Cops, a Cheating Robot, Bodyguards and Presidents

by

© William Kellough

A thesis submitted to the School of Graduate Studies in partial fulfillment of the requirements for the degree of Master of Science.

Department of Mathematics and Statistics

Memorial University

August 2024

St. John's, Newfoundland and Labrador, Canada

Abstract

Cops and Robber is a pursuit-evasion game where a set of cops and a robber are placed on the vertices of a graph. The cops and the robber take turns moving onto vertices adjacent to the vertices they currently occupy. In this thesis, we consider a variation of Cops and Robber where the robber is a cheating robot, meaning the cops and the robber move simultaneously and the robber knows how the cops will move each turn. The cheating robot number of a graph G, denoted $c_{cr}(G)$, is the fewest number of cops needed to win the game on G. We study the computational complexity of this cheating robot game and prove that for any bipartite planar graph G, $c_{cr}(G) \leq 4$ is a tight upper bound. We introduce two new parameters, the push number and the bodyguard number, that are used to give new upper bounds on the cheating robot number for various graph products and show that $c_{cr}(\boxtimes_{i=1}^{k} P_{n_i}) \leq \frac{3^k-1}{2}$.

Lay summary

Pursuit-evasion games are models that mathematicians use to study scenarios in which a group of pursuers are chasing an evader in a fixed environment. The main application for these games is in robotics where the algorithms developed from solving these games are translated into algorithms robots can use to accomplish important tasks. This allows pursuit-evasion games to be useful in a wide range of fields, such as in emergency response to optimize search and rescue missions and in aeronautics to optimize flight paths.

Among pursuit-evasion games that are played in discrete environments, the most well studied is the game of Cops and Robber. The pursuers in this model are cops and the evader is a robber. If any of the cops are able to reach the location of the robber, then the cops win. If the robber can indefinitely avoid the cops, then the robber wins. In this thesis, we study a variation of Cops and Robber where the cops and the robber move simultaneously but the robber is given intelligence on how the cops will move throughout the game. The parameter of interest for this game is the fewest number of cops needed to capture this knowledgeable robber on a given playing field. We study this parameter on various playing fields. Studying this parameter on certain playing fields naturally produces a new pursuit-evasion game where the pursuers' goal is to indefinitely surround the evader. We study this new pursuit-evasion game and show how this game relates to the model with a knowledgeable robber.

Acknowledgements

I would like to start by thanking my supervisors Dr. Danny Dyer and Dr. Nancy Clarke for the amount of support they have given throughout my degree. I am extremely grateful for all of the encouragement and advice they have given me and I feel incredibly lucky to have had the opportunity to work with them. I would like to thank the Mathematics and Statistics department at Memorial University for the financial support and for the opportunity to work as a teaching assistant. Special thanks to NSERC and AARMS for the financial support as well.

I would also like to thank my family for their extraordinary support. In particular, I would like to thank my parents Clay and Laura Kellough for their consistent support throughout all of my mathematical endeavours.

The research for this thesis was funded by NSERC's CGS M scholarship, the AARMS Graduate Scholarship, and the School of Graduate Studies at Memorial University.

Statement of contribution

The research in this thesis was supervised by Dr. Danny Dyer and Dr. Nancy Clarke. The graphs in Figure 2.4 were found through a discussion with Alfie Davies who is a graduate student at Memorial University. All other results in this thesis were developed by myself, Dr. Danny Dyer, and Dr. Nancy Clarke.

Table of contents

List of figures

Chapter 1

Introduction

1.1 History and thesis overview

Pursuit-evasion games are a way for mathematicians to model various scenarios in which a group of pursuers are trying to capture an evader. Cops and Robber is a pursuit-evasion game that was first studied by Quilliot [41] and independently by Nowakowski and Winkler [35]. In Cops and Robber, a group of cops chase after a robber on the vertices of a graph. If a cop can move onto the vertex occupied by the robber, the cops win. Otherwise, if the robber can avoid the cops indefinitely, then the robber wins. One of the main problems we are interested in with Cops and Robber is finding bounds on the fewest number of cops needed to win the game for a given graph. This problem has been tackled for many graph families including planar graphs [1], outerplanar graphs [12], graphs with genus $g \geq 1$ [9] and graphs constructed using various graph products [32, 33, 34, 40]. For a survey on Cops and Robber, we direct the reader to the book [8].

The main focus of this thesis is on a variation of Cops and Robber first introduced by Huggan and Nowakowski [23]. In this variation, the robber is a "cheating robot." The term "cheating robot" refers to a robot that was built to always win in Rock Paper Scissors [26, 39]. We direct the reader to [43] for a description of the rules and terminology of Rock Paper Scissors. During each round in a game of Rock Paper Scissors, the robot would analyze the movements of its human opponent's hand as they were throwing their move and accurately predict what its opponent was going to throw. By doing this, the robot would look like it was moving at the same time as its opponent, but it would know in advance what move its opponent was going to make. Applying the same idea to Cops and Robber yields a game where a group of cops and a robber are playing on the vertices of a graph simultaneously, but the robber knows how the cops are going to move throughout the entirety of the game. The fewest number of cops needed to win this game on a given graph is referred to as the cheating robot number. In their paper, Huggan and Nowakowski [23] study how the cheating robot number behaves with respect to trees, cycles, retracts, Cartesian products of graphs, strong grids, and outerplanar graphs.

The goal of this thesis is to expand on the work that was done in [23]. In the rest of Chapter 1, we give the prerequisite knowledge of graph theory needed to read this thesis and we discuss the various Cops and Robber models that are studied in this thesis. In Chapter 2 we give general bounds on the cheating robot number that apply to all graphs, we show that there exists a graph with a cheating robot number that is lower than the cheating robot number of one of its subgraphs, and we introduce the push number which is a new parameter that considers the strategies the cops implement to capture the robber. In Chapter 3 we give a proof that the cheating robot number of any bipartite planar graph is at most four and we give an example of a bipartite planar graph with a cheating robot number of four. In Chapter 4 we give a polynomial time algorithm for checking whether a graph's cheating robot number is at most a given integer. In Chapter 5 we introduce a new pursuit-evasion game Bodyguards and Presidents where the goal of the pursuers is to, after some finite number of moves, surround the evader by the end of each of their infinite number of turns. We also study the fewest number of pursuers needed to win Bodyguards and Presidents for various families of graphs. In Chapter 6 we make use of the model introduced in Chapter 5 to obtain new results on the behaviour of the cheating robot number with respect to the strong product of graphs. In Chapter 7 we end the thesis with open questions and some areas where further research can be done.

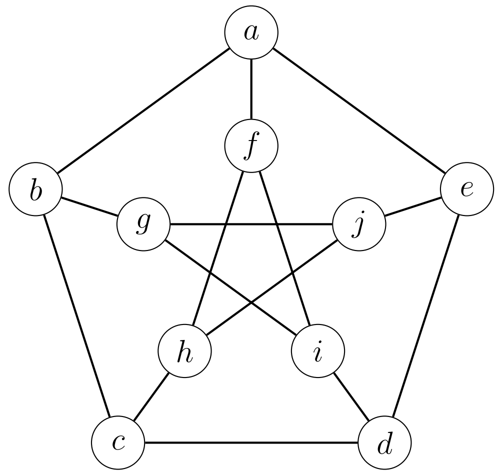


Figure 1.1: An example of a graph.

1.2 Introduction to graph theory

A *graph* is an ordered pair (V, E) where V is a set and E is a set of 2-element subsets of V. In this thesis, V is always assumed to be a finite set. For example,

$$G = (\{a, b, c, d, e, f, g, h, i, j\},$$
$$\{\{a, b\}, \{a, e\}, \{a, f\}, \{b, c\}, \{b, g\}, \{c, d\}, \{c, h\}, \{d, e\},$$
$$\{d, i\}, \{e, j\}, \{f, h\}, \{f, i\}, \{g, i\}, \{g, j\}, \{h, j\}\})$$

is a graph. The elements of V are referred to as *vertices* and the elements of E are referred to as *edges*. We will use $V(G)$ to denote the set of all vertices of G and $E(G)$ to denote the set of all edges of G. In graph theory, the edges of a graph are sometimes considered to be multisets, which allows for the edges to be 2-element subsets of the vertex set with both elements being the same. Such edges are referred to as *loops*. In this thesis, graphs will be assumed to not have loops unless stated otherwise. Instead of writing graphs using set notation, it is usually easier to understand the properties of a graph by using diagrams. In a diagram of a graph, circles are used to represent the vertices and lines between circles represent the edges. Figure 1.1 is an illustration of the above graph G. These diagrams are also referred to as graphs.

While edges are sets, it is convenient to write edges without the curly brackets or the comma between the two vertices. For example, instead of $\{x, y\}$ we may write xy. Two vertices in a graph G, say $u, v \in V(G)$, are *adjacent* if $uv \in E(G)$. If

$uv \in E(G)$, we say that u is *incident* to the edge uv. A set of vertices $S \subseteq V(G)$ is said to be *independent* if for any pair of vertices $u, v \in S$, u and v are not adjacent. For example, the set of vertices $\{a, c, g\}$ in the graph in Figure 1.1 is an independent set. The cardinality of the largest independent set of vertices in G is denoted $\alpha(G)$. As an example, if G is the graph in Figure 1.1 then $\alpha(G) = 4$. The *degree* of a vertex v, denoted $\deg(v)$, is the total number of vertices adjacent to v. Equivalently, $\deg(v)$ is the total number of edges incident to v. For $k \in \mathbb{Z}^+$, a graph is k-regular if every vertex in the graph has degree k. For example, since every vertex in the graph in Figure 1.1 has a degree of three, the graph is 3-regular. For a graph G, we denote the *maximum degree* as $\Delta(G) = \max_{v \in V(G)}\{\deg(v)\}$ and the *minimum degree* as $\delta(G) = \min_{v \in V(G)}\{\deg(v)\}$. The *open neighbourhood* of a vertex v, denoted $N(v)$, is the set of all vertices adjacent to v. The *closed neighbourhood* of a vertex v is the set $N(v) \cup \{v\}$ and is denoted $N[v]$. Since we will talk about open neighbourhoods more often than closed neighbourhoods, we will use the phrase *neighbourhood* in place of open neighbourhood throughout this thesis for convenience. To help specify the graph that contains the neighbourhood of interest, say the graph G for example, then we write the neighbourhood as $N_G(v)$ and the closed neighbourhood as $N_G[v]$.

A *walk* is a sequence of vertices in a graph $v_0, v_1, \ldots, v_k$ such that v_i is adjacent to v_{i+1} for each $0 \leq i \leq k-1$. The sequence of vertices a, e, j, e, d is a walk in Figure 1.1. The *length of a walk* is one less than the number of vertices contained in the walk. A *path* is a walk that does not contain a repeated vertex. Thus, a, e, j, e, d is not a path since it contains the vertex e twice. On the other hand, a, e, j, g would be an example of a path in Figure 1.1. If, for example, the first and last vertices in a path P are u and v, then we say that the *endpoints* of P are u and v and we call P a *u-v path*. If u and v are vertices in a graph G, then a *u-v geodesic* is a u-v path P in G such that for any other u-v path P' in G, the length of P is at most the length of P'. In other words, a geodesic is a shortest path between two vertices. While a, e, j, g is an a-g path in Figure 1.1, the path a, b, g is a shorter path. Thus a, e, j, g is not a geodesic and it can be seen by inspection that a, b, g is a geodesic. We say that a graph G is *connected* if and only if for every pair of vertices $u, v \in V(G)$, there exists a walk whose first vertex is u and whose last vertex is v. Equivalently, a graph G is connected if and only if for every pair of vertices $u, v \in V(G)$ there exists a u-v path in G. If a graph is not connected we say that it is *disconnected*. The *distance* between two vertices u and v in a graph G is the length of a u-v geodesic. This distance is

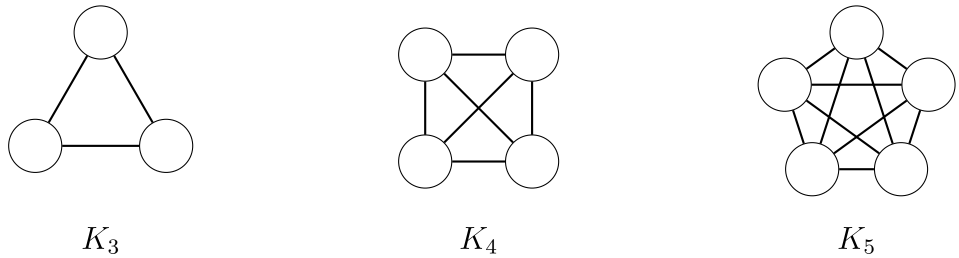


Figure 1.2: The graphs K_3, K_4, and K_5.

denoted $d(u,v)$. The *eccentricity* of a vertex v is defined as $\max_{u \in V(G)}\{d(v,u)\}$. A *center* vertex is a vertex with minimum eccentricity in a graph. It is possible for a graph to have more than one center vertex. For example, in Figure 1.1 every vertex has an eccentricity of three and so every vertex is a center vertex.

A *cycle* in a graph is a sequence of distinct vertices $v_0, \ldots, v_k$ such that v_i is a adjacent to v_{i+1} for each $0 \le i \le k-1$ and v_k is adjacent to v_0. The sequence of vertices a, b, c, h, f form a cycle in Figure 1.1. A connected graph that does not contain a cycle is called a *tree*. The *girth* of a graph G is the minimum number of vertices in any cycle contained in G. For example, the girth of the graph in Figure 1.1 is five. If G is a tree, then the girth of G is defined to be infinity. Any vertex in a tree that has degree one is called a *leaf*. Vertices of degree one in graphs that are not trees are called *pendants*.

Let $v_0, v_1, \ldots, v_{n-1}$ be a set of n vertices. The graph P_n is obtained by including the edges v_iv_{i+1} for each $0 \le i \le n-2$. That is, P_n is a path containing n vertices. The graph C_n is obtained by including the same edges as P_n with the additional edge $v_{n-1}v_0$. That is, C_n is a cycle with n vertices. We may also refer to the graph C_n as the n-cycle. The *complete graph*, K_n, is formed by including every possible edge. So $E(K_n) = \{v_iv_j \mid 0 \le i < j \le n-1\}$. Figure 1.2 illustrates three complete graphs. An *empty graph* is a graph that does not have any edges.

A graph G is *bipartite* if $V(G)$ can be partitioned into two sets $X, Y \subsetneq V(G)$ such that X and Y are independent sets. The three graphs in Figure 1.3 are all bipartite graphs. For example, the vertices of the left graph in Figure 1.3 can be partitioned into two sets of independent vertices, one of size two and one of size three. A *vertex colouring* of a graph G using a set C is a mapping $c : V(G) \to C$. The elements of the set C are referred to as *colours*. A *proper k-colouring* of G is a vertex colouring

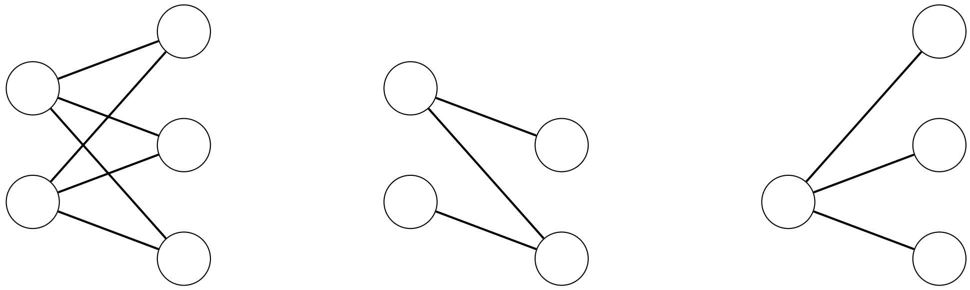


Figure 1.3: The graph $K_{2,3}$ (left), a subgraph of $K_{2,3}$ that is not an induced subgraph (middle), and an induced subgraph of $K_{2,3}$ (right).

that uses at most k colours and no pair of adjacent vertices share the same colour. If a graph can be coloured this way, then we say that the graph can be *properly k-coloured*. Vertex colourings allow for a second way of defining bipartite graphs. A graph is bipartite if and only if it contains at least two vertices and it can be properly 2-coloured. There is a third way bipartite graphs can be defined. It is commonly known that a graph is bipartite if and only if it does not contain a cycle with an odd number of vertices. As an example, the graph in Figure 1.1 contains the cycle a, b, c, d, e of length five and so the graph is not bipartite. The *complete bipartite graph $K_{n,m}$* is the graph with vertex set $V = X \cup Y$ where $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_m\}$ and edge set $E = \{xy \mid x \in X, y \in Y\}$. A *star* is a graph of the form $K_{1,n}$ where $n \in \mathbb{Z}^+$.

Let G be a graph. We say that a graph H is a *subgraph* of G, denoted $H \subseteq G$, if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. By this definition, every graph is a subgraph of itself. In the case where $H \subseteq G$ such that $V(H) \subset V(G)$, we call H a *proper subgraph* of G. We call H an *induced subgraph* of G if $H \subseteq G$ and $E(H) = \{uv \in E(G) \mid u, v \in V(H)\}$. If H is an induced subgraph of G with vertex set $V(H) \subseteq V(G)$, then we say that H is the subgraph induced by the set of vertices $V(H)$. Figure 1.3 illustrates the graph $K_{2,3}$ and two subgraphs of $K_{2,3}$ where one is an induced subgraph while the other is not. We say that H is a *component* of G if H is an induced subgraph of G and for every $u \in V(H)$ and $v \in V(G) \backslash V(H)$, $uv \notin E(G)$. If $S \subseteq V(G)$, then the graph $H = G \backslash S$ is the graph with vertex set $V(H) = \{v \in V(G) \mid v \notin S\}$ and edge set $E(H) = \{uv \in E(G) \mid u, v \notin S\}$. If $T \subseteq E(G)$, then the graph $H = G \backslash T$ is the graph with vertex set $V(H) = V(G)$ and edge set $E(H) = \{uv \in E(G) \mid uv \notin T\}$. In the case where S and T only have one element, say $\{v\} = S$ and $\{uv\} = T$, then we may instead write $G \backslash v$ to indicate the

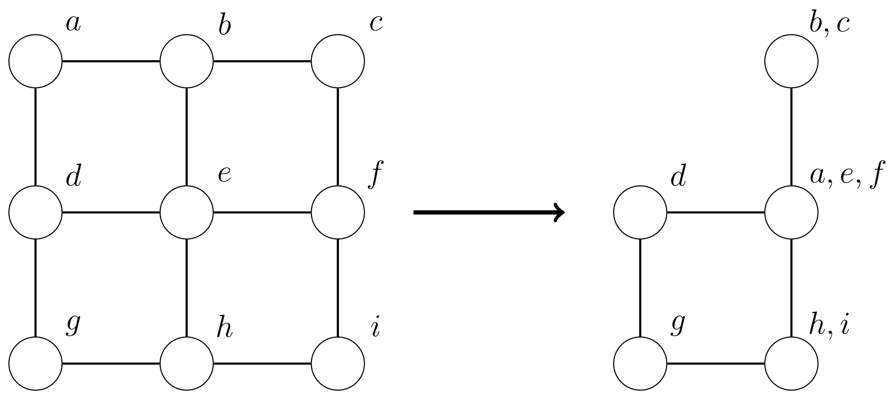


Figure 1.4: An example of a retract with the mapping of the vertices labelled.

removal of a single vertex or $G \backslash uv$ to indicate the removal of a single edge. We call $H \subseteq G$ a *k-core* if H is an induced subgraph of G and $\delta(H) \geq k$. For example, since every graph is an induced subgraph of itself, the graph $K_{2,3}$ is a 2-core of $K_{2,3}$. On the other hand, the proper subgraph on the right in Figure 1.3 is a 1-core.

If there exists a mapping $\varphi : V(G) \to V(H)$ such that for every pair of adjacent vertices $u, v \in V(G)$ the pair of vertices $\varphi(u), \varphi(v) \in V(H)$ are either adjacent or $\varphi(u) = \varphi(v)$, then we say that G is *homomorphic* to H and we call the map φ a *homomorphism*. An *isomorphism* is a homomorphism that is also a bijection. Two graphs are *isomorphic* if there exists an isomorphism that maps the vertices of one graph to the other. If G and H are isomorphic graphs, we use the notation $G \cong H$. We say that H is a *retract* of G if H is an induced subgraph of G and there exists a homomorphism $\varphi : V(G) \to V(H)$ such that for every $v \in V(H) \subseteq V(G)$, $\varphi(v) = v$. In this case we call the homomorphism φ a *retraction map*. Figure 1.4 gives an example of a retract.

For more on graph theory, see the textbook by West [42].

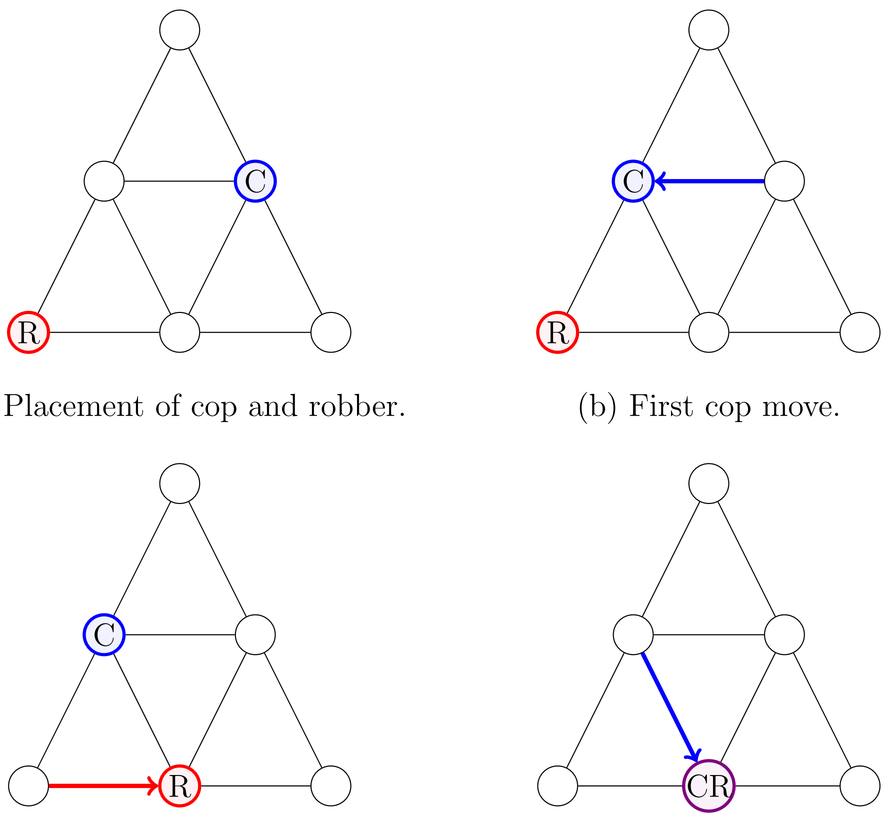
1.3 Cops and Robber models

Cops and Robber is a two-player, discrete time-step, pursuit-evasion game played on the vertices of a graph. This game was first studied by Quilliot [41] and then later, independently, by Nowakowski and Winkler [35]. One player controls a set of cops while the other player controls a robber. To help distinguish the cops and the robber, the cops are given the pronouns she/her while the robber is given the pronouns he/him. The game begins with the cops being placed on the vertices of a graph, then the robber is placed on a vertex. A legal move for either a cop or the robber is to either stay at the vertex they currently occupy or move to a vertex that is adjacent to the vertex they currently occupy. After the cops and the robber are placed on the graph, the cops get the first turn. On a cop turn, every cop makes one legal move. After the cops' turn, it is the robber's turn where he makes one legal move. The cops and the robber alternate taking turns for the rest of the game. If, after a finite number of moves, a cop and a robber occupy the same vertex, then the robber is considered to be *captured* and the cops win. If the robber can indefinitely avoid sharing a vertex with any of the cops, then the robber wins.

The *cop number* of a graph, introduced by Aigner and Fromme [1], is the fewest number of cops needed to win Cops and Robber on that graph. If G is a graph then its cop number is denoted $c(G)$. We say that G is *copwin* if $c(G) = 1$. Figure 1.5 illustrates an example of Cops and Robber being played on a graph that is copwin. In Figure 1.5, the red vertex labelled "R" represents Robert while the blue vertex labelled "C" represents a cop. The red arrow indicates movement from Robert while the blue arrow indicates movement from a cop. For a survey on Cops and Robber, see [8]. While the game Cops and Robber is usually referred to as Cops and Robbers in newer publications, for example the book written about the game [8], we will use the name Cops and Robber for the game since the minimum number of cops needed to win does not change based on the number of robbers. Once a robber is captured in Cops and Robbers, he is eliminated from the game. Thus given any graph G and $r \in \mathbb{Z}^+$ robbers, $c(G)$ cops can win by capturing the robbers one at a time.

In this thesis, we focus on a variation of Cops and Robber first introduced by Huggan and Nowakowski [23] where the robber is a *cheating robot*. We will refer to this game as the *cheating robot variant*. The term "cheating robot" refers to a player in a two-player game that moves at the same time as their opponent, but always knows in advance the move of their opponent. The term originally came from a robot who would cheat in Rock Paper Scissors by throwing the winning move after determining what its opponent was throwing [26, 39]. Studying games where one of the players is a cheating robot has been done in [22, 23, 24].

The setup and legal moves for the cheating robot variant is the same as in Cops and



(a) Placement of cop and robber.

(b) First cop move.

(c) First robber move.

(d) The cop captures the robber.

Figure 1.5: Example of possible moves in a game of Cops and Robber played on the graph shown.

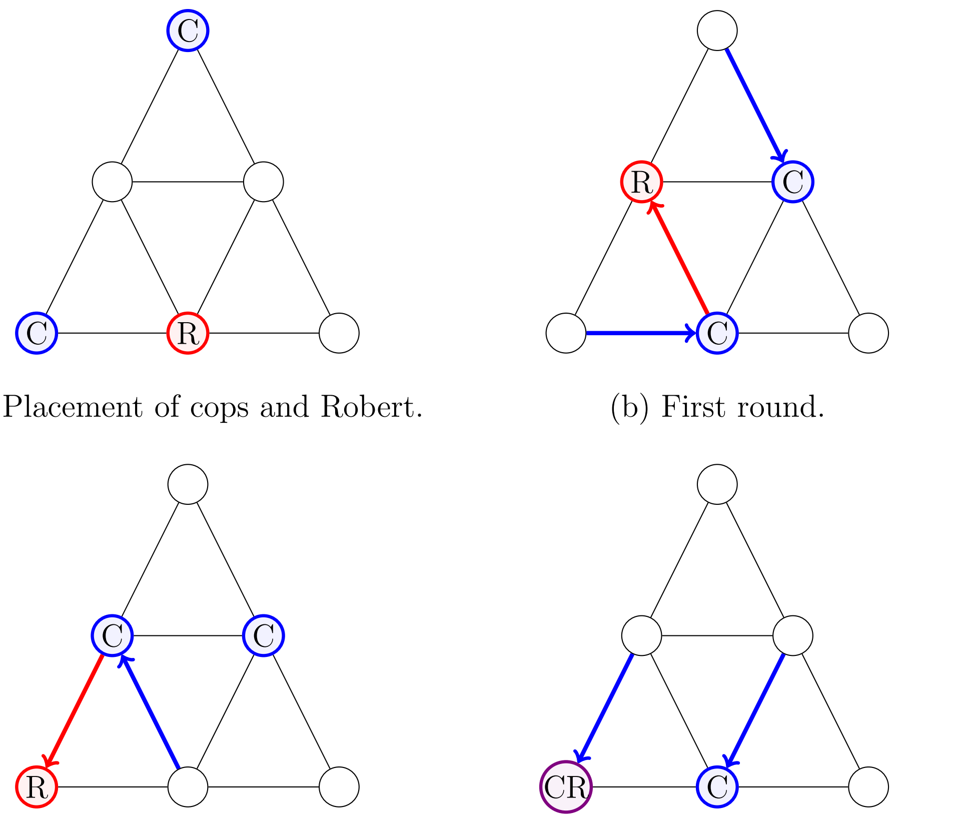
Robber. The difference is instead of the cops and the robber alternately making legal moves, all of the cops and the robber move simultaneously. However, the robber has the advantage of knowing how the cops will move before each move. Since the robber is also a cheating robot, we will give him the nickname *Robert* to help distinguish him from a robber in Cops and Robber. The win condition for the cops is to either move onto the vertex occupied by Robert or to traverse the same edge as Robert during a move. If Robert can avoid this indefinitely, Robert wins. We define a *cop winning strategy* for $k \in \mathbb{Z}^+$ cops $c_1, \ldots, c_k$ on the graph G to be a pair (X, A) where X set of vertices $X = \{x_1, \ldots, x_k\} \subseteq V(G)$ and A is an algorithm for the cops' moves such that when c_i starts the game on x_i for each $1 \leq i \leq k$, regardless of how Robert moves the cops can capture Robert in finitely many moves by following the algorithm A. The fewest number of cops needed to win the cheating robot variant on a given graph G is called the *cheating robot number* of the graph and is denoted $c_{cr}(G)$. Figure 1.6 illustrates an example of the cheating robot variant being played. It is clear that one cop alone cannot capture Robert on the graph in Figure 1.6 since Robert could pick any C_3 contained in the graph and avoid the cop indefinitely. Since the graph in Figure 1.6 is copwin, this is an example where the cheating robot number is strictly larger than the cop number.

We can see that for any graph G, $c_{cr}(G) \geq c(G)$ if we translate the rules of the cheating robot variant to a game where the players alternately move as was done in [23]. To do this, we need to change the win condition for the cops.

Theorem 1.3.1. For any $k \in \mathbb{Z}^+$, k cops win the cheating robot variant if and only if they win the game where the cops and Robert alternately move the same as in Cops and Robber but the cops win only if either

- Robert ends his turn on a vertex occupied by a cop, or
- Robert traverses an edge that was traversed by a cop on her previous legal move.

