The Role of Planarity in Connectivity Problems Parameterized by Treewidth

Julien Baste and Ignasi Sau$^{(\boxtimes)}$

AlGCo Project-team, CNRS, LIRMM, Montpellier, France
jbaste@ens-cachan.fr, ignasi.sau@lirmm.fr

Abstract. For some years it was believed that for "connectivity" problems such as HAMILTONIAN CYCLE, algorithms running in time $2^{O(\mathbf{tw})} \cdot n^{O(1)}$ –called *single-exponential*– existed only on planar and other sparse graph classes, where $\mathbf{tw}$ stands for the treewidth of the n-vertex input graph. This was recently disproved by Cygan *et al.* [FOCS 2011], Bodlaender *et al.* [ICALP 2013], and Fomin *et al.* [SODA 2014], who provided single-exponential algorithms on general graphs for most connectivity problems that were known to be solvable in single-exponential time on sparse graphs. In this article we further investigate the role of planarity in connectivity problems parameterized by treewidth, and convey that several problems can indeed be distinguished according to their behavior on planar graphs. Known results from the literature imply that there exist problems, like CYCLE PACKING, that *cannot* be solved in time $2^{o(\mathbf{tw} \log \mathbf{tw})} \cdot n^{O(1)}$ on general graphs but that can be solved in time $2^{O(\mathbf{tw})} \cdot n^{O(1)}$ when restricted to planar graphs. Our main contribution is to show that there exist natural problems that can be solved in time $2^{O(\mathbf{tw} \log \mathbf{tw})} \cdot n^{O(1)}$ on general graphs but that *cannot* be solved in time $2^{o(\mathbf{tw} \log \mathbf{tw})} \cdot n^{O(1)}$ even when restricted to planar graphs. Furthermore, we prove that PLANAR CYCLE PACKING and PLANAR DISJOINT PATHS *cannot* be solved in time $2^{o(\mathbf{tw})} \cdot n^{O(1)}$. The mentioned negative results hold unless the ETH fails. We feel that our results constitute a first step in a subject that can be further exploited.

Keywords: Parameterized complexity · Treewidth · Connectivity problems · Single-exponential algorithms · Planar graphs · Dynamic programming

1 Introduction

Motivation and previous work. Treewidth is a fundamental graph parameter that, loosely speaking, measures the resemblance of a graph to a tree. It was introduced, among other equivalent definitions given by other authors, by Robertson and Seymour in the early stages of their monumental Graph Minors

project [18], and its algorithmic importance was significantly popularized by Courcelle's theorem [3], stating that any graph problem that can be expressed in CMSO logic can be solved in time $f(\mathbf{tw}) \cdot n$ on graphs with n vertices and treewidth $\mathbf{tw}$, where f is some function depending on the problem. Nevertheless, the function $f(\mathbf{tw})$ given by Courcelle's theorem is unavoidably huge [10], so from an algorithmic point of view it is crucial to identify problems for which $f(\mathbf{tw})$ grows *moderately* fast.

Many problems can be solved in time $2^{O(\mathbf{tw} \log \mathbf{tw})} \cdot n^{O(1)}$ when the n-vertex input (general) graph comes equipped with a tree-decomposition of width $\mathbf{tw}$. Intuitively, this is the case of problems that can be solved via dynamic programming on a tree-decomposition by enumerating all *partitions* or *packings* of the vertices in the bags of the tree-decomposition, which are $\mathbf{tw}^{O(\mathbf{tw})} = 2^{O(\mathbf{tw} \log \mathbf{tw})}$ many. In this article we only consider this type of problems and, more precisely, we are interested in which of these problems can be solved in time $2^{O(\mathbf{tw})} \cdot n^{O(1)}$; such a running time is called *single-exponential*. This topic has been object of extensive study during the last decade. Let us briefly overview the main results on this line of research.

It is well known that problems that have *locally checkable certificates*[1], like VERTEX COVER or DOMINATING SET, can be solved in single-exponential time on general graphs. Intuitively, for this problems it is enough to enumerate *subsets* of the bags of a tree-decomposition (rather than partitions or packings), which are $2^{O(\mathbf{tw})}$ many. A natural class of problems that do *not* have locally checkable certificates is the class of so-called *connectivity problems*, which contains for example HAMILTONIAN CYCLE, STEINER TREE, or CONNECTED VERTEX COVER. These problems have the property that the solutions should satisfy a *connectivity* requirement (see [2,4,19] for more details), and using classical dynamic programming techniques it seems that for solving such a problem it is necessary to enumerate partitions or packings of the bags of a tree-decomposition.

A series of articles provided single-exponential algorithms for connectivity problems when the input graphs are restricted to be sparse, namely planar [9], of bounded genus [7,19], or excluding a fixed graph as a minor [8,20]. The common key idea of these works is to use special types of branch-decompositions (which are objects similar to tree-decompositions) with nice combinatorial properties, which strongly rely on the fact that the input graphs are sparse.

