A Game of Cops and Robbers on Graphs with Periodic Edge-Connectivity

Thomas Erlebach[0000−0002−4470−5868] and Jakob T. Spooner[0000−0003−3816−6308]

School of Informatics, University of Leicester, Leicester, England
{te17,jts21}@leicester.ac.uk

Abstract. This paper considers a game in which a single cop and a single robber take turns moving along the edges of a given graph G. If there exists a strategy for the cop which enables it to be positioned at the same vertex as the robber eventually, then G is called cop-win, and robber-win otherwise. In contrast to previous work, we study this classical combinatorial game on edge-periodic graphs. These are graphs with an infinite lifetime comprised of discrete time steps such that each edge e is assigned a bit pattern of length l_e, with a 1 in the i-th position of the pattern indicating the presence of edge e in the i-th step of each consecutive block of l_e steps. Utilising the known framework of reachability games, we obtain an $O(\mathsf{LCM}(L) \cdot n^3)$ time algorithm to decide if a given n-vertex edge-periodic graph G^τ is cop-win or robber-win as well as compute a strategy for the winning player (here, L is the set of all edge pattern lengths l_e, and $\mathsf{LCM}(L)$ denotes the least common multiple of the set L). For the special case of edge-periodic cycles, we prove an upper bound of $2 \cdot l \cdot \mathsf{LCM}(L)$ on the minimum length required of any edge-periodic cycle to ensure that it is robber-win, where $l = 1$ if $\mathsf{LCM}(L) \geq 2 \cdot \max L$, and $l = 2$ otherwise. Furthermore, we provide constructions of edge-periodic cycles that are cop-win and have length $1.5 \cdot \mathsf{LCM}(L)$ in the $l = 1$ case and length $3 \cdot \mathsf{LCM}(L)$ in the $l = 2$ case.

1 Introduction

Pursuit-evasion games are games played between two teams of players, who take turns moving within the confines of some abstract arena. Typically, one team – the *pursuers* – are tasked with catching the members of the other team – the *evaders* – whose task it is to evade capture indefinitely. The study of such games has led to their application in a number of real-world scenarios, one widely-studied example of which would be their application to the problem of guiding robots through real-world environments [8]. From a theoretical standpoint, other variants of the game have been studied for their intrinsic links to important graph parameters; for example, in one particular variant in which each pursuer can, in a single turn, move to an arbitrary vertex of the given graph G, it is well known that establishing the minimum number of pursuers it takes to catch one evader also establishes the treewidth of G [20].

The variant most closely resembled by the one considered in this paper was first studied separately by Quilliot [18], and by Nowakowski and Winkler [15], as the discrete *Cops and Robbers* game: One cop (pursuer) and one robber (evader) take turns moving across an edge (or remaining at their current vertex) in a given graph G, with the cop aiming to catch the robber, and the robber attempting to avoid capture. (By 'catching the robber' we mean that the cop occupies the same vertex as the robber.) In this paper, we consider a variant of this game where the game arena is an *edge-periodic* graph [7]. We call this game *Edge-Periodic Cops and Robbers*, or EPCR for short. Such graphs can be thought of as traditional static graphs equipped with an additional function mapping each edge e to a *pattern* of length l_e that dictates in which time steps e is present within each consecutive period of l_e steps. Formal definitions of edge-periodic graphs, which can be seen as a subclass of temporal graphs [14], and EPCR are given in Section 2. As far as we are aware, pursuit-evasion games have not yet been studied in the context of temporal graphs.

Paper Outline and Our Results. The remainder of this section discusses related work. Section 2 gives preliminaries. In Section 3, we consider the problem of deciding, given an edge-periodic graph G^τ, whether a game of edge-periodic cops and robbers played on G^τ is won by the cop or won by the robber. We exploit the connection (which was previously noted, e.g., in [11]) between the game of cops and robbers and *reachability games* to solve the one cop, one robber variant of cops and robbers on edge-periodic graphs. Our algorithm runs in polynomial-time whenever the lowest common multiple of the lengths of each edges appearance-pattern is n and $\max L$; we remark, however, that the algorithms has exponential running-time in the worst-case (more in Section 5). In Section 4, we consider edge-periodic graphs whose underlying graph is a cycle. We prove an upper bound of $2 \cdot l \cdot \mathsf{LCM}(L)$ on the length required of any such cycle C^τ in order to guarantee that it is robber-win, where $l = 1$ if $\mathsf{LCM}(L) \geq 2 \cdot \max L$, and $l = 2$ otherwise. Here, L is the set of the lengths of the bit patterns assigned to the edges of the cycle, and $\mathsf{LCM}(L)$ their least common multiple. We also give lower bound constructions showing that there exist cop-win edge-periodic cycles of length $\frac{3}{2} \cdot \mathsf{LCM}(L)$ and $3 \cdot \mathsf{LCM}(L)$ in the $l = 1$ and $l = 2$ case, respectively. Section 5 concludes the paper.

