Edge tree spanners

Fernanda Couto – Universidade Federal Rural do Rio de Janeiro, Brasil
Luís Cunha – Universidade Federal Fluminense, Brasil
Daniel Posner – Universidade Federal Rural do Rio de Janeiro, Brasil

18th CTW, 2020

September, 14, 2020

A min-max problem

- Finding a spanning tree of a connected graph is an easy well-studied problem

A min-max problem

- Finding a spanning tree of a connected graph is an easy well-studied problem
- However, it can become a challenge if we add some constraints on it

A min-max problem

- Finding a spanning tree of a connected graph is an easy well-studied problem
- However, it can become a challenge if we add some constraints on it
- In this work, we deal with a problem that makes two opposite optimality analysis
  - Goal: determining trees that minimize the maximum distance between vertices (or edges) of connected graphs.

Tree-spanners

- A tree t-spanner of a graph G is a spanning tree T of G such that the distance in T between each pair of adjacent vertices of G is at most t.

Tree-spanners

- A tree t-spanner of a graph G is a spanning tree T of G such that the distance in T between each pair of adjacent vertices of G is at most t.
- G is called a t-admissible graph

Tree-spanners

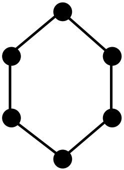
- A tree t-spanner of a graph G is a spanning tree T of G such that the distance in T between each pair of adjacent vertices of G is at most t.
- G is called a t-admissible graph
- The stretch index of G, denoted by $\sigma_T(G)$, is the smallest t for which a graph G is t-admissible

Tree-spanners

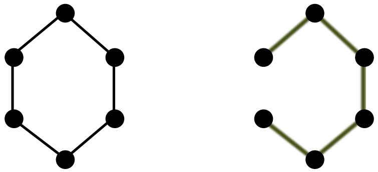
- A tree t-spanner of a graph G is a spanning tree T of G such that the distance in T between each pair of adjacent vertices of G is at most t.
- G is called a t-admissible graph
- The stretch index of G, denoted by $\sigma_T(G)$, is the smallest t for which a graph G is t-admissible

The MSST problem
Instance: A connected graph G and an integer t.
Question: Is G $\sigma_T(G)$-admissible graph, $\sigma_T(G) \leq t$?

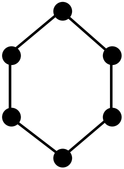
Examples

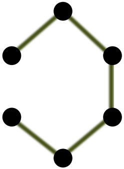


Examples



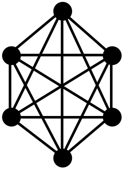
Examples



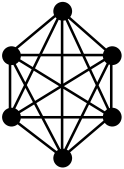


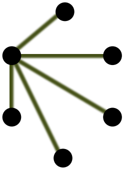
$$\sigma_T(C_6) = 5$$

Examples

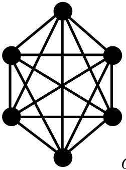


Examples

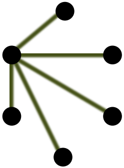




Examples



$\sigma_T(K_6) = 2$



Edge tree-spanners

- An edge tree t-spanner of a graph G is a spanning tree T of $L(G)$ such that the distance in T between each pair of adjacent edges of G is at most t.
  - $L(G)$ is the line graph of G, i.e. the intersection graph of the edges of G.

Edge tree-spanners

- An edge tree t-spanner of a graph G is a spanning tree T of $L(G)$ such that the distance in T between each pair of adjacent edges of G is at most t.
  - $L(G)$ is the line graph of G, i.e. the intersection graph of the edges of G.
- G is called an edge t-admissible graph

Edge tree-spanners

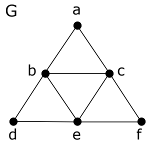
- An edge tree t-spanner of a graph G is a spanning tree T of $L(G)$ such that the distance in T between each pair of adjacent edges of G is at most t.
  - $L(G)$ is the line graph of G, i.e. the intersection graph of the edges of G.
- G is called an edge t-admissible graph
- The edge stretch index of G, denoted by $\sigma'_T(G)$, is the smallest t for which a graph G is edge t-admissible.

