



**HAL**  
open science

# Study of a combinatorial game in graphs through Linear Programming

Nathann Cohen, Fionn Mc Inerney, Nicolas Nisse, Stéphane Pérennes

► **To cite this version:**

Nathann Cohen, Fionn Mc Inerney, Nicolas Nisse, Stéphane Pérennes. Study of a combinatorial game in graphs through Linear Programming. 28th International Symposium on Algorithms and Computation (ISAAC 2017), 2017, Phuket, Thailand. 10.4230/LIPIcs . hal-01582091

**HAL Id: hal-01582091**

**<https://inria.hal.science/hal-01582091v1>**

Submitted on 5 Sep 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Study of a combinatorial game in graphs through Linear Programming

Nathann Cohen<sup>1</sup>, Fionn Mc Inerney<sup>2</sup>, Nicolas Nisse <sup>\*2</sup>, and Stéphane Pérennes<sup>2</sup>

<sup>1</sup> CNRS, Univ Paris Sud, LRI, Orsay, France

<sup>2</sup> Université Côte d’Azur, Inria, CNRS, I3S, France

---

## Abstract

In the *Spy Game* played on a graph  $G$ , a single spy travels the vertices of  $G$  at speed  $s$ , while multiple slow guards strive to have, at all times, one of them within distance  $d$  of that spy. In order to determine the smallest number of guards necessary for this task, we analyze the game through a Linear Programming formulation and the *fractional strategies* it yields for the guards. We then show the equivalence of fractional and integral strategies in trees. This allows us to design a polynomial-time algorithm for computing an optimal strategy in this class of graphs. Using duality in Linear Programming, we also provide non-trivial bounds on the fractional guard-number of grids and torus. We believe that the approach using fractional relaxation and Linear Programming is promising to obtain new results in the field of combinatorial games.

**1998 ACM Subject Classification** G.2.2 Graph Theory, F.2.2 Nonnumerical Algorithms and Problems, G.1.6 Optimization

**Keywords and phrases** Turn-by-turn games in graphs, Graph algorithms, Linear Programming

**Digital Object Identifier** 10.4230/LIPIcs...

## 1 Introduction

Turn-by-turn combinatorial games in graphs involve two players placing their pawns on vertices of a graph and moving them along its edges in order to achieve some task. For instance, in *Cops and Robber* games, Player 1 has a team of *cops* that must collaborate to capture a *robber* moved by the second player [5, 22, 23]. In the *surveillance game*, Player 1 has no pawn but is allowed to cover vertices at each of its turns, while the goal of Player 2 is to move its surfer to an uncovered vertex [11]. Another example is the *Eternal Domination Problem* in which Player 2 has no pawn but is allowed to *attack* any vertex at each of its turns, while the goal of Player 1 is to always be able to move at least one of its cops to the attacked vertex [15]. Most of these games have been studied because they model natural problems involving mobile agents cooperating to perform some task (e.g., network security, robot motion planning). Surprisingly, these games can also be used to provide a novel understanding of problems arising in telecommunication networks: for instance, the surveillance game was introduced for modelling resources prefetching [11].

In all these games, the goal is to minimize the amount of resources (e.g., the number of cops) ensuring the victory of one of the players. These combinatorial problems are generally “hard”: Cops and Robber games are EXPTIME-complete [18] and W[2]-hard [12] and the Surveillance game is NP-hard and even PSPACE-complete in directed graphs [11]. Moreover,

---

\* Email address: nicolas.nisse@inria.fr



many longstanding (probably difficult) open questions have not been solved yet for these games. For instance, the celebrated Meyniel’s conjecture states that  $O(\sqrt{n})$  cops are sufficient to capture a robber in any  $n$ -node graph [25], Schröder asks whether  $g + 3$  cops are sufficient in any graph with genus at most  $g$  [24], the status of the complexity of Eternal Domination is still unknown, etc. A classical approach to tackle these open problems has been to study variants of these games in which one of the players often has some restrictions. For instance, the robber may be faster than the cops [2, 12], the cops may capture at some distance [4], the surveyed area may be forced to be connected [13], etc. Another approach may be to restrict the games to particular graph classes such as trees [19], grids [16, 21], planar graphs [1], bounded treewidth graphs [17, 20], etc.

Recently, some of the authors of the present paper proposed a new framework that considers a fractional variant of these games (roughly where pawns may be split into arbitrarily small entities) and uses Linear Programming to obtain new bounds and algorithms [8, 14]. While this approach seems not to be successful to handle Cops and Robber games, it has been fruitful in designing approximation algorithms for other combinatorial games. Precisely, it allowed to design polynomial-time approximation algorithms for various (NP-hard) variants of the surveillance game [14]. In this paper, we present a new successful application of this approach. In particular, we consider the *Spy-game* [8, 9] and show that it can be solved in polynomial-time in trees using this approach. We emphasize that, as far as we know, it is the first exact algorithm for such combinatorial games using a Linear Programming approach and that we were not able to solve it without this technique. We hope that our results will encourage people to use this framework to study combinatorial games and we believe it will enable progress toward solutions of the difficult open problems.

**Spy-game.** The *Spy-game* has been defined as it is closely related to the Cops and fast robber game and it generalizes the Eternal Domination Problem [8, 9]. The Spy-game is a turn-by-turn 2-Player game with perfect information. The first player has a *spy* which is first placed at some vertex of a graph  $G$ . The second player has  $k \in \mathbb{N}$  *guards* that are then placed at some vertices of  $G$ . Turn-by-turn, the spy may move along at most  $s \geq 1$  edges ( $s \in \mathbb{N}^*$  is the *speed* of the spy), and then each guard may move along one edge. Any number of guards and the spy may occupy the same vertex. The goal of the game is to minimize the number of guards, called *guard-number* and denoted by  $gn_{s,d}(G)$ , ensuring that, at every step after the guards’ turn, the spy is at distance at most  $d \in \mathbb{N}$  from at least one guard (we say that the spy is *controlled* at distance  $d$ ). Note that, when  $d = 0$  and  $s$  is large (at least the diameter of  $G$ ), the Spy-game is equivalent to the Eternal Domination Problem. The guard-numbers of paths and cycles, and corresponding optimal guards’ strategies, have been characterized in [8]. To tackle the more difficult case of trees, we consider the fractional variant of the Spy-game, in which the rules are unchanged for the spy but the guards can be split into arbitrarily small entities.

**Fractional Spy-game.** Formally, the fractional Spy-game proceeds as follows in a graph  $G = (V, E)$ . Let  $s \geq 2$ ,  $d \geq 0$  be two integers and let  $k \in \mathbb{R}$  such that  $k > 0$ . First, the spy is placed at a vertex. Then, each vertex  $v$  receives some amount  $g_v \in \mathbb{R}^+$  (a non negative real) of guards such that the total amount of guards is  $\sum_{v \in V} g_v = k$ . Then, turn-by-turn, the spy may first move at distance at most  $s$  from its current position. Then, the “fractional” guards move following a flow constrained as follows. For any  $v \in V$ , and for any  $u \in N[v]^1$ , there is a flow  $f(v, u) \in \mathbb{R}^+$  of guards going from  $v$  to  $u \in N[v]$ , such that  $\sum_{u \in N[v]} f(v, u) = g_v$ , i.e., the

---

<sup>1</sup> For any graph  $G$ , any integer  $\ell$  and  $v \in V(G)$ , let  $N_\ell[v]$  be the set of vertices at distance at most  $\ell$  from  $v$  in  $G$  and let  $N[v] = N_1[v]$ .

amount of guards leaving  $v$  and staying at  $v$  is exactly what was at  $v$ . Finally, for any vertex  $v \in V$ , the amount of guards occupying  $v$  at the end of the turn is  $g'_v = \sum_{u \in N[v]} f(u, v)$ . We now need to rephrase the fact that the guards control the spy at distance  $d$  at every turn. This is the case if, after every guards' turn,  $\sum_{w \in N_d[x]} g'_w \geq 1$ , where  $x$  is the vertex occupied by the spy. Let  $fgn_{s,d}(G)$  denote the minimum total amount of fractional guards needed to always control at distance  $d$  a spy with speed  $s$  in a graph  $G$ . Note that, by definition, since the fractional game is a relaxation of the ‘‘integral’’ Spy-game: for any graph  $G$  and any  $s \geq 2$ ,  $d \geq 0$ ,  $fgn_{s,d}(G) \leq gn_{s,d}(G)$ . The fractional variant of the Spy-game has been used to show the first non-trivial lower bound on the guard-number of grids [8]. In this paper, we will give the first exact algorithm using the framework of fractional combinatorial games.

## 1.1 Our results

We study the Spy-game in the classes of trees and grids. We prove that the guard-number of any tree can be computed in polynomial-time and give a non-trivial upper bound on the fractional guard-number of grids. More precisely, for every  $s \geq 2$  and  $d \geq 0$ :

- We design a Linear Program that computes  $fgn_{s,d}(T)$  and a corresponding strategy in polynomial-time for any tree  $T$ . Then, we show that any fractional strategy (winning for the guards) using  $k$  guards in a tree can be turned into a winning (integral) strategy using  $\lfloor k \rfloor$  guards. The key argument is that we can restrict the study to what we call *Spy-positional* strategies. Altogether, this shows that, in any tree  $T$ ,  $fgn_{s,d}(T) = gn_{s,d}(T)$ , and that  $gn_{s,d}(T)$  and a corresponding winning strategy can be computed in polynomial-time.
- Then, we show that there exists a constant  $0 < \alpha \leq \log(3/2)$  such that  $fgn_{s,d}(G_{n \times n}) = O(n^{2-\alpha})$ . Note that the best known upper bound for  $gn_{s,d}(G_{n \times n})$  is  $O(n^2)$ . A similar bound holds for the  $n \times n$  torus.

We believe that the methods using Linear Programming used in this paper are a promising way to better understand other combinatorial games in graphs.

## 1.2 Related Work

**Spy-game.** The Spy-game has been defined in [8, 9]. It has been shown that, for every  $d \geq 0$  and  $s \geq 2$ , computing  $gn_{s,d}(G)$  is NP-hard in a subclass of chordal graphs (precisely, graphs obtained from a clique and some paths, where one end of each path is connected to some vertices of the clique) [8, 9]. The guard-number of paths is also characterized and almost tight lower and upper bounds are given in the case of cycles. More precisely,  $gn_{s,d}(P) = \left\lceil \frac{n}{2d+2+\lfloor \frac{2d}{s-1} \rfloor} \right\rceil$  for any  $n$ -node path  $P$ . Moreover, the strategy consists of partitioning the path into  $gn_{s,d}(P)$  subpaths with one guard assigned to each one [8, 9]. We show that such a strategy (assigning disjoint subtrees to each guard) is not necessarily optimal in trees (see Section 4). Finally, it was proven using a fractional relaxation of the game that there exists  $\beta > 0$ , such that  $\Omega(n^{1+\beta})$  guards are necessary to win in an  $n \times n$ -grid  $G_{n \times n}$  [8]. Precisely, it was proven that  $fgn_{s,d}(G_{n \times n}) = \Omega(n^{1+\beta})$  and the result follows from the fact that  $fgn_{s,d}(G) \leq gn_{s,d}(G)$  for any graph  $G$ . Note that no direct (without using the fractional variant) proof is known. Note also that the best known upper bound in grids is the trivial one:  $O(n^2)$ . The difficulty of finding the exact value of the guard-number of grids can be related to the difficulty of finding the exact number of cops required to capture a fast robber in a grid [2, 12]. We believe that obtaining results or tools in one case would lead to further progress in the other case.

**Eternal Domination.** The Spy game generalizes the *Eternal Domination Problem* [15]. In the latter game, a team of mobile agents (cops) occupy some vertices of a graph. At every

turn, the second player *attacks* some vertex  $v$  and then each of the cops is allowed to move to one of its neighbors or may stay idle such that at least one cop occupies  $v$  (note that in the original variant, only one agent was allowed to move at each turn [7]). In other words, the agents must always occupy a dominating set  $D$ , such that for any vertex  $v \notin D$ , the agents can move to another dominating set containing  $v$ . The minimum number of agents ensuring to win the game in a graph  $G$  is denoted by  $\gamma^m(G)$ . It is easy to see that the Eternal Domination Problem is equivalent to the Spy game when the spy is arbitrarily fast and  $d = 0$ , i.e.,  $\gamma^m(G) = gn_{s,0}(G)$  for any  $s$  which is at least the diameter of the graph. Therefore, our results apply to the Eternal Domination Problem.