Related Work. The introduction of pursuit-evasion type combinatorial games is most often attributed to Parsons, who studied a problem in which a team of rescuers search for a lost spelunker in a circular cave system [16]. By representing the cave as a cycle graph, he showed that one rescuer is not enough to guarantee that the spelunker is found, but that two are. In a similar vein, the *Cop and Robber* problem, in which one cop attempts to catch a robber in a given graph G, was introduced independently by Quilliot [18], and by Nowakowski and Winkler [15]. Their papers characterise precisely those graphs for which one cop is enough to guarantee that the robber is caught. Aigner and Fromme [1] considered a generalised variant of the game, in which k cops attempt to catch a single robber; their paper introduced the notion of the *cop-number* of a graph, i.e., the minimum number of cops required to guarantee that the robber is caught.

Reductions from the standard game of cops and robbers to a game played on a directed graph, and algorithms that can decide, for a given graph, whether cop or robber wins, were given in [10, 4, 2]. Kehagias and Konstantinidis [11] note a connection between these approaches and reachability games. Reachability games are a well-studied class of 2-player *token-pushing* games, in which two players push a token along the edges of a directed graph in turn – one with the aim to push the token to some vertex belonging to a prespecified subset of the graph's vertex set, and the other with the aim to ensure the token never reaches such a vertex [9]. The winner of a reachability game played on a given directed graph G can be established in polynomial time [3, 9]. For more information regarding cops and robbers/pursuit-evasion games, as well as their connection to reachability games, we refer the reader to [9, 3, 11, 12, 5, 17, 8].

In this paper, we consider the game of cops and robbers within the context of *temporal graphs*. Temporal graphs are a relatively new object of interest, and incorporate an aspect of time-variance into the combinatorial structure of traditional static graphs [14]. One previously considered way of viewing a temporal graph $\mathcal{G}$ is as a sequence of L subgraphs of a given *underlying graph* G (where L is the *lifetime* of the graph) [13], with each subgraph indexed by the time steps $t \in [L]$. For problems within this model, it is often natural to assume that each subgraph G_t in all time steps $t \in [L]$ is connected [13]. The edge-periodic graphs considered in this paper differ in that this connectivity assumption is dropped – similar graphs were introduced in [7]. For further related work on temporal graphs, we refer the reader to, e.g., [6, 13, 14].

2 Graph Model and Game Rules

For any positive integer k we write $[k]$ for the set $\{0, 1, \dots, k-1\}$.

Definition 1 (Edge-periodic graph G^τ). *An edge-periodic graph $G^\tau = (V, E, \tau)$ is a temporal graph with underlying (directed or undirected) graph $G = (V, E)$ and infinite lifetime, and an additional function $\tau : E \to \{0,1\}^*$ that maps each edge $e \in E$ to a pattern $\tau(e) = b_e(0)b_e(1)\cdots b_e(l_e - 1)$ of length $l_e > 0$. Each $\tau(e)$ consists of l_e Boolean values, such that e is present in a time step $t \geq 0$ if and only if $b_e(t \bmod l_e) = 1$. We can assume that for any edge $e \in E(G^\tau)$, $b_e(i) = 1$ for at least one $i \in [l_e]$, so that every edge e is present at least once in any period of l_e time steps.*

For a given temporal graph $G^\tau = (V, E, \tau)$, we refer to the length l_e of the bit pattern assigned to edge e as the *period* of e. Furthermore, we use $L = \{l_e : e \in E\}$ to denote the set of all edge periods and $\mathsf{LCM}(L)$ to denote the least common multiple of the elements in L. When the set L is clear from the context, we omit it from the notation, writing LCM in place of $\mathsf{LCM}(L)$.

We consider a game of cops and robbers identical in its rule set to the one introduced in [18] and [15] (in particular, the variant with 1 cop and 1 robber), but with edge-periodic graphs as the game arenas. We call the resulting game *edge-periodic cop(s) and robber(s)*, or EPCR for short. In this paper we only consider the undirected case, but all results translate to the directed case easily.

Rules of EPCR. Initially, the two players (cop C and robber R) each select a start vertex on a given edge-periodic graph G^τ. C chooses first, followed by R, whose choice is made in full knowledge of C's choice. After the start vertices have been chosen, in each time step $t \geq 0$, players take alternating turns moving over an edge in the graph that is incident to their current vertex or choosing to remain at their current vertex, following the convention that in any particular time step, C moves first, in full knowledge of R's position, followed by R; again, R's move is made with full knowledge of the move that C just made. Whenever C or R are situated at a vertex $v \in V(G^\tau)$ during some time step t and it is their turn to make a move, they may only traverse those edges $\{v, u\}$ with $b_{\{v,u\}}(t \bmod l_{\{v,u\}}) = 1$. The game terminates only when, at the end of either player's move, C and R are situated at the same vertex in G^τ. If there exists a strategy for C that ensures that the game terminates, we say that G^τ is *cop-win*. Otherwise, there must exist a strategy for R that enables infinite evasion of C; in this case we call G^τ *robber-win*.

