Immediate versus Eventual Conversion: Comparing Geodetic and Hull Numbers in P_3 Convexity

Dr. Carmen C. Centeno

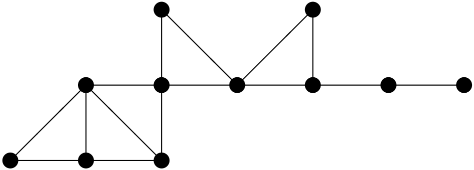
Federal University of Rio de Janeiro

joint work with L. D. Penso, D. B. Rautenbach and V. G. P. de Sá

Starting from the Start - P_3-Convexity

- What is the P_3-convex Hull of a subset C of vertices from $G = (V(G), E(G))$?

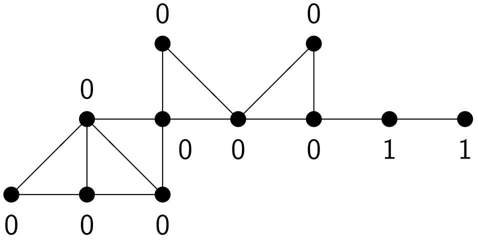
The P_3-convex Hull of C is obtained by iteratively adding to C^* every vertex which neighbours two vertices in C^*, where initially $C^* = C$.



Starting from the Start - P_3-Convexity

- What is the P_3-convex Hull of a subset C of vertices from $G = (V(G), E(G))$?

The P_3-convex Hull of C is obtained by iteratively adding to C^* every vertex which neighbours two vertices in C^*, where initially $C^* = C$.

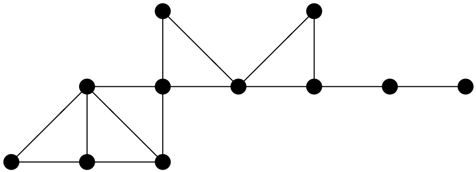


Example 1

Starting from the Start - P_3-Convexity

- What is the P_3-convex Hull of a subset C of vertices from $G = (V(G), E(G))$?

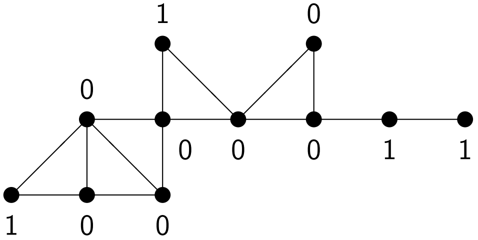
The P_3-convex Hull of C is obtained by iteratively adding to C^* every vertex which neighbours two vertices in C^*, where initially $C^* = C$.



Starting from the Start - P_3-Convexity

- What is the P_3-convex Hull of a subset C of vertices from $G = (V(G), E(G))$?

The P_3-convex Hull of C is obtained by iteratively adding to C^* every vertex which neighbours two vertices in C^*, where initially $C^* = C$.

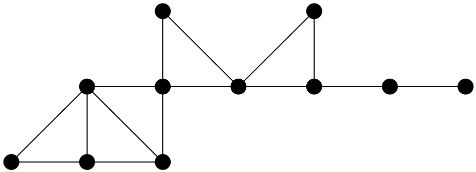


Example 2

Starting from the Start - P_3-Convexity

- What is the P_3-convex Hull of a subset C of vertices from $G = (V(G), E(G))$?

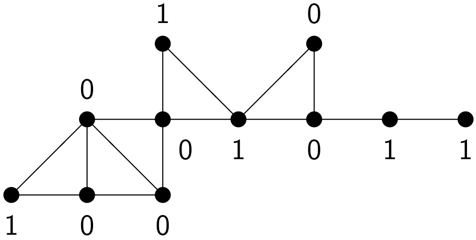
The P_3-convex Hull of C is obtained by iteratively adding to C^* every vertex which neighbours two vertices in C^*, where initially $C^* = C$.



Starting from the Start - P_3-Convexity

- What is the P_3-convex Hull of a subset C of vertices from $G = (V(G), E(G))$?

The P_3-convex Hull of C is obtained by iteratively adding to C^* every vertex which neighbours two vertices in C^*, where initially $C^* = C$.

