$\mathcal{H}$-Clique-Width and a Hereditary Analogue of Product Structure

Petr Hliněný
Masaryk University, Brno, Czech Republic

Jan Jedelský
Masaryk University, Brno, Czech Republic

Abstract

We introduce a novel generalization of the notion of clique-width which aims to bridge the gap between classical hereditary width measures and the recently introduced graph product structure theory. Bounding the new $\mathcal{H}$-clique-width, in the special case of $\mathcal{H}$ being the class of paths, is equivalent to admitting a hereditary (i.e., induced) product structure of a path times a graph of bounded clique-width. Furthermore, every graph admitting the usual (non-induced) product structure of a path times a graph of bounded tree-width, has bounded $\mathcal{H}$-clique-width and, as a consequence, it admits the usual product structure in an induced way. We prove further basic properties of $\mathcal{H}$-clique-width in general.

2012 ACM Subject Classification Mathematics of computing → Graph theory

Keywords and phrases product structure, hereditary class, clique-width, twin-width

Digital Object Identifier 10.4230/LIPIcs.MFCS.2024.61

Related Version *Full Preprint*: `https://arxiv.org/abs/2403.16789` [17]

Acknowledgements We acknowledge helpful discussions with Jakub Gajarský on the questions and problems posed in Section 5, and especially on the question of $\mathcal{H}$-clique-width possibly being complement-closed.

1 Introduction

A prominent structural result by Dujmović, Joret, Micek, Morin, Ueckerdt and Wood [11], known as the *Planar product structure theorem*, claims that every planar graph can be found as a subgraph in the strong product ($\boxtimes$) of a path and a graph of small tree-width. We refer to Section 2 and Theorem 2.3 for the definitions and details.

The original motivation for this rather recent Product structure theorem was to bound the queue number of planar graphs, but the theorem has quickly found interesting applications and follow-up results, among which we may mention [1, 9, 10, 12, 13, 23]. Namely, the product structure theory has been used to study non-repetitive colourings [10], to design short labelling schemes [1, 9], or to bound the twin-width of planar graphs [4, 16, 19].

The basic goal of the product structure theory can be seen in studying graph classes which admit such product structure, that is, they can be constructed as subgraphs of the strong product of a path and a graph of small tree-width. Within this setting, there are two major restrictions; first that the containment (subgraph) relation is not induced, and second that this kind of a superstructure can exist only for sparse graph classes.

We aim to give a different perspective on the product structure (Definition 2.1) addressing both mentioned issues, that is, getting graphs which admit the traditional product structure as *induced* subgraphs in such strong product, and allowing also dense graphs to occur.

Our alternative view is two-sided and is closely related to another classical structural notion in graph theory – the clique-width measure. On one hand, any graph admitting the traditional product structure can be obtained as an induced subgraph of the strong

49th International Symposium on Mathematical Foundations of Computer Science (MFCS 2024).
Editors: Rastislav Královič and Antonín Kučera; Article No. 61; pp. 61:1–61:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

product of a path and a suitable graph of bounded clique-width and even bounded tree-width (Theorem 4.6). On the other hand, a graph G admits the induced product structure with bounded clique-width (of the relevant factor), if and only if G has bounded $\mathcal{H}$-clique-width where $\mathcal{H}$ is the class of reflexive paths (Theorem 4.1). This dual nature of our view is another promising enhancement. Moreover, we believe this view can contribute to finding potential algorithmic applications of product structure theory.

The wide scope of our definition suggests to study $\mathcal{H}$-clique-width for other graph families $\mathcal{H}$ in addition to paths, too. For instance, in relation to the aforementioned product-structure works, one may consider $\mathcal{H}$ to be the class of the graphs $P_n \boxtimes K_k$ or of $P_n \boxtimes P_m$.

We study and characterize relations of $\mathcal{H}$-clique-width to ordinary clique-width (Theorem 3.1), to local clique-width (Theorem 3.4), and in parts to twin-width (Corollary 4.3). The full potential of this new concept when $\mathcal{H}$ is a family of specific graphs other than paths is yet to be explored, especially in the case of $\mathcal{H}$ formed by suitable dense graphs. We conclude with a number of open questions related to the new concept (Section 5).

2 Preliminaries

We consider finite simple graphs, i.e., graphs without parallel edges or loops, but in one specific context (Definition 2.1) we allow graphs with optional (self-)loops, thereafter called loop graphs. Precisely, a *loop graph* is a multigraph allowing loops (at most one per vertex), but not allowing parallel edges. In the context of loop graphs, we specially call a graph G a *reflexive (loop) graph* if every vertex of G has a loop. We naturally use terms *reflexive path*, *reflexive clique*, and *reflexive independent set* to denote ordinary paths, cliques, and independent sets, respectively, with loops added to all vertices. We write $G_1 \subseteq_i G_2$ to say that G_1 is an induced subgraph of G_2. Note that if $G_1 \subseteq_i G_2$ for loop graphs, then possible loops of G_2 on vertices of G_1 are also inherited.

A graph G is a *matching* if G is simple and all vertex degrees in G are 1. A graph $G \subseteq K_{n,n}$ is an *antimatching* if G is obtained from $K_{n,n}$ by removing a matching of n edges. A graph G is a *half-graph* if G is a bipartite simple graph with the bipartition $\{u_1, \ldots, u_n\}$ and $\{v_1, \ldots, v_n\}$, such that $u_i v_j \in E(G)$ if and only if $i \leq j$.

Width measures. As a traditional structural decomposition, a *tree decomposition* of a graph G is a tuple $(T, \mathcal{X})$ where T is a tree, and $\mathcal{X} = \{X_t : t \in V(T)\}$ where $X_t \subseteq V(G)$ is a collection of *bags* which satisfy the following: (i) $\bigcup_{t \in V(T)} X_t = V(G)$, (ii) for every vertex $v \in V(G)$, the set of the nodes $t \in V(T)$ such that $v \in X_t$ forms a subtree in T, and (iii) for every edge $uv \in E(G)$, there exists $t \in V(T)$ such that $\{u, v\} \subseteq X_t$. The *tree-width* of G is the minimum of $\max_{t \in V(T)} |X_t| - 1$ over all tree decompositions of G.

Our work is closely related to another measure, which is a "dense counterpart" of tree-width. The *clique-width* of a graph G is the minimum integer ℓ such that G (irrespective of labelling) is the value of an algebraic *ℓ-expression* defined by the following operations:

- create a new vertex of label (colour) i for some $i \in \{1, \ldots, \ell\}$;
- take the disjoint union of two labelled graphs;
- for $1 \leq i \neq j \leq \ell$, add all (missing) edges between a vertex of label i and a vertex of label j;
- for $1 \leq i \neq j \leq \ell$, recolour each vertex of label i to have label j.

In the same direction, let the *local clique-width* of a graph G be the integer function λ defined as follows; for an integer distance $r \geq 1$, $\lambda(r)$ is the maximum clique-width of the r-neighbourhood of a vertex v in G, over all $v \in V(G)$. We say that a graph class $\mathcal{G}$ is of *bounded local clique-width* if there exists an integer function upper-bounding the local clique-width of every member of $\mathcal{G}$. For instance, the class of grids is of bounded local clique-width, but of unbounded clique-width.

The last measure we mention, twin-width, was introduced a few years ago by Bonnet et al. in [3]. A *trigraph* is a simple graph G in which some edges are marked as *red*, and with respect to the red edges only, we naturally speak about *red neighbours* and *red degree* in G. For a pair of (possibly not adjacent) vertices $x_1, x_2 \in V(G)$, we define a contraction of the pair x_1, x_2 as the operation creating a trigraph G' which is the same as G except that x_1, x_2 are replaced with a new vertex x_0 (said to *stem from* x_1, x_2) such that:

- the (full) neighbourhood of x_0 in G' (i.e., including the red neighbours), denoted by $N_{G'}(x_0)$, equals the union of the neighbourhoods $N_G(x_1)$ of x_1 and $N_G(x_2)$ of x_2 in G except x_1, x_2 themselves, that is, $N_{G'}(x_0) = (N_G(x_1) \cup N_G(x_2)) \setminus \{x_1, x_2\}$, and
- the red neighbours of x_0, denoted here by $N^r_{G'}(x_0)$, inherit all red neighbours of x_1 and of x_2 and add those in $N_G(x_1)\Delta N_G(x_2)$, that is, $N^r_{G'}(x_0) = \big(N^r_G(x_1) \cup N^r_G(x_2) \cup (N_G(x_1)\Delta N_G(x_2))\big) \setminus \{x_1, x_2\}$, where Δ denotes the symmetric set difference.

A *contraction sequence* of a trigraph G is a sequence of successive contractions turning G into a single vertex, and its width d is the maximum red degree of any vertex in any trigraph of the sequence. The *twin-width* of a trigraph G is the minimum width over all possible contraction sequences of G. The twin-width of a multigraph or a loop graph is defined as the twin-width of its simplification.

