Directed Elimination Games

Autoren:
Viktor Engelmann, Stephan Kreutzer, Sebastian Ordyniak

Dezember 2013
Directed Elimination Games

Viktor Engelmann\(^1\), Stephan Kreutzer\(^1\), Sebastian Ordyniak\(^2\)

\(^1\)Chair for Logic and Semantics, Technische Universität Berlin, viktor.engelmann@tu-berlin.de, stephan.kreutzer@tu-berlin.de
\(^2\)Faculty of Informatics, Masaryk University Brno, sordyniak@gmail.com

Abstract

While tools from structural graph theory such as tree- or path-width have proved to be highly successful in coping with computational intractability on undirected graphs, corresponding width measures for directed graphs have not yet fulfilled their promise for broad algorithmic applications on directed graphs. One reason for this is that in most existing digraph width measures the class of acyclic digraphs has small width which immediately implies hardness of problems such as computing directed dominating sets.

In [11], Fernau and Meister introduce the concept of elimination width and a corresponding graph searching game which overcomes this problem with acyclic digraphs. In their paper, the focus was on structural characterisations of classes of digraphs of bounded elimination width.

In this paper we study elimination width from an algorithmic and graph searching perspective. We analyse variations of the elimination width game commonly studied in the literature and show that this leads to width measures on which the directed dominating set problem and some variants of it become tractable.

1 Introduction

Structural graph theory has proved to be a powerful tool for coping with computational intractability. It provides a wealth of concepts and results that can be used to design efficient algorithms for hard computational problems on specific classes of graphs that occur naturally in applications. For instance, it has been shown that many problems which in general are NP-complete become tractable on planar graphs or graphs of bounded genus. Here, tractable can mean solvable in polynomial time or fixed-parameter tractable, the analogous concept of efficient solvability in parameterized complexity (see e.g. [10]).

Of particular importance in this context are structural properties such as tree-width or path-width introduced by Robertson and Seymour as part of their celebrated graph minor project [22]. See e.g. [9] for an introduction to these concepts. Most problems that are tractable on trees also remain tractable on classes of graphs of bounded tree-width and a huge number of papers is devoted to developing efficient algorithms for a wide range of problems on such classes of graphs. See e.g. [5, 6] and references therein.

In many applications in computer science, directed graphs are the most natural model. Unfortunately, so far research in structural graph theory has
almost exclusively focused on undirected graphs and no structure theory for directed graphs has been developed that would provide for a similar set of tools and concepts to deal with hard computational problems on digraphs.

Reed [21] and Johnson et al. [16] initiated the development of a decomposition theory for directed graphs with the aim of defining an analogue of the concept of undirected tree-width for directed graphs. Following their definition of a directed tree-width, several alternative notions have been introduced, for instance in [23, 3, 4, 15]. For each of these decompositions and associated width measures it was shown that several problems become tractable if the width of digraphs with respect to these measures is bounded by a fixed constant. However, most examples of problems becoming tractable are either linkage problems, i.e. problems asking for the existence of certain pairwise disjoint paths in the digraph, or certain combinatorial games arising in verification. The only exception is bi-rank-width, which was introduced by Kanté et al. in [17] for a very different purpose than the other digraph width measures as part of the ongoing quest to find suitable structural parameters generalising the concept of clique-width to more general structures than undirected graphs. On classes of digraphs of bounded bi-rank-width all problems definable in monadic second-order logic (MSO) become tractable. However, in the context of algorithmic applications to digraphs, monadic second-order logic is too powerful, as it allows to specify on a directed graph properties of its underlying undirected graph. Bi-rank-width is therefore a special case in digraph width measures as bounded bi-rank-width implies bounded clique-width of the underlying undirected graphs whereas the other width measures were introduced specifically so that classes of bounded width do not also have bounded tree- or clique-width.

Following these initial proposals for directed analogues of tree-width, several papers investigated how broad the algorithmic theory of classes of digraphs of bounded width with respect to these measures is. Unfortunately, for many interesting problems other than those mentioned before, strong intractability results were obtained showing that the algorithmic applicability of the existing directed width measures is very limited. See e.g. [19, 18, 7, 13] and references therein.

In nearly all existing digraph width measures the class of acyclic digraphs (DAGs) has very small width. The negative results reported above show very clearly that exactly this assumption, that DAGs are “simple”, is a major obstacle for algorithmic applications of directed width measures, as standard problems such as the directed dominating set problem are NP-complete and fixed-parameter intractable already on very simple acyclic digraphs (see Section 7 for details). Hence, if we aim for a width measures which can be used in the analysis of problems such as directed dominating sets, then it must necessarily split the class of DAGs into simple and hard instances.

In [20], a concept called nowhere crownful classes of digraphs was introduced which achieves this goal. Nowhere crownful classes are very general, in particular they include all classes of digraphs whose underlying undirected graphs exclude a fixed minor or are nowhere dense. And yet problems such as the directed dominating set problem and others become fixed-parameter tractable. However, this generality comes at a price as it is easily seen that disjoint paths problems remain hard on nowhere crownful classes of digraphs. Hence, more research and different ideas are needed to find parameters on which more and different problems become tractable.
Elimination width. An elegant and promising proposal for such a parameter was given by Fernau and Meister in [11]. Like many other digraph width parameters, in particular directed tree-width [16], DAG-width [4] and Kelly-width [15], it is based on graph searching games. In a graph searching game, a number of cops tries to catch a robber hiding on the vertices of a graph or digraph. That is, in every round of the play, the robber occupies a vertex and so does each of the cops. The game then proceeds as follows. In every round the cop player can lift some cops and place them on arbitrary new vertices. While the cops are in transit from their old to the new positions, the robber can also change his position. The different variants of the game are determined by the rules the robber has to follow when moving to a new position, but in all graph searching games, the robber is not allowed to move through cops that are not moved. The cops win if they can place a cop on the position occupied by the robber without the robber being able to escape. Otherwise, i.e. if the robber can escape forever, then he wins (see Section 3 for details).

