On Coloring of Arc graphs
Iskandar Karapetian

Aris Tentes
NTUA
Network Algorithms

The problem

- We have a circular graph and some paths (arcs) on it
- We want to color these arcs in such a way that no two overlapping arcs are colored with the same color
- We also want to use as few colors as possible

Notation

- L=max load
- ω=max clique of intersection graph
- p=point with load L
- χ=chromatic number
- $f=[f^-,f^+]$,the arc endpoints clockwise
- a≤b, if a in [p,b] (clockwise)
- a≥b, if a not in [p,b] (clockwise)

Alan Tucker

- It is easy to see, that $\chi \leq 2L-1$
- **Theorem**: If no three arcs cover the circle, then $\chi \leq 3/2\, L$

Steps of proof

- Induction. If L=1, trivial.
- Suppose it holds for L-1
- Take the arc containing p with the shortest length counter-clock size of p.
- Take successively the nearest, non intersecting arcs, starting with p, until we pass p three times.
- The inductive hypothesis holds for the rest of the arcs
- The selected arcs can be three colored.

This Paper

- **Theorem:** For all circular arc graphs $\chi \le 3/2\omega$.
- **Corollary:** Tucker's Theorem.

Sweep Subroutines

- **Clockwise-Sweep(F,f)**
  1. Start with f.
  2. Select from F:=F\{arcs not intersecting with f} the arc, g, with the nearest g^- to f^+.
  3. Repeat 2 with f:=g, until F is empty.
- **Counter-Clockwise-Sweep(F,f)**
  1. Start with f.
  2. Select from F:=F\{arcs not intersecting with f} the arc, g, with the nearest g^+ to f^-.
  3. Repeat 2 with f:=g, until F is empty.

- The set of arcs selected by a sweep routine contains pair wise independent arcs.
- Let F denote the set of all arcs.
- Let $\{f_1, \ldots, f_L\}$ be the set of all arcs containing p, s.t. $f_i^+ \leq f_{i+1}^+$

The Coloring Algorithm

1. G:=F\ $\{ f_1,\ldots, f_L \}$
2. For i:=1,…,L/2
   1. Select g in G with the nearest to p starting point
   2. The set A_i = Clockwise-Sweep(G,g) is colored with color L+i.
   3. G:=G\ A_i
3. G:=G∪$\{ f_1,\ldots, f_L \}$
4. For i:=L,…,1
   1. The set B_i =Counter-Clockwise-Sweep(G,f_i) is colored with color i.
   2. G:=G\ B_i

Proof of Coloring Algorithm

- Our objective is to show that the set G is empty after step 4.

- Then, we will obviously have a 3/2L coloring.

- We suppose that $L=\omega$ wlog, because if $L<\omega$ we can add arcs, s.t. $L=\omega$ without changing χ.

- We are going to use the following three lemata

Lemma 1

If after step 2, g in G, then for each i=1,…,L/2, there exists a_i in A_i, s.t. a_i contains $\bar{g}$

- Or else g, would be selected for some i.
- Observation: $\bar{g}$ has load at least L/2

Lemma 2

If after step 4, g in G, then there exists k, 2≤k≤L, s.t. $g^- \geq f_{k-1}^+$ and $g^- \leq f_k^+$

Moreover k-1>L/2.

- **Proof:**
  - If k=1, then g overlaps with $\{f_1, \ldots, f_L\}$,which implies a load of L+1
  - If k>L, then by a similar argument as before we are lead to a load of L+1
  - The lemma 1 holds for g. In addition the arcs in $\{g, f_k, \ldots, f_L\}$ contain g^-. By counting we have the above result.

Lemma 3

If after step 4, g in G, s.t. $g^- \geq f_{k-1}^+$ and $g^- \leq f_k^+$ then for each $h \in (\cup_{i=k}^{L} B_i) \setminus \{f_k, \ldots, f_L\}, h^- \leq g^+$.

