ABSTRACT. We characterise the classes of tournaments with tractable first-order model checking. For every hereditary class of tournaments $T$, first-order model checking is either fixed parameter tractable or $AW[\ast]$-hard. This dichotomy coincides with the fact that $T$ has either bounded or unbounded twin-width, and that the growth of $T$ is either at most exponential or at least factorial. From the model-theoretic point of view, we show that NIP classes of tournaments coincide with bounded twin-width. Twin-width is also characterised by three infinite families of obstructions: $T$ has bounded twin-width if and only if it excludes at least one tournament from each family. This generalises results of Bonnet et al. on ordered graphs.

The key for these results is a polynomial time algorithm which takes as input a tournament $T$ and computes a linear order $<$ on $V(T)$ such that the twin-width of the birelation $(T, <)$ is at most some function of the twin-width of $T$. Since approximating twin-width can be done in polynomial time for an ordered structure $(T, <)$, this provides a polynomial time approximation of twin-width for tournaments.

Our results extend to oriented graphs with stable sets of bounded size, which may also be augmented by arbitrary binary relations.

1. Introduction

1.1. Parameterized problems in tournaments. Tournaments can represent the outcome of a ranking procedure on a set of candidates, which in general need not be a total order. A well-known example is the Condorcet voting paradox, where three referees whose preference lists are $(A, B, C)$, $(B, C, A)$, and $(C, A, B)$ lead to a cycle $A \leftarrow B \leftarrow C \leftarrow A$ in the ‘preferred over’ relation. This interpretation leads to classical algorithmic problems when trying to choose a subset of winners. For instance the DOMINATING SET (DS) problem on tournaments can be understood as asking for a subset $D$ which is collectively preferred to any other candidate, in the sense that for any $y \notin D$, there is $x \in D$ such that $x$ is preferred to $y$. The FEEDBACK VERTEX SET (FVS) problem attempts to build a preference order—i.e. an acyclic graph—over the candidates by ignoring a subset of candidates.

One can consider the parameterized version of these problems, asking for a solution of size $k$—they are then denoted by $k$-DS, $k$-FVS. A problem
parameterized by \( k \) is fixed parameter tractable (FPT) if it admits an algorithm running in time \( O(f(k)n^c) \) where \( n \) is the size of the input. For instance \( k \)-FVS is well-known to be FPT for tournaments (see Kumar and Lokshenov [24]), whereas \( k \)-DS is probably not FPT. However general tournaments may not be representative of usual instances, for example, majority voting tournaments involving \( 2r + 1 \) referees form a very restricted class. A cornerstone paper by Alon et al. [2], based on Vapnik-Chervonenkis dimension, shows that \( k \)-DS is trivially FPT on \((2r + 1)\)-majority tournaments because the minimum size of a dominating set is bounded by \( f(r) \). This exemplifies how difficult problems can become easy on restricted classes, and how complexity parameters (here bounded VC-dimension) can help to define classes of tournaments.

To put these questions in a much broader perspective, remark that the previous problems can be expressed in first-order logic (FO). The existence of a \( k \)-DS is described by the formula

\[
\exists x_1, x_2, \ldots, x_k. \forall y. (y \to x_1) \lor \cdots \lor (y \to x_k)
\]

(assuming \( x \to x \) for all \( x \) for simplicity). That \( k \)-FVS is also expressible in first-order logic is worth mentioning, as this is only true in tournaments, and not in general graphs. It is based on the simple remark that a tournament is acyclic if and only if it is transitive, i.e. it has no directed 3-cycle—the latter condition is easily expressed in FO. This makes \( k \)-DS and \( k \)-FVS very specific instances of the FO Model Checking (or FOMC) problem, which given as input a tournament \( T \) and a first-order formula \( \phi \), asks if \( T \models \phi \).

FO model checking is difficult on the class of all graphs [15], and using back-and-forth encodings expressible in first-order logic, one can show that it is just as hard in general tournaments. Thus, one cannot hope for substantial algorithmic results on first-order model checking for arbitrary tournaments. Instead, we study the classes of tournaments in which the problem is simple, meaning the classes of tournaments which admit an FPT algorithm for FO model checking parameterized by the size of the given formula.

1.2. Main results. We prove a dichotomy: in any class of tournaments \( \mathcal{T} \) (closed under taking subtournaments), FOMC is either FPT or AW\[\ast\]-hard. More precisely, we show that the key of this dichotomy is twin-width, a complexity parameter introduced by Bonnet et al. [7], defined on arbitrary binary structures. If \( \mathcal{T} \) has bounded twin-width, then FOMC in \( \mathcal{T} \) is FPT, whereas it is AW\[\ast\]-hard when \( \mathcal{T} \) has unbounded twin-width. This dichotomy also coincides with a model theoretic characterisation: the class \( \mathcal{T} \) has bounded twin-width if and only if it is (monadically) NIP, which roughly speaking means that arbitrary graphs cannot be described by use of tournaments in \( \mathcal{T} \), and a fixed first-order formula. In the vocabulary of [11], this proves that tournaments are delineated: every subclass is NIP if and only if it has bounded twin-width, answering a question posed in [11]. The equivalence between NIP and FPT FO model checking also confirms the nowhere FO dense conjecture of Gajarský et al. [19] in the case of tournaments.

The dichotomy for FO model checking in tournaments also coincides with a gap in the growth function \( g_T(n) \) of the class \( \mathcal{T} \), defined as the number of tournaments of \( \mathcal{T} \) with \( n \) vertices counted up to isomorphism. We show
that $T$ has bounded twin-width if and only if its growth is at most $c^n$ for some constant $c$. To the contrary, when twin-width is not bounded, the growth is at least $\lfloor n/2 \rfloor! - 1)!$. This exponential/factorial gap can be seen as a generalization of the Marcus-Tardos theorem on permutations avoiding a fixed pattern [25]. It may also be compared to results of Boudabbous and Pouzet [12] which imply that hereditary classes of tournaments have growth either at most polynomial or at least exponential—the growth being polynomial if and only if the class does not contain arbitrarily large tournaments with no acyclic modules.

The following theorem summarises our results.

**Theorem 1.1.** Let $T$ be a hereditary class of tournaments. Under the assumption $FPT \neq AW[\ast]^*$, the following are equivalent:

1. $T$ has bounded twin-width,
2. FO model checking in $T$ is FPT,
3. FO model checking in $T$ is not $AW[\ast]$-hard,
4. $T$ does not FO interpret the class of all graphs,
5. $T$ is monadically NIP, i.e. does not FO transduce all graphs,
6. the growth of $T$ is at most $c^n$ for some constant $c$,
7. the growth of $T$ is less than $\left(\lfloor n/2 \rfloor - 1\right)!$.

These equivalences are completed by three minimal classes of obstructions, characterising twin-width in terms of excluded substructures. These excluded tournaments are encodings of arbitrary permutations.

**Theorem 1.2.** There are three hereditary classes $F_\leq, F_\geq, F_\geq$ such that a hereditary class $T$ of tournaments has unbounded twin-width if and only if one of $F_\leq, F_\geq, F_\geq$ is a subclass of $T$.

Finally, we show that there is a fixed parameter tractable algorithm which approximates twin-width of tournaments up to some function.

**Theorem 1.3.** There are functions $f, g : \mathbb{N} \rightarrow \mathbb{N}$ and an algorithm which given a tournament $T$ with twin-width $k$, produces a witness that the twin-width of $T$ is at most $f(k)$ in FPT time $g(k) \cdot |T|^{O(1)}$.

This algorithm is crucial to obtain the FPT FO model checking algorithm in classes with bounded twin-width.

The former three theorems generalise to oriented graphs with independence number bounded by some fixed constant. Furthermore, the tournaments (or oriented graphs with bounded independence) can be augmented by any fixed number of arbitrary binary relations: the theorems still hold for the resulting class of binary relational structures. In particular, this work generalises the analogous results of [9] on ordered graphs and ordered binary structures: we replace linear orders with tournaments. In these generalisations, however, the classes of obstructions for Theorem 1.2 are more numerous, and we do not give precise descriptions.

---

1The assumption is only used for conditions 2 and 3.
1.3. Overview of the proof. A fundamental idea regarding twin-width is that upper bounds on twin-width can be witnessed by orders on vertices which exclude grid-like structures in the adjacency matrix. This idea appears in the founding works of Guillemot and Marx [23] and Bonnet et al. [7], and the relation between twin-width and orders has been deeply explored in [9]. However it is difficult to provide witnesses for lower bounds on twin-width using this approach: one needs to somehow prove that all orders contain grids. To this purpose, in any tournament $T$, we want to construct an order $<$ which, if $T$ has small twin-width, is a witness of this fact. An equivalent and more precise requirement is that if $T$ has twin-width $k$, then the bi-relation $(T, <)$ should have twin-width at most $f(k)$ for some function $f$.

A tentative approach to obtain such an order is to describe it in FO logic—that is, to give a FO transduction which produces a total ordering on any given tournament. Indeed, FO transductions preserve twin-width up to some function [7, Theorem 39]. Thus, if there is a universal transduction $\Phi$ which on any tournament $T$ computes some order $<$, we immediately obtain $\text{tww}(T, <) \leq f(\text{tww}(T))$ as desired. With such a transduction, and some additional requirements such as $<$ being efficiently computable, it would be straightforward to obtain our results from the known case of ordered graphs [9]. Unfortunately, this approach fails: to transduce a total order on an iterated lexicographic product of the 3-cycle with itself, one needs a first-order formula with size increasing in the number of iterations [5]. This class of counter-examples also has twin-width 1, meaning that it is not even possible to transduce an order on the class of tournaments of twin-width $k$ for a fixed $k$.

Instead, our approach is the following: we design a candidate total order $<$ on $T$, computable in polynomial time. If the bi-relation $(T, <)$ has small twin-width, we are done. On the other hand, if $(T, <)$ has large twin-width, then its adjacency matrix w.r.t. $<$ must contain a large high-rank grid by [9]. We then extract a subtournament $T' \subset T$ in which the adjacency matrix w.r.t. $<$ still has a substantial (but logarithmically smaller) high-rank grid, and on which $<$ is described by a FO transduction. This is enough to witness that $T$ has large twin-width. Using Ramsey arguments, we are also able to extract from $T'$ some obstruction from one of the classes $F_{\leq}$, $F_{<}$, $F_{\geq}$.

The construction of the order is simple: we consider a binary search tree (BST) on the vertex set, i.e. a tree in which the left, resp. right, branch of a node $x$ consists only of in-, resp. out-neighbours of $x$. The order $<$ is the left-to-right order on nodes of the tree. The extraction of smaller high-rank grids corresponds to the choice of a branch $B$, and the restriction of $<$ to the relevant subtournament is a lexicographic ordering according to $B$.

To summarize, the crucial property of BST orders is the following.

**Lemma 1.4.** There is a function $f$ such that for any tournament $T$ and any BST order $<$, $\text{tww}(T, <) \leq f(\text{tww}(T))$

Lemma 1.4 implies Theorem 1.3 to approximate the twin-width of $T$, it suffices to compute any BST order, which takes polynomial time, and then apply the approximation algorithm for ordered structures [9, Theorem 2], which is fixed parameter tractable. This last algorithm produces either a
contraction sequence (which is valid for \((T, <)\) and a fortiori for \(T\)), or a witness that \((T, <)\) has large twin-width, which in turn implies that \(T\) has large twin-width by Lemma 1.4.

Our main technical result is about extracting the three classes of obstructions to twin-width \(F_e, F_\leq, F_\geq\), which consist of encodings of arbitrary permutations into tournaments, see section 3 for the definition.

**Theorem 1.5.** Let \(T\) be a hereditary class of tournaments with the property that there are tournaments \(T \in T\) and BST orders \(<\) such that \(\text{tww}(T, <)\) is arbitrarily large. Then \(T\) contains one of the classes \(F_e, F_\leq, F_\geq\) as a subclass.

We also prove that the classes \(F_e, F_\leq, F_\geq\) are complex in all the senses considered by Theorem 1.1, that is

**Theorem 1.6.** For each \(R \in \{=, \leq, \geq\}\), the class \(F_R\)

1. has unbounded twin-width;
2. contains at least \(((\lfloor n/2 \rfloor - 1)! \cdot n)!\) tournaments on \(n\) vertices counted up to isomorphism;
3. contains at least \(((\lfloor n/2 \rfloor - 1)! \cdot n)!\) tournaments on vertex set \(\{1, \ldots, n\}\) counted up to equality;
4. efficiently interprets the class of all graphs;
5. and has \(AW[2]\)-hard FO model checking problem.

Theorems 1.5 and 1.6 together imply Theorem 1.2. They also imply Lemma 1.4 when applied to the class of tournaments with twin-width at most \(k\): this class cannot contain any of \(F_e, F_\leq, F_\geq\), hence its tournaments must still have bounded twin-width when paired with BST orders. Finally, Theorems 1.2 and 1.6 directly imply that if \(T\) is a hereditary class with unbounded twin-width, then \(T\) satisfies none of the conditions of Theorem 1.1. The remaining implications of Theorem 1.1—that is, when \(T\) has bounded twin-width, all other conditions hold—follow from known results on twin-width. By [7, Theorem 1], FO model checking has an FPT algorithm when a witness of bounded twin-width is given. Combined with Theorem 1.3, we obtain an FPT algorithm for classes of tournaments with bounded twin-width. By [7, Theorem 39], a class of structures with bounded twin-width cannot transduce all graphs. Finally, by [6, Corollary 7.3], a class of structures with bounded twin-width contains at most \(c^n\) structures on \(n\) vertices up to isomorphism, for some constant \(c\) depending on the class.

