JOURNAL OF ALGORITHMS **28,** 272–289 (1998)
ARTICLE NO. AL980936

Minimum Fill-in on Circle and Circular-Arc Graphs

T. Kloks*

Faculty of Applied Mathematics, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands

D. Kratsch†

Fakultät für Mathematik und Informatik, Friedrich-Schiller-Universität, 07740 Jena, Germany

and

C. K. Wong‡

Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong

Received November 11, 1996; revised March 10, 1998

We present two algorithms solving the minimum fill-in problem on circle graphs and on circular-arc graphs in time $O(n^3)$.

1. INTRODUCTION

The MINIMUM FILL-IN problem is a well-known and often-studied problem. It stems from the optimal performance of Gaussian elimination on sparse matrices.

Gaussian elimination of sparse matrices is one of the major problems in computational linear algebra. The problem is to find a pivoting such that the number of nonzeros created during the elimination process is mini-

* This research was done while the author was with the Department of Computer Science and Engineering of the Chinese University of Hong Kong, Shatin, Hong Kong, as a research fellow. E-mail: ton@win.tue.nl.

† E-mail: kratsch@minet.uni-jena.de.

‡ On leave from IBM T.J. Watson Research Center, P.O. Box 218, Yorktown Heights, NY 10598.

0196-6774/98 $25.00

mized. Due to the lack of efficient algorithms for finding an optimal solution, in practice one usually has to work with certain heuristics for "approximating" a minimum fill-in (such as the minimum degree and nested dissection heuristics [1]).

The knowledge on the algorithmic complexity of the MINIMUM FILL-IN problem when restricted to special graph classes is relatively small compared to that of other problems, such as, for example, INDEPENDENT SET, CLIQUE, DOMINATING SET, TREEWIDTH, and PATHWIDTH.

The MINIMUM FILL-IN problem "Given a graph G and a positive integer k, decide whether there is a minimum fill-in of G with at most k edges" remains NP-complete on cobipartite graphs [24] and on bipartite graphs [22]. The only known graph classes for which the minimum fill-in can be computed by a polynomial time algorithm were for almost 10 years the relatively small classes of cographs [7] and bipartite permutation graphs [20]. Now polynomial time algorithms for chordal bipartite graphs [6], multitolerance graphs [17], and d-trapezoid graphs [3] are known.

In a sense, the MINIMUM FILL-IN problem has many similarities with the TREEWIDTH problem. Both problems ask for a chordal embedding of the graph. In the TREEWIDTH problem, one wishes to keep the maximum clique size as small as possible. An $O(n^3)$ algorithm computing the treewidth of circle graphs is given in [12, 13] and an $O(n^3)$ algorithm computing the treewidth of circular-arc graphs is given in [21].

We are not aware of any graph class for which the two problems TREEWIDTH and MINIMUM FILL-IN have different algorithmic complexity, although the solution for the two problems can be far apart [2] (see also [12]).

Both graph classes that we consider, circle graphs as well as circular-arc graphs, are defined as intersection graphs of geometrical objects of a circle. Circle graphs are the intersection graphs of a collection of chords of a circle and circular-arc graphs are the intersection graphs of a collection of arcs of a circle. This leads to a number of similarities between the two graph classes which allows somewhat similar algorithms for two different classes. To emphasize the similarities, we shall present all the theoretical background results for our algorithms "in parallel."

We present two simple algorithms to compute the minimum fill-in of circle and circular arc-graphs. Both algorithms compute a minimum weight triangulation of a certain convex polygon and have overall running time $O(n^3)$.

This shows that circle graphs and circular-arc graphs are two more graph classes with polynomial time algorithms solving the MINIMUM FILL-IN problem. Thus the algorithmic complexity of the problems TREEWIDTH and MINIMUM FILL-IN is the same on both classes. In this way we erase two graph classes from the relatively small list of candidates for graph classes with different algorithmic complexity of TREEWIDTH and MINIMUM FILL-IN.

Since the class of permutation graphs is properly contained in the class of circle graphs, our circle graph algorithm extends a result of [3], where an $O(n^2)$ time algorithm is given that computes the minimum fill-in of trapezoid graphs, and thus of permutation graphs.

2. PRELIMINARIES

Let $G = (V, E)$ be a graph. We denote the number of vertices of G by n. For a set $W \subseteq V$, we denote by $G[W]$ the subgraph of G induced by W.

2.1. *Preliminaries on Triangulations*

We start by considering triangulations and minimal separators.

DEFINITION 1. A graph is *chordal* if it does not contain a chordless cycle of length greater than 3.

DEFINITION 2. A *triangulation* of a graph G is a chordal graph H with the same vertex set as G, such that G is a subgraph of H. A triangulation H of a graph G is called a *minimal triangulation* of G, if no proper subgraph of H is a triangulation of G.

The following theorem has been shown in [18].

THEOREM 3. *Let H be a triangulation of a graph G. Then H is a minimal triangulation of G if and only if each edge $e \in E(H) \setminus E(G)$ is the unique chord of a cycle of length 4 in H.*

DEFINITION 4. Let $G = (V, E)$ be a graph and a, b two nonadjacent vertices of G. The set $S \subseteq V$ is an *a, b-separator* if the removal of S separates a and b into distinct connected components. If no proper subset of S is an a, b-separator then S is a *minimal a, b-separator*. A *minimal separator* is a set of vertices S for which there exist nonadjacent vertices a and b such that S is a minimal a, b-separator.

For a proof of the following lemma (which is well known), see, e.g., [10].

LEMMA 5. *Let S be a minimal a, b-separator of the graph $G = (V, E)$ and let C_a and C_b be the connected components of $G[V \setminus S]$ containing a and b, respectively. Then every vertex of S has at least one neighbor in C_a and at least one neighbor in C_b.*

We denote by $\Delta(H)$ the set of all minimal separators of a graph H. In [14] the following characterization of minimal triangulations is given.

