Long Paths Make Pattern-Counting Hard, and Deep Trees Make It Harder

Vít Jelínek
Computer Science Institute, Charles University, Prague, Czech Republic

Michal Opler
Computer Science Institute, Charles University, Prague, Czech Republic

Jakub Pekárek
Department of Applied Mathematics, Charles University, Prague, Czech Republic

Abstract

We study the counting problem known as #PPM, whose input is a pair of permutations π and τ (called *pattern* and *text*, respectively), and the task is to find the number of subsequences of τ that have the same relative order as π. A simple brute-force approach solves #PPM for a pattern of length k and a text of length n in time $O(n^{k+1})$, while Berendsohn, Kozma and Marx have recently shown that under the exponential time hypothesis (ETH), it cannot be solved in time $f(k)n^{o(k/\log k)}$ for any function f. In this paper, we consider the restriction of #PPM, known as $\mathcal{C}$-Pattern #PPM, where the pattern π must belong to a hereditary permutation class $\mathcal{C}$. Our goal is to identify the structural properties of $\mathcal{C}$ that determine the complexity of $\mathcal{C}$-Pattern #PPM.

We focus on two such structural properties, known as the *long path property* (LPP) and the *deep tree property* (DTP). Assuming ETH, we obtain these results:

1. If $\mathcal{C}$ has the LPP, then $\mathcal{C}$-Pattern #PPM cannot be solved in time $f(k)n^{o(\sqrt{k})}$ for any function f, and
2. if $\mathcal{C}$ has the DTP, then $\mathcal{C}$-Pattern #PPM cannot be solved in time $f(k)n^{o(k/\log^2 k)}$ for any function f.

Furthermore, when $\mathcal{C}$ is one of the so-called monotone grid classes, we show that if $\mathcal{C}$ has the LPP but not the DTP, then $\mathcal{C}$-Pattern #PPM can be solved in time $f(k)n^{O(\sqrt{k})}$. In particular, the lower bounds above are tight up to the polylog terms in the exponents.

2012 ACM Subject Classification Mathematics of computing → Permutations and combinations; Theory of computation → Pattern matching; Theory of computation → Problems, reductions and completeness

Keywords and phrases Permutation pattern matching, subexponential algorithm, conditional lower bounds, tree-width

Digital Object Identifier 10.4230/LIPIcs.IPEC.2021.22

Related Version *Full Version*: `https://arxiv.org/abs/2111.03479`

Funding *Vít Jelínek*: Supported by project 18-19158S of the Czech Science Foundation.
Michal Opler: Supported by project 21-32817S of the Czech Science Foundation and by project SVV–2020–260578.

1 Introduction

One of the most frequently studied algorithmic problems related to permutations is known as Permutation pattern matching (or PPM). The input of PPM is a pair of permutations τ (the "text") of length n and π (the "pattern") of length k, and the goal is to determine whether τ contains π as a subpermutation (see Section 2 for formal definitions).

In full generality, PPM is NP-complete, as shown by Bose et al. [4]. Thus most research into PPM focuses either on improved exact algorithms, or on identifying special types of inputs for which the PPM can be solved in polynomial time, or at least in subexponential time. Note that a direct brute-force approach solves PPM in time $O(n^{k+1})$.

16th International Symposium on Parameterized and Exact Computation (IPEC 2021).
Editors: Petr A. Golovach and Meirav Zehavi; Article No. 22; pp. 22:1–22:17
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

A particularly fruitful technique to solving PPM has been proposed by Ahal and Rabinovich [1], who showed that PPM can be solved in time $n^{O(\mathrm{tw}(\pi))}$, where $\mathrm{tw}(\pi)$ denotes the tree-width of the so-called *incidence graph* of the pattern π. The bound was subsequently tightened to $n^{\mathrm{tw}(\pi)+1}$ by Berendsohn, Kozma and Marx [3], who have used it to show that PPM can be solved in time $n^{k/4+o(k)}$.

Another approach to PPM, due to Guillemot and Marx [9] (with a slight improvement by Fox [8]) shows that the problem can be solved in time $n \cdot 2^{O(k^2)}$, implying that the problem is fixed-parameter tractable with parameter k.

Closely related to PPM is its counting version #PPM, whose goal is to compute the number of occurrences of the pattern π in the text τ. Berendsohn et al. [3] show that their bounds of $O(n^{\mathrm{tw}(\pi)+1})$ and $n^{k/4+o(k)}$ for PPM also apply to solving #PPM. In contrast, the FPT result for PPM by Guillemot and Marx [9] likely does not extend to #PPM, since Berendsohn et al. [3] show that, under the exponential time hypothesis (ETH), #PPM cannot be solved in time $f(k)n^{o(k/\log k)}$, for any function f.

Given that both PPM and #PPM are hard in general, it is natural to consider their complexity on restricted inputs. A common approach is to fix a hereditary class $\mathcal{C}$ of permutations, and study the restriction of PPM or #PPM to inputs where the pattern π belongs to $\mathcal{C}$. Such restriction is known as $\mathcal{C}$-Pattern PPM and $\mathcal{C}$-Pattern #PPM, respectively. It follows from the results of Ahal and Rabinovich [1] and Berendsohn et al. [3], that the restricted problems are polynomial whenever the function $\mathrm{tw}(\pi)$ is bounded on the class $\mathcal{C}$. This idea is the basis for previous results establishing sharp thresholds between polynomial and NP-hard cases of $\mathcal{C}$-Pattern PPM [11, 12]. In fact, in all the known cases when $\mathcal{C}$-Pattern PPM and $\mathcal{C}$-Pattern #PPM are polynomial, the class $\mathcal{C}$ has bounded tree-width.

While distinguishing the polynomial cases of $\mathcal{C}$-Pattern PPM from the NP-hard ones is obviously the main focus of research, it is also of interest to distinguish subexponential cases from those cases which (under suitable complexity assumptions, such as the ETH) require exponential or near-exponential time. Here again, the tree-width plays a key role. It is convenient to associate to a class $\mathcal{C}$ its *tree-width growth function*

$$\mathrm{tw}_{\mathcal{C}}(k) = \max\{\mathrm{tw}(\pi);\ \pi \in \mathcal{C} \wedge |\pi| = k\}.$$

Indeed, Berendsohn et al. [3], extending previous results by Guillemot and Vialette [10], have shown that when $\mathcal{C}$ is the class of 2-monotone permutations (i.e., the permutations merged from two monotone sequences), then $\mathrm{tw}_{\mathcal{C}}(k) = O(\sqrt{k})$, and consequently $\mathcal{C}$-Pattern #PPM can be solved in the subexponential time $n^{O(\sqrt{k})}$. They show, however, that for the class of 3-monotone permutations, the tree-width growth is of order $\Omega(k/\log k)$. Later Berendsohn [2, Theorem 4.1] showed that for the class $\mathcal{C} = \mathrm{Av}(654321)$, consisting of permutations that can be merged from 5 increasing subsequences, $\mathcal{C}$-Pattern #PPM cannot be solved in time $f(k)n^{o(k/\log^4 k)}$ for any function f, unless ETH fails.

In the context of $\mathcal{C}$-Pattern PPM and $\mathcal{C}$-Pattern #PPM, most of the research focuses on the cases when $\mathcal{C}$ is a *principal class*, i.e., the class $\mathrm{Av}(\sigma)$ of all the permutations that avoid a single forbidden pattern σ. Unfortunately, principal classes seldom admit a suitable structural characterisation of their elements, and even in those cases where such characterisations exist, they are very different from one class to another. This makes it hard to obtain general results that apply uniformly to a large set of principal classes.

To sidestep this issue, we mostly avoid dealing with individual principal classes directly, and instead we primarily focus on a different type of permutation classes, the so-called monotone grid classes. We then consider two structural properties of a general permutation class $\mathcal{C}$, called the *long path property* (LPP) and the *deep tree property* (DTP). Both these properties can be viewed as stating that $\mathcal{C}$ contains monotone grid subclasses of a particular type. We establish lower bounds for the complexity of $\mathcal{C}$-Pattern #PPM applicable to any class $\mathcal{C}$ with LPP or DTP. The definitions of LPP and DTP are somewhat technical (see Section 3); however, it is usually not too hard to verify whether a given class has these properties. Indeed, we are able to identify all the principal classes that have LPP, as well as all those that have DTP; see Subsection 3.2.

The LPP has already played a central part in a dichotomy result of the authors [12], and implicitly also in the work of Berendsohn [2] and Berendsohn et al. [3]. These previous results imply that for a monotone grid class $\mathcal{C}$ these properties are equivalent (assuming $P \neq NP$): (i) $\mathcal{C}$ has LPP, (ii) $\mathrm{tw}_{\mathcal{C}}(k)$ is unbounded, (iii) $\mathrm{tw}_{\mathcal{C}}(k) = \Omega(\sqrt{k})$, and (iv) $\mathcal{C}$-Pattern PPM is NP-complete. For all we know, the equivalence might hold for an arbitrary hereditary class $\mathcal{C}$, i.e., not just a monotone grid class. However, we do not even know whether every class of unbounded tree-width has LPP.