Edge tree-spanners

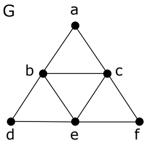
- An edge tree t-spanner of a graph G is a spanning tree T of $L(G)$ such that the distance in T between each pair of adjacent edges of G is at most t.
  - $L(G)$ is the line graph of G, i.e. the intersection graph of the edges of G.
- G is called an edge t-admissible graph
- The edge stretch index of G, denoted by $\sigma'_T(G)$, is the smallest t for which a graph G is edge t-admissible.

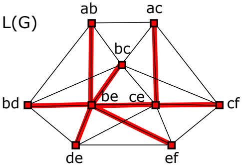
Equivalently:
Determining the edge stretch index of G is equivalent to determine the stretch index of $L(G)$.

Example

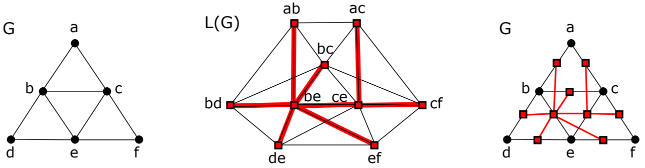


Example

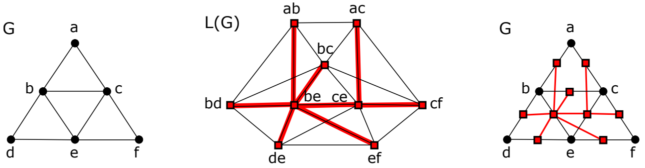




Example



Example



G is edge 3-admissible.

Computational complexity

- Tree-spanners:

  Determining whether $\sigma_T(G) \leq 2$ is polynomial-time solvable

Computational complexity

- Tree-spanners:

  Determining whether $\sigma_T(G) \leq 2$ is polynomial-time solvable

  Determining whether $\sigma_T(G) \leq t, t \geq 4$ is NP-complete

Computational complexity

- Tree-spanners:

  Determining whether $\sigma_T(G) \leq 2$ is polynomial-time solvable

  Determining whether $\sigma_T(G) \leq t, t \geq 4$ is NP-complete

  Determining whether $\sigma_T(G) = 3$ is open

Computational complexity

- Tree-spanners:

  Determining whether $\sigma_T(G) \leq 2$ is polynomial-time solvable

  Determining whether $\sigma_T(G) \leq t, t \geq 4$ is NP-complete

  Determining whether $\sigma_T(G) = 3$ is open

  - L. Cai, D. G. Corneil, Tree spanners, SIAM J. Discrete Math, 1995.

Computational complexity

- Tree-spanners:

  Determining whether $\sigma_T(G) \leq 2$ is polynomial-time solvable

  Determining whether $\sigma_T(G) \leq t, t \geq 4$ is NP-complete

  Determining whether $\sigma_T(G) = 3$ is open

  - L. Cai, D. G. Corneil, Tree spanners, SIAM J. Discrete Math, 1995.

- Edge tree-spanners:

  There is no study regarding the admissibility for line graphs.

  Hence, . . .

In this work, we ...

- prove that edge t-admissibility is a polynomial time solvable problem, for $t \leq 3$.

In this work, we ...

- prove that edge t-admissibility is a polynomial time solvable problem, for $t \leq 3$.
  - Hence, 3-admissibility is a polynomial time solvable problem for line graphs.

In this work, we ...

- prove that edge t-admissibility is a polynomial time solvable problem, for $t \leq 3$.
  - Hence, 3-admissibility is a polynomial time solvable problem for line graphs.
- settle sufficient conditions of graphs to be edge 4-admissible or edge 5-admissible.

In this work, we ...

- prove that edge t-admissibility is a polynomial time solvable problem, for $t \leq 3$.
  - Hence, 3-admissibility is a polynomial time solvable problem for line graphs.

