Robbing, Surfing and Rioting Games on Graphs: Some Results

Ioannis Lamprou

COATI, INRIA, I3S, CNRS, Sophia Antipolis, France
ioannis.lamprou@inria.fr
Department of Informatics & Telecommunications, University of Athens, Greece
ilamprou@di.uoa.gr

August 27, 2013

Abstract

The focus is on 3 different combinatorial pursuit-evasion games. For the famous *Cops & Robber* game, two variants are examined. That is, the cases of fractional cops or/and fast robber. For the recently introduced *Surveillance* game modeling Web prefetching, a hardness result for bounded degree graphs is provided. Moreover, a generalized *Benefit-Deficit* approach is cited as an alternative field of study. Finally, *Eternal Domination* is studied. Combinatorial and complexity issues are taken into consideration.

Keywords: combinatorial game, cops & robber, fractional, grid, surveillance, marker, surfer, eternal domination, guard, rioter, graph theory, computational complexity

Acknowledgements

I would like to express my gratitude towards Dr. Nicolas Nisse and Dr. Stéphanne Pérennes for their guidance and co-operation during this internship. Many thanks go to PhD candidate Ronan Pardo Soares especially for his assistance on the early stages of the internship. Moreover, I acknowledge all COATI members for the invaluable time I spent with them either on academic or on entertainment level. For the University of Athens, I would like to thank Prof. Ioannis Emiris for his assistance on arranging this internship and Prof. Vassilis Zissimopoulos for his constant advising and co-operation. Finally, I would like to thank my family and friends, whose continuous care and interest in me I deeply appreciate.

Contents

Chapter 1

Introduction

Historically, games have always appeared in human societies and in many different forms. In recent years, mathematical studying of games has received great interest by researchers in many fields in an attempt to formalize them. *Game Theory*, in general, is a broad notion that encompasses a variety of situations; applications come along in Computer Science, Economics, Business, Biology and other areas. We concentrate on a particular subset of Game Theory, namely *Combinatorial Game Theory*, which includes games with specific characteristics. The interest is about games where two players play alternately and both enjoy perfect information over the game. That is, at each turn, each player is familiar with the current and all previous game configurations and picks an action out of a set of publicly known predefined actions. Moreover, the games are deterministic in the sense that no action performed is dependent on any source of randomness. By the end of the game, one player wins and the other loses; there may be no tie. However, notice that definitions are subject to discrepancy due to the vast amount of literature present. For a survey on Combinatorial Game Theory, see [2]. A large bibliography on the area can be found in [17].

In this report, we focus on 3 specific combinatorial games played on graphs. Consequently, we identify a strong interaction with *Graph Theory* for these games. In other words, the features of the graphs, on which the games are described and played, have a major impact on the dynamics emerging between the 2 players. The games under consideration in this report are the *Cops & Robber* game, the *Surveillance* game and the *Eternal Domination* game. In *Cops & Robber*, one player handles a set of *cops* and the other a single *robber* in the sense of tokens lying on certain vertices of the graph. The players alternately move their tokens (in a way to be specified) and they seek victory: the robber hopes to always escape, while the cops strive for a way to capture her. In *Surveillance*, one player handles a token, namely the *surfer*, while the other deposits marks on the vertices of the graph. The surfer's objective is to reach an unmarked node, while the *marker* wishes to mark all graph vertices before the surfer accomplishes that. Finally, in *Eternal Domination*, a set of tokens, namely the *guards*, lie on the graph and are controlled by one player, while the other (say the *rioter*) attacks at each turn a specific vertex. Guards win if at least one of them can always immediately move to a just-attacked vertex. If they fail to do so, then the rioter wins. More information on the definition and bibliography for these games is given at the beginning of each corresponding chapter.

This introduction reaches its end after a short section with preliminary notions and notation to be used later on in the report. Specialized notation is introduced and used when needed inside the specific chapter. Chapter 2 deals with some results on *Cops & Robber* where some variations, in the way that the cops or the robber can move, are taken into consideration. In Chapter 3, we examine the newly-proposed *Surveillance Game*: a result for an open question is given and a generalized extension is attempted. In Chapter 4, the topic is *Eternal Domination* with a focus on computational complexity. Finally, the report concludes with a summary of the work presented and some directions for further study.

1.1 Preliminaries

1.1.1 Graph Theory

A *graph* G is defined as a pair of two sets: the set of *vertices* $V(G)$ and the set of *edges* $E(G)$. A vertex is otherwise called a *node*, while an edge is an unordered pair of two nodes. The set of edges is therefore a subset of the set of all possible unordered pairs of vertices $\{\{x,y\} : x,y \in V(G)\}$ and we write $e = \{x,y\} \in E(G)$ when edge e connects nodes x and y of $V(G)$. Furthermore, x and y are called the *endnodes* of e. For the sets' cardinalities, we use $n = |V(G)|$ and $m = |E(G)|$. Unless otherwise stated, all graphs mentioned are *simple* graphs, i.e. they contain neither *loops* ($\{x,x\}$ edges) nor *parallel* edges (multiple edges between the same two nodes).

A *directed* graph G is similar to an (undirected) graph with the variation that the edge set $E(G)$ is now a subset out of the set of all possible *ordered* pairs of vertices $\{(x,y) : x,y \in V(G)\}$. That is, an edge $e = (x,y)$ is now an *arc* whose origin is x and whose destination is y.

Two vertices connected by an edge are called *adjacent* or *neighboring* vertices and they are *incident* to that edge. The (open) *neighborhood* of a node $v \in V(G)$ is defined as $N(v) = \{u \in V(G) : \{v,u\} \in E(G)\}$, while the *closed* neighborhood is defined as $N[v] = N(v) \cup \{v\}$. The *degree* of a node $v \in V(G)$ is defined as $d(v) = |N(v)|$. The mininum degree of G is denoted by $\delta(G)$, where $\delta(G) = \min_{v \in V(G)} d(v)$. The maximum degree of G is denoted by $\Delta(G)$, where $\Delta(G) = \max_{v \in V(G)} d(v)$. A graph is called *regular* if all its nodes have equal degree. In this case, $\delta(G) = \Delta(G) = k$ and the graph is referred to as k-regular.

A *path* in G is a sequence of nodes $v_0, v_1, \ldots, v_k$, where $v_i \neq v_j$ for any $0 \leq i,j \leq k$ and $\{v_i, v_{i+1}\} \in E(G)$ for any $0 \leq i \leq k-1$. The case when $v_0 = v_k$ is called a *cycle*. The path (cycle) is then said to be of *length* k and is denoted as P_k (C_k). The definition for a *directed* path (cycle) is similar; (v_i, v_{i+1}) needs to be an arc in $E(G)$, where G is a directed graph. A graph is *connected* if there is a path between any two nodes of it.

A graph that contains no cycles is called a *tree*. A tree can be considered *rooted* under any of its nodes and drawn in a specific way on the plane. In this case, for any node there is a *father*: the node above him to which it is connected. Nodes connected and a level below of a certain node are called the *children* of that node. A *leaf* is a tree node of degree 1.

The *cartesian product* of two graphs G_1 and G_2 is defined as a new graph $G_3 = G_1 \times G_2$, where $V(G_3) = V(G_1) \times V(G_2)$ and $E(G_3) = \{\{x,y\} : x = \{v_1, v_2\} \in V(G_1) \times V(G_2), y = \{v_3, v_4\} \in V(G_1) \times V(G_2)$ and $v_1 = v_3$ or $v_2 = v_4\}$. The cartesian product of two paths is called a *grid*.

A graph is *chordal* if each of its cycles of four or more nodes has a *chord*, which is an edge joining two nodes that are not adjacent in the cycle. An *interval* graph is the intersection graph of a multiset of intervals on the real line.

The *girth* of a graph is the length of its shortest cycle. A graph is called *planar* if it can be drawn on the plane without any intersecting edges.

A *dominating set* is a subset of vertices of G such that every vertex of $V(G)$ is either included in it or has at least one neighbor in it. The *domination number* of G, denoted $\gamma(G)$ is the cardinality of a minimum dominating set of G.

An *independent set* is a subset of vertices of G such that there is no edge between any two of them. *The indepence number* of G, denoted $\alpha(G)$, is the cardinality of a maximum independent set of G.

A subset of nodes of G such that each edge $e \in E(G)$ has either one or both of its endpoints in it, is called a *vertex cover* of G. $\tau(G)$ stands for the cardinality of a minimum vertex cover of G, namely the *vertex cover number* of G.

A subset of nodes, for which all possible edges are present, is called a *clique*. A partitioning of the vertex set into disjoint subsets, such that each subset forms a clique, is called a *clique cover* of G. The corresponding size of a minimum such partitioning of G is called the *clique cover number* of G and it is denoted by $\theta(G)$.

For any disambiguation or further information on basic graph-theoretic notions, please refer to a graph theory textbook, e.g. [40, 8].

1.1.2 Complexity

For notions regarding the field of Computational Complexity, the reader is referred to any standard textbook, e.g. [34].

Chapter 2

Cops & Robber

2.1 The Game

2.1.1 Description

Cops & Robber is a pursuit-evasion combinatorial game played on a graph. From now on, we will always assume that the graph is connected, simple and finite. There are two players: one that controls the cop tokens and another who controls the robber token. Let us call them player $\mathscr{C}$ and player $\mathscr{R}$ respectively. Initially, player $\mathscr{C}$ places his k tokens on the vertices of the graph. Notice that more than one cop tokens may lie on the same node. Then, player $\mathscr{R}$ chooses an initial placement for the robber. Round 0 is over. From now on, every *round* consists of 2 turns (one for $\mathscr{C}$ and one for $\mathscr{R}$), where $\mathscr{C}$ may or may not move any of his cops to a vertex adjacent to the one he currently lies on and $\mathscr{R}$ moves the robber to an adjacent vertex with respect to her current position or does not move her at all. $\mathscr{C}$ wins the game if, after any player's turn, the robber lies on the same vertex with a cop. $\mathscr{R}$ wins if she can perpetually avoid the realization of the aforementioned condition. Both players try to devise *strategies* in order that they win against any possible strategy of their adversary. A strategy is a set of movement rules for either the cops or the robber.

2.1.2 Background

From the optimization point of view, the important question in mind is what is the minimum number of cops needed to capture the robber either on a specific graph or in general. For this purpose, we define this number as the *cop number* of a graph.

Definition 1. *The cop number of a graph G, denoted $cn(G)$, is the minimum number of cops needed to ensure that the robber is captured, regardless of her strategy.*

Problems related to the cop number have been studied heavily over the last 30 years. Originally, Quillot [37] and Nowakowski and Winkler [33] characterized graphs with cop number equal to 1. Since then, there has been a lot of literature in proving different lower and upper bounds for the cop number of specific graph classes. For instance, Aigner and Fromme [1] proved that $cn(G) \leq 3$ for any planar graph G. Frankl [18] proved a lower bound for graphs of large girth. Other work includes [4, 13, 31]. Moving onto general graphs, Meyniel conjectured that $\sqrt{n}$ cops are always sufficient to capture the robber. Chiniforooshan [10] proved an $\mathcal{O}(n/logn)$ upper bound, which was improved by Scott and Sudakov [39] and Lu and Peng [29] to $\mathcal{O}(n2^{(1-o(1))\sqrt{logn}})$. Thus, yet the conjecture remains open. On the contrary, the conjecture was recently proved positive on random graphs [36]; previous work included [5, 7, 30]. For an introduction to random graphs, see [25]. Finally, there exists a book capturing all the activity on Cops & Robber until recently: see [6].

The computational complexity of the specific decision problem is also worth a note. The question to be answered is: *given a graph G and an integer k, does $cn(G) \leq k$ hold?*. Goldstein and Reingold [24] proved that the problem is EXPTIME-complete given that the graph is directed or the initial positions are given. Recently, Mamino [32] proved PSPACE-hardness again by using a restriction to enable the proof.

2.2 Fractional Cop Number

Relaxing the description is a general technique followed in order to augment the understanding on a hard combinatorial problem. Furthermore, the relaxed version could provide an approximation for the original one. Thus, studying the integrality gap for a (later modified or not) relaxed solution has become a significant field of research. Authors of [21] study a natural relaxation for the cop number of a graph, namely the *fractional cop number* of a graph. The fractional cop number (in short *fcn*) refers to the original Cops & Robber game, but with the relaxation that the cops can now split into infinitely small and infinitely many pieces and move such pieces along the edges of the graph. The robber remains integral (i.e. she cannot split). Furthermore, in [21] it is proved that splitting would not assist her towards escaping. The sum of all cop pieces remains always equal to a constant $k \in \Re^+$. In order for the robber to be captured, a quantity of cops ≥ 1 needs to lie on the same vertex as her. We now review a cop-strategy and then present some new remarks on it. This game relaxation is referred to as *Fractional Cops & Robber*.

