Cops and Robber Game Without Recharging*

Fedor V. Fomin† Petr A. Golovach‡ Daniel Lokshtanov†

Abstract

Cops & Robber is a classical pursuit-evasion game on undirected graphs, where the task is to identify the minimum number of cops sufficient to catch the robber. In this work, we consider a natural variant of this game, where every cop can make at most f steps, and prove that for each $f \geq 2$, it is PSPACE-complete to decide whether k cops can capture the robber.

1 Introduction

The study of pursuit-evasion games is driven by many real-world applications where a team of agents/robots must reach a moving target. The mathematical study of such games has a long history, tracing back to the work of Pierre Bouguer, who in 1732 studied the problem of a pirate ship pursuing a fleeing merchant vessel. In 1960s the study of pursuit-evasion games, mostly motivated by military applications like missile interception, gave a rise to the theory of Differential Games [11]. Besides the original military motivations, pursuit-evasion games have found many applications reaching from law enforcement to video games and thus were studied within different disciplines and from different perspectives. The necessity of algorithms for pursuit tasks occur in many real-world domains. In the Artificial Intelligence literature many heuristic algorithms for variations of the problem like Moving Target Search have been studied extensively [8, 12, 13, 17, 18]. In computer games, for instance, computer-controlled agents often pursue human-controlled players and making a good strategy for pursuers is definitely a challenge [15]. The algorithmic study of pursuit-evasion games is also an active area in Robotics [10, 22] and Graph Algorithms [6, 16].

*Extended abstract of results in this paper appeared in the proceedings of SWAT 2010 [5].

†Department of Informatics, University of Bergen, N-5020 Bergen, Norway, emails: {fedor.fomin|daniello}@ii.uib.no.

‡School of Engineering and Computing Sciences, Durham University, South Road, DH1 3LE Durham, UK, email: petr.golovach@durham.ac.uk. Supported by EPSRC under project EP/G043434/1.

One of the classical pursuit-evasion problems is the Man and Lion problem attributed to Rado by Littlewood in [14]: A lion (pursuer) and a man (evader) in a closed arena have equal maximum speeds. What tactics should the lion employ to be sure of his meal? See also for more recent results on this problem [3, 21]. The discrete version of the Man and Lion problem on graphs was introduced by Winkler and Nowakowski [19] and Quilliot [20]. Aigner and Fromme [1] initiated the study of the problem with several pursuers. This game, named Cops & Robber, is played on a graph G by two players: the cop-player $\mathcal{C}$ and the robber player $\mathcal{R}$, which make moves alternately. The cop-player $\mathcal{C}$ has a team of k cops who attempt to capture the robber. At the beginning of the game this player selects vertices and put cops on these vertices. Then $\mathcal{R}$ puts the robber on a vertex. The players take turns starting with $\mathcal{C}$. At every turn each of the cops can be either moved to an adjacent vertex or kept on the same vertex. Let us note that several cops can occupy the same vertex at some move. Similarly, $\mathcal{R}$ responds by moving the robber to an adjacent vertex or keeping him on same vertex. It is said that a cop *catches* (or captures) the robber at some move if at that move they occupy the same vertex. The cop-player wins if one of his cops catches the robber, and the player $\mathcal{R}$ wins if he can avoid such a situation. The game was studied intensively and there is an extensive literature on this problem. We refer to surveys [2, 6] for references on different pursuit-evasion and search games on graphs.

In the game of Cops & Robber there are no restrictions on the number of moves the players can make. Such model is not realistic for most of the applications: No lion can pursuit a man without taking a nap and no robot can move permanently without recharging batteries. In this work, we introduce a more realistic scenario of Cops & Robber, the model capturing the fact that each of the cops has a limited amount of power or fuel.

