Kernels using composition

Abhisekh Sankaran

Parameterized Complexity Week, IMSc

October 26, 2017

Fixed parameter tractability

- Fixed parameter tractability (f.p.t.) is a subarea of algorithms introduced by Downey and Fellows in the mid-80s, to solve hard problems by identifying and fixing parameters.
- One of the approaches to getting f.p.t. algorithms is kernelization, in which a given problem instance is preprocessed to obtain a smaller structure, analysing which yields an answer to given problem instance.
- Algorithmic meta-theorems are a class of f.p.t. results that provide uniform algorithms for checking over a given graph class, any property of graphs that can be expressed in a logic.
- Starting with Courcelle's theorem (1990), algorithmic meta-theorems have been very well-studied and are being actively investigated in the context of sparse and dense graph classes.

Feferman-Vaught composition

- In the 1950s, Tarski asked the following of his students in Berkeley: Given the FO theories of structures $\mathcal{A}$ and $\mathcal{B}$, what can be said about the FO theories of $\mathcal{A} \sqcup \mathcal{B}$ and $\mathcal{A} \times \mathcal{B}$?
- Building upon the work of Mostowski, Feferman and Vaught showed that the FO theories of $\mathcal{A}$ and $\mathcal{B}$ completely determine FO theories of $\mathcal{A} \sqcup \mathcal{B}$ and $\mathcal{A} \times \mathcal{B}$.
- Subsequently, the above composition property was shown for various natural operations on structures and various logics.
- The first algorithmic application of the composition property was MSO characterization of regular languages by Büchi (1960). Subsequently, it has been used to prove the decidability of various theories, and algorithmic meta-theorems for various graph classes.

Talk outline

- Background from logic
- Feferman-Vaught composition
- Kernels for cographs
- Generalizations
- Implications for well-quasi-ordered classes
- Bootstrapping on kernels

Background from logic

FO and MSO

- First order logic (FO):

$$E(x,y) \mid x = y \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \neg\varphi \mid \exists x \varphi \mid \forall x \varphi$$

  E.g.: 2-vertex cover:

$$\exists x_1 \exists x_2 \forall y_1 \forall y_2 \quad \Big((x_1 \neq x_2) \wedge (E(y_1, y_2) \rightarrow \bigvee_{i,j \in \{1,2\}} y_i = x_j) \Big)$$

- Monadic second order logic (MSO):

$$E(x,y) \mid x = y \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \neg\varphi \mid \exists x \varphi \mid \forall x \varphi \mid \exists X \varphi \mid \forall X \varphi$$

  E.g.: 3-colourability:

$$\exists X_1 \exists X_2 \exists X_3 \quad \big(\forall x \bigvee_{1 \leq i < j \leq 3} \neg (X_i(x) \wedge X_j(x)) \quad \bigwedge$$
$$\forall x \forall y\, E(x,y) \rightarrow \bigwedge_{1 \leq i \leq 3} \neg (X_i(x) \wedge X_i(y)) \big)$$

Quantifier rank of a formula

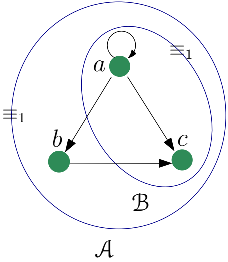
- The quantifier rank, or simply, rank of a formula is the maximum nesting depth of quantifiers appearing in it.
- E.g. the rank of the sentence for 2-vertex cover is 4, while the 3-colourability sentence has rank 5 (though number of quantifiers is 6).
- Let $\mathcal{L}$ denote FO or MSO, and let $\mathcal{L}[m]$ be the class of all $\mathcal{L}$ sentences of rank at most m.

Fact

The number of non-equivalent $\mathcal{L}[m]$ sentences is finite, and further, bounded by a computable function of m.

$\mathcal{L}[m]$-similarity of graphs

- We say graphs G and H are $\mathcal{L}[m]$-similar, denoted $G \equiv_{m,\mathcal{L}} H$, if no sentence of $\mathcal{L}[m]$ distinguishes G and H.



$\mathcal{A}$ and $\mathcal{B}$ are FO[1]-similar, but not FO[2]-similar.

$\mathcal{L}[m]$-similarity of graphs

- We say graphs G and H are $\mathcal{L}[m]$-similar, denoted $G \equiv_{m,\mathcal{L}} H$, if no sentence of $\mathcal{L}[m]$ distinguishes G and H.
- The finiteness of $\mathcal{L}[m]$ upto equivalence implies the following.

Fact

The set Δ_m of equivalence classes of the $\mathcal{L}[m]$-similarity relation is finite. Further, there is a computable function $\Lambda : \mathbb{N} \to \mathbb{N}$ such that $|\Delta_m| \leq \Lambda(m)$.

Feferman-Vaught composition

Feferman-Vaught composition property

Definition

Given a class $\mathcal{S}$ of graphs and an n-ary operation $\mathsf{Op} : \mathcal{S}^n \rightarrow \mathcal{S}$, we say Op satisfies the Feferman-Vaught $\mathcal{L}$-composition property, or simply, $\mathcal{L}$-composition, if the following holds:

$$
\begin{array}{ccccccc}
G_1 & G_2 & \cdots & G_n & \longrightarrow & \mathsf{Op}(G_1, G_2, \ldots, G_n) \\
\updownarrow \equiv_{m,\mathcal{L}} & \updownarrow \equiv_{m,\mathcal{L}} & & \updownarrow \equiv_{m,\mathcal{L}} & & \updownarrow \equiv_{m,\mathcal{L}} \\
H_1 & H_2 & \cdots & H_n & \longrightarrow & \mathsf{Op}(H_1, H_2, \ldots, H_n)
\end{array}
$$

