PACE challenge and potential maximal cliques

Tuukka Korhonen

November 28, 2017

PACE

- The Parameterized Algorithms and Computational Experiments Challenge
- Two tracks in 2017: treewidth and minimum fill-in
- Solutions evaluated with 100 or 200 test instances, 30 minutes timeout, use of parameterized algorithms encouraged

Minimum fill-in

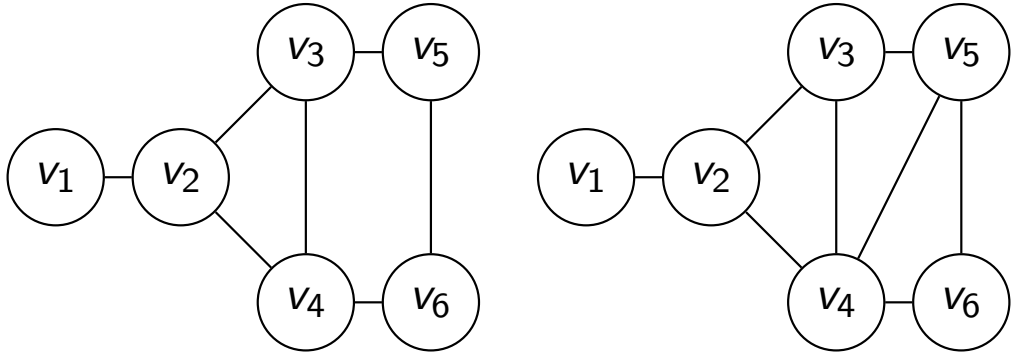
- Find a triangulation of a graph with the least number of edges
- Has exact algorithms with $O(1.8899^n)$ complexity and $2^{O(k^{0.5} \log k)} + O(k^2 mn)$ parameterized complexity
- Both of the algorithms are based on the algorithm by Bouchitte and Todinca (2001). They use potential maximal cliques

PACE 2017 minimum fill-in track

- **1st place:** Yasuaki Kobayashi (Kyoto University) and Hisao Tamaki (Meiji University). Solved 54 of 100 instances. Uses extension of BT-algorithm.
- **2nd place:** Jeremias Berg, Matti Järvisalo, and Tuukka Korhonen (University of Helsinki). Solved 45 of 100 instances. Uses BT-algorithm with kernelization.
- **3rd place:** Édouard Bonnet (University Paris-Dauphine), R.B. Sandeep (Hungarian Academy of Sciences), and Florian Sikora (University Paris-Dauphine). Solved 23 of 100 instances. Uses BT-algorithm.

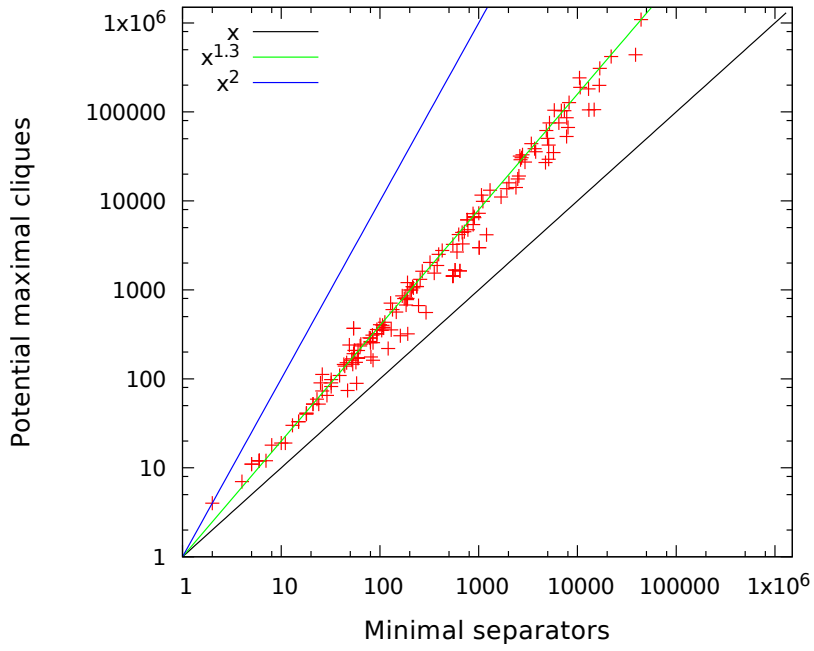
There seems to be no literature about implementation of BT-algorithm before this competition

