Linear bounds between Cliquewidth and Component twin-width and applications

Ambroise Baril, Miguel Couceiro, Victor Lagerkvist

Université de Lorraine, CNRS, LORIA & Linköpings Universitet

April 2023

Contents

1. #*H*-COLORING
2. First bound
3. Second bound
4. Complexity consequences

k-COLORING

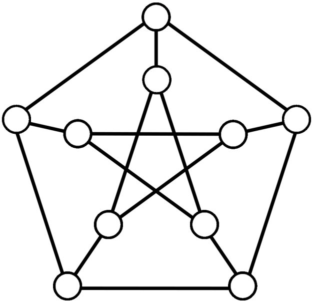


Figure: Instance of 3-COLORING

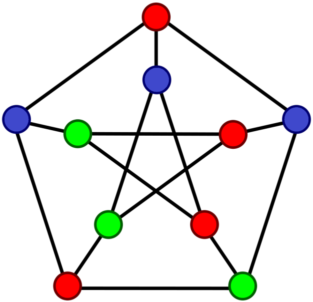
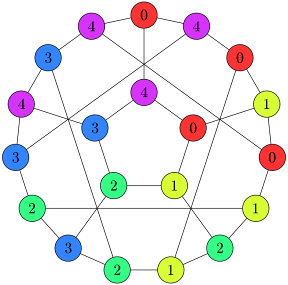
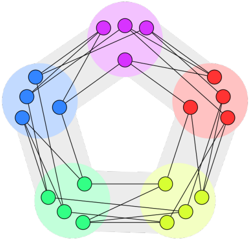


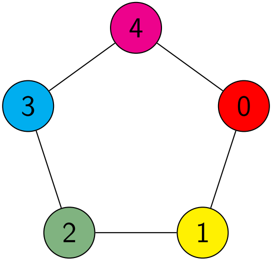
Figure: Solution of the instance

$$c : V_G \mapsto [k] \text{ such that } \forall (u,v) \in E_G,\ c(u) \neq c(v)$$

H-COLORING







Example of a C_5-COLORING

$f : V_G \to V_H$
$\forall (u,v) \in E_G, (f(u), f(v)) \in E_H$
f is an **Homomorphism**

k-COLORING = K_k-COLORING

Hard problem

Naive algorithm in time $|V_H|^{|V_G|}$

No algo in time $F(H) \times |V_G|^{F(H)}$ unless P=NP ($H = K_3$)

No algo in time $F(G) \times |V_H|^{O(1)}$ unless FPT=W[1] ($G = K_k$)

How to solve in practice ?

Hard problem

Naive algorithm in time $|V_H|^{|V_G|}$

No algo in time $F(H) \times |V_G|^{F(H)}$ unless P=NP ($H = K_3$)

No algo in time $F(G) \times |V_H|^{O(1)}$ unless FPT=W[1] ($G = K_k$)

How to solve in practice ?

Use structural properties of the graphs involved

Clique-width

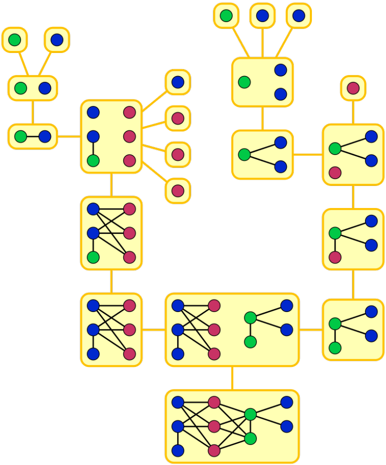


Figure: 3-expression of a graph

$\bullet_i$: vertex labelled by i

$G_1 \oplus G_2$: disjointed union

$\rho_{j \to i}(G)$: relabel the j with i

$\eta_{i,j}(G)$: construct an edge between every i and j

$\mathrm{cw}(G)$: number of labels

$\mathrm{linearcw}(G)$: number of labels where every $\oplus$ contains a $\bullet_i$ member

