

# Interval Routing Schemes for Circular-Arc Graphs

Frank Gurski  
University of Düsseldorf  
Institute of Computer Science  
Algorithmics for Hard Problems Group  
D-40225 Düsseldorf  
frank.gurski@hhu.de

Patrick Gwydion Poullie\*  
University of Zurich  
Department of Informatics  
Communication Systems Group  
CH-8050 Zürich  
poullie@ifi.uzh.ch

November 2, 2018

## Abstract

Interval routing is a space efficient method to realize a distributed routing function. In this paper we show that every circular-arc graph allows a shortest path strict 2-interval routing scheme, i.e., by introducing a global order on the vertices and assigning at most two (strict) intervals in this order to the ends of every edge allows to depict a routing function that implies exclusively shortest paths. Since circular-arc graphs do not allow shortest path 1-interval routing schemes in general, the result implies that the class of circular-arc graphs has strict compactness 2, which was a hitherto open question. Additionally, we show that the constructed 2-interval routing scheme is a 1-interval routing scheme with at most one additional interval assigned at each vertex and we outline an algorithm to calculate the routing scheme for circular-arc graphs in  $\mathcal{O}(n^2)$  time, where  $n$  is the number of vertices.

**Keywords:** interval routing, compact routing, circular-arc graphs, cyclic permutations

## 1 Introduction

Routing is an essential task that a network of processors or computers must be able to perform. Interval routing is a space-efficient solution to this problem. Sets of consecutive destination addresses that use the same output port are grouped into intervals and then assigned to this port. In this way, the storage space required is greatly reduced compared to the straight forward approach, in which an output port is stored specifically for every destination address. Of course, the advantage in space efficiency depends heavily on the number of intervals assigned to the output ports, which, in turn, depends on the address and network topology. Interval routing was first introduced in [SK82, SK85]; for a discussion of interval routing we refer the reader to [vLT87, FJ88, BvLT91] and for a detailed survey to [Gav00]. When theoretical aspects of interval routing are discussed, the network is represented by a directed (symmetric) graph  $G = (V, A)$ . As with many routing methods, the problem is that space efficiency and path optimality are conflicting goals [PU89]: as shown in [GG98], for every  $n \in \mathbb{N}$ , there exists an  $n$ -vertex graph  $G_n$  such that for every shortest path interval routing scheme (for short IRS) for  $G_n$  the maximal number of intervals per directed edge is only bounded by  $\Omega(n)$ . Also, it is NP-hard to determine the most space efficient shortest path IRS for a given graph [EMZ02, Fla97]. Other worst case results can be found in [TL99]. On the other hand, there are many special graph classes [FG98] including random graphs [GP98] that are known to allow shortest path IRSs with a constantly bounded number of intervals on all directed edges. If this number is a tight bound, it is denoted the compactness of the graph. The compactness of undirected graphs is defined as the compactness of their directed symmetric

---

\*Corresponding author.

version<sup>1</sup> and the compactness of a graph class as the smallest  $k$ , such that every graph in this class has compactness at most  $k$ , if such  $k$  exists.

In this paper we show that the class of circular-arc graphs has (strict) compactness 2 by presenting an algorithm to construct a corresponding IRS. A different approach to realize space efficient, shortest path routing in circular-arc graphs can be found in [DL02]. In [GP08] an  $\mathcal{O}(\log(|V|))$ -bit distance labeling scheme is developed that allows for each pair of vertices in circular-arc graphs to compute their exact distance in  $\mathcal{O}(1)$  time. In [DYL06] an  $\mathcal{O}(\log^2(|V|))$ -bit routing labeling scheme is developed that allows to make a routing decision in  $\mathcal{O}(1)$  time for every vertex in an arbitrary circular-arc graph. The resulting routing path has at most two more edges than the respective shortest path. Nevertheless, our result is interesting, since interval graphs and unit circular-arc graphs are included in the class of circular-arc graphs and known to have compactness 1 [FG98, NS96], while for the compactness of circular-arc graphs hitherto only a lower bound of 2 was known (cf. the graph in Fig. 1) and the solutions from [FG98] and [NS96] cannot be extended to the class of circular-arc graphs.

While interval graphs can be represented by intervals on a line, circular-arc graphs can be represented by arcs on a circle, that is to say, every circular-arc graph is the intersection graph of a set of arcs on a circle. The first polynomial-time ( $\mathcal{O}(|V|^3)$ ) algorithm to recognize circular-arc graphs and give a corresponding circular-arc model can be found in [Tuc80]. An algorithm with linear runtime for the same purpose is given in [McC03]. In circular-arc graphs maximum cliques can be computed much faster than in arbitrary graphs; an algorithm that determines a maximum clique in a circular-arc graph in  $\mathcal{O}(|E| + |V| \cdot \log(|V|))$  time or even in  $\mathcal{O}(|E|)$  time, if the circular-arc endpoints are given in sorted order, is presented in [BK97]. A discussion of circular-arc graphs can be found in [Hsu95]. Circular-arc graphs have, amongst others, applications in cyclic scheduling, compiler design [Gol04, Tuc75], and genetics [Rob76].

This paper is organized as follows. In Section 2 we define cyclic permutations, circular-arc graphs, and interval routing schemes. In Section 3 we prove the main result of this paper, namely that every circular-arc graph allows a shortest path strict 2-interval routing scheme. To this end, we choose an arbitrary circular-arc graph  $G = (V, A)$  and fix a cyclic permutation  $\mathcal{L}$  on  $V$ , in Section 3.1. In Section 3.2 we fix an arbitrary vertex  $v$  and partition  $V - \{v\}$  into the three sets  $A_v$ ,  $B_v$ , and  $C_v$ ; the vertices in these sets appear consecutively in  $\mathcal{L}$  and are referred to as the *vertices to the right of  $v$* , the *vertices face-to-face with  $v$* , and the *vertices to the left of  $v$* , respectively. In Section 3.3 we show that the vertices in the three sets  $A_v$ ,  $B_v$ , and  $C_v$  can be assigned to the edges incident to  $v$ , such that the conditions of a shortest path strict 2-interval routing scheme are fulfilled. In Section 4 we show that the upper bound of the number of intervals in a shortest path strict 2-interval routing scheme constructed in this way is close to the number in an 1-IRS. In Section 5 we outline how to implement the implied IRS in  $\mathcal{O}(|V|^2)$  time.

## 2 Preliminaries

### 2.1 Cyclic Permutations

Cyclic permutations are relevant for the two main subjects of this paper, namely circular-arc graphs and non-linear interval routing schemes. Just like a permutation, a cyclic permutation implies an order on a set with the only difference being that there is no distinct first, second, ..., or  $n$ -th element. We also define cyclic permutations on multisets, where we assume that multiple appearances of an element are distinguishable, i.e., a multiset  $A = \{a_1, \dots, a_n\}$  on  $n$  elements is identified with  $A' = \{(1, a_1), \dots, (n, a_n)\}$ .

**Definition 2.1 (Cyclic Permutation)** *A cyclic permutation on a (multi-)set  $A$  with  $n$  elements is a function  $\mathcal{C}: A \rightarrow A$  such that  $\forall a \in A: \forall i \in \mathbb{N}^+: \mathcal{C}^i(a) = a \Leftrightarrow i \bmod n = 0$ .<sup>2</sup> We call  $\mathcal{C}(a)$  the successor of  $a$  (in  $\mathcal{C}$ ) and we write  $a \succ_{\mathcal{C}} \mathcal{C}(a)$ .*

<sup>1</sup>The directed symmetric version of an undirected graph  $G_u = (V, E)$  is the directed graph  $G_d = (V, A)$  with  $A = \{(v, w), (w, v) \mid \{v, w\} \in E\}$ .

<sup>2</sup>If the condition holds for one element in  $A$ , it holds for all elements in  $A$ .

Every cyclic permutation is a bijective mapping. Let  $\mathcal{C}$  be a cyclic permutation on a (multi-)set  $A = \{a_1, a_2, \dots, a_n\}$ . One could imagine  $\mathcal{C}$  as an arrangement of  $A$  on a clock face, for an  $n$ -hour clock. To go through  $\mathcal{C}$  beginning at  $a_i \in A$  means to consider the elements in  $A$  as they appear in  $\mathcal{C}$ , beginning at  $a_i$ . Next we define subsets of  $A$  that appear consecutively in  $\mathcal{C}$ .

**Definition 2.2 (Ring-Interval, Ring-Sequence)** For a cyclic permutation  $\mathcal{C}$  on a (multi-)set  $A$  and  $a, b \in A$ , we recursively define the ring-interval from  $a$  to  $b$  (in  $\mathcal{C}$ ) as

$$[a, b]_{\mathcal{C}} := \begin{cases} \{a\} & \text{if } a = b \\ \{a\} \cup [\mathcal{C}(a), b]_{\mathcal{C}} & \text{else.} \end{cases}$$

By  $]a, b]_{\mathcal{C}}$  we indicate that the left endpoint is excluded, by  $[a, b[_{\mathcal{C}}$  that the right endpoint is excluded, and by  $]a, b[_{\mathcal{C}}$  that both endpoints are excluded. Therefore we have  $]a, a]$ ,  $[a, a[_$ ,  $]a, a[_ = \emptyset$

The ring-sequence from  $a$  to  $b$  (in  $\mathcal{C}$ ), denoted by  $\mathcal{S}[a, b]_{\mathcal{C}} = (a, \dots, b)$ , is the order in which elements of  $[a, b]_{\mathcal{C}}$  appear when going through  $\mathcal{C}$  beginning at  $a$ .

**Construction 2.3** Let  $\mathcal{C}$  be a cyclic permutation on a (multi-)set  $A$  and  $a, b, c, d \in A$ . If  $[a, b]_{\mathcal{C}} \cap [c, d]_{\mathcal{C}} = \emptyset$  and  $b \succ_{\mathcal{C}} c$ , we have  $[a, b]_{\mathcal{C}} \cup [c, d]_{\mathcal{C}} = [a, d]_{\mathcal{C}}$ .