Graph searching games yield a natural graph invariant, namely the minimal number of cops needed to catch a robber on a given graph. Naturally, different versions of the game yield different graph invariants and there are games corresponding exactly to the tree- or the path-width of a graph. For directed graphs, many width measures can also be defined by graph searching games, namely the strongly connected visible robber game for directed tree-width, monotone directed visible robber game for DAG-width, the inert invisible robber game for Kelly-width and the agile, invisible robber game for directed path-width. In all of these games, the robber can only move along directed paths and this immediately implies that on DAGs a very small number of cops suffices to capture the robber: all they have to do is to force the robber along a directed path until he reaches a sink, i.e. a vertex with no successor.

The novel idea proposed in [11] is to consider games, called directed elimination games, where the robber can run along a directed path or against a directed path, i.e. choose a new position reachable from his current position by a directed path or from which there is a directed path to his current position. In this way they obtain games where the number of cops needed to catch a robber on the class of DAGs is unbounded. The main purpose of their work in [11] is to study the structure of graphs of bounded elimination width, i.e. on which a bounded number of cops suffices to capture the robber in the elimination game. In particular, it turns out that such classes can equivalently be defined by perfect elimination orderings and other structural characterisations, i.e. form a robust class of directed graphs.

Our contributions. The purpose of this paper is to further explore elimination games and in particular study their applications for algorithmic digraph problems. The variant of the game studied in [11] is what is called the inert invisible robber game in the graph searching literature. As explained above, on directed graphs, different variants of graph searching games yield very different width measures with completely different properties. In this paper we therefore first study the other standard variants of graph searching games equipped with the elimination width idea of robber movement. Of particular importance in the context of graph searching games are monotone variants of the games, as monotone strategies for the cop player often yield simple and useful recursive decompositions of the graphs such as tree- or DAG-decompositions. We there-
fore analyse monotonicity of the resulting games and compare the corresponding width measures.

In the second part of the paper, then, we study algorithmic applications of classes of graphs of bounded search number in these games. As we will show, we are indeed able to solve problems such as the directed dominating set problem efficiently, which remains computationally intractable on classes of bounded width in the existing digraph width measures. These results demonstrate that the idea of elimination games holds the promise for further algorithmic applications and to overcome the problems with acyclic digraphs encountered by current directed width measures.

2 Preliminaries

We use \( \mathbb{N} \) to denote the set of natural numbers (non-negative integers). For a set \( X \) we denote by \( \mathcal{P}(X) \) the powerset of \( X \).

**Directed and Undirected Graphs.** We use standard notation from graph theory as can be found in, e.g., [9, 2]. All graphs and directed graphs in this paper are finite and simple. Let \( G \) be a (directed) graph. We denote the vertex set of \( G \) by \( V(G) \) and the edge set of \( G \) by \( E(G) \). Let \( X \subseteq V(G) \) be a set of vertices of \( G \). The sub-graph of \( G \) induced by \( X \), denoted \( G[X] \), is the graph with vertex set \( X \) and edges \( E(G) \cap (X \times X) \) if \( G \) is directed and edges \( E(G) \cap [X]^2 \) if \( G \) is undirected. By \( G \setminus X \) we denote the sub-graph of \( G \) induced by \( V(G) \setminus X \). Similarly for \( Y \subseteq E(G) \) we define \( G \setminus Y \) to be the sub-graph of \( G \) obtained by deleting all edges in \( Y \) from \( G \). For an undirected graph \( G \) and a vertex \( v \in V(G) \) we denote by \( N_G(v) \) and \( N_G[v] \) the open and closed neighborhood of \( v \) in \( G \), respectively. Similarly, if \( G \) is directed we denote by \( N^+_G(v) \) and \( N^-_G[v] \) the open and closed out-neighborhood of \( v \) and \( G \) and by \( N^-_G(v) \) and \( N^+_G[v] \) the open and closed in-neighborhood of \( v \) and \( G \). We omit the subscript \( G \) if the graph \( G \) is clear from the context. Furthermore, if \( G \) is undirected we denote by \( \Delta(G) \) the maximum degree of any vertex in \( G \) and if \( G \) is directed we denote by \( \Delta^+(G) \) and \( \Delta^-(G) \) the maximum out-degree and the maximum in-degree of any vertex \( v \) of \( G \), respectively. For a directed graph \( D \) we define its underlying undirected graph, denoted by \( \bar{D} \), as the undirected graph with vertex set \( V(D) \) and edge set \( \{ (u, v) : (u, v) \in E(D) \} \).

**Parameterized Complexity.** An instance of a parameterized problem is a pair \((I, k)\) where \( I \) is the main part and \( k \) is the parameter; the latter is usually a non-negative integer. A parameterized problem is fixed-parameter tractable if there exist a computable function \( f \) and a constant \( c \) such that instances \((I, k)\) can be solved in time \( O(f(k)||I||^c) \) where \( ||I|| \) denotes the size of \( I \). FPT is the class of all fixed-parameter tractable decision problems and algorithms which run in the time specified above are called FPT algorithms.

The Weft Hierarchy consists of parameterized complexity classes \( W[1] \subseteq W[2] \subseteq \cdots \) which are defined as the closure of certain parameterized problems under FPT-reductions (see [10, 12] for definitions). There is strong theoretical evidence that parameterized problems that are hard for classes \( W[i] \) are not fixed-parameter tractable. For example \( \text{FPT} = W[1] \) implies that the Exponential Time Hypothesis (ETH) fails; that is, \( \text{FPT} = W[1] \) implies the existence of a \( 2^{o(n)} \) algorithm for \( n \)-variable 3SAT [12].
3 Directed Elimination Games

In this section we formally introduce directed elimination games. We first explain one particular variant of the game, the visible robber game played on directed graphs, and then comment on the variations studied in the literature later on. As explained in the introduction, elimination games where first introduced in [11]. The games studied there broadly correspond to the inert, invisible robber game below, except that in [11] two different types of cops are used which we do not in our paper. The corresponding width measures therefore can differ by a factor of 2.

3.1 Games with a Visible Robber

The graph searching game we are going to describe is played by two players, the cop and the robber, on a directed graph $G$. The game is played in rounds where in each round the players can place or remove tokens on and from the vertices of the graph. The cop controls an unbounded amount of tokens whereas the robber only has one token.

Initially, there are no tokens on the graph $G$. The game begins by the cops placing some tokens on the graph followed by the robber placing his token on a vertex that is not yet occupied by a cop. This completes the first round.

