Lacon- and Shrub-Decompositions

A New Characterization of First-Order Transductions of Bounded Expansion Classes

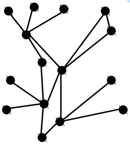
Jan Dreier
TU Wien, Austria

LICS 2021

Algorithmic Meta-Theorems

Algorithmic Meta-Theorems

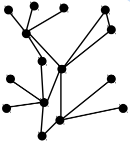
MSO on treewidth

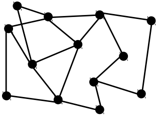


Algorithmic Meta-Theorems

MSO on treewidth

FO on sparse graphs

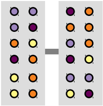


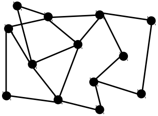


Algorithmic Meta-Theorems

MSO on cliquewidth

FO on sparse graphs

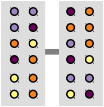


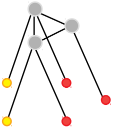


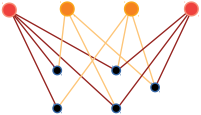
Algorithmic Meta-Theorems

MSO on cliquewidth

?







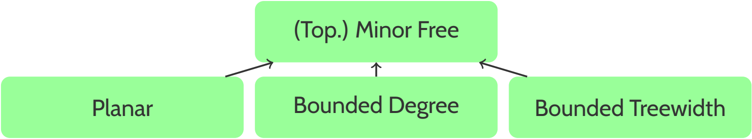
Some Sparse Graph Classes

Planar

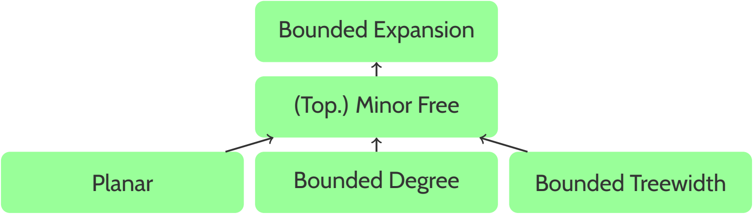
Bounded Degree

Bounded Treewidth

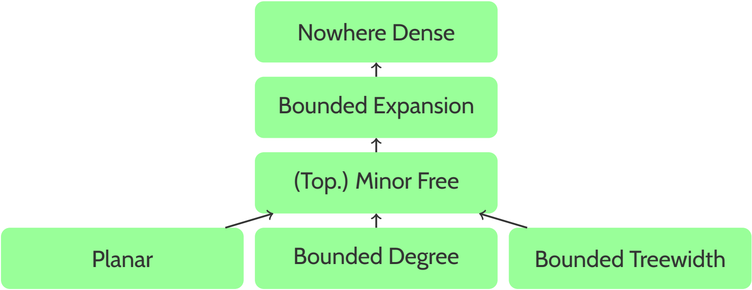
Some Sparse Graph Classes



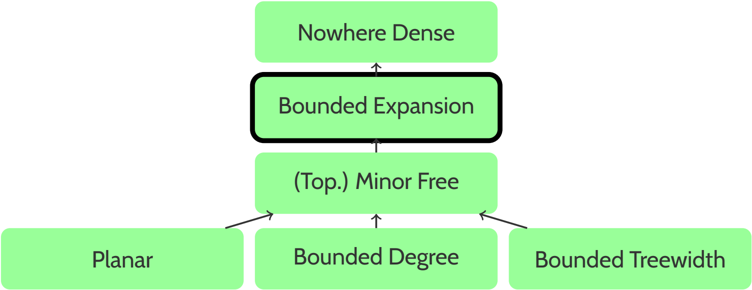
Some Sparse Graph Classes



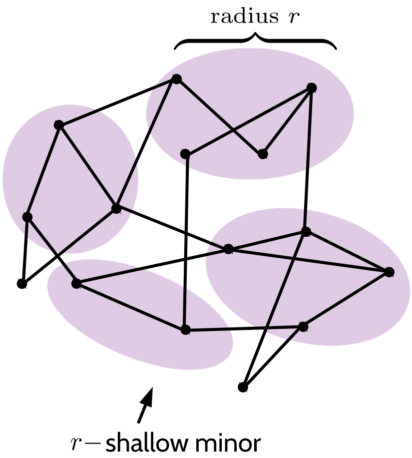
Some Sparse Graph Classes



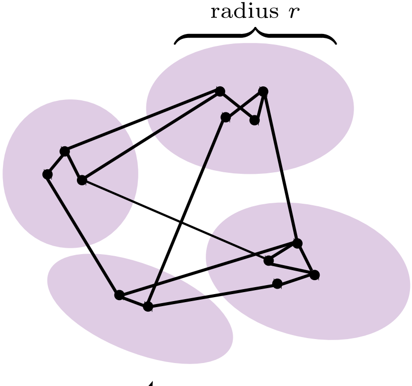
Some Sparse Graph Classes



Definition of Bounded Expansion

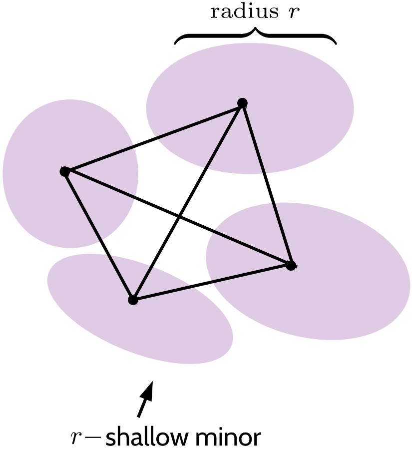


Definition of Bounded Expansion

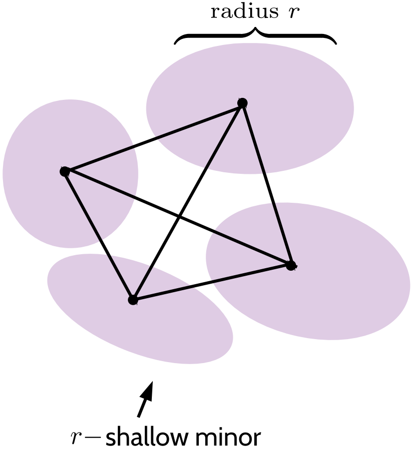


$r-$shallow minor

Definition of Bounded Expansion



Definition of Bounded Expansion



$$\frac{|E|}{|V|} \leq f(r)$$

for every r-shallow minor of every graph in the graph class.