## 2.2 Circular-Arc Graphs

We assume that the reader is familiar with basic graph theoretical definitions. The *intersection graph* of a family of sets is the graph where the vertices are the sets, and the edges are the pairs of sets that intersect. Every graph is the intersection graph of some family of sets. A graph is an *interval graph* if it is the intersection graph of a finite set of intervals (line segments) on a line and a *unit interval graph* if these intervals have unit length. A graph  $G$  is a *circular-arc graph* if it is the intersection graph of a finite set of arcs on a circle; the latter we call an *arc model* of  $G$ . Since an interval graph is a special case of a circular-arc graph, namely a circular-arc graph that can be represented with a set of arcs that do not cover the entire circle, we define the set of *strict circular-arc graphs* as the set of circular-arc graphs that are not interval graphs. Every strict circular-arc graph is connected. A *unit circular-arc graph* is a circular-arc graph that has an arc model in which the arcs have unit length. For a survey on circular-arc graphs see [LS09] and for the definition of further special graph classes see [BLS99]. Fig. 1 illustrates a strict circular-arc graph.

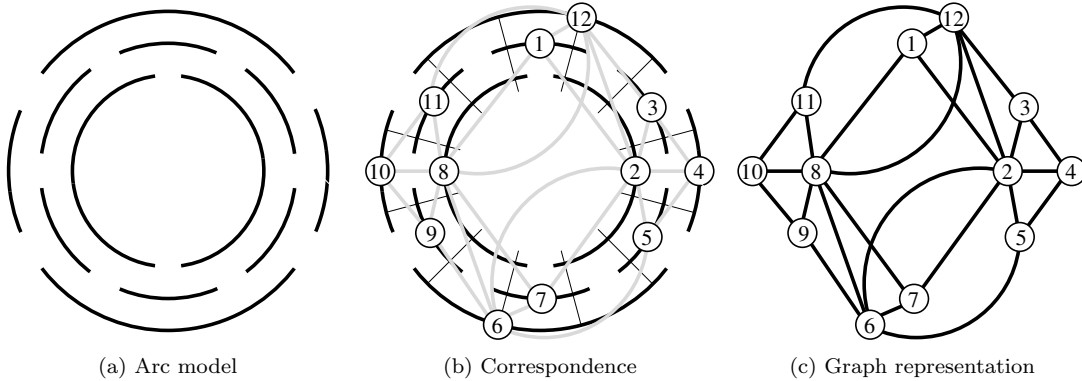


Figure 1: A (strict) circular-arc graph

A difference between interval graphs and circular-arc graphs, that is worth mentioning, is that the maximal cliques of interval graphs can be associated to points of the “interval model” and therefore an interval graph can have no more maximal cliques than vertices. In contrast, circular-arc graphs may contain maximal cliques that do not correspond to points of some arc model

[LS09]. In fact, just as in arbitrary graphs, the number of maximal cliques in circular-arc graphs can grow exponentially in the size of the graph [Tuc80]. For the special case that there is an arc model where no three arcs cover the whole circle the number of maximal cliques is bounded by the number of vertices of the graph [CFZ08]. Also maximal cliques may occur several times within some arc model, which is addressed by multisets in this paper.

The following definition and the subsequent corollary formalize important graph-theoretic properties of circular-arc graphs.

**Definition 2.4 (Clique-Cycle, Left Clique  $\alpha(v)$ , and Right Clique  $\beta(v)$ )** Let  $G = (V, E)$  be a circular-arc graph,  $M_G$  be an arc model of  $G$ , and  $C$  be the circle of  $M_G$ . To each point  $p$  on  $C$  corresponds a clique that contains the vertices whose corresponding arcs contain  $p$ . Let multiset  $Y$  contain all cliques that correspond to points on  $C$  and define  $X \subseteq Y$  as the multiset of cliques we obtain, when we remove all cliques from  $Y$  that are not maximal with respect to inclusion among the cliques in  $Y$  and all but one cliques that are equal. Pair  $(X, \mathcal{C})$ , where  $\mathcal{C}$  is the cyclic permutation on  $X$  that is implied by ordering the elements in  $X$  as their corresponding points appear (clockwise) on  $C$ , is called a clique-cycle for  $G$ . For every vertex  $v \in V$ , there exist two cliques  $a, b \in X$  such that a clique  $x \in X$  contains  $v$  if and only if  $x \in [a, b]_{\mathcal{C}}$ . We call  $a$  the left clique of  $v$ , denoted by  $\alpha(v)$ , and  $b$  the right clique of  $v$ , denoted by  $\beta(v)$ .

When circular-arc graphs are discussed in this paper, the argumentation is based on one of their clique-cycles deduced from some arc model.

A frequently used notation in our paper is that of a *dominating vertex*, which is a vertex adjacent to all other vertices of the graph. The set of all dominating vertices of graph  $G$  is denoted by  $\mathcal{D}(G)$ . Since a dominating vertex  $d$  is adjacent to all the vertices, all the maximal cliques contain it, wherefore  $d$ 's left and right clique form a cyclic interval that covers the entire clique cycle.

**Corollary 2.5** Let  $(X, \mathcal{C})$  be a clique-cycle for a circular-arc graph  $G = (V, E)$  and  $d \in \mathcal{D}(G)$  a dominating vertex. The left clique  $\alpha(d)$  and the right clique  $\beta(d)$  of  $d$  are not unique, but can be chosen arbitrarily with  $\beta(d) \succ_{\mathcal{C}} \alpha(d)$  being the only constraint.

Consider the arc model of a circular-arc graph and two intersecting arcs  $v, w$ . The arcs  $v$  and  $w$  can intersect in the following two ways: 1. The intersection of  $v$  and  $w$  constitutes another arc. This is the case, (a) if  $v$  is included in  $w$  or vice versa, (b) if  $v$  and  $w$  are congruent, i.e., cover the exact same part of the cycle, or (c) if exactly one endpoint of  $v$  lies in  $w$  and vice versa. 2. The intersection of  $v$  and  $w$  constitutes two arcs (the corresponding vertices will be defined as counter vertices in the subsequent definition). This is the case, if  $v$  and  $w$  jointly cover the circle and meet at both ends.

**Definition 2.6 (Counter Vertex  $\mathcal{C}(v)$ )** Let  $(X, \mathcal{C})$  be a clique-cycle for a circular-arc graph  $G = (V, E)$ . We call  $w \in V$  a counter vertex of  $v \in V$ , if  $\alpha(v), \beta(v) \in [\alpha(w), \beta(w)]_{\mathcal{C}}$  and  $\alpha(w), \beta(w) \in [\alpha(v), \beta(v)]_{\mathcal{C}}$  and  $\alpha(v) \neq \alpha(w)$  (or, equivalently,  $\beta(v) \neq \beta(w)$ ). We call  $(v, w)$  a pair of counter vertices and denote the set of all counter vertices of  $v$  by  $\mathcal{C}(v)$ .<sup>3</sup>

The next corollary follows directly from the arc model of a circular-arc graph.

**Corollary 2.7** Let  $G = (V, E)$  be a circular-arc graph and  $v \in V$  be a vertex with  $\mathcal{C}(v) \neq \emptyset$ . Then every vertex in  $V$  is adjacent to  $v$  or to every vertex in  $\mathcal{C}(v)$ , in other words,  $v$  and any of its counter vertices constitute a dominating set.

**Definition 2.8 (Reaching to the Left/Right)** Let  $(X, \mathcal{C})$  be a clique-cycle for a circular-arc graph  $G = (V, E)$ ,  $v, w$  two adjacent vertices with  $v \notin \mathcal{C}(w)$  and  $x \in X$  a clique with  $v, w \in x$ . We say,  $w$  reaches at least as far to the right as  $v$  if  $[x, \beta(v)]_{\mathcal{C}} \subseteq [x, \beta(w)]_{\mathcal{C}}$ , and  $w$  reaches further to the right than  $v$  if  $[x, \beta(v)]_{\mathcal{C}} \subset [x, \beta(w)]_{\mathcal{C}}$ . Analogously, we define  $w$  reaches at least as far to the left as  $v$  by  $[\alpha(v), x]_{\mathcal{C}} \subseteq [\alpha(w), x]_{\mathcal{C}}$  and  $w$  reaches further to the left than  $v$  by  $[\alpha(v), x]_{\mathcal{C}} \subset [\alpha(w), x]_{\mathcal{C}}$ .

---

<sup>3</sup>Note that,  $w \in \mathcal{C}(v) \Leftrightarrow v \in \mathcal{C}(w)$ .

Although the arcs corresponding to counter vertices cover very different areas of the circle, it is impossible to say which of the two arcs “reaches further to the left or right”, when the point of view is the middle of the circle.

## 2.3 Interval Routing Schemes

We assume that the reader is familiar with basic concepts of interval routing and refer to [Gav00] for an exhaustive introduction and survey. If an IRS assigns at most  $k$  intervals to each (directed) edge and only implies shortest paths, we denote it by  $k$ -IRS. If this  $k$ -IRS is furthermore strict, which means that for every vertex  $v \in V$  no interval assigned to the outgoing edges of  $v$  contains  $v$ 's number, it is denoted by  $k$ -SIRS. Definition 2.9 is needed for Definition 2.10, which is an alternative definition for shortest path strict interval routing schemes that is equivalent to definitions in literature but better suited for our purposes. To apply Definition 2.10 to an undirected graph (as for example a circular-arc graph), the graph is converted to its directed, symmetric version. While directed edges are often called arcs, we choose the former term to avoid confusion with the arcs of a graph's arc model.

**Definition 2.9 (First Vertex  $\mathcal{F}(u, w)$ )** *Let  $G = (V, A)$  be a directed graph and  $u, v, w \in V$ . Vertex  $v \in V$  is a first vertex from  $u$  to  $w$ , if there exists a shortest directed path  $(u, v, \dots, w)$ , i.e., there is a shortest directed path from  $u$  to  $w$ , that first traverses  $v$ . The set of first vertices from  $u$  to  $w$  is denoted by*

$$\mathcal{F}(u, w) = \{v \in V \mid v \text{ is a first vertex from } u \text{ to } w\}.$$

**Definition 2.10 (Shortest Path Strict Interval Routing Scheme)** *Let  $G = (V, E)$  be a directed graph. A shortest path strict interval routing scheme for  $G$  is a pair  $R = (\mathcal{L}, \mathcal{I})$ , where  $\mathcal{L}$  is a cyclic permutation on  $V$ , called the vertex order, and  $\mathcal{I}$ , called the directed-edge-labeling, maps every directed edge to a set of ring-intervals in  $\mathcal{L}$  such that for every vertex  $v \in V$ ,*

1.  $\mathcal{I}$  maps the outgoing directed edges of  $v$  to a set of ring-intervals in  $\mathcal{L}$ , such that the intervals assigned to different directed edges never intersect,
2. for every vertex  $u \neq v$  one of these these ring-intervals contains  $u$  (vertex  $v$  must not appear in one of these intervals), and
3. if  $u$  is contained in a ring-interval in  $\mathcal{I}((v, w))$ , then  $w$  is a first vertex from  $v$  to  $u$ .

