PhD defence

Friday 17 November 2023

MONOIDAL WIDTH

Elena Di Lavore

Tallinn University of Technology

MOTIVATION

checking MSO formulae

Some problems on graphs are easier to solve on structurally simple inputs.

bounded tree width
or clique width

time linear
in the size

[Courcelle 1990, Courcelle, Makowsky, Rotics 2000]

OVERVIEW

Compositional problems on morphisms in monoidal categories are easier to solve on structurally simple inputs.

bounded monoidal width

- Introduce monoidal width
- Capture tree width and clique width
- Some compositional problems

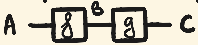
OUTLINE

- [Monoidal decompositions]
- Categories with copy and biproducts
- Tree width
- Clique width
- Fixed-parameter tractability

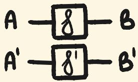
DECOMPOSING MORPHISMS IN MONOIDAL CATEGORIES

There are two operations in monoidal categories :

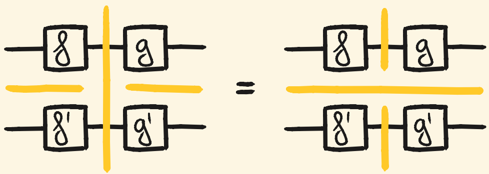
- composition $;_B$ ⇝ resource sharing, synchronisation
  ⇒ COSTLY



- monoidal product $\otimes$ ⇝ processes side-by-side
  ⇒ CHEAP



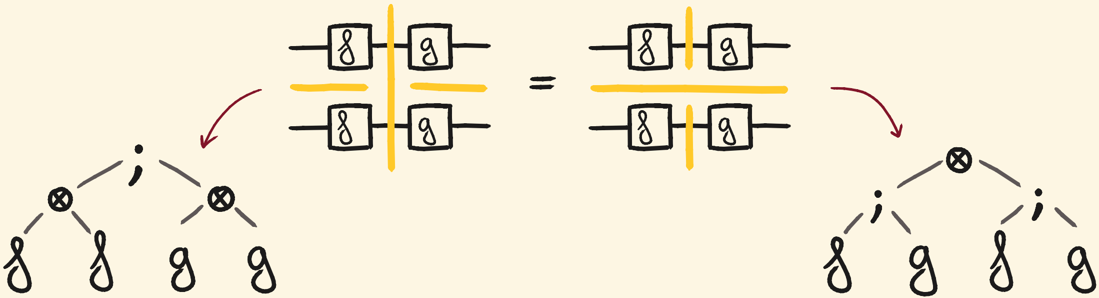
$$(f \otimes f')\,;(g \otimes g') = (f\,;g) \otimes (f'\,;g')$$



MONOIDAL DECOMPOSITIONS

A monoidal decomposition $d \in \mathcal{D}_f$ of $f : X \to Y$ is

$$
\begin{aligned}
d \; ::= \; & (f) \\
\mid \; & d_1 \,;\, d_2 \quad \text{if } f = f_1 ; f_2 ,\ d_1 \in \mathcal{D}_{f_1},\ d_2 \in \mathcal{D}_{f_2} \\
\mid \; & d_1 \otimes d_2 \quad \text{if } f = f_1 \otimes f_2 ,\ d_1 \in \mathcal{D}_{f_1},\ d_2 \in \mathcal{D}_{f_2}
\end{aligned}
$$



MONOIDAL WIDTH

WEIGHT FUNCTION

$w : \mathrm{morph}\,\mathcal{C} \to \mathbb{N}$ such that

- $w(f ;_Y g) \leq w(f) + w(\mathbb{1}_Y) + w(g)$
- $w(f \otimes g) \leq w(f) + w(g)$

WIDTH OF A DECOMPOSITION $\rightsquigarrow$ cost of the most expensive operation

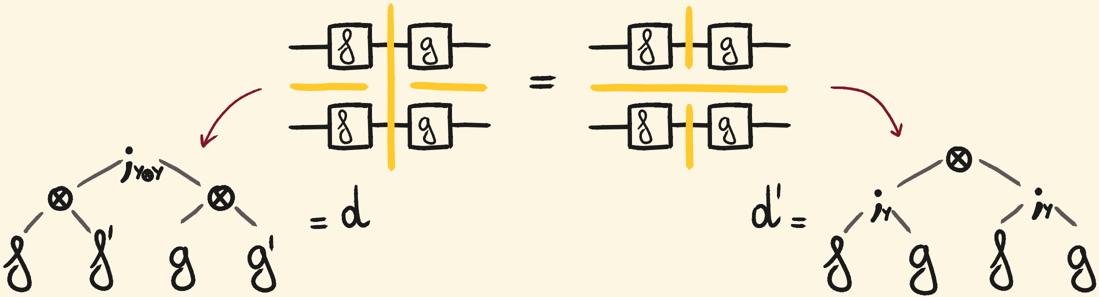
$$
\begin{aligned}
wd(d) := \;& w(f) && d = (f) \\
\mid\;& \max\{wd(d_1), w(\mathbb{1}_Y), wd(d_2)\} && d = d_1 \overset{;_Y}{\wedge} d_2 \\
\mid\;& \max\{wd(d_1), wd(d_2)\} && d = d_1 \overset{\otimes}{\wedge} d_2
\end{aligned}
$$

MONOIDAL WIDTH

$$mwd(f) := \min_{d \in \mathcal{D}_f} wd(d)$$

cost of a cheapest decomposition

MONOIDAL WIDTH INCENTIVISES PARALLELISM

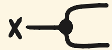


$$wd(d) = \max\{w(f), w(g), 2 \cdot w(\mathbb{1}_Y)\} \geq \max\{w(f), w(g), w(\mathbb{1}_Y)\} = wd(d')$$

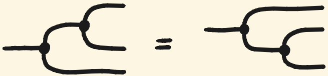
OUTLINE

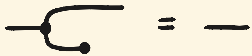
- Monoidal decompositions
- [Categories with copy and biproducts]
- Clique width
- Tree width
- Fixed-parameter tractability