In the following rounds the cops can move some of their tokens to new vertices, they can remove some of their tokens from the graph and they can place new tokens on arbitrary vertices. However, moving tokens takes some time during which the robber can move his token to any position on the graph that can either be reached from his current position by a directed path that contains no vertex occupied by a cop or that can reach his current position by a directed path that contains no vertex occupied by a cop. Formally, suppose that after some rounds the current position is $(X,v)$, i.e. the cops occupy the vertices in $X \subseteq V(G)$ and the robber is on $v \in V(G) \setminus X$. In the next round, the cops must first announce their new position $X' \subseteq V(G)$. The robber can then choose any position $v' \in V(G) \setminus X'$ that is either reachable from $v$ by a directed path in $G \setminus (X \cap X')$ or that can reach $v$ by a directed path in $G \setminus (X \cap X')$. The play continues at $(X',v')$.

If there is no such position, i.e., if $v \in X'$ and there is no vertex to which the robber can escape, he is caught and has lost. Otherwise, if the robber can escape forever he wins the play. Hence, the robber’s goal is to avoid capture by the cops.

Clearly, on any graph the cops have a very simple winning strategy by placing one token on every vertex of the graph. We can therefore associate with every directed graph $G$ the minimal number of tokens required by the cops to guarantee capture of the robber. We call this number the visible search number $sw_{vis}(G)$ of $G$. $sw_{vis}(G)$ is a graph invariant closely related to the internal connectivity of $G$.

A strategy for the cops in a visible directed elimination game is a function that tells the cops where to move to from the current position $(X,v)$ of the play. Formally, cop strategies in the visible game are functions $f : \mathcal{P}(V(G)) \times V(G) \to \mathcal{P}(V(G))$ assigning to each pair $(X,v)$, with $X \subseteq V(G)$ and $v \in V(G)$, a new set $f(X,v) \subseteq V(G)$.

For graph searching games such as the directed elimination game one gen-
erally considers two types of monotonicity, i.e., one distinguishes between cop monotonicity and robber monotonicity. In particular, a cop strategy is *cop monotone* if at no point during a play where the cop is following this strategy he has to move to a vertex from which he has removed a token before. This means, once a token is removed from a vertex, the cop is not allowed to put a token back on that vertex. Furthermore, a strategy for the cop is *robber monotone* if whenever at some point of the play the robber cannot reach a vertex \( v \), then it will not be able to reach \( v \) at any later point in the play. However, due to the nature of the directed elimination game we are not sure how to properly define robber monotonicity in the case of a visible robber. We therefore refrain from using robber monotonicity in the visible case and refer to cop monotonicity simply as monotonicity. Therefore, a strategy \( f : P(V(G)) \times V(G) \to P(V(G)) \) for the cop is monotone if the cop never revisits a previously vacated vertex in every play that is consistent with the strategy \( f \).

Obviously monotone strategies are more restrictive than general strategies. We can therefore define the *monotone search width* \( \text{mon-sw}_{vis}(G) \) as the minimal number of tokens required for a monotone winning strategy of the cop.

### 3.2 Games with an Invisible Robber

The games described in the previous section are played between the cop trying to catch a robber which he can see at all times. An important variation of this idea is to make the robber invisible. Historically, this was the first definition of graph searching modelling the idea that a search party is trying to find a person that is lost in a system of tunnels whose position they do not know in advance.

With the exception of the robber being invisible to the cop, the rules of the game are unchanged otherwise. That is, initially the board is empty and the game begins by the cop placing some tokens on the graph followed by the robber choosing his initial position. In each round, from position \( X \) the cop announces his new position \( X' \) and the robber can move from his current position along a directed path not containing a vertex of \( X \cap X' \) to a new position. He is caught if the cop places a token on his current position but he is unable to escape to a safe place.

The crucial difference between the visible and invisible case is the type of strategies the cop employs. For, in the visible case the strategy was essentially a tree, where for each position \( X \) of the cop the strategy provides a possible move for the cop for each possible robber response. In the invisible case there is no need to distinguish between robber moves as the cop does not know where the robber is. Hence they have to search the complete graph linearly to catch the robber independent of his actions.

We can therefore represent a cop strategy on a graph \( G \) in the invisible graph searching game by a finite or infinite sequence \( \mathcal{S} := (X_1, \ldots, X_n) \) or \( \mathcal{S} := (X_1, \ldots) \) of cop positions. With any such strategy we associate the corresponding sequence of robber spaces \( \mathcal{R} := (R_1, \ldots, R_n) \) or \( \mathcal{R} := (R_1, \ldots) \), where \( R_1 := V(G) \setminus X_1 \) and

\[
R_{i+1} := \left\{ u \in V(G) \setminus X_{i+1} : \begin{array}{l}
\text{there exists } v \in R_i \text{ and a directed path} \\
\text{from } v \text{ to } u \text{ or from } u \text{ to } v \text{ in } G \setminus (X_i \cap X_{i+1})
\end{array} \right\},
\]

for all \( i \geq 1 \).
A strategy is winning if \( R_i := \emptyset \) for some \( i \geq 1 \). Clearly, \( S \) is cop monotone if for every \( 1 \leq i < l < j \) it holds that \( X_i \cap X_j \subseteq X_l \). Similarly, \( S \) is robber monotone if \( R_i \supseteq R_{i+1} \) for all \( i \). Again, \( S \) is monotone if it is both cop and robber monotone. The following proposition shows that there is no need to distinguish cop and robber monotone strategies for the invisible variant of the game.

**Proposition 3.3.** Let \( D \) be a digraph and \( k \) a natural number. If \( k \) cops have a robber monotone or a cop monotone winning strategy on \( D \) then \( k \) cops have a monotone winning strategy on \( D \).