Introducing $\mathcal{H}$-clique-width. Our main contribution builds on the following new concept.

▶ **Definition 2.1** ($\mathcal{H}$-clique-width)**.** *Let $\mathcal{H}$ be a family of loop graphs, and $\ell > 0$ be an integer. Consider* labels *of the form (i, v) where $i \in \{1, \dots, \ell\}$ and $v \in V(H)$ for some (fixed) $H \in \mathcal{H}$.*

a) *For $H \in \mathcal{H}$, let an (H, ℓ)*-expression *be an algebraic expression using the following four operations on vertex-labelled graphs:*
- *creating a new vertex with single label (i, v) for some $i \in \{1, \dots, \ell\}$ and $v \in V(H)$;*
- *taking the disjoint union of two labelled graphs;*
- *for $1 \le i \ne j \le \ell$,* adding edges between i and j*, which means to add all edges between the vertices of label (i, v) and the vertices of label (j, w) over all pairs $(v, w) \in V(H)^2$ such that $vw \in E(H)$ (including the case of a single vertex $v = w$ with a loop, which will often be assumed to exist for the graphs H); and*
- *for $1 \le i \ne j \le \ell$,* recolouring i to j*, which means to relabel all vertices with label (i, v) where $v \in V(H)$ to label (j, v).*

b) *The* $\mathcal{H}$-clique-width $\mathcal{H}\text{-cw}(G)$ *of a simple graph G is defined as the smallest integer ℓ such that (some labelling of) G is the value of an (H, ℓ)-expression for some $H \in \mathcal{H}$. If it is not possible to build G this way, then let $\mathcal{H}\text{-cw}(G) = \infty$.*

Given an (H, ℓ)-expression of value (a labelled graph) G, we use the following terminology; the graph H is the parameter *of the expression, and when referring to a label (i, v) of $x \in V(G)$, the integer i is the* colour *and v the* parameter vertex *of x.*

Observe that, throughout an (H, ℓ)-expression φ valued G, the colours of vertices of G may arbitrarily change by the recolouring operations, but the parameter vertex of every $x \in V(G)$ stays the same (is uniquely determined for x) in φ.

It is obvious that $\mathcal{H}$-clique-width (similarly to ordinary clique-width) is monotone under taking induced subgraphs. On the other hand, it is not apriori clear whether $\mathcal{H}$-clique-width is (at least functionally) closed under taking the complement of a graph; we will address this interesting issue in the concluding section. Another remark concerns the family $\mathcal{H}$ which should be generally treated as an infinite class of (finite) loop graphs, due to $\mathcal{H}$-clique-width being asymptotically the same as ordinary clique-width in the case of finite $\mathcal{H}$ – see Theorem 3.1.

To further briefly illustrate Definition 2.1, we add few more easy observations:

▷ Claim 2.2.

a) If $\mathcal{H} = \{K_1\}$, then $\mathcal{H}$-cw$(G) < \infty$ if and only if G has no edges. If $\mathcal{H} = \{K_1^\circ\}$, where K_1° stands for a single vertex with a loop, then $\mathcal{H}$-cw$(G) = \mathrm{cw}(G) < \infty$.

b) If $\mathcal{H}$ contains a loop graph with at least one loop, then $\mathcal{H}$-cw$(G) \le \mathrm{cw}(G)$.

c) If $\mathcal{H} = \{H\}$, then $\mathcal{H}$-cw$(H_1) \le 2$ holds for every simple graph H_1 obtained from an induced subgraph of H by removing loops.

d) If $\mathcal{H} = \{H\}$ and H is disconnected, then for every connected simple graph G we have $\mathcal{H}$-cw$(G) = \{H_0\}$-cw(G) for some connected component H_0 of H.

e) If $\mathcal{H} = \{K_2\}$ (no loops), then $\mathcal{H}$-cw$(G) < \infty$ if and only if G is a simple bipartite graph. More generally, for any $\mathcal{H}$, we have $\mathcal{H}$-cw$(G) < \infty$ for a simple graph G, if and only if G has a homomorphism into some $H \in \mathcal{H}$.

f) For any $k \ge 3$ and $\mathcal{H} = \{K_k\}$, it is NP-hard to decide whether $\mathcal{H}$-cw$(G) < \infty$.

g) Let $\mathcal{H}$ be a family containing arbitrarily long reflexive paths. If G is any square grid, then $\mathcal{H}$-cw$(G) \le 5$ (while cw(G) is unbounded in such case).

Proof.

a) There is no edge in K_1, and so Definition 2.1 cannot create an edge in G. On the other hand, (K_1°, ℓ)-expressions in Definition 2.1 exactly coincide with traditional ℓ-expressions of clique-width (replacing every label i with (i, v) where $\{v\} = V(K_1^\circ)$).

b) We pick $v \in V(H)$ for $H \in \mathcal{H}$ such that v has a loop in H. Then, in an ordinary cw(G)-expression for G, we replace every label i with (i, v) to get an $(H, \mathrm{cw}(G))$-expression for G, similarly to part a).

c) We simply make an $(H, 2)$-expression for H_1 as follows; in an arbitrary order $V(H_1) = \{v_1, \ldots, v_n\}$ of the vertices, for $k = 1, \ldots, n$, we add a new vertex labelled $(2, v_k)$, add edges between 1 and 2, and recolour 2 to 1. This creates exactly the non-loop edges of H_1.

d) This follows from the facts that the recolouring operation of Definition 2.1 does not allow to change the initially assigned parameter vertex of H, and hence every edge of G created within an (H, ℓ)-expression has a preimage edge in H. So, an expression creating connected G may only use parameter vertices of a connected component of H.

e) Considering the previous argument turned around, every edge created within an (H, ℓ)-expression has a unique homomorphic image in H (possibly a loop). In the opposite direction, for a homomorphism $h : G \to H \in \mathcal{H}$, we make an $(H, |V(G)|)$-expression starting with the disjoint union of vertices labelled $(i_x, h(x))$ for all $x \in V(G)$ where $i_x \ne i_y$ for $x \ne y$, and then simply add the edges of G one by one using the colours (i.e., i_x).

f) By e), we have $\mathcal{H}$-cw$(G) < \infty$ if and only if G is k-colourable.

g) Let G be an $a \times b$ grid, i.e., $|V(G)| = ab$. We take $H \in \mathcal{H}$ such that $|V(H)| \ge b$, choose a consecutive subpath on $\{v_1, \ldots, v_b\} \subseteq V(H)$, and make an $(H, 5)$-expression valued G as follows. As in c), we define an $(H, 3)$-(sub)expression creating a "vertical" copy P_1 of the path on b vertices, but now using three colours such that the resulting labels of P_1 are with alternating colours 2 and 3, precisely as $(2, v_1)$, $(3, v_2)$, $(2, v_3)$, $(3, v_4), \ldots$. We likewise create a copy P_2 of the same path with alternating colours in labels $(4, v_1)$, $(5, v_2)$, $(4, v_3)$, $(5, v_4), \ldots$. Then we make a disjoint union and add edges between colours $2, 4$ and between $3, 5$ – this creates precisely the "horizontal" edges between the labels $(2, v_i)$ and $(4, v_i)$, and between $(3, v_{i+1})$ and $(5, v_{i+1})$, for $i = 1, 3, \ldots$. In a subsequent round, we recolour colours $2, 3$ to 1 (this concerns only P_1), and continue the same process with adding a path P_3 with alternating colours again 2 and 3, and adding the "horizontal" edges. After $a - 1$ rounds, we build the desired $a \times b$ square grid G. ◁

Product structure theory. The *strong product* $G_1 \boxtimes G_2$ of two simple graphs is the graph G on the vertex set $V(G) := V(G_1) \times V(G_2)$ such that, for any $[u, u'], [v, v'] \in V(G)$, we have $\{[u, u'], [v, v']\} \in E(G)$ if and only if ($uv \in E(G_1)$ and $u'v' \in E(G_2)$) or ($u = v$ and $u'v' \in E(G_2)$) or ($uv \in E(G_1)$ and $u' = v'$).

For an illustration, the strong product $P \boxtimes Q$ of two paths P, Q is the square grid with diagonals. It may be interesting to observe that, in the context of loops graphs, if both G_1 and G_2 are reflexive, then the definition of the strong product $G_1 \boxtimes G_2$ could be shortened as "$uv \in E(G_1)$ and $u'v' \in E(G_2)$", and the result would be the same except that all vertices would have loops.

Origins of graph product structure theory go back to the mentioned seminal paper [11]:

▶ **Theorem 2.3** ([11], improved in [23])**.** *Every planar graph is a subgraph of the strong product $P \boxtimes M$ where P is a path and M is a planar graph of tree-width at most 6.*

There exist alternative refined formulations of Theorem 2.3, such as using the strong product $P \boxtimes K_3 \boxtimes M$ where M is now of tree-width at most 3 which is of importance in some applications (such as in refining the upper bound on the queue number of planar graphs).