*Let  $\mathcal{F}$  be a vertex order. We say that  $v \in V$  given  $\mathcal{F}$  suffices a shortest path  $k$ -SIRS, if there exists a directed edge-labeling for  $G$  that maps every outgoing directed edge of  $v$  to at most  $k$  ring-intervals in  $\mathcal{F}$ , such that  $v$  satisfies the three constraints above.*

**Corollary 2.11** *A directed graph  $G = (V, A)$  supports a shortest path  $k$ -SIRS, if a vertex order  $\mathcal{F}$  exists, such that every vertex  $v \in V$  given  $\mathcal{F}$  suffices a shortest path  $k$ -SIRS.*

## 3 Main result

The compactness of the class of circular-arc graphs has not been determined until now, which is somehow surprising, since the class of circular-arc graphs is closely related to the class of interval graphs and unit circular-arc graphs and both classes are known to have strict compactness 1 [NS96, FG98]. However, for the compactness of the class of circular-arc graphs only a lower bound of 2 was known. In particular, circular-arc graphs exist that do not allow for optimal 1 interval routing schemes, as, for example, a wheel graph, i.e., a cycle together with a dominating vertex, with six outer vertices [FG98]. Also the circular-arc graph shown in Fig. 1 does not allow optimal 1 interval routing schemes. In this section we prove the main result of this paper, which is given by the following theorem and shows that the lower bound of 2 for the compactness of circular-arc graphs is indeed sharp.

**Theorem 3.1 (Main Theorem)** *The class of circular-arc graphs has strict compactness 2.*

Since every non-strict circular-arc graph  $G$  is an interval graph and therefore has strict compactness 1 [NS96], we only need to consider strict circular-arc graphs in the proof. For the rest of this section let  $(X, \mathcal{C})$  be a clique-cycle for an arbitrary strict circular-arc graph  $G_u = (V, E)$  and  $G = (V, A)$  be the directed symmetric version of  $G_u$ .

**Outline of the proof** To show that there exists a shortest path 2-SIRS for  $G$  and thus Theorem 3.1 holds, we proceed as follows. In Section 3.1 we construct a cyclic permutation  $\mathcal{L}$  on  $V$ . In Section 3.2 we choose an arbitrary vertex  $v \in V$  and partition the vertices in  $V - \{v\}$  in three disjoint ring-intervals  $A_v, B_v, C_v$  in  $\mathcal{L}$  in dependency on  $v$ . In order to show that  $v$  given  $\mathcal{L}$  suffices a shortest path 2-SIRS, we show how to define an directed edge-labeling  $\mathcal{I}_v$  for the outgoing directed edges of  $v$  that satisfies the corresponding constraints given in Definition 2.10. We consider each of the three ring-intervals  $A_v, B_v, C_v$  in Section 3.3.1 and 3.3.2 and show how the vertices in the respective ring-interval can be sufficiently mapped to by  $\mathcal{I}_v$ . Since we have chosen  $v$  arbitrarily, it follows that every vertex in  $V$  suffices a shortest path 2-SIRS given  $\mathcal{L}$  and therefore, by Corollary 2.11,  $G$  supports a shortest path 2-SIRS. Since  $G_u$  is an arbitrary strict circular-arc graph, this proves Theorem 3.1.

The following notations are straight forward but may be formalized for convenience. Let  $(v, w) \in E$  be a directed edge with  $\mathcal{I}_v((v, w)) = S$ , where  $S$  is a set of ring-intervals in  $\mathcal{L}$ , and  $R$  be a ring-interval in  $\mathcal{L}$ . *Assigning  $R$  to directed edge  $(v, w)$*  means to define  $\mathcal{I}_v: (v, w) \mapsto S \cup \{R\}$ . Of course, when we start constructing  $\mathcal{I}_v$ , every directed edge is mapped to the empty set. Let  $[a, b]_{\mathcal{L}}$  and  $[c, d]_{\mathcal{L}}$  be two ring-intervals that can be joined to one ring-interval  $[a, d]_{\mathcal{L}}$  by Construction 2.3. If  $[a, b]_{\mathcal{L}}$  and  $[c, d]_{\mathcal{L}}$  are assigned to the same directed edge  $(v, w)$ , we can redefine

$$\mathcal{I}_v: (v, w) \mapsto \{[a, b]_{\mathcal{L}}, [c, d]_{\mathcal{L}}\} \text{ by } \mathcal{I}_v: (v, w) \mapsto \{[a, d]_{\mathcal{L}}\}$$

and therefore save one ring-interval on directed edge  $(v, w)$ . This redefinition is called *compressing the (two) ring-intervals on directed edge  $(v, w)$  (to one ring-interval)*. Let  $V' \subseteq V$  be a set of vertices. *To distribute  $V'$  (over the outgoing directed edges of  $v$ )* means to partition  $V'$  into ring-intervals and assign these to outgoing directed edges of  $v$  (sufficiently). For convenience, we sometimes refer to ring-intervals as *intervals*.

### 3.1 Definition of the Vertex Order

In this section we consider a circular-arc graph  $G = (V, E)$  together with a clique-cycle  $(X, \mathcal{C})$  and show how to construct a cyclic permutation  $\mathcal{L}$  on  $V$  that serves as the given vertex order. The ordering is obtained by sorting the arcs using their left cliques as primary sort key and right cliques as secondary sort key, making the first arc the successor of the last. A pseudocode for this purpose is presented in Listing 1 and its idea is explained next.

By Corollary 2.5 the left and right clique of a dominating vertex are not unique. Since it simplifies the proof, if all dominating vertices have the same left and thus also the same right clique, these cliques are unified in Line 1. For a fixed  $z$  in Line 1, the generated cyclic permutation is fully deterministic, except for the ordering of true twins.<sup>4</sup> The dummy vertex introduced in Line 4 is needed to close the cyclic order once all vertices are integrated in  $\mathcal{L}$ . The loop in Line 6 runs once through all cliques in  $X$  (beginning at  $f$ , that is arbitrarily chosen in Line 2) in the order defined by  $\mathcal{C}$  (Line 13). For every visited clique  $n$  the loop in Line 8 integrates every vertex, whose left clique is  $n$ , in  $\mathcal{L}$ . By Line 9, vertices with the same left clique are ordered with respect to their right clique. An example for the defined vertex ordering can be found in Fig. 1.

- 1 Fix  $z \in X$  arbitrarily and choose  $\alpha(d) := \mathcal{C}(z)$  and  $\beta(d) := z$  for all  $d \in \mathcal{D}(G)$
- 2 Choose  $f \in X$  arbitrarily
- 3  $n := f$

<sup>4</sup>Two vertices in a graph are called *true twins* if they are adjacent to the same set of vertices and to each other.

```

4   $V := V \cup \{t\}$  /* add dummy vertex  $t$  */
5   $p := t$ 
6  do
7       $A := \{v \in V \mid \alpha(v) = n\}$ 
8      while( $A \neq \emptyset$ )
9          Choose  $a \in A$  such that every vertex in  $A$  reaches at least as far to the right as  $a$ .
10          $\mathcal{L}(p) := a$ 
11          $p := a$ 
12          $A := A - \{a\}$ 
13      $n := \mathcal{C}(n)$ 
14 while( $n \neq f$ )
15  $\mathcal{L}(p) := \mathcal{L}(t)$ 
16  $V = V - \{t\}$  /* remove dummy vertex  $t$  */
17 return  $\mathcal{L}$ 

```

Listing 1: Definition of a cyclic order  $\mathcal{L}$  on  $V$  that serves as the given vertex order.

The following observations are crucial for the rest of this paper.

- Remark 3.2** 1. The vertices are ordered primarily by their left clique, that is to say, vertices having the same left clique appear consecutively in  $\mathcal{L}$ .
2. Vertices with the same left clique are ordered in ascending order by their right clique.
3. For two vertices  $v, w \in V$  with  $v \succ_{\mathcal{L}} w$ , we know that  $v$  and  $w$  are adjacent and either
- (a)  $\alpha(v) = \alpha(w)$  and  $w$  reaches at least as far to the right as  $v$ , or
  - (b)  $\alpha(v) \succ_{\mathcal{C}} \alpha(w)$ , in other words,  $v$  reaches one clique further to the left than  $w$ .<sup>5</sup>

The next definition is based on Assertion 1 of the preceding remark, that is to say, on the fact that, since vertices having the same left clique appear consecutively in  $\mathcal{L}$ , for a given clique  $c \in X$ , two vertices  $v, w \in c$  exist, such that the vertices in ring-interval  $[v, w]_{\mathcal{L}}$  are exactly those with left clique  $c$ .

**Definition 3.3 (Head Vertex  $\mathcal{H}(c)$  and Tail Vertex  $\mathcal{T}(c)$ )** Let  $(X, \mathcal{C})$  be a clique-cycle for some circular-arc graph  $G = (V, E)$ ,  $\mathcal{L}$  given by Listing 1,  $c \in X$ , and  $v, w \in c$  the unique vertices such that  $[v, w]_{\mathcal{L}} = \{x \in V \mid \alpha(x) = c\}$  ( $v$  and  $w$  are not necessarily distinct). We call  $v$  the head vertex of  $c$  and  $w$  the tail vertex of  $c$  and denote them by  $\mathcal{H}(c)$  and  $\mathcal{T}(c)$ , respectively.