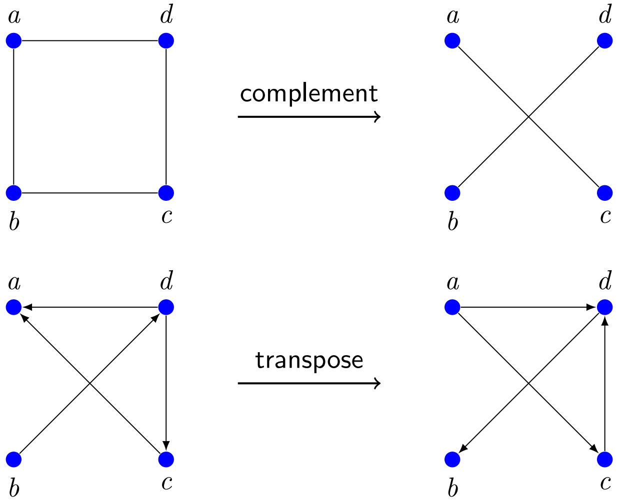
Feferman-Vaught composition property

Definition

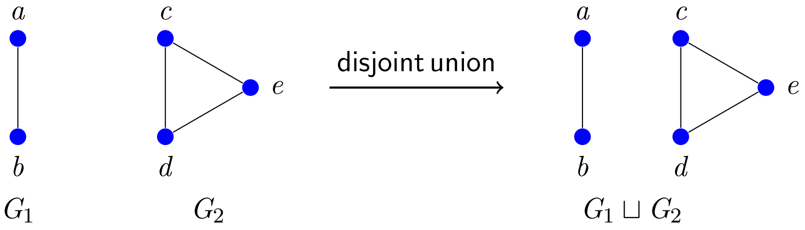
Given a class $\mathcal{S}$ of graphs and an n-ary operation $\mathsf{Op} : \mathcal{S}^n \to \mathcal{S}$, we say Op satisfies $\mathcal{L}$-composition, if there exists a composition function $f_{(\mathsf{Op},m)} : (\Delta_m)^n \to \Delta_m$ such that if $\delta_m(G)$ is the $\mathcal{L}[m]$-similarity class of G, then

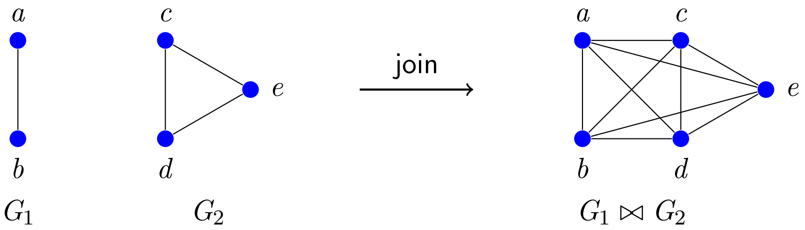
$$\delta_m(\mathsf{Op}(G_1, \ldots, G_n)) = f_{(\mathsf{Op},m)}(\delta_m(G_1), \ldots, \delta_m(G_n))$$

Operations satisfying MSO-composition

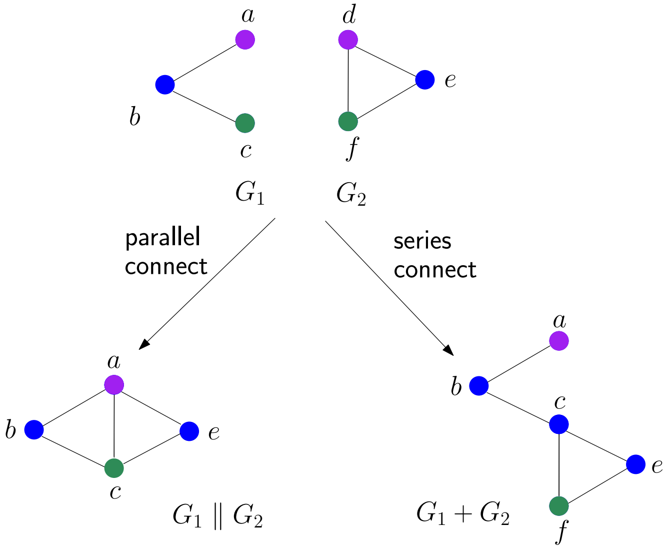


Operations satisfying MSO-composition

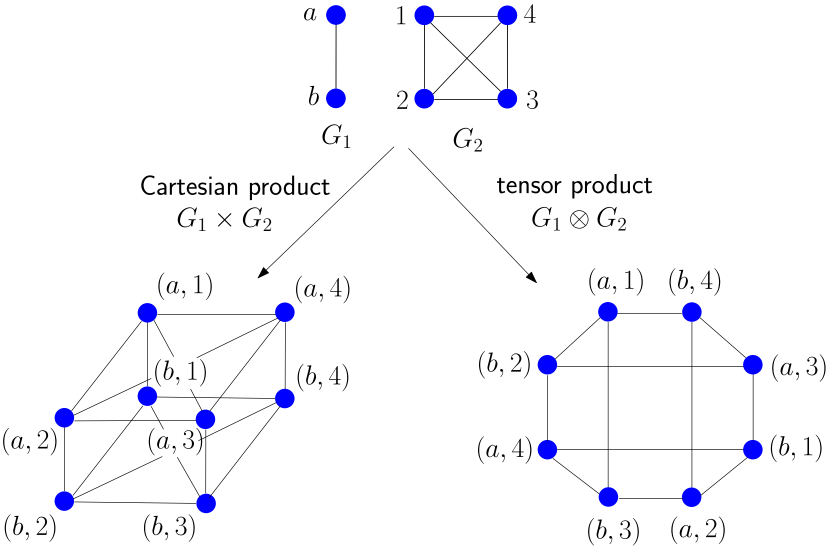




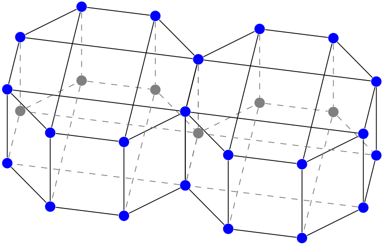
Operations satisfying MSO-composition

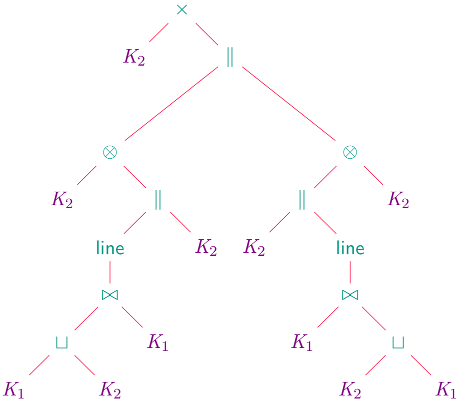


Operations satisfying FO-composition



Composite operations satisfying composition



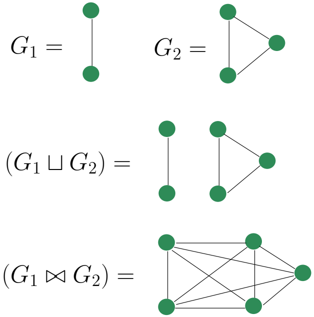


K_1 = single vertex; K_2 = single edge

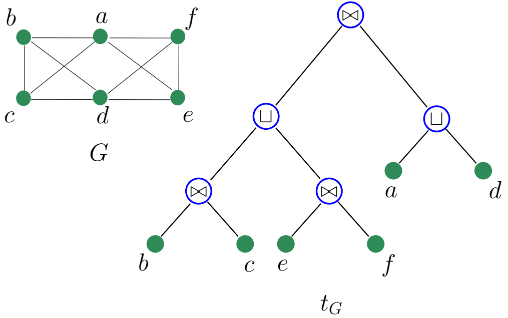
Kernels for cographs using $\mathcal{L}$-composition

Cographs

Generated from point graphs using disjoint union and join.



disjoint union $= \sqcup$; join $= \bowtie$



Cograph G and its cotree t_G

Revisiting model-theoretic facts

Recall the following facts.