THEOREM 6. *A triangulation H of a graph G is a minimal triangulation of G if and only if the following three conditions are satisfied*:

1. *If a and b are nonadjacent vertices of H, then every minimal a, b-separator of H is also a minimal a, b-separator of G.*

2. *If S is a minimal separator of H and C a connected component of $H[V \setminus S]$, then the vertex set of C also induces a connected component in $G[V \setminus S]$.*

3. *$H = G_{\Delta(H)}$, where $G_{\Delta(H)}$ is the graph obtained from G by adding edges between every pair of vertices contained in the same set S, for any $S \in \Delta(H)$.*

Now it is convenient to define the MINIMUM FILL-IN problem as follows.

DEFINITION 7. The MINIMUM FILL-IN problem is the problem of finding a triangulation H of the given graph $G = (V, E)$ with the least possible number of edges. The *minimum fill-in* of the graph G, denoted by mfi(G), is the minimum number of edges which have to be added to make G chordal.

In other words, solving the MINIMUM FILL-IN problem is equivalent to finding a (minimal) triangulation H of the input graph G with smallest possible number of edges. Then any perfect elimination ordering of H is a minimum elimination ordering of G (see [18]).

2.2. *Preliminaries on Circle and Circular-Arc Graphs*

We give the necessary background material concerning the two graph classes studied in this paper. For more information on circle graphs, circular-arc graphs, and related classes of graphs, we refer to [10].

DEFINITION 8. A *circle graph* $G = (V, E)$ is a graph for which one can associate with each vertex $v \in V$ a chord of a circle $\mathscr{C}$ such that two vertices of G are adjacent if and only if the corresponding chords have a nonempty intersection. The set of chords and the circle $\mathscr{C}$ are said to be a *circle model* $\mathscr{D}(G)$.

Without loss of generality we assume that no two chords of the circle model share an end point. We also assume that a circle model of the input graph is given, since there is an $O(n^2)$ time recognition algorithm for circle graphs, that also computes a circle model of the input graph, if it is a circle graph [19]. (For other recognition algorithms, see [4, 5, 9, 16].)

DEFINITION 9. A *circular-arc graph* is a graph $G = (V, E)$ for which one can associate with each vertex $v \in V$ an arc on a circle $\mathscr{C}$ such that two vertices of G are adjacent if and only if the corresponding arcs have a nonempty intersection. The set of arcs and the circle $\mathscr{C}$ are said to be a *circular-arc model* $\mathscr{D}(G)$.

Without loss of generality we assume that no two arcs of a circular-arc model share an end point and that no arc covers the whole circle. Hence all end points in a circular-arc model are pairwise different. If the circle $\mathscr{C}$ contains a point that is not contained in any arc, then the graph is an interval graph and its minimum fill-in is 0. Thus, we assume that every point of $\mathscr{C}$ is contained in at least one arc.

We also assume that a circular-arc model of the given graph is available. Note that, if no circular-arc model is part of the input, then we can compute one by an $O(n^2)$ time algorithm [8]. This is a recognition algorithm for circular-arc graphs that also computes a circular-arc model of the given graph, if it is a circular-arc graph. (For other recognition algorithms, see [11, 23].)

In the following sections, we show that any minimal triangulation of a circle graph and a circular-arc graph can be represented in terms of a (planar) triangulation of a well-defined convex polygon. This is the property exploited by our algorithms. This property enables the design of simple algorithms computing the minimum fill-in.

3. SCANLINES

We show how to represent the minimal separators of circle graphs and circular-arc graphs by means of scanlines.

3.1. *Circle Graphs*

Let $G = (V, E)$ be a circle graph with circle model $\mathscr{D}(G)$.

DEFINITION 10. Place a new point on the circle $\mathscr{C}$ between every two consecutive end points of chords. These new points are called *scanpoints* and the set of the $2n$ scanpoints of $\mathscr{D}(G)$ is denoted by Z.

DEFINITION 11. A *scanline* of $\mathscr{D}(G)$ is a chord of the circle $\mathscr{C}$, connecting two scanpoints.

Consequently, there are $\binom{2n}{2}$ different scanlines in $\mathscr{D}(G)$.

DEFINITION 12. Two scanlines *cross* if they have a nonempty intersection but no scanpoint in common.

DEFINITION 13. Let c_1 and c_2 be two chords of $\mathscr{C}$ with empty intersection. A scanline s is *between* c_1 and c_2 if every path from an end point of c_1 to an end point of c_2 along $\mathscr{C}$ passes through a scanpoint of s.

For any scanline s of $\mathscr{D}(G)$, we denote by $S(s)$ the set of all vertices v of G, for which the corresponding chord intersects s.

For the following theorem and corollary, see [12, 13, 15].

THEOREM 14. *Let a and b be nonadjacent vertices of the circle graph $G = (V, E)$. For every minimal a, b-separator S of G, there exists a scanline s of $\mathscr{D}(G)$ between the chords of a and b such that $S = S(s)$.*

COROLLARY 15. *A circle graph on n vertices has $O(n^2)$ minimal separators.*

3.2. *Circular-Arc Graphs*

Let $G = (V, E)$ be a circular-arc graph with circular-arc model $\mathscr{D}(G)$. For any point p of the circle $\mathscr{C}$ of $\mathscr{D}(G)$, we denote by $S(p)$ the set of all vertices v, for which the corresponding arc contains the point p.

DEFINITION 16. Place new points on the circle $\mathscr{C}$ as follows. Consider two consecutive end points x and y of arcs and let p be a point on the circle between x and y. If $|S(p)| < \min(|S(x)|, |S(y)|)$ then we call p a *scanpoint*. The set of scanpoints of $\mathscr{D}(G)$ is denoted by Z^*.

Also in this case we call a chord of the circle connecting two scanpoints a *scanline*.

