Computing the K-terminal Reliability of Circle Graphs

Min-Sheng Lin*, Chien-Min Chen
Department of Electrical Engineering
National Taipei University of Technology
Taipei, Taiwan

Abstract

Let $G$ denote a graph and let $K$ be a subset of vertices that are a set of target vertices of $G$. The K-terminal reliability of $G$ is defined as the probability that all target vertices in $K$ are connected, considering the possible failures of non-target vertices of $G$. The problem of computing $K$-terminal reliability is known to be #P-complete for polygon-circle graphs, and can be solved in polynomial-time for t-polygon graphs, which are a subclass of polygon-circle graphs. The class of circle graphs is a subclass of polygon-circle graphs and a superclass of t-polygon graphs. Therefore, the problem of computing $K$-terminal reliability for circle graphs is of particular interest. This paper proves that the problem remains #P-complete even for circle graphs. Additionally, this paper proposes a linear-time algorithm for solving the problem for proper circular-arc graphs, which are a subclass of circle graphs and a superclass of proper interval graphs.

Keywords: K-terminal reliability; Circle graphs; Proper circular-arc graphs; Algorithms; Complexity

1. Introduction

Consider a probabilistic graph $G$ in which the edges are perfectly reliable, but vertices may fail with known probabilities. A subset $K$ of vertices of $G$ is chosen as the set of target vertices of $G$. The $K$-terminal reliability (KTR) of $G$ is defined as the probability that all target vertices in $K$ are connected.

The KTR problem has been proved to be #P-complete for general graphs and to remain so even for chordal graphs [1]. Valiant [2] defined the class of #P problems as those that involve counting access computations for problems in NP, while the class of #P-complete problems includes the hardest

* Corresponding author. E-mail: mslin@ee.ntut.edu.tw
problems in \#P. As is well known, all algorithms for solving these problems have exponential time complexity, so the development of efficient algorithms for solving this class of problems is almost impossible. However, considering only a restricted subclass of \#P-complete problems can reduce this complexity. Many studies have investigated KTR problems for restricted subclasses of intersection graphs.

A graph \(G\) is an intersection graph of a set of objects if an isomorphism exists between the vertices of \(G\) and the set of objects such that two vertices are adjacent in \(G\) if and only if their corresponding objects have a nonempty intersection. Well known special classes of intersection graphs include the following. Interval graphs [3] are the intersection graphs of a set of intervals on a line; proper interval graphs [3] are interval graphs in which no interval properly contains another; circular-arc graphs [3] are the intersection graphs of a set of arcs on a circle; proper circular-arc graphs [3] are circular-arc graphs in which no arc properly contains another; permutation graphs [3] are the intersection graphs of a set of line segments between two parallel lines; trapezoid graphs [4] are the intersection graphs of a set of trapezoids between two parallel lines; chordal graphs [3] are the intersection graphs of a set of subtrees of a tree; circle graphs [5] are the intersection graphs of a set of chords on a circle; polygon-circle graphs [6] are the intersection graphs of a set of convex polygons with corners on a circle; circle-trapezoid graphs [7] are the intersection graphs of a set of circle trapezoids on a circle, and t-polygon graphs [8] are the intersection graphs of a set of chords in a convex t-sided polygon. Figure 1 depicted the inclusions between these graph classes. Some of the inclusions are trivial: for example, proper interval \(\subseteq\) \{interval, proper circular-arc\} \(\subseteq\) circular-arc, interval \(\subseteq\) chordal, permutation \(\subseteq\) t-polygon \(\subseteq\) circle \(\subseteq\) circle-trapezoid \(\subseteq\) polygon-circle, \{interval, permutation\} \(\subseteq\) trapezoid, and \{circle, circular arc, trapezoid\} \(\subseteq\) circle-trapezoid. However, some of them are non-trivial: for example, chordal \(\subseteq\) polygon-circle [9], and proper circular-arc \(\subseteq\) circle graph [3].