We also find the Cops & Robber problem with restricted power interesting from combinatorial point of view because it generalizes the *Minimum Dominating Set* problem, one of the fundamental problems in Graph Theory and Graph Algorithms. Indeed, with fuel limit 1 every cop can make at most one move, then k cops can win on a graph G if and only if G has a dominating set of size k. Thus two classical problems— *Minimum Dominating Set* (fuel limit is 1) and Cops & Robber (unlimited fuel) are the extreme cases of our problem. It would be natural to guess that if the amount of fuel the cops possess is some fixed integer f, then the problem is related to distance f domination. Indeed, for some graph classes (e.g. for trees), the problems coincide. Surprisingly, the intuition that Cops & Robber and *Minimum f-Dominating Set* (the classical NP-complete problem) should be similar from the computational complexity point of view is wrong. The main result of this paper is that the problem deciding if k cops can win on an undirected graph is PSPACE-complete even for $f = 2$. Another motivation for our work is the long time open question on the computational complexity of the Cops & Robber problem (without power constrains) on undirected graphs. In 1995, Goldstein and Reingold [9] have shown that the classical Cops & Robber game is E-complete (the complexity class E = DTIME($2^{O(|I|)}$), where $|I|$ is the input size) on *directed* graphs and conjectured that similar holds for undirected graphs. However, even NP-hardness of the problem was not known until very recently [4]. By our results, in the game on an n-vertex undirected graph if the number of steps each cop is allowed to make is at most some polynomial of k, then deciding if k cops can win is PSPACE-complete.

2 Basic definitions and preliminaries

We consider finite undirected graphs without loops or multiple edges. The vertex set of a graph G is denoted by $V(G)$ and its edge set by $E(G)$, or simply by V and E if this does not create confusion. If $U \subseteq V(G)$ then the subgraph of G induced by U is denoted by $G[U]$. For a vertex v, the set of vertices which are adjacent to v is called the *(open) neighborhood* of v and denoted by $N_G(v)$. The *closed neighborhood* of v is the set $N_G[v] = N_G(v) \cup \{v\}$. The *distance* $\mathrm{dist}_G(u,v)$ between a pair of vertices u and v in a connected graph G is the number of edges in a shortest u,v-path in G. For a positive integer r, $N^r_G[v] = \{u \in V(G)\colon \mathrm{dist}_G(u,v) \le r\}$. Whenever there is no ambiguity we omit the subscripts.

The Cops & Robber game can be defined as follows. Let G be a graph, and let $f > 0$ be an integer. The game is played by two players: the cop-player $\mathcal{C}$ and the robber player $\mathcal{R}$. The players use the same rules as in the original game introduced Winkler, Nowakowski and Quilliot [19, 20] with one additional condition: during the whole game each of the cops can be moved from a vertex to another vertex at most f times in total. In other words, each of the cops has an amount of fuel which allows him to move at most f steps. Notice that even if a cop cannot move to adjacent vertex (run out of fuel), he is still active and the robber cannot move to the vertex occupied by the cop without being caught. Observe also that the player $\mathcal{R}$ wins if he can survive for $kf + 1$ moves, since it can be assumed that at least one cop is moved at each step (otherwise the robber can either keep his position or improve it). For an integer f and a graph G, we denote by $c_f(G)$ the minimum number k of cops sufficient for $\mathcal{C}$ to win on graph G.

We define the *position* of a cop as a pair (v,s) where $v \in V(G)$ and s is an integer, $0 \le s \le f$. Here v is the vertex occupied by the cop, and s is the number of moves along edges (amount of fuel) which the cop can do. The *position of a team* of k cops (or *position of cops*) is a multiset $((v_1,s_1),\ldots,(v_k,s_k))$, where (v_i,s_i) is the position of the i-th cop. For the *initial* position, all $s_i = f$. The *position of the robber* is a vertex of the graph occupied by him.

We consider the following Cops and Robber decision problem:

Input: A connected graph G and two positive integers k, f.

Question: Is $c_f(G) \le k$?