DEFINITION 17. Let a and b be two nonadjacent vertices of G. A scanline s is *between* the arcs of a and b if every path from an end point of the arc of a to an end point of the arc of b along the circle $\mathscr{C}$ passes through a scanpoint of s.

A vertex is called *simplicial* if its neighborhood is a clique.

LEMMA 18. *If an arc in a circular-arc model $\mathscr{D}(G)$ does not contain any scanpoint then its corresponding vertex is simplicial.*

Proof. The definition of Z^* implies that between any two nonintersecting arcs a and b in $\mathscr{D}(G)$ there is a scanpoint u and a scanpoint z such that the scanline connecting u and z is between a and b. Consequently, if a vertex has nonadjacent neighbors than there is a scanpoint contained in the arc corresponding to that vertex. ∎

For any scanline s of $\mathscr{D}(G)$, we denote by $S(s)$ the set of all vertices v of G, for which the corresponding arc contains at least one scanpoint of s.

Proofs of the following theorem and corollary were given in [12, 13, 15].

THEOREM 19. *Let a and b be nonadjacent vertices of a circular-arc graph $G = (V, E)$. For every minimal a, b-separator S of G, there exists a scanline s between the arcs of a and b such that $S = S(s)$.*

Proof. Let C_a and C_b be the components of $G[V \setminus S]$ containing a and b, respectively. Then there is a "generalized arc" corresponding to C_a and C_b that arises as the union of the arcs of all vertices in C_a and C_b, respectively.

Clearly, these generalized arcs have an empty intersection. Moreover, when going from one generalized arc to the other in clockwise direction along $\mathscr{C}$, there is at least one scanpoint in the part of the circle not contained in a generalized arc.

We construct the scanline s as follows. Starting from the generalized arc of C_a in clockwise direction, let x be the first scanpoint not contained in the generalized arc. Similarly, let y be the first scanpoint outside the generalized arc of C_b, when going from C_b in clockwise direction. Then s is the scanline connecting the scanpoints x and y. By construction, s is between the arcs of a and b.

Note that x and y cannot be contained in any arc corresponding to a vertex of $V \setminus S$. Hence $S(s) \subseteq S$. On the other hand, every arc corresponding to a vertex of S intersects the generalized arc corresponding to C_a and also that corresponding to C_b by Lemma 5. Thus $S(s) = S$. ∎

COROLLARY 20. *A circular-arc graph on n vertices has $O(n^2)$ minimal separators.*

Characterizations of the minimal separators in terms of scanlines as in Theorems 14 and 19 and their consequences for the number of minimal separators as in Corollaries 15 and 20 have been presented for various classes of intersection graphs (see, e.g., [3, 12, 15]).

4. REALIZERS AND TRIANGULATIONS

We introduce two similar types of convex polygons and show how certain planar triangulations of these polygons and the triangulations of the corresponding graph relate to each other.

4.1. *Circle Graphs*

Let $G = (V, E)$ be a circle graph. Consider a circle model $\mathscr{D}(G)$ with the set Z of scanpoints.

DEFINITION 21. Let $Y \subseteq Z$ and $|Y| \geq 3$. We denote by $\mathscr{P}(Y)$ the convex polygon with vertex set Y. The *candidate component* $G(Y)$ is the subgraph of G induced by the set of vertices corresponding to chords of $\mathscr{D}(G)$ that have a nonempty intersection with the interior region of $\mathscr{P}(Y)$.

Hence the edges of the polygon $\mathscr{P}(Y)$ are scanlines. Notice that $G(Z) = G$.

DEFINITION 22. Let $Y \subseteq Z$ and $|Y| \geq 3$. For each scanline s that is an edge of the polygon $\mathscr{P}(Y)$, add an edge between any pair of nonadjacent vertices of the candidate component $G(Y)$ of which the two corresponding chords intersect the scanline s. The graph obtained in this way is called the *realizer* $R(Y)$ of the candidate component $G(Y)$.

Notice that $R(Z) = G$. Furthermore, if $|Y| = 3$ then $\mathscr{P}(Y)$ is a triangle and each chord corresponding to a vertex of $G(Y)$ intersects exactly two edges of $\mathscr{P}(Y)$. Thus $R(Y)$ is a clique.

LEMMA 23. *Let $Y \subseteq Z$ and $|Y| \geq 3$. Then the realizer $R(Y)$ is a circle graph.*

DEFINITION 24. Let $G(Y)$ be a candidate component with realizer $R(Y)$. A scanline s of $\mathscr{D}(R(Y))$ is *Y-nice* if the scanpoints of s are elements of Y.

LEMMA 25. *Let $Y \subseteq Z$ and $|Y| \geq 3$ and let S be a minimal a, b-separator in the realizer $R(Y)$. Then there is a Y-nice scanline s such that $S = S(s)$ in $\mathscr{D}(R(Y))$.*

For the proofs of Lemmas 23 and Lemma 25, we refer the reader to [12, 13].

DEFINITION 26. Let $\mathscr{P}$ be a convex polygon with m vertices. A (*planar*) *triangulation* of $\mathscr{P}$ is a set of $m - 3$ noncrossing diagonals in $\mathscr{P}$ that divide the interior of $\mathscr{P}$ into $m - 2$ triangles.

DEFINITION 27. Let $Y \subseteq Z$ and $|Y| \geq 3$. Let T be a triangulation of $\mathscr{P}(Y)$. Then $H(T)$ is defined as the graph with the same vertex set as $G(Y)$ and vertices u and v of $H(T)$ are adjacent if there exists a triangle Q in T such that the two chords corresponding to u and v both intersect Q.

Note that $R(Y)$ is a spanning subgraph of $H(T)$.

LEMMA 28. *The graph $H(T)$ is chordal and therefore a triangulation of $R(Y)$.*

Proof. Let $y \in Y$ be a vertex of $\mathscr{P}(Y)$ that is not incident with a diagonal of the triangulation T. Hence y is incident with exactly one triangle Q of T. If there is a chord intersecting Q but no other triangle of T, then the vertex x corresponding to this chord is a simplicial vertex of $H(T)$. Remove x from $H(T)$ and the chord from the circle model.