FO Model-Checking [Dvořák, Král, Thomas 2010]

First-order formulas φ can be evaluated on bounded expansion classes in time $f(|\varphi|)n$.

Algorithmic Meta-Theorems

FO Model-Checking [Dvořák, Král, Thomas 2010]

First-order formulas φ can be evaluated on bounded expansion classes in time $f(|\varphi|)n$.

- dominating set of size k:

$$\exists x_1 \ldots \exists x_k \forall y \bigvee_i y \sim x_i \vee y = x_i$$

FO Model-Checking [Dvořák, Král, Thomas 2010]

First-order formulas φ can be evaluated on bounded expansion classes in time $f(|\varphi|)n$.

- dominating set of size k:

$$\exists x_1 \ldots \exists x_k \forall y \bigvee_i y \sim x_i \vee y = x_i$$

- independent set of size k:

$$\exists x_1 \ldots \exists x_k \bigwedge_{i,j} x_i \not\sim x_j \wedge x_i \neq x_j$$

Algorithmic Meta-Theorems

FO Model-Checking [Dvořák, Král, Thomas 2010]

First-order formulas φ can be evaluated on bounded expansion classes in time $f(|\varphi|)n$.

- dominating set of size k:

$$\exists x_1 \ldots \exists x_k \forall y \bigvee_i y \sim x_i \vee y = x_i$$

- independent set of size k:

$$\exists x_1 \ldots \exists x_k \bigwedge_{i,j} x_i \not\sim x_j \wedge x_i \neq x_j$$

Best algorithms on general graphs: $n^{O(k)}$

FO Model-Checking [Dvořák, Král, Thomas 2010]

First-order formulas φ can be evaluated on bounded expansion classes in time $f(|\varphi|)n$.

- dominating set of size k:

$$\exists x_1 \ldots \exists x_k \forall y \bigvee_i y \sim x_i \vee y = x_i$$

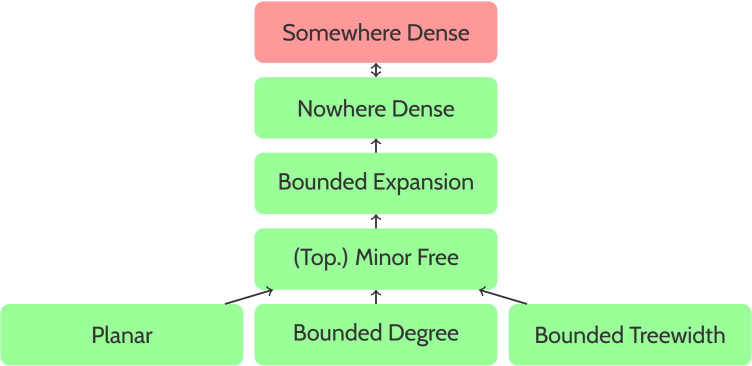
- independent set of size k:

$$\exists x_1 \ldots \exists x_k \bigwedge_{i,j} x_i \not\sim x_j \wedge x_i \neq x_j$$

Best algorithms on general graphs: $n^{O(k)}$

On bounded expansion: $f(k)n$

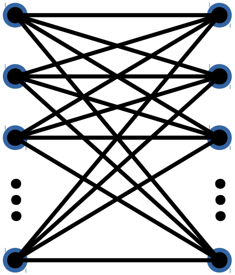
Exact Characterization



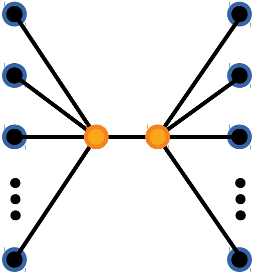
For graph classes $\mathcal{G}$ *closed under subgraphs*,
FO model-checking is tractable iff $\mathcal{G}$ is nowhere dense.

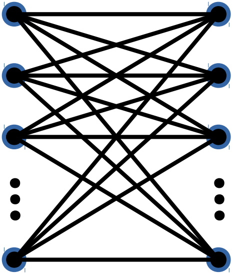
[Grohe, Kreutzer, Sieberz 2011]

Example: Fully Bipartite

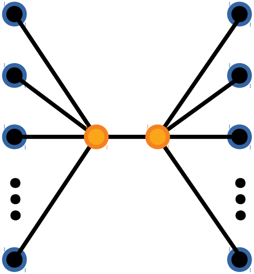


Example: Fully Bipartite

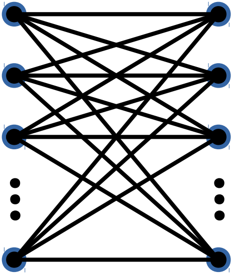




Example: Fully Bipartite

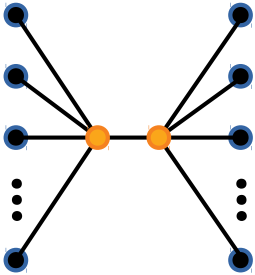


$$\mathrm{dist}(x, y) = 3$$



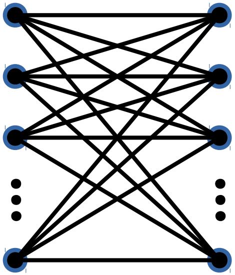
$$x \sim y$$

Example: Fully Bipartite



$\mathrm{dist}(x, y) = 3$

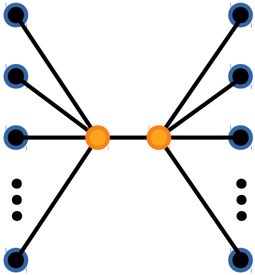
$\exists x \ \mathrm{blue}(x) \wedge \varphi$