Let us finish the section on preliminaries with the proof of relations between Cops & Robber and r-domination announced in Introduction. The Cops & Robber problem with restricted power is closely related to domination problems. Let r be a positive integer. A set of vertices $S \subset V(G)$ of a graph G is called an *r-dominating set* if for any $v \in V(G)$, there is $u \in S$ such that $\mathrm{dist}(u,v) \le r$. The *r-domination number* $\gamma_r(G)$ is the minimum k such that there is an r-dominating set with at most k vertices. Then $\gamma_1(G)$ is the domination number of G.

The proof of the following observation is straightforward.

Observation 1. *For any connected graph G, $c_1(G) = \gamma_1(G)$.*

For $f > 1$, the values $c_f(G)$ and $\gamma_f(G)$ can differ arbitrarily. Consider, for example, the graph G which is the union of k complete graphs K_k with one additional vertex joined with all vertices of these copies of complete graphs by paths of length f. It can be easily seen that $\gamma_f(G) = 1$ but $c_f(G) = k$. Still, for some graph classes (e.g. for trees) these numbers are equal. Recall, that the *girth* of a graph G, denoted by $g(G)$, is the length of a shortest cycle in G (if G is acyclic then $g(G) = \infty$).

Lemma 1. *Let $f > 0$ be an integer and let G be a connected graph of girth at least $4f - 1$. Then $c_f(G) = \gamma_f(G)$.*

Proof. The proof of $\gamma_f(G) \le c_f(G)$ is trivial. To prove that $c_f(G) \le \gamma_f(G)$, we give a winning strategy of $\gamma_f(G)$ cops. Suppose that S is an f-dominating set in G of size $\gamma_f(G)$. The cops are placed on the vertices of S. Suppose that the robber occupies a vertex u. Then the cops from vertices of $S \cap N_G^{2f-1}(u)$ move towards the vertex occupied by the robber at the current moment along the shortest paths. We claim that the robber is captured after at most f moves of the cops. Notice that the robber can move at distance at most $f - 1$ from u before the cops make f moves. Because $g(G) \ge 4f - 1$, the paths along which the cops move are unique. Suppose that the robber is not captured after $f - 1$ moves of the cop-player, and the robber occupies a vertex w after his $f - 1$ moves. Since S is an f-dominating set, there is a vertex $z \in S$ such that $\mathrm{dist}_G(w,z) \le f$. Using the fact that $g(G) \ge 4f - 1$, and since the robber was not captured before, we observe that the cop from z moved to w along the shortest path between z and w and by his f-th move he has to enter w and capture the robber. □

3 PSPACE-completeness

The minimum dominating set problem is one of the classical NP-complete problems [7] and by Observation 1, it is NP-complete to decide whether $c_1(G) \leq k$. Here we prove that for $f \geq 2$, the complexity of the problem changes drastically.

Theorem 1. *For any $f \geq 2$, the* Cops and Robber *problem is* PSPACE-*complete.*

Remaining part of this section contains the proof of this theorem.

3.1 PSPACE-hardness

In this subsection we prove the first part of Theorem 1 that the Cops and Robber problem is PSPACE-complete for every $f \geq 2$.

We reduce to the PSPACE-complete Quantified Boolean Formula in Conjunctive Normal Form (QBF) problem [7]. For a set of Boolean variables $x_1, x_2, \dots, x_n$ and a Boolean formula $F = C_1 \wedge C_2 \wedge \cdots \wedge C_m$, where C_j is a clause, the QBF problem asks whether the expression

$$\phi = Q_1 x_1 Q_2 x_2 \cdots Q_n x_n F$$

is *true*, where for every i, Q_i is either $\forall$ or $\exists$. We assume additionally that $Q_n = \exists$. Clearly, QBF remains PSPACE-complete with this restriction. Given a quantified Boolean formula ϕ, we construct an instance (G, n) of our problem such that ϕ is true if and only if the cop-player can win on G with n cops.

Constructing G. For every $Q_i x_i$ we introduce a gadget graph G_i. For $Q_i = \forall$, we construct the graph $G_i(\forall)$ as follows.

