Characterizing minimal interval completions: Towards better understanding of profile and pathwidth

Pinar Heggernes, Karol Suchan, Ioan Todinca, and
Yngve Villanger

STACS 2007, Aachen

Outline

1. Background
   - Motivation
   - Interval graphs
   - Minimum and minimal interval completion
   - Defining the Characterization
2. Folding
   - A single edge
   - Path-decomposition
   - Defining graphs
   - Defining foldings
   - Algorithm FillFolding
   - No single fill edge
   - Pivots
   - One or two unfolding
3. Unfolding
   - Recognizing a one unfolding
   - Recognizing a two unfolding
   - Extracting a minimal interval completion

Pinar Heggernes, Karol Suchan, Ioan Todinca, and Yngve Villanger

Motivation

Problems: minimum fill-in and tree-width

- Both problems are NP-hard
- The solution can be found among the Minimal triangulations
- Characterizations of minimal triangulations can be used to bound the search space.
- Several characterizations exist.

Problems: profile and path-width

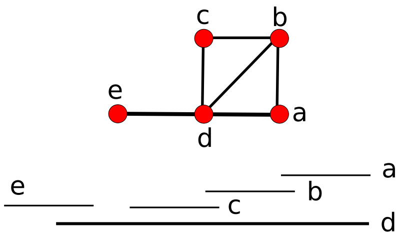
- Both problems are NP-hard
- The solution can be found among the Minimal *interval* completions
- We will now have a look at the first characterization of a minimal interval completion.

Pinar Heggernes, Karol Suchan, Ioan Todinca, and Yngve Villanger

Interval graphs

Definition

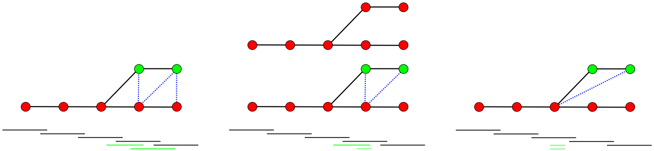
- A **graph** is an **interval graph** if every vertex can be assigned an interval on the real line, such that two lines only intersect if the corresponding vertices are adjacent.
- An interval graph $H = (V, E \cup F)$ where $E \cap F = \emptyset$ is called an interval **completion** of $G = (V, E)$ if $E \subseteq F$.
- The edge set F is called **fill-edges**.



Minimum and minimal interval completion

Definitions

- An interval graph $H = (V, E \cup F)$ is a **minimum** interval completion of $G = (V, E)$ if $E \cap F = \emptyset$ and $H' = (V, E \cup F')$ is not an interval graph for every edge set F' such that $|F'| < |F|$.
- An interval graph $H = (V, E \cup F)$ is a **minimal** interval completion of $G = (V, E)$ if $E \cap F = \emptyset$, and $H' = (V, E \cup F')$ is not an interval graph for every edge set F' such that $F' \subset F$.



Defining the Characterization

The questions

Given a graph $G = (V, E)$ and
an interval completion $H = (V, E \cup F)$ of G where $E \cap F = \emptyset$.

We answer the following question in polynomial time:

- Is H a minimal interval completion of G ?
- **Alternatively**
  Do there exist an edge set $F' \subset F$ such that $H' = (V, E \cup F')$ is an interval graph?
- **Finally**
  If H is not a minimal interval completion, can an edge set F', such that $F' \subset F$ and $H' = (V, E \cup F')$ is an interval graph be found?

Outline

1. Background
    - Motivation
    - Interval graphs
    - Minimum and minimal interval completion
    - Defining the Characterization
2. Folding
    - A single edge
    - Path-decomposition
    - Defining graphs
    - Defining foldings
    - Algorithm FillFolding
    - No single fill edge
    - Pivots
    - One or two unfolding
3. Unfolding
    - Recognizing a one unfolding
    - Recognizing a two unfolding
    - Extracting a minimal interval completion

Pinar Heggernes, Karol Suchan, Ioan Todinca, and Yngve Villanger

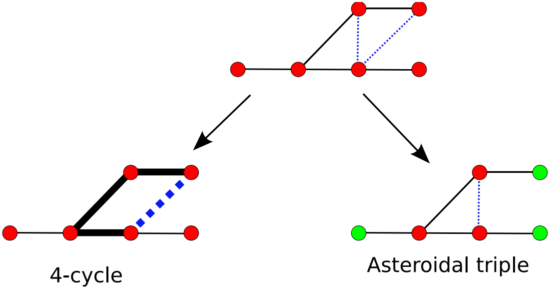
Removing a single edge

First step

If there exists a single edge $e \in F$ such that $H' = (V, E \cup F \setminus \{e\})$ is an interval graph, then we have the answer.