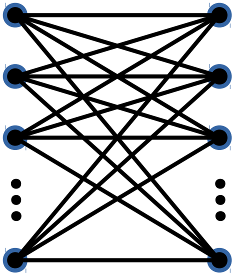
$x \sim y$

$\exists x \ \varphi$

Example: Fully Bipartite



$$\exists x_1 \exists x_2 \exists x_3$$
$$\text{blue}(x_1) \wedge \text{blue}(x_2) \wedge \text{blue}(x_3)$$
$$\text{dist}(x_1, x_2) = 3 \wedge \text{dist}(x_2, x_3) = 3$$
$$\wedge \text{dist}(x_1, x_3) = 3$$



$$\exists x_1 \exists x_2 \exists x_3$$
$$x_1 \sim x_2 \wedge x_2 \sim x_3$$
$$\wedge x_1 \sim x_3$$

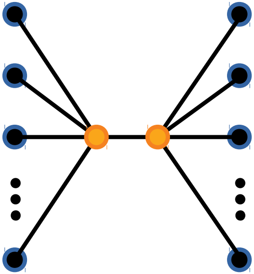
Interpretations

$$I = (\nu(x), \mu(x, y))$$

Interpretations

$$I = (\nu(x), \mu(x, y))$$

G



$I(G)$

vertices: $\{v \mid G \models \nu(v)\}$

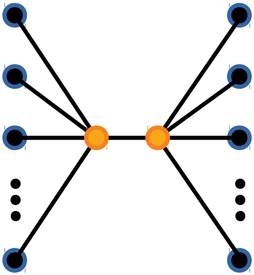
edges: $\{uv \mid G \models \mu(u, v)\}$

Interpretations

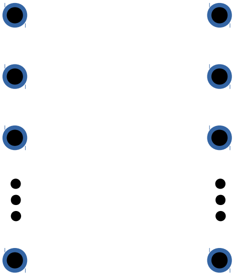
$$I = (\nu(x), \mu(x, y))$$

is blue

G



$I(G)$



vertices: $\{v \mid G \models \nu(v)\}$

edges: $\{uv \mid G \models \mu(u, v)\}$

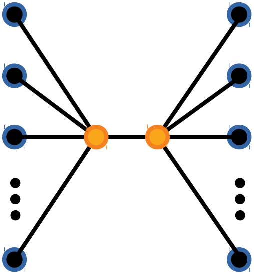
Interpretations

$$I = (\nu(x), \mu(x, y))$$

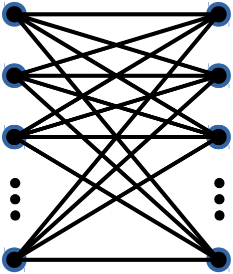
have distance three

is blue

G



$I(G)$



vertices: $\{v \mid G \models \nu(v)\}$

edges: $\{uv \mid G \models \mu(u, v)\}$

Structurally Property X

A graph class $\mathcal{G}$ has *structurally property X* if there exists

Structurally Property X

A graph class $\mathcal{G}$ has *structurally property X* if there exists

- a class $\mathcal{G}'$ with property X,

Structurally Property X

A graph class $\mathcal{G}$ has *structurally property X* if there exists

- a class $\mathcal{G}'$ with property X,
- an interpretation $I = (\nu(x), \mu(x, y))$,

Structurally Property X

A graph class $\mathcal{G}$ has *structurally property X* if there exists

- a class $\mathcal{G}'$ with property X,
- an interpretation $I = (\nu(x), \mu(x, y))$,

such that for every $G \in \mathcal{G}$ there is $G' \in \mathcal{G}'$ with $G = I(G')$.

Structurally Property X

A graph class $\mathcal{G}$ has *structurally property X* if there exists

- a class $\mathcal{G}'$ with property X,
- an interpretation $I = (\nu(x), \mu(x, y))$,

such that for every $G \in \mathcal{G}$ there is $G' \in \mathcal{G}'$ with $G = I(G')$.

The class of all fully bipartite graphs has *structurally treewidth 1*:

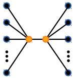
Structurally Property X

A graph class $\mathcal{G}$ has *structurally property X* if there exists

- a class $\mathcal{G}'$ with property X,
- an interpretation $I = (\nu(x), \mu(x, y))$,

such that for every $G \in \mathcal{G}$ there is $G' \in \mathcal{G}'$ with $G = I(G')$.

The class of all fully bipartite graphs has *structurally treewidth 1*:

- The class of all  has treewidth 1

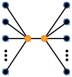
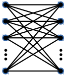
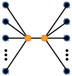
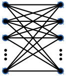
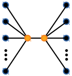
Structurally Property X

A graph class $\mathcal{G}$ has *structurally property X* if there exists

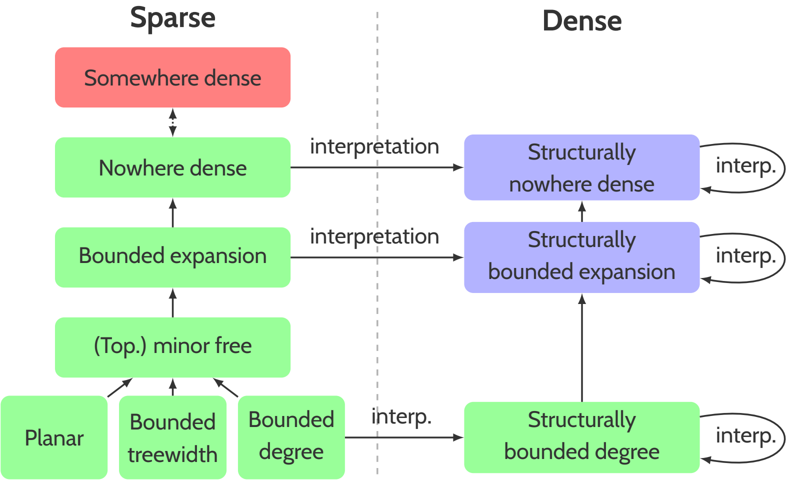
- a class $\mathcal{G}'$ with property X,
- an interpretation $I = (\nu(x), \mu(x, y))$,

such that for every $G \in \mathcal{G}$ there is $G' \in \mathcal{G}'$ with $G = I(G')$.