Our main goal is to refine, using Definition 2.1, the statement of Theorem 2.3 with the induced subgraph relation in Theorem 4.6; admittedly, at the cost of worse absolute constants.

3 Properties of $\mathcal{H}$-Clique-Width

We first characterize the asymptotic difference between the ordinary clique-width and the $\mathcal{H}$-clique-width for families of loop graphs $\mathcal{H}$.

We recall the concept of neighbourhood diversity by Lampis [20]. Two vertices x, y of a simple graph G are of the same *neighbourhood type* if and only if they have the same set of neighbours in $V(G) \setminus \{x, y\}$. We shall use an adjusted version of this concept, suitable for our loop graphs; Two vertices x, y of a simple loop graph G are of the same *total neighbourhood type*, if and only if they have the same set of neighbours in $V(G)$ when x counts as a neighbour of x if there is a loop on x (and likewise with y). A loop graph G is of *total neighbourhood diversity* at most d if $V(G)$ can be partitioned into d parts such that every pair in the same part have the same total neighbourhood type.

The slight, but very important in our context, difference of these two notions in presence of loops can be observed, e.g., on: loopless cliques K_n (neighbourhood diversity 1 and total neighbourhood diversity n) vs. reflexive cliques K_n° (total neighbourhood diversity 1), or loopless stars $K_{1,n}$ (both neighbourhood diversities equal 2) vs. reflexive stars $K_{1,n}^\circ$ (total neighbourhood diversity n).

A loop graph class $\mathcal{G}$ is of *component-bounded* total neighbourhood diversity if there exists an integer d such that each connected component of every graph of $\mathcal{G}$ is of total neighbourhood diversity at most d.

▶ **Theorem 3.1.** *Let $\mathcal{H}$ be a family of loop graphs. There exists a function f such that, $\mathrm{cw}(G) \le f(\mathcal{H}\text{-}\mathrm{cw}(G))$ holds for all simple graphs G, if and only if $\mathcal{H}$ is of component-bounded total neighbourhood diversity.*

Proof. In the "$\Leftarrow$" direction, we may assume G is connected (we will later take the maximum over connected components). By Claim 2.2 d), $\mathcal{H}$-cw$(G) = \{H_0\}$-cw$(G) = \ell$ for a connected component H_0 of some $H \in \mathcal{H}$. The total neighbourhood diversity of H_0 is at most some constant d, by the theorem assumption. Then, in an (H_0, ℓ)-expression for G, we may equivalently replace the parameter vertices of H_0 by d new colours, giving a $d\ell$-expression for G. So, cw$(G) \le d \cdot \mathcal{H}$-cw$(G)$.

A proof of the "$\Rightarrow$" direction is based on the following natural technical claim:

$\triangleright$ Claim 3.2 (Ding et al. [7, Corollary 2.4]). For every k there exists m such that the following holds. If F is a bipartite connected simple graph with the bipartition $V(F) = A \cup B$, $|A| \ge m$ and the vertices of A have pairwise different neighbourhood types (in B), then F contains an induced subgraph isomorphic to one of the following graphs on $2k$ vertices: a matching, an antimatching, or a half-graph.

Having Claim 3.2 at hand, we continue as follows. Assume that $\mathcal{H}$ is *not* of component-bounded total neighbourhood diversity. Let $H \in \mathcal{H}$ (or a component thereof) be a connected loop graph of total neighbourhood diversity $\ge c_1$, and $C \subseteq V(H)$ be vertices representing these c_1 total neighbourhood types. By Ramsey theorem, for sufficiently large c_1 we find a subset $C_1 \subseteq C$, $|C_1| = 2c_2 - 1$, such that C_1 induces a clique or an independent set in G, and then we can select $C_2 \subseteq C_1$, $|C_2| = c_2$ such that either all vertices of C_2 have loops, or none has. We have got one of the two possibilities:

- C_2 is a reflexive independent set or a loopless clique in G.
- Or, all vertices of C_2 have the same total neighbourhood type in C_2 (empty or full C_2), and so they have pairwise different neighbourhood types in $D := V(G) \setminus C_2$. Consequently, we may apply Claim 3.2 to the bipartite subgraph "between" C_2 and D.

Regarding the second point, in more detail, we say that a bipartite graph F_1 with a fixed bipartition $V(F_1) = A_1 \cup B_1$ is a *bi-induced subgraph* of a graph H, if $F_1 \subseteq H$ such that every edge of H with one end in A_1 and the other end in B_1 belongs to F_1. Claim 3.2 hence implies that one of the three claimed subgraphs is bi-induced in H.

Altogether, for every k and sufficiently large c_1 depending on k, we have connected $H \in \mathcal{H}$ containing one of the five said substructures; an induced reflexive independent set or an induced loopless clique on k vertices, or a bi-induced matching, a bi-induced antimatching, or a bi-induced half-graph on $2k$ vertices. In each of these five cases, we can construct a "grid-like" graph of bounded $\mathcal{H}$-clique-width whose ordinary clique-width grows linearly with k. This is provided by subsequent Lemma 3.3, in which one can easily check that its assumptions cover all five cases of $H \in \mathcal{H}$ listed in this proof. ◀

▶ **Lemma 3.3.** *Let $k \ge 3$ be an integer, and H_1 be a loop graph satisfying the following:*

- *H_1 is connected.*
- *There exist sets $A, B \subseteq V(H_1)$, $|A| = |B| = k$, such that either $A = B$, or $A \cap B = \emptyset$.*
- *We can write $A = \{u_1, \dots, u_k\}$ and $B = \{u'_1, \dots, u'_k\}$ such that, for some of the following three conditions on integers $C(i,j) \in \{$'$i < j$','$i = j$','$i \neq j$'$\}$ we have; for all $(i,j) \in \{1, \dots, k\}^2$, $\{u_i, u'_j\} \in E(H_1)$ if and only if $C(i,j)$ is false. (Note that, if $A = B$, we assume $u_i = u'_i$ and deal also with loops.)*

Then, there exists a constant ℓ_0 independent of k such that the class of graphs of $\{H_1\}$-clique-width at most ℓ_0 has ordinary clique-width $\Omega(k)$.

Proof. We construct, via an $(H_1, \mathcal{O}(1))$-expression, a graph G_k of ordinary clique-width $\Omega(k)$ as follows.

Similarly as in Claim 2.2 c), we create a loopless copy G'_1 of the graph H_1, such that every vertex $x \in V(G'_1)$ which is a copy of a vertex $v \in B$ has the label with colour 2 and parameter vertex v and, if $A \neq B$, every vertex $x \in V(G'_1)$ which is a copy of $w \in A$ has the label with colour 1 and parameter vertex w. Vertices of $V(G'_i)$ that are not copies of $A \cup B$ have labels with colour 0 and the respective parameter vertex from $V(H_1) \setminus (A \cup B)$.

We set $G_1 := G'_1$, and for $a = 2, \dots, k$ we do:

- We, likewise, create a loopless copy G'_a of H_1, now with colour 3 in the labels of the copy of A in $V(G'_a)$ and, if $A \neq B$, with colour 4 in the labels of the copy of B in $V(G'_a)$. The labels of $V(G'_a)$ besides the copies of $A \cup B$ are again with colour 0.
- Then we make a disjoint union $G_a := G_{a-1} \dot\cup G'_a$, and add edges between colours 2 and 3.
- If $A = B$, we recolour 2 to 1 and 3 to 2. If $A \neq B$, we recolour 2 and 3 to 1 and 4 to 2.

Altogether, the graph G_k has $k \cdot |V(H_1)|$ vertices, k disjoint copies G'_a of H_1, and every copy G'_a has k vertices which are, in a well-defined way – cf. condition $C(i, j)$, adjacent to corresponding k vertices of the subsequent copy G'_{a+1} (if $a < k$). There are no other edges in G_k. For clarity (and in resemblance to Theorem 4.1), we imagine the copy G'_a of H_1 as "column a" of G_k, and the set of the copies of u_i and u'_i of H_1 as "row i" of G_k. Possible remaining vertices of G'_a (those of colour 0 in their label) are not part of any row, as they do not participate in inter-column edges of G_k. Observe that, if $A \neq B$, the adjacency pattern occurring between columns a and $a + 1$ is exactly the same as the edges between B and A in H_1, and so the same as the "mirrored" adjacency pattern between the copies of A and of B within column a (or $a + 1$).