Until very recently, it was a common belief that all problems solvable in single-exponential time on general graphs should have locally checkable certificates, specially after Lokshtanov *et al.* [16] proved that one connectivity problem, namely DISJOINT PATHS, cannot be solved in time $2^{o(\mathbf{tw} \log \mathbf{tw})} \cdot n^{O(1)}$ on general graphs unless the Exponential Time Hypothesis (ETH) fails[2]. This

---

[1] That is, certificates consisting of a constant number of bits per vertex that can be checked by a cardinality check and by iteratively looking at the neighborhoods of the input graph.

[2] The ETH states that there exists a positive real number s such that 3-CNF-SAT with n variables and m clauses cannot be solved in time $2^{sn} \cdot (n + m)^{O(1)}$. See [15] for more details.

credence was disproved by Cygan *et al.* [4], who provided single-exponential *randomized* algorithms on general graphs for several connectivity problems, like Longest Path, Feedback Vertex Set, or Connected Vertex Cover. More recently, Bodlaender *et al.* [2] presented single-exponential *deterministic* algorithms for basically the same connectivity problems, and an alternative proof based on matroids was given by Fomin *et al.* [11]. These results have been considered a breakthrough, and in particular they imply that most connectivity problems that were known to be solvable in single-exponential time on sparse graph classes [7–9,19,20] are also solvable in single-exponential time on general graphs [2,4].

Our main results. In view of the above discussion, a natural conclusion is that sparsity may not be particularly helpful or relevant for obtaining single-exponential algorithms. However, in this article we convey that sparsity (in particular, planarity) *does* play a role in connectivity problems parameterized by treewidth. To this end, among the problems that can be solved in time $2^{O(\mathbf{tw} \log \mathbf{tw})} \cdot n^{O(1)}$ on general graphs, we distinguish the following three disjoint types:

- **Type 1**: Problems that can be solved in time $2^{O(\mathbf{tw})} \cdot n^{O(1)}$ on general graphs.
- **Type 2:** Problems that *cannot* be solved in time $2^{o(\mathbf{tw} \log \mathbf{tw})} \cdot n^{O(1)}$ on general graphs unless the ETH fails, but that can be solved in time $2^{O(\mathbf{tw})} \cdot n^{O(1)}$ when restricted to planar graphs.
- **Type 3:** Problems that *cannot* be solved in time $2^{o(\mathbf{tw} \log \mathbf{tw})} \cdot n^{O(1)}$ even when restricted to planar graphs, unless the ETH fails.

Problems that have locally checkable certificates are of Type 1 (see also [17] for a logical characterization of such problems). As discussed in Sect. 2, known results [4,13] imply that there exist problems of Type 2, such as Cycle Packing. Our main contribution is to show that there exist (natural) problems of Type 3, thus demonstrating that some connectivity problems can indeed be distinguished according to their behavior on planar graphs. More precisely, we prove the following results:

- In Sect. 2 we provide some examples of problems of Type 2. Furthermore, we prove that Planar Cycle Packing cannot be solved in time $2^{o(\mathbf{tw})} \cdot n^{O(1)}$ unless the ETH fails, and therefore the running time $2^{O(\mathbf{tw})} \cdot n^{O(1)}$ is tight.
- In Sect. 3 we provide an example of problem of Type 3: Monochromatic Disjoint Paths, which is a variant of the Disjoint Paths problem on a vertex-colored graph with additional restrictions on the allowed colors for each path. To the best of our knowledge, problems of this type had not been identified before.

In order to obtain our results, for the upper bounds we strongly follow the algorithmic techniques based on *Catalan structures* used in [7–9,19,20], and for some of the lower bounds we use the framework introduced in [16], and that has

been also used in [4]. Due to space limitations, the proofs of the results marked with '[⋆]' can be found in the full version of this article [1].

Additional results and further research. We feel that our results about the role of planarity in connectivity problems parameterized by treewidth are just a first step in a subject that can be much exploited, and we think that the following avenues are particularly interesting:

- It is known that DISJOINT PATHS can be solved in time $2^{O(\mathbf{tw} \log \mathbf{tw})} \cdot n^{O(1)}$ on general graphs [21], and that this bound is asymptotically tight under the ETH [16]. The fact whether DISJOINT PATHS belongs to Type 2 or Type 3 (or maybe even to some other type in between) remains an important open problem that we have been unable to solve (it is worth noting that Catalan structures do *not* seem to yield an algorithm in time $2^{O(\mathbf{tw})} \cdot n^{O(1)}$). Towards a possible answer to this question, we prove in [1] that PLANAR DISJOINT PATHS cannot be solved in time $2^{o(\mathbf{tw})} \cdot n^{O(1)}$ unless the ETH fails.
- Lokshtanov *et al.* [14] have proved that for a number of problems such as DOMINATING SET or q-COLORING, the best known constant c in algorithms of the form $c^{\mathbf{tw}} \cdot n^{O(1)}$ on general graphs is best possible unless the Strong ETH fails (similar results have been also given by Cygan *et al.* [4]). Is it possible to provide better constants for these problems on planar graphs? The existence of such algorithms would permit to further refine the problems belonging to Type 1.
- Are there NP-hard problems solvable in time $2^{o(\mathbf{tw})} \cdot n^{O(1)}$?
- Finally, it would be interesting to obtain similar results for problems parameterized by pathwidth, and to extend our algorithms to more general classes of sparse graphs.