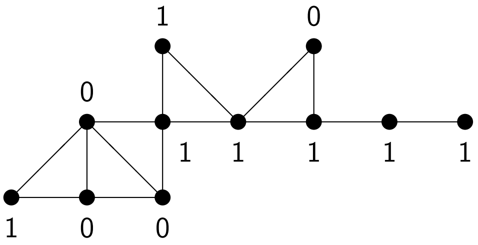


Example 3

Starting from the Start - P_3-Convexity

- What is the P_3-convex Hull of a subset C of vertices from $G = (V(G), E(G))$?

The P_3-convex Hull of C is obtained by iteratively adding to C^* every vertex which neighbours two vertices in C^*, where initially $C^* = C$.

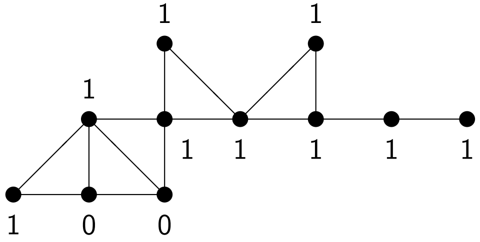


Example 3

Starting from the Start - P_3-Convexity

- What is the P_3-convex Hull of a subset C of vertices from $G = (V(G), E(G))$?

The P_3-convex Hull of C is obtained by iteratively adding to C^* every vertex which neighbours two vertices in C^*, where initially $C^* = C$.

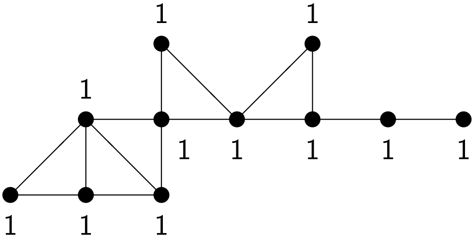


Example 3

Starting from the Start - P_3-Convexity

- What is the P_3-convex Hull of a subset C of vertices from $G = (V(G), E(G))$?

The P_3-convex Hull of C is obtained by iteratively adding to C^* every vertex which neighbours two vertices in C^*, where initially $C^* = C$.



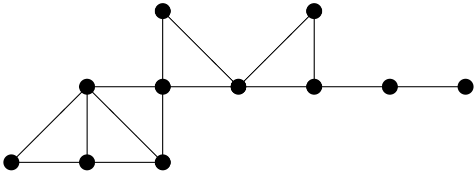
Example 3

Starting from the Start - P_3-Hull

- What is a P_3-Hull Set and the P_3-Hull Number $h(G)$ of a graph $G = (V(G), E(G))$?

For a P_3-Hull Set of G the P_3-convex Hull equals $V(G)$.

The P_3-Hull Number is the minimum cardinality of a P_3-Hull Set in G.

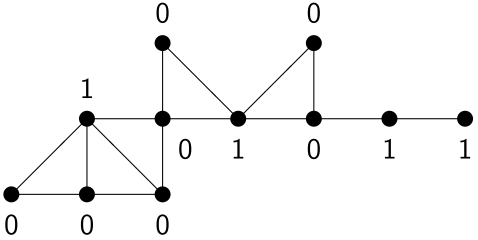


Starting from the Start - P_3-Hull

- What is a P_3-Hull Set and the P_3-Hull Number $h(G)$ of a graph $G = (V(G), E(G))$?

For a P_3-Hull Set of G the P_3-convex Hull equals $V(G)$.

The P_3-Hull Number is the minimum cardinality of a P_3-Hull Set in G.



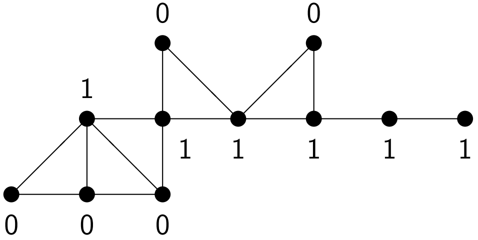
P_3-Hull Number is 4

Starting from the Start - P_3-Hull

- What is a P_3-Hull Set and the P_3-Hull Number $h(G)$ of a graph $G = (V(G), E(G))$?

For a P_3-Hull Set of G the P_3-convex Hull equals $V(G)$.