Observation

Removing a single edges will not always be sufficient.



Path-decomposition

Definition

A path-decomposition of a graph $G = (V, E)$ is a sequence $P = (X_1, X_2, ..., X_r)$ of subsets of V called bags, such that

- Each vertex $u \in V$ appears in some bag X_i,
- For every edge $xy \in E$ some bag X_i contains both x and y.
- The set of bags that contains a vertex x appears consecutively in P.

Definition

A path decomposition P is called a **clique path** of the given graph G if the vertices in every bag induces a **maximal** clique in G.
There exists a clique path P of a graph G if and only if G is an interval graph.

Defining some graphs

Graphs

We need some graphs to work width:

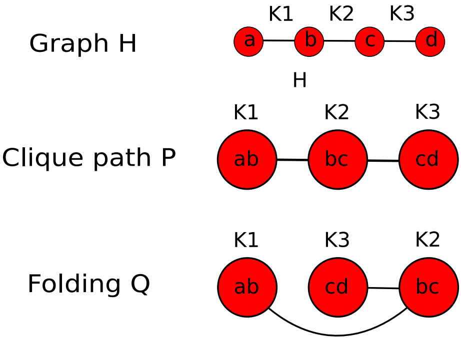
- Let $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ be an **arbitrary** graph.
- Let $\mathbf{H_2} = (\mathbf{V}, \mathbf{E} \cup \mathbf{F_2})$ be an **interval completion** of G $(E \cap F_2 = \emptyset)$.
- Let $\mathbf{H_0} = (\mathbf{V}, \mathbf{E} \cup \mathbf{F_0})$ be a **minimal** interval completion of G $(E \cap F_0 = \emptyset$ and $F_0 \subset F_2)$.
- Let $\mathbf{H_1} = (\mathbf{V}, \mathbf{E} \cup \mathbf{F_1})$ be an interval completion of G, where $F_0 \subset F_1 \subset F_2$.

By slightly abusing notation, we have:
$G \subset H_0 \subseteq H_1 \subset H_2$.

Defining foldings

Definition of folding

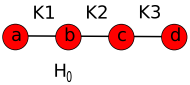
Let H be an interval graph, let Q be any permutation of the set of maximal cliques of H, and let P be a clique path of H. We say that (H, Q, P) is a **folding** of H.

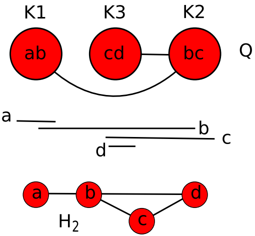


Algorithm FillFolding

Algorithm FillFolding

Input: A folding (H_0, Q, P) of an interval graph H_0
Input: An interval completion H_2 of H_0
(Basic idea) For each vertex x,
let Q_l and Q_r be the left most and right most maximal clique in Q that contains the vertex x. H_2 is obtained by adding x to every maximal clique between Q_l and Q_r.





No single fill edge

Theorem 1

Let H_2 be a an interval completion of an interval graph H_0, where the removal of any single fill edge results in a graph that is not an interval graph.

Then there exists a folding (H_0, Q, P), such that $H_2 = FillFolding(H_0, Q, P)$.

Pivots

Definition of pivot

Let (H_0, Q_0, P_0) be a folding. A clique $K \in Q$ is called a **pivot** in (H_0, Q_0, P_0) if both cliques just next to K (one to the left, the other to the right) in P_0 are on the same side of K in Q_0.

Theorem 2

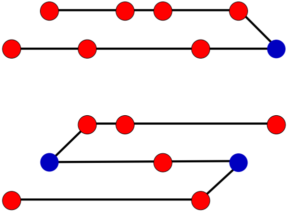
Let $H_2 = FillFolding(H_0, Q_0, P_0)$ be a an interval completion of H_0, where the removal of any single fill edge result in a non interval graph.
Then there exists an interval graph H_1 such that $E(H_0) \subseteq E(H_1) \subset E(H_2)$, and $H_2 = FillFolding(H_1, Q_1, P_1)$ where every pivot of the folding H_1, Q_1, P_1 contains a **simplicial** vertex of H_1.

