A game of cops and robbers on graphs with periodic edge-connectivity

Thomas Erlebach Jakob Spooner

Department of Informatics

UNIVERSITY OF LEICESTER

Algorithmic Aspects of Temporal Graphs II
ICALP 2019 Satellite Workshop
Patras, Greece, 8 July 2019

Cops and Robbers

- One or several cops chase a robber in a graph
- Also known as pursuit-evasion games
- Many variations:
  - Move along edges or arbitrarily
  - Knowledge about position of opponent
  - Turn-based or simultaneous moves
  - ...
- Some variants relate to graph parameters such as treewidth

Discrete Cop and Robber Game

- Undirected graph $G = (V, E)$. One cop, one robber.

Discrete Cop and Robber Game

- Undirected graph $G = (V, E)$. One cop, one robber.
- Cop chooses its start vertex.

Discrete Cop and Robber Game

- Undirected graph $G = (V, E)$. One cop, one robber.
- Cop chooses its start vertex.
- Then robber chooses its start vertex.

Discrete Cop and Robber Game

- Undirected graph $G = (V, E)$. One cop, one robber.
- Cop chooses its start vertex.
- Then robber chooses its start vertex.
- Then cop and robber take turns. In each turn, can stay at a vertex or move over an edge to an adjacent vertex.

Discrete Cop and Robber Game

- Undirected graph $G = (V, E)$. One cop, one robber.
- Cop chooses its start vertex.
- Then robber chooses its start vertex.
- Then cop and robber take turns. In each turn, can stay at a vertex or move over an edge to an adjacent vertex.
- Full knowledge about other player's position.

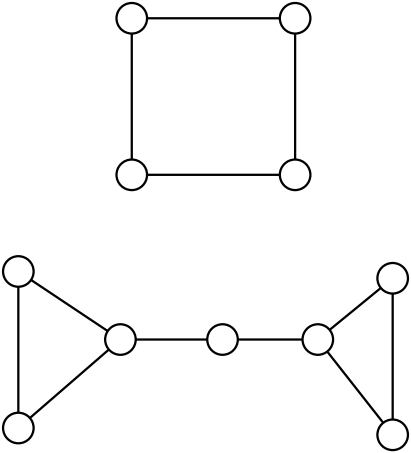
Discrete Cop and Robber Game

- Undirected graph $G = (V, E)$. One cop, one robber.
- Cop chooses its start vertex.
- Then robber chooses its start vertex.
- Then cop and robber take turns. In each turn, can stay at a vertex or move over an edge to an adjacent vertex.
- Full knowledge about other player's position.
- Game ends when cop and robber are on the same vertex (cop wins), or continues indefinitely (robber wins).

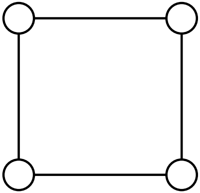
Discrete Cop and Robber Game

- Undirected graph $G = (V, E)$. One cop, one robber.
- Cop chooses its start vertex.
- Then robber chooses its start vertex.
- Then cop and robber take turns. In each turn, can stay at a vertex or move over an edge to an adjacent vertex.
- Full knowledge about other player's position.
- Game ends when cop and robber are on the same vertex (cop wins), or continues indefinitely (robber wins).
- G is **cop-win** if the cop can guarantee to be at the same vertex as the robber eventually, otherwise **robber-win**.