Fact 1

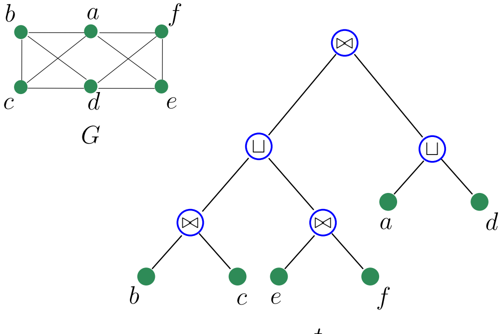
The set Δ_m of equivalence classes of the $\mathcal{L}[m]$-similarity relation is finite. Further, there is a computable function $\Lambda : \mathbb{N} \rightarrow \mathbb{N}$ such that $|\Delta_m| \leq \Lambda(m)$.

Fact 2

Each of $\sqcup$ and $\bowtie$ satisfies $\mathcal{L}$-composition. That is, there exist composition functions $f_m, g_m : (\Delta_m \times \Delta_m) \rightarrow \Delta_m$ for $\sqcup, \bowtie$ resp.

Kernelizing procedure for a fixed quantifier rank

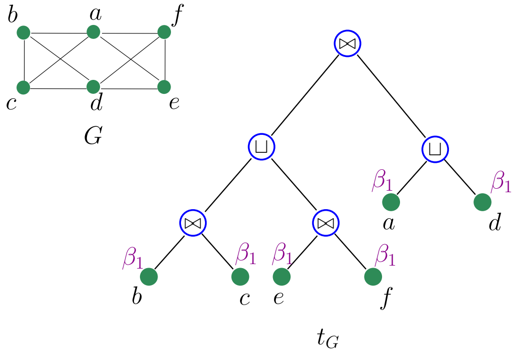
Step I: Label bottom up in the cotree, each node z with the m-similarity class of the graph represented by the tree rooted at z.



G t_G

Kernelizing procedure for a fixed quantifier rank

Step I: Label bottom up in the cotree, each node z with the m-similarity class of the graph represented by the tree rooted at z.



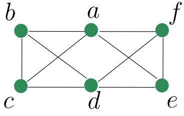
$\beta_1 = \delta_m$ (●)

Kernelizing procedure for a fixed quantifier rank

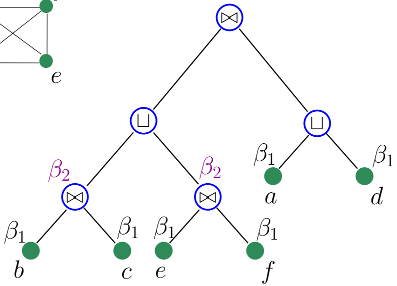
Step I: Label bottom up in the cotree, each node z with the m-similarity class of the graph represented by the tree rooted at z.

$\beta_1 = \delta_m(\bullet)$

$\beta_2 = g_m(\beta_1, \beta_1) = \delta_m(\bullet\!\!-\!\!\!-\!\!\bullet)$



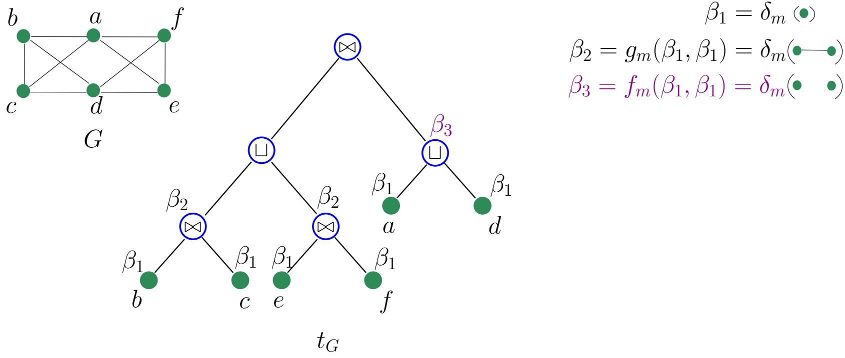
G



t_G

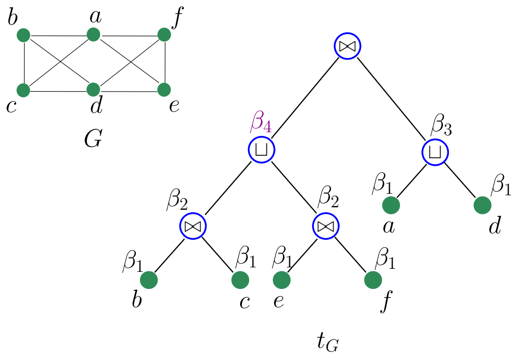
Kernelizing procedure for a fixed quantifier rank

Step I: Label bottom up in the cotree, each node z with the m-similarity class of the graph represented by the tree rooted at z.



Kernelizing procedure for a fixed quantifier rank

Step I: Label bottom up in the cotree, each node z with the m-similarity class of the graph represented by the tree rooted at z.



$\beta_1 = \delta_m(\bullet)$

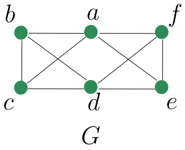
$\beta_2 = g_m(\beta_1, \beta_1) = \delta_m(\bullet\!-\!\bullet)$

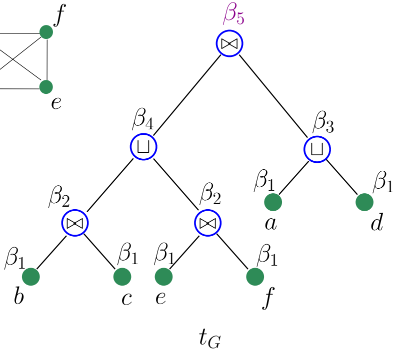
$\beta_3 = f_m(\beta_1, \beta_1) = \delta_m(\bullet \quad \bullet)$

$\beta_4 = f_m(\beta_2, \beta_2) = \delta_m\left(\begin{smallmatrix}\bullet & \bullet\\ | & |\\ \bullet & \bullet\end{smallmatrix}\right)$

Kernelizing procedure for a fixed quantifier rank

Step I: Label bottom up in the cotree, each node z with the m-similarity class of the graph represented by the tree rooted at z.