Now, assume we have an (ordinary) ℓ-expression φ valued G_k for some integer ℓ. We apply an argument which is folklore in this area. There must exist a subexpression φ_1 of φ making a subset of vertices $X \subseteq V(G_k)$ (it is irrelevant which of the edges of $G_k[X]$ this φ_1 makes), such that $\frac{1}{3}|V(G_k)| \leq |X| \leq \frac{2}{3}|V(G_k)|$. Let $\bar{X} = V(G_k) \setminus X$.

Consider any $1 \leq a < k$; then the columns a and $a + 1$ differ with respect to X in at most 3ℓ rows ($\leq \ell$ if $A = B$); meaning that in $\leq 3\ell$ rows i we have a situation that a vertex of row i in one of the columns a or $a + 1$ belongs to X, and a vertex of row i in the other column belongs to $\bar{X}$. This follows since we have at most ℓ different colours in φ_1 which can be used to further distinguish different adjacencies, as given by the condition $C(i, j)$ of the lemma, between the columns a and $a + 1$, or within each one of the columns a or $a + 1$ if $A \neq B$.

Likewise, at most ℓ columns are such that they intersect both X and $\bar{X}$. This follows similarly since every column is a copy of connected H_1, and so it needs in φ_1 a special colour for its (at least one) "private" edge from X to $\bar{X}$. The two latter conditions together are in a clear contradiction with $\frac{1}{3}|V(G_k)| \leq |X| \leq \frac{2}{3}|V(G_k)|$ if $\ell \in o(k)$. ◀

Secondly, there is an interesting relation to established concepts in the case of parameter families $\mathcal{H}$ of bounded degrees.

▶ **Theorem 3.4.** *Let $\mathcal{H}$ be a family of loop graphs of maximum degree Δ. Then the class of graphs of $\mathcal{H}$-clique-width at most ℓ is of bounded local clique-width in terms of Δ and ℓ.*

Proof. Let $H \in \mathcal{H}$ and G be a graph that is a value of an (H, ℓ)-expression φ. Choose $x \in V(G)$, and assume a vertex $y \in V(G)$ at distance at most r from x in G. Let $v, w \in V(H)$ be the parameter vertices in φ of x and y, respectively. As argued in Claim 2.2 e), there is a homomorphism $G \to H$ taking a path between x and y into a walk between v and w in H, and so the distance from v to w in H is at most r. Since $\Delta(H) \leq \Delta$, the r-neighbourhood of v in H has at most $(\Delta + 1)^r$ vertices, and hence φ restricted to the r-neighbourhood of x in G uses only at most $(\Delta + 1)^r$ parameter vertices which can be replaced in φ by unique colours.

This way we obtain an (ordinary) $\ell \cdot (\Delta+1)^r$-expression whose value is the r-neighbourhood of x in G. We can thus set $f(r) := \ell \cdot (\Delta+1)^r$ (independently of $H \in \mathcal{H}$ and G) to certify bounded local clique-width of every G such that $\mathcal{H}$-cw$(G) \le \ell$. ◀

A similar structural relation of $\mathcal{H}$-clique-width to the parameter twin-width is stated later in Corollary 4.3, as a consequence of a product-structure-like characterization.

From Theorem 3.4 we, for instance, immediately get tractability of FO model checking, which is FPT for all classes of bounded local clique-width – this well-known fact follows by a combination of the ideas of Frick and Grohe [15] and of Dawar, Grohe and Kreutzer [5]:

▶ **Corollary 3.5.** *For every family $\mathcal{H}$ of loop graphs, the FO model checking problem of a graph G is in FPT when parameterized by the formula, the maximum degree of $\mathcal{H}$ and the $\mathcal{H}$-clique-width of G.* ◀

Furthermore, it may be interesting to ask to which extent Theorem 3.4 can be reversed. This cannot be done straightforwardly since there are families $\mathcal{H}$ of unbounded degrees, such that classes of bounded $\mathcal{H}$-clique-width not only have bounded local clique-width, but even bounded ordinary clique-width. One example is $\mathcal{H}_1$ the class of all reflexive cliques by Theorem 3.1. On the other hand, e.g., for $\mathcal{H}_2$ being the class of all reflexive stars, there are graphs whose $\mathcal{H}_2$-clique-width is bounded by a constant, and they contain arbitrarily large induced grids and a universal vertex adjacent to everything (a construction similar to Claim 2.2 g)). Such graph hence have unbounded local clique-width.

4 Approaching Induced Product Structure

In this section we restrict our attention to families $\mathcal{H}$ formed by *reflexive* loop graphs (i.e., all vertices have loops in the graphs of $\mathcal{H}$), which makes most natural sense with respect to the strong-product structure studied.

▶ **Theorem 4.1.** *Let $\mathcal{H}$ be a family of reflexive loop graphs, and $\mathcal{H}'$ be the family of simple graphs obtained from the graphs of $\mathcal{H}$ by removing all loops. For every integer $\ell \ge 2$ the following holds. A simple graph G is of $\mathcal{H}$-clique-width at most ℓ, if and only if G is isomorphic to an induced subgraph of the strong product $H' \boxtimes M$ where $H' \in \mathcal{H}'$ and M is a simple graph of clique-width at most ℓ.*

Proof. In the "$\Leftarrow$" direction, it is enough to show that $\mathcal{H}$-cw$(G) \le \ell$ for $G := H' \boxtimes M$. Let φ be an ℓ-expression of the graph M, and let H° be obtained from H' by adding loops to all vertices. We are going to transform φ into an (H°, ℓ)-expression as follows. First, for each $x \in V(M)$ we independently construct a copy H'_x of H', using only $2 \le \ell$ colours by Claim 2.2 c). That is, the parameter vertex of every $v_x \in V(H'_x)$ is the preimage $v \in V(H')$ of v_x. Then, at every moment the expression φ introduces a new vertex $y \in V(M)$ of colour i, we take (substitute) the copy H'_y and recolour it to i. The remaining operations (union, recolouring, and adding edges) stay in place in φ, but are now applied according to Definition 2.1.

We claim that the value G of the resulting transformed (H°, ℓ)-expression σ is $H' \boxtimes M$. Indeed, the vertex set is $V(G) = V(H') \times V(M)$, and for each $m \in V(M)$ the subgraph induced on $V(H') \times \{m\}$ is a copy of H'. For any $[v_1, m_1], [v_2, m_2] \in V(G)$ and $m_1 \ne m_2$; if $\{[v_1, m_1], [v_2, m_2]\} \in E(G)$, then $v_1v_2 \in E(H^\circ)$ by Definition 2.1, and $m_1m_2 \in E(M)$ by the definition of σ. Hence $\{[v_1, m_1], [v_2, m_2]\} \in E(H' \boxtimes M)$. Conversely, if $\{[v_1, m_1], [v_2, m_2]\} \in E(H' \boxtimes M)$, then, by the definition of $\boxtimes$, $v_1v_2 \in E(H')$ or $v_1 = v_2$, meaning $v_1v_2 \in E(H^\circ)$, and $m_1m_2 \in E(M)$. So, the edge $\{[v_1, m_1], [v_2, m_2]\}$ has been created by σ.

In the “⇒” direction, let σ be an (H°, ℓ)-expression valued G, for some $H^\circ \in \mathcal{H}$. Let $H' \in \mathcal{H}'$ be the simple graph of H°. We are going to construct an ℓ-expression φ valued M on the vertex $V(M) = V(G)$, such that $G \subseteq_i H' \boxtimes M$. The expression φ simply discards parameter vertices (cf. Definition 2.1) from the labels in σ. Hence, we clearly get $M \supseteq G$. To prove that $G \subseteq_i H' \boxtimes M$, consider any vertices $x \neq y \in V(G)$ labelled (i, v) and (j, w) by σ (for here, indefinite i and j are irrelevant, and v and w are uniquely determined by σ). We claim that the vertices x and y as of G can be represented by $[v, x]$ and $[w, y]$ of the product $H' \boxtimes M$. If $xy \in E(G)$, then $xy \in E(M)$ by previous $M \supseteq G$, and $vw \in E(H^\circ)$ by Definition 2.1. Consequently, $\{[v, x], [w, y]\} \in E(H' \boxtimes M)$ by $\boxtimes$. On the other hand, if $\{[v, x], [w, y]\} \in E(H' \boxtimes M)$, then $vw \in E(H')$ or $v = w$, and so $vw \in E(H^\circ)$. Moreover, $xy \in E(M)$ since $x \neq y$, and so $xy \in E(G)$ since the (original) (H°, ℓ)-expression σ creates the edge xy by Definition 2.1. ◀

Theorem 4.1 can be used also to bound the twin-width of graphs of bounded $\mathcal{H}$-clique-width. To show this, we first prove the following ad-hoc upper bound.