The DTP is a strengthening of LPP, which we introduce in this paper, with the aim of distinguishing the cases of $\mathcal{C}$-Pattern #PPM that can be solved in the subexponential time $f(k)n^{O(\sqrt{k})}$ from those that cannot be solved in time $f(k)n^{o(k/\log k)}$. While LPP forces tree-width growth of order $\Omega(\sqrt{k})$, DTP forces tree-width growth of order $\Omega(k/\log k)$.

Our main results show that the lower bounds on tree-width imposed by LPP and DTP are accompanied by the corresponding complexity lower bounds for $\mathcal{C}$-Pattern #PPM. Specifically, we show that under ETH, the following holds for any permutation class $\mathcal{C}$ (see Theorem 18):

- If $\mathcal{C}$ has the LPP, then $\mathcal{C}$-Pattern #PPM cannot be solved in time $f(k)n^{o(\sqrt{k})}$ for any function f, and
- if $\mathcal{C}$ has the DTP, then $\mathcal{C}$-Pattern #PPM cannot be solved in time $f(k)n^{o(k/\log^2 k)}$ for any function f.

In addition, we show that for classes with LPP, the Ahal–Rabinovich PPM algorithm with complexity $n^{O(\mathrm{tw}(\pi))}$ is asymptotically optimal. More precisely, we show that if ETH holds, then for a class $\mathcal{C}$ with LPP, no algorithm may solve $\mathcal{C}$-Pattern PPM in time $f(t)n^{o(t)}$ for any function f, where $t = \mathrm{tw}(\pi)$ (see Theorem 15). All these complexity lower-bounds are presented in Section 4.

Recall that by a result of Berendsohn et al. [3], the class $\mathcal{C} = \mathrm{Av}(321)$ has tree-width growth $\mathrm{tw}_{\mathcal{C}}(k) = O(\sqrt{k})$, and therefore $\mathcal{C}$-Pattern #PPM can be solved in time $n^{O(\sqrt{k})}$. It turns out that this class has LPP, which implies, by our results above, that $\mathrm{tw}_{\mathcal{C}}(k) = \Omega(\sqrt{k})$ and that $\mathcal{C}$-Pattern #PPM cannot be solved in time $f(k)n^{o(\sqrt{k})}$ for any function f. In particular, both the tree-width bound and the complexity bound are tight.

For any class $\mathcal{C}$ with DTP, the tree-width lower bound $\Omega(k/\log k)$ and the complexity lower-bound $f(k)n^{o(k/\log^2 k)}$ both match, up to the logarithmic terms, the trivial upper bounds of k and $n^{O(k)}$, respectively.

As we mentioned before, we mostly focus on monotone grid classes. We will show that for a monotone grid class $\mathcal{C}$, both LPP and DTP can be easily characterised in terms of a certain graph associated to a monotone grid class $\mathcal{C}$, called the cell graph, and that these two properties asymptotically determine $\mathrm{tw}_{\mathcal{C}}(\cdot)$. An earlier paper of the authors [12] shows that a monotone grid class has bounded tree-width (and hence neither LPP nor DTP) if and only if its cell graph is acyclic. We extend this result as follows (see Corollary 7):

- If the cell graph of a monotone grid class $\mathcal{C}$ is not acyclic but has at most one cycle in each component, then $\mathcal{C}$ has LPP but not DTP, and $\mathrm{tw}_{\mathcal{C}}(k) \in \Theta(\sqrt{k})$.
- If the cell graph of a monotone grid class $\mathcal{C}$ has a component with at least two cycles, then $\mathcal{C}$ has DTP and $\mathrm{tw}_{\mathcal{C}}(k) \in \Omega(k/\log k)$.

2 Preliminaries

A *permutation of length* n is a sequence in which each element of the set $[n] = \{1, 2, \ldots, n\}$ appears exactly once. When writing out short permutations explicitly, we shall omit all punctuation and write, e.g., 15342 for the permutation $1, 5, 3, 4, 2$. The *permutation diagram* of π is the set of points $S_\pi = \{(i, \pi_i);\ i \in [n]\}$ in the plane. Observe that no two points from S_π share the same x- or y-coordinate. We say that such a set is in *general position*.

For a point p in the plane, we denote its horizontal coordinate as $p.x$, and its vertical coordinate as $p.y$. Two finite sets $S, R \subseteq \mathbb{R}^2$ in general position are *isomorphic* if there is a bijection $f\colon S \to R$ such that for any pair of points $p \neq q$ of S we have $f(p).x < f(q).x$ if and only if $p.x < q.x$, and $f(p).y < f(p).y$ if and only if $p.y < q.y$. The *reduction* of a finite set $S \subseteq \mathbb{R}^2$ in general position is the unique permutation π such that S is isomorphic to S_π. We write $\pi = \mathrm{red}(S)$.

We say that a permutation τ *contains* a permutation π, written $\pi \leq \tau$, if the diagram of τ contains a subset that is isomorphic to the diagram of π. If τ does not contain π, we say that it *avoids* π. A *permutation class* is a set $\mathcal{C}$ of permutations which is *hereditary*, i.e., for every $\sigma \in \mathcal{C}$ and every $\pi \leq \sigma$, we have $\pi \in \mathcal{C}$. For a permutation π, we let $\mathrm{Av}(\pi)$ denote the set of all the permutations that avoid π; this is clearly a permutation class. The class $\mathrm{Av}(21)$ of all the increasing permutations and the class $\mathrm{Av}(12)$ of all the decreasing permutations are denoted by the symbols ⧄ and ⧅, respectively.

We will frequently refer to symmetries that transform permutations into other permutations. For our purposes, it is convenient to describe these symmetries geometrically, as transformations of the plane acting on permutation diagrams. We define the *m-box* to be the set $(\frac{1}{2}, m + \frac{1}{2}) \times (\frac{1}{2}, m + \frac{1}{2})$. Observe that for every permutation π of length at most m, the permutation diagram S_π is a subset of the m-box. We view permutation symmetries as bijections acting of the whole m-box. There are eight such symmetries, generated by:

reversal which reflects the m-box horizontally, i.e. the image of point p is $(m + 1 - p.x, p.y)$,

complement which reflects the m-box vertically, i.e. the image of point p is $(p.x, m+1-p.y)$,

inverse which reflects the m-box through its main diagonal, i.e. the image of point p is $(p.y, p.x)$.

In particular, the reversal of a permutation $\pi = \pi_1, \ldots, \pi_n$ is the permutation $\pi^r = \pi_n \pi_{n-1}, \ldots, \pi_1$, the complement of π is the permutation $\pi^c = n + 1 - \pi_1, n + 1 - \pi_2, \ldots, n + 1 - \pi_n$, and the inverse π^{-1} is the permutation $\sigma = \sigma_1, \ldots, \sigma_n$ such that $\sigma_i = j \iff \pi_j = i$. We also apply these symmetries to sets of permutations, in an obvious way: if Ψ is one of the eight symmetries defined above and $\mathcal{C}$ is a permutation class, we define $\Psi(\mathcal{C})$ as $\{\Psi(\pi);\ \pi \in \mathcal{C}\}$.

The incidence graph G_π of a permutation $\pi = \pi_1, \ldots, \pi_n$ is the graph whose vertices are the n entries $\pi_1, \ldots, \pi_n$, with two entries π_i and π_j connected by an edge if $|i - j| = 1$ or $|\pi_i - \pi_j| = 1$. In particular, the graph G_π is a union of two paths, one of them visiting the entries of π in left-to-right order, and the other in top-to-bottom order. We let $\mathrm{tw}(\pi)$ denote the tree-width of G_π.

Monotone grid classes

An important type of permutation classes are the so-called monotone grid-classes, which we now define. A *gridding matrix of size* $k \times \ell$ is a matrix $\mathcal{M}$ with k columns and ℓ rows, whose every entry is a permutation class. A *monotone gridding matrix* is a gridding matrix whose every entry is one of the three classes $\emptyset$, ⧄ or ⧅. Note that to be consistent with the Cartesian coordinates that we use to describe permutation diagrams, we will number the

$$\mathcal{M} = \begin{pmatrix} \mathrm{Av}(12) & & \mathrm{Av}(21) \\ & \mathrm{Av}(21) & \mathrm{Av}(12) \end{pmatrix}$$

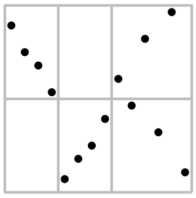

Figure 1 A monotone gridding matrix $\mathcal{M}$ on the left and a permutation equipped with an $\mathcal{M}$-gridding on the right. Empty entries of $\mathcal{M}$ are omitted and the edges of $G_{\mathcal{M}}$ are drawn in $\mathcal{M}$.

rows of a matrix from bottom to top, and we give the column coordinate as the first one. In particular, $\mathcal{M}_{i,j}$ denotes the entry in column i and row j of the matrix $\mathcal{M}$, with $1 \le i \le k$ and $1 \le j \le \ell$.