Eternal Domination has been investigated in several graph classes. In grids, only a few cases are known: for instance, tight bounds are known in  $m \times n$  grids for  $n \leq 4$  [3, 10] and the case  $n = 5$  is considered in [26]. The best known general upper bound in grids is  $\left\lceil \frac{nm}{5} \right\rceil + O(n + m)$  [21]. Note that the minimum size of a dominating set in any grid has only been characterized recently [16]. In the class of trees  $T$ ,  $\gamma^m(T)$  can be computed in polynomial-time [19]. The key property in this simple recursive algorithm is that an optimal strategy consists of partitioning a tree into vertex-disjoint stars, each star being assigned to at most 2 cops. As already mentioned, such a method does not extend for the Spy-game.

## 2 Representation of winning strategies and Spy-positional strategies

In this paper, all graphs are simple (without loops nor multi-edges), connected, and undirected. For any vertex  $v \in V$  in a graph  $G = (V, E)$ , let  $N(v)$  denote the set of neighbors of  $v$  and  $N[v] = N(v) \cup \{v\}$ . Moreover, for any integer  $s \geq 0$  and vertex  $v \in V$ , let  $N_s[v]$  be the set of vertices at distance at most  $s$  from  $v$ .

A strategy for the guards is a function describing the moves of the guards at every step. A strategy is *winning* if it allows the guards to perpetually control the spy. It is easy to show that there is always an optimal winning strategy (using the minimum number of guards) which is *positional*, i.e. such that the next move is only determined by the current position of both the spy and the guards, and not by the history of the game<sup>2</sup>. In other words, there is always an optimal winning strategy which is a function that takes the current positions of the spy and of the guards and returns the new positions of the guards (and so, their moves).

**Representation of (fractional) guards' strategies.** Let  $G = (V, E)$  be an  $n$ -node graph,  $s \geq 2$  and  $d \geq 0$  be two integers. Let  $V = \{v_1, \dots, v_n\}$ . A winning strategy  $\sigma$  using  $k \in \mathbb{R}^+$  guards is defined as a set  $\sigma = \{\mathcal{C}_v\}_{v \in V}$  of sets of configurations. That is, for any  $v \in V$  (a possible position for the spy),  $\mathcal{C}_v$  is a non-empty set of functions, called *configurations*, that represent the possible positions of the guards when the spy is at  $v$ . More precisely, any  $\omega \in \mathcal{C}_v$  is a function  $\omega : V \rightarrow \mathbb{R}^+$ , where  $\omega(u) \in \mathbb{R}^+$  represents the amount of guards at vertex  $u \in V$  when the spy occupies  $v$ , that must satisfy  $\sum_{u \in V} \omega(u) = k$  and  $\sum_{u \in N_d[v]} \omega(u) \geq 1$ . Finally, for any  $v \in V$ , any  $\omega \in \mathcal{C}_v$ , and any  $v' \in N_s[v]$ , there must exist  $\omega' \in \mathcal{C}_{v'}$  such that the guards can go from  $\omega$  to  $\omega'$  in one step. That is, for any possible move of the spy (from  $v$  to  $v'$ ), there must exist a valid flow from  $\omega$  to  $\omega'$  (the guards must be able to reach a configuration controlling the spy in  $v'$ ). A strategy is *integral* if  $k \in \mathbb{N}^+$ , every of its configurations is a function  $V \rightarrow \mathbb{N}$ , and every move is an integral flow. The *size* of a strategy is the number of different configurations necessary to describe the strategy, i.e., the size of  $\sigma$  is  $\sum_{v \in V} |\mathcal{C}_v|$ . Note that, a same position for the spy may correspond to different positions of

<sup>2</sup> That can be easily shown by considering the configurations' graph of the game.

the guards. Therefore, the size of an integral strategy using  $k$  guards in an  $n$ -node graph is  $n^{O(k)}$ . Moreover, the size of a fractional strategy is *a priori* unbounded.

**Spy-positional strategies.** In this paper, we will also consider more constrained strategies. A winning strategy is said to be *Spy-positional* if it depends only on the position of the spy. That is, in a spy-positional strategy  $\sigma = \{\mathcal{C}_v\}_{v \in V}$ , the positions of the guards are only determined by the position of the spy. In particular, every time the spy occupies some vertex  $v$ , the set of vertices occupied by the guards is defined by a unique function  $\sigma_v : V(G) \rightarrow \mathbb{N}$  such that, for every  $u \in V$ ,  $\sigma_v(u)$  is the number of guards occupying  $u$  when the spy is occupying  $v$ . That is,  $\mathcal{C}_v = \{\sigma_v\}$  and  $|\mathcal{C}_v| = 1$  for every  $v \in V$ . An important consequence for our purpose is that any (fractional or integral) spy-positional strategy has size  $O(n)$ .

Let us remark that, in a spy-positional strategy, it is not required that the same guards occupy the same vertices when the spy is at some vertex. That is, assume that, at some step, the spy occupies some vertex  $v$ , some Guard  $A$  occupies a vertex  $a$  and a guard  $B$  occupies a vertex  $b$ . It may happen that, after some steps, the spy goes back to  $v$  and now Guard  $A$  is at  $b$  and Guard  $B$  is at  $a$  (however, the set of vertices occupied by the guards is the same).

Second, there does not always exist an optimal strategy (using the minimum number of guards) that is spy-positional. As an example, consider the cycle  $C_5$  with 5 vertices  $\{a, b, c, d, e\}$ . It is easy to show that  $gn_{2,1}(C_5) = 1$  but that every spy-positional strategy needs 2 guards. One of our main results is to show that, in trees, there always exists an optimal strategy which is spy-positional.

Let  $fgn_{s,d}^*(G)$  be the minimum total amount of fractional guards needed to always control at distance  $d$  a spy with speed  $s$  in a graph  $G$ , when the guards are constrained to play spy-positional strategies. By definition, for any graph  $G$  and any  $s \geq 2$ ,  $d \geq 0$ ,

$$fgn_{s,d}(G) \leq \min\{fgn_{s,d}^*(G), gn_{s,d}(G)\}.$$

### 3 Spy-positional fractional strategies in general graphs

This section is devoted to present a polynomial-time algorithm that computes optimal spy-positional fractional strategies in general graphs. Here, optimal means using the minimum total amount of guards with the extra constraint that guards are restricted to play spy-positional strategies. In other words, we prove that, for any graph  $G$ ,  $s \geq 2$ , and  $d \geq 0$ ,  $fgn_{s,d}^*(G)$  and a corresponding strategy can be computed in polynomial time.

We prove this result by describing a Linear Program with polynomial size that computes such strategies. In Section 4, we will show that in any tree  $T$ ,  $gn_{s,d}(T) = fgn_{s,d}^*(T)$ . More precisely, we will show that in trees, the Linear Program below can be used to compute optimal (integral) strategies in polynomial time.

We describe a Linear Program for computing an optimal fractional spy-positional strategy.

**Variables.** Let  $G = (V, E)$  be a connected  $n$ -node graph. Recall that a spy-positional strategy is defined by, for each position of the spy, the amount of guards that must occupy each vertex. Therefore, for any two vertices  $u, v \in V$ , let  $\sigma_v(u) \in \mathbb{R}^+$  be the non negative real variable representing the amount of guards occupying vertex  $u$  when the spy is at  $v$ .

Moreover, for any  $x \in V$ ,  $y \in N_s[x]$  and for any  $u \in V$  and  $v \in N[u]$ , let  $f_{x,y,u,v} \in \mathbb{R}^+$  be the non negative real variable representing the amount of guards going from vertex  $u$  to  $v \in N[u]$  when the spy goes from  $x$  to  $y \in N_s[x]$ . Finally, a variable  $k$  will represent the total amount of guards. Overall, there are  $O((|E| + 1)n^2) = O(n^4)$  real variables.

These variables fully describe a strategy, since  $\sigma$  encodes a distribution of cops for every position of the spy and  $f$  describes a feasible transition between two successive distributions.

**Objective function.** We aim at minimizing the total amount of guards.

$$\text{Minimize } k. \tag{1}$$

**Constraints.** The first family of constraints states that, for every position  $v \in V$  of the spy, the total amount of guards is at most  $k$ .

$$\forall v \in V, \sum_{u \in V} \sigma_v(u) \leq k. \tag{2}$$

The second family of constraints states that, for every position  $v \in V$  of the spy, the amount of guards at distance at most  $d$  from the spy is at least 1, i.e., the guards always control the spy at distance  $d$ .

$$\forall v \in V, \sum_{u \in N_d[v]} \sigma_v(u) \geq 1. \tag{3}$$

The third family of constraints states that, for any move of the spy (from  $x$  to  $y \in N_s[x]$ ), the corresponding moves of the guards ensure that the amount of guards leaving a vertex  $v \in V$  plus what remains at  $v$  equals the amount of guards that was at  $v$  before the move.

$$\forall x \in V, y \in N_s[x], v \in V, \sum_{w \in N[v]} f_{x,y,v,w} = \sigma_x(v). \tag{4}$$

The fourth family of constraints states that, for any move of the spy (from  $x$  to  $y \in N_s[x]$ ), the corresponding moves of the guards ensure that the amount of guards that are at a vertex  $w \in V$  after the moves equals the amount of guards arriving in  $w$  plus what remains at  $w$ .

$$\forall x \in V, y \in N_s[x], w \in V, \sum_{v \in N[w]} f_{x,y,v,w} = \sigma_y(w). \tag{5}$$

There are  $O(n^4)$  constraints and the above Linear Program has polynomial size and clearly computes an optimal spy-positional fractional strategy. Hence:

► **Theorem 1.** *For any connected graph  $G$ , and any two integers  $s \geq 2$  and  $d \geq 0$ , the above Linear Program computes  $fgn_{s,d}^*(G)$  and a corresponding spy-positional strategy in polynomial time.*

## 4 Spy game is Polynomial in Trees

This section is devoted to the study of the Spy-game in trees (Theorem 4). Before going into the details, we would like to emphasize one difficulty when dealing with guards' strategies.

A natural idea would be to partition the tree into smaller subtrees (with bounded diameter) with a constant number of guards assigned to each of them. That is, each guard would be assigned (possibly with other guards) a subtree  $S$  and would move only when the spy is in  $S$  (in particular, the guard would only occupy some vertices of  $S$ ). As already mentioned, there exist such strategies that are optimal in paths [8, 9] or in trees when  $d = 0$  and  $s$  is large (Eternal Domination) [19]. We show that we cannot expect such strategies for the Spy-game (for any  $s \geq 2$  and  $d > 0$ ) in trees. Precisely, we present a family of trees with unbounded guard-number such that, for each of these trees, there is a strategy of the spy that forces every guard to occupy every non-leaf vertex infinitely often (Claim 9 in Appendix 6.1). Hence, optimal guards' strategies seem difficult to be described in trees.

To overcome this difficulty, we use the power of Linear Programming. Precisely, we prove that, in any tree  $T$  and for any  $s \geq 2$ ,  $d \geq 0$ ,  $gn_{s,d}(T) = fgn_{s,d}^*(T)$ . Therefore, using the Linear Program of Section 3, it follows that computing  $gn_{s,d}(T)$  can be done in polynomial time in trees. The proof is twofold. First, we prove that  $gn_{s,d}(T) = fgn_{s,d}(T)$  for any  $s \geq 2$  and  $d \geq 0$  (i.e., the integrality gap is null in trees), and then that  $fgn_{s,d}(T) = fgn_{s,d}^*(T)$ .



► **Theorem 2.** *For any tree  $T$  and for any  $s \geq 2$ ,  $d \geq 0$ ,  $gn_{s,d}(T) = fgn_{s,d}(T)$ . More precisely, any fractional winning strategy using a total amount of  $k \in \mathbb{R}^+$  guards can be transformed into an integral winning strategy using  $\lfloor k \rfloor$  guards. Moreover, such a transformation can be done in polynomial time in the size of the fractional strategy.*

**Proof.** Let  $\sigma = \{\mathcal{C}_v\}_{v \in V}$  be any fractional winning strategy using a total amount of  $k \in \mathbb{R}^+$  guards to control a spy with speed  $s \geq 2$ , at distance  $d \geq 0$ , and in an  $n$ -node tree  $T = (V, E)$ .