$\beta_1 = \delta_m(\bullet)$

$\beta_2 = g_m(\beta_1, \beta_1) = \delta_m(\bullet\!-\!\!-\!\bullet)$

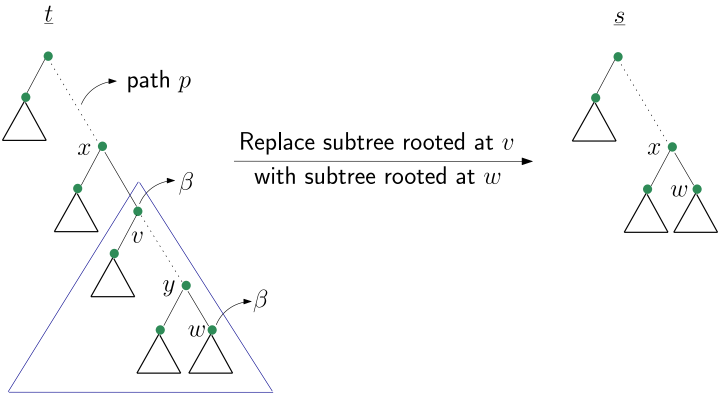
$\beta_3 = f_m(\beta_1, \beta_1) = \delta_m(\bullet \quad \bullet)$

$\beta_4 = f_m(\beta_2, \beta_2) = \delta_m\left(\begin{smallmatrix}\bullet & \bullet \\ | & | \\ \bullet & \bullet\end{smallmatrix}\right)$

$\beta_5 = g_m(\beta_3, \beta_4) = \delta_m(G)$

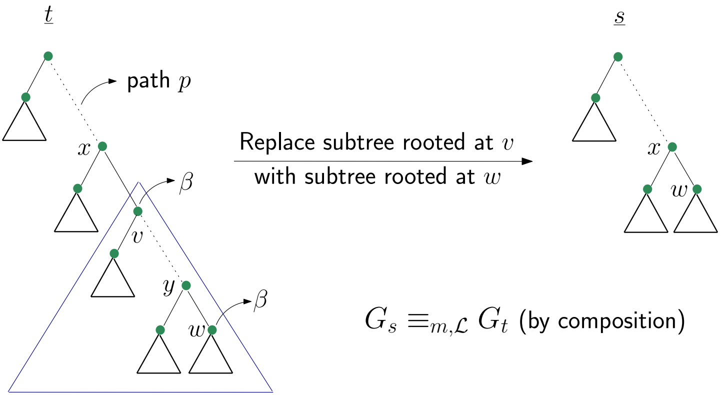
Kernelization procedure for a fixed quantifier rank

Step II: Perform graftings in the cotree whenever a root-to-leaf path has repeated labels.



Kernelization procedure for a fixed quantifier rank

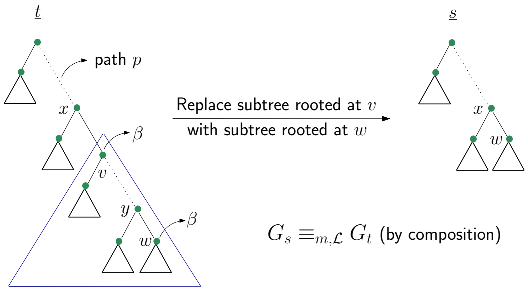
Step II: Perform graftings in the cotree whenever a root-to-leaf path has repeated labels.



$G_s \equiv_{m,\mathcal{L}} G_t$ (by composition)

Kernelization procedure for a fixed quantifier rank

Step II: Perform graftings in the cotree whenever a root-to-leaf path has repeated labels.

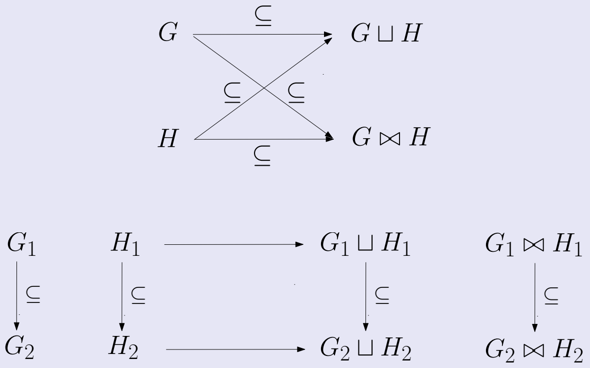


Iterate to get a "rainbow" subtree in which no root-to-leaf path has repeated labels. This subtree represents an $\mathcal{L}[m]$-similar cograph of size $\leq 2^{\Lambda(m)}$. $\square$

An additional fact about $\sqcup$ and $\bowtie$

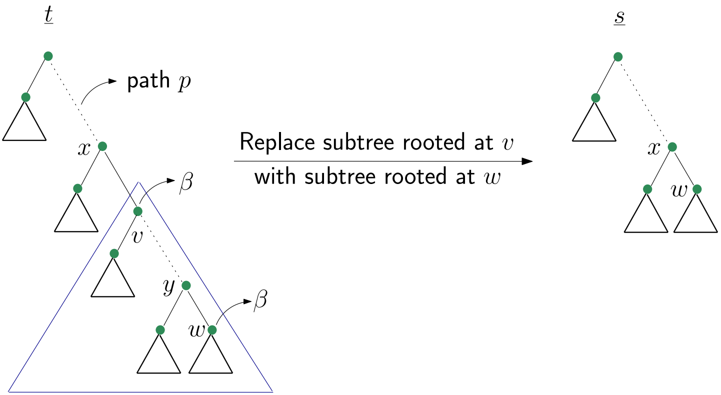
Fact

Each of $\sqcup$ and $\bowtie$ has the following "$\subseteq$-monotonicity" properties.



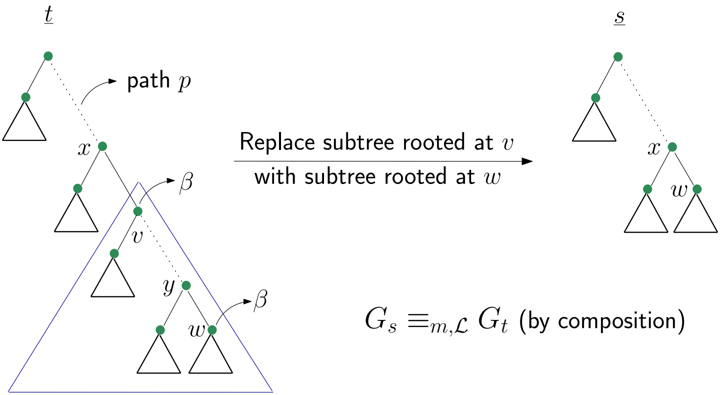
Kernelization procedure revisited

Step II: Perform graftings in the cotree whenever a root-to-leaf path has repeated labels.