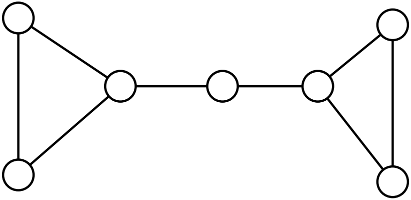
Discrete Cop and Robber Game - Examples



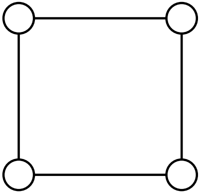
Discrete Cop and Robber Game - Examples



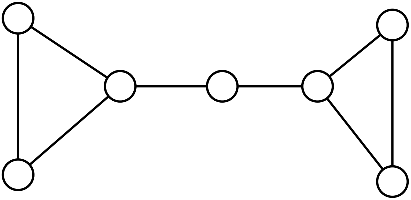
robber-win



Discrete Cop and Robber Game - Examples

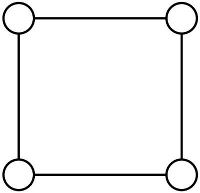


robber-win

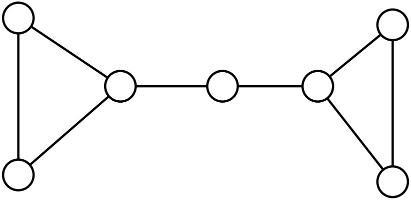


cop-win

Discrete Cop and Robber Game - Examples



robber-win



cop-win

- Studied by Quiliot (1978) and Nowakowski and Winkler (1983).
- G is cop-win if and only if it can be **dismantled**.

Cop and Robber in Temporal Graphs?

- We want to consider graphs with infinite lifetime whose edge set can change in every step.

Cop and Robber in Temporal Graphs?

- We want to consider graphs with infinite lifetime whose edge set can change in every step.
- Assume each edge e has a periodic appearance schedule b_e of length ℓ_e, meaning the edge appears in step t if:

$$b_e[t \bmod \ell_e] = 1$$

Cop and Robber in Temporal Graphs?

- We want to consider graphs with infinite lifetime whose edge set can change in every step.
- Assume each edge e has a periodic appearance schedule b_e of length ℓ_e, meaning the edge appears in step t if:

$$b_e[t \bmod \ell_e] = 1$$

- Example: $b_e = 01001$ means $\ell_e = 5$ and the edge appears in steps 1, 4, 6, 9, 11, 14, ...

Cop and Robber in Temporal Graphs?

- We want to consider graphs with infinite lifetime whose edge set can change in every step.
- Assume each edge e has a periodic appearance schedule b_e of length ℓ_e, meaning the edge appears in step t if:

$$b_e[t \bmod \ell_e] = 1$$

- Example: $b_e = 01001$ means $\ell_e = 5$ and the edge appears in steps 1, 4, 6, 9, 11, 14, . . .
- Such graphs are called **edge-periodic graphs** (Casteigts et al., 2011).
- An edge-periodic graph $\mathcal{G}$ is given by a graph $G = (V, E)$ together with b_e and ℓ_e for each $e \in E$.
- Define LCM as the least common multiple of the edge periods ℓ_e.

Edge-Periodic Cop and Robber Game (EPCR)

- **Edge-periodic graph $\mathcal{G}$.** One cop, one robber.

Edge-Periodic Cop and Robber Game (EPCR)

- **Edge-periodic graph $\mathcal{G}$.** One cop, one robber.
- Cop chooses its start vertex.

Edge-Periodic Cop and Robber Game (EPCR)

- **Edge-periodic graph $\mathcal{G}$.** One cop, one robber.
- Cop chooses its start vertex.
- Then robber chooses its start vertex.

Edge-Periodic Cop and Robber Game (EPCR)

- **Edge-periodic graph $\mathcal{G}$.** One cop, one robber.
- Cop chooses its start vertex.
- Then robber chooses its start vertex.
- Then cop and robber take turns. In each turn, can stay at a vertex or move over an edge to a vertex that is **adjacent in the current time step.**

Edge-Periodic Cop and Robber Game (EPCR)

- **Edge-periodic graph $\mathcal{G}$.** One cop, one robber.
- Cop chooses its start vertex.
- Then robber chooses its start vertex.
- Then cop and robber take turns. In each turn, can stay at a vertex or move over an edge to a vertex that is **adjacent in the current time step. In each time step, the cop moves first, then the robber.**

Edge-Periodic Cop and Robber Game (EPCR)

- **Edge-periodic graph $\mathcal{G}$.** One cop, one robber.
- Cop chooses its start vertex.
- Then robber chooses its start vertex.
- Then cop and robber take turns. In each turn, can stay at a vertex or move over an edge to a vertex that is **adjacent in the current time step. In each time step, the cop moves first, then the robber.**
- Full knowledge about other player's position **and edge appearance schedule**.

