Distance d-Domination Games

Stephan Kreutzer and Sebastian Ordyniak

Oxford University Computing Laboratory {kreutzer,ordyniak}@comlab.ox.ac.uk

Abstract. We study graph searching games where a number of cops try to capture a robber that is hiding in a system of tunnels modelled as a graph. While the current position of the robber is unknown to the cops, each cop can see a certain radius d around his position. For the case $d = 1$ these games have been studied by Fomin, Kratsch and Müller [7] under the name domination games.
We are primarily interested in questions concerning the complexity and monotonicity of these games. We show that dominating games are computationally much harder than standard graph searching games where the cops only see their own vertex and establish strong non-monotonicity results for various notions of monotonicity which arise naturally in the context of domination games. Answering a question of [7], we show that there exists graphs for which the shortest winning strategy for a minimal number of cops must necessarily be of exponential length. On the positive side, we establish tractability results for graph classes of bounded degree.

1 Introduction

Graph searching games are a form of two-player games played on graphs. A wide range of such games have been studied in the literature but they all share the common scheme that a number of cops tries to catch a robber who is hiding in the graph. The problem is to guide a party of as few cops as possible so that the robber is guaranteed to be captured regardless of his moves. In the model of graph searching games known as node searching, the cops and the robber occupy vertices of the graph. At each step of the play, the player controlling the cops can lift some of the cops from the graph and place them somewhere else. While they are in transit, the robber can move in the graph following any path from his current to his new position as long as this path does not go through a vertex occupied or "blocked" by a remaining cop (in which case the robber would be have been captured).

Variants of this game are obtained by varying the abilities of the cops, for instance, whether or not they know the current position of the robber, and by the precise definition of "blocking". The minimal number of cops needed to catch a robber on a graph yields an interesting graph invariant related to the global connectivity of the graph. See [6] for a recent survey on the subject.

Graph searching games have found a wide range of applications in Computer Science in seemingly unrelated areas: there is a strong resemblance of graph searching games to pebble games modelling sequential computation as described in [10]. In [8], graph searching games have been employed as a model for privacy in distributed systems, where the cops model eavesdroppers or intruders in networks. Furthermore, applications of graph searching games can be found in VLSI design as the game theoretical approach to important graph layout parameters providing valuable tools for the design of efficient algorithms. Of particular importance is the connection between graph searching games and well-known graph parameters such as tree-width and path-width (see e.g. [3, 4, 5]). For instance, Seymour and Thomas [12] characterised the tree-width of a graph in terms of a variant of graph searching games where the robber is visible and hides on vertices of the graph.

An important concept in the theory of graph searching games is monotonicity. Intuitively, a strategy for the cops is monotone if they can catch the robber without allowing him to revisit vertices from which he has previously been exspelled. Monotonicity has featured highly in research on graph searching games for a number of reasons. For instance, monotone strategies correspond directly to graph decompositions such as tree- or path-decompositions. Also, for many game variants, winning strategies for the robber can often be characterised by simple combinatorial structures, such as *brambles* for the case of games corresponding to tree-width, and hence provide natural and intuitive obstructions for tree-width and similar measures. However, these structures usually provide a winning strategy even against cops following a non-monotone strategy. Hence, showing for a game variant that the number of cops needed to win against a robber is always the same as the number of cops needed for a monotone strategy brings all these concepts together and establishes a smooth theory of decompositions and games in terms of min-max or duality theorems.

From an algorithmic perspective, an important property of monotone strategies is that their length is usually linearly bounded in the order of the graph, whereas non-monotone strategies can have up to exponential length, although almost no game variant actually requires such long strategies. Hence, monotone strategies often provide polynomial certificates and thereby yield NP-algorithms for deciding the number of cops needed to catch a robber.

Originally, graph searching games were introduced to model the chivvy for a robber that is hiding in a system of tunnels. While the cops do not know the current position of the robber they do have knowledge of the graph modelling the system of tunnels. In this paper we follow this idea of catching an invisible robber but consider games, which we call d*-domination games*, where the cops do not only see their current vertex but have a radius d of visibility. That is, a cop placed on a vertex v can see any other vertex within distance d of v and if this vertex is occupied by the robber then the cop can see the robber and capture him. We are primarily interested in complexity and monotonicity questions related to these games.

