

1-1-2012

The CC-Game: A Variant Of The Game Of Cops And Robbers

Ali Haidar
Ryerson University

Follow this and additional works at: <http://digitalcommons.ryerson.ca/dissertations>

 Part of the [Applied Mathematics Commons](#)

Recommended Citation

Haidar, Ali, "The CC-Game: A Variant Of The Game Of Cops And Robbers" (2012). *Theses and dissertations*. Paper 1391.

This Thesis is brought to you for free and open access by Digital Commons @ Ryerson. It has been accepted for inclusion in Theses and dissertations by an authorized administrator of Digital Commons @ Ryerson. For more information, please contact bcameron@ryerson.ca.

THE CC-GAME: A VARIANT OF THE GAME OF COPS AND ROBBERS

by

Ali Haidar, Bachelor of Science in Computer Science

Arts, Science and Technology University

Jadra, Lebanon, 2009

A thesis presented to
Ryerson University
in partial fulfilment of the
requirements for the degree of
Masters of Science
in the Program of
Applied Mathematics

Toronto, Ontario, Canada, 2012

©Ali Haidar 2012

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my thesis may be made electronically available to the public.

The CC-Game: A Variant Of The Game Of Cops And Robbers

Ali Haidar

Masters of Science in Applied Mathematics

Ryerson University

Toronto, Ontario, Canada, 2012

Abstract

Cops and Robbers is a vertex pursuit game played on graphs. The objective of the game, as the name suggests, is for a set of cops to catch the robber. We study a new variant of this game in which the robber can attack a cop or fight back. This variation restricts the movement of the cops and changes many of the parameters and strategies achieved in the regular game. We explore aspects of this variant such as classifications for certain cop numbers, upper and lower bounds, strategies on special graphs, the cop number on products of graphs, complexity of computations, and density of cops in infinite graphs.

Acknowledgements

I would like to thank the Faculty and Staff at the Department of Mathematics at Ryerson University for all their help and support. Their contributions have been essential for the completion of this thesis. First and foremost, my deepest thanks go to my supervisor Dr. Anthony Bonato. His enthusiasm and interest in Graph Theory and Mathematics is a driving force to everyone who works with him. Dr. Bonato's insight, guidance and patience was essential to this work and to my development as a mathematician. I would like to thank Dr. Peter Danziger and Dr. Dejan Delić for being part of my thesis committee and for their guidance. Special thanks go to all the members, organizers and speakers of Graphs at Ryerson who helped expand my knowledge and interest in Mathematics. I would like to also thank Steve Kanellis for his technical help and support.

I would like to also express my gratitude to my parents, Mohammad and Fatima, and my wife, Emma. Their unwavering support and constant encouragement continues to be an inspiration for all my work and was invaluable for the completion of this thesis.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Graphs: Introduction and Notation	4
1.3	Asymptotic Notation	8
1.4	Graph Theory Concepts	9
1.4.1	Random Graphs	9
1.4.2	Graph Algorithms	11
1.5	Overview of Thesis	13
2	The cc-Game	14
2.1	The cc-number	14
2.1.1	Graph Girth	17
2.2	Cycles	19
2.3	Retracts	22
2.4	Isometric Paths	24
2.5	Corners and Dismantlable Graphs	27

Contents

2.6	Joined Graphs	29
3	Planar and Product Graphs	31
3.1	Introduction	31
3.2	Planar and Outerplanar Graphs	31
3.2.1	Outerplanar Graphs	32
3.3	Graph Products	39
3.3.1	Strong Product	40
3.4	Nearly Irreducible Vertices	44
4	Miscellaneous Results on the cc-Game	49
4.1	Introduction	49
4.2	Complexity	50
4.2.1	Cops, Robbers, and the cc-Game	51
4.3	Random Graphs	53
4.4	Infinite Graphs	56
4.4.1	Chains and Cop Density	59
5	Results and Open Problems	64
5.1	Introduction	64
5.2	Results and Theorems	64
5.3	Open Problems and Conjectures	65
	Bibliography	67

List of Figures

1.1.1 Robber attacks the cop	3
1.1.2 A backup cop	3
1.2.1 A disconnected graph	5
1.2.2 Distance between vertices	6
1.2.3 Families of graphs	6
1.2.4 A homomorphism on Cycles	7
1.4.1 A random graph $G(20, 0.5)$	11
1.4.2 A graph with an Euler circuit.	13
2.1.1 One cop is not enough	17
2.2.1 Dominating set in C_6	20
2.2.2 Large cycles need more than two cops	21
2.2.3 Three cops on large cycles	21
2.3.1 A retraction $u \rightarrow v$	22
2.4.1 An isometric path	24
2.4.2 An isometric path need not be 2-guardable	26
2.5.1 cc-Game on a dismantlable graph	28

List of Figures

2.6.1 Construction of Joined graphs	29
3.2.1 Planar graphs	32
3.2.2 An outerplanar graphs	33
3.2.3 A maximal outerplanar graphs	34
3.2.4 Cop territory on an Outerplanar graphs	36
3.2.5 Truncated Icosahedral	39
3.3.1 Strong Product of two graphs.	41
3.3.2 Cops move diagonally on Strong product graphs	44
3.4.1 Ordering of Nearly irreducible vertices.	45
3.4.2 A kite graph.	46
4.2.1 The four complexity classes.	51
4.4.1 Finite number of cops on an infinite tree	57
4.4.2 Graph with e.c. property	58
4.4.3 A robber escaping finitely many cops in R	59

Chapter 1

Introduction

1.1 Motivation

Cops and Robbers is a vertex pursuit game played on a reflexive, undirected graph (that is, a graph where there is a loop on each vertex). The objective of the game, as the name suggests, is for a set of cops to catch the robber. This game was introduced by Quilliot [14] in his doctoral thesis and further independently considered by Nowakowski and Winkler [15]. Since then, the game of Cops and Robbers has been the subject of much interest in the mathematical and computer science literature; for more background, see the recent book [5].

The game is played over a series of *rounds* and starts with the k cops choosing at most k vertices. The robber then picks a vertex himself to start on. At the beginning of each round, the cops will move to neighboring

1.1. Motivation

vertices. The robber will also move to one of its neighboring vertices after the cops' move. Since the game is played on a reflexive graph, the players can choose to remain on their present vertex. A *winning strategy* for the cops is a set of instructions that enables them to catch the robber after a finite number of rounds. If the robber cannot be caught with a finite number of moves from the cops, then the robber wins. A graph in which a strategy exists for one cop to win is called a *cop-win*. A graph in which a strategy exists for k cops to win is called *k-cop-win*. The minimum number of cops needed to capture a robber in a graph is called the *cop number* of that graph. This graph parameter is well defined with its maximum being the number of vertices of the graph. For a graph G , the cop number is denoted $c(G)$. The cop number was first introduced and defined in by Aigner and Fromme in [1].

Although the game of Cops and Robbers has applications to counter-terrorism and missile guidance systems (see [12]), it sometimes is far from realistic. When we vary some conditions of the game, our understanding of all the aspects of it may be dramatically reduced. Therefore, some researchers started considering variants of the game. Some of the variants studied were a game in which the cop can catch the robber from a distance k and another one in which the cop has imperfect information on the location of the robber; see Chapter 8 of [5].

We consider a new variant of the game of Cops and Robbers. In this variant, the robber is able to attack the cop. We say that the robber *attacks*

1.1. Motivation

the cop if he chooses to move to the vertex on which the cop is present and eliminate her from the game. This variation places many restrictions on the cops' strategy. A cop is unable to be at distance one from a robber unless she is also adjacent to another cop. The job of the second cop would be to capture the robber if he chooses to attack the first cop. In this variant, we write $cc(G)$ for the minimum number of cops needed to capture the robber. We refer to $cc(G)$ as the *cc-number of G* . See Figure 1.1 and Figure 1.2.

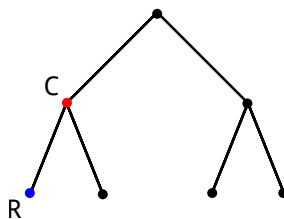


Figure 1.1.1: Robber can attack the cop in next move.

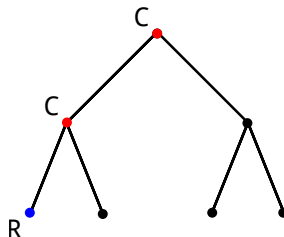


Figure 1.1.2: A second cop protecting a cop adjacent to the robber.

Before we proceed further, in the next section we provide some background and notation in graph theory.

1.2 Graphs: Introduction and Notation

In this section we will introduce graph theory and the necessary notation used in this thesis. A graph G has a non-empty *vertex set* $V(G)$ and an *edge set* $E(G)$. A graph consisting of $V(G)$ and $E(G)$ is sometimes referred to as $G = (V, E)$. We say G is of *order* n if the cardinality of its vertex set $V(G)$ is equal to n . If u and v are two vertices that have an edge between them, then we call that edge uv . We say that vertices u and v are *adjacent* or *joined*. Alternatively, we might simply say that $u \sim v$ meaning that u and v are adjacent. Graphs can be either finite or infinite, and an infinite graph has an infinite number of vertices. We assume for all purposes that all graphs are finite, unless otherwise stated. A *loop* is an edge from a vertex to itself. Graphs can contain loops or multiple edges. Multiple edges consist of two or more edges connecting a pair of vertices u and v in a graph G . We assume that all graphs do not contain any multiple edges since they make no difference in the play of the game.

We define the set $N(u)$ as the set of vertices adjacent to u and the set $N[u]$ as $N(u) \cup \{u\}$. The *degree* of a vertex u is the cardinality of its neighbour set $N(u)$. We refer to the degree of some vertex u as $\deg_G(u)$, or $\deg(u)$ if G is clear from context. A graph in which every vertex has degree k is called *k-regular*. A *path* is a sequence of vertices such that each vertex is joined to the next vertex in the sequence; the *length* of a path is the number of its edges. We write P_n for the path with n vertices. We call a graph

1.2. Graphs: Introduction and Notation

connected if there is a path between any two of its vertices. A *disconnected* graph, on the other hand, is a graph that has two vertices u and v that are not connected by a path (see Figure 1.2.1). All the graphs we study in this thesis are connected, unless otherwise stated. A vertex whose neighbour set is empty is called an *isolated* vertex.

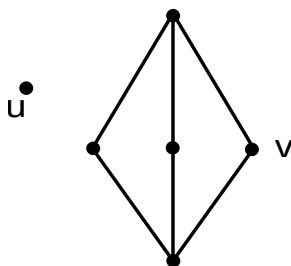


Figure 1.2.1: A disconnected graph with an isolated vertex u .

A vertex whose neighbour set contains exactly one vertex is called an *end-vertex*. Finally, a *universal* vertex is a vertex that is adjacent to every other vertex in the graph. In a graph of order n , the degree of a universal vertex is $n - 1$.

The *distance* between two vertices u and v in some graph G is the number of edges on the shortest path between those two vertices and is referred to as $d(u, v)$ (see Figure 1.2.2). If $u = v$, then $d(u, v) = 0$. If either u or v is disconnected, then the distance is ∞ . The *diameter* of a graph G , referred to as $\text{diam}(G)$, is the maximum of all distances between vertex pairs in G . In a disconnected graph, the diameter is ∞ .

1.2. Graphs: Introduction and Notation

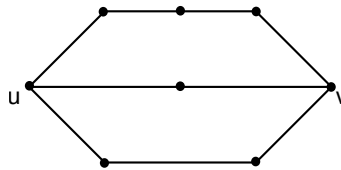


Figure 1.2.2: In this graph, $d(u, v) = 2$

We now define certain families of graphs. A *cycle*, referred to as C_n , is a connected graph of order n in which every vertex has degree equal to 2. A complete graph with n vertices, called a *clique* and referred to as K_n , is a graph containing all possible edges except loops. A cycle along with one vertex adjacent to all its vertices forms a *wheel*. A wheel of order n is referred to as W_n . See Figure 1.2.3 for examples of these graphs.

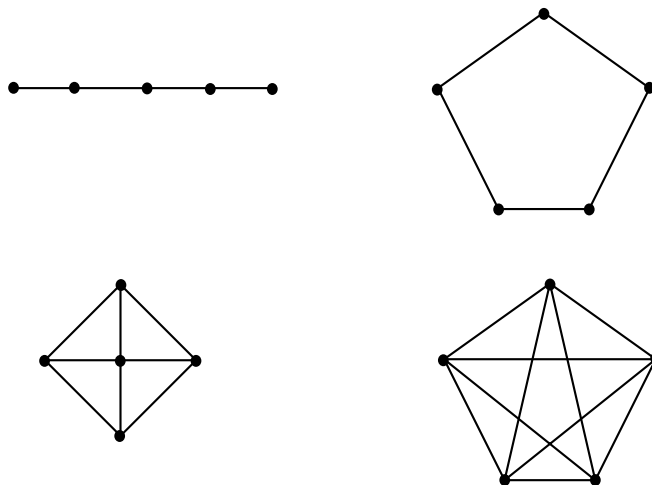


Figure 1.2.3: Clockwise from top left: a path, cycle, clique and wheel of order 5.