3 Determining the Winner of a Game of EPCR

In this section, we prove the following theorem:

Theorem 1. *Let G^τ be an edge-periodic graph with n nodes, and let $L = \{l_e : e \in E(G^\tau)\}$. Then, it can be decided in $O(\mathsf{LCM} \cdot n^3)$ time whether G^τ is cop-win or robber-win. A winning strategy for the winning player can be computed in the same time bound.*

The proof mainly uses a transformation from a given edge-periodic graph G^τ to a finite directed graph G'. The transformation is such that the playing of an instance of EPCR on G^τ is essentially equivalent to the playing of a *reachability game* on G'. For this, we need a way of translating a particular state of an instance of EPCR played on G^τ to a corresponding state in the reachability game played on G'. The following definition introduces the notion of a *position* that represents the current state in a game of EPCR on an edge-periodic graph G^τ.

Definition 2 (Position in G^τ). *A* position *of a game of EPCR played on an edge-periodic graph G^τ is a 4-tuple $P = (c_P, r_P, s_P, t_P)$, where $c_P \in V(G^\tau)$ is C's current vertex, $r_P \in V(G^\tau)$ is R's current vertex, $s_P \in \{\mathsf{C}, \mathsf{R}\}$ is the player whose turn it is to move next, and t_P is the current time step.*

We call any position P such that $c_P = r_P$ a *terminating position*, since this indicates that both players are situated on the same vertex and hence C has won. Next, we formally introduce *reachability games* [9]:

Definition 3 (Reachability game G'). *A reachability game is a directed graph G', given as a 3-tuple*

$$G' = (V_0 \cup V_1, E', F),$$

where $V_0 \cup V_1$ is a partition of the node set V' (also referred to as the state set); $E' \subseteq V' \times V'$ is a set of directed edges; and $F \subseteq V'$ is a set of final states.

The game is played by two opposing players, Player 0 and Player 1; V_0 and V_1 are the (disjoint) sets of Player 0/Player 1 owned nodes, respectively. A token is placed at some initial vertex v_0 at the start of the game. Depending on whether $v_0 \in V_0$ or $v_0 \in V_1$, the corresponding player then selects one of the outgoing edges of v_0 and pushes the token along that edge. When the token arrives at the next vertex, the player who owns that vertex then selects an outgoing edge and pushes the token along it. This process continues, and such a sequence of moves constitutes a *play* of the reachability game on G'. Formally, a play $\phi = v_0, v_1, ...$ is a (possibly infinite) sequence of vertices in V', such that $(v_i, v_{i+1}) \in E'$ for all $i \geq 0$. We say that a play ϕ is *won* by Player 0 if there exists some i such that $v_i \in F$. Otherwise, ϕ is of infinite length and for no i is $v_i \in F$, and ϕ is won by Player 1.

3.1 Transformation

We now detail our transformation from a given edge-periodic graph G^τ to a reachability game $G' = (V', E', F)$.

State set V'. We define the state set (i.e., vertex set) of the directed graph G' to be a set of 4-tuples, each corresponding to a position in the game of EPCR on G^τ as follows:

$$V' = \{(c, r, s, t) : c, r \in V(G^\tau), s \in \{\mathsf{C}, \mathsf{R}\}, t \in [\mathsf{LCM}]\}.$$

Let $V_0 = \{(c, r, s, t) \in V' : s = \mathsf{C}\}$ and $V_1 = \{(c, r, s, t) \in V' : s = \mathsf{R}\}$ be the sets of Player 0 (or C) owned nodes, and Player 1 (or R) owned nodes, respectively. We can restrict the range of t to [LCM] without losing information because all edge periods divide LCM and hence the set of edges present at any time t is the same as the set of edges present at time $t \bmod \mathsf{LCM}$.

Edge set E'. In order to construct the edge set $E' \subseteq (V_0 \times V_1) \cup (V_1 \times V_0)$ we include the edge (S, S') for $S = (c, r, s, t)$ and $S' = (c', r', s', t')$ in E' if and only if the following conditions are satisfied:

$$\begin{aligned}
&(1)\ \ s = \mathsf{C} \implies \big(c = c' \vee (\{c, c'\} \in E(G^\tau) \text{ and } b_{\{c,c'\}}(t \bmod l_{\{c,c'\}}) = 1)\big) \\
&\qquad\qquad\qquad \wedge (r = r') \wedge (t' = t) \wedge (s' = \mathsf{R}), \\
&(2)\ \ s = \mathsf{R} \implies \big(r = r' \vee (\{r, r'\} \in E(G^\tau) \text{ and } b_{\{r,r'\}}(t \bmod l_{\{r,r'\}}) = 1)\big) \\
&\qquad\qquad\qquad \wedge \big(c = c'\big) \wedge \big(t' = (t + 1) \bmod \mathsf{LCM}\big) \wedge (s' = \mathsf{C}).
\end{aligned}$$

Condition (1) ensures that C can only stay at a vertex or move over an adjacent edge that is present in every time step t'' with $t'' \bmod \mathsf{LCM} = t$, and that the next state will be a state in the same time step where R has to move. Condition (2) is the analogous condition for R, but the next state will be in the following time step (modulo LCM) and C will have to move next.