2.2.1 The Strategy

We discuss a variation of the strategy given in [21], which does not substantially differ from the original one, but it helps us simplify the remarks that follow:

Initially, the k cops are placed uniformly on the graph, i.e. k/n cops are placed at each node. Then, the robber places herself to a node υ. Now, it's the cops' turn. The k/n cops that lie on the same vertex as the robber will (from now on) always follow the robber. The rest $k - k/n = k(n-1)/n$ cops are spread uniformly on the graph (we later show that this can be done in exactly 1 cops' step for any graph). That is, each vertex is now guarded by $k(n-1)/n^2$ cops. We do not ever reconsider cop quantities that are bound to always follow the robber. Now, it's the robber's turn. Whatever her move, the cops will repeat the same strategy, i.e. the quantity that lies on the same vertex will always follow her from now and the rest are re-spread uniformly over the graph. Inductively, at step t there will be $x_t = k(\frac{n-1}{n})^t$ cops left on the graph, which are not bound to always follow the robber. The rest $y_t = k - k(\frac{n-1}{n})^t = k(1-(\frac{n-1}{n})^t)$ cops accumulate on the robber until this quantity eventually becomes ≥ 1 at a certain round, hence cops win. From now on, we refer to this strategy as the *spread and follow* strategy.

2.2.2 Background Results

It is proved [21] that the fractional cop number of any graph approaches 1 if we allow a very large number of steps. One need only notice that $\lim_{t\to\infty} x_t = 0$ and so for the quantity that accumulates on the robber $\lim_{t\to\infty} y_t = k$. After an infinite number of steps, all k cops are eventually on the robber. That is, for any $k \geq 1$ the above strategy manages to accumulate all cops on the robber. Obviously, just 1 is enough to capture her, leading us to the following result.

Let $fcn_\infty(G)$ stand for the fractional cop number of graph G for a *Fractional Cops & Robber* game of infinite duration.

Theorem 1. [21] $\forall G : fcn_\infty(G) = 1$.

Finally, they show that if we allow a finite number of steps, then still just a bit more than 1 cop is needed.

Theorem 2. [21] $\forall G\, \forall \varepsilon > 0 : fcn(G) \leq 1 + \varepsilon$.

2.2.3 New Remarks

Bounded time

The aforementioned results suggest that the fractional cop number does not yield any help towards the approximation of the integral cop number, since it is always (almost) 1. Cops indeed become very powerful if we allow them the capability of fractionalizing themselves. A further suggestion to try to reduce the cops' power, in order to possibly narrow the gap between the fractional and the integral cop numbers, is to bound the number of steps allowed to them (i.e. reduce the duration of the game). For example, let us consider y_n, i.e. the quantity that is accumulated on the robber after $n = |V(G)|$ rounds of the game.

$$\begin{aligned}
y_n &= k(1-(\frac{n-1}{n})^n) \\
&= k(1-e^{\log(\frac{n-1}{n})^n}) \\
&= k(1-e^{n\log(1-1/n)}) \\
&\sim k(1-e^{n(-1/n)}) \\
&= k(1-e^{-1}) \\
&= k(e-1)/e
\end{aligned}$$

What we wish for is $y_n \geq 1$, i.e. $k(e-1)/e \geq 1$ which leads to $k \geq e/(e-1) \approx 1.58$. Therefore, we notice that even in a few (linear) number of steps, even less than 2 cops suffice to capture the robber. To conclude, upper-bounding the number of steps allowed in the game still does not help us in our objective to find a relation to the integral cop number, since the bounded-time fractional cop number remains small enough. The following table summarizes the lower bounds on cop quantity obtained in this scope.

Steps Allowed	Cops Required
∞	1
n	1.58
$\sqrt{n}$	$\sqrt{n}$
$\log n$	$O(n)$

Table 2.1: Fractional cop numbers for bounded duration games

Notice that if we further restrict to $\sqrt{n}$ or *logn* steps, the number of fractional cops needed grows very big ($\geq \sqrt{n}$) by using this specific strategy and so it gives no information on the integral cop number. This happens due to the big gap observed between the n steps and the $\sqrt{n}$ steps case.

Spreading Uniformly in 1 Step

For the reasoning made to be completely accurate, we need to make sure that the cops' strategy is feasible. In this paragraph, we prove that, at each step, the remaining cop quantity (the ones who do not follow the robber) can move from its current state ($n-1$ vertices carry $k(n-1)^t/n^{t+1}$ available cops each and 1 vertex -where the robber lies-carries 0 available cops) to a new uniform state (each of the n nodes carries $k(n-1)^{t+1}/n^{t+2}$ available cops) in just one cops' step.

Lemma 1. *The k cops have followed the "spread and follow" strategy for $t \geq 1$ rounds of Fractional Cops & Robber on graph G. The robber moves to vertex υ during her turn at round t. The total cop quantity lying on $V(G) \setminus \{\upsilon\}$ can be moved such that it lies uniformly on $V(G)$ after cops' turn at round $t+1$.*

Proof. Pick any spanning tree T of G (this can be done in $O(|E(G)|)$ time using a depth first search approach) and consider it rooted under υ. Let the quantity $k(n-1)^t/n^{t+2}$ be called a *piece*. The re-spreading algorithm consists of the following statement: Each node $v \in V(T) \setminus \{\upsilon\}$ sends to its father the size of the subtree under it (including itself) many pieces. We show by induction that each node in T carries $k(n-1)^{t+1}/n^{t+2}$ cops after such a procedure. Thus, a total quantity of $k(n-1)^{t+1}/n^{t+1}$ cops is spread uniformly on G.

Any leaf v sends to its father one piece, so the remaining quantity on v is now $k((n-1)^t/n^{t+1} - (n-1)^t/n^{t+2}) = k(n(n-1)^t - (n-1)^t)/n^{t+2} = k(n-1)^{t+1}/n^{t+2}$ like desired. Now, consider a non-leaf node $i \in V(T) \setminus \{\upsilon\}$ and assume that all the subtrees hanging from its children are fixed. Let $child(i)$ stand for the set of children of i and T^i for the subtree of T hanging under node i and including it. Any $c \in child(i)$ sends

$|V(T^c)|$ many pieces to i. Then, i receives $\sum_{c\in child(i)} |V(T^c)|$ pieces and it needs to send $|V(T^i)| = 1+\sum_{c\in child(i)} |V(T^c)|$ many pieces. Therefore, the total quantity left on vertex i is $k((n-1)^t/n^{t+1} - (1+\sum_{c\in child(i)} |V(T^c)|)(n-1)^t/n^{t+2} + \sum_{c\in child(i)} |V(T^c)|(n-1)^t/n^{t+2} = k((n-1)^t/n^{t+1} - (n-1)^t/n^{t+2}) = k(n-1)^{t+1}/n^{t+2}$ which concludes the induction. Finally, node υ will receive $\sum_{c\in child(\upsilon)} |V(T^c)| = n-1$ pieces, which will increase its quantity from 0 to $(n-1)k(n-1)^t/n^{t+2} = k(n-1)^{t+1}/n^{t+2}$. $\square$

Limited Fractionality

Another idea is to restrict the degree of liberty given to fractional cops in order to obtain a measurement *better* than the fractional cop number examined heretofore. In this case, *better* means that it could hopefully relate to the cop number and so present us with more useful information. The restriction we follow is to put a limit on the cops' ability to divide themselves. That is, let the α-fractional cop number of a graph G (denoted $fcn_\alpha(G)$, where α can be either a constant or a function, but always greater than 0 and no more than 1) stand for the minimum number of cops needed to win in G, when cops are allowed to split but only in a way that, after every cops' turn, the cop quantity lying on any node of G is a multiple of α. In a sense, after the cops' turn, there is $k\cdot\alpha$ cop quantity on any node of G, where $k \in \mathbb{N}^*$. Moreover, $\lim_{\alpha\to 0} fcn_\alpha(G) = fcn(G)$.

One can now observe that an α-fractional cop strategy can be transformed to an integral cop strategy: replace any cop quantity α with cop quantity 1 and perform the exact same strategy. Plainly, an upper bound for the cop number can be immediately derived leading us to the following corollary.

Corollary 1. *$cn(G) \leq 1/\alpha \cdot fcn_a(G)$ for any graph G.*

To continue, we derive an upper bound for $fcn_\alpha(G)$ in terms of $cn(G)$.

Lemma 2. *$fcn_\alpha(G) \leq 1+(cn(G)-1)\cdot\alpha$ for any graph G.*

Proof. Initially, pick $cn(G)$ many cop quantities of size α. They follow the integral strategy to catch the robber. Eventually, 1 piece lies on her. Should the robber decide to move on any of these pieces, the strategy halts and the specific piece(s) just follow the robber from now on. Besides, any cop quantity, on which the robber steps on, follows her from now on. Whenever the strategy reaches its end or is halted, another available $cn(G)$ pieces are picked at random and repeat the integral strategy. At some point, either $1/\alpha$ pieces lie on the robber (thus cops win) or $1/\alpha - 1$ pieces lie on her. In the latter case, there exists a remainder of available $cn(G)$ pieces, which follow the integral strategy. Eventually, another piece gets on the robber which, together with the pieces that already follow her, sums up to 1 and hence the game is over. $\square$

By combining these two facts, we reach the conclusion that $fcn_\alpha(G)$ directly provides us with an approximation for $cn(G)$ up to an additive factor of $1/\alpha - 1$.

Corollary 2. *$fcn_\alpha(G)/\alpha + 1 - 1/\alpha \leq cn(G) \leq fcn_\alpha(G)/\alpha$ for any graph G.*

To conclude this part, we provide some lower bounds for fcn_α by adopting results in [1] and [18] regarding graphs with large girth.

Lemma 3. *For any graph G with girth greater than* 4 *it holds* $fcn_\alpha(G) \geq \alpha\delta(G)$.

Lemma 4. *For any graph G with girth greater or equal to* $8t-3$ *and* $\delta(G) > d$ *(where* $t,d \in \mathbb{N}$*) it holds* $fcn_\alpha(G) > \alpha d^t$.

The reader is referred to the proofs in [1, 18] for the integral case, which these ones directly follow. One need only replace each integral cop with α cop quantity and apply the original reasoning. In this way, the evasion strategy proposed for the robber guarantees that she remains at all times cop-free, i.e. no cop quantity ever lies on the same node as she does.

2.3 Fast Robber

We now turn our attention to the *Fractional Cops & Robber* game variant where the robber can move with speed 2, i.e. at each step she can move on a path of length at most 2 from her current position. Several natural questions arise in this case about the definition of the game e.g. *Can the robber jump over some cop quantity? What happens when a robber co-exists with some cop quantity on a vertex?* We try to define the game in a way that it handles such questions, nevertheless it remains as natural as possible. Furthermore, we try to understand whether fractional cops can perform better than integral ones in this context. We focus on square grid graphs and obtain some bounds for the (fractional) cop number on small grids.

2.3.1 Game Definition

Initially, let us restate the assumption that the robber is not allowed to fractionalize herself. Moreover, by following and extending the reasoning in [21] we understand that she has not interest to split. While in the original version cops and robber would pick a move out of a set Δ_G of stochastic matrices, in this case the robber is differentiated in that she picks a move out of another set, say $\Delta_R \supset \Delta_G$. Formally,

$$\Delta_R = \left\{ \quad [a_{ij}]_{1\leq i,j\leq n} \;\middle|\; \begin{array}{l} a_{ij} \geq 0 \\ \forall j : \sum_i a_{ij} = 1 \\ \forall i,j \text{ where } d(v_i,v_j) > 2 : a_{ij} = 0 \end{array} \right\}$$

, where $d(v_i,v_j)$ stands for the shortest path distance between v_i and v_j on G and $a_i j$ for the robber amount moving from node j to node i.

Let us now try to consider some possible answers to the questions posed about the fast robber extension. We take into consideration the *jumping* ability of the robber. A logical thought would be that the robber can jump over certain cop quantity , i.e. move in a position of distance two neighboring to a position at distance one with a certain cop amount. That is, the robber can jump over node i, if $c_i < \alpha$, where $\alpha \in [0,1] \cap \Re$ and c_i is the cop quantity on node i. Such an approach, models all kinds of different situations, e.g. for $\alpha = 0$, the robber cannot jump over any cop quantity (so cops shall easily win) and for $\alpha = 1$ the only restriction is that she may not jump over a whole cop. This approach is quite general and it is the one adopted by the author (with focus on $\alpha = 1$). Besides, another argument is that a robber quantity r could jump over node i if $r > c_i$, but since the robber remains integral, this case reduces to the aforementioned $\alpha = 1$ case. In turn, this indicates that the robber may move wherever she wishes in distance at most 2. Of course, jumping over a whole cop would result in her capturing, since her new position would be dominated. Notice that the same reasoning ($c_i < \alpha$) can apply to the question whether the robber can co-exist with a certain cop quantity on a vertex.

On the other hand, let us consider the *capturing* rules of the game. Suppose that whenever some robber quantity r and some cop quantity $c < r$ co-exist on a specific graph vertex, then the c cop quantity captures exactly the same robber quantity. This approach does indeed make cops really powerful, since given their ability to subdivide themselves, they can always be present on any graph node and thus using a strategy like *spread and follow* (recall the previous section) they could easily and rapidly capture the whole robber (she will be diminishing on every single round until nothing is left). The formulation we are going to follow is that a whole cop is needed to capture the robber. So, the only way we could capture the unsplittable robber is that a cop quantity ≥ 1 lies on the same vertex as her. This approach seems more natural, as it reminds us of the more or less standard way to catch the robber.