We build a winning integral strategy  $\sigma^r$  using  $\lfloor k \rfloor$  guards by “rounding” all configurations of  $\sigma$ . For any configuration  $\omega$  of  $\sigma$ , we will define an integral configuration  $\omega^r$  (which we call a *rounding* of  $\omega$ ) using  $\lfloor k \rfloor$  guards (Claim 1), such that if the spy is controlled in  $\omega$  then it is also controlled in  $\omega^r$  (Claim 2). Moreover, for any two configurations  $\omega_1$  and  $\omega_2$  such that there is a feasible flow from  $\omega_1$  to  $\omega_2$ , we show that there is feasible integral flow from  $\omega_1^r$  to  $\omega_2^r$  (Claim 3). Altogether, this shows that  $\sigma^r$  is a winning integral strategy using  $\lfloor k \rfloor$  guards, which proves the theorem. The omitted proofs of the claims below are in Appendix 6.2.

From now on, let us consider  $T$  to be rooted at some vertex  $r \in V$ .

**Notations.** For any  $u \in V$ , let  $T_u$  be the subtree of  $T$  rooted in  $u$  (i.e., the subtree that consists of  $u$  and all its descendants). For any configuration  $\omega : V \rightarrow \mathbb{R}^+$ , let  $\omega(T_u) = \sum_{v \in V(T_u)} \omega(v)$  and let  $\omega(T) = \omega(T_r)$ . By definition,  $\omega(T_u) \geq \omega(u)$  for every  $u \in V$ . Finally, let  $cont(T, \omega) = \{u \in V : \sum_{v \in N_d[u]} \omega(v) \geq 1\}$  (i.e.,  $cont(T, \omega)$  is the set of vertices  $u$  such that the spy on  $u$  is controlled at distance  $d$  by the guards in the configuration  $\omega$ ).

Let us define the *rounded* configuration  $\omega^r : V \mapsto \mathbb{N}$  as, for every  $u \in V$ ,

$$\omega^r(u) = \left\lfloor \omega(u) + \sum_{v \text{ child of } u} (\omega(T_v) - \lfloor \omega(T_v) \rfloor) \right\rfloor$$

Intuitively, the fractional part of guards that are in each of the subtrees rooted in the children of  $u$  is “pushed” to  $u$ . Then  $u$  “keeps” only the integral part of the sum of what it had plus what it received from its children.

We first prove that rounding a configuration using  $k$  guards provides an integral configuration using  $\lfloor k \rfloor$  guards. A simple induction on  $n$  shows that:

► **Claim 1.** *For any configuration  $\omega : V(T) \rightarrow \mathbb{R}^+$ ,  $\omega^r(T) = \lfloor \omega(T) \rfloor$*

Then, Claim 2 proves that every position of the spy that is controlled by the guards in a configuration  $\omega$  is also controlled by the guards in the configuration  $\omega^r$ . We prove that, for every  $v \in V$  and  $\ell \in \mathbb{N}$ ,  $\sum_{x \in N_\ell[v]} \omega^r(x) \geq \lfloor \sum_{x \in N_\ell[v]} \omega(x) \rfloor$ . Hence, if  $u \in cont(T, \omega)$ , i.e.,  $1 \leq \sum_{x \in N_d[u]} \omega(x)$ , then  $1 \leq \lfloor \sum_{x \in N_d[u]} \omega(x) \rfloor \leq \sum_{x \in N_d[u]} \omega^r(x)$  and so  $u \in cont(T, \omega^r)$ .

► **Claim 2.** *For any configuration  $\omega : V(T) \rightarrow \mathbb{R}^+$ ,  $cont(T, \omega) \subseteq cont(T, \omega^r)$*

Claim 3 shows that the moves that were valid in  $\sigma$  still hold in the “rounded” strategy. Its proof is postponed to Appendix 6.2.

► **Claim 3.** *Let  $\omega_1, \omega_2 : V(T) \mapsto \mathbb{R}^+$  be two configurations such that the guards can go from  $\omega_1$  to  $\omega_2$  in one step (there is feasible flow from  $\omega_1$  to  $\omega_2$ ). Then, the guards can go from  $\omega_1^r$  to  $\omega_2^r$  in one step (there is feasible integral flow from  $\omega_1^r$  to  $\omega_2^r$ ).*

This concludes the proof of Theorem 2. ◀

The second step in this section is to show that there is always an optimal fractional strategy which is spy-positional. For this purpose, we prove the following theorem.



► **Theorem 3.** *For any tree  $T$  and for any  $s \geq 2$ ,  $d \geq 0$ ,  $fgn_{s,d}^*(T) = fgn_{s,d}(T)$ . More precisely, any fractional winning strategy using a total amount of  $k \in \mathbb{R}^+$  guards can be transformed into a spy-positional winning strategy using  $k$  guards.*

**Proof.** Let  $\sigma = \{\mathcal{C}_v\}_{v \in V}$  be any fractional winning strategy using a total amount of  $k \in \mathbb{R}^+$  guards to control a spy with speed  $s \geq 2$ , at distance  $d \geq 0$ , and in an  $n$ -node tree  $T = (V, E)$ . Recall that, for any vertex  $v \in V$ ,  $\mathcal{C}_v$  is the set of possible configurations  $\omega : V \rightarrow \mathbb{R}^+$  for the guards when the spy is at  $v$ .

The proof consists in defining a spy-positional strategy  $\sigma^{min}$  that is a winning strategy using  $k$  guards. For any  $v \in V$ , we will define the function  $\omega_v^{min} : V \rightarrow \mathbb{R}^+$  to be the (unique) configuration of  $\sigma^{min}$  when the spy is at  $v$ , i.e.,  $\sigma^{min} = \{\omega_v^{min}\}_{v \in V}$ . We first prove that  $\sigma^{min}$  is a strategy using  $k$  guards (Claims 4-5), then that the spy at  $v \in V$  is controlled at distance  $d$  by the guards in the configuration  $\omega_v^{min}$  (Claim 6). Finally, we prove that, for any move of the spy from  $v$  to  $v' \in V$ , the guards can move from  $\omega_v^{min}$  to  $\omega_{v'}^{min}$  (Claim 7).

From now on,  $T$  is rooted in an arbitrary vertex  $r \in V$ .

**Notations.** For any weight function  $\omega : V \rightarrow \mathbb{R}^+$ , let  $\omega^+ : V \rightarrow \mathbb{R}^+$  be the *cumulative* function of  $\omega$ , defined by, for every  $u \in V$ ,  $\omega^+(u) = \sum_{v \in V(T_u)} \omega(v) = \omega(T_u)$ . Let  $v \in V$  and  $\mathcal{C}_v = \{\omega_1, \dots, \omega_h\} \in \sigma$  be the set of configurations of the guards, when the spy is in  $v$ . Let  $\alpha_v : V \rightarrow \mathbb{R}^+$  be such that, for every  $u \in V$ ,  $\alpha_v(u) = \min_{1 \leq i \leq h} \omega_i^+(u)$ . Now,  $\omega_v^{min}$  is defined as the (unique) function such that  $\alpha_v$  is its cumulative function, i.e.,  $\alpha_v = (\omega_v^{min})^+$ . Formally, for every  $u \in V$ :  $\omega_v^{min}(u) = \alpha_v(u) - \sum_{x \text{ child of } u} \alpha_v(x)$ .

Claim 4 proves that, for every  $v \in V$ ,  $\omega_v^{min} : V \rightarrow \mathbb{R}^+$  is a configuration.

► **Claim 4.** *For every  $u \in V$ ,  $\omega_v^{min}(u) \geq 0$ .*

*Proof of claim.* Let  $1 \leq i \leq h$  be an integer such that  $\alpha_v(u) = \min_{1 \leq j \leq h} \omega_j^+(u) = \omega_i^+(u)$ . By definition of  $\alpha_v$ , for every  $x \in \text{Children}(u)$ ,  $\alpha_v(x) = \min_{1 \leq j \leq h} \omega_j^+(x) \geq \omega_i^+(x)$ . Hence,  $\omega_v^{min}(u) \geq \omega_i^+(u) - \sum_{x \in \text{Children}(u)} \omega_i^+(x) = \omega_i(u) \geq 0$ . ◊

Claim 5 proves that, for every  $v \in V$ , the configuration  $\omega_v^{min}$  uses  $k$  guards.

► **Claim 5.** *For every  $v \in V$ ,  $\sum_{u \in V} \omega_v^{min}(u) = k$ .*

*Proof of claim.* For every  $1 \leq i \leq h$ ,  $\omega_i^+(r) = k$ . Hence,  $\alpha_v(r) = \min_{1 \leq i \leq h} \omega_i^+(r) = k$ .  $\sum_{u \in V} \omega_v^{min}(u) = (\omega_v^{min})^+(r) = \alpha_v(r) = k$  (since  $\alpha_v$  is the cumulative function of  $\omega_v^{min}$ ). ◊

Claim 6 proves that the guards in the configuration  $\omega_v^{min}$  control a spy located at  $v$ . Finally, Claim 7 shows that the moves that were valid in  $\sigma$  still hold for  $\sigma^{min}$ . Their proofs are postponed to Appendix 6.2.

► **Claim 6.** *For every  $v \in V$ ,  $\sum_{u \in N_d[v]} \omega_v^{min}(u) \geq 1$ .*

► **Claim 7.** *For every  $v \in V$  and  $v' \in N_s[v]$ , there is a feasible flow from  $\omega_v^{min}$  to  $\omega_{v'}^{min}$ .*

This concludes the proof of Theorem 3. ◀

We can now prove the main theorem of this section.

► **Theorem 4.** *Let  $s \geq 2$  and  $d \geq 0$  be two integers. There is a polynomial-time algorithm that computes an integral winning strategy using  $gn_{s,d}(T)$  guards to control a spy with speed  $s$  at distance  $d$  in any tree  $T$ .*

**Proof.** By Theorem 3, there exists an optimal (fractional) winning strategy that is spy-positional. By Theorem 1, such a strategy can be computed in polynomial time. By Theorem 2, an optimal integral winning strategy can be computed in polynomial time from any optimal fractional winning strategy. ◀

## 5 Fractional Spy-game in Grid and Torus

In this section, we provide some progress toward the understanding of the Spy-game in grids. Precisely, we provide the first fractional strategy using a sub-linear (in the number of vertices) number of guards. It is clear that, for any  $n \times n$  grid  $G_{n \times n}$ ,  $gn_{s,d}(G_{n \times n}) = O(n^2)$ , and it is known that  $fgn_{s,d}(G_{n \times n})$  is super-linear in  $n$  [8]. However, the exact order of magnitude of  $gn_{s,d}(G_{n \times n})$  (and of  $fgn_{s,d}(G_{n \times n})$ ) is not known. We prove that  $fgn_{s,d}(G_{n \times n})$  is sub-quadratic in  $n$ .

Let  $n, m \geq 2$  be two integers. We consider the  $n \times m$  toroidal grid  $TG_{n \times m} = (V, E)$ , i.e., the graph with vertices  $v_{i,j} = (i, j)$  and edges  $\{(i, j), (i + 1 \bmod n, j)\}$  and  $\{(i, j), (i, j + 1 \bmod m)\}$ , for all  $0 \leq i < n$  and  $0 \leq j < m$ . The  $n \times m$  grid  $G_{n \times m}$  is obtained from  $TG_{n \times m}$  by removing the edges  $\{(i, m - 1), (i, 0)\}; \{(n - 1, j), (0, j)\} \mid \forall 0 \leq i < n, 0 \leq j < m\}$ .

First, we show that the number of fractional (resp., integral) guards required in the grid and in the torus have the same order of magnitude. Informally, the proof consists in considering a strategy  $\mathcal{S}$  in a grid (resp., in a torus) and in applying in the torus (resp., in the grid) four symmetric strategies, each one mimicking  $\mathcal{S}$  (see Appendix 6.3.1).

► **Lemma 5.** *For every  $n, m \geq 2$ ,  $s \geq 2$ ,  $d \geq 0$ , and for every  $f \in \{gn_{s,d}, fgn_{s,d}, fgn_{s,d}^*\}$ :*

$$f(TG_{n \times m})/4 \leq f(G_{n \times m}) \leq 4 \cdot f(TG_{n \times m}).$$

The remaining part of this section is devoted to prove the following theorem.

► **Theorem 6.** *There exists  $0 < \alpha \leq \log(3/2) \approx 0.58$  such that, for every  $s \geq 2$ ,  $d \geq 0$ ,*

$$fgn_{s,d}^*(TG_{n \times n}) = O(n^{2-\alpha}).$$

To prove Theorem 6, we make use of the Linear Program (LP) of Section 3. Recall that, in a spy-positional strategy, the positions of the guards (configuration) only depends on the position of the spy. In any vertex-transitive graph (so in  $TG_{n \times n}$ ), there is actually a unique configuration to be considered (where the spy is occupying the vertex  $(0, 0)$ ). Therefore, the LP of Section 3 can be reformulated as follows.

