Exact algorithm for the Maximum Induced Planar Subgraph Problem

Fedor Fomin Ioan Todinca Yngve Villanger

University of Bergen, Université d'Orléans

Workshop on Graph Decompositions, CIRM, October 19th, 2010

Outline of the result

Theorem

There is an $O(1.73601^n)$ algorithm for the MAX INDUCED PLANAR SUBGRAPH *problem.*

Ingredients:

1. [Fomin, Villanger, STACS 2010]: an $O(1.73601^n \cdot n^{t+3})$ algorithm for the MAX INDUCED SUBGRAPH OF THEEWIDTH $\leq t$
2. [Robertson, Seymour 1986; Fomin, Thilikos 2006]: planar graphs have treewidth $O(\sqrt{n})$

Outline of the result

Theorem

There is an $O(1.73601^n)$ algorithm for the MAX INDUCED PLANAR SUBGRAPH *problem.*

Ingredients:

1. [Fomin, Villanger, STACS 2010]: an $O(1.73601^n \cdot n^{t+3})$ algorithm for the MAX INDUCED SUBGRAPH OF THEEWIDTH $\leq t$
2. [Robertson, Seymour 1986; Fomin, Thilikos 2006]: planar graphs have treewidth $O(\sqrt{n})$
3. Combinatorial results on minimal triangulations and potential maximal cliques of planar graphs.

Outline of the result

Theorem

There is an $O(1.73601^n)$ algorithm for the MAX INDUCED PLANAR SUBGRAPH *problem.*

Ingredients:

1. [Fomin, Villanger, STACS 2010]: an $O(1.73601^n \cdot n^{t+3})$ algorithm for the MAX INDUCED SUBGRAPH OF THEEWIDTH $\leq t$
2. [Robertson, Seymour 1986; Fomin, Thilikos 2006]: planar graphs have treewidth $O(\sqrt{n})$
3. Combinatorial results on minimal triangulations and potential maximal cliques of planar graphs.
4. An algorithm putting everything together

Motivation and related work

Exact algorithms for NP-hard problems

- [Gödel, 1959]: "how strongly in general the number of steps in finite combinatorial problems can be reduced with respect to simple exhaustive search?"
- Nice combinatorics, nice algorithmic techniques

MAX INDUCED SUBGRAPH WITH PROPERTY Π

- MAX INDEPENDENT SET [Moon, Moser, 1965; Fomin, Grandoni, Kratsch 2009]
- MAX FEEDBACK VERTEX SET [Razgon 2006; Fomin, Villanger 2010]
- MAX INDUCED SUBGRAPH OF TREEWIDTH $\leq t$ [Fomin, Villanger 2010]

Max Induced Subgraph of tw $\leq t$

[Fomin, Villanger 2010]
F an induced subgraph of G, H_F a minimal triangulation of F.

- There is a minimal triangulation H_G of G such that H_F is an induced subgraph of G; we say thay H_F and H_G are compatible

MAX INDUCED SUBGRAPH OF tw $\leq t$

[Fomin, Villanger 2010]

F an induced subgraph of G, H_F a minimal triangulation of F.

- There is a minimal triangulation H_G of G such that H_F is an induced subgraph of G; we say thay H_F and H_G are compatible
- Consequence: for every bag Ω_G of H_G, $\Omega_G \cap V(F)$ is contained is some bag Ω_F of H_F

Max Induced Subgraph of tw $\leq t$

[Fomin, Villanger 2010]

F an induced subgraph of G, H_F a minimal triangulation of F.

- There is a minimal triangulation H_G of G such that H_F is an induced subgraph of G; we say thay H_F and H_G are compatible
- Consequence: for every bag Ω_G of H_G, $\Omega_G \cap V(F)$ is contained is some bag Ω_F of H_F
- Fix a minimal triangulation H_G.

MAX INDUCED SUBGRAPH OF tw $\leq t$

[Fomin, Villanger 2010]

F an induced subgraph of G, H_F a minimal triangulation of F.