Let π be a permutation of length n. A $(k \times \ell)$-*gridding* of π is a pair of weakly increasing sequences $1 = c_1 \le c_2 \le \cdots \le c_{k+1} = n+1$ and $1 = r_1 \le r_2 \le \cdots \le r_{\ell+1} = n+1$. For $i \in [k]$ and $j \in [\ell]$, the (i,j)-*cell* of the gridding of π is the set of points $p \in S_\pi$ satisfying $c_i \le p.x < c_{i+1}$ and $r_j \le p.y < r_{j+1}$. Note that each point of the diagram S_π belongs to a unique cell of the gridding. A permutation π together with a gridding (c, r) forms a *gridded permutation*.

Let $\mathcal{M}$ be a gridding matrix of size $k \times \ell$. We say that the gridding of π is an $\mathcal{M}$-*gridding* if for every $i \in [k]$ and $j \in [\ell]$, the subpermutation of π induced by the points in the (i,j)-cell of the gridding of π belongs to the class $\mathcal{M}_{i,j}$.

We let $\mathrm{Grid}(\mathcal{M})$ denote the set of permutations that admit an $\mathcal{M}$-gridding. This is clearly a permutation class. A *monotone grid class* is any permutation class $\mathrm{Grid}(\mathcal{M})$ for a monotone gridding matrix $\mathcal{M}$.

The *cell graph* of a gridding matrix $\mathcal{M}$, denoted $G_{\mathcal{M}}$, is the graph whose vertices are all the pairs (i,j) for which $\mathcal{M}_{i,j}$ is an infinite permutation class. Two vertices are adjacent if they appear in the same row or the same column of $\mathcal{M}$, and there is no other cell containing an infinite class between them. See Figure 1. A *proper-turning path* in $G_{\mathcal{M}}$ is a path P such that no three vertices of P share the same row or column.

Grid transforms and orientations

Let π be a permutation of length n with a $(k \times \ell)$-gridding (c, r), where $c = (c_1, \ldots, c_{k+1})$ and $r = (r_1, \ldots, r_{\ell+1})$. The *reversal of the i-th column* of π is the operation that transforms π into a new permutation π' by taking the rectangle $[c_i, c_{i+1} - 1] \times [1, n]$ and flipping it along its vertical axis, thus producing the diagram of a new permutation π'. Equivalently, π' is created from π by reversing the order of the entries of π at positions $c_i, c_i + 1, \ldots, c_{i+1} - 1$. We view π' as a gridded permutation, with the same gridding (c, r) as π.

Similarly, the *complementation of the j-th row* transforms the diagram of π by flipping the rectangle $[1, n] \times [r_j, r_{j+1} - 1]$ along its horizontal axis, producing the diagram of a new gridded permutation π'.

We may similarly apply reversals to the columns of a gridding matrix $\mathcal{M}$ and complements to its rows. Reversing the i-th column of $\mathcal{M}$ produces a new gridding matrix, in which all the classes in the i-th column of $\mathcal{M}$ are replaced by their reversals. Row complementation of a gridding matrix is defined analogously. Note that a column reversal or a row complementation in a gridded permutation or in a gridding matrix is an involution, i.e., repeating the same operation twice restores the original permutation or matrix. Note also that when we perform a sequence of column reversals and row complementations, then the end result does not depend on the order in which the operations were performed.

To describe succinctly a sequence of row and column operations, we introduce the notion of $(k \times \ell)$-*orientation*, which is a pair of functions $\mathcal{F} = (f_c, f_r)$ with $f_c\colon [k] \to \{-1, 1\}$ and $f_r\colon [\ell] \to \{-1, 1\}$. *Applying* the orientation $\mathcal{F}$ to a $(k \times \ell)$-gridded permutation π produces a new gridded permutation $\mathcal{F}(\pi)$ with the same gridding as π, obtained by reversing each column i such that $f_c(i) = -1$ and complementing each row j such that $f_r(j) = -1$. The application of $\mathcal{F}$ to a gridding matrix $\mathcal{M}$ is defined analogously, and produces a gridding matrix denoted $\mathcal{F}(\mathcal{M})$. Note that (c, r) is an $\mathcal{M}$-gridding of π if and only if it is an $\mathcal{F}(\mathcal{M})$-gridding of $\mathcal{F}(\pi)$.

An orientation $\mathcal{F}$ is a *consistent orientation* of a monotone gridding matrix $\mathcal{M}$, if every nonempty entry of $\mathcal{F}(\mathcal{M})$ is equal to ⧄. As an example, the matrix $\begin{pmatrix} ⧅ & ⧄ \\ ⧄ & ⧅ \end{pmatrix}$ has a consistent orientation acting by reversing the first column and complementing the first row. On the other hand, the matrix $\begin{pmatrix} ⧅ & ⧄ \\ ⧄ & ⧄ \end{pmatrix}$ has no consistent orientation, since applying any orientation to this matrix yields a matrix with an odd number of ⧅-entries.

The following lemma, due to Vatter and Waton [15], will be later useful.

▶ **Lemma 1.** *Every monotone gridding matrix whose cell graph is acyclic has a consistent orientation.*

Tile assembly

In the hardness reductions that we are about to present, we frequently need to construct permutations whose diagrams have a natural $k \times \ell$ grid-like structure. We describe such a diagram by taking each cell individually and describing the points inside it. For such a description, it is often convenient to assume that each cell has its own coordinate system whose origin is near the bottom-left corner of the cell. This allows us to describe the coordinates of the points inside the cell without referring to the position of the cell within the whole permutation diagram. In effect, we describe the diagram of the gridded permutation by first constructing a set of independent "tiles" $\mathcal{T}_{i,j}$ for $i \in [k]$ and $j \in [\ell]$ of the same size, and then translating each tile $\mathcal{T}_{i,j}$ to column i and row j of the diagram. On top of that, we often need to apply an orientation to the gridded permutation whose diagram we constructed.

We now describe the whole procedure more formally. Fix an integer m and recall that an m-box is a square of the form $(\frac{1}{2}, m + \frac{1}{2}) \times (\frac{1}{2}, m + \frac{1}{2})$. An m-tile is a finite set of points inside the m-box. Note that the coordinates of the points in the tile may not be integers. A $(k \times \ell)$-family of m-tiles is a collection $(\mathcal{T}_{i,j};\ i \in [k], j \in [\ell])$ where each $\mathcal{T}_{i,j}$ is an M-tile. Let $\mathcal{F}$ be a $(k \times \ell)$-orientation. The $\mathcal{F}$-*assembly* of the family $(\mathcal{T}_{i,j};\ i \in [k], j \in [\ell])$ is the gridded permutation obtained as follows.

First, we translate each tile $\mathcal{T}_{i,j}$ by adding $m(i-1)$ to each horizontal coordinate and $m(j-1)$ to each vertical coordinate. Thus, the m-tiles will be disjoint. If the union of the translated tiles is not in general position, we rotate it slightly clockwise to reach general position. Notice that we can do so without changing the relative position of any pair of points that were already in general position. This yields a point set isomorphic to a unique permutation π. See Figure 2. Additionally, π has a natural gridding whose cells correspond to the translated tiles. To finish the construction, we apply the orientation $\mathcal{F}$ to π, obtaining the gridded permutation $\mathcal{F}(\pi)$, which is the $\mathcal{F}$-assembly of the family of tiles $(\mathcal{T}_{i,j};\ i \in [k], j \in [\ell])$.

▶ **Observation 2.** *Let $(\mathcal{T}_{i,j};\ i \in [k], j \in [\ell])$ be a family of tiles, let $\mathcal{F}$ be an orientation, and let $\mathcal{M}$ be a gridding matrix such that $\mathcal{T}_{i,j}$ is isomorphic to a permutation from the class $\mathcal{M}_{i,j}$. Then the $\mathcal{F}$-assembly of the family of tiles $(\mathcal{T}_{i,j};\ i \in [k], j \in [\ell])$ is a permutation from the class Grid$(\mathcal{F}(\mathcal{M}))$.*

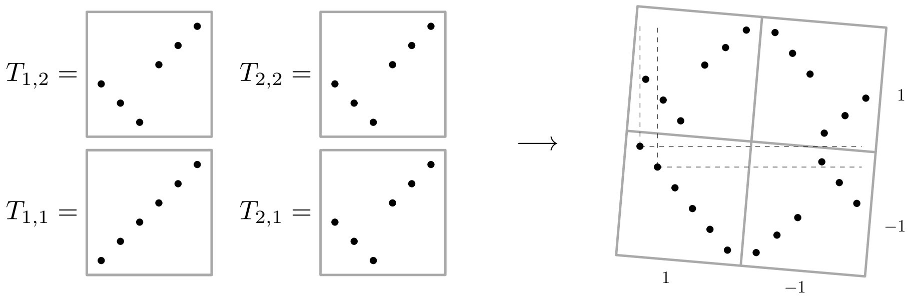

Figure 2 A 2×2 family of tiles $\mathcal{T}$ on the left and its $\mathcal{F}$-assembly on the right for a 2×2 orientation $\mathcal{F}$ given next to each row and column on the right. General position is attained by rotating the resulting point set clockwise. The dashed lines indicate relative positions of two particular points.

