Theoretical Computer Science 794 (2019) 47–58

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Cops, a fast robber and defensive domination on interval graphs [☆]

Dariusz Dereniowski [b,*,2], Tomáš Gavenčiak [a,*], Jan Kratochvíl [a,*,1]

[a] Department of Applied Mathematics, Charles University, Prague, Czech Republic
[b] Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, Gdańsk, Poland

ARTICLE INFO

Article history:
Received 20 September 2017
Received in revised form 3 August 2018
Accepted 26 September 2018
Available online 6 November 2018

Keywords:
Cops and robber
Pursuit-evasion
Combinatorial game
Interval graph
Defensive domination

ABSTRACT

The game of Cops and ∞-fast Robber is played by two players, one controlling c cops, the other one robber. The players alternate in turns: all the cops move at once to distance at most one each, the robber moves along any cop-free path. Cops win by sharing a vertex with the robber, the robber by avoiding capture indefinitely.

The game was proposed with bounded robber speed by Fomin et al. in "Pursuing a fast robber on a graph", generalizing a well-known game of Cops and Robber which has robber speed 1. We answer their open question about the computational complexity of the game on interval graphs with ∞-fast robber, showing it to be polynomially decidable.

We also generalize the concept of k-defensive domination introduced by Farley and Proskurowski in "Defensive Domination" to $\mathcal{A}$*-defensive domination* and use it as a main tool in our proof. The generalization allows specifying arbitrary attacks and limiting the number of defenders of each vertex. While this problem is NP-complete even for split graphs, we show that $\mathcal{A}$-defensive domination is decidable in polynomial time on interval graphs.

1. Introduction

In this article, we explore and extend two independent topics on the class of interval graphs, namely a version of Cops and Robber game with ∞-fast robber and the concept of defensive domination, extending it to arbitrary attack vectors. While we introduce the defensive domination algorithms as a tool to solve the game in polynomial time, we believe that the problem and algorithm are of independent interest. For that reason, the domination is treated independently in Section 3. This is a significantly revised and extended version of a paper presented at TAMC 2011 [12].

The recent development in the area of combinatorial "Cops and Robber" games (also called pursuit-evasion games) includes results on games with various characteristics of the players. The characteristics include, e.g., visibility [7,8,13], speed [6,10] or radius of capture [4]. See also the recent book by Bonato and Nowakowski [5]. In this work we are interested in the Cops and s-fast Robber game that is a generalization of the original Cops and Robber game introduced by Nowakowski and Winkler [20] and by Quilliot [21], allowing the Robber to make up to s steps instead of 1 in one turn.

☆ This is a significantly revised and extended version of a paper presented at TAMC 2011 [12].

* Corresponding authors.
E-mail addresses: deren@eti.pg.gda.pl (D. Dereniowski), gavento@kam.mff.cuni.cz (T. Gavenčiak), honza@kam.mff.cuni.cz (J. Kratochvíl).

1 Partially supported by CE-ITI project GAČR P202/12/606.
2 Partially supported by National Science Centre (Poland) grant number 2015/17/B/ST6/01887.

https://doi.org/10.1016/j.tcs.2018.09.031

The concept of k-defensive domination has been introduced by Farley and Proskurowski [9] as a generalization of the well-known dominating set problem. They see the dominating set as a "defense unit" placement scheme where every single unit may actively defend an attack on one vertex in distance at most 1. The k-defensive domination asks for defense unit placement that can counter any k-vertex attack. A formal definition is given below. We take a slightly different approach and propose the problem with the possible attack sets as a part of the input.

1.1. Cops and s-fast Robber game

We start with a formal definition of the game:

The **Cops and s-fast Robber game**, where $s \geq 1$ and possibly $s = \infty$, is defined as follows: The game is played by k cops controlled by one player and one robber controlled by the second player on a given simple undirected graph G. The cops and the robber are positioned on vertices of G at all times; several cops may share a vertex. Both players have a complete information about G and the game state. At the start of the game, the cops choose starting vertices, then the robber chooses a starting vertex. If there is no vertex for the robber, the cops win immediately.

A *step* is a move of a cop to distance at most one (staying on the same vertex is allowed). One *turn* then consists of a *cop-move*, where every cop makes a step, followed by a *robber-move*, where the robber moves along any cop-free path of length at most s. The robber may never pass through a vertex occupied by a cop. In the case $s = \infty$ the robber may move to an arbitrary vertex of his present component of $G \setminus C$ where C is the (multi)set of cops' position.

Should a cop be at or move to robber's vertex, we say that the robber is captured and the cops immediately win. The robber wins by avoiding the capture indefinitely. The minimum number of cops sufficient to capture the robber in a graph G is denoted by $c_s(G)$. This game is equivalent to the original Cops and Robber game when $s = 1$.

In their paper "Pursuing a fast robber on a graph" [10], Fomin et al. propose the complexity of the Cops and ∞-fast Robber game on interval graphs as an open question. We show that the problem is polynomially decidable in interval graphs. The proof is constructive and shows how to decompose cops' moves of a general winning strategy into certain basic move sequences, thus getting a simpler equivalent game that can be decided polynomially.

Fomin et al. examine several complexity aspects of Cops and s-fast Robber games and show that, for each $s \geq 2$, the problem of computing $c_s(G)$ is NP-hard (and even W[2]-hard in the version parametrized by c_s) even if G is a chordal graph, or even a split graph. On the other hand, $c_s(G)$ can be computed in polynomial time for an interval graph and every fixed $s < \infty$. The hardness results of Fomin et al. easily extend to the game of ∞-fast robber, but the polynomiality proof for interval graphs does not and our proof takes a different approach.

This game has been further studied by Mehrabian [15,18] who gave a 3-approximation polynomial-time algorithm for interval graphs, but the complexity status of the exact problem remained open. In [17], a characterization of graphs with $c_\infty(G) = 1$ is given. For some bounds on $c_s(G)$ see [2,11,16]. For other works on the version of the game with different speeds see, e.g., [6,19]

Our first result is summarized in the following theorem:

Theorem 1. *There exists a polynomial-time algorithm that, given an n-vertex interval graph G, computes $c_\infty(G)$ and also a winning strategy for the cops that captures the robber in $O(n^3)$ turns.*

1.2. $\mathcal{A}$-defensive domination

As a main tool in our analysis we introduce the problem of **$\mathcal{A}$-defensive domination** which we believe to be of independent interest from the game and its analysis. The complexity of the problem is closely examined in Section 3, here introduce the problem and our results, and compare it to the original problem of Farley and Proskurowski.

The notion of defensive domination has been introduced by Farley and Proskurowski [9] in the following way: The input is a graph G and an integer k. A simple set $D \subseteq V_G$ is said to be *k-defensive dominating* if for each set $\{a_1, \dots, a_k\} \subseteq V_G$ of k distinct vertices of G (called a *k-attack*) there exists a set $\{d_1, \dots, d_k\} \subseteq D$ of k distinct vertices of D (called an *assignment of defenders*) such that for each $i \in \{1, \dots, k\}$ we have $d_i \in N[a_i]$. The goal is to find a k-defensive dominating set D of minimum size.

Farley and Proskurowski examine the problem and show that while the problem is generally NP-hard (as it generalizes the problem of smallest dominating set even for $k = 1$), there is a polynomial-time algorithm for any k on trees.

We use an equivalent definition of defensive domination. An *attack* is a (usually given) multiset A of vertices. We say that a vertex multiset D (called defender placement) *defends* A when there is a map $f : A \to D$ which is injective (as a multiset map) and when $f(a) = d$, then $d \in N[a]$. Such f is called a *defense* or a *defending mapping*.

We consider multisets to provide a natural but strong generalization of the problem as well as to address a situation with multiple cops on a vertex in the game analysis. See section Preliminaries for notes on multisets and maps.

$\mathcal{A}$-DEFENSIVE DOMINATION

Input: Graph G, a family of vertex multisets $\mathcal{A}$ (the attacks) and vertex multisets $D_{min} \subseteq D_{max}$.