**Proof.** Suppose \( S := (X_1, \ldots, X_n) \) is a robber monotone winning strategy for \( k \) cops on \( D \) and let \( \mathcal{R} := (R_1, \ldots, R_n) \) be the associated sequence of robber spaces. W.l.o.g. we can assume that \( S \) is inclusion minimal, i.e., there is no index \( 1 \leq i \leq n \) and vertex \( v \in X_i \) such that \( S' := (X_1, \ldots, X_i \setminus \{v\}, \ldots, X_n) \) is a robber monotone winning strategy for \( k \) cops on \( D \). If \( S \) is also cop monotone then there is nothing to show. Hence, we can assume that \( S \) is not cop monotone. Consequently, there are indices \( 1 \leq i < l < j \) and a vertex \( v \in V(D) \) such that \( X_i \cap X_j \subseteq X_l \) and \( v \in X_i \cap X_j \) and \( v \notin X_l \). We will show that the strategy \( S' := (X_1, \ldots, X_i \setminus \{v\}, \ldots, X_n) \) is a robber monotone winning strategy for \( k \) cops on \( D \) contradicting the inclusion minimality of \( S \). To see this let \( \mathcal{R}' := (R'_1, \ldots, R'_n) \) be the sequence of robber spaces associated with \( S' \). It suffices to show that \( R'_r = R_r \) for every \( 1 \leq r \leq n \). This clearly holds for every \( 1 \leq r < j \) because \( X'_r = X_r \) for every such \( r \). As \( v \in X_i \) and \( S \) is robber monotone it follows that \( v \notin R_r \) for every \( r \geq i \). Furthermore, because \( v \notin X_l \) it follows that there is no arc between \( v \) and a vertex in \( R_l \) and hence there is no arc between \( v \) and a vertex in \( R_r \) for any \( r \geq l \). In particular, \( v \notin R_{j-1} = R'_{j-1} \) and there is no arc between \( v \) and a vertex in \( R_{j-1} = R'_{j-1} \). Hence, \( R'_r = R_r \). Applying the same argument again for \( R_j \) instead of \( R_{j-1} \) we obtain \( R'_{j+1} = R_{j+1} \). Because \( X'_r = X_r \) for every \( r \geq j \) it follows that \( R'_r = R_r \) for every such \( r \).

For the reverse direction suppose that \( S := (X_1, \ldots, X_n) \) is a cop monotone winning strategy for \( k \) cops on \( D \) and let \( \mathcal{R} := (R_1, \ldots, R_n) \) be the associated sequence of robber spaces. If \( S \) is also robber monotone there is nothing to show. Hence, we can assume that \( S \) is not robber monotone and there is some \( 1 \leq i \) and some vertex \( v \in V(D) \) such that \( v \notin R_i \) and \( v \in R_{i+1} \). Furthermore, because \( S \) is cop monotone it holds that \( v \notin X_j \) for every \( j > i \). Consequently, \( v \in R_j \) for every \( j > i \) contradicting our assumption that \( S \) is a winning strategy.

Because of the above proposition we only need to consider monotone strategies in the sequel. The minimal number of cops required to catch a robber on a graph \( G \) is the invisible search number of \( G \), denoted \( \text{sw}_{\text{invis}}(G) \). The minimal number of tokens required for a monotone strategy is denoted by \( \text{monsw}_{\text{invis}}(G) \).

### 3.4 Games with an Inert Robber

The last variant of the game that we consider is a variation of the invisible robber game. The crucial difference is that now the robber can only move if the cop is about to place a token on his current position. These games were first introduced for undirected graphs in \( \mathbb{S} \). More formally, a cop strategy in
a graph $G$ is again defined as a finite or infinite sequence $S := (X_1, \ldots, X_n)$ or $S := (X_1, \ldots)$ of sets $X_i \subseteq V(G)$ of vertices but the corresponding robber spaces are now defined as a sequence $R := (R_1, \ldots, R_n)$ or $R := (R_1, \ldots)$, where $R_1 := V(G) \setminus X_1$ and

$$R_{i+1} := \{ u \in V(G) \setminus X_{i+1} : \text{ there exists } v \in R_i \cap X_{i+1} \text{ and a directed path from } v \text{ to } u \text{ or from } u \text{ to } v \text{ in } G \setminus (X_i \cap X_{i+1}) \}.$$ 

Again $S$ is winning if $R_k := \emptyset$. Monotonicity is defined as for the invisible variant of the game. That is, $S$ is cop monotone if for every $1 \leq i < l < j$ it holds that $X_i \cap X_j \subseteq X_l$. Similarly, $S$ is robber monotone if $R_i \supseteq R_{i+1}$ for all $i$. Again, $S$ is monotone if it is both cop and robber monotone.

The following proposition shows that the cop monotone variant of the inert directed elimination game corresponds to the monotone variant of the invisible directed elimination game.

**Proposition 3.5.** Let $G$ be a graph and $k$ an integer. Then $k$ cops have a cop monotone winning strategy in the directed elimination game on $G$ if and only if $k$ cops have a monotone winning strategy in the invisible directed elimination game.

**Proof.** Suppose that $S := (X_1, \ldots, X_n)$ is a cop monotone winning strategy in the directed elimination game on $G$ and let $R := (R_1, \ldots, R_n)$ be the associated sequence of robber spaces. We first show that $S$ is also robber monotone. Suppose this is not the case then there is some $1 \leq i$ and some vertex $v \in V(D)$ such that $v \in R_i$ and $v \notin R_{i+1}$. Furthermore, because $S$ is cop monotone it holds that $v \notin X_j$ for every $j > i$. Consequently, $v \in R_j$ for every $j > i$ contradicting our assumption that $S$ is a winning strategy. It follows that $S$ is monotone. It remains to show that $S$ is also a winning strategy for the cops in the invisible directed elimination game. Let $R' := (R'_1, \ldots, R'_n)$ be the sequence of robber spaces associated with $S$ in the invisible directed elimination game. It suffices to show that $R'_i = R_i$ for every $1 \leq i \leq n$. Suppose not and let $1 \leq i \leq n$ be minimal such that $R'_i \neq R_i$. Clearly, $i > 1$. It follows that there is a vertex $v \in R_{i-1}$ that has an arc to or from some vertex $v$ in $G \setminus (R_{i-1} \cup (X_{i-1} \cap X_i))$. Because $S$ is cop monotone and $v \notin R_{i-1}$ it follows that $v \notin X_l$ for any $l \geq i-1$. Because $S$ is a winning strategy in the inert directed elimination game there is some $i \leq j$ such that $r \in X_j$. Let $j$ be minimal with this property. Because there is an arc between $r$ and $v$ and $S$ is robber monotone it follows that $v \in X_j$ contradicting the cop monotonicity of $S$.