3 Tree-width bounds

3.1 Width of monotone grid classes

We say that a permutation class $\mathcal{C}$ has the *long path property* (LPP) if for every k the class $\mathcal{C}$ contains a monotone grid subclass whose cell graph is a path of length k. The next proposition builds upon the ideas of Berendsohn et al. [3], who proved a similar result for the class Av(321) using the fact that this class contains a staircase-shaped grid path of arbitrary length.

▶ **Proposition 3.** *If a permutation class $\mathcal{C}$ has the LPP then* $\mathrm{tw}_{\mathcal{C}}(n) \in \Omega(\sqrt{n})$.

Proof. First, we show that $\mathcal{C}$ contains for every k a grid subclass whose cell graph is a proper-turning path of length k, i.e. a path in which no three consecutive vertices are in the same row or column of the gridding. For the contrary, assume that there is ℓ such that $\mathcal{C}$ does not contain such path of length ℓ. The LPP then implies that $\mathcal{C}$ contains for every t a class $\mathrm{Grid}(\mathcal{M})$ where $\mathcal{M}$ is either a $1 \times t$ or $t \times 1$ matrix without empty entries. However, any such matrix of dimensions $1 \times n$ or $n \times 1$ contains all permutations of length n and thus, $\mathcal{C}$ must actually be the class of all permutations that contains all possible proper turning paths.

So we can suppose that there is a monotone gridding matrix $\mathcal{M}$ such that $\mathcal{M}$ is a proper-turning path $v_1, \dots, v_{2m-1}$ of length $2m - 1$ and $\mathrm{Grid}(\mathcal{M})$ is contained in $\mathcal{C}$. We explicitly construct a permutation $\pi \in \mathrm{Grid}(\mathcal{M})$ such that G_π contains an $m \times m$ grid graph as a subgraph. The claim then follows since the tree-width of $m \times m$ grid graph is exactly m.

For $i \in [m]$ and $j \in [i]$, let

$$p_{i,j} = (m + 2j - i - 1, m + 2j - i - 1), \quad p_{2m-i,j} = p_{i,j}.$$

We define a family of $2m$-tiles $\mathcal{P}$ by setting $\mathcal{P}_{v_i}$ to be the set of points $p_{i,j}$ for all possible choices of j.

Let $\mathcal{F}$ be a consistent orientation of $\mathcal{M}$ guaranteed by Lemma 1 and let π be the $\mathcal{F}$-assembly of $\mathcal{P}$. The sets P_{v_i} were defined in such a way that for every i the points in $P_{v_{2i}}$ have both coordinates odd whereas the points in $P_{v_{2i+1}}$ have both coordinates even. Since $\mathcal{M}$ is a proper turning path, there are always at most two non-empty tiles sharing the same row or column in π and in such case they correspond to neighboring vertices of the path. Moreover, if they share a common row, then the y-coordinates of their points are interleaved, and if they share a common column, the same holds for the x-coordinates.

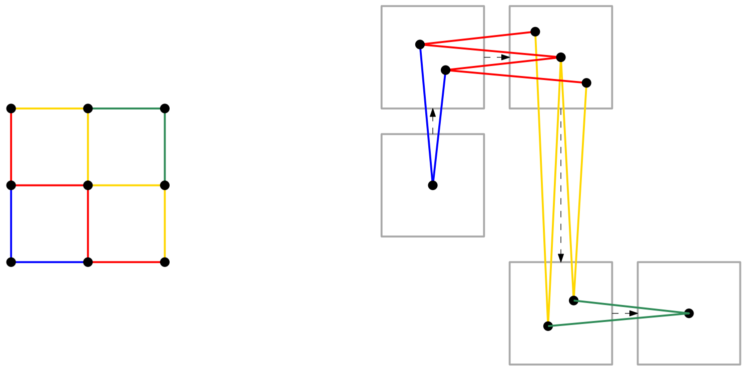

Figure 3 Illustration of the proof of Proposition 3. Embedding a 3×3 grid graph (left) into a permutation from a monotone grid class whose cell graph is a path of length 5 (right).

It remains to show that G_π contains an $m \times m$ grid graph as a subgraph. Let $s_{i,j}$ be the image of $p_{i,j}$ under the $\mathcal{F}$-assembly. We claim that we can map consecutive diagonals of the grid to the tiles P_{v_i}. See Figure 3. More precisely, for $x, y \in [m]$ set

$$g_{x,y} = \begin{cases} s_{x+y-1,x} & \text{if } x + y \leq m + 1, \\ s_{x+y-1,m-y+1} & \text{otherwise.} \end{cases}$$

We start by showing that for any $i \in [m-1]$, there is an edge between $s_{i,j}$ and $s_{i+1,j}$, and also between $s_{i,j}$ and $s_{i+1,j+1}$. This follows since the points of P_{v_i} and $P_{v_{i+1}}$ have their x- or y-coordinates interleaved and there is no other tile occupying their shared row or column. Due to symmetry, it holds that for $i > m$, there is an edge between $s_{i,j}$ and $s_{i-1,j}$ and also between $s_{i,j}$ and $s_{i-1,j+1}$.

If we take $x, y \in [m]$ such that $x + y \leq m$ (i.e. $g_{x,y}$ lies below the anti-diagonal of the grid), the fact proved in the previous paragraph directly translates to the existence of edges between $g_{x,y}$ and $g_{x+1,y}$ and between $g_{x,y}$ and $g_{x,y+1}$. On the other hand for $x, y \in [m]$ such that $x + y \geq m + 2$, the points $g_{x,y}$, $g_{x-1,y}$ and $g_{x,y-1}$ translate to $s_{x+y-1,x}$, $s_{x+y-2,x-1}$ and $s_{x+y-2,x}$. Therefore, in this case there are edges between $g_{x,y}$ and $g_{x-1,y}$ and between $g_{x,y}$ and $g_{x,y-1}$. This concludes the proof as any edge in the $m \times m$ grid graph is of one of the two types whose existence we proved. ◀

It turns out that there is a large family of monotone grid classes for which $\mathrm{tw}_\mathcal{C} \in \Theta(\sqrt{n})$, namely every monotone grid class whose cell graph is not acyclic yet it does not contain two connected cycles. We include the complete proof in the full version, and here we only briefly describe its main ideas.

▶ **Theorem 4.** *If $\mathcal{M}$ is a connected monotone gridding matrix that contains a single cycle in its cell graph then $\mathrm{tw}_{Grid(\mathcal{M})}(n) \in \Theta(\sqrt{n})$*

Proof idea. First, we show that $\mathrm{tw}_{\mathrm{Grid}(\mathcal{M})}(n) \in \Omega(\sqrt{n})$. It has been previously proved by the authors in [12, Lemma 3.5] that a cycle in a grid class implies the LPP. The lower bound readily follows from Proposition 3.

For the upper bound, let π be a permutation of $\mathrm{Grid}(\mathcal{M})$ with a given $\mathcal{M}$-gridding. We observe that there is only $O(1)$ edges whose endpoints share neither a common row nor a common column. Therefore, we can focus on the graph G' obtained from G_π by removing these edges. We subsequently show that G' can be drawn on a surface of Euler genus 1 with $O(n)$ total crossings. Standard techniques [7] then imply that $\mathrm{tw}(G') \in O(\sqrt{n})$.

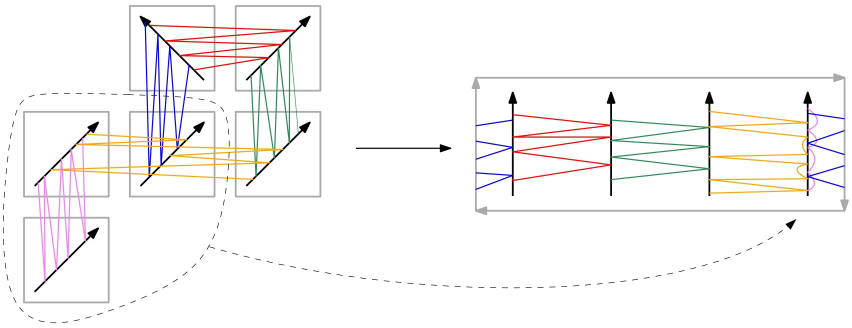

Figure 4 A schematic drawing of G_π for π from a unicyclic grid class on the projective plane. Instead of drawing the specific points of π, we place arrows to indicate the orientation of each cell. Different color is used for each set of edges that share a single row or column, and the exceptional edges are omitted.