Edge-Periodic Cop and Robber Game (EPCR)

- **Edge-periodic graph $\mathcal{G}$.** One cop, one robber.
- Cop chooses its start vertex.
- Then robber chooses its start vertex.
- Then cop and robber take turns. In each turn, can stay at a vertex or move over an edge to a vertex that is **adjacent in the current time step. In each time step, the cop moves first, then the robber.**
- Full knowledge about other player's position **and edge appearance schedule**.
- Game ends when cop and robber are on the same vertex (cop wins), or continues indefinitely (robber wins).

Edge-Periodic Cop and Robber Game (EPCR)

- **Edge-periodic graph $\mathcal{G}$.** One cop, one robber.
- Cop chooses its start vertex.
- Then robber chooses its start vertex.
- Then cop and robber take turns. In each turn, can stay at a vertex or move over an edge to a vertex that is **adjacent in the current time step. In each time step, the cop moves first, then the robber.**
- Full knowledge about other player's position **and edge appearance schedule**.
- Game ends when cop and robber are on the same vertex (cop wins), or continues indefinitely (robber wins).
- $\mathcal{G}$ is **cop-win** if the cop can guarantee to be at the same vertex as the robber eventually, otherwise **robber-win**.

Result for Edge-Periodic Cop and Robber Games

Theorem

There is an algorithm to decide if an edge-periodic graph $\mathcal{G}$ with n vertices is cop-win or robber-win (and to compute a winning strategy for the winning player) in time $O(n^3 \cdot LCM)$.

Result for Edge-Periodic Cop and Robber Games

Theorem

There is an algorithm to decide if an edge-periodic graph $\mathcal{G}$ with n vertices is cop-win or robber-win (and to compute a winning strategy for the winning player) in time $O(n^3 \cdot LCM)$.

Proof outline.

- Edge appearance schedule for $\mathcal{G}$ repeats every LCM steps.
- Translate game into a **reachability game** on a suitable directed graph with $O(n^2 \cdot LCM)$ vertices and $O(n^3 \cdot LCM)$ edges.
- Apply known algorithm for reachability games.

□

Main Tool in the Proof: Reachability Games

- Directed graph $G = (V_1 \cup V_2, E)$ with set $F \subseteq V_1 \cup V_2$ of winning positions for player 1.
- In state $v \in V_i$, player i chooses an outgoing edge to determine the next state.
- Player 1 wins if the game reaches a state in F eventually. Otherwise, player 2 wins.

Main Tool in the Proof: Reachability Games

- Directed graph $G = (V_1 \cup V_2, E)$ with set $F \subseteq V_1 \cup V_2$ of winning positions for player 1.
- In state $v \in V_i$, player i chooses an outgoing edge to determine the next state.
- Player 1 wins if the game reaches a state in F eventually. Otherwise, player 2 wins.

Theorem (Berwanger (2011), Grädel et al. (2002))

For a given reachability game, one can determine in linear time the winning states for each player and corresponding memoryless winning strategies.

Translating EPCR into a Reachability Game

- Define a game state (c, r, p, t) for each possible configuration of the EPCR:
  - $c \in V$ is the cop's location
  - $r \in V$ is the robber's location
  - $p \in \{\mathrm{cop}, \mathrm{robber}\}$ is the player making the next move
  - $t \in \{0, 1, 2, \ldots LCM - 1\}$ is the current time step modulo LCM

Translating EPCR into a Reachability Game

- Define a game state (c, r, p, t) for each possible configuration of the EPCR:
  - $c \in V$ is the cop's location
  - $r \in V$ is the robber's location
  - $p \in \{\text{cop}, \text{robber}\}$ is the player making the next move
  - $t \in \{0, 1, 2, \ldots LCM - 1\}$ is the current time step modulo LCM
- Define directed edges to represent the possible moves of the player whose turn it is.