The P_3-Hull Number is the minimum cardinality of a P_3-Hull Set in G.



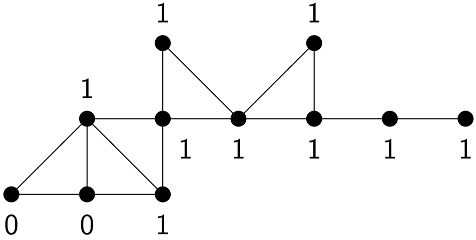
P_3-Hull Number is 4

Starting from the Start - P_3-Hull

- What is a P_3-Hull Set and the P_3-Hull Number $h(G)$ of a graph $G = (V(G), E(G))$?

For a P_3-Hull Set of G the P_3-convex Hull equals $V(G)$.

The P_3-Hull Number is the minimum cardinality of a P_3-Hull Set in G.



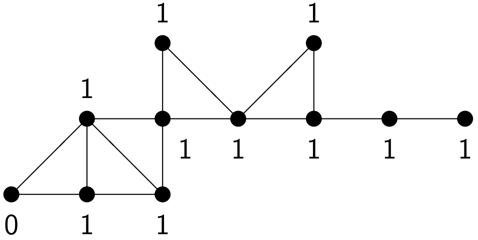
P_3-Hull Number is 4

Starting from the Start - P_3-Hull

- What is a P_3-Hull Set and the P_3-Hull Number $h(G)$ of a graph $G = (V(G), E(G))$?

For a P_3-Hull Set of G the P_3-convex Hull equals $V(G)$.

The P_3-Hull Number is the minimum cardinality of a P_3-Hull Set in G.



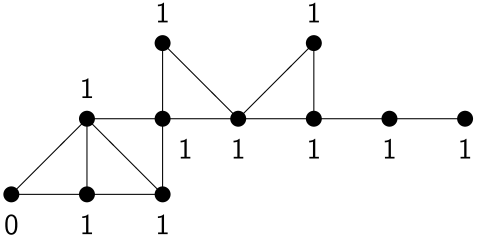
P_3-Hull Number is 4

Starting from the Start - P_3-Hull

- What is a P_3-Hull Set and the P_3-Hull Number $h(G)$ of a graph $G = (V(G), E(G))$?

For a P_3-Hull Set of G the P_3-convex Hull equals $V(G)$.

The P_3-Hull Number is the minimum cardinality of a P_3-Hull Set in G.



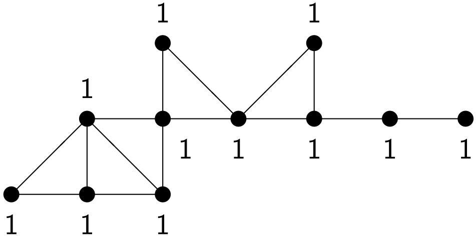
P_3-Hull Number is 4

Starting from the Start - P_3-Hull

- What is a P_3-Hull Set and the P_3-Hull Number $h(G)$ of a graph $G = (V(G), E(G))$?

For a P_3-Hull Set of G the P_3-convex Hull equals $V(G)$.

The P_3-Hull Number is the minimum cardinality of a P_3-Hull Set in G.



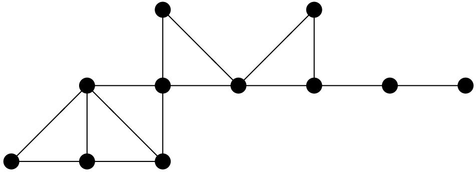
P_3-Hull Number is 4

Starting from the Start - P_3-Geodetic

- What is a P_3-Geodetic Set and the P_3-Geodetic Number of a graph $G = (V(G), E(G))$?

In a P_3-Geodetic Set of G every vertex of G either belongs to the set or has two neighbours in the set.

The P_3-Geodetic Number $g(G)$ is the minimum cardinality of a P_3-Geodetic Set in G. It equals the 2-Domination Number.