The class of all fully bipartite graphs has *structurally treewidth 1*:

- The class of all  has treewidth 1
- For every  there is  with  $= I\Big($  $\Big)$.

Model-Checking in Sparse and Dense Classes

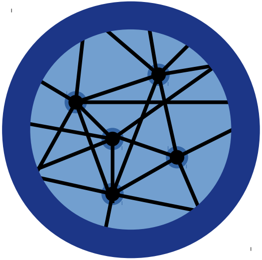


Nowhere Dense: Grohe, Kreutzer, Sieberz 2011
Structurally Bounded Degree: Gajarský, Hlineňỳ, Obdržálek, Lokshtanov, Ramanujan 2016

Big Question

Structurally
Bounded Expansion

$$I(G)$$



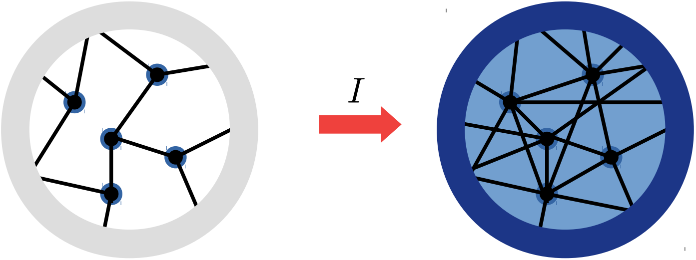
Big Question

Bounded Expansion

G

Structurally Bounded Expansion

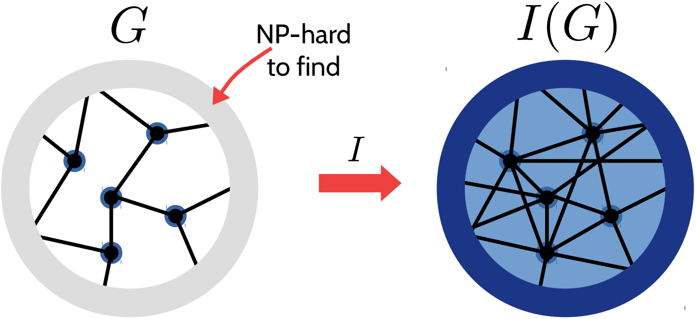
$I(G)$



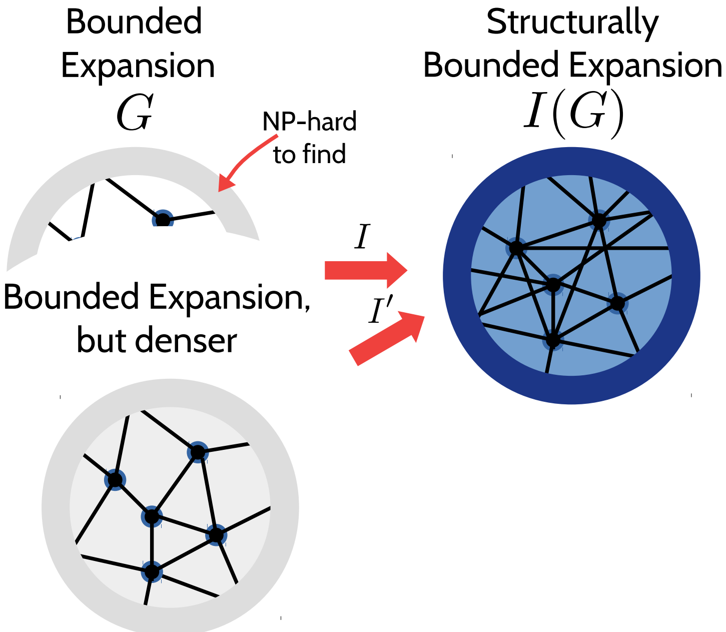
Big Question

Bounded Expansion