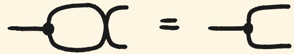
CATEGORIES WITH COPY



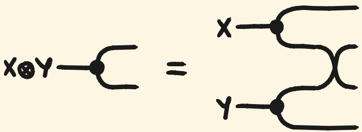
COCOMMUTATIVE COMONOID

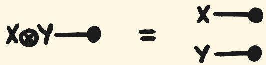






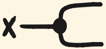
COHERENCE

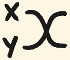




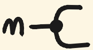
MONOIDAL WIDTH OF COPY

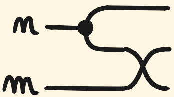
WEIGHT FUNCTION

$w\Big($  $\Big) := 2 \cdot w(X)$

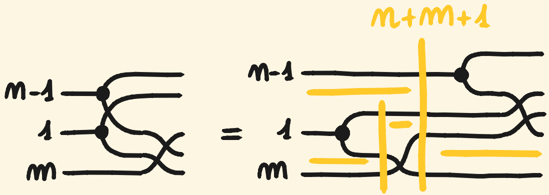
$w\Big($  $\Big) := w(X) + w(Y)$

PROPOSITION

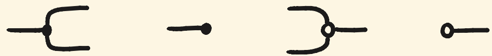
$\mathrm{mwd}\Big($  $\Big) \leq n+1$



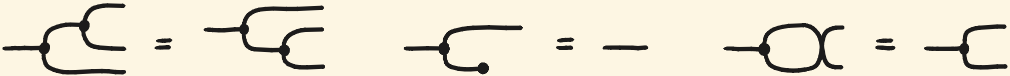
$=$



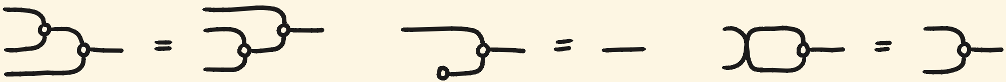
CATEGORIES WITH BIPRODUCTS



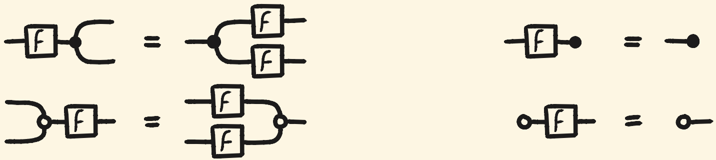
COCOMMUTATIVE COMONOID



COMMUTATIVE MONOID



NATURALITY



[Fox 1976]

MONOIDAL WIDTH OF 'MATRICES'

WEIGHT FUNCTION

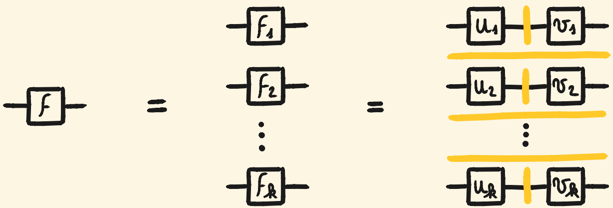
$$w(f : m \to n) := \max\{m, n\}$$

RANK OF MORPHISMS

$$\mathrm{rank}(\boxed{f}) := \min\{k \in \mathbb{N} \mid \boxed{f} = \boxed{u} \overset{k}{-} \boxed{v}\}$$

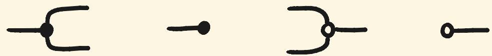
THEOREM

$$\max_i \mathrm{rank}(f_i) \leq \mathrm{mwd}\, f \leq \max_i \mathrm{rank}(f_i) + 1$$

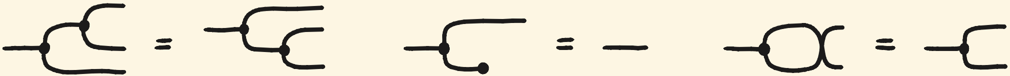


↳ unique ⊗-factorisation

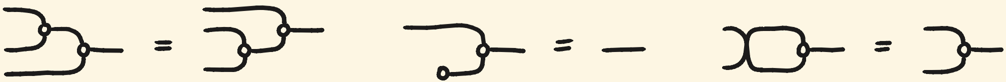
BIALGEBRA : THE PROP OF $\mathbb{N}$ - MATRICES



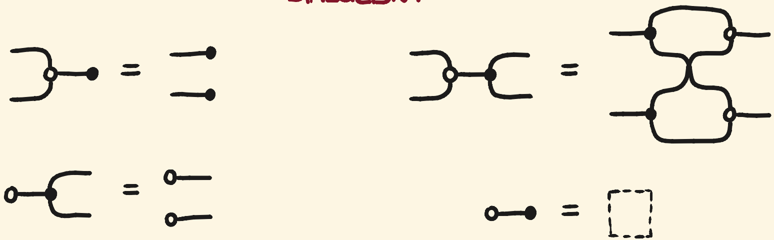
COCOMMUTATIVE COMONOID



COMMUTATIVE MONOID



BIALGEBRA



[Zanasi 2015]

MONOIDAL WIDTH OF MATRICES

FACT

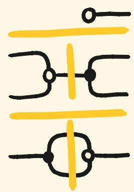
The minimal vertical cut in a matrix is its rank

$$\min\{k \in \mathbb{N} \mid A = B ;_k C\} = \operatorname{rank} A$$

COROLLARY

$$\max_i \operatorname{rank} A_i \leq \operatorname{mwd} A \leq \max_i \operatorname{rank} A_i + 1$$

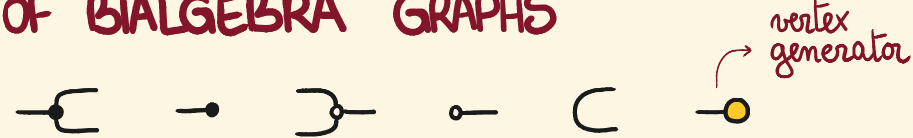
$$A = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix} = A_1 \oplus A_2 \oplus A_3 =$$