In a graph G , a *dominating* set of vertices S has the property that each vertex not in S is joined to some vertex in S . The *domination number*, denoted $\gamma(G)$, is the number of vertices of the smallest dominating set S in

1.2. Graphs: Introduction and Notation

G . For example, a graph G with a universal vertex u will have $\gamma(G) = 1$ and $S = \{u\}$.

Lemma 1.1. *In a graph G , we have that $c(G) \leq \gamma(G)$.*

Proof. Let G be a graph with domination number $\gamma(G)$, and let S be a minimum order dominating set. In the first round, we place one cop on each vertex in S . Since every vertex in the dominating set has a cop on it, then any vertex the robber chooses to start on will be connected to at least one cop. The robber will be captured in the following round. \square

The *minimum degree* is the degree of the vertex of G that has the least number of edges incident to it. We refer to the minimum degree of a graph G as $\delta(G)$.

A *homomorphism* f from G to H is a function $f : V(G) \rightarrow V(H)$ which *preserves edges*, that is, if $xy \in E(G)$, then $f(x)f(y) \in E(H)$. We will refer to such a homomorphism as $f : G \rightarrow H$.

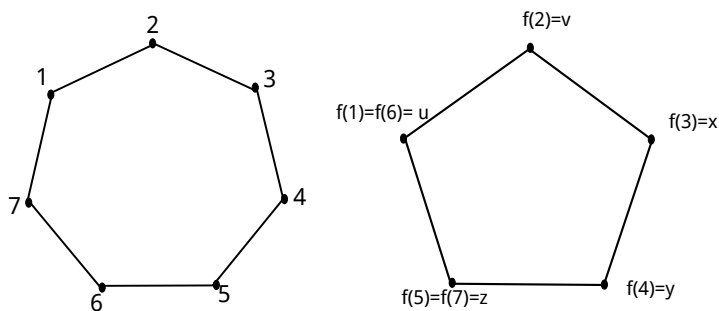


Figure 1.2.4: A homomorphism f from C_7 to C_5 .

In Figure 1.2.4, we have a homomorphism. Note that homomorphisms can add edges between the non-adjacent vertices in G . For example, in the

1.3. Asymptotic Notation

figure, we have that the vertices 1 and 5 are not joined in C_7 but their images are joined. The homomorphism we have in the figure is $f : C_7 \rightarrow C_5$ defined by: $f(1) = f(6) = u, f(2) = v, f(3) = x, f(4) = y$ and $f(5) = f(7) = z$.

An *embedding* from G to H is an injective homomorphism $f : G \rightarrow H$ with the property that $xy \in E(G)$ if and only if $f(x)f(y) \in E(H)$. An *isomorphism* is a bijective embedding of G to H . Isomorphisms are one-to-one mappings from one graph to another that preserve adjacency and non-adjacency. If G and H are isomorphic, then we denote this by $G \cong H$.

1.3 Asymptotic Notation

Several results that will follow in this thesis are asymptotic so we present some asymptotic notations that will be used. Let f and g be two functions whose domains are a subset of \mathbb{R} . We write $f = O(g)$ if

$$\limsup_{x \rightarrow \infty} \frac{f(x)}{g(x)}$$

exists and is finite. This is equivalent to saying that for some constants c and N , $f(x) \leq cg(x)$ for $x > N$. If $f = O(g)$ and $g = O(f)$, then we will say that $f = \Theta(g)$. We also say that $f = o(g)$ if $\lim_{x \rightarrow \infty} \left| \frac{f(x)}{g(x)} \right| = 0$. An example of the use of the big O notation in the context of the game of cops and robbers is Meyniel's conjecture which states that if G is a connected graph of order n , then its cop number satisfies $c(G) = O(\sqrt{n})$. See [9] and Chapter 3 of [5]

1.4. Graph Theory Concepts

for more on Meyniel's conjecture.

1.4 Graph Theory Concepts

We move on in this section to more advanced concepts within graph theory. In particular, we introduce random graphs and graph algorithms. One of the most powerful tools in graph theory is the *probabilistic method*. The probabilistic method, pioneered by Paul Erdős and Alfréd Rényi, is a method used to prove a property for a certain class of combinatorial objects. If every object in a certain class fails to satisfy a certain condition or have a certain property, then the probability of an object picked randomly from this class satisfying that condition or having that property is zero. Hence, if the probability of a randomly chosen object from a certain class having that property is greater than zero, then there must exist at least one object from that class that has that property. Although the probabilistic method uses probabilistic proofs, it can be used to prove results which have no explicit reference to probability in their statements.

1.4.1 Random Graphs

We study in this thesis one of the tools of this probabilistic method, the *random graph*. In random graphs, the adjacency between two vertices is determined by a probability $p \in (0, 1)$. Define a probability space on graphs of a given order $n \geq 1$ as follows. Fix a vertex set V consisting of n distinct

1.4. Graph Theory Concepts

elements, usually taken as $[n]$, and fix $p \in [0, 1]$. Define the space of *random graphs of order n with edge probability p* , written $G(n, p)$, with sample space equalling the set of all $2^{\binom{n}{2}}$ (labelled) graphs with vertex set V , and

$$\mathbb{P}(G) = p^{|E(G)|} (1-p)^{\binom{n}{2}-|E(G)|}$$

A random graph G with n vertices is referred to as $G(n, p)$, where p is the edge probability (see Fig 1.4.1). In $G(n, p)$, the probability p can be either constant or a function of n such that $p = p(n)$. We only study random graphs with a constant p in this thesis. The cop number and the cc-number for a random graph are random variables over the probability space $G(n, p)$. These two variables will be estimated in terms of n and p . A complete survey of the results on the game of cops and robbers on random graphs can be found in Chapter 6 of [5]. We use some of these results and expand them to the cc-game in Chapter 3.

1.4. Graph Theory Concepts

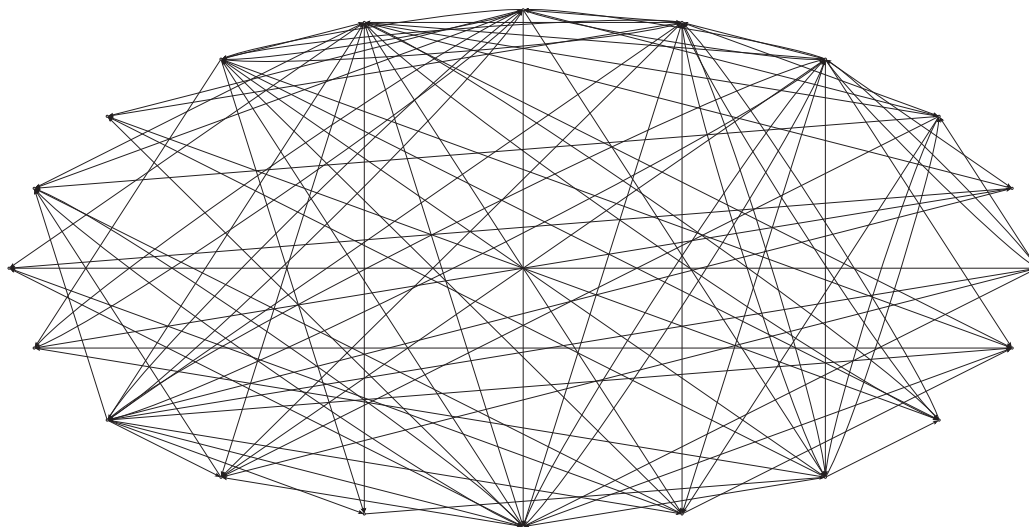


Figure 1.4.1: A randomly drawn graph with 20 vertices, where edges were drawn with probability 0.5. The vertices are located on the perimeter of the figure.

1.4.2 Graph Algorithms

An important notion for a graph parameter is determining the complexity of an algorithm used to compute the parameter. Complexity is the measure of the inherent difficulty of a computational problem. The measure of complexity is often described in the amount of time needed to execute an algorithm or the number of steps needed relative to the size of the problem. In this thesis, we describe the complexity of computing or checking the validity of a cop number and a cc-number. We describe this complexity in terms of the number of vertices in the graph in question. All algorithms we consider focus on *graph decision problems*. In graph decision problems, the *input* is a

1.4. Graph Theory Concepts

graph usually represented in a data structure that can describe the graph's vertices and their adjacencies. The complexity of a computational problem can be in one of several complexity classes, the two most studied of which are polynomial time **P** or non-deterministic polynomial time **NP**. Problems in **P** are those that are *solvable in polynomial time*, that is, their complexity is $O(n^m)$ for some integer $m \geq 0$. **NP** is the set of decision problems such that if the answer is YES, then there is a *certificate* or *proof* of this fact that can be checked in polynomial time. An example of a **P** problem is checking if a connected graph is *Eulerian* which means that there exists a circuit within this graph spanning each edge exactly once. The vertices of an Eulerian graph must all have an even degree (see Figure 1.3.2). An algorithm would calculate the degrees of a graph. The result will be found in polynomial time. An example of an **NP** problem is finding if graph contains a cycle that includes all its vertices. Such a cycle is called a *Hamiltonian cycle*. Without a more efficient algorithm, one can simply exhaustively examine all the permutations of vertices for a cycle spanning all vertices. For n vertices, that is $n!$ permutations which is a huge number of possible solutions even for a relatively small n . Furthermore, not all **NP** problems are of equal complexity. The problem of finding a Hamiltonian cycle is **NP**-complete which means that finding an efficient algorithm is very unlikely. So-called **NP**-hard problems are those which are as hard as any in **NP**. We will discuss the complexity of computing the cc-number of a graph in Chapter 4.

1.5. Overview of Thesis

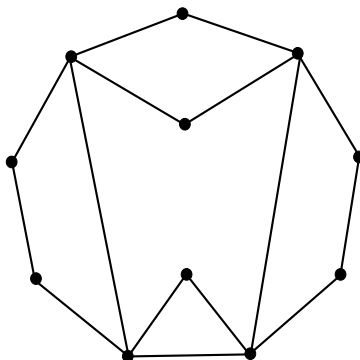


Figure 1.4.2: All the vertices of an Eulerian graph have even degrees.

1.5 Overview of Thesis

In this thesis, we derive results on the cc-number of a graph. We find an upper and lower bound on $cc(G)$ in terms of $c(G)$. We also compute the cc-numbers on cycles and outerplanar graphs and find the winning strategies on these graphs. We then investigate classifications of graphs by cc-number and found a classification for graphs with $cc(G) = 1$ and a family of graphs with $cc(G) = 2$. We also compute upper bounds on the cc-number of a strong product of two graphs, investigate the complexity of computing $cc(G)$, compute the density of cops in infinite graphs, and the order of the cc-number in random graphs. These results and others will be explained in detail in Chapters 2, 3, and 4. In Chapter 5, we summarize the results we found and the problems that remain unsolved.

Chapter 2

The cc-Game

2.1 The cc-number

The game of Cops and Robbers is one of many vertex-pursuit games played on graphs. A great amount of literature has been published about this game and some important theorems have been found; see [5]. We study in this chapter the new variant we briefly introduced in Chapter 1, called the *cc-game*. In the *cc-game*, the only difference from the regular Cops and Robbers game is that a robber is able to “attack” the cop. The rules otherwise are the same. In the first round, the cops choose the vertices on which they can start. The robber then chooses a vertex different than those chosen by the cops in the first round. The game then proceeds with the cops moving according to a strategy that enables them to catch the robber, if such a strategy exists.

We first try to relate the *cc-number* to the cop number in the regular

2.1. The cc-number

game. By comparing the two parameters, we can exploit existing theorems on the cop number. Theorem 2.1 gives us an upper and lower bound on the cc-number in terms of the cop number of graph G and its domination number $\gamma(G)$.

Theorem 2.1. *If G is a graph, then*

$$c(G) \leq cc(G) \leq \min\{2c(G), \gamma(G)\}.$$

Proof. We start by proving that $c(G) \leq cc(G)$. Let \mathcal{S} be a winning strategy of $c(G)$ -many cops on G . Place $cc(G)$ -many cops on G and move them according to the strategy \mathcal{S} . Since $cc(G)$ -many cops are enough to win in the cc-game, it will be enough to capture the robber in the regular Cops and Robbers game (since the robber gains no extra moves by reverting to those rules and the cops' movements are not otherwise restricted).