If there is no chord left that intersects Q but no other triangle of T, then remove y from Y. Notice that this removal does not change $H(T)$ because in this case any chord intersecting Q also intersects the neighboring triangle of Q in T. In this way, we obtain a perfect elimination ordering of the graph $H(T)$. Thus $H(T)$ is chordal. ∎

We give a representation theorem for all minimal triangulizations of a circle graph in terms of planar triangulations of the polygon $\mathscr{P}(Z)$ (see also [12, 13]).

THEOREM 29. *Let $G = (V, E)$ be a circle graph. Let $\mathscr{D}(G)$ be a circle model of G and Z its set of scanpoints. Then for every minimal triangulation H of G there is a (planar) triangulation T of the polygon $\mathscr{P}(Z)$ such that $H = H(T)$.*

Proof. We claim that for any set $Y \subseteq Z$ with $|Y| \geq 3$ and any minimal triangulation H of the realizer $R(Y)$, there is a triangulation T of $\mathscr{P}(Y)$ such that $H = H(T)$. Note that the claim immediately implies the theorem, since $G = R(Z)$.

First let H be a complete graph. Then $\Delta(H) = \emptyset$. Thus Theorem 6 implies that $R(Y)$ is also complete. Hence $H = H(T)$ for any triangulation T of $\mathscr{P}(Y)$.

Now let H be a noncomplete graph. We shall prove the claim by induction on the number of vertices in Y. Let S be a minimal a, b-separator of H. Since H is a minimal triangulation, S is also a minimal a, b-separator of $R(Y)$ by Theorem 6. Then there exists a Y-nice scanline s such that $S = S(s)$ in $\mathscr{D}(R(Y))$ by Lemma 25. Clearly, s divides the polygon $\mathscr{P}(Y)$ into two polygons $\mathscr{P}(Y_1)$ and $\mathscr{P}(Y_2)$. Moreover, $|Y_1| < |Y|$ and $|Y_2| < |Y|$, since s is between the chords of a and b.

Consider the corresponding realizers $R(Y_1)$ and $R(Y_2)$. The subgraphs of H induced by the vertices of $R(Y_1)$ and $R(Y_2)$ are minimal triangulations of $R(Y_1)$ and $R(Y_2)$. Now the claim follows by induction. ∎

The consequence of Theorem 29 and Lemma 28 is a minimum fill-in algorithm for circle graphs, that essentially computes a minimum weight triangulation of the polygon $\mathscr{P}(Z)$.

4.2. *Circular-Arc Graphs*

Let $G = (V, E)$ be a circular-arc graph with circular-arc model $\mathscr{D}(G)$. Consider the set Z^* of scanpoints of $\mathscr{D}(G)$.

Contrary to circle graphs, we only have $1 \leq |Z^*| \leq 2n$. (By our assumption, no arc in $\mathscr{D}(G)$ covers the whole circle. Hence $|Z^*| = 0$ is impossible.) If $|Z^*| = 1$, then G has no minimal separator by Theorem 19. Hence G is complete. If $|Z^*| = 2$ and G is not complete then Theorem 19 implies that $S(s)$ is the unique minimal separator of G, where the scaline s connects the two scanpoints of Z^*. Moreover, $S(s)$ is a clique, since G has no other minimal separator. Furthermore, any vertex of $V \setminus S(s)$ is simplicial by Lemma 18. Hence, removing all simplicial vertices, we obtain the complete graph $G[S(s)]$. Thus G has a perfect elimination ordering and is chordal. (In fact, one can show that G is an interval graph having at most two maximal cliques.) Thus in both cases $\mathrm{mfi}(G) = 0$. Consequently, we may assume $|Z^*| \geq 3$.

Let $\mathscr{P}(Z^*)$ be the convex polygon with vertex set Z^*. Hence the edges of $\mathscr{P}(Z^*)$ are scanlines of $\mathscr{D}(G)$.

DEFINITION 30. Let T be a (planar) triangulation of the polygon $\mathscr{P}(Z^*)$. Then the graph $H(T)$ is defined as follows. The vertex set of $H(T)$ is the same as the vertex set of G. Two vertices u and v of $H(T)$ are adjacent if either they are adjacent in G or there exists a diagonal d in T such that the arcs of u and v both contain a scanpoint of d.

Note that if the arcs of vertices u and v of G both contain a scanpoint of an edge of $\mathscr{P}(Z^*)$ then u and v are adjacent in G, since there is no minimal u, v-separator in G by Theorem 19.

LEMMA 31. *$H(T)$ is chordal and therefore a triangulation of G.*

Proof. Suppose $H(T)$ is not chordal and let $\mathscr{Z}$ be a chordless cycle in $H(T)$ of length greater than 3. Thus there are two vertices x and y of $\mathscr{Z}$ which are nonadjacent in $H(T)$. Clearly, the arc a of x and the arc b of y have empty intersection. By Theorem 19 there are two scanpoints p and q, such that the scanline s connecting p and q is between a and b and $S(s)$ is a minimal x, y-separator in G.

Assume s is not an edge of $\mathscr{P}(Z^*)$. Then there is a scanpoint p' contained in the arc a and a scanpoint q' contained in the arc b. Since T is a triangulation of $\mathscr{P}(Z^*)$ and there is a diagonal of T in the convex polygon $\mathscr{P}(\{p, p', q, q'\})$. By the choice of a and b this cannot be (p', q'), thus $s = (p, q)$ is a diagonal of T.