Application to counting homomorphisms

Parameterized complexity:

k-COLORING in time $(2^{|V_H|}-2)^{\mathrm{cw}(G)}$ [Lam20][1]

Fine-grained complexity:

#H-COLORING in time:
$(2\mathrm{cw}(H)+1)^{|V_G|}$ and $(\mathrm{linearcw}(H)+2)^{|V_G|}$ [Wah11][2]

[1] Lampis
[2] Wahlström

Exemple of a contraction sequence

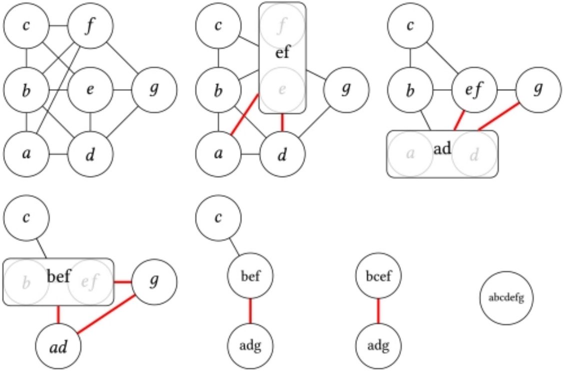


Figure: A contraction sequence of a graph

(Component) twin-width

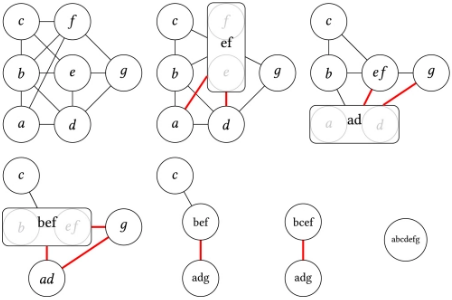


Figure: Contraction sequence of a graph

tww(G): Maximal red-degree [BKTW20][a]
ctww(G): Max red-component size [BKRT22][b]

No FPT algo for 3-COLOR param by tww(G):

3-COLOR is NP-hard on planar graphs

tww is bounded on planar graphs

[a] Bonnet, Kim, Thomassé, Watrigant
[b] Bonnet, Kim, Reinald, Thomassé

Applications to counting homomorphisms

Naive use of component twin-width for $\#H$-COLORING:

Parameterized complexity:

$$(2^{|V_H|} - 1)^{\mathrm{ctww}(G)}$$

Fine-grained complexity:

$$(\mathrm{ctww}(\mathrm{H}) + 2)^{|V_G|}$$

Comparing complexities

Which approach is the best ?

Parameterized complexity:

$$(2^{|V_H|}-1)^{\text{ctww}(G)} \text{ VS } (2^{|V_H|}-2)^{\text{cw}(G)}$$

Fine-grained complexity:

$$(\text{ctww(H)}+2)^{|V_G|} \text{ VS } (\text{2cw(H)}+1)^{|V_G|} \text{ and } (\text{linearcw(H)}+2)^{|V_G|}$$

We need to compare the two parameters cw and ctww.

Functional Equivalence

Using boolean-width (func equiv to cliquewidth) [BKRT22][3]

$$\mathrm{ctww}(G) \leq 2^{\mathrm{boolw}(G)+1} \leq 2^{\mathrm{cw}(G)+1}$$

AND

$$\mathrm{cw}(G) \leq 2^{\mathrm{boolw}(G)} \text{ and } \mathrm{boolw}(G) \leq 2^{\mathrm{ctww}(G)}$$

so

$$\mathrm{cw}(G) \leq 2^{2^{\mathrm{ctww}(G)}}$$

[3] Bonnet, Kim, Reinald, Thomassé

Contents

1 #*H*-COLORING

2 First bound

3 Second bound

4 Complexity consequences

Functional equivalence

We already know:

$$\mathrm{cw}(G) \leq 2^{2^{\mathrm{ctww}(G)}}$$

First contribution: Improved bound

I will prove

$$\text{cw}(G) \leq \text{ctww}(G) + 1$$

Take a contraction sequence of G of ctww k

Build a $(k+1)$-expression of G

Exemple of a contraction sequence

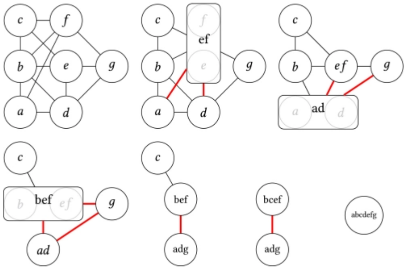


Figure: A contraction sequence of a graph

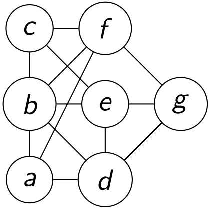
For $C = \{S_1, \dots, S_p\}$ red-component
Build φ_C a $(k+1)$-expression of $G[S_1 \uplus \dots \uplus S_p]$ with $\forall i, label(S_i) = i$

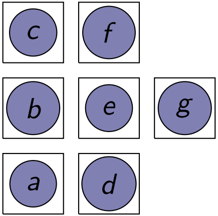
Same red-component = Same formula
Same set = Same label

Base case

Contraction sequence of ctww=3

We will use 4 labels: •, •, •, •: proves cw≤ 4





$\varphi_a =$ •

$\varphi_b =$ •

$\varphi_c =$ •

$\varphi_d =$ •

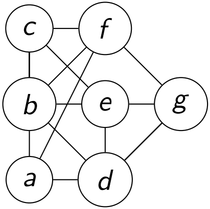
$\varphi_e =$ •

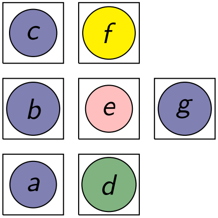
$\varphi_f =$ •

$\varphi_g =$ •

Red-component are singletons $\{a\}, \{b\}, \ldots$

Contracting e and f





$\varphi_a = \bullet$

$\varphi_b = \bullet$

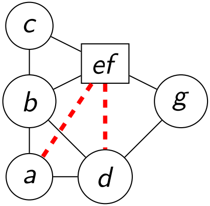
$\varphi_c = \bullet$

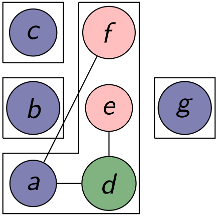
$\varphi_d = \bullet$

$\varphi_e = \bullet$

$\varphi_f = \bullet$

$\varphi_g = \bullet$

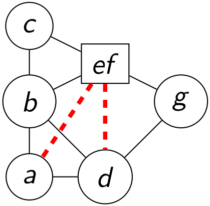


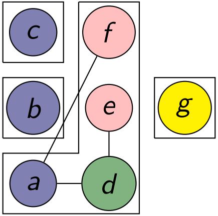


$\varphi_{adef} =$

$\rho_{\bullet \mapsto \bullet}$
$\eta_{\bullet,\bullet}\eta_{\bullet,\bullet}\eta_{\bullet,\bullet}$
$(\varphi_a \oplus \varphi_d \oplus$
$\varphi_e \oplus \varphi_f)$