Suppose that $c_1, c_2, \ldots, c_m$ are the entries of $\mathcal{M}$ that lie on its only cycle in this order. The cell graph $G_\mathcal{M}$ consists of the cycle and trees that are attached to it. If we remove all the edges that participate in the cycle, we end up with m trees $T_1, \ldots, T_m$ called *tendrils* such that the tree T_i contains the entry c_i.

We prove that the points of a single tendril can be drawn on a straight segment in a way such that the points from different cells are ordered consistently, and moreover, there are only $O(n)$ crossings between edges going inside a single tendril. We use this to draw each tendril on a parallel line, called *meridian*, and subsequently, we draw the edges connecting points in different tendrils as polylines that do not cross each other. We are forced to add one crosscap between some pair of meridians if $\mathcal{M}$ does not admit a consistent orientation. Finally, we check that we produced at most $O(n)$ crossing between the edges whose endpoints occupy a single tendril and the edges connecting two different tendrils. See Figure 4. ◀

For integer constants c and d, a *c-subdivided binary tree of depth d* is a graph obtained from a binary tree of depth d by replacing every edge by a path of length at most c. We say that a permutation class $\mathcal{C}$ has the *deep tree property* (DTP) if there is a constant c such that for every d, the class $\mathcal{C}$ contains a monotone grid subclass whose cell graph is a c-subdivided binary tree of depth d. Observe that DTP straightforwardly implies LPP. We say that a class $\mathcal{C}$ has *near-linear width* if $\mathrm{tw}_\mathcal{C}(n) \in \Omega(n/\log n)$.

▶ **Proposition 5.** *If a permutation class $\mathcal{C}$ has the DTP, then it has near-linear width.*

Proof. Inspired by the approach of Berendsohn [3], we want to show that for a graph G of large tree-width, we can find a permutation $\sigma \in \mathcal{C}$ such that G_σ contains G as a minor while the length of σ exceeds the size of G by at most a logarithmic factor.

To that end, fix an arbitrary graph G with vertex set $V_G = [n]$ and edges $\{e_1, \ldots, e_m\}$ where $e_i = \{a_i, b_i\}$. Let $\mathcal{M}$ be a monotone gridding matrix such that $\mathrm{Grid}(\mathcal{M}) \subseteq \mathcal{C}$ and the cell graph of $\mathcal{M}$ is a c-subdivided binary tree with exactly m leaves. Let r denote the root of this tree. It follows that the tree has maximal depth at most $c(\log m + 1)$. We turn $G_\mathcal{M}$ into an oriented graph by orienting all edges consistently away from r. For any vertex v of the tree, the *descendants of* v, denoted by $D(v)$, are all the out-neighbors of v.

We assign a set $A_w \subseteq V_G$ to each vertex w of the tree. First, we arbitrarily order the m leaves of $G_\mathcal{M}$ as $v_1, \ldots, v_m$. Then we inductively define

$$A_w = \begin{cases} \{a_i, b_i\} & \text{if } w = v_i \text{ for } i \in [m] \text{ where } e_i = \{a_i, b_i\}, \\ \bigcup_{v \in D(w)} A_v & \text{otherwise.} \end{cases} \tag{1}$$

We remark that $\sum_v |A_v| = O(m \log m)$ since each vertex $i \in V_G$ is present in exactly $\deg(i)$ leaves and in the paths of length $O(\log m)$ that connect those leaves to r. We proceed to define a family of m-tiles $\mathcal{P}$ by setting $P_v = \{(i,i) \mid i \in A_v\}$ for every vertex v of the tree, and keeping all the other tiles empty.

Let $\mathcal{F}$ be a consistent orientation obtained from the application of Lemma 1 on $\mathcal{M}$ and let π be the $\mathcal{F}$-assembly of $\mathcal{P}$. Since every tile is an increasing point set, it follows that π belongs to $\mathrm{Grid}(\mathcal{M})$.

In order to simplify the rest of the proof, we color S with n colors. We assign a color $i \in V_G$ to a point $p \in S$ with preimage (i,i) in $P_{x,y}$. We claim that S satisfies the following conditions:

(a) The subgraph of G_π induced by a single color is connected;

(b) For each edge $e_i = \{a_i, b_i\}$ of G there is an edge in G_π between a vertex of color a_i and a vertex of color b_i.

Fix a color $i \in V_G$. Let Q_i be the set of all vertices v of $G_\mathcal{M}$ such that $i \in A_v$. Clearly, Q_i induces a connected subtree of $G_\mathcal{M}$. Recall that every point of color i has always the coordinates (i,i) inside any tile. It follows that for points (i,i) in two neighboring tiles, the $\mathcal{F}$-assembly of $\mathcal{P}$ transforms them first to points that share one coordinate and then by rotating slightly clockwise makes them either horizontal or vertical neighbors. Therefore, the subgraph of G_π induced by color i is connected, which proves a.

Every leaf v_i must be the only non-empty vertex in its row or column. Let us assume the latter case as the other one is symmetric. Therefore, the two points contained in the image of P_{v_i} form an edge in G_π since no other point lies in the vertical strip between them. In particular, the leaf v_i satisfies the condition b for edge e_i.

The conditions a and b together imply that we can obtain a supergraph of G by contracting every monochromatic subgraph of G_π to a single vertex and thus, G is a minor of G_π. Observe that the total size of π is equal to $\sum_v |A_v|$ which we showed to be $O(m \log m)$. And since there exist graphs on n vertices with $O(n)$ edges and tree-width $\Omega(n)$, we deduce that $\mathrm{tw}_C(n) \geq \mathrm{tw}_{\mathrm{Grid}(\mathcal{M})}(n) \in \Omega(n/\log n)$. ◀

We continue by introducing a different property that implies the deep tree property and is at the same time easier to show for a specific class $\mathcal{C}$. A permutation class $\mathcal{C}$ has the *bicycle property* if it contains a monotone grid subclass whose cell graph is connected and contains at least two cycles. We include the full, rather technical, proof in the full version, providing here with only a brief sketch.

▶ **Proposition 6.** *If a permutation class $\mathcal{C}$ has the bicycle property, then it also has the DTP (and therefore near-linear width).*

Proof idea. The proof consists of two parts. First, we will show that there is always a grid subclass of $\mathcal{C}$ that contains in its cell graph two connected cycles of a certain special type. To that end, observe that the cell graph $G_\mathcal{M}$ can either be two cycles connected by a path or one cycle with a chord. For the latter case, we show that by replacing each entry in $\mathcal{M}$ with a suitable 3×3 matrix, we obtain a matrix $\mathcal{N}$ such that $\mathrm{Grid}(\mathcal{N})$ is a subclass of $\mathrm{Grid}(\mathcal{M})$ and moreover, the cell graph $G_\mathcal{N}$ contains two cycles joined with a path. See Figure 5.

In the second step, we find a way to wind a c-subdivided binary tree of arbitrary depth into the two cycles joined with a path and thus, showing that $\mathcal{C}$ has the DTP. ◀

For monotone grid classes, the results of this section imply a sharp dichotomy.

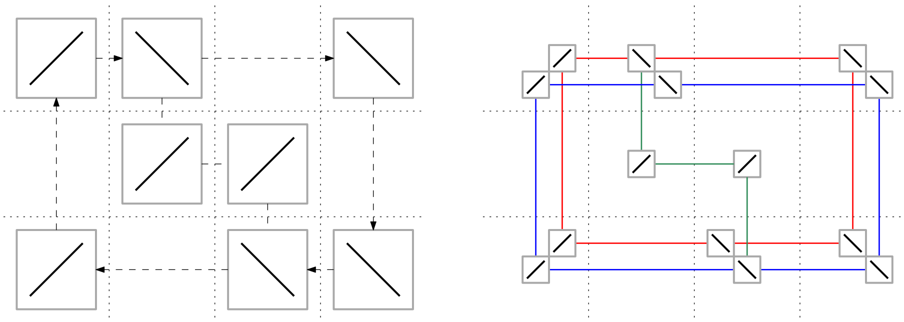

Figure 5 Left: a gridding matrix $\mathcal{M}$ whose cell graph is a cycle with a chord. Right: a gridding matrix $\mathcal{N}$ such that $\mathrm{Grid}(\mathcal{N})$ is contained in $\mathrm{Grid}(\mathcal{M})$ and the cell graph $G_{\mathcal{N}}$ consists of two cycles joined by a path.

▶ **Corollary 7.** *For a monotone grid class Grid($\mathcal{M}$) exactly one of the following holds.*

- $G_{\mathcal{M}}$ *is acyclic and* $\mathrm{tw}_{\mathcal{C}}(k) \in \Theta(1)$.
- $G_{\mathcal{M}}$ *contains at most one cycle in each component,* $\mathcal{C}$ *has LPP and* $\mathrm{tw}_{\mathcal{C}}(k) \in \Theta(\sqrt{k})$.
- $G_{\mathcal{M}}$ *has a component with at least two cycles,* $\mathcal{C}$ *has DTP and* $\mathrm{tw}_{\mathcal{C}}(k) \in \Omega(k/\log k)$.