For the KTR problem, polynomial-time algorithms have been found for interval graphs [1], permutation graphs [1], trapezoid graphs [10], circular-arc graphs [11], and t-polygon graphs [12]. A linear-time algorithm for solving the KTR problem for proper interval graphs was given in [13]. The KTR problem is \#P-complete for chordal graphs [1] and, therefore, for polygon-circle graphs. The complexity of the KTR problem remains unsolved for circle graphs and circle-trapezoid graphs. This paper reveals that the KTR problem remains \#P-complete for circle graphs and circle-trapezoid graphs but that a further restriction to proper circular-arc graphs admits a linear-time solution.
Fig. 1. Inclusion relations between classes of intersection graphs and results of the analysis of the problem of computing KTR. The * symbol indicates a main contribution of this paper.

2. Hardness of computing KTR of circle graphs

This section reveals that the KTR problem remains \#P-complete for the class of circle graphs. A chord family $C$ is a set of chords in a circle. A graph $G$ is a circle graph if there exist a chord family $C$ and a one-to-one mapping between the vertices of $G$ and the chords in $C$ such that two vertices in $G$ are adjacent if and only if their corresponding chords in $C$ intersect. Such a chord family $C$ is called a circle representation and $G(C)$ denotes the circle graph that is constructed from $C$.

For convenience, this paper will consider chords in $C$ rather than vertices in the corresponding graph $G(C)$. A chord in $C$ is called a target chord if its corresponding vertex in $G(C)$ is a target vertex; otherwise, it is called an non-target chord. For simplicity, if no ambiguity arises, let $K$ denote both the set of target chords in $C$ and the set of target vertices in $G(C)$.

**Theorem 1.** The KTR problem for circle graphs is \#P-complete.
Proof. The KTR problem trivially belongs to \#P. To prove its \#P-hardness, the problem of counting edge covers in a bipartite graph, which is known to be \#P-complete [14], is reduced to the KTR problem for circle graphs.

Given an instance \( B \) of bipartite graphs with bipartition \( U = \{u_1, u_2, \ldots \} \) and \( V = \{v_1, v_2, \ldots \} \), an edge cover of \( B \) is a set of edges \( D \subseteq E(B) \) such that each vertex in \( U \cup V \) is incident to at least one edge of \( D \). In the following steps, the corresponding circle representation \( C \) is constructed from \( B \). First, corresponding to each vertex \( u_i \in U \), chord \( x_i \) is placed on the left half circle such that no two chords \( x_i \) intersect. Similarly, corresponding to each vertex \( v_j \in V \), chord \( y_j \) is placed on the right half circle such that no two chords \( y_j \) intersect. Next, corresponding to each edge \((u_i, v_j) \in E(B)\), chord \( w_{ij} \) is placed on the circle such that chord \( w_{ij} \) intersects both chords \( x_i \) and \( y_j \). Finally, an additional chord \( z \) is placed between the left circle and the right circle and it intersects all chords \( w_{ij} \) but neither \( x_i \) nor \( y_j \). Figure 2 shows an example of the above construction.

Let \( X = \{ x_i \mid u_i \in U \} \), \( Y = \{ y_j \mid v_j \in V \} \), and \( W = \{ w_{ij} \mid (u_i, v_j) \in E(B) \} \). Now, \( C = X \cup Y \cup \{ z \} \cup W \).

Let \( K = X \cup Y \cup \{ z \} \) be the set of target chords and \( W \) be the set of non-target chords in \( C \). A success set \( S \) is defined herein as a subset of \( W \) such that all target chords in \( K \) are connected to each other when all chords in \( S \) work and all chords in \( WS \) fail. Let \( SS(C) \) denote the collection of all success sets in \( C \). Let \( EC(B) \) denote the collection of all edge covers in \( B \). The equality \( |SS(C)| = |EC(B)| \) is established as follows.

Let \( D \in EC(B) \) be an arbitrary edge cover of \( B \) and \( S' \subseteq W \) be the corresponding subset of non-target chords of \( G \), so \( S' = \{ w_{ij} \mid (x_i, y_j) \in D \} \). Since \( D \) covers all vertices in \( U \cup V \), each chord in \( X \cup Y \) intersects at least one chord of \( S' \). Furthermore, \( z \) intersects all chords in \( S' \). Consequently, all target chords in \( X \cup Y \cup \{ z \} \) are connected to each other by the set of non-target chords in \( S' \), and thus \( S' \) is a success set of \( C \). On the other hand, let \( S \subseteq SS(C) \) be an arbitrary success set of \( C \) and \( D' \subseteq E(B) \) be the corresponding subset of edges of \( B \), so \( D' = \{ (u_i, v_j) \mid w_{ij} \in S \} \). Because each target chord in \( X \cup Y \) intersects at least one chord of \( S \), \( D' \) is an edge cover of \( B \). Therefore, an isomorphism exists between the edge covers of \( B \) and the success sets of \( C \). Accordingly, \( |SS(C)| = |EC(B)| \). Then, the correlation between the KTR of \( G(C), R_k(G(C)) \), and the number of edge cover in \( B, |EC(B)| \), is established.
According to the definition of success sets, $R_k(G(C))$ is given by