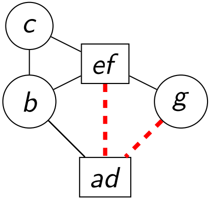
Contracting a and d

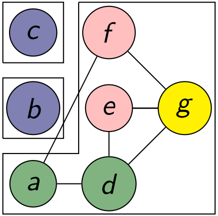




φ_{adef}

$\varphi_{g} = \bullet$





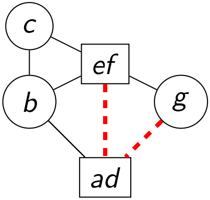
$\varphi_{adefg} =$

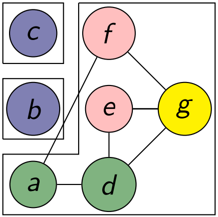
$\rho_{\bullet \mapsto \bullet}$

$\eta_{\bullet,\bullet}, \eta_{\bullet,\bullet}$

$(\varphi_{adef} \oplus \varphi_{g})$

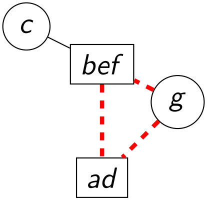
Contracting b and ef

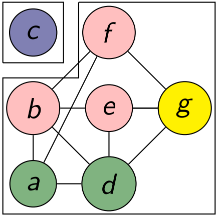




φ_{adefg}

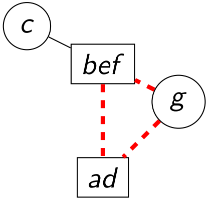
φ_{b}

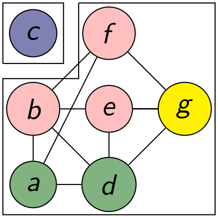




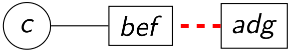
$$\varphi_{adbefg} = \rho_{\bullet \mapsto \bullet}\, \eta_{\bullet,\bullet}\, \eta_{\bullet,\bullet} (\varphi_{adefg} \oplus \varphi_{b})$$

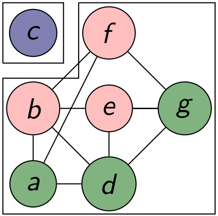
Contracting *ad* and *g*





$\varphi_{ad\,bef\,g}$



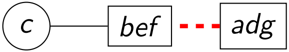


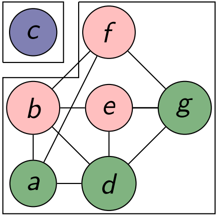
$\varphi_{adg\,bef} =$

$\rho_{\bullet \mapsto \bullet}$

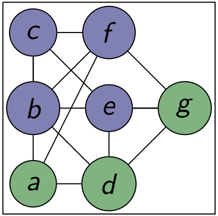
$\varphi_{ad\,bef\,g}$

Contracting c and bef





$\varphi_{adg\,bef}$
φ_{c}



$$\varphi_{adg\,bcef} = \rho_{\bullet \mapsto \bullet}\, \eta_{\bullet,\bullet}\, (\varphi_{adg\,bef} \oplus \varphi_{c})$$

Consequence

Contraction of comp.width $k \implies (k+1)$-expression

$$\boxed{\mathrm{cw}(G) \leq \mathrm{ctww}(G) + 1}$$

Tight for cographs ($\mathrm{cw} = 2$, $\mathrm{ctww} = 1$)

Contents

1 #*H*-COLORING

2 First bound

3 Second bound

4 Complexity consequences

Functional equivalence

We already know:

$$\mathrm{ctww}(G) \leq 2^{\mathrm{ctww}(G)+1}$$

Second contribution: Improved bound on component twin-width

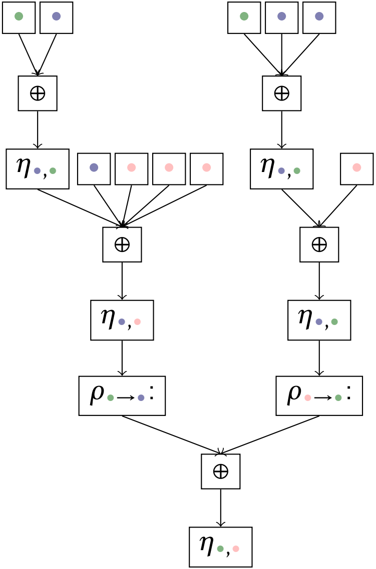
I will prove

$$\text{ctww}(G) \leq 2\text{cw}(G) - 1 \text{ and } \text{ctww}(G) \leq \text{linearcw}(G)$$