Now we prove that $cc(G) \leq 2c(G)$. Let \mathcal{S} be the strategy followed by the $c(G)$ -many cops to capture the robber in the regular game. Place two cops on each vertex where \mathcal{S} tells you to place one cop, thus, doubling the number of cops. Follow the strategy \mathcal{S} by moving the two partnering cops at once as if they were one cop. Since there are always two cops together on any vertex occupied by the set of cops, the robber can never attack the cops and the strategy \mathcal{S} can be followed to catch the robber.

Lastly, we prove that $cc(G) \leq \gamma(G)$. Assume there are $\gamma(G)$ -many cops in the cc-game. Start by placing a cop on each vertex belonging to the

2.1. The cc-number

dominating set of G called X . The robber, by the rules of the cc-game, cannot attack a cop on his first move and therefore, will occupy any vertex $u \notin X$. Since there is a cop on each vertex of the dominating set, at least one cop will be joined to the vertex u . The cop will capture the robber in the following step. \square

We know by Theorem 2.1 that the cc-number is at most twice the cop number (hence, the name of the parameter). However, we conjecture something stronger. For all the graphs that we know, Conjecture 2.2 holds.

Conjecture 2.2. *For all graphs G ,*

$$cc(G) \leq c(G) + 1.$$

An important part of studying the game of Cops and Robbers is classifying graphs with a specific cop number. The same is also relevant in the cc-game. We are interested in classifying the graphs that share the same cc-number. Theorem 2.3 classifies all graphs G that have cc-number $cc(G) = 1$.

Theorem 2.3. *For a graph G , $cc(G) = 1$ if and only if $\gamma(G) = 1$.*

Proof. Suppose that for a graph G , we have that $cc(G) = 1$. If the cop captures the robber in the first round, then the vertex the cop occupies must be adjacent to all other vertices in G . Hence, $\gamma(G) = 1$. Conversely, if $\gamma(G) = 1$, then G has a universal vertex. Hence, $cc(G) = 1$. \square

2.1. The cc-number

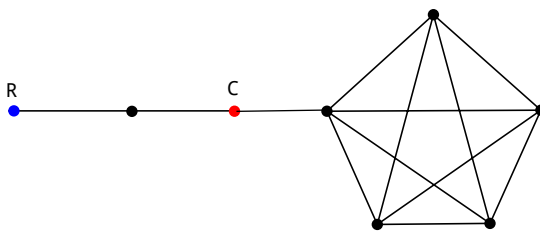


Figure 2.1.1: A lone cop can only catch the robber in the first round.

A large class of graphs that has been studied intensively is cop-win graphs (see [15] and Chapter 2 of [5]). Corollary 2.4 follows directly from Theorems 2.1 and 2.3 and shows that cop-win graphs have a cc-number equal to 2 unless they contain a universal vertex.

Corollary 2.4. *If $c(G) = 1$ and $\gamma(G) \neq 1$, then $cc(G) = 2$.*

2.1.1 Graph Girth

The *girth* of a graph is the length of the shortest cycle contained in that graph. For example, a graph of girth at least 4 contains no triangles. Aigner and Fromme proved in [1] Theorem 2.5 that places a lower bound on the cop number of a graph G if that graph has girth greater than four.

Theorem 2.5 ([1]). *If G has a girth at least 5, then*

$$c(G) \geq \delta(G),$$

where $\delta(G)$ is the minimum degree in G .

Theorem 2.5, with some modifications, can be applied to the cc-game.

2.1. The cc-number

Theorem 2.6 finds a lower bound for the cc-number in graphs whose girth is greater than or equal to 5.

Theorem 2.6. *If G has a girth at least 5 and $c(G) \neq \gamma(G)$, then*

$$cc(G) \geq \delta(G) + 1$$

where $\delta(G)$ is the minimum degree in G .

Proof. This proof is analogous to the proof given by Aigner and Fromme in Theorem 2.5. We start with $\delta(G)$ cops and prove that the robber survives indefinitely. We must first show that the robber survives in the first round. For the robber to be caught in the first round, $cc(G) = \gamma(G)$, which is an upper bound for $cc(G)$ in any graph. Since we have $\delta(G)$ cops, this means that $\delta(G) = \gamma(G)$. Theorem 1 implies that $c(G) \leq cc(G) \leq \gamma(G)$ and from [1] we know that $c(G) \geq \delta(G)$. Putting all these inequalities together we obtain that :

$$\gamma(G) \leq c(G) \leq cc(G) \leq \gamma(G),$$

which implies that all these values are equal. Hence, for the robber to be caught in the first round, $c(G)$ must be equal to $\gamma(G)$. This violates the hypothesis of the theorem.

Let C be the set of vertices occupied by the cops. The set C has at most $\delta(G)$ cops. The set $N(R)$ is the set of vertices adjacent to the robber. Since there are no 3-cycles, none of the vertices of $N(R)$ are joined and

2.2. Cycles

$|N(R)| \geq \delta(G)$. None of the vertices of C can be joined to two vertices of $N(R)$ since that would result in a four cycle, hence, at most $\delta(G)$ cops on $\delta(G)$ vertices of C will be joined to $\delta(G)$ vertices belonging to $N(R)$. Now if $|N(R)| > \delta(G)$, then the robber can move to a vertex not adjacent to the $\delta(G)$ cops. If $|N(R)| = \delta(G)$, then the robber will pass indefinitely with none of the cops being able to move to any of the vertices of $N(R)$ since she will be attacked by the robber. Since none of the vertices of $N(R)$ are adjacent, any two cops moving to any two vertices of $N(R)$ will not catch the robber. We therefore need at least one extra cop to capture the robber, and so, $cc(G) \geq \delta(G) + 1$. \square

2.2 Cycles

Cycles are special graphs in which the degree of each vertex is equal to two. The cop number, in any cycle C_n where $n \geq 4$, is 2. In Theorem 2.7, we prove that the cc-number in cycles can be either 2 or 3 depending on the number of vertices.

Theorem 2.7. *A cycle C_n with n vertices has $cc(C_n) = \begin{cases} 2 & \text{if } 4 \leq n \leq 6, \\ 3 & \text{if } n \geq 7. \end{cases}$*

Proof. Since each vertex in a cycle is joined to two vertices, a single vertex will dominate three vertices (two neighbours and itself). So if in a cycle C_n we have $4 \leq n \leq 6$, then C_n has domination number $\gamma(C_n) = 2$ (see Figure 2.2.1). From Theorem 2.1 we obtain that $1 \leq cc(G) \leq 2$ and since $\gamma(C_n) = 2$,

2.2. Cycles

then by Theorem 2.3 we can know that $cc(C_n) > 1$. Therefore, $cc(C_n) = 2$ when $4 \leq n \leq 6$.

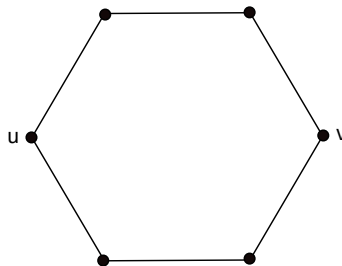


Figure 2.2.1: A cycle C_6 with dominating set $S = \{u, v\}$ and $\gamma(G) = 2$.

When $n \geq 7$, since $c(C_n) = 2$, by Theorem 2.1 $cc(G)$ must be between 2 and 4. Place two cops on any two vertices. The first option would be that both cops move, in tandem or on the same vertex, in the same direction towards the robber (counter-clockwise or clockwise). Assuming without loss of generality that the two cops are moving clockwise towards the robber, then whenever the two cops are at distance 1 from the robber, the robber moves clockwise increasing the distance to 2. This happens indefinitely and the robber wins. The second option would be moving one cop clockwise and the second cop counter-clockwise. After a finite number of rounds, the robber will be stuck with his two neighbours adjacent to cops ($N(r) = \{u, v\}$, see Figure 2.2.2). The robber will pass whenever it is his turn to move and either cop cannot move to u or v since they will be at distance 1 from the robber and can therefore be attacked. The robber passes indefinitely and wins.

2.2. Cycles

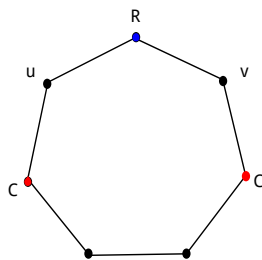


Figure 2.2.2: Robber passes indefinitely.

Now assume $cc(C_n) = 3$. Move one cop clockwise and two cops, on the same vertex, counter-clockwise. After some round $t > 0$, the robber will be at distance 2 from one cop on one side and at distance 2 from two cops on the other side (see Figure 2.2.3). The robber will have to pass when it is his turn since moving to an adjacent vertex means being at distance 1 from the cop and thus, being caught in the following move. The two cops that are moving together then move to be at distance 1 from the robber forcing him to be caught by the lone cop if he moves and by them if he passes. \square

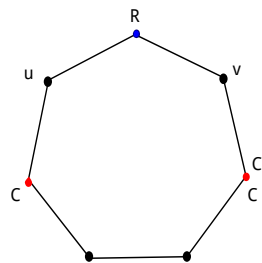


Figure 2.2.3: 2 cops move counter-clockwise.

2.3 Retracts

Retracts play a major role in some theorems and their proofs in the Cops and Robbers game. Let H be a subgraph of graph G attained by deleting a vertex from G . We say H is a *retract* of G if there is a homomorphism f from G onto H so that $f(x) = x$ for every vertex x in $V(H)$. The mapping f , therefore, is the identity on H , and is called a *retraction*. For such a mapping to be a retraction, distances between any two vertices do not increase in H compared to those in G . In Figure 2.3.1, $N[u] \subseteq N[v]$ and the mapping

$$f(x) = \begin{cases} v & \text{if } x = u, \\ x & \text{otherwise.} \end{cases}$$

is a retraction. In this case, we can say $u \rightarrow v$.

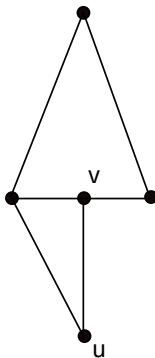


Figure 2.3.1: A retraction $u \rightarrow v$

We will mention two theorems related to retracts and the cop number in the game of Cops and Robbers. Both theorems were proven by Berarducci

2.3. Retracts

and Intrigila in [2]. Theorem 2.8 proves that the cop number never increases in a retract.

Theorem 2.8. *If H is a retract of G , then $c(H) \leq c(G)$.*

Theorem 2.9 finds an upper bound on the cop number of graphs using retracts.

Theorem 2.9. *If H is a retract of G , then*

$$c(G) \leq \max\{c(H), c(G - H) + 1\}.$$

We were able in Theorem 2.10, through an analogous approach to the one used by Berarducci and Intriglia, to find an upper bound on the cc-number of graphs using retracts.

Theorem 2.10. *If H is a retract of G , then*

$$cc(G) \leq \max\{cc(H), cc(G - H) + 2\}.$$

Proof. Let $m = \max\{cc(H), cc(G - H) + 2\}$, so $m \geq cc(H)$. Start with m cops in the subgraph H playing the winning strategy there. If the robber remains in H , then he will be caught eventually. If the robber goes to $G - H$, then we leave two cops in H sending at least $cc(G - H)$ cops to $G - H$ since $m \geq cc(G - H) + 2$. The two cops remaining in H will occupy the robber's retraction in H . If the robber decides to go to H , then the retraction is equal

2.4. Isometric Paths

to the actual position (that is, $f(x) = x$) of the robber and he is caught. If he stays in $G - H$, then he will be caught by the cops there. \square

2.4 Isometric Paths

As is the case for retracts, isometric paths play an important role in several key theorems in the game of Cops and Robbers, such as the the cop number in planar graphs (See Chapter 4 in [5]). We call a path P in a graph G *isometric* if the shortest distance between any two vertices is equal in P and G ; that is, for all vertices u and v in P , we have that $d_P(u, v) = d_G(u, v)$. An example of an isometric path is shown in Figure 2.4.1.

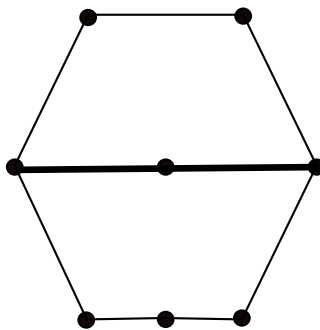


Figure 2.4.1: The bolded path is an isometric path in this graph.