We are looking for a function  $\omega : \{0, \dots, n - 1\}^2 \rightarrow \mathbb{R}^+$  such that  $\omega(i, j)$  is the amount of guards occupying the vertex  $(i, j)$  when the spy is occupying the vertex  $(0, 0)$ . This function must be defined such as to minimize the number of guards, i.e.,  $\sum_{0 \leq i, j < n} \omega(i, j)$  must be minimum, subject to the following constraints. The spy must be controlled, i.e.,  $\sum_{(i,j) \in N_s[(0,0)]} \omega(i, j) \geq 1$ . Moreover, for any move of the spy from  $(0, 0)$  to  $(x, y) \in N_s[(0, 0)]$ , there must be a feasible flow from the configuration  $(\omega(i, j))_{(i,j) \in V(TG_{n \times n})}$  to  $(\omega(i - x, j - y))_{(i,j) \in V(TG_{n \times n})}$ . Before going further, let us simplify the latter constraint. Indeed, instead of considering every possible move of the spy in  $N_s[(0, 0)]$ , we only consider the *extremal moves* from  $(0, 0)$  to one of the vertices in  $\{(0, s), (s, 0), (-s, 0), (0, -s)\}$ , i.e., we weaken the spy by allowing it to move only “horizontally” or “vertically” at full speed. We prove in Lemma 6 in Appendix 6.3.2 that it does not change the order of magnitude of an optimal solution.

The above LP, restricted to vertex-transitive graphs, is more efficient than the one presented in Section 3 since there is only one configuration to be considered and less flow constraints (and so, much less variables and constraints). In particular, it gives interesting experimental results as presented in the conclusion. In what follows, we present and analyze a function using a sub-quadratic (in  $n$ ) number of guards that satisfies the above LP.

Precisely, let  $0 < \alpha < 1$  and let  $d(v)$  (resp.,  $d(i, j)$ ) denote the distance between vertex  $v$  (resp.,  $(i, j)$ ) and vertex  $(0, 0)$  in  $TG_{n \times n}$ .

► **Definition 7** (Strategies  $\omega_\alpha$ ). Let us consider the spy-positional strategy  $\omega_\alpha$  of the form  $\omega_\alpha(i, j) = \frac{B}{(d(i, j) + 1)^\alpha}$  for every  $(i, j) \in V(TG_{n \times n})$  and for some constant  $B$  defined later.

Note that  $\omega_\alpha$  is *symmetric*, i.e.,  $\omega_\alpha(i, j) = \omega_\alpha(n - i, j) = \omega_\alpha(i, n - j \bmod n) = \omega_\alpha(n - i, n - j \bmod n)$ . Therefore, by symmetry, we only need to check that there is a feasible flow from the configuration  $(\omega_\alpha(i, j))_{(i, j) \in V(TG_{n \times n})}$  to the one  $(\omega_\alpha(i - s, j))_{(i, j) \in V(TG_{n \times n})}$ , i.e., when the spy goes from  $(0, 0)$  to  $(s, 0)$ .

Equivalently, the flow constraints can be defined as a flow problem in a *transportation bipartite auxiliary network*  $H$  defined as follows (i.e., the constraints are satisfied if and only if there is feasible flow in  $H$ ). Let  $H = (V_1 \cup V_2, E(H))$  be the graph such that  $V_1$  and  $V_2$  are two copies of  $V(TG_{n \times n})$ . There is an arc from  $u \in V_1$  to  $v \in V_2$  if  $\{u, v\} \in E(TG_{n \times n})$ . Each vertex  $(i, j) \in V_1$  has a supply  $\omega_\alpha(i, j)$  and every vertex  $(i', j') \in V_2$  has a demand  $\omega_\alpha(i - s, j)$ . By Hall's Theorem [6], there is a feasible flow in  $H$  if and only if, for every  $A \subseteq V_1$ , the total supply in  $N[A]$  is at least the demand in  $A \subseteq V_2$ , i.e., at least  $\sum_{(i, j) \in A} \omega_\alpha(i - s, j)$ .

To summarize, the flow constraints can be stated as:

$$\forall A \subseteq V(TG_{n \times n}), \quad \sum_{(i, j) \in N[A]} \omega_\alpha(i, j) \geq \sum_{(i, j) \in A} \omega_\alpha(i - s, j). \quad (6)$$

We aim at deciding the range of  $\alpha$  such that the function  $\omega_\alpha$  satisfies constraint 6. For this purpose, we first aim at finding a set  $\mathcal{H}_s \subseteq V(TG_{n \times n})$  such that  $\kappa_\alpha(\mathcal{H}_s) = \sum_{(i, j) \in N[\mathcal{H}_s]} \omega_\alpha(i, j) - \sum_{(i, j) \in \mathcal{H}_s} \omega_\alpha(i - s, j)$  is minimum. For such a set  $\mathcal{H}_s$ , if  $\kappa_\alpha(\mathcal{H}_s) \geq 0$ , it implies that  $\omega_\alpha$  satisfies constraint 6.

Let  $\mathcal{H}_s$  be the set of vertices  $(i, j) \in V(TG_{n \times n})$  defined by:

$$\mathcal{H}_s = \{(i, j) \mid s/2 \leq i \leq (n + s)/2 \bmod n, 0 \leq j < n\}.$$

The proof of the following technical lemma is postponed to Appendix 6.3.3.

► **Lemma 8.** *Let  $\alpha > 0$  and  $s \leq n/2$ . For every  $A \subseteq V(TG_{n \times n})$ ,  $\kappa_\alpha(A) \geq \kappa_\alpha(\mathcal{H}_s)$ .*

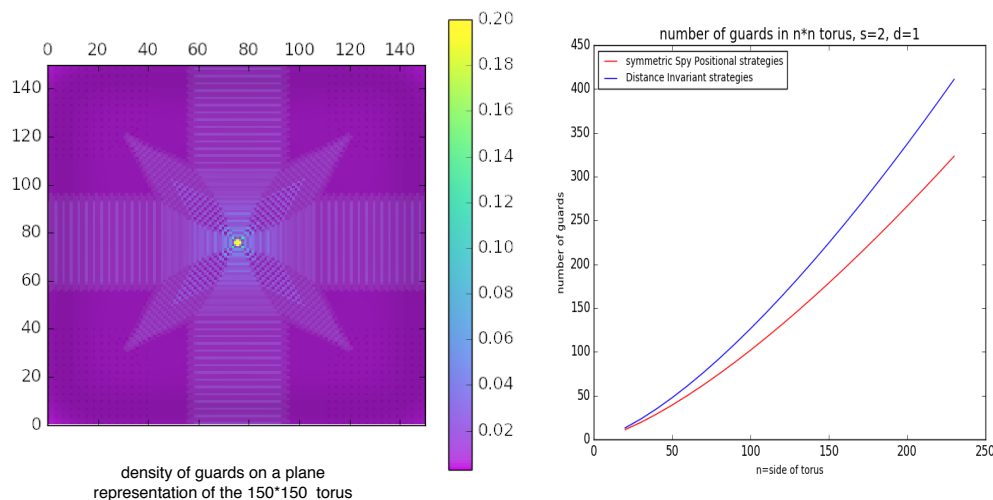
Finally, we are ready to present a winning strategy in the  $n \times n$  torus which proves Th. 6.

► **Lemma 9.** *Let  $n, s \geq 2$ ,  $s \leq n/2$ ,  $d \geq 0$  and  $0 < \alpha \leq \log(3/2)$ . There exists a constant  $B > 0$  (independent of  $n$ ) such that the function  $\omega_\alpha : V(TG_{n \times n}) \rightarrow \mathbb{R}^+$  where  $\omega_\alpha(v) = \frac{B}{(d(v) + 1)^\alpha}$  for every  $v \in V(TG_{n \times n})$  is a spy-positional winning fractional strategy that uses  $O(n^{2-\alpha})$  guards to control a spy with speed  $s$  at distance  $d$  in  $TG_{n \times n}$ .*

*Sketch of Proof.* See Appendix 6.3.3 for full proof. To verify that  $\omega_\alpha$  is a winning strategy, we need to prove that it satisfies constraints 3 and 6. Let  $B_d$  be the set of vertices at distance at most  $d$  from  $(0, 0)$  and let  $B = 1 / \sum_{v \in B_d} \frac{1}{(d(v) + 1)^\alpha}$ . Constraint 3 is satisfied by the choice of  $B$ . Some computations allow us to show that  $\kappa_\alpha(\mathcal{H}_s) \geq 0$  if  $0 < \alpha \leq \log(3/2)$  and therefore, by Lemma 8, Constraint 6 is satisfied. Finally, a simple summation shows that the strategy uses  $\sum_{v \in V(TG_{n \times n})} \omega_\alpha(v) = O(n^{2-\alpha})$  guards.  $\diamond$

## Conclusion

Concerning the Spy-game, the main open question is to determine the exact value of  $gn_{s,d}(G_{n \times n})$  in any  $n \times n$  grid  $G_{n \times n}$  (or torus). A first step towards such a result would be to prove that  $gn_{s,d}(G_{n \times n}) = O(gn_{s',d'}(G_{n \times n}))$  for any  $s, s' \geq 2$  and  $d, d' \geq 0$ . To get more intuition on optimal strategies for guards, we used Cplex to solve the LP described



■ **Figure 1** Experimental results,  $s = 2$  and  $d = 1$ . (Left) Density of guards on a plane representation of the  $150 \times 150$  torus in an optimal symmetrical Spy-positional configuration. (Right) Minimum number of guards for symmetrical (red) and distance-invariant (blue) Spy-Positional strategies.

in Section 3 with additional constraints of symmetry. The left drawing in Fig. 1 represents the density of guards in the torus of side 150 (where the central vertex is the position of the spy) for  $s = 2$  and  $d = 1$ . It shows that optimal symmetric Spy-positional (SSP) strategies may be much more intricate than the strategy  $\omega_\alpha$  we studied. For instance, it is not monotone when the distance to the spy's position increases. On the right, we plotted the number of guards used by optimal SSP (in red) which is much less than  $n^{2-\log(3/2)}$  for  $n \leq 250$  (it is difficult to extrapolate further intuition from such small values of  $n$ )<sup>3</sup>. Even the optimal distance-invariant strategies (i.e., the density of guards is only a function of the distance to the spy's position) computed using the LP (plotted in blue) use much less guards than  $n^{2-\log(3/2)}$  (we did not plot the function  $n^{2-\log(3/2)}$  for more readability, indeed,  $50^{2-\log(3/2)} > 500$  and  $250^{2-\log(3/2)} > 6600$ ). In trees, it would be interesting to design a combinatorial algorithm (i.e., not relying on the solution of a Linear Program) that computes optimal strategies for controlling a spy with speed  $s$  at distance  $d$ .

More importantly, using the fractional framework to obtain new results in two-player combinatorial games in graphs seems promising.

---

## References

- 1 M. Aigner and M. Fromme. A game of cops and robbers. *Discrete Applied Mathematics*, 8:1–12, 1984.
- 2 P. Balister, S. Binski, B. Bollobás, and B. P. Narayanan. Catching a fast robber on the grid. *CoRR*, abs/1609.01002, 2016. URL: <http://arxiv.org/abs/1609.01002>.

