Computing and counting longest paths on circular-arc graphs in polynomial time

George B. Mertzios\textsuperscript{a,}\textsuperscript{*}, Ivona Bezáková\textsuperscript{b}

\textsuperscript{a} School of Engineering and Computing Sciences, Durham University, Durham, United Kingdom
\textsuperscript{b} Department of Computer Science, Rochester Institute of Technology, Rochester, NY, USA

\textbf{Abstract}

The longest path problem asks for a path with the largest number of vertices in a given graph. In contrast to the Hamiltonian path problem, until recently polynomial algorithms for the longest path problem were known only for small graph classes, such as trees. Recently, a polynomial algorithm for this problem on interval graphs has been presented in Ioannidou et al. (2011) \cite{19} with running time \(O(n^4)\) on a graph with \(n\) vertices, thus answering the open question posed in Uehara and Uno (2004) \cite{32}. Even though interval and circular-arc graphs look superficially similar, they differ substantially, as circular-arc graphs are not perfect; for instance, several problems – e.g. coloring – are NP-hard on circular-arc graphs, although they can be efficiently solved on interval graphs. In this paper, we prove that for every path \(P\) of a circular-arc graph \(G\), we can appropriately ”cut” the circle, such that the obtained (not induced) interval subgraph \(G'\) of \(G\) admits a path \(P'\) on the same vertices as \(P\). This non-trivial result is of independent interest, as it suggests a generic reduction of a number of path problems on circular-arc graphs to the case of interval graphs with a multiplicative linear time overhead of \(O(n)\). As an application of this reduction, we present the first polynomial algorithm for the longest path problem on circular-arc graphs. In addition, by exploiting deeper the structure of circular-arc graphs, we manage to get rid of the linear time overhead of the reduction, and thus this algorithm turns out to have the same running time \(O(n^4)\) as the one on interval graphs. Our algorithm, which significantly simplifies the approach of Ioannidou et al. (2011) \cite{19}, computes in the same time an \(n\)-approximation of the (exponentially large in worst case) number of different vertex sets that provide a longest path; in the case where \(G\) is an interval graph, we compute the exact number. Moreover, in contrast to Ioannidou et al. (2011) \cite{19}, this algorithm can be directly extended with the same running time to the case where every vertex has an arbitrary positive weight.

© 2012 Elsevier B.V. All rights reserved.

\textbf{1. Introduction}

The Hamiltonian path problem, i.e. the problem of deciding whether a given graph contains a simple path that visits all its vertices, is one of the most well known and well studied NP-complete problems \cite{13}, with numerous applications. The most natural optimization version of this problem is the longest path problem, where the task is to find a path with the largest number of vertices. This problem has been also extensively studied over the past several decades and it plays an important role in a number of applications, for instance in computational biology \cite{29,7}.

\textsuperscript{*} Corresponding author. Tel.: +44 1913342429; fax: +44 1913341701.
E-mail addresses: george.mertzios@gmail.com, george.mertzios@durham.ac.uk (G.B. Mertzios), ib@cs.rit.edu (I. Bezáková).

0166-218X/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.dam.2012.08.024
In addition to both problems being NP-hard for general graphs, several prohibitive inapproximability results for the longest path problem appeared in [22]. In particular, for any \( \varepsilon \in (0, 1) \), it is NP-hard to compute a path of length \( n - n^\varepsilon \) in a graph with \( n \) vertices, even if it is known that the graph admits a Hamiltonian path. Moreover, there is no polynomial time constant-factor approximation algorithm for the longest path problem unless \( P = \text{NP} \) [22]. To the best of our knowledge, the best known approximation algorithms achieve approximation ratio \( O(n(\log \log n / \log n)^2) \) for general graphs [3]. Furthermore, the Hamiltonian path (and thus also the longest path) problem is NP-hard on many restricted classes of graphs, namely split graphs, chordal bipartite graphs, split strongly chordal graphs, directed path graphs, circle graphs, planar graphs, and grid graphs; see e.g. [25] for a list of related works.

On the positive side, polynomial time algorithms have been developed for the Hamiltonian path (and the related Hamiltonian cycle) problem on several graph classes, notably proper interval graphs [2], interval graphs [1,23], circular-arc graphs [8,30,18], and cocomparability graphs [9,10]. Another natural generalization of the Hamiltonian path problem, namely the minimum path cover problem, has also received considerable attention in the literature. The objective is to find the smallest number of vertex disjoint simple paths that cover all the vertices; polynomial time algorithms were given for this problem on interval graphs [1,6], cocomparability graphs [9], and lately also on circular-arc graphs [17,18].

In contrast to the Hamiltonian path problem, until recently only a few polynomial algorithms were known for the longest path problem, and these were restricted to trees [5], weighted trees and block graphs [32], bipartite permutation graphs [33], and ptolemaic graphs [31]. Very recently, prompted by an open problem statement in [32], a polynomial time algorithm has been developed for interval graphs with running time \( O(n^3) \) on a graph with \( n \) vertices [19]. This algorithm has been followed by two independent polynomial algorithms for the longest path problem on the much greater class of cocomparability graphs (one with running time \( O(n^4) \) [26] and one with running time \( O(n^5) \) [20]).

Circular-arc graphs naturally extend interval graphs: interval graphs are the intersection graphs of intervals on the real line, while circular-arc graphs are intersection graphs of arcs on a circle. That is, a graph \( G \) is interval (resp. circular-arc) if its vertices can be put in a one-to-one correspondence with a family of intervals (resp. arcs) on the real line (resp. on the circle), such that two vertices are adjacent in \( G \) if and only if their corresponding intervals (resp. arcs) intersect. Such an intersection model with intervals (resp. arcs) of an interval (resp. circular-arc) graph \( G \) is also called an interval (resp. circular-arc) representation of \( G \). Several NP-complete problems have been studied on these graph classes, for example, a maximum independent set and a maximum clique be found in polynomial time [16], while, for example, the achromatic number problem is NP-complete for both classes of graphs [4].

Although circular-arc graphs look superficially similar to interval graphs, several combinatorial problems behave very differently on these classes of graphs. For example, the coloring problem is NP-complete for circular-arc graphs [14] while it can be solved greedily in linear time on interval graphs.

All optimization problems have a corresponding counting version. For the case of Hamiltonian paths, the counting version asks for the overall number of all Hamiltonian paths in a given graph. Counting problems are related to sampling [21], where, for example, for the case of Hamiltonian paths, the task is to sample one of the Hamiltonian paths uniformly at random. Sampling plays an important role in machine learning and other applied areas, and the problems of counting and sampling of paths in graphs have been studied extensively; see e.g. [24]. The problem of counting all Hamiltonian paths is \#P-complete for general graphs [11], while approximation algorithms were given for several special classes of graphs, including dense graphs for which there exists a fully polynomial randomized approximation scheme (FPRAS) [11], and nearly regular graphs [12]. An importance sampling based framework, combined with cross and minimumentropy methods, achieved fast empirical results, closely approximating the optimum [28]. The problem of counting and sampling paths in graphs, especially the scenario of self-avoiding walks in lattice graphs, has been researched since the 1960s; see e.g. [24]. Most of the algorithms are heuristic without proofs of correctness; notable exceptions include [27].

Our contribution. In this article we present the first polynomial algorithm for the longest path problem on circular-arc graphs by showing that the problem reduces to the case of interval graphs. The significance of our reduction comes from the fact that a path in a circular-arc graph can have a spiral-like form and this makes it hard to "cut" the circle to create an interval graph that maintains the length of a longest path. Note here that also other problems on circular arc graphs have been reduced to the interval graph case. However, for problems that search for a set (e.g. an independent set) the reduction is fairly natural, since "cutting" the circle does not destroy the set. On the other hand, "cutting" a sequence (such as a path) breaks the sequence into many parts. In this article we overcome this issue by showing that for every path \( P \) of a circular-arc graph \( G \), we can appropriately "cut" the circle, such that the obtained (not necessarily induced) interval subgraph admits a path \( P' \) on the same vertices as \( P \).

This result suggests a generic reduction of a number of path problems (such as the Hamiltonian and the longest path problems) on circular-arc graphs to the corresponding problem on interval graphs with a multiplicative linear time overhead of \( O(n) \). However, by exploiting deeper the structure of circular-arc graphs, we manage to get rid of this overhead for the longest path problem. In particular, we introduce the crucial notion of normal paths in circular-arc graphs, which can be thought of as "monotone representatives" of all paths. Indeed, we prove that every path \( P \) of a circular-arc graph \( G \) can be restructured as a normal path on the same vertices.

Our dynamic programming algorithm searches for a longest normal path in a circular-arc graph and it runs in time \( O(n^4) \) on a graph with \( n \) vertices. Our algorithm significantly simplifies the approach of [19] that shows polynomial time solvability of this problem on interval graphs. This simplification consists of the elimination of the introduced "dummy vertices" that
were essential in [19] (the algorithm of [19] has three phases, during which it adds these dummy vertices to construct a second auxiliary graph).

By getting rid of these dummy vertices, our algorithm computes in the same time bound also the total number of different longest normal paths in the given circular-arc representation. This number constitutes an \( n \)-approximation of the (exponentially large in the worst case) number of different vertex sets that provide a longest path of \( G \). Moreover, in the case where \( G \) is an interval graph, we are able to compute within the same time bound the exact number of vertex sets of \( G \) that give a longest path. Moreover, in contrast to [19] where the introduced “dummy vertices” played a crucial role in the algorithm, all the above results can be directly extended (with the same running time as well) to the weighted case, i.e. to the case where a positive weight is assigned to every vertex of the input graph. However, for simplicity of the presentation, we present here only the unweighted case.

An interesting problem for further research is to exactly compute the number of all different vertex sets of a circular-arc graph that provide a longest path, as well as the number of all different longest paths (as opposed to their “representatives”, i.e. normal paths). It is noted that the same problems remain open also on the class of Helly circular-arc graphs, which lie strictly between interval graphs and circular-arc graphs.

Organization of the paper. Formal definitions, notation, and other preliminaries are introduced in Section 2. Section 3 describes the reduction from the problem on circular-arc graphs to interval graphs. The final algorithm and its analysis are presented in Section 4.

2. Notation and preliminaries

In this article we follow standard notation and terminology; see for instance [15]. We consider simple undirected graphs with no loops or multiple edges. In a graph \( G = (V, E) \), the edge between vertices \( u \) and \( v \) is denoted by \( uv \), and in this case \( u \) and \( v \) are said to be adjacent in \( G \), or equivalently, vertex \( u \) sees vertex \( v \). Let \( S \subseteq V \) be a set of vertices of a graph \( G = (V, E) \). Then, the cardinality of the set \( S \) is denoted by \( |S| \) and the subgraph of \( G \) induced by \( S \) is denoted by \( G[S] \).

Furthermore, let \( S' \subseteq S \) and \( \sigma \) be some ordering of the vertices of \( S \). Then, we denote by \( \sigma \setminus S' \) the ordering of the vertices of \( S \setminus S' \) that is obtained from \( \sigma \) after removing all vertices of \( S' \). The set \( N(v) = \{ u \in V \mid uv \in E \} \) is called the neighborhood of the vertex \( v \in V \) in \( G \), sometimes denoted by \( N_G(v) \) for clarity reasons.

A simple path \( P \) of a graph \( G \) is a sequence of distinct vertices \( v_1, v_2, \ldots, v_k \) such that \( v_i v_{i+1} \in E \) for each \( i \in \{1, 2, \ldots, k-1\} \), and is denoted by \( P = (v_1, v_2, \ldots, v_k) \); throughout the paper all paths considered are simple. Furthermore, \( v_1 \) (resp. \( v_k \)) is called the first (resp. last) vertex of \( P \). Given a path \( P = (v_1, v_2, \ldots, v_k) \) of a graph \( G \), the reverse path of \( P \) is the path \( \overline{P} = (v_k, v_{k-1}, \ldots, v_1) \) of \( G \). If \( P \) has at least two vertices, we consider the paths \( P \) and \( \overline{P} \) as two different paths.

We denote by \( V(P) \) the set of vertices of the path \( P \), and define the length \( |P| \) of \( P \) to be the number of vertices in \( P \), i.e. \( |P| = |V(P)| \). Additionally, if \( P = (v_1, v_2, \ldots, v_{i-1}, v_i, v_{i+1}, \ldots, v_k) \) is a path of a graph and \( P_0 = (v_j, \ldots, v_l) \) is a subpath of \( P \), we sometimes equivalently use the notation \( P = (v_1, v_2, \ldots, v_{i-1}, P_0, v_{i+1}, \ldots, v_k) \).

Given a circular-arc (resp. interval) graph \( G = (V, E) \) along with a circular-arc (resp. interval) representation \( R \) of \( G \), we may not distinguish in the following between a vertex \( v \) of \( G \) and the corresponding arc (resp. interval) \( l_v \) in \( R \), whenever it is clear from the context. Furthermore, by possibly performing a small shift of the endpoints of the arcs (resp. intervals), we may assume without loss of generality that all endpoints of the arcs (resp. intervals) are distinct [15].

We will denote the arc \( l_v \) of a vertex \( v \) with endpoints \( \ell_v \) and \( r_v \) by \( l_v = [\ell_v, r_v] \). We always consider the arcs in the counter-clockwise direction. That is, \( \ell_v \) (resp. \( r_v \)) is the first (resp. last) point of \( [\ell_v, r_v] \) (also referred to as the left and right endpoint of \( [\ell_v, r_v] \), respectively) when traveled in the counter-clockwise direction. Using this notation, the next two observations follow easily.

**Observation 1.** For every two distinct points \( a \) and \( b \) on a circle, the arcs \([a, b]\) and \([b, a]\) of the circle are different, while \([a, b] \cup [b, a] \) covers the whole circle.

**Observation 2.** Let \( G = (V, E) \) be a circular-arc graph, \( R \) be a circular-arc representation of \( G \), and \( u, v \in V \) such that \( uv \in E \). Then \( r_u \in l_v \) or \( r_v \in l_u \) (or both).