- settle sufficient conditions of graphs to be edge 4-admissible or edge 5-admissible.
  - Hence, we determine graph classes which are edge 5-admissible, and determining their edge stretch indexes is a polynomial time solvable problem.

In this work, we ...

- prove that edge t-admissibility is a polynomial time solvable problem, for $t \leq 3$.
  - Hence, 3-admissibility is a polynomial time solvable problem for line graphs.
- settle sufficient conditions of graphs to be edge 4-admissible or edge 5-admissible.
  - Hence, we determine graph classes which are edge 5-admissible, and determining their edge stretch indexes is a polynomial time solvable problem.
- prove that Edge 8-admissibility problem is NP-complete, even for bipartite graphs.

In this work, we ...

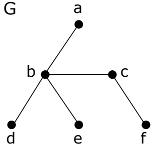
- prove that edge t-admissibility is a polynomial time solvable problem, for $t \leq 3$.
  - Hence, 3-admissibility is a polynomial time solvable problem for line graphs.
- settle sufficient conditions of graphs to be edge 4-admissible or edge 5-admissible.
  - Hence, we determine graph classes which are edge 5-admissible, and determining their edge stretch indexes is a polynomial time solvable problem.
- prove that Edge 8-admissibility problem is NP-complete, even for bipartite graphs.
  - Hence, we can adapt such a NP-completeness in order to prove that Edge $2k$-admissibility problem is NP-complete, for $k \geq 5$.

Edge k-admissibility, $k \in \{1, 2\}$

Trees are 1-admissible

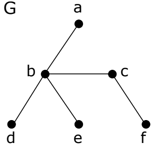
Edge k-admissibility, $k \in \{1, 2\}$

Trees are 1-admissible



Edge k-admissibility, $k \in \{1, 2\}$

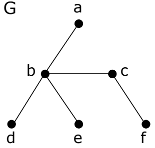
Trees are 1-admissible



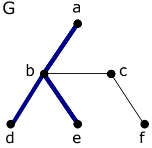
Line graphs are claw-free

Edge k-admissibility, $k \in \{1, 2\}$

Trees are 1-admissible

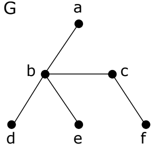


Line graphs are claw-free

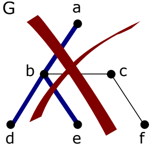


Edge k-admissibility, $k \in \{1, 2\}$

Trees are 1-admissible



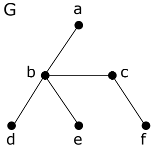
Line graphs are claw-free



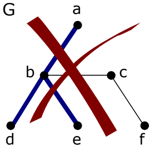
Since line graphs are claw-free,

Edge k-admissibility, $k \in \{1, 2\}$

Trees are 1-admissible



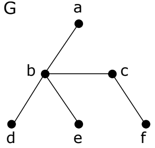
Line graphs are claw-free



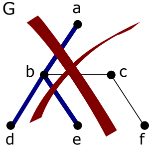
Since line graphs are claw-free, then path graphs are the unique edge 1-admissible graphs.

Edge k-admissibility, $k \in \{1, 2\}$

Trees are 1-admissible



Line graphs are claw-free

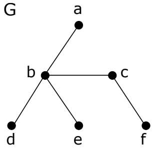


Since line graphs are claw-free, then path graphs are the unique edge 1-admissible graphs.

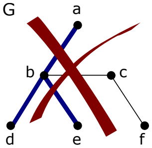
Proposition: Let G be a tree.

Edge k-admissibility, $k \in \{1, 2\}$

Trees are 1-admissible

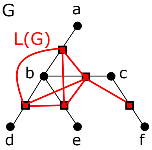


Line graphs are claw-free



Since line graphs are claw-free, then path graphs are the unique edge 1-admissible graphs.

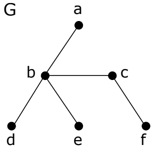
Proposition: Let G be a tree. If G is a path graph, then $\sigma'(G) = 1$, otherwise $\sigma'(G) = 2$.