Set of final states F. Let $F = \{(c, r, s, t) \in V' : c = r\}$, so that the set of final states consists of all states that correspond to a position in G^τ where C is positioned on the same vertex as R (i.e., where C has won the game).

3.2 Proof of Theorem 1

We first introduce the elements of the theory of reachability games that are required for the proof of Theorem 1, starting with the definition of the *attractor set*:

Definition 4 (Attractor set $Attr(F)$ [3]). *The sequence $(Attr_i(F))_{i\geq 0}$ is recursively defined as follows:*

$$\begin{aligned} Attr_0(F) &= F \\ Attr_{i+1}(F) &= Attr_i(F) \cup \{v \in V_0 \mid \exists (v,u) \in E' : u \in Attr_i(F)\} \cup \\ &\quad \{v \in V_1 \mid \forall (v,u) \in E' : u \in Attr_i(F)\} \end{aligned}$$

The sets $Attr_i(F)$ are a sequence of subsets of V' that is monotone with respect to set-inclusion. Let

$$Attr(F) = \bigcup_{i\geq 0} Attr_i(F).$$

Since G' is finite, $Attr(F)$ is the least fixed point of the sequence $(Attr_i(F))_{i\geq 0}$.

Intuitively, the states in $Attr(F)$ are the states from which Player 0 can win the game. For $x \in \{0,1\}$, a *memoryless* strategy of Player x is a partial function $\sigma_x : V_x \to V'$ that specifies for each state in V_x (except states in $V_x \cap F$) the state to which Player x pushes the token from that state. The strategy is called memoryless because the move a player selects only depends on the current state, not on the history of the game. A winning strategy of Player 0 from any state in $Attr(F)$ consists of selecting for each state u in $(Attr_{i+1}(F) \setminus Attr_i(F)) \cap V_0$, for any $i \geq 0$, an arbitrary outgoing edge leading to a state in $Attr_i(F)$. The states in $Attr_i(F)$, for any $i \geq 0$, have the property that Player 0 wins the game after at most i further moves (in total for both players) when following that strategy. Similarly, $V' \setminus Attr(F)$ is the set of states from which Player 1 can win the game. The winning strategy for Player 1 from any such state consists of selecting for each state u in $V_1 \setminus Attr(F)$ an arbitrary outgoing edge leading to a state that is not in $Attr(F)$. These winning strategies are memoryless.

Theorem 2 (Berwanger [3], Grädel et al.[9]). *In a given reachability game $G' = (V', E', F)$, Player 0 has a winning strategy from any state $S \in Attr(F)$, and Player 1 has a winning strategy from any state $S \in V' \setminus Attr(F)$. There exists an algorithm which computes the set $Attr(F)$ and a memoryless winning strategy for the winning player in time $O(|V'| + |E'|)$.*

Our transformation produces, from a given edge-periodic graph G^τ, a directed graph $G' = (V', E', F)$ such that there is a correspondence between positions in the game of EPCR on G^τ and states in V'. Let $Attr(F)$ be the attractor set for G'. Winning strategies for G' translate directly into winning strategies for EPCR on G^τ from any winning position by moving according to the outgoing edges chosen by the winning strategy in G'. Using the notation S_P to refer to the state in V' that corresponds to the position P in the game of EPCR on G^τ, Theorem 2 then implies the following:

Lemma 1. *C can force a win from a position P if and only if the state $S_P \in V'$ satisfies $S_P \in Attr(F)$. Starting from a position P such that $S_P \notin Attr(F)$, R can force the sequence of moves to never reach any state $S \in F$, and, as such, the EPCR game can be won by R.*

Lemma 2. *An edge-periodic graph G^τ is cop-win if and only if there exists a vertex $v \in V(G^\tau)$ such that $(v, r, \mathsf{C}, 0) \in Attr(F)$ for all $r \in V(G^\tau)$.*

Proof. ($\Rightarrow$) Assume not, so that G^τ is cop-win but there exists no vertex $v \in V(G^\tau)$ such that $(v, r, \mathsf{C}, 0) \in Attr(F)$ for all $r \in V(G^\tau)$. Then for every start vertex c that C can choose, there exists at least one vertex u such that the state $(c, u, \mathsf{C}, 0) \notin Attr(F)$. Let R choose such a vertex u as its start vertex. Since R chooses u in full knowledge of C's choice of c, it follows that R can force the equivalent reachability game on G' to begin from a state $S_{(c,u,\mathsf{C},0)} \notin Attr(F)$, hence winning the reachability game regardless of C's choice of c. This is a contradiction since, by assumption, G^τ is cop-win.