Let C_x and C_y be the components of $G[V \setminus S(s)]$ containing x respectively y. By the definition of $H(T)$, there is no edge between a vertex of C_x and a vertex of C_y in $H(T)$. Consequently, the cycle $\mathscr{Z}$ contains two nonconsecutive vertices u and z that both belong to $S(s)$. If s is an edge of $\mathscr{P}(Z^*)$ then $S(s)$ is a clique of G as we noticed above. Thus $S(s)$ is also a clique of $H(T)$, a contradiction. Otherwise consider the diagonal $s = (p, q)$. The arcs of u and z either both contain p respectively q or one contains p and the other q. Therefore u and z are adjacent in $H(T)$, which is a contradiction. ∎

Now we give a representation theorem for the minimal triangulations of circular-arc graphs in terms of (planar) triangulations of the convex polygon $\mathscr{P}(Z^*)$.

THEOREM 32. *Let $G = (V, E)$ be a circular-arc graph. Let $\mathscr{D}(G)$ be a circular-arc model of G and Z^* its set of scanpoints. Then for every minimal triangulation H of G there is a triangulation T of the polygon $\mathscr{P}(Z^*)$ such that $H = H(T)$.*

Proof. The proof is by induction on the number of vertices of the graph.

Suppose H is a complete graph. Then G is also complete. Hence $H(T)$ is a complete graph for any triangulation T of $\mathscr{P}(Z^*)$.

Now assume that H has two nonadjacent vertices a and b. Let S be a minimal a, b-separator in H. Since H is chordal, S is a clique in H. By Theorem 6, S is a minimal a, b-separator in G.

By Theorem 19, there exists a scanline s of $\mathscr{D}(G)$ such that $S(s) = S$ and s is between the arcs of a and b. Let α and β be the scanpoints of s. The removal of α and β from the circle $\mathscr{C}$ creates two "halves," which we call *a-half* if it contains the arc of a and *b-half* if it contains the arc of b.

By Theorem 6, the vertex set of every connected component of $H[V \setminus S]$ induces a connected component of $G[V \setminus S]$. Let $V_1 \subseteq V$ be the union of S and the vertex sets of all components of $G[V \setminus S]$ for which all arcs are in the a-half. Analogously, let $V_2 \subseteq V$ be the union of S and the vertex sets of all components for which all arcs are in the b-half. Clearly, $V_1 \cap V_2 = \emptyset$, $|V_1| < |V|$, and $|V_2| < |V|$.

For $i \in \{1, 2\}$, let $R(V_i)$ be the graph obtained from $G[V_i]$ by making S a clique. Since H is a minimal triangulation of G, S is a clique in $R(V_i)$ as well as in H, any component of $G[V \setminus S]$ is either contained in $G[V_1]$ or in $G[V_2]$, and the vertex set of each component of $H[V \setminus S]$ is also the vertex set of a component of $G[V \setminus S]$, we obtain that $H[V_i]$ is a minimal triangulation of $R(V_i)$, which can be seen best by using Theorem 3.

Our aim is to construct a circular-arc model of $R(V_1)$ and one of $R(V_2)$. Let $Z_1^* \subseteq Z^*$ (respectively $Z_2^* \subseteq Z^*$) be the set consisting of α, β and all scanpoints in the a-half (respectively b-half). Thus $\min(|Z_1^*|, |Z_2^*|) \geq 2$.

Case 1. $\min(|Z_1^*|, |Z_2^*|) \geq 3$.

Consider the a-half. Clearly, $H[V_1]$ is chordal and thus a triangulation of $G[V_1]$. We obtain a circular-arc model of $G[V_1]$ by removing the arcs of all vertices of $V_2 \setminus S$ from $\mathscr{D}(G)$. Then replace any arc of a vertex v of S containing only one of α and β by a new arc that has the original end point in the a-half and a new end point in the b-half, such that all these new arcs contain a fixed point of the b-half, say one close to α.

First, assume that there is one arc of a vertex in S containing α but not β, and one arc containing β but not α. Then the new circular-arc model can be constructed in such a way that the set of scanpoints in the new model is exactly Z_1^*. Clearly, the new model is a circular-arc model of the graph $R(V_1)$ since S is a clique and adjacencies between a vertex of S and a vertex of V_1 remain unchanged. As we have seen above, $H[V_1]$ is a minimal triangulation of $R(V_1)$. Thus, by induction, we obtain that there is a triangulation R_1 of $\mathscr{P}(Z_1^*)$ such that $H[V_1] = H(T_1)$.

In the remaining case all arcs of vertices in S contain one particular end point of s, say α. Hence S is a clique in G and $R(V_1) = G$. Thus removing all arcs of vertices of $V_2 \setminus S$ creates a circular-arc model of $R(V_1)$. However, the set of scanpoints Z_1' in the new model is either $Z_1^* \setminus \{\alpha, \beta\}$ or $Z_1^* \setminus \{\beta\}$. Similar to the case considered first, using induction, we obtain that there is a triangulation T_1' of $\mathscr{P}(Z_1')$ such that $H[V_1] = H(T_1')$. (Possibly $|Z_1'| < 3$. Then take just the "degenerate triangulation" of the "polygon" on one or two vertices, which simply means no diagonals.) Let α' and β' be the scanpoints in the a-half that are closest to α and β, respectively. Then we obtain a triangulation T_1 of the polygon $\mathscr{P}(Z_1^*)$ in $\mathscr{D}(G)$ by adding to the triangulation of T_1' of $\mathscr{P}(Z_1')$ the scanlines s and a scanline between α' and α if $Z_1' = Z_1^* \setminus \{\beta\}$, and additionally a diagonal between β and α' and a scanline between β' and β if $Z_1' = Z_1^* \setminus \{\alpha, \beta\}$. Then $H[V_1] = H(T_1')$, the fact that S is a clique in G, $R(V_1)$, and H, and the construction of T_1 imply $H[V_1] = H(T_1)$.

Analogously, there is a triangulation T_2 of $\mathscr{P}(Z_2^*)$ such that $H[V_2] = H(T_2)$.