For the above discussion, the resulting definition generated sounds actually quite simple and natural; only the robber's speed is changed, while the rest can remain the same.

Definition 2. *Fractional Cops & Fast Robber* is the same game as *Fractional Cops & Robber* with the extension that the robber has speed 2, i.e. she can move from her current position to any vertex at distance at most 2.

A special case of this game is not to allow cops to divide themselves. This is exactly the integral game with a robber of speed 2.

2.3.2 $N \times N$ Grids

We focus our attention on square grid graphs and provide some useful intuition and preliminary results on the fast robber case. Thence, let $cn_2(G)$ stand for the cop number of graph G for the *Cops & Fast Robber* game and $fcn_2(G)$ for the corresponding fractional one. $fdn(G)$ embodies the fractional domination number of G.

Background Results

One can check that for a normal-speed robber, 2 cops suffice to capture her on the integral game [31]. For fractional cops, just 1 is necessary as pinpointed in the previous section. As far as the fast robber case is concerned, there exists a $O(\sqrt{logn})$ lower bound [16] and the best known upper bound does not escape from $O(n)$ [11]. Notice that n cops can capture a fast robber in a $n \times n$ grid; just put them on any horizontal or vertical line and move towards her.

Discussion

Let the k-neighborhood ($k \geq 0$) of a node $\upsilon \in V(P_n \times P_n)$ stand for the set of nodes at distance exactly k from υ and denote it by $N_k(\upsilon)$. In addition, let $N_{[k]}(\upsilon) = \bigcup_{0 \leq i \leq k} N_i(\upsilon)$. Suppose that the robber has just moved and she lies on vertex υ. It comes to our understanding that if we manage to dominate all nodes in $N_k(\upsilon)$ in a way that the robber may only move in nodes residing in $N_{[k-1]}(\upsilon)$ for the rest of the game, then cops will eventually win the game by progressively dominating $N_{k-1}(\upsilon), N_{k-2}(\upsilon), \ldots, N_0(\upsilon) = \{\upsilon\}$. The cops form a diamond-like shape around the robber and they steadily narrow the diamond as they progress towards υ. Unfortunately, intuition suggests that in a big grid, a big number of cops is needed as well to make use of this remark.

Some Bounds

On the next table, we present some values obtained for the fractional as well as the integral cop number for small $n \times n$ grids ($n \leq 5$). The fractional numbers proven mostly derive from a domination analysis of $N_{[2]}(\upsilon)$ for any possible robber position υ. Finally, our pursuit ends with a derived lower bound for the fractional cop number of a big grid.

n	$fcn_2(P_n \times P_n)$	$cn_2(P_n \times P_n)$
1	1	1
2	$4/3$	2
3	2	2
4	2	2
5	$\in [2,3]$	3

Table 2.2: Small grids' (fractional) cop numbers for a fast robber

Lemma 5. $fcn_2(P_1 \times P_1) = cn_2(P_1 \times P_1) = 1$.

Proof. Trivially place a cop on the single vertex available. □

Lemma 6. $fcn_2(P_2 \times P_2) = fdn(P_2 \times P_2) = 4/3$.

Proof. Suppose $fcn_2(P_2 \times P_2) < fdn(P_2 \times P_2)$. Then, there is at least one undominated vertex where the robber can initially place herself. The cops move in any way they desire. Due to the robber's speed 2, she may move to any out of the 4 vertices. Since less than $fdn(P_2 \times P_2)$ cops exist, there will be at least one undominated vertex for her to move. The robber repeats this strategy indefinitely and escapes capture. Hence, $fcn_2(P_2 \times P_2) \geq fdn(P_2 \times P_2)$.

Given $fdn(P_2 \times P_2)$ fractional cops, they can place themselves such that they dominate all 4 vertices. No matter which vertex she picks for her initial move, her position is dominated and thus she gets captured. Formally, $fcn_2(P_2 \times P_2) \leq fdn(P_2 \times P_2)$ which concludes the proof. The value of $4/3$ is easily obtained by solving the corresponding linear program for the domination of $P_2 \times P_2$. □

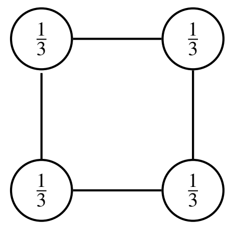

Figure 2.1: The optimal solution for fractionally dominating $P_2 \times P_2$

Notice that the above lemma can be extended to any graph of diameter ≤ 2, since at each step of the game the robber may move to any vertex of the graph. Thus, for her to be captured, all the vertices need to be dominated at some step.

Corollary 3. *For any graph G, where* $diameter(G) \leq 2$*:* $fcn_2(G) = fdn(G)$.

Lemma 7. $cn_2(P_2 \times P_2) = 2$.

Proof. Suppose $cn_2(P_2 \times P_2) = 1$. The cop places his token in any out of the four vertices and thus dominates 3 out of 4 vertices of the graph (the one he lies and the two neighboring ones). The robber places her token at the only undominated vertex. The robber repeats the following strategy indefinitely: if the cop reaches a neighboring vertex to her, then the robber picks the other neighboring vertex which is undominated (there is such a vertex due to the graph's topology); otherwise she stays put. Hence, $cn_2(P_2 \times P_2) > 1$.

Given 2 integral cops, they initially place themselves in any 2 nodes of the graph. Observe that the whole graph is now dominated. The robber places her token in any vertex and the cops win in at most 1 move. □

Let us continue with the 3×3 grid, where we note 3 different sets of positions. Let us call these sets $core_{3\times3}$, $side_{3\times3}$ and $corner_{3\times3}$, whose names correspond to the vertices they contain. Moreover, let us name the vertices horizontally and from left to right, i.e. the first line of the grid being $\upsilon_1, \upsilon_2, \upsilon_3$, the second $\upsilon_4, \upsilon_5, \upsilon_6$ and the third $\upsilon_7, \upsilon_8, \upsilon_9$. Then, $core_{3\times3} = \{\upsilon_5\}$, $side_{3\times3} = \{\upsilon_2, \upsilon_4, \upsilon_6, \upsilon_8\}$ and $corner_{3\times3} = \{\upsilon_1, \upsilon_3, \upsilon_7, \upsilon_9\}$.

Lemma 8. $cn_2(P_3 \times P_3) = 2$.

Proof. Since $P_2 \times P_2$ is an isometric subgraph of $P_3 \times P_3$, then it holds $cn_2(P_3 \times P_3) \geq cn_2(P_2 \times P_2) = 2$. We show a strategy for 2 integral cops to capture the robber: initially put 1 cop at υ_4 and 1 at υ_6 (symmetrically, υ_2 and υ_8 would have worked the same; we just need two opposite side nodes). The robber can now place her token either at υ_2 or at υ_8, since all other vertices are dominated. Let us assume that she picks υ_2 for her initial placement (the other case works symmetrically). Now, the cops move such that they inhabit υ_5 and υ_1 (or υ_3; symmetric case). The robber has a sole option to move to υ_3. Then, cops move to υ_2 and υ_6. It is the robber's turn, but we notice that all vertices within distance 2 from her current position are dominated. Whatever the robber's move, in the next round the cops win. □

In the figure below (and the ones to follow), each subfigure represents the state of the game after one round (cops and robber turn). The game proceeds from left to right.

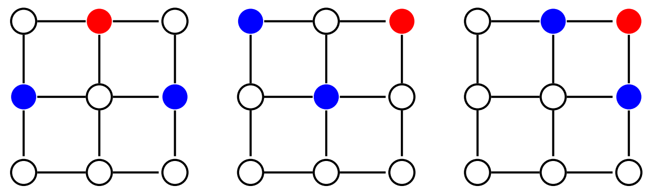

Figure 2.2: A strategy for 2 cops to capture the robber in the 3×3 grid

We now introduce a new notion to assist us on the fractional analysis. Let $fdn_{v,[s]}(G)$ stand for the the fractional domination quantity needed to dominate all nodes at distance at most s from v in graph G, but with the extra restriction that the quantity on v is strictly less than 1. From now on, $fdn_{v,[s]}(G)$ will also be refered to as the $(v,[s])$-fractional domination number of G. We make use of this quantity, since we wish to consider the amount of cops needed to capture the robber after she places herself on vertex v. The extra restriction is put, since there cannot be a ≥ 1 cop quantity on v, otherwise the game would be immediately over. The linear program below computes $fdn_{v,[s]}(G)$ for any node $v \in V(G)$. In the following figure, let c_i stand for the cop quantity at node i.

Minimize $\sum_{i \in V(G)} c_i$	*such that*
$\sum_{j \in N[i]} c_j \geq 1$	$\forall\, i \in N_{[s]}(v)$
$c_v < 1$	

Figure 2.3: Linear program for the $fdn_{v,[s]}(G)$ of any node $v \in V(G)$

Notice that the final constraint can be otherwise stated as $c_v \leq 1 - \varepsilon$ for any $\varepsilon > 0$ in order to be made appropriate for a linear programming solver. Below, we focus on the $(v,[2])$ case, since the robber is restricted to speed 2. Therefore, she can move to a node within distance at most 2 and thus these nodes capture our interest.

Lemma 9. *The $(v,[2])$-fractional domination numbers for any node $v \in V(P_3 \times P_3)$ are:*

- *if $v \in core_{3\times3}$, then $fdn_{v,[2]}(P_3 \times P_3) = fdn(P_3 \times P_3) = 5/2$,*
- *if $v \in side_{3\times3}$, then $fdn_{v,[2]}(P_3 \times P_3) = 2 + \varepsilon$ for any $\varepsilon > 0$,*
- *if $v \in corner_{3\times3}$, then $fdn_{v,[2]}(P_3 \times P_3) = 2$.*

Proof. By solving the corresponding linear programs for domination of the specific nodes. See the figures below for an illustration. ☐

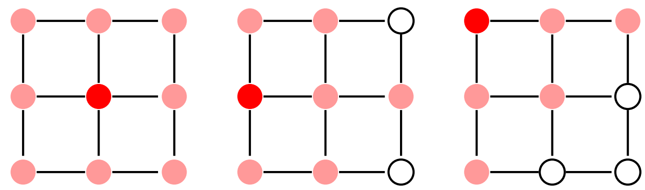

Figure 2.4: Vertices to be dominated for robber positions on the 3×3 grid

Lemma 10. $fcn_2(P_3 \times P_3) = 2$.

Proof. To start with, $fcn_2(P_3 \times P_3) \leq cn_2(P_3 \times P_3) = 2$. We provide a strategy for the robber to avoid $2 - \varepsilon$ fractional cops for any $\varepsilon > 0$: Initially, there is at least 1 undominated vertex for the robber to place herself, since $fdn(P_3 \times P_3) = 5/2 > 2 - \varepsilon$. At any later round, the robber finds herself in a core, side or corner vertex. By the previous lemma, no matter where she lies, ≥ 2 cops are needed to dominate all nodes lying within distance at most 2. Since only $2 - \varepsilon$ cops are available, there is at least one available vertex $v \in N_{[2]}(v)$. The robber moves to v and escapes capture for this round. Repeat for any round and the robber can always escape. □

Let us move on to the 4×4 grid. Again, we consider the vertices named left-to-right and horizontally $\upsilon_1, \ldots, \upsilon_{16}$. We partition the vertices in 3 sets as follows: $core_{4\times4} = \{\upsilon_5, \upsilon_6, \upsilon_9, \upsilon_{10}\}$, $side_{4\times4} = \{\upsilon_2, \upsilon_3, \upsilon_5, \upsilon_8, \upsilon_9, \upsilon_{12}, \upsilon_{14}, \upsilon_{15}\}$ and finally $corner_{4\times4} = \{\upsilon_1, \upsilon_4, \upsilon_{13}, \upsilon_{16}\}$.

Lemma 11. $cn_2(P_4 \times P_4) = 2$.

Proof. $cn_2(P_4 \times P_4) \geq cn_2(P_3 \times P_3) = 2$, since $P_3 \times P_3$ is an isometric subgraph of $P_4 \times P_4$. Moreover, we demonstrate a stategy for 2 cops to win against any possible robber strategy: Initially, one cop is placed on υ_6 and the other on υ_{11}. Then, the robber may pick an initial position out of $\upsilon_1, \upsilon_3, \upsilon_4, \upsilon_8, \upsilon_9, \upsilon_{13}, \upsilon_{14}, \upsilon_{16}$ as depicted in the next figure. Notice that without loss of generality, we can safely bypass positions υ_9 (symmetrical to υ_8), υ_{13} (symmetrical to υ_4), υ_{14} (symmetrical to υ_3) and υ_{16} (symmetrical to υ_1). Furthermore, υ_8 is symmetrical to υ_3 respecting cops position, thus the strategy followed is symmetrical and we can ignore this case as well. That is, let us focus our attention to the case where the robber initially places herself on any node out of $\upsilon_1, \upsilon_3, \upsilon_4$.

- The robber initially places herself on node υ_1. Cops move such that they now lie on nodes υ_5 and υ_7. The robber has only one plausible escape, which is to move to υ_2; all other nodes in $N_{[2]}(\upsilon_1)$ are dominated. Now, the 2 cops move to υ_6 and υ_3, respectively. Again, the robber is restrained to move to her sole option, i.e. υ_1, otherwise she loses. After that, the cops can and will move to υ_5 and υ_2. Clearly, no escape exists for the robber anymore. The cops win after at most 1 step.