▶ **Proposition 4.2.** *Let P be a reflexive path and G a simple graph. Then the twin-width of G is at most $5 \cdot (\{P\}\text{-cw}(G)) - 2$. Consequently, denoting by $\mathcal{P}^\circ$ the class of all reflexive paths, the twin-width of any simple graph G is at most $5 \cdot (\mathcal{P}^\circ\text{-cw}(G)) - 2$.*

Proof. Let G be the value of a (P, ℓ)-expression φ, where $\ell = \{P\}\text{-cw}(G)$. When constructing a contraction sequence for G, we proceed recursively (bottom-up) along the expression tree of φ; processing only the union and recolouring nodes, and at each node contracting together all vertices of the same label.

Consider a situation at a node with a subexpression φ_0 of φ, where $X_0 \subseteq V(G)$ is the vertex set generated by φ_0, and let $x^0_{(i,v)}$ denote the vertex resulting from the contractions of all vertices of X_0 that are of label (i, v) by φ_0. The core observation is that every vertex of $V(G) \setminus X_0$ has the same adjacency to all vertices forming $x^0_{(i,v)}$ by Definition 2.1, and so the only possible red neighbours of $x^0_{(i,v)}$ in a contraction of G are those that stem from X_0.

The only possible neighbours of $x^0_{(i,v)}$ in the described contraction of the induced subgraph $G[X_0]$ are $x^0_{(j,w)}$ where $j \in \{1, \dots, \ell\}$ and $vw \in E(P)$ – altogether at most $3\ell - 1$ choices of potential red neighbours of $x^0_{(i,v)}$ in the contraction of $G[X_0]$. If a recolouring operation i to j is encountered after the node of φ_0, we simply contract each former $x^0_{(i,v)}$ with $x^0_{(j,v)}$ over all $v \in X_0$, not increasing the previous bound on the red degree.

Consider now a union node making $X_2 := X_0 \dot\cup X_1$, where X_1 has been generated by a sibling subexpression φ_1 of φ, and let $x^1_{(i,v)}$ analogously denote the vertices resulting from contractions of X_1. Let the vertices of P be $V(P) = (v_1, \dots, v_a)$ in the natural order along the path. For $k = 1, \dots, a$, and subsequently for $i = 1, \dots, \ell$, we make $x^2_{(i,v_k)}$ by contracting $x^0_{(i,v_k)}$ with $x^1_{(i,v_k)}$. Considering the corresponding successive contractions of the induced subgraph $G[X_2]$, the only possible red neighbours of $x^2_{(i,v_k)}$ are the ℓ vertices $x^2_{(i',v_{k-1})}$, the up to $2(\ell - 1)$ vertices $x^2_{(j,v_k)}$ for $j < i$, or $x^0_{(j,v_k)}$, $x^1_{(j,v_k)}$ for $j > i$, and the 2ℓ vertices $x^2_{(i'',v_{k+1})}$, $x^2_{(i'',v_{k+1})}$. The maximum possible encountered red degree is thus $\ell + 2(\ell - 1) + 2\ell = 5\ell - 2$. ◀

▶ **Corollary 4.3.** *Let $\mathcal{H}$ be a family of reflexive loop graphs of maximum degree Δ and twin-width at most t. Then the twin-width of any simple graph G is at most $\mathcal{O}(t + \Delta \cdot \mathcal{H}\text{-cw}(G))$.*

Proof. By Theorem 4.1, we have $G \subseteq_i H' \boxtimes M$, where H' is of maximum degree at most Δ and twin-width at most t and M is of clique-width at most $\ell := \mathcal{H}\text{-cw}(G)$. Since twin-width is monotone under taking induced subgraphs, it is enough to bound it for the graph $H' \boxtimes M$.

By Proposition 4.2 applied to P being a single reflexive vertex, and Claim 2.2 a), M is of twin-width at most $k := 5\ell - 2$. Then, by Bonnet et al. [2] (bounding twin-width of a strong product), $H' \boxtimes M$ is of twin-width at most $\max\{t + \Delta,\, k(\Delta + 1) + 2\Delta\} = \mathcal{O}(t + \Delta\ell)$. ◀

Notice that, for constant Δ, the bound $\mathcal{O}(t+\Delta\cdot\mathcal{H}\text{-cw}(G))$ in Corollary 4.3 is asymptotically best possible; a linear dependence on t (the maximum twin-width of $\mathcal{H}$) is necessary due to Claim 2.2 c), and a linear dependence on $\mathcal{H}$-cw(G) is, on the other hand, required already by the subcase of ordinary clique-width. It is not clear whether the linear dependence on Δ in the bound of Corollary 4.3 is really necessary, however, the next construction shows that the bound has to depend on Δ, the maximum degree of $\mathcal{H}$:

▶ **Proposition 4.4.** **(*)** *Let H_n denote the half-graph (cf. Section 2) on $2n$ vertices. Then the twin-width of the graphs $H_n \boxtimes H_n$ grows with n. Furthermore, the class $\{H_n \boxtimes H_n : n \in \mathbb{N}_+\}$ is monadically independent.*

Since Proposition 4.4 was not part of the reviewed MFCS submission, we leave its proof to the full preprint [17].

If $\mathcal{H}$ is the family of reflexive half-graphs (which is of unbouded maximum degree), then $\mathcal{H}$-cw$(H_n \boxtimes H_n) \leq$ cw$(H_n) \leq 3$ using Theorem 4.1 and a trivial 3-expression for H_n, and likewise the twin-width of $\mathcal{H}$ is constant. So, from Proposition 4.4 we get that the bound in Corollary 4.3 must grow with $n = \Delta(H_n)$, too.

4.1 Relations to the traditional product structure

In regard of the Planar product structure theorem, as introduced in Section 2, we are especially interested in $\mathcal{H}$-clique-width for $\mathcal{H} = \mathcal{P}^\circ$ where $\mathcal{P}^\circ$ is the *class of reflexive paths*. We get the following as another immediate consequence of Theorem 4.1:

▶ **Corollary 4.5.** *For every integer $\ell \geq 2$ the following holds. A graph G is isomorphic to an induced subgraph of the strong product $P \boxtimes M$ where P is a path and M is a simple graph of clique-width at most ℓ, if and only if $\mathcal{P}^\circ$-cw$(G) \leq \ell$.* ◀

There is, however, a more direct connection between our concept and the original Planar product structure theorem, which constitutes the main new contribution of the paper:

▶ **Theorem 4.6.** *Assume that a graph G is a subgraph (not necessarily induced) of the strong product $G \subseteq P \boxtimes M$ where P is a path and M is a simple graph of tree-width at most k. Then $\mathcal{P}^\circ$-cw$(G) \leq 6(k + 1) \cdot 8^{k+1}$. Moreover, there exists a graph M_1 of tree-width at most $6(k+1)\cdot 8^{k+1}$ such that G is isomorphic to an* induced *subgraph of the strong product $P \boxtimes M_1$.*

Proof. We start with proving the first part of the statement, that $\mathcal{P}^\circ$-cw$(G) \leq 6(k+1)\cdot 8^{k+1}$. Although, we remark that we could as well jump straight into a proof of the second part, the product $P \boxtimes M_1$, and then refer to Theorem 4.1 to conclude with a bound (albeit weaker) on $\mathcal{P}^\circ$-cw(G). We believe that the presented approach to the proof is more accessible for the readers.

We assume a rooted tree decomposition $(T, \mathcal{X})$ of width k of the graph M, such that every node of T has at most two children. For a node $t \in V(T)$, let $X_t^+ \subseteq V(M)$ denote the union of X_s where s ranges over t and all descendants of t. Let $p(t)$ denote the parent node of t in T, and let $Y_t = X_t^+ \setminus X_{p(t)}$ denote the vertices of M which occur *only* in the bags of t and its descendants. For the root r of T, let specially $Y_r = X_r^+ = V(M)$. Observe that all neighbours of a vertex $m \in Y_t$ in $V(M) \setminus Y_t$ must belong to the set $X_t \setminus Y_t$, by the interpolation property of a tree decomposition. Let $q : V(M) \to \{0, 1, \ldots, k\}$ be a function such that q is injective on each of the bags X_t over $t \in V(T)$ – such q is easily constructed along the tree T in the root-to-leaves order (in fact, q can be seen as a monotone cop search strategy on the decomposition of M).

Analogously to the treatment in the proof of Theorem 4.1, we refer to the vertices of $G \subseteq P \boxtimes M$ as to the pairs $[p, m]$ where $p \in V(P)$ and $m \in V(M)$ in the natural correspondence. When constructing an expression for the graph G, we follow on a high level the tree T; at a node $t \in V(T)$, we will construct precisely the subgraph G^t of G induced on the vertex set $(V(P) \times Y_t) \cap V(G)$. By the previous, all neighbours of $V(G^t)$ in the rest of G belong to the set $W_t := \big(V(P) \times (X_t \setminus Y_t)\big) \cap V(G)$ where $|X_t \setminus Y_t| \leq |X_t| \leq k+1$. It will thus be enough to encode in the colour of each $x \in V(G^t)$ information about which vertices of W_t are actual neighbours of x in G and, moreover, the colours used can be "recycled modulo 3" along the path P. This way we will prove that $\mathcal{P}^\circ$-cw(G) is bounded in terms of k; more precisely, that $\{P^\circ\}$-cw$(G) \leq 6(k+1) \cdot 8^{k+1}$ for P° being the reflexive closure of P.