- Construct vertices $x_i, \overline{x}_i, y_i, \overline{y}_i, z_i$ and edges $z_i x_i, x_i y_i, z_i \overline{x_i}, \overline{x_i y}_i, x_i \overline{x}_i$.
- Add vertices u_i and v_i assuming that $u_i = z_i$ for $f = 2$.
- Join z_i and u_i by the path P_i of length $f - 2$ and join u_i and v_i by the path P'_i of length f.

For $Q_i = \exists$, we construct the graph $G_i(\exists)$:

- Construct vertices $x_i, \overline{x}_i, y_i, z_i$ and edges $z_i x_i, x_i y_i, z_i \overline{x_i}, \overline{x_i} y_i, x_i \overline{x}_i$.
- Add vertices u_i and v_i assuming that $u_i = z_i$ for $f = 2$.
- Join z_i and u_i by the path P_i of length $f - 2$ and join u_i and v_i by the path P'_i of length f.

The graphs $G_i(\forall)$ and $G_i(\exists)$ are shown in Fig 1. Let $X_i = \{x_i, \overline{x}_i\}$, $Y_i = \{y_i, \overline{y}_i\}$ for $G_i(\forall)$ and $Y_i = \{y_i\}$ for $G_i(\exists)$.

Using these gadgets we construct G as follows.

- Construct gadget graphs G_i for $i \in \{1, \dots, n\}$.
- For each $i \in \{2, \dots, n\}$, join vertices from the set Y_{i-1} with all vertices from $X_i \cup Y_i$.
- For each $i \in \{2, \dots, n\}$, introduce vertices $w_i^{(1)}$ and $w_i^{(2)}$, add edges $z_i w_i^{(1)}$ and $z_i w_i^{(2)}$, and join the vertices $w_i^{(1)}, w_i^{(2)}$ with all vertices from the sets Y_j for $j \in \{1, \dots, i-1\}$.
- Add vertices $C_1, C_2, \dots, C_m$ corresponding to clauses and join them with the unique vertex of Y_n (recall that $Q_n = \exists$) by edges.
- For $i \in \{1, \dots, n\}$ and $j \in \{1, \dots, m\}$, the vertex x_i is joined with C_j by an edge if C_j contains the literal x_i, and $\overline{x}_i$ is joined with C_j if C_j contains the literal $\overline{x}_i$.

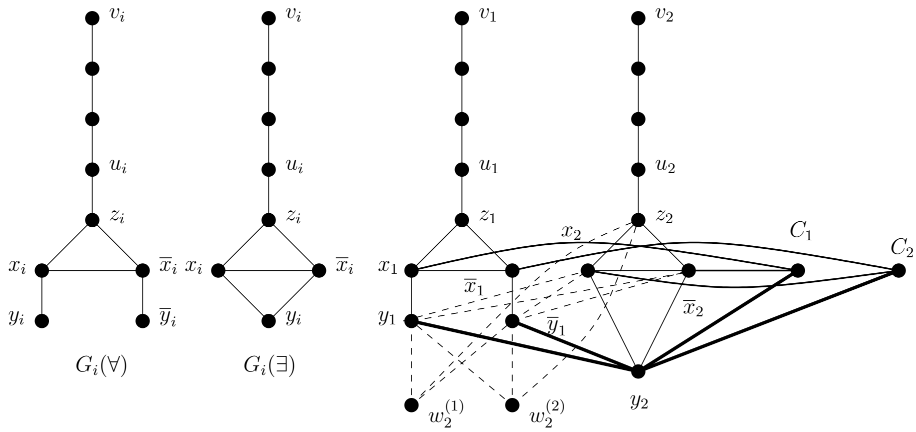

Figure 1: Graphs $G_i(\forall)$, $G_i(\exists)$ and G for $\phi = \forall x_1 \exists x_2\ (x_1 \vee \overline{x}_2) \wedge (\overline{x}_1 \vee x_2)$, $f = 3$. The edges shown by dashed lines are used to force the payers to move in a special way.

Now we prove the following two lemmata.