1.4. Context and related works. An important question is how twin-width compares to other classical complexity measures for tournaments. Bounded twin-width implies bounded Vapnik-Chervonenkis dimension, because unbounded VC-dimension implies all possible bipartite tournaments appearing as subgraphs—which is against single-exponential growth, and thus twin-width is unbounded. The notion of cutwidth was introduced by Chudnovsky, Fradkin and Seymour [14] to study tournament immersions. The vertex ordering used to certify that cutwidth is bounded can be shown to exclude grids, meaning that it is a witness of bounded twin-width. Another parameter, closely related to subdivisions in tournaments, is pathwidth, studied by Fradkin and Seymour [18]. Bounded pathwidth of tournaments
implies bounded cliquewidth, which in turn also implies bounded twin-width, see [7]. Thus, we have the following chain of inclusions of complexity parameters (if a parameter is bounded, all the ones listed after are also bounded): cutwidth, pathwidth, cliquewidth, twin-width, and VC-dimension. For more on the subject, we invite to read Fomin and Pilipczuk [17] or Pilipczuk’s thesis [27].

Our results are a direct generalisation of those of [9], from ordered graphs to tournaments. This generalisation is non-trivial: as explained in section 1.3, one cannot FO transduce orders from arbitrary tournaments—which would be the obvious way to reduce the problem from tournaments to ordered graphs. On the other hand, there is a natural way to form a tournament $T$ from an ordered graph $(G, <)$, namely $x \rightarrow y$ is an arc of $T$ if and only if $x < y$ and $xy$ is an edge of $G$, or $y < x$ and $xy$ is not an edge of $G$. This gives a way to create tournaments with bounded twin-width: starting with any graph $G$ with bounded twin-width, consider a total order $<$ on its vertices which is a witness of its twin-width, and interpret $T$ from $(G, <)$ as before. This paper can be seen as the reverse operation, that is a way of producing an order $<$ from a tournament $T$.

The binary search tree we use to order tournaments correspond to the KWIKSORT algorithm of Ailon, Charikar and Newman for approximating the minimum feedback arc set [1]. The only difference is that their result requires the BST to be randomly chosen, whereas arbitrary BST provide approximations of twin-width.

The generalization to oriented graphs with bounded independence number contains in particular partially ordered sets with bounded width (i.e. antichains of bounded size). These classes in fact always have bounded twin-width, see Balabán and Hlinený [3], and before that were already known to be tractable with respect to FO model checking, see Gajarský et al. [20].

Finally let us mention another notable result on tournaments of bounded twin-width: the polynomial time isomorphism test of Grohe and Neuen [22]. Their algorithm is remarkable in that it leverages twin-width without needing or manipulating a contraction sequence witness of twin-width. The techniques involved are very different from ours.

1.5. Organisation of the paper. Section 2 summarises definitions and notations used in this paper. In section 3 we define the classes $\mathcal{F}_=, \mathcal{F}_\leq, \mathcal{F}_\geq$ of obstructions to twin-width in tournaments, and prove Theorem 1.6. Section 4 introduces binary search trees, and proves a crucial lemma, which from a partition into intervals of a BST order, extracts some FO definable ordered substructure. Section 5 combines this key lemma with results of [9] to prove Lemma 1.4: from a witness that the tournament $T$ augmented by a BST order has large twin-width, the key lemma extracts a witness that $T$ itself has large twin-width. Section 6 refines this argument into a proof of Theorem 1.5 using Ramsey results. Finally, section 7 explains how our results generalise to oriented graphs with bounded independence number, and to binary relational structures.
2. Preliminaries

This section summarizes the notions and notations used in this work. For \( n \in \mathbb{N} \), we denote by \([n]\) the interval of integers \( \{1, \ldots, n\} \). All the graphs, matrices, and relational structures considered in this work are finite.

2.1. Graphs, Oriented graphs, Tournaments. A directed graph \( D \) consists of a vertex set \( V(G) \), and a set \( A(G) \) of arcs which are ordered pairs of vertices. An arc \((u, v)\) is simply denoted by \( uv \), or sometimes \( u \to v \) to insist on the orientation. In this work, we do not consider loops, i.e. arcs of the form \( v \to v \). The underlying graph of \( D \) is the undirected graph with the same vertices, and an edge \( uv \) whenever either \( uv \) or \( vu \) is an arc of \( D \).

A digon in a directed graph is a pair of vertices \( u, v \) with both edges \( u \to v \) and \( v \to u \). An oriented graph is a directed graph without digons.

A tournament is an oriented graph whose underlying graph is complete, or equivalently it is a choice of orientation \( u \to v \) or \( v \to u \) (but not both) for every pair of vertices. A tournament is transitive if it contains no directed cycle. A transitive tournament represents some order \( \prec \) on vertices: there is an edge \( u \to v \) if and only if \( u \prec v \). In a directed graph \( D \), a subset \( X \subseteq V(D) \) which induces a transitive tournament is also called a chain.

A (directed) graph \( H \) is an induced subgraph of \( G \) if \( V(H) \subseteq V(G) \), and the edges (or arcs) of \( H \) are exactly the edges of \( G \) which are between vertices of \( V(H) \). If \( X = V(H) \), we also say that \( H \) is the subgraph of \( G \) induced by \( X \), denoted \( H = G[X] \). A class of graphs is a family of graphs closed under isomorphism. A class is hereditary if it is also closed under taking induced subgraphs.

A subset \( S \) of a (directed) graph is called independent or stable if no edge (arc) joins two vertices of \( S \). The independence number \( \alpha(G) \) is the maximum size of an independent set in \( G \). If \( x \) is a vertex of a directed graph \( D \), then \( N^+(x) = \{ y \mid x \to y \} \) denotes its out-neighbourhood, and \( N^-(x) = \{ y \mid y \to x \} \) its in-neighbourhood. The complete neighbourhood is \( N(x) = N^+(x) \cup N^-(x) \).

2.2. Relational structures. Relational structures are a natural generalisation of graphs. A relational signature is a finite set \( \Sigma \) of relation symbols \( R \), each with an associated arity \( r \in \mathbb{N} \). A relational structure \( S \) over \( \Sigma \), or simply \( \Sigma \)-structure, consists of a domain \( A \), and for each relation symbol \( R \in \Sigma \) of arity \( r \), an interpretation of \( R \) as \( R^S \subseteq A^r \). One may think of \( A \) as a set of vertices, and each \( R^S \) as a set of hyperedges. The notions of induced substructure, hereditary classes, etc. generalise to relational structures in the obvious way.

We will only consider binary relational structures, i.e. over signatures where all symbols have arity 2. Directed graphs are binary structures with a single relation for arcs.

An ordered structure \( S \) is a structure over a relation \( \Sigma \) with a special symbol, typically denoted \( \prec \), whose interpretation \( <^S \) is a strict total order on the domain of \( S \).

2.3. Matrices. We consider matrices as maps \( M : R \times C \to \Gamma \), where \( R, C \) are ordered sets of rows and columns of the matrix, and \( \Gamma \) and its alphabet.
The order of rows and columns is very important in the context of twin-width. Usually, the alphabet will simply be \( \{0,1\} \). The value \( M(x,y) \) is called the entry at position \( (x,y) \), and an \( a \)-entry means an entry whose value is \( a \). A submatrix of \( M \) is the restriction of \( M \) to some subsets of rows and columns. The notion is comparable to that of induced substructure, thinking of rows and columns as vertices, and of the content of the matrix as a relation. Accordingly, a class of matrices is called hereditary if it is closed under submatrices.

Given a (directed) graph \( G \) and a total order \( < \) on \( V(G) \), the adjacency matrix \( A(G,<) \) is a 0,1-matrix with rows and columns \( V(G) \), and with a ‘1’ at position \( (u,v) \) if and only if \( uv \) is an edge or arc of \( G \). A common alternative definition for oriented graphs, which we do not use, is to place ‘−1’ at position \( (v,u) \) when \( uv \) is an arc, so as to keep the matrix antisymmetric. Adjacency matrices generalise easily to binary relational structures over any signature \( \Sigma \), the alphabet being \( \{0,1\}^\Sigma \) in this case.

A division \( D \) of a matrix consist of partitions \( \mathcal{R},\mathcal{C} \) of its rows and its columns respectively into intervals. It is a \( k \)-division if the partitions have \( k \) parts each. A cell of the division is the submatrix induced by \( X \times Y \) for some \( X \in \mathcal{R},Y \in \mathcal{C} \). A \( k \)-grid in a 0,1-matrix is a division in which every cell contains a ‘1’-entry.

2.4. Orders, Quasi-orders. A quasi-order \( \preceq \) is a reflexive and transitive binary relation. Any quasi-order induces an equivalence relation \( x \sim y \) defined as \( x \preceq y \land y \preceq x \), which is trivial if and only if \( \preceq \) is an order. One can also describe a quasi-order by giving this equivalence relation, and the order on the equivalence classes. The strict component of the quasi-order is defined by \( x < y \) if and only if \( x \preceq y \) and \( y \not\preceq x \). The quasi-order is total if for all \( x,y \), either \( x \preceq y \) or \( y \preceq x \).

An interval of a quasi-order \( \preceq \) is a set of the form \( \{ z \mid x \preceq z \preceq y \} \) for some \( x,y \), called endpoints of the interval. Remark that an interval is always a union of equivalence classes of \( \sim \).

2.5. Permutations. We denote by \( \mathfrak{S}_n \) the group of permutations on \( n \) elements. The permutation matrix \( M_\sigma \) of \( \sigma \in \mathfrak{S}_n \) is the sparse \( n \times n \) matrix with a ‘1’ at position \( (i,j) \) if and only if \( j = \sigma(i) \). A permutation \( \tau \) is a pattern of \( \sigma \) if \( M_\tau \) is a submatrix of \( M_\sigma \).

We say that \( \sigma \) contains a \( k \)-grid if \( M_\sigma \) contains a \( k \)-grid. When this is the case, any permutation in \( \mathfrak{S}_k \) is a pattern of \( \sigma \). For example, the permutation \( \sigma \) on \( k^2 \) elements defined by \( \sigma(ki + j + 1) = kj + i + 1 \) for any \( 0 \leq i,j < k \) contains a \( k \)-grid. Grids in permutations are deeply related to pattern-closed classes of permutations, which are precursors of the work on twin-width, see Marcus and Tardos [25] and Guillemot and Marx [23].

A permutation can be represented as a bi-order, i.e. the superposition of two total orders. Precisely, for \( \sigma \in \mathfrak{S}_n \), the structure \( \mathcal{O}_\sigma \) has domain \([n]\), and has for relations the natural order \( < \), and the permuted order \( <_\sigma \) defined as \( i <_\sigma j \) if and only if \( \sigma(i) < \sigma(j) \). Any bi-order is isomorphic to some \( \mathcal{O}_\sigma \). Remark that \( \tau \) is a pattern of \( \sigma \) if and only if \( \mathcal{O}_\tau \) is isomorphic to an induced substructure of \( \mathcal{O}_\sigma \). We write \( \mathcal{O}_{\mathfrak{S}} \) for the class of all finite bi-orders.
For any numbers \(x, y\) let the order type \(\text{ot}(x, y)\) be 1 if \(x < y\), \(-1\) if \(x > y\), and 0 if \(x = y\). We also write \(\text{ot}_\sigma(x, y)\) for \(\text{ot}(\sigma(x), \sigma(y))\), the order type with regards to the permuted order \(\prec_\sigma\). In a bi-order \(\mathcal{O}_\sigma = ([n], \prec, \prec_\sigma)\), consider a coloring of pairs \(\lambda : [n]^2 \to \Gamma\) for some finite alphabet \(\Gamma\). We say that this coloring depends only on the orders \(\prec, \prec_\sigma\) if there is a \(\eta : \{-1, 0, 1\}^2 \to \Gamma\) such that \(\lambda\) factors as

\[
\lambda(x, y) = \eta(\text{ot}(x, y), \text{ot}_\sigma(x, y))
\]

For example, interpreting the edges of a graph on \([n]\) as a coloring, the permutation graph with an edge \(xy\) if only if \((x < y \land x \prec_\sigma y)\) or \((x > y \land x \prec_\sigma y)\) is one which only depends on \(\prec\) and \(\prec_\sigma\).

The following result of Ramsey theory, stated as such in [9, Lemma 17], will be used extensively to extract the minimal classes of obstructions in tournaments with large twin-width. It is a corollary of the product Ramsey theorem, see [4, Proposition 3.3] or [21, page 97].

**Lemma 2.1.** For any permutation \(\sigma \in S_n\) and any finite alphabet \(\Gamma\), there is a permutation \(\tau\) satisfying the following. Let \(\mathcal{O}_\tau = ([n], \prec, \prec_\tau)\). For any coloring of pairs \(\lambda : [n]^2 \to \Gamma\), there exists \(X \subset N\) of size \(n\) such that the substructure \(\mathcal{O}_\tau[X]\) is isomorphic to \(\mathcal{O}_\sigma\), and the coloring \(\lambda\) restricted to \(X\) only depends on the orders \(\prec, \prec_\tau\).

If the coloring \(\lambda\) represents the edges of a tournament on \([n]\), i.e. \(\lambda(x, y) = 1\) if and only if \(x \to y\) is an edge, then the ‘depends only on \(\prec, \prec_\tau\)’ conditions becomes particularly useful.

**Lemma 2.2.** Let \(T\) be a tournament and \(\prec_1, \prec_2\) two orders on \(V(T)\) such that the direction of edges of \(T\) only depends on \(\prec_1, \prec_2\). Then \(T\) is transitive, and the direction of edges coincides with one of \(\prec_1, \prec_2\), the reverse of \(\prec_1\), or the reverse of \(\prec_2\).