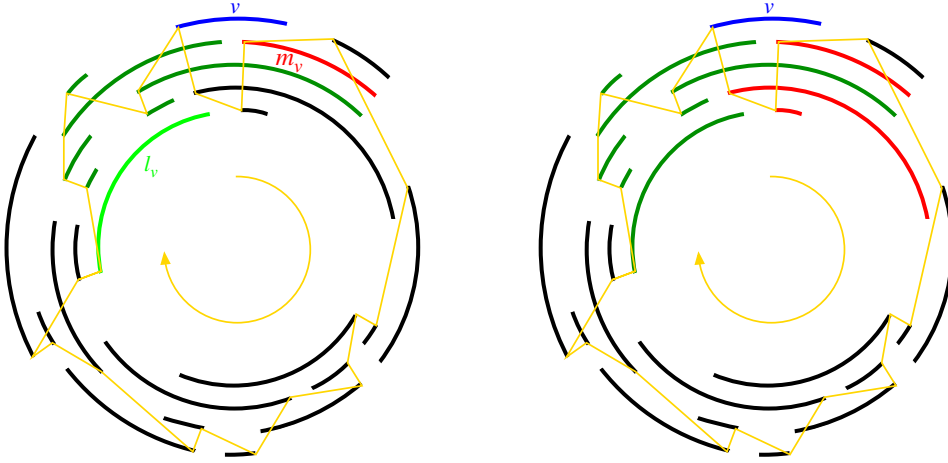
**Corollary 3.4** Let  $(X, \mathcal{C})$  be a clique-cycle for a circular-arc graph  $G = (V, E)$ ,  $\mathcal{L}$  given by Listing 1, and  $c \in X$ . The successor (in  $\mathcal{L}$ ) of the tail vertex of  $c$  is the head vertex of the successor (in  $\mathcal{C}$ ) of  $c$ , i.e.,  $\mathcal{L}(\mathcal{T}(c)) = \mathcal{H}(\mathcal{C}(c))$ .

### 3.2 Partition of the Vertex Order

The cyclic permutation  $\mathcal{L}$  on  $V$  allows to state further definitions. We now fix  $v \in V$  arbitrarily for the rest of the proof and show that  $v$  given  $\mathcal{L}$  suffices a shortest path 2-SIRS. If  $v$  is a dominating vertex, we can define  $\mathcal{I}_v: (v, w) \mapsto [w, w]_{\mathcal{L}}$  for every vertex  $w \in V - \{v\}$  and have thereby shown that  $v$  given  $\mathcal{L}$  suffices a shortest path 1-SIRS. Therefore, we now assume that  $v$  is not a dominating vertex. Fig. 2a illustrates the subsequent Definition.

**Definition 3.5 (Left Vertex  $l_v$ )** Let  $(X, \mathcal{C})$  be a clique-cycle for some circular-arc graph  $G = (V, E)$ ,  $\mathcal{L}$  given by Listing 1, and  $v \in V$  not a dominating vertex. Let  $L \subset V$  be the union of the following two vertex sets. (i) The set of vertices that are adjacent to  $v$  and reach further to the

<sup>5</sup>Because the relation “reaching to the right” is not defined, if  $v$  and  $w$  are counter vertices, we intuitively extend Definition 2.8 to this case by fixing  $x = \alpha(w)$ .



(a) The green vertices are in set  $L$  of Definition 3.5. The left vertex of  $v$ , which is also in  $L$ , is colored in light green and the middle vertex of  $v$  in red. (b) The green vertices are to the left of  $v$ , the red vertices are to the right of  $v$ , and the black vertices are face-to-face with  $v$ .

Figure 2: Two examples for the Definitions made in this section. The fixed vertex  $v$  is colored blue. The yellow line connects the vertices as they appear in vertex order  $\mathcal{L}$  as generated by Listing 1.

left than  $v$  but are neither counter vertices of  $v$  nor dominating vertices. (ii) The set of vertices in  $[\mathcal{H}(\alpha(v)), v]_{\mathcal{L}}$ .<sup>6</sup> If  $L \neq \emptyset$ , we define the left vertex of  $v$ , denoted by  $l_v$ , as the unique vertex in  $L$  such that

$$\forall w \in L: [w, v]_{\mathcal{L}} \subseteq [l_v, v]_{\mathcal{L}} \text{ (or, equivalently, } L \subseteq [l_v, v]_{\mathcal{L}}). \quad (1)$$

**Theorem 3.6** Let  $(X, \mathcal{C})$  be a clique-cycle for a circular-arc graph  $G = (V, E)$ ,  $\mathcal{L}$  given by Listing 1,  $v \in V$  not a dominating vertex,  $l_v$  the left vertex of  $v$ , and  $u \neq l_v$  be a vertex that is adjacent to  $v$  and reaches further to the left than  $v$  but is neither a dominating vertex nor a counter vertex of  $v$ . Then  $l_v$  reaches at least as far to the left as  $u$ .

**Proof** Assume  $u$  reaches further to the left than  $l_v$ . Then by Definition 2.8 we know that  $[\alpha(l_v), \alpha(v)]_{\mathcal{C}} \subset [\alpha(u), \alpha(v)]_{\mathcal{C}}$ , which implies that

$$\mathcal{S}[\alpha(u), \alpha(v)]_{\mathcal{C}} = (\alpha(u), \dots, \alpha(l_v), \dots, \alpha(v)).$$

Since  $\mathcal{L}$  is primary ordered by the left cliques of the vertices, it follows

$$u \succ_{\mathcal{L}} \dots \succ_{\mathcal{L}} l_v \succ_{\mathcal{L}} \dots \succ_{\mathcal{L}} v$$

and therefore  $[l_v, v]_{\mathcal{L}} \subset [u, v]_{\mathcal{L}}$ . This contradicts (1) in Definition 3.5, since  $u \in L$  reaches further to the left than  $v$  and is neither a dominating vertex nor a counter vertex of  $v$ .  $\square$

In Definition 3.8 we partition the vertices in  $V - \{v\}$  into three ring-intervals in  $\mathcal{L}$ . The following argumentation gives the idea behind this partitioning and also proves Theorem 3.9. Consider we go through  $\mathcal{L}$  beginning at  $v$ . Since the vertices in  $\mathcal{L}$  are primary ordered by their left cliques, the left clique of the vertices first considered is  $\alpha(v)$  (or  $\mathcal{C}(\alpha(v))$ , if  $v$  is the tail vertex of clique  $\alpha(v)$ ). The next vertices traversed have left clique  $\mathcal{C}(\alpha(v))$ , followed by vertices with left clique  $\mathcal{C}^2(\alpha(v))$ , and so on. We eventually reach the set of vertices with left clique  $\beta(v)$ . The last vertex we come across in this set is  $\mathcal{T}(\beta(v))$  and fixed in Definition 3.7. The vertices we came across so far (excluding  $v$ , including  $\mathcal{T}(\beta(v))$ ) will be defined as the “vertices to the right of  $v$ ” in Definition 3.8 and are adjacent to  $v$ , as  $v$  was contained in their left clique.

<sup>6</sup>We have  $\mathcal{D}(G) \cap [\mathcal{H}(\alpha(v)), v]_{\mathcal{L}} = \emptyset$  and  $\mathcal{C}(v) \cap [\mathcal{H}(\alpha(v)), v]_{\mathcal{L}} = \emptyset$ .



**Definition 3.7 (Middle Vertex  $m_v$ )** Let  $(X, \mathcal{C})$  be a clique-cycle for a circular-arc graph  $G = (V, E)$ ,  $\mathcal{L}$  given by Listing 1, and  $v \in V$  not a dominating vertex. We define the middle vertex of  $v$ , denoted by  $m_v = \mathcal{T}(\beta(v))$ , as the tail vertex of the right clique of  $v$ .

When we continue visiting the vertices after  $m_v$  in the same manner, the next vertex  $u$  we come across that is not a dominating vertex but adjacent to  $v$  has to be  $l_v$ , as we could find a contradiction in the same manner as in the proof of Theorem 3.6, if we had  $u \neq l_v$ . These vertices after  $m_v$  and before  $l_v$  are defined as the “vertices face-to-face with  $v$ ” in Definition 3.8. When we continue to go through  $\mathcal{L}$ , we eventually reach  $v$  again. These vertices from  $l_v$  to  $v$  (excluding  $v$ ) are defined as the “vertices to the left of  $v$ ” in Definition 3.8, which is illustrated in Fig. 2b.

**Definition 3.8 ( $A_v, B_v, C_v$ )** Let  $(X, \mathcal{C})$  be a clique-cycle for a circular-arc graph  $G = (V, E)$ ,  $\mathcal{L}$  given by Listing 1,  $v \in V$  not a dominating vertex,  $l_v$  the left vertex of  $v$ , if  $l_v$  exists, and  $m_v$  the middle vertex of  $v$ . We define

1. the vertices to the right of  $v$  as  $A_v = ]v, m_v[_{\mathcal{L}}$ , and
2. the vertices face-to-face with  $v$ 
  - (a) as  $B_v = ]m_v, v[_{\mathcal{L}}$ , if  $l_v$  does not exist, or else
  - (b) as  $B_v = ]m_v, l_v[_{\mathcal{L}}$ , and
3. the vertices to the left of  $v$ 
  - (a) as  $C_v = [l_v, v[_{\mathcal{L}}$ , if  $l_v$  exists, or else
  - (b) as  $C_v = \emptyset$ .<sup>7</sup>

The next theorem follows from the argumentation between Theorem 3.6 and Definition 3.8.

**Theorem 3.9** Let  $(X, \mathcal{C})$  be a clique-cycle for a circular-arc graph  $G = (V, E)$ ,  $\mathcal{L}$  given by Listing 1, and  $v \in V$  but not a dominating vertex.

1. Every vertex to the right of  $v$  is adjacent to  $v$ .
2. Every vertex face-to-face with  $v$  that is not a dominating vertex is not adjacent to  $v$ .

### 3.3 Definition of the Directed Edge-labeling

Next we show that  $v$  given  $\mathcal{L}$  suffices a shortest path 2-SIRS by constructing a mapping  $\mathcal{I}_v$  from the set of outgoing directed edges of  $v$  to at most two ring-intervals in  $\mathcal{L}$ , according to Definition 2.10. We investigate the vertices to the left of  $v$  and the vertices to the right of  $v$  in Section 3.3.1 and the vertices face-to-face with  $v$  in Section 3.3.2. For every vertex  $u$  in the respective interval, we determine a first vertex  $w$  from  $v$  to  $u$  such that  $u$  and every vertex that is hitherto assigned to directed edge  $(v, w)$  can be embraced by at most two ring-intervals in  $\mathcal{L}$ . This is a simple task for the vertices in  $A_v$  and  $C_v$  and even for the vertices in  $B_v$  the approach is straight forward, if there is a dominating vertex or a counter vertex of  $v$ . In fact, it only gets tricky, if there are no dominating vertices and no counter vertices.