The reverse direction is obvious. \qed

We will hence refrain from using cop monotonicity in the context of the inert directed elimination game in the sequel and will refer to robber monotonicity simply as monotonicity.

The minimal number of cops required to catch a robber on a graph $G$ is the inert search number of $G$, denoted $\text{sw}_{\text{inert}}(G)$. The minimal number of tokens required for a monotone strategy is denoted by $\text{mon-sw}_{\text{inert}}(G)$.

The following lemma gives an upper bound on $\text{mon-sw}_{\text{inert}}(D)$ for a DAG $D$ that we use frequently in the sequel.

**Lemma 3.6.** If $D$ is a DAG, then $\text{mon-sw}_{\text{inert}}(D) \leq \Delta^- (D) + 1$ and $\text{mon-sw}_{\text{inert}}(D) \leq \Delta^+ (D) + 1$. 

8
Figure 1: Alternatingly oriented grid.

Proof. Because $D$ is a DAG there is an ordering $v_1, \ldots, v_n$ of the vertices of $D$ such that $D$ contains no arc from a vertex $v_j$ to a vertex $v_i$ for every $1 \leq i < j \leq n$.

We first show that $\text{mon-sw}_{\text{inert}}(D) \leq \Delta^-(D) + 1$ by giving a monotone winning strategy $S$ for $\Delta^-(D) + 1$ cops in the inert directed elimination game on $D$. The strategy $S$ is defined as follows: $S := (N^-(v_n), N^-[v_n], \ldots, N^-(v_1), N^-[v_1])$.

Similarly, we show that $\text{mon-sw}_{\text{inert}}(D) \leq \Delta^+(D) + 1$ by giving a monotone winning strategy $S$ for $\Delta^+(D) + 1$ cops in the inert directed elimination game on $D$. The strategy $S$ is defined as $S := (N^-(v_1), N^-[v_1], \ldots, N^-(v_n), N^-[v_n])$.

4 Comparison to Directed and Undirected Width Measures

In this section we compare the variants of the directed elimination game to the known width measures on directed and undirected graphs.

4.1 Kelly-Width and DAG-width

The relationship between Kelly-Width and the inert directed elimination game has been studied in [11]. There it is shown that the inert search number is always larger than the Kelly-width and that the gap between the two can be arbitrary high even on acyclic graphs. Almost identical arguments can be used to establish the same relationship between the DAG-width and the visible search number.

4.2 Tree-Width

Because the monotone variants of the visible and inert directed elimination game correspond to treewidth on undirected graphs it follows that $\text{mon-sw}_{\text{vis}}(D) \leq \text{mon-sw}_{\text{vis}}(\overline{D}) = \text{tw}(D)$ and $\text{mon-sw}_{\text{vis}}(D) \leq \text{mon-sw}_{\text{vis}}(\overline{D}) = \text{tw}(D)$ for every directed graph $D$. Here we show that the gap between the treewidth and both the visible and inert directed elimination game can be arbitrary large.

Theorem 4.3. For every $q \in \mathbb{N}$ there is a digraph $G_q$ with $\text{tw}(\overline{G_q}) = q$, but $\text{mon-sw}_{\text{vis}}(G_q)$ and $\text{mon-sw}_{\text{inert}}(G_q)$ are bounded by a constant.

Proof. Let $q \in \mathbb{N}$ and let $G_{q,q}$ be the undirected $q \times q$-grid. Then the digraph $G_q$ is a directed orientation of $G_{q,q}$ such that every vertex in $G_q$ is either a source or a sink, i.e., has in-degree 0 or out-degree 0, respectively. More formally,
Every edge $\{u,v\}$, $1 \leq i,j \leq q$, and the arc set of $G_q$ is the orientation of the edge set of $G_q$. Such that a vertex $v_{i,j}$ is a sink if $i + j - 1 \mod 2 = 0$, and a source otherwise. See Figure 1 for an illustration of $G_q$.

We start by showing that $\text{tw}(G_q) = q$. It is well-known that the treewidth of the $q \times q$-grid $G_{q,q}$ is $q$. Because $G_q$’s underlying undirected graph is $G_{q,q}$, it follows that $\text{tw}(G_q) = q$.

To show that $\text{mon-sw}_{\text{inv}}(G_q)$ is bounded by a constant, we observe that the function $f$ with $f(\emptyset, r) = N_{G_q}(r)$ for every $r \in V(G_q)$ is a monotone winning strategy for $|N_{G_q}(r)| \leq 5$ cops in the visible directed elimination game on $G_q$.

To show that $\text{mon-sw}_{\text{inert}}(G_q)$ is bounded by a constant, we observe that $G_q$ is a directed acyclic graph. Hence, it follows from Lemma 3.6 that $\text{mon-sw}_{\text{inert}}(G_q) \leq \Delta^+(G_q) + 1 = 5$. This concludes the proof of the theorem.

4.4 Path-Width

Because the monotone variants of the invisible directed elimination game correspond to pathwidth on undirected graphs, it follows that $\text{mon-sw}_{\text{inv}}(D) \leq \text{mon-sw}_{\text{inert}}(D) = \text{pw}(D)$ for every directed graph $D$. Here we show that the two graph invariants are actually equal.

A path decomposition of an undirected graph $G$ is a sequence $P := (X_1, \ldots, X_n)$ of vertex set of $G$ that satisfies the following properties:

- **PW1** $\bigcup_{1 \leq i \leq n} X_i = V(G)$;
- **PW2** For every edge $\{u, v\} \in E(G)$ there is an $1 \leq i \leq n$ such that $\{u, v\} \subset X_i$;
- **PW3** For every $1 \leq i < l < j \leq n$ it holds that $(X_i \cap X_j) \subseteq X_l$.

The width of a path decomposition $P := (X_1, \ldots, X_n)$ is $\min_{1 \leq i \leq n} |X_i| - 1$. The path-width of a graph $G$ is the minimum width of any path decomposition of $G$.