Proof: Suppose there is an h, take the closest to g. We are going to find a point with load L+1

- For 1≤i≤k-1
  - An analogous to lemma 1 holds, thus define the sets

$U = \{b_i \in B_i : g^+ \in b_i \wedge h^- \in b_i, 1 \leq i \leq k-1\}$ and $V = \{b_i \in B_i : g^+ \in b_i \wedge h^- \notin b_i, 1 \leq i \leq k-1\} \cup \{g$

  - Observe that k-1+L/2-L<|U|≤L/2

…Proof

- For k≤i≤L
  - Define the sets

$I=\{i: \text{there is } b_i \in B_i \text{ s.t. } h^- \in b_i, k \le i \le L\}$ and $J=\{i: \text{there is no } b_i \in B_i \text{ s.t. } h^- \in b_i, k \le i \le L\}$

  - We can prove that |J|≥L-(k-1)-L/2+|U|
  - Let j in J be the smallest, s.t. none of the arcs in B_j contain g^+ (there is one..) and $J' = J|_{i<j}$
  - Observe that each element of (V∪{b_i:i in J'}) must intersect with f_j and hence with the rest through f_L.

…Proof

- Let q be the intersection beginning point of $(V \cup \{b_i : i \text{ in } J'\})$
- Lemma 1 holds for this point
- Therefore the load at q is

$$|(V \cup \{b_i : i \text{ in } J'\})| + |\{f_j, \ldots, f_L\}| + L/2 \geq |V| + |J| + L/2$$

$$\geq \ldots \geq L+1$$

…Contradiction…

…Proof of theorem

- Assume that the set G is not empty (there is a g in G) after step 4.
- We are going to construct a set of L+1, pair wise intersecting arcs, which is a contradiction.

- Let the sets $H_{k-1},\ldots,H_L$ be defined as follows:
  - $H_{k-1}=\{g\} \cup \{b_i \in B_i : g^+ \in b_i, 1 \le i \le k-1\}$
  - $H_{i+1}= \begin{cases} H_i \cup \{f_{i+1}\}, & \text{if } f_i \text{ intersects all arcs of } H_i \\ H_i \cup \{s_{i+1}\}, & \text{else} \end{cases}$

  where s_i the second arc added in Counter-Clockwise Sweep

...Proof of theorem

- Suppose that j is the smallest index, s.t. the arcs of H_j are not mutually intersecting.
- Clearly $H_j = H_{j-1} \cup \{s_j\}$ and there is an h in H_{j-1} which does not intersect with s_j.
- Either
  - $h \in (H_{j-1} \cap \{s_k, \ldots, s_{j-1}\}) \cup H_{k-1}$ or
  - $h \in H_{j-1} \cap \{f_k, \ldots, f_{j-1}\}$.
- However, the arcs in the first set are all overlapping and the second case leads us to a contradiction.

…first case

- All arcs in H_{k-1} contain g^+
- The same holds for $H = H_j \cap \{s_k, ..., s_j\} = \{s_i, ..., s_l\}$
- By induction. Suppose it holds for $\{s_i, \ldots, s_{m-1}\}$. We will see, that it holds for s_m also.
- Because s_m in H_j there is a u which does not intersect f_m but contains g^+ (induction hypoth.)
- Since u was not selected in B_m, $u^+ \leq s_m^+$
- Because o lemma 3, we have the inclusion of g^+

…second case

- Namely, $h \in H_{j-1} \cap \{f_k, \ldots, f_{j-1}\}$
- Since, $H_j = H_{j-1} U\{s_j\}$ there is an arc $v \in H_{j-1} \setminus \{f_k, \ldots, f_{j-1}\}$ which does not intersect with f_j.
- Since v was not selected in B_j, we have $v^+ \leq s_j^+$
- Observe that v does not intersect with h.
- Contradiction due to the definition of j.

- **G is empty. The proof is complete.**