Another concept that is related to retracts is *k-guardability*. For a fixed integer $k \geq 1$, an induced subgraph H of G is *k-guardable* if, after finitely many moves, k cops can move only in the vertices of H in such a way that if the robber moves into H at round t , then he will be captured at round $t + 1$. For example, a clique in a graph is 1-guardable.

2.4. Isometric Paths

The concepts of isometric paths and k -guardability were used by Aigner and Fromme in [1] to prove that an isometric path is 1-guardable.

Theorem 2.11 ([1]). *An isometric path is 1-guardable.*

By providing an analogous proof to the one used by Aigner and Fromme, we prove in Theorem 2.12 that an isometric path is 2-guardable in the cc-game.

Theorem 2.12. *An isometric path is 2-guardable, but need not be 1-guardable.*

Proof. Let $P = \{v_0, v_1, \dots, v_k\}$ be an isometric path in a graph G , and let $D_i = \{x \in V(G) : d(x, v_0) = i\}$. Since P is an isometric path, it follows that $v_i \in D_i$ for $i = 0, 1, \dots, k$. The cops, restricted to P , play as if the robber is on v_j when the robber is on some vertex of D_j , $j = 0, 1, \dots, k - 1$, and on v_k when $j \in D_k$. We will call this the robber's *image*. If the robber is in D_j , then he can only move to vertices of D_{j-1} , D_j and D_{j+1} , so his image can only move from v_j to v_{j-1} , or v_{j+1} , or remain at v_j . Start both cops on v_0 . As far as they and the image are concerned, they are playing the game on a path, and in this game the cops win. Note that the cops will always move together occupying the same vertex at any round. After the image has been caught, the actual robber can still move in G , but the robber's image moves to an adjacent vertex on P or is stationary. The cops now move to recapture the image. Suppose that the robber tries to enter P after his image has been caught. Before his move onto P , he is in D_j for some j . If $j < k$, then his image is on v_j and so are the cops. The robber can only move to one of

2.4. Isometric Paths

v_{j-1} , v_j , or v_{j+1} . Whichever vertex he chooses, the cops will capture him on the next move. If $j \geq k$, then the image is on v_k and so are the cops. The robber's only possible moves are to v_k or v_{k-1} and the cops captures him on the next move.

The same proof may fail in the cc-game if only one cop is used. A robber can leave and enter an isometric path an infinite number of times in some cases because the cop cannot be at distance 1 from him. This restricts the movement of the cop. Consider, for example, Figure 2.4.2

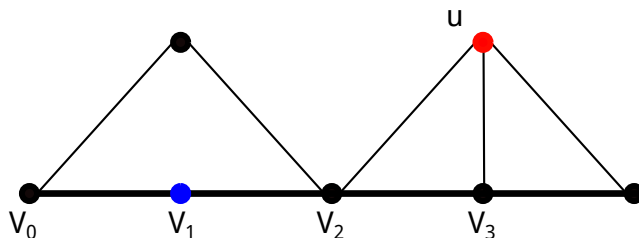


Figure 2.4.2: The cop cannot move to v_2 on the isometric path (bolded).

The cop starts on v_0 and the robber starts on u . The cop then moves, according to the strategy we used in the proof so far, to v_1 while the robber remains on u . Now the cop is supposed to move to v_2 . If she does, however, she will be at distance 1 from the robber and therefore can be attacked. The robber can move freely now between u and v_3 (into and out of the isometric path) and the cop can't capture him or stop him from entering. \square

2.5 Corners and Dismantlable Graphs

The cornerstone of characterizing cop-win graphs in the game of Cops and Robbers is the notion of a corner. Consider a graph G with vertices $u \in V(G)$ and $v \in V(G)$. We say u is a *corner* in G if we have $N[u] \subseteq N[v]$. The importance of corners comes from the fact that if the robber occupies a corner, then there exists a vertex which can be occupied by the cop that is adjacent to the robber and all its neighbours. The cop can catch the robber in the following move. The vertex v is said to *cover* or *dominate* u .

A concept that follows from corners is *dismantlable* graphs. A graph is dismantlable if some sequence of deleting corners results in the graph K_1 . For example, each tree is dismantlable: delete end-vertices repeatedly until a single vertex remains. An example of a graph that is not dismantlable is a chordless cycle.

Theorem 2.13 on dismantlable graphs characterizes all cop-win graphs.

Theorem 2.13. *A graph is cop-win if and only if it is dismantlable.*

We studied corners to see if they have any significance in the cc-game. Corollary 2.14 follows from Theorems 2.3 and 2.13. Theorem 2.15 studies the effect of adding corners to dismantlable graphs in the cc-game.

Corollary 2.14. *If G is a dismantlable graph and $\gamma(G) > 1$, then $cc(G) = 2$.*

Theorem 2.15. *Let G be a cop-win graph with $\gamma(G) > 1$. If a corner u is added such that $N[u] \subseteq N[v]$, where v is some vertex in G , then the resulting graph G' satisfies $cc(G') = 2$.*

2.5. Corners and Dismantlable Graphs

Proof. Since $\gamma(G) > 1$ and $c(G) = 1$, then $cc(G) = 2$ as shown in Corollary 2.3. Add a corner u such that $N[u] \subseteq N[v]$ resulting in a new graph G' . Note that all the neighbours of u are adjacent to a vertex v in G . The two cops will play the same strategy \mathcal{S} that they played in G . If the robber stays in G , then he will be captured. If the robber goes to the newly added vertex u , the cops will assume that the robber is on v and follow the strategy \mathcal{S} accordingly. The strategy on G will have both cops capture the robber on v (or his image in this case), or otherwise, the robber would attack the lone cop. Once both cops are on v , and since $N[u] \subseteq N[v]$, the robber cannot escape and will be captured in the following step. (See Figure 2.5.1.) \square

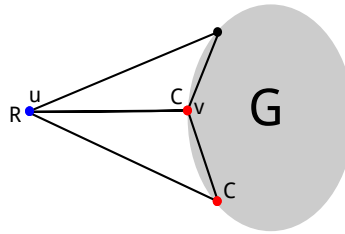


Figure 2.5.1: The robber will be caught in the following move.

Theorem 2.15 reinforces our finding in Theorem 2.1. Adding a corner to a cop-win graph will result in G' being cop-win as well which means that $cc(G')$ equals one or two. Theorem 2.3 shows that it cannot be one unless $\gamma(G) = 1$.

2.6 Joined Graphs

Define a graph G'' as the resultant graph of connecting two graphs G and G' at one vertex. Let $v_1 \in V(G)$ and $v_2 \in V(G')$ be the two vertices that will superimpose to create a new vertex u . We construct the *joined* graph by adding a new vertex u that we then join to every neighbour of v_1 and v_2 . That is, $N(u) = N(v_1) \cup N(v_2)$. Vertices v_1 and v_2 are then deleted along with all the edges previously incident to them. An example is shown in Figure 2.6.1.

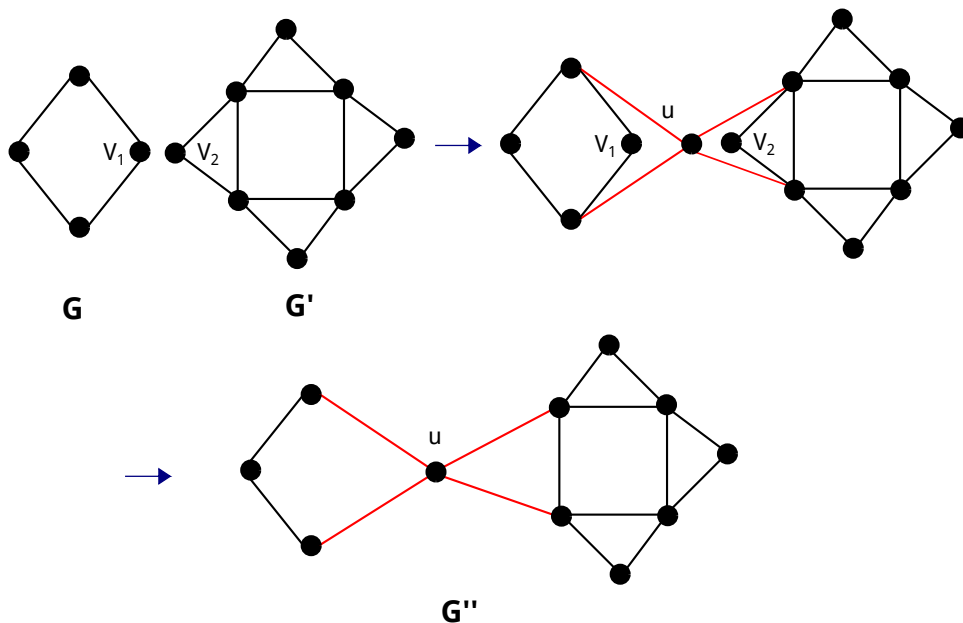


Figure 2.6.1: Constructing a joined graph.

For joined graphs, Theorem 2.16 shows how the cop number and cc-number relate between the initial graphs and the joined graph.

2.6. Joined Graphs

Theorem 2.16. *If $c(G) = cc(G) > 1$, $c(G) > c(G')$ and $cc(G) > cc(G')$, then*

$$c(G'') = cc(G'') = c(G) = cc(G).$$

Proof. We need to prove that $c(G'') = c(G)$ and $cc(G'') = cc(G)$. We start with the regular game, where we place $c(G)$ cops in G according to a winning strategy there. The vertex u is the vertex that connects the two graphs G and G' . Let $c(G)$ -many cops use their winning strategy to catch the robber in G . If the robber stays in G , then he will eventually be caught. The robber must move to G' to avoid capture. Since $c(G) > c(G')$, $c(G')$ -many cops will follow the robber into G' leaving at least one cop on u . The robber will be caught in G' . If he attempts to go back to G , then the cops who were left on u will catch him.

An analogous approach is used to prove that $cc(G'') = cc(G)$ in the cc-game. We start in G and capture the robber if he stays in G . If he chooses to go to G' , then we move all the cops to u first to force the robber to leave any vertex adjacent to u . We then leave one cop on u and the rest will go into G' . The robber will ultimately be caught by the $cc(G) - 1$ cops (as $cc(G) - 1 \geq cc(G')$), or by the lone cop if he attempts to go back to G through one of the vertices adjacent to u . \square

Chapter 3

Planar and Product Graphs

3.1 Introduction

In Chapter 2, we introduced the cc-game and presented some of the results and proofs we have found. In this chapter, we continue exploring this game in more complex families of graphs such as planar graphs and product graphs. We also look at a classification of a family of graphs with cc-number equal to 2.

3.2 Planar and Outerplanar Graphs

Planar graphs are one of the most studied graphs in graph theory. A graph G is called *planar* if it can be embedded in a plane without two of its edges crossing. A generalization of planar graphs are graphs which can be drawn

3.2. Planar and Outerplanar Graphs

on a surface of a given *genus*. Planar graphs, therefore, have graph genus 0, since the plane is a surface of genus 0. For more on genus, see Chapter 4 of [5].

Cycles, K_3 and K_4 are planar graphs are planar whereas K_5 and $K_{3,3}$ are not planar. Figure 3.2.1 shows some planar graphs.

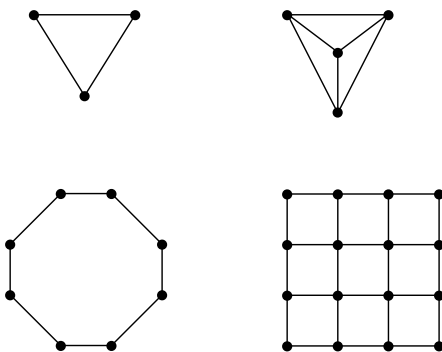


Figure 3.2.1: Clockwise from top left: K_3 , K_4 , grids, and cycles are planar graphs.

Aigner and Fromme studied the Cops and Robbers game on planar graphs and found that planar graphs require at most three cops to catch the robber; see [1]. An alternative proof of Theorem 3.1 was given in [5].

Theorem 3.1 ([1]). *If G is a planar graph, then $c(G) \leq 3$.*

3.2.1 Outerplanar Graphs

Before we set out to study the cc-game on planar graphs, we study a simpler but related class of graphs. *Outerplanar* graphs are graphs that satisfy the following three conditions; see [5].

3.2. Planar and Outerplanar Graphs

1. Every vertex lies on a circle.
2. Every edge of G either joins two consecutive vertices around the circle or is a chord across the circle.
3. If two chords intersect, then they do so at a vertex.

Figure 3.2.2 shows an outerplanar graph.

