Spy-Game on graphs

Nathann Cohen\textsuperscript{1}  Mathieu Hilaire\textsuperscript{2}  Nícolas A. Martins\textsuperscript{3}  Nicolas Nisse\textsuperscript{4,5}  Stéphane Pérennes\textsuperscript{5,4}

\textsuperscript{1} CNRS, Univ Paris Sud, LRI, Orsay, France
\textsuperscript{2} ENS Cachan, France
\textsuperscript{3} Universidade Federal do Ceará, Fortaleza, Brazil
\textsuperscript{4} Inria, France
\textsuperscript{5} Univ. Nice Sophia Antipolis, CNRS, I3S, UMR 7271, Sophia Antipolis, France

Search Games: Theory and Algorithms

Leiden, Netherlands, June 28th, 2016
Many Two-player games in Graphs

**Cops and Robber** [Nowakowski and Winkler; Quilliot 1983]
A team of *Cops* attempts to capture one *Robber*

**Angels and Devils** [Conway 1996]
*Angel* moves on the graph, *Devil* blocks vertices

**Eternal Domination** [Burger *et al* 2004]
A team of *Defenders* protects nodes from one *Attacker*

**Spy Game** [this work]
A team of *Guards* attempts to stay close to the *Spy*
Many Two-player games in Graphs

Cops and Robber [Nowakowski and Winkler; Quilliot 1983]
A team of *Cops* attempts to capture one *Robber*

Angels and Devils [Conway 1996]
*Angel* moves on the graph, *Devil* blocks vertices

Eternel Domination [Burger et al 2004]
A team of *Defenders* protects nodes from one *Attacker*

Spy Game [this work]
A team of *Guards* attempts to stay close to the *Spy*
Motivations to define yet another game

Cops & Robber Games [Nowakowski and Winkler; Quilliot, 1983]

Rules of the $C&R$ game

Place

$k \geq 1$ Cops

$C$ on nodes

Visible Robber

$R$ at one node

Turn by turn (1) each $C$ slides along $\leq 1$ edge

(2) $R$ slides along $\leq 1$ edge

Goal of the $C&R$ game

Robber must avoid the Cops

Cops must capture Robber (i.e., occupy the same node)

Cop Number of a graph $G$

$cn(G)$: min # Cops to win in $G$
Motivations to define yet another game

Cops & Robber Games [Nowakowski and Winkler; Quilliot, 1983]

Rules of the C&R game

1. Place $k \geq 1$ Cops $C$ on nodes
Motivations to define yet another game

Cops & Robber Games [Nowakowski and Winkler; Quilliot, 1983]

Rules of the C&R game
1. Place $k \geq 1$ Cops C on nodes
2. Visible Robber R at one node
Motivations to define yet another game

Cops & Robber Games [Nowakowski and Winkler; Quilliot, 1983]

Rules of the \(C&\mathcal{R}\) game

1. Place \(k \geq 1\) Cops \(C\) on nodes
2. Visible Robber \(\mathcal{R}\) at one node
3. Turn by turn
   (1) each \(C\) slides along \(\leq 1\) edge
Motivations to define yet another game

Cops & Robber Games [Nowakowski and Winkler; Quilliot, 1983]

Rules of the $C&R$ game

1. Place $k \geq 1$ Cops $C$ on nodes
2. Visible Robber $R$ at one node
3. Turn by turn
   (1) each $C$ slides along $\leq 1$ edge
   (2) $R$ slides along $\leq 1$ edge
Motivations to define yet another game

Cops & Robber Games [Nowakowski and Winkler; Quilliot, 1983]

Rules of the $C\&R$ game

1. Place $k \geq 1$ Cops $C$ on nodes
2. Visible Robber $R$ at one node
3. Turn by turn
   (1) each $C$ slides along $\leq 1$ edge
   (2) $R$ slides along $\leq 1$ edge
Motivations to define yet another game

Cops & Robber Games [Nowakowski and Winkler; Quilliot, 1983]

Rules of the $C&R$ game

1. Place $k \geq 1$ Cops $C$ on nodes
2. Visible Robber $R$ at one node
3. Turn by turn
   (1) each $C$ slides along $\leq 1$ edge
   (2) $R$ slides along $\leq 1$ edge

Goal of the $C&R$ game

- Robber must avoid the Cops
Motivations to define yet another game

Cops & Robber Games [Nowakowski and Winkler; Quilliot, 1983]

Rules of the C&R game
1. Place $k \geq 1$ Cops $C$ on nodes
2. Visible Robber $R$ at one node
3. Turn by turn
   (1) each $C$ slides along $\leq 1$ edge
   (2) $R$ slides along $\leq 1$ edge

Goal of the C&R game
- Robber must avoid the Cops
- Cops must capture Robber (i.e., occupy the same node)
Motivations to define yet another game