#### 3.3.1 Vertices to the Left and Right

By the first assertion of Theorem 3.9 every vertex to the right of  $v$  is adjacent to  $v$ , wherefore these vertices can be distributed by assigning  $[w, w]_{\mathcal{L}}$  to directed edge  $(v, w)$ , for every vertex  $w \in A_v$ . Assume that we have  $C_v \neq \emptyset$ . In general, not every vertex in  $C_v$  is adjacent to  $v$ . Although  $l_v$  is adjacent to all vertices in  $C_v$ , it is generally not possible to assign  $C_v$  to directed edge  $(v, l_v)$ , since  $C_v$  may contain vertices that are adjacent to  $v$  (other than  $l_v$ ), and therefore must be assigned to

---

<sup>7</sup> We have  $C_v = L$  from Definition 3.5.

“their own directed edge”. Let  $v_i \in C_v$  be a vertex that is adjacent to  $v$  and assume we start at  $v_i$  to go through  $\mathcal{L}$  until we come across the next vertex that is adjacent to  $v$  or  $v$  itself. Denote this vertex  $v_j$ . We can show that  $v_i$  is adjacent to every in  $]v_i, v_j]_{\mathcal{L}}$ , wherefore we can assign  $[v_i, v_j]_{\mathcal{L}}$  to directed edge  $(v, v_i)$ . Therefore the vertices to the left of  $v$  can be distributed by assigning every vertex  $w \in C_v$  to directed edge  $(v, w)$ , if  $w$  is adjacent to  $v$ , and else to directed edge  $(v, w')$ , where  $w'$  is the first vertex that precedes  $w$  in  $\mathcal{L}$  and is adjacent to  $v$ . Theorem 3.10 proves the outlined idea formally. When distributing the vertices in sets  $A_v$  and  $C_v$  as just outlined, every outgoing directed edge of  $v$  (except for edges incident to dominating vertices face-to-face with  $v$ , which are not yet labeled) gets one ring-interval assigned.

**Theorem 3.10** *Let  $(X, \mathcal{C})$  be a clique-cycle for a circular-arc graph  $G = (V, E)$ ,  $\mathcal{L}$  given by Listing 1,  $v \in V$  but not a dominating vertex, and  $l_v$  the left vertex of  $v$  (then the vertices in  $[l_v, v]_{\mathcal{L}}$  are the vertices to the left of  $v$ ). Let  $S[l_v, v]_{\mathcal{L}} = (l_v = v_1, v_2, \dots, v_n = v)$  be the sequence of vertices that we obtain, when we order the vertices in  $[l_v, v]_{\mathcal{L}}$  that are adjacent to  $v$  as they appear in ring-sequence  $S[l_v, v]_{\mathcal{L}}$  and append  $v$ . For  $1 \leq i < n$  and  $x \in [v_i, v_{i+1}]_{\mathcal{L}}$ , vertex  $v_i$  is a first vertex from  $v$  to  $x$ .*

**Proof** For some  $i$ ,  $1 \leq i < n$ , let  $x \in [v_i, v_{i+1}]_{\mathcal{L}}$ . Clearly the theorem holds if  $x = v_i$ . If  $x \neq v_i$ ,  $x$  does not appear in  $(v_1, v_2, \dots, v_n)$  and therefore is not adjacent to  $v$ . Since the vertices in  $\mathcal{L}$  are primary ordered by their left clique and  $v_i$  appears before  $x$  in  $S[l_v, v]_{\mathcal{L}}$ , it follows that the left clique of  $x$  is in  $[\alpha(v_i), \alpha(v)]_{\mathcal{C}}$ . Since  $v_i$  is adjacent to  $v$ ,  $v_i$  is contained in every clique in  $[\alpha(v_i), \alpha(v)]_{\mathcal{C}}$ . This implies that  $v_i$  is contained in the left clique of  $x$  and thus  $v_i$  and  $x$  are adjacent. Therefore  $(v, v_i, x)$  is a shortest path between  $v$  and  $x$  and  $v_i$  is a first vertex from  $v$  to  $x$ .  $\square$

### 3.3.2 Vertices Face-to-Face

This section discusses the distribution of vertices face-to-face with  $v$ . It is probably the common case, that neither counter nor dominating vertices exist, since only then path lengths are unbounded. Thus, this case is discussed in the next paragraph and the subsequent paragraph discusses the remaining cases.

**3.3.2.1 Neither Dominating nor Counter Vertices exist** In this case,  $l_v$  always reaches further to the left than  $v$  and there is at least one vertex adjacent to  $v$  that reaches further to the right. Let  $w \in V$  be a vertex that is not adjacent to  $v$ ,  $L$  be the set of vertices adjacent to  $v$  that reach farthest to the left and  $R$  be the set of vertices adjacent to  $v$  that reach farthest to the right ( $L$  and  $R$  might intersect). As evident from the arc model of  $G$ , every vertex in  $L$  or every vertex in  $R$  is a first vertex from  $v$  to  $w$  (cf. Corollary 3.13). Since we have  $l_v \in L$ , we now fix a vertex adjacent to  $v$  that reaches farthest to the right.

**Definition 3.11 (Right Vertex  $r_v$ )** *Let  $(X, \mathcal{C})$  be a clique-cycle for a strict circular-arc graph  $G = (V, E)$  without dominating vertices and without counter vertices,  $\mathcal{L}$  given by Listing 1,  $v \in V$ ,  $l_v$  the left vertex of  $v$ ,  $m_v$  the middle vertex of  $v$ , and  $R \subset V$  be the set of vertices that are adjacent to  $v$  and reach farthest to the right. We define the right vertex of  $v$ , denoted by  $r_v$ , as  $l_v$ , if  $l_v \in R$ , or as  $m_v$ , if  $m_v \in R$ , or else as an arbitrary vertex in  $R$ .*

In order to show that  $v$  suffices a shortest path 2-SIRS given  $\mathcal{L}$ , we could choose  $r_v$  arbitrarily in  $R$  even if  $l_v \in R$  or  $m_v \in R$ . However, we will show that  $v$  given  $\mathcal{L}$  suffices a shortest path 1-SIRS, if  $l_v \in R$  or  $m_v \in R$ . The next definition redefines the notation of the left and right vertex as a function, in order to allow recursive usage to easily determine a first vertex from  $v$  to every vertex. The subsequent corollary is clear when illustrated.

**Definition 3.12 ( $l(v)$ ,  $r(v)$  and  $\mathcal{L}_v(u)$ ,  $\mathcal{R}_v(u)$ )** *Let  $(X, \mathcal{C})$  be a clique-cycle for a strict circular-arc graph  $G = (V, E)$  without dominating vertices and without counter vertices, cyclic permutation*

$\mathcal{L}$  given by Listing 1, and  $v \in V$ . We define  $l(v)$  as the left vertex of  $v$  and  $r(v)$  as the right vertex of  $v$ .<sup>8</sup> For  $u \in V$ , we define  $\mathcal{L}_v(u)$  as the smallest  $i$  such that  $u$  is adjacent to  $l^i(v)$  and  $\mathcal{R}_v(u)$  as the smallest  $i$  such that  $u$  is adjacent to  $r^i(v)$ .

**Corollary 3.13** *Let  $(X, \mathcal{C})$  be a clique-cycle for a strict circular-arc graph  $G = (V, E)$  without dominating vertices and without counter vertices, cyclic permutation  $\mathcal{L}$  given by Listing 1, and  $v \in V$ . The following three assertions hold for every vertex  $u \in V$  that is not adjacent to  $v$ .*

1. *If  $\mathcal{L}_v(u) < \mathcal{R}_v(u)$ , then  $l_v$  is a first vertex from  $v$  to  $u$ .*
2. *If  $\mathcal{L}_v(u) > \mathcal{R}_v(u)$ , then  $r_v$  is a first vertex from  $v$  to  $u$ .*
3. *If  $\mathcal{L}_v(u) = \mathcal{R}_v(u)$ , then  $l_v$  and  $r_v$  are first vertices from  $v$  to  $u$ .*

Since no vertex in  $B_v$  is adjacent to  $v$ , the corollary implies that  $l_v$  or  $r_v$  is a first vertex from  $v$  to every vertex in  $B_v$ . Since sequence  $r^1(v), r^2(v), \dots$  “runs” through the clique-cycle as implied by  $\mathcal{C}$  and sequence  $l^1(v), l^2(v), \dots$  “runs” through the clique-cycle as implied by  $\mathcal{C}^{-1}$ , that is to say, in the other direction, there exists an  $i \in \mathbb{N}$  such that  $l^i(v)$  and  $r^i(v)$  “meet”, or more formally, are adjacent or equal. This number is fixed in the following definition.

**Definition 3.14 (Apex Number)** *Let  $(X, \mathcal{C})$  be a clique-cycle for a strict circular-arc graph  $G = (V, E)$  without dominating vertices and without counter vertices,  $\mathcal{L}$  given by Listing 1, and  $v \in V$ . If we have  $\alpha(l^1(v)) = \beta(r^1(v))$  or  $[\alpha(v), \beta(v)]_{\mathcal{C}} \subset [\beta(r^1(v)), \alpha(l^1(v))]_{\mathcal{C}}$  we set  $i = 1$ , otherwise we define  $i$  as the smallest number greater than 1 such that  $l^i(v)$  and  $r^i(v)$  are adjacent or equal. We call integer  $i$  the apex number of  $v$ .*

The separate treatment for apex number  $i = 1$  in Definition 3.14 ensures that the arcs of  $l^i(v)$  and  $r^i(v)$  intersect/meet adverse to  $v$  on the circle and therefore the arcs of  $l^1(v), \dots, l^i(v)$ ,  $r^1(v), \dots, r^i(v)$ , and  $v$  to cover the entire circle. In particular, if  $l_v$  and  $r_v$  intersect in  $v$ 's arc they cannot cover the entire cycle, since they are not counter vertices, but the apex number would be defined as 1, if only the second part of the case distinction were in place.

By the definition, for apex number  $i > 2$  vertices  $r^i(v)$  and  $l^{i-1}(v)$  are adjacent if and only if  $l^i(v)$  and  $r^{i-1}(v)$  are adjacent. Furthermore, for  $2 \leq j, k < i$  vertices  $r^j(v)$  and  $l^k(v)$  are not adjacent.

The following Theorem 3.15 shows that there exists a vertex  $s_v \in B_v$  with  $l^i(v), r^i(v) \in \alpha(s_v)$ , where  $i$  is the apex number of  $v$ , such that  $r_v$  is a first vertex from  $v$  to every vertex in  $B_r = ]m_v, s_v]_{\mathcal{L}}$  and vertex  $l_v$  is a first vertex from  $v$  to every vertex in  $B_l = ]s_v, l_v]_{\mathcal{L}}$ . Since Theorem 3.15 implies that the vertices in  $B_v$  can be distributed by assigning  $B_l$  to directed edge  $(v, l_v)$  and  $B_r$  to directed edge  $(v, r_v)$ , the main theorem (Theorem 3.1) follows for the case in which neither dominating vertices nor counter vertices exist.