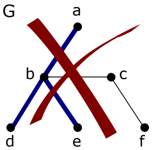
Each internal node of G corresponds to a maximal clique in $L(G)$

Edge k-admissibility, $k \in \{1, 2\}$

Trees are 1-admissible

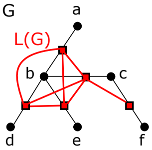


Line graphs are claw-free



Since line graphs are claw-free, then path graphs are the unique edge 1-admissible graphs.

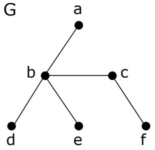
Proposition: Let G be a tree. If G is a path graph, then $\sigma'(G) = 1$, otherwise $\sigma'(G) = 2$.



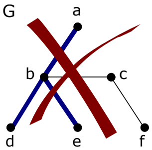
Each internal node of G corresponds to a maximal clique in $L(G)$ and two maximal cliques share at most a vertex in $L(G)$.

Edge k-admissibility, $k \in \{1, 2\}$

Trees are 1-admissible

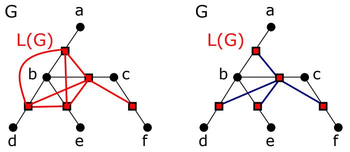


Line graphs are claw-free



Since line graphs are claw-free, then path graphs are the unique edge 1-admissible graphs.

Proposition: Let G be a tree. If G is a path graph, then $\sigma'(G) = 1$, otherwise $\sigma'(G) = 2$.

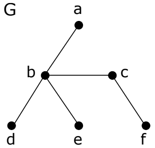


Each internal node of G corresponds to a maximal clique in $L(G)$ and two maximal cliques share at most a vertex in $L(G)$.

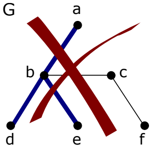
For each clique in $L(G)$, obtain a spanning star.

Edge k-admissibility, $k \in \{1, 2\}$

Trees are 1-admissible

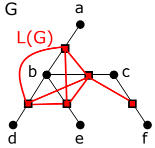


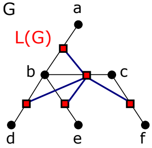
Line graphs are claw-free



Since line graphs are claw-free, then path graphs are the unique edge 1-admissible graphs.

Proposition: Let G be a tree. If G is a path graph, then $\sigma'(G) = 1$, otherwise $\sigma'(G) = 2$.





Each internal node of G corresponds to a maximal clique in $L(G)$ and two maximal cliques share at most a vertex in $L(G)$.

For each clique in $L(G)$, obtain a spanning star.

Corollary: Edge 2-admissibility is a polynomial time solvable problem.

Edge 3-admissibility

Lemma: Edge 3-admissibility is a hereditary property for induced subgraphs.

Edge 3-admissibility

Lemma: Edge 3-admissibility is a hereditary property for induced subgraphs.

Sketch of proof:

Edge 3-admissibility

Lemma: Edge 3-admissibility is a hereditary property for induced subgraphs.

Sketch of proof: Assume G is edge 3-admissible and it contains an induced subgraph H that is not edge 3-admissible, but H is edge k-admissible, for $k \geq 4$.

Edge 3-admissibility

Lemma: Edge 3-admissibility is a hereditary property for induced subgraphs.

Sketch of proof: Assume G is edge 3-admissible and it contains an induced subgraph H that is not edge 3-admissible, but H is edge k-admissible, for $k \geq 4$.

Suppose also that H is a subgraph of G', G' is a subgraph of G and G' is edge 3-admissible.

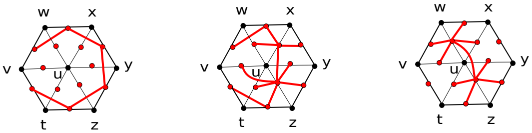
Edge 3-admissibility

Lemma: Edge 3-admissibility is a hereditary property for induced subgraphs.

