A linear-time Algorithm for Paired Neighbhourhood Set on Circular-Arc Graphs

Saranya C. R.*[1], K. S. P. Sowndarya[2], Y. Lakshmi Naidu[2]

[1] Department of Mathematics, Smt. Eswaramma High School, Prasanthinilayam, A.P. (515134) India
[2] Department of Mathematics, Sri Sathya Sai Institute of Higher Learning, Anantapur Campus, A.P. (515001) India

Abstract: *The basic difference between a dominating set of a graph and neighbourhood set of a graph is, dominating set covers the vertices of the graph G, but may not cover all edges; whereas a neighbourhood set covers the entire graph. In this paper we provide an algorithm to find minimum paired neighbhourhood set of circular arc graphs. Given a circular-arc graph G with end points sorted, we give an $O(n)$ - time algorithm, where n denote the number of vertices. To the best of our knowledge, this algorithm is first of its kind related to finding minimum paired neighbhourhood set on circular-arc graphs.*

Keywords: *Algorithm, paired neighbourhood set, circular-arc graph.*

I. INTRODUCTION

Let $G(V, E)$ be a graph. The closed neighbhourhood of a vertex $v \in V$ in G is defined as the set of vertices that are adjacent to v in G, including the vertex v. The closed neighbhourhood is denoted by $N[v]$. A set S of vertices in G is called a neighbhourhood set of G, if $G = \bigcup_{v \in S} < N[v] >$, where *<Nv>* is the subgraph induced by *N[v]*. A neighbhourhood set with minimum cardinality is called minimum paired neighbhourhood set. A neighbhourhood set S is called a paired neighbhourhood set, if $< S >$ has a perfect matching.

A graph $G(V, E)$ is called a circular-arc graph, if there is a *one-to-one* correspondence between $V(G)$ and a family $\mathcal{F}$ of arcs on a circle such that two vertices in G are adjacent *if and only if* the corresponding arcs in $\mathcal{F}$ intersect. Thus in $G(V, E)$, every vertex represents an arc and every edge represents pair of vertices whose corresponding arcs intersect. An arc is said to be *maximal*, if it is not contained in any other arc of $\mathcal{F}$.

Circular-arc graphs are useful in modeling resource allocation problem in operations research, traffic control problems, genetics and many more area as mentioned in [1]. Several algorithms have been developed for circular-arc graphs as mentioned in [2]. Most of the algorithms focused on finding the paired dominating set [3] – [5] . Algorithms on paired neighbhourhood set on interval graphs are given in [6] – [8]. The minimum paired neighbhourhood set problem on circular-arc graphs have not been explored yet by many researchers. In this paper, we give the algorithm to find paired neighbhourhood set on circular-arc graphs.

The following is the plan of the paper. In next section, we present the basic definitions and notations. Section 3 contains the algorithm to find minimum paired neighbhourhood set of a circular-arc graph and the following section gives the proof of correctness and time complexity analysis of the algorithm.

II. PRELIMINARIES

The following are the notations and definitions needed for the algorithm. Let $A = \{1,2, \dots, n\}$ be the family $\mathcal{F}$ of arcs on a circle. Let $\{ 1, 2, 3, \dots , n \}$ be the corresponding vertices in circular-arc graph G. We have chosen the same label for both arc and vertex to reduce the ambiguity in algorithm and subsequent proofs. For each arc i, the arc starts from the end point $h(i)$ and along the clock-wise directions ends at the end point $t(i)$. $h(i)$ is called the head point of arc i and $t(i)$ is called the tail point of arc i. Since $h(i)$and $t(i)$ are endpoints on a circle, it is assumed that all $h(i)$ and $t(i)$ are distinct and no single arc in $\mathcal{F}$ covers the whole circle. We assume that, for any arbitrarily chosen arc k from A, the ordering is done starting from the head point of arc k, in the increasing order of head points in clockwise direction. If $h(i)$ is encountered before $h(j)$ in clockwise direction then $i < j$. Fig.1 gives ordering of arcs and corresponding circular arc graph.