($\Leftarrow$) If C chooses a vertex v with the stated property as its initial vertex, the resulting position P will correspond to a state $S_P \in Attr(F)$ no matter which vertex R chooses as its initial vertex, and by Lemma 1 C has a winning strategy. $\square$

Proof (of Theorem 1). Since $n = |V(G^\tau)|$, our transformation produces, given an edge-periodic graph G^τ, a directed graph $G' = (V', E', F)$, such that $|V'| = O(\mathsf{LCM} \cdot n^2)$. This is because V' contains tuples (c, r, s, t) for n choices of c, n choices of r, two choices of s, and LCM choices of t. Next, note that each state $S_P \in V'$ has at most n outgoing edges because the player whose turn it is can only stay at its vertex or move to one of at most $n-1$ neighbouring vertices. It follows that $|E'| = O(\mathsf{LCM} \cdot n^3)$. Furthermore, the transformation can be done in $O(|V'| + |E'|) = O(\mathsf{LCM} \cdot n^3)$ time.

By Theorem 2, the attractor set $Attr(F)$ of G' can be computed in time $O(\mathsf{LCM} \cdot n^3)$. By Lemma 2, we can then determine whether G^τ is cop-win by checking if there exists at least one vertex $c \in V(G^\tau)$ such that $(c, r, \mathsf{C}, 0) \in Attr(F)$ for all $r \in V(G^\tau)$: if such a c exists, G^τ is cop-win, otherwise it is robber-win. This check can be done in $O(n^2)$ time.

By Theorem 2, we also obtain a memoryless winning strategy σ_0 for $\mathsf{Player\ 0}$ from all states in $Attr(F)$, and a memoryless winning strategy σ_1 for $\mathsf{Player\ 1}$ from all states in $V' \setminus Attr(F)$, in $O(|V'| + |E'|)$ time. If G^τ is cop-win, we obtain a winning-strategy for C by letting C choose as its initial vertex any vertex satisfying the condition of Lemma 2 and then behave in line with σ_0: When it is $\mathsf{C}'s$ turn in a current position $P = (c_P, r_P, \mathsf{C}, t_P)$, C constructs from it the state S_P, looks up the state $\sigma_0(S_P) = (c', r', \mathsf{R}, t_P \bmod \mathsf{LCM})$, and moves to c' (or stays at c_P if $c_P = c'$). Similarly, if G^τ is robber-win, we obtain a winning-strategy for R by letting R choose its initial vertex r (in response to C's choice of its initial vertex c) in such a way that $S_P \notin Attr(F)$ for $P = (c, r, \mathsf{C}, 0)$ and then behave in line with σ_1. $\square$

We remark that, as long as LCM is polynomial in n and $\max L$, the winner of EPCR on a given graph G^τ can be determined in polynomial time. In particular, if the periods l_e are bounded by some constant for all $e \in E(G^\tau)$, the winner can be determined in $O(n^3)$ time.

Finally, we note that Theorem 1 can be generalised to a setting with k cops at the expense of increasing the algorithm's running time to $O(\mathsf{LCM} \cdot k \cdot n^{k+2})$. The idea is to fix an arbitrary ordering of the cops and create $k+1$ layers of states during every time step $t \in [\mathsf{LCM}]$ (one for each of the k cops' moves, followed finally by the robber's move). By allowing the players to play their moves in each time step in this serialised fashion the resulting game graph requires $O(\mathsf{LCM} \cdot k)$ layers with n^{k+1} states in each, with at most n edges leading from every state to states in the following layer.

4 An Upper Bound on the Length Required to Ensure an Edge-Periodic Cycle is Robber-Win

In this section, we consider edge-periodic cycles, a restricted subclass of edge-periodic graphs where the underlying graph is a cycle. We are interested in how long (in terms of number of edges) the cycle needs to be to ensure that the robber can escape the cop indefinitely. First, we show that any edge-periodic infinite path for which the set L of its edge periods is finite is robber-win. After this, we show how the strategy for such infinite paths can be adapted to the cycle case. Let the given edge periodic cycle be $C^\tau = (V, E, \tau)$, and let $L = \{l_e : e \in E\}$ denote the set of edge periods. In the remainder of this section, we write LCM as short-hand for LCM.

We first consider infinite paths, which will later allow us to handle the case in which the cop chases the robber around the cycle in a fixed direction.

Lemma 3. *Let P be an infinite edge-periodic path, $L = \{l_e : e \in E(P)\}$, and assume that $|L|$ is finite. Then, starting from any time step t, there exists a winning strategy for R from any vertex with distance at least $2 \cdot \mathsf{LCM}$ from C's start vertex if $\mathsf{LCM} = \max L$, and with distance at least LCM otherwise.*

Proof. First, notice that since we assume that $|L|$ is finite, so must be LCM. Let C's vertex at the start of time step t be $c_t \in P$. Denote R's initial vertex by r_t, and assume without loss of generality that r_t is a vertex in P that lies to the right of c_t. Assume from now onward that P is a path starting at c_t and extending infinitely to the right, and that C moves right whenever possible (it is clear that this is the best strategy for capturing R).

