Proper Interval Vertex Deletion

Yngve Villanger
University of Bergen

CIRM - Luminy (Marseille, France)

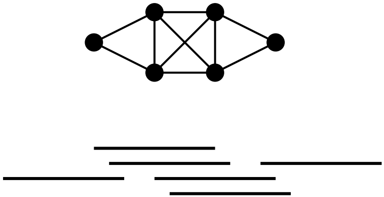
Overview

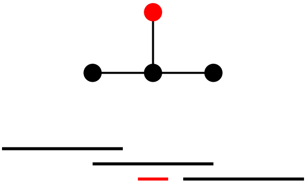
1. Characterization of proper interval graphs
2. The vertex deletion problem
3. Known algorithms
4. A small discussion about the proof
5. An approximation algorithm
6. Deleting edges to get a Proper Interval graph

Proper Interval, Unit Interval, or Indifference Graphs

Characterization

A graph is a proper interval graph if each vertex can be represented by an interval on the real line, such that two vertices are adjacent if and only if the intervals intersect, and no interval is a sub-interval of another interval.

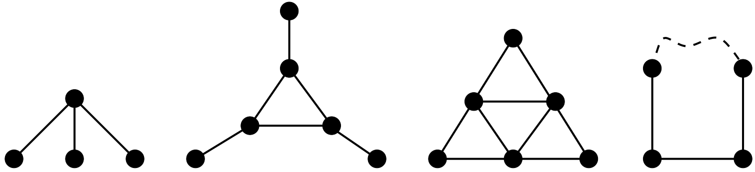




Proper Interval Graphs

Characterization [Wegner 67]

A graph is a proper interval graph if it does not contain claw, net, tent, C_i(Hole) for $4 \leq i$, as an induced sub-graph. (C_i is an induced cycle of length i.)



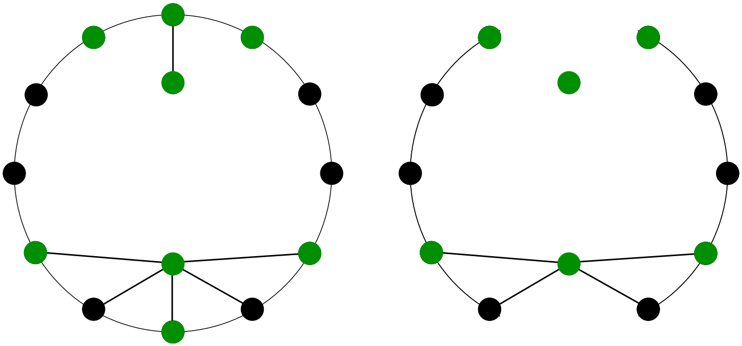
Claw Net Tent Hole

The problem

Problem: PROPER INTERVAL VERTEX DELETION

Input: A simple undirected graph G and an integer k.

Question: Is it possible to delete k vertices such that a proper interval graph remains?



Motivation and related problems

Related problems where k vertices are deleted:

Motivation and related problems

Related problems where k vertices are deleted:

- Maximum Induced Independent Set
  (**Vertex Cover**)

Motivation and related problems

Related problems where k vertices are deleted:

- Maximum Induced Independent Set
  (**Vertex Cover**)
- Maximum Induced Bipartite graph
  (**Odd Cycle Deletion**)

Motivation and related problems

Related problems where k vertices are deleted:

- Maximum Induced Independent Set
  (**Vertex Cover**)
- Maximum Induced Bipartite graph
  (**Odd Cycle Deletion**)
- Maximum Induced Sub-graph of Treewidth One
  (**Feedback Vertex Set**)

Motivation and related problems

Related problems where k vertices are deleted:

- Maximum Induced Independent Set
  (**Vertex Cover**)
- Maximum Induced Bipartite graph
  (**Odd Cycle Deletion**)
- Maximum Induced Sub-graph of Treewidth One
  (**Feedback Vertex Set**)
- Maximum Induced Chordal Sub-graph
  (**Chordal Vertex Deletion**)

Some history

Theorem [Lewis,Yannakakis 80]

It is NP-complete to decide if k vertices can be deleted, such that a proper interval graph remains.

Some history

Theorem [Lewis, Yannakakis 80]

It is *NP*-complete to decide if k vertices can be deleted, such that a proper interval graph remains.

Theorem [Cai 96]

Deleting k vertices such that any hereditary graph class remains is *FPT*, as long as the number of forbidden induced sub-graphs is bounded by some function of k.

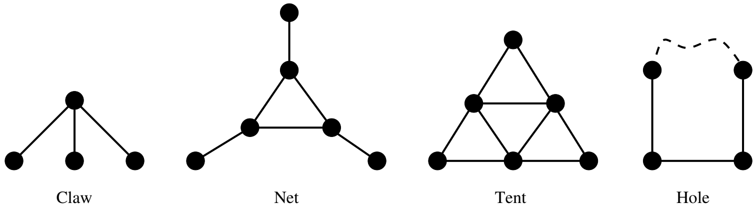
Some history

Theorem [Lewis, Yannakakis 80]

It is *NP*-complete to decide if k vertices can be deleted, such that a proper interval graph remains.