Finally, take the scanline s and all the diagonals of T_1 and T_2. This gives a triangulation T of $\mathscr{P}(Z^*)$. Furthermore, $H = H(T)$ since any edge of H that does not have both end vertices in S is either an edge of $H[V_1]$ or $H[V_2]$, and thus represented by some diagonal of T.

Case 2. $\min(|Z_1^*|, |Z_2^*|) = 2$.

Without loss of generality assume $|Z_2^*| = 2$. Then consider $H[V_1]$ and the set $Z_1^* = Z^*$. There is no scanpoint in the b-half, thus S is a clique in G and $R(V_1) = G$. Hence, removing all arcs of vertices in $V_2 \setminus S$ from $\mathscr{D}(G)$, we obtain a circular-arc model of $R(V_1)$. $H[V_1]$ has fewer vertices than H and it is a minimal triangulation of $R(V_1)$. Thus, by induction, there is a triangulation T_1 of $\mathscr{P}(Z_1^*)$ such that $H[V_1] = H(T_1)$. By Lemma 18, all vertices of $V_2 \setminus S$ are simplicial in G. No minimal triangulation of a graph G adds an edge for which an incident vertex is simplicial. (This follows easily from Theorem 3 as well as from Theorem 6 and the fact that no minimal separator from Theorem 3 as well as from Theorem 6 and the fact that no minimal separator of a graph contains a simplicial vertex.) Consequently, $H(T) = H$, where T is the triangulation T_1 of $\mathscr{P}(Z_1^*)$ taken as a triangulation of $\mathscr{P}(Z^*)$ with respect to G. ∎

5. END TRIANGLES

The concept of an end triangle is important for obtaining efficient algorithms that compute the minimum fill-in on circle graphs and circular-arc graphs.

5.1. *Circle Graphs*

Let $G = (V, E)$ be a circle graph with circle model $\mathscr{D}(G)$ and let $\mathscr{P}(Z)$ be the convex polygon with vertex set Z.

DEFINITION 33. Let a and b be nonadjacent vertices of G. A triangle Q in $\mathscr{P}(Z)$, that is, all vertices of Q are in Z, is an *end triangle for* $\{a, b\}$, if the chords of a and b both intersect Q, but there is only one edge of Q that is crossed by both chords.

LEMMA 34. *Let T be any triangulation of $\mathscr{P}(Z)$. If the chords of two nonadjacent vertices a and b both intersect some triangle of T, then there are exactly two end triangles for $\{a, b\}$ in T.*

Proof. Suppose the chords of a and b both intersect a triangle Q of T. Then the chords of a and b both cross at least one common edge, say r, of Q. Hence the chords of a and b also intersect a neighboring triangle Q' which shares the edge r with Q. In this way we find a path of triangles which must end with an end triangle for $\{a, b\}$.

Notice that the set of triangles having nonempty intersection with the chords of a and b is exactly the path of triangles between the two end triangles for $\{a, b\}$. This shows that there are exactly two end triangles for $\{a, b\}$. ∎

DEFINITION 35. Let Q be a triangle of $\mathscr{P}(Z)$. Then $w(Q)$, the *weight* of Q, is the number of unordered pairs $\{a, b\}$ of noncrossing chords for which Q is an end triangle. The *weight $w(T)$ of a triangulation T* is the sum of the weights of all the triangles in T.

COROLLARY 36. *Let $G = (V, E)$ be a circle graph. Then*

$$\mathrm{mfi}(G) = \tfrac{1}{2}\min\{w(T) | T \textit{ triangulation of } \mathscr{P}(Z)\}.$$

Proof. Let T be a triangulation of $\mathscr{P}(Z)$. Consider $H(T)$. The weight of T is twice the number of edges of $H(T)$ minus the number of edges in G, since Lemma 34 implies that every edge in $H(T)$ that is not an edge in G is counted exactly twice, namely once for each end triangle. Thus the minimum weight of a triangulation T of $\mathscr{P}(Z)$ is exactly twice the minimum fill-in of G. ∎

5.2. *Circular-Arc Graphs*

Let $G = (V, E)$ be a circular-arc graph with circular-arc model $\mathscr{D}(G)$ and let $\mathscr{P}(Z^*)$ be the convex polygon with vertex set Z^*.

DEFINITION 37. Let a and b be two nonadjacent vertices of G. A triangle Q of $\mathscr{P}(Z^*)$ is an *end triangle for* $\{a, b\}$ if the arcs of a and b each contain exactly one vertex of Q.

LEMMA 38. *Let T be any triangulation of $\mathscr{P}(Z^*)$. If there is a diagonal s of T such that the arcs of the two adjacent vertices a and b both contain one scanpoint of s, then there are exactly two end triangles for $\{a, b\}$.*

Proof. Consider the diagonal s. Let us say that it is horizontal. We claim that there is exactly one end triangle above s and exactly one end triangle below s.

Since s is a diagonal, there are two triangles incident with s, one below and one above s. Consider the triangle Q above s. Let s have end points α and β and let γ be the third vertex of Q.

First, assume γ is not contained in the arcs a or b. Then Q is an end triangle. All triangles above Q are above one of the scanlines between α and γ or β and γ. This shows that there can be no other end triangle above Q.

Now, without loss of generality, assume γ is contained in b. Let s' be the scanline between α and γ. By Theorem 19, there is a scanline between a and b. This scanline clearly intersects s'. It follows that s' must be a diagonal in T rather than an edge of $\mathscr{P}(Z^*)$. Now the claim follows by induction. ∎

DEFINITION 39. Let Q be a triangle of $\mathscr{P}(Z^*)$. Then $w(Q)$, the *weight of Q*, is the number of unordered pairs $\{a, b\}$ for which Q is an end triangle.