Output: A smallest multiset of vertices D defending every attack $A \in \mathcal{A}$ such that $D_{min} \subseteq D \subseteq D_{max}$, or information that no such D exists.

Here D_{min} specifies pre-placed defenders, D_{max} of the defender capacity of a vertex. The parameters D_{min} and D_{max} can be considered optional with default values $D_{min} = \emptyset$ and $D_{max} = t * V$ (t copies of V) where $t = \max A \in \mathcal{A}|A|$. Note that setting $D_{max} = V$ would force the defender multiset to be a simple set.

Our statement of the problem differs from the one in [9] in two ways. First, we may allow more defenders per vertex. This can be forbidden by setting $D_{max} = V$ as above. However, in this article we only show an algorithm for unbounded capacities D_{max}. Note that only considering unbounded capacities does not interfere with use in the game part of the paper and allows us use simpler the algorithm and technical arguments. See Conclusion for a discussion of possible extensions.

Second, and more importantly, in our case the collection $\mathcal{A}$ is a part of the input and may contain multisets of any size, as opposed containing all the subsets of the vertices of G of given size k. Specifying $\mathcal{A}$ brings more flexibility but having all the sets $\binom{V}{k}$ as part of the input may blow it up substantially even for medium values of k.

Our main result on $\mathcal{A}$-defensive domination is the following:

Theorem 2. *There is an algorithm that solves $\mathcal{A}$-DEFENSIVE DOMINATION on interval graphs in time polynomial in the input size for any D_{min} and unbounded capacities D_{max}.*

2. Preliminaries

In this paper, we use standard graph- and game-theoretic notation. For introduction to these areas, we recommend the books "Modern Graph Theory" [3] and "Lessons in Play: An Introduction to Combinatorial Game Theory" [1].

Throughout the paper G denotes the input graph, V its vertex set and E its undirected edge set. We use $N_G[v]$ and $N_G[X]$ to denote the *closed neighborhood* of a vertex v or a subset of vertices X (including v, resp. X), and $N_G(v)$ to denote the open neighborhood of v (not containing v). We say that a set A of vertices *dominates* a vertex set B when every $v \in B$ belongs to A or has a neighbor in A, that is, $B \subseteq N_G[A]$. We drop the subscript when the graph is clear from the context. For any set A of vertices of G, we denote by $G[A]$ the subgraph of G *induced by* A, that is, the subgraph with the vertex set A and with an edge between each pair of vertices of A that are adjacent in G.

In the problem of $\mathcal{A}$-defensive domination it is natural to consider multiple attackers on a single vertex as well as multiple defenders, and this is also the case with multiple cops per vertex in the game. Therefore in our treatment all the attacker, defender and cop sets are multisets, and we point out this where it is most important. Namely, a function (map) from a multiset maps every element x to a multiset with the size same as the multiplicity of x. Similarly, for a map to be injective means $|f^{-1}(x)|$ is at most the multiplicity of x. Also, $X \subseteq Y$ takes multiplicity into account and we naturally talk about partial multiset functions, bijections e.t.c.

We briefly mention some of the less well-known graph classes and their properties:

A graph G is *chordal* (also *triangulated*) if there are no induced cycles of length at least 4 in G. A graph is a *split graph* if its vertices can be partitioned into two sets I and K such that I is an independent set and $G[K]$ is a complete subgraph. Every split graph is also chordal.

A graph is an *interval graph* if it can be realized as the intersection graph of a family of intervals on a real line. For a family of intervals $\mathcal{I}$, the associated intersection graph $\mathcal{G}(\mathcal{I})$ has one vertex for each of the intervals and an edge between the vertices corresponding to intervals I_1 and I_2 from $\mathcal{I}$ if and only if $I_1 \cap I_2 \neq \emptyset$. Note that every interval graph is chordal. In the following, we may assume without loss of generality that the intervals are open and their endpoints are distinct integers $\{1 \dots 2|V|\}$, fixing a representation for every graph. Note that such an interval representation can be reconstructed from G in linear time, as shown by Korte and Möhring [14].

For each vertex v of G, let $\overline{v}$ be the interval representing v. For any subset X of vertices of an interval graph G, denote by $\overline{X}$ the union of intervals representing the vertices in X, $\overline{X} = \bigcup_{v \in X} \overline{v}$. Note that for a connected interval graph G and any $X \subseteq V_G$ the subgraph $G[X]$ is connected if and only if $\overline{X}$ is an interval.

For any integer i, let $V[i]$ be the set of vertices v of G such that $i \in \overline{v}$. Let (i, j) denote the open interval from i to j, $(i, j) = \emptyset$ if $i \geq j$. Then let $V(i, j) = \{v \in V \mid \overline{v} \subseteq (i, j)\}$. Note that $V(i, j) \cap V[i] = \emptyset$ for every i and j. If $I = (i, j)$ is an interval, then we also write for brevity $V(I)$ in place of $V(i, j)$.

Given an interval I, denote by $\mathrm{L}(I)$ and $\mathrm{R}(I)$ the left and right endpoints of I, respectively, i.e., $\mathrm{L}(I) = \inf(I)$ and $\mathrm{R}(I) = \sup(I)$.

The intervals of a representation are naturally ordered in two ways – by their left and right endpoints. We use these to define two orders on V_G. Let $u <_R v$ if and only if $\mathrm{R}(\overline{u}) < \mathrm{R}(\overline{v})$. Similarly, $u <_L v$ if and only if $\mathrm{L}(\overline{u}) < \mathrm{L}(\overline{v})$. Note that these orders are linear thanks to the assumption that all endpoints are different.

In the algorithmic sections, $<_R$ is the commonly and sometimes implicitly used interval order while $<_L$ usually plays an auxiliary role. This is due to our choice of sweeping the graph representation left-to-right. In particular we use the following properties:

Lemma 3. *If we have $a <_R b <_R c$ and $ac \in E$, then also $bc \in E$. Similarly, if $a <_L b <_L c$ and $ac \in E$, then also $ab \in E$.*

Proof. Since $ac \in E$ we have $\mathrm{L}(\overline{c}) < \mathrm{R}(\overline{a})$ and therefore $\mathrm{R}(\overline{b}) \in \overline{c}$. The proof of the second statement is symmetric. □

The integers $1, \ldots, 2|V|$ are also called *cutpoints*, as every $V(i)$ is a vertex cut in the interval graph between the vertex sets $V(-\infty, i)$ and $V(i, \infty)$. Unlike the usual definition of a cut, here either may be empty.

3. $\mathcal{A}$-defensive domination on interval graphs

In this section we closely examine the problem of $\mathcal{A}$-*defensive domination* and prove the results stated in Section 1.2. Note that the entire section and all the proofs are completely independent from the game and Section 4.

We build on the notation from Section 1.2. Additionally, with D a multiset of defenders let a *partial defense* (against an attack A) be a function from $A' \subseteq A$ to D mapping every vertex to distance at most 1, similarly to defense defined above. Two partial defenses f_1 and f_2 *agree on a set* $X \subseteq A$ if $f(x) = f'(x)$ for each $x \in X$.

Before the algorithm, we introduce a property of a defense we use through the algorithm. For a given attack A and defenders D, order the attackers $a_1 <_R \ldots <_R a_k$. A (partial) defense f is called *leftmost (partial) defense* if the defender $f(a_i)$ assigned to attacker a_i is $<_R$-leftmost among the neighbors of a_i from $D \setminus \{f(a_1), \ldots f(a_{i-1})\}$. Such (partial) defense is called *maximal* if f assigns a defender to a_i whenever $N(a_i) \cap (D \setminus \{f(a_1), \ldots f(a_{i-1})\}) \neq \emptyset$.