Consider the set L and its constituent elements. Either (1) there exists $x \in L$ such that $\max L$ is not a multiple of x – then $\mathsf{LCM} \geq 2 \cdot \max L$, since it cannot be the case that $\mathsf{LCM} = j \cdot \max L$ for any $j < 2$; or (2) for every $x \in L$, $\max L = x \cdot i$ for some integer $i \geq 1$; then $\mathsf{LCM} = \max L$. With this in mind, define $B = \mathsf{LCM}$ if (1) holds and $B = 2 \cdot \mathsf{LCM}$ if (2) holds. Now, let us define the *strips* S_i ($i \geq 1$) to be finite subpaths of P, such that for all edges $e \in S_i$, e is first traversed by C in some time step $t_e \in [t + (i-1)B, t + iB - 1]$ (assuming that C moves right whenever it can). Note that $B \geq 2 \cdot \max L$ and hence each S_i must contain at least two edges. By convention, we call the leftmost and rightmost edges (vertices) of any S_i its *first* and *last* edges (vertices), respectively. Note also that the last vertex of S_i and the first vertex of S_{i+1} are one and the same, for all $i \geq 1$.

Note that the first vertex of S_2 is at most B edges away from c_t. By the condition of the lemma, R is located at least B edges away from c_t. For the remainder of the analysis, we assume that R is located at the first vertex of S_2 and moves right whenever possible. If R can escape C indefinitely under this assumption, it is clear that R can also do so if it starts further to the right.

We now demonstrate that R wins the game. Note that the set of edges that are present in each step repeats every B time steps as LCM divides B. Thus, we have that C and R traverse strips in a synchronised fashion: For any $i \geq 1$, during the interval $[t+(i-1)B, t+iB-1]$ of time steps, C traverses S_i and R traverses S_{i+1}. The only possibility for C to catch R would be for C to reach the last vertex of S_i before R leaves the first vertex of S_{i+1}. However, C reaches the last vertex of S_i in a time step $t' = t + iB - j$ for some $1 \leq j \leq \max L$, as the last edge of S_i is available at least once in $\max L$ consecutive time steps. On the other hand, R leaves the first vertex of S_{i+1} in a time step $t'' = t + (i-1)B + j'$ for some $0 \leq j' < \max L$. As $B \geq 2 \max L$, it follows that $t' > t''$, showing that C cannot catch R. □

Theorem 3. *Let $C^\tau = (V, E, \tau)$ be an edge-periodic cycle on n vertices and $L = \{l_e : e \in E\}$. If $n \geq 2 \cdot l \cdot \mathsf{LCM}(L)$, then C^τ is robber-win (where $l = 1$ if $\mathsf{LCM}(L) \geq 2 \cdot \max L$, and $l = 2$ otherwise).*

Proof. For any $t \geq 0$, we let c_t and r_t denote the vertex at which C and R are positioned at the start of time step t, respectively. Consider now some edge $e \in E(C^\tau)$ and classify its vertices as a 'left' and 'right' vertex arbitrarily; let the left vertex of each edge be the right vertex of the following edge in the cycle. We proceed by specifying a strategy for R. Initially, let C choose c_0; R chooses r_0 to be the vertex antipodal to c_0 in C^τ. (If n is odd then R selects r_0 to be either of the two vertices that are furthest away from c_0; we will refer to both these vertices as antipodal to c_0, and treat vertices in all steps $t \geq 0$ in the same way.) We now distinguish between two modes of play, *Hide* and *Escape*, and specify R's strategy in each of them.

Hide mode: A *Hide period* begins in step 0 and in any step $t \geq 2$ such that c_t and r_t are antipodal, but c_{t-1} and r_{t-1} were not. Note that any game in which R follows our strategy begins in a Hide period. The Hide period beginning at step t is the interval $[t, t+x]$ such that $c_{t'}$ and $r_{t'}$ are antipodal for all $t' \in [t, t+x]$, but c_{t+x+1} and r_{t+x+1} are not. (If no such step $t+x+1$ exists, the Hide period is $[t, \infty)$.) Any Hide period (except if it is of the form $[t, \infty)$) is followed directly by an Escape period, which will start in step $t+x+1$.

R's **Hide strategy**: If the game is in a Hide period during step t, R observes C's choice of c_{t+1} and tries to move to (or stay at) a vertex antipodal to it. Clearly, R cannot be caught in any step belonging to a Hide period, as regardless of whether $\mathsf{LCM} = \max L$ or $\mathsf{LCM} \geq 2 \cdot \max L$, we have that $n \geq 4 \cdot \max L \geq 4$. As a result, antipodal vertices in C^τ are at least distance 2 from one another.

Escape mode: An *Escape period* always begins in a step t such that step $t-1$ was the last step of some Hide period. As such, an Escape period is an interval $[t, t+x]$ such that $c_{t'}$ and $r_{t'}$ are not antipodal for any $t' \in [t, t+x]$, but c_{t+x+1} and r_{t+x+1} are. The last step of the Escape period is then $t+x$, and the first step of the next Hide period is $t+x+1$. If there is no step $t+x+1$ in which c_{t+x+1} and r_{t+x+1} are antipodal, the Escape period is $[t, \infty)$.