COROLLARY 40. *Let $G = (V, E)$ be a circular-arc graph. Then*

$$\text{mfi}(F) = \tfrac{1}{2}\min\{w(T) \mid T \textit{ triangulation of } \mathscr{P}(Z^*)\}.$$

Proof. Let T be a triangulation of $\mathscr{P}(Z^*)$. Consider $H(T)$. Lemma 38 implies that the weight of T is twice the number of edges of $H(T)$ minus the number of edges in G, since every edge in $H(T)$ that is not an edge in G is counted exactly twice, namely once for each end triangle. Consequently, the minimum weight of a triangulation T of $\mathscr{P}(Z^*)$ is exactly twice the minimum fill-in of G. ∎

6. THE ALGORITHMS COMPUTING THE MINIMUM FILL-IN

In this section we describe simple polynomial time algorithms to find the minimum fill-in of circle graphs and circular-arc graphs.

Luckily, for both graph classes we were able to define a weight function on the triangles such that the minimum weight of a (planar) triangulation is equal to twice the minimum fill-in of the input graph. Hence both algorithms compute the minimum weight of a triangulation of the polygon

$\mathscr{P}(Z)$ and $\mathscr{P}(Z^*)$, respectively, using a classical dynamic programming algorithm for this problem which has also been applied in [13, 21].

Let $s_1, s_2, \ldots, s_h$ be the scanpoints of the set Z (respectively Z^*) in some clockwise order. Then $h = 2n$ for circle graphs and $h \leq 2n$ for circular-arc graphs. (Note that all indices in this section are to be taken modulo h.)

Let $c(i, j, k)$ be the weight of the triangle with vertices s_i, s_j, s_k. Suppose the weights of all triangles of $\mathscr{P}(Z)$ (respectively $\mathscr{P}(Z^*)$) are given. Then there is an $O(h^3)$ algorithm that computes the minimum weight of a triangulation of $\mathscr{P}(Z)$ (respectively $\mathscr{P}(Z^*)$).

Define $w(i, t)$ as the minimum weight of a triangulation of the polygon with vertices $s_i, s_{i+1}, \ldots, s_{i+t-1}$. Then, for all i, $w(i, 2) = 0$, and, for all $t \in \{3, \ldots, 2n\}$,

$$w(i,t) = \min_{2 \leq j < t} w(i,j) + w(i+j-1, t-j+1) + c(i, i+j-1, i+t-1).$$

By Corollaries 36 and 40, the minimum fill-in of the input graph can be computed in time $O(n^3)$ plus the time for computing the weights of all the $O(n^3)$ triangles of the polygon $\mathscr{P}(Z)$ and $\mathscr{P}(Z^*)$. It is not hard to see that the weights of all triangles can be computed in time $O(n^5)$ for circle and circular-arc graphs. This would give $O(n^5)$ algorithms computing the minimum fill-in. In the remainder we show how to get faster algorithms.

6.1. *Circle Graphs*

The weight of the triangle Q is the number of nonadjacent vertices a and b of G for which the chords of a and b both cross Q, but there is only one edge of Q that is crossed by both chords. The weight $c(p, q, r)$ of all triangles of $\mathscr{P}(Z)$ can be computed as follows.

Let $L(i, k; j)$ (where s_j is not on the part of the circle going clockwise from s_i to s_{i+k-1}) be the number of chords that have exactly one end point in clockwise order between s_i and s_{i+k-1} and that cross the scanline between s_i and s_j. The numbers $L(i, 2; j)$ are easy to determine, since there is a unique chord with an end point on the part of the circle going clockwise from s_i to s_{i+1}. The numbers $L(i, k+1; j)$ an be determined as follows. Check if the chord with an end point on the part of the circle going clockwise from s_{i+k-1} to s_{i+k}, crosses the scanline between s_i and s_j. If it does, then $L(i, k+1; j) = L(i, k; j) + 1$ and if it does not, then $L(i, k+1; j) = L(i, k; j)$.

Let $A(i, k; j)$ (where s_j is not in the part of the circle going clockwise from s_i to s_{i+k-1}) be the number of pairs of nonadjacent vertices a and b such that chords of a and b have one end point on the part of the circle going clockwise from s_i to s_{i+k-1}, the chord of a crosses the scanline between s_i and s_j, and the chord of b crosses the scanline between s_{i+k-1} and s_j. Then $A(i, 2; j) = 0$ for all i and j, since there is only one chord with an end point on the part going clockwise from s_i to s_{i+1}. Then consider the chord with an end point on the part of the circle going clockwise from s_{i+k-1} to s_{i+k}. If this chord crosses the scanline between s_{i+k} and s_j, then $A(i, k+1; j) = A(i, k; j) + L(i, k; j)$. Otherwise $A(i, k + 1; j) = A(i, k; j)$.

Now the weight of the triangle with vertices s_p, s_q, and s_r, where s_p, s_q, s_r is in clockwise order, can be computed as

$$\begin{aligned} c(p, q, r) = {} & A(p, q - p + 1; r) + A(q, r - q + 1; p) \\ & + A(r, p - r + 1; q). \end{aligned}$$

Consequently, the weight of all triangles of $\mathscr{P}(Z)$ can be determined in $O(n^3)$ time.

THEOREM 41. *There is an $O(n^3)$ time algorithm computing the minimum fill-in of a circle graph.*

6.2. *Circular-Arc Graphs*

First, check in linear time (see [10]) whether the graph is a chordal graph. If so, its minimum fill-in is 0. Thus, we may assume that the input graph G is not an interval graph, implying $|Z^*| \geq 3$.

We shall demonstrate that the weights of all triangles of $\mathscr{P}(Z^*)$ can be computed by solving the corresponding problem on a suitable circle model. First, add scanpoints to the original set Z^* such that there is exactly one scanpoint between any two consecutive end points of arcs. Similar to the circle graph terminology, we call the set of all these points Z. Clearly, $Z^* \subseteq Z$. Now transform $\mathscr{D}(G)$ into a circle model $\mathscr{D}'(G)$ by replacing any arc with end points x and y with a chord connecting x and y.