Cops & Robber Games [Nowakowski and Winkler; Quilliot, 1983]

Rules of the C&R game
1. Place $k \geq 1$ Cops $C$ on nodes
2. Visible Robber $R$ at one node
3. Turn by turn
   (1) each $C$ slides along $\leq 1$ edge
   (2) $R$ slides along $\leq 1$ edge

Goal of the C&R game
- Robber must avoid the Cops
- Cops must capture Robber (i.e., occupy the same node)

Cop Number of a graph $G$  
$cn(G)$: min # Cops to win in $G$
Motivations to define yet another game

Let’s play a bit
Motivations to define yet another game

Let’s play a bit

Cohen et al
Spy-Game on graphs
Motivations to define yet another game

Let’s play a bit
Motivations to define yet another game

Let’s play a bit
Motivations to define yet another game

Let’s play a bit

Spy-Game on graphs
Motivations to define yet another game

Let’s play a bit
Motivations to define yet another game

Let’s play a bit

cn(tree)=1
Motivations to define yet another game

Let’s play a bit

- cn(tree) = 1
- cn(clique) = ?
- cn(cycle) = ?
- cn(Petersen) = ?
Motivations to define yet another game

Let’s play a bit

\[
\begin{align*}
cn(\text{tree}) &= 1 \\
cn(\text{clique}) &= ? \\
cn(\text{cycle}) &= ? \\
cn(\text{Petersen}) &= ?
\end{align*}
\]
Motivations to define yet another game

Let’s play a bit

- $cn(\text{tree}) = 1$
- $cn(\text{clique}) = ?$
- $cn(\text{cycle}) = ?$
- $cn(\text{Petersen}) = ?$
Motivations to define yet another game

Let’s play a bit

c_{\text{tree}} = 1

\begin{align*}
c_{\text{clique}} &= 1 \\
c_{\text{cycle}} &= ? \\
c_{\text{Petersen}} &= ?
\end{align*}
Motivations to define yet another game

Let’s play a bit

cn(tree) = 1

cn(clique) = 1

cn(cycle) = ?

cn(Petersen) = ?
Motivations to define yet another game

Let’s play a bit

- cn(tree) = 1
- cn(clique) = 1
- cn(cycle) = ?
- cn(Petersen) = ?

Spy-Game on graphs
Motivations to define yet another game

Let’s play a bit

cn(tree) = 1

cn(clique) = 1

Question marks indicate unknown values.

Cohen et al. 
Spy-Game on graphs
Motivations to define yet another game

Let’s play a bit

cn(tree) = 1

cn(clique) = 1

cn(cycle) = ?

cn(Petersen) = ?

Cohen et al.  Spy-Game on graphs
Motivations to define yet another game

Let’s play a bit

- cn(tree) = 1
- cn(clique) = 1
- cn(cycle) = ?
- cn(Petersen) = ?
Motivations to define yet another game

Let’s play a bit

- cn(tree) = 1
- cn(clique) = 1
- cn(cycle) = ?
- cn(Petersen) = ?
Motivations to define yet another game

Let’s play a bit

- cn(tree) = 1
- cn(clique) = 1
- cn(cycle) = 2
- cn(Petersen) = ?
Motivations to define yet another game

Let’s play a bit

- cn(tree) = 1
- cn(clique) = 1
- cn(cycle) = 2
- cn(Petersen) = ?
Motivations to define yet another game

Let’s play a bit

\[
\begin{align*}
\text{cn}(\text{tree}) &= 1 \\
\text{cn}(\text{clique}) &= 1 \\
\text{cn}(\text{cycle}) &= 2 \\
\text{cn}(\text{Petersen}) &= ?
\end{align*}
\]
Motivations to define yet another game

Let’s play a bit

cn(tree) = 1

cn(clique) = 1

cn(cycle) = 2

cn(Petersen) = ?
Motivations to define yet another game

Let’s play a bit

\[
\begin{align*}
\text{cn(tree)} &= 1 \\
\text{cn(clique)} &= 1 \\
\text{cn(cycle)} &= 2 \\
\text{cn(Petersen)} &= ?
\end{align*}
\]
Motivations to define yet another game

Let’s play a bit

cn(tree)=1
cn(clique)=1
cn(cycle)=2

cn(Petersen)=?
Motivations to define yet another game

Let’s play a bit

- **cn(tree) = 1**
- **cn(clique) = 1**
- **cn(cycle) = 2**
- **cn(Petersen) = 3**
Motivations to define yet another game

Meyniel Conjecture (1985): \( \forall \) connected \( n \)-node graph \( G \), \( cn(G) = O(\sqrt{n}) \)
Motivations to define yet another game

Meyniel Conjecture (1985): \( \forall \) connected \( n \)-node graph \( G \), \( cn(G) = O(\sqrt{n}) \)