R's **Escape strategy**: Assume that some Escape period starts in step t. Then, at the start of step $t-1$, c_{t-1} and r_{t-1} were antipodal to one another, and during step $t-1$, we had a situation in which C was able to move towards R in some direction, but the edge incident to r_{t-1} leading in the same direction was not present. Now, recall that if $l = 2$, so that $\mathsf{LCM} = \max L$, then $n \geq 4 \cdot \mathsf{LCM}$; and if $l = 1$, so that $\mathsf{LCM} \geq 2 \cdot \max L$, then $n \geq 2 \cdot \mathsf{LCM}$. Therefore, since c_{t-1} and r_{t-1} are antipodal in C^τ , if $l = 2$ holds we have that the distance between them is at least $2 \cdot \mathsf{LCM}$ and if $l = 1$ holds, the distance between them is at least LCM. Observe now that we are able to view any edge-periodic cycle of finite length as an infinite path whose edge patterns repeat infinitely often. We can thus view the Escape period as an instance of the game on an infinite edge-periodic path starting at time step $t-1$, to which Lemma 3 applies. Hence, R can evade C until the Escape period ends (or indefinitely, in case the Escape period never ends).

Since every step t belongs to either a Hide period or an Escape period, we have shown that C can never catch R, and the proof is complete. □

We now give lower bounds on the length required of a strictly edge-periodic cycle to ensure that it is robber-win.

Theorem 4. *There exists an edge-periodic cycle of length* $3 \cdot \mathsf{LCM}$ *with edge pattern lengths in the set* L *that is both cop-win and satisfies* $\mathsf{LCM} = \max L$.

Proof. Let $M > 1$ be an integer and consider an edge-periodic cycle C with $3M$ edges and with edge pattern lengths in $L = \{1, M\}$. Let two consecutive edges have patterns 0...01 of length M, and all $3M-2$ remaining edges have pattern 1. We refer to the subpath of C consisting of the two edges with period M as the M-path, and the subpath with edges labelled with 1 as the 1-path.

We now specify a strategy for C and show that it is in fact a winning strategy: Let C position itself initially at either of the two vertices belonging to the 1-path that are distance $M-1$ from one extreme point of the M-path, and $2M-1$ from the opposite extreme point (where distance is taken to mean the length of the path to that extreme point that avoids the edges of the M-path). Call that chosen vertex c_0, and notice that it splits the 1-path into two subpaths that intersect only in c_0 – one of length $M-1$ which we will call P^-, the other of length $2M-1$ which we call P^+. If R chooses its initial position to be some vertex lying on P^-, then C can move along all edges of P^- in the first $M-1$ steps and catch R. If R chooses its initial position as some vertex lying on P^+, then in the first $2M-1$ steps C can traverse all edges of P^+. The only way for R to leave P^+ without encountering C is via the M-path. R can traverse only one edge of the M-path (in the M-th step) and will be stuck at the middle vertex of the M-path until the $2M$-th step (i.e., until time step $2M-1$). In step $2M-1$, C will be positioned at the vertex that lies on both the M-path and P^+. C will move first and catch R. It remains to be shown that R will be caught if it chooses the middle vertex of the M-path as its start vertex: here, R will not be able to move until step $M-1$, so C can traverse all edges of P^- in the first $M-1$ steps and then, in step $M-1$, catch R before R can make its move. □

A small amount of modification to the construction in the proof of Theorem 4 yields the following lower bound for the case when $\mathsf{LCM} \geq 2 \cdot \max L$:

Theorem 5. *There exists an edge-periodic cycle of length* $1.5 \cdot \mathsf{LCM}$ *with edge periods in the set* L *that is both cop-win and satisfies* $\mathsf{LCM} \geq 2 \cdot \max L$.

Proof. Perform the construction from the proof of Theorem 4, taking $M > 1$ to be odd. Again let C select one of the vertices that has distance $M-1$ and $2M-1$ from opposite ends of the M-path as its start vertex, calling that vertex x. Consider the strategy from the proof of Theorem 4 and observe that there are two edges that C may cross in the second step. Select either one of these edges and replace its pattern of 1 with the pattern 01 (with period 2). C can now follow the strategy in the proof of Theorem 4 – this works since the edge with pattern 01 has been selected so that it is present whenever C's strategy crosses that edge. Since M is odd, we have that $\mathsf{LCM} = 2M$. Since the constructed cycle has length $3M$, the theorem follows. □

5 Conclusion

We have introduced a cops and robbers game on edge-periodic graphs and shown that there exists an algorithm with running time $O(\mathsf{LCM} \cdot n^3)$ that decides whether the cop or robber wins and computes a winning strategy for the winning player. The running-time of the algorithm is polynomial if $\mathsf{LCM}(L)$ is polynomial in n and $\max L$. A natural open question is: What is the complexity of deciding whether cop or robber wins when the least common multiple of the edge periods is exponential in the size of the input? We note that $\mathsf{LCM}(\{1, ..., n\}) = \mathsf{e}^{\phi(n)}$, where $\phi(n) \in \Theta(n)$ is Chebyshev's function [19], and thus there are edge-periodic graphs where the running-time of our algorithm is exponential. It would be interesting to establish whether there exists a better algorithm or whether the problem is *NP*-hard for this case. More generally, one could also examine the cops and robbers game within the context of other temporal graph models. It would also be interesting to reduce the gap between our upper and lower bounds on the minimum length required of an edge-periodic cycle to be guaranteed to be robber-win.