Let a and b be two nonadjacent vertices of G. Then, for every triangle Q of $\mathscr{P}(Z)$, the chords corresponding to a and b in $\mathscr{D}'(G)$ intersect Q such that exactly one edge of Q is crossed by both chords if and only if the arcs of a and b in $\mathscr{D}(G)$ have empty intersection and each contains exactly one vertex of Q.

Consequently, the weight of all triangles of $\mathscr{P}(Z^*)$ can be computed in time $O(n^3)$ by using the algorithm for circle graphs, given in Section 6.1, additionally counting a pair of noncrossing chords in $\mathscr{D}'(G)$ only if the corresponding vertices are nonadjacent in the circular-arc graph G, which can be determined easily in $\mathscr{D}(G)$.

THEOREM 42. *There is an $O(n^3)$ time algorithm computing the minimum fill-in of a circular-arc graph.*

7. CONCLUSIONS

In this paper we described elegant and efficient algorithms for solving the MINIMUM FILL-IN problem on circle graphs and circular-arc graphs, which are based on representation theorems for the minimal triangulations of such graphs. Representation theorems of this type are powerful tools for designing treewidth and minimum fill-in algorithms. Similar theorems for other graph classes could be a step forward in clarifying the relation of TREEWIDTH and MINIMUM FILL-IN. Graph classes for which the algorithmic complexity of the problems TREEWIDTH and MINIMUM FILL-IN might be different, up to our knowledge, are, e.g., circular permutation and weakly triangulated graphs.

Earlier results for permutation graphs were generalized to d-trapezoid graphs in [3]. It would be interesting to see a suitable definition of a higher-dimensional circle graph and circular-arc graph, respectively, to which the algorithms presented in this paper can be generalized.

ACKNOWLEDGMENTS

We are grateful to two anonymous referees for many helpful comments.

REFERENCES

1. A. Agrawal, P. Klein, and R. Ravi, Cutting down on fill-in using nested dissection, *in* "Sparse Matrix Computations: Graph Theory Issues and Algorithms," (A. George, J. Gilbert, and J. W.-H. Liu, Eds.), IMA Volumes in Mathematics and Its Applications, Vol. 56, pp. 31–55, Springer-Verlag, New York, 1993.
2. H. Bodlaender, L. van der Gaag, and T. Kloks, Some remarks on minimum edge and minimum clique triangulations, manuscript.
3. H. Bodlaender, T. Kloks, D. Kratsch, and H. Müller, "Treewidth and Minimum Fill-in on d-Trapezoid Graphs," Technical Report UU-CS-1995-34, Utrecht University, The Netherlands.
4. A. Bouchet, A polynomial algorithm for recognizing circle graphs, *C. R. Acad. Sci. Paris Sér. I Math.* **300** (1985), 569–572.
5. A. Bouchet, Reducing prime graphs and recognizing circle graphs, *Combinatorica* **7** (1987), 243–254.
6. M.-S. Chang, Algorithms for maximum matching and minimum fill-in on chordal bipartite graphs, *in* "Proceedings of ISAAC '96," Lecture Notes in Comput. Sci., Vol. 1178, pp. 146–155, Springer-Verlag, New York, 1996.
7. D. G. Corneil, Y. Perl, and L. K. Stewart, Cographs: Recognition, applications and algorithms, *Congr. Numer.* **43** (1984), 249–258.
8. E. M. Eschen and J. P. Spinrad, An $O(n^2)$ algorithm for circular-arc graph recognition, *in* "Proceedings of SODA '93," pp. 128–137.

9. C. P. Gabor, W. L. Hsu, and K. J. Supowit, Recognizing circle graphs in polynomial time, *J. Assoc. Comput. Mach.* **36** (1989), 435–474.
10. M. C. Golumbic, "Algorithmic Graph Theory and Perfect Graphs," Academic Press, New York, 1980.
11. W.-L. Hsu, $O(mn)$ algorithms for the recognition and isomorphism problems on circular-arc graphs, *SIAM J. Comput.* **24** (1995), 411–439.
12. T. Kloks, "Treewidth–Computations and Approximations," Lecture Notes in Comput. Sci., Vol. 842, Springer-Verlag, New York, 1994.
13. T. Kloks, Treewidth of circle graphs, *Internat. J. Found. Comput. Sci.* **7** (1996), 111–120.
14. T. Kloks, D. Kratsch and H. Müller, Approximating the bandwidth of AT-free graphs, *in* "Proceedings of ESA '95," Lectures Notes in Comput. Sci., Vol. 979, pp. 434–447, Springer-Verlag, New York, 1995.
15. D. Kratsch, "The Structure of Graphs and the Design of Efficient Algorithms," Habilitation (thesis), F.-Schiller-Universität, Jena, 1996.
16. W. Naji, Reconnaissance des graphes de cordes, *Discrete Math.* **54** (1985), 329–337.
17. A. Parra, "Triangulating Multitolerance Graphs," Technical Report 392/1994, Technische Universität Berlin, Germany.
18. D. J. Rose, R. E. Tarjan, and G. S. Lueker, Algorithmic aspects of vertex elimination on graphs, *SIAM J. Comput.* **5** (1976), 266–283.
19. J. Spinrad, Recognition of circle graphs, *J. Algorithms* **16** (1994), 264–282.
20. J. Spinrad, A. Brandstädt, and L. K. Stewart, Bipartite permutation graphs, *Discrete Appl. Math.* **18** (1987), 279–292.
21. R. Sundaram, K. Sher Singh, and C. Pandu Rangan, Treewidth of circular arc graphs, *SIAM J. Discrete Math.* **7** (1994), 647–655.
22. R. E. Tarjan, Decomposition by clique separators, *Discrete Math.* **55** (1985), 221–232.
23. A. Tucker, An efficient test for circular-arc graphs, *SIAM J. Comput.* **9** (1980), 1–24.
24. M. Yannakakis, Computing the minimum fill-in is NP-complete, *SIAM J. Algebraic Discrete Methods* **2** (1981), 77–79.