# Isomorphism for Tournaments of Small Twin Width 

Martin Grohe $\square$ (i)<br>RWTH Aachen University, Germany<br>Daniel Neuen $\square$ (다<br>University of Regensburg, Germany


#### Abstract

We prove that isomorphism of tournaments of twin width at most $k$ can be decided in time $k^{O(\log k)} n^{O(1)}$. This implies that the isomorphism problem for classes of tournaments of bounded or moderately growing twin width is in polynomial time. By comparison, there are classes of undirected graphs of bounded twin width that are isomorphism complete, that is, the isomorphism problem for the classes is as hard as the general graph isomorphism problem. Twin width is a graph parameter that has been introduced only recently (Bonnet et al., FOCS 2020), but has received a lot of attention in structural graph theory since then. On directed graphs, it is functionally smaller than clique width. We prove that on tournaments (but not on general directed graphs) it is also functionally smaller than directed tree width (and thus, the same also holds for cut width and directed path width). Hence, our result implies that tournament isomorphism testing is also fixed-parameter tractable when parameterized by any of these parameters.

Our isomorphism algorithm heavily employs group-theoretic techniques. This seems to be necessary: as a second main result, we show that the combinatorial Weisfeiler-Leman algorithm does not decide isomorphism of tournaments of twin width at most 35 if its dimension is $o(n)$. (Throughout this abstract, $n$ is the order of the input graphs.)


2012 ACM Subject Classification Theory of computation $\rightarrow$ Fixed parameter tractability; Mathematics of computing $\rightarrow$ Graph algorithms; Theory of computation $\rightarrow$ Finite Model Theory

Keywords and phrases tournament isomorphism, twin width, fixed-parameter tractability, WeisfeilerLeman algorithm

Digital Object Identifier 10.4230/LIPIcs.ICALP.2024.78
Category Track A: Algorithms, Complexity and Games
Related Version Full Version: https://arxiv.org/abs/2312.02048
Funding Martin Grohe: The research is funded by the European Union (ERC, SymSim, 101054974). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

Acknowledgements We thank the anonymous reviewers of an earlier version of this paper for helpful feedback which, in particular, resulted an improved bound in Theorem 1.2.

## 1 Introduction

The tournament isomorphism problem (TI) was recognized as a particularly interesting special case of the graph isomorphism problem (GI) early-on. Already in 1983, Babai and Luks [3] proved that TI is solvable in time $n^{O(\log n)}$; it took 33 more years for Babai [2] to prove that the general GI is in quasi-polynomial time. An important fact that makes TI more accessible than GI is that tournaments always have solvable automorphism groups. This is a consequence of the observation that the automorphism groups of tournaments have odd order and the famous Feit-Thompson Theorem [18] stating that all groups of odd order are solvable. However, even Babai's powerful new machinery did not help us to

© Martin Grohe and Daniel Neuen
licensed under Creative Commons License CC-BY 4.0
51st International Colloquium on Automata, Languages, and Programming (ICALP 2024). Editors: Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson; Article No. 78; pp. 78:1-78:20

Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany
improve the upper bound for TI, as one might have hoped. But TI is not only special from a group-theoretic perspective. Another remarkable result, due to Schweitzer [42], states that TI reduces to the problem of deciding whether a tournament has a nontrivial automorphism; the so-called rigidity problem. It is an open question whether the same holds for general graphs.

While there is an extensive literature on GI restricted to classes of graphs (see [25] for a recent survey), remarkably little is known for restrictions of TI. Ponomarenko [40] proved that TI is in polynomial time for tournaments whose automorphism group contains a regular cyclic subgroup, and recently Arvind, Ponomarenko, and Ryabov [1] proved that TI is in polynomial time for edge-colored tournaments where at least one edge color induces a (strongly) connected spanning subgraph of bounded degree (even fixed-parameter tractable when parameterized by the out-degree). While both of these results are very interesting from a technical perspective, they consider classes of tournaments that would hardly be called natural from a graph-theoretic point of view. Natural graph parameters that have played a central role in the structural theory of tournaments developed by Chudnovsky, Seymour and others $[13,14,15,16,19]$ are cut width and path width. The more recent theory of structural sparsity $[20,21,36,37]$ highlights clique width and twin width. Here twin width is the key parameter. Not only is it functionally smaller than the other parameters, which means that if cut width, path width, or clique width is bounded, then twin width is bounded as well, it is also known [23] that a class of tournaments has bounded twin width if and only if it has a property known as monadic dependence (NIP). Dependence is a key property studied in classical model theory. A class of graphs is monadically dependent if and only if all set systems definable in this class by a first-order transduction have bounded VC dimension. This property seems to characterize precisely the graph classes that are regarded as structurally sparse. Since twin width of graphs and binary relational structures has been introduced in [11], it received a lot of attention in algorithmic structural graph theory $[5,6,7,8,9,10,21,22,23,29,44]$. (We defer the somewhat unwieldy definition of twin width to Section 2.3.) Our main result states that tournament isomorphism is fixed-parameter tractable when parameterized by twin width.

- Theorem 1.1. The isomorphism problem for tournaments of twin width at most $k$ can be solved in time $k^{O(\log k)} \cdot n^{O(1)}$.

Interestingly, isomorphism testing for undirected graphs of bounded twin width is as hard as the general GI. This follows easily from the fact that an $\Omega(\log n)$-subdivision of every graph with $n$ vertices has bounded twin width [5]. Once more, this demonstrates the special role of the tournament isomorphism problem, though here the reason is not group-theoretic, but purely combinatorial.

Note that the dependence on the twin width $k$ of the algorithm in Theorem 1.1 is subexponential, so our result implies that TI is in polynomial time even for tournaments of twin width $2^{O(\sqrt{\log n})}$. Since twin width is functionally smaller than clique width, our result implies that TI is also fixed-parameter tractable when parameterized by clique width. Additionally, we prove that the twin width of a tournament is functionally smaller than its directed tree width, a graph parameter originally introduced in [31]. Since the directed tree width of every directed graph is smaller than its cut width or directed path width, the same also holds for these two parameters. Hence, TI is fixed-parameter tractable also when parameterized by directed tree width, directed path width or cut width. To the best of our knowledge, this was not known for any of these parameters. The fact that twin width is functionally smaller than directed tree width, directed path width and cut width on tournaments is interesting in its own right, because this result does not extend to general directed graphs (for any of the three parameters).

Our proof of Theorem 1.1 heavily relies on group-theoretic techniques. In a nutshell, we show that bounded twin width allows us to cover a tournament by a sequence of directed graphs that have a property resembling bounded degree sufficiently closely to apply a grouptheoretic machinery going back to Luks [33] and developed to great depth since then (see, e.g., $[2,3,27,35,38])$. Specifically, we generalize arguments that have been introduced by Arvind et al. [1] for TI on edge-colored tournaments where at least one edge color induces a spanning subgraph of bounded out-degree.

Yet one may wonder if this heavy machinery is even needed to prove our theorem, in particular in view of the fact that on many natural graph classes, including, for example, undirected graphs of bounded clique width [26], the purely combinatorial Weisfeiler-Leman algorithm is sufficient to decide isomorphism (see, e.g., $[24,32]$ ). We prove that this is not the case for tournaments of bounded twin width.

- Theorem 1.2. For every $k \geq 2$ there are non-isomorphic tournaments $T_{k}$ and $T_{k}^{\prime}$ of order $\left|V\left(T_{k}\right)\right|=\left|V\left(T_{k}^{\prime}\right)\right|=O(k)$ and twin width at most 35 that are not distinguished by the $k$-dimensional Weisfeiler-Leman algorithm.

We remark that it was known before that the Weisfeiler-Leman algorithm fails to decide tournament isomorphism. Indeed, Dawar and Kopczynski (unpublished) proved that for every $k \geq 2$ there are non-isomorphic tournaments $U_{k}$ and $U_{k}^{\prime}$ of order $\left|V\left(U_{k}\right)\right|=\left|V\left(U_{k}^{\prime}\right)\right|=O(k)$ that are not distinguished by the $k$-dimensional Weisfeiler-Leman algorithm. Theorem 1.2 strengthens this result by constructing tournaments where additionally the twin width is bounded by a fixed constant.

The paper is organized as follows. After introducing the necessary preliminaries in Section 2, Theorem 1.1 is proved in Sections 3 and 4. First, we give our main combinatorial arguments in Section 3. After that, the mainly group-theoretic isomorphism algorithm of Theorem 1.1 is presented in Section 4. Theorem 1.2 is proved in Section 5. Finally, in Section 6 we compare twin width to other width measures for directed graphs. All omitted proofs can be found in the full version.

## 2 Preliminaries

### 2.1 Graphs

Graphs in this paper are usually directed. We often emphasize this by calling them "digraphs". However, when we make general statements about graphs, this refers to directed graphs and includes undirected graphs a special case (directed graphs with a symmetric edge relation). We denote the vertex set of a graph $G$ by $V(G)$ and the edge relation by $E(G)$. The vertex set $V(G)$ is always finite and non-empty. The edge relation is always anti-reflexive, that is, graphs are loop-free, and there are no parallel edges. For a digraph $G$ and a vertex $v \in V(G)$, we denote the set of out-neighbors and in-neighbors of $v$ by $N_{+}(v)$ and $N_{-}(v)$, respectively. Also,
 respectively. Furthermore, $E_{+}(v)$ and $E_{-}(v)$ denote the set of outgoing and incoming edges into $v$, respectively. For $X \subseteq V(G)$, we write $G[X]$ to denote the subgraph of $G$ induced on $X$. For two sets $X, Y \subseteq V(G)$ we write $E_{G}(X, Y):=\{(v, w) \in E(G) \mid v \in X, w \in Y\}$ to denote the set of directed edges from $X$ to $Y$.

Let $G$ be an undirected graph. A directed graph $\vec{G}$ is an orientation of $G$ if, for every undirected edge $\{v, w\} \in E(G)$, exactly one of $(v, w)$ and $(w, v)$ is an edge of $\vec{G}$, and there are no other edges present in $\vec{G}$. A tournament is an orientation of a complete graph.

A tournament $T$ is regular if $\operatorname{deg}_{+}(v)=\operatorname{deg}_{+}(w)$ for all $v, w \in V(T)$. In this case, $\operatorname{deg}_{+}(v)=\operatorname{deg}_{-}(v)=\frac{|V(G)|-1}{2}$ for all $v \in V(T)$. This implies that every regular tournament has an odd number of vertices.

Let $G_{1}, G_{2}$ be two graphs. An isomorphism from $G_{1}$ to $G_{2}$ is a bijection $\varphi: V\left(G_{1}\right) \rightarrow$ $V\left(G_{2}\right)$ such that $(v, w) \in E\left(G_{1}\right)$ if and only if $(\varphi(v), \varphi(w)) \in E\left(G_{2}\right)$ for all $v, w \in V\left(G_{1}\right)$. We write $\left.\varphi: G_{1}\right) \cong G_{2}$ to denote that $\varphi$ is an isomorphism from $G_{1}$ to $G_{2}$. Also, Iso $\left(G_{1}, G_{2}\right)$ denotes the set of all isomorphisms from $G_{1}$ to $G_{2}$. The graphs $G_{1}$ and $G_{2}$ are isomorphic if $\operatorname{Iso}\left(G_{1}, G_{2}\right) \neq \emptyset$. The automorphism group of $G_{1}$ is $\operatorname{Aut}\left(G_{1}\right):=\operatorname{Iso}\left(G_{1}, G_{1}\right)$.

An arc coloring of a digraph $G$ is a mapping $\lambda:(E(G) \cup\{(v, v) \mid v \in V(G)\}) \rightarrow C$ for some set $C$ of "colors". An arc-colored graph is a triple $G=(V, E, \lambda)$, where $(V, E)$ is a graph an $\lambda$ an arc coloring of $(V, E)$. Isomorphisms between arc-colored graphs are required to preserve the coloring.