Take a (linear) k-expression

Build a contraction sequence of G, where every red-component has size $\leq 2k - 1$ (resp. $\leq k$).

k-expression



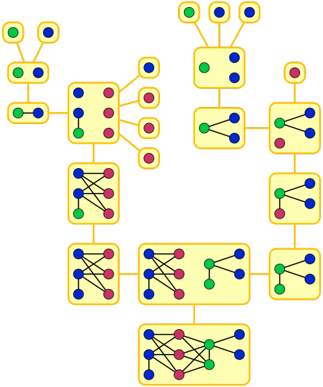
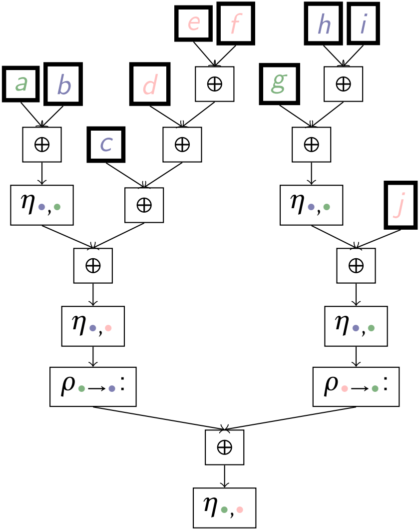


Figure: *k*-expression tree structure

Severe abuse of notation: ⊕ must be binary

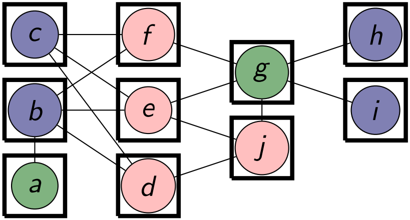
Intuition: contract same colors in ⊕



Build larger and larger "parks" following the k-expressions.

Contract similar colors:

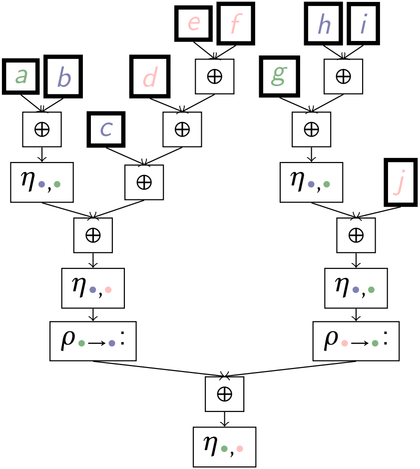
- Parks size $\leq 2k$
- No red-edges crossing parks

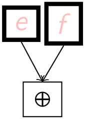


Initial parks are single vertices

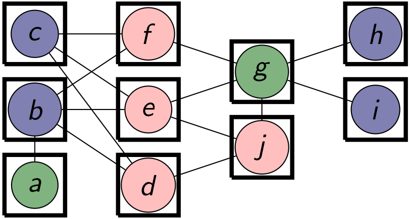
Free contraction of twins

Here, d, e and f (as well as h and i) are introduced together with the same labels: they are twins