Lemma 2. *If $\phi = true$ then n cops have a winning strategy on G.*

Proof. We describe a winning strategy for the cop-player. The cops start by occupying vertices $u_1, \dots, u_n$. If the robber occupies a vertex of some path P'_i, then the cop from the vertex u_i moves toward him and captures the robber in at most f steps. Suppose that there is no robber on vertices of the paths $P'_1, \dots, P'_n$. For each $i \in \{1, \dots, n\}$, the cop from u_i moves to

z_i in $f-2$ steps. Now the cops occupy the vertices $z_1,\dots,z_n$ and each of them can do two moves along edges. Assume that the robber is not captured after these moves of the cops. Observe that if after the robber's last move he occupies a vertex of some set X_i or $\{w_i^{(1)},w_i^{(1)}\}$ for $i\in\{1,\dots,n\}$, then he is captured by the cop from z_i in one step. Hence, we assume that the robber is in a vertex of some set Y_i or in $\{C_1,\dots,C_m\}$. We consider two cases.

Case 1. The robber occupies a vertex of some set Y_i. For each $j\in\{1,\dots,i-1\}$, the cop from z_j moves either to x_j or to $\overline{x}_j$. We assume that the choice of x_j corresponds to the value *true* of the variable x_j and the choice of $\overline{x}_j$ corresponds to the value $false$ of x_j. Since $\phi = true$, the variables $x_1,\dots,x_{i-1}$ can be assigned values such that $Q_ix_i\dots Q_nx_nF = true$. The cop from the vertex z_j moves according to the value of x_j. If $x_j = true$ then he moves to the vertex x_j, otherwise he moves to $\overline{x}_j$. Now inductively for $j\in\{i,\dots,n\}$, we assume that the robber occupies a vertex of Y_j and we move the cop from z_j according to the following subcases.

a) $Q_j=\forall$. If the robber occupies the vertex y_j then the cop from z_j moves to x_j, and if the robber is in $\overline{y}_j$ then the cop moves to $\overline{x}_i$. We again suppose that a placement of a cop in x_j corresponds to the value *true* of the variable x_j, and a moving a cop to $\overline{x}_j$ corresponds to the value $false$. Notice that now the robber choses the value of the variable x_j. Now the robber should make his move:

   - If the robber stays in his old position then he will be captured in one step by the cop which is either in x_i or $\overline{x}_i$.
   - If the robber moves to a vertex of Y_{j-1} then again he will be captured in one step by the cop which is either in x_i or $\overline{x}_i$.
   - If the robber moves to a vertex $w_s^{(1)}$ or $w_s^{(2)}$ for $s>j$ then he will be captured in one step by the cop from z_s.

   Hence he should move to a vertex of Y_{j+1} or to one of the vertices $C_1,\dots,C_m$ if $j=n$ to avoid the capture.

b) $Q_j=\exists$. Then the robber occupies y_j. The cop from z_j moves either to x_i or to $\overline{x}_i$. The vertex is chosen in such a way that it corresponds to the value of the variable x_i for which (and for already assigned values of the variables $x_1,\dots,x_{j-1}$) $Q_{j+1}x_{j+1}\dots Q_nx_nF = true$. Then similarly to Subcase a), the robber is either captured by the next step or moves to a vertex of Y_{j+1} or to one of the vertices $C_1,\dots,C_m$ if $j=n$.

Finally, the robber is either captured or occupies some vertex C_s. Observe, that the cops have chosen the vertices of the sets $X_1,\dots,X_n$ such that

$F = true$ for the corresponding values of boolean variables. Hence there is a cop in a vertex adjacent with C_s and he captures the robber by the next move.

Case 2. The robber occupies some vertex C_j. For each $i \in \{1, \dots, n\}$, the cop from z_i moves either to x_i or to $\overline{x}_i$. Again we assume that the choice of x_i corresponds to the value *true* of the variable x_i and the choice of $\overline{x}_i$ corresponds to the value $false$ of x_i. Since $\phi = true$, the variables $x_1, \dots, x_n$ can be assigned values such that $F = true$. The cop from the vertex z_i moves according to the value of x_i. If $x_i = true$ then he moves to the vertex x_i and he moves to $\overline{x}_i$ otherwise. Now the robber makes his move:

- If the robber moves to the vertex y_n then he will be captured in one step by the cop which is either in x_n or $\overline{x}_n$.
- If the robber moves to a vertex of X_i unoccupied by a cop then he will be captured in one step by the cop from another vertex of this set.