Acknowledgements

The authors would like to thank Maciej Gazda for helpful discussions regarding reachability games, as well as an anonymous reviewer for a suggestion leading to the running-time for the variant with k cops mentioned at the end of Section 3.

References

1. Aigner, M., Fromme, M.: A game of cops and robbers. Discrete Applied Mathematics **8**(1), 1 – 12 (1984). https://doi.org/10.1016/0166-218X(84)90073-8
2. Berarducci, A., Intrigila, B.: On the cop number of a graph. Advances in Applied Mathematics **14**(4), 389 – 403 (1993). https://doi.org/10.1006/aama.1993.1019
3. Berwanger, D.: Graph games with perfect information. arXiv:1407.1647 (2013)
4. Bonato, A., MacGillivray, G.: A general framework for discrete-time pursuit games. Unpublished manuscript (2015)
5. Bonato, A., Nowakowski, R.: The Game of Cops and Robbers on Graphs, Student Mathematical Library, vol. 61. American Mathematical Society (2011). https://doi.org/10.1090/stml/061
6. Casteigts, A.: A Journey Through Dynamic Networks (with Excursions). Habilitation à diriger des recherches, University of Bordeaux (June 2018), https://tel.archives-ouvertes.fr/tel-01883384
7. Casteigts, A., Flocchini, P., Quattrociocchi, W., Santoro, N.: Time-varying graphs and dynamic networks. In: Frey, H., Li, X., Ruehrup, S. (eds.) ADHOC-NOW 2011. LNCS, vol. 6811, pp. 346–359. Springer, Heidelberg (2011)
8. Chung, T.H., Hollinger, G.A., Isler, V.: Search and pursuit-evasion in mobile robotics. Autonomous Robots **31**(4), 299–316 (2011). https://doi.org/10.1007/s10514-011-9241-4
9. Grädel, E., Thomas, W., Wilke, T. (eds.): Automata Logics, and Infinite Games: A Guide to Current Research. Springer, New York, NY, USA (2002)
10. Hahn, G., MacGillivray, G.: A note on k-cop, l-robber games on graphs. Discrete Mathematics **306**(19), 2492–2497 (2006). https://doi.org/10.1016/j.disc.2005.12.038, Creation and Recreation: A Tribute to the Memory of Claude Berge
11. Kehagias, A., Konstantinidis, G.: Cops and robbers, game theory and Zermelo's early results. arXiv:1407.1647 (2014)
12. Kehagias, A., Mitsche, D., Pralat, P.: The role of visibility in pursuit/evasion games. Robotics **4**, 371–399 (2014)
13. Michail, O.: An introduction to temporal graphs: An algorithmic perspective. Internet Mathematics **12**(4), 239–280 (2016). https://doi.org/10.1080/15427951.2016.1177801
14. Michail, O., Spirakis, P.G.: Elements of the theory of dynamic networks. Communications of the ACM **61**(2), 72–72 (Jan 2018). https://doi.org/10.1145/3156693
15. Nowakowski, R., Winkler, P.: Vertex-to-vertex pursuit in a graph. Discrete Mathematics **43**(2), 235–239 (1983). https://doi.org/10.1016/0012-365X(83)90160-7
16. Parsons, T.D.: Pursuit-evasion in a graph. In: Alavi, Y., Lick, D.R. (eds.) Theory and Applications of Graphs. pp. 426–441. Springer Berlin Heidelberg (1978)
17. Patsko, V., Kumkov, S., Turova, V.: Pursuit-evasion games. In: Basar, T., Zaccour, G. (eds.) Handbook of Dynamic Game Theory, pp. 1–87. Springer International Publishing (2017). https://doi.org/10.1007/978-3-319-27335-8-30-1
18. Quilliot, A.: Jeux et pointes fixes sur les graphes. Ph.D. thesis, University of Paris VI (1978)
19. Rankin, B.A.: Ramanujan: Twelve lectures on subjects suggested by his life and work. The Mathematical Gazette **45**(352), 166 (1961). https://doi.org/10.1017/S0025557200044892
20. Seymour, P., Thomas, R.: Graph searching and a min-max theorem for tree-width. Journal of Combinatorial Theory, Series B **58**(1), 22–33 (1993). https://doi.org/10.1006/jctb.1993.1027