Proof. If Robert can win against $k \in \mathbb{Z}^+$ cops in the cheating robot variant, then he has a strategy where he never traverses an edge traversed by a cop during his move and he never ends his move on a vertex occupied by a cop. If he uses the same strategy in the turn-based game described in the theorem, Robert will never traverse an edge that was traversed by a cop on the previous cop move and he will not end his turn on



(a) Placement of cops and Robert.

(b) First round.

(c) Second round.

(d) Third round: Robert is captured.

Figure 1.6: Example of possible moves in a game of the cheating robot variant played on the graph shown.

a vertex occupied by a cop. Thus, Robert can win against k cops in the turn-based game.

If instead Robert has a winning strategy against k cops in the turn-based game, then he has a strategy where, for the entirety of the game, he neither ends his turn on a vertex occupied by a cop nor traverses an edge that was traversed by a cop during the previous cop turn. By using the same strategy against k cops in the cheating robot variant, Robert will never occupy a vertex that is also occupied by a cop and he will never traverse an edge at the same time as a cop. That is, Robert can win the cheating robot variant against k cops. $\square$

For the rest of this thesis, we will consider the game as turn-based and also refer to this version as the cheating robot variant.

One way for the cops to win against Robert is to first occupy every vertex in his neighbourhood, and then have these cops move onto Robert's vertex. When this occurs, Robert will lose if he stays on the vertex he is currently occupying since he would be ending his turn on a vertex occupied by a cop and Robert will lose if he moves to a different vertex since he will be traversing an edge that was just traversed by a cop. If the cops are able to occupy every vertex in Robert's neighbourhood, then the cops have *surrounded* Robert. If the cops are able to win against Robert, then by changing their final move they can surround him before capturing him.

Lemma 1.3.2. A set of k cops can surround Robert at some vertex in G if and only if $k \geq c_{cr}(G)$.

Proof. If k cops can surround Robert, they can win the game on their next move by the previous discussion. Thus k cops can win the cheating robot variant on G and so $k \geq c_{cr}(G)$.

Instead suppose $k \geq c_{cr}(G)$ and consider the game being played with k cops. Assume that Robert is playing in such a way that he will never purposefully make a move that would result in a loss. Since the cops win, Robert is eventually put into a position where every move results in a loss. Let v be the vertex Robert is occupying when Robert is put into such a position.

Since Robert staying on v is a losing move, there is a cop on v. For every vertex u that is adjacent to v, moving to u is a losing move. Thus either a cop is occupying

u or a cop traversed the edge $\{u, v\}$ on their last turn. Suppose the cops replay the game using the same strategy as before with the following change to their move the turn before they capture Robert. For each cop c that moved onto v to capture Robert, instead c remains on the vertex she was occupying on the previous turn. By making this change, every vertex adjacent to v becomes occupied by a cop. So Robert is surrounded. $\square$

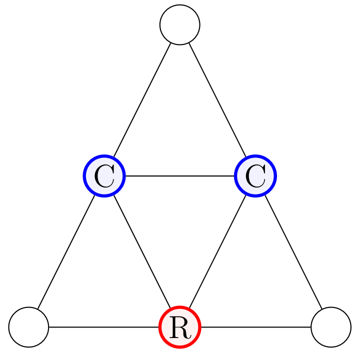
Another variation of Cops and Robber that we will discuss is *Surrounding Cops and Robbers* which was first introduced in [11]. For this variation of Cops and Robber, we will keep the plurality of the name Surrounding Cops and Robbers to be consistent with the original paper on the game [11]. Surrounding Cops and Robbers has the same ruleset as Cops and Robber with the only exception being the winning condition for the cops. In Surrounding Cops and Robbers, the cops win if they can surround the robber, or if the robber ends his turn on a vertex occupied by a cop. For convenience, we will also refer to this game as the *surrounding variant*. The minimum number of cops needed to win the surrounding variant on a graph G is called the *surrounding number* and is denoted $\sigma(G)$. Figure 1.7 demonstrates how in the surrounding variant, three cops are needed to win on the graph shown unlike in the cheating robot variant where two cops were shown to win in Figure 1.6. For more on the surrounding variant, see [2, 10, 11, 27].

Lemma 1.3.2 shows that the cheating robot variant and the surrounding variant have almost identical win conditions for the cops. The only difference between the two games is that Robert cannot traverse edges that were traversed by cops while the robber in the surrounding variant can. In Section 2.3 we will further discuss the relationship between these two games.

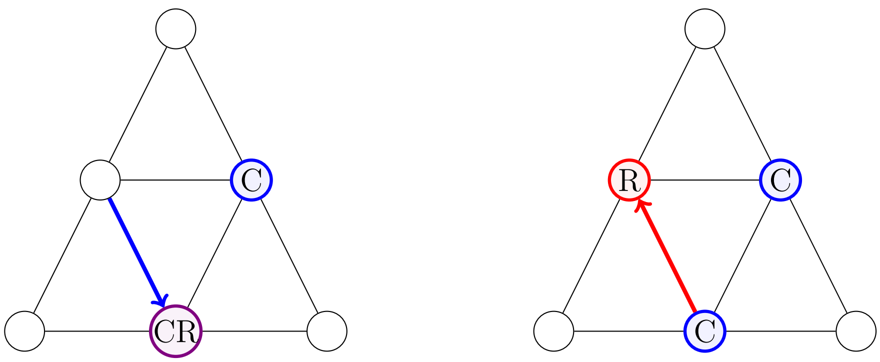
1.4 Graph classes

Let G and H be graphs. The *Cartesian product* of G and H, denoted $G \square H$, has vertex set $V(G \square H) = \{(u, v) \mid u \in V(G) \text{ and } v \in V(H)\}$ and edge set

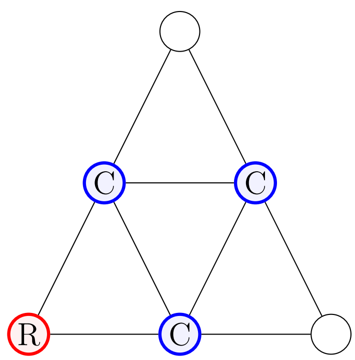
$$E(G \square H) = \{\{(u, v), (x, y)\} \mid ux \in E(G) \text{ and } v = y, \\ \text{or } u = x \text{ and } vy \in E(H)\}.$$



(a) Placement of cops and robber.



(b) Regardless of how the two cops move, the robber can avoid being surrounded.



(c) With three cops, the robber loses immediately.

Figure 1.7: An illustration that the graph from Figures 1.5 and 1.6 has a surrounding number of three.

The *strong product* of G and H, which is denoted $G \boxtimes H$, has the vertex set $V(G \boxtimes H) = \{(u, v) \mid u \in V(G) \text{ and } v \in V(H)\}$ and edge set

$$\begin{aligned} E(G \boxtimes H) = \{\{(u,v),(x,y)\} \mid\ & ux \in E(G) \text{ and } v = y, \\ & \text{or } u = x \text{ and } vy \in E(H), \\ & \text{or } ux \in E(G) \text{ and } vy \in E(H)\}. \end{aligned}$$

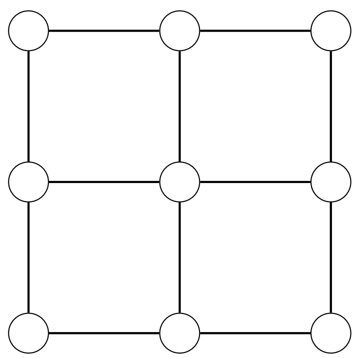
The *lexicographic product* of G and H, which is denoted $G \bullet H$, has vertex set $V(G \bullet H) = \{(u, v) \mid u \in V(G) \text{ and } v \in V(H)\}$ and edge set

$$\begin{aligned} E(G \bullet H) = \{\{(u,v),(x,y)\} \mid\ & u = x \text{ and } vy \in E(H), \\ & \text{or } ux \in E(G)\}. \end{aligned}$$

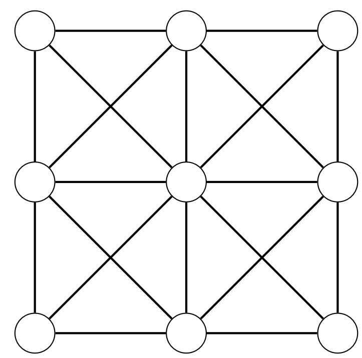
Figure 1.8 illustrates what these products look like when applying them to two copies of P_3. Notably, the Cartesian product and the strong product are *symmetric* meaning that for any two graphs G and H, $G \square H$ ($G \boxtimes H$) is the same graph as $H \square G$ ($H \boxtimes G$). However, the lexicographic product is not symmetric and an example is illustrated in Figure 1.9.

If we take the product of two graphs $G \otimes H$ where $\otimes$ is either the Cartesian product, the strong product or the lexicographic product, then the graphs G and H are called the *factors* of the graph product. The subgraph induced by the vertices $\{(u, v) \mid u \in V(G)\} \subseteq V(G \otimes H)$ is denoted $G.\{v\}$. Similarly, the subgraph of $G \otimes H$ induced by the vertices $\{(u, v) \mid v \in V(H)\}$ is denoted $\{u\}.H$. We can think of $G.\{v\}$ and $\{u\}.H$ as being "copies" of G and H respectively within the graph $G \otimes H$. We say that two copies of G within $G \otimes H$, say $G.\{v_1\}$ and $G.\{v_2\}$, are adjacent if $v_1v_2 \in E(H)$. Similarly, two copies of H, say $\{u_1\}.H$ and $\{u_2\}.H$, are adjacent if $u_1u_2 \in E(G)$.

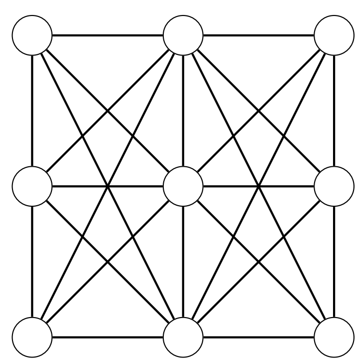
If we want to take the Cartesian product of the graph G with itself k times we use the notation $\square_{i=1}^{k} G$. If we want to do the same with the strong and lexicographic products we use the notation $\boxtimes_{i=1}^{k} G$ and $\bullet_{i=1}^{k} G$ respectively. If $n_1, \ldots, n_k \geq 2$, the graphs of the form $\square_{i=1}^{k} P_{n_i}$, $\boxtimes_{i=1}^{k} P_{n_i}$ and $\bullet_{i=1}^{k} P_{n_i}$ are called the Cartesian grid, the strong grid, and the lexicographic grid respectively. In particular, we will refer to these graphs as *k-dimensional grids*. The graph $\square_{i=1}^{k} P_2$ is referred to as the kth dimensional *hypercube* and is denoted Q_k. Usually we will label the vertices of a k-dimensional



$P_3 \square P_3$

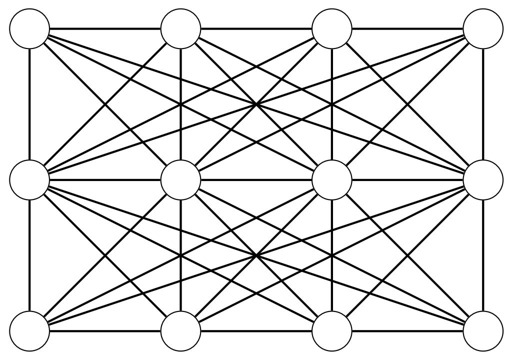


$P_3 \boxtimes P_3$

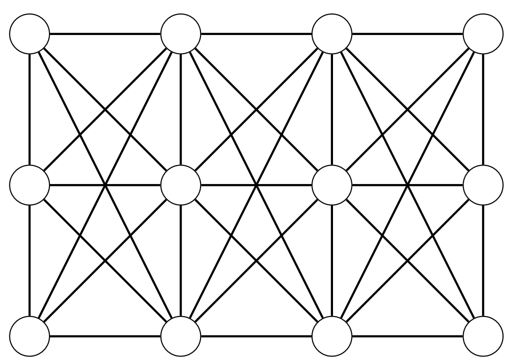


$P_3 \bullet P_3$

Figure 1.8: Cartesian, strong, and lexicographic products with factors P_3 and P_3.



$P_3 \bullet P_4$



$P_4 \bullet P_3$

Figure 1.9: An example where $G \bullet H$ is not the same as $H \bullet G$.

Not a planar embedding　　　　A planar embedding

Figure 1.10: Two diagrams of the graph K_4 where one is a non-planar embedding while the other is a planar embedding.

grid $\otimes_{i=1}^{k} P_{n_i}$ with k-coordinate vectors $(x_1, x_2, \ldots, x_k)$ where for each $1 \leq i \leq k$, $1 \leq x_i \leq n_i$ and whether two vertices corresponding to the vectors $(v_1, \ldots, v_k)$ and $(v_1', \ldots, v_k')$ are adjacent depends on the product. For more on graph products, we direct the reader to [25].

If Robert is on the vertex (u, v), then we say that Robert's *shadow* occupies the vertices of the form (u, y) for any $y \neq v$ and (x, v) for any $x \neq u$. In particular, Robert's shadow in the graph $G.\{t\}$ occupies the vertex (u, t) and Robert's shadow in the graph $\{s\}.H$ occupies the vertex (s, v). It is useful to reference these vertices when developing winning strategies for the cops on graphs constructed using graph products.

A graph is *planar* if it can be drawn on a flat surface in such a way that none of its edges intersect. Such a drawing of a graph is referred to as a *planar embedding*. Figure 1.10 illustrates two ways of drawing the graph K_4 where one is a planar embedding while the other is not a planar embedding. Since K_4 has a planar embedding, K_4 is a planar graph. The *faces* in a planar embedding are the bounded, empty regions that are enclosed by vertices and edges as well as the unbounded region in which the graph is contained. For example, the planar embedding in Figure 1.10 has three bounded faces and one unbounded face. A graph is *outerplanar* if it has a planar embedding with every vertex adjacent to the unbounded face. Some examples of outerplanar graphs are trees and the graph C_n for any $n \geq 3$.

Instead of graphs being drawn on a flat surface, we can generalize the idea of planarity to graphs drawn on other surfaces. The *genus* of a surface, intuitively, is the number of "holes" the surface has. For more on the genus of a surface, we direct the reader to [21]. If a graph G can be drawn on a surface of genus g such that none of its edges intersect, and G cannot be drawn on a surface of genus $g-1$ without having intersecting edges, then we say that the *genus of the graph G* is g. As an example, planar graphs are graphs that have genus zero. Graphs that can be drawn on surfaces of genus one, such as a torus, are called *toroidal* graphs.

Chapter 2

Preliminary Results

In this chapter, we will review some of the results from the original paper on the cheating robot number [23]. These results will be used throughout the thesis. We will also provide new, general upper bounds on the cheating robot number. In the original Cops and Robber game, it is easy to find graphs G and H satisfying $H \subseteq G$ and $c(G) < c(H)$. For example, for any $n \geq 4$, $c(K_n) = 1$ but $c(C_n) = 2$. We will show that such examples exist for the cheating robot number. In Chapter 1 we alluded to a connection between the cheating robot number and the surrounding number. In Section 2.3 we will give new bounds for the surrounding number in terms of the cheating robot number.

2.1 General bounds on the cheating robot number

First, we give a useful lower bound on the cheating robot number that was first proved in [23].

Theorem 2.1.1. [23] If G is a graph with a k-core where $k \in \mathbb{Z}^+$, then $c_{cr}(G) \geq k$.

Proof. Suppose Robert is playing against $k-1$ cops on a graph with a k-core. If Robert starts on a vertex u in the k-core, then there are at least k vertices adjacent to him that are still in the k-core. Thus if a cop moves to u, there will always be at least one vertex, say v, in the k-core such that v is adjacent to Robert, v is not occupied by a cop, and the edge uv was not traversed by a cop on their previous turn.

Therefore Robert can avoid the $k-1$ cops indefinitely by staying in the k-core. $\square$

While the size of the largest k such that a graph G contains a k-core provides a lower bound for $c_{cr}(G)$, the difference $c_{cr}(G) - k$ may be arbitrarily large.

Theorem 2.1.2. If $N \in \mathbb{Z}^+$ then there exists a graph G where k is the largest integer such that G contains a k-core and $c_{cr}(G) - k > N$.

Proof. To prove the theorem, we give an infinite family of graphs where the every graph in the family contains a 2-core, none of the graphs in the family contain a 3-core and for any $N \in \mathbb{Z}^+$ there exists a graph G in the family such that $c_{cr}(G) > N$. Consider the family of graphs $\{G_n\}_{n=2}^{\infty}$ where G_n is obtained by replacing all of the edges in the hypercube Q_{n-1} with 4-cycles. The graphs G_2 and G_3 are illustrated in Figure 2.1. Every vertex of G_n either has degree 2 or degree $2(n-1)$. Furthermore, all of the neighbours of each vertex of degree $2(n-1)$ have degree 2. Thus for all $n \geq 2$, G_n contains a 2-core but not a k-core for any $k \geq 3$. Next, we claim that $c_{cr}(G_n) > n-1$.

Fix $n \geq 2$ and suppose Robert plays against $n-1$ cops on the graph G_n. Since there are 2^{n-1} vertices of degree $2(n-1)$, regardless of where the cops place themselves at the start of the game, Robert can place himself on a vertex of degree $2(n-1)$. There are $n-1$ other vertices of degree $2(n-1)$ that are each of distance two away from Robert. Let U denote this set of $n-1$ vertices each of degree $2(n-1)$. There are two internally disjoint paths from Robert's vertex to each of the vertices in U. Since for all $u, v \in U$, u and v do not have any neighbours in common, it is not possible for one cop to be adjacent to more than one vertex in U. Therefore Robert can avoid capture indefinitely by waiting at a vertex of degree $2(n-1)$ until a cop moves to his position, and then moving to a vertex of degree $2(n-1)$ before any of the other $n-2$ cops can stop him.

So, $c_{cr}(G_n) \geq n$. Thus, if $N \in \mathbb{Z}^+$, then $c_{cr}(G_{N+2}) - 2 \geq N$. Since each G_n contains a 2-core and not a 3-core, this proves the result. $\square$

As a direct consequence of Theorem 2.1.1, Huggan and Nowakowski [23] gave a characterization of all graphs with a cheating robot number of one. Here we provide the characterization as well as their proof.

Theorem 2.1.3. [23] Let G be a graph. Then $c_{cr}(G) = 1$ if and only if G is a tree.

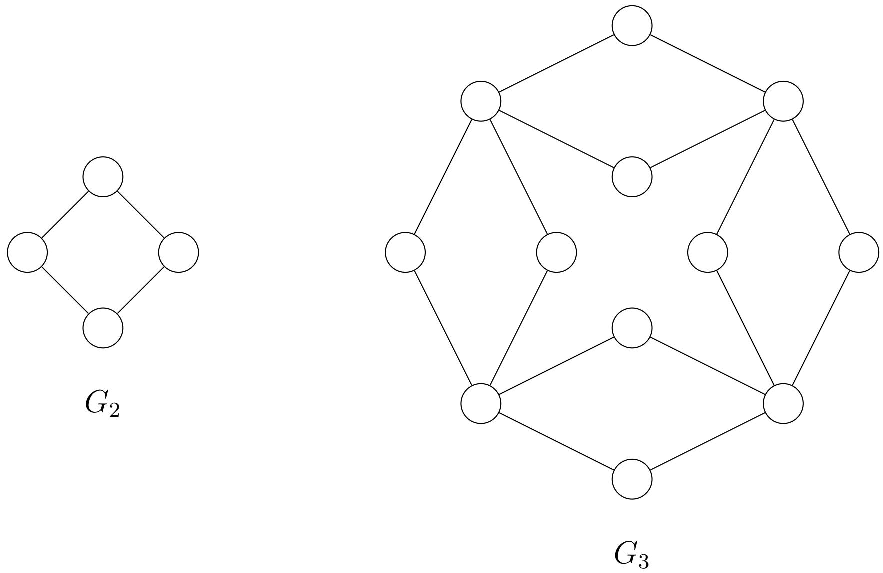


Figure 2.1: The two smallest graphs in the family of graphs discussed in the proof of Theorem 2.1.2.

Proof. If G is a tree then a winning strategy for a single cop is to always move such that the distance between her and Robert decreases. Since G does not contain a cycle, in finitely many moves Robert will be forced to move onto a leaf where he will be surrounded. So $c_{cr}(G) = 1$. If G is not a tree, then it contains a cycle which is a 2-core. So $c_{cr}(G) \geq 2$ by Theorem 2.1.1. $\square$

Using a technique from [13], we can prove that the girth of a graph has an effect on the cheating robot number.

Theorem 2.1.4. If G is a graph with $\delta(G) \geq 3$ and girth $g \geq 6$, then $c_{cr}(G) \geq 4$.

Proof. Suppose three cops are able to surround Robert at some vertex. We consider the possibilities for the positions of the cops and of Robert on Robert's last move before he is surrounded. Since we are assuming that every move Robert makes results in him being surrounded, including Robert's option to pass, Robert is on a vertex of degree three and every vertex adjacent to him that is not occupied by a cop has degree three. There are three cases for the position of the cops: none of the cops are within distance one of Robert, exactly one cop is within distance one of Robert and exactly two cops are within distance one of Robert. For convenience, we will say that a cop is *near* Robert if she is at most distance one from him.

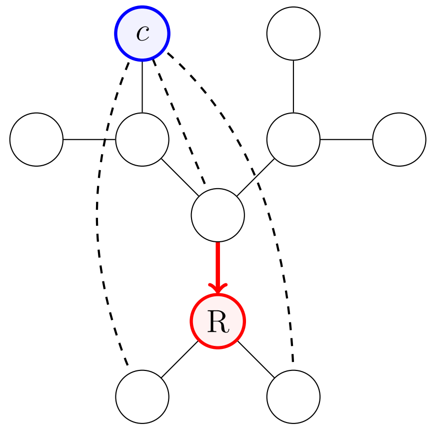


Figure 2.2: An illustration of the case in the proof of Theorem 2.1.4 where one cop is near Robert.

No cops near Robert: Let v be the vertex on which Robert starts on his final move. By moving to an adjacent vertex, Robert is able to avoid being surrounded for an extra turn since no cop can move onto v in one move. This contradicts our assumption that the cops were one move away from surrounding Robert.

One cop near Robert: There are two possibilities for the cop that is near Robert, either she is adjacent to Robert or she has moved onto Robert's vertex. In both cases, Robert has the same options for his final move and the other two cops are of distance two away from Robert before he moves. Let c be one of the cops that is distance two away from Robert. Since G does not contain a cycle of size five or less, Robert can move to a vertex that is of distance three away from c. This makes it impossible for c to move to a vertex adjacent to Robert in one move, and so Robert cannot be surrounded. Figure 2.2 gives an illustration of this argument. In Figure 2.2, the dashed lines indicate edges that cannot exist because the girth of the graph is at least six.

Two cops near Robert: If the two cops are adjacent to Robert and Robert moves on his final turn, one of those cops will have to move to a vertex adjacent to Robert that is different from Robert's initial vertex. Thus the graph contains a C_4. Figure 2.3 illustrates this argument. A similar argument holds for the case where one cop is on the vertex occupied by Robert before his final move while the other cop is adjacent to him. If both cops are on the vertex occupied by Robert before his final move then, since Robert's initial vertex has degree three, Robert can move to a vertex that has two neighbours that are not adjacent to these two cops. Thus the

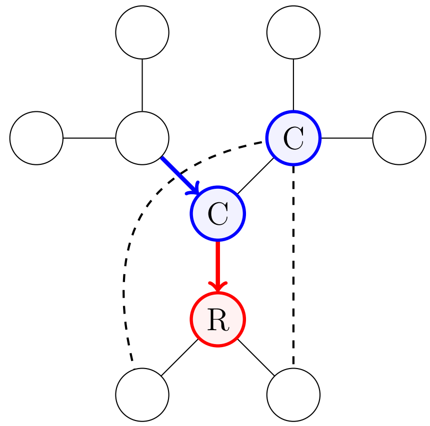


Figure 2.3: An illustration of the case in the proof of Theorem 2.1.4 where two cops are near Robert.

cops cannot surround Robert.

By our case analysis, it is impossible for three cops to surround Robert. Thus, $c_{cr}(G) \geq 4$. □

Next, we focus on upper bounds for the cheating robot number. It is trivial that for any graph G, $c_{cr}(G) \leq |V(G)|$. We can do slightly better by placing a cop on every vertex except on a set of independent vertices. This forces Robert to begin the game on a vertex that is already surrounded by cops. Since the largest independent set of G has size $\alpha(G)$, we have the following theorem.

Theorem 2.1.5. For any graph G, $c_{cr}(G) \leq |V(G)| - \alpha(G)$.

Next, we give a new upper bound on the cheating robot number by making use of the idea that Robert is unable to access vertices that are occupied by cops. Depending on the placement of the cops, it is possible for a vertex v unoccupied by the cops to be inaccessible to Robert. This occurs when every walk from Robert's vertex to v contains a vertex occupied by a cop. For example, on the path $P_n = v_1, \ldots, v_n$, if a cop is on v_i and Robert is on v_j where $j < i$, then it is not possible for Robert to move onto any of $v_{i+1}, \ldots, v_n$ no matter how many moves he is given. Suppose on a graph G we place a cop on each of the vertices $x_1, \ldots, x_k \in V(G)$. Then Robert will not be able to move onto x_i for any $1 \leq i \leq k$. If we delete the vertices $x_1, \ldots, x_k$ from G then the resulting graph H may be disconnected, meaning Robert is on some component of H and is unable to walk to any other component of H. Suppose

m is the maximum cheating robot number out of all components of H. Then $m+k$ cops can capture Robert in G by placing a cop on each x_i where $1 \leq i \leq k$ and then using m cops to capture Robert who is restricted to a component of the subgraph H. The following theorem generalizes this idea to obtain an upper bound on the cheating robot number.

Theorem 2.1.6. Let G be a graph on n vertices. If S_k is the smallest set of vertices in G such that the components of $G \backslash S_k$ have cheating robot number at most k, then

$$c_{cr}(G) \leq \min_{1 \leq k \leq |V(G)| - \alpha(G)} \{|S_k| + k\}.$$

Proof. Let $j \in \mathbb{Z}^+$ such that

$$\min_{1 \leq k \leq |V(G)| - \alpha(G)} \{|S_k| + k\} = |S_j| + j.$$

The cops can win the game by first placing one cop on each vertex of S_j. Then Robert will be restricted to playing on some component of $G \backslash S_j$ that has cheating robot number at most j. The rest of the j cops can move into the component containing Robert and then implement a winning strategy to capture Robert. $\square$

First introduced by Beineke and Vandell [3], the *decycling number* of a graph G, denoted $\nabla(G)$, is the minimum number of vertices needed to be deleted from G so that the resulting graph does not contain a cycle. The decycling number gives us another way of bounding the cheating robot number from above.

Corollary 2.1.7. If G is a graph, then $c_{cr}(G) \leq \nabla(G) + 1$.

Proof. By Theorem 2.1.3, we know that any graph without a cycle has a cheating robot number of one. Therefore, from Theorem 2.1.6 we have

$$c_{cr}(G) \leq \min_{1 \leq k \leq |V(G)| - \alpha(G)} \{|S_k| + k\} \leq |S_1| + 1 = \nabla(G) + 1. \qquad \square$$

2.2 Subgraphs

Huggan and Nowakowski [23] asked whether it is true that $c_{cr}(H) \leq c_{cr}(G)$ for any graph G and subgraph H. Here we provide examples where this is not true.

Theorem 2.2.1. There exists a graph G with a connected subgraph H such that $c_{cr}(H) > c_{cr}(G)$.

Proof. Consider the graphs G and H in Figure 2.4.

Suppose Robert is playing against two cops on the graph H and that he starts on a vertex of degree four. It should be noted that there are two other vertices of degree four, v_i and v_j, that are of distance two away from Robert and it is not possible for a single cop to be of distance less than two away from both v_i and v_j. Thus, if Robert waits on a vertex of degree four until a cop moves to that vertex then the cops cannot prevent him from moving to another vertex of degree four. So $c_{cr}(H) > 2$.

Since G contains a 2-core, $c_{cr}(G) \geq 2$ by Theorem 2.1.1. Suppose Robert is playing against two cops, c_1 and c_2, on G. To win the game c_1 starts at v_1 and c_2 starts at v_3. If Robert does not start the game on either v_2 or v_4 then the cops can surround Robert on their first turn. If Robert starts on v_2 then the cop c_2 can move to v_2 which forces Robert to move on his next turn. If Robert then moves closer to v_1, he is surrounded. If Robert instead moves further away from v_1 then, since Robert could not have traversed the edge v_2v_3, Robert's only option is to move to a vertex that is adjacent to only v_2 and v_3. In response, c_1 can move to v_2 and c_2 can move to v_3 to surround Robert. Similarly, if Robert starts on v_4 the cops can surround him in at most two moves. Therefore $c_{cr}(G) = 2 < c_{cr}(H)$. □

Theorem 2.2.1 shows that adding edges to a graph can decrease the cheating robot number. Next, we show that adding both vertices and edges can also decrease the cheating robot number.

Theorem 2.2.2. There exists a graph G with an induced subgraph H such that $c_{cr}(H) > c_{cr}(G)$.

Proof. Consider the graphs G and H in Figure 2.5. If Robert is playing against two cops on the graph H, then he can use the same strategy as in the proof for Theorem 2.2.1 to evade the cops. Thus $c_{cr}(H) > 2$.