Let the vertices of P be $V(P) = (p_1, \ldots, p_n)$ in the natural order along the path, and let $t_m \in V(T)$ for $m \in V(M)$ denote the node closest to the root such that $m \in X_{t_m}$ (so, $m \in Y_{t_m}$). In more detail, for each $m \in V(M)$ we create, in a trivial way, a subexpression for the subgraph G_m of G induced by $\big(V(P) \times \{m\}\big) \cap V(G)$ (which is a copy of a subpath of P), such that the labels in G_m are as follows.

For each vertex $x \in V(G_m)$ where $x = [p_i, m]$, we give x the label (c_x, p_i) such that the colour of x is a tuple $c_x = (0, i \bmod 3,\ b_0, b_1, \ldots, b_k,\ d)$ satisfying the following:

- i is the index of p_i, and $b_j \in \{0,1\}^3$ and $d \in \{0, \ldots, k\}$ are prescribed below;
- for every $j \in \{0, 1, \ldots, k\} \setminus q(X_{t_m})$ we let $b_j = (0, 0, 0)$;
- for every $j = q(m')$ where $m' \in X_{t_m}$ (recall that q is injective on each bag, and so m' is unique such), we define $b_j = (b^1, b^2, b^3)$ where
  - $b^1 = 1$ if and only if $\{x, [p_{i-1}, m']\} \in E(G)$,
  - $b^2 = 1$ if and only if $\{x, [p_i, m']\} \in E(G)$, and
  - $b^3 = 1$ if and only if $\{x, [p_{i+1}, m']\} \in E(G)$;
- we set $d = q(m'')$ where $m'' \in V(M)$ is a neighbour of m in M such that all neighbours of m belong to $Y_{t_{m''}}$ (informally, m'' is any topmost w.r.t. T neighbour of m in M).

For an informal explanation in relation the above "sketch of encoding", the colour c_x in the label of x covers all desired information about the neighbours of x in the set W_{t_m} in G, for which adjacencies will created later in the coming expression for G. And again on an informal level, the purpose of the component d of the colour c_x is to encode the moment at which all edges of m in M are already created going bottom-up along the tree T.

Let $\Gamma_k := \{0,1\} \times \{0,1,2\} \times \{0,1\}^{3\cdot(k+1)} \times \{0,1,\ldots,k\}$ be our set of colours (where, as above, we have $c_x \in \Gamma_k$ for every x, and "a half" of the colour space – with the first component equal to 1, remains unused so far), and let $\ell = |\Gamma_k| = 2 \cdot 3 \cdot 2^{3(k+1)} \cdot (k+1) = 6(k+1) \cdot 8^{k+1}$. We construct a (P°, ℓ)-expression $\varphi = \varphi^r$ valued G recursively, making subexpressions φ^t valued G^t along the nodes $t \in V(T)$, as follows:

I. For a node $t \in V(T)$ (including leaves), in the leaf-to-root tree order, we start with an empty expression φ_0 and $G^t_0 = \emptyset$ if t is a leaf. If t has one child s, then we take the expression φ_0 already constructed at s, and $G^t_0 = G^s$. If t has two children s, s', then we let φ_0 be the union operation over the expressions constructed at s and s', and $G^t_0 = G^s \dot\cup G^{s'}$. Note that, in the latter case, M has no edges between the sets Y_s and $Y_{s'}$ by the interpolation property, and so there are no edges between the (disjoint) subgraphs G^s and $G^{s'}$ in G.

II. Subsequently, for $Y_t' := Y_t \cap X_t$ (informally, Y_t' are the vertices of M whose last bag in T is right at t) we choose an arbitrary order $Y_t' = (m_1, \dots, m_a)$, $a = |Y_t'|$, of these vertices – possibly $a = 0$ if $Y_t' = \emptyset$. For $i = 1, \dots, a$, we repeat the following:

a) We start the expression φ_i by making a union of previous φ_{i-1} (if nonempty) and of the above subexpression constructing G_{m_i}.

b) Now we add in φ_i all required edges between G_{m_i} and $\big(G_0^t \cup G_{m_1} \cup \cdots \cup G_{m_{i-1}}\big)$. Using information stored in the labels of φ_{i-1} and the labels of vertices of G_{m_i}, this is a routine task as follows. For simplicity, we write $*$ for an arbitrary value.

- Let $j = q(m_i)$. For $(\alpha, \beta) \in \{(0,2), (1,0), (2,1)\}$, we add edges between each colour $(1, \alpha,\, *, \dots, b_j, \dots, *,\, *)$ where $b_j = (1, *, *)$, and each colour $(0, \beta,\, *, \dots, *,\, *)$. (Note that only vertices of G_{m_i} may currently hold colours starting with 0.)
- Similarly, for $(\alpha, \beta) \in \{(0,0), (1,1), (2,2)\}$, we add edges between each colour $(1, \alpha,\, *, \dots, b_j, \dots, *,\, *)$ where $b_j = (*, 1, *)$, and each colour $(0, \beta,\, *, \dots, *,\, *)$.
- And again, for $(\alpha, \beta) \in \{(0,1), (1,2), (2,0)\}$, we add edges between each colour $(1, \alpha,\, *, \dots, b_j, \dots, *,\, *)$ where $b_j = (*, *, 1)$, and each colour $(0, \beta,\, *, \dots, *,\, *)$.

c) Then we recolour every colour $c = (0, \beta,\, b_0, \dots, b_k,\, d)$ in the previous, where $\beta \in \{0,1,2\}$ and $b_0, \dots, b_k,\, d$ are arbitrary, to colour $c' = (1, \beta,\, b_0, \dots, b_k,\, d)$. This finishes the expression φ_i constructing a subgraph on $V(G_0^t \cup G_{m_1} \cup \cdots \cup G_{m_i})$.

III. Continuing on the expression φ_a from the previous point, we for all $i \in \{1, \dots, a\}$ do the following. We recolour every colour $c = \big(1, \beta,\, b_0, \dots, b_k,\, q(m_i)\big)$ in φ_a, where $\beta \in \{0,1,2\}$ and $b_0, \dots, b_k$ are arbitrary, to colour $c' = \big(1, \beta,\, (0,0,0)^{k+1},\, q(m_i)\big)$. (The purpose is to prevent creation of further edges from the recoloured vertices which got finished.) This finishes the sought expression φ^t with intended value G^t at the node t.

Now, the constructed (P°, ℓ)-expression $\varphi = \varphi^r$ clearly creates (precisely) all vertices and (at least) all edges of G, and uses at most $\ell = 6(k+1) \cdot 8^{k+1}$ colours. The proof will be finished once we prove that no other edges than those of G have been created by φ in G^r.

There are three points in verification of the last task.

- First, colouring in the process of construction of φ ensures that no additional edges are created within each of the graphs G_m above, and no edges are ever created between G_m and $G_{m'}$ if $mm' \notin E(M)$. The only operation adding edges in φ, besides the subexpressions making each G_m, is as defined in item IIb) above, and so it always adds only edges from $V(G_{m_i})$ to the rest of the current subgraph.
- Second, the operations in IIb) indeed add precisely those edges between G_{m_i} and $\big(G_0^t \cup G_{m_1} \cup \cdots \cup G_{m_{i-1}}\big)$ which exist in G, thanks to our definition of the colours c_x.
- And third (which relates to both previous points), the "hash" function $q : V(M) \to \{0, 1, \dots, k\}$ used in the construction of our colours indeed unambiguously identifies neighbours we want to make adjacent as in G thanks to assumed injectivity of q on each bag of $(T, \mathcal{X})$ and the recolouring performed in item III.

Regarding the second part of the Theorem, the graph M_1, we cannot directly employ Theorem 4.1 since that would give us only a factor (of the strong product) of bounded clique-width, but containing unbounded bipartite cliques in the worst case. We instead provide an ad-hoc construction of the desired factor M_1 which is closely related to fine details of the (P°, ℓ)-expression φ^r of G described above.

Let $\Gamma_k' := \{0,1\} \times \{0,1,2\} \times \{0,1\}^{3\cdot(k+1)}$, i.e., we have got $\Gamma_k' \times \{0, 1, \dots, k\} = \Gamma_k$ as above. Recall also that $Y_t' := Y_t \cap X_t$ for $t \in V(T)$ denotes the vertices of the graph M whose last bag in T (going bottom-up) is right at t, and that, when defining the expression φ of G, we have ordered the members of Y_t' where $a = |Y_t'|$ as $Y_t' = (m_1, \dots, m_a)$.

