remote
9th LAWCG
and
MDA
November 25th, 2020

Efficient characterizations and algorithms of tree *t*-spanners

Fernanda Couto – UFRRJ – fernandavdc@ufrrj.br, Luís Felipe I. Cunha – UFF – lfignacio@ic.uff.br,
Diego Amaro Ferraz –UFRJ – ferrazda@cos.ufrj.br

Introduction

The t-admissibility problem has been widely studied specially because determining if a graph G is 3-admissible is still an open problem since it was proposed [2]. Although recognizing if a graph is 2-admissible is a polynomial time solvable problem, we realized that for some classes could be easier. Hence, in this work we present simple and efficient algorithms in order to characterize 2 and 3-admissible graphs for some graphs classes as cographs, split graphs, P_4-sparse and other superclasses.

Tree t-spanners

A tree *t*-spanner of a graph G is a spanning tree T of G in which the distance between adjacent vertices of G is at most *t* in T. In this case, we say that G is a t-admissible graph and the t-admissibility problem concerns in deciding if G is t-admissible. The minimum t for which G is t-admissible is the stretch index of G.

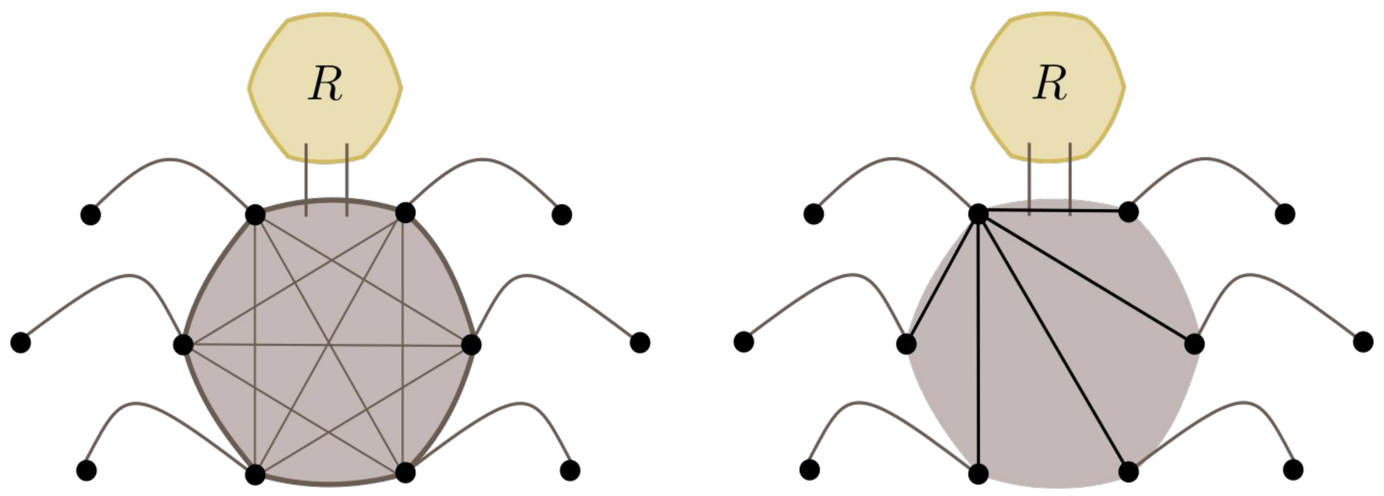

Figure 1: Thin spider graph and its tree 2-spanner T. Two parallel lines represent a join operation between the touched parts. Each vertex in the spider's body is connected to all other vertices in the spider's body and the vertices on the spider's head R. Thus there is a spanning star with respect to the body and R. Since spider's paws have degree one, we make them pendant in T, and then, the stretch index is equal 2.

Deciding whether G is 2-admissible can be solved in *O(n+m)* time, where *n* and *m* are the number of vertices and edges of G, respectively. t-admissibility is NP-complete for $t \geq 4$, and 3-admissibility remains an open problem.

Our goal is to provide simple and fast characterizations of tree *t*-spanners for graph classes in order to check 2- or 3-admissibility for them.

3-admissibility has been already efficiently solved for some graph classes, such as cographs, split graphs, cycle-power graphs and (2,1)-chordal graphs [1,3].

2-admissible P4-sparse graphs and (0,2)-graphs

For P_4-sparse graphs (graphs obtained from trivial graphs, by applying in any order union, join and spider operations), we have that, if G is not a thin spider (Figure 1) and has not a universal vertex, its stretch index is equal to 3.

Moreover, given a P_4-sparse graph G, G is 2-admissible if and only if either G has universal vertex; or G is a thin spider.