Theorem [Cai 96]

Deleting k vertices such that any hereditary graph class remains is *FPT*, as long as the number of forbidden induced sub-graphs is bounded by some function of k.



Algorithms for Proper Interval Vertex Deletion

Theorem [Marx 06]

Deleting k vertices to get a hole-free(Chordal) graph is FPT.

Algorithms for Proper Interval Vertex Deletion

Theorem [Marx 06]

Deleting k vertices to get a hole-free(Chordal) graph is FPT.

Alg 1

Branch on, claw, net, tent‘s
For each reduced instance use Marx’s algorithm
Running time : $O(f(k) * poly(n))$

Algorithms for Proper Interval Vertex Deletion

Theorem [Marx 06]

Deleting k vertices to get a hole-free(Chordal) graph is FPT.

Alg 1

Branch on, claw, net, tent's
For each reduced instance use Marx's algorithm
Running time : $O(f(k) * poly(n))$

Alg 2 [van Bevern et al. 10]

Branch on, claw, net, tent, C_4, C_5, C_6
Use iterative compression and structural arguments.
Running time : $O((14k+14)^{k+1} * kn^6)$

Algorithms for Proper Interval Vertex Deletion

Theorem [Marx 06]

Deleting k vertices to get a hole-free(Chordal) graph is FPT.

Alg 1

Branch on, claw, net, tent‘s
For each reduced instance use Marx's algorithm
Running time : $O(f(k) * poly(n))$

Alg 2 [van Bevern et al. 10]

Branch on, claw, net, tent, C_4, C_5, C_6
Use iterative compression and structural arguments.
Running time : $O((14k + 14)^{k+1} * kn^6)$

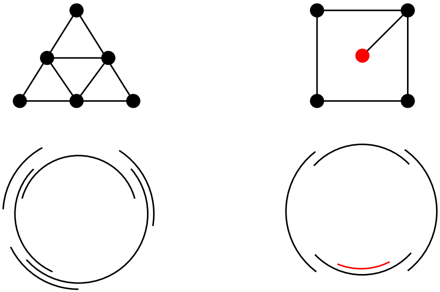
Alg 3 [this paper]

Branch on, claw, net, tent, C_4, C_5, C_6
Solve remaining problem in polynomial time.
Running time : $O(6^k * kn^6)$

Proper Circular Arc Graphs

Characterization

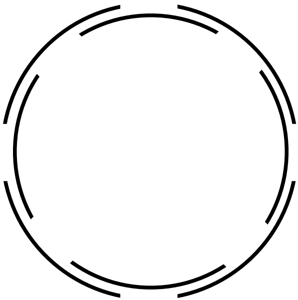
A graph is a proper circular arc graph if each vertex can be represented by an interval(arc) of a circle such that two vertices are adjacent if and only if the intervals intersect, and no interval is a sub-interval of another interval.



Main result

Main Theorem

A connected component of a *claw*, *net*, *tent*, C_4, C_5, C_6-free graph is a proper circular arc graph.



A sketch of the proof

A sketch of the proof

- Consider a connected *claw*, *net*, *tent*, C_4, C_5, C_6-free graph G which contains a hole C

A sketch of the proof

- Consider a connected *claw*, *net*, *tent*, C_4, C_5, C_6-free graph G which contains a hole C
- Let $w_1, w_2, \ldots, w_r$ for $7 \leq r$ be the vertices of induced cycle C

A sketch of the proof

- Consider a connected *claw*, *net*, *tent*, C_4, C_5, C_6-free graph G which contains a hole C
- Let $w_1, w_2, \ldots, w_r$ for $7 \leq r$ be the vertices of induced cycle C
- Let $v_1, v_2, \ldots, v_{n-r}$ be an ordering of the vertices of $V \setminus C$ such that $G_i = G[C \cup \{v_1, \cdots, v_i\}]$ is connected.

A sketch of the proof

- Consider a connected *claw*, *net*, *tent*, C_4, C_5, C_6-free graph G which contains a hole C
- Let $w_1, w_2, \ldots, w_r$ for $7 \leq r$ be the vertices of induced cycle C
- Let $v_1, v_2, \ldots, v_{n-r}$ be an ordering of the vertices of $V \setminus C$ such that $G_i = G[C \cup \{v_1, \cdots, v_i\}]$ is connected.
- The proof is by induction on i.