We define the sought graph M_1 such that $V(M_1) := V(M) \times \Gamma'_k$, and $E(M_1) \subseteq F$ where $F = \{\{(m,c),(m',c')\} : m = m' \vee mm' \in E(M),\ c,c' \in \Gamma'_k\}$. This setting clearly implies that tree-width of M_1 is going to be at most $(k+1)\cdot|\Gamma'_k| < 6(k+1)\cdot 8^{k+1}$, regardless of the detailed definition of its edges. We finish the definition of M_1 as follows:

i. For each $m \in V(M)$ and $\iota \in \{0,1\}$, we have $\{(m,c),(m,c')\} \in E(M_1)$ if and only if $c = (\iota, *, *, \ldots, *)$ and $c' = (\iota, *, *, \ldots, *) \neq c$.

ii. For each $m \neq m' \in V(M)$ such that $mm' \in E(M)$, up to symmetry, we either have $m \in Y'_t$ and $m' \in Y'_u$ where u is closer to the root of T than t, or $m, m' \in Y'_t$ and m precedes m' in the order $Y'_t = (m_1, \ldots, m_a)$ mentioned above. We define $\{(m,c),(m',c')\} \in E(M_1)$ if and only if, for $j = q(m')$ and some $\alpha \in \{0,1,2\}$, one of the following holds:
- $c = (*, \alpha, *, \ldots, b_j, \ldots, *)$, $c' = (*, (\alpha+2) \bmod 3, *, \ldots, *)$ and $b_j = (1,*,*)$, or
- $c = (*, \alpha, *, \ldots, b_j, \ldots, *)$, $c' = (*, \alpha, *, \ldots, *)$ and $b_j = (*,1,*)$, or
- $c = (*, \alpha, *, \ldots, b_j, \ldots, *)$, $c' = (*, (\alpha+1) \bmod 3, *, \ldots, *)$ and $b_j = (*,*,1)$.

iii. No other edges exist in M_1.

It remains to identify an isomorphism of $G \subseteq P \boxtimes M$ to an induced subgraph of the product $P \boxtimes M_1$. To each vertex $x \in V(G)$ such that $x = [p_i, m]$ in $P \boxtimes M$, we assign $x \mapsto [p_i, (m, c_i)]$ in $P \boxtimes M_1$ where $c_i = (a_i,\ i \bmod 3,\ b_0, \ldots, b_k)$ is determined in the following:

(I) The first component a_i of c_i is defined inductively by i (for each fixed m) as follows; it is $a_1 = 0$, and for each $i > 1$ we let $a_i = 1 - a_{i-1}$ if there is $y = [p_{i-1}, m] \in V(G)$ such that $xy \notin E(G)$, and $a_i = a_{i-1}$ otherwise.

(II) For each $j \in \{0,1,\ldots,k\}$ where $j = q(m')$ for some $m' \in X_{t_m}$ (and recall that there is at most one such m' for j since q is injective on each bag), the component b_j is determined as $b_j = (b^1, b^2, b^3)$ where, for $k = 1,2,3$, $b^k = 1$ if and only if $\{x, [p_{i+k-2}, m']\} \in E(G)$.

(III) If undetermined by the previous point, b_j may be chosen arbitrarily.

Let G' be the induced subgraph of $P \boxtimes M_1$ determined by the previous assignment $\mapsto$; our remaining task is to prove that $\mapsto$ is an isomorphism of G to G'. By $\boxtimes$ and the definition of M_1, we know that edges of G and of G' are of the form $e = \{[p_i, m], [p_j, m']\}$ and $e' = \{[p_i, (m, *)], [p_j, (m', *)]\}$, respectively, where $j \in \{i-1, i, i+1\}$ and $mm' \in E(M)$ or $m = m'$.

In the case of $m = m'$, we get $e \in E(G) \iff e' \in E(G')$ already by the definition of the component a_i (a_j) in the point (I). For $mm' \in E(M)$, we get the same straightforwardly from the definition of the edge set of M_1, precisely the point ii. above, and from the (matching) point (II) of the definition of $\mapsto$. ◀

▶ Remark 4.7. Theorem 4.6 has a natural and straightforward extension to graphs $P \in \mathcal{H}$ where $\mathcal{H}$ is any graph class of bounded maximum degree (instead of the class of paths). We skip details due to their additional technical difficulty.

5 Concluding Remarks

The primary focus of our paper is an introduction of a new concept of potential interest, and as such it naturally brings many questions and open problems, (some of) which we briefly survey in this last section.

From the TCS perspective, the probably most important question is about the complexity of computing the $\mathcal{H}$-clique-width. Computing traditional clique-width exactly is NP-hard [14], and hence the same holds for computing $\mathcal{H}$-clique-width exactly in general. However, a question is whether for some special classes $\mathcal{H}$ one could compute exact $\mathcal{H}$-clique-width faster. This is trivially possible, by Claim 2.2 c), when $\mathcal{H}$ is the class of all graphs – which is uninteresting. Is it true that computing $\mathcal{H}$-clique-width exactly is NP-hard for every fixed family $\mathcal{H}$ except in "similarly trivial" cases?

On the other hand, traditional clique-width can be approximated in FPT time with respect to the solution value [18,21]. A big goal would be to extend this approximation result to $\mathcal{H}$-clique-width, perhaps with an additional parameter capturing some properties of $\mathcal{H}$. In particular, with respect to Section 4, we emphasize:

▷ Problem 5.1. Let $\mathcal{P}^\circ$ denote the class of reflexive paths. Can one, for input graph G, approximate $\mathcal{P}^\circ$-cw(G) in FPT time with respect to the solution value?

Next group of questions concerns combinatorial properties proved in this paper. In regard of Section 3, we bring the following one:

▷ Problem 5.2 (cf. Theorem 3.4). Can we characterize families $\mathcal{H}$ of loop graphs such that, for all graphs, bounded $\mathcal{H}$-clique-width implies bounded local clique-width?
This question may be interesting even when restricted to particular graph classes which are of unbounded local clique-width.

A more interesting and natural question, however, comes in a direct relation to the Planar product structure theorem and to Theorem 4.6. We know that graphs of bounded clique-width that do not contain large $K_{t,t}$ subgraphs are as well of bounded tree-width. A natural counterpart of this claim in the context of $\mathcal{P}^\circ$-clique-width would be:

▷ Problem 5.3 (cf. Theorem 4.1, Theorem 4.6). Assume a fixed integer t and an arbitrary graph G such that $\mathcal{P}^\circ$-cw$(G) \leq t$ and G has no $K_{t,t}$ subgraph. Is it then true that $G \subseteq P \boxtimes M$ where P is a path and M is a suitable graph of tree-width bounded in terms of t?

Another question, already mentioned in Section 2, is whether $\mathcal{H}$-clique-width is (at least asymptotically) closed under taking graph complement. This is a prominent and desired property of ordinary clique-width. It would be natural to ask whether, having any simple graph G and its complement $\bar{G}$, we can bound $\mathcal{H}$-cw(G) in terms of $\mathcal{H}$-cw$(\bar{G})$. However, classes $\mathcal{H}$ of bounded degree have bounded local clique-width (Theorem 3.4) and this property is not closed under taking complement.

Instead, we ask whether, for every graph H, there is a graph H' such that for all graphs G, the $\{H'\}$-clique-width of the complement $\bar{G}$ is bounded by a function of the $\{H\}$-clique-width of G. Although we do not have a simple concrete counterexample at hand, we conjecture this is not possible with arbitrary H. One can thus, when being closed under complements is a desirable property, consider only classes $\mathcal{H}$ which are closed under complements themselves (but even that subcase is not trivial), or enrich Definition 2.1 with an operation of adding edges between labels (i, v) and (j, w) over all pairs $(v, w) \in V(H)^2$ such that $vw \notin E(H)$ (note that the latter is much stronger than simply requiring $\mathcal{H}$ to be complement-closed).

We also suggest to study the special case of $\mathcal{T}^\circ$-clique-width when $\mathcal{H} = \mathcal{T}^\circ$ is the family of (all) reflexive trees. The related question in the context of the traditional product structure – that is which graph classes (other than, say, planar graphs) can be expressed as subgraphs of the strong products $T \boxtimes M$ where T is an arbitrary tree and M is of bounded tree-width, does not seem to be explicitly studied yet. We, however, do not have any progress in this direction so far.

Our last batch of questions concerns possible relations of $\mathcal{H}$-clique-width to the currently hot trend of studying structural graph properties through the lens of FO logic on graphs and of FO transductions – transformations of one graph into another defined by FO formulas. In this we use some logic-oriented terms which are not formally defined here (such as, in particular, of *FO transductions*) and we refer for their definitions, e.g., to [6].