- There is a minimal triangulation H_G of G such that H_F is an induced subgraph of G; we say thay H_F and H_G are compatible
- Consequence: for every bag Ω_G of H_G, $\Omega_G \cap V(F)$ is contained is some bag Ω_F of H_F
- Fix a minimal triangulation H_G. One can compute a maximum induced subgraph F of G s.t. $\text{tw}(F) \leq t$ and F has an optimal triangulation compatible with H_G in $O(n^{t+cst})$ time

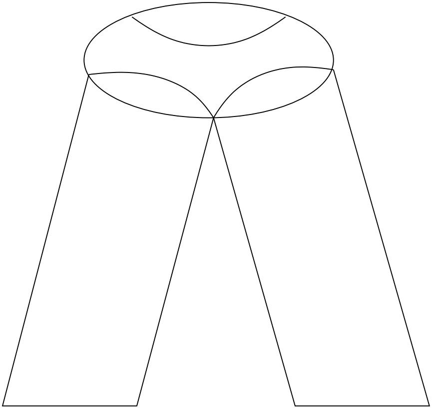
Max Induced Subgraph of tw $\leq t$

[Fomin, Villanger 2010]

F an induced subgraph of G, H_F a minimal triangulation of F.

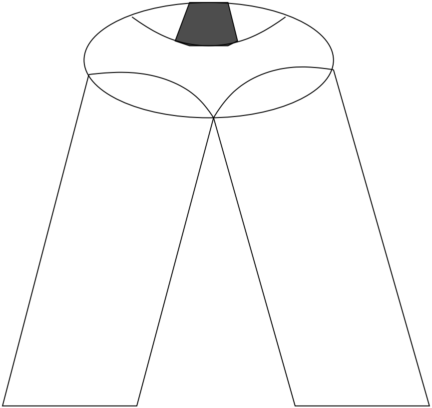
- There is a minimal triangulation H_G of G such that H_F is an induced subgraph of G; we say thay H_F and H_G are compatible
- Consequence: for every bag Ω_G of H_G, $\Omega_G \cap V(F)$ is contained is some bag Ω_F of H_F
- Fix a minimal triangulation H_G. One can compute a maximum induced subgraph F of G s.t. $\text{tw}(F) \leq t$ and F has an optimal triangulation compatible with H_G in $O(n^{t+cst})$ time
- Dynamic programming over all minimal triangulations H_G using potential maximal cliques: $O(1.73601^n \cdot n^{t+cst})$ time

Computing F for a fixed tree decomposition of G



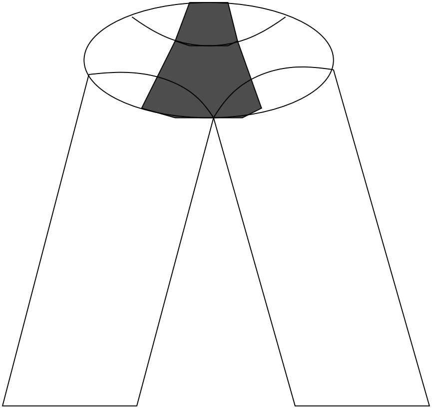
- $\alpha(S, W, C)$: the size of the largest partial solution intersecting S in W

Computing F for a fixed tree decomposition of G



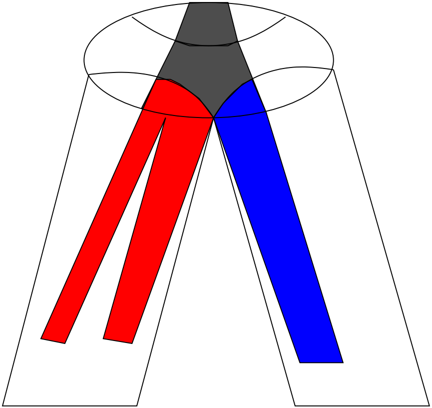
- $\alpha(S, W, C)$: the size of the largest partial solution intersecting S in W
- $\alpha(S, W, C) =$

Computing F for a fixed tree decomposition of G



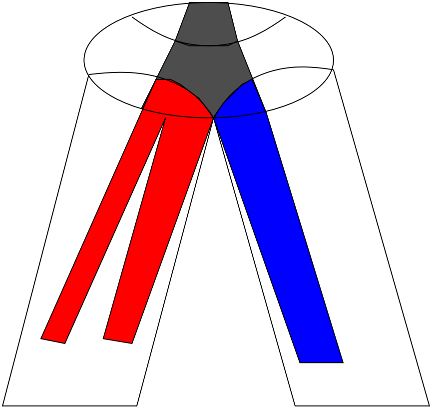
- $\alpha(S, W, C)$: the size of the largest partial solution intersecting S in W
- $\alpha(S, W, C) =$ $max_{W' \subset \Omega}(W' \oplus$