By Theorem 2.1.1, $c_{cr}(G) \geq 2$. Suppose Robert is playing against two cops, c_1 and c_2, on G with c_1 starting on v_1 and c_2 starting on v_3. There are five cases, up to symmetry, for Robert's position; either Robert is on v_2, Robert is on a vertex that is distance one away from v_2 and c_2, Robert is of distance one away from c_2 and of

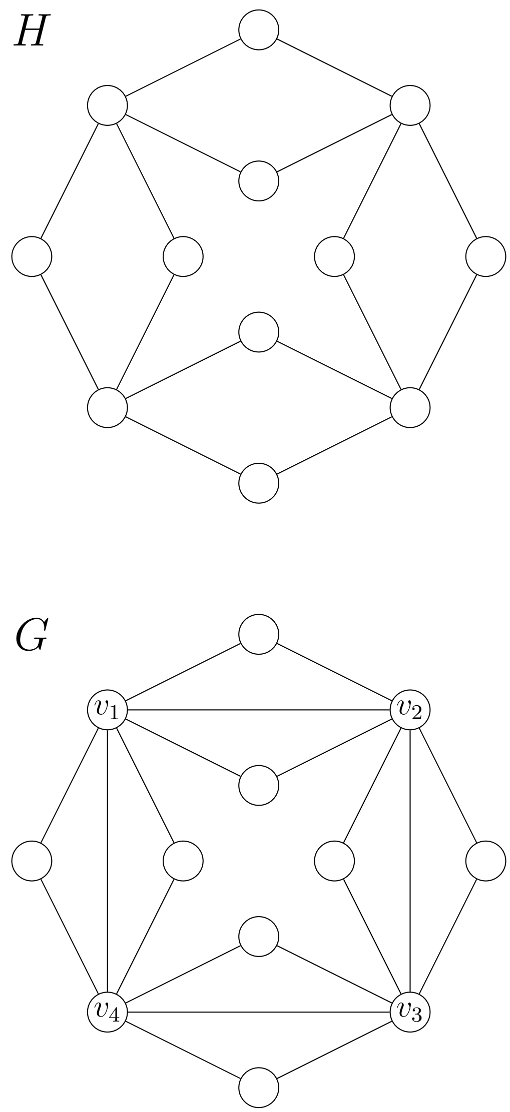


Figure 2.4: A graph, G, with a cheating robot number of two that contains a subgraph, H, with fewer edges and a cheating robot number of at least three.

distance three away from v_2, Robert is of distance three away from c_2 and of distance one away from v_2, or Robert is of distance two away from v_2 and c_2.

Case (i): Suppose Robert is of distance one away from c_2 and distance three away from v_2. Let v_2, x, y, z, v_3 be the path that Robert is on. Then Robert is starting the game at vertex z. For the rest of the game, the cop c_2 will react to how Robert moves and prevent Robert from moving onto either v_2 or v_3. To accomplish this, c_2 will move as follows: if Robert moves to z, c_2 will move to v_3; if Robert moves to y, c_2 will move to the vertex adjacent to both v_2 and v_3; if Robert moves to x, c_2 will move to v_2. With this *shadowing* strategy, c_2 can keep Robert on the path x, y, z indefinitely. Meanwhile, c_1 begins taking the shortest path to Robert, and eventually Robert is surrounded on the vertex z.

Case (ii): Suppose Robert is of distance two away from c_2 and v_2. If c_2 moves to the vertex adjacent to both v_2 and v_3, then she can use the same shadowing strategy from Case (i) to catch Robert.

Case (iii): Suppose Robert is of distance three away from c_2 and distance one away from v_2. On the cops' first turn, c_2 will move to the vertex adjacent to both v_2 and v_3 while c_1 does not move. If Robert moves towards v_3 on his first turn, then c_2 can use the shadowing strategy from Case (i) to catch Robert. If Robert moves to v_2, then c_2 will move to v_2 on her next turn which forces Robert to move off v_2 on his next turn. By the rules of the game, Robert is not allowed to move to the vertex adjacent to both v_2 and v_3. If Robert moves to the vertex adjacent to v_1 and v_2, then he is surrounded. If Robert makes any other move, c_2 can use the shadowing strategy from Case (i).

Case (iv): Suppose Robert is on v_2. On the cops' first turn, c_2 will move to the vertex adjacent to v_2 and v_3 and c_1 will remain at v_1. If Robert does not move, then by using the strategy from Case (iii) Robert is eventually captured. If Robert moves off v_2, then c_2 will move to v_2 and either the cops immediately surround Robert or c_2 can use the shadowing strategy from Case (i).

Case (v): Suppose Robert is adjacent to c_2 and v_2. On the cops' first turn, c_2 will move to the vertex occupied by Robert and c_1 will not move. This forces Robert to move to v_2. From here we are in a scenario described in Case (iv). Thus Robert is eventually captured.

Therefore $c_{cr}(G) = 2 < c_{cr}(H)$. □

While it is not true in general that graphs have higher cheating robot numbers than their subgraphs, it is true in the case when the subgraph is a retract.

Theorem 2.2.3. [23] If G is a graph and $H \subseteq G$ is a retract of G, then $c_{cr}(H) \leq c_{cr}(G)$.

Suppose we want to show that $k \in \mathbb{Z}^+$ cops can win against Robert on a graph G. One way we can do this is to find a graph G' such that G is a retract of G' with retraction map $f : V(G') \to V(G)$ and k cops can capture Robert on G' when he is restricted to moving on G. If the cops can win in this way, a strategy for the k cops to win on G can be obtained by making use of the retraction map f. Suppose on the nth turn, a given cop c is required to move from the vertex u to the vertex v as part of a winning strategy for the cops playing on G' with Robert restricted to G. Consider the change in movement of c on the nth turn of the game when both the cops and Robert are restricted to G, where instead of moving from u to v, c moves from $f(u)$ to $f(v)$. We can create a winning strategy for the cops on G by translating all of the cops' moves on every turn in the same way. This strategy works since either $f(x) = x$ or $f(x)$ is adjacent to x for all $x \in V(G')$. Thus, translating a cop move in G' to a cop move in G by using the retraction map f is always possible. Furthermore, since $f(x) = x$ when $x \in V(G)$, Robert will have at most the same options for moves on each turn as he did when the cops were using the original winning strategy and playing on G'. Therefore, this new strategy that utilizes the retraction map f will result in Robert being captured. This technique of creating a new winning strategy for the cops by using retraction maps is especially useful when studying pursuit-evasion games on grids which we will see in Chapters 5 and 6.

2.3 The push number

Recall that a *winning strategy* for the cops on a graph G is a pair (X, A). The set $X \subseteq V(G)$ is the set of vertices the cops start the game on and A is an algorithm for the cops' moves such that no matter how Robert moves he is eventually captured. Let $\mathcal{S}(G, k)$ denote the set of all cop winning strategies for k cops against Robert on the graph G. Since capturing Robert happens if and only if the cops can surround

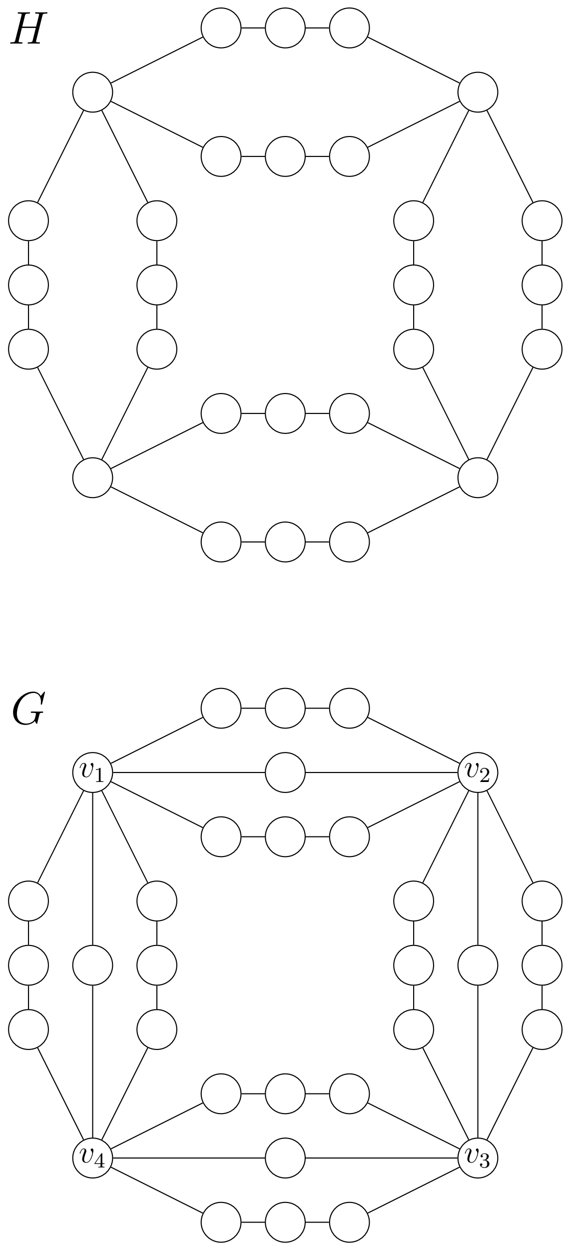


Figure 2.5: A graph, G, with a cheating robot number of two that contains an induced subgraph, H with a cheating robot number of at least three (top).

him by Lemma 1.3.2, and since a strategy for capturing Robert can be changed into a strategy for surrounding Robert by changing the cops' last move, without loss of generality we assume that all of the algorithms for these winning strategies end with the cops surrounding Robert. While following some winning strategy, if a cop moves onto the vertex occupied by Robert forcing him to move and Robert does not lose at the end of his next turn then we say that the cop has *pushed* Robert. In the case where more than one cop moves onto Robert's vertex at the same time, we say that each of those cops has pushed Robert. For a given graph G and a winning strategy S, let $p_{cr}(G, S)$ denote the maximum number of distinct cops that push Robert over all possible ways Robert could play the game. Let $p_{cr}(G) = \min_{S \in \mathcal{S}} p_{cr}(G, S)$. We will call $p_{cr}(G)$ the *push number* of G.

Theorem 2.3.1. If G is a graph, then $p_{cr}(G) \leq c_{cr}(G)$.

Proof. At most the number of cops needed to push Robert is the number of cops needed to capture Robert. □

Since for any graph G, $0 \leq p_{cr}(G) \leq c_{cr}(G)$, the push number is well defined. Next, we give some families of graphs that have a push number of zero.

Theorem 2.3.2. If $n \geq 3$, then $p_{cr}(C_n) = 0$.

Proof. Since $c_{cr}(C_n) = 2$, suppose Robert is playing against two cops, c_1 and c_2, on C_n with vertices $v_0, v_1, \ldots, v_{n-1}$ where v_i is adjacent to v_{i+1} and the addition is modulo n. The cops will begin by placing c_1 on v_0 and c_2 on v_1. The cops will move as follows. If c_1 is on the vertex v_i and Robert is not on v_{i-1}, then c_1 will move to v_{i-1}. If c_2 is on the vertex v_j and Robert is not on v_{j+1}, then c_2 will move to v_{j+1}. By using this strategy, Robert will be surrounded in finitely turns without any pushes. Once he is surrounded, the cops can capture him to end the game without any pushes. □

The following lemma gives a condition that easily guarantees a push number of zero.

Lemma 2.3.3. Let G be a graph. If $c_{cr}(G)$ cops can capture Robert in one move on G, then $p_{cr}(G) = 0$.

Proof. If the $c_{cr}(G)$ cops capture Robert in one move, then Robert will not have been pushed by definition. □

Theorem 2.3.4. If $m, n \geq 2$, then $p_{cr}(K_n) = 0$ and $p_{cr}(K_{m,n}) = 0$.

Proof. By Lemma 2.3.3, it suffices to show that $c_{cr}(K_n)$ and $c_{cr}(K_{m,n})$ cops can win in one move on K_n and $K_{m,n}$ respectively.

Since $c_{cr}(K_n) = n - 1$, Robert has no choice but to place himself on a vertex that is already surrounded by cops. Therefore, the cops win in one move.

Without loss of generality, assume $n \leq m$. Since $K_{m,n}$ contains an n-core, $c_{cr}(K_{m,n}) \geq n$ by Theorem 2.1.1. Let X be the independent set of vertices in $K_{m,n}$ of size m. The cops can win by placing one cop on each vertex not in X. This forces Robert to place himself on a vertex in X. However, every vertex in X is surrounded by cops and so the cops can win in one move. □

While there is no known characterization for when the push number is zero, we can give a simple condition for when it is nonzero.

Theorem 2.3.5. If G is a graph with at least $c_{cr}(G) + 1$ vertices of degree at least $c_{cr}(G) + 1$, then $p_{cr}(G) \geq 1$.

Proof. Robert can begin the game on a vertex of degree at least $c_{cr}(G) + 1$ and then wait until a cop is forced to push him. □

Consider, as an example, the graph G in Figure 2.4. We showed in the proof of Theorem 2.2.1 that $c_{cr}(G) = 2$. Since G has four vertices of degree six, G satisfies the condition in Theorem 2.3.5. Thus, we know $p_{cr}(G) \geq 1$ without determining a winning strategy for the cops.

Note that there exist examples of graphs for which the converse of Theorem 2.3.5 is not true. Let G be the graph in Figure 2.6. Since G is not a tree, $c_{cr}(G) \geq 2$. Two cops can win by beginning on the vertices x and y and then taking the shortest path to Robert's vertex until he is captured. So $c_{cr}(G) = 2$. We claim that regardless of where the cops start, one of them is forced to push Robert. If Robert is able to start on either x or y, then the cops cannot surround Robert without pushing him since $\deg(x) = \deg(y) = 3$. If the two cops start on x and y, then Robert can start on the vertex that is adjacent to x but not adjacent to y. From here, the only way for the cops to win is to push Robert onto the pendant. Therefore $p_{cr}(G) > 0$.

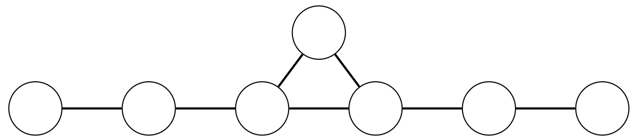


Figure 2.6: A graph illustrating that the converse of Theorem 2.3.5 does not hold.

It was first observed without proof in [23] that the surrounding number is always at least as large as the cheating robot number. Here we will give a proof for this observation.

Lemma 2.3.6. If G is a graph, then $c_{cr}(G) \leq \sigma(G)$.

Proof. Consider the same cop and robber positions in both games and suppose it is the robber's turn to move. In the cheating robot variant, Robert can only move to a vertex that is not occupied by a cop and he cannot traverse an edge that was just traversed by a cop. On the other hand, the robber in the surrounding game is only not allowed to move to a vertex that is occupied by a cop. Therefore, if the cops play the same way in both games, the robber in the surrounding game will always have at least as many options for moves as Robert in the cheating robot variant. Furthermore, we know by Lemma 1.3.2 that the win condition for the cops is the same in both games. Thus every winning strategy for the cops in the surrounding game is a winning strategy for the cops in the cheating robot variant. □

By using the push number, we can also give an upper bound on the surrounding number in terms of the cheating robot number. This gives the following theorem.

Theorem 2.3.7. If G is a graph, then $c_{cr}(G) \leq \sigma(G) \leq c_{cr}(G) + p_{cr}(G)$.

Proof. For convenience, let $k = c_{cr}(G)$. By Lemma 2.3.6, $\sigma(G) \geq k$. Fix a winning strategy S for the cops in the cheating robot variant such that regardless of how Robert plays, at most $p_{cr}(G)$ different cops push Robert before surrounding him.

We describe a winning strategy for $k + p_{cr}(G)$ cops in Surrounding Cops and Robbers as follows. We will label the cops c_1, c_2, $\ldots$, c_k, d_1, d_2, $\ldots$, $d_{p_{cr}(G)}$. To start the game, the cops c_1, c_2, $\ldots$, c_k will place themselves on the vertices used in the strategy S. These cops will play using strategy S. Since $p_{cr}(G) \leq k$ by Theorem 2.3.1, without loss of generality assume that the cops that need to push Robert in order to

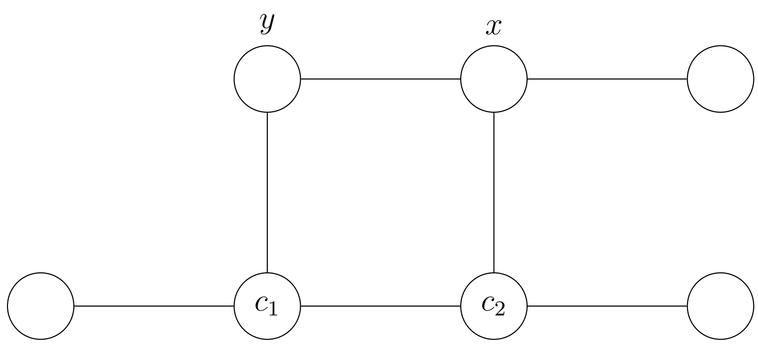


Figure 2.7: A graph G such that $c_{cr}(G) = \sigma(G)$ but $p_{cr}(G) > 0$.

execute the strategy are the cops $c_1, c_2, \ldots, c_{p_{cr}(G)}$. For every $1 \leq i \leq p_{cr}(G)$, the cop d_i will start at the same vertex as c_i and will always move to the vertex occupied by c_i on the previous turn. With this strategy, the robber is unable to traverse an edge previously traversed by $c_1, c_2, \ldots, c_{p_{cr}(G)}$. Since by assumption these k cops are using a strategy that surrounds Robert in the cheating robot variant, the $k + p_{cr}(G)$ cops are able to surround the robber using the same strategy. $\square$

There exist graphs where the upper bound in Theorem 2.3.7 is not tight. Let G be the graph in Figure 2.7. Two cops, c_1 and c_2, can win Surrounding Cops and Robbers by starting on the vertices shown in Figure 2.7. If the robber starts on a leaf or on the vertex y, he gets surrounded in one move. Suppose the robber starts on the vertex x. The cop c_2 can move to x while the cop c_1 moves to the vertex c_2 was just on. This forces the robber to move to either a leaf, where he is surrounded, or to y, where the cops can surround him in one additional move. Therefore $\sigma(G) \leq 2$. Since G contains a cycle, by Theorem 2.1.3 $c_{cr}(G) > 1$ and so $c_{cr}(G) = \sigma(G) = 2$. However, by Theorem 2.3.5, $p_{cr}(G) \geq 1$ since there are three vertices of degree larger than two. So we have $\sigma(G) < c_{cr}(G) + p_{cr}(G)$.

Chapter 3

Planar Graphs

For almost as long as the game of Cops and Robber has existed, mathematicians have been interested in how the cop number behaves with respect to the genus of the graph. It was first shown by Aigner and Fromme [1] that for any planar graph G, $c(G) \leq 3$. In the case where the planar graph is also outerplanar, it was shown by Clarke [12] that $c(G) \leq 2$. For graphs with genus g, Schroeder [38] conjectured that $c(G) \leq g+3$. Currently, the best known upper bound is $c(G) \leq \frac{4}{3}g + \frac{10}{3}$ [9]. Schroeder's conjecture is known to hold for planar graphs [1] and toroidal graphs [30] where $c(G) \leq 3$ for both classes of graphs.

Since there are known bounds for the cop number of a graph based on the graph's genus, it is natural to ask whether bounds exist for the cheating robot number based on the genus of the graph. For Surrounding Cops and Robbers, it was shown by Bradshaw and Hosseini [10] that for any planar graph G, $\sigma(G) \leq 7$. Since we know $c_{cr}(G) \leq \sigma(G)$ for any graph, it follows that a planar graph has a cheating robot number of at most seven. By Theorem 2.1.1, any planar graph with a minimum degree of five will have a cheating robot number of at least five. It is unknown if there exists a planar graph G with $c_{cr}(G) \geq 6$. For outerplanar graphs, it was shown by Huggan and Nowakowski [23] that $c_{cr}(G) \leq 2$. In this chapter, we will use Bradshaw and Hosseini's work in [10] to show that for any planar graph, $c_{cr}(G) \leq 7$ and for any bipartite planar graph, $c_{cr}(G) \leq 4$. We will also discuss a new operation on graphs introduced in [13] which we will use to construct a bipartite planar graph with a cheating robot number of 4.

3.1 Double subdivision

Let G be a graph and let $xy \in E(G)$. A *subdividing* of the edge xy is a graph operation where the edge xy is deleted, a new vertex z is added to the graph, and the edges xz and yz are added to the graph. A *subdivision* of G is a graph operation where every edge of G is subdivided. Clarke, Finbow and Mullen [13] define the graph operation *double subdivision* as follows:

- for every edge $uv \in E(G)$ add a vertex x along with the edges xu and xv; and
- for every edge $uv \in E(G)$, subdivide uv.

Alternatively, double subdividing a graph can be thought of as replacing every edge of the graph with a copy of C_4.

The graph obtained by double subdividing a graph G is denoted $\mathrm{DS}(G)$. As an example, the first graph in Figure 2.4 is isomorphic to $\mathrm{DS}(C_4)$. Since $c_{cr}(C_4) = 2$ and, as it was proven in Theorem 2.2.1, $c_{cr}(\mathrm{DS}(C_4)) = 3$, this example shows that the double subdivision operation can increase the cheating robot number. By double subdividing any tree, we obtain a graph that is no longer a tree since the resulting graph will contain cycles. Therefore, we have infinitely many graphs where double subdividing increases the cheating robot number.

One important property of the double subdivision operation is that it turns planar graphs into bipartite planar graphs.

Lemma 3.1.1. If G is a planar graph, then $\mathrm{DS}(G)$ is a bipartite planar graph.

Proof. Planarity is preserved when applying the double subdivision operation. The graph $\mathrm{DS}(G)$ can always be properly 2-coloured by colouring the vertices that were originally in G red and colouring the vertices that were added in by the double subdivision operation blue. Since a graph is bipartite if and only if it can be properly 2-coloured, $\mathrm{DS}(G)$ is a bipartite planar graph. $\square$

The following lemma gives another important property of double subdividing.

Lemma 3.1.2. Let G be a graph with vertex set V. If $u, v, w \in V \subsetneq V(\mathrm{DS}(G))$, then there does not exist a vertex $x \in V(\mathrm{DS}(G))$ such that x is adjacent to u, v and w.

Proof. Suppose $x \in V$. Then in $\mathrm{DS}(G)$, the distance between x and any other vertex in V is at least two. Now suppose $x \notin V$. That is, x is a vertex that was created by the double subdivision. By the definition of double subdivision operation, x is adjacent to exactly two vertices. Therefore, no vertex in $\mathrm{DS}(G)$ is adjacent to three vertices in V. $\square$

In Section 3.2 we will give an example where the double subdivision operation decreases the cheating robot number. There are no known characterizations for graphs where double subdividing increases the cheating robot number, decreases the cheating robot number, or leaves the cheating robot number unchanged.

3.2 Bipartite planar graphs

Bradshaw and Hosseini [10] proved that for any bipartite, planar graph G, $\sigma(G) \leq 4$. From Theorem 2.3.7, we know that $c_{cr}(G) \leq \sigma(G)$ and so the cheating robot number of any planar graph is at most seven. In this section, for completeness, we go through Bradshaw and Hosseini's proof from the perspective of the cheating robot number instead of the surrounding number.

Lemma 3.2.1. [10] Let $P = v_0, \ldots, v_k$ be a geodesic path with v_i adjacent to v_{i+1} for each $0 \leq i \leq k-1$ in a bipartite graph G. If w is a vertex adjacent to $v_i \in V(P)$, then $d(w, v_0) \in \{i-1, i+1\}$.

Proof. First, we show that $i - 1 \leq d(w, v_0) \leq i + 1$. Since $v_0, \ldots, v_i, w$ is a $v_0 - w$ path with $i + 2$ vertices, we have $d(w, v_0) \leq i + 1$. Suppose for a contradiction that $d(w, v_0) \leq i - 2$. Then there exists a path $v_0, x_1, \ldots, x_{i-3}, w, v_i, \ldots, v_k$ in G with at most $k - 1$ vertices, which contradicts P being a geodesic. So $d(w, v_0) > i - 2$.

Now, suppose for another contradiction that $d(w, v_0) = i$. Then there exists a $j < i$ and a path $v_0, \ldots, v_{j-1}, v_j, y_{j+1}, \ldots, y_{i-2}, y_{i-1}, w$ containing $i + 1$ vertices where $y_\ell \notin V(P)$ for each $j + 1 \leq \ell \leq i - 1$. Since w is adjacent to v_i, let the path $P_1 = v_i, w, y_{i-1}, \ldots, y_{j+1}, v_j$. Note that P_1 contains $i - j + 2$ vertices and the path $P_2 = v_i, v_{i-1}, \ldots, v_j$ contains $i - j + 1$ vertices. Therefore the cycle formed by P_1 and P_2 contains $2i - 2j + 1$ vertices. Thus G contains an odd cycle which contradicts G being bipartite. So $d(w, v_0) \neq i$.

Therefore either $d(w, v_0) = i - 1$ or $d(w, v_0) = i + 1$. □

Let G be a graph with a subgraph H. We will say that H is *geodesically closed* if for every $u, v \in V(H)$, every u-v geodesic in G is contained in H.

Lemma 3.2.2. [10] Let $P = v_0, \ldots, v_k$ be a geodesically closed path with v_i adjacent to v_{i+1} for each $0 \leq i \leq k-1$ in a bipartite graph G. If w is a vertex outside of P that is adjacent to $v_i \in V(P)$, then $d(w, v_0) = i + 1$.

Proof. Since a geodesically closed path is a geodesic, by Lemma 3.2.1 we have that $d(w, v_0) \in \{i-1, i+1\}$. Suppose for a contradiction that $d(w, v_0) = i - 1$. Then there exists a path $w, x_1, x_2, \ldots, v_0$ containing i vertices and so the path P' defined by $P' = v_i, w, x_1, x_2, \ldots, v_0$ contains $i+1$ vertices. The length of P' is i. Thus P' has the same length as the subpath $v_0, \ldots, v_i$ contained in P. This contradicts P being geodesically closed since P' is a v_0-v_i geodesic that is not contained in P. Therefore, $d(w, v_0) = i + 1$. □

We will say that $H \subseteq G$ can be *guarded* with k cops, or that H is *k-guardable*, if, after some finite number of moves, k cops can move on the vertices of H such that if Robert moves onto H then he lands on a vertex occupied by a cop.

Lemma 3.2.3. [10] Let G be a bipartite graph and let $P = v_0, \ldots, v_k$ be a path with v_i adjacent to v_{i+1} for each $0 \leq i \leq k-1$ in G. If P is a geodesic, then P is 2-guardable.

Proof. Let r denote the vertex Robert is on and let $d = d(r, v_0)$. To guard P, two cops c_1 and c_2 will implement the following strategy:

- if $1 \leq d \leq k-1$, then c_1 will move to v_{d-1} and c_2 will move to v_{d+1}, and
- if $d \geq k$ then c_1 will move to v_{k-1} and c_2 will move to v_k.

The above strategy is illustrated in Figure 3.1.

Every time Robert moves, d changes value by at most one. Furthermore, paths are copwin. Thus after finitely many turns, c_1 and c_2 are able to occupy the vertices listed in the above strategy. Suppose c_1 and c_2 are implementing the above strategy and Robert is not on P. If Robert is not adjacent to any vertex in P, then it is not

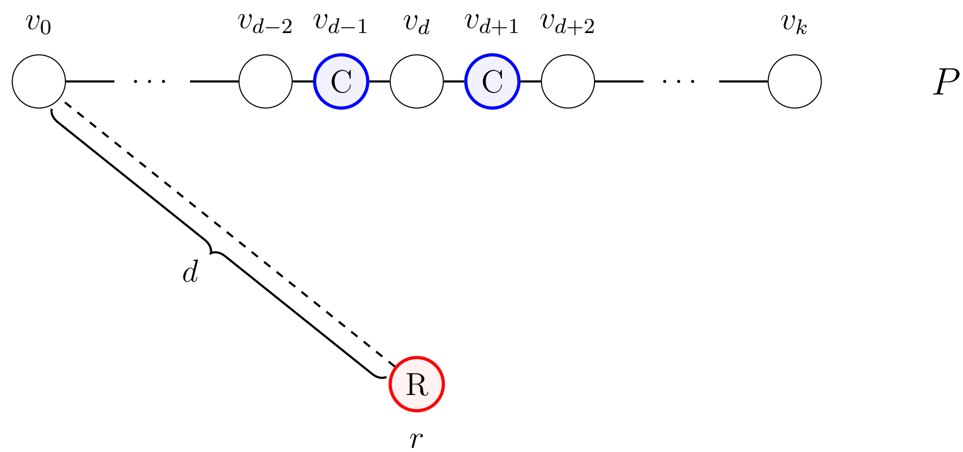


Figure 3.1: A diagram of the cops' strategy to guard a geodesic in a bipartite graph.

possible for him to move onto P in one move. Suppose instead that Robert is adjacent to a vertex on P, say v_i. By Lemma 3.2.1, $d \in \{i-1, i+1\}$. If $d = i+1$ then since P is a geodesic and G is a bipartite graph, v_i is the only vertex Robert could be adjacent to. However, v_i is occupied by c_1 following the above strategy. If $d = i-1$ then since P is a geodesic and G is bipartite, the only two vertices Robert could be adjacent to are v_{i-2} and v_i which are occupied by c_1 and c_2 respectively following the above strategy. So, Robert cannot access P when he is adjacent to a vertex in P.

Suppose c_1 and c_2 move into the positions described by the above strategy while Robert is on P. If Robert is still on P once c_1 and c_2 are in position, then Robert must be on the vertex v_j where $j = d$. To force Robert off P, c_1 will move from v_{j-1} to v_j. After Robert moves off of v_j, by Lemma 3.2.1 either $d = j-1$ or $d = j+1$. If $d = j+1$ after Robert moves off of v_j, then c_1 and c_2 can move into position to implement their strategy. If $d = j-1$, then c_1 will move to v_{j-1} and c_2 will move to v_j. If Robert does not decrease d on his next move, then c_1 and c_2 can move into their positions to implement the above strategy. Otherwise, consider the scenario where Robert continues to decrease d every move. Every time Robert decreases d, c_1 and c_2 will continue to move along P towards v_0. If at any point d increases or stays the same by the end of a Robert move, then c_1 and c_2 can move into position to implement their strategy. If Robert moves onto P at any point, then when c_1 and c_2 move towards v_0, c_1 will move onto the vertex occupied by Robert and force him off of P. This creates the same scenario as before where either d decreases on Robert's next move or c_1 and c_2 can get into position. Suppose Robert decreases d until $d = 0$.