Note that for given attack and defense, a unique maximal leftmost (partial) defense is computed by the following straightforward greedy algorithm which closely follows the definition of a leftmost (partial) defense:

procedure LeftmostDefense(G, A, D)
  Input: Represented interval (multi)graph G
  Input: Multisets $A \subseteq V_G$, $D \subseteq V_G$
  Output: Maximal leftmost (partial) defense $f : A' \to D$, $A' \subseteq A$
  $A' \leftarrow \emptyset$
  Order vertices of A as $a_1 \leq_R a_2 \leq_R \ldots \leq_R a_k$
  for all $i \in \{1, \ldots k\}$ **do**
    $D_i \leftarrow N(a_i) \cap (D \setminus f[A'])$
    if $D_i \neq \emptyset$ **then**
      $A' \leftarrow A' \cup \{a_i\}$
      $f(a_i) \leftarrow \min_{<_R} D_i$
    end if
  end for
  return f
end procedure

The procedure obviously returns a valid partial defense map, runs in polynomial time and is easily implementable in time $O(k|V|)$.

Whenever there is no full defense for A and D, the $<_R$-leftmost undefended attacker is called the *leftmost greedily undefended attacker*. Note that f is injective (w.r.t. D) and so the partial function $f^{-1} : D \to A$ is well-defined and injective whenever f is a full map. We can also additionally assume that f^{-1} is monotone$_R$ on every group of assigned defenders on a single vertex.

This computed defense also has the following useful properties.

Lemma 4. *The partial function f returned by* LeftmostDefense *satisfies: Whenever $a_i <_R a_j$ and $f(a_j) \in N[a_i]$, then $f(a_i) \leq_R f(a_j)$.*

Proof. Assume $f(a_i) >_R f(a_j)$. But in that case, a_i would get assigned $f(a_j)$ as a left-most available defender, which is a contradiction. □

Lemma 5. *Whenever there is a defensive map f from A to D on an interval graph G,* LeftmostDefense *returns a valid full defense map.*

Proof. Let f_{ALG} be as returned by LeftmostDefense(G, A, D) and among all defense maps, take $f : A \to D$ such that f and f_{ALG} agree on the longest prefix $\{a_1 \ldots a_{i-1}\}$. Now either $f_{\text{ALG}} = f$ and we are done, or we have either $f_{\text{ALG}}(a_i)$ undefined or $f(a_i) \neq f_{\text{ALG}}(a_i)$.

Note that we have $f(a_i) \in D_i$ (with D_i as in the algorithm) since f and f_{ALG} agree on $\{a_1 \ldots a_{i-1}\}$. This immediately rules out the possibility of undefined $f_{\text{ALG}}(a_i)$. Additionally, this shows that $f_{\text{ALG}}(a_i) \leq_R f(a_i)$, since $f_{\text{ALG}}(a_i)$ is minimal from D_i.

The last possibility to examine is $f_{\text{ALG}}(a_i) <_R f(a_i)$. In this case we show that there is a full defense map $f' : A \to D$ that agrees with f_{ALG} on a longer prefix, a contradiction with the choice of f. Let $d_i^{\text{ALG}} = f_{\text{ALG}}(a_i)$ and $d_i = f(a_i)$. Moreover let $a_j = f^{-1}(d_i^{\text{ALG}})$ if defined, in which case also note that necessarily $j > i$. See Fig. 1 for an illustration.

We now construct the new full defense map: Set f' to be f except for $f'(a_i) = d_i^{\text{ALG}}$, and $f'(a_j) = d_i$ when a_j is defined. In case a_j is undefined, validity of f' is straightforward. When a_j is defined, we only need to show $a_j \in N[d_i]$: when $a_i <_R a_j \leq_R d_i$ observe that $a_i \in N[d_i]$, when $d_i^{\text{ALG}} <_R d_i \leq_R a_j$ observe that $a_j \in N[d_i^{\text{ALG}}]$. Therefore f' is a full defense map. □

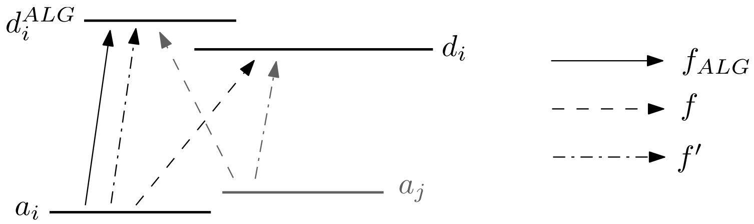

Fig. 1. Illustration of the update step and map preservation in Lemma 5.

Additionally, when two defenses differ at some point, the algorithm finds defenses that agree up to that point:

Lemma 6. *Let $D = \{d_1 \le_R \dots\}$ and $D' = \{d'_1 \le_R \dots\}$ be defenses against an attack $A = \{a_1 \le_R \dots\}$ and let f and f' be the defense maps computed by* LeftmostDefense*. Assume D and D' agree on $\{d_1 = d'_1, \dots, d_{i-1} = d'_{i-1}\}$, $d_i <_R d'_i$. Then f^{-1} and f'^{-1} agree on $\{d_1 \dots d_{i-1}\}$ (being either equal or both undefined).*

Proof. Assume that f^{-1} and f'^{-1} do not agree on a left-most d_j, $j < i$. Let $a_l = \min_R\{f^{-1}(d_j), f'^{-1}(d_j)\}$ (note that at least one is defined). Now if $f(a_l) = d_i$, then d_i was a left-most candidate defender for a in LeftmostDefense not only for f but also for f' since by assumption we have

$$f[\{a_1 \dots a_{l-1}\}] \cap \{d_1 \dots d_{j-1}\} = f'[\{a_1 \dots a_{l-1}\}] \cap \{d_1 \dots d_{j-1}\}$$

which is a contradiction with existence of such d_j. □

Now let us state and analyze the main algorithm. Informally, the algorithm incrementally builds a defense by adding a defender for a left-most currently undefendable vertex.

procedure $\mathcal{A}$-DefensiveDomination(G, $\mathcal{A}$, D_{min})
 Input: Represented interval (multi)graph G
 Input: Multiset family $\mathcal{A} \subseteq 2^{V_G}$ (attack)
 Input: Multiset $D_{min} \subseteq V_G$
 Output: Smallest defender placement $D_{min} \subseteq D \subseteq V_G$
 $D \leftarrow D_{min}$
 loop
  for all $A \in \mathcal{A}$ **do**
   $f_A \leftarrow$LeftmostDefense(G, A, D)
  end for
  if all attackers defended in every f_A **then**
   return D
  end if
  $u \leftarrow$ left-most$_R$ vertex undefended in some f_A.
  $d' \leftarrow$ right-most$_R$ vertex of $N[u]$.
  $D \leftarrow D \cup \{d'\}$ ▷ A and u are referred to as *reason* for this d'.
 end loop
end procedure

We show that this algorithm optimally solves the problem of $\mathcal{A}$-defensive domination on interval graphs.

Proof of Theorem 2. The algorithm always returns a solution, and any returned solution is a valid defense against all attacks, as certified by the computed maps f_A. Additionally, all the intervals of $D \setminus D_{min}$ are always inclusion-maximal. We only need to show size-optimality.

Let us call the computed defense $D_{\text{ALG}} = \{d_1^{\text{ALG}} \le_R \dots\}$ and take an optimum-size defense $D_{\text{OPT}} = \{d_1^{\text{OPT}} \le_R \dots\}$ such that it agrees with D_{ALG} on a longest possible prefix $\{d_1^{\text{ALG}} = d_1^{\text{OPT}}, \dots, d_{i-1}^{\text{ALG}} = d_{i-1}^{\text{OPT}}\}$. We additionally assume that all the intervals of $D_{\text{OPT}} \setminus D_{min}$ are inclusion-maximal (as are those of $D_{\text{ALG}} \setminus D_{min}$). Let $\{f_{\text{ALG},A}\}_{A\in\mathcal{A}}$ and $\{f_{\text{OPT},A}\}_{A\in\mathcal{A}}$ be the defense maps computed by LeftmostDefense for D_{ALG} and D_{OPT} respectively. Note that $f_{\text{OPT},A}$ are well-defined thanks to Lemma 5.