### 2.2 Partitions and Colorings

Let $S$ be a finite set. A partition of $S$ is a set $\mathcal{P} \subseteq 2^{S}$ whose elements we refer to as parts, such that any two parts are mutually disjoint, and the union of all parts is $S$. A partition $\mathcal{P}$ refines another partition $\mathcal{Q}$, denoted by $\mathcal{P} \preceq \mathcal{Q}$, if for all $P \in \mathcal{P}$ there is some $Q \in \mathcal{Q}$ such that $P \subseteq Q$. We say a partition $\mathcal{P}$ is trivial if $|\mathcal{P}|=1$, which means that the only part is $S$, and it is discrete if $|P|=1$ for all $P \in \mathcal{P}$.

Every mapping $\chi: S \rightarrow C$, for some set $C$, induces a partition $\mathcal{P}_{\chi}$ of $S$ into the sets $\chi^{-1}(c)$ for all $c$ in the range of $\chi$. In this context, we think of $\chi$ as a "coloring" of $S$, the elements $c \in C$ as "colors", and the parts $\chi^{-1}(c)$ of the partition as "color classes". If $\chi^{\prime}: S \rightarrow C^{\prime}$ is another coloring, then we say that $\chi$ refines $\chi^{\prime}$, denoted by $\chi \preceq \chi^{\prime}$, if $\mathcal{P}_{\chi} \preceq \mathcal{P}_{\chi^{\prime}}$. The colorings are equivalent (we write $\chi \equiv \chi^{\prime}$ ) if $\chi \preceq \chi^{\prime}$ and $\chi \preceq \chi^{\prime}$, i.e., $\mathcal{P}_{\chi}=\mathcal{P}_{\chi^{\prime}}$.

### 2.3 Twin Width

Twin width [11] is defined for binary relational structures, which in this paper are mostly directed graphs. We need one distinguished binary relation symbol $E_{\text {red }}$ that plays a special role in the definition of twin width. Following [11], we refer to elements of $E_{\text {red }}$ as red edges. For every structure $A$, we assume the relation $E_{\text {red }}(A)$ to be symmetric and anti-reflexive, that is, the edge relation of an undirected graph, and we refer to the maximum degree of this graph as the red degree of $A$. If $E_{\text {red }}(A)$ is not explicitly defined, we assume $E_{\text {red }}(A)=\emptyset$ (and the red degree of $A$ is 0 ).

Let $A=\left(V(A), R_{1}(A), \ldots, R_{k}(A)\right)$ be a binary relational structure, where $V(A)$ is a non-empty finite vertex set and $R_{i}(A) \subseteq(V(A))^{2}$ are binary relations on $V(A)$ (possibly, $R_{i}=E_{\text {red }}$ for some $i \in[k]$ ). We call a pair $(X, Y)$ of disjoint subsets of $V(A)$ homogeneous if for all $x, x^{\prime} \in X$, and all $y, y^{\prime} \in Y$ it holds that
(i) $(x, y) \in R_{i}(A) \Leftrightarrow\left(x^{\prime}, y^{\prime}\right) \in R_{i}(A)$ and $(y, x) \in R_{i}(A) \Leftrightarrow\left(y^{\prime}, x^{\prime}\right) \in R_{i}(A)$ for all $i \in[k]$, and
(ii) $(x, y) \notin E_{\text {red }}(A)$ and $(y, x) \notin E_{\text {red }}(A)$.

For a partition $\mathcal{P}$ of $V(A)$, we define $A / \mathcal{P}$ to be the structure with vertex set $V(A / \mathcal{P}):=\mathcal{P}$ and relations

$$
R_{i}(A / \mathcal{P}):=\left\{(X, Y) \in \mathcal{P}^{2} \mid(X, Y) \text { is homogeneous and } X \times Y \subseteq R_{i}(A)\right\}
$$

for all $R_{i} \neq E_{\text {red }}$, and

$$
E_{\text {red }}(A / \mathcal{P}):=\left\{(X, Y) \in \mathcal{P}^{2} \mid(X, Y) \text { is not homogeneous and } X \neq Y\right\}
$$

A contraction sequence for $A$ is a sequence of partitions $\mathcal{P}_{1}, \ldots, \mathcal{P}_{n}$ of $V(A)$ such that $\mathcal{P}_{1}=\{\{v\} \mid v \in V(A)\}$ is the discrete partition, $\mathcal{P}_{n}=\{V(A)\}$ is the trivial partition, and for every $i \in[n-1]$ the partition $\mathcal{P}_{i+1}$ is obtained from $\mathcal{P}_{i}$ by merging two parts, i.e., there are distinct $P, P^{\prime} \in \mathcal{P}_{i}$ such that $\mathcal{P}_{i+1}=\left\{P \cup P^{\prime}\right\} \cup\left(\mathcal{P}_{i} \backslash\left\{P, P^{\prime}\right\}\right)$. The width of a contraction sequence $\mathcal{P}_{1}, \ldots, \mathcal{P}_{n}$ of $A$ is the minimum $k$ such that for every $i \in[n]$ the structure $A / \mathcal{P}_{i}$ has red degree at most $k$. The twin width of $A$, denoted by $\operatorname{tww}(A)$, is the minimum $k \geq 0$ such that $A$ has a contraction sequence of width $k$.

Note that red edges are introduced as we contract parts of the partitions. However, the structure $A$ we start with may already have red edges, which then have direct impact on its twin width. In particular, the twin width of a graph $G$ may be smaller than the twin width of the structure $G^{\text {red }}$ obtained from $G$ by coloring all edges red. This fact is used later.

We also remark that for our isomorphism algorithms, we never have to compute a contraction sequence of minimum width or the twin width.

We state two simple lemmas on basic properties of twin width.

- Lemma 2.1 ([11]). Let $A$ be a binary relational structure and $X \subseteq V(A)$. Then $\operatorname{tww}(A[X]) \leq \operatorname{tww}(A)$.
- Lemma 2.2. Let $A$ be a structure over the vocabulary $\tau$. Then there is a linear order $<$ on $V(A)$ such that $\operatorname{tww}(A,<)=\operatorname{tww}(A)$.


### 2.4 Weisfeiler-Leman

In this section, we describe the $k$-dimensional Weisfeiler-Leman algorithm ( $k$-WL). The algorithm has been originally introduced in its 2-dimensional form by Weisfeiler and Leman [46] (see also [45]). The $k$-dimensional version, coloring $k$-tuples, was introduced later by Babai and Mathon (see [12]).

Fix $k \geq 2$, and let $G$ be a graph. For $i \geq 0$, we describe the coloring $\chi_{(i)}^{k, G}$ of $(V(G))^{k}$ computed in the $i$-th iteration of $k$-WL. For $i=0$, each tuple is colored with the isomorphism type of the underlying ordered induced subgraph. So if $H$ is another graph and $\bar{v}=$ $\left(v_{1}, \ldots, v_{k}\right) \in(V(G))^{k}, \bar{w}=\left(w_{1}, \ldots, w_{k}\right) \in(V(H))^{k}$, then $\chi_{(0)}^{k, G}(\bar{v})=\chi_{(0)}^{k, H}(\bar{w})$ if and only if, for all $i, j \in[k]$, it holds that $v_{i}=v_{j} \Leftrightarrow w_{i}=w_{j}$ and $\left(v_{i}, v_{j}\right) \in E(G) \Leftrightarrow\left(w_{i}, w_{j}\right) \in E(H)$. If $G$ and $H$ are arc-colored, then the colors are also taken into account.

Now let $i \geq 0$. For $\bar{v}=\left(v_{1}, \ldots, v_{k}\right)$ we define

$$
\chi_{(i+1)}^{k, G}(\bar{v}):=\left(\chi_{(i)}^{k, G}(\bar{v}), \mathcal{M}_{i}(\bar{v})\right)
$$

where

$$
\mathcal{M}_{i}(\bar{v}):=\left\{\left\{\left(\chi_{(i)}^{k, G}(\bar{v}[w / 1]), \ldots, \chi_{(i)}^{k, G}(\bar{v}[w / k])\right) \mid w \in V(G)\right\}\right\}
$$

and $\bar{v}[w / i]:=\left(v_{1}, \ldots, v_{i-1}, w, v_{i+1}, \ldots, v_{k}\right)$ is the tuple obtained from $\bar{v}$ by replacing the $i$-th entry by $w$ (and $\{\{\ldots\}\}$ denotes a multiset).

Clearly, $\chi_{(i+1)}^{k, G} \preceq \chi_{(i)}^{k, G}$ for all $i \geq 0$. So there is a unique minimal $i_{\infty} \geq 0$ such that $\chi_{\left(i_{\infty}+1\right)}^{k, G} \equiv \chi_{\left(i_{\infty}\right)}^{k, G}$ and we write $\chi^{k, G}:=\chi_{\left(i_{\infty}\right)}^{k, G}$ to denote the corresponding coloring.

The $k$-dimensional Weisfeiler-Leman algorithm takes as input a (possibly colored) graph $G$ and outputs (a coloring that is equivalent to) $\chi^{k, G}$. This can be done in time $O\left(k^{2} n^{k+1} \log n\right)$ [30].

Let $H$ be a second graph. The $k$-dimensional Weisfeiler-Leman algorithm distinguishes $G$ and $H$ if there is a color $c \in C$ such that

$$
\left|\left\{\bar{v} \in(V(G))^{k} \mid \chi^{k, G}(\bar{v})=c\right\}\right| \neq\left|\left\{\bar{w} \in(V(H))^{k} \mid \chi^{k, H}(\bar{w})=c\right\}\right| .
$$

We write $G \simeq_{k} H$ to denote that $k$-WL does not distinguish between $G$ and $H$.

A graph $G$ is $k$-WL-homogeneous if for all $v, w \in V(G)$ it holds that $\chi^{k, G}(v, \ldots, v)=$ $\chi^{k, G}(w, \ldots, w)$.

### 2.5 Group Theory

For a general background on group theory we refer to [41], whereas background on permutation groups can be found in [17]. Also, basics facts on algorithms for permutation groups are given in [43].

Basics for Permutation Groups. A permutation group acting on a set $\Omega$ is a subgroup $\Gamma \leq \operatorname{Sym}(\Omega)$ of the symmetric group. The size of the permutation domain $\Omega$ is called the degree of $\Gamma$. If $\Omega=[n]:=\{1, \ldots, n\}$, then we also write $S_{n}$ instead of $\operatorname{Sym}(\Omega)$. For $A \subseteq \Omega$ and $\gamma \in \Gamma$ let $\gamma(A):=\{\gamma(\alpha) \mid \alpha \in A\}$. The set $A$ is $\Gamma$-invariant if $\gamma(A)=A$ for all $\gamma \in \Gamma$.

Let $\theta: \Omega \rightarrow \Omega^{\prime}$ be a bijection. We write $\Gamma \theta:=\{\gamma \theta \mid \gamma \in \Gamma\}$ for the set of bijections from $\Omega$ to $\Omega^{\prime}$ obtained from concatenating a permutation from $\Gamma$ and $\theta$. Note that $(\gamma \theta)(\alpha)=\theta(\gamma(\alpha))$ for all $\alpha \in \Omega$.

A set $S \subseteq \Gamma$ is a generating set for $\Gamma$ if for every $\gamma \in \Gamma$ there are $\delta_{1}, \ldots, \delta_{k} \in S$ such that $\gamma=\delta_{1} \ldots \delta_{k}$. In order to perform computational tasks for permutation groups efficiently the groups are represented by generating sets of small size (i.e., polynomial in the size of the permutation domain). Indeed, most algorithms are based on so-called strong generating sets, which can be chosen of size quadratic in the size of the permutation domain of the group and can be computed in polynomial time given an arbitrary generating set (see, e.g., [43]).

Group-Theoretic Methods for Isomorphism Testing. In this work, we shall be interested in a particular subclass of permutation groups. Let $\Gamma$ be a group and let $\gamma, \delta \in \Gamma$. The commutator of $\gamma$ and $\delta$ is $[\gamma, \delta]:=\gamma^{-1} \delta^{-1} \gamma \delta$. The commutator subgroup $[\Gamma, \Gamma]$ of $\Gamma$ is the unique subgroup of $\Gamma$ generated by all commutators $[\gamma, \delta]$ for $\gamma, \delta \in \Gamma$. Note that $[\Gamma, \Gamma]$ is a normal subgroup of $\Gamma$. The derived series of $\Gamma$ is the sequence of subgroups $\Gamma^{(0)} \unrhd \Gamma^{(1)} \unrhd \Gamma^{(2)} \unrhd \ldots$ where $\Gamma^{(0)}:=\Gamma$ and $\Gamma^{(i+1)}:=\left[\Gamma^{(i)}, \Gamma^{(i)}\right]$ for all $i \geq 0$. A group $\Gamma$ is solvable if there is some $i \geq 0$ such that $\Gamma^{(i)}$ is the trivial group (i.e., it only contains the identity element). The next theorem follows from the Feit-Thompson Theorem stating that every group of odd order is solvable.