For the case $d = 1$ these games correspond to *domination games* as introduced by Fomin, Kratsch and Müller [7]. This variant is related to the notion of "see-catch" games studied in Computational Geometry and Robotics, for instance motivated by applications in robotics such as surveillance with a mobile robot equipped with a camera. In their paper, the authors develop the fundamental theory of domination games and establish a relationship between domination games and the size of a minimum dominating set of a graph and an interesting connection between these games and a graph parameter called *domination target number* introduced in [11]. The focus of [7] is on establishing bounds on the domination search number – the minimal number of cops that are required to guarantee capture of the robber – for various classes of graphs such as k-dimensional cubes, *asteroidal-triple free* graphs, *claw-free* graphs, and graphs with certain types of spanning trees and caterpillars. They also exhibit an example showing that domination games are non-monotone.

In this paper we study d-domination games with a focus on complexity and monotonicity. Following the initial results on monotonicity of graph searching games mentioned above, monotonicity proofs for a large number of graph searching games and also non-monotonicity proofs for some games have been obtained (see e.g. [6]). Most variants of graph searching games are either monotone or, if not, at least a bound on the difference between the number of cops needed for arbitrary or monotone strategies can be established. As it turns out, d-domination games exhibit a completely different behaviour in this respect.

Organisation and results. In Section 4, we establish very strong non-monotonicity results by exhibiting classes of graphs on which two cops can win on any graph in this class but the number of cops required for monotone winning strategies is unbounded. Hence, domination games are one of only very few types of games for which such a difference has been proved.

In [7, Problem 7], Fomin et al. raise the question whether any polynomial bound could be proved for the length of winning strategies in domination games. We give a negative answer to this question by exhibiting a class of graphs where two cops have a winning strategy but only with an exponential number of steps. To the best of our knowledge, this is the first type of graph search games for which such a lower bound has been proved.

In terms of complexity, domination games are also much harder than standard cops and robber games. In particular, we show that deciding if two cops have a (non-monotone) winning strategy is PSPACE-complete. Again, to the best of our knowledge, this is the first type of graph searching games exhibiting this worst-case complexity. This result is in sharp contrast to other variants of graph searching games on undirected graphs, which often are in polynomial time for a fixed number of cops and often even fixed-parameter tractable with the numbers of cops being the parameter. For monotone strategies we also prove that it is NP-hard to decide whether two cops have a monotone winning strategy in domination games. The complexity results are the focus of Section 5.

Finally, we establish a relation between domination games and Robber and Marshal games played on hypergraphs. Robber and Marshal games were introduced in [9] to provide a game theoretical characterisation of hypertree-width. In particular, we show that every Robber and Marshal game on a hypergraph can be translated into a domination game on an undirected graph and derive interesting consequences from this fact.

2 Preliminaries

We use standard notation from graph theory as can be found in, e.g., [5]. In particular, we write $V(G)$ for the vertex set of a graph G and $E(G)$ for its edge set. All graphs in this paper are simple and undirected and all graphs and hypergraphs are finite. Let G be a graph and $d \geq 1$. The (open) d-neighbourhood of a vertex v in G is $N_d^G(v) := \{u : 0 < \text{dist}_G(u,v) \leq d\}$, where $\text{dist}_G(u,v)$ is the distance between u and v in G.

The closed d-neighbourhood of v is $N_d^G[v] := N_d^G(v) \cup \{v\}$. If X is a set, we define $N_d^G[X] := \bigcup_{v \in X} N_d^G[v]$. For the case $d = 1$, we omit the index d and e.g. write $N^G(v)$ for $N_1^G(v)$. Also, we omit the index G whenever G is clear from the context.

The notions of tree-width and path-width were introduced by Robertson and Seymour as part of their work on graph minors. We refer to [3,5] for definitions and further information. We write $\mathrm{pw}(G)$ for the path-width of a graph G and $\mathrm{tw}(G)$ for its tree-width.

3 d-Domination Games

In this section we introduce d-*domination games* and present basic results.

A d-*domination game* on a graph G is played between two players, the cop and the robber, where the goal of the cops is to capture the robber. At each step of the play, the robber occupies a vertex of the graph and the cop player controls a finite number of cops each occupying vertices. A play starts by the robber choosing an initial position. In each step of the game, the cop either places a new cop on a vertex or removes an already placed cop from the graph. Suppose X is the set of vertices currently occupied by the cops and they want to place a new cop on vertex v. They first have to announce this to the robber. The robber can then run away, but is not allowed to run through a vertex that is in the d-neighbourhood of a vertex occupied by a cop, i.e. he can pick a new position u anywhere on the graph as long as there is a path from his current position to u that contains no vertex in $N_d[X]$.