**Proof.** Let \(\eta : \{-1, 0, 1\}^2 \to \{0, 1\}\) be such that \(\lambda\) factors as \(\lambda(x, y) = \eta(\text{ot}(x, y), \text{ot}_\sigma(x, y))\). Since \(T\) is a tournament, it is then simple to verify that \(\eta(-1, -1) = 1 - \eta(1, 1)\) and \(\eta(-1, 1) = 1 - \eta(1, -1)\). It follows that the value of \(\eta\) only actually depends on one of the two coordinates. It is then simple to verify that \(T\) is transitive, with either the same order or the reverse of the order corresponding to this coordinate. \(\square\)

**2.6. Twin-width.** We recall the definition of twin-width for the sake of completeness, however we will not use it directly, relying on higher level results instead.

In a binary relational structure \(S\), two disjoint subsets \(X, Y\) of the domain are homogeneous if for all relations \(R\) of \(S\), if there are \(x \in X, y \in Y\) with \(xRy\), then for all \(x' \in X, y' \in Y, x'Ry'\), and symmetrically. Thus, the relations between \(X\) and \(Y\) do not distinguish any vertex. For example, in a graph, \(X, Y\) are homogeneous if and only if the bipartite graph between \(X\) and \(Y\) is either empty or complete; and in a tournament, \(X\) and \(Y\) are homogeneous if and only if all edges are oriented from \(X\) to \(Y\), or all from \(Y\) to \(X\).

A contraction sequence for a structure \(S\) is a sequence \(\mathcal{P}_n, \ldots, \mathcal{P}_1\) of partitions of the domain of \(S\), where \(\mathcal{P}_n\) is the partition into singletons, \(\mathcal{P}_1\) is
the partition in a single part, and \( P_i \) is obtained by merging two parts of \( P_i \).

For a given part \( X \in P_i \), the error degree of \( X \) in \( (S, P_i) \) is the number of parts \( Y \in P_i \setminus \{X\} \) which are not homogeneous to \( X \). The width of the contraction sequence is the maximum error degree of \( X \) in \( (S, P_i) \) over all choices of \( i \in [n] \) and \( X \in P_i \). Finally, the twin-width of \( S \), denoted \( \text{tww}(S) \), is the minimum width of a contraction sequence for \( S \).

Twin-width is characterised by grid-like structures in adjacency matrices. For a given part \( X \in P_i \), the contraction sequence is the maximum error degree of \( X \) in \( (S, P_i) \) over all choices of \( i \in [n] \) and \( X \in P_i \). Finally, the twin-width of \( S \), denoted \( \text{tww}(S) \), is the minimum width of a contraction sequence for \( S \).

Twin-width is characterised by grid-like structures in adjacency matrices.

2.7. First-order logic. We assume general familiarity with first-order logic on graphs and relational structures. Recall from the introduction that we are primarily interested in the First-Order Model Checking algorithmic problem, which asks given as input a structure \( S \) and a first-order formula \( \phi \), if \( \phi \) is satisfied by \( S \), which is denoted by \( S \models \phi \). We consider the complexity parametrized by the size \( |\phi| \). In general, this problem is \( \text{AW}[*]\)-complete [15, Theorem 1]. In fact, the proof shows more precisely that

**Theorem 2.4** ([15, Proposition 2]).

**First-Order Model Checking on the class of all graphs is AW[*]-complete.**

On classes of structures with bounded twin-width, FO model checking is FPT as long as a witness of twin-width is given.

**Theorem 2.5** ([7, Theorem 1]). **There is an algorithm which, given a binary structure \( S \) on \( n \) vertices, a contraction sequence of width \( k \) for \( S \), and a FO formula \( \phi \), decides if \( S \models \phi \) in time \( f(k, \phi) \cdot n \).**

Interpretations and transductions are ways to describe transformations of structures using logical formulæ. For two relational signatures \( \Sigma, \Delta \), a FO interpretation \( \Phi \) from \( \Sigma \) to \( \Delta \) consists of, for each relation \( R \in \Delta \) of arity \( r \), a FO formula \( \phi_R(x_1, \ldots, x_r) \) over the language \( \Sigma \), and one last formula \( \phi_{\text{dom}}(x) \) again over \( \Sigma \). Given a \( \Sigma \)-structure \( S \), the result \( \Phi(S) \) of the interpretation is the \( \Delta \)-structure obtained by

- choosing the same domain as \( S \),
- interpreting each relation \( R \in \Delta \) as the set of tuples \( (v_1, \ldots, v_r) \) such that \( S \models \phi_R(v_1, \ldots, v_r) \),
- and finally taking the substructure induced by \( \{v \mid S \models \phi_{\text{dom}}(v)\} \).

For instance, the square of a graph \( G \) is defined as the graph with vertices \( V(G) \), and in which \( x, y \) are adjacent if and only if their distance in \( G \)
is at most 2. This is a first-order interpretation in which the edge relation is interpreted by the formula
\[ \phi(x, y) = E(x, y) \lor (\exists z. E(x, z) \land E(z, y)) \]
where \( E(x, x) \) denotes adjacency. The domain formula is trivial since we do not wish to delete vertices in this interpretation. It is well-known that the composition of two FO interpretations can itself be written as a FO interpretation.

Transductions generalise interpretation with a non-deterministic coloring step. Consider relational signatures \( \Sigma \) and \( \Sigma^+ \), where \( \Sigma^+ \) consists of \( \Sigma \) augmented by \( r \) new unary relations \( C_1, \ldots, C_r \). If \( S \) is a structure over the signature \( \Sigma \), we denote by \( S^+ \) the set of structures over \( \Sigma^+ \) whose domain and interpretation of \( \Sigma \) coincide with that of \( S \). Thus, a \( T \in S^+ \) is uniquely described by the interpretation of \( C_1, \ldots, C_r \) within \( V(S) \). A FO transduction \( \Phi : \Sigma \to \Delta \) is described by the choice of \( \Sigma^+ \) augmenting \( \Sigma \) with unary relations, together with a FO interpretation \( \Phi_I \) from \( \Sigma^+ \) from \( \Delta \). The result of the transduction \( \Phi \) on a \( \Sigma \)-structure \( S \) is then the set of structures
\[ \Phi(S) = \{ \Phi_I(T) \mid T \in S^+ \} \]
Additional operations such as duplication of vertices are often allowed in transductions, but these will not be needed in this work. Like interpretations, transductions can be composed.

Given classes \( C, D \) of relational structures, we say that \( C \) interprets (resp. transduces) \( D \) if there is a FO interpretation (resp. transduction) \( \Phi \) such that \( \Phi(C) \supseteq D \). The interpretation \( \Phi \) is allowed to also output structures which are not in \( D \). We furthermore say that \( C \) efficiently interprets \( D \) if there is also an algorithm which given as input \( D \in D \), finds in polynomial time some \( C \in C \) such that \( \Phi(C) = D \). It is straightforward to show that this additional condition gives an FPT reduction for model checking.

**Lemma 2.6.** If \( C \) efficiently interprets \( D \), then there is an FPT reduction from FO Model Checking on \( D \) to FO Model Checking on \( C \).

Recall that \( \mathcal{O}_5 \) denotes the class of bi-orders, which are encodings of permutations. The following is a folklore result, see e.g. [9, Lemma 10] for a very similar claim.

**Lemma 2.7.** The class \( \mathcal{O}_5 \) of bi-orders efficiently interprets the class of all graphs.

Thus, using Lemma 2.6 and Theorem 2.3, FO Model Checking on \( \mathcal{O}_5 \) is \( \text{AW}[^s] \)-complete.

FO transductions also preserve twin-width, up to some function.

**Theorem 2.8 ([7, Theorem 39]).** If \( S \) is a class of binary structures with bounded twin-width and \( \Phi \) is a FO transduction defined on \( S \), then \( \Phi(S) \) also has bounded twin-width.

A class of structures \( S \) is said to be monadically NIP if \( S \) does not transduce the class of all graphs. Since the class of all graphs does not have bounded twin-width, Theorem 2.8 implies that classes with bounded twin-width are monadically NIP. The weaker notion of (non-monadicly)
NIP also exists, with a slightly more technical definition. Braunfeld and Laskowski recently proved that NIP and monadically NIP are equivalent notions for hereditary classes of structures [13], thus the distinction is irrelevant in this work.

Gajarsky et al. conjectured that NIP is equivalent to FPT model checking.

**Conjecture 2.9** ([19] Conjecture 8.2). A hereditary class of graphs has an FPT algorithm for first-order model checking if and only if it is NIP.

The first implication of Conjecture 2.9 was only recently proved by Dreier, Mählmann, and Toruńczyk [16]: they show that if \( C \) is hereditary and transduces the class of all graphs, then it in fact efficiently interprets all graphs, hence model checking in \( C \) is \( AW[1] \)-complete by Lemma 2.6. The converse implication (finding a model checking algorithm in NIP classes) is open.

Our main result confirms Conjecture 2.9 for tournaments.

2.8. Enumerative combinatorics. A class \( S \) of graphs (or binary relational structures) is small if there exists \( c \) such that \( S \) contains at most \( c^n \cdot n! \) structures on the vertex set \([n]\). For instance, the class of trees is small, and more generally proper minor closed classes of graphs are small as shown by Norine et al. [26]. This was further generalised to classes of bounded twin-width by Bonnet et al.

**Theorem 2.10** ([8, Theorem 2.4]). Classes of structures with bounded twin-width are small.

The reverse implication does not hold for graphs: there are small hereditary classes of graphs with unbounded twin-width [10]. These counterexamples have bounded maximum degree. One of the main results of [9] is that for classes of ordered graphs, bounded twin-width is equivalent to being small. We generalise it to tournaments.

One may also count structures up to isomorphism instead of counting labelled structures as in the definition of small classes. A strengthening of Theorem 2.10 when counting up to isomorphism was proved in [6].

**Theorem 2.11** ([6, Corollary 7.3]). For any class \( C \) of structures with bounded twin-width, there is a constant \( c \) such that \( C \) has at most \( c^n \) structures on \( n \) vertices counted up to isomorphism.

3. Forbidden classes of tournaments

This section defines the three minimal classes \( F=, F\leq, \) and \( F\geq \) of obstructions to twin-width in tournaments. Each of them corresponds to some encoding of the class of all permutations. For \( R \in \{=, \leq, \geq\} \) and any permutation \( \sigma \), we will define a tournament \( F_R(\sigma) \). The class \( F_R \) is the hereditary closure of all \( F_R(\sigma) \).

Let \( R \in \{=, \leq, \geq\} \), and let \( \sigma \in S_n \) be a permutation on \( n \) elements. The tournament \( F_R(\sigma) \) consists of \( 2n \) vertices, called \( x_1, \ldots, x_n, y_1, \ldots, y_n \). Let \( X = \{x_1, \ldots, x_n\} \) and \( Y = \{x_1, \ldots, y_n\} \). Each of \( X, Y \) is a chain under the natural order, i.e. there is an edge from \( x_i \) to \( x_j \), resp. from \( y_i \) to \( y_j \), if and only if \( i < j \). The edges between \( X \) and \( Y \) encode \( \sigma \) in a way specified by the relation \( R \): there is an edge oriented from \( y_j \) to \( x_i \) if and only if \( i R \sigma^{-1}(j) \). See Figure 1 for an example.
Thus in $F_=$($\sigma$) the edges oriented from $Y$ to $X$ form a matching which encodes $\sigma$. In $F_\leq$($\sigma$) and $F_\geq$($\sigma$), these edges form a half-graph which orders $X$ and $Y$ by inclusion of neighbourhoods, so that the order on $X$ is the natural one, and the order on $Y$ encodes $\sigma$. Precisely, in $F_\geq$($\sigma$), for any $i, j \in [n]$, we have

\begin{align*}
(1) & \quad N^-(x_i) \cap Y \subseteq N^-(x_j) \cap Y \iff i \leq j \\
(2) & \quad N^-(y_i) \cap X \subseteq N^-(y_j) \cap X \iff \sigma^{-1}(i) \leq \sigma^{-1}(j),
\end{align*}

while in $F_\leq$($\sigma$), the direction of inclusions is reversed.

**Lemma 3.1.** For each $R \in \{=, \leq, \geq\}$, the class $F_R$ efficiently interprets the class $O_\mathfrak{S}$ of permutations represented as bi-orders. Precisely, there is an interpretation $\Phi_R$, and for any permutation $\sigma \in \mathfrak{S}_n$, $n \geq 2$, there is a $\sigma' \in \mathfrak{S}_{n+1}$ computable in polynomial time such that $O_\sigma = \Phi_R(F_R(\sigma'))$.

**Proof.** We will first show that $F_R(\sigma)$ transduces $O_\sigma$, and then how to remove the coloring step of the transduction by slightly extending $\sigma$.

Let $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_n\}$ be as in the definition of $F_R(\sigma)$. The transduction uses coloring to guess the set $X$. It then defines two total orders on $Y$, which together describe $\sigma$. The first ordering is given by the direction of edges inside $Y$. The second depends on $R$:

- If $R$ is $=$, edges oriented from $Y$ to $X$ are a perfect matching. The direction of edges in $X$, interpreted through this matching, defines the second order on $Y$.
- If $R$ is $\geq$ or $\leq$, the second order is inclusion, respectively inverse inclusion, of in-neighbourhoods intersected with $X$, see (2).

With the knowledge of which subset is $X$, each of these orders is clearly definable with a first-order formula. Finally, the transduction deletes vertices of $X$, leaving only $Y$ and the two orders which encode $\sigma$. 

**Figure 1.** The three classes of obstructions to twin-width in tournaments. For readability, edges oriented from some $x_i$ to some $y_j$ have been omitted. Each class consists of some encoding of the class of all permutations, represented here with the permutation $\sigma = 31452$. 

\[
F = (\sigma) \quad F \leq (\sigma) \quad F \geq (\sigma)
\]
Let us now show how to deterministically define the partition $X, Y$, at the cost of extending $\sigma$ with one fixed value. Here, we assume $n \geq 2$.