Given an interval graph \( G = (V, E) \) along with an interval representation \( R \) of \( G \), we can define an ordering of the set \( V \) by sorting the intervals in \( R \) according to their right endpoints. Such an ordering of the vertices of an interval graph \( G \), which is called a right-end ordering of \( G \), has been proved useful for a number of problems on interval graphs (see e.g. [19, 11]). In a right-end ordering \( \pi \) of an interval graph \( G = (V, E) \), we can define a total order \( <_\pi \) in \( V \) as follows: \( u <_\pi v \) for two vertices \( u, v \in V \) if \( u \) appears to the left of \( v \) in \( \pi \).

In a similar fashion, we consider in the following the right-end circular ordering \( \pi = (u_0, u_1, \ldots, u_{n-1}) \) of the set \( V \) of vertices of a circular-arc graph \( G = (V, E) \), which results after sorting the arcs of a circular-arc representation \( R \) of \( G \) according to their right endpoints. It is easy to see by definition that the notion of a right-end circular ordering in a circular-arc graph extends in a natural way the notion of a right-end ordering in an interval graph. However, in contrast to interval graphs, we cannot define a total order \( <_\pi \) on the vertices of \( V \), since there are two ways to travel from one point to another on a circle.
Definition 1. Let $G = (V, E)$ be a circular-arc graph and $R$ be a circular-arc representation of $G$. Let $P = (v_1, v_2, \ldots, v_k)$ be a path of $G$. For an arbitrary $i \in \{1, 2, \ldots, k-1\}$,

- if $r_{v_{i+1}} \not\in I_{v_i}$ in $R$ (in this case $r_{v_i} \in I_{v_{i+1}}$), then the path-arc of $P$ between $v_i$ and $v_{i+1}$ is $P(v_i, v_{i+1}) = [r_{v_i}, r_{v_{i+1}}]$; such a path-arc is called right-going,

- if $r_{v_i} \not\in I_{v_{i+1}}$ in $R$ (in this case $r_{v_{i+1}} \in I_{v_i}$), then the path-arc of $P$ between $v_i$ and $v_{i+1}$ is $P(v_i, v_{i+1}) = [r_{v_{i+1}}, r_{v_i}]$; such a path-arc is called left-going,

- if $r_{v_i} \in I_{v_{i+1}}$ and $r_{v_{i+1}} \in I_{v_i}$ in $R$, then the path-arc of $P$ between $v_i$ and $v_{i+1}$ is either $P(v_i, v_{i+1}) = [r_{v_i}, r_{v_{i+1}}]$ or $P(v_i, v_{i+1}) = [r_{v_{i+1}}, r_{v_i}]$; such a path-arc is called right-going or left-going, respectively.

The set of all path-arcs $P(v_i, v_{i+1})$ for every $i \in \{1, 2, \ldots, k-1\}$ is the path-arc representation $P_{arc}$ of $P$.

Note that, given a circular-arc representation $R$ of a circular-arc graph $G$, a path-arc representation $P_{arc}$ uniquely determines a path $P$ of $G$. However, a path $P$ of $G$ may have by Definition 1 more than one path-arc representation $P_{arc}$, since some edges of $P$ may be represented by a right-going or by a left-going path-arc in $P_{arc}$. As an example, a circular-arc graph $G$ is depicted in Fig. 1(a). For the path $P = (u_1, u_2, u_3, u_4, u_5, u_6, u_7)$ of $G$, two different path-arc representations of $P$ are illustrated in Fig. 1(b) and (c); note that the path-arc representation of Fig. 1(b) covers the whole circle, while that of Fig. 1(c) does not. The next two observations follow by Definition 1.

Observation 3. Let $G = (V, E)$ be a circular-arc graph, $R$ be a circular-arc representation of $G$, and $P = (v_1, v_2, \ldots, v_k)$ be a path of $G$. Then the union of all path-arcs of $P_{arc}$ either cover the arc $[r_{v_1}, r_{v_k}]$ or the arc $[r_{v_k}, r_{v_1}]$ (or both).

Observation 4. Let $G = (V, E)$ be a circular-arc graph and $R$ be a circular-arc representation of $G$. Let $P = (v_1, v_2, \ldots, v_k)$ be a path of $G$. $P_{arc}$ be the path-arc representation of $P$, and $i \in \{1, 2, \ldots, k-1\}$. If $P(v_i, v_{i+1}) = [r_{v_i}, r_{v_{i+1}}]$, then $P(v_i, v_{i+1}) \subseteq I_{v_{i+1}}$. Furthermore, if $P(v_i, v_{i+1}) = [r_{v_{i+1}}, r_{v_i}]$, then $P(v_i, v_{i+1}) \subseteq I_{v_i}$.

Definition 2. Let $G = (V, E)$ be a circular-arc graph and $R$ be a circular-arc representation of $G$. Let $P = (v_1, v_2, \ldots, v_k)$ be a path of $G$ and $P_{arc}$ be a path-arc representation of $P$. For every vertex $v_j$ of $P$, where $j \in \{1, 2, \ldots, k\}$,

- rightCut$_{arc}(v_j)$ is the number of path-arcs $P(v_j, v_{j+1})$ in $P_{arc}$, where $j \not\in \{i, i+1\}$, such that $r_{v_j} \in P(v_i, v_{i+1})$ and $P(v_i, v_{i+1}) = [r_{v_i}, r_{v_{i+1}}]$ (i.e. $P(v_j, v_{j+1})$ is right-going),

- leftCut$_{arc}(v_j)$ is the number of path-arcs $P(v_j, v_{j+1})$ in $P_{arc}$, where $j \not\in \{i, i+1\}$, such that $r_{v_j} \in P(v_i, v_{i+1})$ and $P(v_i, v_{i+1}) = [r_{v_{i+1}}, r_{v_i}]$ (i.e. $P(v_j, v_{j+1})$ is left-going),

- cut$_{arc}(v_j)$ is the number of path-arcs $P(v_j, v_{j+1})$ of $P_{arc}$, where $j \not\in \{i, i+1\}$, such that $r_{v_j} \in P(v_i, v_{i+1})$, i.e. cut$_{arc}(v_j) =$ rightCut$_{arc}(v_j) + $ leftCut$_{arc}(v_j)$.

Furthermore, we define sumCut$_{arc} = \sum_{j=1}^{k}$ cut$_{arc}(v_j)$.

Finally, we want to define an interval graph obtained from a circular-arc graph $G$ by “cutting” the circle and the arcs of a circular-arc representation $R$ of $G$ appropriately. Intuitively, given a vertex $u_i$ of $G$, we define in the following definition...
the interval graphs $G_u$ and $G'_{u_i}$, which we obtain by “cutting” $R$ immediately to the left and immediately to the right of $r_{u_i}$, respectively, where $r_{u_i}$ is the right endpoint of the arc $l_{u_i} = [l_{u_i}, r_{u_i}]$ of $R$ that corresponds to $u_i$. In the following definition, we denote for simplicity of the presentation by $\frac{[a, b]}{2}$ the mid-point of the arc $[a, b]$, for any two points $a$, $b$ on the circle.

**Definition 3.** Let $G = (V, E)$ be a circular-arc graph and $R$ be a circular-arc representation of $G$. Let $\pi = (u_0, u_1, \ldots, u_{n-1})$ be the right-end circular ordering of $G$ (in $R$) and $i \in \{0, 1, \ldots, n-1\}$. The circular-arc representation $R_{u_i}$ (resp. $R'_{u_i}$) is obtained by replacing in $R$ every arc $[l_{u_i}, r_{u_i}]$ such that $[\frac{r_{u_i}+n_{u_i}}{2}] \in [l_{u_i}, r_{u_i}]$ (resp. $[\frac{n_{u_i}+r_{u_i}+1}{2}] \in [l_{u_i}, r_{u_i}]$) by the arc $[\frac{r_{u_i}+n_{u_i}}{2}, r_{u_i}]$ (resp. $[\frac{n_{u_i}+r_{u_i}+1}{2}, r_{u_i}]$). Then, $G_{u_i}$ (resp. $G'_{u_i}$) is the graph induced by the representation $R_{u_i}$ (resp. $R'_{u_i}$).

Note by **Definition 3** that the graphs $G_{u_i}$ and $G'_{u_i}$ are (not necessarily induced) subgraphs of $G$. As an example, the subgraph $G_{u_3}$ of the circular-arc graph $G$ of Fig. 1(a) is illustrated in Fig. 2(a); in this figure, the arcs present in $G_{u_3}$ are drawn black, while all the others are drawn gray, for better visibility. The next observation follows now easily by **Definition 3**, as well as by the definitions of a right-end circular ordering (resp. a right-end ordering) in a circular-arc (resp. interval) graph.

**Observation 5.** Let $G = (V, E)$ be a circular-arc graph and $R$ be a circular-arc representation of $G$. Let $\pi = (u_0, u_1, \ldots, u_{n-1})$ be the right-end circular ordering of $G$ (in $R$) and $i \in \{0, 1, \ldots, n-1\}$. Then,

- $R_{u_i}$ and $R'_{u_i}$ can be rewritten as interval representations, while both $G_{u_i}$ and $G'_{u_i}$ are interval graphs; furthermore, $R'_{u_i} = R_{u_{i+1}}$ and $G'_{u_i} = G_{u_{i+1}}$.
- the right-end circular ordering $\pi$ of $V = V(\overline{G})$ defines in a natural way a right-end ordering $\pi_i = (u_i, u_{i+1}, \ldots, u_{n-1}, u_1, u_2, \ldots, u_{i-1})$ of the interval graph $G_{u_i}$.

## 3. Reduction of the problem to the case of interval graphs

In this section we show that the longest path problem for circular-arc graphs can be reduced to the longest path problem for interval graphs. In particular, given a circular-arc graph $G$ and a circular-arc representation $R$ of $G$, we prove that for every path $P$ of $G$ there exists a path $P'$ on the same vertices, which has a path-arc representation $P'_{\text{arc}}$, that does not cover the whole circle in $R$ (cf. **Theorem 1**), or equivalently, $P'$ is a path of the interval graph $G_{u_i}$, as defined in **Definition 3**, where $u_i$ is a vertex of $G$.

**Lemma 1.** Let $G = (V, E)$ be a circular-arc graph and $R$ be a circular-arc representation of $G$. Let $P = (v_1, v_2, \ldots, v_k)$ be a path in $G$ such that $\text{cut}_{\text{arc}}(v_1) = 0$. Then $P$ is also a path of $G_{v_1}$ or $G'_{v_1}$ (or of both).

**Proof.** If $k = 1$, the lemma holds trivially. Let in the following $k \geq 2$. Consider the path-arc representation $P_{\text{arc}}$ of $P$ and the path-arc $P(v_1, v_2)$ of $P$ between $v_1$ and $v_2$, and recall by **Definition 1** that either $P(v_1, v_2) = [r_{v_1}, r_{v_2}]$ or $P(v_1, v_2) = [r_{v_2}, r_{v_1}]$.

Suppose first that $P(v_1, v_2) = [r_{v_1}, r_{v_2}]$; then note that $P(v_1, v_2)$ is a path-arc of $R_{v_1}$. We will prove that in this case $P$ is a path of $G_{v_1}$, or equivalently that $P_{\text{arc}}$ is also a path-arc representation in $R_{v_1}$. Suppose otherwise that $P_{\text{arc}}$ is not a path-arc representation in $R_{v_1}$. Then, since $P_{\text{arc}}$ is a path-arc representation in $R$, there exists at least one index $i \in \{1, \ldots, k-1\}$ such that $r_{v_i} \in P(v_1, v_{i+1})$. However, this implies by **Definition 2** that $\text{cut}_{\text{arc}}(v_1) \geq 1$, which is a contradiction to the assumption that $\text{cut}_{\text{arc}}(v_1) = 0$. Therefore $r_{v_i} \notin P(v_1, v_{i+1})$ for every index $i \in \{2, \ldots, k-1\}$, i.e. $P(v_1, v_{i+1})$ is a path-arc of $R_{v_1}$ for every index $i \in \{2, \ldots, k-1\}$. Thus, since also $P(v_1, v_2)$ is a path-arc of $R_{v_1}$, it follows that $P_{\text{arc}}$ is a path-arc representation in $R_{v_1}$, i.e. $P$ is a path of $G_{v_1}$.

Suppose now that $P(v_1, v_2) = [r_{v_2}, r_{v_1}]$; then note that $P(v_1, v_2)$ is a path-arc of $R'_{v_1}$. We will prove that in this case $P$ is a path of $G'_{v_1}$, or equivalently that $P_{\text{arc}}$ is also a path-arc representation in $R'_{v_1}$. Suppose otherwise that $P_{\text{arc}}$ is not a path-arc
Lemma 2. Let $G$ be a circular-arc graph and $R$ be a circular-arc representation of $G$. Let $P = (v_1, v_2, \ldots, v_k)$ be a path of $G$ such that $cut_{arc}(v_k) = 0$. Then $P$ is also a path of $G$ or $G^R_k$ (or of both).

Corollary 2. Let $G = (V, E)$ be a circular-arc graph and $R$ be a circular-arc representation of $G$. Let $P = (v_1, v_2, \ldots, v_k)$ be a path of $G$ that has the smallest possible value $\text{sumCut}_{arc}$ among all paths of $G$ on the vertices $\{v_1, v_2, \ldots, v_k\}$. Then $\text{rightCut}_{arc}(v_1) = 0$.

Proof. Suppose otherwise that $\text{rightCut}_{arc}(v_1) \geq 1$. Then, there exists by Definition 2 at least one index $i \in \{2, \ldots, k - 1\}$ such that $r_{i+1} \in P(v_i, v_{i+1})$, which implies by Definition 2 that $cut_{arc}(v_1) \geq 1$. This is a contradiction to the assumption that $cut_{arc}(v_1) = 0$. Therefore $r_{i+1} \notin P(v_i, v_{i+1})$ for every index $i \in \{2, \ldots, k - 1\}$, i.e. $P(v_i, v_{i+1})$ is a path-arc of $R^R_k$ for every index $i \in \{2, \ldots, k\}$. Thus, since also $P(v_1, v_2)$ is a path-arc of $R^R_k$, it follows that $P_{arc}$ is a path-arc representation in $R^R_k$, i.e. $P$ is a path of $G_{R^R_k}$. This completes the proof of the lemma. □

The next corollary follows easily by considering Lemma 1 the reverse path $\overline{P} = (v_k, v_{k-1}, \ldots, v_1)$ of the path $P = (v_1, v_2, \ldots, v_k)$.