Notation and definitions. We use standard graph-theoretic notation, and the reader is referred to [6] for any undefined term. All the graphs we consider are undirected and contain neither loops nor multiple edges. We denote by $V(G)$ the set of vertices of a graph G and by $E(G)$ its set of edges. We use the notation $[k]$ for the set of integers $\{1, \ldots, k\}$. In the set $[k] \times [k]$, a *row* is a set $\{i\} \times [k]$ and a *column* is a set $[k] \times \{i\}$ for some $i \in [k]$. If $\mathbf{P}$ is a problem defined on graphs, we denote by PLANAR $\mathbf{P}$ the restriction of $\mathbf{P}$ to planar input graphs.

A *subgraph* $H = (V_H, E_H)$ of a graph $G = (V, E)$ is a graph such that $V_H \subseteq V$ and $E_H \subseteq E \cap \binom{V_H}{2}$. The *degree* of a vertex v in a graph G, denoted by $\deg_G(v)$, is the number of edges of G containing v. The *grid* $m * k$ is the graph $Gr_{m,k} = (\{a_{i,j} | i \in [m], j \in [k]\}, \{(a_{i,j}, a_{i+1,j}) | i \in [m-1], j \in [k]\} \cup \{(a_{i,j}, a_{i,j+1}) | i \in [m], j \in [k-1]\})$. When $m = k$ we just speak about the *grid of size k*. We say that there is a *path* $s \ldots t$ in a graph G if there exist $m \in \mathbb{N}$ and $x_0, \ldots, x_m$ in $V(G)$ such that $x_0 = s$, $x_m = t$, and for all $i \in [m]$, $(x_{i-1}, x_i) \in E(G)$.

A *tree-decomposition* of width w of a graph $G = (V, E)$ is a pair (T, σ), where T is a tree and $\sigma = \{B_t | B_t \subseteq V, t \in V(T)\}$ such that:

- $\bigcup_{t \in V(T)} B_t = V$;
- For every edge $\{u, v\} \in E$ there is a $t \in V(T)$ such that $\{u, v\} \subseteq B_t$;

- $B_i \cap B_k \subseteq B_j$ for all $\{i, j, k\} \subseteq V(T)$ such that j lies on the path between i and k in T;
- $\max_{t \in V(T)} |B_t| = w + 1$.

The sets B_t are called *bags*. The *treewidth* of G, denoted by $\mathbf{tw}(G)$, is the smallest integer w such that there is a tree-decomposition of G of width w. An *optimal tree-decomposition* is a tree-decomposition of width $\mathbf{tw}(G)$. A *path-decomposition* of a graph $G = (V, E)$ is a tree-decomposition (T, σ) such that T is a path. The *pathwidth* of G, denoted by $\mathbf{pw}(G)$, is the smallest integer w such that there is a path-decomposition of G of width w. Clearly, for any graph G, we have $\mathbf{tw}(G) \leq \mathbf{pw}(G)$.

Throughout the paper, in order to simplify the notation, when the problem and the input graph under consideration are clear, we let n denote the number of vertices of the input graph, $\mathbf{tw}$ its treewidth, and $\mathbf{pw}$ its pathwidth.

2 Problems of Type 2

In this section we deal with CYCLE PACKING. Other problems of Type 2 are discussed in [1].

CYCLE PACKING
Input: An n-vertex graph $G = (V, E)$ and an integer ℓ_0.
Parameter: The treewidth $\mathbf{tw}$ of G.
Question: Does G contain ℓ_0 pairwise vertex-disjoint cycles?

It is proved in [4] that CYCLE PACKING *cannot* be solved in time $2^{o(\mathbf{tw} \log \mathbf{tw})} \cdot n^{O(1)}$ on general graphs unless the ETH fails. On the other hand, a dynamic programming algorithm for PLANAR CYCLE PACKING running in time $2^{O(\mathbf{tw})} \cdot n^{O(1)}$ can be found in [13]. Therefore, it follows that CYCLE PACKING is of Type 2. In Lemma 1 below we provide an alternative algorithm for PLANAR CYCLE PACKING running in time $2^{O(\mathbf{tw})} \cdot n^{O(1)}$, which is a direct application of the techniques based on *Catalan structures* introduced in [9]. We include its proof for completeness, as it yields slightly better constants than [13].

Lemma 1. [⋆] PLANAR CYCLE PACKING *can be solved in time* $2^{O(\mathbf{tw})} \cdot n^{O(1)}$.