**Theorem 4.5.** For every digraph $D$ it holds that $\text{pw}(\overline{D}) = \text{mon-sw}_{\text{inv}}(D) - 1$.

**Proof.** The path-width of a graph can be defined in terms of a variant of a cops and robber game where the robber is agile and invisible and can move along undirected paths in the graph that are not occupied by a cop. If $G$ is an undirected graph, then the variant of the cops and robber game that defines pathwidth corresponds to the monotone invisible directed elimination game. Hence, if $D$ is a directed graph, we obtain $\text{mon-sw}_{\text{inv}}(D) - 1 \leq \text{mon-sw}_{\text{inert}}(D) - 1 = \text{pw}(D)$.

It remains to show that $\text{mon-sw}_{\text{inert}}(D) - 1 \geq \text{pw}(D)$. To show that let $S = (X_1, \ldots, X_n)$ be a monotone winning strategy for the cop in the invisible directed elimination game on $D$. We show that $P := (X_1, \ldots, X_n)$ is a path decomposition of the undirected graph underlying $D$. Property PW1 follows from the fact that $S$ is a winning strategy. Towards showing Property PW2 suppose that there is an edge $\{u, v\} \in E(G)$ such that $\{u, v\} \not\subseteq X_i$ for every $1 \leq i \leq n$. The edge $\{u, v\}$ corresponds to at least 1 arc in $D$, i.e., either the arc $(u, v)$ or the arc $(v, u)$. Consider the following strategy for the robber:

- the robber starts by going on the vertex $v$;
- if the robber is on $v$ and the cop places a token on $v$ then the robber moves along the arc between $v$ and $u$ to $u$;
if the robber is on $u$ and the cop places a token on $u$ then the robber moves along the arc between $v$ and $u$ to $v$;

Because no set $X_i$ contains both vertices $u$ and $v$ the robber can always escape using the above strategy. This contradicts our assumption that $S$ is a winning strategy for the cop player and shows Property PW2. Property PW3 now follows from the cop monotonicity of $S$. Because the width of $P$ is the number of cops used by $S$ minus 1 we obtain that $\text{mon-sw}_{\text{invis}}(D) - 1 \geq \text{pw}(D)$.

5 Non-Monotonicity of the Invisible Game

In this section we show that the invisible directed elimination game is not monotone. In fact there is an arbitrary difference between the monotone and non-monotone search number of the game.

Theorem 5.1. For every $q \in \mathbb{N}$ there is a graph $T_q$ with $\text{sw}_{\text{invis}}(T_q) \leq 3$ and $\text{mon-sw}_{\text{invis}}(T_q) \geq q - 1$.

Proof. For every $q \in \mathbb{N}$ the graph $T_q$ is a directed graph whose underlying undirected graph is a rooted tree. In the following we will denote the root of $T_q$ by $r(T_q)$. The graph $T_q$ is inductively constructed as follows. $T_1$ consists of an isolated vertex that acts also as the root of $T_1$. The graph $T_q$ consists of 2 copies, i.e., the left copy $L$ and the right copy $R$, of $T_{q-1}$, a new vertex $r$, which will act as the root vertex of $T_q$, an arc from $r(L)$ to $r$ and an alternating path between $r$ and $r(R)$ of length at least $l_{q-1}$ where $l_{q-1}$ is the length of a winning strategy for 3 cops in the invisible directed elimination game on $T_{q-1}$ (such a strategy is described below). An alternating path is a path on which the directions of the arcs are alternating. More formally, every vertex on an alternating path is either a source or a sink vertex, i.e., every vertex on this path has either in-degree 0 or out-degree 0. Because the directions of the arcs on such a path are alternating the robber can traverse at most 1 arc of the alternating path at every step of the game. This completes the construction of the graph $T_q$.

We first show that $\text{mon-sw}_{\text{invis}}(T_q) \geq q - 1$. It follows from Theorem 4.5 that $\text{mon-sw}_{\text{invis}}(T_q) = \text{pw}(T_q)$. It is well-known that the path width of a complete binary tree of height $q$ is at least $q - 1$ and that the path-width of a graph $G$ is at least the path-width of any graph that is obtained from $G$ by subdividing the edges of $G$. It is straightforward to verify that the undirected graph underlying $T_q$ can be obtained from a complete binary tree of height $q$ by subdividing its edges. Consequently, $\text{pw}(T_q) \geq q - 1$ and hence $\text{mon-sw}_{\text{invis}}(T_q) \geq q - 1$.

It remains to show that $\text{sw}_{\text{invis}}(T_q) \leq 3$. To show this we inductively construct a winning strategy $S_q := (S_1, \ldots, S_n)$ for 3 cops in the invisible directed elimination game on $T_q$ that satisfies the following properties:

P1 $r(T_q) \in S_n$, and

P2 Let $G$ be a graph that contains a copy of $T_q$ such that there is no arc between a vertex of $G \setminus T_q$ and a vertex of $T_q \setminus \{r(T_q)\}$ and let $\mathcal{R} := (R_1, \ldots, R_n)$ be the sequence of robber spaces associated to $S_q$ played on the copy of $T_q$ in $G$. Then $R_n \cap V(T_q) = \emptyset$. 

11
Figure 2: The digraph $D_4$ from the proof of Theorem 6.1.

We set $S_1 := \{r(T_1)\}$. Because $r(T_1)$ is the only vertex of $T_1$ the strategy $S_1$ is a winning strategy in the directed elimination game that uses only 1 cop and satisfies Properties P1 and P2.

Now assume that $q > 1$ and let $L$ be the left copy of $T_{q-1}$ in $T_q$, $R$ be the right copy of $T_{q-1}$ in $T_q$, and let $P = (p_1, \ldots, p_{q-1})$ be the sequence of the internal vertices of the alternating path from $r(T_q)$ to $r(R)$ (in the order that they appear on the alternating path and such that $p_1$ is adjacent to $r(T_q)$). Furthermore, let $S(L)$ and $S(R)$ be winning strategies for 3 cops on $L$ and $R$, respectively given by the induction hypothesis. Then $S_q$ is the concatenation of the following sequences:

- $S(R)$ (clear $R$);
- $\{r(R), p_{q-1}, \ldots, p_2, p_1, r(T_q)\}$ (clear the alternating path starting from $r(R)$ towards $r(T_q)$);
- $S(L)$ (clear $L$);
- $\{r(L), r(T_q), r(R)\}$ (prepare to clear the alternating path again);
- $\{r(T_q), r(R), p_{q-1}, r(T_q), p_{q-1-1}, \ldots, r(T_q), p_2, p_1\}$ (clear the alternating path again from $r(R)$ towards $r(T_q)$ but this time always keep a cop on $T_q$).