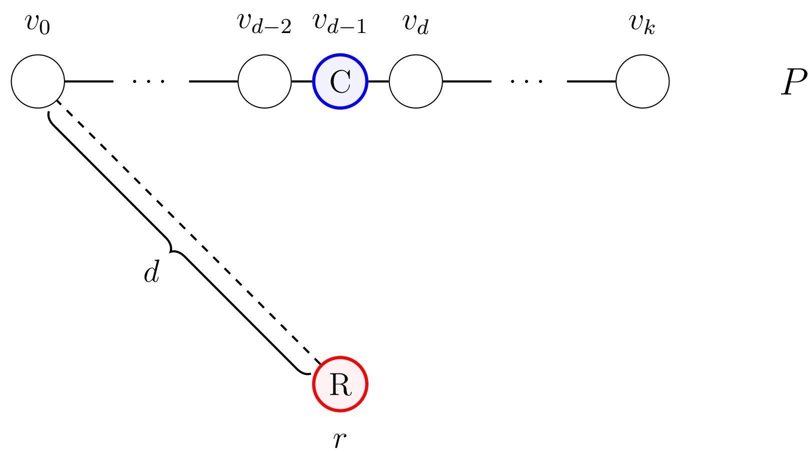


Figure 3.2: A diagram of a cops' strategy to guard a geodesically closed path in a bipartite graph.

That is, Robert moves onto v_0. Then c_1 and c_2 will move onto v_0 and v_1 respectively. On Robert's next move, he is forced to move off of v_0 and so d increases. Thus the cops can move into position on their next move.

Therefore, in finitely many moves, c_1 and c_2 can implement the above strategy to guard P. $\square$

Lemma 3.2.4. [10] Let G be a bipartite graph and let $P = v_0, \ldots, v_k$ be a path with v_i adjacent to v_{i+1} for each $0 \leq i \leq k-1$ in G. If P is geodesically closed, then P is 1-guardable.

Proof. Let r denote the vertex Robert is on and let $d = d(r, v_0)$. To guard P, one cop c will implement the following strategy:

- if $1 \leq d \leq k$, then c will move to v_{d-1}, and
- if $d \geq k+1$ then c will move to v_k.

Figure 3.2 illustrates the above strategy.

Using the same argument as in the proof of Lemma 3.2.3, c can get into position to implement this strategy in finitely many turns. Consider Robert's position when c moves onto the vertex the above strategy instructs her to move to. If Robert is not adjacent to a vertex on P, then he cannot move onto P in one move. If Robert is adjacent to a vertex in P, say v_j, then by Lemma 3.2.2, $d = j+1$ and so c is occupying

v_j. Furthermore, since P is geodesically closed and since G is bipartite, v_j is the only vertex Robert could possibly be adjacent to. Thus Robert cannot access P. If instead Robert is on P, then Robert is on the vertex v_d. In response, c can move from v_{d-1} to v_d until Robert is forced off of P. Once Robert moves off of P, c will already be in position to implement the above strategy. Therefore, in finitely many moves, c can guard P. □

Together, Lemma 3.4 and Theorem 3.5 in [10] prove that for any bipartite planar graph G, $\sigma(G) \leq 4$. The proof of the following theorem is an adaptation of the proofs of Lemma 3.4 and Theorem 3.5 in [10].

Theorem 3.2.5. If G is a connected, bipartite, planar graph, then $c_{cr}(G) \leq 4$.

Proof. In this paragraph, we give an overview of the proof. First, after fixing an embedding of the graph, we will describe a strategy that three cops can use to force Robert to stay within a region enclosed by two paths with one cop guarding one path and two cops guarding the other path. Next we will consider an arbitrary point in the game where Robert is enclosed by two paths with one cop guarding one path and two cops guarding the other path. We will then show that with the fourth cop, in finitely many moves the cops can enclose Robert in a smaller region where the boundary is two paths with one cop guarding one path and two cops guarding the other. By repeating this process, Robert will eventually be in a region containing only a single vertex. Once this occurs, he will be surrounded and the cops will win.

We define the *Robert territory* as the set of all vertices $x \in V(G)$ such that there exists a path from x to the vertex occupied by Robert that does not contain a vertex on a guarded path. Note that since Robert can walk to the vertices in the Robert territory, the subgraph induced by the vertices in the Robert territory is connected.

Fix an embedding of G. First, we show that three cops can enclose Robert in a region. By Theorem 2.1.3, if G is a tree, then $c_{cr}(G) = 1$. Suppose G is not a tree. Since G contains a cycle, let $uv \in E(G)$ such that $G \backslash uv$ is connected. Note that the edge uv is a geodesically closed path in G. Thus, since G is bipartite, by Lemma 3.2.4 one cop can guard uv. Let P be a u-v geodesic in $G \backslash uv$. Since Robert is unable to access uv due to it being guarded by a cop, Robert is restricted to moving within the subgraph $G \backslash uv$. Therefore, since P is a geodesic in the graph Robert is playing on, by Lemma 3.2.3 two cops can guard P. The paths uv and P divide the graph into two regions: the region inside $\{uv\} \cup P$ and the region outside of $\{uv\} \cup P$. Without loss of generality, assume G is embedded onto the plane such that Robert is in the region enclosed by uv and P. Since Robert cannot access the vertices of uv and P because of the cops guarding those two paths, Robert is only able to access vertices enclosed by the two paths.

Now we consider any point in the game where the cops and Robert are positioned in the following way:

- two cops, c_1 and c_2, are guarding an x-y path P_1;
- a cop, c_3 is guarding an x-y path P_2;
- the Robert territory, R, is enclosed by P_1 and P_2;
- P_1 is a geodesic with respect to the induced subgraph $P_1 \cup R$; and
- P_2 is geodesically closed with respect to the induced subgraph $P_2 \cup R$.

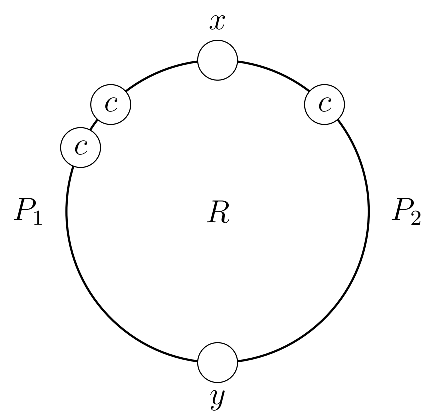
This positioning of the cops is illustrated in Figure 3.3. From here we will show that with the use of a fourth cop c_4, the cops can decrease the Robert territory from R to $R' \subsetneq R$ by transitioning to guarding two x'-y' paths, P_1' and P_2', where P_1' is a geodesic with respect to $P_1' \cup R'$ and P_2' is geodesically closed with respect to $P_2' \cup R'$. There are three cases to consider:

Case (i): P_1 is geodesically closed with respect to $P_1 \cup R$ and there exists an x-y path in $P_1 \cup P_2 \cup R$ distinct from P_1 and P_2;

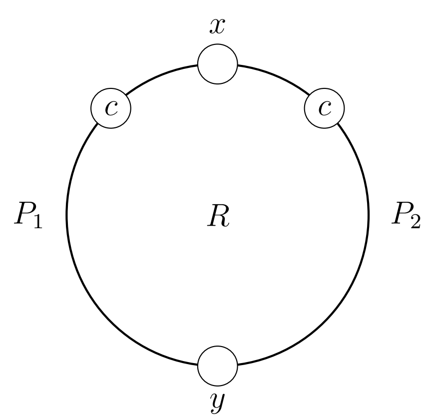
Case (ii): P_1 is geodesically closed with respect to $P_1 \cup R$ and there does not exist an x-y path in $P_1 \cup P_2 \cup R$ distinct from P_1 and P_2; and

Case (iii): P_1 is not geodesically closed with respect to $P_1 \cup R$.

Suppose we are in Case (i). Since there exists an x-y path in $P_1 \cup P_2 \cup R$ distinct from P_1 and P_2, let P_3 be a shortest x-y path distinct from P_1 and P_2. Then P_3 is a geodesic with respect to $P_3 \cup R$ and so it can be guarded with two cops. Since P_1 and P_2 are geodesically closed, only one cop is needed to guard each by Lemma 3.2.4. This leaves two cops to move onto and begin guarding P_3 using the strategy in the proof of Lemma 3.2.3. Furthermore, P_3 splits the Robert territory into smaller



P_1 is not geodesically closed



P_1 is geodesically closed

Figure 3.3: A diagram of the position of the cops before implementing a strategy to decrease the Robert territory.

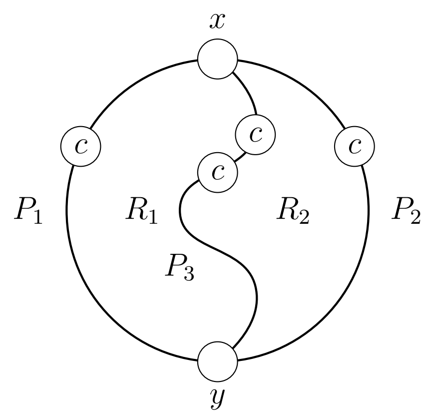


Figure 3.4: A diagram of the outcome of the cops implementing the strategy described in Case (i). Robert is trapped in either the region R_1 or the region R_2.

regions. Without loss of generality, the embedding and orientation of $P_1 \cup P_2 \cup R$ is such that every vertex in $P_1 \backslash \{x, y\}$ is to the left of every vertex in $P_2 \backslash \{x, y\}$ and every vertex in R is within the enclosure of $P_1 \cup P_2$. The path P_3 can be embedded such that for every vertex $a \in V(P_3) \backslash (V(P_1) \cap V(P_2))$, a is not to the left of P_1 and not to the right of P_2. Thus the vertices and edges of P_3 can be drawn such that no edge or vertex is to the left of P_1 nor to the right of P_2. By embedding the paths in this way, it is not possible for a region within $P_1 \cup P_2 \cup R$ to have all three of P_1, P_2 and P_3 on its boundary. Therefore each region is either enclosed by P_1 and P_3 or enclosed by P_2 and P_3. Figure 3.4 illustrates the result of the cops implementing the above strategy. Let R' be the region occupied by Robert. Without loss of generality, assume R' is enclosed by P_1 and P_3. Then we have that P_1 is geodesically closed with respect to $P_1 \cup R'$, P_3 is a geodesic with respect to $P_3 \cup R'$, and P_1 and P_3 are guarded by one and at most two cops respectively. Therefore the game is in a position described by Case (i), (ii), or (iii), with a smaller Robert territory.

Now suppose we are in Case (ii). Let v_r denote the vertex occupied by Robert. Since there is no x-y path distinct from P_1 and P_2 in $P_1 \cup P_2 \cup R$, we claim that there exists a unique vertex $z \in V(P_1 \cup P_2)$ such that for any vertex $r \in R$ either r is adjacent to z or r is not adjacent to any vertex in $P_1 \cup P_2$. If no vertex in $P_1 \cup P_2$ is adjacent to a vertex in R, then G would be disconnected, which is a contradiction. Therefore, there exists at least one vertex in $P_1 \cup P_2$ that is adjacent to at least one vertex in R. Suppose for a contradiction there exists two vertices in $P_1 \cup P_2$ that are adjacent to vertices in R. Let $x = v_0, v_1, \ldots, v_k$ be the vertices of the path P_1 where v_i is adjacent

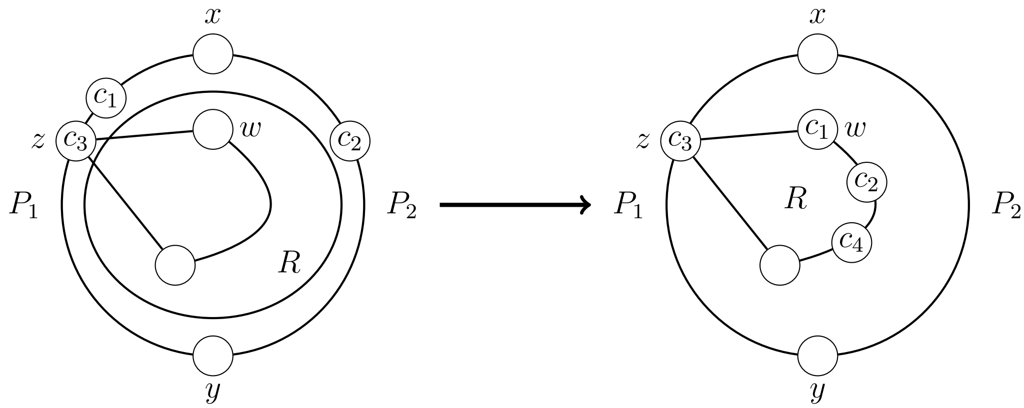


Figure 3.5: A diagram of the cops implementing the strategy to decrease the Robert territory in Case (ii).

to v_{i+1} for each $0 \leq i \leq k-1$. Without loss of generality, assume two distinct vertices, v_i and v_j with $i < j$, in P_1 are adjacent to at least one vertex in R. Since R is connected, there exists a path $v_i = r_1, r_2, \ldots, r_\ell = v_j$ where for each $2 \leq r \leq \ell - 1$, $r_i \in R$. Thus the path formed by the vertices $v_0, \ldots, v_i, r_2, \ldots, r_{\ell-1}, v_j \ldots, v_k$ is an x-y path in $P_1 \cup P_2 \cup R$ that is distinct from P_1 and P_2 which contradicts the assumption of being in Case (ii). Thus the claim holds. Let z be the vertex described in the claim. Without loss of generality, assume $z \in V(P_1)$.

Since P_1 and P_2 are geodesically closed, only one cop is needed to guard each path. Let c_1 be the cop guarding P_1 and c_2 be the cop guarding P_2. Let c_3 and c_4 be the other two cops. To decrease the Robert territory, the cops begin their strategy with c_3 moving to z. There are two cases for z, either z is adjacent to more than one vertex in R or exactly one vertex in R. Suppose z is adjacent to more than one vertex in R. Let w be one of the vertices in R that is adjacent to z. Since c_3 is on z and the only vertex in P_1 that is adjacent to any vertex in the Robert territory is z, Robert cannot access P_1. So, c_1 can stop guarding P_1 and move onto w. Since there are no vertices in R that are adjacent to any vertex in P_2, c_2 does not need to guard P_2. Thus c_2 and c_4 can move onto and guard a z-w geodesic P_1' in the subgraph induced by the vertices $R \backslash \{z, w\}$ using the strategy in the proof of Lemma 3.2.3. Since the edge zw is a geodesically closed path, either c_1 or c_3 can guard zw using the strategy in the proof of Lemma 3.2.4. This setup is illustrated in Figure 3.5.

Suppose instead z is adjacent to only one vertex, y, in R. In this case c_3 will move onto y. If $\deg(y) = 2$, c_3 will keep moving onto vertices in R she had not previously occupied. If every vertex in R has degree at most two, then the subgraph induced by the vertices in R is a path. Thus, c_3 will capture Robert in finitely many moves. Suppose instead that c_3 moves onto a vertex of degree at least three. Let y' be this vertex of degree at least three and let w' be one of the vertices adjacent to y' that c_3 has not yet occupied. From here, the cops can use a similar setup as before where one cop guards the edge $y'w'$ and two cops guard a y'-w' geodesic in $R\backslash\{y', w'\}$. In both cases, after the cops have set up, Robert is enclosed in a region $R' \subsetneq R$ by two paths, P_1' and P_2', with P_2' being geodesically closed with respect to $P_2' \cup R'$ and guarded by one cop, and P_1' being a geodesic with respect to $P_1' \cup R'$ and guarded by two cops. Thus the position of the game is in either Case (i), (ii) or (iii), with a smaller Robert territory.

Finally, suppose we are in Case (iii). Let $x = v_0, v_1, \ldots, v_k = y$ be the vertices of P_1 where v_i is adjacent to v_{i+1} for all $0 \leq i \leq k-1$. Let c_1 and c_2 be the two cops guarding P_1, let c_3 be the cop guarding P_2 and let c_4 be the last cop that has yet to be assigned a strategy. We assume that c_1 and c_2 are using the strategy from the proof of Lemma 3.2.3 and c_3 is using the strategy from the proof of Lemma 3.2.4. That is, when Robert is distance $d \leq k-1$ away from v_0, c_1 and c_2 occupy the vertices v_{d-1} and v_{d+1} respectively. Since P_1 is not geodesically closed, there exists a geodesic with respect to $P_1 \cup R$ that is not contained in P_1 and has endpoints in P_1. Out of all such geodesics, let S be a geodesic from v_i to v_j, with $i < j$, such that $j - i$ is minimized. Assume that G is embedded such that S is within the region enclosed by P_1 and P_2. Furthermore, assume that out of all v_i-v_j geodesics not contained in P_1, S minimizes the region enclosed by S and P_1. Let $\aleph$ be the region enclosed by P_1 and S. Let $P_1(i,j)$ be the subpath $v_i, v_{i+1}, \ldots, v_j$ contained in P_1.

Let P_3 be the x-y path formed by the vertices in $(V(P_1)\backslash V(P_1(i,j))) \cup V(S)$. Figure 3.6 gives an illustration of all the paths. Note that since S is a v_i-v_j geodesic with respect to $P_1 \cup R$, P_3 contains the same number of vertices as P_1. We label the vertices of P_3 to $z_0 = v_0, \ldots, z_k = v_k$. The four cops' strategy will be to transition from two cops guarding P_1 to two cops guarding P_3 while one cop remains on P_1. The cop c_3 will continue guarding P_2 during this transition. We will show that during this transition, Robert will not be able to move onto P_1. We will also show that once the transition is finished, the Robert territory will have decreased and the position of the game will be in either Case (i), (ii) or (iii).

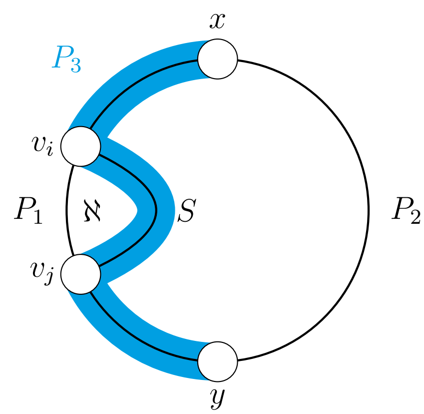


Figure 3.6: A diagram of P_1, P_2, P_3, S, and $\aleph$ in Case (iii).

First, we claim that if $\alpha \in R$ is adjacent to $v_\ell \in V(P_1(i,j))\backslash\{v_i, v_j\}$ then it holds that $d(\alpha, v_0) = \ell + 1$. By Lemma 3.2.1, if α is adjacent to $v_\ell \in V(P_1(i,j))\backslash\{v_i, v_j\}$ then $d(\alpha, v_0) \in \{\ell - 1, \ell + 1\}$. Note that by our embedding of G, α is in the region $\aleph$. Suppose for a contradiction that $d(\alpha, v_0) = \ell - 1$. Then there exists a α-v_0 geodesic with respect to $P_1 \cup R$ of length $\ell - 1$. Let $\alpha = \alpha_1, \alpha_2, \ldots, \alpha_\ell = v_0$ be the vertices of this α-v_0 geodesic. The path formed by the vertices $v_\ell, \alpha_1, \ldots, \alpha_\ell$ is a v_ℓ-v_0 path of length ℓ and so this is a v_ℓ-v_0 geodesic. Furthermore, this v_ℓ-v_0 geodesic shares a vertex v_m with P_1 for some $0 \leq m \leq \ell - 1$. So we have a v_ℓ-v_0 geodesic of the form $v_\ell, \alpha_1, \ldots, v_m, v_{m+1}, \ldots, v_0$. Note that since there exists one such v_ℓ-v_0 geodesic, there may exist many such geodesics. Out of all of these v_ℓ-v_0 geodesics, let S' be the subpath $v_\ell, \alpha_1, \ldots, v_m$ of length $\ell - m$ from the v_ℓ-v_0 geodesic that maximizes m. There are two cases for m, either $i \leq m < \ell - 1$ or $m < i$. If $i \leq m < \ell - 1$, then the length of S' is less than the length of S which contradicts our choice of S. Suppose $m < i$. Since S' and S must intersect due to α being within the region $\aleph$, by the planarity of G and by our choice of S minimizing the region between S and $P_1(i,j)$, S and S' share a vertex, say w. Then the path formed by the vertices $v_\ell, \alpha_1 \ldots, w, \ldots, v_i$, where every vertex listed before w is on the path S' and every vertex listed after w is on the path S, is a v_ℓ-v_i geodesic. The existence of this v_ℓ-v_i geodesic contradicts our choice of the v_i-v_j geodesic S minimizing $j - i$. Therefore, $d(\alpha, v_0) \neq \ell - 1$ and so $d(\alpha, v_0) = \ell + 1$.

Let v_r denote the vertex occupied by Robert and let $d = d(r, z_0)$. Since c_1 and c_2 are using the strategy from the proof of Lemma 3.2.3 to guard P_1, we can assume that c_1 occupies v_{d-1} and c_2 occupies v_{d+1}. The cops will begin transitioning from

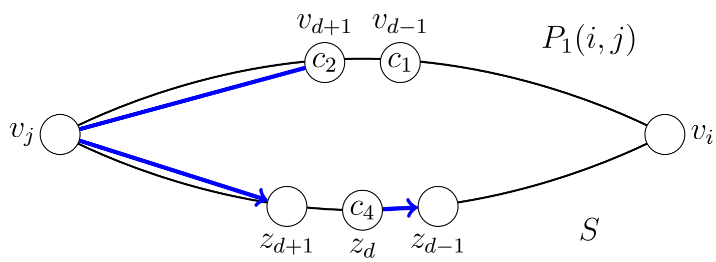


Figure 3.7: A diagram of the cops' strategy in Case (iii) when the distance between Robert and v_0 stays within the interval (i, j).

guarding P_1 to guarding P_3 as follows. First, c_4 will move to z_d in finitely many moves. Next, c_2 will move to z_{d+1} in finitely many moves while P_1 stays guarded. To show that this strategy results in decreasing the Robert territory, we consider the value of d while the cops are executing their strategy. Suppose d stays within the interval (i, j) during the entire execution of the cops' strategy. If Robert is not adjacent to a vertex in P_1, then Robert cannot move onto P_1 in one move. If instead Robert is adjacent to a vertex $v_\ell \in V(P_1)$, then by the previous claim $d = \ell + 1$ and so c_1 occupies v_ℓ. By using the strategy from the proof of Lemma 3.2.4, c_1 can guard P_1. Since $d(v_{\ell+2}, v_j) < j - \ell - 1$, c_2 can make it to v_j before Robert can. Thus Robert cannot access P_1 while c_2 executes her strategy. After c_2 moves onto v_j, c_2 will move towards z_{d+1}. On the cop turn that c_2 moves to z_{d+1}, c_4 will move from z_d to z_{d-1}. This way, c_2 and c_4 can use the strategy from the proof of Lemma 3.2.3 to guard P_3. Figure 3.7 illustrates the movements of c_2 and c_4. Once the cops have finished executing their strategy, Robert will either be in the region enclosed by P_1 and P_3, or he will be in the region enclosed by P_3 and P_2.

Now we consider the cases where d does not stay within the interval (i, j) during the execution of the cops' strategy. Suppose that at some point during the cops' execution of their strategy, d attains the value i. On the cop turn immediately following d attaining the value i, c_1 moves to $v_{i-1} = z_{i-1}$ and c_4 moves to z_{i+1}. Note that since $d \leq i$, by the embedding of the graph Robert is not in the region $\aleph$ and is instead in the region enclosed by P_3 and P_2. Therefore, if c_1 and c_4 use the strategy from the proof of Lemma 3.2.3 to guard P_3, Robert will not be able to access P_3. Suppose instead that at some point during the cops' execution of their strategy, d attains the value j. Then on the cop turn immediately following d attaining the value j, c_2 can move to $v_{j+1} = z_{j+1}$ and c_4 can move to z_{j-1}. Since $d \geq j$, Robert is not in the region

$\aleph$ and is instead in the region enclosed by P_3 and P_2. Therefore, c_2 and c_4 can use the strategy from the proof of Lemma 3.2.3 to guard P_3. In both of the subcases where d does not stay within the interval (i, j), Robert is, in finitely many turns, enclosed in the region bounded by the paths P_3 and P_2. Let R' be this region. Since $R' \subsetneq R$, P_2 is geodesically closed with respect to $P_2 \cup R'$ and guarded by one cop, and P_3 is a geodesic with respect to $P_3 \cup R'$ that is guarded by two cops, the position of the game coincides with either Case (i), (ii) or (iii), with a smaller Robert territory.

Since the cops are able to repeatedly decrease the Robert territory, eventually the Robert territory will contain only a single vertex. Once this occurs, Robert will be surrounded. Thus the cops can capture Robert by Lemma 1.3.2. □

Next, we will show that the upper bound in Theorem 3.2.5 is tight. We will do this by making use of the double subdivision operation on a planar graph with a "high" cheating robot number to obtain a bipartite planar graph with a cheating robot number of four.

Theorem 3.2.6. There exists a planar, bipartite graph with a cheating robot number of four.

Proof. Consider the double subdivided icosahedron, $\mathrm{DS}(I_{20})$, as illustrated in Figure 3.8. We will show that three cops are not enough to capture Robert on $\mathrm{DS}(I_{20})$.

There are more than three vertices of degree ten, so Robert can begin the game on one of these vertices. Robert's strategy will be to wait at this vertex until a cop moves to his vertex. Note that Robert is of distance two away from five other vertices of degree ten and Robert is able to move closer to any one of them on his next move without traversing an edge that was just traversed by a cop. It is not possible for a single cop to be distance one away from more than two of these vertices by Lemma 3.1.2. Therefore, there exists a vertex of degree ten that Robert can move to before any of the cops can. Once Robert has done this, by the symmetry of the graph Robert can repeat the strategy of waiting until he is forced to move indefinitely. □

The strategy Robert uses to beat three cops in the proof of Theorem 3.2.6 will work on any double subdivided 5-regular planar graph. Suppose G is a 5-regular planar graph. Then in $\mathrm{DS}(G)$, one cop can be adjacent to at most two vertices of degree ten by Lemma 3.1.2. Thus two cops can access at most four vertices of degree

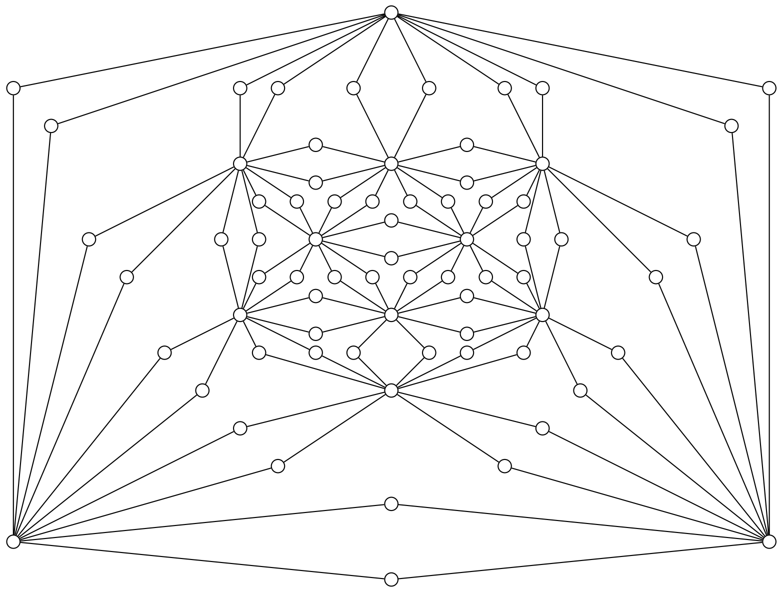


Figure 3.8: The graph DS(I_{20}).

ten in one move. Therefore if Robert waits at a vertex of degree ten until he is pushed by a cop, then he can move to another vertex of degree ten in two moves for the same reason he was able to in the proof of Theorem 3.2.6. There are infinitely many connected, 5-regular planar graphs, see [20] for a way of constructing all connected, 5-regular planar graphs, and so there are infinitely many bipartite planar graphs with a cheating robot number of four.

Chapter 4

Complexity

For an introduction to complexity theory, see [17]. Berarducci and Intrigila [4] were the first to prove that for any fixed positive integer k, the problem of determining for a graph G whether $c(G) \leq k$ is in P. This fact has also been proven in other papers, see for example [6, 14, 19]. In the case where k is not fixed, Goldstein and Reingold [18] proved that determining whether $c(G) \leq k$ is EXPTIME-complete when the initial positions of the cops and the robber are given. They conjectured that even when the initial positions are not given, determining whether $c(G) \leq k$ is EXPTIME-complete. Fomin, Golovach, Kratochvíl, Nisse and Suchan [16] showed that the decision problem is NP-hard and Mamino [31] showed that the decision problem is PSPACE-hard. Kinnersley [29] proved Goldstein and Reingold's conjecture and showed that determining whether $c(G) \leq k$ is indeed EXPTIME-complete. The computational complexities of different variations of Cops and Robber have also been studied such as Surrounding Cops and Robbers [11], Zombies and Survivors [28] and generalized Cops and Robber games [7]. Because complexity results are known for Cops and Robber and many of its variants, in this chapter we will explore the complexity of the cheating robot variant. We will show that determining whether $c_{cr}(G) \leq k$ for any graph G and for any fixed $k \in \mathbb{Z}^+$ can be done in polynomial time.