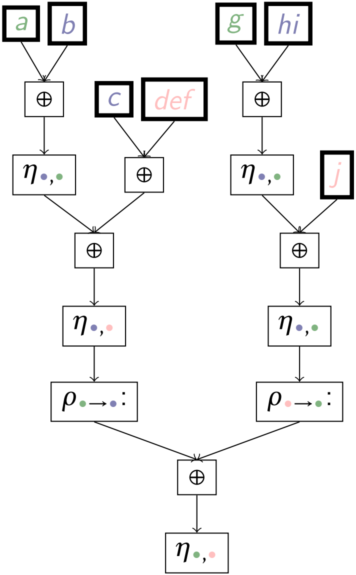




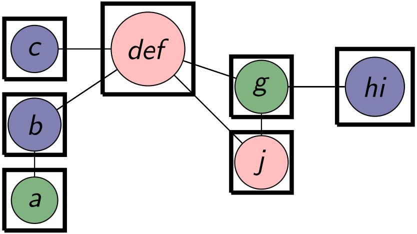
becomes



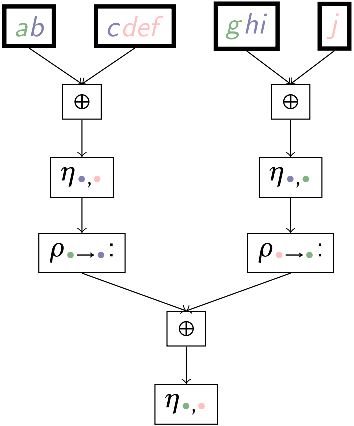
Contracting similar colors in a park



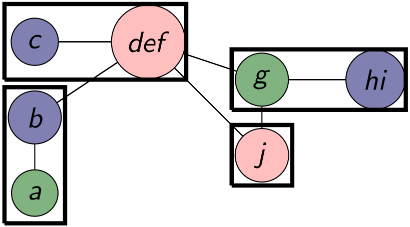
- Merge the parks of a and b, of c and def and of g and hi.
- Collapse the k-expression
- No 2 different colors in the same park: no contraction.



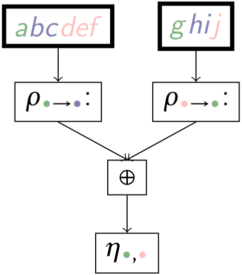
Joining different colors in a park



- Merge the parks of $\{a, b\}$ and $\{c, def\}$ and of $\{g, hi\}$ and $\{j\}$.
- b and c are both blue in the same park: contract them.



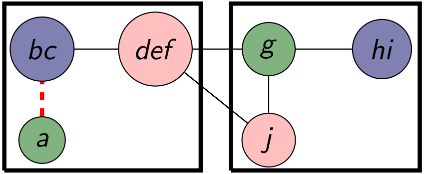
Main argument: no red-edge crossing parks



b and c will have eternally the same label

b and c have exactly the same neighbors in $\{g, h, i, j\}$: no red-edge crossing parks

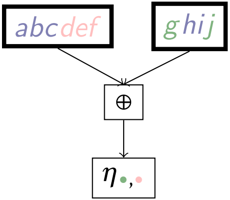
b and c have been contracted.



a will become blue: contract a and bc

j will become green: contract j and g

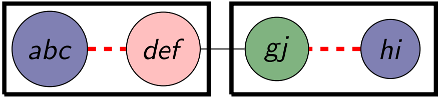
Renaming in a park: no red-edge crossing parks



g and j will have eternally the same label

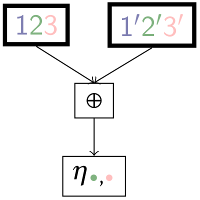
g and j have exactly the same neighbors in $\{a, b, c, d, e, f\}$

a and bc have been contracted.



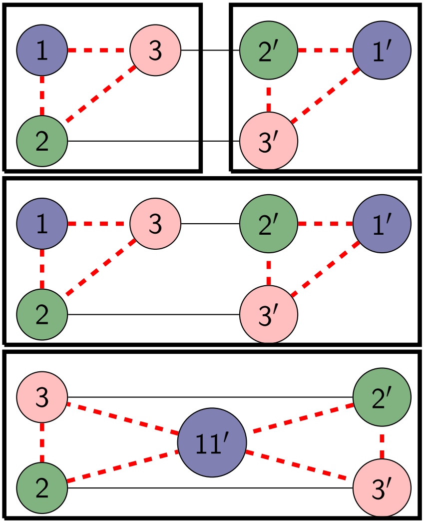
Next step: merge parks.
One park left: Ends.
Finish the contraction sequence randomly

Largest possible red-component



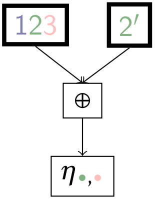
k labels on both side.
Red-comp of size k on both side.