Corollary 1. Let $G = (V, E)$ be a circular-arc graph and $R$ be a circular-arc representation of $G$. Let $P = (v_1, v_2, \ldots, v_k)$ be a path in $G$ such that $cut_{arc}(v_k) = 0$. Then $P$ is also a path of $G$ or $G^R_k$ (or of both).

Lemma 2. Let $G = (V, E)$ be a circular-arc graph and $R$ be a circular-arc representation of $G$. Let $P = (v_1, v_2, \ldots, v_k)$ be a path of $G$ that has the smallest possible value $\text{sumCut}_{arc}$ among all paths of $G$ on the vertices $\{v_1, v_2, \ldots, v_k\}$. Then $\text{leftCut}_{arc}(v_1) = 0$.

Proof. Consider the reverse path $\overline{P} = (v_k, v_{k-1}, \ldots, v_1)$ of $P$. Note that both paths $\overline{P}$ and $P$ can be represented by the same path-arc representation $P_{arc} = P_{arc}$, i.e. $\overline{P}(v_i, v_{i+1}) = P(v_i, v_{i+1})$ for every $i \in \{1, 2, \ldots, k - 1\}$. Furthermore, since $P$ visits the vertices $\{v_1, v_2, \ldots, v_k\}$ in the reverse order than $P$, observe by Definition 2 that rightCut$_{arc}(v_i) =$ leftCut$_{arc}(v_i)$ and that leftCut$_{arc}(v_i) =$ rightCut$_{arc}(v_i)$, for every $i \in \{1, 2, \ldots, k\}$. Therefore cut$_{arc}(v_i) =$ cut$_{arc}(v_i)$, for every $i \in \{1, 2, \ldots, k\}$, and thus also sumCut$_{arc} =$ sumCut$_{arc}$. Therefore $\overline{P}$ has also the smallest possible value $\text{sumCut}_{arc}$ among all paths of $G$ on the vertices $\{v_1, v_2, \ldots, v_k\}$. Thus Lemma 2 can be applied to the path $\overline{P} = (v_k, v_{k-1}, \ldots, v_1)$, from which it follows that rightCut$_{arc}(v_1) = 0$, and thus leftCut$_{arc}(v_1) = 0$. □

Lemma 3. Let $G = (V, E)$ be a circular-arc graph and $R$ be a circular-arc representation of $G$. Let $P = (v_1, v_2, \ldots, v_k)$ be a path of $G$ that has the smallest possible value $\text{sumCut}_{arc}$ among all paths of $G$ on the vertices $\{v_1, v_2, \ldots, v_k\}$. Then cut$_{arc}(v_1) = 0$ or cut$_{arc}(v_k) = 0$.

Proof. Suppose otherwise that cut$_{arc}(v_1) \geq 1$ and cut$_{arc}(v_k) \geq 1$. Recall by Lemma 2 and Corollary 2 that rightCut$_{arc}(v_1) = 0$ and leftCut$_{arc}(v_k) = 0$, respectively. Therefore cut$_{arc}(v_1) =$ leftCut$_{arc}(v_1) \geq 1$ and cut$_{arc}(v_k) =$ rightCut$_{arc}(v_k) \geq 1$. Let $i \in \{1, 2, \ldots, k - 1\}$ be an index such that $r_i \in P(v_i, v_{i+1})$ and $i \notin \{1, 2, \ldots, k - 1\}$ be an index such that $r_i \notin P(v_i, v_{i+1})$. Then, since cut$_{arc}(v_1) =$ leftCut$_{arc}(v_1)$ and cut$_{arc}(v_k) =$ rightCut$_{arc}(v_k)$, it follows that $P(v_i, v_{i+1}) = [r_{i+1}, r_j]$ and $P(v_i, v_{i+1}) = [r_{i-1}, r_j]$. That is, $P(v_i, v_{i+1})$ is a left-going and $P(v_i, v_{i+1})$ is a right-going path-arc, which is a contradiction by Corollary 2. Therefore $r_i \notin [r_{i+1}, r_j]$. Similarly $r_i \notin [r_j, r_{i-1}]$, since otherwise rightCut$_{arc}(v_1) \geq 1$, which is again a contradiction by Lemma 2. Recall now that for every two distinct points $a$ and $b$ on the circle, $[a, b] \cap [b, a]$ covers the whole circle by Observation 1. Therefore, since $r_i \notin [r_{i+1}, r_j]$ and $r_i \notin [r_j, r_{i-1}]$, it follows that $r_i \notin [r_{i-1}, r_j]$ and $r_i \in [r_j, r_{i+1}]$, respectively.

Suppose that $j = i+1$, i.e. $P(v_i, v_{i+1}) = [r_{i+1}, r_j]$. Note that either $[r_{i+1}, r_{i+2}] \subseteq [r_{i+1}, r_j]$ or $[r_{i+1}, r_{i+2}] \subseteq [r_{i+1}, r_j]$. If $[r_{i+1}, r_{i+2}] \subseteq [r_{i+1}, r_j]$, then $v_k \in [r_{i+1}, r_j]$, which is a contradiction by the previous paragraph. Therefore $[r_{i+1}, r_j] \subseteq [r_{i+1}, r_{i+2}]$, and thus $r_i \in [r_{i+1}, r_{i+2}] = P(v_{i+1}, v_{i+2})$. Therefore rightCut$_{arc}(v_1) \geq 1$, since $P(v_{i+1}, v_{i+2})$ is a right-going path-arc, which is a contradiction by Lemma 2.
Thus $j \neq i + 1$. By symmetry, it follows similarly that also $i \neq j + 1$. We distinguish in the following the cases where $i < j$ and $i > j$, respectively.

**Case 1.** $i < j$. Then $P = (v_1, P_1, v_i, v_{i+1}, P_2, v_j, v_{j+1}, P_3, v_k)$ for some (possibly empty) subpaths $P_1, P_2, P_3$ of $P$. We will prove that $r_{v_i} \in [r_{v_{i+1}}, r_k]$ and $r_{v_j} \in [r_{v_{j+1}}, r_k]$. To the sake of contradiction, suppose first that $r_{v_j} \notin [r_{v_{j+1}}, r_k]$, i.e. $r_{v_j} \notin [r_{v_{j+1}}, r_k]$. Then $r_{v_j} \notin [r_{v_{j+1}}, r_k]$. Suppose that $r_{v_{j+1}} \in [r_{v_j}, r_k]$. Then $[r_{v_j}, r_{v_{j+1}}] \subseteq [r_{v_{j+1}}, r_k]$, which is a contradiction, as we proved above. Therefore $r_{v_{j+1}} \notin [r_{v_j}, r_k]$. Then $r_{v_j} \in [r_{v_{j+1}}, r_k]$, which is again a contradiction, as we proved above. Therefore $r_{v_j} \notin [r_{v_{j+1}}, r_k]$. Consider the subpath $\hat{P} = (v_{i+1}, P_2, v_j)$ of $P$. Then, **Observation 3** implies that all path-arcs of $\hat{P}_{arc}$ either cover the arc $[r_{v_{i+1}} r_{v_j}]$ or the arc $[r_{v_j}, r_{v_{i+1}}]$ (or both). Suppose first that the path-arcs of $\hat{P}_{arc}$ cover the arc $[r_{v_{i+1}}, r_{v_j}]$. Then, since $r_{v_j} \in [r_{v_{i+1}}, r_k]$, there must exist at least one index $\ell \in \{i+1, i+2, \ldots, j-1\}$, such that for the path-arc $\hat{P}_{arc}(v_{\ell}, v_{\ell+1})$ of $\hat{P}_{arc}$ it holds $r_{v_{\ell}} \in [r_{v_{i+1}}, r_{v_{\ell+1}}]$. This implies that rightCut$_{arc}(v_{\ell}) \geq 1$, and thus also rightCut$_{arc}(v_{\ell}) \geq 1$, since $P(v_{\ell}, v_{\ell+1}) = P(v_0, v_{\ell+1})$ is a right-going path-arc. This is a contradiction, since rightCut$_{arc}(v_{\ell}) = 0$. Suppose now that the path-arcs of $\hat{P}_{arc}$ cover the arc $[r_{v_j}, r_{v_{i+1}}]$. Then, since $r_{v_{i+1}} \in [r_{v_j}, r_k]$, there must exist at least one index $\ell \in \{i+1, i+2, \ldots, j-1\}$, such that for the path-arc $\hat{P}_{arc}(v_{\ell}, v_{\ell+1})$ of $\hat{P}_{arc}$ it holds $r_{v_{\ell}} \in [r_{v_j}, r_{v_{i+1}}]$; this is a contradiction, since leftCut$_{arc}(v_{\ell}) \geq 1$, and thus also leftCut$_{arc}(v_{\ell}) \geq 1$, since $\hat{P}(v_{\ell}, v_{\ell+1}) = \hat{P}(v_{0}, v_{\ell+1})$ is a left-going path-arc. This is again a contradiction, since leftCut$_{arc}(v_{\ell}) = 0$. Therefore cut$_{arc}(v_{\ell}) = 0$ or cut$_{arc}(v_{\ell}) = 0$ for the case where $i < j$.

**Case 2.** $i > j$. Then $P = (v_1, P_1, v_i, v_{i+1}, P_2, v_j, v_{j+1}, P_3, v_k)$ for some (possibly empty) subpaths $P_1, P_2, P_3$ of $P$. Recall that $r_{v_i} \in [r_{v_{i+1}}, r_k]$ and $r_{v_j} \in [r_{v_{j+1}}, r_k]$, as we proved above. Similarly to Case 1, we will prove that $r_{v_i} \in [r_{v_{i+1}}, r_k]$ and $r_{v_j} \in [r_{v_{j+1}}, r_k]$. To the sake of contradiction, suppose first that $r_{v_i} \notin [r_{v_{i+1}}, r_k]$, i.e. $r_{v_i} \notin [r_{v_{i+1}}, r_k]$. Then $r_{v_i} \notin [r_{v_{i+1}}, r_k]$. Suppose that $r_{v_j} \in [r_{v_i}, r_{v_{i+1}}]$. Then $[r_{v_i}, r_{v_{i+1}}] \subseteq [r_{v_{i+1}}, r_k]$, which is a contradiction. Therefore $r_{v_i} \notin [r_{v_j}, r_{v_{i+1}}]$, i.e. $r_{v_i} \notin [r_{v_{i+1}}, r_k]$. Then $r_{v_i} \notin [r_{v_{i+1}}, r_k]$, which is again a contradiction. Therefore $r_{v_i} \notin [r_{v_{i+1}}, r_k]$. Consider the subpath $\hat{P} = (v_{j+1}, P_2, v_j)$ of $P$. Then, **Observation 3** implies that all path-arcs of $\hat{P}_{arc}$ either cover the arc $[r_{v_{j+1}} r_{v_j}]$ or the arc $[r_{v_j}, r_{v_{i+1}}]$ (or both). Suppose first that the path-arcs of $\hat{P}_{arc}$ cover the arc $[r_{v_{j+1}}, r_{v_j}]$. Then, since $r_{v_{j+1}} \in [r_{v_{j+1}}, r_k]$, there must exist at least one index $\ell \in \{j+1, j+2, \ldots, i-1\}$, such that for the path-arc $\hat{P}_{arc}(v_{\ell}, v_{\ell+1})$ of $\hat{P}_{arc}$ it holds $r_{v_{\ell}} \in [r_{v_{j+1}}, r_{v_{\ell+1}}]$, which is a contradiction, since rightCut$_{arc}(v_{\ell}) = 0$. Suppose now that the path-arcs of $\hat{P}_{arc}$ cover the arc $[r_{v_j}, r_{v_{j+1}}]$. Then, since $r_{v_{j+1}} \in [r_{v_j}, r_k]$, there must exist at least one index $\ell \in \{j+1, j+2, \ldots, i-1\}$, such that for the path-arc $\hat{P}_{arc}(v_{\ell}, v_{\ell+1})$ of $\hat{P}_{arc}$ it holds $r_{v_{\ell}} \in [r_{v_j}, r_{v_{j+1}}]$, which is a contradiction. Therefore $r_{v_{j+1}} \notin [r_{v_j}, r_{v_{j+1}}]$, i.e. $r_{v_{j+1}} \notin [r_{v_{j+1}}, r_k]$. Then $r_{v_{j+1}} \notin [r_{v_{j+1}}, r_k]$, which is again a contradiction. Therefore $r_{v_{j+1}} \notin [r_{v_{j+1}}, r_k]$.

We are now ready to present the main theorem of this section.

**Theorem 1.** Let $G = (V, E)$ be a circular-arc graph, $R$ be a circular-arc representation of $G$, and $P$ be any path of $G$. Then there exists a vertex $v \in V$ and a path $P'$ with $V(P') = V(P)$, such that $P'$ is also a path of the interval graph $G_v$.

**Proof.** Let $\pi = (u_0, u_1, \ldots, u_{n-1})$ be the right-end circular ordering of $G$ and let $V(P) = \{v_1, v_2, \ldots, v_k\}$. Among all paths of $G$ on the vertices $\{v_1, v_2, \ldots, v_k\}$, let $P' = (v_1, v_2, \ldots, v_k)$ have the smallest possible value $\text{sumCut}_{arc}$. Then, **Lemma 3** implies that cut$_{arc}(v_1) = 0$ or cut$_{arc}(v_k) = 0$. If cut$_{arc}(v_1) = 0$ (resp. cut$_{arc}(v_k) = 0$), then it follows by **Corollary 1** that $P'$ is also a path of $G_{v_1}$ or $G_{v_k}$ (resp. $G_{v_k}$ or $G_{v_1}$). If $P'$ is a path of $G_{v_1}$ (resp. of $G_{v_k}$), then clearly the vertex $v = v_1$ (resp. $v = v_k$) satisfies the conditions of the theorem. Let $P'$ be a path of $G_{v'}$ (resp. of $G_{v'}$). Furthermore, let $v_1 = u_i$ (resp. $v_k = u_i$) for some $i \in \{0, 1, \ldots, n-1\}$. Then $C_{v'} = G_{v_{i+1}}$ (resp. $C_{v'} = G_{v_{i+1}}$) by **Observation 5**, i.e. the vertex $v = u_{i+1}$ satisfies the conditions of the theorem. This completes the proof of the theorem. □

This structural theorem suggests a generic reduction of a number of path problems on circular-arc graphs to the corresponding problem on interval graphs. For instance, in order to decide the Hamiltonian path problem on a circular-arc graph $G = (V, E)$, we can use one of the known algorithms for the Hamiltonian path problem on interval graphs (e.g. [1]).
and apply it to the interval graph \( G_v \), for every vertex \( v \in V \). Then, Theorem 1 implies that \( G \) has a Hamiltonian path if and only if \( G_v \) has a Hamiltonian path for at least one \( v \in V \). Since the algorithm of [1] has running time \( O(n + m) \) for an interval graph with \( n \) vertices and \( m \) edges, such a reduction provides an \( O(n(n + m)) \) time algorithm for the Hamiltonian path problem on a circular-arc graph. This problem can be also solved by any of the other known Hamiltonian path or cycle algorithms on circular-arc graphs (for instance see [17,30,18,8]); some of these known algorithms have running time faster than \( O(n(n + m)) \) as they exploit directly the structure of circular-arc graphs, rather than reducing the problem to the case of interval graphs.