Computing F for a fixed tree decomposition of G



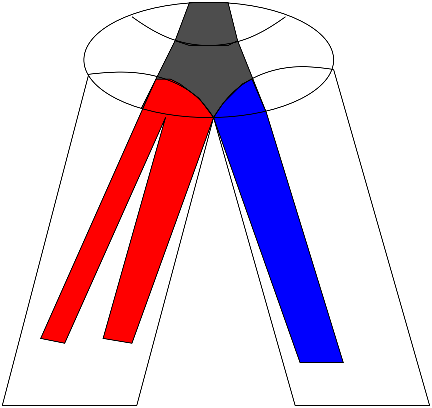
- $\alpha(S, W, C)$: the size of the largest partial solution intersecting S in W
- $\alpha(S, W, C) = max_{W' \subset \Omega}(W' \oplus \alpha(S_1, W_1, C_1) \oplus \alpha(S_2, W_2, C_2))$

Computing F for a fixed tree decomposition of G



- $\alpha(S, W, C)$: the size of the largest partial solution intersecting S in W
- $\alpha(S, W, C) = max_{W' \subset \Omega}(W' \oplus \alpha(S_1, W_1, C_1) \oplus \alpha(S_2, W_2, C_2))$
- Running time: $O(n^{t+cst})$

Computing F for a fixed tree decomposition of G



- $\alpha(S, W, C)$: the size of the largest partial solution intersecting S in W
- $\alpha(S, W, C) = max_{W' \subset \Omega}(W' \oplus \alpha(S_1, W_1, C_1) \oplus \alpha(S_2, W_2, C_2))$
- Running time: $O(n^{t+cst})$
- Also constructs a minimal triangulation of F

Browsing through all minimal triangulations of G

Definition

A vertex subset Ω_G of G is a potential maximal clique if there exists a minimal triangulation H_G such that Ω_G is a maximal clique of H_G.

- One can "browse" through all minimal triangulations of a graph, in time $O^*(\sharp p.m.c.)$ [Bouchitté, Todinca 2001, Fomin, Kratsch, Todinca 2008]
- An n-vertex graph has $O^*(1.73601^n)$ potential maximal cliques [Fomin Villanger 2010]

A Maximum Induced Subgraph of $tw \leq t$ can be computed in $O(1.73601^n \cdot n^{t+cst})$ time [Fomin Villanger 2010].

Towards an extension to the MAX INDUCED PLANAR GRAPH problem

- Good news: planar graphs have treewidth at most $3.182\sqrt{n}$ [Fomin, Thilikos 2006]; an $O(\sharp p.m.c. \cdot n^{c\sqrt{n}+cst}) = O(\sharp p.m.c. \cdot 2^{o(n)}) = O(1.73601^n)$ algorithm?

Towards an extension to the MAX INDUCED PLANAR GRAPH problem

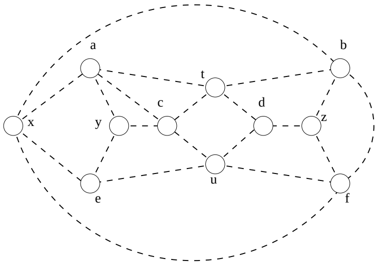
- Good news: planar graphs have treewidth at most $3.182\sqrt{n}$ [Fomin, Thilikos 2006]; an $O(\sharp p.m.c. \cdot n^{c\sqrt{n}+cst}) = O(\sharp p.m.c. \cdot 2^{o(n)}) = O(1.73601^n)$ algorithm?
- Bad news: in the algorithm of Fomin and Villanger, even when we "glue" two planar graphs, the result might not be planar.

Towards an extension to the Max Induced Planar Graph problem

- Good news: planar graphs have treewidth at most $3.182\sqrt{n}$ [Fomin, Thilikos 2006]; an $O(\sharp p.m.c. \cdot n^{c\sqrt{n}+cst}) = O(\sharp p.m.c. \cdot 2^{o(n)}) = O(1.73601^n)$ algorithm?
- Bad news: in the algorithm of Fomin and Villanger, even when we "glue" two planar graphs, the result might not be planar.

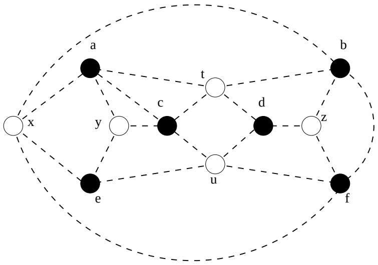
We need more tools for gluing partial solutions. Recall that we glue along potential maximal cliques of the target (planar) graph F.