- If $R$ is $=$, define $\sigma' \in S_{n+1}$ by $\sigma'(n+1) = n+1$ and $\sigma'(i) = \sigma(i)$ for any $i \leq n$. Then, in $F_(\sigma')$, the unique vertex with out-degree 1 is $y_{n+1}$. Its out-neighbour is $x_{n+1}$, which verifies $X = N^- (x_{n+1}) \cup \{x_{n+1}\} \setminus \{y_{n+1}\}$.
- If $R$ is $\leq$, define $\sigma'(1) = n+1$ and $\sigma'(i+1) = \sigma(i)$. Then $y_{n+1}$ is the unique vertex with out-degree 1, and its out-neighbour is $x_1$, which satisfies $X = N^+(x_1) \cup \{x_1\}$.
- If $R$ is $\geq$, we once again define $\sigma'(1) = n+1$ and $\sigma'(i+1) = \sigma(i)$. Then $x_1$ has in-degree 1, and its in-neighbour is $y_{n+1}$. The only other vertex which may have in-degree 1 is $y_1$, and this happens if and only if $\sigma'(2) = 1$. When this is the case, the direction of the edge $x_1 \rightarrow y_1$ still allows to distinguish $x_1$ in FO logic. Then, having defined $x_1$, we obtain $y_{n+1}$ as its in-neighbour, which satisfies $X = N^+ (y_{n+1})$.

In all three cases, $F_R(\sigma')$ contains two extra vertices compared to $F_R(\sigma)$. These extra vertices can be uniquely identified in first-order logic, and can then be used to define $X$. Combined with the previous transduction, this gives an interpretation of $O_\sigma$ in $F_R(\sigma')$. \hfill $\square$

We can now prove that the classes $F_R$ are complex.

**Theorem 1.6.** For each $R \in \{=, \leq, \geq\}$, the class $F_R$

1. has unbounded twin-width;
2. contains at least $\lceil \frac{n+1}{2} \rceil \cdot n!$ tournaments on $n$ vertices counted up to isomorphism;
3. contains at least $\lceil \frac{n+1}{2} \rceil \cdot n!$ tournaments on vertex set $\{1, \ldots, n\}$ counted up to equality;
4. efficiently interprets the class of all graphs;
5. and has AW[\text{log}]-hard FO model checking problem.

**Proof.** Item 3 is straightforward by Lemmas 2.7 and 3.1 since efficient interpretations can be composed. By Lemma 2.6 and Theorem 2.4 this in turn implies Item 5.

Item 3 implies Item 2 by a simple counting argument: in an isomorphism class, there are at most $n!$ choices of labelling of vertices with $\{1, \ldots, n\}$ (less when the graph has automorphism). Furthermore, each of Items 2 and 4 implies Item 1 by Theorem 2.10 and Theorem 2.8 respectively. Thus it only remains to prove Item 3.

By Lemma 3.1, for all $\sigma \in S_n$ there is some $F_R(\sigma')$ on $2n+2$ vertices such that $\Phi_R(F_R(\sigma')) = \sigma$, where $\Phi_R$ is an interpretation. Since interpretations preserve isomorphism, it follows that there are at least $n!$ non-isomorphic tournaments on $2n+2$ vertices in $F_R$. Furthermore, using the same arguments as in Lemma 3.1, it is easy to show that these $F_R(\sigma')$ have no non-trivial automorphism. Thus, there are exactly $(2n+2)!$ distinct labellings of $F_R(\sigma')$ with $\{1, \ldots, 2n+2\}$. In total, this gives $(2n+2)! \cdot n!$ distinct graphs on vertices $\{1, \ldots, 2n+2\}$ in $F_R$, proving Item 3. \hfill $\square$

Thus the classes $F_\leq, F_\geq, F_$ are obstructions to fixed parameter tractability of FO model checking and twin-width. The rest of the paper shows
that they are the only obstructions. One may also verify that all three are minimal, i.e. none of them is contained in another.

4. Binary search tree orders

This section presents the good order for twin-width in tournaments. It is based on binary search trees (BST), which we define in a tournament $T$ as a rooted ordered binary tree $S$ (meaning that each node has a left and right child, either of which may be missing), whose nodes are the vertices of $T$, and such that for any $x \in S$

- the left child of $x$ (if any) and its descendants are in $N^-_T(x)$, and
- the right child of $x$ (if any) and its descendants are in $N^+_T(x)$,

see Figure 2.

The order associated to $S$, denoted $<_S$, is the left-to-right order, i.e. the one which places a node $x$ after its left child and its descendants, but before its right child and its descendants. Such an order is called a BST order.

Remark that because $T$ is only a tournament and not an order as in a standard BST, there is no restriction on the direction of edges between the left and right subtrees of $x$. On the other hand, if $x$ is an ancestor of $y$, then there is an edge oriented from $x$ to $y$ if and only if $x <_S y$. Thus we have

**Remark 4.1.** In a tournament $T$, any branch $B$ of a BST $S$ forms a chain which coincides with $<_S$. That is, for $x, y \in B$, the edge in $T$ is oriented from $x$ to $y$ if and only if $x <_S y$.

We will now define chain quasi-orders, which are FO definable quasi-orders with which we will approximate BST orders. Let $C$ be a chain in $T$. Its chain quasi-order $\leq_C^+$ is defined as follows. Enumerate the vertices of $C$ as $c_1, \ldots, c_k$ so that edges are oriented from $c_i$ to $c_j$ when $i < j$. Define $A_i = \bigcap_{j \leq i} N^+(c_j)$, and $B_i = A_{i-1} \cap N^-(c_i)$. Then each of $B_1, \ldots, B_k$ and $A_k$ is an equivalence class of $\leq_C^+$, and the classes are ordered as

$$B_1 <_C^+ c_1 <_C^+ B_2 <_C^+ c_2 <_C^+ \cdots B_k <_C^+ c_k <_C^+ A_k,$$

see Figure 3. This can be seen as the left-to-right order of a partial BST consisting only of a single branch $c_1, \ldots, c_k$, with $c_1$ as root and $c_k$ as leaf. It is also a coarsening of the lexicographic order: the latter would refine the order inside each class $B_i$ using $c_{i+1}, \ldots, c_k$.

The dual quasi-order $\preceq_C^-$ is defined in the same way, but reversing the direction of all edges. Thus, we now enumerate $C$ so that edges are from $c_i$ to $c_j$ when $i > j$, while $A_i = \bigcap_{j \leq i} N^-(c_i)$ and $B_i = A_{i-1} \cap N^+(c_i)$. The rest of the definition is the same.
Figure 3. Example of construction of the quasi-order $\preceq_C^+$. The quasi-order is from left to right, and the triangles are equivalence classes. The direction of omitted edges (from $B_i$ to $B_j \cup \{c_j\}$ for $i < j$) is not constrained. For $\preceq_C^-$, the direction of all edges would be reversed.

Lemma 4.2. There is a transduction $\Phi$ such that for any tournament $T$ and chain quasi-ordering $\preceq$, we have $(T, \preceq) \in \Phi(T)$.

Proof. The transduction $\Phi$ first uses a colouring step to guess a chain $C$, and an orientation $o \in \{+, -\}$. Depending on this non-deterministic choice, it may yield any chain quasi-ordering. The remainder of $\Phi$ is a deterministic interpretation. For simplicity, we will assume that the orientation $o$ is $+$, the case of $-$ being similar.

Having guessed $C$, the ordering $\preceq_C^+$ inside $C$ is simply given by the orientation of edges. Enumerate $C$ as $\{c_1 \preceq_C^+ \ldots c_k\}$. Given $x \not\in C$, let $i(x) \in \{0, \ldots, k\}$ be maximal such that there are edges oriented $c_j \to x$ for all $1 \leq j \leq i(x)$. Then

- for any $x \not\in C$, we have $c_1 \ldots c_{i(x)} \preceq_C^+ x \preceq_C^+ c_{i(x)+1} \ldots c_k$, and
- for $x, y \not\in C$, we have $x \preceq_C^+ y$ if and only if $i(x) \leq i(y)$.

This characterisation of $\preceq_C^+$ can be expressed by a first-order formula. □

The main result of this section allows in a sense to approximate BST orderings with chain quasi-orders:

Lemma 4.3. There is a function $f(k) = 2^{O(k)}$ such that for any tournament $T$, BST ordering $<_S$, and family $\mathcal{P}$ of at least $f(k)$ disjoint intervals of $<_S$, one can find a chain quasi-ordering $\preceq$ and a subfamily $\mathcal{P}' \subset \mathcal{P}$ of at least $k$ parts such that any parts $X, Y \in \mathcal{P}'$ satisfy $X < Y$ or $Y < X$.

Furthermore, $C$, $o$, and $\mathcal{P}'$ can be computed in linear time.

Proof. Let $T$ be a tournament, $S$ a BST of $T$ and $<_S$ the corresponding order. Consider a family $\mathcal{P}$ of at least $f(k)$ disjoint intervals of $<_S$, where $f(k) = 2^{O(k)}$ will be determined later.

We choose a branch $B = b_0, \ldots, b_p$ of $S$ by the following process. First $b_0$ is the root of $S$. For each (yet to be determined) $b_i$, let $S_i$ be the subtree of $S$ rooted in $b_i$, and define the weight $w_i$ to be the number of classes of $\mathcal{P}$ intersected by $S_i$. Then $b_{i+1}$ is chosen to be the child of $b_i$ which maximizes $w_{i+1}$. This choice ensures that

$$2w_{i+1} + 1 \geq w_i.$$  

For each $i < p$, let $d_i$ be the child of $b_i$ other than $b_{i+1}$ (sometimes $d_i$ does not exist), and let $D_i$ be the subtree of $S$ rooted at $d_i$ ($D_i$ is empty if $d_i$ does
not exist). Furthermore, let $L, R$ be the sets of vertices which are before, resp. after the leaf $b_p$ in the order $<_S$. For any $0 \leq i \leq j \leq p$, let

$$L_{i,j} \overset{\text{def}}{=} \bigcup_{i \leq \ell < j} \{b_{\ell}\} \cup D_\ell, \quad \text{and} \quad R_{i,j} \overset{\text{def}}{=} \bigcup_{i \leq \ell < j} \{b_{\ell}\} \cup D_\ell.\]$$

Roughly speaking, $L_{i,j}$, resp. $R_{i,j}$ consists of subtrees branching out of $B$ on the left, resp. right, between $b_i$ and $b_j$.

**Claim 4.4.** For any $i, j$, the subtree $S_i$ is partitioned into $L_{i,j} <_S S_j <_S R_{i,j}$.

**Proof.** Clearly $L_{i,j}, S_j, R_{i,j}$ partition $S_i$. Furthermore, if $\ell < j$ and $b_{\ell} \in L$, then $b_{\ell} <_S S_j$, and in turn $D_\ell <_S b_{\ell}$. This proves $L_{i,j} <_S S_j$, and symmetrically $S_j <_S R_{i,j}$. $\blacksquare$

**Claim 4.5.** For $0 \leq i < j \leq p$, if $w_i \geq w_j + 3$, then there is a part $P \in \mathcal{P}$ such that $P \subseteq L_{i,j}$ or $P \subseteq R_{i,j}$.

**Proof.** There are at least three parts of $\mathcal{P}$ which intersect $S_i$ but not $S_j$. Since these three parts and $S_i$ are all intervals of $<_S$, one of these parts, say $P$, is contained in $S_i$. Thus $P$ is a subset of $S_i$ but does not intersect $S_j$, which using Claim 4.4 implies that $P \subseteq L_{i,j}$ or $P \subseteq R_{i,j}$. $\blacksquare$

Construct a sequence $i_0 < \cdots < i_{2k}$ of indices in $\{0, \ldots, p\}$ inductively by taking $i_0 = 0$, and choosing $i_{\ell+1}$ minimal such that $w_{i_{\ell+1}} \leq w_{i_\ell} - 3$.

Using (3) and the minimality of $i_{\ell+1}$, we obtain for all $\ell$ that

\begin{equation}
2w_{i_{\ell+1}} + 1 \geq w_{i_{\ell+1} - 1} > w_{i_\ell} - 3,
\end{equation}

\begin{equation}
\text{hence} \quad 2w_{i_{\ell+1}} + 3 \geq w_{i_\ell}.
\end{equation}

We can now define $f$ by $f(0) = 1$ and $f(k+1) = 4f(k) + 9$. By assumption, $w_0 = |\mathcal{P}| \geq f(k)$, and it follows from (3) that the construction of $i_{\ell}$ can be carried out up to $i_{2k}$.

Define $L'_\ell = L_{i_{\ell-1}, i_\ell}$, and similarly $R'_\ell = R_{i_{\ell-1}, i_\ell}$, see Figure 4. By Claim 4.5, for any $\ell \in [2k]$, either $L'_\ell$ or $R'_\ell$ contains a part of $\mathcal{P}$. Thus, either there are at least $k$ distinct $L'_\ell$ containing a part of $\mathcal{P}$, or there are at least $k$ distinct $R'_\ell$ containing a part of $\mathcal{P}$. Without loss of generality, assume that we are in the former case. We will now forget about vertices which are not in $L$.

Define $C = L \cap B$. By Remark 4.4 this is a chain, whose order coincides with $<_S$. Furthermore, at any node $x$ of $C$, the branch $B$ does descend on the right side, since $x <_S b_p$. Thus, the order in $C$ also coincides with the ancestor-descendent order of $S$. (Remark here that if we were in $R$ instead of $L$, the order of $C$ would be the inverse of the ancestor-descendent order.) Now, if $C$ is enumerated as $c_0 <_C \cdots <_C c_t$, and $C_i$ is the subtree branching out on the left of $c_i$, defined similarly to $D_i$, then the chain quasi-order $\preceq^+_C$ restricted to $L$ is exactly

$$C_0 <_C c_0 <_C C_1 <_C c_1 <_C \cdots <_C c_t$$

where each subtree $C_i$ is an equivalence class. (In $R$, we would instead use $\preceq^+_C$. From this description, we obtain that any $L_{i,j}$ is an interval of $\preceq^+_C$ restricted to $L$.}
Figure 4. Sketch of the proof of Lemma 4.3. In the upper half, the BST $T$ with the extracted branch $B$; circled in blue, the extracted subsequence $b_{i_\ell}$; in green arrows, the chain $C = B \cap L = \{b_2, b_4, b_5, b_6\}$. Below the tree, from top to bottom: the partition in $L'_{\ell}$ and $R'_{\ell}$; the initial family (here partition) $P$, with the parts contained in some $L'_{\ell}$ or $R'_{\ell}$ highlighted; the final family $P'$, obtained by selecting a part of $P$ inside each possible $L'_{\ell}$.