- The robber initially places herself on node v_3. The 2 cops move upwards to v_2 and v_7. The robber moves to her only available option, v_4. The cops now move to the right on nodes v_3 and v_8. The robber is trapped in the corner. Cops win in at most 1 step.
- The robber initially places herself on node v_4. The cops move to the right on nodes v_7 and v_{12}. The robber now picks her available move of speed 2 to node v_2, otherwise if she stays put, then the cops will easily trap her in the corner. The cops move such that they attain positions v_6 and v_8. The robber may now move to either v_1 or v_3; all other potential moves would result in her being captured in at most 2 cop steps. We consider both cases:
  - Suppose the robber moves to v_1. Then, the cops move to the left on nodes v_5 and v_7. The robber moves to the only undominated option v_2. Cops move now onto v_6 and v_3. The robber again has only one available option, which is v_1. But, the cops can now move to v_5 and v_2, trap her in the corner and win.
  - Suppose the robber moves to v_3. Then, the cops attain positions v_2 and v_7. The robber picks her only option, which is moving to v_4. Now, the cops need only move to the right on nodes v_3 and v_8. The robber is trapped in the corner and soon the game is over.

□

Refer to the figures below for a visual interpretation of the previous proof.

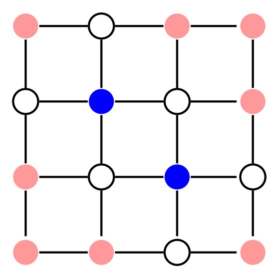

Figure 2.5: Possible robber positions for cops' initial placement at v_6 and v_{11}

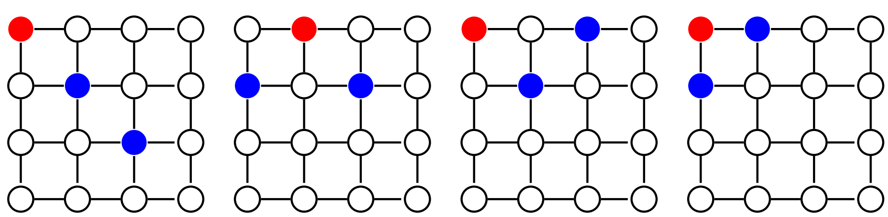

Figure 2.6: Cops' strategy for robber's initial placement on v_1

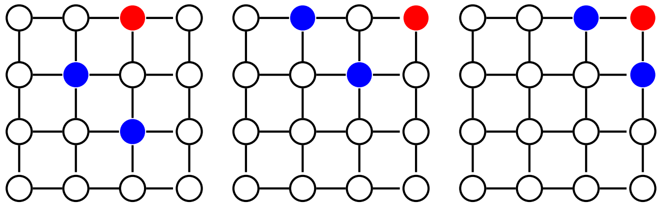

Figure 2.7: Cops' strategy for robber's initial placement on v_3

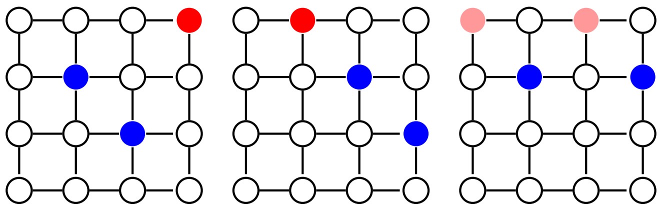

Figure 2.8: Early steps of cops' strategy for robber's initial placement on v_4

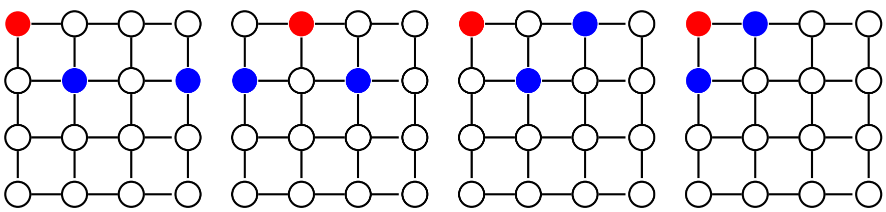

Figure 2.9: First subcase of cops' strategy for robber's initial placement on v_4

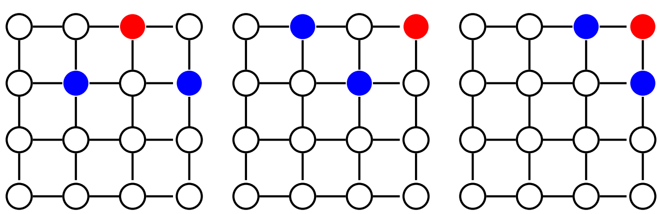

Figure 2.10: Second subcase of cops' strategy for robber's initial placement on v_4

Lemma 12. *The $(v,[2])$-fractional domination numbers for any node $v \in V(P_4 \times P_4)$ are:*

- *if* $v \in core_{4\times4}$, *then* $fdn_{v,[2]}(P_4 \times P_4) = 3$,
- *if* $v \in side_{4\times4}$, *then* $fdn_{v,[2]}(P_4 \times P_4) = 3$,
- *if* $v \in corner_{4\times4}$, *then* $fdn_{v,[2]}(P_4 \times P_4) = 2$.

Proof. Again, by solving the corresponding linear programs for domination of the specific nodes, just like for the 3×3 case. □

The following figure demonstrates the vertices that need to be dominated to eventually capture the robber given her current position. All other cases are symmetrical to one on the figure.

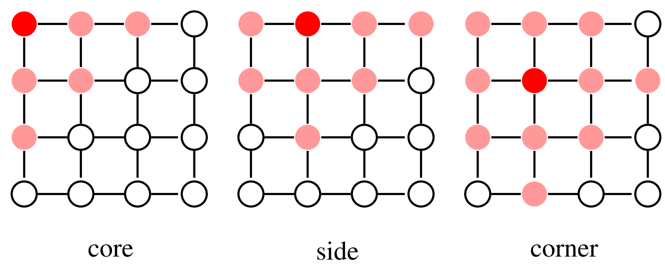

Figure 2.11: Vertices to be dominated for robber positions on the 4×4 grid

Lemma 13. $fcn_2(P_4 \times P_4) = 2$.

Proof. Using the fact that $P_3 \times P_3$ is an isometric subgraph of $P_4 \times P_4$, we derive $2 = fcn_2(P_3 \times P_3) \leq fcn_2(P_4 \times P_4) \leq cn_2(P_4 \times P_4) = 2$. Alternatively, one can again consider the domination number of the 2-neighborhood for any possible position of the robber. Initially, the robber is placed at any undominated vertex. This is possible since $fdn(P_4 \times P_4) = 4$. At each round of the game, the robber may lie on a core, side or corner node. No matter where she lies, at least 2 cops are needed to dominate her 2-neighbourhood. If strictly less than 2 cops were present, there would always be an undominated position for the robber to move to. Consequently, $fcn_2(P_4 \times P_4) \geq 2$. This fact, together with the integral strategy for 2 cops, produce the result. □

We now turn our attention to the 5×5 grid, where we partition the vertices of $P_5 \times P_5$ as follows:

- $corner_{5\times5} = \{\upsilon_1, \upsilon_5, \upsilon_{21}, \upsilon_{25}\}$,
- $next-to-corner_{5\times5} = \{\upsilon_2, \upsilon_4, \upsilon_6, \upsilon_{10}, \upsilon_{16}, \upsilon_{20}, \upsilon_{22}, \upsilon_{24}\}$,
- $side_{5\times5} = \{\upsilon_3, \upsilon_{11}, \upsilon_{15}, \upsilon_{23}\}$,
- $inner-corner_{5\times5} = \{\upsilon_7, \upsilon_9, \upsilon_{17}, \upsilon_{19}\}$,
- $inner-side_{5\times5} = \{\upsilon_8, \upsilon_{12}, \upsilon_{14}, \upsilon_{18}\}$,

- $core_{5\times5} = \{\upsilon_{13}\}$

The following lemma demonstrates the number of cops needed to v-dominate the 2-neighborhood for any possible robber position $v \in V(P_5 \times P_5)$ out of the above 6 disjoint subsets of the vertex set. Let $fdn_{v,s}(G)$ stand for the (v,s)-fractional domination number of G, i.e. the cop quantity needed to dominate all nodes at distance *exactly* s from v. To compute this quantity, one needs to maintain only constraints regarding $N_s(v)$, rather than $N_{[s]}(v)$ in the linear program formulation presented before.

Lemma 14. *The $(v,[2])$-fractional and $(v,2)$-fractional domination numbers for any node $v \in V(P_5 \times P_5)$ are:*

- *if $v \in corner_{5\times5}$, then $fdn_{v,[2]}(P_5 \times P_5) = fdn_{v,2}(P_5 \times P_5) = 2$,*
- *if $v \in next-to-corner_{5\times5}$, then $fdn_{v,[2]}(P_5 \times P_5) = 3$ and $fdn_{v,2}(P_5 \times P_5) = 2$,*
- *if $v \in side_{5\times5}$, then $fdn_{v,[2]}(P_5 \times P_5) = fdn_{v,2}(P_5 \times P_5) = 3$,*
- *if $v \in inner-corner_{5\times5}$, then $fdn_{v,[2]}(P_5 \times P_5) = fdn_{v,2}(P_5 \times P_5) = 3$,*
- *if $v \in inner-side_{5\times5}$, then $fdn_{v,[2]}(P_5 \times P_5) = 4$ and $fdn_{v,2}(P_5 \times P_5) = 3$,*
- *if $v \in core_{5\times5}$, then $fdn_{v,[2]}(P_5 \times P_5) = fdn_{v,2}(P_5 \times P_5) = 4$.*

Proof. By solving the corresponding linear programs for domination of the specific nodes; refer to the linear program earlier in the section. □

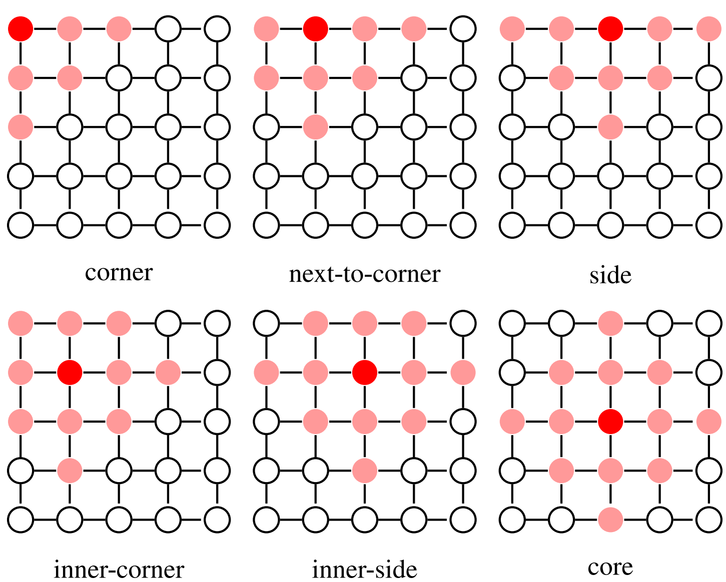

Figure 2.12: Vertices to be dominated for robber positions on the 5×5 grid

Lemma 15. $fcn_2(P_5 \times P_5) \geq 2$.

Proof. Trivially, $2 = fcn_2((P_4 \times P_4) \leq fcn_2(P_5 \times P_5)$. Alternatively, suppose $2-\varepsilon$ (where $\varepsilon > 0$) cops are present. Initially, there is an undominated vertex for the robber to place herself, since $2-\varepsilon < fdn(P_5 \times P_5) = 7$. Then, at each round there is at least one undominated vertex for the robber to move, following the previous lemma. The robber picks the available move and perpetually evades capture no matter how the cops move. $\square$

Lemma 16. $cn_2(P_5 \times P_5) > 2$.

Proof. We suggest an evasion strategy for the robber, against 2 integral cops. We will make use of the partitioning of $P_5 \times P_5$ described earlier.

Initially, place the robber in a vertex $\upsilon \notin corner_{5\times5} \bigcup next-to-corner_{5\times5}$. Notice that such an initial placement is possible since $2 < fdn(V(P_5 \times P_5) \setminus (corner_{5\times5} \bigcup next-to-corner_{5\times5})) = fdn_{\upsilon_{13},[2]}(P_5 \times P_5) = 4$. Now, we provide an evasion strategy for the robber in the form of what move(s) she needs to do to escape capture, when lying in any out of the 6 partition-sets of $V(P_5 \times P_5)$ at any game round:

- Assume that the robber lies on a vertex $\upsilon \in side_{5\times5} \bigcup inner-corner_{5\times5} \bigcup inner-side_{5\times5} \bigcup core_{5\times5}$, just like at the first step. By making use of the 2-neighbourhood domination lemma, there exists an available vertex at distance 2. The robber moves there and escapes capture. Preferably, the robber chooses to move to a vertex $w \notin corner_{5\times5} \bigcup next-to-corner_{5\times5}$ (should such a vertex be available) in order to repeat the same substrategy. If that's not the case, then the robber goes to a (next-to-)corner vertex.