Structurally Bounded Expansion



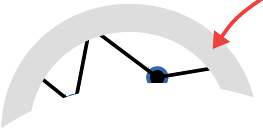
Big Question



Big Question

Bounded Expansion

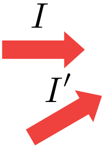
G

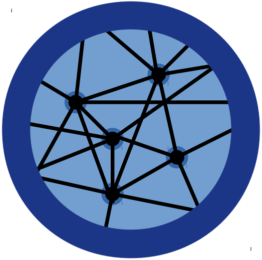


NP-hard to find

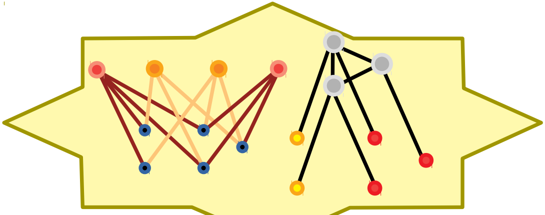
Structurally Bounded Expansion

$I(G)$

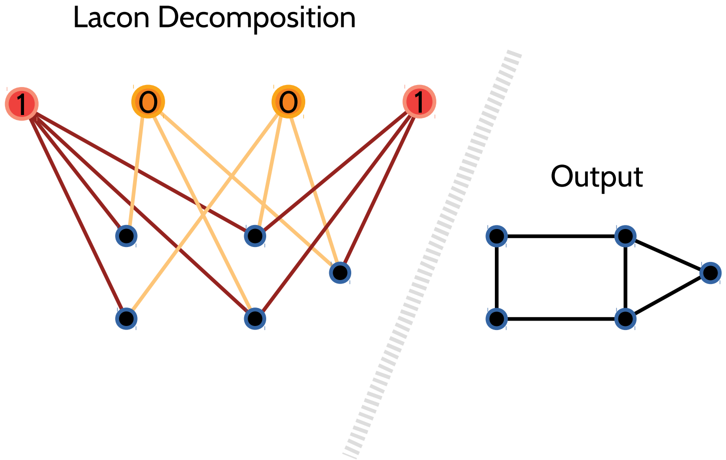




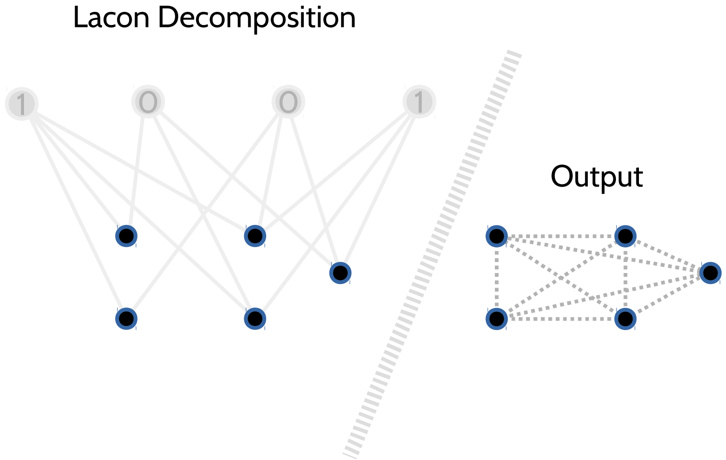
Lacon- or Shrub Decomposition



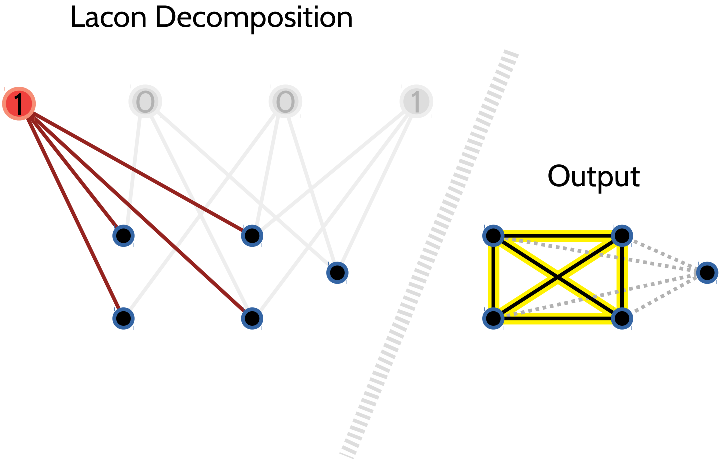
Lacon Decompositions



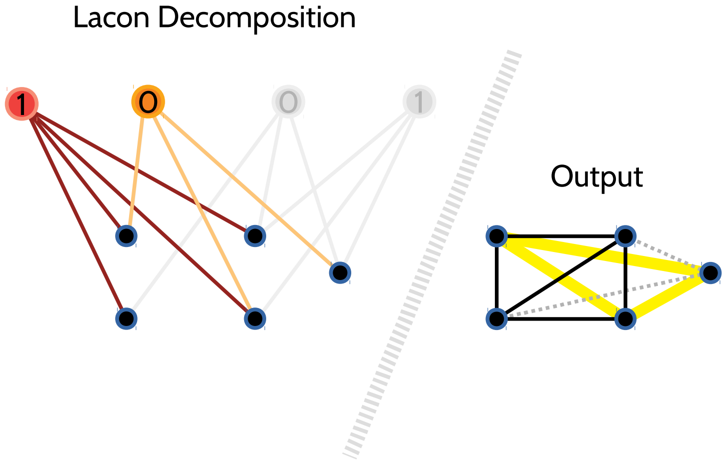
Lacon Decompositions



Lacon Decompositions

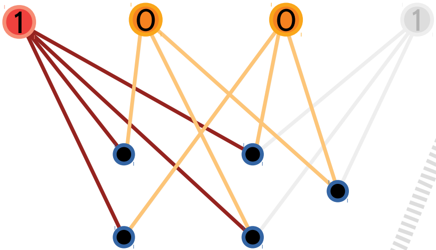


Lacon Decompositions

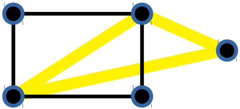