**Theorem 3.15** *Let  $(X, \mathcal{C})$  be a clique-cycle for a strict circular-arc graph  $G = (V, E)$  without dominating vertices and without counter vertices,  $\mathcal{L}$  given by Listing 1,  $v \in V$ ,  $l_v$  the left vertex of  $v$ ,  $m_v$  the middle vertex of  $v$ , and  $r_v$  the right vertex of  $v$ . There exists a vertex  $s_v \in ]m_v, l_v]_{\mathcal{L}}$  such that, for every vertex  $w \in ]m_v, s_v]_{\mathcal{L}}$ , we have  $r_v \in \mathcal{F}(v, w)$ , and for every vertex  $w' \in ]s_v, l_v]_{\mathcal{L}}$ , we have  $l_v \in \mathcal{F}(v, w')$ .*

**Proof** Let  $i$  the apex number of  $v$ .

If  $i = 1$  the theorem holds for  $s_v = \mathcal{L}^{-1}(l_v)$ , since every vertex in  $]m_v, l_v]_{\mathcal{L}}$  is adjacent to  $r_v$ .

If  $i > 1$  and  $l^i(v) = r^i(v)$ , we have

$$\forall w \in ]m_v, \mathcal{T}(\beta(r^{i-1}(v)))]_{\mathcal{L}}: \mathcal{R}_v(w) \leq i - 1 \leq \mathcal{L}_v(w)$$

and

$$\forall w \in ]\mathcal{T}(\beta(r^{i-1}(v))), l_v]_{\mathcal{L}}: \mathcal{L}_v(w) \leq \mathcal{R}_v(w),$$

wherefore, by Corollary 3.13, Theorem 3.15 holds for  $s_v = \mathcal{T}(\beta(r^{i-1}(v)))$ .

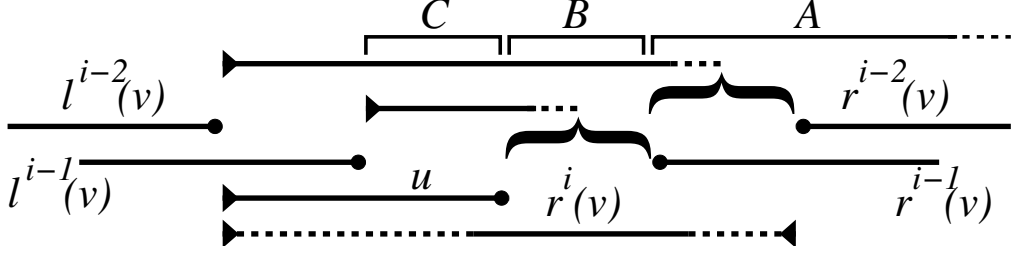


Figure 3: This picture illustrates the proof of Theorem 3.15 and partially shows a circular-arc graph, where the lines represent arcs. A line ends with a triangle, if it may reach not further to left or right, respectively. The upper two lines with a dotted end illustrate that an arc, that has its right end in the interval depicted by the rotated curly bracket, may reach no further to the left than the triangle of the respective line. We begin to search for  $u$  at the right end of the lower curly bracket. The rotated square brackets on top mark the areas of the left cliques of the vertices in sets  $A, B, C$ . The case in which  $l^{i-1}(v) = u$  looks similar.

The following argumentation is partially illustrated in Fig. 3.

If  $i > 1$  and  $l^i(v) \neq r^i(v)$ , let

$$A = ]m_v, \mathcal{T}(\beta(r^{i-1}(v)))[_{\mathcal{L}},$$

and  $a \in A$ . We have  $\mathcal{R}_v(a) \leq i-1$ , because  $a$  is adjacent to a vertex in  $\{r^1(v), r^2(v), \dots, r^{i-1}(v)\}$ . When  $l_v$  reaches further to the right than  $v$ ,  $a$  might appear in a clique in  $]\beta(v), \beta(l_v)]_{\mathcal{C}}$  and, therefore, be adjacent to  $l_v$ , which implies  $\mathcal{L}_v(a) = 1$ . However, in this case, we also have  $\mathcal{R}_v(a) = 1$ . Therefore, we could only have  $\mathcal{L}_v(a) < \mathcal{R}_v(a)$ , when  $a$  reaches so far to the right, such that it is adjacent to  $l^{i-2}(v)$ . However, in this case  $a$  would reach further to the left than  $l^{i-1}(v)$ , because  $a$  is adjacent to  $r^{i-1}(v)$  and  $l^{i-1}(v)$  is “at most” adjacent to  $r^i(v)$ . Because  $l^{i-1}(v)$  is the left vertex of  $l^{i-2}(v)$  this contradicts Theorem 3.6. Thus, Corollary 3.13 implies

$$\forall a \in A: r_v \in \mathcal{F}(v, a). \quad (2)$$

Now we go through  $\mathcal{L}$  beginning at  $\mathcal{L}(\mathcal{T}(\beta(r^{i-1}(v)))) = \mathcal{H}(\mathcal{C}(\beta(r^{i-1}(v))))$ ,<sup>9</sup> in other words, we begin at the first vertex after the “last” vertex in  $A$ , until we eventually come across a vertex  $u$  that either is  $l^{i-1}(v)$  or adjacent to  $l^{i-1}(v)$ .<sup>10</sup> We define

$$B = ]\mathcal{T}(\beta(r^{i-1}(v))), u[_{\mathcal{L}}$$

i.e.  $B$  is the set of vertices after  $\mathcal{T}(\beta(r^{i-1}(v)))$  and before  $u$  in  $\mathcal{L}$ . Note that, by the choice of  $u$ , no vertex in  $B$  is adjacent to  $l^{i-1}(v)$ .

Assume there is a vertex  $x \in B$  that is not adjacent to  $r^i(v)$ . When we go through  $\mathcal{C}$  beginning at  $\alpha(r^{i-1}(v))$  we eventually come across  $\beta(r^{i-1}(v))$  (since  $r^i(v)$  and  $r^{i-1}(v)$  are adjacent, we have  $\alpha(r^i(v)) \in [\alpha(r^{i-1}(v)), \beta(r^{i-1}(v))[_{\mathcal{C}}$ ). Before we come across  $\alpha(x)$ , we come across  $\beta(r^i(v))$ , since otherwise  $x$  and  $r^i(v)$  would be adjacent. We do not come across  $\alpha(l^{i-1}(v))$  until we came across  $\beta(x)$ , since otherwise  $x$  and  $l^{i-1}(v)$  would be adjacent. It follows that  $l^{i-1}(v)$  and  $r^i(v)$  are not adjacent and, because  $l^i(v)$  is adjacent to  $l^{i-1}(v)$  and  $r^i(v)$ , it follows that  $l^i(v)$  reaches further to the right than  $r^i(v)$ . Because  $l^i(v)$  and  $r^i(v)$  are adjacent,  $\alpha(l^i(v))$  must appear in  $[\alpha(r^i(v)), \beta(r^i(v))[_{\mathcal{C}}$ . It cannot appear in  $[\alpha(r^i(v)), \beta(r^{i-1}(v))[_{\mathcal{C}}$ , because then  $l^i(v)$  is adjacent to  $r^{i-1}(v)$ , which contradicts Definition 3.11, since  $l^i(v)$  reaches further to the right than  $r^i(v)$  and

<sup>8</sup>It follows that  $l^2(v)$  is the left vertex of the left vertex of  $v$  and  $r^2(v)$  is the right vertex of the right vertex of  $v$ , etc.

<sup>9</sup>This equality holds by Corollary 3.4.

<sup>10</sup>This vertex  $u$  may be  $l^i(v)$ . Also there may be vertices in  $A$  that are adjacent to  $l^{i-1}(v)$ , in other words, we may have “passed by” a vertex that is adjacent to  $l^{i-1}(v)$  already.

$r^i(v)$  is the right vertex of  $r^{i-1}(v)$ . Therefore, we have  $\alpha(l^i(v)) \in ]\beta(r^{i-1}(v)), \beta(r^i(v))[_{\mathcal{C}}$ . Because we came across  $\beta(r^i(v))$  before  $\alpha(x)$ , we have  $]\beta(r^{i-1}(v)), \beta(r^i(v))[_{\mathcal{C}} \subset ]\beta(r^{i-1}(v)), \alpha(x)[_{\mathcal{C}}$  and, therefore,

$$\alpha(l^i(v)) \in ]\beta(r^{i-1}(v)), \alpha(x)[_{\mathcal{C}}. \quad (3)$$

Since the vertices in  $\mathcal{L}$  are primary ordered by their left clique, Inclusion (3) implies

$$l^i(v) \in [\mathcal{H}(\mathcal{C}(\beta(r^{i-1}(v))))], x]_{\mathcal{L}}. \quad (4)$$

Remember that we began at  $\mathcal{H}(\mathcal{C}(\beta(r^{i-1}(v))))$  to go through  $\mathcal{L}$  until we found a vertex  $u$  that is adjacent to  $l^{i-1}(v)$ . Inclusion (4) implies that we must have come across  $l^i(v)$  before  $x$ . But then we have  $u = l^i(v)$ . Since

$$x \notin [\mathcal{H}(\mathcal{C}(\beta(r^{i-1}(v))))], l^i(v)[_{\mathcal{L}} = [\mathcal{L}(\mathcal{T}(\beta(r^{i-1}(v))))], u[_{\mathcal{L}} = B$$

contradicts the initial assumption, every vertex in  $B$  is adjacent to  $r^i(v)$ .