Peak: Red-comp of size $2k-1$
Then, contract by color until k vertices left in the park
Then, procede to the next ⊕



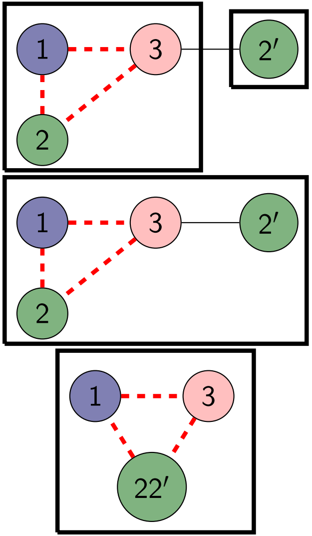
Case of a linear k-expression

Linear k-expression: $G_1 \oplus G_2$ is used $\implies$ G_2 has one vertex



k labels on one side.
1 vertex (so 1 label) on the otherside

Peak: Red-comp of size k



Consequence

(Linear) k-expression $\implies$ contraction sequence with every red-comp having size $\leq 2k-1$ (resp. k)

$$\boxed{\text{ctww(G)} \leq 2\text{cw(G)} - 1 \textbf{ and } \text{ctww(G)} \leq \text{linearcw(G)}}$$

$$\text{tww}(G) \leq 2\text{cw}(G) - 2$$

Tight ?

Contents

1 #*H*-COLORING

2 First bound

3 Second bound

4 Complexity consequences

Parameterized complexity

Use the first bound: $\text{cw} \leq \text{ctww} + 1$

$$(2^{|V_H|} - 2)^{\text{cw}(G)} \text{ VS } (2^{|V_H|} - 1)^{\text{ctww}(G)}$$

Parameterized complexity

Use the first bound: $\mathrm{cw} \leq \mathrm{ctww} + 1$

$$(2^{|V_H|} - 2)^{\mathrm{cw}(G)} \text{ VS } (2^{|V_H|} - 1)^{\mathrm{ctww}(G)}$$

Clique-width approach wins... for the moment (very naive) !

Fine grained complexity

Use the second bound: $\text{ctww} \leq 2\text{cw} - 1$ and $\text{ctww} \leq \text{linearcw}$

$(\text{ctww}(H)+2)^{|V_G|}$ VS $(2\text{cw}(H)+1)^{|V_G|}$ and $(\text{linearcw}(H)+2)^{|V_G|}$

Fine grained complexity

Use the second bound: $\text{ctww} \leq 2\text{cw} - 1$ and $\text{ctww} \leq \text{linearcw}$

$(\text{ctww}(H) + 2)^{|V_G|}$ VS $(2\text{cw}(H) + 1)^{|V_G|}$ and $(\text{linearcw}(H) + 2)^{|V_G|}$

Component twin-width approach wins without effort

References

- Édouard Bonnet, Eun Jung Kim, Amadeus Reinald, and Stéphan Thomassé, *Twin-width vi: the lens of contraction sequences*, SODA-2022, SIAM, 2022, pp. 1036–1056.
- Édouard Bonnet, Eun Jung Kim, Stéphan Thomassé, and Rémi Watrigant, *Twin-width I: tractable FO model checking*, FOCS-2020, IEEE, 2020.
- Michael Lampis, *Finer tight bounds for coloring on clique-width*, SIAM Journal on Discrete Mathematics **34** (2020), no. 3, 1538–1558.
- Magnus Wahlström, *New plain-exponential time classes for graph homomorphism*, Theory of Computing Systems **49** (2011), no. 2, 273–282.

The End

Thank you for your attention !

Questions ?