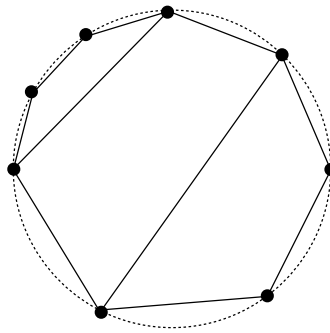


Figure 3.2.2: An outerplanar graph.

Maximal outerplanar graphs are outerplanar graphs that cannot have any additional edges added to them while preserving outerplanarity. Every maximal outerplanar graph with n vertices has exactly $2n - 3$ edges. Figure 3.2.3 shows a maximal outerplanar graph.

3.2. Planar and Outerplanar Graphs

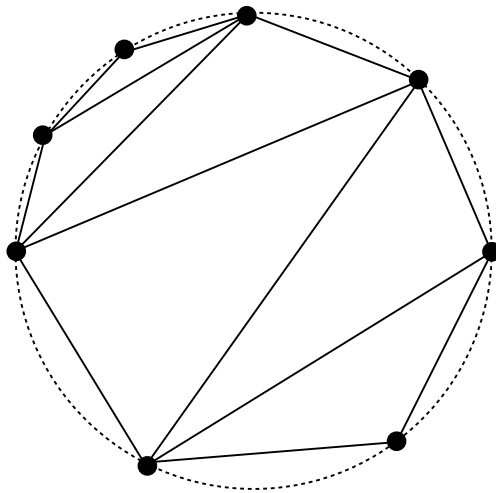


Figure 3.2.3: A maximal outerplanar graph.

Outerplanar graphs can be cop-win graphs and an example of that is a maximal outerplanar graph. A cycle, on the other hand, with $n \geq 7$ is an outerplanar graph with $cc(G) = 3$ and $c(G) = 2$. Clarke proved Theorem 3.2 in her doctoral thesis.

Theorem 3.2 ([6]). *If G is outerplanar, then $c(G) \leq 2$.*

Before we proceed, we define cut vertices. A vertex is called a *cut* vertex if removing it results in increasing the number of connected components in the graph. The proof of the following theorem relies on arguments analogous to those found in [5,6]. Note that Theorem 3.3 verifies Conjecture 2.2 in the case of outerplanar graphs.

Theorem 3.3. *If G is outerplanar, then $cc(G) \leq 3$.*

Proof. We need to show that in any outerplanar graph G in the cc-game, three cops will suffice to win. Assume first that G has no cut vertices. Order

3.2. Planar and Outerplanar Graphs

the vertices around the circle as $\{v_0, v_1, \dots, v_n\}$. Suppose that for a given i that v_i is not adjacent to v_{i+1} . We can renumber the subscripts so that $i = 0$. Since G is connected and the degree of v_0 is at least two, let v_j be the vertex of least index which is adjacent to v_0 . The edge v_0v_j prevents any vertex in $\{v_1, v_2, \dots, v_{j-1}\}$ from being adjacent to any vertex of $\{v_{j+1}, v_{j+2}, \dots, v_{n-1}\}$ since this would create a chord, and no vertex of $\{v_1, v_2, \dots, v_{j-1}\}$ is adjacent to v_0 by assumption. Therefore, v_j is a cut vertex, which is a contradiction. Hence, we may assume that for all i , v_i is adjacent to both v_{i-1} and v_{i+1} , with subscripts taken modulo n .

If the embedding contains no chords, then it is a cycle and three cops suffice (See Theorem 2.6) to capture the robber. Let a_0, a_1, \dots, a_k be the vertices of degree at least 3 in order around the circle. Note that vertices on the cycle between a_i and a_{i+1} are of degree 2 and so the path between a_i and a_{i+1} is well-defined.

Place the cops C_1, C_2 , and C_3 on the vertex v_0 . If v_0 has degree 2, then it is on a path between a_0 and a_k (renumbering the a_i if necessary); if it has degree 3 or more, then renumber so that $a_0 = v_0$. In either case, we can move C_1, C_2 , and C_3 to a_0 and a_k so that the robber is not on the path between these two vertices. We move two cops to a_0 or to a_k depending which one has a shorter distance to the robber.

We now assume more generally that C_1 and C_2 are on a_i and C_3 is on a_j for some $i < j$, respectively, and that the robber is not on and cannot move to any vertex in $\{v_p, v_{p+1}, \dots, v_0, \dots, v_q\}$, where $v_p = a_j$ and $v_q = a_i$. That

3.2. Planar and Outerplanar Graphs

is, every path from the robber to a vertex in $\{v_p, v_{p+1}, \dots, v_0, \dots, v_q\}$ passes through a_i or a_j . Such an area is the *cop territory*. The idea of the proof now is to show that the cops can increase the cop territory, so that it is eventually all of G and the cops win. We need not worry about C_3 being a lone cop on a_j yet. One cop is enough to stop the robber from moving to be at distance 1 with the cop in the cc-game and the robber will not be able to go to $\{v_p, v_{p+1}, \dots, v_0, \dots, v_q\}$. See Figure 3.2.4.

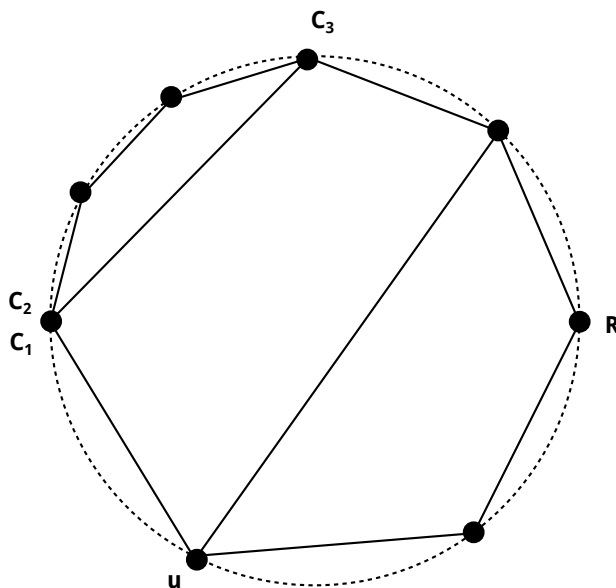


Figure 3.2.4: C_1 and C_2 move to u to expand the cop territory.

Suppose a_i has a chord to a vertex in the robber territory. Let a_r be a vertex adjacent to a_i which is closest to a_j . If the robber is on the arc of the circle from a_i to a_r , then he cannot move off that arc if C_1 on a_i does not move. Therefore, C_2 moves to a_j and then along with C_3 can be moved to a_r and the cop territory has increased. If the robber is between a_j and a_r , then

3.2. Planar and Outerplanar Graphs

C_1 and C_2 move from a_i to a_r again increasing the cop territory. A similar analysis holds for a_j . Hence, the only case to consider is when neither a_i nor a_j have an interior edge to the robber territory. In this case, the only paths to the cop territory from the robber are the ones along the cycle incident to a_i and a_j . That is, every path from the robber to the cop territory passes through a_{i+1} or a_j . Hence, moving C_1 and C_2 along the path from a_i to a_{i+1} does not allow the robber to move into the cop territory, and the cop territory has increased. We will refer to this as the *no-cut-vertex* strategy. The key difference between this strategy and the one in the regular Cops and Robbers game is that whenever a cop has to move into the robber territory, another cop joins her. The third cop stays on guard on the other vertex (that is, passes on a_i or a_j).

Now suppose that G has at least one cut vertex. Let

$$B(G) = \{G_1, G_2, \dots, G_m\}$$

be the set of maximal induced subgraphs of G such that each G_i itself has no cut vertices. Note that each G_i will contain a vertex which is a cut vertex of G , and each G_i has at least two vertices.

We can retract G onto G_i , for any i , by the mapping described as follows. Let $x \in V(G_i)$ and x be a cut vertex of G . All vertices of G that are disconnected from G_i by the deletion of x are mapped to x . Vertices of G_i are mapped to themselves. Let G_i denote this retract. Since G_i is a

3.2. Planar and Outerplanar Graphs

subgraph of an outerplanar graph, G_i is also outerplanar. Fix an embedding for each G_i . Choose some G_i and place the three cops on two vertices in G_i as in the case with no cut vertices. We now use the following strategy on G_i . Employing the strategy of the case with no cut vertices, the cops will capture the robbers image on G_i . Since $|V(G_i)| \geq 2$, at least one more vertex and one more element of $B(G)$ is added to the cop territory. If the robber is actually on G_i , then he has been caught. If not, then the cops have captured the robbers shadow on a vertex x whose deletion separates G_i from the G_k where the robber presently resides. This cut vertex also lies in some G_j that either contains the robber (that is, $j = k$) or contains a cut vertex $y = x$ whose deletion separates G_j from G_k (and j is unique). Fix an outerplanar embedding of G_j . The cops now execute the no-cut-vertex strategy on G_j . Hence, the cops eliminate the subgraphs in $B(G)$, and eventually they capture the robber (rather than just his image). At all times, when capturing the robber or his image, two cops will be moving together to expand the cops territory and one cop will pass. \square

We were unable to find a similar result for the cc-number on planar graphs. In particular, Conjecture 2.2 remains open in the case of general planar graphs. Theorem 3.1 in its proof relies heavily on the fact that isometric paths are 1-guardable in the Cops and Robbers game; a fact that does not apply to the cc-game. We suspect that the following conjecture is true but no solution has been found yet.

3.3. Graph Products

Conjecture 3.4. *If G is a planar graph, then $cc(G) \leq 4$.*

One of the planar graphs that such a conjecture remains open for is the planar projection of a certain geometric structure called the *buckminsterfullerene*. The graph corresponding to that structure is called the *truncated icosahedral graph*. Figure 3.2.5 shows both the geometric structure and its planar graph representation which has 60 vertices.

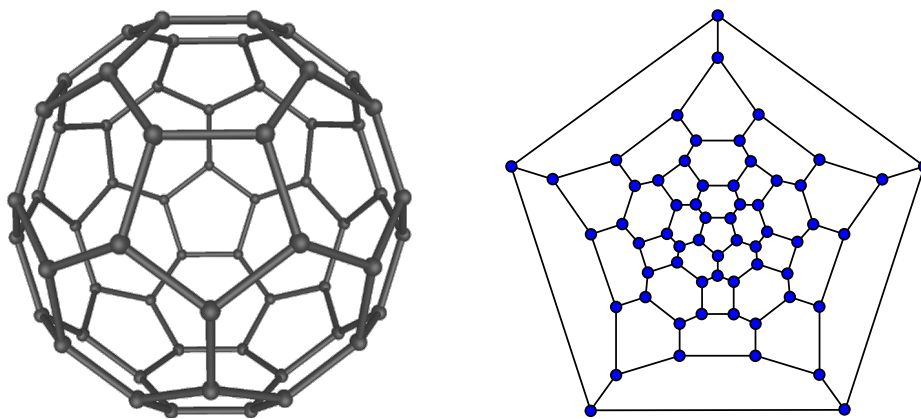


Figure 3.2.5: A buckminsterfullerene (left) and truncated icosahedral graph (right).

3.3 Graph Products

Graph Products are operations that take two graphs G_1 and G_2 and produce a graph H with the following properties [11]:

1. The vertex set $V(H) = V(G_1) \times V(G_2)$ is the Cartesian product of the two sets $V(G_1)$ and $V(G_2)$.

3.3. Graph Products

2. Two vertices in $V(H)$ are adjacent if their components satisfy certain conditions in G_1 and G_2 . These conditions vary between different types of products.

Graph products come in many types, each with different rules on adjacency between two vertices in the product graph. Some of the most studied product graphs are the categorical product, the strong product, the Cartesian product and the lexicographic product. In the *Cartesian* product, (u, v) and (x, y) are adjacent if $u = x$ and $vy \in E(H)$ or $ux \in E(G)$ and $v = y$. The *categorical* product is product in which a pair of vertices (u, v) and (x, y) are adjacent in the product graph if $ux \in E(G)$ and $vy \in E(H)$. The *lexicographic* product is a product where two vertices (u, v) and (x, y) are adjacent if $ux \in E(G)$, or $u = x$ and $vy \in E(H)$.

The game of Cops and Robbers has been studied, with varying degrees of success, on these products. The results can be found in Chapter 4 of [5]. We tried in our research to implement similar approaches in the cc-game and found some interesting results in the cc-game on strong products.

3.3.1 Strong Product

The *strong product* $G \boxtimes H$ of two graphs G and H is defined on the Cartesian product of the vertex sets of G and H , with two distinct vertices (u, v) and (x, y) being adjacent in $G \boxtimes H$ if $u = x$ and $vy \in E(H)$, or $ux \in E(G)$ and $v = y$, or $ux \in E(G)$ and $vy \in E(H)$.