Moreover, Theorem 1 can be used to provide the first polynomial algorithm for the longest path problem on circular-arc graphs, which works as follows. For every vertex \( v \in V \) of the given circular-arc graph \( G = (V, E) \), compute a longest path \( P_v \) in the interval graph \( G_v \) by the algorithm in [19]. Then, the longest path \( P_v \) among all \( v \in V \) is also a longest path of \( G \). Since the complexity of the algorithm in [19] is \( O(n^3) \) when applied to a graph with \( n \) vertices, the complexity of the above algorithm for circular-arc graphs becomes \( O(n^3) \). However, in order to reduce the complexity of this algorithm, we will exploit in the sequel the structure of circular-arc graphs rather than reducing the longest path problem to the case of interval graphs. Moreover, our algorithm also counts all different longest paths of a specific “normal” type in the input circular-arc graph \( G \).

4. Computation and counting of longest paths in circular-arc graphs

In this section we present the first polynomial algorithm (cf. Algorithm 1) that computes a longest path of a circular-arc graph \( G = (V, E) \). In order to present our algorithm, we introduce the notion of a normal path of a circular-arc graph (cf. Definition 6). Using our structural Theorem 1 of the previous section, we are able to prove the basic property that for every path \( P \) of a circular-arc graph \( G \), there exists another path \( P' \) on the same vertices, which is normal in \( G \). Therefore, normal paths can be thought as “representatives” of several non-normal paths. Furthermore, using this notion, our Algorithm 1 computes also the number \( N \) of all different normal paths in the given circular-arc representation of \( G \). This number \( N \) constitutes an \( n \)-approximation of the number of different vertex sets \( S \subseteq V \) such that \( V(P) = S \) for a longest path \( P \) of \( G \); note here that the number of such sets \( S \) is exponential in the worst case. Moreover, with a slight modification of the algorithm, we manage to compute the exact number of such sets \( S \) that provide a longest path of \( G \), in the case where \( G \) is an interval graph. First, we recall the notion of a normal path in an interval graph \( G \), given a right-end ordering \( \pi \) of \( G \) [19].

Definition 4 ([19]). Let \( G = (V, E) \) be an interval graph and \( \pi \) be a right-end ordering of \( G \). A path \( P = (v_1, v_2, \ldots, v_k) \) of \( G \) is normal if \( v_1 \) is the leftmost vertex of \( V(P) \) in \( \pi \) and \( v_i \) is the leftmost vertex of \( N(v_{i-1}) \cap \{v_1, v_{i+1}, \ldots, v_k\} \) in \( \pi \), for every \( i = 2, \ldots, k \).

The concept of a normal path in interval graphs has first appeared in [23] (referred to as a straight path), as well as later in [8].

Lemma 4 (See [23,19]). Let \( G = (V, E) \) be an interval graph, \( \pi \) be a right-end ordering of \( G \), and \( P \) be a path of \( G \). Then, there exists a normal path \( P' \) of \( G \) such that \( V(P') = V(P) \).

We define in the following the (not necessarily induced) subgraphs \( G_i(j) \) and \( G_i(j) \) of a circular-arc graph \( G = (V, E) \), where \( i, j \in \{0, 1, \ldots, |V| - 1\} \). These subgraphs of \( G \) will be used in the sequel for the analysis of our algorithm for the longest path problem on circular-arc graphs.

Definition 5. Let \( G = (V, E) \) be a circular-arc graph, \( R \) be a circular-arc representation of \( G \), and \( \pi = (u_0, u_1, \ldots, u_{n-1}) \) be the right-end circular ordering of \( G \) (in \( R \)). For every pair of indices \( i, j \in \{0, 1, \ldots, n - 1\} \),

- \( G_i(j) = G_i(\{u_i, u_{i+1}, \ldots, u_j\}) \), and

- \( G_i(j) \) is the subgraph of \( G \) induced by the vertex set \( S = \{u_i, u_{i+1}, \ldots, u_j\} \) \( \setminus \{u_k \in V \mid r_{u_k-1} \in I_{u_k}\} \).

Observation 6. Let \( i, j \in \{0, 1, \ldots, n - 1\} \) be a pair of indices, where \( i \neq j \). Then \( G_{u_i} = G_i(i - 1) \) and \( G_{u_i} = G_i(i - 1) \) in \( \{u_i\} \).

It is easy to see by Definition 5 that both \( G_i(j) \) and \( G_i(j) \) are induced subgraphs of the interval graph \( G_{u_i} \), while they are (not necessarily induced) subgraphs of the circular-arc graph \( G \). Furthermore, \( G_i(j) \) is an induced subgraph of \( G_{i-1}(j) \), for every pair \( i, j \) of indices such that \( j \neq i - 1 \) mod \( n \). Note that always \( u_{i-1} \notin V(G_{i}(j)) \), since the graph \( G_{i}(j) \) is defined in Definition 5 only for pairs of indices \( i, j \) such that \( j \neq i - 1 \) mod \( n \). Moreover, note that the vertices \( u_i \) and \( u_{j+m} \) may or may not belong to \( G_{i}(j) \), since they may or may not belong to the set \( \{u_k \in V \mid r_{u_k-1} \in I_{u_k}\} \). In the following, for an arbitrary \( i \in \mathbb{Z} \), we may refer for simplicity to the interval graphs \( G_i(\{u_i \mod n\}) \) \( \{u_i \mod n\} \) and \( G_i(\{u_i \mod n \}) \) \( \{u_i \mod n \} \) as \( G_i(j) \) and \( G_i(j) \), respectively. As an example, the subgraphs \( G_i(6) \) and \( G(1, 6) \) of the circular-arc graph \( G \) of Fig. 1(a) are illustrated in Fig. 2(b) and (c), respectively. In these figures, the arcs present in the graphs \( G(1, 6) \) and \( G(1, 6) \) are drawn black, while all the others are drawn gray, for better visibility. That is, \( V(G(1, 6)) = \{u_1, u_2, u_3, u_4, u_5, u_6\} \) and \( V(G(1, 6)) = \{u_1, u_3, u_4, u_5\} \).
Notation 1. A right-end circular ordering $\pi = (u_0, u_1, \ldots, u_{n-1})$ of a circular-arc graph $G = (V, E)$ determines in a natural way a right-end ordering $\sigma$ of the interval graph $G(\pi)$ (resp. $G(i, j)$), for every possible pair of indices $i, j$. More specifically, this right-end ordering $\sigma$ of the vertices of $G(i, j)$ is $\sigma = (u_i, u_{i+1}, \ldots, u_j)$. Furthermore, in the case of $G(i, j)$, $\sigma = (u_i, u_{i+1}, \ldots, u_j) \setminus \{u_k \in V \mid r_{u_k} \in I_{u_j}\}$. In the following, given a right-end circular ordering $\pi$ of $G$, we will refer to such an ordering $\sigma$ of the vertices of $G(i, j)$ (resp. of $G(i, j)$) as the “right-end ordering $\sigma$ of $G(i, j)$ (resp. of $G(i, j)$) induced by $\pi$”. Furthermore, for two vertices $u_i, u_j$ of $G(i, j)$ (resp. of $G(i, j)$), we will write $u_i <_\sigma u_j$ if $u_i$ appears before $u_j$ in the ordering $\sigma$.

Now, using Notation 1, we can extend in a natural way the notion of normal paths (cf. Definition 4) to the case of circular-arc graphs.

Definition 6. Let $G = (V, E)$ be a circular-arc graph and $\pi$ be a circular right-end ordering of $G$. A path $P$ of $G$ is normal if $P$ is a normal path in the interval graph $G_\pi$ for some vertex $u \in V$ (with respect to the right-end ordering $\sigma$ of $G_\pi$ induced by $\pi$).

For example, the path $P = (u_3, u_4, u_5, u_6, u_7, u_2, u_1)$ is a normal path of the circular-arc graph $G$ of Fig. 1(a), since $P$ is a normal path of the interval subgraph $G_\pi$ of $G$; cf. Fig. 2(a). Normal paths in circular-arc graphs, as defined in Definition 6, behave similarly to normal paths in interval graphs. Indeed, the next theorem (which is the extension of Lemma 4 to the case of circular-arc graphs) follows directly by Lemma 4 and by the structural Theorem 1.

Theorem 2. Let $G = (V, E)$ be a circular-arc graph, $\pi$ be a circular right-end ordering of $G$, and $P$ be a path of $G$. Then there exists a normal path $P'$ of $G$ with $V(P') = V(P)$.

In the following, given a path $P$ of a circular-arc graph $G$, we may assume without loss of generality by Theorem 2 that $P$ is a normal path of $G$. The next observations can be obtained using Notation 1 and Definition 5.

Observation 7. Let $G = (V, E)$ be a circular-arc graph, $R$ be a circular-arc representation of $G$, and $\pi = (u_0, u_1, \ldots, u_{n-1})$ be a right-end circular ordering of $G$ in $R$. Let $i \neq j$ be two indices and $\sigma$ be the right-end ordering of $G(i, j)$ (resp. of $G(i, j)$) induced by $\pi$. Let $u_i$ and $u_j$ be two vertices of $G(i, j)$ (resp. of $G(i, j)$), such that $u_i <_\sigma u_j$. If $u_i \in N_G(u_j)$ (resp. $u_i \in N_G(u_j)$), then $r_{u_i} \in I_{u_j}$ and $v_{u_i} \in R$.

Observation 8. Let $P_1 = (P_0, u_i)$ be a normal path of $G(i,j)$ (resp. of $G(i,j)$), for some pair of indices, where $i \neq j$, and let $r_{P_0} \in I_{u_i}$ in the circular-arc representation of $G$. Then $P_1 = (P_0, u_i)$ is a normal path of $G(i,j)$ (resp. if $u_j \in V(G(i,j))$, then $P = (P_1, u_i)$ is a normal path of $G(i,j)$).

Observation 9. Let $P = (P_1, u_j)$ be a normal path of $G(i,j)$ (resp. of $G(i,j)$), for some pair of indices $i, j$, where $i \neq j$. Then $P_1$ is a normal path of both $G(i, j)$ and $G(i,j)$ (resp. of both $G(i, j)$ and $G(i,j)$).

Now we provide two auxiliary lemmas that will be used in the sequel of this section.

Lemma 5. Let $G = (V, E)$ be a circular-arc graph, $R$ be a circular-arc representation of $G$, and $\pi$ be the right-end circular ordering of $G$ in $R$. Let $\sigma$ be the right-end ordering of $G(i,j)$ (resp. of $G(i,j)$) induced by $\pi$, where $i \neq j$. Let $P = (u_1, u_2, \ldots, u_k)$ be a path of $G$, and let $v_i \not\in V(P)$ be a vertex of $G(i,j)$ (resp. of $G(i,j)$) such that $v_i \not\in \sigma v_t \not\in v_k$ and $v_j \not\in v_k$. Then, there exist two consecutive vertices $v_{t-1}$ and $v_{t+1}$ in $P$, $1 \leq t \leq k$, such that $r_{v_{t-1}} \in I_{v_k}$ in $R$.

Proof. Since $v_i <_\sigma v_t <_\sigma v_k$ and $P$ is a path from $v_1$ to $v_k$, there must exist two consecutive vertices $v_{t-1}, v_t$, where $2 \leq t \leq k$, such that $v_{t-1} < v_t < v_k$. Note that $u_{i-1} \in N_G(v_i)$ (resp. $u_{i-1} \in N_G(v_i)$), since $u_{i-1}$ and $v_i$ are consecutive in $P$. Therefore $r_{v_{t-1}} \in I_{v_k}$ in $R$ by Observation 7, and thus also $r_{v_{t-1}} \in I_{v_k}$ in $R$, since $v_{t-1} < v_t < v_k$. Let now $v_j$ be the last vertex in $P$, such that $r_{v_{t-1}} \in I_{v_k}$ in $R$. Note that $v_j \neq v_k$, since $v_j \not\in v_k$ in $R$ by assumption, and thus the vertex $v_{t+1}$ exists in $P$. If $v_{t+1} < v_{t+1} v_j$, then there exists similarly to the above a vertex $v_{t+1}$ in $P$, where $t^* > i + 1$, such that $r_{v_j} \in I_{v_{t+1}}$, in $R$. This is a contradiction to the assumption that $v_j$ is the last vertex in $P$, such that $r_{v_j} \in I_{v_k}$ in $R$. Therefore $v_j < v_{t+1}$. This completes the proof of the lemma.

Lemma 6. Let $G = (V, E)$ be a circular-arc graph and $\pi$ be a right-end circular ordering of $G$. Let $\sigma$ be the right-end ordering of $G(i,j)$ (resp. of $G(i,j)$) induced by $\pi$, where $i \neq j$. If $u \in V(G(i,j))$, then $u \in V(G(i,j))$, such that $x \neq j - 1$, then $V(G(x + 1, j - 1)) \subseteq V(G(i,j))$ (resp. $V(G(x + 1, j - 1)) \subseteq V(G(i,j))$).