3.2 The case of principal classes

In this section, we investigate the long path and deep tree properties of principal classes, i.e., the classes of the form $\mathrm{Av}(\pi)$. Combined with the results of Subsection 3.1, it allows us to infer lower bounds for the tree-width growth function of $\mathrm{Av}(\sigma)$. Whereas together with the results of Section 4, we obtain conditional lower bounds for counting patterns from $\mathrm{Av}(\sigma)$. Let us note that previously Berendsohn [2] has shown that for any π of length at least 4 that is not symmetric to one of $\{3412, 3142, 4213, 4123, 42153, 41352, 42513\}$, the class $\mathrm{Av}(\pi)$ has near-linear width. We reproduce and improve this result in a concise way with the tools that we have built up.

The *k-step increasing $(\mathcal{C}, \mathcal{D})$-staircase*, denoted by $\mathrm{St}_k(\mathcal{C}, \mathcal{D})$ is a grid class $\mathrm{Grid}(\mathcal{M})$ of a $k \times (k+1)$ gridding matrix $\mathcal{M}$ such that the only non-empty entries in $\mathcal{M}$ are $\mathcal{M}_{i,i} = \mathcal{C}$ and $\mathcal{M}_{i,i+1} = \mathcal{D}$ for every $i \in [k]$. In other words, the entries on the main diagonal are equal to $\mathcal{C}$ and the entries of the adjacent lower diagonal are equal to $\mathcal{D}$. The *increasing $(\mathcal{C}, \mathcal{D})$-staircase*, denoted by $\mathrm{St}(\mathcal{C}, \mathcal{D})$, is the union of $\mathrm{St}_k(\mathcal{C}, \mathcal{D})$ over all $k \in \mathbb{N}$.

The authors [13] recently showed that $\mathrm{Av}(\sigma)$ contains a certain staircase class for three patterns of length 3 and certain 2×2 grid classes for four patterns of length 4. Moreover, at least one of these patterns or their symmetries is contained in every permutation of length at least 4 that is not symmetric to one of $3412, 3142, 4213, 4123$ or 41352.

▶ **Proposition 8** (Jelínek et al.[13]). *We have* $\mathrm{St}(\boxslash, \mathrm{Av}(321)) \subseteq \mathrm{Av}(4321)$, $\mathrm{St}(\boxslash, \mathrm{Av}(231)) \subseteq \mathrm{Av}(4231)$ *and* $\mathrm{St}(\boxslash, \mathrm{Av}(312)) \subseteq \mathrm{Av}(4312)$.

▶ **Proposition 9** (Jelínek et al.[13]). *The class* $\mathrm{Av}(\sigma)$ *contains the class Grid($\mathcal{M}$) for the gridding matrix* $\mathcal{M} = \begin{pmatrix} \boxbslash & \boxslash \\ \mathrm{Av}(\pi) & \boxbslash \end{pmatrix}$ *whenever*

- $\pi = 132$ *and* $\sigma = 14523$, *or*
- $\pi = 231$ *and* $\sigma = 24513$, *or*
- $\pi = 321$ *and* $\sigma \in \{32154, 42513\}$.

σ	LPP, DTP of $\mathrm{Av}(\sigma)$	Comment
1, 21, 312	neither LPP nor DTP	$\mathrm{tw}_{\mathrm{Av}(\sigma)} \in \Theta(1)$ by Ahal and Rabinovich [1] which would contradict Proposition 3.
321, 3412, 3142, 4213, 4123, 41352	LPP but not DTP	LPP of 321 and 3412 follows due to Jelínek and Kynčl [11], the rest contains either 123 or 321. The absence of DTP is proved in the full version.
All other	both LPP and DTP	DTP by Proposition 10, LPP follows.

Figure 6 The long path and deep tree properties of principal classes, i.e. classes of form $\mathrm{Av}(\sigma)$. Only one pattern σ from each symmetry group is listed.

▶ **Proposition 10.** *If σ is a permutation of length at least 4 that is not in symmetric to any of* $3412, 3142, 4213, 4123$ *or* 41352*, then* $\mathrm{Av}(\sigma)$ *has the bicycle property and thus,* $\mathrm{Av}(\sigma)$ *has near-linear width.*

Proof. We start by proving that every class defined by forbidding a pattern of length 3 must contain a special type of monotone grid subclass. For arbitrary π of length 3, the class $\mathrm{Av}(\pi)$ contains a grid class $\mathrm{Grid}(\mathcal{M})$ such that $\mathcal{M}$ is a 2×2 monotone gridding matrix with three non-empty entries. Since there are only two different symmetry types of permutations of length 3, it is enough to check that

$$\mathrm{Grid}\begin{pmatrix} \boxslash & \boxslash \\ \boxslash & \cdot \end{pmatrix} \subseteq \mathrm{Av}(321) \quad \text{and} \quad \mathrm{Grid}\begin{pmatrix} \boxbslash & \boxslash \\ \cdot & \boxbslash \end{pmatrix} \subseteq \mathrm{Av}(132).$$

First, we prove the claim for the patterns that appear in Proposition 8. Let $\sigma \in \{4321, 4231, 4312\}$ and take a 3-step increasing staircase $\mathrm{St}_3(\boxslash, \mathrm{Av}(\pi))$ for π of length 3 that is contained in $\mathrm{Av}(\sigma)$. Let $\mathcal{M}'$ be a 6×8 monotone gridding matrix obtained from $\mathrm{St}_3(\boxslash, \mathrm{Av}(\pi))$ by replacing every $\boxslash$-entry by the identity matrix $\begin{pmatrix} \cdot & \boxslash \\ \boxslash & \cdot \end{pmatrix}$ and every $\mathrm{Av}(\pi)$-entry with its 2×2 monotone grid subclass which has three non-empty entries. Clearly, $\mathrm{Grid}(\mathcal{M}')$ is a subclass of $\mathrm{Av}(\sigma)$, and it is easy to check that for any π, the cell graph of $\mathcal{M}'$ is connected and contains two cycles.

We prove the claim for the patterns that appear in Proposition 9 in a similar fashion. Let $\sigma \in \{14523, 24513, 32154, 42513\}$ and take $\mathcal{M}$ to be the grid class $\mathrm{Grid}\begin{pmatrix} \boxbslash & \boxslash \\ \mathrm{Av}(\pi) & \boxbslash \end{pmatrix}$ for π of length 3 that is contained in $\mathrm{Av}(\sigma)$. Similar to before, let $\mathcal{M}'$ be the gridding matrix obtained from $\mathcal{M}$ by replacing the $\boxslash$-entry with the matrix $\begin{pmatrix} \cdot & \boxslash \\ \boxslash & \cdot \end{pmatrix}$, both $\boxbslash$-entries with the matrix $\begin{pmatrix} \boxbslash & \cdot \\ \cdot & \boxbslash \end{pmatrix}$, and $\mathrm{Av}(\pi)$ with its 2×2 monotone grid subclass which has three non-empty entries. Again, $\mathrm{Grid}(\mathcal{M}')$ is a subclass of $\mathrm{Av}(\sigma)$, and it is easy to check that for any π, the cell graph of $\mathcal{M}'$ is connected and contains two cycles. ◀

We can actually show that the DTP cannot get us any further, since for any $\sigma \in \{3412, 3142, 4213, 4123, 41352\}$, the class $\mathrm{Av}(\sigma)$ does not have the DTP. See the full version for the whole discussion. Hereby, we actually obtained a complete knowledge of LPP and DTP for principal classes. See Figure 6.

4 Hardness of #PPM

In this section, we provide conditional lower bounds for modified variants of $\mathcal{C}$-PATTERN PPM given LPP or DTP. The results of this section are proved under a slightly stronger assumptions about the classes. Apart from the LPP or DTP property, we furthermore require an algorithm that provides a witnessing long path or deep tree. Formally, a class $\mathcal{C}$ has the *computable LPP* if it has the LPP and there is an algorithm that, for a given k, outputs the description of a monotone grid subclass of $\mathcal{C}$ whose cell graph is a path of length k. Similarly, a class $\mathcal{C}$ has the *computable DTP* if it has the DTP and there is an algorithm that, for a given k, outputs the description of a monotone grid subclass of $\mathcal{C}$ whose cell graph is a c-subdivided binary tree of depth k. Observe that all the specific examples of classes we encountered (and especially the principal classes in Subsection 3.2) posses the computable version of their corresponding properties.