3.3. Graph Products

This means that two vertices have to either be equal or adjacent in both factors of the product [11]. Figure 3.3.1 shows two graphs G and H with their strong product $G \boxtimes H$.

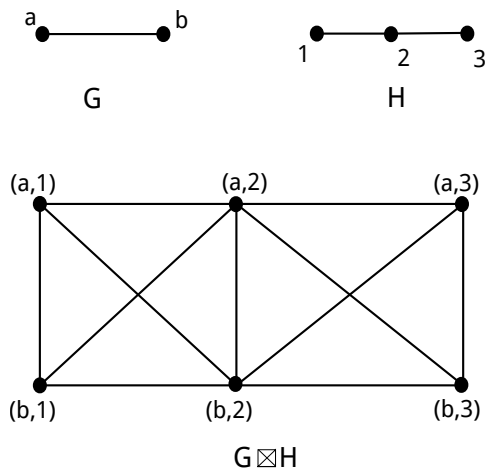


Figure 3.3.1: Strong product of G and H .

The first theorem we find is that strong product of two graphs with cc-number equal to 1 also has cc-number equal to 1.

Theorem 3.5. *If $cc(G) = 1$ and $cc(H) = 1$, then $cc(G \boxtimes H) = 1$.*

Proof. Consider two graphs G and H with $cc(G) = 1$ and $cc(H) = 1$. We know from Theorem 2.2 that both graphs must contain a universal vertex. Let u and v be the universal vertices in graphs G and H respectively. Consider the vertex (u, v) in the graph $G \boxtimes H$. The vertex (u, v) is joined to every vertex in $G \boxtimes H$ that has a neighbour of u in G or a neighbour of v in H . The vertex (u, v) is also adjacent to every vertex containing either u or v and neighbouring the other vertex of that pair (for example, vertices

3.3. Graph Products

(u_1, v) and (u, v_1) are connected to (u, v)). Hence, (u, v) is a universal vertex in $G \boxtimes H$, and so $cc(G \boxtimes H) = 1$ \square

If we have two graphs G and H , and $v \in H$, then we define $G.v$ as the induced subgraph in $G \boxtimes H$ with vertices $\{(u_1, v), \dots, (u_n, v)\}$, where $V(G) = \{u_1, \dots, u_n\}$. Since v is common as the second component of all those vertices in $G.v$, then $G.v$ is an isomorphic copy of G . Theorem 3.5 uses this concept to find the cc-number on strong products of any graph with a graph containing a universal vertex.

Theorem 3.6. *If $cc(G) = k \geq 2$ and $cc(H) = 1$, then $cc(G \boxtimes H) = k$.*

Proof. We start by placing k cops on some vertex (u_1, v) , where v is the universal vertex in H . Since we have k cops and $cc(G) = k$, we play the strategy of the k cops on $G.v$. This can be done on $G \boxtimes H$ because of the properties of the strong product. For example, if the strategy on G tells a cop to go from u_1 to u_2 and these two vertices are adjacent, the cop on $G \boxtimes H$ can move from (u_1, v) to (u_2, v) . The projection of the robber on $G.v$ will be caught. Once that happens, if the robber is on $G.v$, then he has been caught and not his image. If he is on some vertex $G.v_1$, then he must be adjacent to the cop who caught his image since v is connected to v_1 . The robber is caught in the following step. \square

Theorem 3.6 finds an upper bound on the cc-number of the strong product graph of any two graphs in terms of their cc-numbers.

3.3. Graph Products

Theorem 3.7. *If $cc(G) \geq 2$ and $cc(H) \geq 2$, then*

$$cc(G \boxtimes H) \leq cc(G) + cc(H) - 1.$$

Proof. We start with $cc(H)$ cops on a vertex (u, v) , where $u \in G$, $v \in H$ and u is one of the starting positions for one of the cops in G according to the winning strategy in that graph. We then place $cc(G) - 1$ cops on $G.v$ in the remaining starting positions for that strategy. We play on $G.v$ as if we are playing in G while assuming that the $cc(H)$ cops are one cop. Once the cops capture the robber, move $cc(G)$ cops to where the robber's image was caught and shadow his image on $G.v$. Since $cc(G) \geq 2$, then we will be left with at least $cc(H) - 1$ cops. Now we execute the strategy for H using $cc(G)$ and $cc(H) - 1$ cops. The $cc(G)$ cops move as one cop on the projection on H and shadow the image of the robber on G at the same time. This is possible because we are playing on the strong product where $cc(G)$ can move on G , H , or at both at the same time (see Figure 3.3.2). When the robber's image is caught on H while his image on G is shadowed by $cc(G)$, the robber is caught and the cops win. \square

3.4. Nearly Irreducible Vertices

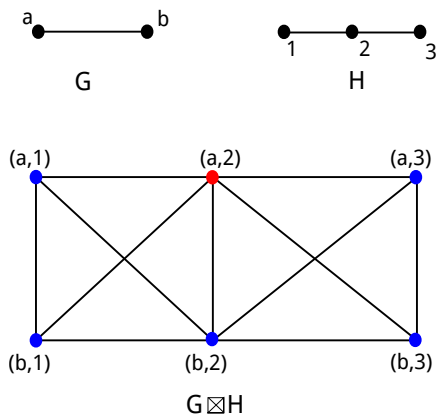


Figure 3.3.2: The cop (red) can in one move shadow the movement on G , H or both.

It remains an open problem to find bounds or values for the cc-number on other types of products such as the Cartesian product, categorical product, and lexicographic product.

3.4 Nearly Irreducible Vertices

In this section, we attempt to classify a family of graphs whose cc-number is less than or equal to two. We have already classified graphs with cc-number equal to 1 in Chapter 2. An important concept in this classification is a nearly irreducible vertex. In a graph G , a vertex $v \in G$ is *nearly irreducible* if there exists another vertex $u \in G$ such that $N(v) \subseteq N(u)$. That is, all the vertices adjacent to v are also adjacent to some other vertex u .

A *nearly irreducible ordering* in a graph G is an ordering (v_1, \dots, v_n) of the vertices of a graph G such that for each $i < n$, v_i is a nearly irreducible

3.4. Nearly Irreducible Vertices

vertex in $G_i = G - \{v_1, \dots, v_{i-1}\}$. Figure 3.4.1 shows a graph with an ordering of nearly irreducible vertices v_1 to v_9 .

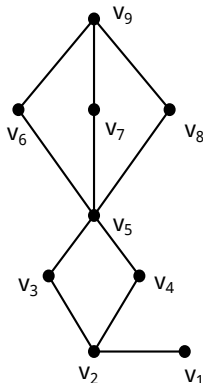


Figure 3.4.1: An ordering of nearly irreducible vertices v_1 to v_9 .

Theorem 3.8 proves that a certain class of graphs has cc-number at most 2. Before we proceed to its proof, we define a kite graph and a triangle-free graph. A *kite* is a graph in which a vertex v is nearly irreducible with respect to a vertex u and some vertex z exists that is adjacent to u but not to v (See Figure 3.4.2). This means if v is a nearly irreducible vertex in a kite-free graph, then $N(v) = N(u)$. A graph is *triangle-free* if it does not contain K_3 as a subgraph.

3.4. Nearly Irreducible Vertices

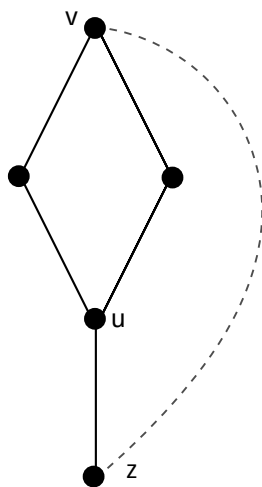


Figure 3.4.2: A kite graph.

Theorem 3.8. *If G is a triangle-free and kite-free graph, then $cc(G) \leq 2$ if and only if there exists an ordering of nearly irreducible vertices.*

Proof. We start by proving the reverse direction and we do so by induction. For $n = |V(G)|$, we have that $G = K_1$ and $cc(G) = 1 < 2$ and the result holds. Now we assume that for $|V(G)| = n$, the result holds and $cc(G) \leq 2$. For the inductive step, suppose that G is kite-free and triangle-free with $|V(G)| > 1$, and admits a nearly irreducible ordering. We show that $cc(G) \leq 2$. Let v be the last vertex in a nearly irreducible ordering of G . Hence, there is a vertex u such that $N(u) \subseteq N(v)$. As G is kite-free, we must have that $N(u) = N(v)$. The graph $G - v$ inherits the nearly irreducible ordering from G . Therefore, $G - v$ has cc-number at most 2 by inductive hypothesis. Let \mathcal{S} be the winning strategy for the two cops in G .

We fix the following retract,

3.4. Nearly Irreducible Vertices

$$f(x) = \begin{cases} u & \text{if } x = v, \\ x & \text{otherwise.} \end{cases}$$

Now we will play a similar strategy \mathcal{S}' in $G' = G - v$. If the robber goes to the newly added vertex v , then the robber's image $f(R)$ will be on the vertex u as per the retract we defined. Since $cc(G) \leq 2$, then the robber will be caught by the cops if he was to remain in G . Using \mathcal{S}' , the robber moves to v and the cops move to u . Since there are no triangles, then v is not adjacent to u and one cop cannot be attacked by the robber directly from v . The cops will catch the robber's image on u . The robber cannot leave v and $N(v)$ without passing through u and therefore remains there. Since $N(v) \subseteq N(u)$, the robber will remain on v to avoid being caught. One cop, say C_1 , will move to one of the common neighbours of u and v while the other cop, say C_2 , will stay on u . The robber can pass and will be caught by C_1 in the following step or he could move to one of the vertices adjacent to both u and v and be caught by C_2 in the following step. Therefore, $cc(G') \leq 2$, and the reverse direction follows by induction.

We will prove the forward direction using induction. For the base step, consider a graph G with $cc(G) \leq 2$. We consider the last move by the robber after which he is supposed to be caught according to the winning strategy of the cops. Suppose that the cops apply their winning strategy, and consider the second-to-last move of the cops before the robber is captured. We must have that some cop, say C_1 , on a vertex that is joined to the robber (as the

3.4. Nearly Irreducible Vertices

robber is captured in the next round). As there are no triangles, C_1 is not on a vertex that is adjacent to any other neighbours of R . To avoid the robber either attacking C_1 or escaping to some neighbour, the remaining cop C_2 must be on a vertex that is joined to the vertex C_1 is on (or possibly C_1 and C_2 are on the same vertex) and to each neighbour of R . Hence, the position of the robber is a nearly irreducible vertex. Now since $cc(G) \leq 2$ and we already proved that it has at least one nearly irreducible vertex, delete a fixed nearly irreducible vertex which we call v in G with $N(u) = N(v)$ for some u . Note that here we use that G is kite-free, so we have the equality of the two neighbour sets, rather than just $N(u) \subseteq N(v)$. All the vertices of $G - v$ remain unchanged except for the neighbours of v . We play the same strategy on $G - v$ as the one we would have played on G . If R was to go to v , then we play in $G - v$ as if he is on u . This means that the robber will gain no extra moves by deleting $G - v$. Since the winning strategy in G allows the cops to catch the robber on v , the same strategy can be used to catch the robber on u or one of the neighbours of u and v . Therefore, $cc(G - v) \leq 2$. We may now use induction on $|V(G)|$ to obtain a nearly irreducible ordering for G . Hence, if $cc(G) \leq 2$, then there is an ordering of nearly irreducible vertices. □

Chapter 4

Miscellaneous Results on the cc-Game

4.1 Introduction

In this chapter, we will be revisiting three concepts that were mentioned in Chapter 1. The concepts of complexity, random graphs, and infinite graphs will be explained in further detail and their relevance to the cc-game will be explored. We will mention some of the results on the three topics in regards to the game of Cops and Robbers as well as the results we found in the cc-game. These results are accompanied by proofs and explanatory figures.

4.2 Complexity

We introduced the computational complexity of algorithms in Chapter 1, and we will be exploring this subject further in this chapter. For our applications(although this will not be used explicitly), graphs are represented as data structures such as adjacency lists or adjacency matrices. The two major complexity classes for algorithms used in graph decision problems are **P** and **NP**. **P** is the class of problems with polynomial time complexity. **NP** is the class of problems decided by non-deterministic polynomial algorithms. The relation between **P** and **NP** is not entirely clear. One of the deepest conjectures in mathematics is if the classes **P** and **NP** are equal. It is clear that complexity class $\mathbf{P} \subseteq \mathbf{NP}$ but someone has yet to prove that there is a problem in **NP** that is not in **P**.