If $D_{\text{ALG}} = D_{\text{OPT}}$ we are done. Otherwise, let $d_i^{\text{ALG}} \neq d_i^{\text{OPT}}$ be the first difference in the solutions. Note that thanks to Lemma 6 the maps $f_{\text{ALG},A}^{-1}$ and $f_{\text{OPT},A}^{-1}$ agree on $\{d_1^{\text{ALG}}, \dots, d_{i-1}^{\text{ALG}}\}$. Consider the two possibilities:

1. $d_i^{\text{ALG}} <_R d_i^{\text{OPT}}$. Let A' and u' be the reason for including d_i^{ALG} in D_{ALG}. Now opt can not defend u' in A': We have $d_i^{\text{OPT}} \notin N[u']$ since $d_i^{\text{OPT}} >_R d_i^{\text{ALG}}$ and d_i^{ALG} is the right-most neighbor of u'. At the same time, all the vertices of D_{OPT} before

d_i^{OPT} are either already used in $f_{\mathrm{OPT},A'}$ (as they are in $f_{\mathrm{ALG},A'}$) or too far (otherwise u' would not be chosen as a reason for $d' \in D_{\mathrm{ALG}}$ at that point).

2. $d_i^{\mathrm{ALG}} >_R d_i^{\mathrm{OPT}}$. Take as d_j^{ALG} the *left-endpoint* left-most$_L$ with $j \geq i$ and then set $D'_{\mathrm{OPT}} = D_{\mathrm{OPT}} - d_i^{\mathrm{OPT}} + d_j^{\mathrm{ALG}}$. We show that D'_{OPT} is a valid defense: take any attack $A \in \mathcal{A}$ and let $a = f^{-1}_{\mathrm{OPT},A}(d_i^{\mathrm{OPT}})$. If such a is undefined, let $f'_{\mathrm{OPT},A} = f_{\mathrm{OPT},A}$ and this map trivially defends A using D'_{OPT} as well.

If such a exists, let $d_l^{\mathrm{ALG}} = f_{\mathrm{ALG},A}(a)$ and note that $l \geq i$ thanks to Lemma 6. Now in both cases, $a <_R d_i^{\mathrm{OPT}}$ and $a >_R d_i^{\mathrm{OPT}}$, we straightforwardly obtain $d_j^{\mathrm{ALG}} \in N[a]$. The map $f'_{\mathrm{OPT},A}$ derived from $f_{\mathrm{OPT},A}$ by replacing $f'_{\mathrm{OPT},A}(a_j) = d_j^{\mathrm{ALG}}$ is then a valid defense map. In either case we get a contradiction with the choice of D_{OPT}. □

4. Cops and robber on interval graphs

In this section we focus on the game at hand on interval graphs. First we examine some properties of the game and introduce more terminology specific to the game on interval graphs, and then we introduce the *restricted game* and sketch the equivalence with the original game.

4.1. Closer look at the original game

Observe that in a disconnected graph the cops have to decide at the beginning of the game on a distribution among the components before the robber chooses a component to play in and after that only the cops placed in that component are relevant. In this light we get the following:

Proposition 7. *For every s including ∞, $c_s(G) = \sum_i c_s(G_i)$ where G_i are the connected components of G.*

In the rest of the paper we assume that G is connected.

To avoid special treatment of the game-states at the start of the game, we work with a game with modified starting state, which is equivalent to the unmodified game on connected graphs:

Proposition 8. *A connected interval graph G is k-cop-win if and only if k cops win starting all positioned at the leftmost vertex a of G (w.r. to $<_R$) with the robber starting on the rightmost vertex b (w.r. to $<_L$) such that* $\mathrm{dist}(a,b) \geq 2$. *If there is no such b, then the graph is* 1*-cop-win.*

Proof. To show this, assume that k cops have a wining strategy in the unmodified game starting at a configuration C. In the modified game, the cops can first use several moves to get to C ignoring the robber (or even capturing him on the way accidentally) and then play out the winning strategy according to the position of the robber at that point.

If the robber has a wining strategy in the unmodified game for any starting position of the cops, then let C be the cop's positions after their first move from a and let w be the desired robber's starting position for C. The cases $w = a$ or $w \in C$ can happen only if the radius of the graph is ≤ 1 and therefore G can not be robber-win. Also, we can assume $w \notin N[C]$ as such strategy would lose immediately. Otherwise w and b belong to the same component of $G \setminus C$, since every such component is either contained in $N[C]$ (and therefore not containing w) or contains b as the right-most vertex. In that case, the robber can move to w and play out his winning strategy. □

Formally, we denote the game state before cops' move (also *cop-state*) by $\mathcal{C}(C, w)$, where C is a multiset of vertices occupied by the cops and w is the vertex occupied by the robber. Note that we normally do not distinguish the cops on one vertex. The game state before robber's move (also *robber-state*) is $\mathcal{R}(C, A)$ with C as above and A is the set of all vertices the robber may reach in his following move (thus, A is the vertex set of the connected component of $G - C$ containing the robber).

Note that before any robber's move, two states with the robber in the same component of $G - C$ offer the same moves to the robber and this notation already slightly reduces the complexity of the examined states. Also note that A is by definition always connected which gives us the following:

Proposition 9. *For any game state $\mathcal{R}(C, A)$, $\overline{A}$ is an interval.*

For a state $\mathcal{R}(C, A)$ we call the interval $\overline{A}$ the *playground* of the state. The left and right endpoints of the playground (which is an open interval) are cutpoints called specifically the *left and right barrier*, usually denoted by l and r, respectively.

Note that all vertices containing the barriers, i.e., $V[l] \cup V[r]$, are occupied by the cops (otherwise the playground would be bigger). The vertices of $V[l]$ and in $V[r]$ are called the *barriers' support*. Note that the support of either barrier may be empty and in such case, due to the connectedness of G, we may assume that such a barrier is at either 1 or $2|V|$. The vertices in $V(l, r)$ are called the *playground support*. Note that by definition, the support of a playground is always disjoint from the supports of the barriers.

Among the cops occupying a barrier's support, we choose and fix one cop per vertex. Let us call these cops the cops *holding* the barrier. Note that a cop may hold both barriers at once, but as we see below, that this may happen only just before capturing the robber.

A playground (l, r) is *feasible*, if $|V(l) \cup V(r)| \leq k$, that is, if the cops are able to hold both barriers at once. A playground (l, r) is *nontrivial* if $V(l, r)$ is nonempty and it does not contain all vertices of G.

For every feasible and nontrivial playground (l, r), we fix a *canonical* game state

$$\S(l,r) = \mathcal{R}(V[l] \cup V[r], V(l,r))$$

in which the cops occupy all vertices in $V[l] \cup V[r]$ and the extra cops, if any, are positioned on an arbitrary (but fixed) vertex in $V[l] \cup V[r]$. A game state won by the cops, canonical to every playground with $V(l, r) = \emptyset$ is denoted by $\mathcal{WIN}$.

Proposition 10. *If $V(l, r) \neq \emptyset$, then the playground of $\S(l, r)$ is (l, r).*

The cops occupying the vertices in C *threaten* a vertex set T if the cops can occupy all vertices of T after one cop-move. This is equivalent to an existence of a partial surjective mapping from the cops on C to T such that every cop is assigned to a vertex at distance at most one. If the cops threaten $V[i]$ for some $i \in \{1, \dots, 2|V|\}$, then we also say that the cops *threaten* the cut i. When considering a set of cops threatening T, we fix a matching between the threatening cops and the vertices in T for the moment. In the rest of this section we introduce some additional notation that allows us to formally define the sets T we are interested in.

We say that a cop/robber *is over* an interval I if it is located on a vertex v such that $I \subseteq \overline{v}$. Then, the cops $c_1, \dots, c_p$ *are over* I if c_i is at vertex v_i, $i = 1, \dots, p$, and $I \subseteq \overline{v_1} \cup \dots \cup \overline{v_p}$. Any maximal interval B such that some cops are over B is called a *base*. Given a base B, $\xi(B) = \bigcup_{x \cap B \neq \emptyset} \overline{x}$ is called the *cover* of B.