It is usually believed that NP-hard problems parameterized by $\mathbf{tw}$ cannot be solved in time $2^{o(\mathbf{tw})} \cdot n^{O(1)}$ under some reasonable complexity assumption. This has been proved in [12] for problems on *general* graphs such as q-COLORABILITY, INDEPENDENT SET, and VERTEX COVER, or in [5] for PLANAR HAMILTONIAN CYCLE, all these results assuming the ETH. Nevertheless, such a result requires an ad-hoc proof for each problem. For instance, to the best of our knowledge, such lower bounds are not known for CYCLE PACKING when the input graph is restricted to be *planar*. We now prove that the running time given by Lemma 1 is asymptotically tight.
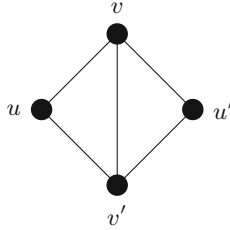
Fig. 1. The choice gadget.

Theorem 1. Planar Cycle Packing *cannot be solved in time* $2^{o(\sqrt{n})} \cdot n^{O(1)}$ *unless the ETH fails. Therefore,* Planar Cycle Packing *cannot be solved in time* $2^{o(\mathbf{tw})} \cdot n^{O(1)}$ *unless the ETH fails.*

Proof. To prove this theorem, we reduce from the Planar Independent Set problem, which consists in finding a maximum-sized set of vertices in a given planar graph that are pairwise nonadjacent. It is shown in [15] that Planar Independent Set cannot be solved in time $2^{o(\sqrt{n})} \cdot n^{O(1)}$ unless the ETH fails[3], where n stands for the number of vertices of the input graph.

Let G be a planar graph on which we want to solve Planar Independent Set. We construct from G a graph H as follows. For each vertex a of G, we add to H a cycle of length equal to the degree of a. If the degree of a is smaller than 3, then the added cycle is of size 3. For each edge (a, b) of G, we add the choice gadget depicted in Fig. 1, by identifying vertex u of the choice gadget with one of the vertices of the cycle representing a, and identifying vertex u' of the choice gadget with one of the vertices of the cycle representing b, in such a way that all the choice gadgets are vertex-disjoint. That concludes the construction of H. Note that a planar embedding of H can be obtained in a straightforward way from a planar embedding of G.

Claim 1. *G contains an independent set of size k if and only if H admits a packing of $m + k$ cycles, where $m = |E(G)|$.*

Proof. Assume first that there is an independent set S of size k in G. Then we choose in H the k cycles that represent the k elements of S, plus a cycle of size 3 in each of the m choice gadgets. Note that we always can take a cycle of size 3 in the choice gadgets, as two adjacent vertices in G cannot be in S. This gives $m + k$ cycles in H.

Conversely, assume that H contains a packing $\mathcal{P}$ of $m + k$ vertex-disjoint cycles. We say that a cycle in $\mathcal{P}$ *hits* a choice gadget if it contains vertex v

---

[3] In [15] that Planar Vertex Cover problem is mentioned, which is equivalent to solving Planar Independent Set, as the complement of a vertex cover is an independent set.

or v', and note that each choice gadget can be hit by at most one cycle, which is necessarily a triangle. We transform $\mathcal{P}$ into a packing $\mathcal{P}'$ with $|\mathcal{P}'| \geq |\mathcal{P}|$ and such that every choice gadget is hit by $\mathcal{P}'$, as follows. Start with $\mathcal{P}' = \mathcal{P}$. Then, for each choice gadget in H, corresponding to an edge (a, b) of G, that is not hit by a cycle in $\mathcal{P}$, we arbitrarily choose one of a or b (say, a), remove from $\mathcal{P}'$ the cycle containing a (if any), and add to $\mathcal{P}'$ the triangle in the choice gadget containing vertex u. As there are only m choice gadgets in H, and as $|\mathcal{P}'| \geq |\mathcal{P}| \geq m + k$, it follows that at least k cycles in $\mathcal{P}'$ do not hit any choice gadget. By construction of H, it clearly means that each of these k cycles is a cycle corresponding to a vertex of G. Let R be the set of these vertices, so we have that $|R| \geq k$. Finally, note that R is an independent set in G, because for each edge $(a, b) \in E(G)$, the corresponding choice gadget is contained in $\mathcal{P}'$, so at most one of a and b can be in R. $\square$

Thus, if we have an algorithm solving PLANAR CYCLE PACKING in time $2^{o(\sqrt{n})} \cdot n^{O(1)}$, then by Claim 1 (and using that $m = O(n)$ because G is planar) we could solve PLANAR INDEPENDENT SET in time $2^{o(\sqrt{n})} \cdot n^{O(1)}$, which is impossible unless the ETH fails [15]. $\square$

3   Problems of Type 3

In this section we prove that the MONOCHROMATIC DISJOINT PATHS problem is of Type 3. We first need to introduce some definitions. Let $G = (V, E)$ be a graph, let k be an integer, and let $c : V \to \{0, \ldots, k\}$ be a color function. Two colors c_1 and c_2 in $\{0, \ldots, k\}$ are *compatible*, and we denote it by $c_1 \equiv c_2$, if $c_1 = 0$, $c_2 = 0$, or $c_1 = c_2$. A path $P = x_1 \ldots x_m$ in G is *monochromatic* if for all $i, j \in [m]$, $i \neq j$, $c(x_i)$ and $c(x_j)$ are two compatible colors. We abuse notation and also define $c(P) = \max_{i \in [m]}(c(x_i))$. We say that P is *colored* x if $x = c(P)$. Two monochromatic paths P and P' are *color-compatible* if $c(P) \equiv c(P')$.