For each $i \epsilon A$, the head partner of i is denoted by $P_h(i)$, be the arc that intersects i and contains $h(i)$, and whose head point is last encountered in counter-clockwise direction. Similarly, the tail partner of i is denoted by $P_t(i)$, be the arc that intersects i and contains $t(i)$, and whose tail point is last encountered in clockwise direction. From Fig.1, $P_t(3) = 4, P_h(3) = 2$ and $P_t(5) = 6$.

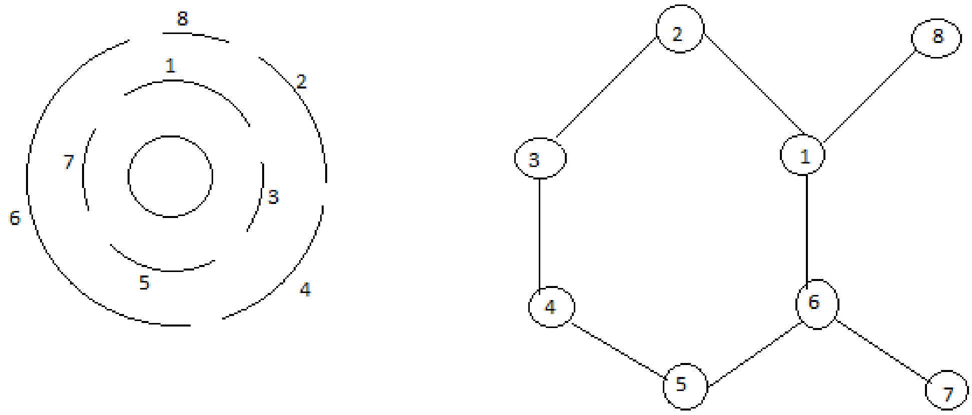

Fig.1: Circular Arcs and corresponding graphs

For each $i \in A$, we define $Pred_h(i)$ as the arc that intersects i and whose head point is between $h(i)$ and $t(i)$, and its head point is first encountered from $t(i)$ in counter clockwise direction. Similarly, we define $Pred_t(i)$ as the arc that intersects i and whose tail point is between $h(i)$ and $t(i)$, and its tail point is first encountered from $h(i)$ in clockwise direction. From Fig.1, $Pred_h(4) = 5$ and $Pred_t(4) = 3$ If there is no such arc, then $Pred_h(i) = i$ or $Pred_t(i) = i$.

$FNA_h(i)$ is defined as the first non-intersecting arc of arc i, whose tail is first encountered from $h(i)$ in counter clockwise direction. $FNA_t(i)$ is defined as the first non-intersecting arc of arc i, whose head is first encountered from $t(i)$ in clockwise direction. From Fig.1, $FNA_h(2) = 8$ and $FNA_t(2) = 4$.

An arc is member of $\mathcal{F}$, and a *segment* is a part of circle between two end points. A segment $\big(h(i), t(j)\big)$ is part of circle from $h(i)$ to $t(j)$ for any i, j, $1 \le i, j \le n$ and $i < j$. From Fig.1, the segment $\big(h(1), t(4)\big)$ contains the arcs 1,2,3, and 4.

If S is set of arcs, $FNA_h(S)$ is defined as the first non intersecting arc of the last head point in S, taken in counter clockwise direction. $FNA_t(S)$ is defined as the first non intersecting arc of the last tail point in S, taken in clockwise direction. From Fig. 1, $FNA_h(\{1,2,3,6\}) = 4$ and $FNA_t(\{1,2,3,4\}) = 7$.

Using the above mentioned notations we present the minimum paired neighbhourhood set on circular arc graph algorithm (MPNC) in the following section.

III. ALGORITHM MPNC

Input: A circular arc graph G with sorted endpoints.