We will reduce from the well-known problem *partitioned subgraph isomorphism* (PSI) defined as follows. We receive on input two graphs $G = (V_G, E_G)$ and $H = (V_H, E_H)$ together with a coloring $\chi\colon V_H \to V_G$ of vertices of H, using the vertices of G as colors. We have to decide if there is a mapping $\phi\colon V_G \to V_H$ such that whenever $\{u, v\} \in E_G$ then also $\{\phi(u), \phi(v)\} \in E_H$ and moreover $\chi(\phi(v)) = v$ for every $v \in V_G$. Less formally, we aim to find G as a subgraph of H, but we prescribe in advance where each vertex can be mapped to. It is a well-known fact that PSI is hard to solve.

▶ **Theorem 11** (Marx [14], Bringmann et al. [5])**.** *Unless ETH fails, PSI cannot be solved in time $f(k) \cdot n^{o(k/\log k)}$ for any function f, where $n = |V_H|$ and $k = |E_G|$. This is true even when we require G to have exactly as many vertices as edges.*

If we additionally fix G to be the clique on k vertices we obtain the problem called Partitioned Clique. Formally, the input to Partitioned Clique consists of a graph $H = (V_H, E_H)$ together with a coloring $\chi\colon V_H \to [k]$ and we have to decide if there is a k-clique in H that hits all k available colors. It is easy to see that Partitioned Clique can be solved in time $f(k) \cdot n^{O(k)}$. However, there is also a matching conditional lower bound.

▶ **Theorem 12** (Cygan et al. [6])**.** *Unless ETH fails, Partitioned Clique cannot be solved in time $f(k) \cdot n^{o(k)}$ for any function f, where $n = |V_H|$.*

We shall also not reduce directly to the problems of interest. Rather, we first reduce to the *$\mathcal{C}$-Pattern Anchored PPM* ($\mathcal{C}$-Pattern APPM) problem, defined as follows. The input consists of permutations $\pi \in \mathcal{C}$ and arbitrary τ together with pairs of points A in π and B in τ that are called *anchors*. We are promised that arbitrary inflation of the points in A with either two increasing or two decreasing sequences creates π' that is still contained in $\mathcal{C}$. The goal is to decide whether there is an embedding of π into τ that maps A to B.

For $\mathcal{C}$ with the computable LPP, we are able to reduce Partitioned Clique to $\mathcal{C}$-Pattern APPM such that the size of the pattern π is linear in the number of vertices of the clique. And for $\mathcal{C}$ with the computable DTP, we provide a reduction from PSI to $\mathcal{C}$-Pattern APPM such that the size of π is almost linear in the size of the graph G. Due to the space constraints and technicality of the reductions, we include here only brief overviews and describe both of them, including the proofs of correctness in the full version.

▶ **Lemma 13.** *Let $\mathcal{C}$ be a class with the computable LPP. An instance (G, χ) of Partitioned Clique can be reduced to an instance (π, τ, A, B) of $\mathcal{C}$-Pattern APPM where $|\pi| \in O(k^2)$ and $|\tau| \in O(|V_H|^2)$ in time $f(k) \cdot |V_H|^{O(1)}$ for some function f. Moreover,* $\mathrm{tw}(\pi) \in O(k)$*.*

Proof idea. Using the computable LPP, we obtain a monotone gridding matrix $\mathcal{M}$ such that $\mathrm{Grid}(\mathcal{M})$ is a subclass of $\mathcal{C}$ and the cell graph of $\mathcal{M}$ is a proper-turning path with $4k - 2$ vertices $v_1, \ldots, v_{4k-2}$. We construct the pattern π via an $\mathcal{F}$-assembly from a family of tiles $\mathcal{P}$ and the text τ from a family of tiles $\mathcal{T}$ where the only non-empty tiles in both families correspond to the non-empty entries of $\mathcal{M}$ and moreover, each non-empty tile in $\mathcal{P}$ is an increasing sequence.

The first tiles P_{v_1} and T_{v_1} both contain only pair of elements whose images under the $\mathcal{F}$-assembly are taken as the anchors A and B. Their role is to guarantee that any embedding of π into τ that maps A to B must be grid-preserving, i.e. it maps the image of the tile P_{v_i} to the image of the tile T_{v_i} for every i. The second pair of tiles P_{v_2} and T_{v_2} then simulates a mapping $\phi\colon [k] \to V_H$ that respects the coloring χ. And finally for every $i \in [k]$, the tiles corresponding to vertices v_{4a-1}, v_{4a}, v_{4a+1} and v_{4a+2} verify that there is an edge in H between the vertex $\phi(i)$ and $\phi(j)$ for every $j > i$. ◀

▶ **Lemma 14.** *Let $\mathcal{C}$ be a class with the computable DTP. An instance (G, H, χ) of PSI can be reduced to an instance (π, τ, A, B) of $\mathcal{C}$-Pattern APPM where $|\pi| \in O(|E_G| \cdot \log |E_G|)$ and $|\tau| \in O(|E_H| + |V_H| \cdot |E_G|)$ in time $f(|E_G|) \cdot |V_H|^{O(1)}$ for some function f.*

Proof idea. In this reduction, we combine the ideas of the reduction for the LPP (Lemma 13) with the proof that DTP implies near-linear tree-width (Proposition 5).

Using the computable DTP, we obtain a monotone gridding matrix $\mathcal{M}$ such that Grid($\mathcal{M}$) is a subclass of $\mathcal{C}$ and the cell graph of $\mathcal{M}$ is a c-subdivided binary tree with $|E_G|$ leaves. Additionally, we require that the root r of the tree has a single child r', and that each parent of a leaf has no other children. We again construct the pattern π via an $\mathcal{F}$-assembly from a family of tiles $\mathcal{P}$ and the text τ from a family of tiles $\mathcal{T}$ where the only non-empty tiles in both families correspond to the non-empty entries of $\mathcal{M}$ with each non-empty tile in $\mathcal{P}$ being an increasing sequence.

We set the tiles P_r and T_r to contain each a pair of elements which become the anchors A and B under the $\mathcal{F}$-assembly and which guarantee that any embedding of π into τ that respects the anchors must be grid-preserving. Using the same idea as in the proof of Lemma 13, the pair of tiles $P_{r'}$ and $T_{r'}$ is used to simulate a mapping $\phi\colon V_G \to V_H$ that respects the coloring χ. But now instead of verifying sequentially the neighborhood of each vertex in G, we aim to verify the existence of each edge in a particular leaf.

Set $k = |E_G|$. Following along the proof of Proposition 5, we orient the edges of the cell graph $G_{\mathcal{M}}$ consistently away from r and for any vertex v, the descendants of v, denoted by $D(v)$, are all the out-neighbors of v. We arbitrarily order the edges $E_G = \{e_1, \dots, e_k\}$ and also the k leaves of $G_{\mathcal{M}}$ as $v_1, \dots, v_k$, and we define the sets A_w exactly as in (1). We additionally assume that $A_r = [k]$ which corresponds to G having no isolated vertices. We again have $\sum_v A_v \in O(k \log k)$.

Now we spread the information about the mapping ϕ from r' to each leaf while keeping in each vertex only the information necessary to decide the existence of edges assigned to leaves in its subtree. In other words for a vertex v, we force the mapping of P_v into T_v to encode the mapping ϕ restricted to A_v. This in particular allows us to bound the size of π by $O(k \log k)$. Finally, we use the leaf v_i and its parent to test the existence of an edge $\{\phi(a_i), \phi(b_i)\} \in E_H$ where $e_i = \{a_i, b_i\}$ using the same construction as in Lemma 13. ◀

Observe that both reductions produce π and τ as gridded permutations belonging to some monotone grid class Grid($\mathcal{M}$) via an $\mathcal{F}$-assembly from families of tiles. Importantly, they share the property that any embedding of π into τ that maps A to B must be grid-preserving, i.e., it maps the (i, j)-cell of the gridding of π to th (i, j)-cell of the gridding of τ for every i and j. Moreover, both A and B are pairs of consecutive points in the left-to-right order.

4.1 Consequences

▶ **Theorem 15.** *If $\mathcal{C}$ has the computable long-path property then $\mathcal{C}$-Pattern PPM cannot be solved in time $f(t) \cdot n^{o(t)}$ where $t = \mathrm{tw}(\pi)$ for any function f, unless ETH fails.*

Proof. Let (π, τ, A, B) be the instance of C-Pattern APPM produced by Lemma 13 and let m be the length of τ. We define π' as the permutation obtained from π by inflating both of the anchors in A with either two increasing or decreasing sequences of length m such that π' is still contained in $\mathcal{C}$. Recall that one of these inflations is always possible. And similarly, we let τ' be the permutation obtained from τ by inflating both of the anchors in B with the same type of monotone sequences of length m as in π'.

We claim that π' is contained in τ' if and only if (π, τ, A, B) is a positive instance of C-Pattern APPM. It is clear that if there is an embedding of π into τ that maps A to B, then there is an embedding of π' into τ'.