Sketch of proof: Assume G is edge 3-admissible and it contains an induced subgraph H that is not edge 3-admissible, but H is edge k-admissible, for $k \geq 4$.

Suppose also that H is a subgraph of G', G' is a subgraph of G and G' is edge 3-admissible.

Assume $u \in (V(G') \setminus V(H))$.



A k-path in H must be decreased to a 3-path through edges incident to u.

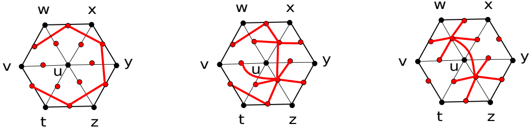
Edge 3-admissibility

Lemma: Edge 3-admissibility is a hereditary property for induced subgraphs.

Sketch of proof: Assume G is edge 3-admissible and it contains an induced subgraph H that is not edge 3-admissible, but H is edge k-admissible, for $k \geq 4$.

Suppose also that H is a subgraph of G', G' is a subgraph of G and G' is edge 3-admissible.

Assume $u \in (V(G') \setminus V(H))$.



A k-path in H must be decreased to a 3-path through edges incident to u.

The addition of u does not imply in a edge 3-admissible graph.

Graphs not edge 3-admissible

Corollary: Any complete graph K_n has $\sigma'(K_n) = 4$, for $n \geq 5$.

Graphs not edge 3-admissible

Corollary: Any complete graph K_n has $\sigma'(K_n) = 4$, for $n \geq 5$.

Sketch of proof:

Graphs not edge 3-admissible

Corollary: Any complete graph K_n has $\sigma'(K_n) = 4$, for $n \geq 5$.

Sketch of proof:

1. K_5 are edge 4-admissible,

Graphs not edge 3-admissible

Corollary: Any complete graph K_n has $\sigma'(K_n) = 4$, for $n \geq 5$.

Sketch of proof:

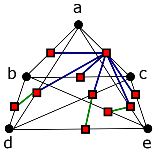
1. K_5 are edge 4-admissible, proved by a case analysis;

Graphs not edge 3-admissible

Corollary: Any complete graph K_n has $\sigma'(K_n) = 4$, for $n \geq 5$.

Sketch of proof:

1. K_5 are edge 4-admissible, proved by a case analysis;

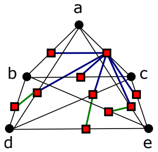


Graphs not edge 3-admissible

Corollary: Any complete graph K_n has $\sigma'(K_n) = 4$, for $n \geq 5$.

Sketch of proof:

1. K_5 are edge 4-admissible, proved by a case analysis;



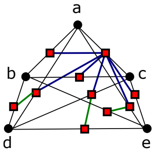
2. K_n are edge 4-admissible, for $n \geq 5$.

Graphs not edge 3-admissible

Corollary: Any complete graph K_n has $\sigma'(K_n) = 4$, for $n \geq 5$.

Sketch of proof:

1. K_5 are edge 4-admissible, proved by a case analysis;



2. K_n are edge 4-admissible, for $n \geq 5$.

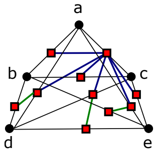
3. Edge 3-admissibility is hereditary and any K_n, for $n \geq 5$, contains a K_5 as induced subgraph, which is not edge 3-admissible, hence, K_n is not edge 3-admissible, as well.

Graphs not edge 3-admissible

Corollary: Any complete graph K_n has $\sigma'(K_n) = 4$, for $n \geq 5$.

Sketch of proof:

1. K_5 are edge 4-admissible, proved by a case analysis;



2. K_n are edge 4-admissible, for $n \geq 5$.

3. Edge 3-admissibility is hereditary and any K_n, for $n \geq 5$, contains a K_5 as induced subgraph, which is not edge 3-admissible, hence, K_n is not edge 3-admissible, as well.

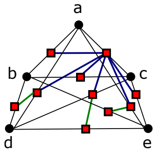
Proposition: C_n are not edge 3-admissible, for $n \geq 5$.