$$\mathcal{R}_k(G(C)) = \sum_{S \in SS(C)} \left( \prod_{c \in S} (1-q_c) \times \prod_{c \in W \setminus S} q_c \right),$$

where $q_c$ is the failure probability of chord $c$.

Let all $q_c$ for $c \in W$ in the above equation have the same value $\frac{1}{2}$; now,

$$\mathcal{R}_k(G(C)) = \sum_{S \in SS(C)} \left( \left(\frac{1}{2}\right)^{|S|} \times \left(\frac{1}{2}\right)^{|W \setminus S|} \right) = \sum_{S \in SS(C)} \left( \left(\frac{1}{2}\right)^{|W|} \right) = |SS(C)| \times \left(\frac{1}{2}\right)^{|E(B)|}.$$

Since $|SS(C)| = |EC(B)|$, the number of edge cover of the bipartite graph $B$ is expressed as

$$|EC(B)| = \mathcal{R}_k(G(C)) \times 2^{|E(B)|}.$$

Therefore, an efficient algorithm for determining the KTR of the circle graph $G(C)$ yields an efficient algorithm for counting edge covers in the bipartite graph $B$. However, as is well known, the latter counting problem is $\#P$-complete, so the KTR problem must also be $\#P$-complete. □

Since the class of circle graphs is a subclass of circle trapezoids and polygon-circle graphs, the following corollaries immediately follow.

**Corollary 1.** The KTR problem for circle-trapezoid graphs is $\#P$-complete.

**Corollary 2.** [1] The KTR problem for polygon-circle graphs is $\#P$-complete.

Fig. 2. A bipartite graph $B$, a circle representation $C$ constructed from $B$, and the circle graph $G(C)$. 

\begin{itemize}
  \item $\bigcirc$: target vertex
  \item $\bigcirc$: non-target vertex
\end{itemize}
3. Linear-time algorithm for computing KTR of proper circular-arc graphs

This section presents a linear time algorithm for computing the KTR of proper circular-arc graphs, which are a subclass of circle graphs and a superclass of proper interval graphs. First, some necessary terminology associated with proper circular-arc graphs is introduced; it is similar to that associated with circle graphs in Section 2. A circular-arc family $\mathcal{A}$ is a set of arcs on a circle. A graph $G$ is a circular-arc graph if a circular-arc family $\mathcal{A}$ and a one-to-one mapping between the vertices of $G$ and the arcs in $\mathcal{A}$ exist such that two vertices in $G$ are adjacent if and only if their corresponding arcs in $\mathcal{A}$ intersect. Such a circular-arc family $\mathcal{A}$ is called a circular-arc representation for $G$ and $G(\mathcal{A})$ denotes the circular-arc graph constructed from $\mathcal{A}$. If $\mathcal{A}$ can be chosen so that no arc contains another, then $G(\mathcal{A})$ is called a proper circular-arc graph and $\mathcal{A}$ is called a proper circular-arc representation.

As mentioned previously, a polynomial-time algorithm exists for computing the KTR of circular-arc graphs [11]. Since every proper circular-arc graph is a circular-arc graph, the same algorithm can be used to compute the KTR of proper circular-arc graphs. This section presents a more efficient algorithm for computing the KTR of proper circular-arc graphs. Consider a proper circular-arc representation $\mathcal{A}$ with $n$ arcs. For convenience, arcs in $\mathcal{A}$ rather than vertices in $G(\mathcal{A})$ will be considered. An arc in $\mathcal{A}$ is called a target arc if its corresponding vertex in $G(\mathcal{A})$ is a target vertex; otherwise, it is called a non-target arc.