First, one may ask for which families $\mathcal{H}$, classes of bounded $\mathcal{H}$-clique-width are monadically dependent, i.e., such that one cannot FO-transduce all finite graphs from graphs of bounded $\mathcal{H}$-clique-width. A partial answer is provided by Theorem 3.4 and Proposition 4.4, but a full characterization of such classes $\mathcal{H}$ is currently out of our reach. As witnessed by Proposition 4.4, classes of bounded $\mathcal{H}$-clique-width can be monadically independent even if $\mathcal{H}$ itself is monadically dependent.

Second, it is interesting to investigate whether and when, having a graph class $\mathcal{G}$ obtained as an FO transduction of a class of bounded $\mathcal{H}$-clique-width, one can find a class $\mathcal{H}^+$ depending on $\mathcal{H}$ (e.g., $\mathcal{H}^+$ an FO transduction of $\mathcal{H}$), such that the $\mathcal{H}^+$-clique-width of $\mathcal{G}$ is bounded. In relation to the Planar product structure, we formulate the following two specific questions in this direction:

▷ Problem 5.4. Assume that a graph class $\mathcal{G}$ is obtained from the class of planar graphs by an FO transduction τ. Is it true that one can give an FO transduction σ, depending on τ, such that the $\mathcal{P}^\sigma$-clique-width of $\mathcal{G}$ is bounded where $\mathcal{P}^\sigma$ is the class of loop graphs obtained from the class of all paths by σ?

Problem 5.4 seems to be much easier if we, instead of requiring bounded $\mathcal{P}^\sigma$-clique-width of every member of $\mathcal{G}$, require only that every graph from $\mathcal{G}$ has a bounded perturbation of bounded $\mathcal{P}^\sigma$-clique-width. This is possibly extensible to classes $\mathcal{H}$ of bounded degree.

▷ Problem 5.5. Let $\mathcal{P}^\sigma$ be the class of loop graphs obtained from the class of all paths by an FO transduction σ. Assume that $\mathcal{G}$ is a graph class of bounded $\mathcal{P}^\sigma$-clique-width and $\mathcal{G}$ is monadically stable, meaning that one cannot define on graphs of $\mathcal{G}$ an arbitrarily long linear order using FO formulas. Is it then true that there exists an FO transduction τ, such that $\mathcal{G}$ is obtained, by τ, from a graph class that admits the traditional product structure?

Third, Theorem 3.4 and Corollary 3.5 can be read as that if a class $\mathcal{H}$ is of bounded degree, then the FO model checking problem is FPT on all classes of bounded $\mathcal{H}$-clique-width. Classes of bounded degree are a prime example of those having FO model checking in FPT [22]. Unfortunately, a bold conjecture claiming that for every class $\mathcal{H}$ having complexity of the FO model checking problem in FPT, the complexity of the FO model checking problem is again in FPT on every class of bounded $\mathcal{H}$-clique-width (perhaps assuming a given decomposition), is very likely false due to the construction given in Proposition 4.4 – the constructed graphs there actually FO transduce all graphs. Hence, it follows from the results of Dreier, Mählmann, and Toruńczyk [8] that FO model checking on classes of bounded $\mathcal{H}_{half}$-clique-width, where $\mathcal{H}_{half}$ denotes the class of reflexive half-graphs, is AW[$\star$]-hard. However, a weaker version of this conjecture, with a suitable additional restriction on $\mathcal{H}$, might still be true.

References

1 Marthe Bonamy, Cyril Gavoille, and Michal Pilipczuk. Shorter labeling schemes for planar graphs. *SIAM J. Discret. Math.*, 36(3):2082–2099, 2022. doi:10.1137/20M1330464.

2 Édouard Bonnet, Colin Geniet, Eun Jung Kim, Stéphan Thomassé, and Rémi Watrigant. Twin-width II: small classes. In *SODA*, pages 1977–1996. SIAM, 2021. doi:10.1137/1.9781611976465.118.

3 Édouard Bonnet, Eun Jung Kim, Stéphan Thomassé, and Rémi Watrigant. Twin-width I: tractable FO model checking. *J. ACM*, 69(1):3:1–3:46, 2022. doi:10.1145/3486655.

4 Édouard Bonnet, O-joung Kwon, and David R. Wood. Reduced bandwidth: a qualitative strengthening of twin-width in minor-closed classes (and beyond). *CoRR*, abs/2202.11858, 2022. arXiv:2202.11858.

5 Anuj Dawar, Martin Grohe, and Stephan Kreutzer. Locally excluding a minor. In *LICS*, pages 270–279. IEEE Computer Society, 2007. doi:10.1109/LICS.2007.31.

6 Patrice Ossona de Mendez. First-order transductions of graphs (invited talk). In *STACS*, volume 187 of *LIPIcs*, pages 2:1–2:7. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPICS.STACS.2021.2.

7 Guoli Ding, Bogdan Oporowski, James G. Oxley, and Dirk Vertigan. Unavoidable minors of large 3-connected binary matroids. *J. Comb. Theory, Ser. B*, 66(2):334–360, 1996. doi:10.1006/JCTB.1996.0026.

8 Jan Dreier, Nikolas Mählmann, and Szymon Torunczyk. Flip-breakability: A combinatorial dichotomy for monadically dependent graph classes. In Bojan Mohar, Igor Shinkar, and Ryan O'Donnell, editors, *Proceedings of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024, Vancouver, BC, Canada, June 24-28, 2024*, pages 1550–1560. ACM, 2024. doi:10.1145/3618260.3649739.

9 Vida Dujmovic, Louis Esperet, Cyril Gavoille, Gwenaël Joret, Piotr Micek, and Pat Morin. Adjacency labelling for planar graphs (and beyond). *J. ACM*, 68(6):42:1–42:33, 2021. doi:10.1145/3477542.

10 Vida Dujmovic, Louis Esperet, Gwenaël Joret, Bartosz Walczak, and David R. Wood. Planar graphs have bounded nonrepetitive chromatic number. *Advances in Combinatorics*, March 2020. doi:10.19086/aic.12100.

11 Vida Dujmovic, Gwenaël Joret, Piotr Micek, Pat Morin, Torsten Ueckerdt, and David R. Wood. Planar graphs have bounded queue-number. *J. ACM*, 67(4):22:1–22:38, 2020. doi:10.1145/3385731.

12 Vida Dujmovic, Pat Morin, and David R. Wood. Graph product structure for non-minor-closed classes. *J. Comb. Theory, Ser. B*, 162:34–67, 2023. doi:10.1016/J.JCTB.2023.03.004.

13 Zdenek Dvořák, Tony Huynh, Gwenaël Joret, Chun-Hung Liu, and David R. Wood. Notes on graph product structure theory. In *2019-20 MATRIX Annals*, pages 513–533, Cham, 2021. Springer International Publishing.

14 Michael R. Fellows, Frances A. Rosamond, Udi Rotics, and Stefan Szeider. Clique-width is NP-complete. *SIAM J. Discret. Math.*, 23(2):909–939, 2009. doi:10.1137/070687256.

15 Markus Frick and Martin Grohe. Deciding first-order properties of locally tree-decomposable structures. *J. ACM*, 48(6):1184–1206, 2001. doi:10.1145/504794.504798.

16 Petr Hliněný and Jan Jedelský. Twin-width of planar graphs is at most 8, and at most 6 when bipartite planar. In *ICALP*, volume 261 of *LIPIcs*, pages 75:1–75:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPICS.ICALP.2023.75.

17 Petr Hliněný and Jan Jedelský. H-clique-width and a hereditary analogue of product structure. *CoRR*, abs/2403.16789, 2024. arXiv:2403.16789.

18 Petr Hliněný and Sang-il Oum. Finding branch-decompositions and rank-decompositions. *SIAM J. Comput.*, 38(3):1012–1032, 2008. doi:10.1137/070685920.

19 Hugo Jacob and Marcin Pilipczuk. Bounding twin-width for bounded-treewidth graphs, planar graphs, and bipartite graphs. In *WG*, volume 13453 of *Lecture Notes in Computer Science*, pages 287–299. Springer, 2022. doi:10.1007/978-3-031-15914-5_21.

20 Michael Lampis. Algorithmic meta-theorems for restrictions of treewidth. *Algorithmica*, 64(1):19–37, 2012. doi:10.1007/S00453-011-9554-X.

21 Sang-il Oum and Paul Seymour. Approximating clique-width and branch-width. *J. Comb. Theory, Ser. B*, 96(4):514–528, 2006. doi:10.1016/J.JCTB.2005.10.006.

22 D. Seese. Linear time computable problems and first-order descriptions. *Math. Structures Comput. Sci.*, 6(6):505–526, 1996. doi:10.1016/S1571-0661(05)80203-8.

23 Torsten Ueckerdt, David R. Wood, and Wendy Yi. An improved planar graph product structure theorem. *Electron. J. Comb.*, 29(2), 2022. doi:10.37236/10614.