For the other direction, assume there is an embedding ϕ of π' into τ'. The inflated anchors in π' contain exactly $2m$ points while τ' contains only $m-2$ points outside of its inflated anchors. Therefore, at least $m+2$ points of the inflated anchors in π' are mapped by ϕ to the inflated anchors in τ' and in particular, there must be at least one point in each of the anchors in π' mapped to the corresponding anchor in τ'. Since the anchors A and B are pairs of consecutive points, observe that we can, in fact, map the whole inflated anchors in π' to the inflated anchors in τ'. It follows that we obtain a desired anchored embedding of π into τ by deflating the anchors back to a single point.

Finally, we show that $\mathrm{tw}(\pi') \leq \mathrm{tw}(\pi) + 2$. The desired bound follows as otherwise, we could use a faster algorithm for $\mathcal{C}$-Pattern PPM to decide the instance (π, τ, A, B) of C-Pattern APPM and consequently refute ETH by the "moreover" part of Lemma 13. We claim that in general, if σ' is obtained from σ by inflating one point with a monotone sequence then $\mathrm{tw}(\sigma') \leq \mathrm{tw}(\sigma) + 1$. To see that, notice that when we inflate a point of σ with a monotone sequence of length 2, we get σ' such that $\mathrm{tw}(\sigma') \leq \mathrm{tw}(\sigma) + 1$. However, if we inflate the same point by a longer monotone sequence and obtain a permutation σ'' then $G_{\sigma''}$ can be obtained by edge subdivisions from $G_{\sigma'}$, and it is well-known that subdividing en edge does not increase tree-width. ◀

In order to show the hardness of $\mathcal{C}$-Pattern #PPM, we first reduce to an intermediate problem called *$\mathcal{C}$-Pattern Surjective Colored PPM* ($\mathcal{C}$-Pattern SCPPM) whose input consists of a pattern $\pi \in \mathcal{C}$, a text τ and a coloring $\chi\colon \tau \to [t]$. We need to decide whether there is an embedding of π into τ that hits all t possible colors. This intermediate reduction allows us to infer conditional lower bounds for $\mathcal{C}$-Pattern #PPM via the following lemma.

▶ **Lemma 16** (Berendsohn [2])**.** *Let there be an algorithm that solves $\mathcal{C}$-Pattern #PPM in time $f(k) \cdot n^{O(g(k))}$ for some functions f and g. Then $\mathcal{C}$-Pattern SCPPM can be solved in time $h(k) \cdot n^{O(g(k))}$ for some function h.*

▶ **Lemma 17.** *An instance (π, τ, A, B) of $\mathcal{C}$-Pattern APPM produced by Lemma 13 or 14 can be reduced to an instance (π', τ', χ) of $\mathcal{C}$-Pattern SCPPM where $|\pi'| \in O(|\pi|)$ and $|\tau'| \in O(|\tau|)$ in polynomial time.*

Proof. The general idea of the proof is the same as in Theorem 15 – we force matching of the anchors by inflating them with long monotone sequences. The $\mathcal{C}$-Pattern SCPPM problem, however, allows us to use sequences with length depending only on π. Let k be the length of π and let π' be the permutation obtained by inflating the anchors A with either two increasing or decreasing sequences of length k such that $\pi' \in \mathcal{C}$, and let τ' be the permutation obtained by the same inflation of the anchors B. We define $\chi\colon \tau \to [2k+1]$ by coloring every point added during the inflation with a unique color and using a single additional color for every other point. Clearly, $|\pi'| \in O(|\pi|)$ and $|\tau'| \in O(|\tau|)$.

We need to verify the correctness of our construction. If (π, τ, A, B) is a positive instance of $\mathcal{C}$-Pattern APPM then (π', τ', χ) is a positive instance of $\mathcal{C}$-Pattern SCPPM as we can simply map the inflated anchors of π' to the inflated anchors of τ'. For the other direction, assume there is an embedding ϕ of π' into τ' that hits all the $2k+1$ colors. In other words, the image of π' under ϕ contains the whole inflated anchors of τ'. Since there are only $k-2$ points in π' outside of the anchors, at least $k+2$ points of the anchors in π' maps to the anchors in τ'. In particular, there must be at least one point in each of the two increasing inflated anchors in π' that maps to the corresponding anchor in τ'. By the same argument as in the proof of Theorem 15, we conclude that the inflated anchors map without loss of generality to the inflated anchors. ◀

▶ **Theorem 18.** *Unless ETH fails, $\mathcal{C}$-Pattern #PPM cannot be solved for any function f*
- *in time $f(k) \cdot n^{o(\sqrt{k})}$ if $\mathcal{C}$ has the computable LPP, and*
- *in time $f(k) \cdot n^{o(k/\log^2 k)}$ if $\mathcal{C}$ has the computable DTP.*

Proof. For $\mathcal{C}$ with the computable LPP, a faster algorithm would refute ETH via

$$\text{Partitioned Clique} \xrightarrow{\text{Lemma 13}} \mathcal{C}\text{-Pattern APPM} \xrightarrow{\text{Lemma 17}} \mathcal{C}\text{-Pattern SCPPM} \xrightarrow{\text{Lemma 16}} \mathcal{C}\text{-Pattern \#PPM}$$

Whereas for $\mathcal{C}$ with the computable DTP, a faster algorithm would refute ETH via

$$\text{PSI} \xrightarrow{\text{Lemma 14}} \mathcal{C}\text{-Pattern APPM} \xrightarrow{\text{Lemma 17}} \mathcal{C}\text{-Pattern SCPPM} \xrightarrow{\text{Lemma 16}} \mathcal{C}\text{-Pattern \#PPM}.$$ ◀

References

1 Shlomo Ahal and Yuri Rabinovich. On complexity of the subpattern problem. *SIAM J. Discrete Math.*, 22(2):629–649, 2008. doi:10.1137/S0895480104444776.

2 Benjamin Aram Berendsohn. Complexity of permutation pattern matching. Master's thesis, Freie Universität Berlin, Berlin, 2019.

3 Benjamin Aram Berendsohn, László Kozma, and Dániel Marx. Finding and counting permutations via CSPs. In Bart M. P. Jansen and Jan Arne Telle, editors, *14th International Symposium on Parameterized and Exact Computation, IPEC 2019, September 11-13, 2019, Munich, Germany*, volume 148 of *LIPIcs*, pages 1:1–1:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.IPEC.2019.1.

4 Prosenjit Bose, Jonathan F. Buss, and Anna Lubiw. Pattern matching for permutations. *Inform. Process. Lett.*, 65(5):277–283, 1998. doi:10.1016/S0020-0190(97)00209-3.

5 Karl Bringmann, László Kozma, Shay Moran, and N. S. Narayanaswamy. Hitting set for hypergraphs of low VC-dimension. In *24th Annual European Symposium on Algorithms*, volume 57 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 23, 18. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2016.

6 Marek Cygan, Fedor V. Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized algorithms*. Springer, Cham, 2015. doi:10.1007/978-3-319-21275-3.

7 Vida Dujmović, David Eppstein, and David R. Wood. Structure of graphs with locally restricted crossings. *SIAM Journal on Discrete Mathematics*, 31(2):805–824, 2017. doi:10.1137/16M1062879.

8 Jacob Fox. Stanley–Wilf limits are typically exponential. arXiv:1310.8378v1, 2013.

9 Sylvain Guillemot and Dániel Marx. Finding small patterns in permutations in linear time. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 82–101. ACM, New York, 2014. doi:10.1137/1.9781611973402.7.

10 Sylvain Guillemot and Stéphane Vialette. Pattern matching for 321-avoiding permutations. In *Algorithms and computation*, volume 5878 of *Lecture Notes in Comput. Sci.*, pages 1064–1073. Springer, Berlin, 2009. doi:10.1007/978-3-642-10631-6_107.

11 Vít Jelínek and Jan Kynčl. Hardness of permutation pattern matching. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 378–396. SIAM, Philadelphia, PA, 2017. doi:10.1137/1.9781611974782.24.

12 Vít Jelínek, Michal Opler, and Jakub Pekárek. A complexity dichotomy for permutation pattern matching on grid classes. In Javier Esparza and Daniel Král', editors, *45th International Symposium on Mathematical Foundations of Computer Science, MFCS 2020, August 24-28, 2020, Prague, Czech Republic*, volume 170 of *LIPIcs*, pages 52:1–52:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.MFCS.2020.52.

13 Vít Jelínek, Michal Opler, and Jakub Pekárek. Griddings of Permutations and Hardness of Pattern Matching. In Filippo Bonchi and Simon J. Puglisi, editors, *46th International Symposium on Mathematical Foundations of Computer Science (MFCS 2021)*, volume 202 of *LIPIcs*, pages 65:1–65:22. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.MFCS.2021.65.

14 Dániel Marx. Can you beat treewidth? *Theory Comput.*, 6:85–112, 2010. doi:10.4086/toc.2010.v006a005.

15 Vincent Vatter and Steve Waton. On partial well-order for monotone grid classes of permutations. *Order*, 28(2):193–199, 2011. doi:10.1007/s11083-010-9165-1.