MONOCHROMATIC DISJOINT PATHS
Input: A graph $G = (V, E)$ of treewidth **tw**, a color function $\gamma : V \to \{0, \ldots, \mathbf{tw}\}$, an integer m, and a set $\mathcal{N} = \{\mathcal{N}_i = \{s_i, t_i\} | i \in [m], s_i, t_i \in V\}$.
Parameter: The treewidth **tw** of G.
Question: Does G contain m pairwise vertex-disjoint monochromatic paths from s_i to t_i, for $i \in [m]$?

The proof of the following lemma is inspired from the algorithm given in [21] for the DISJOINT PATHS problem on general graphs.

Lemma 2. [⋆] MONOCHROMATIC DISJOINT PATHS *can be solved in time* $2^{O(\mathbf{tw} \log \mathbf{tw})} \cdot n^{O(1)}$.

We now proceed to provide a matching lower bound for PLANAR MONOCHROMATIC DISJOINT PATHS. For this, we need to define the $k \times k$-HITTING SET problem, first introduced in [16].

$k \times k$-HITTING SET

Input: A family of sets $S_1, S_2, \ldots, S_m \subseteq [k] \times [k]$, such that each set contains at most one element from each row of $[k] \times [k]$.

Parameter: k.

Question: Is there a set S containing exactly one element from each row such that $S \cap S_i \neq \emptyset$ for any $1 \leq i \leq m$?

Theorem 2. (Lokshtanov) et al. [16]. $k \times k$-HITTING SET *cannot be solved in time* $2^{o(k \log k)} \cdot m^{O(1)}$ *unless the ETH fails.*

We state the following theorem in terms of the pathwidth of the input graph, and as any graph G satisfies $\mathbf{tw}(G) \leq \mathbf{pw}(G)$, it implies the same lower bound in the treewidth.

Theorem 3. PLANAR MONOCHROMATIC DISJOINT PATHS *cannot be solved in time* $2^{o(\mathbf{pw} \log \mathbf{pw})} \cdot n^{O(1)}$ *unless the ETH fails.*

Proof. We reduce from $k \times k$-HITTING SET. Let k be an integer and S_1, $S_2, \ldots, S_m \subseteq [k] \times [k]$ such that each set contains at most one element from each row of $[k] \times [k]$. We will first present an overview of the reduction with all the involved gadgets, and then we will provide a formal definition of the constructed planar graph G.

We construct a gadget for each row $\{r\} \times [k]$, $r \in [k]$, which selects the unique pair p of S in this row. First, for each $r \in [k]$, we introduce two new vertices s_r and t_r, a request $\{s_r, t_r\}$, $m + 1$ vertices $v_{r,i}$, $i \in \{0, \ldots, m\}$, and $m + 2$ edges $\{e_{r,0} = (s_r, v_{r,0})\} \cup \{e_{r,i} = (v_{r,i-1}, v_{r,i}) | i \in [m]\} \cup \{e_{r,m+1} = (v_{r,m}, t_r)\}$. That is, we have a path with $m + 2$ edges between s_r and t_r.

Each edge of these paths, except the last one, will be replaced with an appropriate gadget. Namely, for each $r \in [k]$, we replace the edge $e_{r,0}$ with the gadget depicted in Fig. 2, which we call *color-selection* gadget. In this figure, vertex $u_{r,i}$ is colored i. The color used by the path from s_r to t_r in the color-selection gadget will define the pair of the solution of S in the row $\{r\} \times [k]$.

Now that we have described the gadgets that allow to define S, we need to ensure that $S \cap S_i \neq \emptyset$ for any $i \in [m]$. For this, we need the gadget depicted in Fig. 3, which we call *expel* gadget. Each time we introduce this gadget, we add to $\mathcal{N}$ the request $\{s, t\}$. This new requested path uses either vertex u or vertex v, so only one of these vertices can be used by other paths. For each $i \in [m]$, we replace all the edges $\{e_{r,i} | r \in [k]\}$ with the gadget depicted in Fig. 4, which we call *set* gadget. In this figure, $a_{r,i}$ is such that if $(\{r\} \times [k]) \cap S_i = \{\{r, c_{r,i}\}\}$ then $a_{r,i}$ is colored $c_{r,i}$, and if $(\{r\} \times [k]) \cap S_i = \emptyset$ then vertex $a_{r,i}$ is removed from the gadget.