<table>
<thead>
<tr>
<th>True in many graph classes</th>
<th>( cn )</th>
<th>[ \text{folklore} ]</th>
</tr>
</thead>
<tbody>
<tr>
<td>dominating set ( \leq k )</td>
<td>( \leq k )</td>
<td>[ Joret, Kaminski, Theis 09 ]</td>
</tr>
<tr>
<td>treewidth ( \leq t )</td>
<td>( \leq t/2 + 1 )</td>
<td>[ Joret, Kaminski, Theis 09 ]</td>
</tr>
<tr>
<td>chordality ( \leq k )</td>
<td>( \leq k )</td>
<td>[ Kosowski, Li, N., Suchan 12 ]</td>
</tr>
<tr>
<td>genus ( \leq g )</td>
<td>( \leq \lfloor \frac{3g}{2} \rfloor + 3 )</td>
<td>( (\text{Conjecture} \leq g + 3) ) [ Schröder, 01 ]</td>
</tr>
<tr>
<td>planar graphs</td>
<td>( \leq 3 )</td>
<td>[ Aigner, Fromme, 84 ]</td>
</tr>
<tr>
<td>grids</td>
<td>( = 2 )</td>
<td>[ folklore ]</td>
</tr>
<tr>
<td>( H )-minor free</td>
<td>( \leq</td>
<td>E(H)</td>
</tr>
<tr>
<td>degeneracy ( \leq d )</td>
<td>( \leq d )</td>
<td>[ Lu, Peng 12 ]</td>
</tr>
<tr>
<td>diameter 2</td>
<td>( O(\sqrt{n}) )</td>
<td>[ Bollobás et al. 08 ]</td>
</tr>
<tr>
<td>bipartite diameter 3</td>
<td>( O(\sqrt{n}) )</td>
<td>[ Luczak, Pralat 10 ]</td>
</tr>
<tr>
<td>Erdös-Rényi graphs</td>
<td>( O(\sqrt{n}) )</td>
<td>[ Bonato, Pralat, Wang 07 ]</td>
</tr>
<tr>
<td>Power law</td>
<td>( O(\sqrt{n}) )</td>
<td>[ ]</td>
</tr>
</tbody>
</table>
Motivations to define yet another game

Meyniel Conjecture (1985): \( \forall \) connected \( n \)-node graph \( G \), \( cn(G) = O(\sqrt{n}) \)

<table>
<thead>
<tr>
<th>True in many graph classes</th>
<th>( cn )</th>
<th>( [\text{folklore}] )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{dominating set} \leq k )</td>
<td>( \leq k )</td>
<td>( [\text{folklore}] )</td>
</tr>
<tr>
<td>( \text{treewidth} \leq t )</td>
<td>( \leq t/2 + 1 )</td>
<td>( [\text{Joret, Kaminski, Theis 09}] )</td>
</tr>
<tr>
<td>( \text{chordality} \leq k )</td>
<td>( &lt; k )</td>
<td>( [\text{Kosowski, Li, N., Suchan 12}] )</td>
</tr>
<tr>
<td>( \text{genus} \leq g )</td>
<td>( \leq \left\lfloor \frac{3g}{2} \right\rfloor + 3 )</td>
<td>( (\text{Conjecture} \leq g + 3) \ [\text{Schröder, 01}] )</td>
</tr>
<tr>
<td>( \text{planar graphs} )</td>
<td>( \leq 3 )</td>
<td>( [\text{Aigner, Fromme, 84}] )</td>
</tr>
<tr>
<td>( \text{grids} )</td>
<td>( = 2 )</td>
<td>( [\text{folklore}] )</td>
</tr>
<tr>
<td>( \text{( H )-minor free} )</td>
<td>( \leq</td>
<td>E(H)</td>
</tr>
<tr>
<td>( \text{degeneracy} \leq d )</td>
<td>( \leq d )</td>
<td>( [\text{Lu, Peng 12}] )</td>
</tr>
<tr>
<td>( \text{diameter} 2 )</td>
<td>( O(\sqrt{n}) )</td>
<td>( - )</td>
</tr>
<tr>
<td>( \text{bipartite diameter} 3 )</td>
<td>( O(\sqrt{n}) )</td>
<td>( - )</td>
</tr>
<tr>
<td>( \text{Erdös-Rényi graphs} )</td>
<td>( O(\sqrt{n}) )</td>
<td>( [\text{Bollobás et al. 08}] )</td>
</tr>
<tr>
<td>( \text{Power law} )</td>
<td>( O(\sqrt{n}) )</td>
<td>( [\text{Luczak, Pralat 10}] )</td>
</tr>
</tbody>
</table>

Currently known best general upper bound...

\[ cn(G) = O(\frac{n}{2^{(1-o(1))}\sqrt{\log n}}) \]

[Scott, Sudakov 11, Lu, Peng 12]

Note that \( \frac{n}{2^{(1-o(1))}\sqrt{\log n}} \geq n^{1-\epsilon} \) for any \( \epsilon > 0 \)
Motivations to define yet another game

To tackle Meyniel’s conjecture: new variants have been defined.