Hence he should stay in C_j to avoid the capture. Since $F = true$ for the values of boolean variables which correspond to positions of the cops, there is a cop in a vertex adjacent with C_j and he captures the robber by the next move. □

To complete the proof of PSPACE-hardness, it remains to prove the following lemma.

Lemma 3. *If $\phi = false$ then the robber has a winning strategy against n cops on G.*

Proof. We describe a winning strategy for the robber-player. Assume that the cops have chosen their initial positions. If there is a path P_i' such that all vertices of the path are unoccupied by the cops then we place the robber on v_i. Since there are no cops at distance at least f from v_i, the winning strategy for the robber is trivial — he should stay in v_i. Suppose now that for each path P_i', there is a cop in one of the vertices of the path. We have n cops. Hence exactly one cop occupies one vertex of each path. Denote this cop by $\mathcal{C}_i$. Observe that if $\mathcal{C}_i$ move to vertex z_i, then he has no capacity to move further than distance two from this vertex. The robber is placed on a vertex of Y_1. The choice of the vertex and further moves of the robber are described inductively for $i \in \{1, \dots, n\}$.

Suppose that for each $i \leq j \leq n$, the cop $\mathcal{C}_j$ is on the path P_j or P_j', the robber does not occupy a vertex that out of reach of the cops (i.e. he didn't win yet), and currently the robber-player chooses a vertex of Y_i to move the robber there. Assume also that values of the variables $x_1, \dots, x_{i-1}$ are already defined and $Q_i x_i \dots Q_n x_n F = false$ for this assignment. Clearly, these conditions hold for $i = 1$. We consider two cases.

a) $Q_i = \forall$. Since $Q_i x_i \dots Q_n x_n F = false$, there is a value of x_i for which $Q_{i+1} x_{i+1} \dots Q_n x_n F = false$. If this value is *true* then the robber is placed on y_i and he is placed on $\overline{y}_i$ otherwise. Observe, that the value of x_i is chosen by the robber-player. Now the robber stays in his position until the cop $\mathcal{C}_i$ moves from z_i to the vertex of X_i adjacent with his current position (notice that if $\mathcal{C}_i$ moves to this vertex by sliding along two edges then the robber does not move) or some other cop $\mathcal{C}_j$ for $j > i$ moves from z_j to an adjacent vertex t. In the last case if $t \neq w_j^{(1)}$ then the robber moves to $w_j^{(1)}$ and he moves to $w_j^{(2)}$ otherwise. The remaining strategy is trivial — he stays in this vertex, since no cop can reach it. Assume that $\mathcal{C}_i$ have moved from z_i to the vertex of X_i adjacent with the robber's position. Then the robber moves to a vertex of Y_{i+1} or to some vertex of $\{C_1, \dots, C_m\}$ if $i = n$.

b) $Q_i = \exists$. The robber is placed on y_i and he stays in his position until the cop $\mathcal{C}_i$ moves from z_i to the vertex of X_i or some other cop $\mathcal{C}_j$ for $j > i$ moves from z_j to an adjacent vertex t. In the last case if $t \neq w_j^{(1)}$ then the robber moves to $w_j^{(1)}$ and he moves to $w_j^{(2)}$ otherwise. The remaining strategy is trivial — he stays in this vertex, since no cop can reach it. Assume that $\mathcal{C}_i$ moves from z_i to a vertex of X_i. If he moves to x_i then we let $x_i = true$ and $x_{i+1} = false$ otherwise. Now the cop-player chooses the value of x_i. Notice that for both values of x_i and for already defined values of $x_1, \dots, x_{i-1}$, $Q_{i+1} x_{i+1} \dots Q_n x_n F = false$. Then the robber moves to a vertex of Y_{i+1} or to some vertex of $\{C_1, \dots, C_m\}$ if $i = n$.