For each $L'_{\ell}$, select a part of $P$ included in $L'_{\ell}$ if any, and define $P'$ as the collection of selected parts. Thus $P' \subset P$, and we know from the choice of the family $\{L'_\ell\}_{\ell \in [2k]}$ that $|P'| \geq k$. Furthermore, if $X \neq Y$ are parts of $P'$, there are $s \neq t$ such that $X \subseteq L'_{s}$ and $Y \subseteq L'_{t}$. Since the $L'_{\ell}$ are disjoint intervals of $(L, \gtrless_C)$, we have either $L'_{s} \gtrless^+_C L'_{t}$ or $L'_{t} \gtrless^+_C L'_{s}$, hence a fortiori either $X \gtrless^+_C Y$ or $Y \gtrless^+_C X$. Thus $P'$ satisfies all desired properties.

Finally, given the BST $S$ and the family $P$, it is routine to compute the weights $w_i$ of all nodes in $S$ by a bottom-up procedure; this only requires to compute the left-most and right-most parts of $P$ intersecting each subtree. From this, it is simple to choose in linear time the branch $B$, the indices $i_\ell$, the better side $L$ or $R$, and finally to compute $C$ and $P'$.

5. Approximating twin-width

In this section, we prove Lemma 1.4 for any tournament $T$ and BST ordering $\prec$, if $T$ has small twin-width, then so does the ordered structure $(T, \prec)$. This implies an FPT approximation of twin-width in tournaments (Theorem 1.3).

To this end, we show using Theorem 2.3 and Lemma 4.3 that if $(T, \prec)$ has large twin-width, then $T$ contains a so-called chain order representation of a matrix with high grid rank. It is relatively simple to show that these
Figure 5. Chain representation of the matrix $M$ of the permutation 31452. Vertices of $A$ are ordered bottom to top by a chain order $\preceq_{CA}$, and similarly for $B$. Edges oriented from $B$ to $A$ correspond to ‘1’s in $M$. For readability, edges from $A$ to $B$ (corresponding to ‘0’s) are not drawn. The other omitted edges are unconstrained.

chain order representations are obstructions to twin-width. In section 6, we will improve this result by extracting the canonical obstructions $\mathcal{F}_=, \mathcal{F}_<, \mathcal{F}_>$ from chain order representations.

5.1. Chain order representation. Let $M$ be a 0,1-matrix and $T$ a tournament. A chain order representation of $M$ in $T$ consists of two disjoint subsets $A, B$ of vertices and two chain quasi-orders $\preceq^A, \preceq^B$ such that

- restricted to $A$ and $B$ respectively, $\preceq^A$ and $\preceq^B$ are strict orderings,
- and the adjacency matrix of $(A, \preceq^A)$ versus $(B, \preceq^B)$ is $M$.

The subsets $A, B$ should not be confused with the chains from which the orders $\preceq^A, \preceq^B$ are defined. See Figure 5 for an example.

Lemma 5.1. Let $T$ be a class of tournaments which contains chain order representations of matrices with rank-$k$ divisions for arbitrarily large $k$. Then $T$ has unbounded twin-width.

Proof. Consider a chain representation of $M$ in $T$ defined by subsets $A, B$ and chain orders $\preceq^A, \preceq^B$. Construct the ordered structure $(T', <)$ where $T' = T[A \cup B]$ is the subtournament induced by $A$ and $B$, and the ordering $<$ is obtained by combining $\preceq^A$ and $\preceq^B$: $<$ coincides with $\preceq^A$ and $\preceq^B$ inside $A$ and $B$ respectively, and $A < B$. Thus $M$ is contained in the adjacency matrix of $T'$ ordered by $<$. Call $T'$ the class of all such ordered structures constructed from $T \in T$. By assumption, structures in $T'$ have adjacency matrices with arbitrarily high rank divisions, hence by Theorem 2.3 $T'$ has unbounded twin-width.

Using Lemma 4.2, it is simple to show that there is a fixed first-order transduction $\Phi$ which given a tournament $T$, produces any ordering $<$ of $T$ obtained by combining two chain orders as previously. It follows that $T'$ is obtained by first-order transduction from $T$. Since $T'$ has unbounded twin-width, this implies by Theorem 2.3 that $T$ has unbounded twin-width. □
5.2. Self-extraction in high rank divisions. The last tool needed to prove Lemma 5.2 is the following self-extraction lemma for high-rank divisions.

**Lemma 5.2.** Let $M$ be a matrix with a rank-$k^3$ division $(\mathcal{R}, \mathcal{C})$. Then $M$ has a submatrix $M'$ with a rank-$k$ division such that $M'$ contains only one row (resp. column) from each part of $\mathcal{R}$ (resp. $\mathcal{C}$).

Lemma 5.2 will be used as follows: from a high-rank division whose parts are strictly ordered by a chain-quasi order $\leq$ (while allowing elements of the same part to be equivalent under $\leq$), it yields a high-rank division on a subset of elements strictly ordered by $\leq$.

We first prove that a high rank matrix can be extracted.

**Lemma 5.3.** Let $M$ be a matrix with a rank-$k$ division $(\mathcal{R}, \mathcal{C})$. Then there exists a submatrix $M'$ with rank $k$ containing only one row (resp. column) from each part of $\mathcal{R}$ (resp. $\mathcal{C}$).

**Proof.** Enumerate the blocks $\mathcal{R} = \{R_1, \ldots, R_k\}$ and $\mathcal{C} = \{C_1, \ldots, C_k\}$ of the division. Assume by induction that we have already extracted one row (resp. column) from each of $R_1, \ldots, R_{k-1}$ (resp. $C_1, \ldots, C_{k-1}$) to form a submatrix $M_{k-1}$ of rank $k-1$. Now add all rows of $R_k$ and all columns of $C_k$ to $M_{k-1}$. The resulting matrix has rank at least $k$ since it contains $R_k \times C_k$.

Using basis exchange, all but one row of $R_k$ can be removed while preserving rank $k$, and similarly all but one column of $C_k$ can be removed, yielding the desired submatrix. \(\square\)

**Proof of Lemma 5.2.** Let us enumerate the blocks of the division $(\mathcal{R}, \mathcal{C})$ as $\mathcal{R} = \{R_0 < \cdots < R_{k^3-1}\}$ and $\mathcal{C} = \{C_0 < \cdots < C_{k^3-1}\}$. We will use two levels of coarser partitions, regrouping blocks $k$ by $k$: define the partition $\mathcal{R}'$ with parts $R'_i = R_{ki} \cup \cdots \cup R_{k(i+1)-1}$ for $i < k^2$, and $\mathcal{R}''$ with parts $R''_i = R'_{ki} \cup \cdots \cup R'_{k(i+1)-1}$ for $i < k$, and similarly $\mathcal{C}'$ and $\mathcal{C}''$. Thus $(\mathcal{R}', \mathcal{C}')$ is a rank-$k^2$-division of $M$, while $(\mathcal{R}'', \mathcal{C}'')$ is a rank-$k$-division.

Now for each $i, j \in [k]$, we consider the submatrix $R'_{ki+j} \times C''_{kj+i}$. Remark that these submatrices all have disjoint sets of rows and columns. Further, $R'_{ki+j} \times C''_{kj+i}$ contains $k$ parts of $\mathcal{R}$ and of $\mathcal{C}$, hence Lemma 5.3 can be applied to extract a rank-$k$ submatrix $M_{i,j}$ using only one row or column from the relevant blocks of $\mathcal{R}$ and $\mathcal{C}$. Define $M'$ to be the submatrix formed only by the rows and columns used in some $M_{i,j}$. The former construction ensures this is only one row or column from each part of $\mathcal{R}$ and $\mathcal{C}$. In $M'$, $(\mathcal{R}''', \mathcal{C}''')$ is a rank-$k$-division whose cell $R'''_{ki+j} \times C'''_{kj+i}$ contains the rank $k$ submatrix $M_{i,j}$. Therefore $M'$ has a rank-$k$ division. \(\square\)

5.3. Finding high grid rank representations. We are now ready to prove that BST orders yield an approximation of twin-width in tournaments. Given a class $\mathcal{T}$ of tournaments, denote by $\mathcal{T}^{\text{BST}}$ the class of ordered tournaments $(T, <_S)$ with $T \in \mathcal{T}$ and $<_S$ some BST ordering of $T$.

**Lemma 5.4.** Let $\mathcal{T}$ be a class of tournaments. If $\mathcal{T}^{\text{BST}}$ has unbounded twin-width, then $\mathcal{T}$ contains chain order representations of matrices with rank-$k$ divisions for arbitrarily large $k$. 

Proof. Assume that $\mathcal{T}^{\text{BST}}$ has unbounded twin-width. Since it is a class of ordered structures, we can apply the results of [9] to obtain matrices with large grid rank:

Claim 5.5. For any $k$, there exists $(T, <) \in \mathcal{T}^{\text{BST}}$ and intervals $A_1 < \cdots < A_k < B_1 < \cdots < B_k$ of $V(T)$ such that for any $i, j$, the adjacency matrix of $A_i$ versus $B_j$ has rank $k$.

Proof. By Theorem 2.3, ordered structures with sufficiently large twin-width have rank-$k$ divisions in their adjacency matrix, for any given $k$. For $\mathcal{T}^{\text{BST}}$ which has unbounded twin-width, this yields $(T, <) \in \mathcal{T}^{\text{BST}}$ and a rank-$(2k)$ division consisting of two partitions $A_1 < \cdots < A_{2k}$ and $B_1 < \cdots < B_{2k}$ of $V(T)$ such that the adjacency matrix of $A_i$ versus $B_j$ has rank $k$.

Let us ensure that the $A_i$s are disjoint from the $B_j$s. If the last element of $A_k$ is smaller than the first of $B_{k+1}$, then $A_1 < \cdots < A_k < B_{k+1} < \cdots < B_{2k}$ satisfy the desired property. Otherwise, the last element of $B_k$ is smaller than the first of $A_{k+1}$, and symmetrically $B_1 < \cdots < B_k < A_{k+1} < \cdots < A_{2k}$ are as desired.

We now wish to replace the BST ordering $<$ of Claim 5.5 with chain quasi-orders.

Claim 5.6. For any $k$, there exists $T \in \mathcal{T}$, two chain quasi-orders $\preceq^A, \preceq^B$ in $T$, and subsets $A_1 \preceq^A \cdots \preceq^A A_k$ and $B_1 \preceq^B \cdots \preceq^B B_k$ of $V(T)$ with all $A_i$s and $B_j$s pairwise disjoint, such that the adjacency matrix of $A_i$ versus $B_j$ has rank $k$.

Proof. Applying Claim 5.5 with $f(k)$ where $f$ is the function of Lemma 4.3, there exists $T \in \mathcal{T}$ and intervals $A_1 < \cdots < A_{f(k)} < B_1 < \cdots < B_{f(k)}$ of some BST order $<$ such that the adjacency matrix of $A_i$ versus $B_j$ has rank $k$. Applying Lemma 4.3 to the $A_i$s and the $B_j$s separately yields subfamilies of size at least $k$, both of which are strictly ordered by some chain quasi-order.

To obtain the chain order representation from Claim 5.6, we only need to ensure that $\preceq^A, \preceq^B$ are also strict orderings inside each part $A_i$ or $B_j$. This is obtained thanks to Lemma 5.2.

Apply Claim 5.6 with $k^3$, yielding parts $A_i, B_j$ ordered by $\preceq^A, \preceq^B$. Consider the adjacency matrix $M$ of $\bigcup_i A_i$ against $\bigcup_j B_j$ ordered by $\preceq^A$ and $\preceq^B$ respectively, breaking equivalences in the quasi-orderings arbitrarily. Since the matrix of $A_i$ against $B_j$ has rank at least $k$, $M$ has a rank-$(k^3)$ division. We then apply Lemma 5.2 to extract $A, B$ with only one element in each $A_i, B_j$, so that the restriction to $A, B$ has a rank-$k$ division. Then the chain quasi-orderings $\preceq_A, \preceq_B$ strictly order $A$ and $B$. Thus $A, B$ define a chain order representation of a matrix with a rank-$k$ division, as desired.

From Lemmas 5.1 and 5.4, we immediately obtain that if $\mathcal{T}^{\text{BST}}$ has unbounded twin-width, then so does $T$, or equivalently:

Lemma 1.4. There is a function $f$ such that for any tournament $T$ and any BST order $<$,

$$\text{tww}(T, <) \leq f(\text{tww}(T))$$
Using Lemma 1.4, we obtain an FPT approximation algorithm for twin-width of tournaments.

**Theorem 1.3.** There are functions \( f, g : \mathbb{N} \to \mathbb{N} \) and an algorithm which given a tournament \( T \) with twin-width \( k \), produces a witness that the twin-width of \( T \) is at most \( f(k) \) in FPT time \( g(k) \cdot |T|^{O(1)} \).