**When the Robber can run** [Fomin,Golovach,Kratochvil,N.,Suchan’10]

**New variant** with speed: The Robber may move along several edges per turn

\[ cn_s(G) : \text{min } \# \text{ of Cops to capture Robber with speed } s \geq 1. \]

Meyniel Conjecture [Alon, Mehrabian’11] and general upper bound [Frieze,Krivelevich,Loh’12] extend to this variant

... but fundamental differences (recall: planar graphs have \( cn_1 < 3 \))

\( cn_2(G) \) unbounded in grids [Fomin,Golovach,Kratochvil,N.,Suchan’10]

\[ \Omega(\sqrt{\log n}) \leq cn_2(G_{n \times n}) \leq O(n) \text{ in } n \times n \text{ grid } G_{n \times n} \]

Open question: exact value of \( cn_2(G_{n \times n}) \)?
Motivations to define yet another game

To tackle Meyniel’s conjecture: new variants have been defined.

**When the Robber can run**

[Fomin, Golovach, Kratochvil, N., Suchan’10]

**New variant** with speed: The Robber may move along several edges per turn

\( cn_s(G) \): min \# of Cops to capture Robber with speed \( s \geq 1 \).

Meyniel Conjecture [Alon, Mehrabian’11] and general upper bound [Frieze, Krivelevich, Loh’12] extend to this variant

... but fundamental differences (recall: planar graphs have \( cn_1 \leq 3 \))

\( cn_2(G) \) unbounded in grids

[Fomin, Golovach, Kratochvil, N., Suchan’10]

\[ \Omega(\sqrt{\log n}) \leq cn_2(G_{n \times n}) \leq O(n) \text{ in } n \times n \text{ grid } G_{n \times n} \]

Open question: exact value of \( cn_2(G_{n \times n}) \)?
Motivations to define yet another game

To tackle Meyniel’s conjecture: new variants have been defined.

**When the Robber can run**

New variant with speed: The Robber may move along several edges per turn

\[ cn_s(G) \]: min \# of Cops to capture Robber with speed \( s \geq 1 \).

Meyniel Conjecture [Alon, Mehrabian’11] and general upper bound [Frieze,Krivelevich,Loh’12] extend to this variant

... but fundamental differences (recall: planar graphs have \( cn_1 \leq 3 \))

\( cn_2(G) \) unbounded in grids [Fomin,Golovach,Kratochvil,N.,Suchan’10]

\[ \Omega(\sqrt{\log n}) \leq cn_2(G_{n\times n}) \leq O(n) \] in \( n \times n \) grid \( G_{n\times n} \)

Open question: exact value of \( cn_2(G_{n\times n}) \) ?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
More simple (?) question:
In \( \infty \times n \)-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?

Project the whole game on One column
Goal of Robber: go far to all cops
"Find a hole"
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?

Project the whole game on One column

Goal of Robber: go far to all cops
"Find a hole"
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?

Project the whole game on One column

Goal of Robber: go far to all cops
"Find a hole"
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?

Project the whole game on One column

Goal of Robber: go far to all cops
"Find a hole"
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?

Project the whole game on One column

Goal of Robber: go far to all cops
"Find a hole"
Motivations to define yet another game: Rugby!

More simple (?) question:
In $\infty \times n$-grid: number of cops with speed 1 needed to stop a robber with speed 2?

Project the whole game on One column

Goal of Robber: go far to all cops
"Find a hole"
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node
2. Place $k$ guards on nodes (here $k = 3$) (may occupy same nodes)
3. Turn by turn
   (1) Spy moves along $\leq s$ edges (here $s = 2$)
   (2) Guards slide along $\leq 1$ edge

Goal of the Spy game
Spy must reach a node at distance $> d$ from all cops (after Guards' moves)

Guards must always "control" the Spy at distance $\leq d$
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node

Place \( k \) guards on nodes (here \( k = 3 \))

Turn by turn (1) Spy moves along \( \leq s \) edges (here \( s = 2 \))
(2) Guards slide along \( \leq 1 \) edge

Goal of the Spy game
Spy must reach a node at distance \( > d \) from all cops (after Guards' moves)
Guards must always "control" the Spy at distance \( \leq d \)

\( g_n, d(G) \): min # Guards to win in \( G \)
\( d(G), k \): min distance s.t. \( k \) Guards win