Proof. Denote $\pi = (u_0, u_1, \ldots, u_{n-1})$. Consider first the case where $\sigma$ is a right-end ordering of $G(i, j)$. Let $u_x \in V(G(i,j)) = \{u_i, u_{i+1}, \ldots, u_{j-1}\}$, where $x \neq j - 1$. Recall by Definition 5 that $V(G(x + 1, j - 1)) \subseteq \{u_{k+1}, u_{k+2}, \ldots, u_{j-1}\}$. Therefore, since $x \neq j - 1$, it follows that $V(G(x + 1, j - 1)) \subseteq \{u_i, u_{i+1}, \ldots, u_{j-1}\}$, and thus also $V(G(x + 1, j - 1)) \subseteq \{u_i, u_{i+1}, \ldots, u_{j-1}\} = V(G(i,j))$ by Definition 5.

Consider now the case where $\sigma$ is a right-end ordering of $G(i, j)$. Let $u_x \in V(G(i,j)) \subseteq \{u_i, u_{i+1}, \ldots, u_{j-1}\}$, where $x \neq j - 1$. Consider a vertex $u_{y} \in V(G(x + 1, j - 1))$. Then also $u_{y} \in \{u_i, u_{i+1}, \ldots, u_{j-1}\}$ by the previous paragraph. Furthermore, due to Definition 5, $r_{u_x} \not\in I_{u_y}$ in the circular-arc representation $R$ of $G$, and thus also $r_{u_x} \not\in I_{u_y}$ in $R$. Therefore $u_{y} \in V(G(i,j))$ by Definition 5, and thus $V(G(x + 1, j - 1)) \subseteq V(G(i,j))$. □
In the following we state four lemmas (cf. Lemmas 7–10) that are crucial for the proof of the main theorem of this section (i.e. Theorem 3).

**Lemma 7.** Let $G = (V, E)$ be a circular-arc graph and $\pi = (u_0, u_1, \ldots, u_{n-1})$ be a right-end circular ordering of $G$. Let $i \neq j$ be two indices and $\sigma$ be the right-end ordering of $G(i,j)$ induced by $\pi$, where $j \neq i - 1 \mod n$. Let $u_i \in V(G(i,j))$, $u_j \in V(G(i,j-1))$, $u_k \in V(G(x+1, j-1))$, and $u_r \in N_G(u_i)$. Furthermore, let $P_1$ be a normal path of $G(i,j-1)$ with $u_i$ as its last vertex, where $x \neq j - 1$, and $P_2$ be a normal path of $G(x+1, j-1)$ with $u_r$ as its last vertex. Then $P = (P_1, u_i, P_2)$ is a normal path of $G(i,j)$ with $u_i$ as its last vertex.

**Proof.** For simplicity reasons, whenever we refer to $N(u_i)$ in the sequel of the proof, we will mean $N_G(u_i)(u_i)$, i.e. the neighborhood set of vertex $u_i$ in $G(i,j)$. We will first prove that $V(P_1) \subseteq V(G(i,j-1)) \setminus V(G(x+1, j-1))$. Suppose otherwise that $V(P_1) \cap V(G(x+1, j-1)) \neq \emptyset$, and let $u_k$ be the first vertex of $V(G(x+1, j-1))$ in $P_1$. If $u_k \in N_G(u_i)$, then Observation 7 implies that $r_{u_i} \in I_{u_i}$ in the circular-arc representation $R$ of $G$. This is a contradiction, since we assumed that $u_k \in V(G(x+1, j-1))$. Therefore, $u_k \notin N_G(u_i)$.

Furthermore, since we assumed that $u_k \in V(G(x+1, j-1))$, it follows by Definition 5 that $u_k <_\sigma u_i$, and thus $u_k$ is not the leftmost vertex of $P_1$ in $\sigma$. Therefore, since $P_1$ is a normal path by assumption, $u_k$ is not the first vertex of $P_1$, and thus there exists a previous vertex $u_{i-1}$ of $u_i$ in $P_1$. Therefore in particular $u_{i-1} \in N_G(u_i)$; note also that $u_{i-1} \neq u_i$, since $u_i$ is the last vertex of $P_1$. Suppose first that $u_{i-1} \notin N_G(u_i)$, i.e. $u_{i-1} <_\sigma u_i <_\sigma u_k$. Then, since $u_{i-1} \notin N_G(u_i)$, it follows by Observation 7 that also $u_k \notin N_G(u_{i-1})$, which is a contradiction, as we proved above. Suppose now that $u_{i-1} \notin N_G(u_i)$. Then, since $u_{i-1} <_\sigma u_k$ and $u_k$ is unvisited by $P_1$ when $u_k$ is visited, it follows that $u_k$ is not the leftmost unvisited vertex of $N_G(u_i) \cap V(P_1)$ in $\sigma$, when $P_1$ visits $u_k$. This is a contradiction by Definition 4, since $u_k$ is the next vertex of $u_i$ in $P_1$ and $P_1$ is a normal path by assumption. Let $u_k \notin N_G(u_i)$, and thus also $r_{u_i} \notin I_{u_i}$ in the circular-arc representation $R$ of $G$. Therefore $u_k \in V(G(x+1, j-1))$ by Definition 5. This is a contradiction to the assumption that $u_k$ is the first vertex of $V(G(x+1, j-1))$ in $P_1$. Therefore $V(P_1) \cap V(G(x+1, j-1)) = \emptyset$, i.e. $V(P_1) \subseteq V(G(i,j-1)) \setminus V(G(x+1, j-1))$.

Since $V(P_1) \subseteq V(G(i,j-1)) \setminus V(G(x+1, j-1))$ by the previous paragraph and $V(P_2) \subseteq V(G(x+1, j-1))$ by assumption, it follows that, since $u_i \in N_G(u_i)$ by assumption, and since $u_{i-1} <_\sigma u_k <_\sigma u_r$ for every $u_r \in N_G(u_i)$, it follows by Observation 7 that $u_r \notin N_G(u_i)$ for every $u_r \in V(P_2)$. Moreover, recall that $V(P_1) \subseteq V(G(i,j-1)) \subseteq V(G(i,j))$ by Observation 6 and that $V(P_2) \subseteq V(G(x+1, j-1)) \subseteq V(G(i,j))$ by Lemma 6 (since $x \neq j - 1$). Therefore, since $u_i \in V(G(i,j))$ and $u_k \notin N_G(u_i)$ by assumption, it follows that $P = (P_1, u_i, P_2)$ is a path of $G(i,j)$. Moreover, $u_i$ is the last vertex of $P$, since $u_i$ is the last vertex of $P_2$ by assumption.

In the following we prove that $P$ is normal. Let $u_r$ be the first vertex of $P_1$. Note $u_r$ is also the first vertex of $P_2$, since $P = (P_1, u_r, P_2)$. Further, $u_r$ is the leftmost vertex of $P_1$ in $\sigma$ by Definition 4, since $P_1$ is normal by assumption. Furthermore, note that $u_{i-1} \leq \sigma u_r <_\sigma u_i$ for every $u_r \in V(P_2) \cup \{u_i\}$, since $V(P_2) \subseteq V(G(x+1, j-1))$. Therefore, $u_{i-1}$ is also the leftmost vertex of $P_1$ in $\sigma$. Let $u_r$ and $u_{i-1}$ be two consecutive vertices of $P_1$, i.e. $u_r$ is the leftmost unvisited vertex of $N_G(u_i) \cap V(P_1)$ in $\sigma$, when $P_1$ visits $u_r$. We will prove that $u_{i-1}$ is also the leftmost unvisited vertex of $V(u_i) \cap V(P)$ in $\sigma$, when $P$ visits $u_i$. Suppose otherwise that $u_{i-1} \neq u_{i-1}$ is the leftmost unvisited vertex of $V(u_i) \cap V(P)$ in $\sigma$, when $P$ visits $u_i$. Then in particular $u_{i-1} <_\sigma u_{i-1}$ and $u_r \in N_G(u_i)$. If $u_r \in V(P_1)$, then $u_r$ would be also the leftmost unvisited vertex of $V(u_i) \cap V(P)$ in $\sigma$, when $P_1$ visits $u_r$, which is a contradiction. Furthermore, if $u_r \notin V(P_1)$ and $u_{i-1} \neq u_r$, and thus $u_r \notin V(P_2)$.

Since $u_i \in V(P_2) \subseteq V(G(x+1, j-1))$, it follows by particular Definition 5 that $u_i <_\sigma u_r$ and that $r_{u_i} \notin I_{u_i}$ in the circular-arc representation $R$ of $G$. Therefore also $u_i \notin N_G(u_i)$. Suppose that $u_i \in N_G(u_i)$, then, since $u_i <_\sigma u_k$ and $u_k$ is unvisited when $P$ visits $u_i$, it follows that $u_k$ is not the leftmost unvisited vertex of $V(u_i) \cap V(P)$ in $\sigma$, when $P$ visits $u_i$, which is a contradiction to the assumption on $u_i$. Thus $u_i \notin N_G(u_i)$. Suppose that $u_i <_\sigma u_{i-1}$, i.e. $u_i <_\sigma u_k <_\sigma u_{i-1}$. Then, since $u_i \in N_G(u_i)$, it follows by Observation 7 that also $u_r \notin N_G(u_i)$, which is a contradiction as we proved above. Therefore $u_r <_\sigma u_i$. Furthermore, since $u_i \notin N_G(u_i)$, it follows that $r_{u_i} \notin I_{u_i}$ in the circular-arc representation $R$ of $G$. Therefore $u_i \in V(G(x+1, j-1))$ by Definition 5. This is a contradiction, since $u_i \in V(P_1) \subseteq V(G(i,j-1)) \setminus V(G(x+1, j-1))$. Therefore, for any two consecutive vertices $u_i, u_{i-1}$ of $P_1, u_r$ is the leftmost unvisited vertex of $V(u_i) \cap V(P)$ in $\sigma$, when $P$ visits $u_i$.

Recall that $V(P_2) \subseteq V(G(x+1, j-1))$ by assumption, and thus $u_r \notin N_G(u_i)$ for every vertex $u_r \in V(P_2)$. Therefore, $u_r$ is the leftmost unvisited vertex of $V(u_i) \cap V(P)$ in $\sigma$, when $P$ visits $u_i$ (i.e. the last vertex of $P_1$). Note that exactly the vertices of $V(P_2)$ are the unvisited vertices of $V(P)$, when $P$ visits $u_i$. Furthermore recall that, since $u_i \in N_G(u_i)$ by assumption and $u_k <_\sigma u_i <_\sigma u_r$ for every $u_r \in V(P_2) \subseteq V(G(x+1, j-1))$, it follows by Observation 7 that $u_i \notin N_G(u_r)$ for every $u_r \in V(P_2)$. Moreover, recall that $P_2$ is a normal path by assumption. Therefore, the first vertex of $P_2$ is also the leftmost unvisited vertex of $V(u_i) \cap V(P)$ in $\sigma$, when $P$ visits $u_i$. Consider now any pair of consecutive vertices $u_i, u_{i-1}$ of $P_2$. Then, $u_{i-1}$ is the leftmost unvisited vertex of $V(u_i) \cap V(P)$ in $\sigma$, when $P_2$ (resp. $P$) visits $u_i$. Therefore, $P$ is a normal path. This completes the proof of the lemma. \[\square\]

Recall now that always $u_i \in V(G(i,j))$ by Definition 5 for every pair $i,j$ of indices. Furthermore, recall that $G(i,j)$ is an induced subgraph of $G(j)$, for every pair $i, j$ of indices, where $i \neq j$ and $j \neq i - 1 \mod n$. Now, following exactly the same proof as in Lemma 7, where we replace $G(i,j)$ by $G(j)$, we obtain the next lemma.

**Lemma 8.** Let $G = (V, E)$ be a circular-arc graph and $\pi = (u_0, u_1, \ldots, u_{n-1})$ be a right-end circular ordering of $G$. Let $i \neq j$ be two indices and $\sigma$ be the right-end ordering of $G(j)$ induced by $\pi$. Let $u_i \in V(G(i,j-1))$, $u_r \in V(G(x+1, j-1))$, and
and $u_k \in N_{G(i)}(u_i)$. Furthermore, let $P_1$ be a normal path of $G(i-1)$ with $u_i$ as its last vertex and $P_2$ be a normal path of $G(x+1,j-1)$ with $u_j$ as its last vertex. Then $P = (P_1, u_i, P_2)$ is a normal path of $G(i)$ with $u_j$ as its last vertex.

**Notation 2.** Let $G = (V, E)$ be a circular-arc graph and $\pi = (u_0, u_1, \ldots, u_{n-1})$ be a right-end circular ordering of $G$. Let $u_k \in V(G(i,j))$ (resp. $u_k \in V(G(i,j))$), and let $P$ be a normal path of $G(i,j)$ (resp. of $G(i,j)$), for some pair $i, j$ of indices. For simplicity of presentation, we will use the following in the following that “$P$ is a longest normal path of $G(i,j)$ (resp. of $G(i,j)$) with $u_k$ as its last vertex” if $P$ has the greatest number of vertices among those normal paths of $G(i,j)$ (resp. of $G(i,j)$) that have $u_k$ as their last vertex.

**Lemma 9.** Let $G = (V, E)$ be a circular-arc graph and $\pi = (u_0, u_1, \ldots, u_{n-1})$ be a right-end circular ordering of $G$. Let $i \neq j$ be two indices and $\sigma$ be the right-end ordering of $G(i,j)$ induced by $\pi$, where $j \neq i - 1 \mod n$. Let $P$ be a longest normal path of $G(i,j)$ with $u_i \neq u_j$ as its last vertex and let $P = (P_1, u_i, P_2)$. Let $u_j$ be the last vertex of $P_1$. Then $P_1$ is a longest normal path of $G(i,j)$ with $u_j$ as its last vertex and $P_2$ is a longest normal path of $G(x+1,j-1)$ with $u_j$ as its last vertex.