Starting with an arbitrary arc of $\mathcal{A}$ and traversing the circle clockwise, label all arcs of $\mathcal{A}$ from 0 to $n-1$ in the order in which they are encountered. Let $K=\{x_0 < x_1 < \ldots < x_{k-1}\}$ be the set of the given $k$ target arcs of $\mathcal{A}$, and let $A_i$ be the set of arcs that are encountered in a clockwise traversal on the circle from arc $x_i$ to arc $x_{i+1}$. Here arithmetic operations on the index $i$ of $x_i$ are taken modulo $k$. That is, $A_i=\{\text{arc } r | x_i \leq r \leq x_{i+1}\}$ for $0 \leq i \leq k-2$ and $A_{k-1}=\{\text{arc } r | r \geq x_{k-1} \text{ or } r \leq x_0\}$. Let $H_i$ be the subgraph of $G(\mathcal{A})$ that is induced by the vertices corresponding to the arcs of $A_i$. The following proposition follows from the definitions of proper interval graphs.

**Proposition 1.** For $0 \leq i \leq k-1$, $H_i$ is a proper interval graph.
Let $R(H_i)$ be the probability that target arcs $x_i$ and $x_{i+1}$ are connected in $H_i$. According to Proposition 1, $H_i$ is a proper interval graph and thus $R(H_i)$ can be computed using the algorithm in our previous work [13], which proposed a linear-time algorithm for computing the KTR of proper interval graphs. Some of the notation used here differs slightly from that used in that work [13], and is described below.

Consider the proper interval graph $H_i$ with the corresponding set of arcs (intervals) $A_i$ on the circle. For each arc $r$ in $A_i$, define $\alpha(r)$ ($\beta(r)$) as the arc $r'$ in $A_i$ such that $r$ and $r'$ intersect and $r'$ is the first (last) arc that is encountered in a clockwise traversal on the circle from $x_i$ to $x_{i+1}$. Also, define $N^+(r)=\{r+1, r+2, \ldots, \beta(r)\}$. Here arithmetic operations on arc $r$ are taken modulo $n$. Now define the event $F(r)$, for $r\in A_i \backslash \{x_{i+1}\}$, as

$$F(r) = \{\text{there exists no operating path from arc } x_i \text{ to any arc of } N^+(r) \}.$$ 

The following lemma provides a recursive method for computing $\Pr[F(r)]$.

**Lemma 1.** [13] For $r\in A_i \backslash \{x_{i+1}\}$,

$$\Pr[F(r)] = \Pr[F(r-1)] + (1 - \Pr[F(\alpha(r)-1)]) \times (1 - q_r) \times \prod_{r'=r+1}^{\beta(r)} q_{r'}, \quad (1)$$

where the boundary condition $\Pr[F(x_i-1)]=0$ and $q_r$ is the failure probability of arc $r$.

Notably, since $N^+(x_{i+1}-1)$ is the set of arcs that are encountered in a clockwise traversal from $x_{i+1}$ to $\beta(x_{i+1}-1)=x_{i+1}$, it contains a single arc $x_{i+1}$. Therefore, the event $F(x_{i+1}-1)$ is equivalent to the event $\{\text{there exists no operating path from arc } x_i \text{ to arc } x_{i+1} \}$, and it implies that $\Pr[F(x_{i+1}-1)]=1-R(H_i)$.

Let $\mathcal{F}$ denote the collection of all events $F(x_{i+1}-1)$, for $0 \leq i \leq k-1$. Some pair of target arcs in $K$ is easily verified to be disconnected if and only if at least two events in $\mathcal{F}$ occur. Let $f(j, h)$, $1 \leq h \leq j \leq k$, be the probability that at least $h$ of the first $j$ events in $\mathcal{F}$ occur. Therefore, the KTR of the proper circular-arc graph is obtained as $1-f(k, 2)$. For simplicity, let $Q_j = \Pr[F(x_j-1)]$ denote the probability that event $F(x_j-1)$ occurs, for $1 \leq j \leq k$. Since the intersection of $A_i$ and $A_{i'}$, for $0 \leq i \neq i' \leq k-1$, contains only perfect target arcs of $K$, any two events in $\mathcal{F}$ are mutually independent. Accordingly, the following recursive relation for computing $f(j, h)$ is easily verified.
Lemma 2. For $1 \leq j \leq k$ and $h=1$ or 2,

$$f(j, h) = Q_j \times f(j-1, h-1) + (1-Q_j) \times f(j-1, h),$$

under the boundary conditions $f(j, 0)=1$, for $0 \leq j \leq k$, and $f(0, 1)=f(0, 2)=0$.