Not all problems in **NP** are of equal complexity. We call a problem **NP-hard** if a polynomial-time algorithm for it, if found, would mean that there exists a polynomial-time algorithm for every problem in **NP**. In other words, finding a polynomial-time algorithm for an **NP-hard** problem would imply that $\mathbf{P}=\mathbf{NP}$.

An **NP-complete** problem is a problem that belongs to both classes, **NP** and **NP-hard**. This, again, means that if a polynomial-time algorithm is to be found for a problem that is **NP-complete**, then we would have proved that $\mathbf{P}=\mathbf{NP}$. Note that there exist problems which are **NP-hard** and not in **NP** (such as the halting problem). Figure 4.2.1 shows the complexity classes

4.2. Complexity

and the conjectured relations between them. Further readings and references on these classes and the research on them can be found in [13] and [17].

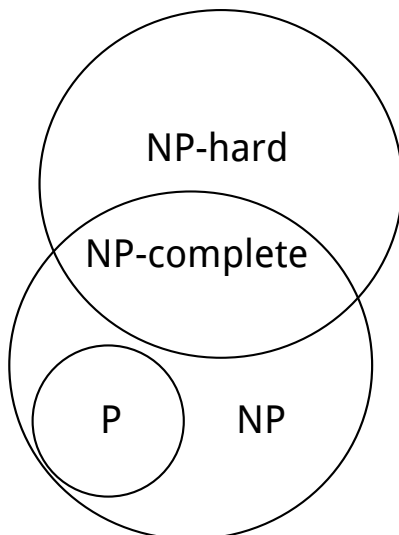


Figure 4.2.1: Four complexity classes, assuming $P \neq NP$.

4.2.1 Cops, Robbers, and the cc-Game

The work that has been done on algorithms and their complexities in the game of Cops and Robbers is fairly limited. We will mention the problems that have been studied and some of the results that have been achieved. More on this topic can be found in Chapter 5 of [5], and in [8].

Two of the problems that have been studied are the following.

k -COP NUMBER: Given a positive integer k , for the input (G, k) is $c(G) \leq k$?

k -FIXED COP NUMBER: Given a fixed positive integer k , for the input

4.2. Complexity

G is $c(G) \leq k$?

The difference between the two is that in the latter the cop number k is a fixed positive integer that does not change based on the input G . In the first problem, k can be thought of as a function of n , the number of vertices of G . For example, an instance of this is determining whether $c(G) \leq \sqrt{n}$.

We will not give the proofs of the following two theorems about the mentioned problems but only use their results in the cc-game. Theorem 4.1 shows that the problem with a fixed cop number can be solved with a polynomial-time algorithm. It first appeared in [2] and was later reproved in [10].

Theorem 4.1 ([2],[10]). *The problem k -FIXED COP NUMBER is in \mathbf{P} .*

Theorem 4.2 shows that the problem with cop number k that is a function of the number of vertices n of a graph G is an \mathbf{NP} -hard problem.

Theorem 4.2 ([8]). *The problem k -COP NUMBER is in \mathbf{NP} -hard.*

We consider analogous results in the cc-game. The problem we were working on can be defined as follows.

k -CC NUMBER: Given a positive integer k , for the input (G, k) is $cc(G) \leq k$?

This means that k can be a function of n in G . Theorem 4.3 shows that the problem k -CC NUMBER is \mathbf{NP} -hard.

We first define *approximation ratios* and *approximation functions*. Approximation algorithms calculate solutions that are close to the optimal solutions and are used when an optimal solution cannot be found. An *approx-*

4.3. Random Graphs

imation ratio $f(n)$ for an input n of an algorithm is a multiplicative factor which is a ratio of the cost of a solution C' to the cost of the optimal solution C . A $f(n)$ -*approximation-algorithm* is an algorithm that computes a solution within an approximation ratio $f(n)$ [16].

Theorem 4.3. *Computing $cc(G)$ is **NP**-hard.*

Proof. It was proved in [8] that there is a constant $d > 0$ such that there is no polynomial time algorithm to approximate $c(G)$ within a multiplicative factor $f(n) = d \log n$, unless $\mathbf{P} = \mathbf{NP}$.

By Theorem 2.1, we know that $c(G) \leq cc(G) \leq 2c(G)$. Thus, by Theorems 2.1 and 4.2, we have that $cc(G)$ will fall within the multiplicative logarithmic factor of $c(G)$ where computing it is **NP**-hard. \square

Further discussion of the algorithmic aspects of the cc-game is beyond the scope of this thesis. However, a number of open problems remain to be solved. The most important of those problems is whether the computing the cop number, or the cc-number of a graph G is in **NP**.

4.3 Random Graphs

In Chapter 1, we defined random graphs and gave some background on them. We continue our work on these graphs in this section and highlight some of the results in the game of Cops and Robbers and the cc-game when played on random graphs.

4.3. Random Graphs

A random graph, denoted $G(n, p)$, is a graph with n vertices in which the probability of any two vertices being adjacent is p . Further, edges are independently chosen. There are two types of random graphs. The simpler type which we mainly study here is a random graph where p is constant. For example, the random graph that was shown in Chapter 1 is an instance of $G(10, 0.5)$ which means that it has 10 vertices and the probability of two vertices being adjacent is $p = \frac{1}{2}$. The more general type of random graphs is one in which the probability p is a function of the number of vertices n ; that is, $p = p(n)$.

The domination number of a graph G is an elementary upper bound for the cop number and the cc-number. It is definitely not the best upper bound in general graphs. A path P_n , for example, has $c(P_n) = 1$ and $cc(P_n) = 2$ for all n . However, the domination number for a path is $\gamma(P_n) = \lceil \frac{n}{3} \rceil$. In random graphs, the cop number, cc-number, and the domination number are of the same order as we will show.

Before we move on to the theorems and results, we define the term asymptotically almost surely that will be used later on. We say that an event holds *asymptotically almost surely* (or *a.a.s.*) if it is true with probability tending to 1 as $n \rightarrow \infty$. Hence, in a random graph $G(n, p)$, a condition holds a.a.s. if as the number of vertices tends to infinity, the probability of such a event being true tends to 1.

Dreyer was the first to study the domination number in a random graph in his doctoral thesis [7]. The result was asymptotic bounds on the domination

4.3. Random Graphs

number of a random graph $G(n, p)$ with p a fixed constant. If $p \in (0, 1)$, then we define

$$\mathbb{L}_n = \log_{\frac{1}{1-p}} n.$$

Theorem 4.4 ([7]). *Let $0 < p < 1$ be fixed. For every real $\epsilon > 0$, a.a.s.*

$$(1 - \epsilon)\mathbb{L}_n \leq \gamma(G(n, p)) \leq (1 + \epsilon)\mathbb{L}_n.$$

In particular,

$$\gamma(G(n, p)) = \Theta(\log n).$$

The cop number on a random graph was studied in [4] and a similar result was found to the one in Theorem 4.5.

Theorem 4.5 ([4]). *Let $0 < p < 1$ be fixed. For every real $\epsilon > 0$, a.a.s.*

$$(1 - \epsilon)\mathbb{L}_n \leq c(G(n, p)) \leq (1 + \epsilon)\mathbb{L}_n.$$

In particular,

$$c(G(n, p)) = \Theta(\log n).$$

We used these two theorems and Theorem 2.1 to deduce Corollary 4.6.

Corollary 4.6. *Let $0 < p < 1$ be fixed. For every real $\epsilon > 0$, a.a.s.*

$$(1 - \epsilon)\mathbb{L}_n \leq cc(G(n, p)) \leq (1 + \epsilon)\mathbb{L}_n.$$

In particular,

$$cc(G(n, p)) = \Theta(\log n).$$

4.4. Infinite Graphs

Proof. By Theorems 4.4 and 4.5 we have that

$$(1 - \epsilon)\mathbb{L}_n \leq \gamma(G(n, p)) \leq (1 + \epsilon)\mathbb{L}_n, \quad (4.1)$$

$$(1 - \epsilon)\mathbb{L}_n \leq c(G(n, p)) \leq (1 + \epsilon)\mathbb{L}_n. \quad (4.2)$$

Recall from Theorem 2.1 that for any graph G :

$$c(G) \leq cc(G) \leq \min\{\gamma(G), 2c(G)\}. \quad (4.3)$$

Therefore, by Equations (4.1)(or by(4.2)) and (4.3), we have that

$$(1 - \epsilon)\mathbb{L}_n \leq cc(G(n, p)) \leq (1 + \epsilon)\mathbb{L}_n,$$

so

$$cc(G(n, p)) = \Theta(\log n). \quad \square$$

An open problem is determining the cc-number on a random graph $G(n, p)$, where $p = p(n)$.

4.4 Infinite Graphs

So far, in all the graphs we have studied including the random graph, there was a finite value for the number of vertices n . An *infinite graph* is a graph in which the number of vertices is infinite. These graphs may exhibit special properties that are different from finite graphs, and this is also true for the Cops and Robbers games. For example, a tree with a finite number of vertices

4.4. Infinite Graphs

has cop number equal to one and cc-number equal to one or two. A tree that contains a *ray*, which is a one-way infinite path, would need an infinite number of cops and cc-cops. Figure 4.4.1 shows a tree that contains a ray.

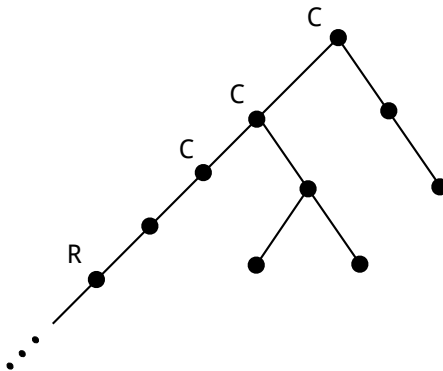


Figure 4.4.1: A finite number of cops cannot catch the robber.

The concept of random graphs and a countably infinite set of vertices can be combined to give the *infinite random graph* or the *Rado graph*, written \mathcal{R} . The graph \mathcal{R} is defined as the probability space $G(\mathbb{N}, p)$ which is graphs whose vertex set are the non-negative integers (countably infinite), and the probability of adjacency between any pair of vertices is p . We will refer to the Rado graph by the symbol \mathcal{R} .

An important property that we will need in this section is the e.c. property of graphs. We say a graph is *existentially closed* or *e.c.* if for all finite disjoint sets of vertices A and B , there is a vertex u joined to all vertices of A and no vertices of B . We say that u is *correctly joined* to A and B . Figure 4.4.2 depicts the e.c. property.

4.4. Infinite Graphs

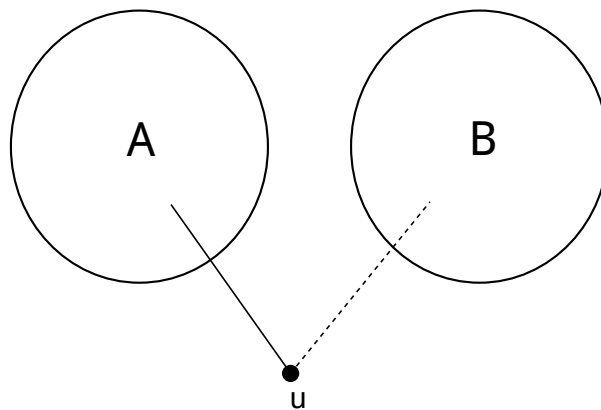


Figure 4.4.2: u is correctly joined to A and B .

A weaker version of the e.c. property is when a graph is *strongly n -e.c.* We say a graph G is strongly n -e.c. if for all disjoint sets A and B in G with $|A| \leq n$, there is a vertex u correctly joined to A and B . Note that a graph is e.c. if and only if it is strongly n -e.c. for all $n \geq 1$.

For all the theorems and proofs that follow in this chapter, we will be working on infinite graphs that are countable. The set of vertices for such graphs is \mathbb{N} .

Theorem 4.7 shows two results. These two results combined mean that the Rado graph is the unique e.c. countable graph.

Theorem 4.7 ([5]). *With probability 1, the following is true.*

- (1) *The graph $G(\mathbb{N}, p)$ is e.c.*
- (2) *The graph $G(\mathbb{N}, p)$ is unique up to isomorphism.*

4.4. Infinite Graphs

4.4.1 Chains and Cop Density

The cop number of \mathcal{R} is infinite, and this can be shown directly. Let the robber occupy at some round t a vertex u and the cops occupy a finite set of vertices X . By the e.c. property, there is a vertex v joined to u and not joined or equal to any vertex of X . The robber moves to v and avoids being caught. This holds for all vertices of \mathcal{R} . See Figure 4.4.3.