- Theorem 2.3. Let $T$ be a tournament. Then $\operatorname{Aut}(T)$ is solvable.

Next, we state several basic group-theoretic algorithms for isomorphism testing.

- Theorem 2.4 ([3, Theorem 4.1]). There is an algorithm that, given two tournaments $T_{1}$ and $T_{2}$, computes $\operatorname{Iso}\left(T_{1}, T_{2}\right)$ in time $n^{O(\log n)}$.

Note that $\operatorname{Iso}\left(T_{1}, T_{2}\right)$ may be of size exponential in the number of vertices of $T_{1}$ and $T_{2}$. However, if $T_{1}$ and $T_{2}$ are isomorphic (i.e., $\operatorname{Iso}\left(T_{1}, T_{2}\right) \neq \emptyset$ ), we have $\operatorname{Iso}\left(T_{1}, T_{2}\right)=\operatorname{Aut}\left(T_{1}\right) \varphi$ where $\varphi \in \operatorname{Iso}\left(T_{1}, T_{2}\right)$ is an arbitrary isomorphism from $T_{1}$ to $T_{2}$. Hence, the set $\operatorname{Iso}\left(T_{1}, T_{2}\right)$ can be represented by a generating set for $\operatorname{Aut}\left(T_{1}\right)$ of size polynomial in $\left|V\left(T_{1}\right)\right|$ and a single element $\varphi \in \operatorname{Iso}\left(T_{1}, T_{2}\right)$. Let us stress at this point that all isomorphism sets computed by the various algorithms discussed in this work are represented in this way.

Let $G_{1}$ and $G_{2}$ be two (colored) directed graphs. Also let $\Gamma \leq \operatorname{Sym}\left(V\left(G_{1}\right)\right)$ be a permutation group and let $\theta: V\left(G_{1}\right) \rightarrow V\left(G_{2}\right)$ be a bijection. We define

$$
\operatorname{Iso}_{\Gamma \theta}\left(G_{1}, G_{2}\right):=\operatorname{Iso}\left(G_{1}, G_{2}\right) \cap \Gamma \theta=\left\{\varphi \in \Gamma \theta \mid \varphi: G_{1} \cong G_{2}\right\}
$$

and $\operatorname{Aut}_{\Gamma}\left(G_{1}\right):=\operatorname{Iso}_{\Gamma}\left(G_{1}, G_{1}\right)$. Note that $\operatorname{Aut}_{\Gamma}\left(G_{1}\right) \leq \Gamma$ and, if $\operatorname{Iso}_{\Gamma \theta}\left(G_{1}, G_{2}\right) \neq \emptyset$, then $\operatorname{Iso}_{\Gamma \theta}\left(G_{1}, G_{2}\right)=\operatorname{Aut}_{\Gamma}\left(G_{1}\right) \varphi$ where $\varphi \in \operatorname{Iso}\left(G_{1}, G_{2}\right)$ is an arbitrary isomorphism from $G_{1}$ to $G_{2}$.

- Theorem 2.5 ([3, Corollary 3.6]). Let $G_{1}=\left(V_{1}, E_{1}, \lambda_{1}\right)$ and $G_{1}=\left(V_{2}, E_{2}, \lambda_{2}\right)$ be two arc-colored directed graphs. Also let $\Gamma \leq \operatorname{Sym}\left(V_{1}\right)$ be a solvable group and $\theta: V_{1} \rightarrow V_{2} a$ bijection. Then $\operatorname{Iso}_{\Gamma \theta}\left(G_{1}, G_{2}\right)$ can be computed in polynomial time.


## 3 Small Degree Partition Sequences

In the following, we design an isomorphism test for tournaments of twin width $k$ which runs in time $k^{O(\log k)} n^{O(1)}$. On a high level, the algorithm essentially proceeds in three phases. First, we use well-established group-theoretic methods going back to [3, 33] to reduce to the case where both input tournaments are 2-WL-homogeneous (without increasing the twin width). In the second step, we identify a substructure of an input tournament $T$ (that is 2-WL-homogeneous) that has some kind of bounded-degree property. More concretely, we apply the 2-dimensional Weisfeiler-Leman algorithm and compute a sequence of colors $c_{1}, \ldots, c_{\ell}$ in the image of the 2 -WL coloring $\chi^{2, T}$ so that the subgraph induced by all edges with a color from $c_{1}, \ldots, c_{\ell}$ has a certain type of bounded-degree property. After that, we rely on the computed bounded-degree structure to determine isomorphisms based on the group-theoretic graph isomorphism machinery. Similar tools have also been used in [1] to solve isomorphism of $k$-spanning tournaments. However, as we shall see below, the bounded-degree property guaranteed by the second step is weaker than the notion of $k$-spanning tournaments, which requires us to further extend the methods from [1].

In this section, we implement the second phase and prove the key combinatorial lemma (Lemma 3.6) underlying our isomorphism algorithm. Our arguments rely on the notion of mixed neighbors for a pair of vertices. For a pair $v, w \in V(T)$ of vertices we let

$$
\begin{equation*}
M(v, w):=\left(N_{-}(v) \cap N_{+}(w)\right) \cup\left(N_{+}(v) \cap N_{-}(w)\right) . \tag{1}
\end{equation*}
$$

We call the elements of $M(v, w)$ the mixed neighbors of $(v, w)$, and we call $\operatorname{md}(v, w):=$ $|M(v, w)|$ the mixed degree of $(v, w)$. The following simple observation links the mixed degree to twin width.

- Observation 3.1. There is an edge $(v, w) \in E(T)$ such that $\operatorname{md}(v, w) \leq \operatorname{tww}(T)$.

Proof. Let $k:=\operatorname{tww}(T)$ and let $\mathcal{P}_{1}, \ldots, \mathcal{P}_{n}$ be a contraction sequence of $T$ of width $k$. Let $\{v, w\}$ be the unique 2-element part in $\mathcal{P}_{2}$. Then $\operatorname{md}(v, w)=\operatorname{md}(w, v) \leq k$, and either $(v, w) \in E(T)$ or $(w, v) \in E(T)$.

In the following, let $G_{T}$ be the directed graph with vertex set $V\left(G_{T}\right):=V(T)$ and edge set $E\left(G_{T}\right):=\{(v, w) \in E(T) \mid \operatorname{md}(v, w) \leq \operatorname{tww}(T)\}$. The next lemma implies that $G_{T}$ has maximum out-degree at most $2 \cdot \operatorname{tww}(T)+1$.

- Lemma 3.2. Suppose $k \geq 1$. Let $T$ be a tournament and let $v \in V(T)$. Also let

$$
W:=\left\{w \in N_{+}(v) \mid \operatorname{md}(v, w) \leq k\right\} .
$$

Then $|W| \leq 2 k+1$.

Proof. Let $\ell:=|W|$. The induced subtournament $T[W]$ has a vertex $w$ of in-degree at least $(\ell-1) / 2$. Since $\operatorname{md}(v, w) \leq k$ and $\left(v, w^{\prime}\right) \in E(T)$ for all $w^{\prime} \in W$, we have

$$
\left|\left\{w^{\prime} \in W \mid\left(w^{\prime}, w\right) \in E(T)\right\}\right| \leq k
$$

Thus $\frac{\ell-1}{2} \leq k$, which implies that $|W|=\ell \leq 2 k+1$.
So $G_{T}$ is a subgraph of $T$ of maximum out-degree $d:=2 \cdot \operatorname{tww}(T)+1$. We remark that similar arguments also show that $G_{T}$ has maximum in-degree at most $d$ (technically, this property is not required by our algorithm, but it is helpful for the following explanations). Now, first suppose that $G_{T}$ is strongly connected. Then the edges of $G_{T}$ define a (strongly) connected spanning subgraph of maximum degree $2 d$ (in-degree plus out-degree). In this situation, we can directly use the algorithm from [1] to test isomorphism in time $d^{O(\log d)} n^{O(1)}$.

So suppose $G_{T}$ is not strongly connected. If $T$ is 2-WL-homogeneous, then $G_{T}$ is also not weakly connected (i.e., the undirected version of $G_{T}$ is not connected). In this case, the basic idea is to identify further edges to be added to decrease the number of connected components while keeping some kind of bounded-degree property.

In the following, let $\mathcal{Q}$ be a partition of $V(T)$ that is non-trivial, that is, has at least two parts. The reader is encouraged to think of $\mathcal{Q}$ as the partition into the (weakly) connected components of $G_{T}$, but the following results hold for any non-trivial partition $\mathcal{Q}$. We call an edge $\left(v, v^{\prime}\right) \in E(T)$ cross-cluster with respect to $\mathcal{Q}$ if it connects distinct $Q, Q^{\prime} \in \mathcal{Q}$. For a cross-cluster edge ( $v, v^{\prime}$ ) with $Q \ni v, Q^{\prime} \ni v^{\prime}$, we let

$$
\mathcal{M}_{\mathcal{Q}}\left(v, v^{\prime}\right):=\left\{Q^{\prime \prime} \in \mathcal{Q} \backslash\left\{Q, Q^{\prime}\right\} \mid Q^{\prime \prime} \cap M\left(v, v^{\prime}\right) \neq \emptyset\right\}
$$

and $\operatorname{md}_{\mathcal{Q}}\left(v, v^{\prime}\right):=\left|\mathcal{M}_{\mathcal{Q}}\left(v, v^{\prime}\right)\right|$.
The next two lemmas generalize Observation 3.1 and Lemma 3.2.

- Lemma 3.3. Let $T$ be a tournament and suppose $\mathcal{Q}$ is a non-trivial partition of $V(T)$. Then there is a cross-cluster edge $(v, w) \in E(T)$ such that $\operatorname{md}_{\mathcal{Q}}(v, w) \leq \operatorname{tww}(T)$.

Proof. Let $k:=\operatorname{tww}(T)$ and let $\mathcal{P}_{1}, \ldots, \mathcal{P}_{n}$ be a contraction sequence of $T$ of width $k$. Note that $\mathcal{P}_{1}$ refines $\mathcal{Q}$ and $\mathcal{P}_{n}$ does not refine $\mathcal{Q}$, because $\mathcal{Q}$ is nontrivial. Let $i \geq 1$ be minimal such that $\mathcal{P}_{i+1}$ does not refine $\mathcal{Q}$.

Let $P, P^{\prime} \in \mathcal{P}_{i}$ denote the parts merged in the step from $\mathcal{P}_{i}$ to $\mathcal{P}_{i+1}$. Since $\mathcal{P}_{i}$ refines $\mathcal{Q}$, there are $Q, Q^{\prime} \in \mathcal{Q}$ such that $P \subseteq Q$ and $P^{\prime} \subseteq Q^{\prime}$. Moreover, $Q \neq Q^{\prime}$, because $\mathcal{P}_{i+1}$ does not refine $\mathcal{Q}$. We pick arbitrary elements $v \in P$ and $w \in P^{\prime}$ such that $(v, w) \in E(T)$ (if $(w, v) \in E(T)$, we swap the roles of $P$ and $P^{\prime}$ ). Then $(v, w)$ is a cross-cluster edge with respect to $\mathcal{Q}$.

Let $P_{1}, \ldots, P_{k^{\prime}}$ be a list of all $P^{\prime \prime} \in \mathcal{P}_{i+1} \backslash\left\{P \cup P^{\prime}\right\}$ such that the pair $\left(P \cup P^{\prime}, P^{\prime \prime}\right)$ is not homogeneous. Then $k^{\prime} \leq k$ by the definition of twin width. Since $\mathcal{P}_{i} \backslash\left\{P, P^{\prime}\right\}=\mathcal{P}_{i+1} \backslash\left\{P \cup P^{\prime}\right\}$ and $\mathcal{P}_{i}$ refines $\mathcal{Q}$, there are $Q_{1}, \ldots, Q_{k^{\prime}} \in \mathcal{Q}$ such that $P_{i} \subseteq Q_{i}$ for all $i \in\left[k^{\prime}\right]$.