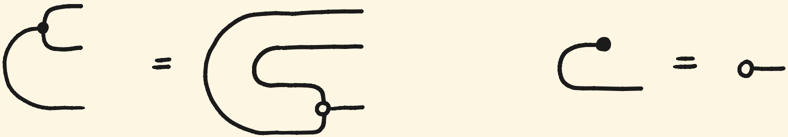
OUTLINE

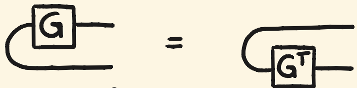
- Monoidal decompositions
- Categories with copy and biproducts
- [Clique width]
- Tree width
- Fixed-parameter tractability

A PROP OF BIALGEBRA GRAPHS



bialgebra equations +

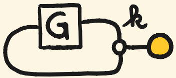


⇝ the cup transposes 

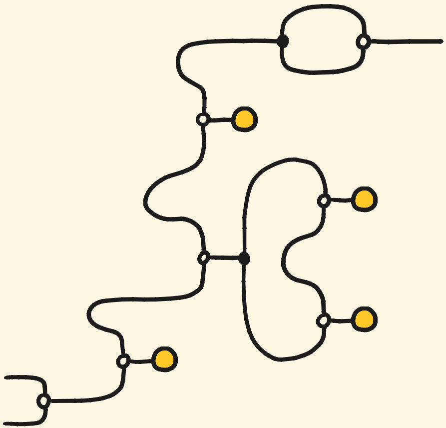
and captures equivalence of adjacency matrices

$G + G^T = H + H^T \iff$ 

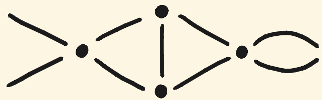
GRAPHS AS MORPHISMS – EXAMPLE



$\rightsquigarrow$ graph on k vertices given by the adjacency matrix $[G]$

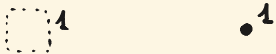


$\rightsquigarrow$

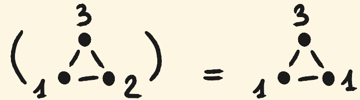
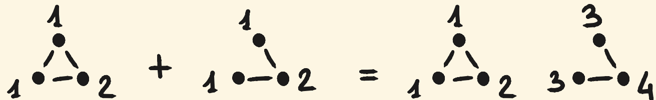


CLIQUE DECOMPOSITIONS

Terms constructed with constants:

- $\emptyset$ • v $\rightsquigarrow$ 

and operations:

- $\mathrm{Rename}^n_{i\to j}$ $\rightsquigarrow$ $\mathrm{Rename}^3_{2\to 1}$ 
- $\mathrm{Edge}^n_{i,j}$ $\rightsquigarrow$ $\mathrm{Edge}^2_{1,2}$ 
- $+$ $\rightsquigarrow$ 

<u>ex</u>

 $= \mathrm{Rename}^2_{2\to 1}\mathrm{Edge}^2_{1,2}(v + \mathrm{Rename}^2_{2\to 1}\mathrm{Edge}^2_{1,2}(v+v))$

[Courcelle, Engelfriet, Rozenberg 1993]

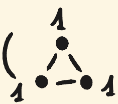
CLIQUE WIDTH

Each constant and operation has a cost.

- $w(\emptyset) = 1$
- $w(v) = 1$
- $w(\mathrm{Rename}^n_{i \to j}) = n$
- $w(\mathrm{Edge}^n_{i,j}) = n$
- $w(+) = 0$

CLIQUE WIDTH

$$\mathrm{cwd}(G) := \min_{t \in T_G} \max_{o \in t} w(o)$$

<u>ex</u> cwd(

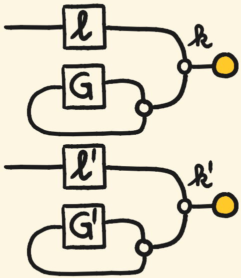
) = 2

[Courcelle, Engelfriet, Rozenberg 1993]

CLIQUE OPERATIONS, CATEGORICALLY

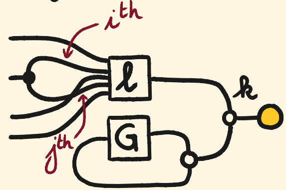
- $\emptyset$ = 
- v = 
- $(G, \ell) + (G', \ell')$

=



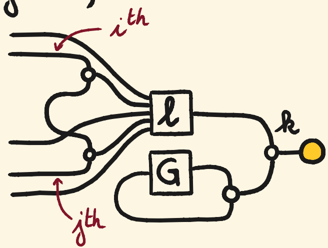
- $\mathrm{Rename}^n_{i \to j}(G, \ell)$

=



- $\mathrm{Edge}^n_{i,j}(G, \ell)$

=

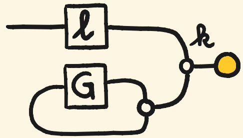


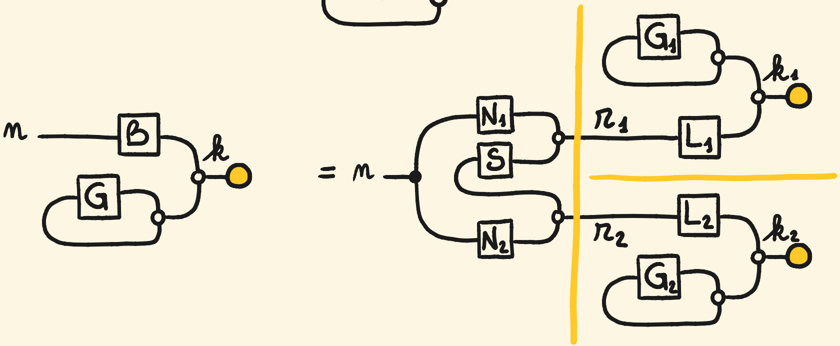
CLIQUE WIDTH & MONOIDAL WIDTH

WEIGHT FUNCTION

$$w([G], B) := \mathrm{rank}\, G + \mathrm{rank}\, B$$

THEOREM

$\frac{1}{2}\, \mathrm{rwd}(G, \ell) \leq \mathrm{mwd}($  $) \leq 2\, \mathrm{rwd}(G, \ell)$



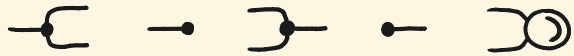
COROLLARY

Clique width is equivalent to monoidal width in BGraph.

