The k-in-a-path problem for claw-free graphs

Jiří Fiala*, Marcin Kamiński+, Bernard Lidický*,
Daniël Paulusma**

Charles University*
Université Libre de Bruxelles+
University of Durham**

22.1.2010 - CSASC 2010

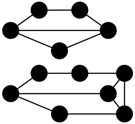
Motivation

3-IN-A-TREE: Find an induced tree containing given 3 vertices.

Theorem (Chudnovsky, Seymour, to appear)

The 3-IN-A-TREE *problem is solvable in polynomial time.*

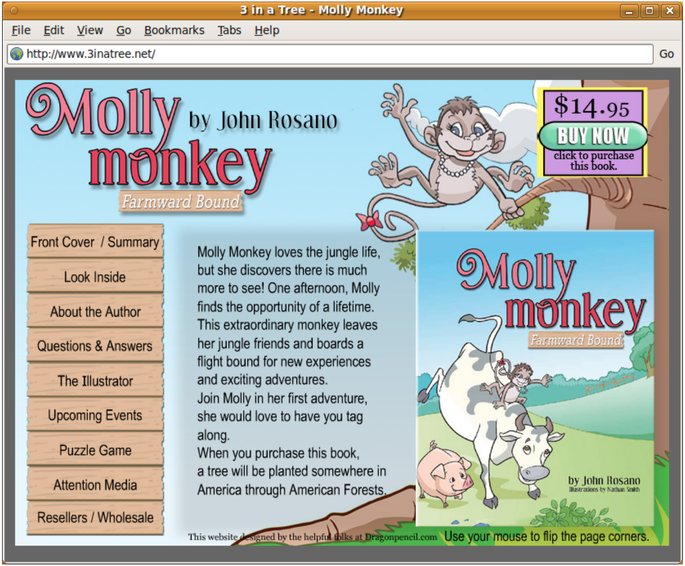
Algorithmic consequences and generalizations



- detecting thetas
- detecting pyramids
- 4-IN-A-TREE in triangle-free graphs [Derhy et. al. '09]
- *k*-IN-A-TREE in graphs of girth *k* [Trotignon and Wei '1X]

...

We: *k*-IN-A-TREE for claw-free graphs

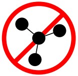


http://www.3inatree.net/

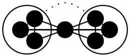
Definitions - quick reminder

A graph G is

- *claw-free* - no induced claw



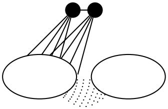
- *quasi-line* - $N(v)$ is union of two cliques



- a *line graph* - $G = L(H)$ for some H
- an *interval graph*

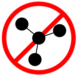
A graph G has

- a *k-hole* - induced k-cycle ($k \geq 4$)
- an *anti-hole* - induced complement of k-cycle
- a *homogeneous clique*



k-IN-A-PATH for claw-free graph

- *k*-IN-A-PATH: Find an induced path containing given *k* vertices (terminals).
- *k*-IN-A-TREE is *k*-IN-A-PATH for claw-free graphs



Theorem

k-IN-A-PATH *is solvable in polynomial time for claw-free graphs. (k constant)*

Theorem

k-IN-A-PATH *is NP-complete if k is part of the input for line graphs.*

Algorithm overview

Theorem

k-IN-A-PATH is solvable in polynomial time for claw-free graphs. (k constant)

claw-free G, terminals T, $|T| = k$, find induced P s.t. $T \subseteq P$

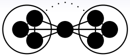
- fix the path a bit
- make G quasi-line
- make G quasi-line with no homogeneous clique
- make G circular interval or composition of interval graphs
- solve circular interval graph
- or solve intervals and k-DISJOINT-PATHS for a line graph

Algorithm overview - the path

claw-free G, terminals T, $|T| = k$, find induced P s.t. $T \subseteq P$

- fix the path a bit
  - $k \geq 3$
  - terminals are ordered $t_1, t_2, \ldots, t_k$
  - terminals and their neighbour are of degree ≤ 2
- make G quasi-line
- make G quasi-line with no homogeneous clique
- make G circular interval or composition of interval graphs
- solve circular interval graph
- or solve intervals and k-DISJOINT-PATHS for a line graph

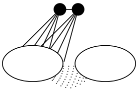
Algorithm overview - quasi-line



claw-free G, terminals T, $|T| = k$, find induced P s.t. $T \subseteq P$

- fix the path a bit
- make G quasi-line
  - clean G (no odd $\geq$ 7-anti-hole) [Hof, Kamiński, Paulusma '09]
  - remove vertices which have 5-anti-hole in neighbourhood
  - result is quasi-line as no odd anti-hole among neighbour
- make G quasi-line with no homogeneous clique
- make G circular interval or composition of interval graphs
- solve circular interval graph
- or solve intervals and k-DISJOINT-PATHS for a line graph

Algorithm overview - no homogeneous clique



claw-free G, terminals T, $|T| = k$, find induced P s.t. $T \subseteq P$

- fix the path a bit
- make G quasi-line
- make G quasi-line with no homogeneous clique
  - easy to check if edge is a homogeneous clique
  - contract homogeneous edge
- make G circular interval or composition of interval graphs
- solve circular interval graph
- or solve intervals and k-DISJOINT-PATHS for a line graph

Algorithm overview - intervals

claw-free G, terminals T, $|T| = k$, find induced P s.t. $T \subseteq P$

- fix the path a bit
- make G quasi-line
- make G quasi-line with no homogeneous clique
- make G circular interval or composition of interval graphs
  - find all homogeneous pairs of cliques [King, Reed '08] and contract them
  - result is circular interval or the composition [Chudnovsky, Seymour '05]
  - decide circular or composition [Deng, Hell, Huang '96]
- solve circular interval graph
- or solve intervals and k-DISJOINT-PATHS for a line graph

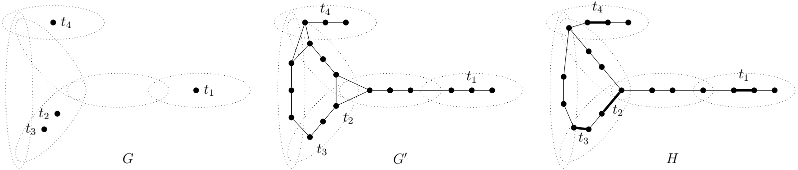
Algorithm overview - circular interval

claw-free G, terminals T, $|T| = k$, find induced P s.t. $T \subseteq P$

- fix the path a bit
- make G quasi-line
- make G quasi-line with no homogeneous clique
- make G circular interval or composition of interval graphs
- solve circular interval graph
    - get circular representation [Deng, Hell, Huang '96]
    - solve
- or solve intervals and k-DISJOINT-PATHS for a line graph

Algorithm overview - composition of interval graphs

- find the composition [King, Reed '08]
- solve each interval graph separately
- replace strips by short paths - G'



- get a graph H such that $G' = L(H)$
- get an instance of k-DISJOINT-PATHS on H
- solve k-DISJOINT-PATHS [Robertson, Seymour '95]

Corollaries

Theorem

The following problems are polynomial time solvable on claw-free graphs for a fixed k:

- k-INDUCED DISJOINT PATHS
- k-INDUCED CYCLE
- 2 MUTUALLY INDUCED HOLES

Open problems

Determine the computational complexity for

- ODD HOLE
- 2 MUTUALLY INDUCED HOLES

both are polynomial time solvable for claw free graphs
mutually induced odd holes is NP-complete