It remains to define the strategy for the case when the robber moves to a vertex of $\{C_1, \dots, C_m\}$. Now we can assume that the variables $x_1, \dots, x_n$ have values for which $F = false$. Hence, there is a clause $C_j = false$. The vertex C_j cannot be occupied by the cops. The robber moves to this vertex and stays there. It remains to observe that no cop can reach C_j. Therefore the robber wins. □

By Lemmata 2 and 3, we have that Cops and Robber is PSPACE-hard for $f \geq 2$.

3.2 Inclusion in PSPACE

To complete the proof of the theorem, it remains to show that our problem is in PSPACE.

Lemma 4. *For every integers $f, k \geq 1$ and an n-vertex graph G, it is possible to decide whether $c_f(G) \leq k$ by making use of space $O(k \cdot f \cdot n^{O(1)})$.*

Proof. The proof is constructive. We describe a recursive algorithm which solves the problem. Note that we can consider only strategies of the cop-player such that at least one cop is moved to an adjacent vertex. Otherwise, if all cops are staying in old positions, the robber can only improve his position.

Our algorithm uses a recursive procedure $W(P, u, l)$, which for a non negative integer l, position of the cops $P = ((v_1, s_1), \ldots, (v_k, s_k))$ such that $l = s_1 + \ldots + s_k$, and a vertex $u \in V(G)$, returns $true$ if k cops can win starting from the position P against the robber which starts from the vertex u, and the procedure returns $false$ otherwise. Clearly, k cops can capture the robber on G if and only if there is an initial position P_0 such that for any $u \in V(G)$, $W(P_0, u, l) = true$ for $l = kf$.

If $l = 0$ then $W(P, u, l) = true$ if and only if $u = v_i$ for some $1 \leq i \leq k$. Suppose that $l > 0$. Then $W(P, u, l) = true$ in the following cases:

- $u = v_i$ for some $1 \leq i \leq k$,
- $u \in N_G(v_i)$ and $s_i > 0$ for some $1 \leq i \leq k$,
- there is a position $P' = ((v'_1, s'_1), \ldots, (v'_k, s'_k))$ such that the cops can go from P to P' in one step, and for any $u' \in N_G[u]$, $W(P', u', l') = true$ where $l' = s'_1 + \ldots + s'_k < l$.

Observe that all positions of the cops can be listed (without storing them) by using polynomial space, and the number of possible moves of the robber is at most n. When the procedure $W(P, u, l)$ is called recursively, we should keep the previous positions of the players which were used to reach the current position. Since the depth of the recursion is at most kf, it can be done in space $O(kfn^{O(1)})$. We conclude that the algorithm uses space $O(kfn^{O(1)})$. □

Observe that our proof shows that the problem is in PSPACE only for $f = n^{O(1)}$. Hence, for the case when f is a part of the input, we can claim only PSPACE-hardness of the problem.

4 Conclusion

In this paper we introduced the variant of the Cops & Robber game with restricted resources and have shown that the problem is PSPACE-complete for every $f > 1$. In fact, our proof also shows that the problem is PSPACE-complete even when f is at most some polynomial of the number of cops. One of the long standing open questions in Cops & Robber games, is the computational complexity of the classical variant of the game on undirected graphs without restrictions on the power of cops. In 1995, Goldstein and Reingold [9] conjectured that this problem is E-hard. On the other hand, we do not know any example, where to win cops are required to make exponential number of steps (or fuel). This lead to a very natural question: How many steps along edges each cop needs in the Cops & Robber game without fuel restrictions?

References

[1] M. Aigner and M. Fromme, *A game of cops and robbers*, Discrete Appl. Math., 8 (1984), pp. 1–11.

[2] B. Alspach, *Searching and sweeping graphs: a brief survey*, Matematiche (Catania), 59 (2006), pp. 5–37.