**Proof.** Given a tournament \( T \), construct an arbitrary BST ordering \(<\). By Lemma 1.4, \( \text{tww}(T, <) \) is bounded by some function of \( \text{tww}(T) \). We then apply to the ordered structure \((T, <)\) the approximation algorithm of ordered structures [9, Theorem 2], which is FPT. It finds a contraction sequence for \((T, <)\) of width bounded by a function of \( \text{tww}(T, <) \), hence by a function of \( \text{tww}(T) \). The same contraction sequence is also valid for \( T \), and forgetting the ordering \(<\) does not increase its width. \(\square\)

6. Extracting forbidden tournaments

The goal of this section, the last part of the proof of our main result, is to show that any class of tournaments with unbounded twin-width contains one of the classes of obstructions described in section 3.

Recall that \( T_{\text{BST}} \) denotes the class of ordered tournaments \((T, <_S)\) where \( T \in T \) and \( <_S \) is some BST ordering of \( T \). The precise statement proved in this section is the following.

**Theorem 6.1** (Theorem 1.5 restated). Let \( T \) be a hereditary class of tournaments. If \( T_{\text{BST}} \) has unbounded twin-width, then \( T \) contains one of the classes \( F_\leq, F_\geq \).

Since the classes \( F_\leq, F_\geq \) have unbounded twin-width, this strengthens Lemma 1.4. The proof builds upon that of Lemma 5.4 in the previous section. The latter constructed chain order representations of matrices with high rank divisions. From these, we will extract encodings of arbitrary permutations thanks to a major technical result of [9], and further refine these encodings to obtain the classes \( F_\leq, F_\geq \) using Ramsey arguments.

6.1. Encoding permutations in matrices. Let us describe the encodings of permutations in matrices used as obstructions to twin-width in [9]. They define six classes \( M_\leq, M_\geq, M_{\leq R}, M_{\geq R}, M_{\leq C}, M_{\geq C} \) of matrices. Like the classes of tournaments described in section 3, each class \( M_\sigma \) consists of the hereditary closure of encodings \( M_\sigma(\sigma) \) of arbitrary permutations \( \sigma \). For a permutation \( \sigma \) on \( n \) elements, the \( n \times n \) matrix \( M_\sigma(\sigma) = (m_{i,j})_{i,j \in [n]} \) is described by

\[
\begin{align*}
M_\leq(\sigma) : & \quad m_{i,j} = 1 \iff j = \sigma(i) \\
M_\geq(\sigma) : & \quad m_{i,j} = 1 \iff j \neq \sigma(i) \\
M_{\leq R}(\sigma) : & \quad m_{i,j} = 1 \iff j \leq \sigma(i) \\
M_{\geq R}(\sigma) : & \quad m_{i,j} = 1 \iff j \geq \sigma(i) \\
M_{\leq C}(\sigma) : & \quad m_{i,j} = 1 \iff i \leq \sigma^{-1}(j) \\
M_{\geq C}(\sigma) : & \quad m_{i,j} = 1 \iff i \geq \sigma^{-1}(j)
\end{align*}
\]

The classes are called \( F_\leq \), etc. in [9], we use \( M_\leq \) to avoid ambiguity with tournaments defined in section 3.
Thus $M_{=}(\sigma)$ is the usual permutation matrix, $M_{\neq}(\sigma)$ is its complement, and $M_{\leq_R}, M_{\geq_R}, M_{\leq_C}, M_{\geq_C}$ are obtained from the permutation matrix by propagating the ‘1’s down, up, left, and right respectively.

The main technical result of \cite{22} is the following.

**Theorem 6.2 (\cite{22} Theorem 27).** There is a function $f$ such that in any matrix $M$ with a rank-$f(k)$ division, there is an $s \in \{=, \neq, \leq_R, \geq_R, \leq_C, \geq_C\}$ such that $M_s(\sigma)$ is a submatrix of $M$ for any permutation $\sigma$ on $k$ elements.

Consider now a class $\mathcal{T}$ such that $\mathcal{T}^{\text{BST}}$. By Lemma \ref{lem:bst} $\mathcal{T}$ contains chain order representations of matrices with arbitrarily high rank divisions. Applying Theorem \ref{thm:main} to these matrices, we obtain the following.

**Claim 6.3.** For any permutation $\sigma$, there exists $s \in \{=, \neq, \leq_R, \geq_R, \leq_C, \geq_C\}$ such that some $T \in \mathcal{T}$ contains a chain order representation of $M_s(\sigma)$.

### 6.2. Permutations between chains

Claim \ref{clm:chain_order} proves that there are chain order representations of arbitrary permutations. Such a representation involves four subsets: the $A, B$ which induce $M_s(\sigma)$ as adjacency matrix, and the two chains from which the orderings of $A$ and $B$ are defined. To simplify this picture, this section shows that one can find encodings of permutations between two chains themselves, in the following sense.

**Claim 6.4.** For any permutation $\sigma \in \mathfrak{S}_n$, there exists in some $T \in \mathcal{T}$ one of the following two structures:

1. Either two disjoint subsets $A = \{a_1 <_A \cdots <_A a_n\}$, $B = \{b_1 <_B \cdots <_B b_n\}$ of vertices such that
   - $A$ is a chain which corresponds to the order $<_A$ up to reversal, meaning that either for all $i < j$ there is an edge $a_i \rightarrow a_j$, or for all $i < j$ there is an edge $a_j \rightarrow a_i$,
   - similarly $B$ is a chain corresponding to $<_B$ up to reversal, and
   - the adjacency matrix of $(A, <_A)$ versus $(B, <_B)$ is $M_s(\sigma)$ for some $s \in \{=, \neq, \leq_R \geq_R, \leq_C, \geq_C\}$.

2. Or a subset $A = \{a_1 < \cdots < a_n\}$ and a chain order $\leq$ such that
   - $\leq$ totally orders $A$,
   - $A$ is a chain which corresponds to $\leq$ up to reversal, and
   - the bi-order $(A, <, \leq)$ is isomorphic to either $O_{\sigma}$ or $O_{\sigma-1}$.

**Proof.** Let $\tau$ be a permutation, which will be chosen as a function of $\sigma$, so as to apply Lemma \ref{lem:chain_order}. By Claim \ref{clm:chain_order} there exists a chain order representation of $\tau$, consisting of disjoint subsets $A, B$ ordered by $\leq_1, \leq_2$ respectively, such that the adjacency matrix of $A$ versus $B$ for these orders is $M_s(\tau)$ for some $s \in \{=, \neq, \leq_R \geq_R, \leq_C, \geq_C\}$. We enumerate $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_n\}$ so that

$$a_i \prec_1 a_j \iff i < j \quad \text{and} \quad b_i \prec_2 b_j \iff \tau(i) < \tau(j)$$

Thus, the map $\phi : A \rightarrow B$ defined as $\phi(a_i) = b_i$ is the bijection implicitly defined by $M_s(\tau)$. We further define the order $\leq_1'$ on $B$ as the image of $\leq_1$ by $\phi$, and similarly $\leq_2'$ is the inverse image of $\leq_2$. Or, more simply

$$a_i \prec_2 a_j \iff \tau(i) < \tau(j) \quad \text{and} \quad b_i \prec_1' b_j \iff i < j$$

Recall that the biorder $O_{\tau} = ([n], <, \tau)$ consists of the natural order $<$, and the permutation order defined by $i <_{\tau} j$ if and only if $\tau(i) < \tau(j)$. It
is isomorphic to \((A, \preceq_1, \preceq'_2)\), and to \((B, \preceq'_1, \preceq_2)\), by the maps \(i \mapsto a_i\) and \(i \mapsto b_i\) respectively. On this structure, we define a coloring \(\lambda : [n]^2 \to \{0, 1\}^2\) which combines the adjacency matrices of \(T[A]\) and \(T[B]\), that is

\[
\lambda(i, j) = (a, b) \quad \text{where } a = 1 \text{ (resp. } b = 1) \text{ iff } a_i \to a_j \quad (b_i \to b_j) \text{ is an edge}
\]

We now suppose that \(\tau\) was chosen so that Lemma 2.1 can be applied to obtain a monochromatic embedding of \(\sigma\). That is, there exists \(Q \subset [n]\) such that \(O_\tau[Q]\) is isomorphic to \(O_\sigma\), and the coloring \(\lambda\) restricted to \(Q \times Q\) depends only on the orders \(<_\sigma\). Let \(A' \equiv \{a_i \mid i \in Q\}\) and \(B' \equiv \{b_i \mid i \in Q\}\) be the corresponding subsets. They form a chain representation of \(M_\sigma\). Furthermore, the direction of edges in \(T[A']\), resp. \(T[B']\), depends only on the orders \(\preceq_1, \preceq'_2\), resp. \(\preceq'_1, \preceq_2\). By Lemma 2.2, this means that \(T[A']\) is a transitive tournament whose order corresponds to—in the sense of equal to or inverse of—either \(\preceq_1\) or \(\preceq'_2\), and similarly in \(B'\). We now have to consider two cases.

1. If edges in \(T[A']\) correspond to the order \(\preceq_1\), and edges in \(T[B']\) correspond to the order \(\preceq_2\), then the chain orders become redundant: we have obtained \(M_\sigma\) as adjacency matrix between two chains, up to some reversal of orders. This is Case 1 of the claim.

2. Otherwise, if for instance the edges in \(T[A']\) correspond to the order \(\preceq'_2\), then \(A'\) is equipped with both the orders \(\preceq'_1\) (as chain order) and \(\preceq'_2\) (from the direction of edges), which together encode \(\sigma\) as a bi-order. This is Case 2 of the claim. The situation is similar when the edges of \(T[B']\) correspond to \(\preceq'_1\).

6.3. Canonical obstructions. In this section, we finally extract the obstructions of section 3 from the structure described by Claim 6.4.

Claim 6.5. For any \(k\) there exists a permutation \(\sigma\) containing a \(k\)-grid and some \(s \in \{=, \leq, \geq\}\) such that \(F_s(\sigma) \in T\).

Proof. We start with a permutation \(\sigma_1\) containing sufficiently large grids, which we will choose function of the desired \(k\). We apply Claim 6.4 to \(\sigma_1\), and consider the two possible cases.

Case 1. Assume first that we are in Case 1 of Claim 6.4 i.e. there are two chains \(A, B\) with adjacency matrix \(M_\sigma(\sigma_1)\) for some \(s \in \{=, \neq, \leq_R, \geq_R, \leq_C, \geq_C\}\). We assume that \(\sigma_1\) was chosen to contain a \((k + 1)\)-grid.

In this representation of \(\sigma_1\), the direction of edges inside \(A, B\) is only controlled up to reversal of their order. This is not an issue: reversing the order in \(A\), resp. \(B\), corresponds to reversing the order of columns, resp. rows, of \(M_\sigma(\sigma_1)\), which yields \(M_\sigma'(\sigma_2)\) for some other \(\sigma'(\sigma_2)\), and some new permutation \(\sigma_2\). This \(\sigma_2\) is obtained by pre- or post-composing \(\sigma_1\) with the map \(i \mapsto (n + 1) - i\). This transformation preserves grids, hence \(\sigma_2\) also has a \((k + 1)\)-grid. Thus we can assume that the direction of edges in \(A, B\) exactly matches the order of rows and columns in the matrix.

Next, we can assume that \(s\) is one of \(=, \leq_R, \geq_R\). Indeed, if this is not the case, we swap the roles of \(A\) and \(B\), which corresponds to transposing and complementing (i.e. swapping ‘0’ for ‘1’ and vice versa) the adjacency matrix. If the matrix was \(M_{=}(\sigma_2)\) before, this yields \(M_{=}(\sigma_2)^{-1}\) instead. If the
matrix was \( M_{\leq C}(\sigma_2) \), the situation is only slightly more complex: transposing and complementing yields \( M_{\geq R}(\sigma_2^{-1}) \), meaning the matrix defined as \( M_{\leq R}(\sigma_2^{-1}) \), except for entries \((i, \sigma_2^{-1}(i))\) which are ‘0’ instead of ‘1’. However, dropping the first row and the column with index \( \sigma_2(n) \)—both of which contain only ‘0’s—yields a matrix of the form \( M_{\geq R}(\sigma_3) \). One can then verify that \( \sigma_3 \) still contains a \( k \)-grid.

In all cases, we obtain some permutation \( \sigma_3 \) with a \( k \)-grid, and two chains between which the adjacency matrix is \( M_{=}(\sigma_3) \), \( M_{\leq R}(\sigma_3) \), or \( M_{\geq R}(\sigma_3) \). This gives exactly one of the structures \( F_\sigma(\sigma_1^{-1}) \), \( F_\sigma(\sigma_2^{-1}) \), \( F_\sigma(\sigma_3^{-1}) \), with the rows of the matrix corresponding to the set called \( Y \) in section 3.

\[ \text{Case 2} \]

We now assume that we are in case 2 of Claim 6.4, i.e. there is a set of vertices \( A \) with a chain order \( \leq \), and a second order \( < \), such that the edges of \( A \) coincide with \( < \), possibly up to reversal, and \( (A, <, \leq) \) is isomorphic to either \( O_{\sigma_1} \) or \( O_{\sigma_2^{-1}} \). The reversal of \( < \), and swapping \( \sigma_1 \) for \( \sigma_1^{-1} \) both preserve the size of grids. Thus we can assume that edges in \( A \) are \( u \to v \) if and only if \( u < v \), and that \( (A, <, \leq) \) is isomorphic to \( O_{\sigma_1} \).

Consider the chain \( C \) and orientation \( o \) which induce \( \leq \), i.e. \( \leq = \leq_Y^o \). For simplicity, we can assume that the orientation \( o \) is ‘+’, the proof is similar in the case of ‘−’.

A simple grid argument (and very specific case of Lemma 2.1) allows to extract a similar substructure for a smaller \( \sigma_2 \), with the assumption that either \( C \) and \( A \) are disjoint, or \( A \subseteq C \). But the latter is clearly impossible unless \( \sigma_2 \) is the identity or its reverse, which we can assume is not the case. Thus, we can now assume that \( A \) and \( C \) are disjoint, and that the encoded permutation \( \sigma_2 \) still contains large grids.