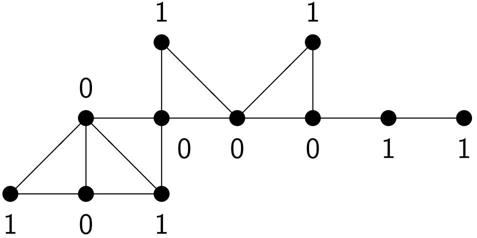


Starting from the Start - P_3-Geodetic

- What is a P_3-Geodetic Set and the P_3-Geodetic Number of a graph $G = (V(G), E(G))$?

In a P_3-Geodetic Set of G every vertex of G either belongs to the set or has two neighbours in the set.

The P_3-Geodetic Number $g(G)$ is the minimum cardinality of a P_3-Geodetic Set in G. It equals the 2-Domination Number.



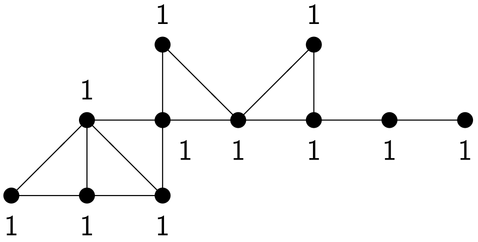
P_3-Geodetic Number is 6

Starting from the Start - P_3-Geodetic

- What is a P_3-Geodetic Set and the P_3-Geodetic Number of a graph $G = (V(G), E(G))$?

In a P_3-Geodetic Set of G every vertex of G either belongs to the set or has two neighbours in the set.

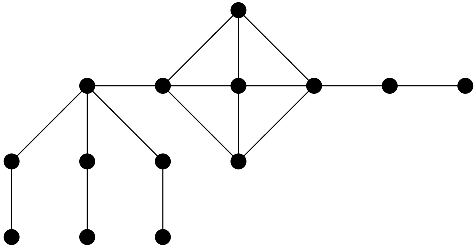
The P_3-Geodetic Number $g(G)$ is the minimum cardinality of a P_3-Geodetic Set in G. It equals the 2-Domination Number.



P_3-Geodetic Number is 6

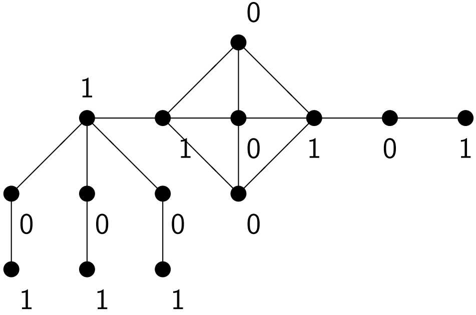
Main Feature - Question 1

- In this talk: for which graphs is P_3-Geodetic equal to P_3-Hull ($g(G) = h(G)$)?



Main Feature - Question 1

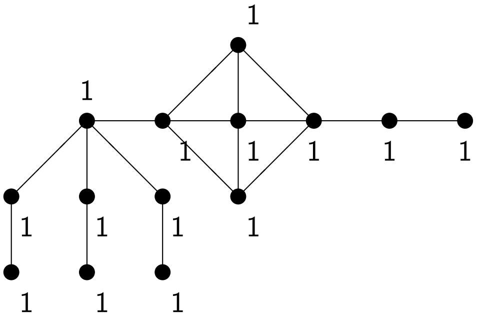
- In this talk: for which graphs is P_3-Geodetic equal to P_3-Hull ($g(G) = h(G)$)?



P_3-Geodetic Equals P_3-Hull ($g(G) = h(G) = 7$)

Main Feature - Question 1

- In this talk: for which graphs is P_3-Geodetic equal to P_3-Hull ($g(G) = h(G)$)?



P_3-Geodetic Equals P_3-Hull ($g(G) = h(G) = 7$)

Main Feature - Question 1

Clearly, every geodetic set is a hull set, which implies

$h(G) \leq g(G)$

for every graph G.

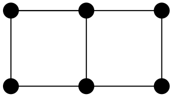
Both parameters for other convexities are computationally hard in general!

(Efficient algorithms are only known for very few exceptions: only two.)

Main Feature - Question 2

- For which all induced subgraphs have both numbers equal?

Are there many? Some examples: a star, a path, a cycle or ...



All Induced Subgraphs have Both Numbers Equal

Fundaments

Let W be a geodetic set of G of minimum order and let $B = V(G) \setminus W$.