Graphs not edge 3-admissible

Corollary: Any complete graph K_n has $\sigma'(K_n) = 4$, for $n \geq 5$.

Sketch of proof:

1. K_5 are edge 4-admissible, proved by a case analysis;



2. K_n are edge 4-admissible, for $n \geq 5$.
3. Edge 3-admissibility is hereditary and any K_n, for $n \geq 5$, contains a K_5 as induced subgraph, which is not edge 3-admissible, hence, K_n is not edge 3-admissible, as well.

Proposition: C_n are not edge 3-admissible, for $n \geq 5$.

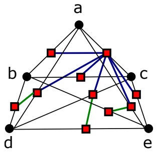
Sketch of proof:

Graphs not edge 3-admissible

Corollary: Any complete graph K_n has $\sigma'(K_n) = 4$, for $n \geq 5$.

Sketch of proof:

1. K_5 are edge 4-admissible, proved by a case analysis;



2. K_n are edge 4-admissible, for $n \geq 5$.
3. Edge 3-admissibility is hereditary and any K_n, for $n \geq 5$, contains a K_5 as induced subgraph, which is not edge 3-admissible, hence, K_n is not edge 3-admissible, as well.

Proposition: C_n are not edge 3-admissible, for $n \geq 5$.

Sketch of proof: $L(C_n) \approx C_n$ and any C_n is $(n-1)$-admissible.

How about K_4's and C_4's?

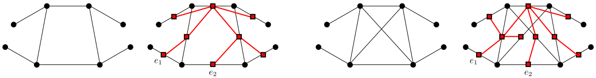
- Note that every edge 3-spanner graph is K_n-free and C_n-free, for $n \geq 5$.

How about K_4's and C_4's?

- Note that every edge 3-spanner graph is K_n-free and C_n-free, for $n \geq 5$.
- Moreover, induced C_4's of K_4's may not be edge 3-admissible.

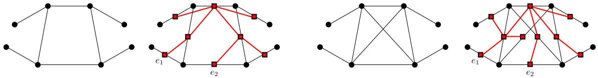
How about K_4's and C_4's?

- Note that every edge 3-spanner graph is K_n-free and C_n-free, for $n \geq 5$.
- Moreover, induced C_4's of K_4's may not be edge 3-admissible.



How about K_4's and C_4's?

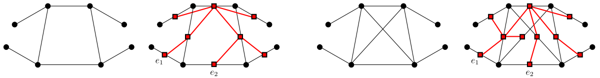
- Note that every edge 3-spanner graph is K_n-free and C_n-free, for $n \geq 5$.
- Moreover, induced C_4's of K_4's may not be edge 3-admissible.



Corollary: For any induced $X \in \{C_4, K_4\}$ in an edge 3-spanner in G, there is a vertex in X adjacent only to vertices in X.

How about K_4's and C_4's?

- Note that every edge 3-spanner graph is K_n-free and C_n-free, for $n \geq 5$.
- Moreover, induced C_4's of K_4's may not be edge 3-admissible.

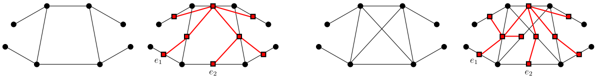


Corollary: For any induced $X \in \{C_4, K_4\}$ in an edge 3-spanner in G, there is a vertex in X adjacent only to vertices in X.

Equivalently: If X is a C_4, there is a vertex of degree 2.

How about K_4's and C_4's?

- Note that every edge 3-spanner graph is K_n-free and C_n-free, for $n \geq 5$.
- Moreover, induced C_4's of K_4's may not be edge 3-admissible.

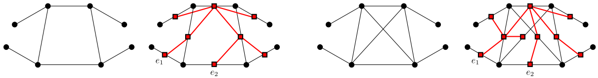


Corollary: For any induced $X \in \{C_4, K_4\}$ in an edge 3-spanner in G, there is a vertex in X adjacent only to vertices in X.

Equivalently: If X is a C_4, there is a vertex of degree 2. If X is a K_4, there is a vertex of degree 3.