Translating EPCR into a Reachability Game

- Define a game state (c, r, p, t) for each possible configuration of the EPCR:
  - $c \in V$ is the cop's location
  - $r \in V$ is the robber's location
  - $p \in \{\text{cop}, \text{robber}\}$ is the player making the next move
  - $t \in \{0, 1, 2, \ldots LCM - 1\}$ is the current time step modulo LCM
- Define directed edges to represent the possible moves of the player whose turn it is.
- There are $2n^2 \cdot LCM$ game states, and at most n outgoing edges per step.

Translating EPCR into a Reachability Game

- Define a game state (c, r, p, t) for each possible configuration of the EPCR:
  - $c \in V$ is the cop's location
  - $r \in V$ is the robber's location
  - $p \in \{\mathrm{cop}, \mathrm{robber}\}$ is the player making the next move
  - $t \in \{0, 1, 2, \ldots LCM - 1\}$ is the current time step modulo LCM
- Define directed edges to represent the possible moves of the player whose turn it is.
- There are $2n^2 \cdot LCM$ game states, and at most n outgoing edges per step.
- An edge-periodic cop and robber game $\mathcal{G}$ is cop-win if and only if there is a vertex $c \in V$ such that $(c, r, \mathrm{cop}, 0)$ is a cop-winning state in the reachability game for all choices of r.
- The winning strategies of the reachability game translate into winning strategies of EPCR.

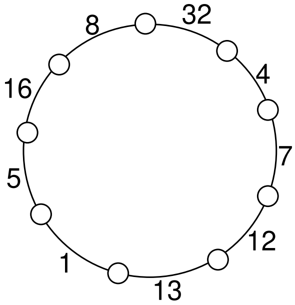
Strictly Edge-Periodic Cycles

Strictly Edge-Periodic Graphs and Cycles

- An edge-periodic graph $\mathcal{G}$ is **strictly edge-periodic** if each edge appearance schedule is of the form $000 \cdots 01$.
- In other words, each e is present once every ℓ_e steps, namely in the i-th step (step $i-1$) where i is a multiple of ℓ_e.

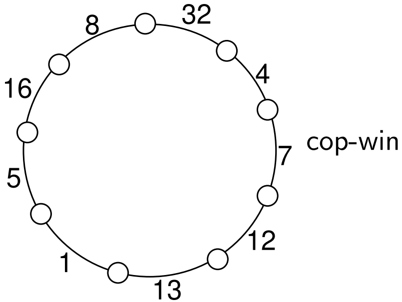
Strictly Edge-Periodic Graphs and Cycles

- An edge-periodic graph $\mathcal{G}$ is **strictly edge-periodic** if each edge appearance schedule is of the form $000\cdots01$.
- In other words, each e is present once every ℓ_e steps, namely in the i-th step (step $i-1$) where i is a multiple of ℓ_e.
- A strictly edge-periodic cycle is a strictly edge-periodic graph whose edge set forms a cycle.
- Example:



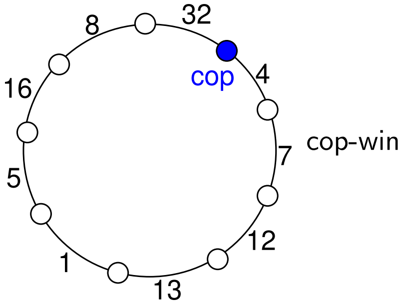
Strictly Edge-Periodic Graphs and Cycles

- An edge-periodic graph $\mathcal{G}$ is **strictly edge-periodic** if each edge appearance schedule is of the form $000\cdots01$.
- In other words, each e is present once every ℓ_e steps, namely in the i-th step (step $i-1$) where i is a multiple of ℓ_e.
- A strictly edge-periodic cycle is a strictly edge-periodic graph whose edge set forms a cycle.
- Example:



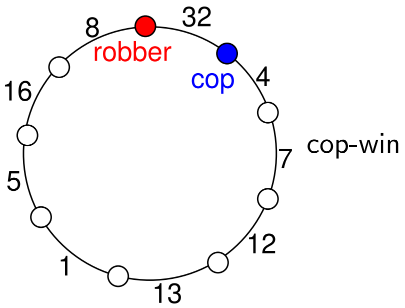
Strictly Edge-Periodic Graphs and Cycles