**Proof.** For simplicity reasons, whenever we refer to $N(u)$ in the sequel of the proof, we will mean $N_{G(i,j)}(u)$, i.e. the neighborhood set of vertex $u$ in $G(i,j)$. Note that $P$ has at least two vertices, since $u_i, u_j \in V(P)$. Therefore, since $u_k <_\pi u_i$ for every $u_k \in V(P) \setminus \{u_i\}$, it follows that $u_i$ is not the leftmost vertex of $V(P)$ in $\pi$. Therefore, since $P$ is normal by assumption, $u_i$ is not the first vertex of $P$, and thus $P_1 \neq \emptyset$. Note that $V(P_1) \subseteq V(G(i,j-1))$, i.e. $V(P_1) \subseteq V(G(i,j)) \setminus \{u_i\}$ by Observation 6, since $u_j \notin V(P_1)$. Furthermore, since $P$ is a normal path by assumption and $P_1$ is a subpath of $P$, it follows that $P_1$ is a normal path of $G(i,j-1)$ with $u_j$ as its last vertex.

Note that $P_2$ has at least one vertex since, $u_j \in V(P_2)$ and $u_j \neq u_i$ by assumption. We will now prove that $x \neq j - 1$ and $V(P_2) \subseteq V(G(x+1,j-1))$. Consider an arbitrary vertex $u_k \in V(P_2)$ and note that $u_k <_\pi u_j$. Furthermore, note that both $u_i$ and $u_k$ are unvisited by $P$ when $u_j$ is visited. Suppose that $u_k \in N(u_i)$. Then, since $u_k <_\pi u_j$, it follows that $u_j$ is not the leftmost unvisited vertex of $N(u_i) \cap V(P)$ in $\sigma$, when $P$ visits $u_j$. Thus, since $P$ is normal by assumption, it follows by Definition 4 that $u_j$ is not the next vertex of $u_i$ in $P$, which is a contradiction. Therefore $u_j \notin N(u_i)$ for every $u_k \in V(P_2)$.

Suppose first that $u_k <_\pi u_i$. Let $u_i$ be the first vertex of $P$. Then $u_i$ is the leftmost vertex of $V(P)$ in $\pi$, since $P$ is a normal path by assumption. Therefore, since $u_i \notin V(P_1)$, it follows that $u_k \neq u_i$, and thus $u_i <_\pi u_k <_\pi u_j$. Thus, since $u_i$ and $u_k$ are the first and the last vertices of $P_1$, respectively, and since $u_k \notin V(P_1)$, there exist by Lemma 5 two consecutive vertices $u_p$ and $u_q$ in $P_1$, such that $r_{u_p} \in I_{u_q}$ in $R$ and $u_k <_\pi u_j$. Note that both $u_i$ and $u_q$ are unvisited when $P$ visits $u_j$. Thus, $u_q$ is not the leftmost unvisited vertex of $N(u_q) \cap V(P)$ in $\sigma$, when $P$ visits $u_j$. This is a contradiction, since $P$ is a normal path by assumption. Therefore $u_k <_\pi u_i$, i.e. $u_i <_\pi u_k <_\pi u_j$, for every $u_k \in V(P_2)$, and thus in particular $x \neq j - 1$. Therefore, since also $u_j \notin N(u_k)$ for every $u_k \in V(P_2)$ by the previous paragraph, it follows that $r_{u_j} \notin I_{u_k}$ in the circular-arc representation $R$ of $G$. That is, $x \neq j - 1$ and $V(P_2) \subseteq V(G(x+1,j-1))$ by Definition 5.

Recall that $u_i <_\pi u_j$ and $u_i \in N(u_j)$. Therefore, Observation 7 implies that $u_k \in N(u_j)$ for every vertex $u_k$ such that $u_k <_\pi u_i$. Thus, in particular $u_k \in N(u_j)$ for every $u_k \in V(P_2)$, since $V(P_2) \subseteq V(G(x+1,j-1))$ by the previous paragraph. Therefore, since $P = (P_1, u_i, P_2)$ is a normal path by assumption, the first vertex of $P_2$ is the leftmost vertex of $V(P_2)$ in $\sigma$. Consider now any two consecutive vertices $u_i, u_j$ of $P_2$. Then, since $P = (P_1, u_i, P_2)$ is a normal path, it follows that $u_j$ is the leftmost unvisited vertex of $N(u_i) \cap V(P)$ (resp. of $N(u_j) \cap V(P)$) in $\sigma$, when $P$ (resp. $P_2$) visits $u_i$. Therefore, since also $u_j$ is the last vertex of $P$ by assumption, $P_2$ is a normal path of $V(G(x+1,j-1))$ with $u_j$ as its last vertex.

Recall that $u_i \in V(G(i,j))$, since $P = (P_1, u_i, P_2)$ is a path of $G(i,j)$ by assumption. Suppose now that there exists a normal path $P' \subseteq V(P)$ (resp. $P'_{j'} \subseteq V(G(i,j))$) with $u_i$ as its last vertex, such that $x \neq j - 1$ and $|P'_{j'}| > |P|$. Then, $P'$ is a normal path of $G(i,j)$ with $u_i$ as its last vertex, such that $|P'_{j'}| > |P|$. This is a contradiction to the assumption that $P$ is a longest normal path of $G(i,j)$ with $u_i$ as its last vertex. Therefore, there exists no such path $P'_{j'}$ (resp. $P'_{j''}$), and thus $P_1$ (resp. $P_2$) is a longest normal path of $G(i,j-1)$ (resp. of $G(x+1,j-1)$) with $u_j$ (resp. with $u_j$) as its last vertex. This completes the proof of the lemma.$\blacksquare$

Now, following exactly the same proof as in Lemma 9, where we replace $G(i,j)$ by $G(i,j)$, we obtain the next lemma.

**Lemma 10.** Let $G = (V, E)$ be a circular-arc graph and $\pi = (u_0, u_1, \ldots, u_{n-1})$ be a right-end circular ordering of $G$. Let $i \neq j$ be two indices and $\sigma$ be the right-end ordering of $G(i,j)$ induced by $\pi$. Let $P$ be a longest normal path of $G(i,j)$ with $u_i \neq u_j$ as its last vertex and let $P = (P_1, u_i, P_2)$. Let $u_i$ be the last vertex of $P_1$. Then $P_1$ is a longest normal path of $G(i,j)$ with $u_i$ as its last vertex and $P_2$ is a longest normal path of $G(x+1,j-1)$ with $u_j$ as its last vertex.

**4.1. The algorithm**

In the following we present our Algorithm 1 that computes a longest path of a given circular-arc graph $G = (V,E)$. The same algorithm also computes the number $N$ of all different normal paths of $G$. This number $N$ constitutes an $n$-approximation of the number of different vertex sets $S \subseteq V$ such that $V(P) = S$ for a longest path $P$ of $G$, while the number of such sets $S$ is exponential in the worst case. In the case where the input graph $G$ is an interval graph, a slight modification of the algorithm allows us to compute the exact number of different vertex sets $S$ of longest path of $G$. For simplicity of the presentation of this algorithm, we make the following two conventions.
Algorithm 1 Computing and counting longest (normal) paths of a circular-arc graph

Input: A circular-arc graph $G = (V, E)$ with $|V| = n$, a circular-arc representation $R$ of $G$, and a right-end circular ordering $\pi = (u_0, u_1, \ldots, u_{n-1})$ of $G$ (in $R$)

Output: A longest path $P$ of $G$ and the number $N$ of different longest normal paths of $G$ in $R$

1: for $i = 0$ to $n - 1$ do
2: for $i = 0$ to $n - 1$ do
3: $j \leftarrow i + t \mod n$
4: Execute Procedure 1 (initialization)
5: if $j \neq i$ then
6: Execute Procedure 2 \{update phase for the graph $G(i)$\}
7: if $j \neq i - 1 \mod n$ then
8: Execute Procedure 3 \{update phase for the graph $G(i, j)$\}
9: Compute a path $P = P_i(u_k; i - 1)$ with $|P| = \max\{\ell_i(u_k; i - 1) \mid u_k \in V, i \in \{0, 1, \ldots, n - 1\}\}$
10: $N \leftarrow 0$
11: for $i = 0$ to $n - 1$ do
12: for every $u_k \in V$ do
13: if $\ell_i(u_k; i - 1) = |P|$ then
14: $N \leftarrow N + N''(u_k; i - 1)$
15: return the path $P$ and the number $N$

Notation 3. Let $G = (V, E)$ be a circular-arc graph, $\pi = (u_0, u_1, \ldots, u_{n-1})$ be a right-end circular ordering of $G$, and $i, j \in \{0, 1, \ldots, n - 1\}$ be two indices. Then, for every vertex $u_k \in V(G(i, j))$, we denote by $P_i(u_k; j)$ a longest normal path of $G_i(j)$ with $u_k$ as its last vertex and by $\ell_i(u_k; j)$ the length $|P_i(u_k; j)|$ of $P_i(u_k; j)$, i.e. the number of vertices of $P_i(u_k; j)$. Similarly, if $j \neq i - 1 \mod n$, then for every vertex $u_k \in V(G(i, j))$, we denote by $P(u_k; i, j)$ a longest normal path of $G(i, j)$ with $u_k$ as its last vertex and by $\ell(u_k; i, j)$ the length $|P(u_k; i, j)|$ of $P(u_k; i, j)$.

Notation 4. Let $G = (V, E)$ be a circular-arc graph, $\pi = (u_0, u_1, \ldots, u_{n-1})$ be a right-end circular ordering of $G$, and $i, j \in \{0, 1, \ldots, n - 1\}$ be two indices. Then, for every vertex $u_k \in V(G(i, j))$, we denote by $\ell_i(u_k; j)$ (resp. by $\ell''(u_k; j)$) the number of normal paths of $G_i(j)$ with $u_k$ as the last vertex, such that $|P| = \ell_i(u_k; j)$ and $P$ does not include (resp. includes) vertex $u_k$. Similarly, if $j \neq i - 1 \mod n$, then for every vertex $u_k \in V(G(i, j))$, we denote by $N''(u_k; i, j)$ (resp. by $N''(u_k; i, j)$) the number of normal paths of $G(i, j)$ with $u_k$ as the last vertex, such that $|P| = \ell_i(u_k; i, j)$ and $P$ does not include (resp. includes) vertex $u_k$. Furthermore, we denote $N_i(u_k; j) = N'_i(u_k; j) + N''(u_k; j)$ (resp. $N_i(u_k; j) = N''(u_k; i, j)$), i.e. $N_i(u_k; j)$ is the number of normal paths of $P_i(j)$ (resp. of $G(i, j)$) with $u_k$ as the last vertex, such that $|P| = \ell_i(u_k; j)$ (resp. $|P| = \ell''(u_k; j)$).

The next two main theorems of this section prove that, given a circular-arc representation $R$ of a circular-arc graph $G = (V, E)$ with $n$ vertices, Algorithm 1 computes in $O(n^4)$ time a longest path $P$ of $G$, as well as the number $N$ of different longest normal paths of $G$ in $R$. Moreover, this number is an $n$-approximation of the number of different sets $S \subseteq V$, such that $V(P) = S$ for some longest path $P$ of $G$; cf. Theorem 3.

Theorem 3. Let $G = (V, E)$ be a circular-arc graph with $n$ vertices and $\pi$ be a right-end circular ordering of $G$. Algorithm 1 computes a longest path $P$ of $G$ and the number $N$ of different longest normal paths of $G$ in $O(n^4)$ time.

Proof. Let $R$ be a circular-arc representation of $G$ and $\pi = (u_0, u_1, \ldots, u_{n-1})$ be the right-end circular ordering of $G$ (in the representation $R$). Let now $P$ be a longest path of $G$. Recall by Theorem 2 that there exists a normal path $P'$ of $G$ on the same vertices as $P$. That is, there exists by Definition 6 a vertex $u_k \in V$, such that $P'$ is a normal path of the interval graph $G_{u_k}$, where $G_{u_k} = G_i(t - 1)$ by Observation 6. Therefore, in order to compute a longest path of $G$, it suffices to compute for every $i = 0, 1, \ldots, n - 1$ a longest normal path of $G_i(t - 1)$ (with respect to the ordering $\pi$ of $G_i(t - 1)$ induced by $\pi$), i.e. a longest path among the normal ones in $G_i(t - 1)$.

In lines 1–8, Algorithm 1 iterates for every pair of indices $t \in \{0, 1, \ldots, n - 1\}$. During these iterations, it computes a path $P_i(u_k; i + t)$ and four values $\ell_i(u_k; i + t)$, $N''(u_k; i + t)$, $N''(u_k; i + t)$, and $N_i(u_k; i + t)$ for every vertex $u_k \in V(G_i(i + t))$. Furthermore, if $t \neq n - 1$ (i.e. if $i + t \neq i - 1 \mod n$), it computes also a path $P_i(u_k; i + t)$ and four values $\ell_i(u_k; i + t)$, $N''(u_k; i + t)$, $N''(u_k; i + t)$, and $N_i(u_k; i + t)$ for every vertex $u_k \in V(G_i(i + t))$. Note that at every iteration, i.e. for all pairs $i, k$ of indices, we denote for simplicity reasons in line 3 of Algorithm 1 the index $i + t$ by $j$ in the corresponding iteration. At every iteration, the algorithm calls the Procedures 1, 2, and 3. During the call of Procedure 1, the algorithm initializes the values $P_i(u_k; i + t)$, $\ell_i(u_k; i + t)$, $N''(u_k; i + t)$, $N''(u_k; i + t)$, and $N_i(u_k; i + t)$ (resp. $P_i(u_k; i + t)$, $\ell_i(u_k; i + t)$, $N''(u_k; i + t)$, $N''(u_k; i + t)$, and $N_i(u_k; i + t)$), for every possible vertex $u_k \in V(G_i(i + t))$. 