After the robber has chosen his new position, the new cop is placed on v and the play continues. The cops win a play if they can capture the robber, i.e. if they can place a cop occupying or dominating the vertex occupied by the robber so that the robber is not able to escape. If the robber can escape forever, he wins.

d-domination games are a variant of the well-known cops and robber games used to characterise graph parameters such as tree-width or path-width (see e.g. [12]). The difference is that in a cops and robber game, a cop only occupies his current position but does not block the d-neighbourhood of this position.

We will distinguish between two variants of d-domination games, i.e. the visible and invisible variant. In the *visible* case, the cops can see the robber and can adapt their strategy accordingly. In the *invisible* case, the cops do not see the robber and hence have to search the graph independently of the robbers current position. In this case, we are essentially dealing with a one player game and in describing the game, we can discard the robber positions. In both cases, the aim of the cop player is to capture the robber using as few cops as possible. In this paper we primarily consider the invisible case and will therefore present the relevant notation and definitions in terms of the invisible domination game. We briefly comment on the visible case in Section 6.

In the invisible domination game, the cops have to capture the robber without being able to see him – and hence without being able to react to his actions. We can therefore represent any cop strategy on a graph G in the invisible d-domination game by a sequence $\mathcal{S} := (S_1, \dots, S_n)$, where, for $1 \leq i \leq n$, $S_i \subseteq V(G)$ is the cop position after step i. With any strategy $\mathcal{S} := (S_1, \dots, S_n)$ we associate the corresponding sequence $R_0, \dots, R_n$ of *robber spaces* as follows: $R_0 := V(G)$ and for all

$i > 0$, $R_i := \{v \in V(G) \setminus N_d[S_i] :$ there is $u \in R_{i-1}$ and a path from u to v in $G \setminus N_d[S_{i-1} \cap S_i]\}$, where we take $S_0 := \emptyset$. Hence, R_i is the set of vertices available to the robber after i steps of the play. Vertices in $V(G) \setminus R_i$ are called *clear at stage* i.

Definition 3.1. *Let $\mathcal{S} := (S_1, \ldots, S_n)$ be a strategy and $(R_0, \ldots, R_n)$ be the corresponding robber spaces.*

1. *$\mathcal{S}$ is a* winning strategy *if it is finite and $R_n = \emptyset$.*
2. *The* width $w(\mathcal{S})$ *of $\mathcal{S}$ is defined as $w(\mathcal{S}) := \max\{|S_i| : 1 \leq i \leq n\}$.*
3. *The* d-domination search number $ds_d(G) := \min\{w(\mathcal{S}) :$ *$\mathcal{S}$ is a winning strategy on $G\}$ of G is the minimal number of cops required to win the invisible d-domination game on G.*

Clearly, every graph of order n can be searched by n cops. Hence $ds_d(G)$ is well-defined. We next introduce a general construction that will be used frequently throughout the paper. As a first application of this we show that questions about complexity and monotonicity of d-domination games for $d > 1$ can be reduced to the corresponding questions for the case of $d = 1$.

For $k > 0$, let K_k be the k-clique, i.e. the complete graph on k vertices. Further, if X is a set, we write $K[X]$ for the complete graph with vertex set X. For each $k > 0$ and $d > 0$, we define S_k^d as the graph (up to isomorphism) obtained from K_k by subdividing each edge $2d$ times, i.e. replacing each edge by a path of length $2d + 1$. We call S_k^d a *d-subdivided k-clique*. Note that S_k^d contains more than k vertices but in the rest of the paper the vertices in the paths replacing edges will usually not play a role. We say that S is the d-subdivided clique *over* a set X if S is obtained from $K[X]$ by subdividing each edge $2d$ times. We write $S^d[X]$ for this graph and call X the *original vertices* of $S^d[X]$. As before, we omit the indices in case $d = 1$. The following lemma, whose proof is straightforward, will be used frequently in the sequel.