4.1 Cops and Cheating Robot complexity

Recall that a *loop* is an edge, a 2-element multisubset of the vertex set, where both vertices in the edge are the same. A graph G is *reflexive* if $\{v, v\} \in E(G)$ for every $v \in V(G)$. For convenience, we will assume that all graphs in this chapter are reflexive. This will allow us to say that a vertex is adjacent to itself. Recall that for an integer $n \in \mathbb{Z}^+$, the graph $\boxtimes_{k=1}^n G$ is defined to be the strong product of the graph G with itself n times. For convenience, we will shorten the notation $\boxtimes_{k=1}^n G$ to $\boxtimes^n G$. Vertices of $\boxtimes^n G$ will be written as n-vectors $(x_1, \ldots, x_n)$. The *tensor product* of two graphs G and H, denoted $G \times H$, has vertex set $V(G \times H) = \{(u, v) \mid u \in V(G), v \in V(H)\}$ and edge set $E(G \times H) = \{\{(u, v), (x, y)\} \mid ux \in E(G) \text{ and } vy \in E(H)\}$. When both G and H are reflexive graphs, the graph products $G \times H$ and $G \boxtimes H$ are isomorphic. While the theorems and proofs in this section can be written using the tensor product instead of the strong product, we will use the strong product to be consistent with the paper [6] that our work is based on.

Bonato, Chiniforooshan and Prałat [6] showed that determining whether the cop number of a graph was less than or equal to a fixed integer k can be solved in polynomial time. Here we use the same technique to show the analogous decision problem for the cheating robot number can be solved in polynomial time.

Theorem 4.1.1. Let $2^{(V(G),V(G))}$ denote the power set of $\{(u, v) \mid u, v \in V(G)\}$. Let $k \in \mathbb{Z}^+$. For a graph G, $c_{cr}(G) > k$ if and only if there exists a function $\psi : (V(\boxtimes^k G), V(\boxtimes^k G)) \to 2^{(V(G),V(G))}$ with the following properties.

(i) For all $T_1T_2 \in E(\boxtimes^k G)$,

$$\psi((T_1, T_2)) \neq \emptyset.$$

(ii) For all $T_1 = (v_1, \ldots, v_k), T_2 = (u_1, \ldots, u_k) \in V(\boxtimes^k G)$ such that $T_1T_2 \in E(\boxtimes^k G)$,

$$\begin{aligned} \psi((T_1, T_2)) \subseteq \{(r_1, r_2) \mid & r_1 \in V(G) \backslash (V_{T_1} \cup S_{T_1}), \\ & r_2 \in V(G) \backslash (V_{T_2} \cup S_{T_2} \cup X_{T_1,T_2,r_1}), \\ & r_1r_2 \in E(G)\} \end{aligned}$$

where V_{T_1} (analogously V_{T_2}) is the set of all vertices in the k-tuple T_1 (T_2), S_{T_1} (S_{T_2}) is the set of all vertices that can be surrounded by T_1 (T_2) in one move, and $X_{T_1,T_2,r_1} = \{v_i \in T_1 \mid r_1 = u_i \in T_2\}$.

(iii) For all $T_1, T_2, T_3 \in V(\boxtimes^k G)$ such that $T_1T_2, T_2T_3 \in E(\boxtimes^k G)$,

$$\psi_2((T_1, T_2)) \subseteq \psi_1((T_2, T_3))$$

where $\psi_2((T_1, T_2))$ is the set of all second entries of $\psi((T_1, T_2))$ and $\psi_1((T_2, T_3))$ is the set of all first entries of $\psi((T_2, T_3))$.

Before proving Theorem 4.1.1, here we explain the meaning of Theorem 4.1.1. Every vertex of the graph $\boxtimes^k G$ corresponds to a position for k cops on the graph G. Two adjacent vertices in $\boxtimes^k G$ corresponds to a legal move that the cops can make on G. The purpose of ψ is to map legal cop moves to legal Robert moves that allow him to win the game. Condition (i) ensures that a legal move for Robert exists given any cop move. Condition (ii) ensures that ψ outputs only Robert moves that do not result in him being captured by the cops by the end of their next move. More specifically, condition (ii) has ψ only output Robert moves where he does not start on a vertex occupied by a cop, moves where he does not end his turn on a vertex occupied by a cop, moves where he cannot be surrounded by the cops by the end of their next move, and moves where he is not traversing an edge that the cops traversed. Condition (iii) ensures that for any two consecutive cop moves, Robert can make two consecutive moves that allow him to evade capture.

Proof. For two vertices $T_1, T_2 \in V(\boxtimes^k G)$, instead of writing $\psi((T_1, T_2))$ we will write $\psi(T_1T_2)$ for convenience.

Suppose $c_{cr}(G) > k$ and Robert is playing against k cops on G. For all vertices $T_1, T_2 \in V(\boxtimes^k G)$, define $\psi(T_1T_2)$ to be the set of all ordered pairs of vertices (r_1, r_2) with $r_1, r_2 \in V(G)$ such that if the cops start at T_1 and move to T_2, then Robert can win by starting on r_1 and then moving to r_2. Since $c_{cr}(G) > k$, Robert can always win regardless of how the cops play. Thus $\psi(T_1T_2) \neq \emptyset$. Furthermore, we know that r_1 is not one of the vertices occupied by the cops in T_1, the cops cannot surround r_1 in one move when moving off of T_1, r_2 is not one of the vertices occupied by the cops in T_2, the cops cannot surround r_2 in one move when they move off of T_2, and as Robert moves from r_1 to r_2 he does not traverse an edge that the cops traverse moving from T_1 to T_2. Therefore properties (i) and (ii) hold.

Let $T_1, T_2, T_3 \in V(\boxtimes^k)$ such that T_1 is adjacent to T_2 and T_2 is adjacent to T_3. That is, (T_1, T_2) and (T_2, T_3) are two consecutive cop moves. Let $(r_1, r_2) \in \psi(T_1T_2)$.

If Robert is on r_1 when the cops are on T_1, then Robert can win when the cops move to T_2 by moving to r_2. That is, if the cops move from T_2 to T_3 Robert can move to some vertex $r_3 \in V(G)$ and continue his winning strategy. So $(r_2, r_3) \in \psi(T_2T_3)$. Thus property (iii) holds.

Now suppose a mapping ψ exists satisfying properties (i), (ii) and (iii). Using induction, we now construct a winning strategy for Robert against k cops by using ψ. Let $T_t \in V(\boxtimes^k G)$ denote the position of the cops in round t of the game. If the cops begin the game at T_0, Robert can begin the game on a vertex r_0 in the second coordinate of a pair of vertices in $\psi(T_0T_0)$. We know such a vertex exists since, by property (i), $\psi(T_0T_0) \neq \emptyset$. By property (ii), $r_0 \notin V_{T_0} \cup S_{T_0}$ and so there is no cop on r_0 and the cops cannot surround Robert in one move. Assume that Robert is able to move along edges in $\psi(T_{t-1}T_t)$ for all rounds $t \leq a$ where $a \geq 0$ is fixed. If the cops move from T_a to T_{a+1}, then by property (iii) Robert, starting from some vertex $r_a \in \psi_2(T_{a-1}T_a)$, can move to an adjacent vertex r_{a+1} such that $(r_a, r_{a+1}) \in \psi(T_aT_{a+1})$. By property (ii), $r_{a+1} \in V_{T_{a+1}} \cup S_{T_{a+1}} \cup X_{T_aT_{a+1}}$. Thus r_{a+1} does not have a cop on it, the cops are not able to capture Robert in one move from T_{a+1}, and Robert is not traversing an edge that the cops traversed while moving from T_a to T_{a+1}. Therefore, Robert can avoid capture in the $(a+1)$th round. So, by the principle of mathematical induction, Robert can indefinitely avoid capture. $\square$

In step 1 of Algorithm 1, we begin by defining ψ to be a function that maps every cop move to all possible Robert moves that do not immediately result in a Robert loss. Then we delete entries from ψ_2 until the conditions of Theorem 4.1.1 are satisfied. Whenever an entry of ψ_2 is deleted, say the ith entry, then the ith entry of ψ_1 will also be deleted. Consequently, if either $\psi_1(T_1T_2) = \emptyset$ or $\psi_2(T_1T_2) = \emptyset$, then $\psi(T_1T_2) = \emptyset$. If by the time Algorithm 1 finishes, $\psi(T_1T_2) = \emptyset$ for some cop move $T_1T_2 \in E(\boxtimes^k G)$, this tells us that Robert has no safe moves he can make and so $c_{cr}(G) \leq k$. If, by the time the algorithm finishes, for every cop move (T_1, T_2) where $T_1, T_2 \in V(\boxtimes^k G)$ we have $\psi(T_1T_2) \neq \emptyset$, then Robert is able to always safely move regardless of how the cops play and so $c_{cr}(G) > k$.

Theorem 4.1.2. Algorithm 1 runs in polynomial time.

Proof. Note that $|V(\boxtimes^k G)| = n^k$ and $|E(\boxtimes^k G)| \leq \binom{n^k}{2} = O(n^{2k})$. During each iteration of the repeat loop on lines 2–7, $|\psi(T_1T_2)|$ will decrease by at least one for

Algorithm 1 CHECK CHEATING ROBOT NUMBER k

Require: $G = (V, E), k \geq 0$

1: initialize $\psi(T_1T_2)$ to $\{(r_1, r_2) \mid r_1 \in V(G)\backslash(V_{T_1} \cup S_{T_1}), r_2 \in V(G)\backslash(V_{T_2} \cup S_{T_2} \cup X_{T_1T_2}), r_1r_2 \in E(G)\}$ for all $T_1, T_2 \in V(\boxtimes^k G)$
2: **repeat**
3:     **for all** $T_1, T_2, T_3 \in V(\boxtimes^k G)$ such that $T_1T_2, T_2T_3 \in E(\boxtimes^k G)$ **do**
4:         $\psi(T_1T_2) \leftarrow \psi(T_1T_2) \cap \{(r_1, r_2) \mid r_1 \in \psi_1(T_1T_2), r_2 \in \psi_2(T_1T_2) \cap \psi_1(T_2T_3)\}$
5:         $\psi(T_2T_3) \leftarrow \psi(T_2T_3) \cap \{(r_1, r_2) \mid r_1 \in \psi_1(T_2T_3) \cap \psi_2(T_1T_2), r_2 \in \psi_2(T_2T_3)\}$
6:     **end for**
7: **until** the value of ψ is unchanged
8: **if** there exists $T_1, T_2 \in V(\boxtimes^k G)$ such that
9:     $\psi(T_1T_2) = \emptyset$
10: **then**
11:     **return** $c_{cr}(G) \leq k$
12: **else**
13:     **return** $c_{cr}(G) > k$
14: **end if**

some (T_1, T_2) where $T_1, T_2, \in V(\boxtimes^k G)$ except for the last iteration of the loop where ψ is unchanged. Since there are $(n^k)(n^k) = n^{2k}$ ways of choosing an ordered pair of vertices in $\boxtimes^k G$, (T_1, T_2), and since $|\psi(T_1, T_2)| = O(n^{2k})$, the repeat loop in lines 2–7 will finish in at most $O(n^{4k})$ steps.

Now we consider the number of steps within one iteration of the repeat loop. There are at most n^{3k} ways to choose $T_1, T_2, T_3 \in V(\boxtimes^k G)$ such that $T_1T_2, T_2T_3 \in E(\boxtimes^k G)$. There are at most n^k vertices in $\psi_1(T_1, T_2)$ and $\psi_2(T_1, T_2)$ for a given pair of vertices (T_1, T_2). Calculating the intersection between two sets of vertices can be done in $O(n^{2k})$ and calculating the intersection between two sets of ordered pairs of vertices can be done in $O(n^{4k})$ steps. Thus in each iteration of the for loop, there are at most $O(n^{4k})O(n^{2k}) = O(n^{6k})$ steps.

Therefore we have a total of

$$O(n^{4k})O(n^{3k})O(n^{6k}) = O(n^{13k})$$

steps for Algorithm 1. So Algorithm 1 finishes in polynomial time. $\square$

Chapter 5

Bodyguards and Presidents

In this chapter we define a new pursuit-evasion game called Bodyguards and Presidents that will be used to obtain results in Chapter 6. This new model plays a role in the cheating robot variant when considering the game on strong products. Recall that we defined the strong product in Section 1.4. When taking the strong product of two graphs, say G and H, the resulting graph $G \boxtimes H$ contains a subgraph that is isomorphic to G, $G.\{v\}$, for each $v \in V(H)$. Suppose v_1 and v_2 are two adjacent vertices in H. The question that motivates the Bodyguards and Presidents model is as follows: how can the cops prevent Robert from moving from $G.\{v_1\}$ to $G.\{v_2\}$ in the graph $G \boxtimes H$? When playing on $G \boxtimes H$, we know that if Robert is on $u \in V(G.\{v_1\})$, then Robert is adjacent to every vertex in $N_{G.\{v_2\}}[u]$. So if the cops want to stop Robert from moving onto $G.\{v_2\}$ from $G.\{v_1\}$ they need at least $\deg_{G.\{v_2\}}(u) + 1$ cops on $G.\{v_2\}$. However, if Robert moves from (u, v_1) onto some other vertex in $G.\{v_1\}$, say (u', v_1), and the cops still want to prevent Robert from moving onto $G.\{v_2\}$, they have to go from occupying every vertex in $N_{G.\{v_2\}}[u]$ to occupying every vertex of $N_{G.\{v_2\}}[u']$.

For example, let G be the graph in Figure 5.1 and consider $G \boxtimes H$ for any graph H. Suppose Robert is on the vertex $(x, v) \in V(G \boxtimes H)$. If $v' \in V(H)$ is adjacent to v in H and the cops want to prevent Robert from moving onto $G.\{v'\}$, they need to occupy every vertex in $N_{G.\{v'\}}[x]$. This can be done by occupying four vertices. Now suppose Robert moves from (x, v) to (y, v). Four cops are now needed to occupy the vertices in $N_{G.\{v'\}}[y]$. However, as illustrated in Figure 5.1, it is not possible to go from occupying all of the vertices $N_G[x]$ to occupying all of the vertices in $N_G[y]$

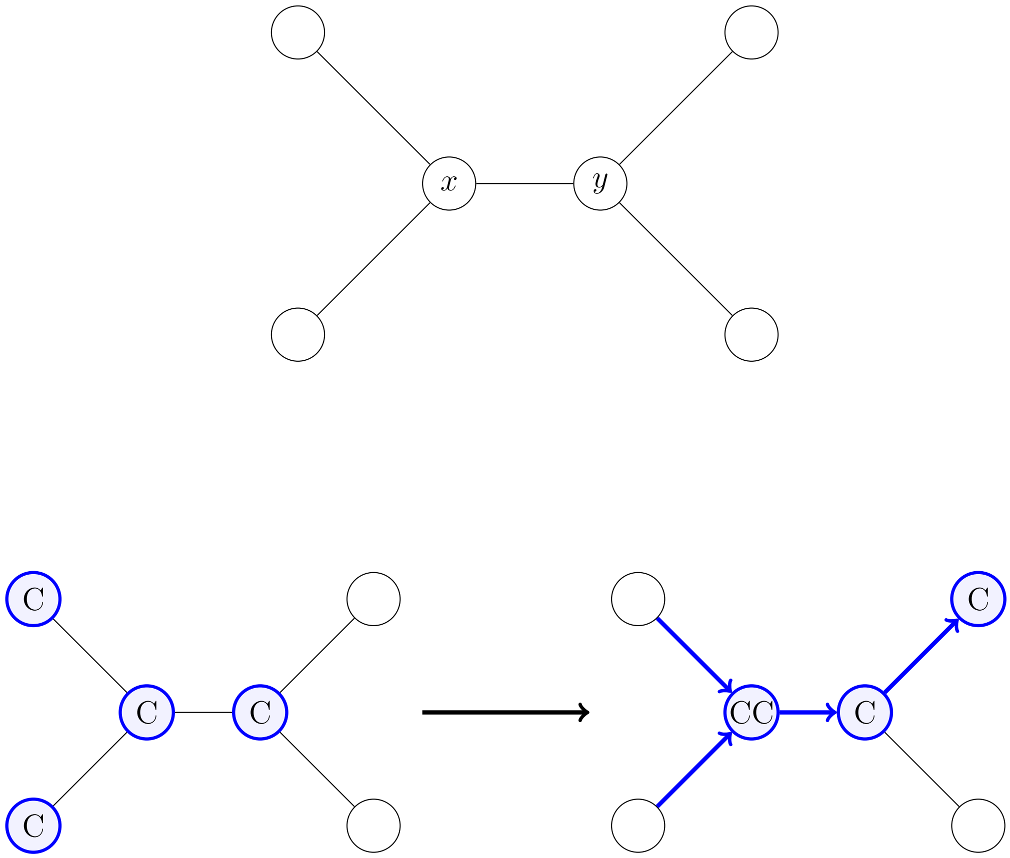


Figure 5.1: A graph with two vertices x and y such that if four cops are covering $N[x]$, then they cannot cover $N[y]$ in one move.

in one move. This gives Robert an opening to move onto $G.\{v'\}$ which means that more cops were needed to prevent Robert moving from $G.\{v\}$ to $G.\{v'\}$. The goal of the Bodyguards and Presidents model is to quantify how many cops are needed to accomplish the task of occupying closed neighbourhoods of vertices indefinitely regardless of how Robert moves.

5.1 Rules for Bodyguards and Presidents

Bodyguards and Presidents is played with two players on the vertices of a graph. One player controls a set of bodyguards while the other player controls a president. To help distinguish the bodyguards and the president, the bodyguards will be given the pronouns she/her and the president will be given the pronouns he/him. At the beginning of the game, the player controlling the bodyguards chooses which vertices they start on followed by the player controlling the president choosing their starting vertex. Once all of the bodyguards and the president are placed on the graph, the players take turns moving with the bodyguards moving first. A legal move in this game for either a bodyguard or a president consists of either staying at the vertex they were already occupying or moving to an adjacent vertex. For the player controlling the bodyguards, when it is their turn they are allowed to move any number of bodyguards and there are no restrictions on which bodyguards are allowed to move. If there exists an $N \in \mathbb{Z}^+$ such that on the Nth turn for the bodyguards they surround the president and for every $n \geq N$, by the end of the bodyguards nth turn they surround the president, the bodyguards win. We say that the bodyguards can *indefinitely surround the president* if after a finite number of turns, say $N \in \mathbb{Z}^+$, the bodyguards can surround the president at the end of each of their moves after their Nth turn. If the president can avoid this, in other words there always exists a u-v walk the president can take starting on u such that the bodyguards cannot surround him in one move once he moves to v, then the president wins.

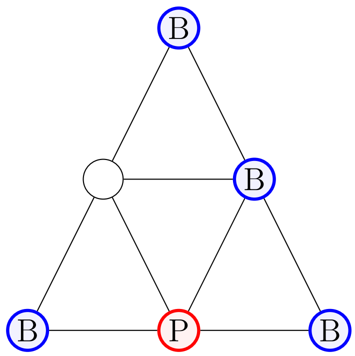
We note that while Bodyguards and Presidents has many similarities to Cops and Robber, there are some important differences. In Bodyguards and Presidents, it does not make a difference to either the bodyguards or the president if a vertex is occupied by both a bodyguard and a president. On the other hand, in Cops and Robber having a cop and a robber occupying the same vertex is the cops' goal. In Cops and Robber, the game ends when the cops' win condition is satisfied. However, in Bodyguards and Presidents, regardless of who wins the game, there are an infinite number of rounds. If the bodyguards surround the president, the bodyguards and the president continue playing. The bodyguards have to surround the president by the end of infinitely many, consecutive bodyguard turns, but the cops only have to capture the robber a single time in Cops and Robber.

Let $B(G)$ denote the minimum number of bodyguards needed to win Bodyguards and Presidents on the graph G. We will call $B(G)$ the *bodyguard number* of G. In the case where G is the empty graph, we define $B(G) = 0$. If a bodyguard is placed on every vertex of a graph, then the president is always surrounded regardless of how he plays. Thus the parameter $B(G)$ is well defined. Figure 5.2 illustrates the first few rounds of a game with four bodyguards. Since the game never ends, it is not possible to draw every round. However, seeing how the first few rounds of the game can play out can sometimes be enough to show who wins. Consider the starting positions of the bodyguard and the president in Figure 5.2. It is shown that from that position, the bodyguards can surround the president in one move. From there, the president has two options; either move to a vertex of degree two or degree four. If the president moves to a vertex of degree four, the bodyguards can reposition themselves in a way that is rotationally symmetrical to the position shown in (b). If the president moves to a vertex of degree two, the bodyguards can surround him. Thus if the president has any hope of winning from here, he must move off his vertex of degree two and onto a vertex of degree four. If the president does so, the bodyguards can again position themselves in a way that is rotationally symmetrical to the position shown in (b). Therefore, if the bodyguards and the president start the game as shown in (a), the president will lose. It is easy to see that regardless of where the president starts on the graph in Figure 5.2, the bodyguards can surround the president in one move such that the position of the bodyguards and the president is rotationally symmetric to one of the positions shown in (c). Thus the bodyguards can indefinitely surround the president regardless of where he starts. So the bodyguard number of this graph is at most four. In Section 5.2, we will introduce lower bounds of the bodyguard number that will prove that the bodyguard number of this graph is exactly four.

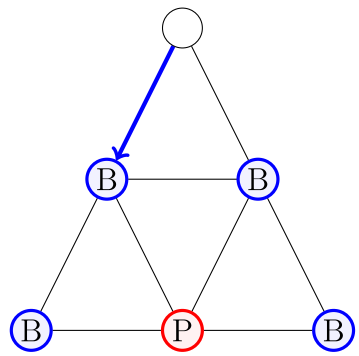
By the way the game is defined, we only require the bodyguards to occupy the president's open neighbourhood in order to win. Instead, we could have considered a similar game where instead of surrounding the president indefinitely, the bodyguards are required to occupy every vertex in the president's closed neighbourhood by the end of each of their moves indefinitely. When using the Bodyguards and Presidents game in Chapter 6 to obtain results for the cheating robot number on strong products of graphs, we will need the cops to occupy the closed neighbourhood of Robert's shadow in order to capture him. Let $B[G]$ denote the fewest number of bodyguards needed to win this version of Bodyguards and Presidents on the graph G. The following lemma shows that $B[G]$ attains only two possible values.

Theorem 5.1.1. If G is a graph, then $B(G) \leq B[G] \leq B(G) + 1$.

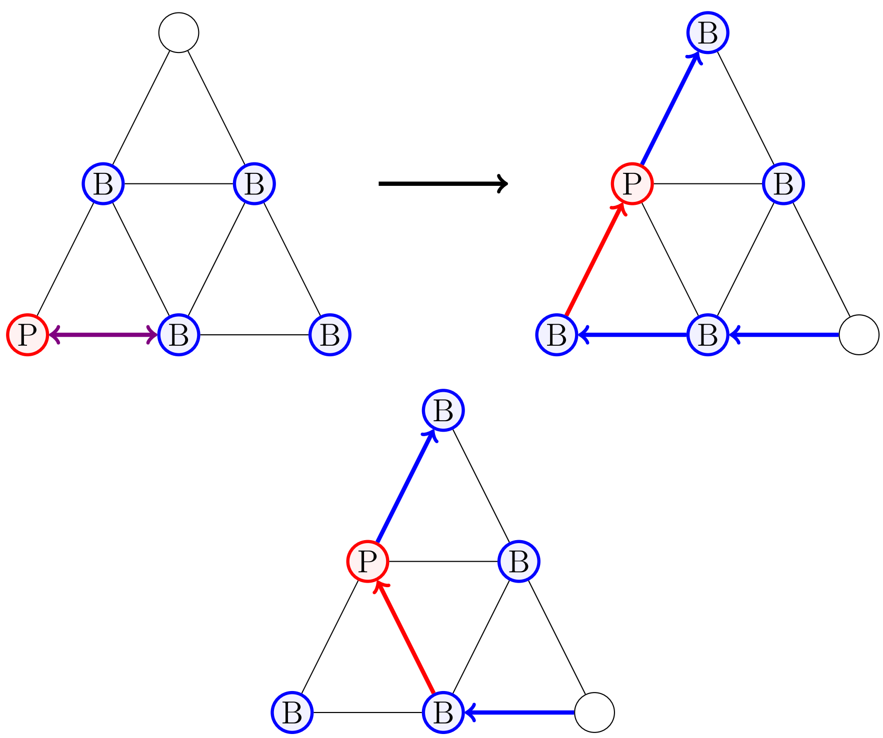
Proof. Suppose $B(G) + 1$ bodyguards are aiming to indefinitely occupy the closed neighbourhood of the president. If they first use a cop winning strategy, then after finitely many moves one bodyguard can always move to the vertex occupied by the president. The remaining $B(G)$ bodyguards can use a bodyguard winning strategy



(a) Initial placement of the bodyguards and the president.



(b) Bodyguards surround the president in one move.



(c) Regardless of how the president moves, the bodyguards can surround him.

Figure 5.2: Example of Bodyguards and Presidents being played.

to surround the president indefinitely. Thus, after finitely many moves, the $B(G)+1$ bodyguards can always occupy the closed neighbourhood of the president.

Suppose for a contradiction that $B(G) > B[G]$. Then if $B[G]$ bodyguards are playing Bodyguards and Presidents, the president can always find a u-v walk in G such that by starting on u and ending at v the bodyguards cannot surround the president on their turn immediately after the president finishes the walk. However, if $B[G]$ bodyguards cannot surround the president after the u-v walk, then they cannot occupy the closed neighbourhood of the president. So the $B[G]$ bodyguards can never indefinitely occupy the president's closed neighbourhood, which is a contradiction. Therefore $B(G) \leq B[G]$. □

5.2 General bounds

In this section we give a lower bound on the bodyguard number, we characterize all graphs G with $B(G) = |V(G)| - 1$, and we give a result analogous to Theorem 2.2.1 for the bodyguard number.

Lemma 5.2.1. For any graph G, $B(G) \geq \max\{\Delta(G), \sigma(G)\}$.

Proof. On any given turn the president can either move to any adjacent vertex or he can remain at the vertex he is currently occupying. The president's moves are not restricted by the position of the bodyguards. Comparatively, the robber in Surrounding Cops and Robbers has the same movement options as the president with the exception that he cannot move to a vertex occupied by a cop. Thus, if the robber has a winning strategy against k cops where $k < \sigma(G)$, the president can use the same strategy against k bodyguards to avoid being surrounded. Therefore, at least $\sigma(G)$ bodyguards are needed to win.

If the president stays on the vertex of maximum degree, the bodyguards cannot win without first surrounding the president on that vertex. Thus, $B(G) \geq \Delta(G)$. □

Naturally, the next question to ask is how large the bodyguard number can be for any graph. Consider any graph on n vertices. From Lemma 5.2.1, the bodyguard number is at least the maximum degree of the graph. However, the maximum degree can be as large as $n-1$. So there are graphs that need at least $n-1$ bodyguards.

We claim that the bodyguards can always win with $n-1$ bodyguards on any graph with n vertices. Suppose $n-1$ bodyguards are placed on the graph with no two bodyguards sharing the same vertex. If the president is placed on the lone vertex without a bodyguard, then he is surrounded. If he is placed on any vertex occupied by a bodyguard, then the bodyguards can shift themselves along a path from the president to the vertex that has no bodyguard on it. This results in bodyguards occupying every vertex except the one occupied by the president, and so the president is surrounded. Whenever the president moves after being surrounded, he will move to a vertex with a bodyguard on it. That bodyguard can then move back to the vertex the president came from to surround him again. Thus, the president gets surrounded by the end of every bodyguard turn indefinitely. So $B(G) \leq n-1$ for any graph G.

Since we know $n-1$ bodyguards will always win, which graphs need exactly that many for the bodyguards to win? As an immediate consequence of the previous lemma, we have the following.

Lemma 5.2.2. If G is a graph on n vertices with $\Delta(G) = n-1$, then $B(G) = n-1$.

Proof. From Lemma 5.2.1, $B(G) \geq \Delta(G) = n-1$. □

A graph on n vertices has maximum degree $n-1$ if and only if the graph contains a vertex that is adjacent to all other vertices in the graph. Such a vertex is called a *universal vertex*.

Theorem 5.2.3. For any graph G on n vertices, $B(G) = n-1$ if and only if $\Delta(G) = n-1$.

Proof. Consider the cases when $n \leq 3$. The only graph with one vertex is the empty graph which we defined to have a bodyguard number of zero. Furthermore, the only graphs that have a bodyguard number of zero are empty graphs since a bodyguard number of zero implies a maximum degree of zero by Lemma 5.2.1. So the theorem holds for $n = 1$. If G is connected and contains exactly two vertices, then G is P_2. Since one bodyguard can win on P_2 by always moving to the vertex not occupied by the president, $B(G) = 1$. Furthermore, for a graph G to have $B(G) = 1$, the maximum degree is at most one by Lemma 5.2.1. Thus either G is empty, which we have established has a bodyguard number of zero, or G is P_2, which indeed has a bodyguard number of one. Thus the theorem holds for $n = 2$. Suppose G has three vertices. If $\Delta(G) = 2$, then G is either the path P_3 or the complete graph K_3. By Lemma 5.2.1, $B(P_3) \geq 2$ and $B(K_3) \geq 2$. Two bodyguards can win on P_3 and K_3 by occupying whichever two vertices the president is not on and so $B(P_3) = B(K_3) = 2$. If $\Delta(G) < 2$, then either G is an empty graph or G has one edge. If G is an empty graph then $B(G) = 0$. If G has one edge then G has one isolated vertex and two adjacent vertices. One bodyguard is needed in case the president starts on the two adjacent vertices and no bodyguards are needed if the president starts on the isolated vertex. Thus $B(G) = 1$ when G only has one edge. Therefore the theorem holds when $n \leq 3$. From here, we assume that G contains at least four vertices.

Lemma 5.2.2 establishes sufficiency. To prove the other direction, we show the contrapositive is true. That is, if G does not have a universal vertex then $B(G) \leq n-2$. First, we will show that the bodyguards can surround the president in finitely many moves such that for any pendant in G that is not occupied by the president, there is at most one pendant that is not occupied by a bodyguard. Afterwards, regardless of how the president moves, we show that the bodyguards can indefinitely surround the president by making use of the fact that, excluding the vertex occupied by the president, there is at most one pendant without a bodyguard on it and the bodyguards can maintain this condition for the entirety of the game.