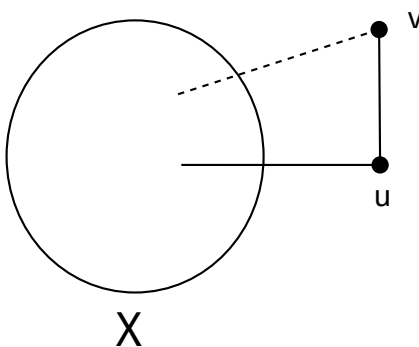


Figure 4.4.3: A robber escaping finitely many cops in R .

When working with infinite graphs that are countable, we know that such graphs are made up of the limits of chains of finite graphs. Studying these chains provides insight into some properties that are not possible to study for the graph as a whole.

The *cop density* of a finite graph, denoted $D_c(G)$, is a property that was introduced in [4]. The cop density is the ratio of cops needed in a finite graph to the number of vertices in that graph. That is,

$$D_c(G) = \frac{c(G)}{|V(G)|}.$$

4.4. Infinite Graphs

Note that since the cop number is well defined and bounded, we have $D_c(G) \in [0, 1]$. The same concept can be extended to infinite graphs. Every countably infinite graph is the limit of a chain of finite graphs, and there are infinite chains with the same limit. Suppose that $G = \lim_{n \rightarrow \infty} G_n$, where $\mathcal{C} = (G_n : n \in \mathbb{N})$ is a chain of induced subgraphs of G . Define

$$D(G, \mathcal{C}) = \lim_{n \rightarrow \infty} D_c(G_n)$$

as the cop density of G relative to \mathcal{C} , assuming the limit exists. We simply refer to this as the cop density of G . We define a similar term that describes the ratio of the cc-number in a graph to the number of vertices in that graph. We call this value $D_{cc}(G)$, where

$$D_{cc}(G) = \frac{cc(G)}{|V(G)|}.$$

The limit $D_{cc}(G, \mathcal{C})$ is defined analogously as $D(G, \mathcal{C})$.

For example, a cycle C_{10} has $cc(C_{10}) = 3$ which means its cop density is

$$D_{cc}(C_{10}) = \frac{cc(G)}{|V(G)|} = \frac{3}{10} = 0.3.$$

Now let G be a disjoint union of infinitely many cycles with 10 vertices, and let G_n be the disjoint union of the first n of those cycles. If the chain \mathcal{C} is G_n , then the density $D_{cc}(G, \mathcal{C})$ is equal to the limit of the density of G_n which is a constant and is equal to 0.3.

4.4. Infinite Graphs

Theorem 4.8 was proved in [4] and it shows that in an infinite graph G , a chain can be found with cop density equal to any real number in $[0, 1]$. This is true for both the cop number and the cc-number. The proof below is analogous to the approach in [4] and Chapter 7 in [5] with some added details on why the same proof works for the cc-game.

Theorem 4.8. *If G is strongly 1-e.c., then for all $r \in [0, 1]$, there is a chain \mathcal{C} in G such that $D_{cc}(G, \mathcal{C}) = r$.*

Proof. Let $(p_n : n \in \mathbb{N})$ be a sequence of rationals in $[0, 1]$ such that

$$\lim_{n \rightarrow \infty} p_n = r,$$

with $p_0 = 1$. For example, if r has a decimal expansion $0.r_1r_2r_3\dots$, then we can choose $p_i = 0.r_1r_2\dots r_i$. We construct a chain $\mathcal{C} = (G_n : n \in \mathbb{N})$ in G such that $G = \lim_{n \rightarrow \infty} G_n$, and with the property that $D_c(G_n) = p_n$. Enumerate $V(G)$ as $\{x_n : n \in \mathbb{N}\}$. We proceed inductively on n . For $n = 0$, let G_0 be the subgraph induced by x_0 . Then

$$\frac{cc(G_0)}{|V(G_0)|} = 1 = p_0.$$

Fix $n \geq 1$. Suppose the induction hypothesis holds for all $k \leq n$, and let $p_{n+1} = \frac{a}{b}$, where a and b are positive integers. Further suppose for an inductive hypothesis that $\{x_0, \dots, x_n\} \subseteq V(G_n)$. Without loss of generality, as $r \in [0, 1]$ we may assume $a < b$, and $\gcd(a, b) = 1$. We add vertices to G_n in

4.4. Infinite Graphs

stages. Define G'_{n+1} to be the graph induced by $V(G_n) \cup \{x_{n+1}\}$. Suppose that $c(G'_{n+1}) = a'$ and also that $|V(G'_{n+1})| = b'$. If $\frac{a'}{b'} = \frac{a}{b}$, then let $G_{n+1} = G'_{n+1}$. Otherwise, we add some new vertices to adjust the density $D_c(G'_{n+1})$. Each time an isolated vertex is added to a graph, the cc-number increases by one. That cop would have to start on that isolated vertex. Adding an end-vertex to a graph does not change the cop number or the cc-number. We may assume that $\frac{a'}{b'} < \frac{a}{b}$ by adding an appropriate number of end-vertices. In this way, b' will become larger, while a' will remain unchanged. We may add an arbitrary finite number of isolated vertices and end-vertices to G'_{n+1} by the strongly 1-e.c. property. We add x isolated vertices and y end-vertices to G'_{n+1} to form G_{n+1} so that

$$D_{cc}(G_{n+1}) = \frac{cc(G_{n+1})}{|V(G_{n+1})|} = \frac{a}{b}.$$

This is possible if we can solve the equation

$$\frac{a}{b} = \frac{a' + x}{b' + x + y},$$

which is equivalent to

$$(b - a)x - ay = ab' - a'b. \tag{4.4}$$

Note that $ab' - a'b > 0$, since otherwise $ab' \leq a'b$ which is contrary to the

4.4. Infinite Graphs

hypothesis. Hence, we obtain a linear Diophantine equation

$$cx + dy = e,$$

where $c = b - a > 0$, $d = -a < 0$, and $e = ab' - a'b > 0$. As $\gcd(b - a, -a) = \gcd(a, b) = 1$, then (4.4) has infinitely many solutions. The general integer solution of (4.4) is

$$x = x_0 - at, y = y_0 - (b - a)t, \quad (4.5)$$

where (x_0, y_0) is a particular fixed solution, and t is an integer. (For example, we may take $(x_0, y_0) = (-a, a - b)$.) As the coefficients of t in (4.5) are both negative, we may choose an appropriate $t < 0$ to ensure an integer solution of (4.4) (x, y) with $x, y \geq 0$. This completes the induction step in constructing G_{n+1} . As $\{x_0, \dots, x_n\} \subseteq V(G_n)$ for all $n \in \mathbb{N}$, we have that $\mathcal{C} = G_n : n \in \mathbb{N}$ is a full chain for G . Further,

$$D_{cc}(G, \mathcal{C}) = \lim_{n \rightarrow \infty} p_n = r. \quad \square$$

Now since the infinite random graph \mathcal{R} is e.c. (and so is strongly 1-e.c.) we have the following corollary.

Corollary 4.9. *For all $r \in [0, 1]$, there is a chain \mathcal{C} in \mathcal{R} such that $D_{cc}(\mathcal{R}, \mathcal{C}) = r$.*

Chapter 5

Results and Open Problems

5.1 Introduction

In this chapter, we will summarize the results and theorems that we have found. We will then mention the open problems and conjectures that remain unsolved.

5.2 Results and Theorems

We started by finding results that relate the cc-number of a graph to its cop number. In Chapter 2, we found an upper and lower bound on $cc(G)$ in terms of $c(G)$ and this later allowed us to utilize some results of the game of Cops and Robbers in the cc-game. We then explored and computed the cc-numbers of some special graphs such as cycles in Chapter 2 and outerplanar

5.3. Open Problems and Conjectures

graphs in Chapter 3. One of the more significant results in this thesis was classifying graphs by their cc -number. Such classifications were found for graphs with $cc(G) = 1$ in Chapter 2 and a family of graphs with $cc(G) = 2$ in Chapter 3. Product graphs were explored in Chapter 3 and results were found in the cc -game on strong products such as the upper bound of the cc -number of the strong product of two graphs. In Chapter 4, we investigated the complexity of computing $cc(G)$. Some corollaries and results were then found when playing the cc -game on random graphs. Finally in the same chapter, we considered infinite graph as well as the Rado graph, and the implications of their properties on the cc -game.

5.3 Open Problems and Conjectures

Throughout this thesis we have mentioned some of the problems that remain unsolved. We will summarize those here. In Theorem 2.1, we established that

$$c(G) \leq cc(G) \leq 2c(G).$$

By observing the strategies for the cc -game on special graphs, we noticed that we seem to only need one more cop to play the cc -game as opposed to the original game. The following conjecture remains unsolved, and it is arguably the main open problem in this area of research.

Conjecture 5.1. *For a graph G , we have $cc(G) \leq c(G) + 1$.*

5.3. Open Problems and Conjectures

We studied planar and outerplanar graphs in Chapter 3 and while we found results on the cc-game on the latter, an upper bound on the cc-number on planar graphs remains to be found. Note that if G is planar, then $c(G) \leq 3$. Conjecture 5.2 however, remains open.

Conjecture 5.2. *If G is a planar graph, then $cc(G) \leq 4$.*

In Chapter 3, we classified a family of graphs with $cc(G) = 2$. It remains unknown if a classification exists for all graphs with $cc(G) = 2$.

Another open problem that we were not able to examine was the cc-number on product graphs. Results were found in Chapter 3 on the cc-number on strong products of graphs but similar results are yet to be found on other types of products such as Cartesian, categorical, and lexicographic products.

In Chapter 4, an open problem was mentioned when we studied algorithmic complexity for the cc-game. That problem is whether computing the cc-number of a graph is **NP**-complete, and therefore is **NP**. Note that this mentioned problem is **NP**-hard but even the relationship between the different complexity classes remains the subject of research and debate.

When we studied random graphs, we found some results on graphs where the probability p of $G(n, p)$ is constant. An open problem that was not examined was finding the cc-number, or some bounds on the cc-number in $G(n, p)$, where $p = p(n)$.

Bibliography

- [1] M. Aigner, M. Fromme, A game of cops and robbers, *Discrete Applied Mathematics* **8** (1984) 1–12.
- [2] A. Berarducci, B. Intrigila, On the cop number of a graph, *Advances in Applied Mathematics* **14** (1993) 389–403.
- [3] A. Bonato, *A Course on the Web Graph*, American Mathematical Society, 2008
- [4] A. Bonato, G. Hahn, C. Wang, The cop density of a graph, *Contributions to Discrete Mathematics* **2** (2007) 133144.
- [5] A. Bonato, R.J. Nowakowski, *The Game of Cops and Robbers on Graphs*, American Mathematical Society, 2011.
- [6] N.E. Clarke, *Constrained Cops and Robber*, Ph.D. Thesis, Dalhousie University, 2002.

-
- [7] P.A. Dreyer, Applications and variations of domination in graphs, Ph.D. Dissertation, Department of Mathematics, Rutgers University, 2000.
- [8] F.V. Fomin, P.A. Golovach, J. Kratochvíl, N. Nisse, Pursuing fast robber in graphs, *Theoretical Computer Science* **411** (2010) 1167–1181.
- [9] P. Frankl, Cops and robbers in graphs with large girth and Cayley graphs, *Discrete Applied Mathematics* **17** (1987) 301-305.
- [10] G. Hahn, G. MacGillivray, A characterization of k -cop-win graphs and digraphs, *Discrete Mathematics* **306** (2006) 2492-2497.
- [11] W. Imrich, S. Klavžar, *Product Graphs*, Wiley-Interscience, 2011.
- [12] R. Isaacs, *Differential Games: A Mathematical Theory with Applications to Warfare and Pursuit, Control and Optimization*, New York: John Wiley, 1965.
- [13] C.H. Papadimitriou, *Computational Complexity*, Addison-Wesley, 1994.
- [14] A. Quilliot, Jeux et pointes fixes sur les graphes, *Thèse de 3ème cycle*, Université de Paris VI, 1978, 131–145.
- [15] R.J. Nowakowski, P. Winkler, Vertex-to-vertex pursuit in a graph, *Discrete Mathematics* **43** (1983) 235–239.

-
- [16] V.V. Vazirani, *Approximation Algorithms*, Springer-Verlag, Berlin, 2001.
- [17] A. Wigderson, P, NP and Mathematics-a computational complexity perspective, In: *Proceedings of the ICM'06*, 2007.