Based on the above formulations, the formal algorithm for computing the KTR of a proper circular-arc graph is presented as follows.

Algorithm 1. Compute the KTR of a proper circular-arc graph

Input: proper circular-arc representation $A$ of $n$ arcs labeled clockwise from 0 to $n-1$,

- closed neighborhood structure, $N[r]$, for each arc $r$ of $A$, and

- set of target arcs $K=\{x_0 < x_1 \ldots < x_{k-1}\}$ with $k \geq 2$.

Output: KTR of $G(A)$

1. for $i \leftarrow 0$ to $k-1$ step 1 do begin
2. \quad $s \leftarrow x_i$; \quad $t \leftarrow x_i+1 (mod\ k)$; \quad $j \leftarrow i+1$; \quad // for simplicity //
3. \quad if ( arcs $s$ and $t$ intersect ) then
4. \quad \quad $Q_j \leftarrow 0$;
5. \quad else begin
6. \quad \quad // find the values of $\alpha(r)$ and $\beta(r)$, for all arc $r$ of $A$ //
7. \quad \quad \quad $r \leftarrow s$;
8. \quad \quad \quad while ( $r \neq t$ ) do begin
9. \quad \quad \quad \quad $\alpha(r) \leftarrow r$; \quad $\beta(r) \leftarrow r$; \quad // initial values //
10. \quad \quad \quad \quad \quad for each $r' \in N[r]$ do begin
11. \quad \quad \quad \quad \quad \quad if ( arcs $s$, $r'$, $\alpha(r)$, and $t$ occur in clockwise order ) then $\alpha(r) \leftarrow r'$;
12. \quad \quad \quad \quad \quad \quad if ( arcs $s$, $\beta(r)$, $r'$ and $t$ occur in clockwise order ) then $\beta(r) \leftarrow r'$;
13. \quad \quad \quad \quad end-for-each
14. \quad \quad \quad $r \leftarrow r+1 (mod\ n)$;
15. \quad \quad end-while
16. \quad // compute the values of $Pr[F(r)]$ in Eq.(1), for each arc $r$ of $A$ //
17. \quad $Pr[F[s-1 (mod\ n)]] \leftarrow 0$; \quad // boundary condition //
18. \quad $r \leftarrow s$;

8
while \(( r \neq t )\) do begin 
\[ q^* \leftarrow q[\beta( r )]; \quad // \text{initial value} // \]
\[ r' \leftarrow r + 1 \mod n; \]
while \(( r' \neq \beta( r ) )\) do begin 
\[ q^* \leftarrow q^* \times q[r']; \]
\[ r' \leftarrow r' + 1 \mod n; \]
end-while 
\[ PrF[ r ] \leftarrow PrF[ r - 1 \mod n ] + (1 - PrF[ \alpha( r ) - 1 \mod n ]) \times (1 - q_r ) \times q^*; \quad // \text{Eq. (1)} // \]
\[ r \leftarrow r + 1 \mod n; \]
end-while 
\[ Q_j \leftarrow PrF[ t - 1 \mod n ]; \]
end-if-else 
end-for 
\[ // \text{compute the values of } f( j, h ), \text{for } 1 \leq j \leq k \text{ and } h = 1 \text{ or } 2 // \]
for \( j \leftarrow 0 \) to \( k \) step 1 do begin 
\[ f( 0, 1 ) \leftarrow 0; \quad // \text{boundary conditions} // \]
\[ f( 0, 2 ) \leftarrow 0; \quad // \text{boundary conditions} // \]
end-for 
for \( j \leftarrow 1 \) to \( k \) step 1 do begin 
\[ f( j, 1 ) \leftarrow Q_j \times f( j - 1, 0 ) + (1 - Q_j ) \times f( j - 1, 1 ); \quad // \text{Eq. (2), for } h = 1 // \]
\[ f( j, 2 ) \leftarrow Q_j \times f( j - 1, 1 ) + (1 - Q_j ) \times f( j - 1, 2 ); \quad // \text{Eq. (2), for } h = 2 // \]
end-for 
return \(( 1 - f( k, 2 ) )\); \quad // \text{return the KTR of } G( \mathcal{A} ) // 
end-algorithm 

**Theorem 2.** Given a proper circular-arc graph \( G \) with \( n \) vertices and \( m \) edges, the KTR problem on \( G \) can be solved in \( O( n + m ) \) time.