Suppose we are playing on G with $n - 2$ bodyguards. Begin by placing one bodyguard on every pendant in G. Note that if G has $n - 1$ pendants, then G is a star which means G has a universal vertex. So G has at most $n - 2$ pendants. Then place the rest of the bodyguards on all but two of the remaining vertices of G such that at most one bodyguard occupies every vertex in G. Let x and y be the two vertices that are not occupied by bodyguards. There are five cases for the initial placements of the bodyguards and the president. Each case is illustrated in Figure 5.3.

Case (i): The president is on a vertex that does not contain a bodyguard and is completely surrounded by bodyguards. That is, the president is starting the game on either x or y and x is not adjacent to y. This is the position we want the bodyguards to be in. So for their first turn, they will not move. Since x and y are not pendants, we have that every pendant is occupied by a bodyguard.

Case (ii): Every vertex in the president's closed neighbourhood has a bodyguard on it. Let $v_1, v_2, \ldots, v_n$ be a path in G such that v_1 is the vertex occupied by the president and $v_n = x$. Let b_i be the bodyguard on v_i. On the bodyguards' first turn, for each $1 \le i \le n-1$ the bodyguard b_i will move from v_i to v_{i+1}. This results in a bodyguard on every vertex in the president's closed neighbourhood except for the vertex he is occupying and there is at most one bodyguard on every vertex. Since y is not a pendant and the vertex the president starts on may be a pendant, we have that at most one pendant is not occupied by a bodyguard.

Case (iii): The president is adjacent to a vertex that does not have a bodyguard on it and shares a vertex with a bodyguard. Without loss of generality, assume the president is adjacent to x. On the first bodyguard turn, the only bodyguard that will move is the one on the president's vertex and she will move to x. For the same reason as in Case (ii), after the bodyguard moves there is at most one pendant that does not contain a bodyguard.

Case (iv): The president is on a vertex without a bodyguard on it and is adjacent to a vertex that is not occupied by a bodyguard. Without loss of generality, assume the president is on x and so he is adjacent to y. Since y is not a pendant, there exists a vertex y' adjacent to y that has a bodyguard on it. On the bodyguards first turn, the bodyguard on y' will move to y which results in the president being surrounded. This also leaves x and y' to be the only vertices to not contain a bodyguard. Since x is not a pendant and y' may be a pendant, at most one pendant is not occupied by a bodyguard.

Case (v): The president is adjacent to both x and y. Suppose y is adjacent to a vertex z where $z \neq x$ and z is not the vertex that the president is starting on. We know such a vertex can exist since we are assuming G contains at least four vertices. The bodyguard that is occupying the same vertex as the president can move to x and the bodyguards can use the same strategy from Case (iv) to surround the president. Besides possibly the vertex occupied by the president, this leaves at most one pendant unoccupied. If x is adjacent to a vertex z where $z \neq y$ and z is not the vertex that the president is starting on, then a similar strategy can be applied to surround the president in one turn.

Now suppose instead that x is only adjacent to y and the president's vertex while y is only adjacent to x and the president's vertex. Let v_p denote the vertex the president is starting on. Let $\alpha_1 \notin N[v_p]$ and let $\alpha_1, \alpha_2, \ldots, \alpha_{k-2}, v_p = \alpha_{k-1}, x = \alpha_k$ be a α-x path containing v_p. Let b_i be the bodyguard on α_i for each $1 \le i \le k-1$. On

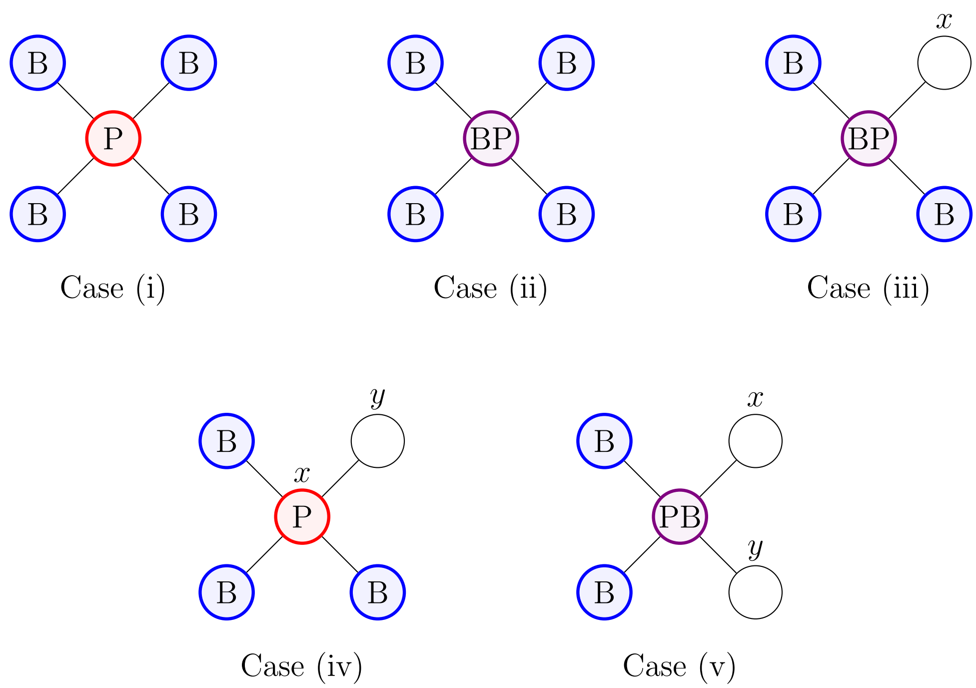


Figure 5.3: Illustrations of the five cases in the proof of Theorem 5.2.3.

the bodyguards' first turn, the bodyguard b_i will move from α_i to α_{i+1}. This results in v_p and x having a bodyguard on it and the only two vertices not occupied by bodyguards are α_1 and y. Now we consider the president's first turn. If the president moves onto x, moves onto y, or stays at v_p, then the president is adjacent to only one vertex that is not occupied by a bodyguard and he shares a vertex with a bodyguard. So, the bodyguards can use the strategy from Case (iii) to surround the president. If the president moves to some vertex that is not v_p, x or y, then at the end of the president's move either every vertex in his closed neighbourhood contains a bodyguard or the president is adjacent to α_1 and he shares a vertex with a bodyguard. From here the bodyguards can use either Case (ii) or Case (iii) respectively to surround the president in one move.

Note that by the end of each case, the bodyguards have positioned themselves such that excluding the vertex occupied by the president which may or may not be a pendant, there is at most one pendant that is not occupied by a bodyguard. Furthermore, by the end of each case the president is surrounded by bodyguards and there is no bodyguard on the president's vertex. Now, consider the kth president move where $k \geq 1$ and assume that the bodyguards are positioned such that out of all the pendants that do not contain the president, at most one of them is not occupied by a bodyguard. Every move the president makes from this position has three cases; either the president does not move, the president moves to a vertex that is adjacent to one vertex that is not occupied by a bodyguard, or he moves to a vertex that is adjacent to two vertices that are not occupied by bodyguards. Let v be the vertex the president is moving from and let u be the vertex the president is moving to. Next we will show that the bodyguards can respond to the president's move in both cases such that they maintain the condition that out of all the pendants that do not contain the president, at most one of them is not occupied by a bodyguard.

If v and u are the same vertex, then the bodyguards will not move. On the other hand, suppose the only vertex adjacent to u that does not have a bodyguard on it is v. After the president moves to u, the bodyguard on u can move to v, resulting in the president being surrounded.

Finally, suppose u is adjacent to a vertex other than v that does not have a bodyguard on it, say w. We claim that it is not possible for both v and w to both be pendants. Suppose for a contradiction that they are. Consider the moment the president first occupied v. If the president was placed onto v during the setup of the game, then by the end of the first bodyguard turn w would still have had a bodyguard on it. If the president moved onto v during the game from u, then as the president moved from u to v the only bodyguard that moved the next turn was the bodyguard that was already on v. Therefore, if both v and w are not occupied by a bodyguard, then when the president was at u there was no bodyguard at w and so he was not surrounded. This contradicts the outcomes of the strategies the bodyguards have been implementing up until this point in the game. So, at most one of v or w is a pendant.

Thus, at least one of v or w is adjacent to a vertex other than u that has a bodyguard on it. Since u is not a universal vertex, without loss of generality assume that w is adjacent to a vertex q such that $q \notin N[u]$ and a bodyguard occupies q. After the president moves from v to u, the bodyguard on u can move to v and the bodyguard on q can move to w. Since u is not a vertex of degree one, at most one vertex of degree one is not occupied by a bodyguard. Therefore, regardless of how the president moves every following turn, the bodyguards can use one of the two strategies described above to continuously surround the president. So, $n - 2$ bodyguards can

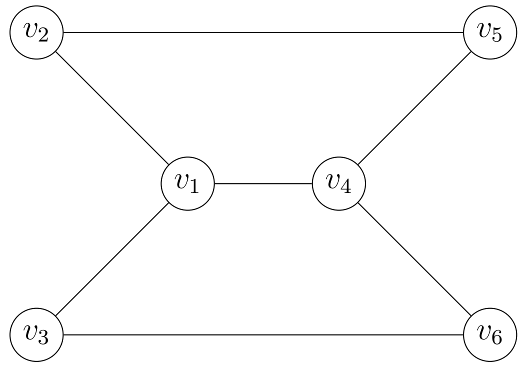


Figure 5.4: The graph G with $B(G) = 3$ in the proof of Theorem 5.2.4.

win against the president on G when $\Delta(G) < n - 1$. □

In Chapter 2, we showed that there exists a graph with a smaller cheating robot number than one of its subgraphs. The same holds true for the bodyguard number.

Theorem 5.2.4. There exists a graph G with a connected subgraph H such that $B(H) > B(G)$.

Proof. Let H be the graph in Figure 5.1 and let G be the graph in Figure 5.4. At the beginning of the chapter, we showed that $B(H) > 3$. To show that $B(H) > B(G)$, we will show that $B(G) \leq 3$.

Let b_1, b_2 and b_3 be the three bodyguards. Consider the two positions b_1 at v_1, b_2 at v_5 and b_3 at v_6; and b_1 at v_4, b_2 at v_2 and b_3 at v_3. We will call the positions A_1 and A_2 respectively. If the president is on one of v_1, v_5 or v_6, then the president will be surrounded when the bodyguards are in position A_2. If the president is on one of v_2, v_3 or v_4, then the president will be surrounded when the bodyguards are in position A_1. Note that the bodyguards can move from A_1 to A_2 in one move. Therefore, the bodyguards can surround the president every round indefinitely. So $B(G) \leq 3$. □

By Lemma 5.2.1 and the proof of Theorem 5.2.4, the bodyguard number of the graph in Figure 5.4 is exactly three. In Section 5.3, we will prove a theorem that tells us the graph in Figure 5.1 has a bodyguard number of four.

Theorem 2.2.3 shows how the cheating robot number behaves with respect to retracts. The following theorem shows that the bodyguard number behaves the same way.

Theorem 5.2.5. If H is a retract of the graph G, then $B(H) \leq B(G)$.

Proof. Let $r : V(G) \to V(H)$ be a retraction map from G to H. Suppose we have $B(G)$ bodyguards on H. Given a winning strategy for the bodyguards on G, we claim that the bodyguards on H can win by mapping the bodyguards' moves on G using the retraction map r. That is, if the winning strategy for a bodyguard in G would require her to move to the vertex $v \in V(G)$, when playing on H she will move to the vertex $r(v) \in V(H)$.

Retracts preserve adjacency and so if a bodyguard moves from u to v when playing on G, the bodyguard can move from $r(u)$ to $r(v)$ when playing on H. Thus it suffices to show that following this strategy when playing on H will result in the president being indefinitely surrounded after a finite number of moves. Since $B(G)$ bodyguards can win on G, $B(G)$ bodyguards can also win on G when the president is restricted to playing on the subgraph H. Fix a winning strategy for the $B(G)$ bodyguards when playing on G while the president is restricted to playing on H. Suppose it takes $N \in \mathbb{Z}^+$ bodyguard moves to begin indefinitely surrounding the president when playing on G. Let $v \in V(H)$ be the vertex the president is on immediately following the Nth bodyguard move. Since the president is surrounded on v, every vertex in $N(v)$ is occupied by a bodyguard. Since r is a retraction map, for each $u \in N_G(v)$ we have $r(u) \in N_H(v)$. Furthermore, for each $u \in N_G(v)$ either $r(u) = u$ or $r(u)$ is adjacent to u and so $|N_G(v)| \geq |N_H(v)|$. Thus if every vertex in $N_G(v)$ is occupied by a bodyguard, then when playing on H every vertex in $N_H(v)$ will be occupied by a bodyguard. So when playing on H, the bodyguards will have the president surrounded at the end of their Nth move. This argument shows that if the bodyguards have surrounded the president when the bodyguards are on G and the president is restricted to H, then the bodyguards have surrounded the president when everyone is restricted to H.

Now we consider any president move after the bodyguards' Nth move. Suppose in the game where the bodyguards are playing on G while the president is restricted to H that on the president's mth move, where $m \geq N$, the president moves from x to y where $x, y \in V(H)$. We assume that the bodyguards have been using the strategy of mapping their moves on G from their fixed winning strategy onto H up until the president's mth move. Since the bodyguards, when playing on G, surround the president after he moves to y, by mapping the bodyguards' moves using r we have that the president is surrounded when everyone is restricted to H.

Therefore, by the principle of mathematical induction, when playing on H, the bodyguards can indefinitely surround the president. □

In the above proof, we have the bodyguards play on a graph G, restrict the president to a retract of G, H, and then map the bodyguards' movements by the retraction map from G to H. This is an adaptation of a classic proof technique in the Cops and Robber literature. In the classic proof technique, both the bodyguards and the president play on the retract H and the bodyguards' moves are the image of the winning strategy on G. We will use the same proof technique from the proof of Theorem 5.2.5 for Lemmas 5.4.2 and 5.4.3.

5.3 Bodyguard numbers for some graph families

In this section, we determine the bodyguard numbers of some simple families of graphs.

Theorem 5.3.1. If $n \geq 2$, then

$$B(P_n) = \begin{cases} 1 & \text{if } n = 2 \\ 2 & \text{if } n \geq 3. \end{cases}$$

Proof. One bodyguard can win on P_2 by always moving to the vertex not occupied by the president. From Lemma 5.2.1, for any $n \geq 3$ we have $B(P_n) \geq \Delta(P_n) = 2$. Let b_1 and b_2 be the two bodyguards playing against the president on P_n and let $v_1, v_2, \ldots, v_n$ be the vertices of P_n where v_i is adjacent to v_{i+1} for each $1 \leq i \leq n-1$. The bodyguard b_1 will start on v_1 while b_2 will start on v_n. Let v_i denote the vertex the president is occupying on any given round. Since the president is restricted in how far he can move in each direction, in finitely many moves the bodyguard b_1 can move onto v_{i-1} and the bodyguard b_2 can move onto v_{i+1}. Once the president is surrounded by the two bodyguards, regardless of whether he moves to v_{i-1} or v_{i+1} the bodyguards can follow him and continue to surround him. If the president moves onto v_1 then b_1 will stay on v_1 and if the president moves onto v_n then b_2 will stay on v_n. Using this strategy, the bodyguards win the game. Therefore, $B(P_n) = 2$ for all $n \geq 3$. □

Theorem 5.3.2. If $n, m \in \mathbb{Z}^+$, then $B(K_{n,m}) = \max\{n, m\}$.

Proof. From Lemma 5.2.1, $B(K_{n,m}) \geq \Delta(K_{n,m}) = \max\{n, m\}$.

Without loss of generality, assume $n \geq m$. Let $X, Y \subsetneq V(K_{n,m})$ be the independent sets of vertices in $K_{n,m}$ such that $|X| = n$, $|Y| = m$ and $X \cap Y = \emptyset$. If the president moves onto Y, then the bodyguards can move to X such that one bodyguard occupies each vertex of X. If the president is on X, then all of the bodyguards can move onto Y such that every vertex of Y is occupied by at least one bodyguard. The bodyguards can repeat this strategy indefinitely, and so $\max\{n, m\}$ bodyguards win on $K_{n,m}$. □

Theorem 5.3.3. If $n \geq 3$, then

$$B(C_n) = \begin{cases} 2 & \text{if } n \leq 5 \\ 3 & \text{if } n > 5. \end{cases}$$

Proof. Let $v_0, v_1, \ldots, v_{n-1}$ be the vertices of the cycle C_n where v_i is adjacent to $v_{i-1 \bmod n}$ and $v_{i+1 \bmod n}$. Once the president is surrounded by two bodyguards, regardless of whether the president moves clockwise or counterclockwise around the cycle, the bodyguards can follow his movements indefinitely. Thus, to determine the bodyguard number of a cycle it suffices to determine how many bodyguards are needed to surround the president.

Consider the cycle C_5. Place a bodyguard on each of the vertices v_0 and v_2. If the president starts on v_0, the bodyguards can shift to v_4 and v_1. If the president starts on v_1 then he is already surrounded. If the president starts on v_2 the bodyguards can move to v_1 and v_3. If the president starts on v_3 then the bodyguard on v_0 can move to v_4. Lastly, if the president is on v_4 then the bodyguard on v_2 can move to v_3. Therefore, regardless of where the president starts, two bodyguards are enough to surround him. Thus $B(C_5) \leq 2$. From Lemma 5.2.1, $B(C_5) \geq 2$. So $B(C_5) = 2$.

Note that for any two cycles C_k and C_{k-1}, C_{k-1} is a retract of C_k via the retraction map $f : V(C_k) \to V(C_{k-1})$ defined by

$$f(v_i) = \begin{cases} v_i & \text{if } 0 \leq i \leq k-2 \\ v_{k-2} & \text{if } i = k-1. \end{cases}$$

Therefore, by Lemma 5.2.1 and Theorem 5.2.5,

$$2 \leq B(C_3) \leq B(C_4) \leq B(C_5) = 2.$$

So $B(C_n) = 2$ for $n \leq 5$.

Now consider the cycle C_n for a fixed $n \geq 6$. First, we will show that two bodyguards are not enough to win the game. Then we will show that three bodyguards can win. Suppose two bodyguards b_1 and b_2 are playing against the president. Regardless of where the bodyguards begin the game, the president can start on a vertex that is distance at least three away from b_1. From here, the president can always maintain a distance of at least three by moving in the same direction, either clockwise or counterclockwise, as b_1. This strategy is illustrated in Figure 5.5. Therefore, b_1 can never be adjacent to the president and so the president never gets surrounded. Thus $B(C_n) > 2$.

Suppose the president is playing against three bodyguards, b_1, b_2 and b_3, on C_n. Note that $c(C_n) = 2$. For the bodyguards to win, they must move onto the two vertices adjacent to the president. The bodyguards can use a cop winning strategy on C_n to force one of the vertices adjacent to the president to be occupied by one of the bodyguards, say b_1. Afterwards, b_2 and b_3 can use a cop winning strategy again to move onto the other vertex adjacent to the president. So, the president is surrounded in finitely many moves. Therefore $B(C_n) = 3$. □

As a consequence of Theorem 5.3.1 and Theorem 5.3.3, we can completely characterize all graphs with bodyguard numbers one and two.

Theorem 5.3.4. Let G be a connected graph. Then $B(G) = 1$ if and only if $G \cong P_2$. Furthermore, $B(G) = 2$ if and only if either $G \cong P_n$ for some $n \geq 3$ or $G \cong C_m$ where $3 \leq m \leq 5$.

Proof. If $B(G) = 1$, then $\Delta(G) = 1$ by Lemma 5.2.1. The only connected graph with maximum degree one is P_2 and so $G \cong P_2$. From Theorem 5.3.1, $B(P_2) = 1$.

If $B(G) = 2$, then $\Delta(G) \leq 2$. Since $\Delta(G) = 1$ implies $G \cong K_2$ and since $B(K_2) = 1$, any graph with a bodyguard number of two has a maximum degree of two. The only connected graphs with maximum degree two are paths and cycles. From Theorem 5.3.1, all paths on at least three vertices have a bodyguard number of two and from Theorem 5.3.3, the cycles with a bodyguard number of two are C_3, C_4 or C_5. □

Next we will show that the bodyguard number of a tree on three or more vertices

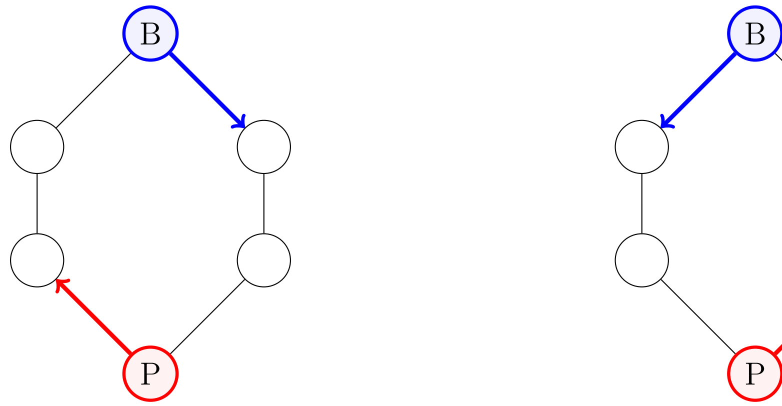


Figure 5.5: The president can indefinitely maintain a large distance from one bodyguard on a large cycle.

is exactly the number of leaves on the tree. We begin by showing that if there is one bodyguard for each leaf on a tree, then the bodyguards have a winning strategy.

Lemma 5.3.5. If T is a tree with ℓ leaves, then $B(T) \leq \ell$.

Proof. We proceed by describing a winning strategy for ℓ bodyguards. Let $v_1, \dots, v_\ell$ be the leaves of T and let v_P denote the vertex that is occupied by the president. Let $B_1, \dots, B_\ell$ be the bodyguards and for each i, the bodyguard B_i will be assigned to start on v_i. For each $1 \leq i \leq \ell$, let P_i denote the unique geodesic from v_i to v_P. Let u_i denote the unique vertex on P_i that is adjacent to v_P. Note that by the way the u_i's are defined they are not necessarily distinct and as the president moves each turn, P_i and u_i may change.

Since $c(T) = 1$, for each $1 \leq i \leq \ell$ the bodyguard B_i can move to u_i in finitely many moves. Thus $u_1 \dots, u_\ell$ can be occupied indefinitely by the bodyguards after finitely many turns. Next we claim that $N(v_P) = \{u_1, \dots, u_\ell\}$. By the way the u_i's are defined, it is clear that $\{u_1, \dots, u_\ell\} \subseteq N(v_P)$. Suppose there exists a vertex $x \in N(v_P)$ such that $x \neq u_i$ for any i. Note that $x \notin \{v_1, \dots, v_\ell\}$ since otherwise the edge xv_P would be a geodesic from a leaf to v_P and so $x \in \{u_1, \dots, u_\ell\}$ by definition. Let P_x^i be the path from x to v_i. If $v_P \notin V(P_x^i)$ for some $1 \leq i \leq \ell$, then P_x^i and P_i share a vertex y. Thus the path contained in P_x^i from x to y, the path contained in P_i from y to v_P, and the edge xv_P form a cycle which is a contradiction. If for all $1 \leq i \leq \ell$, P_x^i contains v_P then the only vertex adjacent to x is v_P. So x is a leaf which we have already shown leads to a contradiction.

Thus, after finitely many turns, every vertex adjacent to the president can be indefinitely occupied by the ℓ bodyguards and so $B(T) \leq \ell$. $\square$

To prove that the bodyguard number of a tree is the number of leaves, all that remains to show is that if the president plays against any fewer number of bodyguards then he has a winning strategy.

Theorem 5.3.6. If T is a tree on at least three vertices with ℓ leaves, then $B(T) = \ell$.

Proof. Let $v_1, \ldots, v_\ell$ be the leaves of T and let c be a center vertex of T. Assume that there are $\ell - 1$ bodyguards in play. The president will begin the game on c. If the bodyguards cannot surround the president at c, then $\ell - 1$ bodyguards would not be enough to win. Suppose instead that the bodyguards surround the president on c after finitely many moves. Since there are $\ell - 1$ bodyguards and ℓ leaves in T, there exists a component X in $T \backslash v_P$ that contains more vertices in the set $\{v_1, \ldots, v_\ell\}$ than bodyguards. On his next turn, the president will move onto X. The president will repeat this process without traversing the same edge during two consecutive moves until either the bodyguards fail to surround him on one of their moves, in which case the president walks back to c and begins this process again, or the president moves to a vertex u adjacent to the leaves $v_{n_1}, \ldots, v_{n_k}$. Since the president always moves onto a component of $T \backslash v_P$ that contains fewer bodyguards than vertices in the set $\{v_1, \ldots, v_\ell\}$, once the president moves onto u there will be at most $k - 1$ bodyguards that are either on u or on one of the v_{n_i}'s. Thus on the bodyguards' next move, it is not possible for the bodyguards to occupy all of the leaves $v_{n_1}, \ldots, v_{n_k}$ and so the president does not get surrounded. Then the president can move back to c and repeat this process indefinitely so that the bodyguards cannot win. So $B(G) \geq \ell$. Since $B(G) \leq \ell$ by Lemma 5.3.5, we have $B(G) = \ell$. $\square$

Theorem 5.3.6 gives an example of how the difference between the bodyguard number and the lower bound in Lemma 5.2.1 can be arbitrarily large. It was shown in [11] that the surrounding number on trees is at most two and so $B(G) - \sigma(G)$ can attain any nonnegative integer value. Furthermore, if we force $\Delta(G) = 3$ we can construct a tree with $\ell \geq 3$ leaves. Therefore $B(G) - \Delta(G)$ can attain any nonnegative integer value as well. So $B(G) - \max\{\Delta(G), \sigma(G)\}$ can be any nonnegative integer value. We also know from Theorem 2.1.3 that the cheating robot number of any tree is one. So $B(G) - c_{cr}(G)$ can also attain any nonnegative integer value since the tree P_2 has a bodyguard number of one.

5.4 Cartesian product

Here we will find the exact bodyguard numbers for all two dimensional Cartesian grids and give an upper bound on the bodyguard number for all Cartesian grids. To help find these bodyguard numbers, we first give an upper bound on the bodyguard number of the Cartesian product of any two graphs.

Theorem 5.4.1. If G and H are graphs, then

$$B(G\Box H) \leq B(G) + B(H) + c(G\Box H) - 1.$$

Proof. Let (u_p, v_p) denote the president's vertex on $G\Box H$. If the bodyguards are able to set up a bodyguard winning strategy on $G.\{v_p\}$ with $B(G)$ bodyguards, then the president will indefinitely be surrounded on $G.\{v_p\}$. Similarly, if the bodyguards can set up a bodyguard winning strategy on $\{u_p\}.H$ with $B(H)$ bodyguards, then the president will indefinitely be surrounded on $\{u_p\}.H$.

Fix a bodyguard winning strategy on $G.\{v_p\}$ and a bodyguard winning strategy on $\{u_p\}.H$. If, by using a set of $B(G)+B(H)$ bodyguards, the bodyguards can set up both of these winning strategies simultaneously, then they can indefinitely surround the president since every move the president makes either puts him in a new copy of G or a new copy of H and he cannot move to both a new copy of G and H in one move. The vertices that the $B(G)+B(H)$ bodyguards need to occupy to successfully implement these strategies change relative to the president. By using $c(G\Box H) - 1$ bodyguards plus 1 bodyguard from a set of $B(G)+B(H)$ bodyguards, the bodyguards can use a cop winning strategy to move a bodyguard b onto a desired vertex relative to the president's position after a finite number of moves. If the president moves in such a way that the desired vertex changes position, b can move onto the new position of the desired vertex. When a bodyguard has been set up in this way, we will say that the vertex is *captured* by the bodyguard. Once one vertex is captured, this leaves $c(G\Box H)-1$ bodyguards to join with another one of the $B(G)+B(H)$ bodyguards to capture another vertex needed for the winning strategy. In finitely many turns, the $B(G) + B(H)$ bodyguards can be set up to indefinitely surround the president. $\Box$

Next, we will study 2-dimensional Cartesian grids through multiple lemmas.

Lemma 5.4.2. If $n \geq 2$, then

$$B(P_2 \square P_n) = \begin{cases} 2 & \text{if } n = 2, \\ 3 & \text{if } n \geq 3. \end{cases}$$

Proof. If $n = 2$, then $P_2 \square P_2$ is isomorphic to C_4. From Theorem 5.3.3, $B(C_4) = 2$. Suppose $n \geq 3$. Since $\Delta(P_2 \square P_n) = 3$, $B(P_2 \square P_n) \geq 3$. To show that three bodyguards can win on $P_2 \square P_n$, assume that the bodyguards play on $P_2 \square P_{n+2}$ with vertices labelled (x, y) where $1 \leq x \leq 2$ and $0 \leq y \leq n+1$ while the president plays on the subgraph induced by the vertices $\{(u, v) \mid 1 \leq u \leq 2, 1 \leq v \leq n\}$. Note that two vertices (x, y) and (x', y') are adjacent if either $x = x'$ and $|y - y'| = 1$ or $|x - x'| = 1$ and $y = y'$. We will begin by giving a strategy for the bodyguards to win on $P_2 \square P_{n+2}$. Next, we will define a retract from $P_2 \square P_{n+2}$ to $P_2 \square P_n$. This retract will translate the winning bodyguard strategy on $P_2 \square P_{n+2}$ with the president restricted to the subgraph $P_2 \square P_n$, to a winning bodyguard strategy where both the bodyguards and the president play on $P_2 \square P_n$.