It is straightforward to verify that $S_q$ is a winning strategy in the invisible directed elimination game on $T_n$ that satisfies Properties P1 and P2.

6 Comparison of the Visible and the Inert Game

In this section we show that in contrast to their undirected counterparts the visible and inert directed elimination game do not coincide. In fact the difference between the visible search number and the inert search number can be arbitrary high.

**Theorem 6.1.** For every $q \in \mathbb{N}$ there is a digraph $D_q$, such that $\text{mon-sw}_{\text{inert}}(D_q) \leq 5$, but $\text{sw}_{\text{invis}}(D_q) \geq q$ and $\text{sw}_{\text{vis}}(D_q) \geq q$.

**Proof.** Let $q \in \mathbb{N}$. Let $T_q$ be a binary tree of lowest height that has $q$ leaves, let $T_q^u$ be the directed graph obtained from $T_q$ after directing all edges towards the root of $T_q$ and similarly let $T_q^d$ be the directed graph obtained from $T_q$ after directing all edges away from the root of $T_q$. If $T$ is one of $T_q^u$ or $T_q^d$, ...
we denote by $r(T)$ the root of $T$ and by $l(T, i)$ the $i$-th leave of $T$ for every $1 \leq i \leq q$. Now the graph $D_q$ consists of $q$ copies $T^u(1), \ldots, T^u(q)$ of $T^u$ and $q$ copies $T^d(1), \ldots, T^d(q)$ of $T^d$. Additionally, the graph $D_q$ contains the arcs \( (l(T^d(i), j), l(T^u(j), i)) : 1 \leq i, j \leq q \}. \) This completes the construction of the graph $D_q$. See Figure 2 for an illustration of the graph $D_q$.

Because $D_q$ is a directed acyclic graph where both the maximum in-degree and the maximum out-degree are bounded by 2 it follows from Lemma 3.6 that $\text{mon-sw}_{\text{inert}}(D_q) \leq \Delta^-(D_q) + 1 = 3$.

To show that $\text{sw}_{\text{invis}}(D_q) \geq q$ and $\text{sw}_{\text{vis}}(D_q) \geq q$ consider the following winning strategy for the robber against $q$ cops in the invisible and the visible directed elimination game on $D_q$:

- The robber starts by playing on $r(T^u(1))$;
- If the robber is on $r(T^u(i))$ for some $1 \leq i \leq q$ and the cop player places its first token on a vertex of $T^u(i)$, then the robber moves to $r(T^d(j))$ where $1 \leq j \leq q$ such that there is no cop placed on a vertex of $T^d(j)$. Note that because the cop player has only $q$ tokens such a $j$ does always exist;
- If the robber is on $r(T^d(i))$ for some $1 \leq i \leq q$ and the cop player places its first token on a vertex of $T^d(i)$, then the robber moves to $r(T^u(j))$ where $1 \leq j \leq q$ such that there is no cop placed on a vertex of $T^u(j)$. Note that because the cop player has only $q$ tokens such a $j$ does always exist.

7 Algorithmic Applications

In this section we study algorithmic applications of elimination games. As explained in the introduction, directed width measures such as directed tree-width or DAG-width have been introduced with algorithmic applications for digraph problems in mind. However, one of the major obstacles for such applications is the fact that DAGs all have small width in these measures.

This becomes apparent when considering the directed dominating set problem, that is, the problem, given a digraph $G$ and $k \in \mathbb{N}$, to decide whether there is a set $S \subseteq V(G)$ of size at most $k$ such that $N^+[S] = V(G)$. As the following theorem shows, the problem is computationally intractable already on acyclic digraphs and hence on classes of bounded directed tree-width, Kelly-width or DAG-width.

**Theorem 7.1** ([13, 20]). The directed dominating set problem is NP-complete and $W[2]$-hard on the class of acyclic digraphs.

We have already seen that elimination games have unbounded search number on the class of acyclic digraphs. Hence, digraph decompositions based on these games hold the promise to overcome the problem with DAGs encountered by existing width measures. As evidence for this we show next that the directed dominating set problem is fixed-parameter tractable on classes of bounded elimination with.
Theorem 7.2. The directed dominating set problem is fixed-parameter tractable in $k + w$, where $k$ is the solution size and $w$ the search number in any of the variants of the elimination-width games considered in this paper.

Proof. Let $D$ a digraph. We first show that the degeneracy of $\bar{D}$ is smaller than the search number of any variant of the directed elimination game considered in this paper. Let $H$ be the induced subgraph of $\bar{D}$ with $\delta(H) = \deg(\bar{D})$ which exists due to [9, Proposition 5.2.2]. It is straightforward to verify that the robber has a winning strategy against $\deg(D)$ cops by staying on the vertices of $H$. Hence, for any variant of the directed elimination game considered in this paper we obtain that the degeneracy is bounded by the search width of the variant. It follows that $D$ has bounded degeneracy. Hence the result follows from the fixed-parameter tractability of Directed Dominating Set parameterized by $k + \deg(D)$ [1].

As the previous result shows, the directed dominating set problem does indeed become tractable on graph classes of bounded elimination width even though it is intractable in other directed width measures. However, the proof relies on the fact that the problem is already tractable on degenerate classes of digraphs. It would be interesting to see whether the distance-$d$-version of the problem becomes tractable on such classes, that is, the problem given a digraph $G$ and numbers $k, d \in \mathbb{N}$, to decide whether there is a set $S \subseteq V(G)$ of size at most $k$ such that every vertex $V \in V(G)$ can be reached from some $w \in S$ in at most $d$ steps. We have not been able to prove this but we can prove a slightly weaker result.

For this, we consider a slight variation of the elimination game. For any $q \in \mathbb{N} \cup \{\infty\}$, the $q$-alternation elimination game on a digraph $G$ is played according to the same rules as the standard elimination game, with the exception that in each round, the robber is no longer bound to follow a directed path or the reversal of a directed path, but he can now move to a new position as long as there is a path from his current position to the new position in which he changes at most $q$ times from following edges in the right direction to using reversed edges or back. Hence, for $q = 0$, this is the standard elimination game and for $q = \infty$ this becomes graph searching games on the underlying undirected graphs and hence equivalent to tree- or path-width, depending on the variant of the game played.