How about K_4's and C_4's?

- Note that every edge 3-spanner graph is K_n-free and C_n-free, for $n \geq 5$.
- Moreover, induced C_4's of K_4's may not be edge 3-admissible.



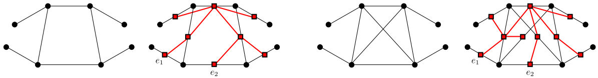
Corollary: For any induced $X \in \{C_4, K_4\}$ in an edge 3-spanner in G, there is a vertex in X adjacent only to vertices in X.

Equivalently: If X is a C_4, there is a vertex of degree 2. If X is a K_4, there is a vertex of degree 3.

Hence:

How about K_4's and C_4's?

- Note that every edge 3-spanner graph is K_n-free and C_n-free, for $n \geq 5$.
- Moreover, induced C_4's of K_4's may not be edge 3-admissible.



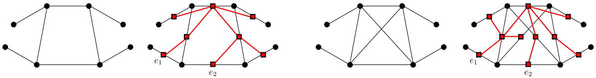
Corollary: For any induced $X \in \{C_4, K_4\}$ in an edge 3-spanner in G, there is a vertex in X adjacent only to vertices in X.

Equivalently: If X is a C_4, there is a vertex of degree 2. If X is a K_4, there is a vertex of degree 3.

Hence: For each induced X satisfying corollary above, apply the following construction:

How about K_4's and C_4's?

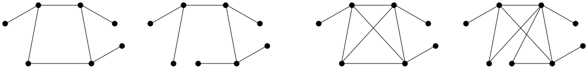
- Note that every edge 3-spanner graph is K_n-free and C_n-free, for $n \geq 5$.
- Moreover, induced C_4's of K_4's may not be edge 3-admissible.



Corollary: For any induced $X \in \{C_4, K_4\}$ in an edge 3-spanner in G, there is a vertex in X adjacent only to vertices in X.

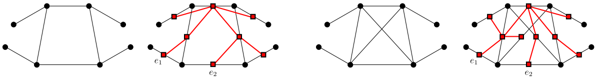
Equivalently: If X is a C_4, there is a vertex of degree 2. If X is a K_4, there is a vertex of degree 3.

Hence: For each induced X satisfying corollary above, apply the following construction:



How about K_4's and C_4's?

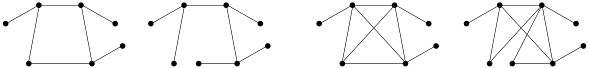
- Note that every edge 3-spanner graph is K_n-free and C_n-free, for $n \geq 5$.
- Moreover, induced C_4's of K_4's may not be edge 3-admissible.



Corollary: For any induced $X \in \{C_4, K_4\}$ in an edge 3-spanner in G, there is a vertex in X adjacent only to vertices in X.

Equivalently: If X is a C_4, there is a vertex of degree 2. If X is a K_4, there is a vertex of degree 3.

Hence: For each induced X satisfying corollary above, apply the following construction:



Solving edge 3-admissibility

- A 2-tree is a graph obtained from a K_3 by repeatedly adding vertices in such a way that each added vertex v has exactly 2 neighbors defining a clique of size 3.

Solving edge 3-admissibility

- A 2-tree is a graph obtained from a K_3 by repeatedly adding vertices in such a way that each added vertex v has exactly 2 neighbors defining a clique of size 3. A partial 2-tree is a subgraph of a 2-tree.

Solving edge 3-admissibility

- A 2-tree is a graph obtained from a K_3 by repeatedly adding vertices in such a way that each added vertex v has exactly 2 neighbors defining a clique of size 3. A partial 2-tree is a subgraph of a 2-tree.

Lemma: Let G be edge 3-admissible. H can be obtained from G from previous construction and H is a chordal partial 2-tree graph.

Solving edge 3-admissibility

- A 2-tree is a graph obtained from a K_3 by repeatedly adding vertices in such a way that each added vertex v has exactly 2 neighbors defining a clique of size 3. A partial 2-tree is a subgraph of a 2-tree.