\( \iff \) Cops and Robber for \( s = 1, d = 0 \)
\( \iff \) Eternal Domination for \( s = \infty, d = 0 \)
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node
2. Place $k$ guards on nodes (here $k = 3$) (may occupy same nodes)
Spy Game in graphs

**Rules of the Spy game**

1. Place the Spy on one node
2. Place $k$ guards on nodes (here $k = 3$) (may occupy same nodes)
3. Turn by turn
   (1) Spy moves along $\leq s$ edges
   (here $s = 2$)
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node
2. Place $k$ guards on nodes (here $k = 3$) (may occupy same nodes)
3. Turn by turn
   (1) Spy moves along $\leq s$ edges (here $s = 2$)
   (2) Guards slide along $\leq 1$ edge
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node
2. Place $k$ guards on nodes (here $k = 3$) (may occupy same nodes)
3. Turn by turn
   (1) Spy moves along $\leq s$ edges (here $s = 2$)
   (2) Guards slide along $\leq 1$ edge
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node
2. Place $k$ guards on nodes (here $k = 3$) (may occupy same nodes)
3. Turn by turn
   (1) Spy moves along $\leq s$ edges (here $s = 2$)
   (2) Guards slide along $\leq 1$ edge
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node
2. Place $k$ guards on nodes (here $k = 3$) (may occupy same nodes)
3. Turn by turn
   (1) Spy moves along $\leq s$ edges (here $s = 2$)
   (2) Guards slide along $\leq 1$ edge

Goal of the Spy game

- Spy must reach a node at distance $> d$ from all cops (after Guards’ moves)
- Guards must always “control” the Spy at distance $\leq d$
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node
2. Place $k$ guards on nodes (here $k = 3$) (may occupy same nodes)
3. Turn by turn
   (1) Spy moves along $\leq s$ edges (here $s = 2$)
   (2) Guards slide along $\leq 1$ edge

Goal of the Spy game

- Spy must reach a node at distance $> d$ from all cops (after Guards’ moves)
- Guards must always “control” the Spy at distance $\leq d$
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node
2. Place \( k \) guards on nodes (here \( k = 3 \)) (may occupy same nodes)
3. Turn by turn
   (1) Spy moves along \( \leq s \) edges (here \( s = 2 \))
   (2) Guards slide along \( \leq 1 \) edge

Goal of the Spy game

- Spy must reach a node at distance > \( d \) from all cops (after Guards’ moves)
- Guards must always “control” the Spy at distance \( \leq d \)
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node
2. Place $k$ guards on nodes (here $k = 3$) (may occupy same nodes)
3. Turn by turn
   (1) Spy moves along $\leq s$ edges (here $s = 2$)
   (2) Guards slide along $\leq 1$ edge

Goal of the Spy game

- Spy must reach a node at distance $> d$ from all cops (after Guards’ moves)
- Guards must always “control” the Spy at distance $\leq d$
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node
2. Place \( k \) guards on nodes (here \( k = 3 \)) (may occupy same nodes)
3. Turn by turn
   (1) Spy moves along \( \leq s \) edges (here \( s = 2 \))
   (2) Guards slide along \( \leq 1 \) edge

Goal of the Spy game

- Spy must reach a node at distance \( > d \) from all cops (after Guards’ moves)
- Guards must always “control” the Spy at distance \( \leq d \)
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node
2. Place $k$ guards on nodes (here $k = 3$) (may occupy same nodes)
3. Turn by turn
   (1) Spy moves along $\leq s$ edges (here $s = 2$)
   (2) Guards slide along $\leq 1$ edge

Goal of the Spy game

- Spy must reach a node at distance $> d$ from all cops (after Guards’ moves)
- Guards must always “control” the Spy at distance $\leq d$

$gn_{s,d}(G)$: min # Guards to win in $G$

$d_{s,k}(G)$: min distance s.t. $k$ Guards win
Spy Game in graphs

Rules of the Spy game

1. Place the Spy on one node
2. Place $k$ guards on nodes (here $k = 3$) (may occupy same nodes)
3. Turn by turn
   (1) Spy moves along $\leq s$ edges (here $s = 2$)
   (2) Guards slide along $\leq 1$ edge

Goal of the Spy game

- Spy must reach a node at distance $> d$ from all cops (after Guards’ moves)
- Guards must always “control” the Spy at distance $\leq d$

$gn_{s,d}(G)$: min $\#$ Guards to win in $G$
$d_{s,k}(G)$: min distance s.t. $k$ Guards win

$\Leftrightarrow$ Cops and Robber for $s = 1$, $d = 0$
$\Leftrightarrow$ Eternal Domination for $s = \infty$, $d = 0$
Spy Game in graphs

Results

Complexity

- Computing $g_{n_s,d}$ is NP-hard (reduction to Set Cover)
- Computing $g_{n_s,d}$ is PSPACE-hard in DAGs if Guards are placed first