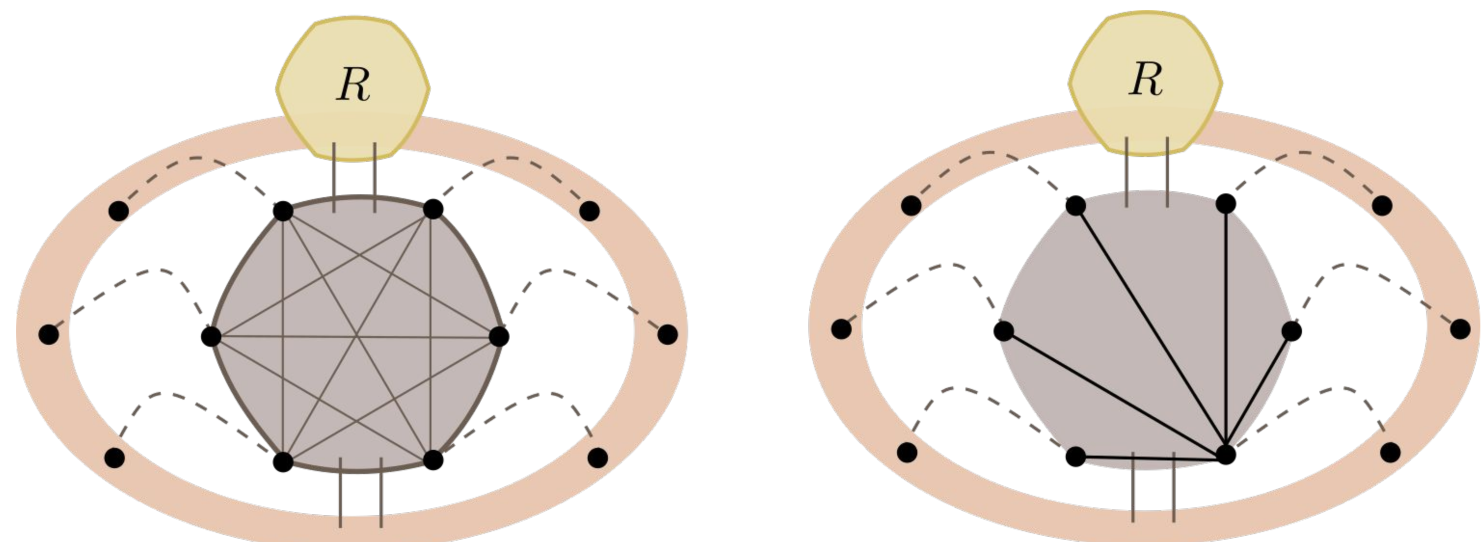

Figure 2: Thick spider graph and its tree 3-spanner. Two parallel lines represent a join operation between the touched parts. Dashed lines represent non-edges. Since each spider's paw is adjacent to all vertices of the body, except one, there is a spanning star with respect to the body and the head R with any vertex v of the spider's body as the center of the star. The paw that is not adjacent to v is placed in any of the leaves of the spider's body. And, thus, the stretch index of the graph is 3.

We present a linear time algorithm to decide 2-admissibility for P_4-sparse graphs. The algorithm consists in verifying the existence of a universal vertex and if the given graph is a thin spider. For this second part, we calculate its spider partition (S, K, R) and check the degrees of the vertices in order to differ the thin form the thick spider (Figure 2), which is not 2-admissible.

Considering (0,2)-graphs (graphs that can be partitioned into 0 independent set and 2 cliques) we also present a linear time algorithm to check the 2-admissibility.
Given a (0,2)-graph G, G is 2-admissible if and only if G has a universal vertex, a cut-vertex or between the parts of the (0,2)-partition is a strict 2-connected graph that has not an induced C_4.

Further work

In addition to the results presented above, we determined linear time algorithms to check 2-admissibility for P_4-tidy graphs, graphs that generalize P4-sparse graphs, as described above.

We also considered the *t*-admissibility problem for a superclass of (0,2)-graphs, the (k,l)-graphs. Specifically: split graphs (i.e. (1,1)-graphs) and (0,l)-graphs. We presented linear time algorithms to verify the existence of a tree 2-spanner.

As future work, we intend to extend this study to other graph classes and to deal with a recent study that is a variation of t-admissibility, called edge admissibility [4], concerning in obtaining a spanning tree of the line graph of G in which the distance between adjacent edges of G is at most *t*.

References

[1] F. COUTO; L. F. I. CUNHA; D. FERRAZ. Determining optimum tree t-spanners for split graphs and cographs. VIII Latin American Workshop on Cliques in Graphs, p.10, 2018.
[2] L. CAI ; D. G. CORNEIL. Tree spanners. SIAM J. Discrete Math., 8(3):359–387, 1995.
[3] F. COUTO; L. F. I. CUNHA. Hardness and efficiency on minimizing maximum distances for graphs with few P4's and (k,l)-graphs. Electronic Notes in Theoretical Computer Science, 346:355–367, 2019.
[4] F. COUTO; L. F. I. CUNHA.; D. POSNER. Edge tree spanner, CTW 2020. 1-12, 2020.