Fundaments

Let W be a geodetic set of G of minimum order and let $B = V(G) \setminus W$.

By definition, every vertex in B has at least two neighbors in W. Therefore, G has a spanning bipartite subgraph G_0 with bipartition $V(G_0) = W \cup B$ such that every vertex in B has degree exactly 2 in G_0.

Fundaments

Let W be a geodetic set of G of minimum order and let $B = V(G) \setminus W$.

By definition, every vertex in B has at least two neighbors in W. Therefore, G has a spanning bipartite subgraph G_0 with bipartition $V(G_0) = W \cup B$ such that every vertex in B has degree exactly 2 in G_0.

Let E_1 denote the set of edges in $E(G) \setminus E(G_0)$ between vertices in the same component of G_0 and let E_2 denote the set of edges in $E(G) \setminus E(G_0)$ between vertices in distinct components of G_0.

Fundaments

Let W be a geodetic set of G of minimum order and let $B = V(G) \setminus W$.

By definition, every vertex in B has at least two neighbors in W. Therefore, G has a spanning bipartite subgraph G_0 with bipartition $V(G_0) = W \cup B$ such that every vertex in B has degree exactly 2 in G_0.

Let E_1 denote the set of edges in $E(G) \setminus E(G_0)$ between vertices in the same component of G_0 and let E_2 denote the set of edges in $E(G) \setminus E(G_0)$ between vertices in distinct components of G_0.

Note that, by construction, W is a geodetic set of G_0.

Fundaments

Since $|W| = g(G) = h(G) \leq h(G_0) \leq g(G_0) \leq |W|$, we obtain $h(G_0) = g(G_0) = |W|$, that is, G_0 has no geodetic set and no hull set of order less than $|W|$.

Fundaments

Since $|W| = g(G) = h(G) \leq h(G_0) \leq g(G_0) \leq |W|$, we obtain $h(G_0) = g(G_0) = |W|$, that is, G_0 has no geodetic set and no hull set of order less than $|W|$.

Thus, if C is a component of G_0, then $W \cap V(C)$ is a minimum geodetic set of C as well as a minimum hull set of C.

Key Idea

Let $\mathcal{G}_0$ denote the set of all bipartite graphs G_0 with a fixed bipartition $V(G_0) = B \cup W$ such that every vertex in B has degree exactly 2.

Key Idea

Let $\mathcal{G}_0$ denote the set of all bipartite graphs G_0 with a fixed bipartition $V(G_0) = B \cup W$ such that every vertex in B has degree exactly 2.

We are going to generate exactly all graphs such that $g(G) = h(G)$ from this set of bipartite graphs of type G_0, by applying Operations on it!

Key Idea

Let $\mathcal{G}_0$ denote the set of all bipartite graphs G_0 with a fixed bipartition $V(G_0) = B \cup W$ such that every vertex in B has degree exactly 2.

We are going to generate exactly all graphs such that $g(G) = h(G)$ from this set of bipartite graphs of type G_0, by applying Operations on it!

With that, we will be able to recognize the class in polynomial time!

Operations for Generation and Recognition

For that, we consider four distinct operations that can be applied to a graph G_0 from $\mathcal{G}_0$.

Operations for Generation and Recognition

For that, we consider four distinct operations that can be applied to a graph G_0 from $\mathcal{G}_0$.

- **Operation** $\mathcal{O}_1$
  Add one arbitrary edge to G_0.

Operations for Generation and Recognition

For that, we consider four distinct operations that can be applied to a graph G_0 from $\mathcal{G}_0$.

- **Operation** $\mathcal{O}_1$
  Add one arbitrary edge to G_0.

- **Operation** $\mathcal{O}_1'$
  Select two vertices w_1 and w_2 from W and arbitrarily add new edges between vertices in $\{w_1, w_2\} \cup (N_{G_0}(w_1) \cap N_{G_0}(w_2))$.

Operations for Generation and Recognition

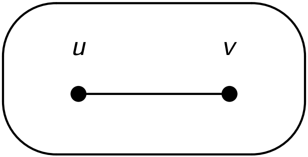
- **Operation** $\mathcal{O}_2$
  Add one arbitrary edge between vertices in distinct components of G_0.