Case of Paths and Cycles on $n$ vertices

- Paths: $\left\lfloor \frac{n(s-1)}{2ks} \right\rfloor \leq d_{s,k}(P_n) \leq \left\lceil \frac{(n+1)(s-1)}{2ks} \right\rceil$
- Cycles: $\left\lfloor \frac{(n-1)(s-1)}{k(2s+2)-4} \right\rfloor \leq d_{s,k}(C_n) \leq \left\lceil \frac{(n+1)(s-1)}{k(2s+2)-4} \right\rceil$

Case of grids: # of guards is super-linear in the side $n$

$\exists \epsilon > 0$ such that $g_{n_s,d}(G_{n \times n}) = \Omega(n^{1+\epsilon})$ in any $n \times n$ grid $G_{n \times n}$

Fractional relaxation of the game

9/16
Spy Game in graphs

Complexity

- Computing $g_{n,s,d}$ is NP-hard (reduction to Set Cover)
- Computing $g_{n,s,d}$ is PSPACE-hard in DAGs if Guards are placed first

Case of Paths and Cycles on $n$ vertices

- Paths: $\left\lceil \frac{n(s-1)}{2ks} \right\rceil \leq d_{s,k}(P_n) \leq \left\lceil \frac{(n+1)(s-1)}{2ks} \right\rceil$
- Cycles: $\left\lceil \frac{(n-1)(s-1)}{k(2s+2)-4} \right\rceil \leq d_{s,k}(C_n) \leq \left\lceil \frac{(n+1)(s-1)}{k(2s+2)-4} \right\rceil$

Case of grids: # of guards is super-linear in the side $n$

$\exists \epsilon > 0$ such that $g_{n,s,d}(G_{n \times n}) = \Omega(n^{1+\epsilon})$ in any $n \times n$ grid $G_{n \times n}$

Fractional relaxation of the game

Cohen et al: Spy-Game on graphs
Spy Game in graphs

Results

Complexity

- Computing $g_{n,s,d}$ is NP-hard (reduction to Set Cover)
- Computing $g_{n,s,d}$ is PSPACE-hard in DAGs if Guards are placed first

Case of Paths and Cycles on $n$ vertices

- Paths: $\left\lfloor \frac{n(s-1)}{2ks} \right\rfloor \leq d_{s,k}(P_n) \leq \left\lceil \frac{(n+1)(s-1)}{2ks} \right\rceil$
- Cycles: $\left\lfloor \frac{(n-1)(s-1)}{k(2s+2)-4} \right\rfloor \leq d_{s,k}(C_n) \leq \left\lceil \frac{(n+1)(s-1)}{k(2s+2)-4} \right\rceil$

Case of grids:

- # of guards is super-linear in the side $n$
  - $\exists \epsilon > 0$ such that $g_{n,s,d}(G_{n \times n}) = \Omega(n^{1+\epsilon})$ in any $n \times n$ grid $G_{n \times n}$
  - Fractional relaxation of the game

Cohen et al
Spy-Game on graphs
The case of paths

\[ d_{s,k}(P_n) = \Theta(n \cdot \frac{s-1}{2ks}) \]

Lower Bound: Spy starts from one end and runs!
The case of paths

\[ d_{s,k}(P_n) = \Theta(n \cdot \frac{s-1}{2ks}) \]

**Lower Bound:** Spy starts from one end and runs! One guard at distance \( \leq d \)
The case of paths

\[ d_{s,k}(P_n) = \Theta(n \cdot \frac{s-1}{2ks}) \]

Lower Bound: Spy starts from one end and runs!
The case of paths

\[ d_{s,k}(P_n) = \Theta(n \cdot \frac{s-1}{2ks}) \]

Lower Bound: Spy starts from one end and runs!
The case of paths

\[ d_{s,k}(P_n) = \Theta(n \cdot \frac{s-1}{2ks}) \]

Lower Bound: Spy starts from one end and runs! One guard is “consumed” after \(2d/(s-1)\) steps
The case of paths

\[ d_{s,k}(P_n) = \Theta(n \cdot \frac{s-1}{2ks}) \]

**Lower Bound:** Spy starts from one end and runs! Another one at distance \( \leq d \)
The case of paths

$$d_{s,k}(P_n) = \Theta(n \cdot \frac{s-1}{2ks})$$

Lower Bound: Spy starts from one end and runs! Hence, $n \leq k \cdot 2ds/(s - 1)$
The case of paths

$$d_{s,k}(P_n) = \Theta(n \cdot \frac{s-1}{2ks})$$

**Upper Bound:** Each guard is assigned its own area of length $\leq \frac{2ds}{(s - 1)}$. 
Upper Bound: Each guard is assigned its own area of length $\leq \frac{2ds}{(s - 1)}$. 