Now let $Q^{\prime \prime} \in \mathcal{Q} \backslash\left\{Q, Q^{\prime}, Q_{1}, \ldots, Q_{k^{\prime}}\right\}$. Suppose for contradiction that $Q^{\prime \prime} \cap M(v, w) \neq \emptyset$, and pick an element $w^{\prime} \in Q^{\prime \prime} \cap M(v, w)$. Then there is a $P^{\prime \prime} \in \mathcal{P}_{i} \backslash\left\{P, P^{\prime}, P_{1}, \ldots, P_{k^{\prime}}\right\}=\mathcal{P}_{i+1} \backslash$ $\left\{P \cup P^{\prime}, P_{1}, \ldots, P_{k^{\prime}}\right\}$ such that $w^{\prime} \in P^{\prime \prime}$. But then the pair $\left(P \cup P^{\prime}, P^{\prime \prime}\right)$ is not homogeneous, which is a contradiction. So $Q^{\prime \prime} \cap M(v, w)=\emptyset$. This implies that $\mathcal{M}_{\mathcal{Q}}(v, w) \subseteq\left\{Q_{1}, \ldots, Q_{k^{\prime}}\right\}$. In particular, $\operatorname{md}_{\mathcal{Q}}(v, w) \leq k^{\prime} \leq k$.

- Lemma 3.4. Suppose $k \geq 1$. Let $T$ be a tournament and let $\mathcal{Q}$ be a non-trivial partition of $V(T)$. Also let $Q \in \mathcal{Q}$ and $v \in Q$. Let

$$
\mathcal{W}:=\left\{Q^{\prime} \in \mathcal{Q} \backslash\{Q\} \mid \exists w \in Q^{\prime}:(v, w) \in E(T) \wedge \operatorname{md}_{\mathcal{Q}}(v, w) \leq k\right\}
$$

Then $|\mathcal{W}| \leq 2 k+1$.


Figure 1 The figure shows part of a tournament $T$. The colors $c_{1}$ and $c_{2}$ are shown in blue and green, respectively. Also, the parts of the partition $\mathcal{Q}_{1}$ are highlighted in gray. Note that only green edges, which are outgoing from the middle part, are shown.

The proof is very similar to the proof of Lemma 3.2.
Proof. Let $\ell:=|\mathcal{W}|$ and suppose $\mathcal{W}=\left\{Q_{1}, \ldots, Q_{\ell}\right\}$. For every $i \in[\ell]$ pick an element $w_{i} \in Q_{i}$ such that $\left(v, w_{i}\right) \in E(T)$ and $\operatorname{md}_{\mathcal{Q}}\left(v, w_{i}\right) \leq k$. We define $W:=\left\{w_{1}, \ldots, w_{\ell}\right\}$. Then there is some $w \in W$ such that

$$
\left|\left\{w^{\prime} \in W \mid\left(w^{\prime}, w\right) \in E(T)\right\}\right| \geq \frac{\ell-1}{2}
$$

because the induced subtournament $T[W]$ has a vertex of in-degree at least $(\ell-1) / 2$.
Since $\operatorname{md}_{\mathcal{Q}}(v, w) \leq k$ and $\left(v, w^{\prime}\right) \in E(T)$ for all $w^{\prime} \in W$, it follows that

$$
\left|\left\{w^{\prime} \in W \mid\left(w^{\prime}, w\right) \in E(T)\right\}\right| \leq k
$$

Thus $\frac{\ell-1}{2} \leq k$, which implies that $|\mathcal{W}|=\ell \leq 2 k+1$.
Now, suppose we color all edges $(v, w)$ of $T$ with $\operatorname{md}(v, w) \leq \operatorname{tww}(T)$ using the color $c_{1}=$ blue (see Figure 1). Let $\mathcal{Q}_{1}$ be the partition into the (weakly) connected components of the graph induced by the blue edges and suppose that $\mathcal{Q}_{1}$ is non-trivial. We can compute isomorphisms between the different parts of $\mathcal{Q}_{1}$ using the algorithm from [1]. Next, let us color all cross-cluster edges $(v, w) \in E(T)$ with $\operatorname{md}_{\mathcal{Q}}(v, w) \leq \operatorname{tww}(T)$ using the color $c_{2}=$ green. Then every vertex has outgoing green edges to at most 2 tww $(T)+1$ other parts of $\mathcal{Q}_{1}$ (see Lemma 3.4). However, since a vertex may have an unbounded number of green neighbors in a single part, the out-degree of the graph induced by the green edges may be unbounded. So it is not possible to use the algorithm from [1] as a black-box on the components induced by blue and green edges. Luckily, the methods used in [1] can be extended to work even in this more general setting (see Section 4). So if the graph induced by the blue and green edges is connected, then we are again done. Otherwise, we let $\mathcal{Q}_{2}$ denote the partition into (weakly) connected components of the graph induced by the blue and green edges. Now, we can continue in the same fashion identifying colors $c_{3}, c_{4}, \ldots$ and corresponding partitions $\mathcal{Q}_{3}, \mathcal{Q}_{4}, \ldots$ until the graph induced by all edges of colors $c_{1}, \ldots, c_{\ell}$ is eventually connected.

Below, we provide a lemma that computes the corresponding sequence of partitions and edge colors using $2-\mathrm{WL}$. To state the lemma in its cleanest form, we restrict our attention to tournaments that are 2-WL-homogeneous. Recall that a tournament $T$ is 2-WL-homogeneous if for all $v, w \in V(T)$ is holds that $\chi^{2, T}(v, v)=\chi^{2, T}(w, w)$.

We also require another piece of notation. For a directed graph $G$ and a set of colors $C \subseteq\left\{\chi^{2, G}(v, w) \mid(v, w) \in E(G)\right\}$ we write $G[C]$ for the directed graph with vertex set $V(G[C]):=V(G)$ and edge set

$$
E(G[C]):=\left\{(v, w) \in E(G) \mid \chi^{2, G} \in C\right\} .
$$

To prove Lemma 3.6 we need the following lemma about the connected components of the graphs $G[C]$.

- Lemma 3.5. Let $G$ be a 2 -WL-homogeneous graph, and let $C$ a set of colors in the range of $\chi^{2, G}$. Then the weakly connected components of $G[C]$ equal the strongly connected components of $G[C]$.
- Lemma 3.6. Let $T$ be a 2-WL-homogeneous tournament of twin width $\operatorname{tww}(T) \leq k$. Then there is a sequence of partitions $\{\{v\} \mid v \in V(T)\}=\mathcal{Q}_{0}, \ldots, \mathcal{Q}_{\ell}=\{V(T)\}$ of $V(T)$ where $\mathcal{Q}_{i-1}$ refines $\mathcal{Q}_{i}$ for all $i$, and a sequence of colors $c_{1}, \ldots, c_{\ell}$ in the range of $\chi^{2, T}$ such that
(1) $\mathcal{Q}_{i}$ is the partition into the strongly connected components of $T\left[\left\{c_{1}, \ldots, c_{i}\right\}\right]$ for every $i \in[\ell]$, and
(2) for every $i \in[\ell]$ and every $v \in V(T)$ it holds that

$$
\left|\left\{Q \in \mathcal{Q}_{i-1} \mid \exists w \in Q: \chi^{2, T}(v, w)=c_{i}\right\}\right| \leq 2 k+1
$$

Moreover, there is a polynomial-time algorithm that, given a tournament $T$ and an integer $k \geq 1$, computes the desired sequences $\mathcal{Q}_{0}, \ldots, \mathcal{Q}_{\ell}$ and $c_{1}, \ldots, c_{\ell}$ or concludes that $\operatorname{tww}(T)>$ $k$.

Proof. We set $\mathcal{Q}_{0}=\{\{v\} \mid v \in V(T)\}$ and inductively define a sequence of partitions and colors as follows. Let $i \geq 0$ and suppose we already defined partitions $\mathcal{Q}_{0} \prec \cdots \prec \mathcal{Q}_{i}$ and colors $c_{1}, \ldots, c_{i}$. If $\mathcal{Q}_{i}=\{V(T)\}$, we set $\ell:=i$ and complete both sequences. Otherwise, there is a cross-cluster edge $\left(v_{i+1}, w_{i+1}\right)$ with respect to $\mathcal{Q}_{i}$ such that $\operatorname{md}_{\mathcal{Q}_{i}}\left(v_{i+1}, w_{i+1}\right) \leq k$ by Lemma 3.3. We set $c_{i+1}:=\chi^{2, T}\left(v_{i+1}, w_{i+1}\right)$ and define $\mathcal{Q}_{i+1}$ to be the set of weakly connected components of $T\left[c_{1}, \ldots, c_{i+1}\right]$. By Lemma 3.5, these are also the strongly connected components.

First observe that $\mathcal{Q}_{i} \prec \mathcal{Q}_{i+1}$ since $\left(v_{i+1}, w_{i+1}\right)$ is a cross-cluster edge with respect to $\mathcal{Q}_{i}$, and $v_{i+1}, w_{i+1}$ are contained in the same part of $\mathcal{Q}_{i+1}$ by Lemma 3.5. Also, Property 1 is satisfied by definition. For Property 2 note that every edge $(v, w) \in E(T)$ such that $\chi^{2, G}(v, w)=c_{i+1}$ is a cross-cluster edge with respect to $\mathcal{Q}_{i}$. So Property 2 follows directly from Lemma 3.4.

Finally, it is clear from the description above that $\mathcal{Q}_{0} \prec \cdots \prec \mathcal{Q}_{\ell}$ and $c_{1}, \ldots, c_{\ell}$ can be computed in polynomial time, or we conclude that $\operatorname{tww}(T)>k$.

## 4 The Isomorphism Algorithm

Based on the structural insights summarized in Lemma 3.6, we now design an isomorphism test for tournaments of small twin width.

The strategy of our algorithm is the following. We are given two tournaments $T_{1}$ and $T_{2}$, and we want to compute $\operatorname{Iso}\left(T_{1}, T_{2}\right)$. First, we reduce to the case where both $T_{1}$ and $T_{2}$ are 2-WL-homogeneous.

Towards this end, we start by applying 2 -WL and, for $j=1,2$, compute the coloring $\chi^{2, T_{j}}$. If 2 -WL distinguishes the two tournaments, we can immediately conclude that they are non-isomorphic and return $\operatorname{Iso}\left(T_{1}, T_{2}\right)=\emptyset$.

So suppose that 2 -WL does not distinguish the tournaments. Then $T_{1}$ is 2 -WLhomogeneous if and only if $T_{2}$ is 2-WL-homogeneous.