Procedure 1 Initialization phase for Algorithm 1

1: \( P(0; i, j) \leftarrow (u_i); \ell(0; i, j) \leftarrow 1 \)
2: \( N'(0; i, j) \leftarrow 0; N'(u_i; i, j) \leftarrow 1 \)
3: \( N(0; i, j) \leftarrow 1 \)

4: if \( u_j \in V(G(i, j)) \) then
5: \( P(u_j; i, j) \leftarrow (u_i); \ell(u_j; i, j) \leftarrow 1 \)
6: \( N'(u_j; i, j) \leftarrow 0; N'(u_j; i, j) \leftarrow 1 \)
7: \( N(u_j; i, j) \leftarrow 1 \)

8: if \( j \neq i \) then
9: for every \( u_y \in V(G(j - 1)) \) do
10: \( P(u_y; i, j) \leftarrow P(u_y; i, j - 1); \ell(u_y; i, j) \leftarrow \ell(u_y; i, j - 1) \)
11: \( N'(u_y; i, j) \leftarrow N(u_y; i, j - 1); N'(u_y; i, j) \leftarrow 0 \)
12: \( N(u_y; i, j) \leftarrow N(u_y; i, j - 1) \)
13: if \( j \neq i - 1 \mod n \) then
14: for every \( u_y \in V(G(i, j - 1)) \) do
15: \( P(u_y; i, j) \leftarrow P(u_y; i, j - 1); \ell(u_y; i, j) \leftarrow \ell(u_y; i, j - 1) \)
16: \( N'(u_y; i, j) \leftarrow N(u_y; i, j - 1); N'(u_y; i, j) \leftarrow 0 \)
17: \( N(u_y; i, j) \leftarrow N(u_y; i, j - 1) \)

We will prove by induction on \( t \) that \( P(u_k; i, t) \) (resp. \( P(u_k; i, i + t) \)) is indeed a longest normal path of \( G(i + t) \) (resp. \( G(i, i + t) \)) with \( u_k \) as its last vertex and that \( \ell(u_k; i, t) = |P(u_k; i, t)| \) (resp. \( \ell(u_k; i, i + t) = |P(u_k; i, i + t)| \)). Furthermore, we will also prove by the same induction on \( t \) that the values \( N'_k(u_k; i, t) \), \( N'(u_k; i, i + t) \), \( N(u_k; i, i + t) \) are the correct values according to Notation 4.

Induction basis. Let \( t = 0 \). In this case \( j = i \) (cf. line 3 of Algorithm 1), and thus only lines 1–7 of Procedure 1 are executed, while Procedures 2 and 3 are not executed at all (cf. line 5 of Algorithm 1). In line 1 of Procedure 1, the algorithm computes the path \( P(u_j; i, j) = (u_j) \), which is clearly the only (and thus also the longest) normal path of \( G(j, j) \) with \( u_j \) as its last vertex. Similarly, in the case where \( u_j \in V(G(j, i)) \), i.e. in the case where \( r_{u_j} \neq I_{u_j} \) in the circular-arc representation \( R \) of \( G \) (cf. Definition 5), the algorithm computes in line 5 of Procedure 1 the path \( P(u_j; j, j) = (u_j) \), which is the only (and thus also the longest) normal path of \( G(j, j) \) with \( u_j \) as its last vertex. Therefore, in this case the algorithm also correctly computes in lines 2–3 (resp. in lines 6–7) of Procedure 1 the values \( N'(u_j; j) = 0 \) and \( N'(u_j; j) = N'(u_j; j) = 1 \) (resp. \( N'(u_j; j) = 0 \) and \( N'(u_j; j) = N'(u_j; j) = 1 \)). This proves the induction basis.

Induction hypothesis. Let \( t \geq 1 \). In this case \( j \neq i \). For every vertex \( u_k \in V(G(j - 1)) \) (resp. \( u_k \in V(G(i, j - 1)) \)), the induction hypothesis implies that \( P(u_k; j - 1) \) (resp. \( P(u_k; i, j - 1) \)) is a longest normal path of \( G(j - 1) \) (resp. \( G(i, j - 1) \)) with \( u_k \) as its last vertex and that \( \ell(u_k; j - 1) = |P(u_k; j - 1)| \) (resp. \( \ell(u_k; i, j - 1) = |P(u_k; i, j - 1)| \)). Furthermore, for every vertex \( u_k \in V(G(j - 1)) \), the value \( N'(u_k; j - 1) \) (resp. \( N'(u_k; i, j - 1) \)) equals the number of normal paths \( P(u_k; j - 1) \) with \( u_k \) as the last vertex, such that \( |P| = \ell(u_k; j - 1) \) and \( P \) does not include (resp. includes) vertex \( u_j \). Similarly, for every vertex \( u_k \in V(G(i, j - 1)) \), the value \( N'(u_k; i, j - 1) \) (resp. \( N'(u_k; i, j - 1) \)) equals the number of normal paths \( P(u_k; j - 1) \) with \( u_k \) as the last vertex, such that \( |P| = \ell(u_k; i, j - 1) \) and \( P \) does not include (resp. includes) vertex \( u_j \).

Induction step. Let \( t \geq 1 \). In this case \( j \neq i \). Consider the iteration of the algorithm for any \( i \in \{0, 1, \ldots, n - 1\} \). First, the algorithm initializes in Procedure 1 the values \( P(u_j; j) \) and \( \ell(u_j; j) \) (resp. \( P(u_j; i, j) \) and \( \ell(u_j; i, j) \)), as well as the values \( N'(u_j; j) \), \( N'(u_j; i, j) \), and \( N(u_j; i, j) \) (resp. \( N'(u_i; i, j) \), \( N'(u_i; i, j) \), and \( N(u_i; i, j) \)), for every \( u_j \in V(G(j)) \) (resp. \( u_k \in V(G(j)) \)). Then, the algorithm updates these values if necessary in Procedure 2 (resp. Procedure 3). In order to prove the induction step, we distinguish in the following the cases where the last vertex of the computed path is \( u_j \), or some vertex \( u_k \in V(G(j - 1)) \) (resp. \( u_k \in V(G(i, j - 1)) \)).

Case 1. The last vertex of the computed path is \( u_j \). The algorithm initializes in line 1 of Procedure 1 the values \( P(u_j; j) = (u_j) \) and \( \ell(u_j; j) = 1 \). Regarding the number of longest paths, it initializes in lines 2–3 of Procedure 1 the values \( N'(u_j; j) = 0 \) and \( N(u_j; j) = N'(u_j; j) = 1 \). Moreover, in the case where \( u_j \in V(G(i, j)) \), the algorithm initializes in line 5 of Procedure 1 the values \( P(u_j; i, j) = (u_j) \) and \( \ell(u_j; i, j) = 1 \). Regarding the number of longest paths, it initializes in lines 6–7 of Procedure 1 the values \( N'(u_j; i, j) = 0 \) and \( N(u_j; i, j) = N'(u_j; i, j) = 1 \). Otherwise, in the case where \( u_j \notin V(G(i, j)) \), the algorithm does not execute lines 5–7 of Procedure 1, since in this case neither the values \( P(u_j; i, j) \) and \( \ell(u_j; i, j) \), nor the values \( N'(u_j; i, j) \), \( N'(u_j; i, j) \), and \( N(u_j; i, j) \) can be defined (cf. Notations 3 and 4).

Recall by Observation 8 that for any normal path \( P_i \) of \( G(j - 1) \) with a vertex \( u_j \) as its last vertex, such that \( r_{u_j} \in I_{u_j} \) in \( R \), the path \( (P_i, u_j) \) is a normal path of \( G(j) \). Conversely, recall by Observation 9 that the path \( (P_i, u_j) \setminus \{u_j\} \) (if not empty) is a normal path of \( G(j - 1) \). Therefore, in order to update the value of \( P(u_j; j) \), the algorithm correctly computes in lines 3–9 of Procedure 2 the paths \( (P_i, u_j; j - 1) \), \( u_j \) for every \( u_k \in V(G(i, j - 1)) \), such that \( r_{u_k} \in I_{u_k} \) in \( R \), and keeps the longest of them.
Procedure 2 Update phase for the values of the graph $G_i(j)$

1: for every $u_s \in V(G_i(j - 1))$ do
2:   if $r_{us} \in I_y$ in $R$ then
3:     if $\ell_i(u_s; j) < \ell_i(u_s; j - 1) + 1$ then
4:       $P_i(u_s; j) \leftarrow (P_i(u_s; j - 1), u_j)$
5:       $\ell_i(u_s; j) \leftarrow \ell_i(u_s; j - 1) + 1$
6:     $N''_i(u_s; j) \leftarrow N_i(u_s; j - 1)$
7:     if $\ell_i(u_s; j) = \ell_i(u_s; j - 1) + 1$ then
8:       $N''_i(u_s; j) \leftarrow N''_i(u_s; j) + N_i(u_s; j - 1)$
9:   $N_i(u_s; j) \leftarrow N''_i(u_s; j)$
10:  for every $u_y \in V(G(x + 1, j - 1))$ do
11:     if $\ell_i(u_y; j) < \ell_i(u_y; j - 1) + \ell_i(u_y; x + 1, j - 1) + 1$ then
12:       $P_i(u_y; j) \leftarrow (P_i(u_y; j - 1), u_j)$
13:       $\ell_i(u_y; j) \leftarrow \ell_i(u_y; j - 1) + 1$
14:       $N'_i(u_y; j) \leftarrow 0; N''_i(u_y; j) \leftarrow N_i(u_y; j - 1) \cdot N_i(u_y; x + 1, j - 1)$
15:     if $\ell_i(u_y; j) = \ell_i(u_y; j - 1) + \ell_i(u_y; x + 1, j - 1) + 1$ then
16:       $N''_i(u_y; j) \leftarrow N''_i(u_y; j) + N_i(u_y; j - 1) \cdot N_i(u_y; x + 1, j - 1)$
17:     $N_i(u_y; j) \leftarrow N''_i(u_y; j)$

Procedure 3 Update phase for the graph $G(i, j)$

1: for every $u_s \in V(G(i, j - 1))$ do
2:   if $u_s \in V(G(i, j))$ and $r_{us} \in I_y$ in $R$ then
3:     if $\ell_i(u_s; i, j) < \ell_i(u_s; i, j - 1) + 1$ then
4:       $P_i(u_s; i, j) \leftarrow (P_i(u_s; i, j - 1), u_j)$
5:       $\ell_i(u_s; i, j) \leftarrow \ell_i(u_s; i, j - 1) + 1$
6:     $N''_i(u_s; i, j) \leftarrow N_i(u_s; i, j - 1)$
7:     if $\ell_i(u_s; i, j) = \ell_i(u_s; i, j - 1) + 1$ then
8:       $N''_i(u_s; i, j) \leftarrow N''_i(u_s; i, j) + N_i(u_s; i, j - 1)$
9:     $N_i(u_s; i, j) \leftarrow N''_i(u_s; i, j)$
10:  for every $u_y \in V(G(x + 1, j - 1))$ do
11:     if $\ell_i(u_y; i, j) < \ell_i(u_y; i, j - 1) + \ell_i(u_y; x + 1, j - 1) + 1$ then
12:       $P_i(u_y; i, j) \leftarrow (P_i(u_y; i, j - 1), u_j)$
13:       $\ell_i(u_y; i, j) \leftarrow \ell_i(u_y; i, j - 1) + 1$
14:       $N'_i(u_y; i, j) \leftarrow 0; N''_i(u_y; i, j) \leftarrow N_i(u_y; i, j - 1) \cdot N_i(u_y; x + 1, j - 1)$
15:     if $\ell_i(u_y; i, j) = \ell_i(u_y; i, j - 1) + \ell_i(u_y; x + 1, j - 1) + 1$ then
16:       $N''_i(u_y; i, j) \leftarrow N''_i(u_y; i, j) + N_i(u_y; i, j - 1) \cdot N_i(u_y; x + 1, j - 1)$
17:     $N_i(u_y; i, j) \leftarrow N''_i(u_y; i, j) + N''_i(u_y; i, j)$

Suppose now that $u_y \in V(G(i, j))$; then the path $P_i(u_y; i, j)$ is well defined (cf. Notation 3). Then, it follows similarly by Observations 8 and 9 that, in order to update the value of $P_i(u_y; i, j)$, the algorithm correctly computes in lines 3–9 of Procedure 3 the paths $(P_i(u_s; i, j - 1), u_y)$ for every $u_s \in V(G(i, j - 1))$, such that $r_{us} \in I_y$ in $R$, and keeps the longest of them.

Regarding the number of longest paths, consider the execution of lines 3–9 of Procedure 2 (resp. of lines 3–9 of Procedure 3) for a vertex $u_s$. Suppose that the algorithm computes at this iteration the path $(P_i(u_s; j - 1), u_j)$ (resp. $(P_i(u_s; i, j - 1), u_i)$) that has greater length than the actual value of $P_i(u_s; j)$ (resp. of $P_i(u_s; i, j)$); cf. line 3 of Procedure 2 (resp. line 3 of Procedure 3). Recall by the induction hypothesis there exist exactly $N_i(u_s; j - 1)$ (resp. $N_i(u_s; i, j - 1)$) normal paths $P_i(u_s; j)$ (resp. $G_i(j)$) with $u_i$ as the last vertex, such that $|P| = |P_i(u_s; j - 1)|$ (resp. $|P| = |P_i(u_s; i, j - 1)|$). Then, the algorithm correctly sets $N''_i(u_s; j)$ (resp. $N''_i(u_s; i, j)$) to be equal to $N_i(u_s; j - 1)$ (resp. $N_i(u_s; i, j - 1)$); cf. line 6 of Procedure 2 (resp. line 6 of Procedure 3).