This completes the construction of the graph G, which is illustrated in Fig. 5. Note that G is indeed planar.

The color function γ of G is defined such that for each $r \in [k]$ and $c \in [k]$, $\gamma(u_{r,c}) = c$, and for each $i \in [m]$ and $(r, c) \in S_i$, $\gamma(a_{r,i}) = c$. For any other vertex $v \in V(G)$, we set $\gamma(v) = 0$. Finally, the input of PLANAR MONOCHROMATIC
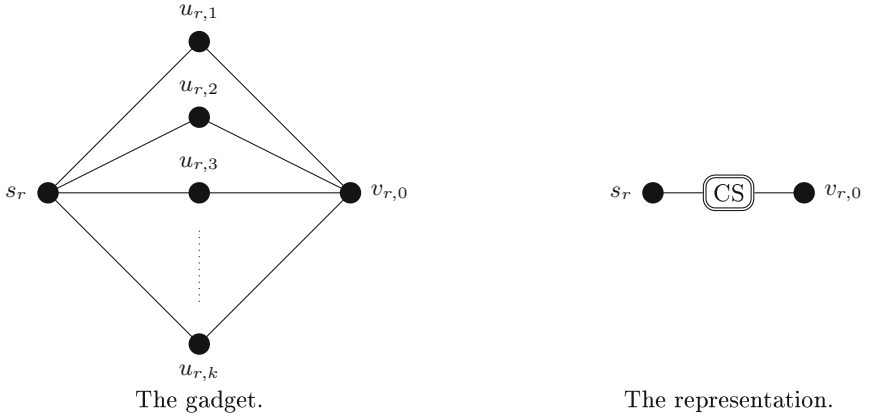
The gadget.   The representation.

Fig. 2. Color-selection gadget, where $u_{r,i}$ is colored c_i for each $i \in [k]$.

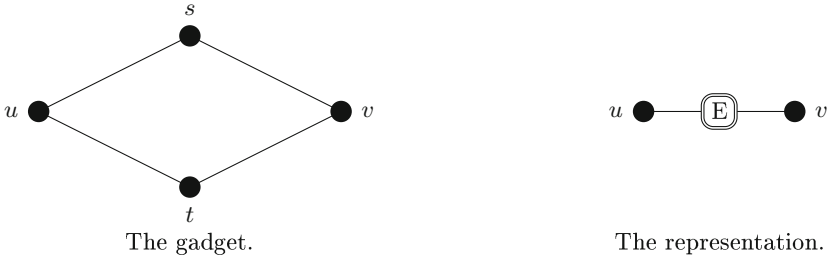The gadget.   The representation.

Fig. 3. Expel gadget.

DISJOINT PATHS is the planar graph G, the color function γ, and the $k+(k-1)\cdot m$ requests $\mathcal{N} = \{\{s_r, t_r\}|r \in [k]\} \cup \{\{s_{r,i}, t_{r,i}\}|r \in [k-1], i \in [m]\}$, the second set of requests corresponding to the ones introduced by the expel gadgets.

Note that because of the expel gadgets, the request $\{s_r, t_r\}$ forces the existence of a path between $v_{r,i-1}$ and $v_{r,i}$ for each $r \in [k]$. Note also that because of the expel gadgets, at least one of the paths between $v_{r,i-1}$ and $v_{r,i}$ should use an $a_{r,i}$ vertex, as otherwise at least two paths would intersect. Conversely, if one path uses a vertex $a_{r,i}$, then we can find all the desired paths in the corresponding set gadgets by using the vertices $w_{r,i,b}$.

Given a solution of PLANAR MONOCHROMATIC DISJOINT PATHS in G, we can construct a solution of $k \times k$-HITTING SET by letting $S = \{(r,c)|r \in [k]$ such that the path from s_r to t_r is colored with color $c\}$. We have that S contains exactly one element of each row, so we just have to check if $S \cap S_i \neq \emptyset$ for each $i \in [m]$. Because of the property of the set gadgets mentioned above, for each $i \in [m]$, the set gadget labeled i ensures that $S \cap S_i \neq \emptyset$.