Three bodyguards can win the game by starting on the vertices $(1, 0)$, $(2, 0)$ and $(1, 1)$. All three bodyguards will begin the game by increasing their second coordinates every round. Let (x_p, y_p) denote the vertex occupied by the president at the time when the bodyguards have moved to the vertices $(1, y_p - 1)$, $(1, y_p)$ and $(2, y_p - 1)$. From here, the bodyguards' strategy changes depending on how the president moves from this position. If the president moves by increasing y_p at any point, the bodyguards can respond by moving towards the president without changing their strategy. Since the president can only increase y_p finitely many times, the president will eventually only have three options: decrease his second coordinate, change his first coordinate, or stay at the vertex he is currently on. There are two cases for the president's position, either he is at $(1, y_p)$ or he is at $(2, y_p)$.

Case (i): The president is at $(1, y_p)$**.** If the president stays at $(1, y_p)$, the bodyguards can move to $(1, y_p - 1)$, $(1, y_p + 1)$ and $(2, y_p)$ to surround the president. If the president moves to $(1, y_p - 1)$, the bodyguards can move to $(1, y_p - 2)$, $(1, y_p)$ and $(2, y_p - 1)$ to surround the president. If the president moves to $(2, y_p)$, the bodyguards can move to $(2, y_p - 1)$, $(1, y_p)$ and $(2, y_p)$. From here the president can either stay at $(2, y_p)$, move to $(1, y_p)$ or move to $(2, y_p - 1)$. If the president stays at $(2, y_p)$, the bodyguards can move to $(2, y_p - 1)$, $(1, y_p)$ and $(2, y_p + 1)$ to surround the president. If the president moves to $(1, y_p)$, the bodyguards can move to $(1, y_p - 1)$, $(2, y_p)$ and $(1, y_p + 1)$ to surround the president. If the president moves to $(2, y_p - 1)$, the bodyguards can move to $(2, y_p - 2)$, $(1, y_p - 1)$ and $(2, y_p)$ to surround the president.

Case (ii): The president is at $(2, y_p)$**.** If the president stays on $(2, y_p)$ then the bodyguards can move to $(1, y_p - 1)$, $(2, y_p)$ and $(2, y_p - 1)$. By the symmetry of the graph, the bodyguards can surround the president from this position by using a strategy similar to the one from Case (i). If the president moves to $(1, y_p)$, then the bodyguards can follow the strategy from Case (i). If the president moves to $(2, y_p - 1)$, the bodyguards can move to $(2, y_p - 2)$, $(1, y_p - 1)$ and $(2, y_p)$ to surround the president.

Therefore, in the case where the bodyguards play on $P_2 \Box P_{n+2}$ while the president is restricted to the subgraph $P_2 \Box P_n$, regardless of how the president moves three bodyguards are enough to surround him. Once the president is surrounded, the bodyguards can indefinitely surround the president by either changing the value of their second coordinates in the same way as the president or by changing the value of the first coordinate whenever the president does the same.

To obtain a winning strategy for three bodyguards on $P_2 \Box P_n$, consider the following retract $f : V(P_2 \Box P_{n+2}) \to V(P_2 \Box P_n)$.

$$f((u, v)) = \begin{cases} (u, v) & \text{if } 1 \leq v \leq n, \\ (u, v + 2) & \text{if } v = 0, \\ (u, v - 2) & \text{if } v = n + 1. \end{cases}$$

Fix a bodyguard winning strategy on $P_2 \Box P_{n+2}$. If a bodyguard was to move to the vertex (u, v) when using the winning strategy on $P_2 \Box P_{n+2}$, to win on $P_2 \Box P_n$ the bodyguard will instead move to $f((u, v))$. Since retracts preserve adjacency, by mapping the bodyguards movements using the retract f, the bodyguards will surround the president in finitely many moves and then indefinitely surround him. This creates a winning strategy where both the bodyguards and the president are playing on $P_2 \Box P_n$. $\square$

We note that in the proof of Lemma 5.4.2 we showed that, for a given $n \geq 2$, if the bodyguards are able to win when playing on $P_2 \Box P_{n+2}$ with the president restricted to a retract $P_2 \Box P_n$ then the bodyguards can win when everyone is restricted to the retract

$P_2 \Box P_n$. The reason we described a winning strategy for the bodyguards playing on a larger graph instead of describing a winning strategy directly on $P_2 \Box P_n$ was to decrease the number of cases needed to describe the winning bodyguard strategy. For example, suppose both the bodyguards and the president are playing on $P_2 \Box P_n$. If the bodyguards have already surrounded the president and he moves from $(1, 2)$ to $(1, 1)$, then the bodyguard that was on $(1, 1)$ when the president was on $(1, 2)$ would now either have to move to $(1, 2)$ or stay at $(1, 1)$. In other words, the bodyguard would no longer be able to move in the same direction as the president. However, by extending the graph that the bodyguards play on to $P_2 \Box P_{n+2}$, the bodyguard that was on $(1, 1)$ when the president was on $(1, 2)$ can respond to the president's move to $(1, 1)$ by moving in the same direction as the president. Extending the graph in this way eliminates the cases for when the president moves to either $(1, 1)$, $(1, n)$, $(2, 1)$, or $(2, n)$. We will use a similar technique of having the president restricted to a retract of the graph the bodyguards are playing on in the proof of Lemma 5.4.3.

Lemma 5.4.3. If $n \geq 2$, then

$$B(P_3 \Box P_n) = \begin{cases} 3 & \text{if } n = 2, \\ 4 & \text{if } n \geq 3. \end{cases}$$

Proof. If $n = 2$, $B(P_3 \Box P_n) = 3$ by Lemma 5.4.2. Let $n \geq 3$. By Lemma 5.2.1, we have $B(P_3 \Box P_n) \geq \Delta(P_3 \Box P_n) \geq 4$. To show that four bodyguards can win on $P_3 \Box P_n$, we use a similar argument as in the proof of Lemma 5.4.2. Assume the bodyguards are playing on $P_3 \Box P_{n+3}$ with vertices labelled (u, v) where $1 \leq u \leq 3$ and $0 \leq v \leq n + 2$ while the president plays on the subgraph induced by the vertices $\{(u, v) \mid 1 \leq u \leq 3, 2 \leq v \leq n + 1\}$. The bodyguards start the game on the vertices $(1, 1)$, $(2, 0)$, $(2, 2)$ and $(3, 1)$. The bodyguards' strategy begins with increasing their second coordinates every round. Let (x_p, y_p) denote the vertex occupied by the president at the time when the bodyguards have moved to the vertices $(1, y_p - 1)$, $(2, y_p - 2)$, $(2, y_p)$ and $(3, y_p - 1)$. The bodyguards' strategy changes from this position. To show that the four bodyguards win, we will give two positions for the bodyguards and the president and show that once in one of the two positions, the bodyguards can indefinitely surround the president. Following this, we will show that in finitely many turns, regardless of how the president moves, the bodyguards can force the game into one of these two positions.

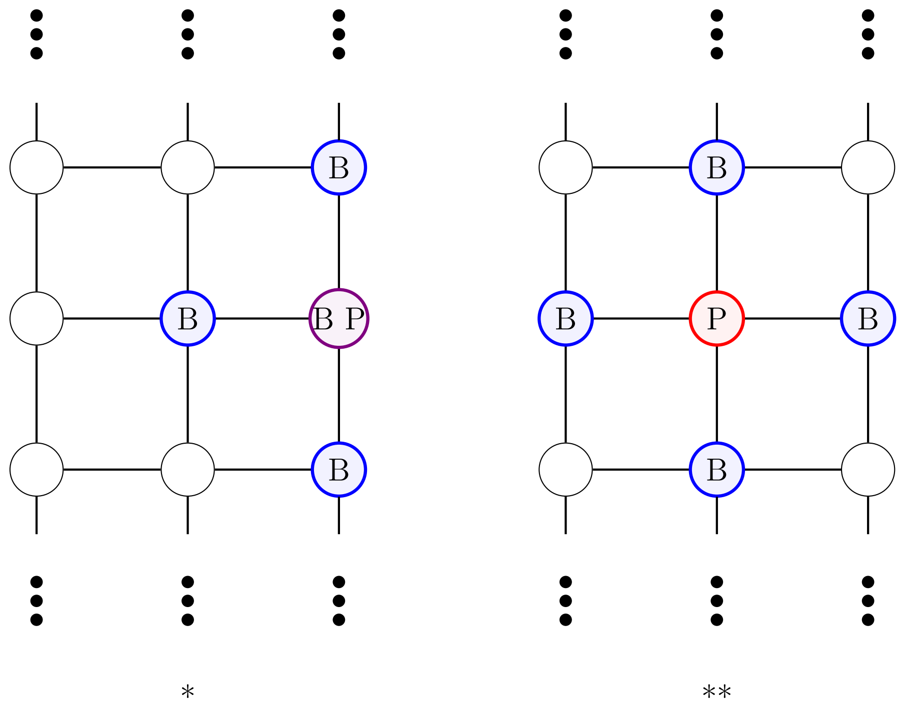


Figure 5.6: Two winning positions for four bodyguards on $P_3 \square P_n$.

First, we describe two winning positions for the bodyguards. Suppose the president is on $(1, y)$ for some $2 \leq y \leq n+1$. We claim that if there are bodyguards occupying the vertices $(1, y-1)$, $(1, y)$, $(2, y)$ and $(1, y+1)$ as labelled * in Figure 5.6, then the president loses. In this position the president can either move to $(1, y-1)$, $(1, y+1)$ or $(2, y)$. If the president moves to $(1, y-1)$ the bodyguards can move to $(1, y-2)$, $(1, y-1)$, $(2, y-1)$ and $(1, y)$ to surround him again. If the president moves to $(1, y+1)$ the bodyguards can move to $(1, y)$, $(1, y+1)$, $(2, y+1)$ and $(1, y+2)$. If the president moves to $(2, y)$ the bodyguards can move to $(2, y-1)$, $(1, y)$, $(3, y)$ and $(2, y+1)$ which is ** in Figure 5.6. From this point, regardless of how the president moves the bodyguards can continue to surround him indefinitely. A similar argument can be made if the president is on $(3, y)$ and the bodyguards occupy $(3, y-1)$, $(3, y)$, $(2, y)$ and $(3, y+1)$. Therefore, to show that four bodyguards can win it suffices to show that they can force the president into either position * or ** in Figure 5.6.

Next we will show that the bodyguards can, in finitely many turns, force the game into either position * or **. Suppose $x_p = 1$. If the president moves to $(1, y_p - 1)$ or $(2, y_p)$ then the bodyguards can force either position * or ** in Figure 5.6. Suppose the president stays at $(1, y_p)$ for his next turn. The bodyguards can position themselves to have one bodyguard at $(1, y_p - 1)$ and $(2, y_p)$ and have two bodyguards at $(2, y_p - 1)$. If the president moves to $(1, y_p - 1)$, the bodyguards can move into position *. If the president moves to $(2, y_p)$ then the bodyguards can move to $(1, y_p)$, $(2, y_p - 1)$, $(2, y_p)$ and $(3, y_p)$. Regardless of how the president moves from this position the bodyguards can either move to position * or **. Suppose the president again stays at $(1, y_p)$. The bodyguards can then move to $(1, y_p - 1)$, $(1, y_p)$, $(2, y_p - 1)$ and $(2, y_p)$. If the president either does not move, moves to $(1, y_p - 1)$ or moves to $(2, y_p)$ the bodyguards can respond by either moving to positions * or ** or to a position from a previous case. If at any point the president had chosen to move to $(x, y_p + 1)$ then the bodyguards could have responded by increasing their second coordinates until they obtain the same position as before. Since the president can only increase his second coordinate finitely many times, he will eventually be forced to move in any other direction, allowing the bodyguards to continue the above strategy.

Suppose instead $x_p = 2$ or $x_p = 3$. If $x_p = 3$ then a similar argument to above holds by the symmetry of the graph. If $x_p = 2$ then the bodyguards can move into position **.

Finally, to obtain a winning strategy for the bodyguards where everyone is playing on $P_3 \square P_n$, we make use of the following retract $f : V(P_3 \square P_{n+3}) \to V(P_3 \square P_n)$.

$$f((u, v)) = \begin{cases} (u, v) & \text{if } 2 \leq v \leq n + 1, \\ (u, v + 2) & \text{if } v = 1 \text{ or } v = 0, \\ (u, v - 2) & \text{if } v = n + 2. \end{cases}$$

By mapping the bodyguards' moves when using their strategy on $P_3 \square P_{n+3}$ with the retract f, we have a winning strategy for the bodyguards on $P_3 \square P_n$ when both the bodyguards and the president are playing on $P_3 \square P_n$. $\square$

Lemma 5.4.4. If $n = 4$, then $B(P_n \square P_n) = 5$.

Proof. By Theorem 5.4.1, $B(P_4 \square P_4) \leq 5$. Suppose the president is playing against four bodyguards. We label the vertices (x, y) where $1 \leq x, y, \leq 4$ and (x_1, y_1) is adjacent to (x_2, y_2) if either $x_1 = x_2$ and $|y_1 - y_2| = 1$ or $|x_1 - x_2| = 1$ and $y_1 = y_2$. The president will stay on the vertices $(2, 2)$, $(2, 3)$, $(3, 2)$, $(3, 3)$. If the bodyguards are to win this game, they need to be able to surround the president in finitely many turns. Consider the last move the president makes from (u_p, v_p) to (u'_p, v'_p) before he is surrounded. The four bodyguards need to occupy the four vertices adjacent to the president after he makes his final move. Therefore, if any bodyguard is distance three away from (u_p, v_p), then the president can avoid being surrounded by staying on (u_p, v_p) for his final turn. This is a contradiction and so we assume that every bodyguard is at most distance two away from the president before his final move.

Let C be the set of vertices $\{(2,2),(2,3),(3,2),(3,3)\}$. There are multiple cases for the position of the bodyguards. Since the president has four neighbours, there are either zero, one, two, three or four bodyguards adjacent to the president. Furthermore, each of these bodyguards are either in C or not in C. Figure 5.7 illustrates an example for each of the cases we consider. First, we consider the case where a vertex adjacent to the president contains more than one bodyguard. Suppose there are two bodyguards adjacent to the president that occupy the same vertex. Without loss of generality, assume the president is on the vertex $(2,2)$. If the president does not move, then it is not possible for either one of the two bodyguards to move to a different vertex adjacent to the president in one move. Therefore, the bodyguards cannot surround the president in one move. Using the same argument, if a vertex adjacent to the president contains two or more bodyguards then the bodyguards cannot surround the president in one move.

Now suppose instead that every vertex adjacent to the president is occupied by at most one bodyguard. We split the rest of the proof into cases based on positions of the bodyguards while under the assumption that no vertex adjacent to the president contains more than one bodyguard. Without loss of generality, assume that in all of these cases the president is on the vertex $(2,2)$. First, we consider the case where there is a bodyguard that is neither adjacent to the president nor in C. If there is a bodyguard outside of C that is not adjacent to the president then, since the bodyguards are at most distance two away from the president, that bodyguard is either on $(1,1)$, $(1,3)$, $(2,4)$, $(3,1)$ or $(4,2)$. If there is a bodyguard on $(1,1)$, $(1,3)$ or $(2,4)$, the president can move to $(3,2)$ and now that bodyguard cannot move to a vertex adjacent to the president on their next move. If that bodyguard is on $(3,1)$ or $(4,2)$, then the president can move to $(2,3)$ and again that bodyguard cannot move to a vertex adjacent to the president in one move.

Suppose instead that every bodyguard is either adjacent to the president or in C.

Then, every bodyguard that is not adjacent to the president is forced to be in C. Since the president is one move away from being surrounded, there can be anywhere between zero and three bodyguards adjacent to the president. So, we have the following four cases:

(i): there are exactly three bodyguards adjacent to the president that may or may not be in C and the remaining bodyguard is in C and not adjacent to the president,

(ii): there are exactly two bodyguards adjacent to the president that may or may not be in C and the remaining two bodyguards are in C and not adjacent to the president,

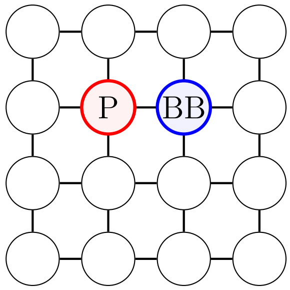
(iii): there is exactly one bodyguard adjacent to the president that may or may not be in C and the remaining three bodyguards are in C and not adjacent to the president, and

(iv): no bodyguards are adjacent to the president, but all four are in C.

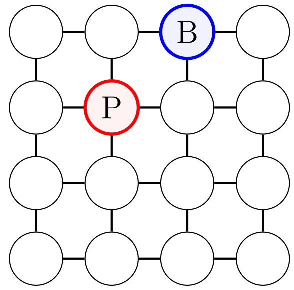
Case (i): Let b be the bodyguard not adjacent to the president. The bodyguard b is either on $(2,2)$ or $(3,3)$. Suppose either $(2,3)$ or $(3,2)$ is unoccupied. If the president moves to whichever of $(2,3)$ or $(3,2)$ is unoccupied, then the bodyguards cannot cover $(2,4)$ and $(4,2)$ respectively. Suppose, on the other hand, that there is a bodyguard on $(2,3)$ and $(3,2)$. Then, without loss of generality, the third bodyguard adjacent to the president is on $(1,2)$. Since there is no bodyguard on $(2,1)$, the president can move to $(3,2)$ and the bodyguards cannot cover $(3,1)$ and $(4,2)$ in one move.

Case (ii): There are three subcases: either there is no bodyguard on $(2,2)$, there is one bodyguard on $(2,2)$, or there are two bodyguards on $(2,2)$. Suppose there are no bodyguards on $(2,2)$. Then the two bodyguards not adjacent to the president, but are still in C are on $(3,3)$. Furthermore, since the president can be surrounded in one move if he does not move off of $(2,2)$, there must be bodyguards on $(1,2)$ and $(2,1)$. If the president moves to $(3,2)$ then no bodyguard can cover $(4,2)$.

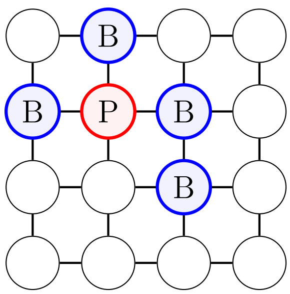
Suppose instead that one bodyguard is on $(2,2)$, forcing the other bodyguard onto $(3,3)$. Note that the two bodyguards adjacent to the president may or may not be on C. So one of the two bodyguards adjacent to the president is forced to be on



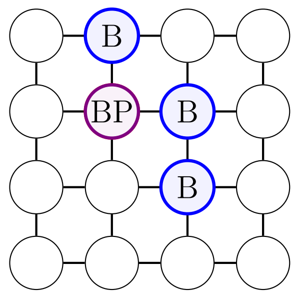
More than one bodyguard on a vertex adjacent to the president.



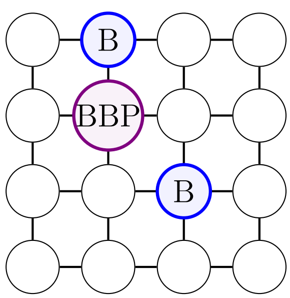
One bodyguard is neither in C nor adjacent to the president.



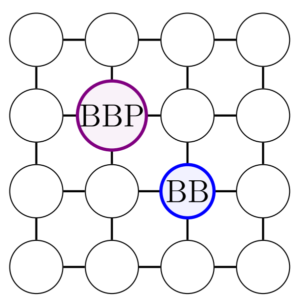
Case (i).



Case (ii).



Case (iii).



Case (iv).

Figure 5.7: Examples of each of the cases considered in the proof of Lemma 5.4.4.

either $(1,2)$ or $(2,1)$. If there is a bodyguard on $(1,2)$ and $(2,1)$ then the president can move to $(3,2)$ and avoid being surrounded since the vertex $(4,2)$ is not covered by the bodyguards. Suppose there is a bodyguard on $(1,2)$ and not $(2,1)$. Then the fourth bodyguard is either on $(2,3)$ or $(3,2)$. Whichever one is not occupied by a bodyguard, if the president moves there then either $(4,2)$ or $(2,4)$ respectively will not be covered by the bodyguards. If instead there is a bodyguard on $(2,1)$ and not $(1,2)$ then by the symmetry of the graph, the same strategy of moving to an empty vertex will prevent the president from being surrounded in one move. Therefore, there cannot be one bodyguard on $(2,2)$.

Finally, suppose the two bodyguards not adjacent to the president are on $(2,2)$. If the president moves to $(3,2)$, then neither of the bodyguards on $(2,2)$ can get to $(3,1)$, $(3,3)$ nor $(4,2)$ in one move. So it is not possible for all four bodyguards to cover those vertices in one move. Thus the president cannot be surrounded in one move.

Case (iii): If there are no bodyguards on $(2,2)$, then there are three bodyguards on $(3,3)$. If the president does not move, it is not possible for the bodyguards to surround the president from this position in one move. If instead there are at least two of the bodyguards on $(2,2)$, then by the work done in Case (ii) the president can move to $(3,2)$ and avoid being surrounded in one move. Suppose one bodyguard is on $(2,2)$ and two bodyguards are on $(3,3)$. Then the last bodyguard must be on either $(1,2)$ or $(2,1)$. In the analysis of Case (ii), we showed that the president can avoid being surrounded in one move when two bodyguards are on $(2,2)$ and one bodyguard is on either $(1,2)$ or $(2,1)$. By using the same strategy for the president as described in the analysis of Case (ii), the president can move and avoid being surrounded in one move.

Case (iv): If all four bodyguards are in C, at least two of them must be on $(2,2)$. Otherwise, two of the vertices in the president's open neighbourhood would be distance at least two away from three of the bodyguards, so the bodyguards would not be able to surround the president on their next move. Thus at least two of the bodyguards are on $(2,2)$. However, by the argument in Case (ii), the president can avoid being surrounded.

Therefore, against four bodyguards, the president can always avoid being surrounded. Thus $B(P_n \Box P_n) > 4$ and so $B(P_4 \Box P_4) = 5$. $\square$

To determine the bodyguard numbers for every 2-dimensional Cartesian grid, we make use of Maamoun and Meyniel's result in [32] on the cop number of the Cartesian product of trees.

Theorem 5.4.5. [32] If $T_1, \ldots, T_k$ are trees such that each T_i has more than one vertex, then

$$c(\square_{i=1}^{k} T_i) = \left\lceil \frac{k+1}{2} \right\rceil.$$

However, we only need the case when each tree in the Cartesian product is a path. So we use the following corollary.

Corollary 5.4.6. If $n_1, \ldots, n_k \geq 2$, then

$$c(\square_{i=1}^{k} P_{n_i}) = \begin{cases} \frac{k+1}{2} & \text{if } k \text{ is odd} \\ \frac{k}{2}+1 & \text{if } k \text{ is even.} \end{cases}$$

Theorem 5.4.7. If $n \leq m$, then

$$B(P_n \square P_m) = \begin{cases} 2 & \text{if } n, m = 2, \\ 3 & \text{if } n = 2 \text{ and } m \geq 3, \\ 4 & \text{if } n = 3 \text{ and } m \geq 3, \\ 5 & \text{if } n, m \geq 4. \end{cases}$$

Proof. From Lemma 5.4.2 we have $B(P_2 \square P_2) = 2$ and $B(P_2 \square P_m) = 3$ for any $m \geq 3$. From Lemma 5.4.3 we have $B(P_3 \square P_m) = 4$ for any $m \geq 3$. From Lemma 5.4.4 we know that $B(P_4 \square P_4) = 5$. Since $P_4 \square P_4$ is a retract of $P_n \square P_m$ where $n, m \geq 4$, by Theorem 5.2.5 $B(P_n \square P_m) \geq 5$. From Corollary 5.4.6, $c(P_n \square P_m) = 2$ for all $n, m \in \mathbb{Z}^+$. Therefore by Theorem 5.4.1, $B(P_n \square P_m) \leq 5$ and so $B(P_n \square P_m) = 5$ when $n, m \geq 4$. $\square$

Next, we give bounds on k-dimensional Cartesian grids.

Theorem 5.4.8. If $n_1, \ldots, n_k \geq 3$, then

$$2k \leq B(\square_{i=1}^{k} P_{n_i}) \leq \begin{cases} \frac{5k-1}{2} & \text{if } k \text{ is odd} \\ \frac{5k}{2} & \text{if } k \text{ is even.} \end{cases}$$

Proof. The lower bound is obtained by considering $\Delta(\square_{i=1}^k P_{n_i})$ and Lemma 5.2.1.

Suppose the bodyguards are playing on the graph $\square_{i=1}^k P_{n_i+2}$ with vertices labeled $(x_1, \ldots, x_k)$ where $0 \le x_i \le n_i+1$ while the president plays on the subgraph induced by the vertices $(y_1, \ldots, y_k)$ where $1 \le y_i \le n_i$. If the president is on the vertex $(v_1, \ldots, v_k)$ and the bodyguards can occupy the vertices $(v_1, \ldots, v_{i-1}, v_i + a_i, v_{i+1}, \ldots, v_k)$ where $1 \le i \le k$ and for each i, $a_i \in \{1, -1\}$, then the president can never escape being surrounded regardless of how he moves. Therefore, it suffices to show that $\left\lfloor \frac{5k}{2} \right\rfloor$ bodyguards can move from a position where the president is not surrounded to a position where the president is surrounded.

Suppose we are playing with $2k + c(\square_{i=1}^k P_{n_i}) - 1$ bodyguards. Note that we have $\Delta(\square_{i=1}^k P_{n_i}) = 2k$. Using a cop winning strategy, $c(\square_{i=1}^k P_{n_i})$ bodyguards can force one of the $2k$ neighbours of the president, say $b_i^+ = (v_1, \ldots, v_{i-1}, v_i + 1, v_{i+1}, \ldots, v_k)$, to become occupied by a bodyguard. Once b_i^+ is occupied by a bodyguard, the bodyguard that moved onto b_i^+ can always move to b_i^+ after every president turn. The $c(\square_{i=1}^k P_{n_i}) - 1$ bodyguards that did not move onto b_i^+ can then combine with one of the remaining $2k-2$ bodyguards to force a bodyguard onto another one of the president's neighbours. They can continue this until eventually all of the president's neighbours are occupied by bodyguards.

From Corollary 5.4.6, we know that $c(\square_{i=1}^k P_{n_i}) = \frac{k+1}{2}$ if k is odd and $c(\square_{i=1}^k P_{n_i}) = \frac{k}{2} + 1$ if k is even. Therefore, to utilize the above strategy we need

$$2k + \frac{k+1}{2} - 1 = \frac{5k-1}{2}$$

bodyguards if k is odd and

$$2k + \frac{k}{2} + 1 - 1 = \frac{5k}{2}$$

bodyguards if k is even. $\square$

If $n_i = 2$ for all $1 \le i \le k$, then the graph $\square_{i=1}^k P_{n_i}$ is called the *k-dimensional hypercube* and is denoted Q_k. Note that $\Delta(Q_k) = k$ and so by Lemma 5.2.1, $B(Q_k) \ge k$. Theorem 5.4.8 gives an upper bound on $B(Q_k)$. In the following theorem, we give tighter bounds on $B(Q_k)$ than the bounds given by Lemma 5.2.1 and Theorem 5.4.8.

Theorem 5.4.9. If $k \geq 3$, then

$$k+1 \leq B(Q_k) \leq \begin{cases} \frac{3k-1}{2} & \text{if } k \text{ is odd} \\ \frac{3k}{2} & \text{if } k \text{ is even.} \end{cases}$$

Proof. We will label the vertices in Q_k with vectors $(a_1, \ldots, a_k)$ where $a_i \in \{0, 1\}$ for each $1 \leq i \leq k$. By the structure of Q_k, two vertices are adjacent in Q_k if they differ in exactly one of the k coordinates in their labels.

First, we will show that $B(Q_k) \geq k+1$ by showing that k bodyguards cannot surround the president. At the start of the game the president can place himself on a vertex $(x_1, \ldots, x_k)$ such that one of the bodyguards is on $(x'_1, \ldots, x'_k)$ where $x_i \not\equiv x'_i \mod 2$. Since $k \geq 3$, the president is beginning the game distance at least three away from one of the bodyguards. Thus, the bodyguards cannot surround the president on their first move.

Now, suppose for a contradiction that k bodyguards can surround the president on some move after their first move. Consider the president's last move before being surrounded. Let $(v_1, \ldots, v_k)$ be the vertex occupied by the president. Since the president is one bodyguard turn away from being surrounded, there is a bodyguard b that is distance two away from the president. Let $(v_1, \ldots, v_{i-1}, v'_i, v_{i+1}, \ldots, v_{j-1}, v'_j, v_{j+1}, \ldots, v_k)$ where $v_i \not\equiv v'_i \mod 2$ and $v_j \not\equiv v'_j \mod 2$ be the vertex occupied by b. To avoid being surrounded, the president can move to a vertex whose label differs from $(v_1, \ldots, v_{i-1}, v'_i, v_{i+1}, \ldots, v_{j-1}, v'_j, v_{j+1}, \ldots, v_k)$ in one coordinate that is neither the ith coordinate nor the jth coordinate. By making this move, the president puts himself at distance three away from b, and so b cannot move adjacent to the president on the next bodyguard turn. Since the degree of every vertex in Q_k is k, the k bodyguards cannot surround the president without having every bodyguard adjacent to the president. Therefore, we have a contradiction and so the president can indefinitely avoid being surrounded by k bodyguards.

Next, we will show that $B(Q_k) \leq \left\lfloor \frac{3k}{2} \right\rfloor$ by using the technique from the proof of Theorem 5.4.8. Let $(v_1, \ldots, v_k)$ be the vertex occupied by the president. Suppose we are playing with $k + c(Q_k) - 1$ bodyguards. Using a cop winning strategy, $c(Q_k)$ bodyguards can force one of the k neighbours of the president, say $u_i = (v_1, \ldots, v_{i-1}, v'_i, v_{i+1}, \ldots, v_k)$ where $v_i \not\equiv v'_i \mod 2$, to become occupied by a bodyguard. The bodyguard that moves onto the vertex u_i can always move to u_i following every president move for the entirety of the game. The $c(Q_k)-1$ bodyguards that did not move onto u_i can then combine with one of the remaining $k-2$ bodyguards to force a bodyguard onto another one of the president's neighbours. The bodyguards can continue this strategy until eventually all of the president's neighbours are occupied by bodyguards. By Corollary 5.4.6, $c(Q_k) = \lceil \frac{k+1}{2} \rceil$. Therefore if k is odd, then $k + \frac{k+1}{2} - 1 = \frac{3k-1}{2}$ bodyguards can surround the president and if k is even, then $k + \frac{k}{2} + 1 - 1 = \frac{3k}{2}$ bodyguards can surround the president. It remains to show that once the president is surrounded, the bodyguards can indefinitely surround him.