Operations for Generation and Recognition

- **Operation** $\mathcal{O}_2$
  Add one arbitrary edge between vertices in distinct components of G_0.

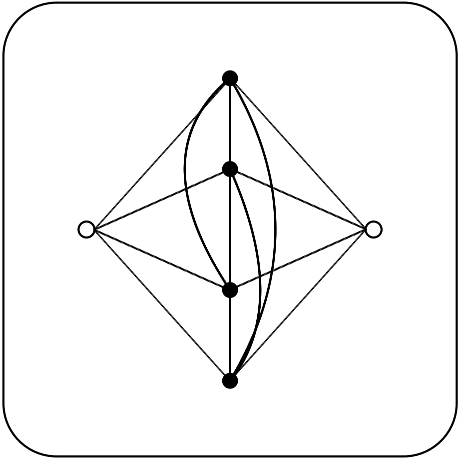
- **Operation** $\mathcal{O}_3$
  Choose a non-empty subset X of B such that all vertices in X are cut vertices of G_0 and no two vertices in X lie in the same component of G_0. Add arbitrary edges between vertices in X so that X induces a connected subgraph of the resulting graph. For every component C of G_0 that does not contain a vertex from X, add one arbitrary edge between a vertex in C and a vertex in X.

O_1

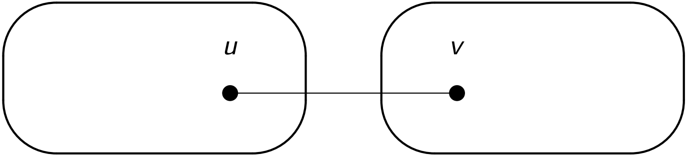


uv
$=$
bb
ww
wb

O_1'

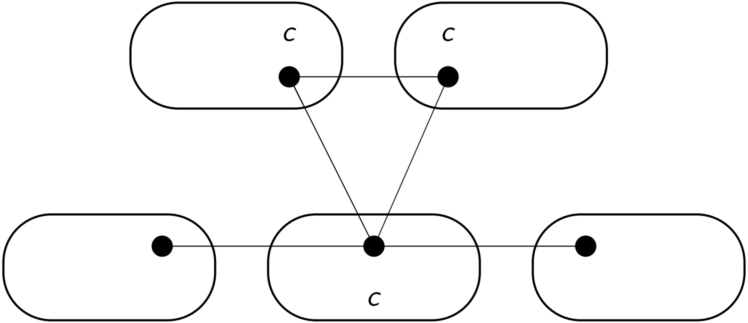


O_2



uv
$=$
bb
ww
wb

O_3



c = cutvertex

Classes of Graphs obtained with Operations

- Let $\mathcal{G}_1$ denote the set of graphs that are obtained by applying operation $\mathcal{O}_1$ once to a connected graph G_0 in $\mathcal{G}_0$.
- Let $\mathcal{G}_1'$ denote the set of graphs that are obtained by applying operation $\mathcal{O}_1'$ once to a connected graph G_0 in $\mathcal{G}_0$.
- Let $\mathcal{G}_2$ denote the set of graphs that are obtained by applying operation $\mathcal{O}_2$ once to a graph G_0 in $\mathcal{G}_0$ that has exactly two components.
- Let $\mathcal{G}_3$ denote the set of graphs that are obtained by applying operation $\mathcal{O}_3$ once to a graph G_0 in $\mathcal{G}_0$ that has at least three components. Note that $\mathcal{O}_3$ can only be applied if G_0 has at least one cut vertex that belongs to B.

Since the operation $\mathcal{O}_1'$ allows that no edges are added, the set $\mathcal{G}_1'$ contains all connected graphs in $\mathcal{G}_0$.

Classes of Graphs obtained with Operations

Finally, let

$$\mathcal{G} = \mathcal{G}_1 \cup \mathcal{G}_1' \cup \mathcal{G}_2 \cup \mathcal{G}_3. \tag{1}$$

Theorem

Let be

$$\mathcal{H} = \{G \mid G \text{ is a connected graph with } h(G) = g(G)\}.$$

Theorem

$\mathcal{G} \subseteq \mathcal{H}$.