Lacon Decompositions

Lacon Decomposition

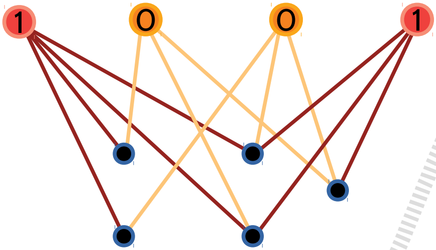


Output

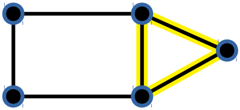


Lacon Decompositions

Lacon Decomposition

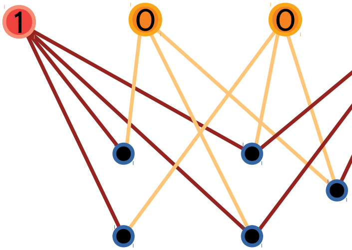


Output

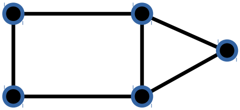


Lacon Decompositions

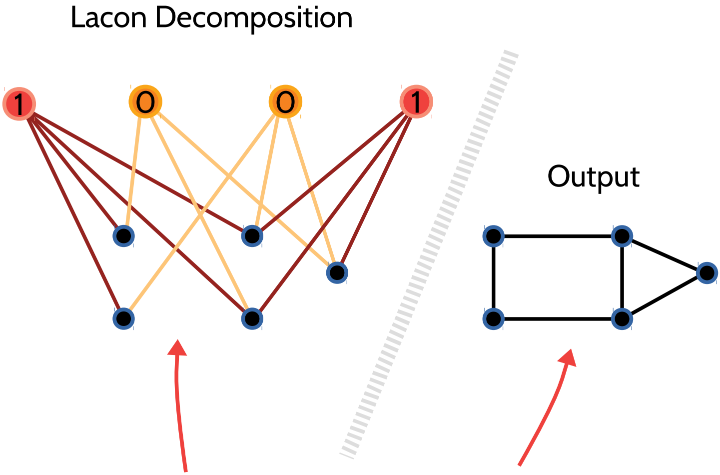
Lacon Decomposition



Output

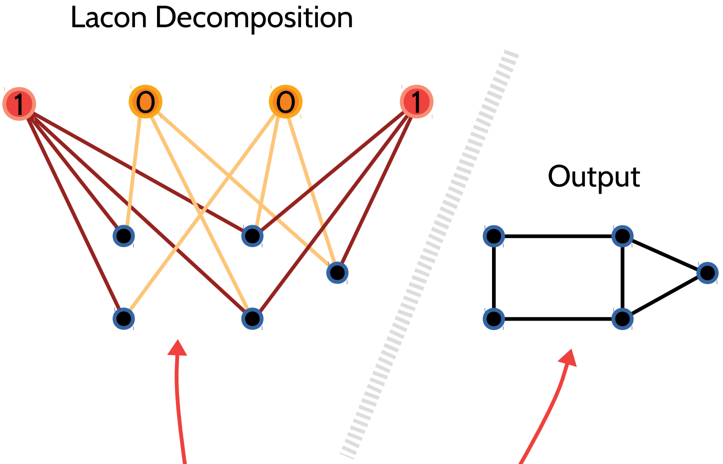


Lacon Decompositions



has lacon decomposition with property X $\implies$ structurally property X

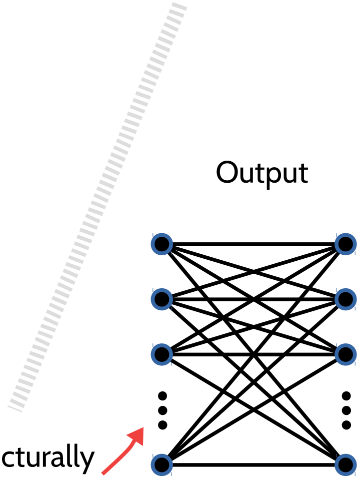
Lacon Decompositions



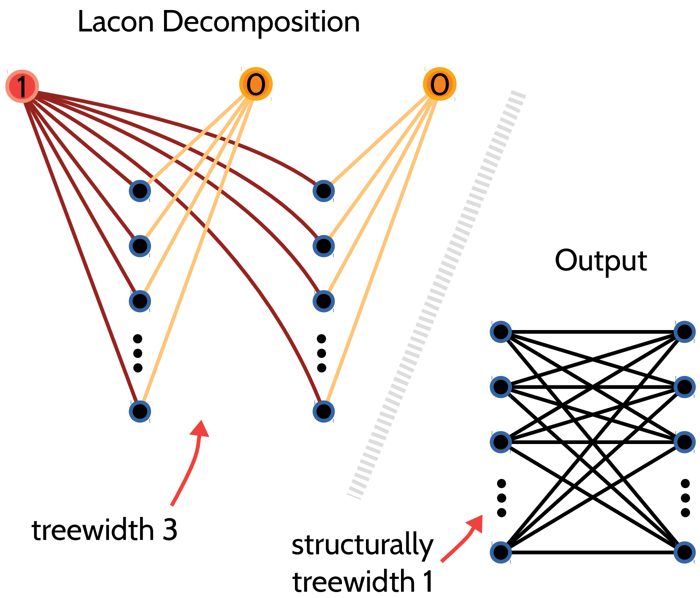
has lacon decomposition with property X $\iff$ structurally property X