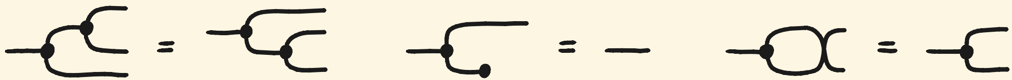
OUTLINE

- Monoidal decompositions
- Categories with copy and biproducts
- Clique width
- [• Tree width]
- Fixed-parameter tractability

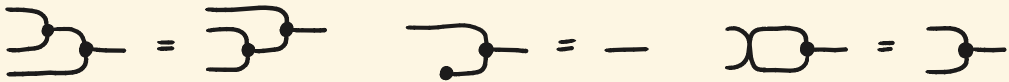
A PROP OF FROBENIUS GRAPHS



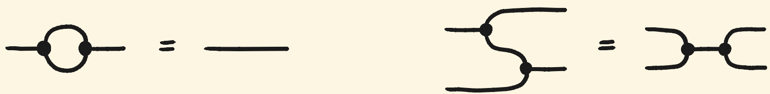
COCOMMUTATIVE COMONOID



COMMUTATIVE MONOID

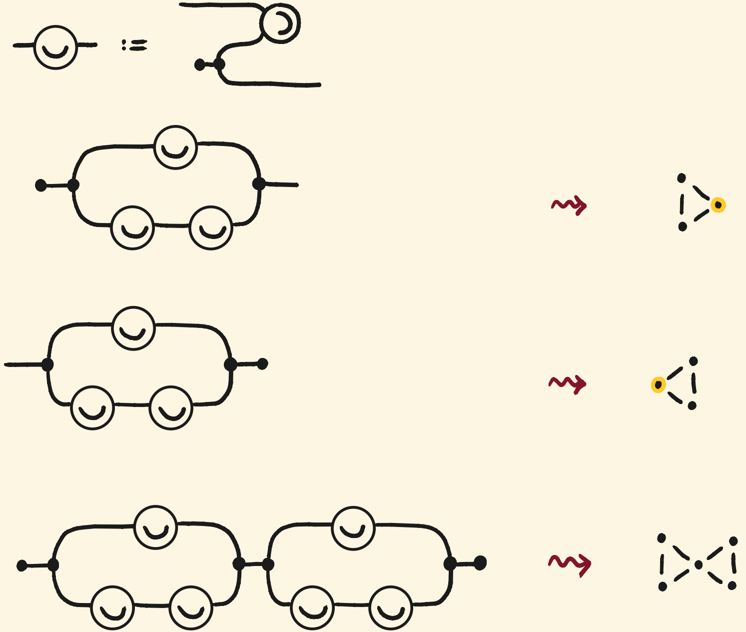


FROBENIUS



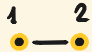
[Bonchi, Seeber & Sobociński 2018]

GRAPHS AS MORPHISMS - EXAMPLE

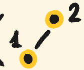
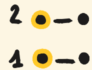


TREE DECOMPOSITIONS

Terms constructed with constants:

- $\emptyset$ • e $\rightsquigarrow$  

and operations:

- Relab^n_f $\rightsquigarrow$ $\mathrm{Relab}^3_f\big($  $\big) =$  $f : 2 \mapsto 3$
- Vert^n_i $\rightsquigarrow$ $\mathrm{Vert}^2_2\big($  $\big) =$ 
- $\mathrm{Fuse}^n_{i,j}$ $\rightsquigarrow$ $\mathrm{Fuse}^2_{1,2}\big($  $\big) =$ 
- $+$ $\rightsquigarrow$  $+$  $=$ 

<u>ex</u>  $= \mathrm{Relab}^3_i\, \mathrm{Fuse}^4_{2,3}(e + \mathrm{Relab}^3_i\, \mathrm{Fuse}^4_{2,3}(e+e))$

[Robertson & Seymour 1983, Courcelle 1990]

TREE WIDTH

Each constant and operation has a cost.

- $w(\emptyset) = 0$
- $w(e) = 2$
- $w(\mathrm{Relab}^n_g) = n$
- $w(\mathrm{Vert}^n_i) = n$
- $w(\mathrm{Fuse}^n_{i,j}) = n$
- $w(+) = 0$

TREE WIDTH

$$\mathrm{twd}(G) := \min_{t \in T_G} \max_{o \in t} w(o)$$

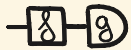
ex $\mathrm{twd}($  $) = 4$

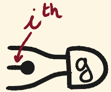
[Robertson & Seymour 1983, Courcelle 1990]

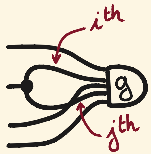
TREE OPERATIONS, CATEGORICALLY

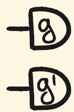
- $\emptyset \quad =$ 

- $e \quad =$ 

- $\mathrm{Relab}^{n}_{g}(G,c) \quad =$ 

- $\mathrm{Vert}^{n}_{i}(G,c) \quad =$ 

- $\mathrm{Fuse}^{n}_{i,j}(G,c) \quad =$ 

- $(G,c) + (G',c') \quad =$ 

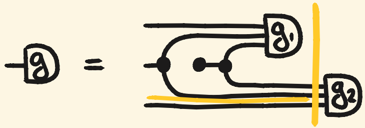
TREE WIDTH & MONOIDAL WIDTH

WEIGHT FUNCTION

$$w(X \to G \leftarrow Y) := |\mathrm{vertices}\, G|$$

THEOREM

$$\frac{1}{2}\,\mathrm{bwd}(G,c) \leq \mathrm{mwd}(n \xrightarrow{c} G \leftarrow \emptyset) \leq \mathrm{bwd}(G,c) + 1$$



COROLLARY

Tree width is equivalent to monoidal width in FGraph

OUTLINE

- Monoidal decompositions
- Categories with copy and biproducts
- Clique width
- Tree width
- [Fixed-parameter tractability]

COMPOSITIONAL ALGORITHMS

$\mathcal{C}, \mathcal{D}$ monoidal categories

$P : \mathcal{C} \to \mathcal{D}$ monoidal functor

space of solutions

$w : \mathrm{morph}\,\mathcal{C} \to \mathbb{N}$ weight function

$\rightsquigarrow$ cf. MSOL-smooth operations and MSOL-inductive classes of τ-structures

$\rightsquigarrow$ cf. number of vertices

a <u>compositional algorithm</u> for P w.r.t. w computes

1. $P(f)$ in time $\mathcal{O}(c(w(f)))$
2. $P(f) ;_Y P(g)$ in $\mathcal{D}$ in time $\mathcal{O}(c(w(Y)))$
3. $P(f) \otimes P(f')$ in $\mathcal{D}$ in time $\mathcal{O}(c(0))$

for some increasing function $c : \mathbb{N} \to \mathbb{N}$.