Lemmas

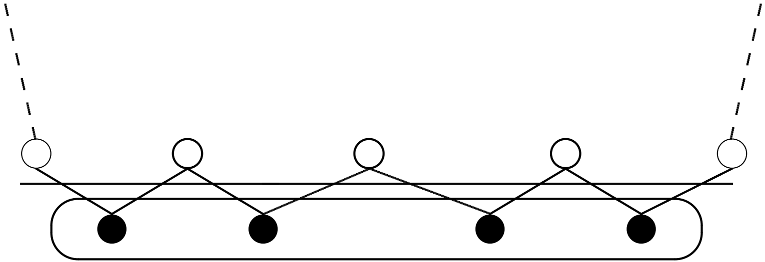
Lemma

Let C be a component of G_0.

(i) *No two vertices in C are incident with edges in E_2.*

(ii) *If some vertex u in C is incident with at least two edges in E_2, then u belongs to B and u is a cut vertex of C.*

Key Idea for Lemma 1



Lemmas

Lemma

If G_0 is not connected, no two vertices in W that belong to the same component of G_0 are adjacent.

Lemma

If G_0 is not connected and C is a component of G_0, then there are no two vertices w in $V(C) \cap W$ and b in $V(C) \cap B$ such that $wb \in E(G) \setminus E(G_0)$.

Lemmas

Lemma

Let G_0 be disconnected and let b and b' be two vertices in B that belong to the same component C of G_0 satisfying $bb' \in E_1$.

(i) *Neither b nor b' is incident with an edge in E_2.*

(ii) *If some vertex w in $V(C) \cap W$ is incident with an edge in E_2 and $P : w_1 b_1 \ldots w_l b_l$ is a path in C between $w = w_1$ and a vertex b_l in $\{b, b'\}$, then w_l is adjacent to both b and b', and C contains no path between b and b' that does not contain w_l.*

(iii) *If some vertex b'' in $(V(C) \cap B) \setminus \{b, b'\}$ is incident with an edge in E_2 and $P : b_1 w_1 \ldots w_{l-1} b_l$ is a path in C between $b'' = b_1$ and a vertex b_l in $\{b, b'\}$, then w_{l-1} is adjacent to both b and b' and C contains no path between b and b' that does not contain w_{l-1}.*

Lemmas

Lemma

If C is a component of G_0, then there are no two vertices w and w' of C that belong to W and two edges e and e' that belong to $E(G) \setminus E(G_0)$ such that w is incident with e, w' is incident with e', and e' is distinct from ww'.

Lemma

If C is a component of G_0, then there are no two edges wb and wb' that belong to $E(G) \setminus E(G_0)$ with $w \in W \cap V(C)$ and $b, b' \in B \cap V(C)$.

Lemma

If G_0 is connected and G is triangle-free, then there are no two edges ww' and bb' in G with $w, w' \in W$ and $b, b' \in B$.

Lemmas

Lemma

If G_0 is connected and G is triangle-free, then there are no two edges wb and $b'b''$ in G with $w \in W$ and $b, b', b'' \in B$.

Lemma

If G_0 is connected and G is triangle-free, then there are no two distinct edges bb' and $b''b'''$ in G with $b, b', b'', b''' \in B$.

Question1: Corollary

Corollary

If $\mathcal{T}$ denotes the set of all triangle-free graphs, then $\mathcal{G} \cap \mathcal{T} = \mathcal{H} \cap \mathcal{T}$.

Question1: Polynomial Recognition

Let G be a given connected triangle-free input graph. By the previous Corollary, the graph G belongs to $\mathcal{H}$ if and only if either G belongs to $\mathcal{G}_0 \cup \mathcal{G}_1 \cup \mathcal{G}_2$ or G belongs to $\mathcal{G}_3$.

Lemma

It can be checked in polynomial time whether $G \in \mathcal{G}_0 \cup \mathcal{G}_1 \cup \mathcal{G}_2$.

Question1: Polynomial Recognition

In view of Lemma 12, we may assume from now on that G does not belong $\mathcal{G}_0 \cup \mathcal{G}_1 \cup \mathcal{G}_2$. The following lemma is an immediate consequence of the definition of operation $\mathcal{O}_3$.