Lacon Decompositions

Output

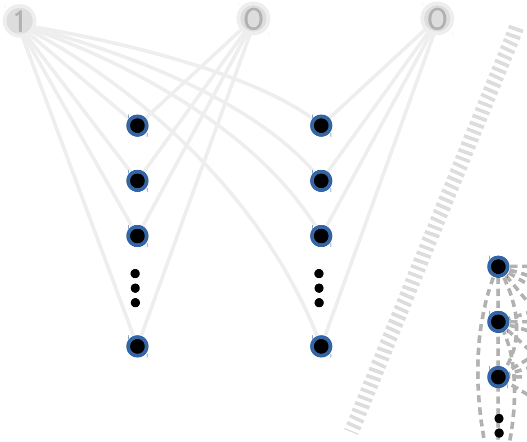


structurally
treewidth 1

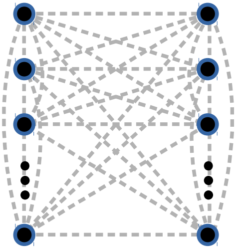

Lacon Decomposition
1
0
0
Output
treewidth 3
structurally
treewidth 1

Lacon Decompositions

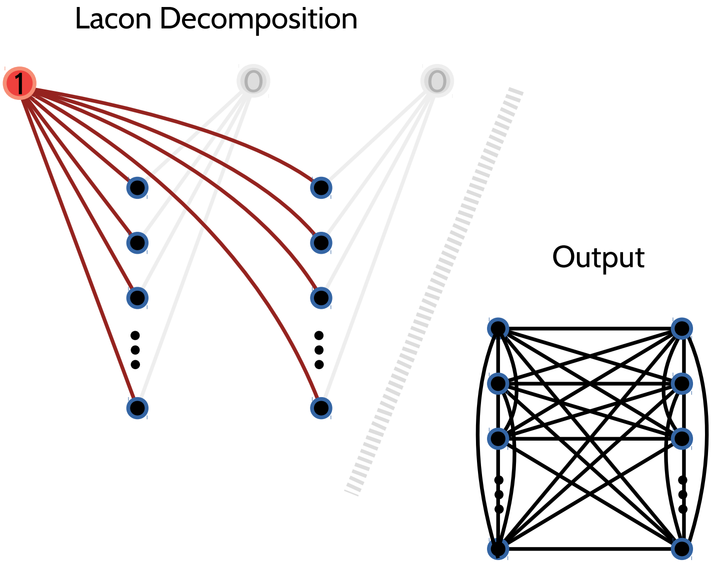
Lacon Decomposition



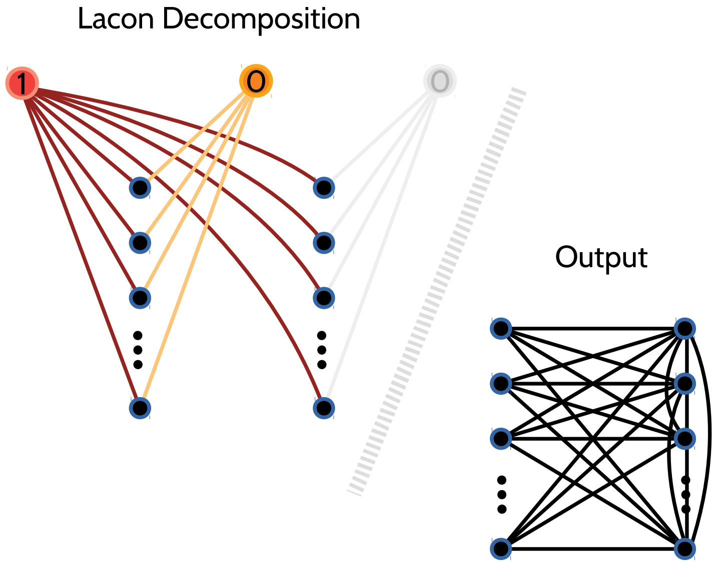
Output



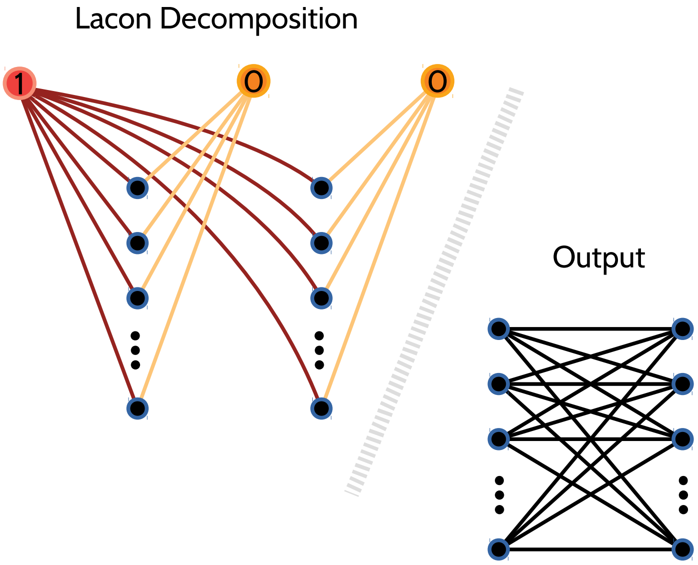
Lacon Decompositions



Lacon Decompositions



Lacon Decompositions

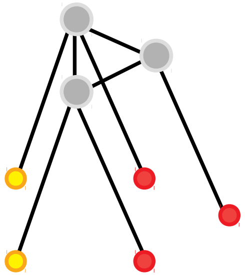


Shrub Decompositions

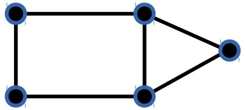
connect vertices with...

- distance 2
- distance 3 and same color

Shrub Decomposition



Output

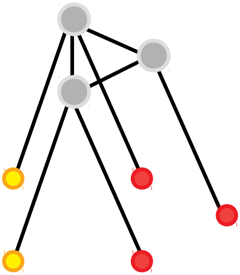


Shrub Decompositions

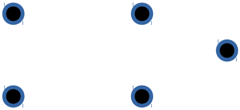
connect vertices with...

-
-

Shrub Decomposition



Output

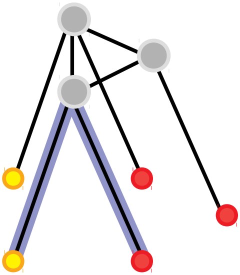


Shrub Decompositions

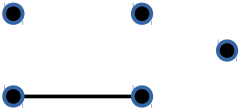
connect vertices with...

- distance 2
-

Shrub Decomposition

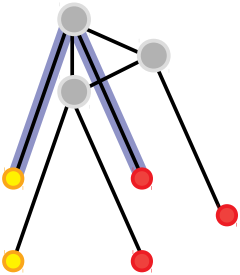


Output



Shrub Decompositions

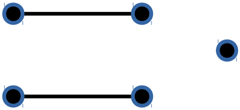
Shrub Decomposition



connect vertices with...

- distance 2
-

Output

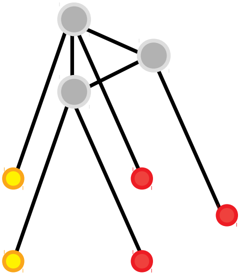


Shrub Decompositions

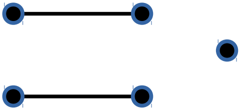
connect vertices with...

- distance 2
- distance 3 and same color

Shrub Decomposition



Output

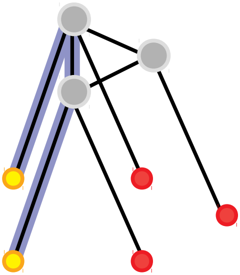


Shrub Decompositions

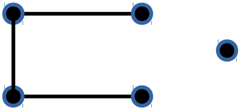
connect vertices with...

- distance 2
- distance 3 and same color

Shrub Decomposition



Output

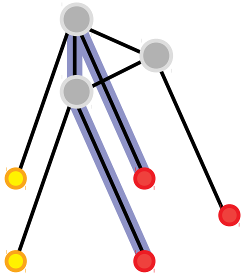


Shrub Decompositions

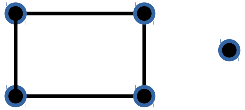
connect vertices with...