Potential maximal cliques in planar graphs



(a)

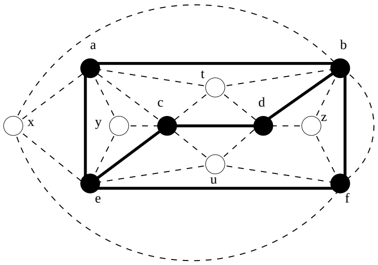
Potential maximal cliques in planar graphs



(a)

- A potential maximal clique Ω_F of a plane graph F forms a

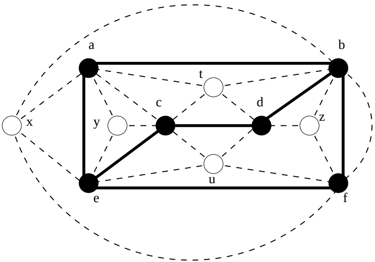
Potential maximal cliques in planar graphs



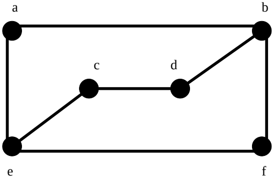
(a)

- A potential maximal clique Ω_F of a plane graph F forms a θ-structure [Bouchitté, Mazoit, Todinca 2003; . . .]

Potential maximal cliques in planar graphs



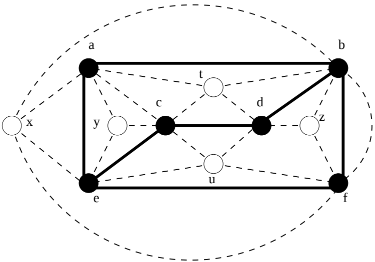
(a)



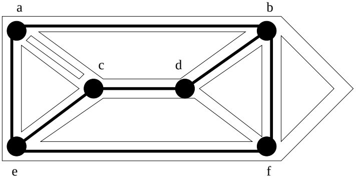
(b)

- A potential maximal clique Ω_F of a plane graph F forms a θ-structure [Bouchitté, Mazoit, Todinca 2003; ...]

Potential maximal cliques in planar graphs



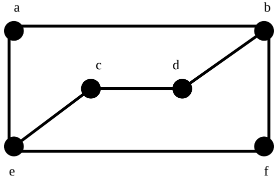
(a)



(b)

- A potential maximal clique Ω_F of a plane graph F forms a θ-structure [Bouchitté, Mazoit, Todinca 2003; ...]
- The neighbourhoods of the components of $F - \Omega_F$ correspond to pairwise non-crossing Jordan curves
- Conversely, if we draw planar pieces "inside" these curves we preserve planarity

θ-structures and neighborhood assignments

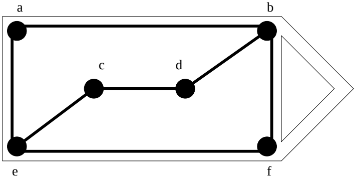


$\mathcal{S}_1 = [e, a, b, f]$
$\mathcal{S}_2 = [e, c, d, b, f]$
$\mathcal{S}_3 = [e, a, b, d, c]$

We can avoid geometry.

- θ-structures $\theta(\Omega_F)$ on Ω_F: three totally ordered subsets, sharing the extremities and forming three cyclic orderings $\mathcal{S}_i$.
- neighboorhood assignment $[\theta(\Omega_F)]$:
  - assigns to each cyclic ordering $\mathcal{S}_i$ of $\theta(\Omega_F)$ a set of pairwise non-crossing subsets of $\mathcal{S}_i$.
  - for each edge xy of $F[\Omega_F]$, the neighborhood $\{x, y\}$ is assigned to some $\mathcal{S}_i$.

θ-structures and neighborhood assignments

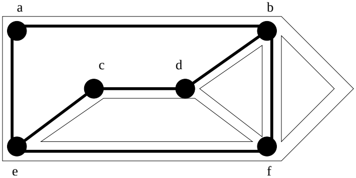


$\mathcal{S}_1 : \{b, f\}, \{a, b, f\}$

We can avoid geometry.

- θ-structures $\theta(\Omega_F)$ on Ω_F: three totally ordered subsets, sharing the extremities and forming three cyclic orderings $\mathcal{S}_i$.
- neighboorhood assignment $[\theta(\Omega_F)]$:
  - assigns to each cyclic ordering $\mathcal{S}_i$ of $\theta(\Omega_F)$ a set of pairwise non-crossing subsets of $\mathcal{S}_i$.
  - for each edge xy of $F[\Omega_F]$, the neighborhood $\{x, y\}$ is assigned to some $\mathcal{S}_i$.