Note that in game state $\mathcal{R}(C, A)$, if the robber positions himself on a vertex v such that $\overline{v}$ intersects a base, then the cops can catch the robber in the next move. Let $A' \subseteq A$ be the union intervals safe for the robber, that is $A' = \overline{V(A) \setminus N[C]}$. This includes vertices v such that $\overline{v}$ is entirely contained in $\xi(B) \setminus B$. Any maximal interval in $\overline{A'}$ is called a *hole in* $\mathcal{R}(C, A)$, or simply a *hole*, if the state is clear from the context. Hence, if $\mathcal{C}(C, r)$ is the state that follows $\mathcal{R}(C, A)$, then either the cops can immediately catch the robber or $\overline{r}$ is contained in some hole in $\mathcal{R}(C, A)$.

Given a state $\mathcal{R}(C, A)$, a *base collection* $\mathcal{B}$ is a set of bases $\{B_1, \dots, B_l\}$ such that $\bigcup_{i=1}^{l} \xi(B_i)$ is an interval. Denote the latter interval by $\xi(\mathcal{B})$. A base collection $\mathcal{B}$ is *maximal* if $\mathcal{B} \cup \{B\}$ is not a base collection for any base B, $B \notin \mathcal{B}$.

4.2. Maneuvers and the restricted game

The main tool of our result is to transform an arbitrary cops' winning strategy to a *restricted* strategy in an equivalent but simpler *restricted game* on a smaller state space. Informally, a restricted game state only describes which cuts are held or threatened by the cops and the current playground of the robber or his choices of a new playground.

While a general cops' strategy is mapping from *every* valid state of the game to a move valid in that state, a *restricted cop's strategy* is a mapping from a *restricted game states* $\mathcal{C}(C, r)$ to the maneuvers valid in those states. In the following we fix a constant $Q = 12$, that is the sufficient number of following playgrounds to be considered, as we show later. A *restricted game state* is $\mathcal{WIN}$ or one of the following:

Restricted cop-states $\S(l, r) = \mathcal{R}(V[l] \cup V[r], (l, r))$, where the cops choose the next maneuver to perform. Note that the general game state is a robber-state, which is more convenient as the maneuver ignores the position of the robber possibly except for the last turn (see below).

Restricted robber-states $\mathcal{R}\{(l_1, r_1), \dots (l_q, r_q)\}$, $q \leq Q$, where the robber chooses the next playground he will be restricted to after the next cop-move. This happens only before the last move in a split-maneuver, where the cops threaten multiple cuts and the robber has to decide where to stand before the cops actually choose which two barriers to create and therefore what will be the new playground. Note that the cops actually do not only present robber with the threatened cuts, but additionally give the robber all possible outcomes (playgrounds) to choose from. The conditions on the maneuvers ensure that the list of options is complete and valid.

A *maneuver* is a fixed finite sequence of cop-moves not depending on the robber's movement in the meantime possibly except for the last move and automatically captured the robber if he ends his move adjacent to a cop. A maneuver always starts in a restricted cop-state and ends in a robber-state (to choose the next playground) or a cop-state (if there is only one choice). A maneuver is *valid* if k cops are enough to perform the moves of the maneuver. There are two types of maneuvers:

Endgame from $\S(l, r)$ to $\mathcal{WIN}$. Starting from the state $\S(l, r)$, the cops move so that in each turn hold l and r, into position $\mathcal{R}(C, A)$ such that C contains $V[l] \cup V[r]$ and dominates $V(l, r)$. In their next move the cops capture the robber. Such a multiset C with $|C| \leq k$ is a *witness* of the maneuver. The maneuver itself is, in this case, the sequence of the above moves that lead from $\S(l, r)$ to $\mathcal{WIN}$.

Split from $\S(l,r)$ **to** $\mathcal{R}\{(l_1,r_1),\dots(l_q,r_q)\}$, $q \le Q$, with $l_i \le l_{i+1}$ for each $i \in \{1,\dots,q-1\}$.

Starting from the state $\S(l,r)$, the move (while holding l and r) into position $\mathcal{R}(C,A)$ such that

(i) the cops are holding l and r,
(ii) there exists a base collection $\mathcal{B} = \{B_1,\dots,B_{q-1}\}$ such that $r_i, l_{i+1} \in \xi(B_i)$ for each $i = 1,\dots,q-1$,
(iii) the cops are threatening $V[l_i] \cup V[r_i]$ for each $i \in \{1,\dots,q\}$,
(iv) C dominates $V(r_i,l_{i+1})$ for each $i \in \{1,\dots,q-1\}$,
(v) C dominates $V(l,l_1)$ and $V(r_q,r)$.

Such a multiset C with $|C| \le k$ is called the *witness* of the maneuver. Note that the cops holding l and r may be used in the threatening mapping. Also note that the conditions on dominating $V(r_i,l_{i+1})$ are trivially satisfied if $r_i > l_{i+1}$.

Once the cops take the positions in C, the robber makes a move by selecting his next position w. Either $w \in N[C]$ and the cops win immediately, or $w \in V(l_i,r_i)$ for some $i \in \{1,\dots,q\}$. Then, the cops make a move that results in $\S(l_i,r_i)$. Note that in the restricted game the robber is given the choice of i even in the case that for his choice of w both $w \in V(l_i,r_i)$ and $w \in V(l_j,r_j)$ and the cops would therefore make the choice of the next playground (out of the two) in the real game.

It is not trivial to check for the existence of such witnesses, but we can find a smallest witness using the algorithm for $\mathcal{A}$-defensive domination introduced in Section 1.2:

Lemma 11. *There are polynomial-time algorithms deciding the validity of maneuvers endgame and split.*

Proof. We observe that by the definitions, a multiset D is a smallest witness of a split maneuver from $\S(l,r)$ to one of $\S(l_i,r_i)$, $i \in \{1,\dots p\}$ if and only if D is a smallest $\mathcal{A}$-defensive multiset for $\mathcal{A} = \{V(l_i) \cup V(r_i) | i \in \{1,\dots p\}\} \cup \binom{Y}{1}$ (possible barriers to be taken) with $D_{min} = V[l] \cup V[r] \subseteq D$ (pre-placed defenders/cops) and unbounded capacities D_{max}, where $Y = V(l,l_1) \cup V(r_p,r) \cup \bigcup_{i=1}^{p-1} V(r_i,l_{i+1})$ (the vertices between the playgrounds to be dominated individually, preventing robber from safely moving there).

Indeed, suppose first that D is a witness of size k of a split maneuver. Then D is a solution to $\mathcal{A}$-defensive domination: For each $A \in \mathcal{A}$ generated by a playground, the injective mapping $f_X : X \to D$ is given by the possible cop-move from D to occupy A. For $A \in \mathcal{A}$ arising from Y, the mapping (of a single defender) follows from D (simply) dominating Y. On the other hand, it is easy to check that a solution to $\mathcal{A}$-defensive domination satisfies all the witness conditions.

Similarly, a multiset D is a smallest witness of an endgame maneuver from $\S(l,r)$ if and only if D is a smallest $\binom{V(l,r)}{1}$-defensive multiset with $D_{min} = V[l] \cup V[r] \subseteq D$ and unbounded capacities D_{max}.

The algorithm deciding the problem of $\mathcal{A}$-defensive domination is polynomial for interval graphs according to Theorem 2 proved in Section 3. The input size is polynomial in the size of G and number of playgrounds considered, which is bounded by Q. $\square$

This then allows us to decide the existence of a cops' restricted strategy.

Theorem 12. *There exists a $O(n^{O(1)})$-time algorithm that, given an interval graph G with $|G| = n$ and an integer k, decides the existence of a winning restricted strategy using k cops for G.*

Proof. We construct a game-state digraph D representing the restricted game. V_D consists of all restricted game states including $\mathcal{WIN}$, the initial state is the state corresponding to the initial position of the modified game. The only cop-win state is $\mathcal{WIN}$, there are no robber-win states.