1. Find $N[j]$, where j is an arbitrarily chosen arc in A

2. Let $W \leftarrow \{\, i \in N[j], i \;\; is$ maximal $\}$

3. Let $m \leftarrow |W|$

4. **For** each $i \in W$ do

5. $S_i^t \leftarrow \{i, P_t(i)\}$

6. **Repeat**

7. $v \leftarrow FNA_t(S_i^t)\,, k \leftarrow Pred_t(v)$

8. **If** $P_t\big(P_t(k)\big) \in S_i^t \; then \; S_i^t \;\leftarrow S_i^t \cup \{k, P_t(k)\}$

9. **Otherwise** $S_i^t \;\leftarrow S_i^t \cup \{\, P_t(k), P_t\big(P_t(k)\big)\}$

10. **Until** $every\ vertex\ not\ in\ S_i^t$ is adjacent to a vertex in S_i^t

11. **For** each $i \in W$ do

12. $S_i^h \leftarrow \{i, P_h(i)\}$

13. **Repeat**

14. $v \leftarrow FNA_h\big(S_i^h\big)\,, k \leftarrow Pred_h(v)$

15. **If** $P_h\big(P_h(k)\big) \in S_i^h \; then \; S_i^h \;\leftarrow S_i^h \cup \{k, P_h(k)\}$

16. **Otherwise** $S_i^h \;\leftarrow S_i^h \cup \{\, P_h(k), P_h\big(P_h(k)\big)\}$

17. **Until** $every\ vertex\ not\ in\ S_i^h$ is adjacent to a vertex in S_i^h

18. **For** $\; i = 1\ to\ m$

19. **If** $|S_i^t| \le \big|S_i^h\big| \; then \; S_i \leftarrow S_i^t$

20. **Otherwise** $S_i \leftarrow S_i^h$

21. Let $S = S_1$

22. **For** $\; i = 2\ to\ m$

23. **If** $|S_i| \le |S| \;\; then \; S \leftarrow S_i$

24. **Return** S

Output: A minimum paired neighbhourhood set S of G.

IV. PROOF OF CORRECTNESS

In this section we shall give lemmas and theorems to show that S produced by the algorithm $MPNC$ is a minimum paired neighbourhood set of G.

Lemma 1: If $W = \{i \in N[j], i \text{ is maximal}\}$, for any j, then there exist a minimum paired neighbourhood set S of G such that $S \cap W \neq \emptyset$.

Proof: For every $i \in W$, there exist a minimum paired neighbourhood set S such that $i \in S$ or $i \in N[u] \text{ where } u \in S$.

Case (i): If $i \in S$, then $S \cap W = i, \therefore S \cap W \neq \emptyset$.

Case (ii): If $i \in N[u] \text{ where } u \in S$, then since $i \in N[j]$, we have $i \in N[u] \cap N[j]$.

This implies that $u \in N[i]$ and $j \in N[i]$. Then the set $S' = \{S \setminus u\} \cup i$ is also a minimum paired neighbourhood set of G. $\therefore S \cap W \neq \emptyset$ ■

Lemma 2: Suppose i is a maximal arc and S_i is the minimum paired neighbourhood set of G among all neighbourhoods that contain i, then
$P_t(i) \in S_i \text{ or } P_h(i) \in S_i$.

Proof: This lemma can be proved by using the arguments similar to the arguments in lemma 2 from [2]. ■

Lemma 3: For any arc $u \in A$,
$\text{if } h(u) \text{ or } t(u) \text{ or both segment } \left(h(i), t(P_t(k))\right)$
for $1 \leq i \leq n$, with k as defined:

$$u \in N[i, P_t(i)] \text{ or } N[k, P_t(k)]$$

Proof: In this lemma we show that, for any arc u, if its head point or tail point or the entire arc lies in the $segment\ \left(h(i), t(P_t(k))\right)$, then the arc u is covered by $N[i, P_t(i)] \text{ or } N[k, P_t(k)]$. Three cases arise in this situation.

Case (i): $h(u) \notin \left(h(i), t(P_t(k))\right)$ but

$$t(u) \in \left(h(i), t(P_t(k))\right)$$

In this situation, $t(u)$ is encountered after $h(i)$, in clockwise direction, hence $u \in N[i]$

$$\therefore u \in N[i, P_t(i)] \text{ or } N[k, P_t(k)]$$

Case (ii): $h(u) \in \left(h(i), t(P_t(k))\right)$ but

$$t(u) \notin \left(h(i), t(P_t(k))\right)$$

In this situation, $h(u)$ is encountered before $t(P_t(k))$, hence $u \in N[P_t(k)]$