However, as the following theorem proves, allowing the robber to change direction once every moves yields a graph invariant on which the distance-$d$-dominating set problem becomes fixed-parameter tractable.

Theorem 7.3. For all variants of the elimination-game, the distance-$d$-dominating set problem is fixed-parameter tractable on classes of digraphs of bounded search number in the $1$-alternation elimination game.

The theorem follows immediately from the observation, that any such class of digraphs must be nowhere crownful as defined in [20]. As the distance-$d$-dominating set problem is fixed-parameter tractable on such classes, the result follows.

Whether this result can be improved to 0-alternation elimination games remains open and we leave this for future research.

The previous results establish fixed-parameter tractability results for directed dominating set problems parameterized by the solution size and the
width. We show next that this cannot be improved to fixed-parameter tractability, or even containment in XP, in the width as only parameter. Hence, we cannot expect polynomial time algorithms for these problems on classes of bounded width. This is not too surprising as the dominating set problem is a very hard problem that remains NP-complete on many restricted classes of graphs such as planar graphs [14].

**Theorem 7.4.** **Directed Dominating Set remains NP-Complete on graphs of bounded mon-sw_{inert} and mon-sw_{vis}**

**Proof.** We show the theorem by reducing to 3-SAT-2, i.e., the variant of 3-SAT where every literal occurs in at most 2 clauses, which is well-known to be NP-complete. Let \( \phi \) be a 3-CNF formula with \( n \) variables such that every literal occurs in at most 2 clauses of \( \phi \). We construct a graph \( D \) such that \( \text{mon-sw}_{\text{inert}}(D) \) and \( \text{mon-sw}_{\text{vis}}(D) \) are bounded and \( D \) has a directed dominating set of size at most \( n \) if and only if \( \phi \) is satisfiable. In the following, we denote by \( V(\phi) \) the set of variables of \( \phi \) and by \( C(\phi) \) the set of clauses of \( \phi \). The graph \( D \) has vertex set \( \{ v, \bar{v} : v \in V(\phi) \} \cup \{ C : C \in C(\phi) \} \) and arc set \( \{ (v, \bar{v}), (\bar{v}, v) : v \in V(\phi) \} \cup \{ (l, C) : l \in C \text{ and } C \in C(\phi) \} \).

We first show that \( \phi \) is satisfiable if and only if \( D \) has a dominating set of size at most \( n \). Suppose that \( \phi \) is satisfiable and let \( \beta \) be a satisfying assignment for \( \phi \). It is straightforward to check that \( \{ v : \beta(v) = 1 \} \cup \{ \bar{v} : \beta(v) = 0 \} \) is a directed dominating set of \( D \) of size \( n \). For the reverse direction suppose that \( D \) has a directed dominating set \( S \) of size at most \( n \). Because the vertices \( v \) and \( \bar{v} \) have only themselves as incoming neighbors every directed dominating set has to contain at least 1 of \( v \) and \( \bar{v} \) for every \( v \in V(\phi) \). Because \( |S| \leq n \) the directed dominating set \( S \) has to contain exactly one of \( v \) and \( \bar{v} \) for every \( v \in V(\phi) \). We claim that the assignment \( \beta \) with \( \beta(v) = 1 \) if \( v \in S \) and \( \beta(v) = 0 \) if \( \bar{v} \in S \) is a satisfying assignment for \( \phi \). This follows from the fact that because \( S \) is a directed dominating set it holds that for every \( C \in C(C) \) there is a vertex \( x \in S \) such that \( (x, C) \in E(D) \).

It remains to show that \( \text{mon-sw}_{\text{vis}}(D) \) and \( \text{mon-sw}_{\text{inert}}(D) \) are bounded. We first show that \( \text{mon-sw}_{\text{vis}}(D) \leq 7 \) by defining a monotone winning strategy for 7 cops in the visible directed elimination game on \( D \) as follows. If the robber starts on a variable vertex \( v \) or \( \bar{v} \) for \( v \in V(\phi) \) then the cop places 2 cops on \( v \) and \( \bar{v} \). After that the robber has to be on a clause-vertex \( C \in C(\phi) \). The cop then places 7 cops on \( C \) and on the vertices in \( \{ v, \bar{v} : (v, C) \in E(D) \text{ or } (\bar{v}, C) \in E(D) \} \) and the robber has nowhere to escape. If on the other hand the robber starts on some clause vertex \( C \in C(\phi) \) the cop places 7 cops on \( C \) and on the vertices in \( \{ v, \bar{v} : (v, C) \in E(D) \text{ or } (\bar{v}, C) \in E(D) \} \). Again the robber has nowhere to escape.

The strategy \( S := \{(v_1, \bar{v}_1), \ldots, (v_n, \bar{v}_n), N_D(C_1), N_D(C_1), \ldots, N_D(C_m), N(C_m)\} \) shows that \( \text{mon-sw}_{\text{inert}}(D) \leq 4 \).

Using the same construction as in the proof of the above theorem we can show the same result for the Alternation \( c \) variant of the game.

**8 Conclusion**

In this paper we have studied a variant of the cops and robber games based on the idea of directed elimination games from [11]. We have seen that these games
yield interesting graph invariants that allow to separate the class of DAGs into “simple” and “hard” instances. This has been used in the Section 7 to show that the directed dominating set problem becomes fixed-parameter tractable on classes of bounded elimination width. While this fact followed from the observation, that any such class must have bounded degeneracy, this is no longer true for the distance-$d$-version of the problem. For this problem, we have been able to show that it becomes tractable on a version of the elimination game, where the robber can in each round change the direction once. It would be interesting to improve on this latter result and show it for the standard elimination width.

Also, so far the algorithmic results were obtained by facilitating algorithmic techniques from larger classes of graphs, such as degenerate or nowhere crownful classes. It would be interesting to develop new techniques specifically for classes of bounded elimination width as this may lead to tractability results for new problems on such classes. We leave this for future research. In any case, we believe that the elimination width idea is a promising new approach in digraph width measures with exciting new potential applications.

References