- Suppose that the robber has reached a vertex in $corner_{5\times5}$ after making a 2-move from a vertex in $side_{5\times5}$. This means, that the other 2-moves to a node in $inner-corner_{5\times5} \bigcup core_{5\times5}$ are guarded and that the 2-move towards the opposite corner *may* be guarded. We focus on the case the robber moves from υ_3 to υ_1, as all other cases can be handled symmetrically.

  - Suppose that all the other 2-moves for the robber are guarded, thus the robber moves to υ_1. The best case for cops' proximity towards the robber is that they lie on υ_4 and υ_{17}. In this case, the robber's 2-moves out of υ_1 are already guarded. Notice that any other positioning of the cops is not plausible, since the robber would pick another 2-move out of υ_3. Now, it's the cops' turn. If they both stay put, then the robber stays put as well. If at least one of them increases his distance from the robber, then a move of speed 2 becomes available for the robber in side or inner corner regions, which she picks and escapes. Hence, the cops need to decrease their distance from the robber. If they both move towards the robber (e.g. to υ_3 and υ_{16} or υ_3 and υ_{12}), then again a 2-move becomes available for the robber. So, we are left with the case that one of them moves and the other stays put. Cops being on υ_3 and υ_{17} is the only possible option. In this case, there is a sequence of moves to allow the robber to reach the nearest inner corner and from there she'll follow the inner-corner strategy. Let us refer to this sequence of moves as the *escape-corner* moves. In fact, the robber moves to υ_6. The cops must now place themselves on υ_3 and υ_{16}, otherwise 2-move opportunities open for the robber. The robber moves to υ_7, an inner-corner node and then she makes use of the strategy for such nodes.

  – Suppose that there are two available 2-moves to a corner node for the robber. The robber may pick the one that lies farthest from the cops. In any case, the best possible proximity for the cops is reaching after their step to υ_3 and υ_{17} (should the robber move to υ_1), since for any other cop positioning, there exists a 2-move for the robber to escape the situation. Now, the robber need only follow the same escape-corner sequence of moves.

- Suppose the robber reaches a corner node (say υ_1), following a move of speed 2 from the corresponding inner-corner node. In this case, the cops' need to be on nodes υ_8 and υ_{12}, otherwise there would be an available robber move of speed 2 to a non-corner node. It's the cops turn now. Witness, that if at least one of them increases his distance from the robber, then a 2-move becomes available for her. The same goes if they both choose to decrease their distance from the robber. In case they stay put, then the robber stays put, too. In the pursuit of capturing the robber their only option is that exactly one of them moves closer to the robber. This means that the cops place themselves on either υ_3 and υ_{17} or υ_8 and υ_{16}. The two cases are symmetrical. The robber may now escape to the closest inner corner node again by following the escape-corner strategy.

- Finally, let us assume that the robber reaches a next-to-corner node leaving from an inner-side node. The best proximity case for the cops now is υ_9 and υ_{13} or υ_3 and υ_{13}. They move to either υ_3 and υ_{12} or υ_8 and υ_{17}. To evade, the robber need only move to the corner and repeat the aforementioned escape-corner moves in order to later reside on υ_7, an inner-corner node.

□

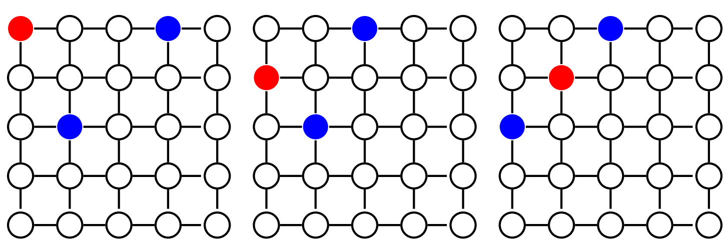

Figure 2.13: The *escape-corner* strategy for the robber

Lemma 17. $cn_2(P_5 \times P_5) \leq 3$.

Proof. We demonstrate a specific strategy for 3 cops to capture the robber whatever her moves. Initially, the 3 cops are placed on υ_7, υ_{13} and υ_{19}. The robber may initially place herself on any node out of v_1, v_3, v_4, v_5, v_9, v_{10}, v_{11}, v_{15}, v_{16}, v_{17}, v_{21}, v_{22}, v_{23} and v_{25}. Notice that, for symmetry reasons we can immediately discard v_{11}, v_{16}, v_{17}, v_{21}, v_{22}, v_{23} and v_{25}. Furthermore, a cop-strategy for the robber commencing on v_{10} would be symmetrical to a strategy for the robber commencing on v_4 and so v_{10} can safely be ignored. In the same spirit, v_{15} can be excluded from our analysis as well due to it being symmetrical to v_3. That is, we shall now focus on the strategy that the 3 cops have to follow in case the robber starts her effort to escape from any node out of v_1, v_3, v_4, v_5 or v_9.

- Assume that the robber initially places herself on v_1. In this case, the 2 cops initially placed on v_7 and v_{13} suffice to capture the robber. They move to v_6 and v_8, respectively. At this point, there does exist only one undominated position for the robber to move herself, that being v_2. The cops now move towards v_7 and v_3. Again, if the robber wishes to evade them, then she must move to her sole option, which is v_1. Now, the cops only have to move leftwards (v_6 and v_2) to accomplish trapping the robber in the corner. The robber can do nothing but stay still. Then, at least 1 cop moves to v_1. The cops win.

- Assume that the robber initially places herself on v_3. The 3 cops now move upwards to v_2, v_8 and v_{14}, respectively. The robber may now move either to v_4 or to v_5.

  - Suppose she chooses a move to v_4. The cops move to the right on v_3, v_9 and v_{15}. Now, the only available option for the robber is to move on v_5. 2 out of the 3 cops move to v_4 and v_{10} and trap the robber in the corner. The cops win in at most 1 step.

  - Suppose she chooses a move to v_5. The cops move to the right on v_3, v_9 and v_{15}. Unfortunately for the robber, her only available move reduces in staying on v_5. The cops move such that they lie on v_4 and v_{10}. The robber is trapped in the corner. The cops win in at most 1 step.

- The robber initially places herself on v_4. The cops move upwards to v_2, v_8 and v_{14}. The robber may now move v_5 or v_{10} or just stay on v_4. In any case, the cops move to the right to v_3, v_9 and v_{15}. Eventually, the robber is forced to move to v_5, then the cops trap her in the corner and they win in at most 1 more step.

- The robber initially places herself on v_5. The cops move rightwards to v_8, v_{14} and v_{20}. The robber may now remain on v_5 or move towards v_4 or v_{10}. In any case, the cops move upwards to v_3, v_9 and v_{15}. The robber is forced to move to v_5. In the next step, the cops will trap her in the corner and finally win the game.

- Finally, suppose that the robber initially places herself on v_9. The cops repeat the same strategy just like in the previous case. Their former step is to move rightwards and the latter to move upwards. One can easily notice that the robber has no evasion strategy against these cops' moves. Eventually, she resorts to the corner. Then, the cops capture her and win.

□

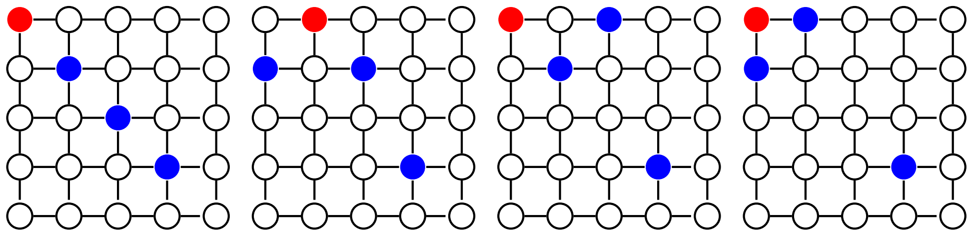

Figure 2.14: Cops' strategy for robber's initial placement on v_1

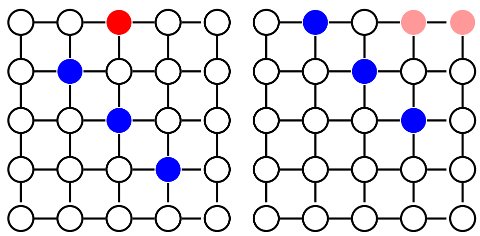

Figure 2.15: Early steps of cops' strategy for robber's initial placement on v_3

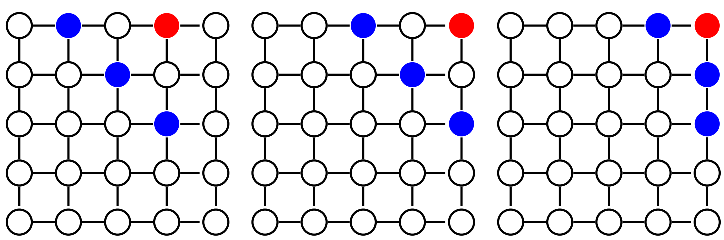

Figure 2.16: First subcase of cops' strategy for robber's initial placement on v_3

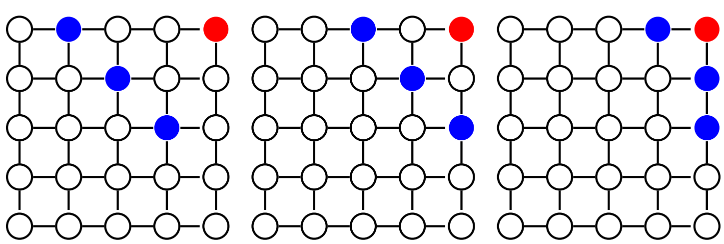

Figure 2.17: Second subcase of cops' strategy for robber's initial placement on v_3

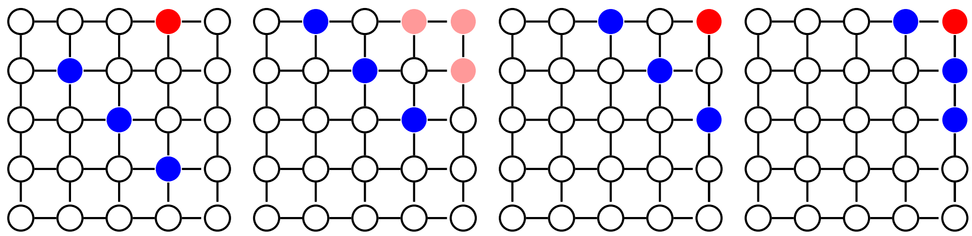

Figure 2.18: Cops' strategy for robber's initial placement on υ_4

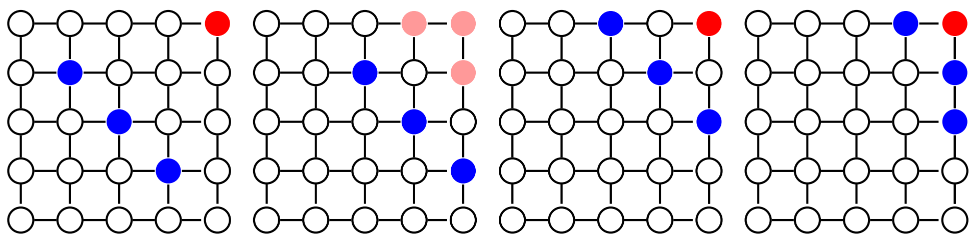

Figure 2.19: Cops' strategy for robber's initial placement on υ_5

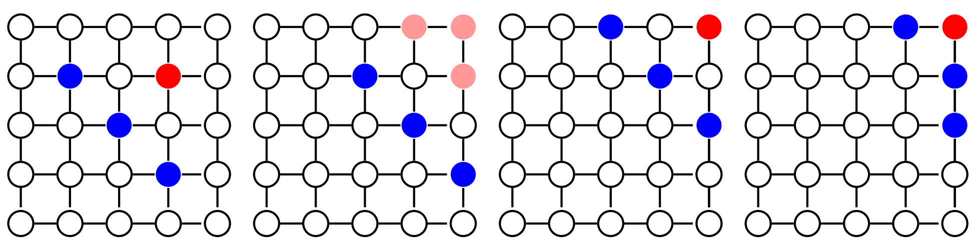

Figure 2.20: Cops' strategy for robber's initial placement on υ_9

Theorem 3. $cn_2(P_5 \times P_5) = 3$.

Proof. Directly follows from the two previous lemmata. □

Notice that a similar partitioning of the vertices can be made for any $n \times n$ grid where $n \geq 5$. The $side_{n\times n}$, $inner - side_{n\times n}$ and $core_{n\times n}$ sets are expanded to contain all such (equivalent under positioning) nodes. Using this approach, we comprehend that in a big grid, given strictly less than 4 cops, the robber can move from turn to turn in such a way that she always remains on core nodes. The following corollary already provides a small lower bound for the fractional cop number in case of speed 2.

Corollary 4. *For n big enough:* $fcn_2(P_n \times P_n) \geq 4$.

2.4 Open Questions

Starting with the fractional case, we would like to know whether there exists a relaxed variant of *Cops & Robber* that can provide an approximation for the cop number. For example, is there a way to compute fcn_a? Moreover, what is the complexity for such a computation? Does it stay within polynomial bounds?