Lemma: Let G be edge 3-admissible. H can be obtained from G from previous construction and H is a chordal partial 2-tree graph.

Theorem: Edge 3-admissibility can be solved in polynomial time for partial 2-tree graphs.

Solving edge 3-admissibility

- A 2-tree is a graph obtained from a K_3 by repeatedly adding vertices in such a way that each added vertex v has exactly 2 neighbors defining a clique of size 3. A partial 2-tree is a subgraph of a 2-tree.

Lemma: Let G be edge 3-admissible. H can be obtained from G from previous construction and H is a chordal partial 2-tree graph.

Theorem: Edge 3-admissibility can be solved in polynomial time for partial 2-tree graphs.

Corollary: Edge 3-admissibility is polynomial-time solvable.

Edge stretch for graph classes

- We prove that some graph classes are edge 4-admissible.

Edge stretch for graph classes

- We prove that some graph classes are edge 4-admissible.
  - Since edge 3-admissible is determined in polynomial time, then their edge stretch index can be determined in polynomial time.

Edge stretch for graph classes

- We prove that some graph classes are edge 4-admissible.
  - Since edge 3-admissible is determined in polynomial time, then their edge stretch index can be determined in polynomial time.
  - Examples are: Split graphs, cographs, P_4-sparse graphs, P_4-tidy graphs, join graphs.

Edge stretch for graph classes

- We prove that some graph classes are edge 4-admissible.
  - Since edge 3-admissible is determined in polynomial time, then their edge stretch index can be determined in polynomial time.
  - Examples are: Split graphs, cographs, P_4-sparse graphs, P_4-tidy graphs, join graphs.
- We prove that some graph classes are edge 5-admissible.
  - Moreover, we characterized when these graphs are edge 4-admissible.

Edge stretch for graph classes

- We prove that some graph classes are edge 4-admissible.
  - Since edge 3-admissible is determined in polynomial time, then their edge stretch index can be determined in polynomial time.
  - Examples are: Split graphs, cographs, P_4-sparse graphs, P_4-tidy graphs, join graphs.
- We prove that some graph classes are edge 5-admissible.
  - Moreover, we characterized when these graphs are edge 4-admissible.
  - Example is: Graphs that can be partitioned into 1 independent set and 2 cliques.

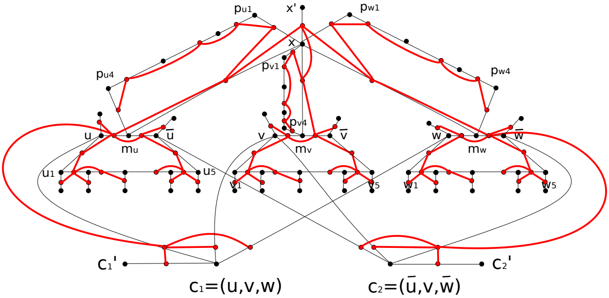
How about Bipartite Graphs?

How about bipartite graphs?

Theorem: Edge 8-admissibility is NP-complete for bipartite graphs.

How about bipartite graphs?

Theorem: Edge 8-admissibility is NP-complete for bipartite graphs.



Graph obtained from instance $I = (\{u, v, w\}, \{(u, v, w), (\overline{u}, v, \overline{w}\})$ and an edge tree 8-spanner of it in red.

Ongoing work

1. What is the relation between the stretch index and the edge stretch index parameters?

Ongoing work

1. What is the relation between the stretch index and the edge stretch index parameters?
2. Last theorem can be adapated (by subdividing the edges) in order to settle that edge $2k$-admissibility is NP-complete, for $k \geq 5$.

Ongoing work

1. What is the relation between the stretch index and the edge stretch index parameters?
2. Last theorem can be adapated (by subdividing the edges) in order to settle that edge $2k$-admissibility is NP-complete, for $k \geq 5$.
   - How about edge $(2k+1)$-admissibility?

Thank you!