[cf. Courcelle & Makowsky 2002]

FEFERMAN-VAUGHT-MOSTOWSKI THEOREM

THEOREM

For τ-structures A, B, A', B' and a set X of sources, if $A \equiv_{MSO(\tau)} A'$ and $B \equiv_{MSO(\tau)} B'$, then $A \oplus_X B \equiv_{MSO(\tau)} A' \oplus_X B'$.
Computing $A \oplus_X B \models \varphi$ given $Th_{MSO_q(\tau)}(A)$ and $Th_{MSO_q(\tau)}(B)$ does not depend on $A \oplus_X B$.

COROLLARY

There is a monoidal functor

$$\begin{aligned} Th : \mathrm{Struct}(\tau) &\longrightarrow \mathrm{Struct}(\tau)/_{\equiv_{MSO(\tau)}} \\ A &\longmapsto Th_{MSO_q(\tau)}(A) \end{aligned}$$

that can be computed with a compositional algorithm.

[Feferman & Vaught 1959, Courcelle & Makowsky 2002]

MONOIDAL FIXED-PARAMETER TRACTABILITY

THEOREM

Functorial problems $P : \mathcal{C} \to \mathcal{D}$ with a compositional algorithm are fixed-monoidal-width tractable: given $f : A \to B$ with a monoidal decomposition d of width $\leq k$, computing $P(f)$ takes $O(c(k) \cdot \mathrm{size}(d))$.

COROLLARY

Checking MSO formulae on graphs is tractable on inputs of bounded monoidal width.

COLIMITS COMPOSITIONALLY

$\mathcal{C}$ category with finite colimits.

CATEGORY OF DIAGRAMS

$\mathsf{Diag}\,\mathcal{C}$ is the slice category $\mathsf{Graph}/|\mathcal{C}|$ restricted to finite graphs. There is a functor $\mathrm{colim}\colon \mathsf{Diag}\,\mathcal{C} \to \mathcal{C}$.

DISCRETE COSPANS OF DIAGRAMS

$\mathsf{CDiag}\,\mathcal{C}$ is the category of cospans of diagrams restricted to discrete diagrams.

THEOREM

There is a monoidal functor $\mathrm{colim}\colon \mathsf{CDiag}\,\mathcal{C} \to \mathsf{Cospan}\,\mathcal{C}$ that extends $\mathrm{colim}\colon \mathsf{Diag}\,\mathcal{C} \to \mathcal{C}$.

[Rosebrugh, Sabadini, Walters 2005]

COMPUTING COLIMITS COMPOSITIONALLY

PROPOSITION

There is a compositional algorithm for computing colimits of fixed shape in FSet and finite presheaf categories.

COROLLARY

Computing colimits of fixed shape in FSet and finite presheaf categories is fixed-monoidal-width tractable.

$d : G \to |\mathrm{FSet}|$

$\operatorname{colim} d = \left(\coprod_{v \in G} d(v) \right) / \sim$ $\qquad$ $U +_Y V = U + V / \sim$

$\rightsquigarrow \mathcal{O}\left(\left| \coprod_{v \in G} d(v) \right|^5 \right)$ $\qquad$ $\rightsquigarrow \mathcal{O}(|Y|^5)$

SUMMARY & FUTURE WORK

- Monoidal width measures structural complexity of morphisms in monoidal categories.
- Monoidal width captures rank width and tree width.
- Compositional problems are fixed-monoidal-width tractable.

- Could we capture more graph parameters?
- More examples of compositional problems
- How do we find efficient decompositions?

APPENDIX

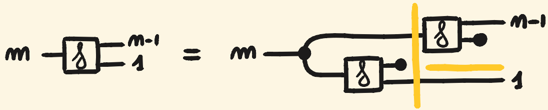
CATEGORIES WITH BIPRODUCTS

THEOREM

$$\max_i \operatorname{rank}(f_i) \leq \operatorname{mwd} f \leq \max_i \operatorname{rank}(f_i) + 1$$

PROOF STRATEGY

- optimal decompositions start with $\otimes$ whenever possible
- morphisms have unique $\otimes$-decompositions
- $f : m \to n \Rightarrow \operatorname{mwd} f \leq \min\{m, n\} + 1$



- $\operatorname{mwd} f \leq \operatorname{rank} f$

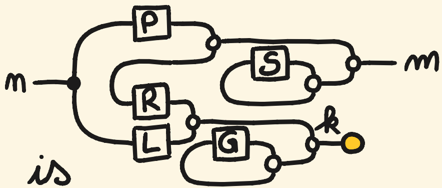
  if f non $\otimes$-decomposable $\Rightarrow \operatorname{mwd} f = \operatorname{rank} f$

$\square$

BIALGEBRA GRAPHS

MGraph is the prop where

- morphisms $g : n \to m$ are $g = (\overset{k\times k}{[G]}, \overset{k\times n}{L}, \overset{k\times m}{R}, \overset{m\times n}{P}, \overset{m\times m}{[S]})$
  quotiented by $([G], L, R, P, [F]) \simeq ([\sigma G \sigma^T], \sigma L, \sigma R, P, [S])$
  for permutations $\sigma : k \to k$