Moving to the fast robber on the grid case, the basic question remains: what is the number of cops needed to capture the fast robber on the $n \times n$ grid? Can we find an escape strategy for the robber where she evades e.g. $\sqrt{n}$ cops? In such case, one needs to describe an "always stay far enough" strategy for the robber. Maybe some intuition from the robber strategies proposed in [18] or recently in [5] could arise.

Overall in this area, the big question that remains open is the Meyniel $\sqrt{n}$-conjecture. Even an upper bound of $n^{1-\varepsilon}$ would be a significant result in this direction. Several other questions arise in the area; one need only investigate the vast literature present today.

Chapter 3

Surfing & Marking

3.1 The Game

3.1.1 Description and Motivation

The *Surveillance* game is a combinatorial game played on a (directed) graph G. There are two players: the *marker* (otherwise called *observer*) and the *surfer* (otherwise called *fugitive*), who take turns alternately. In trivial round 0, the marker marks a predefined vertex v_0 and the surfer places herself on v_0. Then, for all following rounds, the marker marks a selection of at most $k \in \mathbb{N}$ unmarked nodes, while the surfer chooses a neighboring node to move herself to. The marker wins if he manages to mark all vertices of G before the surfer manages to reach an unmarked vertex, in which case she wins. Both players try to develop optimal strategies to ensure victory. Notice that the duration of the game is at most $\lceil (n-1)/k \rceil$ rounds, since the marker marks at most k nodes per round.

3.1.2 Background

Correspondingly to the cop number, the quantity under consideration is the *surveillance number*, defined likewise.

Definition 3. *The surveillance number of a graph G, with respect to vertex $v_0 \in V(G)$, is the minimum number of marks needed such that the marker wins against any surfer strategy for a Surveillance game starting at v_0 and it is denoted $sn(G, v_0)$.*

The game was introduced quite recently by Fomin et al. [15] as a modelization for Web pages' prefetching, where a browser may download potential web pages in advance in order to enhance the user's surfing. They extensively study the computational complexity of the problem of determining $sn(G, v_0)$: they prove PSPACE-completeness and several NP-hardness results. Moreover, the demonstrate polynomial-time algorithms in the case of interval graphs and trees. For trees, a combinatorial characterization is provided as well. Finally, the connected variant is considered as well, where there exists the restriction that, at any round, the subset of nodes marked must be connected. The problem in question is to define the cost of connectivity: how many more marks per round are needed to satisfy the connected variant's restriction. Further attempts on this issue appear on [20].

3.2 Bounded Degree Complexity

An open question in [15] suggested the study of the complexity of Surveillance, when the game is played on graphs whose maximum degree is bounded. Notice that computing $sn(G, v_0)$ is trivial for graphs of maximum degree 3 [15]. The result presented in this section acts as initial step in estimating the surveillance number complexity for graphs whose degree is bounded by a constant bigger than 3. The rest of the section is dedicated to the proof of the following theorem.

Theorem 4. *Deciding whether $sn(G, v_0) \leq 2$, for a directed acyclic graph G of maximum degree* 6 *and a starting vertex $v_0 \in V(G)$, is NP-hard.*

Below, we may refer to the decision problem in question as the *Surveillance Number* problem. To prove NP-hardness, a reduction from a special case of the well-known *Vertex Cover* problem is employed.

Definition 4. *A graph is called cubic, if every node has degree exactly* 3*.*

Definition 5. Vertex Cover for Cubic Graphs: *Given a cubic graph G and a constant k, decide whether there exists a set $V'(G) \subseteq V(G)$ such that for any $e = \{\upsilon_i, \upsilon_j\} \in E(G)$: $\upsilon_i \in V'(G) \vee \upsilon_j \in V'(G)$ and $|V'(G)| \leq k$.*

NP-hardness for the above problem is proved in [19]. From now on, we shall refer to the problem shortly as VC-3.

The reduction

Given an instance (G, k) of VC-3, we transform it into an instance (G', v_0) of the *Surveillance Number* problem as follows: Let $n = |V(G)|$ and $m = |E(G)|$. For each vertex of $\upsilon_i \in V(G)$, we put a corresponding node $u_i \in V(G')$. That is, n new nodes are added, namely $u_1, u_2, \ldots, u_n$. Then, for each edge $e_i \in E(G)$, we put a corresponding node $c_i \in V(G')$. That is, m new nodes are added, namely $c_1, c_2, \ldots, c_m$. Another $k+m-1$ nodes are added to $V(G')$, namely $v_0, v_1, \ldots, v_{k+m-2}$. As far as the edge set is concerned, $v_0, v_1, \ldots, v_{k+m-2}, c_1, c_2, \ldots, c_m$ form a directed path in this order in G'. Moreover, each vertex $c_i \in V(G')$ is connected to $u_j, u_k \in V(G')$, where $e_i = \{\upsilon_j, \upsilon_k\} \in E(G)$, via a specific gadget. For each vertex c_i, 3 more vertices are added, namely $c_i^{left}, c_i^{right}, c_i^{mid}$. Then, the following directed edges are added to the edge set of G': (c_i, c_i^{left}), (c_i, c_i^{right}), (c_i^{left}, c_i^{mid}), (c_i^{right}, c_i^{mid}), (c_i^{left}, u_j), (c_i^{left}, u_k), (c_i^{right}, u_j) and (c_i^{right}, u_k). Notice that since G is cubic, each node $u_i \in V(G')$ receives 6 edges, 2 from each corresponding edge's patch. The construction is complete. Furthermore, it takes polynomial time, since $|V(G')| = |V(G)| + 5|E(G)| + k - 1$ and $|E(G')| = 10|E(G)| + k - 2$.

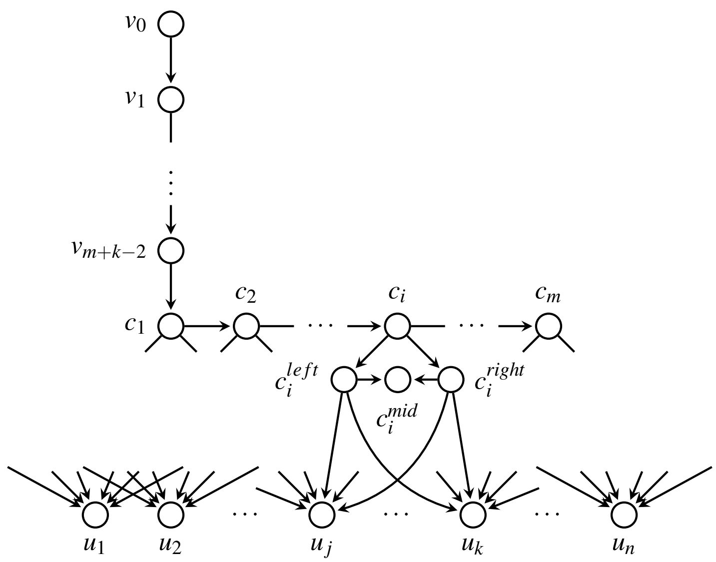

Figure 3.1: The graph G' constructed for the reduction

Lemma 18. *If G has a vertex cover of size at most k, then* $sn(G', v_0) \leq 2$.

Proof. If there exists a vertex cover of size at most k for G, then we show how the marker can win in G' by using 2 marks per round. Initially, the marker trivially marks v_0. When the surfer moves to any node $v_i \in \{v_0, v_1, \ldots, v_{k+m-2}\}$, the marker's strategy is to mark v_{i+1} and an unmarked node in $\{c_i : 0 \leq i \leq m\} \cup \{u_i : 0 \leq i \leq n\}$. Finally, the surfer arrives on v_{k+m-2} and the marker put his two marks on c_1 and another unmarked node in $\{c_i : 0 \leq i \leq m\} \cup \{u_i : 0 \leq i \leq n\}$. Notice that $k+m$ nodes are marked out of this union, since $k+m-1$ marks are used in total and the extra 1 comes from marking c_1 as the next node of v_{k+m-2}. The nodes that the marker chooses to mark until this stage of the game are exactly all the c_i nodes plus the k u_i nodes that correspond to the vertex cover solution. Now, the surfer moves onto c_1. The marker's strategy is to mark c_1^{left} and c_i^{right}, thus covering the only unmarked neighbors. If the surfer decides to move towards c_m for some steps, then the strategy of the marker remains the same. That is, after the surfer moves on c_i, the marker marks c_i^{left} and c_i^{right}. It is clear that the surfer will never reach a node c_i, whose unmarked neighborhood is greater than 2. Hence, at some point, the surfer will move to a node c_j^{left} or c_j^{right} for some j. Without loss of generality, assume that she moves onto c_j^{left}. Now, notice that, due to the existence of the k-cover, at least one node out of u_j, u_k, where $e_j = \{\upsilon_k, \upsilon_j\} \in E(G)$, is already marked. Thus, there are at most 2 unmarked neighbors of c_j^{left}, one of them being c_j^{mid}. The marker marks them and eventually wins the game, since there is no other escape possibility for the surfer. □

Lemma 19. *If G has a vertex cover of size greater than k, then* $sn(G', v_0) > 2$.

Proof. If the vertex cover of G is of size greater than k, we provide a winning strategy for the surfer against a 2-marker in G'. The surfer follows the path from v_0 to v_{k+m-2}. We assume that the marker has so forth protected the path, otherwise the surfer trivially wins. After the surfer places herself on v_{k+m-2}, the marker uses his 2 marks and at this point he has marked in total $k+m$ nodes in $V(G') \setminus \{v_i : 0 \leq i \leq k+m-2\}$. Now, in order to examine the worst possible case for the surfer, assume that there is a vertex cover of size $k+1$ for G. From now on, the surfer moves on c_1 and then on the c_i path. The surfer trivially wins if at some early point the marker does not take care to protect the surfer's neighborhood. Otherwise, the surfer's victory falls under one of the following cases:

- At any point of her traversal, if the marker decides to mark strictly greater than k nodes out of the c_i^{mid}, u_i families, then the surfer just moves upwards onto the c_i path and will eventually arrive to an unmarked node: the marker has to mark all the unmarked nodes out of $c_i, c_i^{left}, c_i^{right}$ (where $i > 1$) and c_1^{left} and c_1^{right} in m rounds (round $m+k$ to round $2m+k-1$). The number of unmarked nodes is at least $3(m-1)+2-(m-2) = 2m+1$, since more than k marks out of the initial $k+m-1$ have been assigned elsewhere. The marker can only mark up to $2m$ of these nodes in m rounds and hence at some point (near the end of the c_i path if the marker produces a decent strategy) there will be an unmarked node neighboring to the surfer's position after the marker's turn.

- On the contrary, suppose that the marker decides to assign at most k marks to nodes in the c_i^{mid}, u_i families. In this case, the counting argument made above does not hold. Nonetheless, the surfer now attacks the cover to escape: for any selection of k nodes u_i, there is an uncovered node c_j. The surfer now follows the path and reaches c_j. Without loss of generality, he then moves towards c_j^{left}. At this time, there are 3 uncovered neighbors of c_j^{left}. The marker marks at most 2 of them and the surfer wins. Notice that if the marker changes his mind and protects the cover when the surfer reaches c_j, then the surfer's strategy and winning reasoning reduces to the previous case. To conclude, it is crucial to observe that in this setting, marking a node c_w^{mid} for some index w is equivalent to marking a node u_k or u_j, where $e_w = \{\upsilon_k, \upsilon_j\} \in E(G)$. This happens because in order for the surfer to win when she reaches either c_w^{left} or c_w^{right}, there must exist 3 unmarked neighbors after the marker's turn. Thus, instead of marking c_w^{mid}, one could mark either of u_j, u_k and the outcome would yet remain unchanged.

□

3.3 Onto the Benefit-Deficit Extension

In this section we consider an extension to the Surveillance Game originally proposed in [35]. The purpose is to increase the understanding over Surveillance and try to establish and correlate results in both contexts.

3.3.1 Game Definition

Description & Intuition

We consider a marking game played on a connected graph. There are two players: the *marker* and the *surfer*, who take turns alternately. Initially, the marker trivially marks a pre-defined vertex and then the surfer commences her walk on the graph from there. At each round of the game, the marker has the opportunity to put some marks on a subset of the vertices of the graph and then the surfer chooses a neighboring vertex to move from her current position. By the time the surfer moves to a certain node, a node-specific quantity of marks (the *benefit*) becomes available for the marker to use only at his turn in the same round for the marking of vertices. Furthermore, each node is related to a node-specific limit of marks (the *deficit*) that it needs to maintain, before the surfer arrives to it. The winning condition for the surfer is to manage to arrive at a node, where this limit is not yet covered by the marker. On the other hand, the marker wins if he makes sure that at each round, the surfer can only move to vertices whose deficit is already covered. Notice that the surfer needs to move at every round, in order for the game to progress. We assume that if the surfer does not move for a certain amount of rounds, then the marker wins. We also assume that the marker does not put more marks to a vertex than the vertex's deficit number and that the surfer does not revisit any node, since that would only provide additional benefit for the marker. The formal definition of the game is given in the following subsection.