$$\therefore u \in N[i, P_t(i)] \text{ or } N[k, P_t(k)]$$

Case (iii): $h(u) \in \left(h(i), t(P_t(k))\right)$ but

$$t(u) \in \left(h(i), t(P_t(k))\right)$$

In this situation, suppose $u \notin N[i, P_t(i)]$, then by definition $FNA_t\ (u) = FNA_t(P_t(i))$. By the definition of k , $t(u) \in segment\ \left(h(k), t(P_t(k))\right)$, this implies $u \in N[k, P_t(k)]$

$$\therefore u \in N[i, P_t(i)] \text{ or } N[k, P_t(k)]$$

Suppose $u \notin N[k, P_t(k)]$, then by definition $FNA_h\ (u) = FNA_h(P_h(i))$. By the definition of k, $h(u) \in segment\ \left(h(i), t(P_t(i))\right)$, this implies $u \in N[i, P_t(i)]$

$$\therefore u \in N[i, P_t(i)] \text{ or } N[k, P_t(k)]$$

Lemma 4: Let $e = (u, v)$ be an edge in a circular arc graph G, where the arcs u and v are such that $t(u)$ and $t(v)$ are contained in the $segment$ $\left(h(i), t\left(P_t(P_t(k))\right)\right)$, for any $i, 1 \leq i \leq n$ and k as defined, then the edge
$(u, v) \in \langle N[i, P_t(i)]\rangle \cup \langle N[P_t(k), P_t(P_t(k))]\rangle$.

Proof: We give proof for this lemma by considering all the possible cases which can arise in the position of intersecting arcs u and v in the given segment.

Case (i): $arc\ u \in N[i]\ and\ v \in N[i]$
In this case the edge $(u, v) \in \langle N[i]\rangle$, which implies $(u, v) \in \langle N[i, P_t(i)]\rangle$
Case (ii): $arc\ u \in N[i]\ and\ v \notin N[i]$ but $v \in N[u]$
In this case, the arc v intersects u but does not intersect i, which implies $FNA_t(i) = v, i.e., v$ must be the first non intersecting arc from $t(i)$. This is a contradiction to the definition of k and the assumption about the arc v.
Case (iii): $arc\ u \in N\left[P_t(P_t(k))\right]\ and$

$$v \in N\left[P_t(P_t(k))\right]$$

In this case the edge $(u, v) \in \langle N\left[P_t(P_t(k))\right]\rangle$, which implies $(u, v) \in \langle N[P_t(k), P_t(P_t(k))]\rangle$
Case (iv): $arc\ u \in N\left[P_t(P_t(k))\right]\ and$

$$v \notin N\left[P_t(P_t(k))\right]$$

In this case, we have two different scenarios,
(a) $h(u)$ is encountered after $h(P_t(P_t(k))$, since u and v intersect, v must intersect $P_t(P_t(k))$, which is a contradiction to $v \notin N\left[P_t(P_t(k))\right]$.
(b) $h(u)$ is encountered before $h(P_t(P_t(k))$, since u and v intersect, v must intersect $P_t(k)$,
$\therefore v \in N[P_t(k)]$,
Hence $v \in N\left[P_t(k), P_t(P_t(k))\right]$
$\therefore (u, v) \in \langle N[i, P_t(i)]\rangle \cup \langle N[P_t(k), P_t(P_t(k))]\rangle$.
Case (v): $arc\ u \in N[i]\ and$

$$v \in N\left[P_t(P_t(k))\right]\ but\ v \notin N[i]$$

In this case, the arc v intersects $P_t(P_t(k))$ but it does not intersect i, then $FNA_t(i) = v$ as in $case\ (ii)$.
From all cases, we conclude that
$(u, v) \in \langle N[i, P_t(i)]\rangle \cup \langle N[P_t(k), P_t(P_t(k))]\rangle$.

Note: The above lemmas holds good for a segment taken in counter clockwise direction too.

Theorem 1: Given a circular arc graph G with end points sorted, the algorithm $MPNC$ produces a minimum paired neighbhourhood set of G.