- An edge-periodic graph $\mathcal{G}$ is **strictly edge-periodic** if each edge appearance schedule is of the form $000\cdots01$.
- In other words, each e is present once every ℓ_e steps, namely in the i-th step (step $i-1$) where i is a multiple of ℓ_e.
- A strictly edge-periodic cycle is a strictly edge-periodic graph whose edge set forms a cycle.
- Example:



Strictly Edge-Periodic Graphs and Cycles

- An edge-periodic graph $\mathcal{G}$ is **strictly edge-periodic** if each edge appearance schedule is of the form $000 \cdots 01$.
- In other words, each e is present once every ℓ_e steps, namely in the i-th step (step $i-1$) where i is a multiple of ℓ_e.
- A strictly edge-periodic cycle is a strictly edge-periodic graph whose edge set forms a cycle.
- Example:



Result for Strictly Edge-Periodic Cycles

Question

Can we determine a value B such that if the strictly edge-periodic cycle has at least B edges, then it must be robber-win?

Result for Strictly Edge-Periodic Cycles

Question

Can we determine a value B such that if the strictly edge-periodic cycle has at least B edges, then it must be robber-win?

- Recall that LCM is the least common multiple of all edge periods l_e.
- Let $f = 1$ if $LCM \geq 2 \max_e \ell_e$, and $f = 2$ otherwise.

Result for Strictly Edge-Periodic Cycles

Question

Can we determine a value B such that if the strictly edge-periodic cycle has at least B edges, then it must be robber-win?

- Recall that LCM is the least common multiple of all edge periods l_e.
- Let $f = 1$ if $LCM \geq 2\max_e \ell_e$, and $f = 2$ otherwise.

Theorem

If a strictly edge-periodic cycle has at least $4f \cdot LCM + 4$ edges, then it is robber-win.

Auxiliary problem: Infinite edge-periodic path

- Infinite path starting at v_0, **finite** set of edge periods l_e.
- Cop initially placed at v_0.

Auxiliary problem: Infinite edge-periodic path

- Infinite path starting at v_0, **finite** set of edge periods l_e.
- Cop initially placed at v_0.

Lemma

The cop and robber game played on an infinite edge-periodic path is robber-win.

Auxiliary problem: Infinite edge-periodic path

- Infinite path starting at v_0, **finite** set of edge periods l_e.
- Cop initially placed at v_0.

Lemma

The cop and robber game played on an infinite edge-periodic path is robber-win.

Proof.

- Partition path into segments, such that the cop passes through one segment in $f \cdot LCM$ steps (assuming it never waits unnecessarily).

Auxiliary problem: Infinite edge-periodic path

- Infinite path starting at v_0, **finite** set of edge periods l_e.
- Cop initially placed at v_0.

Lemma

The cop and robber game played on an infinite edge-periodic path is robber-win.

Proof.

- Partition path into segments, such that the cop passes through one segment in $f \cdot LCM$ steps (assuming it never waits unnecessarily).

- Observe: Each segment consists of at least two edges.

Auxiliary problem: Infinite edge-periodic path

- Infinite path starting at v_0, **finite** set of edge periods l_e.
- Cop initially placed at v_0.

Lemma

The cop and robber game played on an infinite edge-periodic path is robber-win.

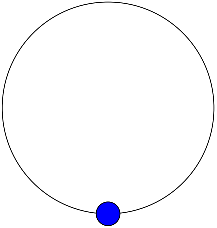
Proof.

- Partition path into segments, such that the cop passes through one segment in $f \cdot LCM$ steps (assuming it never waits unnecessarily).

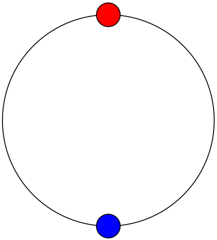
- Observe: Each segment consists of at least two edges.
- Place robber at start of second segment, and always move right

□

Robber Strategy for Cycles with $\geq 4f \cdot LCM + 4$ Edges

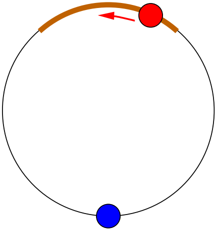


Robber Strategy for Cycles with $\geq 4f \cdot LCM + 4$ Edges



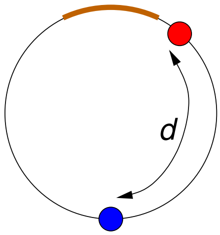
- Robber chooses vertex opposite the cop's start vertex.