θ-structures and neighborhood assignments

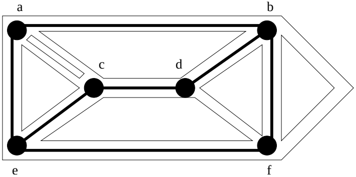


$\mathcal{S}_2 : \{d, b, f\}, \{e, c, d, f\}$

We can avoid geometry.

- θ-structures $\theta(\Omega_F)$ on Ω_F: three totally ordered subsets, sharing the extremities and forming three cyclic orderings $\mathcal{S}_i$.
- neighboorhood assignment $[\theta(\Omega_F)]$:
  - assigns to each cyclic ordering $\mathcal{S}_i$ of $\theta(\Omega_F)$ a set of pairwise non-crossing subsets of $\mathcal{S}_i$.
  - for each edge xy of $F[\Omega_F]$, the neighborhood $\{x, y\}$ is assigned to some $\mathcal{S}_i$.

θ-structures and neighborhood assignments



$\mathcal{S}_3$:
$\{a, c, e\}, \{a, c\}, \{a, b, d, c\}$

We can avoid geometry.

- θ-structures $\theta(\Omega_F)$ on Ω_F: three totally ordered subsets, sharing the extremities and forming three cyclic orderings $\mathcal{S}_i$.
- neighboorhood assignment $[\theta(\Omega_F)]$:
  - assigns to each cyclic ordering $\mathcal{S}_i$ of $\theta(\Omega_F)$ a set of pairwise non-crossing subsets of $\mathcal{S}_i$.
  - for each edge xy of $F[\Omega_F]$, the neighborhood $\{x, y\}$ is assigned to some $\mathcal{S}_i$.

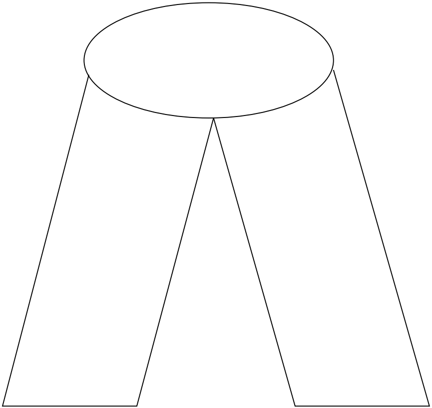
Counting neighborhood assignements

Theorem

Over all subets Ω_F of size at most $c\sqrt{n}$, over all possible θ-structures $\theta(\Omega_F)$, there are $2^{o(n)}$ possible (partial) neighborhood assignements.

- there are $\binom{n}{c\sqrt{n}}$ possible subsets Ω_F
- for each Ω_F, there are $2^{o(n)}$ posible θ-structures
- for a fixed θ-structure $\theta(\Omega_F)$, for each cyclic ordering $\mathcal{S}_i$ the number of posssible neighborhood assignements on $\mathcal{S}_i$ is upper bounded by the Catalan number $CN(|\mathcal{S}_i|) \leq 4^{|\mathcal{S}_i|}$ [Kreweras 1972].

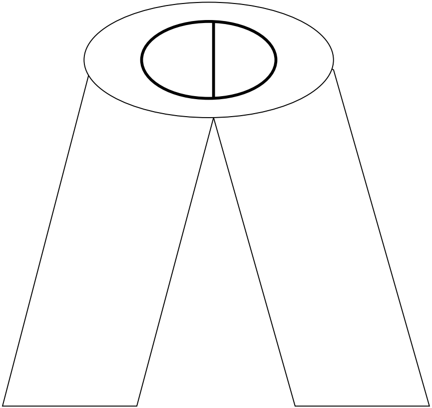
The algorithm



Maximum partial solution for

- (S, C, Ω_G)