---

<sup>3</sup> Solving the LP for  $n \geq 150$  takes more than one hour on a basic laptop

- 3 I. Beaton, S. Finbow, and J.A. MacDonald. Eternal domination numbers of  $4 \times n$  grid graphs. *J. Comb. Math. Comb. Comput.*, 85:33–48, 2013.
- 4 A. Bonato, E. Chiniforooshan, and P. Pralat. Cops and robbers from a distance. *Theor. Comput. Sci.*, 411(43):3834–3844, 2010.
- 5 A. Bonato and R. J. Nowakowski. *The game of Cops and Robbers on Graphs*. American Math. Soc., 2011.
- 6 J. A. Bondy and U. S. R. Murty. *Graph theory*, volume 244 of *Graduate Texts in Mathematics*. Springer, 2008.
- 7 A. Burger, E. J. Cockayne, W. R. Gründlingh, C. M. Mynhardt, J. H. van Vuuren, and W. Winterbach. Infinite order domination in graphs. *J. Comb. Math. Comb. Comput.*, 50:179–194, 2004.
- 8 N. Cohen, M. Hilaire, N. A. Martins, N. Nisse, and S. Pérennes. Spy-game on graphs. In *8th International Conference on Fun with Algorithms, FUN 2016*, pages 10:1–10:16, 2016.
- 9 N. Cohen, N. A. Martins, F. Mc Inerney, N. Nisse, S. Pérennes, and R. Sampaio. Spy-game on graphs. Technical report, INRIA, 2017. RR, <http://hal.inria.fr/hal-161108204>.
- 10 A. Z. Delaney and M. E. Messinger. Closing the gap: Eternal domination on  $3 \times n$  grids. *to appear in Contributions to Discrete Mathematics*, 2015.
- 11 F. V. Fomin, F. Giroire, A. Jean-Marie, D. Mazauric, and N. Nisse. To satisfy impatient web surfers is hard. In *6th Int. Conf. on Fun with Algorithms (FUN)*, volume 7288 of *LNCS*, pages 166–176, 2012.
- 12 F. V. Fomin, P. A. Golovach, J. Kratochvíl, N. Nisse, and K. Suchan. Pursuing a fast robber on a graph. *Theor. Comput. Sci.*, 411(7-9):1167–1181, 2010.
- 13 F. Giroire, D. Mazauric, N. Nisse, S. Pérennes, and R. P. Soares. Connected surveillance game. In *20th International Colloquium on Structural Information and Communication Complexity (SIROCCO)*, Lecture Notes in Computer Science. Springer, 2013.
- 14 F. Giroire, N. Nisse, S. Pérennes, and R. P. Soares. Fractional combinatorial games. Technical report, INRIA, 2013. RR8371, <http://hal.inria.fr/hal-00865345>.
- 15 W. Goddard, S. M. Hedetniemi, and S. T. Hedetniemi. Eternal security in graphs. *J. Comb. Math. Comb. Comput.*, 52:160–180, 2005.
- 16 D. Gonçalves, A. Pinlou, M. Rao, and S. Thomassé. The domination number of grids. *SIAM J. Discrete Math.*, 25(3):1443–1453, 2011.
- 17 G. Joret, M. Kaminski, and D. O. Theis. The cops and robber game on graphs with forbidden (induced) subgraphs. *Contributions to Discrete Mathematics*, 5(2), 2010.
- 18 W. B. Kinnersley. Cops and robbers is exptime-complete. *JCTB*, 111:201–220, 2015.
- 19 W. F. Klostermeyer and G. MacGillivray. Eternal dominating sets in graphs. *J. Comb. Math. Comb. Comput.*, 68, 2009.
- 20 A. Kosowski, B. Li, N. Nisse, and K. Suchan.  $k$ -chordal graphs: From cops and robber to compact routing via treewidth. *Algorithmica*, 72(3):758–777, 2015.
- 21 I. Lamprou, R. Martin, and S. Schewe. Perpetually dominating large grids. *CoRR*, abs/1611.08204, 2016. URL: <http://arxiv.org/abs/1611.08204>.
- 22 R. J. Nowakowski and P. Winkler. Vertex-to-vertex pursuit in a graph. *Discrete Maths*, 43:235–239, 1983.
- 23 A. Quilliot. *Problèmes de jeux, de point fixe, de connectivité et de représentation sur des graphes, des ensembles ordonnés et des hypergraphes*. Doctorat d'état, Univ. Paris 4, 1983.
- 24 B. S. W. Schröder. The copnumber of a graph is bounded by  $\lfloor \frac{3}{2} \text{genus}(g) \rfloor + 3$ . *Categorical perspectives (Kent, OH, 1998)*, *Trends in Mathematics*, pages 243–263, 2001.
- 25 A. Scott and B. Sudakov. A bound for the cops and robbers problem. *SIAM J. Discrete Math.*, 25(3):1438–1442, 2011.
- 26 C. M. van Bommel and M. F. van Bommel. Eternal domination numbers of  $5 \times n$  grid graphs. *J. Comb. Math. Comb. Comput.*, 97:83–102, 2016.

## 6 Appendix

### 6.1 Simple remarks on Spy-positional strategies and strategies in trees

► **Claim 8.** *There are graphs for which no optimal guards' strategy is Spy-positional.*

**Proof.** Let us present an example for  $s = 2$  and  $d = 1$  in the cycle  $C_5$  with 5 vertices. First, it is easy to show that  $gn_{2,1}(C_5) = 1$ . Indeed, a unique guard can initially be placed at a neighbor of the initial position of the spy. Then, whatever be the move of the spy with speed 2, the guard can move to a neighbor of the spy's new position. Hence, one guard can control a spy with speed 2 at distance 1 in  $C_5$ .

However, every spy-positional strategy needs 2 guards. Indeed, it is easy to see that we may assume that, initially, the guard occupies a neighbor of the spy. Initially, the guard has to be either at a neighbor of the spy or at the same vertex. In the latter case, the spy can move at distance two from the guard, and the guard has no choice but to reach a neighbor of the spy. Let  $\{a, b, c, d, e\}$  be the vertex-set of the cycle, in order. W.l.o.g., the spy starts at  $b$  while the guard is at  $a$ . Then, the spy goes to  $c$  and the guard has to go to  $b$ . The spy goes to  $d$  and the guard has to go to  $c$ . Finally, the spy goes back to  $b$  and the guard either stays at  $c$  or goes to  $b$ . Hence, there are two steps where the spy occupies  $b$  and the guard occupies different vertices, i.e., the strategy is not Spy-positional. ◀

► **Claim 9.** *There exists a family of trees with unbounded guard-number such that, for each of these trees, there is a strategy of the spy that forces all guards to occupy all non-leaf vertices infinitely often, whatever be the optimal strategy followed by the guards.*

**Proof.** We present an example in the case  $s = 2$  and  $d = 1$  but it can be generalized to any  $s \geq 2$  and  $d > 0$  (by increasing the branches of the star  $S$  defined below).

Let  $S$  be the tree obtained from a star with three leaves by subdividing each edge exactly twice (i.e.,  $S$  has 10 vertices). Let  $(S_i)_{i \leq k}$  be  $k$  disjoint copies of  $S$  and let  $c_i$  be the unique vertex of degree 3 of  $S_i$ . Finally, let  $T$  be the tree obtained by adding one vertex  $c$  and making it adjacent to every  $c_i$ ,  $i \leq k$ . Note that  $|V(T)| = 10k + 1 = n$ .

First, let us show that  $gn_{2,1}(T) = k + 1 = \Theta(n)$  and that, when the spy is in  $c$ , the guards have to occupy the vertices  $c, c_1, \dots, c_k$ . We label the vertices as follows where  $1 \leq j \leq 3k$ : let  $v_{3j}$  be a leaf, let  $v_{3j-1}$  be the vertex adjacent to the leaf  $v_{3j}$ , and let  $v_{3j-2}$  be the vertex adjacent to  $v_{3j-1}$  and  $c_i$  for  $i = \lceil \frac{j}{3} \rceil$ .

- A strategy using  $k + 1$  guards proceeds as follows. At any step, vertices  $c_1, \dots, c_k$  are occupied (not necessarily by the same guards). Then, if the spy occupies  $c$  or one of the  $c_i$ 's, one guard occupies  $c$ . If the spy occupies  $v_{3j-1}$  or  $v_{3j-2}$  for some  $j \leq 3k$ , then  $v_{3j-2}$  must be occupied by a guard. Finally, if the spy occupies  $v_{3j}$ , then  $v_{3j-1}$  must be occupied by a guard. It is easy to see that, whatever be the strategy of the spy, the guards may move (at most 2 guards per step) as to ensure the desired positions.
- Now we prove the lower bound. The spy starts at a leaf  $v_{3j}$  for some  $1 \leq j \leq 3k$  and moves at full speed to another leaf  $v_{3l}$  for some  $1 \leq l \leq 3k$  such that  $l \neq j$ . Then, there must be a guard at  $v_{3j-1}$  initially and when the spy reaches  $v_{3j-2}$ , there must be at least one other guard at a vertex in the same subtree  $S_i$  for  $i = \lceil \frac{j}{3} \rceil$  as otherwise, the spy could move to one of  $\{v_{3(j+1)}, v_{3(j+2)}\}$  and win. Since there are two guards in the same subtree as the spy, then the spy moves to  $c$ . Neither of the two previous guards can reach any of the other  $k - 1$   $c_i$  on this turn. There must be at least one guard in each of the other  $k - 1$  subtrees  $S_i$  as otherwise, the spy moves to a leaf in one of these subtrees and wins since it would take him two turns but a guard at  $c$  could only be at distance at least 2



## XX:14 Study of a combinatorial game in graphs through Linear Programming

from the spy after two turns. It also follows that, when the spy is at  $c$ , the guards must occupy the vertices  $c, c_1, \dots, c_k$ .

Now, we can prove the main statement of the lemma. For any  $i \leq k$  and any vertex  $v \in V(T)$  of degree two, there is a strategy of the spy that brings the guard initially at  $c_i$  to  $v$  and thus, to any non-leaf vertex. For this purpose, let  $j \neq i$  be such that  $v \in V(S_j)$ . The spy first goes (at full speed) to a leaf of  $S_i$ , then to another leaf of  $S_i$ , then it goes to a leaf of  $S_j$  that is not the neighbor of  $v$  and finally the spy goes to the leaf neighbor of  $v$ . It can be verified that the guard that was initially at  $c_i$  must occupy  $v$ . Repeating infinitely this strategy (for any  $v$  and  $i$ ) gives the strategy announced in the statement. ◀

## 6.2 Spy-game in trees (Section 4)

### 6.2.1 Proof of Theorem 2

In what follows, we consider an  $n$ -node tree  $T = (V, E)$  rooted in some arbitrary node  $r$ . Recall that, for every  $u \in V$ ,  $T_u$  denotes the subtree of  $T$  rooted in  $u$ . Given a configuration  $\omega : V \rightarrow \mathbb{R}^+$ , let  $\omega(T_u) = \sum_{v \in V(T_u)} \omega(v)$ . For any  $x \in \mathbb{R}^+$ , let  $\{x\} = x - \lfloor x \rfloor$ , i.e.,  $\{x\}$  is the fractional part of  $x$ .

Recall that the *rounded* configuration  $\omega^r : V \mapsto \mathbb{N}$  is defined as, for every  $u \in V$ ,

$$\omega^r(u) = \left\lfloor \omega(u) + \sum_{v \text{ child of } u} (\omega(T_v) - \lfloor \omega(T_v) \rfloor) \right\rfloor = \left\lfloor \omega(u) + \sum_{v \text{ child of } u} \{\omega(T_v)\} \right\rfloor$$

► **Claim 1.** For any configuration  $\omega : V(T) \rightarrow \mathbb{R}^+$ ,  $\omega^r(T) = \lfloor \omega(T) \rfloor$

**Proof.** The proof is by induction on  $|V|$ . It clearly holds if  $|V| = 1$ . Let  $T_1, \dots, T_h$  be the subtrees of  $T \setminus r$  rooted in the children of  $r$ . By definition,

$$\begin{aligned} \omega^r(T) &= \omega^r(r) + \sum_{1 \leq i \leq h} \omega^r(T_i) \\ &= \lfloor \omega(r) + \sum_{1 \leq i \leq h} \{\omega(T_i)\} \rfloor + \sum_{1 \leq i \leq h} \omega^r(T_i) \end{aligned}$$

By induction,  $\omega^r(T_i) = \lfloor \omega(T_i) \rfloor$  for every  $1 \leq i \leq h$ , and so:

$$\begin{aligned} \omega^r(T) &= \lfloor \omega(r) + \sum_{1 \leq i \leq h} \{\omega(T_i)\} \rfloor + \sum_{1 \leq i \leq h} \lfloor \omega(T_i) \rfloor \\ &= \lfloor \omega(r) + \sum_{1 \leq i \leq h} \omega(T_i) \rfloor \\ &= \lfloor \omega(T) \rfloor \end{aligned}$$

◀

Recall that  $\text{cont}(T, \omega) = \{u \in V : \sum_{v \in N_d[u]} \omega(v) \geq 1\}$  (i.e.,  $\text{cont}(T, \omega)$  is the set of vertices  $u$  such that the spy on  $u$  is controlled at distance  $d$  by the guards in the configuration  $\omega$ ).