If the $T_{j}$ are not 2-WL-homogeneous, we rely on the following standard argument. Let $c_{1}, \ldots, c_{p}$ be the vertex colors. For $i \in[p]$ and $j=1,2$, let $P_{j, i}$ be the set of all $v \in V\left(T_{j}\right)$ such that $\chi^{2, T_{j}}(v, v)=c_{i}$. We recursively compute the sets $\Lambda_{i}:=\operatorname{Iso}\left(T_{1}\left[P_{1, i}\right], T_{2}\left[P_{2, i}\right]\right)$ for all $i \in[p]$. Note that this is possible since $\operatorname{tww}\left(T_{1}\left[P_{1, i}\right]\right) \leq \operatorname{tww}\left(T_{1}\right)$ and $\operatorname{tww}\left(T_{2}\left[P_{2, i}\right]\right) \leq \operatorname{tww}\left(T_{2}\right)$ for all $i \in[p]$ by Lemma 2.1. If there is some $i \in[p]$ such that $\Lambda_{i}=\emptyset$, then $T_{1}$ and $T_{2}$ are non-isomorphic, and we return $\operatorname{Iso}\left(T_{1}, T_{2}\right)=\emptyset$. Otherwise, the set $\Lambda_{i}$ is a coset of $\Gamma_{i}:=\operatorname{Aut}\left(T_{1}\left[P_{1, i}\right]\right)$ for all $i \in[p]$, i.e., $\Lambda_{i}=\Gamma_{i} \theta_{i}$ for some bijection $\theta_{i}: P_{1, i} \rightarrow P_{2, i}$. As the automorphism group of a tournament, $\Gamma_{i}$ is solvable (see Theorem 2.3). Moreover, since the color classes $P_{1, i}$ are invariant under automorphisms of $T_{1}$, the automorphism group $\Gamma:=\operatorname{Aut}\left(T_{1}\right)$ is a subgroup of the direct product $\prod_{i} \Gamma_{i}$, which is also a solvable group. Also, $\operatorname{Iso}\left(T_{1}, T_{2}\right) \subseteq \Gamma \theta$ where $\theta: V\left(T_{1}\right) \rightarrow V\left(T_{2}\right)$ is the unique bijection defined via $\theta(v):=\theta_{i}(v)$ for all $v \in V\left(T_{1}\right)$, where $i \in[p]$ is the unique index such that $v \in P_{1, i}$. So $\operatorname{Iso}\left(T_{1}, T_{2}\right)=\operatorname{Iso}_{\Gamma \theta}\left(T_{1}, T_{2}\right)$ can be computed in polynomial time using Theorem 2.5.

So we may assume that $T_{1}$ and $T_{2}$ are 2 -WL-homogeneous. In this case, we apply Lemma 3.6 and obtain colors $c_{1}, \ldots, c_{\ell}$ and, for $j=1,2$, a partition sequence $\{\{v\} \mid v \in$ $\left.V\left(T_{j}\right)\right\}=\mathcal{Q}_{j, 0}, \ldots, \mathcal{Q}_{j, \ell}=\left\{V\left(T_{j}\right)\right\}$ where $\mathcal{Q}_{j, i-1}$ refines $\mathcal{Q}_{j, i}$ for all $i \in[\ell]$.

Now, we iteratively compute for $i=0, \ldots, \ell$ the sets $\operatorname{Iso}\left(T_{j}[Q], T_{j^{\prime}}\left[Q^{\prime}\right]\right)$ for all $j, j^{\prime} \in\{1,2\}$ and all $Q \in \mathcal{Q}_{j, i}$ and $Q^{\prime} \in \mathcal{Q}_{j^{\prime}, i}$. For $i=0$ this is trivial since all parts have size 1 . So suppose $i>0$ and consider some elements $j, j^{\prime} \in\{1,2\}$ and $Q \in \mathcal{Q}_{j, i}, Q^{\prime} \in \mathcal{Q}_{j^{\prime}, i}$. For simplicity, let us assume that $j=1$ and $j^{\prime}=2$. Our goal is to compute $\operatorname{Iso}\left(T_{1}[Q], T_{2}\left[Q^{\prime}\right]\right)$. To do so, we exploit that we already computed all isomorphisms between all pairs of subgraphs of $T_{1}[Q]$ and $T_{2}\left[Q^{\prime}\right]$ induced by sets $R \in \mathcal{Q}_{1, i-1} \cup \mathcal{Q}_{2, i-1}$ and for which $R \subseteq Q$ or $R \subseteq Q^{\prime}$. The next lemma describes the key subroutine of the main algorithm which achieves this goal. Note that, on the last level $\ell$, we compute the set $\operatorname{Iso}\left(T_{1}, T_{2}\right)$ since $\mathcal{Q}_{j, \ell}=\left\{V\left(T_{j}\right)\right\}$ for both $j \in\{1,2\}$.

To state the lemma, we need additional terminology. Let $T=(V, E, \lambda)$ be an arc-colored tournament. A partition $\mathcal{Q}$ of $V$ is $\lambda$-definable if there is a set of colors $C \subseteq\{\lambda(v, w) \mid v=$ $w \vee(v, w) \in E\}$ such that

$$
v \sim_{\mathcal{Q}} w \quad \Longleftrightarrow \quad \lambda(v, w) \in C
$$

for all $v, w \in V$ such that $v=w$ or $(v, w) \in E$. We also say that $\mathcal{Q}$ is $\lambda$-defined by $C$. If $\mathcal{Q}$ is $\lambda$-defined by $C$ then we can partition the colors in the range of $\lambda$ into the colors in $C$, which we call intra-cluster colors, and the remaining colors, which we call cross-cluster colors. Note that if a color $c$ is intra-cluster, then for all $(v, w) \in E$ with $\lambda(v, w)=c$ it holds that $v, w \in Q$ for some $Q \in \mathcal{Q}$, and if $c$ is cross-cluster, then for all $(v, w) \in E$ with $\lambda(v, w)=c$ it holds that $(v, w)$ is a cross-cluster edge, that is, $v \in Q$ and $w \in Q^{\prime}$ for distinct $Q, Q^{\prime} \in \mathcal{Q}$.

- Lemma 4.1. There is an algorithm that, given
(A) an integer $d \geq 1$;
(B) two arc-colored tournaments $T_{1}=\left(V_{1}, E_{1}, \lambda_{1}\right)$ and $T_{2}=\left(V_{2}, E_{2}, \lambda_{2}\right)$;
(C) a set of colors $C$ and for $j=1,2$ a partition $\mathcal{Q}_{j}$ of $V_{j}$ that is $\lambda_{j}$-defined by $C$;
(D) a color $c^{*}$ that is cross-cluster with respect to $\mathcal{Q}_{j}$ for $j=1,2$ and
- for every $v \in V_{j}$ it holds that

$$
\left|\left\{Q \in \mathcal{Q}_{j} \mid \exists w \in Q:(v, w) \in E_{j} \wedge \lambda_{j}(v, w)=c^{*}\right\}\right| \leq d
$$

- for

$$
F_{j}:=\left\{\left(Q, Q^{\prime}\right) \in \mathcal{Q}_{j}^{2} \mid Q \neq Q^{\prime}, \exists w \in Q, w^{\prime} \in Q^{\prime}:\left(w, w^{\prime}\right) \in E_{j} \wedge \lambda_{j}\left(w, w^{\prime}\right)=c^{*}\right\}
$$

the directed graph $G_{j}=\left(\mathcal{Q}_{j}, F_{j}\right)$ is strongly connected;
(E) Iso $\left(T_{j}[Q], T_{j^{\prime}}\left[Q^{\prime}\right]\right)$ for every $j, j^{\prime} \in\{1,2\}$ and every $Q \in \mathcal{Q}_{j}, Q^{\prime} \in \mathcal{Q}_{j^{\prime}}$,
computes $\operatorname{Iso}\left(T_{1}, T_{2}\right)$ in time $d^{O(\log d)} \cdot n^{O(1)}$.
The proof builds on the algorithmic ideas presented in [1]. Since it is quite lengthy and technical, we only present a rough idea of the proof; the details can be found in the full version.

Proof Idea. The algorithm fixes an arbitrary vertex $r_{1} \in V_{1}$ and for every $r_{2} \in V_{2}$ computes the set $\operatorname{Iso}\left(\left(T_{1}, r_{1}\right),\left(T_{2}, r_{2}\right)\right)$ of all isomorphisms $\varphi \in \operatorname{Iso}\left(T_{1}, T_{2}\right)$ such that $\varphi\left(r_{1}\right)=r_{2}$. Observe that $\operatorname{Iso}\left(T_{1}, T_{2}\right)$ is the union over all these sets.

The central idea is to iteratively compute larger and larger sets $W_{1, i} \subseteq V_{1}$ and $W_{2, i} \subseteq V_{2}$ such that
(I.1) there is some $\mathcal{W}_{j, i} \subseteq \mathcal{Q}_{j}$ such that $W_{j, i}=\bigcup_{Q \in \mathcal{W}_{j, i}} Q$, and
(I.2) $\varphi\left(W_{1, i}\right)=W_{2, i}$ for every $\varphi \in \operatorname{Iso}\left(\left(T_{1}, r_{1}\right),\left(T_{2}, r_{2}\right)\right)$
and compute the set $\operatorname{Iso}\left(\left(T_{1}\left[W_{1, i}\right], r_{1}\right),\left(T_{2}\left[W_{2, i}\right], r_{2}\right)\right)$. Initially, we set $W_{1,0}:=R_{1}$ and $W_{2,0}:=R_{2}$ where $R_{1}$ and $R_{2}$ are the unique parts of $\mathcal{Q}_{1}$ and $\mathcal{Q}_{2}$ containing $r_{1}$ and $r_{2}$, respectively. Note that the set $\operatorname{Iso}\left(\left(T_{1}\left[W_{1,0}\right], r_{1}\right),\left(T_{2}\left[W_{2,0}\right], r_{2}\right)\right)$ can easily be computed from Item E.

Now suppose we already computed $\operatorname{Iso}\left(\left(T_{1}\left[W_{1, i}\right], r_{1}\right),\left(T_{2}\left[W_{2, i}\right], r_{2}\right)\right)$ for some $W_{1, i} \subseteq V_{1}$ and $W_{2, i} \subseteq V_{2}$ satisfying (I.1) and (I.2). Consider a vertex $u \in W_{1, i}$ that has an outgoing edge of color $c^{*}$ to a vertex outside of $W_{1, i}$. Note that, if $W_{1, i} \neq V_{1}$, such a vertex exists by the second part of Item D . We call such a vertex a boundary vertex. Let $U_{1, i}$ denote the set of boundary vertices. For a boundary vertex $u \in U_{1, i}$ let $\mathcal{L}_{1, i+1}^{u}$ denote the set of all parts $Q \in \mathcal{Q}_{1}$ that are outside of $W_{1, i}$ and contain a vertex $v \in Q$ such that $(u, v) \in E_{1}$ and $\lambda_{1}(u, v)=c^{*}$. Also, we define $L_{1, i+1}^{u}:=\bigcup_{Q \in \mathcal{L}_{1, i+1}^{u}} Q$. A visualization is given in Figure 2. Note that $\left|\mathcal{L}_{1, i+1}^{u}\right| \leq d$ by the first part of Item D.

For every boundary vertex $u \in U_{1, i}$ we construct an isomorphism-invariant tournament $\widetilde{T}_{1, i+1}^{u}$ with vertex set $\mathcal{L}_{1, i+1}^{u}$. Roughly speaking, for two distinct $Q, Q^{\prime} \in \mathcal{L}_{1, i+1}^{u}$, in order to decide whether to add $\left(Q, Q^{\prime}\right)$ or $\left(Q^{\prime}, Q\right)$ to the edge set of $\widetilde{T}_{1, i+1}^{u}$, we take a majority vote on the edges between $Q$ and $Q^{\prime}$ in $T_{1}$ (if there is a tie, we need to invoke further rules; see the full version for details).

Since $\left|\mathcal{L}_{1, i+1}^{u}\right| \leq d$, the automorphism group of $\widetilde{T}_{1, i+1}^{u}$ can be computed in time $d^{O(\log d)}$ by Theorem 2.4. Also, for each $Q \in \mathcal{L}_{1, i+1}^{u}$, the automorphism group of $T[Q]$ is given as part of the input (see Item E). By taking a wreath product, we obtain a solvable permutation $\operatorname{group} \Gamma \leq \operatorname{Sym}\left(L_{1, i+1}^{u}\right)$ such that $\operatorname{Aut}\left(T_{1}\left[L_{1, i+1}^{u}\right]\right) \leq \Gamma$. Using Theorem 2.5, this allows us to compute $\operatorname{Aut}\left(T_{1}\left[L_{1, i+1}^{u}\right]\right)$ and, more generally, compute the isomorphism sets between the corresponding subgraphs for different boundary vertices $u, u^{\prime} \in V_{1} \cup V_{2}$.