Lemma 3.2. *For all $k > 0$ and $d > 0$, $ds_d(S_k^d) = k$.*

For a graph G, $k > 0$ and a function $f : V(G) \to 2^{V(G)}$ we define the *subdivided k-clique graph* of G, denoted by $SC(G, k, f)$, to be the graph obtained from G by 1) replacing each vertex $v \in V(G)$ by a disjoint copy of S_k^1, denoted $SC(v)$, and 2) replacing each edge $\{u, v\} \in E(G)$ by a perfect matching between the original vertices in $SC(u)$ and the original vertices in $SC(v)$ and 3) for each $v \in V(G)$ we add a new vertex denoted $c(v)$ so that $\{c(v) : v \in V(G)\}$ induces a clique in $SC(G, k, f)$ and for each $v \in V(G)$, $SC(G, k, f)$ contains edges between $c(v)$ and all vertices in every $SC(u)$ for $u \in f(v)$.

Now it is easily seen that k cops have a winning strategy in the d-domination game on G if, and only if, k cops have a winning strategy in the 1-domination game on $SC(G, k, N_d^G[])$, where in addition they only play on the new extra vertices $c(v)$, for $v \in V(G)$. The same holds for monotone winning strategies as defined in Section 4 below. Here, $N_d^G[]$ denotes the function $f(v) := N_d^G[v]$. By setting $k := |V(G)|$ we obtain the following corollary.

Corollary 3.3. *Fix $d > 0$. There is a polynomial time algorithm which constructs for each graph G a graph G' such that for all $k > 0$, k cops win the d-domination game on G if, and only if, k cops win the 1-domination game on G'. The analogous statement holds for monotone winning strategies.*

The converse direction is also true. By subdividing each edge $2d$-times, we can construct for each graph G a graph G' so that k cops win the 1-domination game on G if, and only if, k cops win the d-domination game on G'. This construction follows essentially from [7] and also shows that the cops and robber game underlying tree-width can be reduced to the 1-domination game. It follows that all questions concerning monotonicity and complexity about d-domination games can be reduced to the case of $d = 1$. We will therefore only consider this case in the sequel. As described in the introduction, this case was already studied under the name of domination games by Fomin et al. [7]. We will therefore follow their terminology and refer to these games as domination games and write *ds*(G) for the minimal number of cops required to win the domination game on a graph G.

4 Monotonicity of Domination Games

In this section we study monotone strategies of invisible domination games. In particular, we establish strong non-monotonicity results for common notions of monotonicity – *cop-* and *robber-monotonicity* – in showing that in general more cops are needed to catch a robber with a monotone strategy than with an unrestricted strategy and that the ratio between the monotone and the non-monotone case is unbounded. We then consider a third type of monotonicity specific to domination games.

Definition 4.1. *Let $\mathcal{S} := (S_1, \dots, S_n)$ be a strategy and $(R_0, \dots, R_n)$ be the corresponding robber spaces (see Section 3).*

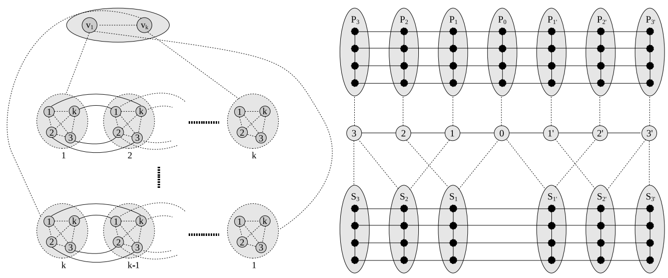
1. *$\mathcal{S}$ is* robber-monotone, *if $R_i \supseteq R_j$ for all $i < j$.*
2. *$\mathcal{S}$ is* cop-monotone *if for all $i < j < l$ and all $v \in V(G)$, if $v \in S_i \setminus S_j$ then $v \notin S_l$.*
3. *The* cop-monotone domination search number *is defined as c-ds$(G) := \min\{w(\mathcal{S}) : \mathcal{S}$ is a cop-monotone winning strategy on $G\}$. The* robber-monotone domination search number *r-ds(G) is defined analogously.*

In a non-monotone strategy, a vertex $v \in R_j \setminus R_i$, for $j > i$, is called recontaminated.

Note that, unlike cops and robber games, in domination games cop-monotone strategies might not be robber-monotone and vice versa. In [7], Stefan Dobrev exhibited an example where three cops can win the domination game but four cops are needed to search the graph using a monotone strategy. We now strengthen this result considerably by showing that the ratio between the (robber- or cop-) monotone and the non-monotone search numbers is unbounded.