$$d_{s,k}(P_n) = \Theta(n \cdot \frac{s-1}{2ks})$$
The case of paths

\[ d_{s,k}(P_n) = \Theta(n \cdot \frac{s-1}{2ks}) \]

Upper Bound: Each guard is assigned its own area of length \( \leq \frac{2ds}{(s-1)} \).
Lemma 1: $g_{n,d}(G_{n \times n}) = \Omega(n \log n)$ in $n \times n$ grid

Consider only "L - strategies"
Lemma 1: $g_{n,s,d}(G_{n \times n}) = \Omega(n \log n)$ in $n \times n$ grid

Consider only "L - strategies"
Lemma 1: \( gn_{s,d}(G_{n \times n}) = \Omega(n \log n) \) in \( n \times n \) grid

Consider only "L - strategies"
Lemma 1: $gn_{s,d}(G_{n \times n}) = \Omega(n \log n)$ in $n \times n$ grid

Consider only "L - strategies"

After at most $O(n)$ steps: Spy far from $n \log n$ Guards
Case of Grids

Lemma 1: \( g_{s,d}(G_{n \times n}) = \Omega(n \log n) \) in \( n \times n \) grid

After at most \( O(n) \) steps: Spy far from \( n \log n \) Guards

Not enough for the announced bound:

\[
g_{s,d}(G_{n \times n}) = \Omega(n^{1+\epsilon}) \quad \text{for some } \epsilon > 0
\]

We would like to reduce guards “density” in order to recurse.
Consider “fractional guards”

$$fg(v) \in \mathbb{R}^+: \text{amount of guard on vertex } v$$

Total amount of guards

$$\sum_{v \in V(G)} fg(v)$$

Moves of guards

It is a flow!

Winning condition: control the Spy at each step

$$\sum_{v \in B(Spy, d)} fg(v) \geq 1$$

$B(Spy, d)$: ball of radius $d$ centered on the Spy
**Fractional Spy-Game**

\( \frac{\text{gns}}{d}(G) \): min amount of fractional guards required to win

**Theorem:** super-linear and sub-quadratic in grids

\[ \exists \epsilon, \beta \text{ such that } \Omega(n^{1+\epsilon}) \leq \frac{\text{gns}}{d}(G)_{n \times n} \leq O(n^{2-\beta}). \]

Clearly, \( \frac{\text{gns}}{d}(G) \leq \text{gns}(G) \) for any graph \( G \)

**Corollary**

\( \text{gns}(G)_{n \times n} = \Omega(n^{1+\epsilon}) \) for some \( \epsilon > 0 \)
**Fractional Spy-Game**

\( \text{frac-} gn_{s,d}(G) \): min amount of fractional guards required to win

**Theorem:** super-linear and sub-quadratic in grids

\[ \exists \epsilon, \beta \text{ such that } \Omega(n^{1+\epsilon}) \leq \text{frac-} gn_{s,d}(G_{n \times n}) \leq O(n^{2-\beta}). \]

Clearly, \( \text{frac-} gn_{s,d}(G) \leq gn_{s,d}(G) \) for any graph \( G \)

**Corollary**

\( gn_{s,d}(G_{n \times n}) = \Omega(n^{1+\epsilon}) \) for some \( \epsilon > 0 \)
Theorem:

\[ \exists \beta > 0 \text{ such that } \text{frac-}gn_{s,d}(G_n \times n) \leq O(n^{2-\beta}). \]

Goal: find such a strategy for the guards, i.e., define

\[ fg_t(v) : \text{amount of cops on } v \text{ at step } t \]

"Trick": Consider very simple strategy
time-independent + decreasing function of the distance

\[ fg(v) = \frac{1}{\text{dist}(v, \text{Spy})^\beta}. \]

main result (using flows and duality)

\[ \exists \beta \text{ such that such a distribution of guards can be preserved} \]

whatever be the move of the spy
Fractional Spy-Game

**Theorem:**

\[ \exists \beta > 0 \text{ such that } \frac{\text{frac}_s \cdot \text{gn}_d}{G_n \times n} \leq O(n^{2-\beta}). \]

**Goal:** find such a strategy for the guards, i.e., define

\[ f_{gt}(v): \text{amount of cops on } v \text{ at step } t \]

**“Trick”:**

Consider very simple strategy

\[ f_g(v) = \frac{1}{\text{dist}(v, \text{Spy})^\beta}. \]

main result (using flows and duality)

\[ \exists \beta \text{ such that such a distribution of guards can be preserved whatever be the move of the spy}. \]
Fractional Spy-Game

**Theorem:**
\[ \exists \beta > 0 \text{ such that } \frac{\text{frac-gn}_{s,d}(G_{n \times n})}{O(n^{2-\beta})}. \]

Goal: find such a strategy for the guards, i.e., define
\[ f_{g_t}(v) : \text{amount of cops on } v \text{ at step } t \]