Kernelization procedure revisited

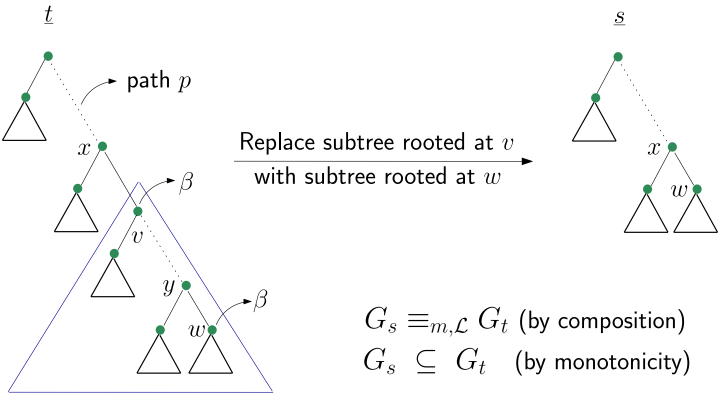
Step II: Perform graftings in the cotree whenever a root-to-leaf path has repeated labels.



$G_s \equiv_{m,\mathcal{L}} G_t$ (by composition)

Kernelization procedure revisited

Step II: Perform graftings in the cotree whenever a root-to-leaf path has repeated labels.

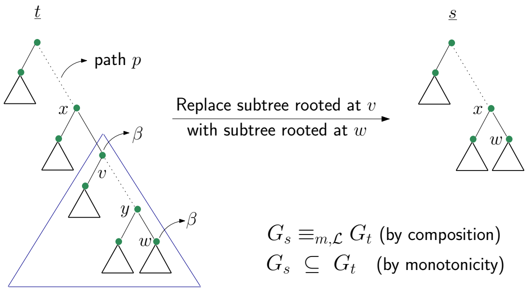


$G_s \equiv_{m,\mathcal{L}} G_t$ (by composition)

$G_s \subseteq G_t$ (by monotonicity)

Kernelization procedure revisited

Step II: Perform graftings in the cotree whenever a root-to-leaf path has repeated labels.



$G_s \equiv_{m,\mathcal{L}} G_t$ (by composition)

$G_s \subseteq G_t$ (by monotonicity)

Iterate to get a "rainbow" subtree in which no root-to-leaf path has repeated labels. This subtree represents an $\mathcal{L}[m]$-similar induced sub-cograph of size $\leq 2^{\Lambda(m)}$. □

Computing the composition functions

We do this in two stages:

1. Compute the list $\mathcal{L}$[m]-classes of $\mathcal{L}[m]$ sentences representing the equivalence classes of the $\equiv_{m,\mathcal{L}}$ relation over cographs.
2. For each $\delta_1, \delta_2 \in$ $\mathcal{L}$[m]-classes, determine $f_m(\delta_1, \delta_2)$ and $g_m(\delta_1, \delta_2)$.

We crucially use the following lemma that follows from the kernelization procedure just described.

Lemma

$\mathcal{L}$ has the small model property over cographs. Whereby

- $\mathcal{L}$-SAT is decidable over cographs.
- There is an algorithm that produces a model for every satisfiable $\mathcal{L}$ sentence.

Stage I: Computing $\mathcal{L}[m]$-classes

- Use the inductive definition of $\mathcal{L}$ to enumerate a set $\mathcal{X}$ of $\mathcal{L}[m]$ sentences, which represents all $\mathcal{L}[m]$ sentences upto equivalence.
- Construct the set $\mathcal{Z}$ of "$\mathcal{L}[m]$-complete" sentences such that all models of any such sentence are $\mathcal{L}[m]$-similar.
- For every sentence of $\mathcal{Z}$, decide if it is satisfiable over cographs. If it is, then the sentence represents an equivalence class of the $\equiv_{m,\mathcal{L}}$ relation over cographs.
- The set of satisfiable sentences above is indeed the list $\mathcal{L}$[m]-classes.

Stage II: Computing f_m and g_m

- Consider $\delta_1, \delta_2 \in \mathcal{L}[\mathsf{m}]$-classes.
- Since each is satisfiable over cographs, generate models $\mathcal{A}_1, \mathcal{A}_2$ for these resp.
- Construct cotrees $\mathsf{t}_1, \mathsf{t}_2$ for $\mathcal{A}_1, \mathcal{A}_2$ resp. (This is actually doable in linear time.)
- Construct the tree s_f, resp. s_g, by making t_1 and t_2 the child subtrees of a new root node labeled with $\sqcup$, resp. $\bowtie$.
- Determine $\delta_f, \delta_g \in \mathcal{L}[\mathsf{m}]$-classes such that the cograph represented by s_f, resp. s_g, models δ_f, resp. δ_g.
- Then $f_m(\delta_1, \delta_2) = \delta_f$ and $g_m(\delta_1, \delta_2) = \delta_g$ (by the $\mathcal{L}$-composition property of $\sqcup$ and $\bowtie$).

Kernelization for cographs

Let $\Lambda(m) = |\mathcal{L}[\text{m}]\text{-classes}|$ be the index of $\equiv_{m,\mathcal{L}}$ over cographs.

Theorem

There is an f.p.t. algorithm $\mathcal{A}$ and a computable function $\rho : \mathbb{N} \to \mathbb{N}$ such that given a cograph G and $m \in \mathbb{N}$, algorithm $\mathcal{A}$ outputs an $\mathcal{L}[m]$-similar cograph H of size $\leq 2^{\Lambda(m)}$ in time $\rho(m) \times |G|$. The graph H is thus a "uniform kernel" for all $\mathcal{L}[m]$ properties of G. Further, H can be constructed to be an induced subgraph of G.

Corollary (Algorithmic meta-theorem for cographs)

There is a linear time f.p.t. algorithm for $\mathcal{L}$ model checking over cographs, where the size of the $\mathcal{L}$ sentence is the parameter.