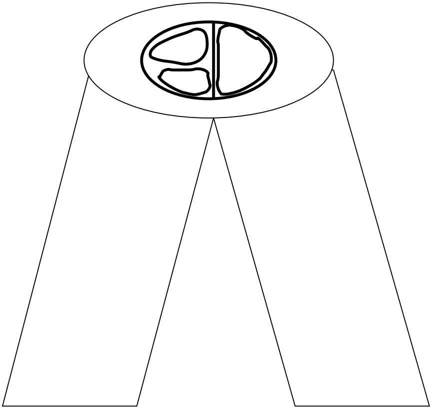
The algorithm



Maximum partial solution for

- (S, C, Ω_G)
- a θ-structure $\theta(\Omega_F)$

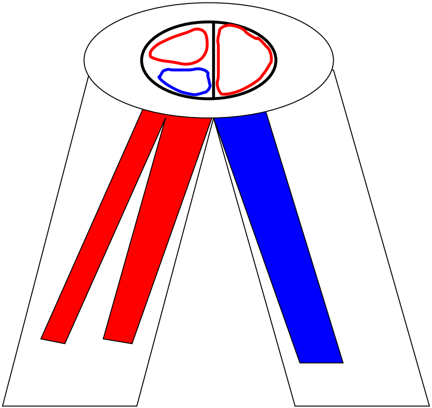
The algorithm



Maximum partial solution for

- (S, C, Ω_G)
- a θ-structure $\theta(\Omega_F)$
- a (partial) neighborhood assignment

The algorithm

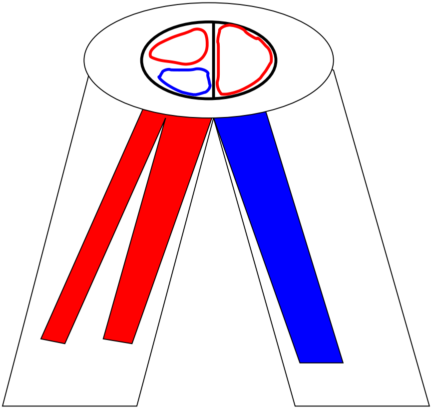


Maximum partial solution for

- (S, C, Ω_G)
- a θ-structure $\theta(\Omega_F)$
- a (partial) neighborhood assignment

Dynamic programming over all p.m.c. Ω_G of G and on all possible (partial) neighborhood assignments $[\theta(\Omega_F)]$, over all "small" subsets Ω_F.

The algorithm



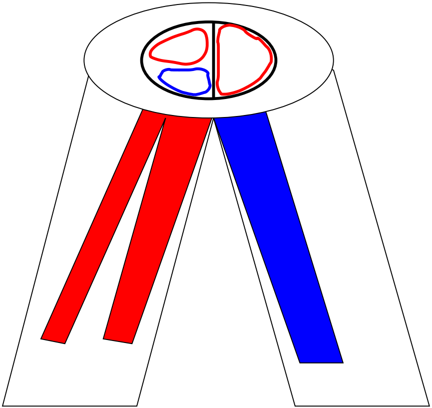
Maximum partial solution for

- (S, C, Ω_G)
- a θ-structure $\theta(\Omega_F)$
- a (partial) neighborhood assignment

Dynamic programming over all p.m.c. Ω_G of G and on all possible (partial) neighborhood assignments $[\theta(\Omega_F)]$, over all "small" subsets Ω_F.

...and many other details!

The algorithm



Maximum partial solution for

- (S, C, Ω_G)
- a θ-structure $\theta(\Omega_F)$
- a (partial) neighborhood assignment

Dynamic programming over all p.m.c. Ω_G of G and on all possible (partial) neighborhood assignments $[\theta(\Omega_F)]$, over all "small" subsets Ω_F.

...and many other details!

Running time: $O(\sharp p.m.c. \cdot (\sharp \text{neighborhood assignments})^3) = O(1.73601^n)$

Conclusion and open questions

An $O(1.73601^n)$ algorithm for the MAX INDUCED PLANAR SUBGRAPH problem.

- MAX INDUCED SUBGRAPH WITH PROPERTY Π?
  - Bounded genus?
  - Excluded minors?
  - Bounded degeneracy?
- Combinatorial questions
  - What is the maximum number of minimal separators in an n-vertex graph?
  - The same for potential maximal cliques? (The current upper bound $O(1.73601^n)$ does not seem tight).

Conclusion and open questions

An $O(1.73601^n)$ algorithm for the MAX INDUCED PLANAR SUBGRAPH problem.

- MAX INDUCED SUBGRAPH WITH PROPERTY Π?
  - Bounded genus?
  - Excluded minors?
  - Bounded degeneracy?
- Combinatorial questions
  - What is the maximum number of minimal separators in an n-vertex graph?
  - The same for potential maximal cliques? (The current upper bound $O(1.73601^n)$ does not seem tight).
- Thank you! Your questions?