Since by the choice of  $u$  no vertex in  $B$  is adjacent to  $l^{i-1}(v)$ , we have

$$\forall b \in B: \mathcal{L}_v(b) \geq i = \mathcal{R}_v(b),$$

which, by Corollary 3.13, implies

$$\forall b \in B: r_v \in \mathcal{F}(v, b). \quad (5)$$

We define  $s_v = \mathcal{L}^{-1}(u)$ . Since the ring-intervals  $A$  and  $B$  appear consecutively in  $\mathcal{L}$ , by Construction 2.3, we have

$$A \cup B = ]m_v, u[_{\mathcal{L}} = ]m_v, s_v]_{\mathcal{L}}. \quad (6)$$

From (2), (5), and (6) it follows

$$\forall w \in ]m_v, s_v]: r_v \in \mathcal{F}(v, w). \quad (7)$$

Since the vertices in  $\mathcal{L}$  are primary ordered by their left cliques, no vertex

$$w \in [\mathcal{H}(\mathcal{C}(\beta(r^{i-1}(v))))], l^{i-1}(v)]_{\mathcal{L}}$$

is adjacent to  $r^{i-1}(v)$  and therefore we have  $\mathcal{R}_v(w) \geq i$ . We define

$$C = ]s_v, l^{i-1}(v)[_{\mathcal{L}}$$

i.e.  $C$  is the set of vertices after  $s_v$  and before  $l^{i-1}(v)$  in  $\mathcal{L}$ . Note that

$$B, C \subseteq [\mathcal{H}(\mathcal{C}(\beta(r^{i-1}(v))))], l^{i-1}(v)]_{\mathcal{L}}.$$

We have

$$\forall c \in [\alpha(\mathcal{L}(s_v)), \alpha(l^{i-1}(v))[_{\mathcal{C}}: l^i(v) \in c \quad (8)$$

since otherwise  $u = \mathcal{L}(s_v)$  would reach further to the left than  $l^i(v)$  although both are adjacent to  $l^{i-1}(v)$  and  $l^i(v)$  is the left vertex of  $l^{i-1}(v)$ . This would contradict Theorem 3.6. It follows

$$\forall c \in C: \mathcal{L}_v(c) \leq i \leq \mathcal{R}_v(c),$$

which, by Corollary 3.13, implies

$$\forall c \in C: l_v \in \mathcal{F}(v, c). \quad (9)$$

Obviously we have

$$\forall w \in [l^{i-1}(v), l_v[_{\mathcal{L}}: l_v \in \mathcal{F}(v, w). \quad (10)$$

By Construction 2.3, we have  $C \cup [l^{i-1}(v), l_v[_{\mathcal{L}} = ]s_v, l_v[_{\mathcal{L}}$  and therefore (9) and (10) imply

$$\forall w \in ]s_v, l_v[_{\mathcal{L}}: l_v \in \mathcal{F}(v, w). \quad (11)$$

The statement follows from (7) and (11).  $\square$

In Section 3.3.1 we assigned  $[l_v, x]_{\mathcal{L}}$  to directed edge  $(v, l_v)$  for some vertex  $x$  to the left of  $v$ . If  $]s_v, l_v[_{\mathcal{L}}$  is assigned to directed edge  $(v, l_v)$  additionally (as implied by Theorem 3.15) the intervals on directed edge  $(v, l_v)$  can be compressed. Therefore, if neither dominating vertices nor counter vertices exist,  $v$  given  $\mathcal{L}$  suffices a shortest path 2-SIRS, but only  $(v, r_v)$  is assigned two ring-intervals.

Below Definition 3.11 we stated that  $v$  given  $\mathcal{L}$  suffices a shortest path 1-SIRS, if we can choose  $r_v = l_v$  or  $r_v = m_v$ . This is the case because then the ring-intervals assigned to directed edge  $(v, r_v)$  can be compressed as well.

Theorem 3.16 covers a further case where  $v$  given  $\mathcal{L}$  suffices a shortest path 1-SIRS. Since Theorem 3.16 is not essential for the proof of Theorem 3.1, we leave the proof to the interested reader.

**Theorem 3.16** *Let  $(X, \mathcal{C})$  be a clique-cycle for a circular-arc graph  $G = (V, E)$ ,  $\mathcal{L}$  given by Listing 1,  $v \in V$  not a dominating vertex,  $l_v$  the left vertex of  $v$ , and  $m_v$  the middle vertex of  $v$ . If  $m_v$  is adjacent to every vertex face-to-face with  $v$  that  $l_v$  is not adjacent to,  $v$  suffices a shortest path 1-SIRS.*

**3.3.2.2 Dominating Vertices or Counter Vertices exist** In the foregoing section the existence of dominating and counter vertices was excluded and is therefore discussed in this section. The vertices to be distributed are those face-to-face with  $v$ , i.e., the vertices in  $B_v$ . By Theorem 3.9, the only vertices in  $B_v$  that are adjacent to  $v$  are dominating vertices; this case is considered first. Next, the case where dominating vertices not face-to-face with  $v$  or counter vertices of  $v$  exist is addressed. Finally, the case where a pair of counter vertices exists (but  $v$  has no counter vertex) is discussed.

**Dominating Face-to-Face Vertices exist** Listing 1 ensures that all dominating vertices appear consecutively in  $\mathcal{L}$ , wherefore we can find two dominating vertices  $d_l$  and  $d_r$  such that  $[d_l, d_r]_{\mathcal{L}} = \mathcal{D}(G)$ . Let  $x \in V$  such that  $]m_v, x]_{\mathcal{L}} = B_v$ . Since by Theorem 3.9,  $\mathcal{D}(G)$  are the only vertices in  $B_v$  that are adjacent to  $v$ , we can assign  $]m_v, d_l]_{\mathcal{L}}$  to directed edge  $(v, d_l)$ ,  $[d_r, x]_{\mathcal{L}}$  to directed edge  $(v, d_r)$ , and for the remaining dominating vertices  $d_i \in ]d_l, d_r[_{\mathcal{L}}$   $[d_i, d_i]_{\mathcal{L}}$  to directed edge  $(v, d_i)$ . It follows that  $v$  given  $\mathcal{L}$  suffices a shortest path 1-SIRS.

**Dominating Face-to-Face Vertices do not exist** If there are no dominating vertices in  $B_v$ , Theorem 3.9 implies that no vertex in  $B_v$  is adjacent to  $v$ . Therefore, by Corollary 2.7, every vertex that is a counter vertex of  $v$  is adjacent to every vertex in  $B_v$ . Let  $u \in \mathcal{C}(v) \cup \mathcal{D}(G)$  and  $w \in B_v$ . Then  $u$  is a first vertex from  $v$  to  $w$ , wherefore we can assign ring-interval  $B_v$  to directed edge  $(v, u)$ , which had hitherto one ring-interval assigned. Thus,  $v$  given  $\mathcal{L}$  suffices a shortest path 2-SIRS, wherein only directed edge  $(v, u)$  has two ring-intervals assigned. By Theorem 3.16,  $v$  given  $\mathcal{L}$  suffices a shortest path 1-SIRS, if  $m_v \in \mathcal{C}(v) \cup \mathcal{D}(G)$ .

**Pairs of Counter Vertices Exist but  $v$  is in none of them** Assume  $G$  does not contain dominating vertices and the considered vertex  $v$  has no counter vertices but there exists at least one pair  $(w, c_w)$  of counter vertices. We have  $B_v = ]m_v, l_v[_{\mathcal{L}}$  and no vertex in  $B_v$  is adjacent to  $v$ . We make a distinction of cases for whether  $v$  is adjacent to one or to both counter vertices.

Assume  $v$  is adjacent to  $w$  as well as  $c_w$ . There is a clique  $c \in X$  that contains  $v$ ,  $w$ , and  $c_w$ . When we begin at  $c$  to go through  $\mathcal{C}$ , we eventually reach a clique  $d$  that contains only one of  $w$  or  $c_w$  and not the other. Without loss of generality, let  $w$  be contained in  $d$  and let  $e$  be the first clique that contains  $c_w$  again.  $e$  is the left clique of  $c_w$  and contains  $l_v$  (we might have  $c_w = l_v$ ), as otherwise,  $c_w$  would reach further to the left than  $l_v$ , which contradicts Theorem 3.6. Since  $w$  and  $c_w$  are counter vertices, all cliques we came across contained  $w$ . Since the vertices in  $\mathcal{L}$  are primary ordered by their left cliques and we also must have traversed  $\beta(v)$ , which contains  $m_v$ , every vertex in  $B_v = ]m_v, l_v[_{\mathcal{L}}$  appeared in at least one of the cliques we came across.

Therefore,  $w$  is adjacent to every vertex in  $B_v$ . Since  $\mathcal{I}_v$  hitherto maps directed edge  $(v, w)$  to one ring-interval, we can assign  $B_v$  to directed edge  $(v, w)$ . Thus,  $v$  given  $\mathcal{L}$  suffices a shortest path 2-SIRS, wherein only directed edge  $(v, w)$  is actually mapped to two ring-intervals.

Now assume  $v$  is adjacent to one of the counter vertices and not to the other. Without loss of generality, let this vertex be  $w$ . Let  $r$  be a vertex adjacent to  $v$  that, similar to Definition 3.11, reaches farthest to the right. Since, by Theorem 3.6,  $l_v$  is one of the vertices adjacent to  $v$  that reaches farthest to the left,  $l_v$  as well as  $r$  are adjacent to  $c_w$ . In fact, we might have  $w = l_v$  or  $w = r$ . Let  $m_r$  be the middle vertex of  $r$  and  $A_r = ]r, m_r[_{\mathcal{L}}$  the set of vertices to the right of  $r$ .  $A_r$  is not empty, since otherwise  $G$  would be an interval graph. Also  $m_v$  is contained in  $A_r$ , if  $r \neq m_v$ . By Theorem 3.9, every vertex in  $A_r$  is adjacent to  $r$ . Consider we go through  $\mathcal{L}$  beginning at  $\mathcal{L}(m_v)$  until we reach  $m_r$  or  $l_v$ . If we have  $l_v = m_r$ , we consider it as reaching  $l_v$ . If we first reach  $l_v$ , we exactly traversed the vertices in  $]m_v, l_v[_{\mathcal{L}} \subseteq B_v$ . It follows that we have  $B_v \subseteq A_r$  and thus assign  $B_v = ]m_v, l_v[_{\mathcal{L}}$  to directed edge  $(v, r)$ . Now assume we first reach  $m_r$ . By Theorem 3.9, every vertex in  $]m_v, m_r[_{\mathcal{L}} \subseteq A_r$  is adjacent to  $r$ , wherefore we can assign  $]m_v, m_r[_{\mathcal{L}}$  to directed edge  $(v, r)$ . If  $\mathcal{L}(m_r) = l_v$ , all vertices in  $B_v$  are distributed. Otherwise let  $X = ]m_r, l_v[_{\mathcal{L}} \subset B_v$  be the vertices left to be distributed. Since, by Theorem 3.6,  $l_v$  is a vertex adjacent to  $v$  that reaches farthest to the left and  $l_v$  is adjacent to  $c_w$ , for every vertex  $u \in X$  either  $(v, l_v, u)$  or  $(v, l_v, c_w, u)$  is a shortest path. Thus, we can assign  $X$  to directed edge  $(v, l_v)$  (and compress the two intervals on this edge), wherefore all vertices are distributed and only directed edge  $(v, r)$  is assigned two ring intervals.