Suppose now that the algorithm computes the path $(P_i(u_s; j - 1), u_j)$ (resp. $(P_i(u_s; i, j - 1), u_i)$) that has the same length as the actual value of $P_i(u_s; j)$ (resp. of $P_i(u_s; i, j)$); cf. line 7 of Procedure 2 (resp. line 7 of Procedure 3). Then, since no longer path has been found, the algorithm correctly increases the value of $N''_i(u_s; j)$ by $N_i(u_s; j - 1)$ (resp. increases the value of $N''_i(u_s; i, j)$ by $N_i(u_s; i, j - 1)$); cf. line 8 of Procedure 2 (resp. line 8 of Procedure 3). Note that always $N''_i(u_s; j) = 0$ (resp. $N''_i(u_s; i, j) = 0$) by Notation 4. Therefore, the algorithm correctly updates $N_i(u_s; j)$ (resp. $N_i(u_s; i, j)$) to be equal to $N''_i(u_s; j)$ (resp. $N''_i(u_s; i, j)$) in line 9 of Procedure 2 (resp. in line 9 of Procedure 3).
Case 2. The last vertex of the computed path is some vertex \( u_p \in V(G(i, j - 1)) \) (resp. \( u_p \in V(G(i, j - 1)) \)). The algorithm initializes in line 10 (resp. line 15) of Procedure 1 for every \( u_p \in V(G(i, j - 1)) \) (resp. every \( u_p \in V(G(i, j - 1)) \)), where \( j \neq i - 1 \mod n \) the values \( P(u_p, j) \) and \( \ell(u_p, j) \) (resp. \( P(u_p, i, j) \) and \( \ell(u_p, i, j) \)) as \( P(u_p, j - 1) \) and \( \ell(u_p, j - 1) \) (resp. \( P(u_p, i, j - 1) \) and \( \ell(u_p, i, j - 1) \)). Regarding the number of longest paths, it initializes in lines 11–12 (resp. in lines 16–17) of Procedure 1 the values \( N(1, j, j - 1) = N'(1, j, j - 1) \) and \( N''(1, j, j - 1) = 0 \) (resp. the values \( N(u_i, i, j) = N'(u_i, i, j) \) and \( N''(u_i, i, j) = 0 \)). Note now by Definition 5 that \( G(i, j - 1) = G(i, j) \backslash \{ u_i \} \). Furthermore, recall by Observation 6 that \( G(i, j - 1) = G(i, j) \backslash \{ u_i \} \). Therefore, due to the induction hypothesis, for every \( u_p \in V(G(i, j - 1)) \) (resp. \( u_p \in V(G(i, j - 1)) \)) the value \( \ell(u_p, j - 1) \) (resp. \( \ell(u_p, i, j - 1) \)) is the greatest length among the normal paths \( P(u_p) \) (resp. \( G(i, j) \)) with \( u_p \) as the last vertex, such that \( P \) does not include \( u_p \).

The algorithm executes lines 11–17 of Procedure 2 for every \( u_p \in V(G(i, j - 1)) \), where \( r_{u_p} \in I_{u_p} \) in \( R \) and for every \( u_p \in V(G(x + 1, j - 1)) \), cf lines 1, 2, and 10 of Procedure 2. Moreover, the algorithm executes lines 11–17 of Procedure 3 for every \( u_p \in V(G(i, j - 1)) \), cf lines 1, 2, and 10 of Procedure 3. For such a pair of vertices \( u_p, u_p \), recall by Lemma 8 (resp. by Lemma 7) that \( P(u_p, u_p, x + 1, j - 1) \) (resp. \( P(u_p, i, j - 1, u_p, P(u_p, x + 1, j - 1)) \)) is a normal path of \( G(i, j) \) (resp. \( G(i, j) \)) with \( u_p \) as its last vertex. Conversely, let \( P \) be a normal path of \( G(i, j) \) (resp. \( G(i, j) \)) with \( u_p \neq u_p \) as its last vertex, let \( P = (P_1, u_p, P_2) \), and let \( u_p \) be the last vertex of \( P_1 \). Then Lemma 10 (resp. Lemma 9) implies that \( P_1 = P(u_p, j - 1) \) (resp. \( P_1 = P(u_p, i, j - 1) \)) and \( P_2 = P(u_p, x + 1, j - 1) \). Therefore, the algorithm correctly computes during the multiple executions of lines 11–17 of Procedure 2 (resp. lines 11–17 of Procedure 3) the greatest length \( \ell \) of a normal path \( P(u_p) \) (resp. \( G(i, j) \)) with \( u_p \) as its last vertex, such that \( P \) includes \( u_p \). If at least one of these paths has greater length than the initial value \( \ell(u_p, j - 1) \) (resp. \( \ell(u_p, i, j - 1) \)) that has been computed in line 10 (resp. in line 15) of Procedure 1, the algorithm correctly keeps in \( P(u_p, j - 1) \) (resp. \( P(u_p, i, j - 1) \)) the longest among these paths.

Regarding the number of longest paths, consider the execution of lines 11–17 of Procedure 2 (resp. lines 11–17 of Procedure 3) for a pair of vertices \( u_p, u_p \). Suppose that the algorithm computes at this iteration the path \( P(u_p, i, j - 1, u_p, P(u_p, x + 1, j - 1)) \) (resp. \( P(u_p, i, j - 1, u_p, P(u_p, x + 1, j - 1)) \)) that has greater length than the actual value of \( P(u_p, j - 1) \) (resp. \( P(u_p, i, j - 1) \)).

Then, since no longer path has been found, the algorithm correctly increases the value of \( N''(u_p, i, j - 1) \) (resp. \( N''(u_p, i, j - 1) \)). Furthermore, there exist by the induction hypothesis exactly \( N(u_p, i, j - 1) \) normal paths \( P(u_p) \) (resp. \( G(i, j) \)) with \( u_p \) as its last vertex, such that \( |P| = \ell(u_p, i, j - 1) \) (resp. \( |P| = \ell(u_p, i, j - 1) \)). Therefore, in case a longer path has been found, the algorithm correctly sets \( N''(u_p, i, j - 1) \) (resp. \( N''(u_p, i, j - 1) \)) equal to \( N(u_p, i, j - 1) \cdot N(u_p, x + 1, j - 1) \) (resp. \( N(u_p, i, j - 1) \cdot N(u_p, x + 1, j - 1) \)).

In case a longer path has been found, finally, the algorithm correctly updates \( N(u_p, i, j) \) (resp. \( N(u_p, i, j) \)) as the sum of \( N''(u_p, i, j) \) and \( N''(u_p, i, j) \) in line 17 of Procedure 2 (resp. in line 17 of Procedure 3). This completes the induction step.

Therefore, for every pair of indices \( i, j \in \{0, 1, \ldots, n - 1\} \) such that \( G(i, j) \neq \emptyset \) (resp. \( G(i, j) \neq \emptyset \)) and every \( u_k \in V(G(i, j)) \) (resp. \( u_k \in V(G(i, j)) \)), the algorithm correctly computes after the execution of lines 1–8 of Algorithm 1 a longest normal path \( P(u_k, i, i + t) \) (resp. \( P(u_k, i, i + t) \)) of \( G(i, i + t) \) with \( u_k \) as its last vertex and its length \( \ell(u_k, i, i + t) = |P(u_k, i, i + t)| \) (resp. \( \ell(u_k, i, i + t) = |P(u_k, i, i + t)| \)).

Moreover, for every index \( i \in \{0, 1, \ldots, n - 1\} \) and every vertex \( u_i \in V(G(1, i)) \), Algorithm 1 correctly computes the number \( N(u_i, i - 1) \) of normal paths \( G(i - 1) \) with \( u_k \) as its last vertex, such that \( |Q| = \ell(u_i, i - 1) \) and \( Q \) includes vertex \( u_i \).

We will now prove that \( N \) is exactly the number of different longest normal paths of \( G \) (with respect to the circular right-end ordering \( \pi \) of \( G \)). To this end, it suffices to consider only the non-trivial case, where the longest path \( G \) of \( P \) has at least two vertices. Suppose otherwise that there exists a longest normal path \( G \) of \( G \) that corresponds to the numbers \( N''(u_i, i - 1) \) and \( N''(u_i, j - 1) \), for some \( i, j \in \{0, 1, \ldots, n - 1\} \) and \( u_k \neq u_p \in V(G(i, j - 1)) \). Let \( \sigma_1, \sigma_2 \) be the right-end ordering of \( G(i - 1) \) (resp. \( G(j - 1) \)) induced by \( \pi \). Note by definition of \( N''(u_i, i - 1) \) and \( N''(u_i, j - 1) \) that \( u_i, u_i \in V(P) \). Then, since the reverse path \( P \) of \( P \) is considered to be different from \( P \) itself, it follows that \( u_k = u_p \) (i.e. \( u_k = u_p \) is the last vertex of \( P \)). Therefore \( \pi \neq \pi \), since we assumed that \( i \neq j \) or \( k \neq p \). Let \( u_k \) be the first vertex of \( P \).

Then, since \( P \) is a normal path, \( u_k \) is the leftmost vertex of \( V(P) \) in both \( \sigma_1 \) and \( \sigma_2 \). Furthermore, since \( u_i \in V(P) \) (resp. \( u_i \in V(P) \)), it follows that \( u_i \) (resp. \( u_i \)) is the rightmost vertex of \( V(P) \) in \( \sigma_1 \) (resp. \( \sigma_2 \)). Therefore, \( u_i \) (resp. \( u_i \)) is the
Proof. Lemma 11. There exists a circular-arc graph $G$ such that $V(P) = S$ for some longest path $P$ of $G$.

Running time. Regarding the running time of Algorithm 1, let us first discuss some implementation details. To avoid the search of the table indicated in line 9, the length and location of the current longest path would be maintained throughout the algorithm. Furthermore, following standard dynamic programming techniques, we do not need to store the path itself, but rather, an indication of how the path is built. In particular, each of lines 1, 5, 10, and 15 of Procedure 1, lines 4 and 12 of Procedure 2, and lines 4 and 12 of Procedure 3, gives “instructions” on how to build the current longest path using information that has already been computed. At the end of the algorithm a simple recursive unwinding of these “instructions” yields a longest path in the given graph. Now, Procedure 1, 2, and 3 are executed at most $O(n^2)$ times each. In Procedure 1, each line lies at most in one loop of $O(n)$ iterations each. Furthermore, in Procedure 2 and 3, each line lies at most in two loops of $O(n)$ iterations each. Finally, line 14 of Algorithm 1 lies in two loops of $O(n)$ iterations each. Therefore, following the implementation details described above, the total running time of Algorithm 1 is $O(n^3)$. This completes the proof of the theorem. \hfill \Box

Theorem 4. Let $G = (V, E)$ be a circular-arc graph with $n$ vertices and $\pi$ be a right-end circular ordering of $G$. Then, the number $N$ computed by Algorithm 1 is an $n$-approximation of the number of different sets $S \subseteq V$, such that $V(P) = S$ for some longest path $P$ of $G$. Furthermore, if $G$ is an interval graph, then the exact number of such different sets $S \subseteq V$ can be computed in $O(n^3)$ time.

Proof. Denote by $\pi = (u_0, u_1, \ldots, u_{n-1})$ the right-end circular ordering of $G$. Let $P$ be a longest path of $G$. Then, we may assume without loss of generality by Theorem 2 that $P$ is a normal path of the circular-arc graph $G$. That is, there exists by Definition 6 an index $i \in \{0, 1, \ldots, n-1\}$, such that $P$ is a normal path of the interval graph $G_i(i - 1)$. Let $V(P) = S$ and $\sigma_i$ be the right-end ordering of $G_i(i - 1)$ induced by $\pi$. It is now easy to see by Definition 4 that the vertex set $S$ determines exactly one normal path $P$ in $\sigma_i$. Therefore, a specific set $S \subseteq V$ may determine at most $n$ different longest normal paths of $G$, one for every index $i \in \{0, 1, \ldots, n-1\}$. Thus, the computed number $N$ computed by Algorithm 1 equals at most $n$ times the number of different sets $S \subseteq V$, such that $V(P) = S$ for some longest path $P$ of $G$.

Suppose now that the input graph $G$ is an interval graph. Then, the right-end circular ordering $\pi = (u_0, u_1, \ldots, u_{n-1})$ of $G$ is actually a right-end ordering, since $G$ is an interval graph. We compute a longest path $P$ of $G$ by executing lines 1–9 of Algorithm 1 (since every interval graph is also a circular-arc graph). Then, instead of executing lines 10–14 of Algorithm 1, we compute the number $N$ by summing up the values $N_0(u_k; n - 1)$, where $0 \leq k \leq n - 1$, such that $\ell_0(u_k; n - 1) = |P|$, where $P$ is a longest path of $G$. That is, $N$ equals the number of different longest normal paths of $G_0(n - 1) = G$. Let $Q$ be such a longest normal path of $G$ and let $V(Q) = S$. Then, it follows similarly to the previous paragraph that the vertex set $S$ determines exactly one normal path $Q$ in $\pi$. Thus, if $G$ is an interval graph, $N$ is the exact number of different sets $S \subseteq V$, such that $V(P) = S$ for some longest path $P$ of $G$. This completes the proof of the theorem. \hfill \Box

The bound of $n$ of Theorem 4 for the approximation ratio of Algorithm 1 is tight. For instance, let the circular-arc graph $G = (V, E)$ be an induced circle with $n$ vertices. The algorithm will return $N = n$, one for each Hamiltonian path of $G$, while $S = V$ is the only set of vertices that provides a longest path of $G$. Moreover, as the following lemma states, there can be exponentially many such different sets $S \subseteq V$ in the worst case, as it is illustrated in the example of Fig. 3.

Lemma 11. There exists a circular-arc graph $G = (V, E)$ with $n$ vertices, such that there exist $2^{O(n)}$ sets $S \subseteq V$, such that $V(P) = S$ for a longest path $P$ of $G$.

Proof. Consider the circular-arc graph $G = (V, E)$ of the example of Fig. 3 with $n = 3k$ vertices $V = \{a_0, a_1, \ldots, a_{k-1}, b_0, b_1, \ldots, b_{k-1}, c_0, c_1, \ldots, c_{k-1}\}$. Any longest path of $G$ has $2k$ vertices; it includes all vertices $\{a_0, a_1, \ldots, a_{k-1}\}$ and exactly one vertex of each pair $\{b_i, c_i\}$. Since there are $k = O(n)$ such sets $\{b_i, c_i\}$, it follows that there exist $2^k = 2^{O(n)}$ different
sets $S \subseteq V$, such that $V(P) = S$ for a longest path $P$ of $G$. Finally, it can be easily seen that $G$ is a circular-arc graph: a circular-arc representation of $G$ has $2k$ “short” arcs and $k$ “long” arcs, where the long arcs correspond to $\{a_0, a_1, \ldots, a_{k-1}\}$ and the short ones correspond to $\{b_0, b_1, \ldots, b_{k-1}\} \cup \{c_0, c_1, \ldots, c_{k-1}\}$. The long arc of $a_i$ intersects with the next and the previous long arc of $a_{i-1}$ and $a_{i+1}$ on the circle, respectively. Furthermore, no two short arcs intersect to each other, while the pair of short arcs of $b_i$ and $c_i$ are intersecting with $a_i$ and $a_{i+1}$. Note that all indices of the vertices are computed mod $n$.

References