For every valid *endgame* maneuver from $\S(l,r)$ add an arc from $\S(l,r)$ to $\mathcal{WIN}$. For every valid *split* from $\S(l,r)$ to one of the $\S(l_i,r_i)$, $i \in \{1,\dots,q \le Q\}$, add an arc from $\S(l,r)$ to $\mathcal{R}\{(l_1,r_1),\dots(l_q,r_q)\}$. From every $\mathcal{R}\{(l_1,r_1),\dots(l_q,r_q)\}$ add arcs to $\S(l_i,r_i)$, $i \in \{1,\dots,q\}$.

We decide the game given by D using a general game theory state-marking algorithm, giving us either a winning restricted strategy for the cops or a non-losing restricted strategy for the robber.

Since Q is a fixed constant, D has polynomial size, every arc can be decided in polynomial time according to Lemma 11, the state-marking algorithm also runs in time polynomial in $|D|$. $\square$

4.3. Equivalence of the restricted game

In this section we prove the following theorem.

Theorem 13. *For an interval graph G and an integer k, k cops have a winning strategy for the Cops and ∞-fast Robber game if and only if k cops have a winning strategy in the restricted game on G.*

With this, we may immediately prove Theorem 1. Before we prove Theorem 13, we show several lemmas which compose the individual steps of the equivalence.

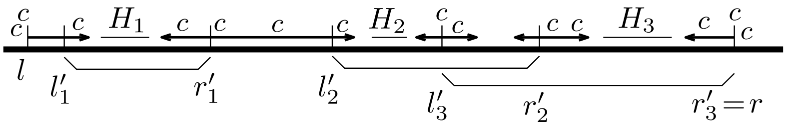

Fig. 2. An illustration of the playgrounds of $\mathcal{P}$ with $q = 3$. Note that the playgrounds may overlap. The intervals H_j are the holes separating the groups of cops of C.

Proof of Theorem 1. According to Theorem 12, we can decide the existence of a winning restricted cops' strategy in polynomial time. By Theorem 13, such a strategy exists if and only if a general winning cops' strategy exists.

Note that any play of a restricted cops' winning strategy visits every restricted cop-state (playground) at most once. There are $O(n^2)$ playgrounds and every canonical robber-state is followed by a maneuver to a canonical cop-state. It is easy to see that playing out any single maneuver takes $O(n)$ cop-moves when the cops follow shortest paths to their destinations. Therefore the game takes $O(n^3)$ turns. $\square$

First, we need additional definitions. A strategy for the Cops and ∞-fast Robber game is called *simple*, if at the beginning of any turn in which the robber occupies a vertex p and in which the playground changes, the following holds: for each base B there exist at most two cuts l and r, $l, r \in \xi(B)$, such that if $\mathrm{R}(p) < \mathrm{L}(B)$, then the cops do not take any cuts in B except possibly for r, and if $\mathrm{L}(p) > \mathrm{R}(B)$, then the cops do not take any cuts in B except possibly for l. We call l and r, respectively, to be the *left* and *right barriers associated with* B. Informally speaking, in a simple strategy if the robber positions himself to the left (right, respectively) of B in a turn in which the playground changes, then the cops either start to hold r (l, respectively) or do not hold any barrier of B.

Note that in a general strategy, in a move that changes the playground many base collections can be formed and each of them can contain many bases (up to one collection per a cop). We prove below that it is enough to consider one base collection at a time. This motivates the following definition:

We say that a strategy for the Cops and ∞-fast Robber game is *semi-restricted* if it is simple and for each cops' move that results in a change of the playground in the preceding robber's move there exists exactly one base collection.

We follow with several lemmas and their proofs:

Lemma 14. *For an interval graph G and an integer k, if k cops have a winning strategy for the Cops and ∞-fast Robber game, then k cops have a simple winning strategy.*

Proof. Let us consider a playground changing move in a winning strategy $\mathcal{S}$ and the two game states $\mathcal{R}(C, A)$ and $\mathcal{C}(C, w)$ preceding the playground change. Let $H_1, \ldots, H_q$ be all holes at the beginning of $\mathcal{R}(C, A)$ ordered so that $\mathrm{R}(H_j) < \mathrm{L}(H_{j+1})$ for each $j \in \{1, \ldots, q-1\}$. Take any base B formed by the cops at the beginning of $\mathcal{R}(C, A)$. If $\mathrm{R}(H_1) \le \mathrm{L}(B)$, then let $p \in \{1, \ldots, q\}$ be the maximum index such that $\mathrm{R}(H_p) \le \mathrm{L}(B)$, and let $p = 0$ otherwise.

Since $\mathcal{S}$ is winning, for each $j \in \{1, \ldots, p\}$ there exists a barrier $r_j \in \xi(B)$ such that if $w \in V(H_j)$, then the cops take in $\mathcal{C}(C, w)$ either no barrier in $\xi(B)$ or they take r_j. Suppose that $r_i \neq r_{i'}$ for some $i, i' \in \{1, \ldots, p\}$. We argue that we can modify the strategy $\mathcal{S}$ so that $r_i = r_{i'}$ and the 'worst-case' length of the new strategy is not greater than the length of $\mathcal{S}$. If r_i is not taken by the cops when $w \in V(H_i)$, then one may change $r_i := r_{i'}$. By using a similar argument for $r_{i'}$, we obtain that both r_i and $r_{i'}$ become the endpoints of the playgrounds in the cases when $w \in V(H_i)$ and $w \in V(H_{i'})$, respectively. Suppose without loss of generality that $i < i'$. Then, however, changing the playground in case of $w \in V(H_{i'})$ to be the same as in case of $w \in V(H_i)$ results in particular in $r_i = r_{i'}$.

We repeat the same argument for B and $l_{p+1}, \ldots, l_q$, where l_j is the border taken by the cops when $w \in V(H_j)$ for each $j = p+1, \ldots, q$. As a result, we obtain a strategy with the same bases as in $\mathcal{S}$, of length not greater than that of $\mathcal{S}$. By repeating the same transformation for each remaining base we obtain the desired simple strategy. $\square$

Lemma 15. *Let $\mathcal{S}$ be a winning strategy using k cops. Given any robber state $\mathcal{R}(C, A)$ of $\mathcal{S}$, let $\mathcal{R}(C_i, A_i)$, $i \in \{1, \ldots, q\}$, be the robber states of $\mathcal{S}$ reachable from $\mathcal{R}(C, A)$ in two moves (a robber move and a cop move). Let $P = (l, r)$ be the playground corresponding to $\mathcal{R}(C, A)$ and $P_i = (l_i, r_i)$ be the playground corresponding to $\mathcal{R}(C_i, A_i)$ for each $i \in \{1, \ldots, q\}$. Then, there is a semi-restricted cops' strategy that uses k cops, starts in state $\S(l, r)$ and either wins or results in state $\S(l_i, r_i)$ for some $i \in \{1, \ldots, q\}$.*

Proof. In view of Lemma 14, we may assume without loss of generality that $\mathcal{S}$ is simple. If C is a witness for *endgame*, then the considered turn does not violate the definition of semi-restricted strategy. Hence, assume that this is not the case and let $H_1, \ldots, H_q$ be the holes at the beginning of $\mathcal{R}(C, A)$. See Fig. 2 for an illustration.

We prove the lemma by induction on the number of base collections of a strategy. If the state $\mathcal{R}(C, A)$ consists of one base collection, then the two following moves of $\mathcal{S}$ themselves form a is semi-restricted strategy. Thus, suppose that $\mathcal{S}$ has more than one base collection.

Let $P'_j = (l'_j, r'_j)$ be the playground at the end of $\mathcal{C}(C, w)$ when $w \in V(H_j)$ for each $j \in \{1, \dots, q\}$. Let the corresponding states be $\mathcal{R}(C'_j, B'_j)$, $j \in \{1, \dots, q\}$. Denote by $P'_{i_1}, \dots, P'_{i_p}$ all inclusion-maximal playgrounds 'between' the base collections, that is, $P'_{i_j} \nsubseteq \mathcal{B}$ for any base collection $\mathcal{B}$ formed at the beginning of $\mathcal{R}(C, A)$, $j \in \{1, \dots, p\}$. By assumption, $p > 1$.