- distance 2
- distance 3 and same color

Shrub Decomposition

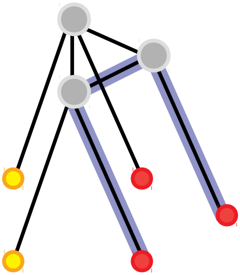


Output



Shrub Decompositions

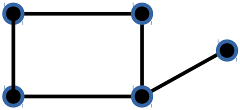
Shrub Decomposition



connect vertices with...

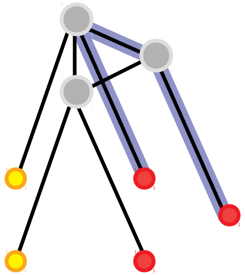
- distance 2
- distance 3 and same color

Output



Shrub Decompositions

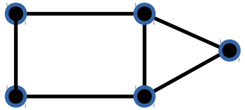
Shrub Decomposition



connect vertices with...

- distance 2
- distance 3 and same color

Output

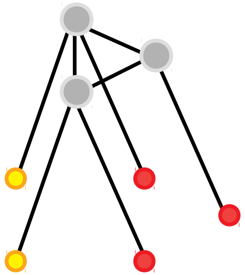


Shrub Decompositions

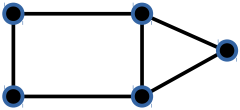
connect vertices with...

- distance 2
- distance 3 and same color

Shrub Decomposition



Output



has shrub decomposition with property X $\Longleftrightarrow$ structurally property X

Result

Theorem

Let $\mathcal{G}$ be a graph class. The following statements are equivalent.

Result

Theorem

Let $\mathcal{G}$ be a graph class. The following statements are equivalent.

- $\mathcal{G}$ has **structurally bounded expansion.**

Result

Theorem

Let $\mathcal{G}$ be a graph class. The following statements are equivalent.

- $\mathcal{G}$ has **structurally bounded expansion.**
- $\mathcal{G}$ has **lacon decompositions** with
  - **bounded expansion,**
  - **bounded target vertex degree.**

Result

Theorem

Let $\mathcal{G}$ be a graph class. The following statements are equivalent.

- $\mathcal{G}$ has **structurally bounded expansion.**
- $\mathcal{G}$ has **lacon decompositions** with
  - **bounded expansion,**
  - **bounded target vertex degree.**
- $\mathcal{G}$ has **shrub decompositions** with
  - **bounded expansion,**
  - **bounded number of colors,**
  - **bounded diameter.**

Result

Theorem

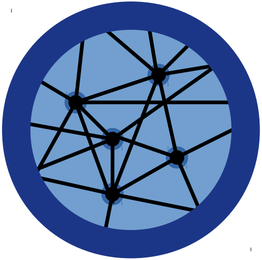
Let $\mathcal{G}$ be a graph class. The following statements are equivalent.

- $\mathcal{G}$ has **structurally bounded expansion.**
- $\mathcal{G}$ has **lacon decompositions** with
  - **bounded expansion,**
  - **bounded target vertex degree.**
- $\mathcal{G}$ has **shrub decompositions** with
  - **bounded expansion,**
  - **bounded number of colors,**
  - **bounded diameter.**
- $\mathcal{G}$ has **low shrubdepth covers** [1].

[1] Gajarský, Kreuzer, Nešetřil, Ossona de Mendez, Siebertz, Toruńczyk 2018

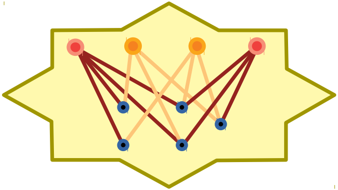
Questions

Structurally
Bounded Expansion

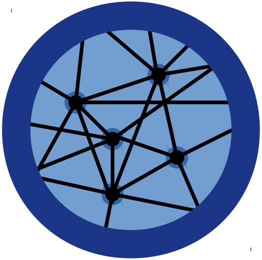


Questions

has Decomposition
with Bounded Expansion

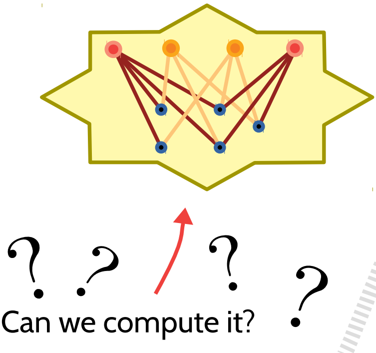


Structurally
Bounded Expansion



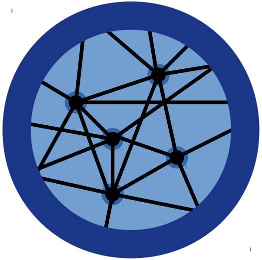
Questions

has Decomposition
with Bounded Expansion



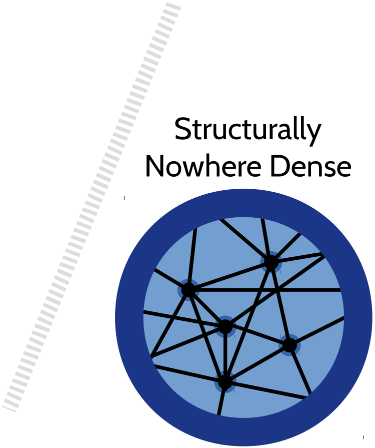
Can we compute it?

Structurally
Bounded Expansion



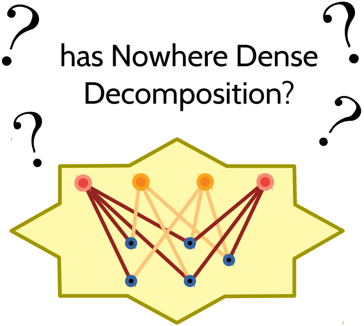
Questions

Structurally
Nowhere Dense

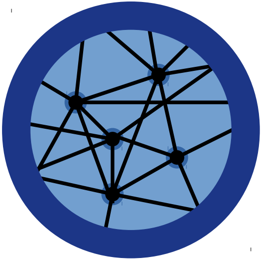


Questions

has Nowhere Dense Decomposition?



Structurally Nowhere Dense



End

Thanks!

dreier@ac.tuwien.ac.at