Formal Definition

Let G be a connected graph and $u_0 \in V(G)$. Let $b : V(G) \rightarrow \mathbb{N}^*$ be a function that assigns at each node $\upsilon \in V(G)$ a benefit $b(\upsilon) \geq 0$. Let $d : V(G) \rightarrow \mathbb{N}^*$ be a function that assigns at each node $\upsilon \in V(G)$ a deficit $d(\upsilon) \geq 0$. Furthermore, we set $d(\upsilon_0) = 0$. Let $m(\upsilon)$ stand for the amount of marks currently on node $\upsilon \in V(G)$. That is, $0 \leq m(\upsilon) \leq d(\upsilon)$, $\forall \upsilon \in V(G)$ and it is updated dynamically during the game. Let $m_i(\upsilon)$ stand for the amount of marks vertex υ receives at round $i \geq 0$. Obviously, $m(\upsilon) = \sum_i m_i(\upsilon)$. Let C stand for the set of nodes $\upsilon \in V(G)$ for which $m(\upsilon) < d(\upsilon)$. Initially, let $C = V(G)$, since $m(\upsilon) = 0 \; \forall \upsilon \in V(G)$. Then, C is updated dynamically at each round. Let S stand for the set of the vertices visited by the surfer; initially $S = \emptyset$ and then it is dynamically updated at every round. Finally, let $\overline{S} = V(G) \setminus S$.

The *Benefit & Deficit* game proceeds as follows:

1. The marker trivially marks υ_0.
2. The surfer moves to υ_0.
3. Let $S = S \bigcup \{\upsilon_0\}$.
4. The marker puts at most $b(\upsilon_0) = \sum_{\upsilon \in C} m_0(\upsilon)$ marks on the vertices of C.

5. For each $\upsilon \in C$:
   Let $m(\upsilon) = m(\upsilon) + m_0(\upsilon)$.
   If $m(\upsilon) = d(\upsilon)$, then let $C = C \setminus \{\upsilon\}$.

6. Let $c = \upsilon_0$ and $i = 1$.

7. Repeat:
   The surfer moves to $v \in N(c) \bigcap \overline{S}$.
   Let $S = S \cup \{v\}$.
   If $m(v) < d(v)$, then THE SURFER WINS.
   The marker puts at most $b(v) = \sum_{\upsilon \in C} m_i(\upsilon)$ marks on the vertices of C.
   For each $\upsilon \in C$:
   Let $m(\upsilon) = m(\upsilon) + m_i(\upsilon)$.
   If $m(\upsilon) = d(\upsilon)$, then let $C = C \setminus \{\upsilon\}$.
   Let $i = i + 1$.
   Let $c = v$.
   Until $C = \emptyset$.

8. THE MARKER WINS.

Notice that the game trivially ends if $b(\upsilon_0) < \sum_{\upsilon \in N(\upsilon_0)} d(\upsilon)$. In contrast, an upper bound on the number of rounds of the game is given by $\left\lceil \frac{\sum_{\upsilon \in V(G)} d(\upsilon)}{\min_{\upsilon \in V(G)} b(\upsilon)} \right\rceil$.

It comes to our attention that the special case of the Benefit & Deficit game where $b(\upsilon) = k \in \mathbb{N}$ and $d(\upsilon) = 1\ \forall \upsilon \in V(G)$ is exactly the Surveillance Game [15].

Variants

One may consider several possible extensions for this game. In an attempt to reinforce the marker, we may consider that he is allowed to use extra marks out of a pool of $M \in \mathbb{N}$ initial marks in any possible way during the game. Otherwise, we could assign such M additional marks at each round, but this case trivially reduces to the transformation $b(\upsilon) = b(\upsilon) + M$, $\forall \upsilon \in V(G)$. Moreover, another variant could be the switching between the laying of the marks and the robber's winning condition checking. That is, when the robber arrives at a node $\upsilon \in V(G)$, the marker deposits his available marks before the surfer questions whether $m(\upsilon) < d(\upsilon)$. Moving on, a rule restricting the power of the marker would be that, at each round and after the robber's arrival on υ, the marker must mark exactly $b(\upsilon)$ nodes, i.e. one node can only receive at most 1 mark at each round. Finally, one may examine the case when $\bigcup_{\upsilon \in V(G): m(\upsilon) > 0} \upsilon$ is required to be connected throughout the whole duration of the game. Such a restriction implies the definition of the connected variant of the game.

3.3.2 Some Contributions

We provide a combinatorial characterization for the case of trees with respect to the notions of benefit and deficit. In the result below, let $b(S) = \sum_{s \in S} b(s)$ (respectively for $d(S)$) for any set S and $N[S]$ represent the closed neighborhood of S. Furthermore, let T^{υ} stand for the subtree of T hanging under and including node υ. Let $l(S) = d(S) - m(S) \geq 0$ denote the number of marks still uncovered on nodes of S.

Theorem 5. *For any tree T and $\upsilon_0 \in V(T)$: The marker wins if and only if $b(S) \geq d(N[S])\ \forall\, S \subseteq V(T)$, where S is an induced subtree of T containing υ_0.*

Proof. For the *if* direction, we give a winning strategy for the marker. Notice that in her quest to escape, the surfer shall follow an induced path $\upsilon_0, \upsilon_1, \ldots, \upsilon_{final}$ on T. This remark is a direct consequence of the assumption that the surfer may not visit any node twice. Notice that even the non-existence of this assumption would not harm the reasoning to follow. Let $S_i = \{\upsilon_j : 0 \leq j \leq i\}$ be the set of vertices that the surfer has so far followed in the first i rounds. Initially, $b(S_0) = b(\upsilon_0) \geq d(N[\upsilon_0]) = d(N[S_0])$ and so the marker cannot lose in this round. Were $b(\upsilon_0) > d(\upsilon_0)$, we would assign the extra $b(\upsilon_0) - d(\upsilon_0)$ marks to nodes in $\bigcup_{v_i \in N(\upsilon_0)} V(T^{v_i})$ in such a way that $b(\upsilon_i) \geq l(\upsilon_i)$ when the surfer arrives at any node υ_i. Respectively, we do so for any step of the surfer. Notice that there must exist such a marking strategy otherwise the fact that $b(S) \geq d(N[S])\ \forall\, S \subseteq V(T)$ induced subtree containing υ_0 holds is contradicted: Suppose that the surfer lies on υ_i and after the marker's turn, there is a node $v \in N(\upsilon_i)$ such that $l(v) > 0$. Indeed, this would mean that $b(\{\upsilon_0, \upsilon_1, \ldots, \upsilon_i\}) < d(N[\{\upsilon_0, \upsilon_1, \ldots, \upsilon_i\}])$, otherwise the marker could have covered all the deficit. The marker makes use of such a strategy at any round. Hence, no matter which course the surfer may follow, the strategy of the marker ensures the early marking of all the necessary nodes.

For the *only if* direction, we demonstrate the contrapositive. Suppose that $\exists S'$: $b(S') < d(N[S'])$. We propose an escape strategy for the surfer where she starts from S' and then moves in always smaller and smaller sets that maintain the same property. Initially, the surfer is placed on υ_0. Let $S'_i = V(T^{\upsilon_i}) \bigcap S'\ \forall\ \upsilon_i \in N(\upsilon_0)$. That is, $\bigcup_i S'_i \bigcup \{\upsilon_0\} = S'$. Suppose $b(\upsilon_0) \geq d(N[\upsilon_0]) = l(N[\upsilon_0])$, otherwise the surfer trivially escapes. We notice that $b(S') = \sum_i b(S'_i) + b(\upsilon_0)$. Moreover, $l(N[S']) = \sum_i l(N[S'_i] \setminus \{\upsilon_i\}) + l(N[\upsilon_0])$, since $l(\upsilon_0) = 0$. It follows $\sum_i b(S'_i) + b(\upsilon_0) < \sum_i l(N[S'_i] \setminus \{\upsilon_i\}) + l(N[\upsilon_0])$. Then, $\sum_i b(S'_i) < \sum_i l(N[S'_i] \setminus \{\upsilon_i\}) - (b(\upsilon_0) - l(N[\upsilon_0]))$. Now, the marker places his marks on some nodes covering at least the uncovered marks in $N[\upsilon_0]$. Let l_1 (l_2) stand for the uncovered mark quantity before (after) the marker's turn. The quantities $l(\cdot)$ are now updated and notice that $\sum_i l_1(N[S'_i] \setminus \{\upsilon_i\}) - (b(\upsilon_0) - l_1(N[\upsilon_0])) \leq \sum_i l_2(N[S'_i] \setminus \{\upsilon_i\}) = \sum_i l_2(N[S'_i])$, since $l_2(\upsilon_i) = 0\ \ \forall \upsilon_i \in N[\upsilon_0]$ and at most $b(\upsilon_0) - l_1(N[\upsilon_0])$ marks can be used on nodes of $\bigcup_i (N[S'_i] \setminus \{\upsilon_i\})$. Since now $\sum_i b(S'_i) < \sum_i l(N[S'_i])$, then $\exists S'_j : b(S'_j) < l(N[S'_j])$. Furthermore, notice that $|S'_j| < |S'|$. The surfer moves to υ_j, for which $S'_j = V(T^{\upsilon_j}) \bigcap S'$. Inductively, suppose that after k steps, the surfer now lies on υ^k and for the set $S^k = V(T^{\upsilon^k}) \bigcap S$ it holds $b(S^k) < l(N[S^k])$. Let $S^k_i = T^{\upsilon^k_i} \bigcap S^k\ \forall\ \upsilon^k_i \in N(\upsilon^k) \bigcap V(T^{\upsilon^k})$. Assume $b(\upsilon^k) \geq l(N[\upsilon^k])$, otherwise the surfer trivially wins. We notice that $b(S^k) = \sum_i b(S^k_i) + b(\upsilon^k)$. Moreover, $l(N[S^k]) = \sum_i l(N[S^k_i] \setminus \{\upsilon^k_i\}) + l(N[\upsilon^k])$, since $l(\upsilon^k) = 0$. It follows $\sum_i b(S^k_i) + b(\upsilon^k) < \sum_i l(N[S^k_i] \setminus \{\upsilon^k_i\}) + l(N[\upsilon^k])$. Then, $\sum_i b(S^k_i) < \sum_i l(N[S^k_i] \setminus \{\upsilon^k_i\}) - (b(\upsilon^k) - l(N[\upsilon^k]))$. Again, the marker places his marks on some nodes covering at least the uncovered marks in $N[\upsilon^k]$. The quantities $l(\cdot)$ are updated and $\sum_i l_1(N[S^k_i] \setminus \{\upsilon^k_i\}) - (b(\upsilon^k) - l_1(N[\upsilon^k])) \leq \sum_i l_2(N[S^k_i] \setminus \{\upsilon^k_i\}) = \sum_i l_2(N[S^k_i])$, since $l_2(\upsilon^k_i) = 0\ \ \forall\ \upsilon^k_i \in N[\upsilon^k]$ and at most $b(\upsilon^k) - l_1(N[\upsilon^k])$ marks can be used on nodes of $\bigcup_i (N[S^k_i] \setminus \{\upsilon^k_i\})$. That is, $\exists S^k_j : b(S^k_j) < l(N[S^k_j])$ and

$|S_j^k| < |S^k|$. The surfer now moves to υ_j^k for which $S_j^k = V(T^{\upsilon_j^k}) \bigcap S^k$. Eventually, the surfer will reach a set S^{final}, where $b(S^{final}) < l(N[S^{final}])$ and $|S^{final}| = 1$. The benefit does not suffice for the marker to cover all $l(N[S^{final}])$ unmarked parts. The marker places his last marks. The surfer now moves to a neighboring vertex υ where $m(\upsilon) < d(\upsilon)$ and wins. □

3.4 Open Questions

For the Surveillance game, one could examine what happens when the degree is even smaller than 6. For instance, when $\Delta = 4$ or $\Delta = 5$. The cost of connectivity remains open, while the best lower and upper bound can be found in [20]. From the complexity point of view, it would be interesting to search for an approximation or an inapproximability result about $sn(G, v_0)$. Perhaps a relation to *Set Cover* could be established, but this insight needs to be investigated more thoroughly.

Regarding the Benefit-Deficit extension, one could try to test similar questions as this context could provide assistance in answering questions on Surveillance. The following is a possible definition for the optimization problem.

Definition 6. *Given a graph G, a starting vertex v_0, two vectors $\vec{b}$ and $\vec{d}$ of size $|V(G)|$ that maintain the benefit and deficit quantities for each node, respectively, and an integer k, decide whether the marker wins in a $(G, v_0, k\vec{b}, \vec{d})$-Benefit-Deficit game.*

Chapter 4

Eternal Security

4.1 The Game

4.1.1 Description

Eternal Domination can be regarded as a combinatorial graph game. There exist two players: one of them controls the *guards*, while the other one controls the *rioter*. Initially, the guard tokens are placed such that they are a dominating set on G. Then, the rioter attacks a token without any guard on it. A guard, that dominates the attacked vertex, must now move on it to counter the attack ensuring that the modified guard positioning remains dominating. The game proceeds in similar fashion in any subsequent rounds. The guards win this game if they can counter any attack of the rioter and perpetually maintain a dominating set. The rioter wins if she manages to force the guards to reach a positioning that is no longer dominating. Finally, notice that more than one guards may lie on the same node.