Robber Strategy for Cycles with $\geq 4f \cdot LCM + 4$ Edges



- Robber chooses vertex opposite the cop's start vertex.
- **Dodge mode**: While robber is in segment of 2–3 edges opposite cop's current vertex, move towards cop's antipodal vertex if possible.

Robber Strategy for Cycles with $\geq 4f \cdot LCM + 4$ Edges



- Robber chooses vertex opposite the cop's start vertex.
- **Dodge mode**: While robber is in segment of 2–3 edges opposite cop's current vertex, move towards cop's antipodal vertex if possible.
- **Escape mode**: If cop moves so that robber leaves the segment, use modified infinite path strategy.
  Note: Initial distance is $d \geq 2f \cdot LCM$ edges.

Escape Mode

- Consider infinite path starting at cop's current position following the ring towards the robber.

Escape Mode

- Consider infinite path starting at cop's current position following the ring towards the robber.
- Partition path into segments, such that the cop passes through the last edge of each segment at i-th step such that i is a multiple of $f \cdot LCM$.

Escape Mode

- Consider infinite path starting at cop's current position following the ring towards the robber.
- Partition path into segments, such that the cop passes through the last edge of each segment at i-th step such that i is a multiple of $f \cdot LCM$.

- Each segment is at most $f \cdot LCM$ edges long.

Escape Mode

- Consider infinite path starting at cop's current position following the ring towards the robber.
- Partition path into segments, such that the cop passes through the last edge of each segment at i-th step such that i is a multiple of $f \cdot LCM$.

- Each segment is at most $f \cdot LCM$ edges long.
- As the initial distance between robber and cop is at least $2f \cdot LCM$ edges, the robber is located in the $\geq$3rd segment.

Escape Mode

- Consider infinite path starting at cop's current position following the ring towards the robber.
- Partition path into segments, such that the cop passes through the last edge of each segment at i-th step such that i is a multiple of $f \cdot LCM$.

- Each segment is at most $f \cdot LCM$ edges long.
- As the initial distance between robber and cop is at least $2f \cdot LCM$ edges, the robber is located in the $\geq$3rd segment.
- **Robber strategy**: Wait until cop reaches end of first segment, then walk through segments in parallel with the (non-waiting) cop.

Escape Mode

- Consider infinite path starting at cop's current position following the ring towards the robber.
- Partition path into segments, such that the cop passes through the last edge of each segment at i-th step such that i is a multiple of $f \cdot LCM$.

- Each segment is at most $f \cdot LCM$ edges long.
- As the initial distance between robber and cop is at least $2f \cdot LCM$ edges, the robber is located in the $\geq$3rd segment.
- **Robber strategy**: Wait until cop reaches end of first segment, then walk through segments in parallel with the (non-waiting) cop.
- If the robber is ever located in the segment opposite the cop again, revert to Dodge mode.

Conclusion

Results for cop and robber games on edge-periodic graphs:

- $O(n^3 \cdot LCM)$ algorithm to determine cop-win or robber-win
  - Can be extended to k cops using $O(n^{2k+1} \cdot LCM)$ time or $O(n^{k+1}k \cdot LCM)$ time.
- Strictly edge-periodic cycles with at least $4f \cdot LCM + 4$ edges are robber-win.

Future work:

- Can we check if an edge-periodic graph is cop-win or robber-win in **polynomial time**?
  (The input size is $O(|V| + |E| + \sum_{e \in E} \ell_e)$, but LCM can be exponential in $\max_{e \in E} \ell_e$.)
- Are there cop-and-robber games on temporal graphs that lead to **useful graph parameters** (similar to treewidth for static graphs)?

Thank you!

Questions?