Now, we extend $W_{1, i}$ by all sets $L_{1, i+1}^{u}, u \in U_{1, i}$, to obtain the next layer $W_{1, i+1}$ (actually, for technical reasons, the final proof proceeds in a slightly different manner); the set $W_{2, i+1}$ is defined analogously. To compute the desired isomorphism set, we proceed as follows. For simplicity, first suppose that all sets $L_{1, i+1}^{u}, u \in U_{1, i}$, are pairwise disjoint. Then, we can again take a wreath product to obtain a solvable permutation group $\Delta \leq$ $\operatorname{Sym}\left(W_{1, i+1} \backslash W_{1, i}\right)$ such that $\operatorname{Aut}\left(T_{1}\left[W_{1, i+1}\right], r_{1}\right) \leq \operatorname{Aut}\left(T_{1}\left[W_{1, i}\right], r_{1}\right) \times \Delta$. Using Theorem 2.5 , this allows us to compute $\operatorname{Aut}\left(T_{1}\left[W_{1, i+1}\right], r_{1}\right)$. Similarly, we can also compute the set $\operatorname{Iso}\left(\left(T_{1}\left[W_{1, i+1}\right], r_{1}\right),\left(T_{2}\left[W_{2, i+1}\right], r_{2}\right)\right)$.


Figure 2 The figure shows the sets $W_{1, i}$ (orange), $U_{1, i}$ (blue) and $L_{1, i+1}^{u}$ computed in the proof sketch of Lemme 4.1. The color $c^{*}$ is shown in green and gray regions depict parts of the partition $\mathcal{Q}_{1}$.

To cover the case that not all sets $L_{1, i+1}^{u}, u \in U_{1, i}$, are pairwise disjoint, we use the following trick. For each vertex $v \in W_{1, i+1} \backslash W_{1, i}$ let $\mu(v)$ denote the number of boundary vertices $u \in U_{1, i}$ such that $v \in L_{1, i+1}^{u}$. We replace each $v \in W_{1, i+1} \backslash W_{1, i}$ with $\mu(v)$ copies of the vertex; each copy is associated with one of the corresponding boundary vertices. Now, we can proceed as in the previous case to compute the set of isomorphisms. Afterwards, we rely on group-theoretic algorithms from [35] to "merge" the different copies of the same vertex again to finally obtain the group $\operatorname{Iso}\left(\left(T_{1}\left[W_{1, i+1}\right], r_{1}\right),\left(T_{2}\left[W_{2, i+1}\right], r_{2}\right)\right)$.

Note that the second part of Item D guarantees that we can continue this process until eventually $W_{1, i}=V_{1}$ and $W_{2, i}=V_{2}$, at which point we have computed the desired set $\operatorname{Iso}\left(\left(T_{1}, r_{1}\right),\left(T_{2}, r_{2}\right)\right)$.

Building on the subroutine from Lemma 4.1, we can now design an isomorphism test for tournaments of bounded twin width following the outline given above.

- Theorem 4.2. There is an algorithm that, given two tournaments $T_{1}$ and $T_{2}$ and an integer $k \geq 1$, either concludes that $\operatorname{tww}\left(T_{1}\right)>k$ or computes $\operatorname{Iso}\left(T_{1}, T_{2}\right)$ in time $k^{O(\log k)} \cdot n^{O(1)}$.


## 5 The WL-Dimension of Tournaments of Bounded Twin Width

In this section, we prove Theorem 1.2. To prove that the WL algorithm on its own is unable to determine isomorphisms between tournaments of bounded twin width, we adapt a construction of Cai, Fürer and Immerman [12]. Towards this end, we first describe a construction of directed graphs with large WL dimension, and then argue how to translate those graphs into tournaments while preserving their WL dimension.

In the following, let $G$ be a connected, 3 -regular (undirected) graph. Let $G^{\text {red }}$ denote the structure obtained from $G$ be replacing every edge with a red edge and let $t:=\operatorname{tww}\left(G^{\text {red }}\right)$.

Let us remark at this point that the Cai-Fürer-Immerman construction [12] replaces each vertex in $G$ with a certain gadget, and those gadgets are connected along the edges of $G$. In order to bound the twin width of the resulting graph, we start with a graph $G$ for which the
twin width of $G^{\text {red }}$ is bounded. ${ }^{1}$ This is because the connections between gadgets are not homogeneous, so when contracting all gadgets in a contraction sequence, we obtain precisely the graph $G^{\text {red }}$ which allows to "complete" the contraction sequence using that $G^{\text {red }}$ has bounded twin width.

Now, let us formally describe the construction of tournaments with large WL dimension. By Lemma 2.2 there is some linear order $<$ on $V(G)$ such that $\operatorname{tww}\left(G^{\text {red }},<\right)=t$. Also, let $\vec{G}$ be an arbitrary orientation of $G$.

Recall that for every $v \in V(G)$ we denote by $E_{+}(v)$ the set of outgoing edges and $E_{-}(v)$ the set of incoming edges in $\vec{G}$. Also, we write $E(v)$ to denote the set of incident (undirected) edges in $G$. For $a \in \mathbb{Z}_{3}$ we define

$$
M_{a}(v):=\left\{f: E(v) \rightarrow \mathbb{Z}_{3} \mid \sum_{(v, w) \in E_{+}(v)} f(\{v, w\})-\sum_{(w, v) \in E_{-}(v)} f(\{v, w\})=a \quad(\bmod 3)\right\}
$$

We also define $F_{a}(v)$ to contain all pairs $(f, g) \in\left(M_{a}(v)\right)^{2}$ such that, for the minimal element $w \in N_{G}(v)$ (with respect to $<$ ) such that $f(v w) \neq g(v w)$, it holds that

$$
f(v w)+1=g(v w) \quad(\bmod 3) .
$$

Observe that, for every distinct $f, g \in M_{a}(v)$, either $(f, g) \in F_{a}(v)$ or $(g, f) \in F_{a}(v)$.
Let $\alpha: V(G) \rightarrow \mathbb{Z}_{3}$ be a function. We define the graph $\mathrm{CFI}_{3}(\vec{G},<, \alpha)$ with vertex set

$$
V\left(\mathrm{CFI}_{3}(\vec{G},<, \alpha)\right):=\bigcup_{v \in V(G)}\{v\} \times M_{\alpha(v)}(v)
$$

and edge set

$$
\begin{aligned}
E\left(\mathrm{CFI}_{3}(\vec{G},<, \alpha)\right):= & \{\{(v, f)(w, g)\} \mid v w \in E(G) \wedge f(v w)=g(v w)\} \\
& \cup\left\{((v, f)(v, g)) \mid(f, g) \in F_{\alpha(v)}(v)\right\}
\end{aligned}
$$

Observe that $\mathrm{CFI}_{3}(\vec{G},<, \alpha)$ is a mixed graph, i.e., it contains both directed and undirected edges. Also, we color the vertices of $\mathrm{CFI}_{3}(\vec{G},<, \alpha)$ using the coloring $\lambda: V\left(\mathrm{CFI}_{3}(\vec{G},<, \alpha)\right) \rightarrow C$ defined via $\lambda(v, f):=v$ for all $(v, f) \in V\left(\mathrm{CFI}_{3}(\vec{G}, \alpha)\right)$, i.e., each set $M_{\alpha(v)}(v)$ forms a color class under $\lambda$.

Now fix an arbitrary vertex $u_{0} \in V(G)$. For every $i \in \mathbb{Z}_{3}$ we define the mapping $\alpha_{i}: V(G) \rightarrow \mathbb{Z}_{3}$ via $\alpha_{i}\left(u_{0}\right):=i$ and $\alpha_{i}(w):=0$ for all $w \in V(G) \backslash\left\{u_{0}\right\}$.

- Lemma 5.1. $\mathrm{CFI}_{3}\left(\vec{G},<, \alpha_{0}\right) \neq \mathrm{CFI}_{3}\left(\vec{G},<, \alpha_{1}\right)$.

Now, we analyse the WL algorithm on the graphs $\mathrm{CFI}_{3}(\vec{G},<, \alpha)$ for different functions $\alpha: V(G) \rightarrow \mathbb{Z}_{3}$. We write $\operatorname{tw}(G)$ to denote the tree width of $G$.

- Lemma 5.2. Let $k$ be an integer such that $\operatorname{tw}(G) \geq k+1$. Also let $\alpha, \beta: V(G) \rightarrow \mathbb{Z}_{3}$ be two functions. Then $\mathrm{CFI}_{3}(\vec{G},<, \alpha) \simeq_{k} \mathrm{CFI}_{3}(\vec{G},<, \beta)$.

Together, Lemmas 5.1 and 5.2 provide pairs of non-isomorphic graphs that are not distinguished by $k$-WL assuming $\operatorname{tw}(G)>k$. Next, we argue how to turn these graphs into tournaments.

[^0]We define the tournament $T=T(\vec{G},<, \alpha)$ with vertex set

$$
V(T):=V\left(\mathrm{CFI}_{3}(\vec{G},<, \alpha)\right)=\bigcup_{v \in V(G)}\{v\} \times M_{\alpha(v)}(v)
$$

and edge set

$$
\begin{aligned}
E(T):= & \left\{((v, f)(v, g)) \mid(f, g) \in F_{\alpha(v)}(v)\right\} \\
& \cup\left\{((v, f)(w, g)) \mid v<w \wedge\{(v, f)(w, g)\} \notin E\left(\mathrm{CFI}_{3}(\vec{G},<, \alpha)\right)\right\} \\
& \cup\left\{((w, g)(v, f)) \mid v<w \wedge\{(v, f)(w, g)\} \in E\left(\mathrm{CFI}_{3}(\vec{G},<, \alpha)\right)\right\} .
\end{aligned}
$$

It can be shown that the relevant properties are preserved by this translation.

- Lemma 5.3. Let $\alpha, \beta: V(G) \rightarrow \mathbb{Z}_{3}$ be two functions. Then $T(\vec{G},<, \alpha) \cong T(\vec{G},<, \beta)$ if and only if $\mathrm{CFI}_{3}(\vec{G},<, \alpha) \cong \mathrm{CFI}_{3}(\vec{G},<, \beta)$.
- Lemma 5.4. Let $k \geq 2$ be an integer such that $\operatorname{tw}(G) \geq k+1$. Also let $\alpha, \beta: V(G) \rightarrow \mathbb{Z}_{3}$ be two functions. Then $T(\vec{G},<, \alpha) \simeq_{k} T(\vec{G},<, \beta)$.

To prove Theorem 1.2, we also need to bound the twin width of the resulting graph. Recall that $t:=\operatorname{tww}\left(G^{\text {red }}\right)$ where $G^{\text {red }}$ denotes the version of $G$ where every edge is red.

- Lemma 5.5. For every function $\alpha: V(G) \rightarrow \mathbb{Z}_{3}$ it holds that $\operatorname{tww}(T(\vec{G},<, \alpha)) \leq \max (35, t)$.

Proof. Throughout the proof, we define $M(v):=M_{\alpha(v)}(v)$ for every $v \in V(G)$. Since $G$ is 3-regular, we have that $|M(v)|=9$ for every $v \in V(G)$. So $|V(T(\vec{G},<, \alpha))|=9 \cdot|V(G)|$. Also note that $(M(v), M(w))$ is homogeneous for all distinct $v, w \in V(G)$ such that $\{v, w\} \notin E(G)$.

We construct a partial contraction sequence as follows. Let $n:=|V(G)|$. We define $\mathcal{P}_{1}, \ldots, \mathcal{P}_{8 n+1}$ arbitrarily such that $\mathcal{P}_{8 n+1}=\{M(v) \mid v \in V(G)\}$. Since $G$ is 3-regular and $|M(v)|=9$ for every $v \in V(G)$, we conclude that $(T(\vec{G},<, \alpha)) / \mathcal{P}_{i}$ has red degree at most $4 \cdot 9-1=35$ for every $i \in[8 n+1]$. Now observe that

$$
\operatorname{tww}\left((T(\vec{G},<, \alpha)) / \mathcal{P}_{8 n+1}\right) \leq \operatorname{tww}\left(G^{\mathrm{red}},<\right)=t
$$

It follows that $\operatorname{tww}(T(\vec{G},<, \alpha)) \leq \max (35, t)$ as desired.
With Lemma 5.5 in hand, we apply the construction $T(\vec{G},<, \alpha)$ to a 3-regular base graph $G$ which has tree width linear in the number of vertices, but the twin width of $G^{\text {red }}$ is bounded. The existence of such graphs has already been observed in [5]. More precisely, the following theorem follows from combining the arguments from [5, Lemma 5.1] and the results from [4, 34].