$g \mapsto$ 

in BGraph

- composition $g_1 ; g_2$ is

$$\left(\left[\begin{pmatrix} G_1 & R_1 L_2^T \\ 0 & G_2 + L_2 S_1 L_2^T \end{pmatrix} \right], \begin{pmatrix} L_1 \\ L_2 P_1 \end{pmatrix}, \begin{pmatrix} R_1 P_2^T \\ R_2 + L_2 (S_1 + S_1^T) P_2^T \end{pmatrix}, P_2 P_1, [S_2 + P_2 S_1 P_2^T] \right)$$

- monoidal product $g_1 \otimes g_2$ is
  $([G_1 \oplus G_2], L_1 \oplus L_2, R_1 \oplus R_2, P_1 \oplus P_2, [S_1 \oplus S_2])$

BIALGEBRA GRAPHS

THEOREM

$$\mathsf{MGraph} \simeq \mathsf{BGraph}$$

PROOF STRATEGY

- $\mathsf{MGraph} \simeq \mathsf{MAdj} + \mathsf{boundP}$

$$\begin{cases} i_1(B,[G]) = ([()], !, !, B, [G]) \\ i_2(k,P) = ([\mathbb{1}_k], P_2, \mathbb{1}, P_1, [\mathbb{1}]) \end{cases} \quad \text{inclusions}$$

any $g : m \to n$ factors as $g = i_1(a) ; i_2(v)$

$\Rightarrow \mathsf{MGraph} = \mathsf{MAdj} \otimes_P \mathsf{boundP}$ and MGraph is the coequaliser of $\mathsf{boundP} \otimes_P \mathsf{MAdj} \overset{s}{\underset{t}{\rightrightarrows}} \mathsf{MAdj} + \mathsf{boundP}$

all prop morphisms $\mathsf{MAdj} + \mathsf{boundP} \to P$ coequalise

$\Rightarrow \mathsf{MGraph} = \mathsf{MAdj} + \mathsf{boundP}$

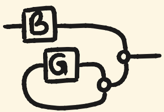
PROOF STRATEGY [CT'D]

- $\mathsf{boundP} \simeq \mathsf{Vert}$
  because boundP is also initial

- $\mathsf{MAdj} \simeq \mathsf{Adj}$ because it is also the coequaliser

$$\begin{tikzcd}
A \arrow[r, shift left, "s"] \arrow[r, shift right, "t"'] & \mathsf{Bialg} + \mathsf{Cup} \arrow[r, "{[b,c]}"] \arrow[dr, "p"'] & \mathsf{MAdj} \arrow[d, dashed, "\bar{p}"] \\
& & P
\end{tikzcd}$$

$$\left.\begin{array}{l} b(\mathrm{Mat}^{-1}(A)) := (A, [\emptyset]) \\ c(C) := (i_2, [\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}]) \end{array}\right\} \to q := [b,c] \text{ is coequalising}$$

$\bar{p}(B, [G]) := p\Big($  $\Big)$ is the unique prop morphism such that $q ; \bar{p} = p$ $\square$

BRANCH DECOMPOSITIONS

G undirected graph

BRANCH DECOMPOSITION

(Y, b) where

- Y is a subcubic tree (= any node has at most 3 neighbours)
- $b : \text{leaves}(Y) \xrightarrow{\sim} \text{edges}(G)$ labelling bijection

WIDTH OF (Y, b)

$$\text{wd}(Y, b) := \max_{e \in \text{edges} Y} |\text{ends}\, A_e \cap \text{ends}\, B_e|$$

$\{A_e, B_e\}$ partition of E given by e through b

BRANCH WIDTH

$$\text{bwd}(G) := \min_{(Y,b)} \text{wd}(Y, b)$$

$\rightsquigarrow$ cost of a cheapest decomposition

INDUCTIVE BRANCH DECOMPOSITIONS

$\Gamma = ((V,E),X)$ hypergraph with sources

- $T = (\Gamma)$ if $|E| \leq 1$
- $T = T_1 \diagup^{\Gamma} \diagdown T_2$ if T_i decomposition of $\Gamma_i = ((V_i, E_i), X_i)$ st
  - $E = E_1 + E_2$
  - $V = V_1 \cup V_2$
  - $X_i = (X \cap V_i) \cup (V_1 \cap V_2)$

$$wd(T) := \begin{cases} |V| & T = (\Gamma) \\ \max\{wd\, T_1, |X|, wd\, T_2\} & T = T_1 \diagup^{\Gamma} \diagdown T_2 \end{cases}$$

BRANCH WIDTH $\geq$ MONOIDAL WIDTH

PROPOSITION

T inductive branch decomposition of $\Gamma = (G, X)$

$\Rightarrow$ there is a monoidal decomposition d of $g = X \leftrightarrow G \leftarrow \emptyset$

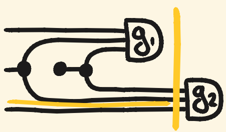
st $wd(d) \leq wd\, T + 1$

PROOF STRATEGY

By induction on T

- $T = (\Gamma) \Rightarrow d := (g)$ and $wd(T) := |V| =: wd(d)$
- $T = \begin{tikzcd}[column sep=small, row sep=small] & \Gamma \arrow[dl, dash] \arrow[dr, dash] & \\ T_1 & & T_2 \end{tikzcd}$, T_i decomposition of Γ_i

  $\Rightarrow \exists\, d_i$ monoidal decomposition of $g_i = X_i \leftrightarrow G_i \leftarrow \emptyset$ st $wd\, d_i \leq wd\, T_i + 1$ by induction

  $\Rightarrow d :=$ 

by lemma on categories with copy

$\square$

BRANCH WIDTH $\leq$ MONOIDAL WIDTH

PROPOSITION

d monoidal decomposition of $g = X \xrightarrow{l} G \xleftarrow{r} Y$

$\Rightarrow$ there is an inductive branch decomposition T of

$\Gamma = (G, l(X) \cup r(Y))$ st $\mathrm{wd}\, T \leq 2 \cdot \max\{\mathrm{wd}\, d, |X|, |Y|\}$

PROOF STRATEGY

By induction on d.