The strategy construction is algorithmic and is done by modifying $\mathcal{S}$. First, the modified strategy forms the base collection $\mathcal{B}$ with minimum $\mathrm{L}(\overline{\mathcal{B}})$. Suppose that the robber occupies a vertex w such that $\mathrm{R}(\overline{w}) < \mathrm{R}(\overline{\mathcal{B}})$. Hence, $\overline{w} \subseteq (l'_j, r'_j)$ for some $j < i_2$. The modified strategy plays the split maneuver to (l'_j, r'_j), as required.

Hence, assume that $\mathrm{R}(\overline{w}) \geq \mathrm{R}(\overline{\mathcal{B}})$. The strategy plays a split to (l'_{i_2}, r). Denote by $\mathcal{S}'$ the strategy that performs this maneuver. There exists a (general) strategy that uses $p-1$ base collections (all base collections, except for $\mathcal{B}$, of the initial strategy $\mathcal{S}$) and in one turn results in a playground $(l'_{j'}, r'_{j'})$ for some $j' \in \{i_2, \dots, q\}$. By the induction hypothesis, there exists a semi restricted strategy $\mathcal{S}''$ that either wins or results in a state $\S(l'_{j'}, r'_{j'})$ for some $j' \in \{i_2, \dots, q\}$. Thus, $\mathcal{S}'$ together with $\mathcal{S}''$ is the strategy that satisfies the conditions of the lemma. Indeed, the number of cops that the latter strategy used is k because, by definition, no cop is simultaneously used in two base collections. $\square$

The following lemma is used to show that it is sufficient to consider a bounded number of playgrounds resulting from a split maneuver.

Lemma 16. *Let $\mathcal{B} = \{B_1, \dots, B_q\}$, $q \geq 12$, be a base collection that is formed by k cops. Let l_i and r_i be the left and right barriers associated with B_i, $i \in \{1, \dots, q\}$. If $1 \leq j < j' \leq q$ and $j' - j \leq 3$, then endgame maneuver using k cops is possible from $\S(l_j, r_{j'})$.*

Proof. We prove the lemma by placing the cops so that they hold l_j and $r_{j'}$ and dominate $V(l_j, r_{j'})$. The cops in $\mathcal{B}$ used to threaten l_j and $r_{j'}$ are selected to occupy the vertices in $V(l_j) \cup V(r_{j'})$. If $j' = j+1$, then two additional cops can dominate $V(l_j, r_{j'})$, which follows directly from the definition of base collection. The two cops are available, because $q \geq 9$. Hence, assume that $j' > j+1$ and we describe how to dominate $V(l_j, r_{j'})$ with the remaining cops.

Place the cops initially used to form $B_{j+1}, \dots, B_{j'-1}$, except for those in Q, in the same way as in the base collection $\mathcal{B}$. The possibly non-dominated vertices are then the ones in $V(l_{j+1}, r_{j+2})$ if $j' - j = 3$, and $V(l_j, r_{j+1}) \cup X \cup V(l_{j'-1}, r_{j'}) \cup Y$, where X (respectively, Y) is the set of vertices initially dominated by the cops in $\mathcal{B}$ present on B_{j+1} (respectively, $B_{j'-1}$). Let $X' \subseteq X$ and $Y' \subseteq Y$ be, respectively, the vertices in X and Y occupied by the cops in Q. The vertex v in X' such that $\mathrm{R}(v) \geq \mathrm{R}(x)$ for each $x \in X'$ dominates X. Another cop can dominate $V(l_j, r_{j+1})$ by definition of base collection. Similarly, two cops can dominate $V(l_{j'-1}, r_{j'}) \cup Y$.

If $j' - j < 3$, then the construction is completed. Otherwise, two additional cops can be used to dominate $V(l_{j+1}, r_{j+2})$. Hence, in the worst case 6 additional cops are used with respect to the ones that initially form $B_j, \dots, B_{j'}$. No cop from a base B_i, $i < j-1$ (respectively, $i < j'+1$), is used to hold l_j ($r_{j'}$, respectively). Since $q \geq 12$, the total number of cops used to construct the witness of endgame does not exceed k. $\square$

Lemma 17. *Let $\mathcal{S}$ be a winning strategy using k cops. For a robber state $\mathcal{R}(C, A)$ in $\mathcal{S}$ let $\mathcal{R}(C_i, A_i)$, $i \in \{1, \dots, q\}$, be the robber states of $\mathcal{S}$ reachable from $\mathcal{R}(C, A)$ in two moves (a robber's move and a cops' move). Let $P = (l, r)$ be a playground corresponding to $\mathcal{R}(C, A)$ and $P_i = (l_i, r_i)$ be the playgrounds corresponding to $\mathcal{R}(C_i, A_i)$, $i \in \{1, \dots, q\}$. Then, there is a restricted cops' strategy that uses k cops, starts in the cop-state canonical to $\S(l, r)$ and either wins or results in a cop-state canonical to one of $\S(l_i, r_i)$.*

Proof. By Lemma 15, there exists a semi-restricted search strategy $\mathcal{S}'$ that uses k cops and results in one of the $\S(l_1, r_1), \dots, \S(l_q, r_q)$. The strategy $\mathcal{S}'$ performs several maneuvers and in the following we analyze two selected moves of $\mathcal{S}'$ after which the playground changes. Namely, let $\mathcal{R}(C', A')$ be a state of $\mathcal{S}'$ such that at the end of the following cops' move the playground changes to one of $(l'_1, r'_1), \dots, (l'_q, r'_q)$. Let (l', r') be the playground in $\mathcal{R}(C', A')$. We argue that there exists a restricted strategy $\mathcal{S}''$ that starts from $\S(l', r')$ and arrives at one of the $\S(l'_1, r'_1), \dots, \S(l'_q, r'_q)$. Let $\mathcal{B} = \{B_1, \dots, B_p\}$ be the base collection in $\mathcal{R}(C', A')$. If $p < 12$, then the two moves of $\mathcal{S}$ following $\mathcal{R}(C', A')$ form a restricted strategy as required. Hence, let $p \geq 12$ in the following.

In the remainder of this proof we assume that $l' \notin \xi(B_1)$ and $r' \notin \xi(B_p)$ as the other cases are similar with minor adjustments to border conditions. Namely having $l' \in \xi(B_1)$ allows to use these cops to threaten $V(l'_1)$ and analogously for r'.

This assumption implies that $l'_1 = l'$, $r'_q = r'$ and $p = q-1$. The strategy $\mathcal{S}''$ performs the first maneuver by holding l' and r' and threatening $V(l'_1) \cup V(r'_1)$ (by using the base B_1). If the robber responds by occupying a vertex in $V(l', r'_1)$, then the strategy plays a split to $(l', r'_1) = (l'_1, r'_1)$, and the lemma follows. Otherwise, the strategy plays a split to (l'_1, r'). Then, the strategy continues by holding l'_1 and r' and threatening $V(l'_q) \cup V(r'_q)$ (by using B_p). If the robber responds by occupying a vertex in $V(l'_q, r')$, then the strategy plays a split to $(l'_q, r') = (l'_q, r_q)$, and the lemma follows.

Otherwise, the strategy plays a split to (l'_1, r'_q) and performs iteratively the following maneuver. Let initially $i = 1$. The strategy holds l'_i and r'_q and threatens $V(l'_{i+2}) \cup V(r'_{i+2})$ by forming the bases B_{i+2} and B_{i+3}. Note that this is possible because no cop that currently holds the barrier l'_i is used in those bases. If the robber decides to occupy a vertex in $V(l'_i, r'_{i+2})$, then the strategy splits to (l'_i, r'_{i+2}) and, by Lemma 16, an endgame maneuver is valid from this state. Otherwise, i.e., if the robber occupies a vertex in (l'_{i+2}, r'_q), then the strategy plays a split to (l'_{i+2}, r'_q). If $i+2 \geq q-3$, then, by Lemma 16, an endgame is possible from (l'_{i+2}, r'_q). If $i+2 < q-3$, then we set $i := i+2$ and repeat the above step. Thus, after a finite number of split maneuver, an endgame maneuver occurs as required. □

Proof of Theorem 13. The "if" part is straightforward, as the cops can play out the maneuvers of a restricted winning strategy. The maneuver properties and witnesses ensure that the moves are possible and that after the maneuver, the robber is inside the respective playground or captured. Note that the capture may occur even playing out a split maneuver.