- Theorem 5.6. There is a family of 3-regular graphs $\left(G_{n}\right)_{n \geq 1}$ such that $\left|V\left(G_{n}\right)\right|=O(n)$, $\operatorname{tww}\left(G_{n}^{\text {red }}\right) \leq 6$ (where $G_{n}^{\text {red }}$ denotes the version of $G_{n}$ where all edges are turned into red edges), and $\operatorname{tw}\left(G_{n}\right) \geq n$ for every $n \geq 1$.

Now, we are ready to give a proof for Theorem 1.2.

- Theorem 5.7. For every $k \geq 2$ there are non-isomorphic tournaments $T_{k}$ and $T_{k}^{\prime}$ such that
(1) $\left|V\left(T_{k}\right)\right|=\left|V\left(T_{k}^{\prime}\right)\right|=O(k)$,
(2) $\operatorname{tww}\left(T_{k}\right) \leq 35$ and $\operatorname{tww}\left(T_{k}^{\prime}\right) \leq 35$, and
(3) $T_{k} \simeq_{k} T_{k}^{\prime}$.

Proof. Let $k \geq 2$. Let $G_{k+1}$ be the 3-regular graph obtained from Theorem 5.6. Note that $\operatorname{tw}\left(G_{k+1}\right) \geq k+1$.

We also fix an arbitrary orientation $\vec{G}_{k+1}$ of $G_{k+1}$ and let $G_{k+1}^{\text {red }}$ denote the version of $G_{k+1}$ where every edge is replaced by a red edge. We have $t:=\operatorname{tww}\left(G_{k+1}^{\text {red }}\right) \leq 6$ by Theorem 5.6. By Lemma 2.2 there is some linear order $<$ on $V\left(G_{k+1}\right)$ such that tww $\left(G_{k+1}^{\text {red }},<\right) \leq t \leq 6$.

Now fix an arbitrary $u_{0} \in V\left(G_{k+1}\right)$. For $p \in \mathbb{Z}_{3}$ we define the mapping $\alpha_{p}: V\left(G_{k+1}\right) \rightarrow \mathbb{Z}_{3}$ via $\alpha_{p}\left(u_{0}\right)=p$ and $\alpha_{p}(w)=0$ for all $w \in V\left(G_{k+1}\right) \backslash\left\{u_{0}\right\}$. We define $T_{k}:=T\left(\vec{G}_{k+1},<, \alpha_{0}\right)$ and $T_{k}^{\prime}:=T\left(\vec{G}_{k+1},<, \alpha_{1}\right)$. We have $\left|V\left(T_{k}\right)\right|=\left|V\left(T_{k}^{\prime}\right)\right|=9 \cdot\left|V\left(G_{k+1}\right)\right|=O(k)$. Also, $\operatorname{tww}\left(T_{k}\right) \leq 35$ and $\operatorname{tww}\left(T_{k}^{\prime}\right) \leq 35$ by Lemma 5.5. Finally, $T_{k} \neq T_{k}^{\prime}$ by Lemmas 5.1 and 5.3, and $T_{k} \simeq_{k} T_{k}^{\prime}$ by Lemma 5.4.

## 6 Twin Width is Smaller Than Other Widths

In this section, we compare twin width with other natural width parameters of tournaments. If $f, g$ are mappings from (directed) graphs to the natural numbers, we say that $f$ is functionally smaller than $g$ on a class $\mathcal{C}$ of graphs if for every $k$ there is a $k^{\prime}$ such that for all graphs $G \in \mathcal{C}$, if $g(G) \leq k$ then $f(G) \leq k^{\prime}$. We write $f \precsim c g$ to denote that $f$ is functionally smaller than $g$ on $\mathcal{C}$. We omit the subscript $\mathcal{c}$ if $\mathcal{C}$ is the class of all digraphs.

Natural width measures for directed graphs are cut width, directed path width, directed tree width, and clique width. On the class of tournaments twin width turns out to be functionally smaller than all of these. For clique width, it has already been shown in [11] that twin width is functionally smaller than clique width on undirected graphs; the proof easily extends to arbitrary binary relational structures and hence to tournaments.

We start by giving definitions for the other width measures. Let $G$ be a digraph. For a linear order $\leq$ on $V(G)$ and a vertex $v \in V(G)$, we let $S_{\leq}(v):=\{w \in V(G) \mid w \leq v\}$ be the set of all vertices smaller than or equal to $v$ in $\leq$. Let $s_{\leq}(v):=\left|E_{G}\left(S_{\leq}(v), V(G) \backslash S_{\leq}(v)\right)\right|$ be the number of edges from $S_{\leq}(v)$ to its complements. The width of $\leq$ is $\max _{v \in V(G)} s_{\leq}(v)$, and the cut width $\operatorname{ctw}(G)$ is the minimum over the width of all linear orders of $V(G)$.

A directed path decomposition of a digraph $G$ is a mapping $\beta:[p] \rightarrow 2^{V(G)}$, for some $p \in \mathbb{N}$, such that for every vertex $v \in V(G)$ there are $\ell, r \in[p]$ such that $v \in \beta(t) \Longleftrightarrow \ell \leq t \leq r$, and for all edges $(v, w) \in E(G)$ there are $\ell, r \in[p]$ with $\ell \leq r$ such that $v \in \beta(r)$ and $w \in \beta(\ell)$. The sets $\beta(t), t \in[p]$, are the bags of the decomposition. The width of the decomposition is $\max _{t \in[p]}|\beta(t)|-1$, and the directed path width $\operatorname{dpw}(G)$ is the minimum width of a directed path decomposition of $G$.

A digraph $R$ is a rooted directed tree if there is a vertex $r_{0} \in V(R)$ such that for every $t \in V(R)$ there is a unique directed walk from $r_{0}$ to $t$. Note that every rooted directed tree can be obtained from an undirected tree by selecting a root $r_{0}$ and directing all edges away from the root. For $t \in V(R)$ we denote by $R_{t}$ the unique induced subgraph of $R$ rooted at $t$.

Let $G$ be a digraph. A directed tree decomposition of $G$ is a triple $(R, \beta, \gamma)$ where $R$ is a rooted directed tree, $\beta: V(R) \rightarrow 2^{V(G)}$ and $\gamma: E(R) \rightarrow 2^{V(G)}$ such that
(D.1) $\{\beta(t) \mid t \in V(R)\}$ is a partition of $V(G)$, and
(D.2) for every $(s, t) \in E(R)$ the set $\gamma(s, t)$ is a hitting set for all directed walks that start and end in $\beta\left(R_{t}\right):=\bigcup_{t^{\prime} \in V\left(R_{t}\right)} \beta\left(t^{\prime}\right)$ and contain a vertex outside of $\beta\left(R_{t}\right)$.
For $t \in V(R)$ we define $\Gamma(t):=\beta(t) \cup \bigcup_{\left(s, s^{\prime}\right) \in E(t)} \gamma\left(s, s^{\prime}\right)$ where $E(t)$ denotes the set of edges incident to $t$. The width of $(R, \beta, \gamma)$ is defined as

$$
\operatorname{width}(R, \beta, \gamma):=\max _{t \in V(R)}|\Gamma(t)|-1
$$

The directed tree width $\operatorname{dtw}(G)$ is the minimum width of a directed tree decomposition of $G$.

Let us first recall the following well-known inequalities.

- Proposition 6.1. For all digraphs $G$, it holds that $\operatorname{dtw}(G) \leq \operatorname{dpw}(G) \leq \operatorname{ctw}(G)$.

The proposition implies that dtw $\precsim \mathrm{dpw} \precsim \mathrm{ctw}$. It can be shown that tww $\not \mathbb{L}$ ctw and hence tww $\not \mathscr{L}$ dpw and tww $\not \mathscr{L}$ dtw on the class of all digraphs. In contrast, it turns out that on the class of tournaments, twin width is functionally smaller than directed tree width. Actually, this even holds for the larger class of semi-complete graphs. A digraph $G$ is semi-complete if for all distinct $v, w \in V(G)$ at least one of the pairs $(v, w),(w, v)$ is an edge. Note that every tournament is semi-complete.

- Theorem 6.2 ([28, Proposition 5]). Let $G$ be a semi-complete graph. Then

$$
\operatorname{dpw}(G) \leq 4(\operatorname{dtw}(G)+2)^{2}+7(\operatorname{dtw}(G)+2)-1
$$

- Theorem 6.3. Let $G$ be a semi-complete graph. Then $\operatorname{tww}(G) \leq \operatorname{dpw}(G)$.

In combination, we get that $\operatorname{tww}(G) \leq 4(\operatorname{dtw}(G)+2)^{2}+7(\operatorname{dtw}(G)+2)-1$ for every semi-complete graph $G$. In particular

$$
\begin{equation*}
\text { tww } \precsim_{\mathcal{S}} \mathrm{dtw}, \tag{2}
\end{equation*}
$$

where $\mathcal{S}$ denotes the class of all semi-complete digraphs. This inequality is strict even on tournaments, that is, dtw $\not \mathscr{L}_{\mathcal{T}}$ tww where $\mathcal{T}$ denotes the class of all tournaments.

## 7 Conclusion

We prove that the isomorphism problem for classes of tournaments of bounded (or slowly growing) twin width is in polynomial time. Many algorithmic problems that can be solved efficiently on (classes of) tournaments can also be solved efficiently on (corresponding classes of) semi-complete graphs, that is, directed graphs where for every pair $(v, w)$ of vertices at least one of the pairs $(v, w),(w, v)$ is an edge (see, e.g., [39]). Contrary to this, we remark that isomorphism of semi-complete graphs of bounded twin width is GI-complete: we can reduce isomorphism of oriented graphs to isomorphism of semi-complete graphs by replacing each non-edge by a bidirectional edge. This reduction preserves twin with.

Classes of tournaments of bounded twin width are precisely the classes that are considered to be structurally sparse. Formally, these are the classes that are monadically dependent, which means that all set systems definable over the tournaments in such a class have bounded VC dimension. The most natural set systems definable within a tournament are those consisting of the in-neighbors of the vertices and of the out-neighbors of the vertices. Bounded twin width implies that the VC dimension of these two set systems is bounded, but the converse does not hold. It is easy to see that the VC dimensions of the in-neighbors and out-neighbors systems as well as the set system consisting of the mixed neighbors of all edges are within a linear factor of one another. As a natural next step, we may ask if isomorphism of tournaments where the VC-dimension of these systems is bounded is in polynomial time.

## References

1 Vikraman Arvind, Ilia N. Ponomarenko, and Grigory Ryabov. Isomorphism testing of kspanning tournaments is fixed parameter tractable. CoRR, abs/2201.12312, 2022. arXiv: 2201.12312.

2 László Babai. Graph isomorphism in quasipolynomial time [extended abstract]. In Daniel Wichs and Yishay Mansour, editors, Proceedings of the 48 th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016, pages 684-697. ACM, 2016. doi:10.1145/2897518. 2897542.
3 László Babai and Eugene M. Luks. Canonical labeling of graphs. In David S. Johnson, Ronald Fagin, Michael L. Fredman, David Harel, Richard M. Karp, Nancy A. Lynch, Christos H. Papadimitriou, Ronald L. Rivest, Walter L. Ruzzo, and Joel I. Seiferas, editors, Proceedings of the 15th Annual ACM Symposium on Theory of Computing, 25-27 April, 1983, Boston, Massachusetts, USA, pages 171-183. ACM, 1983. doi:10.1145/800061.808746.
4 Yonatan Bilu and Nathan Linial. Lifts, discrepancy and nearly optimal spectral gap. Comb., 26(5):495-519, 2006. doi:10.1007/s00493-006-0029-7.
5 Édouard Bonnet, Colin Geniet, Eun Jung Kim, Stéphan Thomassé, and Rémi Watrigant. Twin-width II: small classes. In Dániel Marx, editor, Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10-13, 2021, pages 1977-1996. SIAM, 2021. doi:10.1137/1.9781611976465.118.
6 Édouard Bonnet, Colin Geniet, Eun Jung Kim, Stéphan Thomassé, and Rémi Watrigant. Twin-width III: max independent set, min dominating set, and coloring. In Nikhil Bansal, Emanuela Merelli, and James Worrell, editors, 48 th International Colloquium on Automata, Languages, and Programming, ICALP 2021, July 12-16, 2021, Glasgow, Scotland (Virtual Conference), volume 198 of LIPIcs, pages 35:1-35:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.ICALP.2021.35.
7 Édouard Bonnet, Ugo Giocanti, Patrice Ossona de Mendez, Pierre Simon, Stéphan Thomassé, and Szymon Torunczyk. Twin-width IV: ordered graphs and matrices. In Stefano Leonardi and Anupam Gupta, editors, STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20-24, 2022, pages 924-937. ACM, 2022. doi:10.1145/ 3519935.3520037.