Proof: For any $i \in A, 1 \leq i \leq n$, by lemma 1 and lemma 2, there exists a paired neighbhourhood set S of G, such that $P_t(i) \in S \text{ or } P_h(i) \in S$.
Let $S = \{i_1, P_t(i_1), \dots, i_k, P_t(i_k)\}$ be the set obtained from algorithm $MPNC$, where,

$$v_m = FNA_t\{i_1, P_t(i_1), \dots, i_m, P_t(i_m)\};$$
$$i_{m+1} = Pred(v_m)$$

Then by lemma 3 and lemma 4,

$$G = \langle N[i_1, P_t(i_1)]\rangle \cup \langle N[i_2, P_t(i_2)]\rangle \cup \dots \cup \langle N[i_k, P_t(i_k)]\rangle$$

Hence S is a paired neighbhourhood set of G and $|S| = 2k$.
Suppose S$'$ is a minimum paired neighbhourhood set of G, then it suffices to show that $|S| \leq |S'|$.
Let us consider any two consecutively generated pairs of vertices $\{i_m, P_t(i_m)\}$ and $\{i_{m+1}, P_t(i_{m+1})\}$. Since S' is minimum paired neighbhourhood set of G, for the segment $\left(h(i_m), t\left(P_t(i_{m+1})\right)\right)$ $in\ S$, there exist a corresponding segment $\left(h(p), t(l)\right)$ $in\ S'$, such that

$$\langle N[i_m, P_t(i_m), i_{m+1}, P_t(i_{m+1})]\rangle \subseteq \langle N[p,q]\rangle \cup \langle N[l,r]\rangle$$

$where\ q \in N[p] and\ r \in N[l]$. Thus for every consecutively generated pair in S, there exist a corresponding pair in S'. Since S has k such pairs, $|S'| \geq 2k, \therefore |S'| \geq |S|$.
As S' is minimum paired neighbhourhood set of $G, |S'| \leq |S|$.

$$\therefore |S'| = |S|,$$

hence S is a minimum paired neighbhourhood set of G ∎

V. TIME COMPLEXITY

Theorem 2: Give a circular arc graph G, with end points sorted, the algorithm $MPNC$ produces minimum paired neighbhourhood set of G in $O(n) - time$.

Proof: To find the values of $W, FNA(i)$ and k mentioned in the algorithm, it would take maximum of $O(n) - time$. By the analysis given in [2], the sets $S_1, S_2, \dots, S_m$ can be obtained in $O(n) - time$. Hence the algorithm produces minimum paired neighbhourhood set in $O(n) - time$.

VI. CONCLUDING REMARKS

We have developed an algorithm to solve the minimum paired neighbhourhood problem on circular arc graphs in $O(n) - time$. The study can be extended to find minimum paired neighbhourhood set on weighted interval graphs and weighted circular arc graphs.

REFERENCES

[1] Min Chih Lin, Jayme L. Szwarcfiter, "Characterizations and recognition of circular arc graphs and subclasses: A survey", *Discrete Mathematics*, vol. 309, pp. 5618 – 5635, (2009).

[2] Chig-Chi Lin, Hai-Lun Tu, "A linear-time algorithm for Paired –domination on Circular-arc graphs", *Theoretical Computer Science*, vol. 591, pp. 99 – 105, (2015).

[3] T. Haynas, P. Slater, "Paired-domination in graphs", *Networks*, vol. 32, pp.199 -206 , (1998).

[4] M.S. Chang, "Efficient algorithms for the domination problems on interval and circular-arc graphs", *SIAM Journal on Computing*, vol. 27, pp.1671-1694, (1998).

[5] T.C.E. Cheng, L.Y. Kang, C.T. Ng, "Paired domination on interval and circular-arc graphs", *Discrete Applied Mathematics*, vol. 155(16), pp. 2077-2086, (2007).

[6] E. Sampath Kumar, P.S. Neeralagi, "The line neighbhourhood number of a graph", *Indian journal of Pure and Applied Mathematics, vol.* 17, pp. 142-149, (1986).

[7] B. Maheshwari, Y. Lakshmi Naidu, L. Nagamuni Reddy, "A. Sudhakaraiah, A Polynomial Time algorithm for finding a minimum Independent neighbhourhood set of an interval graph", *Graph Theory Notes of New York XLVI, pp.*9-12, (2004).

[8] Y. Lakshmi Naidu, C.R. Saranya, "Paired Neighbhourhood set on Interval graphs", *International Journal of Pure and Engineering Mathematics, vol.* 4, pp. 21-26, (2016).