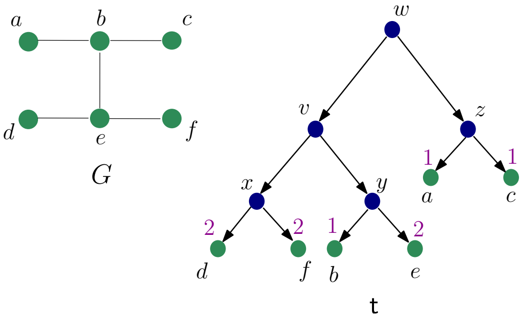
Generalizing the cograph results

Abstracting from the cograph results

- The methods for cographs work seamlessly for any graph class that admits operators satisfying composition that enable constructing the graphs of the class from simple graphs.
- Whereby, we get kernelization and algorithmic meta-theorems for model checking, if the tree representations of the graphs can be computed in polynomial time.
- The kernel sizes are exponential in the index of the $\equiv_{m,\mathcal{L}}$ relation over the class (The price to pay for uniform kernels!).
- If the operators further satisfy $\preceq$-monotonicity with respect to a graph relation $\preceq$ (such as: induced subgraph, subgraph, homomorphic image, minor, etc.), then the kernel obtained is also $\preceq$-related to the given graph.

m-partite cographs

- Hliněný, Nešetřil, et al. introduced in 2012, a special class of bounded clique-width graphs, called m-partite cographs.
- An m-partite cograph G is a graph that has an m-partite cotree representation t:



Label set $= \{1, 2\}$
$f_x = f_z = 0$
$f_y = 1$
$f_v(2, 2) = 1$, else 0
$f_w(1, 1) = 1$, else 0

Important subclasses of m-partite cographs

- Cographs (1-partite cographs): complete graphs, complete k-partite graphs, threshold graphs, Turan graphs, etc.
- Bounded tree-depth graphs
- Bounded shrub-depth graphs
- Bounded $\mathcal{SC}$-depth graphs

All of the above classes are of active current interest for their excellent algorithmic and logical properties!

Composition operators for m-partite cographs

- For each m-partite cograph, fix an m-partite cotree, and let $\mathcal{C}$ be the class of labeled versions of the m-partite cographs given by their chosen cotrees.
- For $f : [1, m]^2 \rightarrow \{0, 1\}$, define $\otimes_f : \mathcal{C}^2 \rightarrow \mathcal{C}$ such that if t, s are m-partite cotrees for inputs G, H resp., then $\otimes_f(G, H)$ is the labeled m-partite cograph given by the tree obtained by making t and s, child subtrees of a new root node labeled f.
- By an Ehrenfeucht-Fräissé game argument, $\otimes_f$ can be shown to satisfy $\mathcal{L}$-composition.
- Further $\otimes_f$ satisfies $\subseteq$-monotonicity, where $\subseteq =$ induced subgraph.

Kernelization for m-partite cographs and its subclasses

Let $\Lambda(n)$ = index of $\equiv_{n,\mathcal{L}}$ over $\mathcal{C}$ (where $\mathcal{C}$ is as defined earlier).

Theorem

Let $\mathcal{S}$ be a hereditary subclass of m-partite cographs. Given $G \in \mathcal{S}$ and $n \in \mathbb{N}$, there is an $\mathcal{L}[n]$-uniform kernel H in $\mathcal{S}$, of size $\leq 2^{\Lambda(n)}$ that is computable in f.p.t linear time, provided G is given by its m-partite cotree in $\mathcal{C}$. Further, H is an induced subgraph of G.

Corollary

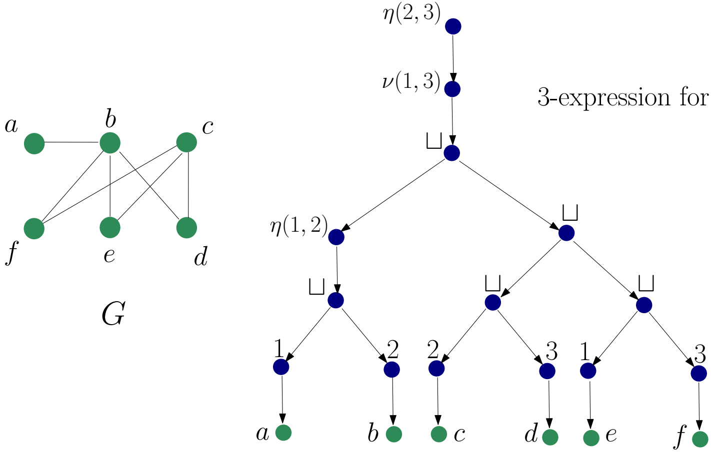
The same statement as above holds of the following classes. Further, for all these classes, the kernel sizes are elementary in n.

1. Any hereditary class of graphs of bounded shrub-depth.
2. Any hereditary class of graphs of bounded $\mathcal{SC}$-depth.
3. Any hereditary class of graphs of bounded tree-depth.

Graphs of bounded clique-width

- The notion of clique-width was introduced by Courcelle, Engelfriet and Rozenberg in 1993.
- Defined in terms of the following 4 operations on labeled graphs whose labels belong to $\{1, \ldots, k\}$:
  1. Create vertex with label i : "$i(v)$"
  2. Disjoint union of labeled graphs G and H: "$G \sqcup H$"
  3. Join i-labeled vertices to j-labeled vertices, $i \neq j$: "$\eta(i, j)$"
  4. Relabel label i to label j: "$\nu(i, j)$"
- A k-expression is a tree formed from the above operations.
- The clique-width of a graph G is the minimum k for which there exists a k-expression which when applied to point graphs, yields G.

Clique width: example



3-expression for G

Clique-width and other graphs

- Cographs are exactly clique-width 2 graphs.
- m-partite cographs have clique-width $\leq 2m$.
- Any graph of tree-width k has clique-width $\leq 3 \cdot 2^{k-1}$. Thus graphs of bounded tree-width have bounded clique-width too.
- The NLC-width of a graph is related to its clique-width as: $\text{NLC-wd}(G) \leq \text{cwd}(G) \leq 2 \cdot \text{NLC-wd}(G)$. Whereby, bounded NLC-width $=$ bounded clique-width.
- The rank-width of a graph and its clique-width are related as: $\text{rwd}(G) \leq \text{cwd}(G) \leq 2^{1+\text{rwd}(G)} - 1$. Whereby, bounded rank-width $=$ bounded clique-width.

Kernelization for bounded clique-width graphs

- Each of the operations used in defining clique-width satisfies $\mathcal{L}$-composition. This again follows by an Ehrenfeucht - Fräïssé game argument.
- There is a polynomial time algorithm (by Oum and Seymour) that, given as input a graph of clique-width k, outputs a $(2^{3k+2} - 1)$-expression for the graph.