Enumerate \( A \) in the order \( a_1 \leq^+ \ldots \leq^+ a_n \). For each \( i \) define \( c_i \in C \) to be minimal w.r.t. the internal order of \( C \) with the property that \( a_i \notin N^+(c_i) \). Because \( C \) and \( A \) are disjoint, such a \( c_i \) must exist except possibly when \( i = n \). At the cost of removing \( a_n \) from \( A \) if necessary, we assume there is also some \( c_n \) matching \( a_n \). The nodes \( c_i \) verify that there is an edge \( c_i \to c_j \) if and only if \( i < j \). We can now assume that the chain \( C \) consists only of \( c_1, \ldots, c_n \), as the other elements are not relevant to the order on \( A \). At this point, we have the following structure: \( C \) is a chain for the natural order \( c_1, \ldots, c_n \), \( A \) is a chain for the permuted order corresponding to \( \sigma_2 \), and there are edges \( a_i \to c_i \) for all \( i \), and \( c_i \to a_j \) for all \( i < j \). What remains unspecified is the direction of edges \( a_ic_j \) with \( i < j \).

We will again apply Lemma 2.1 to control these edges. Consider the bi-order \( O_{\sigma_2} = ([n], <, <_{\sigma_2}) \), and add a tournament on the vertices of this bi-order, where for \( i < j \) the direction of the edge \( ij \) matches that of \( a_ic_j \). Assuming that \( \sigma_2 \) contains a sufficiently large grid, we can apply Lemma 2.1 to this bi-order plus tournament structure to extract an encoding of any specified permutation \( \sigma_3 \). This encoding has exactly the same shape as before: two chains \( C, A \), where \( C \) induces a total chain order on \( A \), but with the additional requirement that for all \( i < j \), the direction of the edge \( a_ic_j \) only depends on whether or not \( i < j \) and whether or not \( i <_{\sigma_3} j \). Since we only consider edges with \( i < j \), the dependency on \( i < j \) is meaningless.

Then, there are only four cases to consider.
(1) If for all $i < j$, there is an edge $c_j \rightarrow a_i$, then $A$ and $C$ induce the structure $F_\prec(\sigma_3)$, and we are done as we can assume that $\sigma_3$ contains the desired permutation $\sigma$ as pattern.

(2) Similarly, if for all $i < j$, there is an edge $a_i \rightarrow c_j$, then $A$ and $C$ induce the structure $F_\succ(\sigma_3)$.

(3) Assume now that for $i < j$, there is an edge $a_i \rightarrow c_j$ if and only if $\sigma_3(i) < \sigma_3(j)$. Given a desired permutation $\sigma$ on $n$ elements, we choose $\sigma_3$ to be the permutation on $2n$ elements defined by

\[
\sigma_3(2i - 1) = \sigma(i) \quad \sigma_3(2i) = n + i
\]

Renaming $A = \{a_1, \ldots, a_{2n}\}$ and $C = \{c_1, \ldots, c_{2n}\}$, we now extract the odd indices in $A$ and the even indices in $C$, that is we define $a_i' = a_{2i-1}$ and $c_i' = c_{2i}$. For any $i, j$, remark that $\sigma_3(2i - 1) < \sigma_3(2j)$ always holds—this will allow us to get rid of the dependency on $<_\sigma_3$.

If $i > j$, then we have $2i - 1 > 2j$, hence there is an edge $c_j' \rightarrow a_i'$ (from the chain order structure) but if $i \leq j$, we instead have the edge $a'_i \rightarrow c'_j$ (by hypothesis on the orientation of edges $a_ic_j$ when $i < j$). Finally, on the extracted vertices $a'_1, \ldots, a'_n$, there is an edge $a'_i \rightarrow a'_j$ if and only if $\sigma_3(2i - 1) < \sigma_3(2j - 1)$, that is if and only if $\sigma(i) < \sigma(j)$. Thus, the structure induced by the $a'_i$ and $b'_j$ is exactly $F_\succ(\sigma)$.

(4) The proof is similar when there is an edge $a_i \rightarrow c_j$ if and only if $\sigma_3(i) < \sigma_3(j)$. \qed

To conclude the proof of Theorem \ref{thm:main} we apply Claim \ref{claim:diagram} to arbitrarily large $k$. By pigeonhole principle, this yields a family of permutations $(\sigma_n)_{n \in \mathbb{N}}$ with arbitrarily large grids, and a fixed $s \in \{=, \leq, \geq\}$ such that $F_s(\sigma_n) \in \mathcal{T}$. Since the pattern closure of such a family contains all permutations, it follows that $F_s \subseteq \mathcal{T}$.

7. Dense oriented graphs and relational structures

In this final section, we generalise our main result in two directions: we replace tournaments by oriented graphs with bounded independent sets, and consider relational structures formed from such oriented graphs augmented by arbitrary binary relations.

7.1. Generalised binary search trees. We consider oriented graphs $D$ (forbidding digons). The independence number $\alpha(D)$ is the maximum size of an independent set (subset of vertices with no arcs between them) in $D$. The oriented graph $D$ is a tournament if and only if $\alpha(D) = 1$. In a sense, oriented graphs whose independence number is bounded by a small constant resemble tournaments; we will generalise our results to them.

We first generalise the notion of BST to arbitrary oriented graphs. A BST on an oriented graph $D$ is a rooted ordered ternary tree $S$ (meaning that each node has a left, center, and right child, any of which may be missing) satisfying the following condition. For any node $x$ of the tree, let $L_x, C_x, R_x$ denote the left, center, and right subtrees of $x$. Then

\[
L_x \subseteq N^-(x) \quad C_x \cap N(x) = \emptyset \quad L_r \subseteq N^+(x).
\]
The left-to-right order $<_S$ corresponding to the tree is the order satisfying for all node $x$

$$L_x <_S x <_S C_x <_S R_x.$$  

The choice of placing $C_x$ after and not before $x$ here is arbitrary.

Any branch $B$ of the tree is an acyclic subgraph in which the direction of edges matches $<_S$. However, it is not necessarily a chain. Indeed, if $x \in B$ is such that the branch $B$ descends into the central subtree of $x$, then $x$ is non-adjacent to all its descendants in $B$. Nonetheless, $B$ can be transformed into a chain at a limited cost when $\alpha(D)$ is small: Let $X \subseteq B$ be the set of all such center-branching nodes in $B$. Then $X$ is an independent set and thus $|X| \leq \alpha(D)$, while the rest $B \setminus X$ of the branch is a chain.

Next, we generalise chain quasi-orders. For $C$ a chain in $D$ enumerated as $C = \{c_1, \ldots, c_k\}$ such that $c_i \rightarrow c_j$ iff $i < j$, we define

$$A_i = \left( \bigcap_{j<i} N^+(c_j) \right) \setminus C \quad \text{and} \quad B_i = A_{i-1} \setminus N^+(c_i).$$

The chain quasi-order $\preceq_C^+$ is then defined by

$$B_1 \prec_C^+ c_1 \prec_C^+ B_2 \prec_C^+ c_2 \prec_C^+ \ldots B_k \prec_C^+ c_k \prec_C^+ A_k,$$

with $B_1, \ldots, B_k$ and $A_k$ being the equivalence classes. That is, we order a node $x$ according to the smallest $i$ such that $x \notin N^+(c_i)$. The dual quasi-order $\preceq_C^-$ is defined in the same way, but reversing the direction of all edges.

We can now generalise the partition extraction lemma to oriented graphs with bounded $\alpha$.

**Lemma 7.1.** Let $D$ be an oriented graph with independence number $\alpha$, and let $S$ be a BST of $D$ with associated order $<_S$. There is a function $f(k) = 2^{O(k)}$ depending only on $\alpha$ satisfying the following. For any family $\mathcal{P}$ of at least $f(k)$ disjoint intervals of $<_S$, there is chain quasi-ordering $\preceq$ and a subfamily $\mathcal{P}' \subset \mathcal{P}$ such that $|\mathcal{P}'| \geq k$ and such that the parts of $\mathcal{P}'$ are non-overlapping for $\preceq$.

Furthermore, $C$, $\alpha$, and $\mathcal{P}'$ can be computed in linear time.

**Sketch of proof.** The proof of Lemma 4.3 can be applied to directed graphs with essentially no modification. The only issue is that it does not yield a chain quasi-order, because the branches of $S$ are not themselves chains. Instead, the quasi-order $\preceq$ compatible with the resulting partition $\mathcal{P}'$ can be described by the following structure (up to inversion of all edges). There is a sequence $C = \{c_1, \ldots, c_k\}$ of vertices. Each $c_i$ has a type: either complete or anti-complete. If $c_i$ is complete, then for all $j > i$ there is an edge $c_i \rightarrow c_j$. If $c_i$ is anti-complete, then for all $j > i$ there is no edge between $c_i$ and $c_j$. We then define subsets $A_j$ by induction: $A_0 = V(T) \setminus C$, and $A_1$ is defined as $A_{i-1} \cap N^+(c_i)$ when $c_i$ is complete or as $A_{i-1} \setminus N^+(c_i)$ when $c_i$ is anti-complete. Then $B_i$ is defined as $A_{i-1} \setminus A_i$, and the quasi-order is

$$B_1 \prec_C^+ c_1 \prec_C^+ B_2 \prec_C^+ c_2 \prec_C^+ \ldots B_k \prec_C^+ c_k \prec_C^+ A_k,$$

When there are no anti-complete nodes in $C$, this structure is exactly a chain quasi-order.

Remark now that the anti-complete nodes in $C$ induce an independent set. It follows that there are at most $\alpha$ anti-complete nodes. Then, we
can simply remove any anti-complete node \( c_i \), and remove from \( P' \) any part which intersects \( B_i \), at the cost of at most \( \alpha \) parts of \( P' \).

7.2. Extracting permutation encodings. Let us now focus on the second generalisation: to a tournament \( T = (V, A) \), we add arbitrary binary relations \( R_1, \ldots, R_\ell \subset V^2 \), yielding a relational structure \( (V, A, R_1, \ldots, R_\ell) \). We will show that the tournament \( A \) in such structures is enough to obtain our main results, without any hypothesis on \( R_1, \ldots, R_\ell \). Remark that when the tournament \( (V, A) \) is transitive, this exactly corresponds to the ordered structures considered in [9]. The relations \( R_i \) are required to be binary for twin-width to be well-defined. Naturally, \( (V, A) \) can also be an oriented graph with bounded independence number instead of a tournament.

Formally, we fix \( \Sigma = \{A, R_1, \ldots, R_\ell\} \) a binary relational signature with a distinguished relation \( A \), and consider \( \Sigma \)-structures in which \( A \) is interpreted as an oriented graph with independence number bounded by some constant \( \alpha \).

A chain order representation of the matrix \( M \) in a \( \Sigma \)-structure \( S \) consists of the following:

- two disjoint ordered subsets \((X, \preceq_X)\) and \((Y, \preceq_Y)\), where \( \preceq_X, \preceq_Y \) are two chain quasi-orderings in \((V, A)\), and
- a relation \( R \) among \( A, R_1, \ldots, R_\ell \) such that the adjacency matrix of \( R \) restricted to \((X, \preceq_X)\) versus \((Y, \preceq_Y)\) is \( M \).

If \( \sigma \) is a permutation, we say that a chain order representation of \( \sigma \) in \( S \) is a chain order representation of the matrix \( M_s(\sigma) \) in \( S \) for any encoding \( s \in \{=, \neq, \leq_R, \geq_R, \leq_C, \geq_C\} \), see section 6.1.

Using Lemma 7.1, it is straightforward to generalise Theorem 1.5.

Theorem 7.2. Let \( S \) be a hereditary class of \( \Sigma \)-structures in which \( A \) is interpreted as oriented graphs with bounded independence number. If there are structures \( S \in S \) with BST orders \( < \) for which \( \text{tww}(S, <) \) is arbitrarily large, then \( S \) contains chain order representations of all permutations.

Sketch of proof. The proof is the same as the first half of Theorem 1.5, up to Claim 6.3. Given \( S = (V, A, R_1, \ldots, R_\ell) \) and the BST ordering \( < \) such that \( \text{tww}(S, <) \) is large, Theorem 2.3 (or rather its natural generalisation to ordered binary structures, see Theorem 2 and Lemma 23 in [9]) yields a high rank division in the adjacency matrix of one of the relations \( A, R_1, \ldots, R_\ell \), with rows and columns ordered by \( < \). Lemma 7.1 and the arguments of section 5 then allow replacing \( < \) with chain orderings, yielding a chain order representation of a matrix with a high rank division. Finally, Theorem 6.2 extracts a permutation encoding \( M_s(\sigma) \) for any permutation \( \sigma \) of size \( k \) with \( k \) an unbounded function of the initial twin-width \( \text{tww}(S, <) \). If the latter can be arbitrarily large in \( S \), we thus retrieve chain order representations of all permutations. \( \square \)

A variant of Lemma 5.1 shows that structures containing chain order representations of all permutations have unbounded twin-width. Following the arguments of section 5, this yields an FPT approximation algorithm for twin-width of such relational structures.
Theorem 7.3. There are functions $f, g$ and an algorithm which given a structure $S = (V, A, R_1, \ldots, R_t)$ with twin-width $t$, where $(V, A)$ is an oriented graph with independence number $\alpha$, computes a contraction sequence of width $f(t, \alpha, t)$ in time $g(t, \alpha, t) \cdot |V|^{O(1)}$.

To obtain the full list of equivalent conditions, we additionally need to extract canonical obstructions from chain order representations by generalising the proofs of section 6.

Consider a chain order representation of a permutation $\sigma$ on $n$ elements. It consists of four disjoint subsets of vertices: $X, Y$ between which the relation $R$ induces the matrix $M_\sigma(\sigma)$, and the chains $C_X, C_Y$ which define the appropriate chain orderings $\preceq^X, \preceq^Y$.