4.1.2 Background

The idea about infiinite order domination was originally considered in [9], as an extension to other domination variants. Later, Goddard et al. [23] focused on the formalisms, which we too follow. More specifically, they consider two variants of the game. In the former, only one guard is allowed to move in each guards' turn, while, in the latter, at most m guards can move in each guards' turn, where $m \in \{2, 3, \dots, n\}$. Below, we define the two corresponding optimization parameters.

Definition 7. *The minimum number of guards, needed to perpetually ensure domination in a graph G against any rioter strategy, is called:*

- *the eternal* 1*-domination number of G and is denoted by $\sigma_1(G)$, if only one guard is allowed to move at each round*
- *the eternal m-domination number of G and is denoted by $\sigma_m(G)$, if at most m guards are allowed to move at each round*

Several bounds are obtained in [23], e.g. $\gamma(G) \leq \sigma_n(G) \leq \alpha(G) \leq \sigma_1(G) \leq \theta(G)$. Many other papers have been published under this setting; see [3, 22, 26, 27]. Finally, there exists a corresponding *Eternal Vertex Cover* game, where the interest is in protecting edges and perpetually maintaining a vertex cover; [28, 14].

4.2 Complexity Issues

We consider the computational complexity of the following decision problem: *Given a graph G and a positive integer k as input, decide whether* $\sigma_{1(m)}(G) \leq k$. As in many combinatorial games, the complexity remains unclarified. The minimal complexity class known to include the problem is EXPTIME. Below, we try to investigate complexity issues of Eternal Domination.

4.2.1 σ_m Hardness & Approximation

To prove NP-hardness for computing $\sigma_m(G)$, we provide a reduction from the NP-hard MINIMUM DOMINATING SET problem: *Given a general graph G and a positive integer k, decide whether there exists a dominating set of size at most k in G.*

The reduction

Given a graph G, we construct a new graph G' as follows: for any vertex $v \in V(G)$, v remains in $V(G')$ and a true twin v' is added as well. Then another vertex, namely u, is added and all edges $\{u, v'\}$ are added. That concludes the construction. Notice that $|V(G')| = 2|V(G)| + 1 = \mathscr{O}(|V(G)|)$ and $|E(G')| = 2|E(G)| + 2|V(G)| = \mathscr{O}(|V(G)|^2)$.

Lemma 20. *If G has a dominating set of size at most k, then* $\sigma_4(G')$ *is at most* $k+2$.

Proof. Given a solution for dominating G with $\rho \leq k$ nodes, we assign ρ guards on the same nodes of G' and another 2 guards on u. Notice that this initial positioning is a dominating set for G'. Now, we show a strategy for these $\rho + 2$ guards to win against any possible rioter attack:

- Suppose an attack happens on a node v' (respectively v) and there exists a guard on v (respectively v'). Then the guard simply follows the transition $v' \to v$ (respectively $v \to v'$) and protects the node together with retaining domination.

- Suppose an attack happens to an original node v and no guards lie on v'. Since there is domination, there exists a guard on a node $g \in N(v) \setminus \{v', u\}$ that dominates v. Then, we have the following guard transitions: $g \to v$ and $u \to g'$. In this way, the attack on v is countered and domination is retained.

  - Now suppose an attack happens on another node, say $v_1 \in N(g) \setminus \{g'\}$. Then, the guards move $g' \to v_1$ and $v \to g'$ and counter it.
  - If there is an attack on g, then the guards simply move $v \to g$ and $g' \to u$.

– Finally, if there is an attack on another node far away, say v_2 guarded by g_1 (otherwise if guarded by v'_2 follow the case before and thus restore initial placement). Then, the guard strategy is to move $v \to g$, $g' \to u$, $g_1 \to v_2$ and $u \to g'_1$.

And this strategy is repeated perpetually against any possible series of attacks. □

Lemma 21. *The domination number of G is at most the domination number of G'.*

Proof. Given a guard placement that dominates G', we transform it in a way that it dominates G using the same amount of guards: For any guard lying on a vertex v or v', where $v, v' \in V(G') \setminus \{u\}$, we put a guard on the corresponding vertex $v \in V(G)$. Now notice that a guard in either v or v' dominates both original and twin nodes lying in $N[v]$ and the union of all these neighborhoods is exactly $V(G')$. Equivalently, guards on the corresponding vertices $v \in V(G)$ dominate the corresponding set of neighborhoods $\{N[v]\}$, whose union is exactly $V(G)$. □

Theorem 6. *Given a general graph G and an integer k (as part of the input), deciding whether $\sigma_4(G) \leq k$ is NP-hard.*

Proof. By the previous two lemmata, $\gamma(G) \leq \gamma(G') \leq \sigma_4(G') \leq \gamma(G) + 2$. Given the well known $\mathcal{O}(log n)$-inapproximability result for computing $\gamma(G)$ [38], NP-hardness follows. □

Definition 8. Set Cover*: Given a universe $\mathcal{U}$ and a collection $\mathcal{C}$ of subsets of the universe, decide whether there exists a cover, i.e. a subcollection $C' \subseteq \mathcal{C}$ such that $\bigcup_{c \in C'} c = \mathcal{U}$.*

Theorem 7. *There exists a polynomial-time $2 ln n$-approximation for $\sigma_2(G)$.*

Proof. We create an instance of *Set Cover*, where $\mathcal{U} = V(G)$ is the universe and $\mathcal{C} = \{N[v] : v \in V(G)\}$ is the collection of subsets. Notice that a solution to this instance is exactly a solution to the dominating set problem. We produce an $ln n$-approximation for this instance by using the greedy algorithm for set cover [12]. Then, for any subset $\{N[v]\}$ in the output solution, put 2 guards on v. Now, if an attack happens on a vertex, at least a pair of guards dominate it. One of them moves to counter the attack and the other stands still such that domination is retained. If an attack happens on the same neighborhood, then the guard that moved before returns back and the other one moves to the attacked node; and so on for any number of attacks in the same neighborhood. If an attack happens in another neighborhood, then the guard who countered the last attack returns to his initial placement and the 2 guards that dominate the other neighborhood follow the same strategy to counter any series of attacks. Let apx stand for the cost of the approximate solution. To conclude, $apx/\sigma_2(G) \leq 2 ln n \cdot \gamma(G)/\sigma_2(G) \leq 2 ln n \cdot \gamma(G)/\gamma(G) \leq 2 ln n$, since $\gamma(G) \leq \sigma_2(G)$. □

4.3 Open Questions

The complexity of the problem remains open. One could look towards PSPACE-hardness as well as PSPACE-completeness. Answering such questions would provide significant contribution that could extend to other combinatorial games. Moreover, the parameterized complexity of the problem could be studied. Finally, (in)approximability conditions may be worth a try.

++ more on combinatorics

Chapter 5

Conclusions

Curse Nico Games

Bibliography

[1] M. Aigner and M. Fromme, A game of cops and robbers, *Discrete Applied Mathematics*, vol. 8, pp. 1-12, 1984.

[2] M.H. Albert, R.J. Nowakowski, D. Wolfe, Lessons in Play: An Introduction to Combinatorial Game Theory, *AK Peters*, 2007.

[3] M. Anderson, C. Barrientos et al. , Maximum demand graphs for eternal security, *Journal of Combinatorial Mathematics and Combinatorial Computing*, vol. 61, pp. 111-128, 2007.

[4] T. Andrae, Note on a pursuit game played on graphs, *Discrete Applied Mathematics*, vol. 9, pp. 111-115, 1984.

[5] B. Bollobás, G. Kun, I. Leader, Cops and robbers in a random graph, *Journal of Combinatorial Theory*, series B, vol. 103, issue 2, pp. 226-236, 2013.

[6] A. Bonato and R.J. Nowakowski, The Game of Cops and Robbers on Graphs, *American Mathematical Society*, 2011.

[7] A. Bonato, P. Pralat, C. Wang, Pursuit-Evasion in Models of Complex Networks, *Internet Mathematics*, vol. 4, pp. 419-436, 2007.

[8] J.A. Bondy and U.S.R. Murty, Graph Theory, *GTM 224, Springer*, 2008.

[9] A.P. Burger, E.J. Cockayne et al. , Infinite order domination in graphs, *Journal of Combinatorial Mathematics and Combinatorial Computing*, vol. 50, pp. 179-194, 2004.

[10] E. Chiniforooshan, A Better Bound for the Cop Number of General Graphs, *Journal of Graph Theory*, vol. 58, pp. 45-48, 2008.

[11] D-K Dang, Cops and Robber Games in Radius Capture and Faster Robber Variants, *Rapport de Stage*, MASCOTTE, INRIA, 2011.

[12] S. Dasgupta, C.H. Papadimitriou, U.V. Vazirani, Algorithms, *McGraw-Hill*, 2006.

[13] S.L. Fitzpatrick and R.J. Nowakowski, Copnumber of graphs with strong isometric dimension two, *Ars Combinatoria*, vol. 59, pp. 65-73, 2001.

[14] F. Fomin, S. Gaspers et al. , Parameterized algorithm for eternal vertex cover, *Information Processing Letters*, vol. 110, pp. 702-706, 2010.

[15] F. Fomin, F. Giroire et al., To satisfy impatient web surfers is hard. *FUN*, pp. 166-176, 2012.

[16] F. Fomin, P. Golovach, J. Kratochvíl, N. Nisse, K. Suchan, Pursuing Fast Robber in Graph, *Theoretical Computer Science*, vol. 411, issues 7-9, pp. 1167-1181, 2010.

[17] A.S. Fraenkel, Combinatorial Games: Selected Bibliography with a Succinct Gourmet Introduction, *Ongoing*.

[18] P. Frankl, Cops and robbers in graphs with large girth and Cayley graphs, *Discrete Applied Mathematics*, vol. 17, issue 3, pp. 301-305, 1987.

[19] M.R. Garey, D.S. Johnson, L. Stockmeyer, Some simplified NP-complete problems, *Proceedings of the sixth annual ACM symposium on Theory of computing*, pp. 47-63, 1974.

[20] F. Giroire, D. Mazauric et al. , Connected Surveillance Game, *20th Colloquium on Structural Information and Communication Complexity (SIROCCO)*, 2013.

[21] F. Giroire, N. Nisse, S. Pérennes, R. P. Soares, Fractional Combinatorial Two-Player Games, *Manuscript*, 2013.

[22] J. Goldwasser and W.F. Klostermeyer, Tight bounds for eternal dominating sets in graphs, *Discrete Mathematics*, vol. 308, pp. 2589-2593, 2008.

[23] W. Goddard,S.M. Hedetniemi, S.T. Hedetniemi, Eternal security in graphs. J. *Journal of Combinatorial Mathematics and Combinatorial Computing*, vol. 52, pp. 169-180, 2005.

[24] A.S. Goldstein and E.M. Reingold, The complexity of pursuit on a graph, *Theoretical Computer Science*, vol. 143, pp. 93-112, 1995.

[25] S. Janson, T. Luczak, A. Rucinski, Random Graphs, *Wiley-Interscience*, 2000.

[26] W.F. Klostermeyer and C.M. Mynhardt, Vertex covers and eternal dominating sets, *Discrete Applied Mathematics*, vol. 160, pp. 1183-1190, 2012.

[27] W.F. Klostermeyer and C.M. Mynhardt, Graphs with equal eternal vertex cover and eternal domination numbers, *Discrete Mathematics*, vol. 311, pp. 1371-1379, 2011.

[28] W.F. Klostermeyer and C.M. Mynhardt, Edge protection in graphs, *Australasian Journal of Combinatorics*, vol. 45, pp. 235-250, 2009.

[29] L. Lu and X.Peng, On Meyniel's conjecture of the cop number, *Journal of Graph Theory*, vol. 71, pp. 192-205, 2012.

[30] T. Luczak and P. Pralat, Chasing Robbers on Random Graphs: Zigzag Theorem, *Random Structures and Algorithms*, vol. 37, pp. 516-524, 2010.

[31] M. Maamoun and H. Meyniel, On a game of policemen and robber, *Discrete Applied Mathematics*, vol. 17, issue 3, pp. 307-309, 1987.

[32] M. Mamino, On the computational complexity of a game of cops and robbers, *Theoretical Computer Science*, vol. 477, pp. 48-56, 2013.

[33] R. Nowakowski and P. Winkler, Vertex-to-vertex pursuit in a graph, *Discrete Mathematics*, vol. 43, pp. 235-239, 1983.

[34] C.H. Papadimitriou, Computational Complexity, *Addison-Wesley*, 1993.

[35] S. Pérennes, Surfer, *Manuscript*, 2012.

[36] P. Pralat and N. Wormald, Meyniel's conjecture holds for random graphs, *Random Structures and Algorithms*, 2013.

[37] A. Quillot, Some results about pursuit games on metric spaces obtained through graph theory techniques, *European Journal of Combinatorics*, vol. 7, pp. 55-66, 1986.

[38] R. Raz and S. Safra, A sub-constant error-probability low-degree test and a sub-constant error-probability PCP characterization of NP, *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing (STOC '97)*, pp. 475-484, 1997.

[39] A. Scott and B. Sudakov, A bound for the Cops and Robbers problem, *SIAM Journal on Discrete Mathematics*, vol. 25, pp. 1438-1442, 2011.

[40] D.B. West, Introduction to Graph Theory, *Pearson*, 2000.