"Trick": Consider very simple strategy

- time-independent + decreasing function of the distance

\[ f_{g}(v) = \frac{1}{\text{dist}(v, \text{Spy})^\beta}. \]

Main result (using flows and duality)
\[ \exists \beta \text{ such that such a distribution of guards can be preserved} \]
\[ \text{whatever be the move of the spy} \]
Theorem: \( \exists \alpha > 0 \) such that \( \Omega(n^{1+\alpha}) \leq \frac{g\text{n}_{s,d}(G_{n \times n})}{d} \).
Theorem: \( \exists \alpha > 0 \) such that \( \Omega(n^{1+\alpha}) \leq \frac{\text{gn}_s,d(G_{n \times n})}{n} \).

“Fractional version” of Lemma 1

Spy wins against \( O(n \log n) \) of guards, in \( O(n) \) steps in a grid.
Fractional Spy-Game

Theorem: \( \exists \alpha > 0 \) such that \( \Omega(n^{1+\alpha}) \leq \text{frac-gn}_{s,d}(G_{n \times n}) \).

“Fractional version” of Lemma 1
Spy wins against \( O(n \log n) \) of guards, in \( O(n) \) steps in a grid.

Key “fractional Lemma”
If Spy wins vs. \( c \) guards in \( t \) steps \( \Rightarrow \)

\( \forall \) strategy of \( k \) guards, \( \exists \) strategy for Spy s.t.

after \( t \) steps, at most \( k/c \) guards are at distance \( \leq d \).
Theorem: \( \exists \alpha > 0 \) such that \( \Omega(n^{1+\alpha}) \leq \frac{gn_s}{d}(G_{n \times n}) \).

"Fractional version" of Lemma 1
Spy wins against \( O(n \log n) \) of guards, in \( O(n) \) steps in a grid.

Key "fractional Lemma"
If Spy wins vs. \( c \) guards in \( t \) steps \( \Rightarrow \)
\( \forall \) strategy of \( k \) guards, \( \exists \) strategy for Spy s.t.
\[ \text{after } t \text{ steps, at most } k/c \text{ guards are at distance } \leq d. \]

In fractional version, the density can be reduced!
for \( a \in \mathbb{N}^* \), after at most \( 2n \) steps against \( k \) guards, the amount of guards at distance \( \leq 2n/a \) from the spy is \( < k(aH(a))^{-1} \).
Fractional Spy-Game

Theorem: \( \exists \alpha > 0 \) such that \( \Omega(n^{1+\alpha}) \leq \frac{\text{frac}-gn_{s,d}(G_{n \times n})}{d} \).

In fractional version, the density can be reduced! \( \Rightarrow \) induction
Theorem: $\exists \alpha > 0$ such that $\Omega(n^{1+\alpha}) \leq \frac{gn_{s,d}}{d}(G_{n\times n})$.

In fractional version, the density can be reduced! $\Rightarrow$ induction.
Theorem: \( \exists \alpha > 0 \text{ such that } \Omega(n^{1+\alpha}) \leq \frac{\text{frac}-\text{gn}_{s,d}(G_{n \times n})}{d(G_{n \times n})}. \)

In fractional version, the density can be reduced! \( \Rightarrow \) induction
On Cops and Robber games:

- **Meyniel Conjecture [1985]:**
  
  For any $n$-node connected graph $G$, $cn(G) = O(\sqrt{n})$

- How many cops with speed 1 to capture a robber with speed 2 in a grid?

On Spy game:

- Complexity of computing $gn_{s,d}$? of computing $frac-gn_{s,d}$?

- actual value of $gn_{s,d}(G)$ in $n \times n$ grid $G$?

  **We know:** $\exists \epsilon, \beta > 0,$
  
  $\Omega(n^{1+\epsilon}) \leq frac-gn \leq gn \leq O(n^2).$

  $frac-gn \leq O(n^{2-\beta}).$

- other graph classes: trees, bounded treewidth...?
On Cops and Robber games:

- **Meyniel Conjecture [1985]:**
  
  For any \( n \)-node connected graph \( G \), \( cn(G) = O(\sqrt{n}) \)

- How many cops with speed 1 to capture a robber with speed 2 in a grid?

On Spy game:

- Complexity of computing \( gn_{s,d} \)? of computing \( frac-gn_{s,d} \)?

- Actual value of \( gn_{s,d}(G) \) in \( n \times n \) grid \( G \)?

  **We know:** \( \exists \epsilon, \beta > 0, \Omega(n^{1+\epsilon}) \leq frac-gn \leq gn \leq O(n^2) \).

  \( frac-gn \leq O(n^{2-\beta}) \).

- Other graph classes: trees, bounded treewidth...?

Thank you!