One or two unfolding

Theorem 3

Let H_2 be a a non minimal interval completion of G, where the removal of any single fill edge result in a non interval graph.
Then there exits a folding (H_1, Q_1, P_1) where

- $E(G) \subset E(H_1) \subset E(H_2)$,
- $H_2 = FillFolding(H_1, Q_1, P_1)$,
- every pivot of (H_1, Q_1, P_1) contains a simplicial vertex in H_1,
- the folding (H_1, Q_1, P_1) contains one or two pivots.



Outline

1. Background
    - Motivation
    - Interval graphs
    - Minimum and minimal interval completion
    - Defining the Characterization
2. Folding
    - A single edge
    - Path-decomposition
    - Defining graphs
    - Defining foldings
    - Algorithm FillFolding
    - No single fill edge
    - Pivots
    - One or two unfolding
3. Unfolding
    - Recognizing a one unfolding
    - Recognizing a two unfolding
    - Extracting a minimal interval completion

Pinar Heggernes, Karol Suchan, Ioan Todinca, and Yngve Villanger

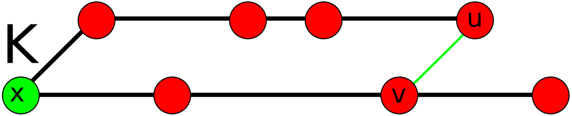
Recognizing a one unfolding

End pivot

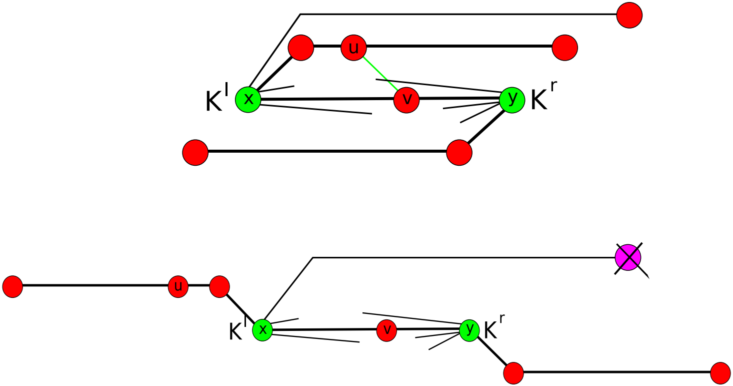
Let (H_0, Q_0, P_0) be a 1-folding and let $H_2 = FillFolding(H_0, Q_0, P_0)$. Then its pivot is a maximal clique in H_2. Moreover, there is a clique path of H_2 such that this pivot corresponds to a leaf.

One unfolding

Guess the right pair (K, u), where K is a maximal clique in H_2 and u is a vertex.



Trouble with the two unfolding

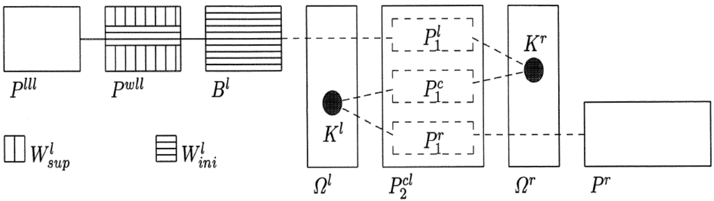


Pinar Heggernes, Karol Suchan, Ioan Todinca, and Yngve Villanger

Solving the two unfolding

Two unfolding

Guess the right tuple $(\Omega^l, \Omega^r, S^l, S^r, C^l, C^r, u, v)$, in H_2.



Extracting a minimal interval completion

Algorithm `ExtractMinimalIntervalCompletion`
Input: A graph $G = (V, E)$, and an int. comp. $H_2 = (V, E \cup F)$.
Output: A minimal interval completion H_1 of G. $[E(H_1) \subseteq E(H_2)]$.

```
$H_1 = H_2$
$H = G$
while $(H \neq H_1)$
    $H = H_1$
    if $H_1 - e$ is an int. graph for an edge $e \in E(H_1) \setminus E(G)$ then
        $H_1 = H_1 - e$
    else if $H_1 \neq \texttt{OneUnfolding}(G, H_1)$ then
        $H_1 = \texttt{OneUnfolding}(G, H_1)$
    else
        $H_1 = \texttt{TwoUnfolding}(G, H_1)$
return $H_1$
```

The end.