Lemma 4.2. *For every $k > 2$, there is a graph G_k such that ds$(G_k) = 2$ but r-ds$(G_k) =$ c-ds$(G_k) = k$.*

Proof. For $k \in \mathbb{N}$ we define G_k as follows. Let $U := \{u_1, \dots, u_k\}$ be a set of size k. For all permutations ρ of $(1, \dots, k)$ and all $1 \leq i \leq k$, let P_i^ρ be a subdivided clique on k vertices and let H_ρ be the graph obtained from the disjoint union $\dot{\bigcup}_i P_i^\rho$ of these subdivided cliques by adding edges forming a perfect matching of the original vertices in P_i^ρ and P_{i+1}^ρ, for $1 \leq i < k$. Then G_k is defined as $K[X] \dot{\cup} \dot{\bigcup}_\rho H_\rho$ augmented by edges $\{\{v_i, v\} : v \in P^{\rho}_{\rho(i)}, 1 \leq i \leq k$ and ρ is a permutation of $(1, \ldots, k)\}$. The construction is illustrated in Figure 1 $a)$. Here, dashed lines represent edges from a vertex u_i to all vertices in a subdivided clique whereas solid lines represent actual edges.



a) Graph G_k from Lemma 4.2. b) Graph G_k from Lemma 4.3.

Fig. 1. Examples for non-monotonicity in domination games.

It is easily seen that two cops can search G_k as follows: for each permutation ρ of $(1, \ldots, k)$ they play $\mathcal{S}_\rho := (\{u_{\rho(1)}, u_{\rho(2)}\}, \{u_\rho(2), u_{\rho(3)}\}, \ldots, \{u_{\rho(k-1)}, u_{\rho(k)}\})$, i.e. they search the "path" P_ρ by going through $u_1, \ldots, u_k$ using the ordering given by ρ. As the only connection between H_ρ and $H_{\rho'}$ is through the vertices in U and these form a clique, they can search the H_ρ independently.

It remains to show that $k - 1$ cops do not have a cop-monotone or a robber-monotone strategy on G_k. We can assume that the cops are only playing on the vertices in $u_1, \cdots, u_k$ as otherwise they need at least k cops to clear a subdivided k-clique.

Suppose the cops start by occupying all but one vertex u_i in U. Then in each H_ρ, the clique $P^{\rho}_{\rho(i)}$ is still contaminated. Furthermore, in the next step the cops have to remove a cop from a vertex u_j. But then, there is a permutation ρ such that $\rho(i)$ and $\rho(j)$ are consecutive numbers and thus in H_ρ the subdivided clique $P^{\rho}_{\rho(j)}$ becomes recontaminated. This shows that the strategy is not robber-monotone. As $P^{\rho}_{\rho(j)}$ can only be cleared again by playing on v_j the strategy for the cops can not be cop-monotone. This concludes the proof.

Considering again the example above exhibiting non-monotone strategies for the cops, the main source for non-monotonicity appears to be that while clearing some parts of the graph, the cops accidentally and unintentionally clear other parts of the graph also – which later on they have to allow to be recontaminated. For instance, in the example above, while clearing a sub-graph H_ρ they also clear parts of other sub-graphs $H_{\rho'}$ but in the wrong order. If we gave the cops the power to choose which vertices in the neighbourhood of a cop they really want to dominate, then they could easily search the graphs G_k with a robber- and cop-monotone strategy. We call this *selective monotonicity*. It seems conceivable, thus, that such *selective* strategies are always sufficient, i.e. whenever k cops can win in any form, they can do so with a selective monotone strategy. Such a result would be extremely interesting as it would imply a linear upper bound for the length of minimal winning strategies for the cop player. This hope is dashed, though, by the following theorem.

Theorem 4.3. *There exists a graph G with ds$(G) = 2$ but 3 cops are needed for any selective monotone winning strategy.*

Proof. The graph G is shown in Figure 1 b). Here, solid lines represent actual edges whereas a dashed line such as between 3 and S_3 indicates that there is an edge between 3 and every vertex in S_3.

Now, *ds*$(G) = 2$ as witnessed by the following two strategies: $\mathcal{S}_1 := \big(\{3,2\},\{2,1\},\{1,0\},\{0,1'\},\{1',2'\},\{2',3'\}\big)$ and $\mathcal{S}_2 := \big(\{3',2'\},\{2',1'\},\{1',0\},\{0,1\},\{1,2\},\{2,3\}\big)$. Note that both strategies are not robber monotone. For instance, in $\mathcal{S}_1$ the vertices in S_1 are recontaminated in the step from $\{2,1\}$ to $\{1,0\}$ and similarly in the symmetric strategy $\mathcal{S}_2$. Further, observe that in order for these strategies to work, at each step all neighbours of every vertex occupied by a cop need to be dominated. Hence, none of the two strategies can be turned into a selective monotone strategy.