Conversely, given a solution S of $k \times k$-HITTING SET, for each $(r,c) \in S$ we color the path from s_r to t_r with color c. We assign an arbitrary coloring
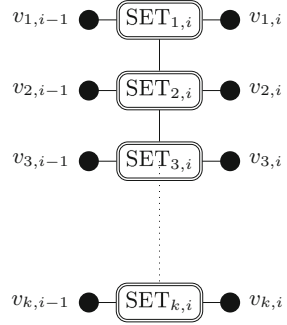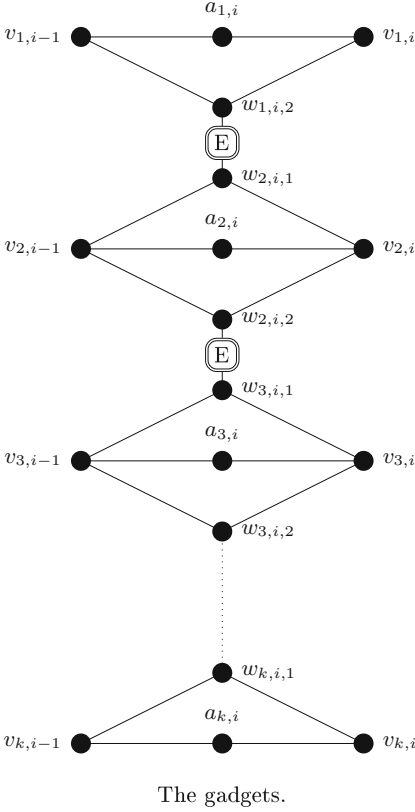
The gadgets.

The representation.

Fig. 4. Set gadgets.

to the other paths. For each $i \in [m]$, we take $(r, c) \in S \cap S_i$ and in the set gadget labeled i, we impose that the path from $v_{r,i-1}$ to $v_{r,i}$ uses vertex $a_{r,i}$. By using the vertices $w_{r,i,b}$ for the other paths, we find the desired $k + (k-1) \cdot m$ monochromatic paths.

Let us now argue about the pathwidth of G. We define for each $r, c \in [k]$ the bag $B_{0,r,c} = \{s_{r'} | r' \in [k]\} \cup \{v_{r',0} | r' \in [k]\} \cup \{u_{r,c}\}$, for each $i \in [m]$, the bag $B_i = \{v_{r,i-1} | r \in [k]\} \cup \{v_{r,i} | r \in [k]\} \cup \{a_{r,i} \in V(G) | r \in [k]\} \cup \{w_{r,i,b} \in V(G) | r \in [k], b \in [2]\} \cup \{s_{r,i} | r \in [k-1]\} \cup \{t_{r,i} | r \in [k-1]\}$, and the bag $B_{m+1} = \{v_{r,m} | r \in [k]\} \cup \{t_r | r \in [k]\}$. Note that for all i in $[m]$, $w_{1,i,1}$ and $w_{k,i,2}$ are not in $V(G)$. The size of each bag is at most $2 \cdot (k-1) + 5 \cdot k - 2 = O(k)$. A path decomposition of G consists of all bags $B_{0,r,c}$, $r, c \in [k]$ and B_i, $i \in [m+1]$ and edges $\{B_i, B_{i+1}\}$ for each $i \in [m]$, $\{B_{0,r,c}, B_{0,r,c+1}\}$ for $r \in [k]$, $c \in [k-1]$, $\{B_{0,r,k}, B_{0,r+1,1}\}$ for $r \in [k]$, and $\{B_{0,k,k}, B_1\}$. Therefore, as we have that $\mathbf{pw}(G) = O(k)$, if one could solve PLANAR MONOCHROMATIC DISJOINT PATHS in time $2^{o(\mathbf{pw} \log \mathbf{pw})} \cdot n^{O(1)}$, then one could also solve $k \times k$-HITTING SET in time $2^{o(k \log k)} \cdot m^{O(1)}$, which is impossible by Theorem 2 unless the ETH fails. □
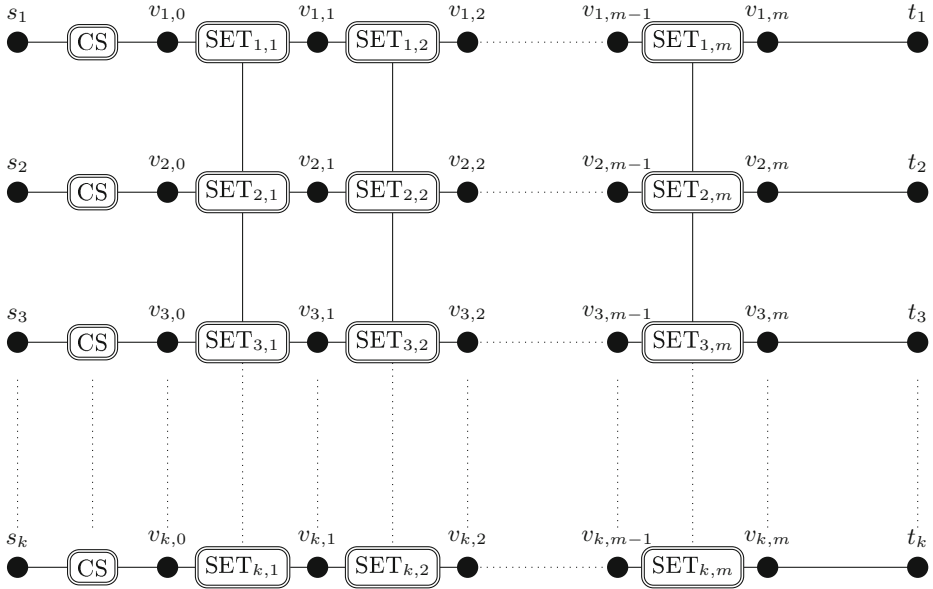
Fig. 5. Final graph G in the reduction of Theorem 3.