Proposition

Let $\mathcal{S}$ be the class of graphs of clique-width $\leq k$ and $\Lambda(n)$ be the index of the $\equiv_{n,\mathcal{L}}$ relation over labeled graphs having labels in $\{1, \ldots, 2^{3k+2} - 1\}$. Then given $G \in \mathcal{S}$ and $n \in \mathbb{N}$, there is an $\mathcal{L}[n]$-uniform kernel H in $\mathcal{S}$, of size $\leq 2^{\Lambda(n)}$ that is computable in f.p.t linear time.

Implication for well-quasi-ordered classes

Well-quasi-ordering

Definition

Given a class $\mathcal{S}$ of graphs and a binary relation $\preceq$ on $\mathcal{S}$, we say $\mathcal{S}$ is w.q.o. under $\preceq$ if for every infinite set $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\}$ of graphs of $\mathcal{S}$, there exist i, j such that $\mathcal{A}_i \preceq \mathcal{A}_j$.

- Words are w.q.o. under subword (Higman, 1952).
- Trees are w.q.o. under subtree (Kruskal, 1960).
- The class of graphs that exclude P_k as a subgraph is w.q.o. under subgraph (Ding, 1992)
- All finite graphs are w.q.o. under minor (Robertson and Seymour, 2004).
- m-partite cographs are w.q.o. under induced subgraph (Hliněný, Nešetřil, et al., 2012).

$\mathcal{L}$-composition bounds sizes of forbidden graphs

Proposition

Let $\mathcal{S}$ be a graph class that admits tree representations using operators satisfying $\mathcal{L}$-composition and $\preceq$-monotonicity.

$\mathcal{L}$-composition bounds sizes of forbidden graphs

Proposition

Let $\mathcal{S}$ be a graph class that admits tree representations using operators satisfying $\mathcal{L}$-composition and $\preceq$-monotonicity. Let $\Lambda(n)$ be the index of the $\equiv_{n,\mathcal{L}}$ relation over $\mathcal{S}$.

$\mathcal{L}$-composition bounds sizes of forbidden graphs

Proposition

Let $\mathcal{S}$ be a graph class that admits tree representations using operators satisfying $\mathcal{L}$-composition and $\preceq$-monotonicity. Let $\Lambda(n)$ be the index of the $\equiv_{n,\mathcal{L}}$ relation over $\mathcal{S}$. If $\mathcal{S}$ is w.q.o. under $\preceq$,

$\mathcal{L}$-composition bounds sizes of forbidden graphs

Proposition

Let $\mathcal{S}$ be a graph class that admits tree representations using operators satisfying $\mathcal{L}$-composition and $\preceq$-monotonicity. Let $\Lambda(n)$ be the index of the $\equiv_{n,\mathcal{L}}$ relation over $\mathcal{S}$. If $\mathcal{S}$ is w.q.o. under $\preceq$, then every $\preceq$-hereditary subclass of $\mathcal{S}$, that is $\mathcal{L}$ definable,

$\mathcal{L}$-composition bounds sizes of forbidden graphs

Proposition

Let $\mathcal{S}$ be a graph class that admits tree representations using operators satisfying $\mathcal{L}$-composition and $\preceq$-monotonicity. Let $\Lambda(n)$ be the index of the $\equiv_{n,\mathcal{L}}$ relation over $\mathcal{S}$. If $\mathcal{S}$ is w.q.o. under $\preceq$, then every $\preceq$-hereditary subclass of $\mathcal{S}$, that is $\mathcal{L}$ definable, is determined by a finite list of forbidden graphs of $\mathcal{S}$, each of size $\leq 2^{\Lambda(n)}$.

$\mathcal{L}$-composition bounds sizes of forbidden graphs

Proposition

Let $\mathcal{S}$ be a graph class that admits tree representations using operators satisfying $\mathcal{L}$-composition and $\preceq$-monotonicity. Let $\Lambda(n)$ be the index of the $\equiv_{n,\mathcal{L}}$ relation over $\mathcal{S}$. If $\mathcal{S}$ is w.q.o. under $\preceq$, then every $\preceq$-hereditary subclass of $\mathcal{S}$, that is $\mathcal{L}$ definable, is determined by a finite list of forbidden graphs of $\mathcal{S}$, each of size $\leq 2^{\Lambda(n)}$. Whereby, the size of the finite list is also bounded by $2^{2^{\Lambda(n)}}$.

The above result therefore holds of the various classes seen earlier.

Bootstrapping on kernels

A closer look at what we have so far

- So far we considered graph classes that, for a given set of operations satisfying composition and monotonicity, are generated using all trees labeled with the operations, and applied to point graphs.
- The class of all trees over a given alphabet forms a trivial regular language.
- And point graphs form a trivial class of graphs admitting kernelization.
- How about generalizing these two scenarios?

Regular languages of operation trees

- Let $\mathcal{S}$ be a graph class that admits $\mathcal{L}[n]$-uniform $\preceq$-related kernels in $\mathcal{S}$, of size $\leq \rho(n)$ for a binary relation $\preceq$ on $\mathcal{S}$ and a (computable) function $\rho : \mathbb{N} \to \mathbb{N}$.

Regular languages of operation trees

- Let $\mathcal{S}$ be a graph class that admits $\mathcal{L}[n]$-uniform $\preceq$-related kernels in $\mathcal{S}$, of size $\leq \rho(n)$ for a binary relation $\preceq$ on $\mathcal{S}$ and a (computable) function $\rho : \mathbb{N} \to \mathbb{N}$.
- Let $\mathcal{O}$ be a set of operations on $\mathcal{S}$, satisfying $\mathcal{L}$-composition and $\preceq$-monotonicity.

Regular languages of operation trees

- Let $\mathcal{S}$ be a graph class that admits $\mathcal{L}[n]$-uniform $\preceq$-related kernels in $\mathcal{S}$, of size $\leq \rho(n)$ for a binary relation $\preceq$ on $\mathcal{S}$ and a (computable) function $\rho : \mathbb{N} \to \mathbb{N}$.
- Let $\mathcal{O}$ be a set of operations on $\mathcal{S}$, satisfying $\mathcal{L}$-composition and $\preceq$-monotonicity.
- Let d be the maximum arity and t be the maximum dimension of any operation of O.