We claim that there is no selective monotone strategy with only two cops. For the sake of contradiction let $\mathcal{S}$ be a selective monotone winning strategy with two cops using a minimal number of steps. We first show that $\mathcal{S}$ cannot use any vertex other than those in $X := \{3,2,1,0,1',2',3'\}$. For, if $v \in S_i$ or $v \in P_i$ is occupied by a cop then at the first step where this cop is lifted from v, v will be recontaminated unless it is dominated by the other cop. Hence, placing a cop on v either can be avoided, as v is dominated anyway, or it leads to non-monotonicity.

Thus, a selective monotone strategy with two cops essentially searches the path $3,2,1,0,1',2',3'$. However, it is easily seen that a path of length 7 can be searched in only two ways by two cops using a monotone strategy: left to right or right to left. If follows that the only possible strategies are $\mathcal{S}_1$ or $\mathcal{S}_2$ and neither is selective monotone. This yields the contradiction.

As argued above, an important aspect of monotonicity for a variant of graph searching games is that in this way a bound on the maximal number of steps in a strategy is obtained. As domination games are strongly non-monotone, no such bound can be achieved using this approach. In Corollary 5.3 below we show that there exist graphs such that the number of steps needed by a strategy in the domination game is exponential in the size of the graph and thus cannot be bounded bounded by a polynomial.

5 Complexity of Domination Games

In this section we study the complexity of deciding whether k cops have a (monotone) winning strategy in the domination game on a graph G. We measure the complexity of this problem in different ways – classically and in the context of parametrised complexity. Let DOMINATION SEARCH be the problem of deciding for a given graph G and $k \in \mathbb{N}$ whether k cops have a winning strategy on G. In [7], Fomin et al. study this problem and show that it is NP-hard.

Theorem 5.1 ([7]). DOMINATION SEARCH *is* NP-*hard.*

No upper bound for the complexity of the problem was given. We settle this problem by giving precise complexity bounds for DOMINATION SEARCH.

Theorem 5.2. DOMINATION SEARCH *is* PSPACE*-complete. More precisely, we show that even deciding whether two cops have a winning strategy on a graph is* PSPACE*-complete.*

In [7, Problem 7], Fomin et al. raise the question whether for every graph G there is a winning strategy of length $\mathcal{O}(n)$ using $ds(G)$ cops in the invisible domination search game. As a consequence of the proof of the previous theorem we answer this question negatively by showing that there exist graphs on which the number of steps needed by a strategy in the domination game is at least exponential in the size of the graph and thus can not be bounded bounded by a polynomial. Clearly, exponential length of strategies is also the worst possible.

Corollary 5.3. *There exists a family $\mathcal{C}$ of graphs such that two cops have a winning strategy in the invisible domination game on each $G \in \mathcal{C}$ but any such strategy is at least of exponential length, i.e. there is no polynomial $p(n)$ so that the length of these strategies is bounded by $p(|G|)$.*

We now consider the problem to decide for a given graph G whether k cops have a monotone winning strategy in the invisible domination game, where we consider cop- and selective-monotonicity. Clearly, as the length of monotone strategies is polynomially bounded in the size of the graph, these problems are necessarily in NP. We again give tight complexity bounds by showing that even deciding whether two (or three, respectively) cops have monotone winning strategies is NP-hard.

Theorem 5.4. *Let G be a graph. Deciding whether two cops have a cop-monotone winning strategy in the domination search game on G is NP-complete.*

Theorem 5.5. *Let G be a graph. Deciding whether three cops have a selective monotone winning strategy in the domination game on G is NP-complete.*

We do not know corresponding results for robber-monotone strategies and leave this as an open problem.

The previous results settle the classical complexity of the domination game problem. We now study the parametrised complexity of this problem. The parametrised domination search problem p-DOMINATION SEARCH is defined as the problem, given a graph G and $k \in \mathbb{N}$ as input, to decide if k cops have a winning strategy in the invisible domination game on G. We take k as the parameter. The problem is in the parametrised complexity class XPif it can be solved in time $|G|^{f(k)}$ for some computable function $f : \mathbb{N} \to \mathbb{N}$. It is *fixed-parameter tractable*, or in FPT, if it can be solved in time $f(k) \cdot |G|^c$, for some $c \in \mathbb{N}$ and computable $f : \mathbb{N} \to \mathbb{N}$. The following is an immediate consequence of Theorem 5.2, 5.4 and 5.5.