For the other direction, let $\mathcal{S}$ be a shortest (i.e., with the minimum number of turns) cop's winning strategy using k cops. Note that if the cops play according to $\mathcal{S}$, the game never revisits a game state.

Let S be the subgraph of the game state digraph representing $\mathcal{S}$, in which the vertices are all the cop- and robber-states of the game. From each cop-vertex, there is exactly one cop-move in S, as dictated by $\mathcal{S}$. From each robber-vertex, all the robber-moves are present in S. Note that we can assume S to be acyclic, since it is a shortest winning strategy. Also, fix any total ordering o of the states of S extending the partial order given by the arcs (moves).

For any robber-state in $\mathcal{S}$ find the maximum (with respect to o) robber-state $\mathcal{R}(C, B)$ in S with the same playground. We construct a winning restricted strategy $\mathcal{T}$ using k cops as follows. Let $\mathcal{R}(C_i, A_i)$ and (l_i, r_i), $i \in \{1, \dots, t\}$, be the states in S and the corresponding playgrounds reachable from $\mathcal{R}(C, A)$ in two moves. By Lemma 17, there exists a restricted strategy that uses k cops, starts in state $\mathcal{R}(C, A)$ and leads to $\mathcal{R}(C_i, A_i)$ for each $i \in \{1, \dots, t\}$.

This leaves the game in a state canonical to one of $\S(l_i, r_i)$ or $\mathcal{WIN}$. Note that all $\S(l_i, r_i)$ are different from $\S(l, r)$, because the state $\mathcal{R}(C, A)$ is latest such state.

Now we have that the latest occurrence of a robber-state with playground $\S(l_i, r_i)$ is (with respect to o) larger than that of $\S(l, r)$. Therefore, by playing $\mathcal{T}$, the latest state (with respect to o) with the same playground as the current one only increases. This proves that $\mathcal{T}$ is acyclic and therefore finite (as $\mathcal{S}$ is) and winning for the cops, as there is no draw or robber-win position in the game. □

5. Conclusions

We have shown a polynomial-time algorithm deciding the Cop and ∞-fast Robber game on interval graphs, therefore answering an open question of Fomin et al. posed in their paper "Pursuing a fast robber on a graph" [10].

Since the game is already NP-hard for general chordal graphs and even split graphs, it might be interesting to consider the complexity of the game on chordal graphs with bounded asteroidal number (or the number of leaves of the underlying tree for the standard intersection representation of chordal graphs) and the class of circular-arc graphs.

It seems that the notion of playgrounds of the reduced game can be extended to such graphs and they might have some common properties, but the analysis does not extend in a straightforward way. We propose the complexity of the game on such graphs as an open question. For chordal graphs, even an algorithm that is exponential in the asteroidal number would be of interest.

The definition of $\mathcal{A}$-defensive domination generalizes k-defensive domination, but explicitly specifying $\mathcal{A} = \binom{V}{k}$ is not practical for even small values of k, making the complexity of the problems incomparable. One direction to take might be to introduce another way to specify $\mathcal{A}$, perhaps inspired by matroid theory.

In this paper we only show an algorithm for $\mathcal{A}$-defensive domination assuming an unbounded number of defenders allowed on individual vertices. We strongly believe that specifying the number of allowed defenders for every vertex (in D_{max}) still allows for a polynomial algorithm very similar to ours, but considering both D_{min} and D_{max} seems to increase the technical complexity way beyond the main scope of this paper.

It would of course be of interest to see efficient $\mathcal{A}$-defensive domination algorithms for other classes, possibly specifying D_{min} and D_{max} as part of the input. Trees and, more generally, bounded tree-width graphs seem like natural candidates.

Acknowledgements

We would like to thank Andrzej Proskurowski and Peter Golovach for their insightful remarks in the initial stages of the work on the game.

References

[1] M.H. Albert, R.J. Nowakowski, D. Wolfe, Lessons in Play: An Introduction to Combinatorial Game Theory, AK Peters, USA, 2007.
[2] N. Alon, A. Mehrabian, On a generalization of Meyniel's conjecture on the cops and robbers game, Electron. J. Combin. 18 (2011) 1.
[3] B. Bollobás, Modern Graph Theory, Graduate Texts in Mathematics, vol. 184, Springer, New York, 1998.
[4] A. Bonato, E. Chiniforooshan, P. Pralat, Cops and robbers from a distance, Theoret. Comput. Sci. 411 (43) (2010) 3834–3844.
[5] A. Bonato, R. Nowakowski, The Game of Cops and Robbers on Graphs, Student Mathematical Library, American Mathematical Society, 2011.
[6] J. Chalopin, V. Chepoi, N. Nisse, Y. Vaxès, Cop and robber games when the robber can hide and ride, SIAM J. Discrete Math. 25 (1) (2011) 333–359.
[7] D. Dereniowski, D. Dyer, R.M. Tifenbach, B. Yang, The complexity of zero-visibility cops and robber, Theoret. Comput. Sci. 607 (2015) 135–148.
[8] D. Dereniowski, D. Dyer, R.M. Tifenbach, B. Yang, Zero-visibility cops and robber and the pathwidth of a graph, J. Comb. Optim. 29 (3) (2015) 541–564.
[9] A. Farley, A. Proskurowski, Defensive domination, Congr. Numer. 168 (2004) 97–107.
[10] F.V. Fomin, P.A. Golovach, J. Kratochvíl, N. Nisse, K. Suchan, Pursuing a fast robber on a graph, Theoret. Comput. Sci. 411 (7) (2010) 1167–1181.

[11] A.M. Frieze, M. Krivelevich, P.-S. Loh, Variations on cops and robbers, J. Graph Theory 69 (4) (2012) 383–402.
[12] T. Gavenčiak, Catching a fast robber on interval graphs, in: Proceedings of the 8th Annual Conference on Theory and Applications of Models of Computation, TAMC'11, Springer-Verlag, Berlin, Heidelberg, 2011, pp. 353–364.
[13] V. Isler, S. Kannan, S. Khanna, Randomized pursuit-evasion with local visibility, SIAM J. Discrete Math. 20 (1) (2006) 26–41.
[14] N. Korte, R.H. Möhring, An incremental linear-time algorithm for recognizing interval graphs, SIAM J. Comput. 18 (1) (1989) 68–81.
[15] A. Mehrabian, Cops and robber game with a fast robber on interval, chordal, and planar graphs, CoRR, arXiv:1008.4210, 2011.
[16] A. Mehrabian, Lower bounds for the cop number when the robber is fast, Combin. Probab. Comput. 20 (4) (2011) 617–621.
[17] A. Mehrabian, Cops and robber game with a fast robber on expander graphs and random graphs, Ann. Comb. 16 (4) (2012) 829–846.
[18] A. Mehrabian, The fast robber on interval and chordal graphs, Discrete Appl. Math. 180 (2015) 188–193.
[19] N. Nisse, K. Suchan, Fast robber in planar graphs, in: Proceedings of the 34th International Workshop on Graph-Theoretic Concepts in Computer Science (WG), in: Lecture Notes in Computer Science, vol. 5344, 2008, pp. 33–44.
[20] R. Nowakowski, P. Winkler, Vertex to vertex pursuit in a graph, Discrete Math. 43 (2) (1983) 235–239.
[21] A. Quilliot, A short note about pursuit games played on a graph with a given genus, J. Combin. Theory Ser. B 38 (1) (1985) 89–92.