## 4 Improving the Space Requirement

This section shows that although circular-arc graphs do not allow shortest path 1-IRSs in general, the number of intervals in a shortest path 2-SIRS for a circular-arc graph is bounded close to the number of intervals in an 1-IRS. Let  $G_d = (V, A)$  be the directed symmetric version of a strict circular-arc graph. A shortest path 2-SIRS for  $G_d$  can be obtained by ordering the vertices of  $G_d$  according to Listing 1 and then labeling the outgoing directed edges of every vertex according to Section 3.3.1 and 3.3.2. As pointed out in these sections, there is at most one outgoing edge per vertex that is assigned two intervals. We have  $|V| \leq |A|/2$ , since  $G$  is a strict circular-arc graph, and  $|V| = |A|/2$  only holds when  $G$  is a ring in which case our labeling yields a shortest path 1-SIRS. Therefore, the number of intervals in the constructed 2-SIRS is less than  $|A| + |V| = 1.5 \cdot |A|$ , while already a 1-IRS for  $G_d$  permits up to  $|A|$  intervals.

## 5 Implementation

This section outlines how the interval routing scheme implied by the proof of Theorem 3.1 can be implemented. Let  $G = (V, E)$  be a circular-arc graph with  $|V| = n$  and  $|E| = m$  and  $(\mathbf{X}, \mathcal{C})$  be a clique-cycle for  $G$  with  $|\mathbf{X}| = c$  maximal cliques ( $\mathbf{X}$  does not need to contain sets of vertices; plain elements representing the cliques are sufficient). We discuss how the vertex order  $\mathcal{L}$  and the critical vertices  $l_v, m_v, r_v, s_v$  for every vertex  $v \in V$  can be determined in  $\mathcal{O}(c + n^2)$  time. We assume that every vertex  $v \in V$  has a pointer  $\alpha(v)$  to its left clique and a pointer  $\beta(v)$  to its right clique in  $\mathbf{X}$ .

We define *broadness of a vertex  $v$*  as the number of cliques in  $\mathbf{X}$  that contain  $v$ . By numbering the elements in  $\mathbf{X}$  as they appear in  $\mathcal{C}$ , beginning at an arbitrary element, we construct a bijective mapping  $\mathcal{N}: \mathbf{X} \rightarrow \{1, 2, \dots, |\mathbf{X}|\}$ , which allows to compute the broadness of a vertex in  $\mathcal{O}(1)$  time. Sorting the vertices ascending by their broadness requires  $\mathcal{O}(n \cdot \log(n))$  time. We introduce a list  $\mathcal{Q}(c)$  for every clique  $c \in \mathbf{X}$ . Next the vertices are traversed in the sorted order and every vertex  $v$  is added to list  $\mathcal{Q}(\alpha(v))$ . By concatenating the lists as their respective cliques appear in  $\mathcal{C}$  we obtain the vertex order  $\mathcal{L}$  in  $\mathcal{O}(n \cdot \log(n))$  time. The left vertex  $l_v$  of a given vertex  $v \in V$  can be determined as follows: we inspect all vertices that are adjacent to  $v$  and store the vertex  $w$  that reaches farthest to the left (for two adjacent vertices  $w, w'$ , we can determine in constant

time which one reaches further to the left, by considering  $\mathcal{N}(\alpha(w))$  and  $\mathcal{N}(\alpha(w'))$  and appears first in  $\mathcal{Q}(\alpha(w))$  (this can also be decided in constant time, by numbering the elements in each list consecutively). Vertex  $w$  must be  $l_v$ . It follows that the set of left vertices can be computed in  $\mathcal{O}(m)$  time. The set of right vertices can be computed in similarly. Determining the set of middle vertices is simpler and can be accomplished in  $\mathcal{O}(n)$ . For a given vertex  $v$ ,  $l^i(v)$  and  $r^i(v)$  can be determined by traversing the sequences  $l^1(v), l^2(v), \dots$  and  $r^1(v), r^2(v), \dots$  synchronously; then the determination of  $s_v$  is a simple task and can be done in  $\mathcal{O}(n)$  for a specific vertex and in  $\mathcal{O}(n^2)$  for all vertices.

Since the algorithm includes constant time operations on the elements in  $\mathbf{X}$  and  $|\mathbf{X}| = c$  is linear within the number of vertices, the overall runtime is  $\mathcal{O}(c + n^2 + m) = \mathcal{O}(n^2)$ . The algorithm can be extended to handle the existence of dominating and counter vertices without exceeding this time bound.

## 6 Conclusions

We showed that the class of circular-arc graphs has strict compactness 2. Throughout the proof special cases of strict circular-arc graphs that allow 1-SIRSs were highlighted (not all cases were pointed out for the sake of brevity), which is in particular interesting, since the class of circular-arc graphs is a super-class of the class of interval and unit circular-arc graphs, which always allow 1-SIRSs. We showed that the constructed 2-SIRS requires a number of intervals that is closer to the maximal number of intervals in an 1-IRS than in a 2-IRS. An open question is, if strict circular-arc graphs exist that allow 1-SIRS but the vertex ordering generated by Listing 1 does not allow these.

## References

- [BK97] Binay K. Bhattacharya and Damon Kaller. An  $\mathcal{O}(m + n \cdot \log(n))$  algorithm for the maximum-clique problem in circular-arc graphs. *Journal of Algorithms*, 25(2):336–358, 1997.
- [BLS99] Andreas Brandstädt, Van Bang Le, and Jereny P. Spinrad. *Graph Classes: A Survey*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1999.
- [BvLT91] Erwin M. Bakker, Jan van Leeuwen, and Richard B. Tan. Linear interval routing. *The Computer Journal*, 30:298–307, 1991.
- [CFZ08] Gabrio Caimi, Martin Fuchsberger, and Rico Zenklusen. About maximal cliques in a certain class of circular-arc graphs. Working paper, Institut für Operations Research, ETH Zurich, CH-8092 Zürich, 2008.
- [DL02] Feodor F. Dragan and Irina Lomonosov. New routing schemes for interval, circular-arc, and permutation graphs. In *IASTED PDCS'02*, pages 78–83, 2002.
- [DYL06] Feodor F. Dragan, Chenyu Yan, and Irina Lomonosov. Collective tree spanners of graphs. *SIAM Journal on Discrete Mathematics*, 20(1):241–260, 2006.
- [EMZ02] Tamar Eilam, Shlomo Moran, and Shmuel Zaks. The complexity of the characterization of networks supporting shortest-path interval routing. *Theor. Comput. Sci.*, 289:85–104, 2002.
- [FG98] Pierre Fraigniaud and Cyril Gavoille. Interval routing schemes. *Algorithmica*, 21(2):155–182, 1998.
- [FJ88] Greg N. Frederickson and Ravi Janardan. Designing networks with compact routing tables. *Algorithmica*, 3:171–190, 1988.



- [Fla97] Michele Flammini. On the hardness of devising interval routing schemes. *Parallel Process. Lett*, 7(1):39–47, 1997.
- [Gav00] Cyril Gavoille. A survey on interval routing. *Theoretical Computer Science*, 245:785–796, 2000.
- [GG98] Eric Guevremont and Cyril Gavoille. Worst case bounds for shortest path interval routing. *Journal of Algorithms*, 27:1–25, 1998.
- [Gol04] Martin Charles Golumbic. *Algorithmic Graph Theory and Perfect Graphs (Annals of Discrete Mathematics, Vol 57)*. North-Holland Publishing Co., Amsterdam, The Netherlands, 2004.
- [GP98] Cyril Gavoille and David Peleg. The compactness of interval routing for almost all graphs. In *Proceedings of Internat. Symp. on Distributed Computing (DISC)*, volume 1499 of *Lecture Notes in Computer Science*, pages 161–174. Springer Verlag, 1998.
- [GP08] Cyril Gavoille and Christophe Paul. Optimal distance labeling for interval graphs and related graph families. *SIAM Journal on Discrete Mathematics*, 22(3):1239–1258, 2008.
- [Hsu95] Wen-Lian Hsu.  $\mathcal{O}(m \cdot n)$  algorithms for the recognition and isomorphism problems on circular-arc graphs. *SIAM Journal of Computing*, 24:411–439, 1995.
- [LS09] Min Chih Lin and Jayme L. Szwarcfiter. Characterizations and recognition of circular-arc graphs and subclasses: A survey. *Discrete Mathematics*, 309(18):5618–5635, 2009.
- [McC03] Ross M. McConnell. Linear-time recognition of circular-arc graphs. *Algorithmica*, 37:93–147, 2003.
- [NS96] Lata Narayanan and Sunil Shende. Characterizations of networks supporting shortest-path interval labeling schemes. *3rd Internat. Coll. on Structural Information and Communication Complexity (SIROCCO)*, pages 73–87, 1996.
- [PU89] David Peleg and Eli Upfal. A trade-off between space and efficiency for routing tables. *Journal of the ACM*, 36(3):510–530, 1989.
- [Rob76] Fred S. Roberts. *Discrete mathematical models, with applications to social, biological, and environmental problems*. Prentice-Hall, Englewood Cliffs, N.J., 1976.
- [SK82] Nicola Santoro and Ramez Khatib. Routing without routing tables. Technical Report SCS-TR-6 School of Computer Science, Carleton University, Ottawa, 1982.
- [SK85] Nicola Santoro and Ramez Khatib. Labelling and implicit routing in networks. *The Computer Journal*, 28(1):5–8, 1985.
- [TL99] Savio S.H. Tse and Francis C.M. Lau. On the space requirement of interval routing. *IEEE Transactions on Computers*, 48:752–757, 1999.
- [Tuc75] Alan Tucker. Coloring a family of circular arcs. *SIAM Journal on Applied Mathematics*, 29(3):493–502, 1975.
- [Tuc80] Alan Tucker. An efficient test for circular-arc graphs. *SIAM Journal of Computing*, 9:1–25, 1980.
- [vLT87] Jan van Leeuwen and Richard B. Tan. Interval routing. *The Computer Journal*, 30(4):298–307, 1987.