References

1. Baste, J., Sau, I.: The role of planarity in connectivity problems parameterized by treewidth. CoRR, abs/1312.2889 (2013)
2. Bodlaender, H.L., Cygan, M., Kratsch, S., Nederlof, J.: Deterministic single exponential time algorithms for connectivity problems parameterized by treewidth. In: Fomin, F.V., Freivalds, R., Kwiatkowska, M., Peleg, D. (eds.) ICALP 2013, Part I. LNCS, vol. 7965, pp. 196–207. Springer, Heidelberg (2013)
3. Courcelle, B.: The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. Inform. Comput. **85**(1), 12–75 (1990)
4. Cygan, M., Nederlof, J., Pilipczuk, M., Pilipczuk, M., van Rooij, J.M.M., Wojtaszczyk, J.O.: Solving connectivity problems parameterized by treewidth in single exponential time. In: Proceeding of the 52nd Annual IEEE Symposium on Foundations of Computer Science (FOCS), pp. 150–159. IEEE Computer Society (2011)
5. Deineko, V.G., Steiner, G., Xue, Z.: Robotic-cell scheduling: special polynomially solvable cases of the traveling salesman problem on permuted monge matrices. J. Comb. Optim. **9**, 381–399 (2005)
6. Diestel, R.: Graph Theory, 3rd edn. Springer, New York (2005)
7. Dorn, F., Fomin, F.V., Thilikos, D.M.: Fast subexponential algorithm for non-local problems on graphs of bounded genus. In: Arge, L., Freivalds, R. (eds.) SWAT 2006. LNCS, vol. 4059, pp. 172–183. Springer, Heidelberg (2006)

8. Dorn, F., Fomin, F.V., Thilikos, D.M.: Catalan structures and dynamic programming in H-minor-free graphs. J. Syst. Sci. **78**(5), 1606–1622 (2012)
9. Dorn, F., Penninkx, E., Bodlaender, H.L., Fomin, F.V.: Efficient exact algorithms on planar graphs: exploiting sphere cut decompositions. Algorithmica **58**(3), 790–810 (2010)
10. Flum, J., Grohe, M.: Parameterized Complexity Theory. Theoretical Computer Science. Springer, Berlin (2006)
11. Fomin, F.V., Lokshtanov, D., Saurabh, S.: Efficient Computation of Representative Sets with Applications in Parameterized and Exact Algorithms. In: Proceeding of SODA'14. CoRR, abs/1304.4626 (2013)
12. Impagliazzo, R., Paturi, R., Zane, F.: Which problems have strongly exponential complexity? J. Comput. Syst. Sci. **63**(4), 512–530 (2001)
13. Kloks, T., Lee, C.M., Liu, J.: New algorithms for k-face cover, k-feedback vertex set, and k-disjoint cycles on plane and planar graphs. In: Kučera, L. (ed.) WG 2002. LNCS, vol. 2573, pp. 282–295. Springer, Heidelberg (2002)
14. Lokshtanov, D., Marx, D., Saurabh, S.: Known algorithms on graphs of bounded treewidth are probably optimal. In: Proceeding of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 777–789 (2011)
15. Lokshtanov, D., Marx, D., Saurabh, S.: Lower bounds based on the exponential time hypothesis. Bull. EATCS **105**, 41–72 (2011)
16. Lokshtanov, D., Marx, D., Saurabh, S.: Slightly superexponential parameterized problems. In: Proceeding of the 22nd Annual ACM-SIAM Symposium on Discrete algorithms (SODA), pp. 760–776 (2011)
17. Pilipczuk, M.: Problems parameterized by treewidth tractable in single exponential time: a logical approach. In: Murlak, F., Sankowski, P. (eds.) MFCS 2011. LNCS, vol. 6907, pp. 520–531. Springer, Heidelberg (2011)
18. Robertson, N., Seymour, P.D.: Graph minors. II. Algorithmic aspects of tree-width. J. Algorithms **7**(3), 309–322 (1986)
19. Rué, J., Sau, I., Thilikos, D.M.: Dynamic programming for graphs on surfaces. In: Short Version in the Proceeding of ICALP'10 in ACM Transactions on Algorithms (TALG). CoRR, abs/1104.2486 (2011)
20. Bhattacharya, B., Kameda, T.: A linear time algorithm for computing minmax regret 1-median on a tree. In: Gudmundsson, J., Mestre, J., Viglas, T. (eds.) COCOON 2012. LNCS, vol. 7434, pp. 1–12. Springer, Heidelberg (2012)
21. Scheffler, P.: A practical linear time algorithm for disjoint paths in graphs with bounded tree-width. Fachbereich 3 Mathematik, Technical Report 396/1994, FU Berlin (1994)