Question1: Polynomial Recognition

Lemma

If G belongs to $\mathcal{G}_3$, then there is a vertex x of G of degree at least three and two edges $e_l = xy_l$ and $e_r = xy_r$ of G incident with x such that, in the graph G' that arises by deleting from G all edges incident with x except for e_l and e_r, the component $C(x, e_l, e_r)$ of G' that contains x has the following properties:

(i) *x is a cut vertex of $C(x, e_l, e_r)$;*

(ii) *$C(x, e_l, e_r)$ has a unique bipartition with partite sets $B_l \cup \{x\} \cup B_r$ and $W_l \cup W_r$;*

(iii) *Every vertex in $B_l \cup \{x\} \cup B_r$ has degree 2 in $C(x, e_l, e_r)$;*

(iv) *$B_l \cup W_l$ and $B_r \cup W_r$ are the vertex sets of the two components of $C(x, e_l, e_r) - x$ such that $y_l \in W_l$ and $y_r \in W_r$;*

(v) *None of the deleted edges connects x to a vertex from $V(C(x, e_l, e_r)) \setminus \{x\}$;*

(vi) *W_l and W_r both contain a vertex of odd degree.*

Question1: Polynomial Recognition

The key observation for the completion of the algorithm is the following lemma, which states that the properties from Lemma 13 uniquely characterize the elements of X.

Lemma

If G belongs to $\mathcal{G}_3$ and a vertex x of G of degree at least three and two edges $e_l = xy_l$ and $e_r = xy_r$ of G incident with x are such that properties (i) to (vi) from Lemma 13 hold, then

(i) *G is obtained by applying operation $\mathcal{O}_3$ to a graph G_0 in $\mathcal{G}_0$ with at least three components such that x belongs to the set X used by operation $\mathcal{O}_3$ and*

(ii) *$C(x, e_l, e_r)$ defined as in Lemma 13 is the component of G_0 that contains x.*

Question1: Polynomial Recognition

Theorem

For a given triangle-free graph G, it can be checked in polynomial time whether $h(G) = g(G)$ holds.

Question1: Polynomial Recognition

Proof: Clearly, we can consider each component of G separately and may therefore assume that G is connected. Let n denote the order of G. By Lemma 12, we can check in $O(n^2)$ time whether G belongs $\mathcal{G}_0 \cup \mathcal{G}_1 \cup \mathcal{G}_2$. If this is the case, then Corollary 11 implies $h(G) = g(G)$. Hence, we may assume that G does not belong to $\mathcal{G}_0 \cup \mathcal{G}_1 \cup \mathcal{G}_2$. Note that there are $O(n^3)$ choices for a vertex x of G and two incident edges e_l and e_r of G. Furthermore, note that for every individual choice of the triple (x, e_l, e_r), the properties (i) to (vi) from Lemma 13 can be checked in $O(n)$ time. Therefore, by Lemmas 13 and 14, in $O(n^4)$ time, we can

- either determine that no choice of (x, e_l, e_r) satisfies the conclusion of Lemma 13, which, by Corollary 11, implies $h(G) \neq g(G)$,
- or find a suitable triple (x, e_l, e_r) and reduce the instance G to a smaller instance $G^- = G - V(C(x, e_l, e_r))$.

Since the order of G^- is at least three less than n, this leads to an overall running time of $O(n^5)$. $\square$

Question2: All Induced Subgraphs

It is an easy exercise to prove $h(G) = g(G)$ whenever G is a path, a cycle, or a star.

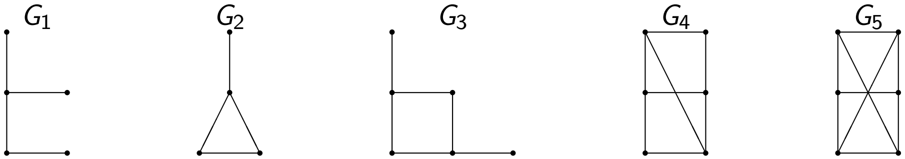


Figure: The five graphs $G_1, \ldots, G_5$.

Theorem

If G is a graph, then $h(H) = g(H)$ for every induced subgraph H of G if and only if G is $\{G_1, \ldots, G_5\}$-free.

Thank you for the attention!