Regular languages of operation trees

- Let $\mathcal{S}$ be a graph class that admits $\mathcal{L}[n]$-uniform $\preceq$-related kernels in $\mathcal{S}$, of size $\leq \rho(n)$ for a binary relation $\preceq$ on $\mathcal{S}$ and a (computable) function $\rho : \mathbb{N} \rightarrow \mathbb{N}$.
- Let $\mathcal{O}$ be a set of operations on $\mathcal{S}$, satisfying $\mathcal{L}$-composition and $\preceq$-monotonicity.
- Let d be the maximum arity and t be the maximum dimension of any operation of O.
- Let $\mathcal{O}$-trees be the class of all trees over O and $\Gamma(r)$ be the index of the $\equiv_{r,\mathsf{MSO}}$ relation over this class.

Regular languages of operation trees

- Let $\mathcal{S}$ be a graph class that admits $\mathcal{L}[n]$-uniform $\preceq$-related kernels in $\mathcal{S}$, of size $\leq \rho(n)$ for a binary relation $\preceq$ on $\mathcal{S}$ and a (computable) function $\rho : \mathbb{N} \rightarrow \mathbb{N}$.
- Let $\mathcal{O}$ be a set of operations on $\mathcal{S}$, satisfying $\mathcal{L}$-composition and $\preceq$-monotonicity.
- Let d be the maximum arity and t be the maximum dimension of any operation of O.
- Let $\mathcal{O}$-trees be the class of all trees over O and $\Gamma(r)$ be the index of the $\equiv_{r,\text{MSO}}$ relation over this class.
- Let $\mathcal{T}$ be a regular subclass of $\mathcal{O}$-trees that is defined by an MSO sentence of rank r.

Kernel bootstrapping using regular tree languages

Theorem

Let $\mathcal{Z}$ be the graph class obtained by "applying" the trees of $\mathcal{T}$ to the graphs of $\mathcal{S}$. Let $\Lambda(n)$ be the index of the $\equiv_{n,\mathcal{L}}$ relation over $\mathcal{Z}$. Then the following holds:

Kernel bootstrapping using regular tree languages

Theorem

Let $\mathcal{Z}$ be the graph class obtained by "applying" the trees of $\mathcal{T}$ to the graphs of $\mathcal{S}$. Let $\Lambda(n)$ be the index of the $\equiv_{n,\mathcal{L}}$ relation over $\mathcal{Z}$. Then the following holds:

For every $n \in \mathbb{N}$ and every graph $G \in \mathcal{Z}$ given in the form of its tree representation over $\mathcal{O}$,

Kernel bootstrapping using regular tree languages

Theorem

Let $\mathcal{Z}$ be the graph class obtained by "applying" the trees of $\mathcal{T}$ to the graphs of $\mathcal{S}$. Let $\Lambda(n)$ be the index of the $\equiv_{n,\mathcal{L}}$ relation over $\mathcal{Z}$. Then the following holds:

For every $n \in \mathbb{N}$ and every graph $G \in \mathcal{Z}$ given in the form of its tree representation over $\mathcal{O}$, there is an $\mathcal{L}[n]$-uniform kernel H in $\mathcal{Z}$,

Kernel bootstrapping using regular tree languages

Theorem

Let $\mathcal{Z}$ be the graph class obtained by "applying" the trees of $\mathcal{T}$ to the graphs of $\mathcal{S}$. Let $\Lambda(n)$ be the index of the $\equiv_{n,\mathcal{L}}$ relation over $\mathcal{Z}$. Then the following holds:

For every $n \in \mathbb{N}$ and every graph $G \in \mathcal{Z}$ given in the form of its tree representation over $\mathcal{O}$, there is an $\mathcal{L}[n]$-uniform kernel H in $\mathcal{Z}$, that is computable in f.p.t time,

Kernel bootstrapping using regular tree languages

Theorem

Let $\mathcal{Z}$ be the graph class obtained by "applying" the trees of $\mathcal{T}$ to the graphs of $\mathcal{S}$. Let $\Lambda(n)$ be the index of the $\equiv_{n,\mathcal{L}}$ relation over $\mathcal{Z}$. Then the following holds:

For every $n \in \mathbb{N}$ and every graph $G \in \mathcal{Z}$ given in the form of its tree representation over $\mathcal{O}$, there is an $\mathcal{L}[n]$-uniform kernel H in $\mathcal{Z}$, that is computable in f.p.t time, and that is of size $\leq \eta(1)$ where for $1 \leq h \leq \Lambda(n) \times \Gamma(r)$, we have $\eta(h) = (d \cdot \eta(h+1))^t$, and $\eta(\Lambda(n) \times \Gamma(r)) = \rho(n)$.

Kernel bootstrapping using regular tree languages

Theorem

Let $\mathcal{Z}$ be the graph class obtained by "applying" the trees of $\mathcal{T}$ to the graphs of $\mathcal{S}$. Let $\Lambda(n)$ be the index of the $\equiv_{n,\mathcal{L}}$ relation over $\mathcal{Z}$. Then the following holds:

For every $n \in \mathbb{N}$ and every graph $G \in \mathcal{Z}$ given in the form of its tree representation over $\mathcal{O}$, there is an $\mathcal{L}[n]$-uniform kernel H in $\mathcal{Z}$, that is computable in f.p.t time, and that is of size $\leq \eta(1)$ where for $1 \leq h \leq \Lambda(n) \times \Gamma(r)$, we have $\eta(h) = (d \cdot \eta(h+1))^t$, and $\eta(\Lambda(n) \times \Gamma(r)) = \rho(n)$. Further, H is $\preceq$-related to G.

Conclusion

Summary

- Background from logic
- Feferman-Vaught composition
- Kernels for cographs
- Generalizations
- Implications for well-quasi-ordered classes
- Bootstrapping on kernels

Future direction

Open question

Is there a class of graphs of unbounded clique-width that can be generated using operations satisfying composition?

Future direction

Open question

Is there a class of graphs of unbounded clique-width that can be generated using operations satisfying composition?

If so, that would disprove (the long-standing) conjecture by Seese that decidability of MSO-SAT implies bounded clique-width!

Mikka Nandri!

References

- A. Sankaran. A finitary analogue of the downward Löwenheim-Skolem property. In *Proceedings of CSL 2017, Stockholm, Sweden, August 20-24, 2017*, Vol. 82, pages 37:1 - 37:21, 2017.
- A. Sankaran, B. Adsul, and S. Chakraborty. A generalization of the Łoś-Tarski preservation theorem. *Annals of Pure and Applied Logic*, 167(3):189 - 210, 2016.
- A. Sankaran, B. Adsul, and S. Chakraborty. A generalization of the Łoś-Tarski preservation theorem over classes of finite structures. In *Proceedings of MFCS 2014, Budapest, Hungary, August 25-29, 2014, Part I*, pages 474 - 485, 2014.