Corollary 5.6. p-DOMINATION SEARCH *is not in XP. This holds true even for the cop- or selective monotone version of the problem.*

The previous results establish fixed-parameter intractability for domination games. Hence, domination games are considerably more complex than standard cops and robber games, which are NP-complete and fixed-parameter tractable. The latter follows from the parametrised tractability of tree-width and path-width and the monotonicity of the games.

We now turn to special cases where tractability can be obtained. A natural choice of graph classes where the problem might be easier are classes of bounded tree- or path-width. One is tempted to think that fixed-parameter tractability of domination search on classes $\mathcal{C}$ of graphs of tree-width at most d could be established along the following lines: given $G \in \mathcal{C}$ and $k \in \mathbb{N}$, we first compute a tree-decomposition of G of width d and then use dynamic programming to decide whether there is a winning strategy of width k. This is the approach taken to show that the analogous questions for cops and robber games (visible and invisible) can be solved by linear time parametrised algorithms. Typically, one proceeds bottom-up along the tree-decomposition and for each node in the decomposition tree one computes a constant size data structure containing information about the sub-graph induced by the vertices in the sub-tree rooted at this node. For domination games, however, this approach fails as a vertex in a bag can be dominated by vertices not contained in this bag. The ways in which this happens can be rather complex and hence a constant size data structure seems difficult to obtain. It is still possible, though, that domination search is fpt on classes of bounded tree-width and we leave this for future work.

We are, however, able to obtain parametrised algorithms for classes of graphs of bounded degree (recall that the problem is already NP-hard on the class of graphs of degree at most 3).

Lemma 5.7. *For $d > 0$ let $\mathcal{C}_d$ be the class of graphs of maximum degree at most d. Then the problem, given $G \in \mathcal{C}_d$ and $k \in \mathbb{N}$, to decide whether k cops have a cop-monotone winning strategy on G is fixed-parameter tractable with parameter $d + k$.*

Furthermore, if k cops have a winning strategy on any $G \in \mathcal{C}_d$, then at most $dk + 1$ cops have a cop- and a selective-monotone winning strategy.

6 Games on Hypergraphs and Visible Robbers

In this section we briefly explore the relation between domination games and Robber and Marshal games on hypergraphs and comment on domination games with a visible robber.

Robber and Marshal games, with a visible robber, have been introduced in [9] as a game-theoretical approach to hypertree-width and have, since then, been studied intensively. Essentially, a Robber and Marshal game is a Cops and Robber game on a hypergraph where the robber occupies a vertex whereas each marshal (= cop) occupies a hyperedge and blocks all vertices contained in it.

We will show next that every hypergraph game can be translated into a domination game – in the visible and the invisible case. There is a small difference between the Robber and Marshal game we use here and the original robber and marshal game in [9]. In the original game the marshals *slide* along edges in the sense that if a marshal moves from hyperedge e to e' then the vertices in $e \cap e'$ remain blocked (an equivalent notion for domination games could easily be defined). Here, we consider the variant of Robber and Marshal games where only the vertices in edges on which a marshal remains are blocked. It is easy to see that both variants are within a constant factor of each other.

Lemma 6.1. *Let H be a hypergraph and $k \geq 1$ be an integer. Then there exists a graph H^{dom}_{k+1}, such that k marshalls have a (marshal-/robber-monotone) winning strategy in the (visible) robber and marshals game on H, if and only if, k cops have a (cop-/robber-monotone) winning strategy in the (visible) domination game on H^{dom}_{k+1} and H^{dom}_{k+1} can be constructed from H in polynomial time.*

The lemma allows us to translate Robber and Marshal games to domination games. It follows immediately from Lemma 6.2 below that there is no translation in the converse direction.

So far, we have primarily considered domination games with an invisible robber. Here, we briefly summarise our knowledge of the visible case. Clearly, notions such as monotonicity and the domination search number translate easily.

In [1], Adler showed that the visible robber and marshall game mentioned above is not robber-monotone. Together with Lemma 6.1, this implies that the visible domination game is also not robber-monotone. However, the robber-monotone and non-monotone variant of the visible robber and marshall Game are within a constant factor of each other (see [2]). We show next that no such bound can be obtained for domination games.