► **Claim 2.** For any configuration  $\omega : V(T) \rightarrow \mathbb{R}^+$ ,  $\text{cont}(T, \omega) \subseteq \text{cont}(T, \omega^r)$

**Proof.** Let  $u \in \text{cont}(T, \omega)$ . By definition,  $\sum_{v \in N_d[u]} \omega(v) \geq 1$ . Let  $r'$  be the vertex in  $N_d[u]$  that is closest to the root  $r$ , and let  $T'$  be the subtree of  $T$  rooted in  $r'$ . Finally, let  $T'_1, \dots, T'_h$

be the subtrees of  $T' \setminus N_d[u]$ . By Claim 1,  $\omega^r(T') = \lfloor \omega(T') \rfloor$  and  $\omega^r(T'_i) = \lfloor \omega(T'_i) \rfloor$  for any  $1 \leq i \leq h$ . Hence,

$$\omega^r(T') = \sum_{v \in N_d[u]} \omega^r(v) + \sum_{1 \leq i \leq h} \omega^r(T'_i) = \sum_{v \in N_d[u]} \omega^r(v) + \sum_{1 \leq i \leq h} \lfloor \omega(T'_i) \rfloor$$

and,

$$\omega^r(T') = \lfloor \omega(T') \rfloor = \left\lfloor \sum_{v \in N_d[u]} \omega(v) + \sum_{1 \leq i \leq h} \omega(T'_i) \right\rfloor$$

Since  $\sum_{v \in N_d[u]} \omega(v) \geq 1$ , it follows that

$$\left\lfloor \sum_{v \in N_d[u]} \omega(v) + \sum_{1 \leq i \leq h} \omega(T'_i) \right\rfloor \geq 1 + \left\lfloor \sum_{1 \leq i \leq h} \omega(T'_i) \right\rfloor \geq 1 + \sum_{1 \leq i \leq h} \lfloor \omega(T'_i) \rfloor$$

Altogether,  $1 + \sum_{1 \leq i \leq h} \lfloor \omega(T'_i) \rfloor \leq \omega^r(T') = \sum_{v \in N_d[u]} \omega^r(v) + \sum_{1 \leq i \leq h} \lfloor \omega(T'_i) \rfloor$ . Therefore,  $\sum_{v \in N_d[u]} \omega^r(v) \geq 1$  and  $u \in \text{cont}(T, \omega^r)$ .  $\blacktriangleleft$

► **Claim 3.** Let  $\omega_1, \omega_2 : V(T) \mapsto \mathbb{R}^+$  be two configurations such that the guards can go from  $\omega_1$  to  $\omega_2$  in one step (there is feasible flow from  $\omega_1$  to  $\omega_2$ ). Then, the guards can go from  $\omega_1^r$  to  $\omega_2^r$  in one step (there is feasible integral flow from  $\omega_1^r$  to  $\omega_2^r$ ).

**Proof.** The proof is by induction on  $k = \omega_1^r(T)$  (Note that  $k = \omega_2^r(T)$ ), the result being trivial when  $k = 0$ . Let  $f$  be the flow representing the move of the guards from  $\omega_1$  to  $\omega_2$ . Clearly, we may assume that,  $\forall u, v \in V$ , at most one of  $f(u, v)$  and  $f(v, u)$  is non-null.

Among all vertices  $v \in V$  such that  $\omega_1(T_v) \geq 1$  or  $\omega_2(T_v) \geq 1$ , let  $x$  be a lowest one (such a vertex furthest from the root). By symmetry (there is a feasible flow from  $\omega_1$  to  $\omega_2$  if and only if there is a feasible flow from  $\omega_2$  to  $\omega_1$ ), up to exchanging  $\omega_1$  and  $\omega_2$ , we may assume that  $\omega_1(T_x) \geq 1$ . Note that, by minimality of  $x$ , for every descendant  $u \in V(T_x) \setminus \{x\}$  of  $x$ ,  $\omega_1(T_u) < 1$  and  $\omega_2(T_u) < 1$ .

Now, let  $\gamma_1$  be the function defined by  $\gamma_1(x) = \omega_1(T_x) - 1$ ,  $\gamma_1(u) = 0$  for every descendant  $u$  of  $x$ , and  $\gamma_1(v) = \omega_1(v)$  for every  $v \in V \setminus V(T_x)$ . Note that  $\gamma_1^r(v) = \omega_1^r(v)$  for every  $v \in V \setminus \{x\}$  and  $\gamma_1^r(x) = \omega_1^r(x) - 1$ . Now, there are two cases to be considered.

■ First, assume that  $\omega_2(T_x) \geq 1$ . In this case, let  $\gamma_2$  be the function defined by  $\gamma_2(x) = \omega_2(T_x) - 1$ ,  $\gamma_2(u) = 0$  for every descendant  $u$  of  $x$ , and  $\gamma_2(v) = \omega_2(v)$  for every  $v \in V \setminus V(T_x)$ . Note that there is a feasible flow  $f'$  from  $\gamma_1$  to  $\gamma_2$ : for any  $u, v \in V(T_x)$ ,  $f'(u, v) = 0$  and for any  $u \in V$ ,  $v \in V \setminus V(T_x)$ ,  $f'(u, v) = f(u, v)$ . Note also that  $\gamma_2^r(v) = \omega_2^r(v)$  for every  $v \in V \setminus \{x\}$  and  $\gamma_2^r(x) = \omega_2^r(x) - 1$ .

By induction (since  $\gamma_1^r(T) = \gamma_2^r(T) = \omega_1^r(T) - 1$ ), there is a feasible integral flow  $f^*$  from  $\gamma_1^r$  to  $\gamma_2^r$ . Since  $\omega_1^r$  (resp.,  $\omega_2^r$ ) is obtained from  $\gamma_1^r$  (resp.,  $\gamma_2^r$ ) by adding 1 guard in  $x$ , this flow  $f^*$  is also a feasible integral flow from  $\omega_1^r$  to  $\omega_2^r$ .

■ Second,  $\omega_2(T_x) < 1$ . Let  $p$  be the parent of  $x$  ( $x$  cannot be the root since  $\omega_2^r(T) \geq 1$ ). Note that, because there is flow from  $\omega_1$  to  $\omega_2$ , then  $\omega_2(p) + \omega_2(T_x) \geq \omega_1(T_x) \geq 1$ .

In this case, let  $\gamma_2$  be the function defined by  $\gamma_2(u) = 0$  for every  $u \in V(T_x)$ ,  $\gamma_2(v) = \omega_2(v)$  for every  $v \in V \setminus (V(T_x) \cup \{p\})$  and  $\gamma_2(p) = \omega_2(p) + \omega_2(T_x) - 1 \geq 0$ .

Note that there is a feasible flow  $f'$  from  $\gamma_1$  to  $\gamma_2$ : for any  $u, v \in V(T_x)$ ,  $f'(u, v) = 0$ , for any  $u, v \in V \setminus V(T_x)$ ,  $f'(u, v) = f(u, v)$ , and  $f'(x, p) = \gamma_1(x)$ . Note also that  $\gamma_2^r(v) = \omega_2^r(v)$  for every  $v \in V \setminus \{p\}$  and  $\gamma_2^r(p) = \omega_2^r(p) - 1$ .

By induction (since  $\gamma_1^r(T) = \gamma_2^r(T) = \omega_1^r(T) - 1$ ), there is a feasible integral flow  $f^*$  from  $\gamma_1^r$  to  $\gamma_2^r$ . Since  $\omega_1^r$  (resp.,  $\omega_2^r$ ) is obtained from  $\gamma_1^r$  (resp.,  $\gamma_2^r$ ) by adding 1 guard in  $x$  (resp., in  $p$ ), there is a feasible integral flow from  $\omega_1^r$  to  $\omega_2^r$  that can be obtained from  $f^*$  by adding to it one unit of flow from  $x$  to  $p$ . ◀

### 6.2.2 Proof of Theorem 3

Let  $T = (V, E)$  be a tree be rooted in some node  $r$ . In what follows, for every  $u \in V$ , let  $Children(u)$  be the set of children of  $u$  in  $T$ .

► **Claim 6.** For every  $v \in V$ ,  $\sum_{u \in N_d[v]} \omega_v^{min}(u) \geq 1$ .

**Proof.** Let  $v^*$  be the vertex of  $N_d[v]$  that is closest to the root  $r$ . Let  $v_1, \dots, v_p$  be the descendants of  $v^*$  that are at distance exactly  $d + 1$  from  $v$ . Since  $\alpha_v$  is the cumulative function of  $\omega_v^{min}$ , we have that  $\sum_{u \in N_d[v]} \omega_v^{min}(u) = \alpha_v(v^*) - \sum_{1 \leq j \leq p} \alpha_v(v_j)$ . Let  $1 \leq i \leq h$  be an integer such that  $\alpha_v(v^*) = \min_{1 \leq j \leq h} \omega_j^+(v^*) = \omega_i^+(v^*)$ . Since the guards in configuration  $\omega_i$  control the spy in  $v$  at distance  $d$ , we have that  $\sum_{u \in N_d[v]} \omega_i(u) = \omega_i^+(v^*) - \sum_{1 \leq j \leq p} \omega_i^+(v_j) \geq 1$ . Hence,  $\sum_{u \in N_d[v]} \omega_v^{min}(u) = \alpha_v(v^*) - \sum_{1 \leq j \leq p} \alpha_v(v_j) = \omega_i^+(v^*) - \sum_{1 \leq j \leq p} \min_{1 \leq j' \leq h} \omega_{j'}^+(v_j) \geq \omega_i^+(v^*) - \sum_{1 \leq j \leq p} \omega_i^+(v_j) \geq 1$ . ◀

► **Claim 7.** For every  $v \in V$  and  $v' \in N_s[v]$ , there is a feasible flow from  $\omega_v^{min}$  to  $\omega_{v'}^{min}$ .

**Proof.** Let  $\mathcal{C}_v = \{\omega_1, \dots, \omega_h\} \in \sigma$  (the configurations of  $\sigma$  when the spy is at  $v$ ) and  $\mathcal{C}_{v'} = \{\omega'_1, \dots, \omega'_{h'}\} \in \sigma$  (the configurations of  $\sigma$  when the spy is at  $v'$ ). Since  $\sigma$  is a winning strategy, it means that, for every  $1 \leq i \leq h$ , there is  $1 \leq \delta(i) \leq h'$ , such that there is a feasible flow from  $\omega_i \in \mathcal{C}_v$  to  $\omega'_{\delta(i)} \in \mathcal{C}_{v'}$ . That is, there is a function  $f^i : V \times V \rightarrow \mathbb{R}^+$  such that, for every  $u \in V$ ,  $\omega'_{\delta(i)}(u) = \omega_i(u) + \sum_{y \in N(u)} (f^i(y, u) - f^i(u, y))$  and  $\sum_{y \in N(u)} f^i(u, y) \leq \omega_i(u)$ . Note that, such a function  $f^i$  can be defined as, for every  $u \in V$  and  $p \in V$ , the parent of  $u$  in  $T$  rooted in  $r$  (if  $u \neq r$ ),  $f^i(u, p) = \max\{\omega_i(u) - \omega'_{\delta(i)}(u), 0\}$  and  $f^i(p, u) = \max\{\omega'_{\delta(i)}(u) - \omega_i(u), 0\}$ .

Let  $u \in V$ ,  $X \subseteq Children(u)$  be any subset of the children of  $u$ , and  $1 \leq i \leq h$ . Because of the existence of the flow  $f^i$ ,  $\sum_{y \in X} (\omega'_{\delta(i)})^+(y) \leq \omega_i(u) + \sum_{y \in X} \omega_i^+(y)$ , hence:

$$\omega_i^+(u) = \omega_i(u) + \sum_{y \in X} \omega_i^+(y) + \sum_{y \in Children(u) \setminus X} \omega_i^+(y) \geq \sum_{y \in X} (\omega'_{\delta(i)})^+(y) + \sum_{y \in Children(u) \setminus X} \omega_i^+(y)$$

and so, since for every  $y \in V$ ,  $\alpha_{v'}(y) = \min_{1 \leq j \leq h'} \omega'_j(y)$  and  $\alpha_v(y) = \min_{1 \leq j \leq h} \omega_j(y)$ :

$$\omega_i^+(u) \geq \sum_{y \in X} \alpha_{v'}(y) + \sum_{y \in Children(u) \setminus X} \alpha_v(y)$$

The above inequality holds for every  $1 \leq i \leq h$ . Since  $\alpha_v(u) = \min_{1 \leq i \leq h} \omega_i^+(u)$ , it follows that:

$$\alpha_v(u) \geq \sum_{y \in X} \alpha_{v'}(y) + \sum_{y \in Children(u) \setminus X} \alpha_v(y)$$

By similar arguments (because, by symmetry, there is a flow from  $\omega'_j$  to some  $\omega_j$  for every  $1 \leq j \leq h'$ ), we get

$$\alpha_{v'}(u) \geq \sum_{y \in X} \alpha_{v'}(y) + \sum_{y \in \text{Children}(u) \setminus X} \alpha_v(y)$$

We need to prove that there exists a function  $f : V \times V \rightarrow \mathbb{R}^+$  such that, for every  $u \in V$ ,  $\omega_{v'}^{\min}(u) = \omega_v^{\min}(u) + \sum_{y \in N(u)} (f(y, u) - f(u, y))$  and  $\sum_{y \in N(u)} f(u, y) \leq \omega_v^{\min}(u)$ .