- $d = (g) \Rightarrow$ take any $T \Rightarrow \mathrm{wd}\, T \leq |V| =: \mathrm{wd}\, d$
- $d = d_1 \diagup^{jc} \diagdown d_2 \Rightarrow g =$

$$\begin{tikzcd}
X \arrow[dr, "l_1"'] & & C \arrow[dl, "r_1"'] \arrow[dr, "l_2"] \arrow[dd, dotted] & & Y \arrow[dl, "r_2"] \\
& G_1 \arrow[dr, "j_1"'] & & G_2 \arrow[dl, "j_2"] & \\
& & G & &
\end{tikzcd}$$

$\Rightarrow \exists T_i$ decompositions of $\Gamma_i = (G_i, l_i \cup r_i)$ by induction

$\Rightarrow \exists j_i(T_i)$ decompositions of $j_i(\Gamma_i) \subseteq G$ by lemma

$\Rightarrow T := j_1(T_1) \diagup^{\Gamma} \diagdown j_2(T_2)$

PROOF STRATEGY [CT'D]

- $d = \begin{array}{ccc} & \otimes & \\ d_1 & & d_2 \end{array} \Rightarrow g = g_1 \otimes g_2$

  $\Rightarrow \exists T_i$ decompositions of $\Gamma_i = (G_i, \ell_i \cup r_i)$ by induction

  $\Rightarrow T := \begin{array}{ccc} & \Gamma & \\ j_1(T_1) & & j_2(T_2) \end{array}$

$\square$

RANK DECOMPOSITIONS

G undirected graph

RANK DECOMPOSITION

(Y, π) where

- Y is a subcubic tree (= any node has at most 3 neighbours)
- π : leaves$(Y) \xrightarrow{\simeq}$ vertices(G) labelling bijection

WIDTH OF (Y, π)

$$\mathrm{wd}(Y, \pi) := \max_{e \in \mathrm{edges}\, Y} \mathrm{rank}(X_e)$$

X_e adjacency matrix of the cut given by e through π

RANK WIDTH

$$\mathrm{rwd}(G) := \min_{(Y, \pi)} \mathrm{wd}(Y, \pi)$$

$\rightsquigarrow$ cost of a cheapest decomposition

INDUCTIVE RANK DECOMPOSITIONS

$\Gamma = ([G], B)$ graph with names

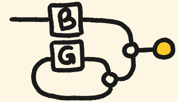
- $T = (\Gamma)$ if $|\text{vertices}\,\Gamma| \leq 1$
- $T = {}_{T_1}\!\diagup^{\Gamma}\diagdown_{T_2}$ if T_i decomposition of $\Gamma_i = ([G_i], B_i)$ st
  - $[G] = \left[\begin{pmatrix} G_1 & \mathcal{E} \\ \Phi & G_2 \end{pmatrix}\right]$
  - $B_1 = (A_1 | C)$, $B_2 = (A_2 | C^T)$, $B = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix}$

$$\text{wd}(T) := \begin{cases} \text{rank}\, B + \text{rank}\, G & T = (\Gamma) \\ \max\{\text{rank}\, B, \text{wd}\, T_1, \text{wd}\, T_2\} & T = {}_{T_1}\!\diagup^{\Gamma}\diagdown_{T_2} \end{cases}$$

RANK WIDTH $\geq$ MONOIDAL WIDTH

PROPOSITION

T inductive rank decomposition of $\Gamma = ([G], B)$

$\Rightarrow$ there is a monoidal decomposition d of $g =$ 

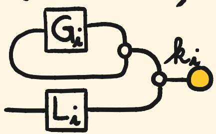
st $\mathrm{wd}\, d \leq 2\, \mathrm{wd}\, T$

PROOF STRATEGY

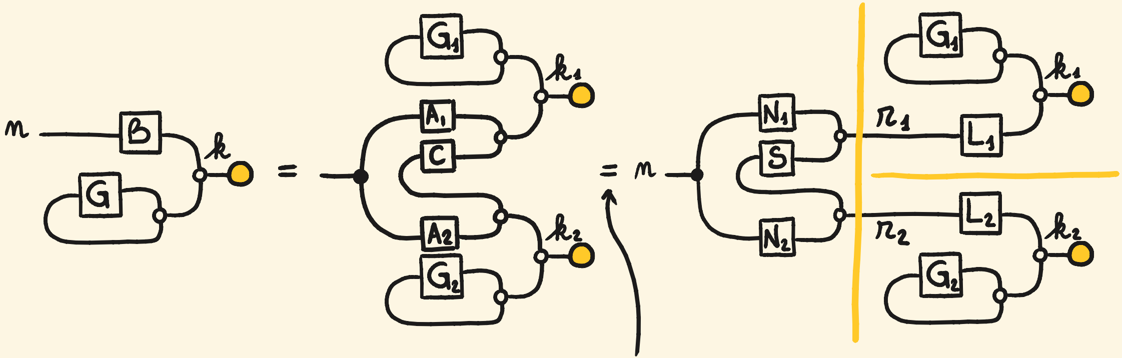
By induction on T

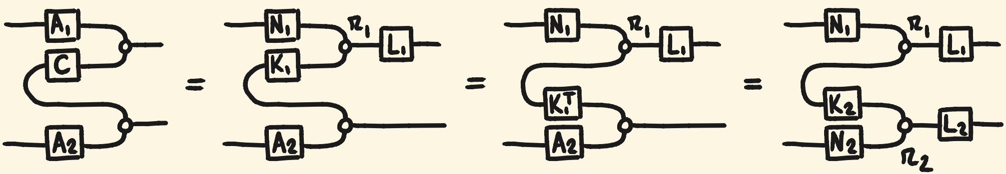
- $T = (\Gamma) \Rightarrow d = (g)$ and $\mathrm{wd}\, d := \mathrm{rank}\, B + \mathrm{rank}\, G$
- $T = T_1 \diagup^{\Gamma} \diagdown T_2$, T_i decomposition of $\Gamma_i = ([G_i], B_i)$

  $\Rightarrow \exists\, T_i'$ decomposition of $\Gamma_i' = ([G_i], L_i)$ by lemma