8 Édouard Bonnet, Ugo Giocanti, Patrice Ossona de Mendez, and Stéphan Thomassé. Twinwidth V: linear minors, modular counting, and matrix multiplication. In Petra Berenbrink, Patricia Bouyer, Anuj Dawar, and Mamadou Moustapha Kanté, editors, 40th International Symposium on Theoretical Aspects of Computer Science, STACS 2023, March 7-9, 2023, Hamburg, Germany, volume 254 of LIPIcs, pages 15:1-15:16. Schloss Dagstuhl - LeibnizZentrum für Informatik, 2023. doi:10.4230/LIPIcs.STACS.2023.15.
9 Édouard Bonnet, Eun Jung Kim, Amadeus Reinald, and Stéphan Thomassé. Twin-width VI: the lens of contraction sequences. In Joseph (Seffi) Naor and Niv Buchbinder, editors, Proceedings of the 2022 ACM-SIAM Symposium on Discrete Algorithms, SODA 2022, Virtual Conference / Alexandria, VA, USA, January 9-12, 2022, pages 1036-1056. SIAM, 2022. doi:10.1137/1.9781611977073.45.
10 Édouard Bonnet, Eun Jung Kim, Amadeus Reinald, Stéphan Thomassé, and Rémi Watrigant. Twin-width and polynomial kernels. Algorithmica, 84(11):3300-3337, 2022. doi:10.1007/ s00453-022-00965-5.
11 Édouard Bonnet, Eun Jung Kim, Stéphan Thomassé, and Rémi Watrigant. Twin-width I: tractable FO model checking. J. ACM, 69(1):3:1-3:46, 2022. doi:10.1145/3486655.
12 Jin-yi Cai, Martin Fürer, and Neil Immerman. An optimal lower bound on the number of variables for graph identification. Comb., 12(4):389-410, 1992. doi:10.1007/BF01305232.
13 Maria Chudnovsky, Alexandra Ovetsky Fradkin, and Paul D. Seymour. Tournament immersion and cutwidth. J. Comb. Theory, Ser. B, 102(1):93-101, 2012. doi:10.1016/j.jctb. 2011.05. 001.

14 Maria Chudnovsky, Ringi Kim, Chun-Hung Liu, Paul D. Seymour, and Stéphan Thomassé. Domination in tournaments. J. Comb. Theory, Ser. B, 130:98-113, 2018. doi:10.1016/j. jctb.2017.10.001.
15 Maria Chudnovsky, Alex Scott, and Paul D. Seymour. Disjoint paths in unions of tournaments. J. Comb. Theory, Ser. B, 135:238-255, 2019. doi:10.1016/j.jctb.2018.08.007.

16 Maria Chudnovsky and Paul D. Seymour. A well-quasi-order for tournaments. J. Comb. Theory, Ser. B, 101(1):47-53, 2011. doi:10.1016/j.jctb.2010.10.003.
17 John D. Dixon and Brian Mortimer. Permutation Groups, volume 163 of Graduate Texts in Mathematics. Springer-Verlag, New York, 1996. doi:10.1007/978-1-4612-0731-3.
18 Walter Feit and John G. Thompson. Solvability of groups of odd order. Pacific J. Math., 13:775-1029, 1963.
19 Fedor V. Fomin and Michal Pilipczuk. On width measures and topological problems on semi-complete digraphs. J. Comb. Theory, Ser. B, 138:78-165, 2019. doi:10.1016/j.jctb. 2019.01.006.

20 Jakub Gajarský, Stephan Kreutzer, Jaroslav Nesetril, Patrice Ossona de Mendez, Michal Pilipczuk, Sebastian Siebertz, and Szymon Torunczyk. First-order interpretations of bounded expansion classes. ACM Trans. Comput. Log., 21(4):29:1-29:41, 2020. doi:10.1145/3382093.
21 Jakub Gajarský, Michal Pilipczuk, and Szymon Torunczyk. Stable graphs of bounded twinwidth. In Christel Baier and Dana Fisman, editors, LICS '22: 37th Annual ACM/IEEE Symposium on Logic in Computer Science, Haifa, Israel, August 2-5, 2022, pages 39:1-39:12. ACM, 2022. doi:10.1145/3531130. 3533356.
22 Robert Ganian, Filip Pokrývka, André Schidler, Kirill Simonov, and Stefan Szeider. Weighted model counting with twin-width. In Kuldeep S. Meel and Ofer Strichman, editors, 25th International Conference on Theory and Applications of Satisfiability Testing, SAT 2022, August 2-5, 2022, Haifa, Israel, volume 236 of LIPIcs, pages 15:1-15:17. Schloss Dagstuhl -Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.SAT.2022.15.
23 Colin Geniet and Stéphan Thomassé. First order logic and twin-width in tournaments. In Inge Li Gørtz, Martin Farach-Colton, Simon J. Puglisi, and Grzegorz Herman, editors, 31st Annual European Symposium on Algorithms, ESA 2023, September 4-6, 2023, Amsterdam, The Netherlands, volume 274 of LIPIcs, pages 53:1-53:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPIcs.ESA.2023.53.
24 Martin Grohe. Descriptive Complexity, Canonisation, and Definable Graph Structure Theory, volume 47 of Lecture Notes in Logic. Cambridge University Press, 2017. doi:10.1017/ 9781139028868.

25 Martin Grohe and Daniel Neuen. Recent advances on the graph isomorphism problem. In Konrad K. Dabrowski, Maximilien Gadouleau, Nicholas Georgiou, Matthew Johnson, George B. Mertzios, and Daniël Paulusma, editors, Surveys in Combinatorics, 2021: Invited lectures from the 28th British Combinatorial Conference, Durham, UK, July 5-9, 2021, pages 187-234. Cambridge University Press, 2021. doi:10.1017/9781009036214.006.
26 Martin Grohe and Daniel Neuen. Canonisation and definability for graphs of bounded rank width. ACM Trans. Comput. Log., 24(1):6:1-6:31, 2023. doi:10.1145/3568025.
27 Martin Grohe, Daniel Neuen, and Pascal Schweitzer. A faster isomorphism test for graphs of small degree. SIAM J. Comput., 52(6):S18-1, 2023. doi:10.1137/19m1245293.
28 Frank Gurski, Dominique Komander, Carolin Rehs, and Sebastian Wiederrecht. Directed width parameters on semicomplete digraphs. In Ding-Zhu Du, Donglei Du, Chenchen Wu, and Dachuan Xu, editors, Combinatorial Optimization and Applications - 15th International Conference, COCOA 2021, Tianjin, China, December 17-19, 2021, Proceedings, volume 13135 of Lecture Notes in Computer Science, pages 615-628. Springer, 2021. doi:10.1007/ 978-3-030-92681-6_48.
29 Petr Hlinený and Jan Jedelský. Twin-width of planar graphs is at most 8, and at most 6 when bipartite planar. In Kousha Etessami, Uriel Feige, and Gabriele Puppis, editors, 50th International Colloquium on Automata, Languages, and Programming, ICALP 2023, July 10-14, 2023, Paderborn, Germany, volume 261 of LIPIcs, pages 75:1-75:18. Schloss Dagstuhl -Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPIcs.ICALP.2023.75.
30 Neil Immerman and Eric Lander. Describing graphs: A first-order approach to graph canonization. In Alan L. Selman, editor, Complexity Theory Retrospective: In Honor of Juris Hartmanis on the Occasion of His Sixtieth Birthday, July 5, 1988, pages 59-81. Springer New York, New York, NY, 1990. doi:10.1007/978-1-4612-4478-3_5.

31 Thor Johnson, Neil Robertson, Paul D. Seymour, and Robin Thomas. Directed tree-width. J. Comb. Theory, Ser. B, 82(1):138-154, 2001. doi:10.1006/jctb.2000. 2031.
32 Sandra Kiefer. The Weisfeiler-Leman algorithm: an exploration of its power. ACM SIGLOG News, 7(3):5-27, 2020. doi:10.1145/3436980.3436982.
33 Eugene M. Luks. Isomorphism of graphs of bounded valence can be tested in polynomial time. J. Comput. Syst. Sci., 25(1):42-65, 1982. doi:10.1016/0022-0000 (82) 90009-5.

34 Adam W. Marcus, Daniel A. Spielman, and Nikhil Srivastava. Interlacing families I: Bipartite Ramanujan graphs of all degrees. Ann. of Math. (2), 182(1):307-325, 2015. doi:10.4007/ annals.2015.182.1.7.
35 Gary L. Miller. Isomorphism of graphs which are pairwise k-separable. Inf. Control., 56(1/2):2133, 1983. doi:10.1016/S0019-9958(83)80048-5.
36 Jaroslav Nesetril and Patrice Ossona de Mendez. Structural sparsity. Russian Math. Surveys, 71(1):79-107, 2016. doi:10.4213/rm9688.
37 Jaroslav Nesetril, Patrice Ossona de Mendez, and Sebastian Siebertz. Structural properties of the first-order transduction quasiorder. In Florin Manea and Alex Simpson, editors, 30th EACSL Annual Conference on Computer Science Logic, CSL 2022, February 14-19, 2022, Göttingen, Germany (Virtual Conference), volume 216 of LIPIcs, pages 31:1-31:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.CSL.2022.31.
38 Daniel Neuen. Hypergraph isomorphism for groups with restricted composition factors. ACM Trans. Algorithms, 18(3):27:1-27:50, 2022. doi:10.1145/3527667.
39 Michal Pilipczuk. Tournaments and Optimality: New Results in Parameterized Complexity. PhD thesis, University of Bergen, 2013.
40 Ilia N. Ponomarenko. Polynomial time algorithms for recognizing and isomorphism testing of cyclic tournaments. Acta Appl. Math., 29(1-2):139-160, 1992. doi:10.1007/BF00053383.
41 Joseph J. Rotman. An Introduction to the Theory of Groups, volume 148 of Graduate Texts in Mathematics. Springer-Verlag, New York, fourth edition, 1995. doi:10.1007/ 978-1-4612-4176-8.
42 Pascal Schweitzer. A polynomial-time randomized reduction from tournament isomorphism to tournament asymmetry. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, 44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland, volume 80 of LIPIcs, pages 66:1-66:14. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2017. doi:10.4230/LIPIcs.ICALP.2017.66.
43 Ákos Seress. Permutation Group Algorithms, volume 152 of Cambridge Tracts in Mathematics. Cambridge University Press, Cambridge, 2003. doi:10.1017/CB09780511546549.
44 Stéphan Thomassé. A brief tour in twin-width (invited talk). In Mikolaj Bojanczyk, Emanuela Merelli, and David P. Woodruff, editors, 49th International Colloquium on Automata, Languages, and Programming, ICALP 2022, July 4-8, 2022, Paris, France, volume 229 of LIPIcs, pages 6:1-6:5. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.ICALP.2022.6.
45 Boris Weisfeiler. On Construction and Identification of Graphs, volume 558 of Lecture Notes in Mathematics. Springer-Verlag, 1976.
46 Boris Weisfeiler and Andrei Leman. The reduction of a graph to canonical form and the algebra which appears therein. NTI, Series 2, 1968. English translation by Grigory Ryabov available at https://www.iti.zcu.cz/wl2018/pdf/wl_paper_translation.pdf.


[^0]:    ${ }^{1}$ We remark that, since $G$ has maximum degree 3 , the twin width of $G^{\text {red }}$ is actually bounded in the twin width of $G$. However, we feel it is more convenient to directly bound the twin width of $G^{\text {red }}$.