**Proof.** The correctness of Algorithm 1 follows from above discussion. Notably, if the corresponding proper circular-arc representation \( \mathcal{A} \) of the proper circular-arc graph \( G \) is not given, then it can be constructed in linear-time using the recognition algorithm [15] for a proper circular-arc graph. First, the values of all \( \alpha( r ) \) and \( \beta( r ) \), for \( r \in \mathcal{A} \), are obtained by executing lines 10 and 11 at most \( O( \sum_{i=0}^{k-1} \sum_{r \in \mathcal{A}} |N[ r ]| ) = O( n + m ) \) times. Next, according to the definition of \( \beta( r ) \), the while-loop of lines 20 to
23 is iterated at most $|N[r]|$ times for each arc $r$. Therefore, the values of all $PrF[r]$, for $r \in A$, can be obtained in $O(\sum_{i=0}^{k-1} |N[r]| )=O(n+m)$ time. Finally, the values of all $f(j,1)$ and $f(j,2)$, $1 \leq j \leq k$, can be computed in $O(k)$ time by executing the for-loop of lines 32 to 35. Accordingly, implementing Algorithm 1 takes $O(n+m)$ time overall, which is linear in the size of $G$.

4. Conclusions

This paper reveals that the KTR problem remains #P-complete even for circle-graphs. This paper also proposes an efficient linear-time algorithm to solve the KTR problem for proper circular-arc graphs, which are a subclass of circle graphs and a superclass of proper interval graphs. Therefore, the classes of intersection graphs with linear-time solvable KTR problems are extended from proper interval graphs to proper circular-arc graphs.

Acknowledgment

The authors would like to thank the Ministry of Science and Technology of Taiwan for financially supporting this research under Contract No. MOST 105-2221-E-027-077-.

References

[1] AboElFotoh, H. M., & Colbourn, C. J. (1990). Efficient algorithms for computing the reliability of permutation and interval graphs. Networks, 20(7), 883-898.
[2] Valiant, L. G. (1979). The complexity of enumeration and reliability problems. SIAM Journal on Computing, 8(3), 410-421.
[3] Golumbic, M. C. (1980). Algorithmic graph theory and perfect graphs. Academic Press.
[4] Dagan, I., Golumbic, M. C., & Pinter, R. Y. (1988). Trapezoid graphs and their coloring. Discrete Applied Mathematics, 21(1), 35-46.
[5] Spinrad, J. (1994). Recognition of circle graphs. Journal of Algorithms, 16(2), 264-282.
[6] Kostochka, A., & Kratochvíl, J. (1997). Covering and coloring polygon-circle graphs. Discrete Mathematics, 163(1), 299-305.
[7] Felsner, S., Müller, R., & Wernisch, L. (1997). Trapezoid graphs and generalizations, geometry and algorithms. Discrete Applied Mathematics, 74(1), 13-32.

[8] Elmallah, E. S., & Stewart, L. K. (1993). Independence and domination in polygon graphs. Discrete Applied Mathematics, 44(1), 65-77.

[9] Janson, S., & Kratochvíl, J. (1992). Thresholds for classes of intersection graphs. Discrete Mathematics, 108(1), 307-326.

[10] Lin, M. S., & Ting, C. C. (2013). Computing K-terminal reliability of d-trapezoid graphs. Information Processing Letters, 113(19), 734-738.

[11] Chen, C. M., & Lin, M. S. (2016). Computing K-terminal reliability of circular-arc graphs. IEICE Transactions on Information and Systems, to be published.

[12] Elmallah, E. S. (1992). Algorithms for K-terminal reliability problems with node failures. Networks, 22(4), 369-384.

[13] Lin, M. S. (2002). A linear-time algorithm for computing K-terminal reliability on proper interval graphs. IEEE Transactions on Reliability, 51(1), 58-62.

[14] Ball, M. O., Provan, J. S., & Shier, D. R. (1991). Reliability covering problems. Networks, 21(3), 345-357.

[15] Lin, M. C., & Szwarcfiter, J. L. (2009). Characterizations and recognition of circular-arc graphs and subclasses: A survey. Discrete Mathematics, 309(18), 5618-5635.