For every  $u \in V$ , let  $p \in V$  be the parent of  $u$  in  $T$  rooted in  $r$  (if  $u \neq r$ ). Let  $f^{\min}(u, p) = \max\{\alpha_v(u) - \alpha_{v'}(u), 0\}$  and let  $f^{\min}(p, u) = \max\{\alpha_{v'}(u) - \alpha_v(u), 0\}$ .

It is clear that, for every  $u \in V$ ,  $\omega_{v'}^{\min}(u) = \omega_v^{\min}(u) + \sum_{y \in N(u)} (f^{\min}(y, u) - f^{\min}(u, y))$ . Hence, we only need to prove that  $\sum_{y \in N(u)} f^{\min}(u, y) \leq \omega_v^{\min}(u)$ .

Let  $u \in V$ ,  $p$  its parent (if  $u \neq r$ ) and let  $X \subseteq \text{Children}(u)$  be the set of vertices such that, for every  $y \in X$ ,  $f^{\min}(u, y) = \alpha_{v'}(y) - \alpha_v(y) > 0$ . There are two cases to be considered.

■ First, let us assume that  $f^{\min}(u, p) = 0$ .

$$\begin{aligned} \omega_v^{\min}(u) &= \alpha_v(u) - \sum_{y \in \text{Children}(u)} \alpha_v(y) \\ &= (\alpha_v(u) - \sum_{y \in \text{Children}(u) \setminus X} \alpha_v(y)) - \sum_{y \in X} \alpha_v(y) \\ &\geq \sum_{y \in X} (\alpha_{v'}(y) - \alpha_v(y)) = \sum_{y \in N(u)} f^{\min}(u, y) \end{aligned}$$

■ Second, assume that  $f^{\min}(u, p) = \delta > 0$ .

$$\begin{aligned} \omega_v^{\min}(u) &= \alpha_v(u) - \sum_{y \in \text{Children}(u)} \alpha_v(y) \\ &= \alpha_{v'}(u) + \delta - \sum_{y \in \text{Children}(u)} \alpha_v(y) \\ &= \delta + (\alpha_{v'}(u) - \sum_{y \in \text{Children}(u) \setminus X} \alpha_v(y)) - \sum_{y \in X} \alpha_v(y) \\ &\geq \delta + \sum_{y \in X} (\alpha_{v'}(y) - \alpha_v(y)) = \sum_{y \in N(u)} f^{\min}(u, y) \end{aligned}$$

◀

## 6.3 Case of grids (Section 5)

### 6.3.1 Grid *versus* Torus

► **Lemma 5.** For every  $n, m \geq 2$ ,  $s \geq 2$ ,  $d \geq 0$ , and for every  $f \in \{gn_{s,d}, fgn_{s,d}, fgn_{s,d}^*\}$ :

$$f(TG_{n \times m})/4 \leq f(G_{n \times m}) \leq 4 \cdot f(TG_{n \times m}).$$

**Proof.** Let us present the proof in the integral case, i.e., when  $f = gn_{s,d}$ , the other two cases are similar.

Let  $\sigma$  be a winning strategy using  $k$  guards in  $TG_{n \times m}$ . We define a winning strategy using  $4k$  guards in  $G_{n \times m}$ . For this purpose, let us label the guards used by  $\sigma$  as  $G_1, \dots, G_k$ . In  $G_{n \times m}$ , the behavior of Guard  $G_i$  ( $1 \leq i \leq k$ ) is “simulated” by four guards as follows. The guard  $G_i$  being at  $(x, y) \in V(TG_{n \times m})$  is simulated by one guard at each of the

four vertices:  $(x, y)$ ,  $(n - 1 - x, y)$ ,  $(x, m - 1 - y)$  and  $(n - 1 - x, m - 1 - y)$ . Hence,  $gn_{s,d}(G_{n \times m}) \leq 4 \cdot gn_{s,d}(TG_{n \times m})$ .

Let  $\sigma$  be a winning strategy using  $k$  guards in  $G_{n \times m}$ . We define a winning strategy using  $4k$  guards in  $TG_{n \times m}$ . Our strategy actually allows to control four spies whose moves are correlated. Precisely, assume that when one spy occupies vertex  $(x, y)$ , the three other spies occupy respectively  $(n - 1 - x, y)$ ,  $(x, m - 1 - y)$  and  $(n - 1 - x, m - 1 - y)$ . We divide the  $4k$  guards into four teams, each of which uses the strategy  $\sigma$  (i.e., they all act as if they were in the grid) to control one of the four spies. When some spies cross an edge of  $E(TG_{n \times m}) \setminus E(G_{n \times m})$ , some teams will exchange their target. Hence,  $gn_{s,d}(TG_{n \times m}) \leq 4 \cdot gn_{s,d}(G_{n \times m})$ . ◀

### 6.3.2 Restricted moves for the spy

Let us weaken the spy by allowing it to move only “horizontally” or “vertically” at full speed. Precisely, when the spy occupies the vertex  $(i, j)$ , it is only allowed to move to one of the four vertices  $(i \pm s \bmod n, j)$  or  $(i, j \pm s \bmod n)$ . We say that such a spy has *restricted* moves. We prove (for the full proof see lemma 6 in the appendix) that it does not change the order of magnitude of the number of guards required to control it.

► **Lemma 6.** *Let  $n, s \geq 2$ , and  $d \geq 0$  be integers. Assume that there exists a (fractional or integral) winning strategy using  $k$  guards to control a spy, with speed  $s$  and restricted moves, at distance  $d$  in the  $n \times n$ -torus. Then, there exists a (fractional or integral) winning strategy using  $O(s^2k)$  guards to control a spy, with speed  $s$ , at distance  $d$  in the  $n \times n$ -torus.*

**Proof.** The proof is written in the integral case. The fractional case is similar.

For any strategy of a (non-restricted) spy, we will define a strategy for a restricted spy, called the *spy's shadow*, that ensures that the shadow is always at distance at most  $2s$  from the non-restricted spy. To control the non-restricted spy, the strategy consists of applying the strategy  $\sigma$  against its shadow (i.e., using  $k$  guards) and replacing each guard  $\gamma$  of  $\sigma$  by  $O(s^2)$  guards, one at every vertex at distance at most  $2s$  from the position of  $\gamma$ .

The shadow starts at the same vertex as the spy and “follows” it but only using restricted moves. The shadow can easily stay at distance  $< 2s$  from the spy if the spy moves from a vertex at distance  $< 2s$  from the shadow to a vertex at distance at least  $2s$  (but  $< 3s$  since the spy has speed  $s$ ) from the shadow. This means, then, that the shadow is at a position such that one of its coordinates differs by at least  $s$  from one of the spy's coordinates. So it can decrease its distance to the spy by exactly  $s$  using a restricted move. This means that after the shadow moves, the distance is still  $< 2s$ . ◀

### 6.3.3 Proof of Theorem 6

The proof of Theorem 6 follows from Lemma 9 that uses Lemma 8.

Let  $\mathcal{H}_s$  be the set of vertices  $(i, j) \in V(TG_{n \times n})$  defined by:

$$\mathcal{H}_s = \{(i, j) \mid s/2 \leq i \leq (n + s)/2 \bmod n, 0 \leq j < n\}.$$

► **Lemma 8.** *Let  $\alpha > 0$  and  $s \leq n/2$ . For every  $A \subseteq V(TG_{n \times n})$ ,  $\kappa_\alpha(A) \geq \kappa_\alpha(\mathcal{H}_s)$ .*

**Proof.** For simplicity of calculations, let us assume that both  $s$  and  $n$  are even. For any  $0 \leq i < n$ , the row  $L_i$  equals  $\{(i, j) \mid 0 \leq j < n\}$  and, for any  $0 \leq j < n$ , the column  $C_j$  equals  $\{(i, j) \mid 0 \leq i < n\}$ .

For any integer  $\ell$ , let  $f_\ell : V \rightarrow \mathbb{R}^+$  be the function such that, for any  $v_{i,j} \in V$ ,

$$f_\ell(i, j) = \frac{1}{(d((i, j), (\ell, 0)) + 1)^\alpha}$$

where  $d(x, y)$  denotes the distance between  $x$  and  $y$  in  $G$ .

Note that

► **Claim 8.** For any  $i, j$ ,  $f_s(i, j) = f_0(i - s \bmod n, j)$ .

For any  $A \subseteq V(TG_{n \times n})$ , let us define the *border*  $\delta(A)$  of  $A$  as  $\delta(A) = \{w \notin A \mid \exists v \in A, \{v, w\} \in E\}$ , i.e., the set of vertices not in  $A$  that have a neighbor in  $A$ .

Note that:

$$\kappa_\alpha(A) = \sum_{v \in N[A]} f_0(v) - \sum_{v \in A} f_s(v) = \sum_{v \in A} (f_0(v) - f_s(v)) + \sum_{v \in \delta(A)} f_0(v).$$

To find a vertex-set minimizing the above function, we actually define another function lower bounding the previous one. We identify a set  $A_{min}$  minimizing this second function such that both functions achieve the same value for  $A_{min}$ . Therefore,  $A_{min}$  also minimizes the first function.

The *vertical border*  $\mu(A)$  equals  $\{(i, j) \notin A \mid (i+1 \bmod n, j) \in A \text{ or } (i-1 \bmod n, j) \in A\}$ , i.e., the set of vertices not in  $A$  that have a neighbor in  $A$  and in the same column. Note that  $\mu(A) \subseteq \delta(A)$  for any  $A \subseteq V$ .

Let us set

$$\gamma(A) = \sum_{v \in A} (f_0(v) - f_s(v)) + \sum_{v \in \mu(A)} f_0(v).$$

Since  $f_0$  is positive and  $\mu(A) \subseteq \delta(A)$ ,

► **Claim 9.**  $\kappa_\alpha(A) \geq \gamma(A)$  for any  $A \subseteq V$ .

A useful property of  $\gamma$  is that columns are somehow “independent”.

► **Claim 10.**  $\gamma(A) = \sum_{0 \leq j < m} \gamma(A \cap C_j)$ .

Note that  $\mathcal{H}_s = \{v_{i,j} \mid s/2 \leq i \leq (n+s)/2 \bmod n, 0 \leq j < n\}$  is the set of vertices  $v$  such that  $f_0(v) - f_s(v) \leq 0$ . Moreover, note that  $\mu(\mathcal{H}_s) = \delta(\mathcal{H}_s)$  and so:

► **Claim 11.**  $\gamma(\mathcal{H}_s) = \kappa_\alpha(\mathcal{H}_s)$ .

Another useful property is that, by the first claim (and telescopic sum),

► **Claim 12.** For any  $0 \leq j < n$ ,

$$\gamma(\mathcal{H}_s \cap C_j) = \sum_{-s/2+1 \leq i \leq s/2+1} f_0(n/2 + i \bmod n, j) - \sum_{-s/2 \leq i \leq s/2-2} f_0(i, j).$$

*Proof of claim.*

$$\begin{aligned} \gamma(\mathcal{H}_s \cap C_j) &= \left[ \sum_{i=s/2}^{(n+s)/2 \bmod n} f_0(i, j) - f_s(i, j) \right] + f_0(s/2 - 1, j) + f_0((n+s)/2 + 1 \bmod n, j) \\ &= \left[ \sum_{i=s/2}^{(n+s)/2 \bmod n} f_0(i, j) - f_0(i-s, j) \right] + f_0(s/2 - 1, j) + f_0((n+s)/2 + 1 \bmod n, j) \\ &= \sum_{i=(n-s)/2+1}^{(n+s)/2+1 \bmod n} f_0(i, j) - \sum_{i=-s/2}^{s/2-2} f_0(i, j) \end{aligned}$$

◊



**XX:20 Study of a combinatorial game in graphs through Linear Programming**

The above proof actually extends to the following. Let

$$H(a, b) \cap C_j = \{(i, j) \mid a \pmod n \leq i \leq b \pmod n\}$$

► **Claim 13.** For any  $|a - b| > 1$ ,

$$\gamma(H(a, b) \cap C_j) = \sum_{-s/2+1 \leq i \leq s/2+1} f_0(b - s/2 + i, j) - \sum_{-s/2 \leq i \leq s/2-2} f_0(a - s/2 + i, j).$$

The remaining part of this section is devoted to prove that  $\mathcal{H}_s$  minimizes  $\kappa_\alpha$ . Precisely, let us prove that  $\gamma(\mathcal{H}_s) = \min_{A \subseteq V} \gamma(A)$ . This follows from the two following claims and previous claims.