  $\Rightarrow \exists\, d_i$ decomposition of  by induction

PROOF STRATEGY [CT'D]





$$\mathrm{rank}(A_2 \mid C^T) = \mathrm{rank}\left((A_2 \mid K_1^T)\begin{pmatrix} 1 & 0 \\ 0 & L_1^T \end{pmatrix}\right) = \mathrm{rank}(A_2 \mid K_1^T)$$

□

RANK WIDTH $\leq$ MONOIDAL WIDTH

PROPOSITION

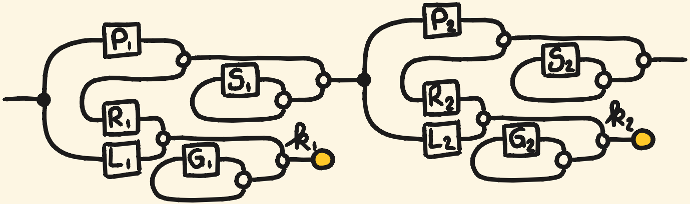
d monoidal decomposition of $g = ([G], L, R, P, [S])$
$\Rightarrow$ there is an inductive rank decomposition T of
$\Gamma = ([G], (L \mid R))$ st $\mathrm{wd}\, T \leq 2 \cdot \max\{\mathrm{wd}\, d, \mathrm{rank}\, L, \mathrm{rank}\, R\}$

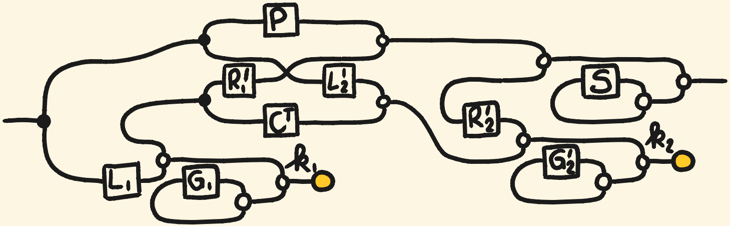
PROOF STRATEGY

By induction on d.

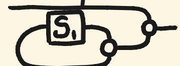
- $d = (g) \Rightarrow$ pick any T
- $d = d_1 \overset{;}{\wedge} d_2 \Rightarrow g = g_1 ; g_2$ (see next page $\circledast$)
- $d = d_1 \overset{\otimes}{\wedge} d_2 \Rightarrow g = g_1 \otimes g_2$

by induction, there are T_i decompositions of $\Gamma_i = ([G_i], (L_i \mid R_i))$
by lemma, there are T_i' decompositions of Γ_1 and Γ_2
$\Rightarrow T := T_1' \overset{\Gamma}{\wedge} T_2'$ $\square$





by induction, there are T_i decompositions of $\Gamma_i = ([G_i], (L_i | R_i))$

by lemma, there is T_2' decomposition of  Γ_i

by lemma, there are $\overline{T}_i$ decompositions of $\overline{\Gamma}_1 = ([G_1], (L_1 | R_1' | C))$ and $\overline{\Gamma}_2 = ([G_2'], (L_2' | R_2' | C^T))$

$\Rightarrow T = \overline{T}_1 \,/\, \Gamma \,\backslash\, \overline{T}_2$

FIXED-MONOIDAL-WIDTH TRACTABILITY

THEOREM

For a compositional problem $P : \mathcal{C} \to \mathcal{D}$, given $f : A \to B$ with a monoidal decomposition d of width $\leq k$, computing $P(f)$ takes $\mathcal{O}(c(k) \cdot \mathrm{size}(d))$.

PROOF STRATEGY

By induction on d.

- $d = (f) \Rightarrow \mathcal{O}(w(f) \cdot c(w(f)))$ by A1
- $d = d_1 ;_Y d_2 \Rightarrow c(k) \cdot (\mathrm{size}(d_1) + \mathrm{size}(d_2)) + c(w(Y))$
- $d = d_1 \otimes d_2 \Rightarrow c(k) \cdot (\mathrm{size}(d_1) + \mathrm{size}(d_2)) + c(0)$

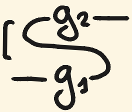
$\square$

MSO CHECKING IS COMPOSITIONAL

$$g = X \xrightarrow{\ell} G \xleftarrow{r} Y$$

$$[g] := [(G, \ell(X) \cup r(Y))]$$

$$[g_1 ;_C g_2] = [$$



$$] := [\mathrm{Fuse}_C((G_1, \ell_1(X) \cup r_1(C)) + (G_2, \ell_2(C) \cup r_2(Y)))]$$

COMPUTING COLIMITS

PROPOSITION

Computing colimits of a fixed shape in FSet is compositional.

PROOF STRATEGY

A1. $d : G \to |FSet| \Rightarrow \operatorname{colim} d = \left(\coprod_{v \in G} d(v) \right) / \sim$

$n := \left| \coprod_{v \in G} d(v) \right|$, $R_0 \in Mat_{Bool}(n, n)$

$R_0(x_1, x_2) = 1 \Leftrightarrow \exists\, e_1 = (u, v_1),\ e_2 = (u, v_2)\ \exists\, y \in d(u)$
st $d(e_1)(y) = x_1$, $d(e_2)(y) = x_2$
$\rightsquigarrow \mathcal{O}(|E| \cdot n)$

R transitive closure of R_0 $\rightsquigarrow \mathcal{O}(n^5)$

$\Rightarrow \mathcal{O}(|E| \cdot n^5)$

A2.

$$\begin{tikzcd}
 & & U+_Y V \arrow[d, phantom, "\vee"] & & \\
 & U \arrow[ur] & & V \arrow[ul] & \\
X \arrow[ur] & & Y \arrow[ul] \arrow[ur] & & Z \arrow[ul]
\end{tikzcd}
\Rightarrow \quad U+_Y V = U+V/\sim$$

$R_0 \in \mathrm{Mat}_{\mathrm{Bool}}(|Y|, |Y|)$

for $y \in Y$ $\quad R_0(u(y), v(y)) = 1 \qquad \rightsquigarrow \Theta(|Y|)$

R transitive closure of $R_0 \qquad \rightsquigarrow \Theta(|Y|^5)$

A3. $\quad U+V \qquad \rightsquigarrow \Theta(1)$

$\square$