Enumerate $X = \{x_i\}_{i \in [n]}$ and similarly $Y = \{y_i\}_{i \in [n]}$ according to $\preceq^X$ and $\preceq^Y$ respectively. In $C_X$, we only need to keep to keep $n - 1$ vertices: one vertex $c_i$ to order $x_i \preceq^X x_{i+1}$ for each $i < n$. Assuming that $\preceq^X$ is the ‘positive’ chain ordering $\preceq^+_X$, we then have the edges $c_i \rightarrow c_j$ and $c_i \rightarrow x_j$ for $i < j$, but $c_i \not\rightarrow x_i$ (meaning either $x_i \rightarrow c_i$ or there is no edge between them). We similarly enumerate $C_Y = \{d_i\}_{i \in [n-1]}$. To summarize, up to reversing the direction of edges for chain orders, the definition of chain order representation enforces the following edges:

- in $A$, the edges $c_i \rightarrow c_j$, $c_i \rightarrow x_j$, $d_i \rightarrow d_j$, $d_i \rightarrow y_j$ for any $i < j$, but $c_i \not\rightarrow x_i$ and $d_i \not\rightarrow y_i$, and
- in $R$, the edges between $x_i$ and $y_j$ according to the matrix $M_\sigma(\sigma)$.

All remaining edges (including all relations in $S$ other than $A, R$) can be arbitrary. To obtain canonical obstruction, we need to eliminate these arbitrary choices.

This is possible thanks to Lemma 2.1. On each of $X, Y, C_X, C_Y$, define two orderings $<_1, <_2$:

- on $X$ and $C_X$, $<_1$ is the natural ordering $x_1 <_1 \cdots <_1 x_n$ (coinciding with $\preceq^X$) while on $Y$ it is permuted by $\sigma$ as $y_{\sigma(1)} <_1 \cdots < y_{\sigma(n)}$,
- and similarly $<_2$ is the natural ordering on $Y$ and $C_Y$, while on $X$ and $C_X$ it is permuted by $\sigma^{-1}$.

Thus on each of the sets $X, Y, C_X, C_Y$, the pair $(<_1, <_2)$ is a biorder representing $\sigma$.

Now Lemma 2.1 shows that for any goal permutation $\tau$, there is a starting permutation $\sigma$ satisfying the following: from a chain order representation of $\sigma$, one may extract a subrepresentation of $\tau$ for which the presence of an edge between vertices $u, v$ in a given relation $R_i$ only depends on (1) which sets $(X, Y, C_X, C_Y)$ they belong to, and (2) the relative orderings of $u$ and $v$ for $<_1$ and $<_2$. The key remark is that the edges enforced by the definition of chain order representation already only depend on the orderings $<_1$ and $<_2$, so that applying Lemma 2.1 preserves the structure of the chain order representation while removing unwanted choices from the remaining edges.

We thus obtain a canonical chain order representation of any $\tau$, where all edges are specified by a collection of rules such as e.g.: if $u \in X, v \in Y$, $u <_1 v$ and $v <_2 u$, then in $A$ there is the edge $u \rightarrow v$. Each choice of set of rules, together with the choice of encoding $s \in \{=, \neq, \leq_R, \geq_R, \leq_C, \geq_C\}$ for
the matrix $M_s(\sigma)$ specifies a family $\mathcal{F}_i$ of representations of permutations. The different choices of rules yield as many classes $\mathcal{F}_1, \ldots, \mathcal{F}_k$ of obstructions. Naturally, the collection of classes $\mathcal{F}_1, \ldots, \mathcal{F}_k$ and their number depend on the signature $\Sigma$. From this sketch of proof, we obtain the following:

**Lemma 7.4.** Let $\mathcal{S}$ be a hereditary class of $\Sigma$-structures in which $A$ is interpreted as an oriented graph with independence number bounded by $\alpha$. Then $\mathcal{S}$ has bounded twin-width if and only if it does not contain any $\mathcal{F}_i$ as a subset.

We finally need to show that these classes are obstructions to the properties we are interested in.

**Lemma 7.5.** Each of the classes of obstructions $\mathcal{F}_1, \ldots, \mathcal{F}_k$

1. has unbounded twin-width;
2. contains at least $\frac{n!}{2^n}$ structures on $n$ vertices counted up to isomorphism for some constant $c$;
3. efficiently interprets the class of all graphs;
4. and has $\text{AW}[\ast]$-hard FO model checking problem.

**Sketch of proof.** In a fixed $\mathcal{F}_i$, there is a single canonical chain order representation $R(\sigma)$ of each permutation $\sigma$, using $4n - 2$ vertices when $\sigma$ is a permutation on $n$ elements. After guessing the sets $(X,Y,C_X,C_Y)$ which describe the chain order representation (at most $4^n$ choices), one can deterministically reconstruct $\sigma$. This implies that there are at least $\frac{n!}{2^n}$ non-isomorphic structures on $4n - 2$ vertices in $\mathcal{F}_i$, proving (2). It follows by Theorem 2.10 that $\mathcal{F}_i$ has unbounded twin-width (1).

Furthermore, this reconstruction of $\sigma$ can be expressed by a first-order interpretation. Since the canonical chain order representation $R(\sigma)$ is easily computed in polynomial time from $\sigma$. Note here that it is crucial that all edges in $R(\sigma)$ are specified by the set of rules corresponding to $\mathcal{F}_i$, as otherwise $\sigma \mapsto R(\sigma)$ would be a well-defined function. Thus $\mathcal{F}_i$ efficiently interprets the class of all permutations, which by Lemma 2.7 implies (3). Model checking hardness (4) follows by Lemma 2.6.

The main result follows from Lemmas 7.4 and 7.5.

**Theorem 7.6.** Let $\mathcal{S}$ be a class of $\Sigma$-structures in which $A$ is interpreted as oriented graphs with independence number at most some constant $k$. Under the assumption $\text{FPT} \neq \text{AW}[\ast]$, the following are equivalent:

1. $\mathcal{S}$ has bounded twin-width,
2. FO model checking in $\mathcal{S}$ is FPT,
3. FO model checking in $\mathcal{S}$ is not $\text{AW}[\ast]$-hard,
4. $\mathcal{S}$ does not interpret the class of all graphs,
5. $\mathcal{S}$ is monadically NIP,
6. the growth of $\mathcal{S}$ contains at most $c^n$ structures on $n$ vertices for some constant $c$, counted up to isomorphism.

**Acknowledgments**

The authors would like to thank Édouard Bonnet for stimulating discussions on the topics of this work, and Szymon Toruńczyk for helpful explanation on model theoretic aspects of section 5.
References

[1] Nir Ailon, Moses Charikar and Alantha Newman. “Aggregating Inconsistent Information: Ranking and Clustering”. In: J. ACM 55.5 (Nov. 2008). issn: 0004-5411. doi: 10.1145/1411509.1411513

[2] Noga Alon et al. “Dominating sets in k-majority tournaments”. In: Journal of Combinatorial Theory, Series B 96.3 (2006), pp. 374–387. issn: 0095-8956. doi: 10.1016/j.jctb.2005.09.003

[3] Jakub Balabán and Petr Hlinény. “Twin-Width Is Linear in the Poset Width”. In: 16th International Symposium on Parameterized and Exact Computation, IPEC 2021, September 8-10, 2021, Lisbon, Portugal. Ed. by Petr A. Golovach and Meirav Zehavi. Vol. 214. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, 6:1–6:13. doi: 10.4230/LIPIcs.IPEC.2021.6

[4] Manuel Bodirsky. “Ramsey classes: examples and constructions.” In: Surveys in combinatorics 424 (2015), p. 1.

[5] Mikolaj Bojańczyk. personal communication. July 2022.

[6] Édouard Bonnet et al. “Twin-width and permutations”. In: Logical Methods in Computer Science Volume 20, Issue 3 (July 2024). doi: 10.46298/lmcs-20(3:4)2024.

[7] Édouard Bonnet et al. “Twin-width I: Tractable FO Model Checking”. In: J. ACM 69.1 (2022), 3:1–3:46. doi: 10.1145/3486655

[8] Édouard Bonnet et al. “Twin-width II: small classes”. In: Combinatorial Theory 2.2 (2022). issn: 2766-1334. doi: 10.5070/C62257876

[9] Édouard Bonnet et al. “Twin-Width IV: Ordered Graphs and Matrices”. In: Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing (Rome, Italy). STOC 2022. New York, NY, USA: Association for Computing Machinery, 2022, pp. 924–937. isbn: 9781450392648. doi: 10.1145/3519935.3520037. arXiv: 2102.03117.

[10] Édouard Bonnet et al. Twin-width VIII: groups. 2022. doi: 10.48550/ARXIV.2204.12330 arXiv: 2204.12330

[11] Édouard Bonnet et al. “Twin-Width VIII: Delineation and Win-Wins”. In: 17th International Symposium on Parameterized and Exact Computation (IPEC 2022). Ed. by Holger Dell and Jesper Nederlof. Vol. 249. LIPIcs. Dagstuhl, Germany: Schloss Dagstuhl, 2022, 9:1–9:18. isbn: 978-3-95977-260-0. doi: 10.4230/LIPIcs.IPEC.2022.9

[12] Youssef Boudabbous and Maurice Pouzet. “The morphology of infinite tournaments; application to the growth of their profile”. In: European Journal of Combinatorics 31.2 (2010). Combinatorics and Geometry, pp. 461–481. issn: 0195-6698. doi: 10.1016/j.ejc.2009.03.027

[13] Samuel Braunfeld and Michael C Laskowski. Existential characterizations of monadic NIP. 2022. doi: 10.48550/ARXIV.2209.05120

[14] Maria Chudnovsky, Alexandra Fradkin and Paul Seymour. “Tournament immersion and cutwidth”. In: J. Comb. Theory, Ser. B 102 (Jan. 2012), pp. 93–101. doi: 10.1016/j.jctb.2011.05.001

[15] Rodney G Downey, Michael R Fellows and Udayan Taylor. “The parameterized complexity of relational database queries and an improved characterization of W[1]”. In: DMCTS 96 (1996), pp. 194–213.
[16] Jan Dreier, Nikolas Mählmann and Szymon Toruńczyk. “Flip-Breakability: A Combinatorial Dichotomy for Monadically Dependent Graph Classes”. In: *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*. STOC 2024. Vancouver, BC, Canada: Association for Computing Machinery, 2024, pp. 1550–1560. ISBN: 9798400703836. DOI: 10.1145/3618260.3649739

[17] Fedor V. Fomin and Michał Pilipczuk. “On width measures and topological problems on semi-complete digraphs”. In: *Journal of Combinatorial Theory, Series B* 138 (2019), pp. 78–165. ISSN: 0095-8956. DOI: 10.1016/j.jctb.2019.01.006

[18] Alexandra Fradkin and Paul Seymour. “Tournament pathwidth and topological containment”. In: *Journal of Combinatorial Theory, Series B* 103.3 (2013), pp. 374–384. ISSN: 0095-8956. DOI: 10.1016/j.jctb.2013.03.001

[19] Jakub Gajarský et al. “A New Perspective on FO Model Checking of Dense Graph Classes”. In: *ACM Trans. Comput. Logic* 21.4 (July 2020). ISSN: 1529-3785. DOI: 10.1145/3383206

[20] Jakub Gajarský et al. “FO model checking on posets of bounded width”. In: *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*. IEEE. 2015, pp. 963–974. DOI: 10.1109/FOCS.2015.63

[21] Ronald L Graham, Bruce L Rothschild and Joel H Spencer. *Ramsey theory*. Vol. 20. John Wiley & Sons, 1991.

[22] Martin Grohe and Daniel Neuen. “Isomorphism for Tournaments of Small Twin Width”. In: *51st International Colloquium on Automata, Languages, and Programming (ICALP 2024)*. Ed. by Karl Bringmann et al. Vol. 297. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024, 78:1–78:20. ISBN: 978-3-95977-322-5. DOI: 10.4230/LIPIcs.ICALP.2024.78

[23] 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, SODA 2014*, Portland, Oregon, USA, January 5-7, 2014. 2014, pp. 82–101. DOI: 10.1137/1.9781611973402.7

[24] Mithilesh Kumar and Daniel Lokshtanov. “Faster Exact and Parameterized Algorithm for Feedback Vertex Set in Tournaments”. In: *33rd Symposium on Theoretical Aspects of Computer Science (STACS 2016)*. Ed. by Nicolas Ollinger and Heribert Vollmer. Vol. 47. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2016, 49:1–49:13. ISBN: 978-3-95977-001-9. DOI: 10.4230/LIPIcs.STACS.2016.49

[25] Adam Marcus and Gábor Tardos. “Excluded permutation matrices and the Stanley-Wilf conjecture”. In: *J. Comb. Theory, Ser. A* 107.1 (2004), pp. 153–160. DOI: 10.1016/j.jcta.2004.04.002

[26] Serguei Norine et al. “Proper minor-closed families are small”. In: *Journal of Combinatorial Theory, Series B* 96.5 (2006), pp. 754–757. ISSN: 0095-8956. DOI: 10.1016/j.jctb.2006.01.006
[27] Michał Pilipczuk. “Tournaments and Optimality: New Results in Parameterized Complexity”. PhD thesis. University of Bergen, Aug. 2013. URL: https://bora.uib.no/bora-xmlui/bitstream/handle/1956/7650/dr-thesis-2013-Michal-Pilipczuk.pdf

**Institute for Basic Science (IBS), 55 EXPO-ro, Yuseong-gu, Daejeon, South Korea 34126.**
*Email address: colin@ibs.re.kr*

**Laboratoire de l’Informatique du Parallélisme, ENS de Lyon, 46 allée d’Italie, 69364 Lyon CEDEX 07, France**
*Email address: stephan.thomasse@ens-lyon.fr*