► **Claim 14.** Let  $X$  be such that  $\gamma(X) = \min_{A \subseteq V} \gamma(A)$ . Then, for any  $0 \leq j < n$ ,  $X \cap C_j$  is connected.

*Proof of claim.* First, assume that there exists a vertex  $v \in C_j \setminus X$  such that its two neighbors in  $C_j$  are in  $X$ . Then,  $\gamma(X \cup \{v\}) = \gamma(X) - f_s(v) < \gamma(X)$ . Therefore, by minimality of  $\gamma(X)$ , there are no such vertices.

If there is  $(n + s)/2 < i < n + s/2 \pmod n$  such that  $u = (i, j) \in X$ ,  $w = (i + 1, j) \notin X$ , and  $(i - 1, j) \in X$ . Note that, by the previous paragraph,  $(i + 2, j) \notin X$ . Therefore,  $\gamma(X \setminus u) = \gamma(X) - f_0(w) + f_s(u) < \gamma(X)$ . The last inequality is because  $f_0(w) > f_s(u)$  because of the choice of  $i$ . This contradicts the minimality of  $\gamma(X)$ . If on the other hand,  $(i - 1, j) \notin X$ , then  $\gamma(X \setminus u) = \gamma(X) - f_0(w) - f_0(u) + f_s(u) < \gamma(X)$  which contradicts the minimality of  $\gamma(X)$ .

“Symmetrically”, if there is  $s/2 \leq i \leq (n + s)/2$  such that  $u = (i, j) \notin X$ ,  $w = (i - 1, j) \in X$ , and  $(i + 2, j) \notin X$ . Note that, by the first paragraph,  $(i + 1, j) \notin X$ . Therefore,  $\gamma(X \cup \{u\}) = \gamma(X) - f_s(u) + f_0(i + 1, j) < \gamma(X)$ . The last inequality is because  $f_0(i + 1, j) < f_s(u)$  because of the choice of  $i$ . This contradicts the minimality of  $\gamma(X)$ . If on the other hand,  $(i + 2, j) \in X$ , then  $\gamma(X \cup \{u\}) = \gamma(X) - f_s(u) < \gamma(X)$  which contradicts the minimality of  $\gamma(X)$ .

If  $X \cap C_j$  would not be connected, one of the cases of the two previous paragraphs should occur. Therefore,  $X \cap C_j$  is connected.  $\diamond$

► **Claim 15.** Let  $0 \leq j < m$ . For any  $X \subseteq V$  such that  $X \cap C_j$  is connected,  $\gamma(\mathcal{H}_s \cap C_j) \leq \gamma(X \cap C_j)$ .

*Proof of claim.* Since  $X \cap C_j$  is connected, it has the form  $H(a, b) \cap C_j$  for some  $a$  and  $b$ . We assume that  $|a - b| > 1$  (the other case can be done similarly). Therefore, by previous claims, it remains to prove that, for any  $a$  and  $b$ ,  $\gamma(\mathcal{H}_s \cap C_j) \leq \gamma(H(a, b) \cap C_j)$ .

$$\gamma(H(a, b) \cap C_j) - \gamma(\mathcal{H}_s \cap C_j) =$$

$$\sum_{-s/2+1 \leq i \leq s/2+1} (f_0(b - s/2 + i, j) - f_0(n/2 + i, j)) - \sum_{-s/2 \leq i \leq s/2-2} (f_0(a - s/2 + i, j) - f_0(i, j)).$$

Since the function  $f_0$  is maximum around  $i = 0$  and minimum around  $i = n/2$ , it is easy to check that, for any  $a$  and  $b$ :

$$\sum_{-s/2+1 \leq i \leq s/2+1} (f_0(b - s/2 + i, j) - f_0(n/2 + i, j)) \geq 0$$

and

$$\sum_{-s/2 \leq i \leq s/2-2} (f_0(a-s/2+i, j) - f_0(i, j)) \leq 0.$$

Hence,  $\gamma(H(a, b) \cap C_j) - \gamma(\mathcal{H}_s \cap C_j) \geq 0$ . ◊

By previous claims,  $\kappa_\alpha(\mathcal{H}_s) = \gamma(\mathcal{H}_s) = \min_{A \subseteq V} \gamma(A) \leq \min_{A \subseteq V} \kappa_\alpha(A)$ .

Hence,  $\kappa_\alpha(\mathcal{H}_s) = \min_{A \subseteq V} \kappa_\alpha(A)$ . ◀

► **Lemma 9.** *Let  $n, s \geq 2$ ,  $s \leq n/2$ ,  $d \geq 0$  and  $0 < \alpha \leq \log(3/2) < 1$ . There exists a constant  $B > 0$  (independent of  $n$ ) such that the function  $\omega_\alpha(v) : V(TG_{n \times n}) \rightarrow \mathbb{R}^+$  where  $\omega_\alpha(v) = \frac{B}{(d(v)+1)^\alpha}$  for every  $v \in V(TG_{n \times n})$  is a spy-positional winning fractional strategy that uses  $O(n^{2-\alpha})$  guards to control a spy with speed  $s$  at distance  $d$  in  $TG_{n \times n}$ .*

**Proof.** To verify that  $\omega_\alpha$  is a winning strategy, we need to prove that it satisfies constraints 3 and 6. Let  $B_d$  be the set of vertices at distance at most  $d$  from  $(0, 0)$  and let  $B = 1 / \sum_{v \in B_d} \frac{1}{(d(v)+1)^\alpha}$ .

The total amount of guards used by the strategy is:

$$\sum_{v \in V(TG_{n \times n})} \frac{B}{(d(v)+1)^\alpha} = B \left( \sum_{0 \leq i \leq \lfloor n/2 \rfloor} \frac{4(i+1)}{(i+1)^\alpha} + \sum_{\lfloor n/2 \rfloor < i < n} \frac{4(n-i)}{(i+1)^\alpha} \right) = O(n^{2-\alpha})$$

Constraint 3 states that  $\sum_{v \in B_d} \omega_\alpha(v) \geq 1$  which is satisfied by the choice of  $B$ .

Constraint 6 states that,  $\forall A \subseteq V(TG_{n \times n})$ ,  $\sum_{(i,j) \in N[A]} \omega_\alpha(i, j) \geq \sum_{(i,j) \in A} \omega_\alpha(i-s, j)$ .

By Lemma 8, we know that  $\kappa_\alpha(A) = \sum_{(i,j) \in N[A]} \omega_\alpha(i, j) - \sum_{(i,j) \in A} \omega_\alpha(i-s, j)$  is minimum for  $A = \mathcal{H}_s$ , where  $\mathcal{H}_s = \{(i, j) \mid s/2 \leq i \leq (n+s)/2 \pmod n, 0 \leq j < n\}$ . Hence, it is sufficient to show that  $\kappa_\alpha(\mathcal{H}_s) \geq 0$ .

Again, for ease of presentation, let us assume that  $s$  and  $n$  are even.

$$\kappa_\alpha(\mathcal{H}_s) = \sum_{s/2-1 \leq i \leq (n+s)/2+1, 0 \leq j < n} \frac{B}{(d(i, j)+1)^\alpha} - \sum_{s/2 \leq i \leq (n+s)/2, 0 \leq j < n} \frac{B}{(d(i-s, j)+1)^\alpha}$$

Because  $s \leq n/2$ , this can be simplified as :

$$\begin{aligned}
\kappa_\alpha(\mathcal{H}_s) &= \sum_{(n-s)/2+1 \leq i \leq (n+s)/2+1, 0 \leq j < n} \frac{B}{(d(i, j) + 1)^\alpha} - \sum_{-s/2 \leq i \leq s/2-2, 0 \leq j < n} \frac{B}{(d(i, j) + 1)^\alpha} \\
&\geq (s+1) \sum_{0 \leq j < n} \frac{B}{(d(n/2, j) + 1)^\alpha} - (s-1) \sum_{0 \leq j < n} \frac{B}{(d(0, j) + 1)^\alpha} \\
&= 2(s+1) \sum_{0 \leq j \leq n/2} \frac{B}{(d(n/2, j) + 1)^\alpha} - \frac{(s+1) \cdot B}{(d(n/2, 0) + 1)^\alpha} - \frac{(s+1) \cdot B}{(d(n/2, n/2) + 1)^\alpha} \\
&\quad - 2(s-1) \sum_{0 \leq j \leq n/2} \frac{B}{(d(0, j) + 1)^\alpha} + \frac{(s-1)B}{(d(0, 0) + 1)^\alpha} + \frac{(s-1)B}{(d(0, n/2) + 1)^\alpha} \\
&= 2(s+1) \sum_{1 \leq j \leq n/2+1} \frac{B}{(n/2 + j)^\alpha} - 2(s-1) \sum_{1 \leq j \leq n/2+1} \frac{B}{j^\alpha} \\
&\quad - \frac{2B}{(n/2 + 1)^\alpha} - \frac{(s+1) \cdot B}{(n+1)^\alpha} + (s-1)B \\
&= 2(s+1) \sum_{0 \leq j \leq n/2-1} \frac{B}{(n/2 + j)^\alpha} - 2(s-1) \sum_{1 \leq j \leq n/2} \frac{B}{j^\alpha} \\
&\quad - \frac{2(s+1) \cdot B}{(n/2)^\alpha} + \frac{2(s+1) \cdot B}{n^\alpha} + \frac{2(s+1) \cdot B}{(n+1)^\alpha} - \frac{2(s-1) \cdot B}{(n/2 + 1)^\alpha} \\
&\quad - \frac{2B}{(n/2 + 1)^\alpha} - \frac{(s+1) \cdot B}{(n+1)^\alpha} + (s-1)B \\
&\geq 2(s+1) \sum_{0 \leq j \leq n/2-1} \frac{B}{(n/2 + j)^\alpha} - 2(s-1) \sum_{1 \leq j \leq n/2} \frac{B}{j^\alpha} + O(1/n^\alpha)
\end{aligned}$$

And so:

$$\kappa_\alpha(\mathcal{H}_s) \geq 2(s+1) \sum_{0 \leq j \leq n/2-1} \frac{B}{(n/2 + j)^\alpha} - 2(s-1) \sum_{1 \leq j \leq n/2} \frac{B}{j^\alpha} + O(1/n^\alpha)$$

Since  $0 < \alpha$ , then  $p(x) = \frac{1}{x^\alpha}$  is decreasing, and  $\int_a^{b+1} p(t) dt \leq \sum_{x=a}^b p(x) \leq \int_{a-1}^b p(t) dt$ .  
Hence,

$$\begin{aligned}
\kappa_\alpha(\mathcal{H}_s)/(2B) &\geq (s+1) \int_0^{n/2} \frac{1}{(n/2 + t)^\alpha} dt - (s-1) \int_0^{n/2} \frac{1}{t^\alpha} dt + O(1/n^\alpha) \\
&= \frac{1}{1-\alpha} [(s+1)((n/2)^\alpha - (n/2)^\alpha) - (s-1)(n/2)^\alpha] + O(1/n^\alpha) \\
&= \frac{n^{1-\alpha}}{1-\alpha} [(s+1)(1 - (1/2)^{1-\alpha}) - (s-1)(1/2)^{1-\alpha}] + O(1/n^\alpha)
\end{aligned}$$

Hence,  $\kappa_\alpha(\mathcal{H}_s) \geq 0$  if  $0 \leq (s+1)(1 - (1/2)^{1-\alpha}) - (s-1)(1/2)^{1-\alpha}$ . In other words,  $\kappa_\alpha(\mathcal{H}_s) \geq 0$  if  $2^\alpha \leq \frac{s+1}{s}$  and  $0 < \alpha \leq \log(1 + 1/s) \leq \log(3/2)$ .  $\blacktriangleleft$