

Status of Open Problems in the Thesis “Bounded Width Graph Classes in Parameterized Algorithms”

Lars Jaffke¹

¹University of Bergen, Norway
lars.jaffke@uib.no

September 19, 2022

Abstract

In this document I keep track of the status of the open problems I mentioned in my thesis. If you are aware of any results I have missed or solved one of the problems, I would appreciate it if you got in touch with me.

1 The List

#	Description	Status	Comments
3.1	Is there an algorithm for TREE DECOMPOSITION running in time $2^{o(k^3)} \cdot n^{O(1)}$?	open	
3.2	Is there some function $f: \mathbb{N} \rightarrow \mathbb{N}$ and an algorithm that given a graph G and an integer k , either decides that $\text{mimw}(G) > k$, or outputs a branch decomposition of G of mim-width at most $f(k)$, and runs in XP time parameterized by k ?	open	
3.3	Is there some constant c and an algorithm that given a graph G , either decides that $\text{mimw}(G) > 1$, or outputs a branch decomposition of G of mim-width at most c , and runs in polynomial time?	open	
3.4	Is there some fragment of MSO/First Order Logic of graphs, containing a problem that is not locally checkable, and whose corresponding MODEL CHECKING problem is solvable in time $n^{f(\phi , \text{mimw})}$, if a branch decomposition of mim-width mimw of the input graph is provided?	solved	A&C DN (B)
4.1	For constant $q \geq 8$, is the clique-width of $(q, q - 2)$ -graphs bounded by a constant or not?	open	
4.2	Is there a polynomial-time algorithm for MAXIMUM INDEPENDENT SET when the input graph is given together with one of its branch decompositions of sim-width 1?	open	

9.1	Characterize the graph class LINEAR MIM-WIDTH 1 .	open	
9.2	Are all connected, acyclic (co-) (σ, ρ) -problems parameterized by the mim-width of a given (linear) branch decomposition of the input graph W[1]-hard ?	open	(C)
9.3	Would an $n^{o(w)}$ -time algorithm for some (σ, ρ) -problem parameterized by the mim-width w of a given (linear) branch decomposition of the input graph refute ETH ?	open	
10.1	Is FALL COLORING parameterized by clique-width W[1]-hard ?	solved	Yes. (A)
10.2	Would an $n^{2^{o(w)}}$ -time algorithm for FALL COLORING , where w denotes the clique-width of the input graph, refute ETH ?	solved	Yes. (A)
10.3	Is CLIQUE COLORING parameterized by clique-width W[1]-hard ?	open	
10.4	Is there a computable function $g: \mathbb{N} \rightarrow \mathbb{N}$ such that each graph of clique-width cw can be clique colored with at most $g(\text{cw})$ many colors?	open	
10.5	Is there an $n^{2^{2^{o(\text{cw})}}}$ -time algorithm for CLIQUE COLORING , where cw denotes the clique-width of the input graph, or would such an algorithm refute ETH ?	open	
10.6	Is there a $2^{2^{2^{o(\text{cw})}}} \cdot n^{\mathcal{O}(1)}$ -time algorithm for 2-CLIQUE COLORING or would such an algorithm refute ETH ?	open	
10.7	Is b - COLORING NP -complete on circular arc graphs, or, more generally, on any graph class of constant mim-width?	solved	NP-c on \bigcup circular arc
10.8	Is b - COLORING parameterized by the mim-width of a given branch decomposition of the input graph plus the number of colors XP ?	solved	Yes. (B)
10.9	For which function $f: \mathbb{N} \rightarrow \mathbb{N}$ does it hold that for all fixed $k \geq 3$, b - COLORING on graphs of clique-width cw can be solved in time $\mathcal{O}^*(f(k)^{\text{cw}})$ while an algorithm running in time $\mathcal{O}^*((f(k) - \epsilon)^{\text{cw}})$, for any $\epsilon > 0$, would refute SETH ? What about FALL COLORING , or CLIQUE COLORING ?	open	
10.10 (I)	Is b - COLORING parameterized by the treewidth of the input graph W[1]-hard ?	solved	Yes. (D)
10.10 (II)	What is the fastest algorithm for b - COLORING parameterized by treewidth under the ETH ?	open	

(A). J., Lima, Lokshtanov [STACS 2021].

(B). Bergougnoux, Dreier, J. [arXiv:2202.13335].

(C). Bakkane, J. [IPEC 2022]: Dichotomies for minimization problems, and for maximization problems when σ and ρ are finite.

(D). J., Lima, Sharma [arXiv:2209.07772]: b -COLORING parameterized by pathwidth is XNLP-complete, which implies it is $W[t]$ -hard for all t .