Lemma 6.2. *For every $k > 2$, there is a graph G_k such that 2 cops have a non-monotone but k cops are needed for a robber-monotone winning strategy in the visible domination game on G_k.*

Finally, we consider the complexity of visible domination games. In terms of classical complexity, we can show the following.

Theorem 6.3. *Let G be a graph. Deciding whether three cops have a selective monotone winning strategy in the visible domination game on G is NP-complete.*

It is easily seen that all visible game variants except for the selective monotone variant are in XP, as the current cop and robber position completely determine the current state of the play and there are only $n^{\mathcal{O}(k)}$ such positions. We show next that the problem is not in FPTunless FPT=W[2].

As observed in [7], domination search is closely related to dominating sets in graphs. A *dominating set* of a graph G is a set X such that for all $v \in V(G)$ either $v \in X$ or there is a $u \in X$ such that $\{u, v\} \in E(G)$. The *domination number* of G, denoted by $\gamma(G)$, is the minimal size of a dominating set of G.

Lemma 6.4 ([7]). *Let G be a graph and H be the graph obtained from G by connecting every pair of non-adjacent vertices in G by a path of length three. Then $\gamma(G) \leq$ $\mathrm{ds}(H) \leq \gamma(G) + 1$.*

We establish a similar but exact correspondence using a slightly different construction.

Theorem 6.5. *For all graphs G, there exists a graph G' such that $\gamma(G) + 1 =$ ds(G') and G' is constructable in polynomial time.*

The theorem immediately gives a parametrised reduction from the dominating set problem, parametrised by the size of the solution, to the domination search problem, parametrised by the number k of cops. The following result follows from the $\mathrm{W}[2]$-hardness of the dominating set problem, where $\mathrm{W}[2]$is a parametrised complexity class strongly believed to be different from FPT.

Theorem 6.6. *The problem p-*DOMINATION SEARCH*: "given a graph G and $k \in \mathbb{N}$, with parameter k, decide whether k cops have a winning strategy in the (in-)visible domination game on G" is* $\mathrm{W}[2]$*-hard.*

However, Lemma 5.7 also applies to the visible case and thus calculating the visible domination search number for graphs of bounded degree is fixed-parameter tractable.

Acknowledgements. We are grateful to Fedor Fomin for bringing domination games to our attention and thereby stimulating the research reported in this paper and to Fedor Fomin, Paul Hunter and Paul Dorbec for valuable discussions on the subject.

References

1. I. Adler. Marshals, monotone marshals, and hypertree-width. *JGT*, 47(4):275–296, 2004.
2. I. Adler, G. Gottlob, and M. Grohe. Hypertree width and related hypergraph invariants. *Journal of Combinatorics*, 28(8):2167–2181, 2007.
3. H. Bodlaender. A partial k-aboretum of graphs with bounded tree-width. *TCS*, 209:1 – 45, 1998.
4. H. L. Bodlaender. Treewidth: Algorithmic techniques and results. In *Proc. of Mathematical Foundations of Computer Science (MFCS)*, volume 1295 of *LNCS*, pages 19–36, 1997.
5. R. Diestel. *Graph Theory*. Springer-Verlag, 3rd edition, 2005.
6. F. Fedor and D.M. Thilikos. An annotated bibliography on guaranteed graph searching. *TCS*, 399(3):236–245, 2008.
7. F. Fomin, D. Kratsch, and H. Müller. On the domination search number. *Discrete Applied Mathematics*, 127(3):565–580, 2003.
8. M. K. Franklin, Z. Galil, and M. Yung. Eavesdropping games: a graph-theoretic approach to privacy in distributed systems. *Journal of the ACM*, 47(2):225–243, 2000.
9. G. Gottlob, N. Leone, and F. Scarcello. Robbers, marshals, and guards: game theoretic and logical characterizations of hypertree width. *Journal of Computer and Systems Science*, 66(4):775–808, 2003.
10. L. M. Kirousis and C. H. Papadimitriou. Searching and pebbling. *TCS*, 47(3):205–218, 1986.
11. T. Kloks, D. Kratsch, and H. Müller. On the structure of graphs with bounded asteroidal number. *Graphs and Combinatorics*, 17(2):295–306, 2001.
12. P. D. Seymour and R. Thomas. Graph searching and a min-max theorem for tree-width. *Journal of Combinatorial Theory, Series B*, 58(1):22–33, 1993.