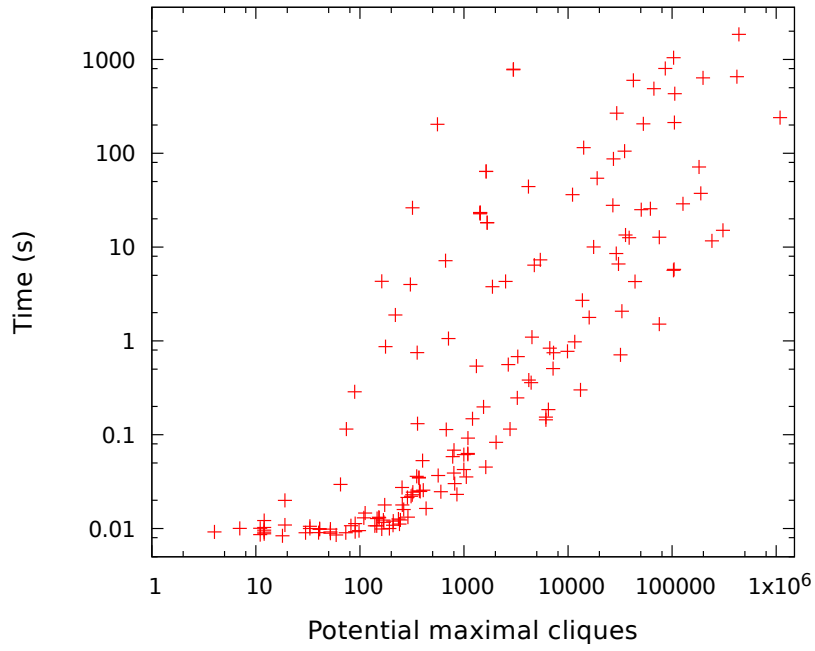
Potential maximal cliques



- A set K of vertices is a potential maximal clique if it is a maximal clique of some minimal triangulation
- Alternatively K is a potential maximal clique if $G - K$ has no full components associated to K and completing $N(G - K)$ makes K a clique.

BT-algorithm



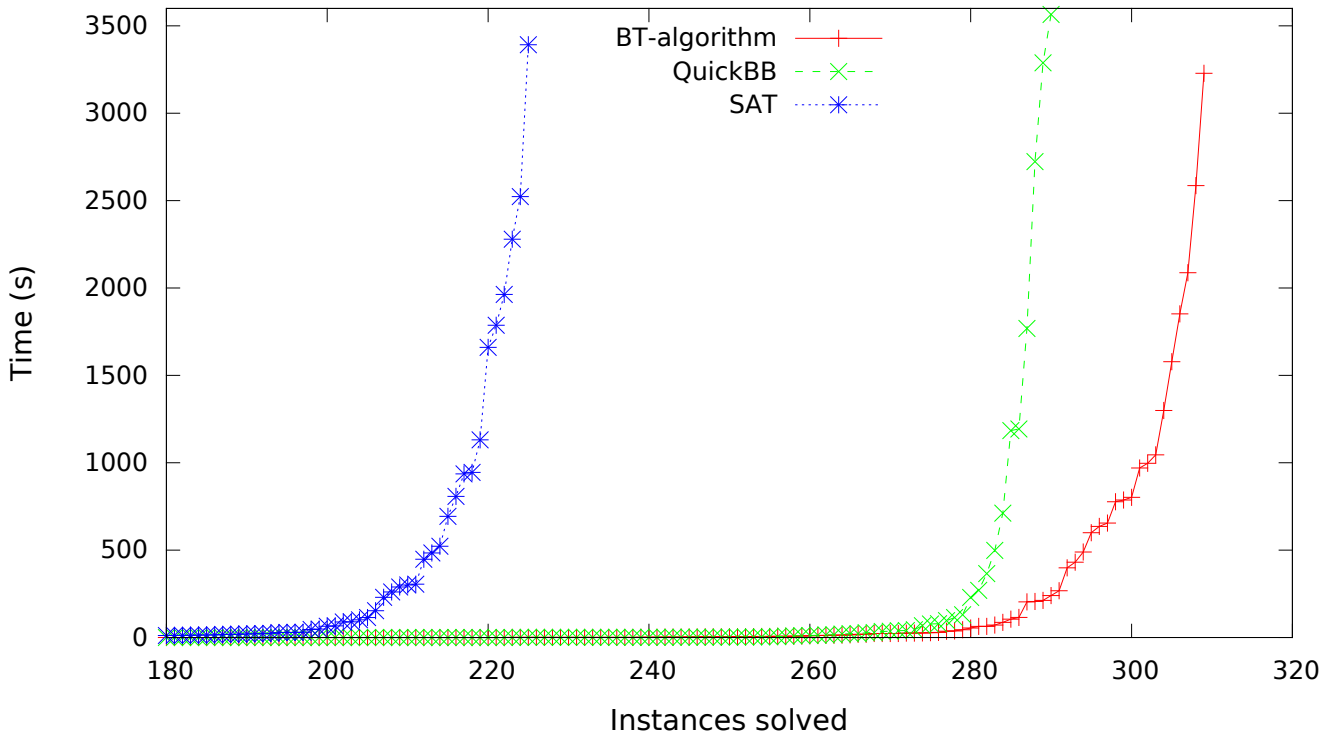


- Preprocess the graph
- Enumerate all potential maximal cliques
- Find the optimal triangulation via dynamic programming over potential maximal cliques
- Enumerating potential maximal cliques is $O(n^2 m|\Delta_G|^2)$, the dynamic programming is $O(|\Pi_G| poly(n))$

Generalization

- The BT-algorithm works for any triangulation cost function where the optimal triangulation is minimal and the costs of cliques can be considered independently of each other
- Examples:
  - Treewidth
  - Minimum fill-in
  - Triangulation of bayesian networks (TTS)
  - Generalized and fractional hypertreewidth
  - Perfect phylogeny

Our work



- All of the mentioned problems have already theory that formulates them in terms of potential maximal cliques
- But no experimental evaluation of BT-algorithm
- Our experiments show that BT-algorithm is competitive with other recent practical algorithms in those problems

Future?

- PACE 2018 is about the Steiner tree problem
- Enumeration of potential maximal cliques is the bottleneck of BT-algorithm. Improve that?