[3] A. Dumitrescu, I. Suzuki, and P. Zylinski, *Offline variants of the "lion and man" problem*, in Proceedings of the 23d annual symposium on Computational Geometry (SCG 07), New York, NY, USA, 2007, ACM, pp. 102–111.

[4] F. V. Fomin, P. A. Golovach, J. Kratochvíl, N. Nisse, and K. Suchan, *Pursuing a fast robber on a graph*, Theor. Comput. Sci., 411(2010), pp. 1167–1181.

[5] F. V. Fomin, P. A. Golovach, and D. Lokshtanov, *Cops and robber game without recharging*, Proceedings of the 12th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2010), vol. 6139 of Lecture Notes in Computer Science, pp. 273–284, Springer-Verlag, 2010.

[6] F. V. Fomin and D. M. Thilikos, *An annotated bibliography on guaranteed graph searching*, Theor. Comput. Sci., 399 (2008), pp. 236–245.

[7] M. R. Garey and D. S. Johnson, *Computers and intractability*, W. H. Freeman and Co., San Francisco, Calif., 1979. A guide to the theory of NP-completeness, A Series of Books in the Mathematical Sciences.

[8] M. Goldenberg, A. Kovarsky, X. Wu, and J. Schaeffer, *Multiple agents moving target search*, in Proceedings of the 18th international joint conference on Artificial Intelligence (IJCAI 03), San Francisco, CA, USA, 2003, Morgan Kaufmann Publishers Inc., pp. 1536–1538.

[9] A. S. Goldstein and E. M. Reingold, *The complexity of pursuit on a graph*, Theoret. Comput. Sci., 143 (1995), pp. 93–112.

[10] L. J. Guibas, J. claude Latombe, S. M. Lavalle, D. Lin, and R. Motwani, *A visibility-based pursuit-evasion problem*, International Journal of Computational Geometry and Applications, 9 (1996), pp. 471–494.

[11] R. Isaacs, *Differential games. A mathematical theory with applications to warfare and pursuit, control and optimization*, John Wiley & Sons Inc., New York, 1965.

[12] T. Ishida and R. E. Korf, *Moving target search*, in Proceedings of the International joint conference on Artificial Intelligence (IJCAI 91), 1991, pp. 204–211.

[13] T. Ishida and R. E. Korf, *Moving-target search: A real-time search for changing goals*, IEEE Trans. Pattern Anal. Mach. Intell., 17 (1995), pp. 609–619.

[14] J. E. Littlewood, *Littlewood's miscellany*, Cambridge University Press, Cambridge, 1986. Edited and with a foreword by Béla Bollobás.

[15] P. K. K. Loh and E. C. Prakash, *Novel moving target search algorithms for computer gaming*, Comput. Entertain., 7 (2009), pp. 1–16.

[16] N. Megiddo, S. L. Hakimi, M. R. Garey, D. S. Johnson, and C. H. Papadimitriou, *The complexity of searching a graph*, J. Assoc. Comput. Mach., 35 (1988), pp. 18–44.

[17] C. Moldenhauer and N. R. Sturtevant, *Evaluating strategies for running from the cops*, in Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009), 2009, pp. 584–589.

[18] ——, *Optimal solutions for moving target search*, in Proceedings of the 8th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2009), IFAAMAS, 2009, pp. 1249–1250.

[19] R. Nowakowski and P. Winkler, *Vertex-to-vertex pursuit in a graph*, Discrete Math., 43 (1983), pp. 235–239.

[20] A. Quilliot, *Some results about pursuit games on metric spaces obtained through graph theory techniques*, European J. Combin., 7 (1986), pp. 55–66.

[21] J. Sgall, *Solution of David Gale's lion and man problem*, Theor. Comput. Sci., 259 (2001), pp. 663–670.

[22] K. Sugihara, I. Suzuki, and M. Yamashita, *The searchlight scheduling problem*, SIAM J. Comput., 19 (1990), pp. 1024–1040.