A sketch of the proof

- Consider a connected *claw*, *net*, *tent*, C_4, C_5, C_6-free graph G which contains a hole C
- Let $w_1, w_2, \ldots, w_r$ for $7 \leq r$ be the vertices of induced cycle C
- Let $v_1, v_2, \ldots, v_{n-r}$ be an ordering of the vertices of $V \setminus C$ such that $G_i = G[C \cup \{v_1, \cdots, v_i\}]$ is connected.
- The proof is by induction on i.
- base case: $i = 0$: $G[C]$ is clearly a proper circular arc graph

A sketch of the proof

- Consider a connected *claw*, *net*, *tent*, C_4, C_5, C_6-free graph G which contains a hole C
- Let $w_1, w_2, \ldots, w_r$ for $7 \leq r$ be the vertices of induced cycle C
- Let $v_1, v_2, \ldots, v_{n-r}$ be an ordering of the vertices of $V \setminus C$ such that $G_i = G[C \cup \{v_1, \cdots, v_i\}]$ is connected.

- The proof is by induction on i.
- base case: $i = 0$: $G[C]$ is clearly a proper circular arc graph
- Induction hypothesis: Let us assume that G_{i-1} is a proper circular arc graph

A sketch of the proof

- Consider a proper circular arc model $\mathcal{I}$ of G_{i-1}.

A sketch of the proof

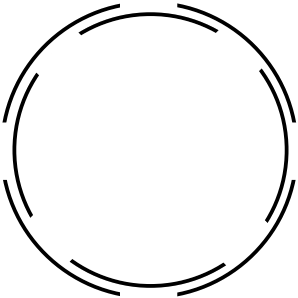
- Consider a proper circular arc model $\mathcal{I}$ of G_{i-1}.
- Cycle C can only be represented by completing the circle of the model $\mathcal{I}$.

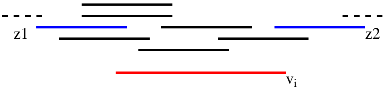
A sketch of the proof

- Consider a proper circular arc model $\mathcal{I}$ of G_{i-1}.
- Cycle C can only be represented by completing the circle of the model $\mathcal{I}$.
- Locally on the proper circular arc model can be represented as a proper interval model

A sketch of the proof

- Consider a proper circular arc model $\mathcal{I}$ of G_{i-1}.
- Cycle C can only be represented by completing the circle of the model $\mathcal{I}$.
- Locally on the proper circular arc model can be represented as a proper interval model
- Let z_1 and z_2 be the left most and right most neighbor of vertex v_i

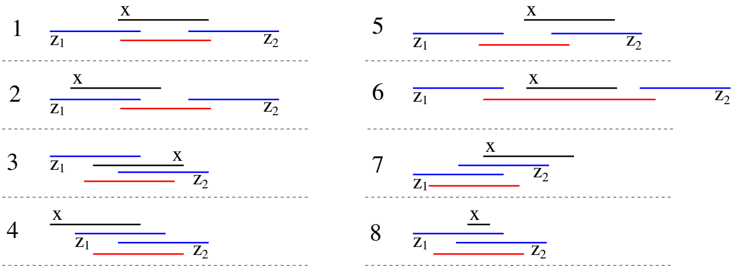




A sketch of the proof

Adding the new interval gives a proper circular arc model if

- No interval is a proper *subset* of the new interval
- The new interval is not a proper *subset* of an existing interval
- An existing interval intersects if and only if it represents a vertex in $N_{G_i}(v_i)$

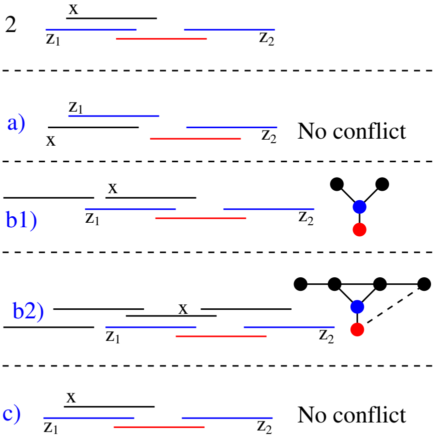


A sketch of the proof

Let us have a closer
look at the proof for Case 2.

- $x \notin N_{G_i}(v_i)$
  - a) $N_{G_{i-1}}[x] = N_{G_{i-1}}[z_1]$
  - b1) $y \in N_{G_{i-1}}[z_1] \setminus N_{G_{i-1}}[x]$
  - b2) $y \in N_{G_{i-1}}[x] \setminus N_{G_{i-1}}[z_1]$

c) $x \in N_{G_i}(v_i)$



Approximation algorithm

Theorem

The proper interval vertex deletion problem has a $6 \cdot OPT$ approximation algorithm.

Proof

While there exists $U \subseteq V$ such that
$G[U]$ is a claw,net,tent, C_4, C_5, or C_6, **then** remove U.
Solve the remaining instance in polynomial time.

Deleting k edges to get a proper interval graph

Theorem

There exists an $O(9^k \cdot poly(n))$ algorithm that decides if a proper interval graph can be obtained by deleting at most k edges.

Proof

- The forbidden induced subgraphs claw,net,tent,C_4, C_5, C_6 contain at most 9 edges. Branch on the 9 different ways of deleting an edge.
- If none of these subgraphs exist, use the model of the proper circular arc graph to find a minimum edge set to remove in poly time.

The end

Thank you for the attention.