Suppose the president is surrounded on the vertex $v = (v_1, \ldots, v_k)$. We claim that if the president moves from v to $v' = (v_1, \ldots, v_{i-1}, v'_i, v_{i+1}, \ldots, v_k)$ where $v_i \not\equiv v'_i \mod 2$, the bodyguards can respond by surrounding the president in one move. If a bodyguard adjacent to the president is on the vertex $u = (u_1, \ldots, u_k)$, then that bodyguard can move to the vertex $u' = (u_1, \ldots, u_{i-1}, u'_i, u_{i+1}, \ldots, u_k)$ where $u_i \not\equiv u'_i \mod 2$. That is, each of the k bodyguards will change their ith coordinates. Since u is adjacent to v, u' is adjacent to v'. Furthermore, since each of the k bodyguards occupied different vertices when they were surrounding the president, if each bodyguard changes their ith coordinate they will still occupy k distinct vertices. Therefore, the bodyguards can surround the president on their turn immediately following the president moving onto v'. Thus, the bodyguards can surround the president indefinitely. $\square$

Bounds on $B(\square_{i=1}^k P_{n_i})$ when $n_i = 2$ for some i can be obtained by using Corollary 5.4.6, Theorem 5.4.8, and Theorem 5.4.9.

Theorem 5.4.10. Let $k, t \in \mathbb{Z}^+$ where $t \leq k$. If $n_1, \ldots, n_t = 2$ and $n_{t+1}, \ldots, n_k > 2$, then

$$2k - t \leq B(\square_{i=1}^k P_{n_i}) \leq \begin{cases} \frac{6k-2t-1}{2} & \text{if } k \text{ is odd} \\ 3k - t + 1 & \text{if } k \text{ is even.} \end{cases}$$

Proof. The lower bound is obtained by considering $\Delta(\square_{i=1}^k P_{n_i})$ and Lemma 5.2.1.

By Theorem 5.4.8, we have

$$\begin{aligned} B(\square_{i=1}^k P_{n_i}) &= B((\square_{i=1}^t P_2)\square(\square_{i=t+1}^k P_{n_i})) \\ &\leq B(Q_t) + B(\square_{i=t+1}^k P_{n_i}) + c(\square_{i=1}^k P_{n_i}) - 1. \end{aligned}$$

By Corollary 5.4.6, Theorem 5.4.8, and Theorem 5.4.9, if k is odd we obtain

$$B(\square_{i=1}^{k} P_{n_i}) \leq \frac{3t-1}{2} + \frac{5(k-t)-1}{2} + \frac{k+1}{2} = \frac{6k-2t-1}{2}$$

and when k is even we obtain

$$B(\square_{i=1}^{k} P_{n_i}) \leq \frac{3t}{2} + \frac{5(k-t)}{2} + \frac{k+2}{2} = 3k - t + 1.$$

$\square$

5.5 Strong grids

In Chapter 6, we will analyze the cheating robot number of strong grids. In this section, we determine the bodyguard number of any k-dimensional strong grid which will be used to bound the cheating robot number of strong grids. First, we show that any k-dimensional strong grid is copwin using a theorem by Neufeld and Nowakowski [34].

Theorem 5.5.1. ([34]) If G and H are graphs, then

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

Corollary 5.5.2. If $n_i \geq 2$ where $1 \leq i \leq k$, then $c(\boxtimes_{i=1}^{k} P_{n_i}) = 1$.

Proof. We know $c(P_n) = 1$ for any $n \in \mathbb{Z}^+$. So, by Theorem 5.5.1,

$$c(P_{n_1} \boxtimes P_{n_2}) = c(P_{n_1}) + c(P_{n_2}) - 1 = 1.$$

By induction, we get $c(\boxtimes_{i=1}^{k} P_{n_i}) = 1$. $\square$

Theorem 5.5.3. If $k \in \mathbb{Z}^+$ and $n_i \geq 3$ for $1 \leq i \leq k$ then

$$B(\boxtimes_{i=1}^{k} P_{n_i}) = 3^k - 1.$$

Proof. Note that $B(\boxtimes_{i=1}^{k} P_{n_i}) \geq \Delta(\boxtimes_{i=1}^{k} P_{n_i}) = 3^k - 1$. We will label the vertices of $B(\boxtimes_{i=1}^{k} P_{n_i})$ by vectors of length k, $(v_1, \ldots, v_k)$ where $1 \leq v_i \leq n_i$ for each $1 \leq i \leq k$.

To show that $3^k - 1$ bodyguards can win on $\boxtimes_{i=1}^k P_{n_i}$, we use the same technique as in the proof of Theorem 5.4.8. Let $G = \boxtimes_{i=1}^k P'_{n_i}$ where P'_{n_i} is obtained by adding a leaf to each end of the path P_{n_i}. The vertices of G will be labeled the same way as the vertices of $\boxtimes_{i=1}^k P_{n_i}$ with additional vertices that have labels containing 0 and $n_i + 1$. The bodyguards will play on G while the president plays on $\boxtimes_{i=1}^k P_{n_i}$.

If the president is on the vertex $(v_1, \ldots, v_k)$ and the bodyguards can occupy vertices of the form $(v_1 + a_1, \ldots, v_k + a_k)$ where $1 \leq i \leq k$, for each i, $a_i \in \{1, 0, -1\}$, and a_i is nonzero for some $1 \leq i \leq k$, then the president can never escape being surrounded regardless of how he moves. By Corollary 5.5.2, $c(\boxtimes_{i=1}^k P_{n_i}) = 1$ and so each of the $3^k - 1$ bodyguards can use a cop winning strategy to move onto each of the president's neighbours. Once each of the president's neighbours is occupied by a bodyguard, the bodyguards can indefinitely surround the president. $\square$

Chapter 6

Graph Products

In this chapter we discuss how the cheating robot number behaves with respect to different graph products. Huggan and Nowakowski [23] showed that for any two graphs G and H, $c_{cr}(G\Box H) \leq c_{cr}(G)+c_{cr}(H)$. Comparatively, it was shown by Tošić [40] that $c(G\Box H) \leq c(G) + c(H)$. Since the cheating robot number has a similar upper bound to the cop number for Cartesian products, it is natural to ask whether the same can be said regarding other graph products. Consider the strong product. It was shown by Neufeld and Nowakowski [34] that $c(G \boxtimes H) \leq c(G) + c(H) - 1$. Is it true that $c_{cr}(G\boxtimes H) \leq c_{cr}(G)+c_{cr}(H)-1$? Burgess et al. [11] showed that $\sigma(G)+\sigma(H)-1$ is not an upper bound for $\sigma(G\boxtimes H)$ by showing that the graph $K_{1,n}\boxtimes K_{1,n}$ contains an n-core. Since $K_{1,n}\boxtimes K_{1,n}$ contains an n-core, by Theorem 2.1.1 $c_{cr}(K_{1,n}\boxtimes K_{1,n}) \geq n$. However, by Theorem 2.1.3 $c_{cr}(K_{1,n}) = 1$ and so $c_{cr}(K_{1,n} \boxtimes K_{1,n}) > c_{cr}(K_{1,n}) + c_{cr}(K_{1,n}) - 1$. Therefore $c_{cr}(G) + c_{cr}(H) - 1$ does not bound $c_{cr}(G \boxtimes H)$ from above in general. To obtain an upper bound on $c_{cr}(G\boxtimes H)$, we will make use of the push number introduced in Chapter 2 and the bodyguard number introduced in Chapter 5. We also give a new upper bound on the cheating robot numbers of the strong grid and the lexicographic product of two graphs. We discuss how we can use similar proof techniques as with the cheating robot number of strong products to obtain a new upper bound on the surrounding number for strong products of graphs and for k-dimensional strong grids.

6.1 General strong product

We begin with an upper bound on the cheating robot number for the strong product of any two graphs.

Theorem 6.1.1. If G and H are graphs with more than one vertex, then

$$c_{cr}(G \boxtimes H) \leq \min\{c_{cr}(H) \cdot (B(G)+1) + \max\{p_{cr}(H), 1\} \cdot c_{cr}(G),$$
$$c_{cr}(G) \cdot (B(H)+1) + \max\{p_{cr}(G), 1\} \cdot c_{cr}(H)\}.$$

Proof. Let S be a winning strategy for $c_{cr}(H)$ cops on H such that at most $p_{cr}(H)$ cops push Robert. Let $x_1, \ldots, x_{c_{cr}(H)}$ be the vertices in H that the cops start on when using strategy S and, without loss of generality, suppose the $p_{cr}(H)$ cops that Robert can force to push him start on the vertices $x_1, x_2, \ldots, x_{p_{cr}(H)}$. Let c_i denote the cop starting on x_i for each $1 \leq i \leq c_{cr}(H)$. We will use S to develop a winning strategy for $c_{cr}(H) \cdot (B(G)+1) + p_{cr}(H) \cdot c_{cr}(G)$ cops on $G \boxtimes H$.

For a given vertex $v \in V(H)$, we will use the same notation as before where $G.\{v\}$ denotes the subgraph of $G \boxtimes H$ induced by the vertex set $\{(u, v) \mid u \in V(G)\}$. At the beginning of the game on $G \boxtimes H$, $B(G)+1$ cops will be placed on $G.\{x_i\}$ for each $1 \leq i \leq c_{cr}(G)$ and an additional $c_{cr}(G)$ cops will be placed on each $G.\{x_i\}$ for $1 \leq i \leq p_{cr}(H)$. Let C_i denote the set of all cops that are starting on $G.\{x_i\}$. In finitely many turns, $B(G)+1$ cops on each $G.\{x_i\}$ will first use a cop winning strategy to catch Robert's shadow on $G.\{x_i\}$. Then while one cop continues moving to remain on Robert's shadow, the other $B(G)$ cops use the bodyguard winning strategy where they treat Robert's shadow as the president.

The cops then proceed to capture Robert as follows. If for some $1 \leq i \leq c_{cr}(H)$ the strategy S has the cop c_i move from vertex v_1 to v_2 where Robert is not on v_2 when playing on H, then the set of cops C_i will move from $G.\{v_1\}$ to $G.\{v_2\}$ while maintaining the bodyguard winning strategy and while capturing Robert's shadow on $G.\{v_2\}$. The cops are able to accomplish this since, by the construction of $G \boxtimes H$, if a cop is on the vertex (u, v_1) then that cop can move to any vertex in the set $\{(x, v_2) \in V(G \boxtimes H) \mid x \in N_G[u]\}$. If for some $1 \leq i \leq p_{cr}(H)$ the strategy S has the cop c_i move from v_1 to v_2 where Robert is on v_2 when playing on H, then the $c_{cr}(G)$ cops from the set C_i that are not a part of the bodyguard winning strategy will move from $G.\{v_1\}$ to $G.\{v_2\}$ while the remaining $B(G)+1$ cops stay on $G.\{v_1\}$ and continue moving onto Robert's shadow and using the bodyguard winning strategy. By using this strategy, the $B(G)+1$ cops prevent Robert from moving onto $G.\{v_1\}$. The $c_{cr}(G)$ cops will use the cop winning strategy on G to catch Robert on $G.\{v_2\}$ and force Robert to move off of $G.\{v_2\}$. Once Robert has moved off of $G.\{v_2\}$, either by force or by his own choice, the $B(G)+1$ cops on $G.\{v_1\}$ will move onto $G.\{v_2\}$ continuing the same strategy of catching Robert's shadow and using the winning bodyguard strategy.

Since strategy S eventually results in Robert's capture at some vertex, say v_f, by using the above strategy on $G \boxtimes H$ the cops are able to force Robert onto $G.\{v_f\}$. Afterwards, $c_{cr}(G)$ cops will move onto $G.\{v_f\}$ while for every $u \in V(H)$ adjacent to v_f, the subgraphs $G.\{u\}$ each contain at least $B(G)+1$ cops that are preventing Robert from moving off of $G.\{v_f\}$. From here the $c_{cr}(G)$ cops use the cop winning strategy for G to capture Robert on $G.\{v_f\}$.

This proves that $c_{cr}(G \boxtimes H) \leq c_{cr}(H) \cdot (B(G)+1) + \max\{p_{cr}(H), 1\} \cdot c_{cr}(G)$. A similar proof can be done to show that $c_{cr}(G \boxtimes H) \leq c_{cr}(G) \cdot (B(H)+1) + \max\{p_{cr}(G), 1\} \cdot c_{cr}(H)$. □

In Section 6.2, we will make use of the following corollary.

Corollary 6.1.2. If G is a graph and $n \geq 2$, then

$$2\delta(G) + 1 \leq c_{cr}(G \boxtimes P_n) \leq B(G) + c_{cr}(G) + 1.$$

Proof. Since $c_{cr}(P_n) = 1$ and $p_{cr}(P_n) = 1$, by Theorem 6.1.1

$$c_{cr}(G \boxtimes P_n) \leq 1(B(G)+1) + \max\{1,1\}(c_{cr}(G)) = B(G) + c_{cr}(G) + 1.$$

Since P_n has a vertex of degree one, $\delta(G \boxtimes P_n) = 2\delta(G) + 1$. Note that for any graph G, G contains a $\delta(G)$-core. So by Theorem 2.1.1

$$c_{cr}(G \boxtimes P_n) \geq \delta(G \boxtimes P_n) = 2\delta(G) + 1.$$ □

We can generalize Corollary 6.1.2 by considering the strong product of two graphs where either one is a tree.

Corollary 6.1.3. Let G and H be graphs, both on at least three vertices. If G is a tree with ℓ leaves then

$$2\delta(H) + 1 \leq c_{cr}(G \boxtimes H) \leq (\ell + 1)c_{cr}(H) + p_{cr}(H).$$

If H is also a tree then

$$3 \leq c_{cr}(G \boxtimes H) \leq \ell + 2.$$

Proof. The upper bound follows directly from Theorem 2.1.3, Theorem 5.3.6 and Theorem 6.1.1. By using the same argument as in the proof of Corollary 6.1.2, $\delta(G \boxtimes H)$ gives the lower bound. □

The following theorem gives the exact value of the cheating robot number for specific strong products.

Theorem 6.1.4. If $n, m \geq 2$, then

$$c_{cr}(C_n \boxtimes P_m) = 5,$$
$$c_{cr}(C_n \boxtimes C_m) = 8,$$
$$c_{cr}(K_n \boxtimes P_m) = 2n - 1,$$
$$c_{cr}(K_n \boxtimes C_m) = 3n - 1,$$

and

$$c_{cr}(K_n \boxtimes K_m) = mn - 1.$$

Proof. We have that $\delta(C_n \boxtimes P_m) = 5$, $\delta(C_n \boxtimes C_m) = 8$, $\delta(K_n \boxtimes P_m) = 2n - 1$, $\delta(K_n \boxtimes C_m) = 3n - 1$ and $\delta(K_n \boxtimes K_m) = mn - 1$. Thus we have that the cheating robot numbers of these products are bounded below by these values by Theorem 2.1.1. For $K_n \boxtimes P_m$, $K_n \boxtimes C_m$ and $K_n \boxtimes K_m$, the upper bound from Theorem 6.1.1 matches their minimum degrees. For the other two products we will describe a winning strategy for the cops.

Let $v_1, \ldots, v_m$ be the vertices of the path P_m where v_i is adjacent to v_{i+1} for each $1 \leq i \leq m-1$. Suppose Robert is playing against five cops on $C_n \boxtimes P_m$. The cops have a winning strategy by starting in the subgraph $C_n.\{v_1\}$. Three of the cops will use the bodyguard winning strategy to surround Robert's shadow on $C_n.\{v_1\}$ while the other two cops use the cop winning strategy to move onto the vertex Robert's shadow occupies. After finitely many moves the closed neighbourhood of Robert's shadow will be occupied by three cops. From here the cops can use a winning strategy from the cheating robot variant to capture Robert on $C_n.\{v_m\}$ as described in the proof for Theorem 6.1.1.

If Robert is playing against eight cops on $C_n \boxtimes C_m$ then the cops can set up their strategy by placing four cops on two different cycles, $C_n.\{x_1\}$ and $C_n.\{x_2\}$ where $x_1, x_2 \in V(C_m)$, using the cop winning strategy to move a cop onto Robert's shadow on the two cycles, and then using the bodyguard winning strategy to eventually occupy the closed neighbourhood of Robert's shadow on the two cycles. Six cops will be occupying these closed neighbourhoods which leaves two cops to push Robert where needed and capture him using a winning cheating robot strategy on C_m. $\square$

6.2 Strong grids

Huggan and Nowakowski [23] gave the following bounds on the cheating robot number of the k-dimensional strong grid.

Theorem 6.2.1 ([23])**.** If $k \in \mathbb{Z}^+$, then

$$3\left(2^{k-1}\right) - 2 \leq c_{cr}\left(\boxtimes_{i=1}^{k} P_{n_i}\right) \leq 3^k.$$

The lower bound is the size of the largest core in a k-dimensional strong grid. In this section we will improve on the upper bound, starting with a direct approach.

Theorem 6.2.2. If $k \in \mathbb{Z}^+$ and $n_i \geq 4$ for $1 \leq i \leq k$, then

$$c_{cr}\left(\boxtimes_{i=1}^{k} P_{n_i}\right) \leq 1 + \sum_{\delta=0}^{k-1} \binom{k}{\delta} \cdot \sum_{\ell=0}^{\left\lfloor \frac{k-\delta}{2} \right\rfloor} \binom{k-\delta}{\ell}.$$

Proof. We proceed by showing that after a finite number of moves, the cops can force Robert's k-coordinates to have more increases than decreases every time he moves until he is captured. Eventually, Robert will have no option but to move to the vertex $(n_1, n_2, \ldots, n_k)$ where he gets surrounded. Suppose Robert moves from the vertex $\overline{v} = (v_1, v_2, \ldots, v_k)$ with $1 \leq v_i \leq n_k$ to the vertex $\overline{v}' = (v_1', v_2', \ldots, v_k')$ such that $v_i = v_i'$ for exactly $0 \leq \delta < k$ different v_is. We want to count the number of ways $\overline{v}'$ can have at least as many coordinates satisfy $v_j' < v_j$ as coordinates that satisfy

$v'_j > v_j$. If this condition on $\overline{v}'$ holds, at most $\left\lfloor \frac{k-\delta}{2} \right\rfloor$ of the coordinates satisfy $v'_j > v_j$. If the δ coordinates that do not change as Robert moves from $\overline{v}$ to $\overline{v}'$ are fixed, the total number of choices for $\overline{v}'$ is

$$\sum_{\ell=0}^{\left\lfloor \frac{k-\delta}{2} \right\rfloor} \binom{k-\delta}{\ell}.$$

Since there are $\binom{k}{\delta}$ ways to choose the δ coordinates that do not change when Robert moves, there are a total of

$$A_k = \sum_{\delta=0}^{k-1} \binom{k}{\delta} \cdot \sum_{\ell=0}^{\left\lfloor \frac{k-\delta}{2} \right\rfloor} \binom{k-\delta}{\ell}$$

vertices Robert could move to such that he would decrease at least as many of his coordinates as he increases. If a cop occupied each of these A_k vertices, then every time Robert moved he would have no choice but to increase more of his coordinates than decrease. It remains to show that the cops can move to these vertices in finitely many moves.

Let $H = \boxtimes_{i=1}^{k} P'_{n_i}$ where P'_{n_i} is the path obtained by adding a leaf at each end of the path P_{n_i}. Thus every vertex $(x_1, \ldots, x_k)$ of H has coordinates $0 \leq x_i \leq n_i + 1$ for all $1 \leq i \leq k$. Then G is a retract of H via the map

$$f(x_i) = \begin{cases} x_i & \text{if } 1 \leq x_i \leq n_i \\ 1 & \text{if } x_i = 0 \\ n_i & \text{if } x_i = n_i + 1 \end{cases}$$

and so by Theorem 2.2.3, $c_{cr}(G) \leq c_{cr}(H)$. We will allow the cops to play on H while restricting Robert to playing on G to ensure that Robert will always be adjacent to the same number of vertices during every round of the game.

The cops' strategy will be to have A_k of the cops move to the A_k vertices adjacent to Robert that were previously discussed while the last cop moves to the vertex occupied by Robert. Let c_y be one of the A_k cops and assume she is attempting to get to the position $\overline{y} = (y_1, \ldots, y_k)$ that is relative to Robert. If Robert moves from $(v_1 \ldots, v_k)$ to $(v_1 + a_1, \ldots, v_k + a_k)$ where $a_i \in \{-1, 0, 1\}$, then the position $\overline{y}$ will also change by adding a_i to each coordinate. By Lemma 5.5.1, $c(H) = 1$. Furthermore, since the

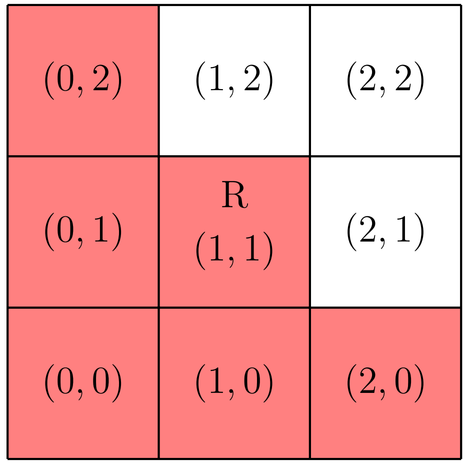


Figure 6.1: The red squares correspond to the vertices Robert is forbidden to move to which are also the vertices whose labels are obtained by adding at most 2 incremental increases of one to the coordinates of $(0, \ldots, 0)$.

position $\overline{y}$ moves based on Robert's movements, the position $\overline{y}$ moves the same way a robber could in the original Cops and Robber. Therefore there exists a strategy for the cop c_y to move onto $\overline{y}$ in finitely many moves. This argument holds for all of the A_k vertices adjacent to Robert as well as the vertex Robert occupies. So, after finitely many moves, the cops can be on the vertices that force Robert to eventually move to the vertex $(n_1, \ldots n_k)$ where he gets surrounded by cops. □

The sequence of values obtained by evaluating the upper bound in Theorem 6.2.2 for different k match the sequence A027914 in the Online Encyclopedia of Integers [36]. The kth integer in the sequence A027914 is defined to be the sum of the coefficients of $x^0, x^1, \ldots, x^k$ in the expansion of $(1 + x + x^2)^k$. To see why these two sequences are the same, note that $(1 + x + x^2)^k$ is the generating function for the number of ways to distribute identical balls into k distinct bins such that each bin contains at most two balls. Thus the sum of the coefficients of $x^0, \ldots, x^k$ in the expansion of $(1+x+x^2)^k$ is equal to the number of ways to distribute at most k identical balls into k distinct bins such that each bin receives at most two balls. The sum in Theorem 6.2.2 is counting the number of vertices on a k-dimensional strong grid that need to be forbidden to force Robert to move closer to the vertex $(n_1, \ldots, n_k)$. Let $(v_1, \ldots, v_k)$ denote the vertex Robert is on. For convenience, we will relabel the vertices such that $(v_1 - 1, \ldots, v_k - 1)$ is mapped to $(0, \ldots, 0)$.

Here the identical balls are the incremental increases by one to the coordinates of $(0,\dots,0)$. The distinct bins are the k-coordinates of the vertices. It takes $2k$ incremental increases by one to go from $(0,\dots,0)$ to $(2,\dots,2)$. Therefore if we add at most k incremental increases, we obtain all of the vertices that are either as close to $(0,\dots,0)$ as they are to $(2,\dots,2)$ or closer to $(0,\dots,0)$ than $(2,\dots,2)$. In other words, we obtain the set of all vertices that have at least as many 0's as 2's in their labels. This is exactly what the sum in Theorem 6.2.2 counts and so the two sequences are the same. Figure 6.1 illustrates the 2-dimensional case of this proof.

Using the bodyguard number, we can obtain a much stronger upper bound on the cheating robot number for k-dimensional strong grids.

Theorem 6.2.3. If $k \in \mathbb{Z}^+$ and $n_i \geq 3$ for $1 \leq i \leq k$, then

$$c_{cr}\left(\boxtimes_{i=1}^{k} P_{n_i}\right) \leq \sum_{j=0}^{k-1} 3^j = \frac{3^k - 1}{2}.$$

Proof. We proceed by induction on k. When $k = 2$, $\sum_{j=0}^{k-1} 3^j = 4$. Huggan and Nowakowski [23] showed that $c_{cr}(P_n \boxtimes P_m) \leq 4$ for any $n, m \geq 2$ and so the theorem holds when $k = 2$. Fix $k \geq 2$ and assume

$$c_{cr}(\boxtimes_{i=1}^{k-1} P_{n_i}) \leq \sum_{j=0}^{k-2} 3^j.$$

We can write $\boxtimes_{i=1}^{k} P_{n_i}$ as $\left(\boxtimes_{i=1}^{k-1} P_{n_i}\right) \boxtimes P_{n_k}$. By applying Theorem 6.1.1 and Theorem 5.5.3, we obtain

$$\begin{aligned}
c_{cr}(\boxtimes_{i=1}^{k-1} P_{n_i} \boxtimes P_{n_k}) &\leq c_{cr}(P_{n_k}) \cdot \left(B(\boxtimes_{i=1}^{k-1} P_{n_i}) + 1\right) + \max\{p_{cr}(P_{n_k}), 1\} \cdot c_{cr}(\boxtimes_{i=1}^{k-1} P_{n_i}) \\
&\leq (1)\left(3^{k-1} - 1 + 1\right) + (1)\left(\sum_{j=0}^{k-2} 3^j\right) \\
&= 3^{k-1} + \sum_{j=0}^{k-2} 3^j \\
&= \sum_{j=0}^{k-1} 3^j
\end{aligned}$$

as required. $\square$

Figure 6.2 gives a table for the values of all the bounds in this section up to

k	$3(2^{k-1})-2$	3^k	$1+\sum_{\delta=0}^{k-1}\binom{k}{\delta}\cdot\sum_{\ell=0}^{\lfloor\frac{k-\delta}{2}\rfloor}\binom{k-\delta}{\ell}$	$\frac{3^k-1}{2}$
1	1	3	2	1
2	4	9	6	4
3	10	27	17	13
4	22	81	50	40
5	46	243	147	121
6	94	729	435	364
7	190	2187	1290	1093
8	382	6561	3834	3280
9	766	19683	11411	9841
10	1534	59049	34001	29524

Figure 6.2: A table comparing all of the proven bounds on the cheating robot of k-dimensional strong grids up to $k = 10$.

10-dimensional strong grids.

6.3 Surrounding numbers for strong products

Using the same technique as in the proof for Theorem 6.1.1, we can obtain a new upper bound on the surrounding number of the strong product of any two graphs.

Theorem 6.3.1. If G and H are graphs, then

$$\begin{aligned}\sigma(G \boxtimes H) \leq \min\{&\sigma(H)\cdot(B(G)+1)+\sigma(G),\\ &\sigma(G)\cdot(B(H)+1)+\sigma(H)\}.\end{aligned}$$

Proof. Let S be a winning strategy in Surrounding Cops and Robbers for $\sigma(H)$ cops on H and let $x_1, \ldots, x_{\sigma(H)}$ be the vertices in H that the cops start on when using strategy S. We now give a winning strategy for $\sigma(H)\cdot(B(G)+1)+\sigma(G)$ cops in Surrounding Cops and Robbers on the graph $G \boxtimes H$.

At the beginning of the game, $B(G)+1$ cops will be placed on $G.\{x_i\}$ for each $1 \leq i \leq \sigma(H)$ and an additional $\sigma(G)+1$ cops will be placed on $G.\{x_1\}$. In finitely many turns, the $B(G)+1$ cops on each $G.\{x_i\}$ will use the cop winning strategy and the bodyguard strategy to indefinitely capture the closed neighbourhood of the robber's shadow in the same way as in the proof of Theorem 6.1.1. These cops will also move to adjacent copies of G as needed according to the strategy S in the same way as in the proof of Theorem 6.1.1. In finitely many moves, the robber will be forced onto the subgraph $G.\{v_f\}$ and for each $u \in V(H)$ such that $uv_f \in E(H)$, the robber will be unable to move onto $G.\{u\}$ due to the $B(G)+1$ cops on each of the $G.\{u\}$. To finish the game, the remaining $\sigma(G)$ cops will move to $G.\{v_f\}$ and use a winning strategy in Surrounding Cops and Robbers to eventually surround the robber in $G.\{v_f\}$. Since the rest of the robber's adjacent vertices outside of $G.\{v_f\}$ are occupied by the groups of $B(G)+1$ cops, the robber is surrounded and the cops win.

This proves that $\sigma(G \boxtimes H) \leq \sigma(H) \cdot (B(G)+1) + \sigma(G)$. A similar proof can be done to show that $\sigma(G \boxtimes H) \leq \sigma(G) \cdot (B(H)+1) + \sigma(H)$. $\square$

The bound in Theorem 6.3.1 is tight. Consider the game being played on $C_n \boxtimes C_m$ where $n \leq 5$. Since $\delta(C_n \boxtimes C_m) = 8$, $\sigma(C_n \boxtimes C_m) \geq 8$. It is easy to see that $\sigma(C_n) = \sigma(C_m) = 2$. From Theorem 5.3.3 we know that $B(C_n) = 2$. Therefore the upper bound is $2(2+1)+2 = 8$. Thus, $\sigma(C_n \boxtimes C_m) = 8$.

We can also obtain an upper bound on the surrounding number of k-dimensional strong grids by leveraging off of Theorem 6.2.3.

Theorem 6.3.2. If $k \in \mathbb{Z}^+$ and $n_i \geq 3$ for $1 \leq i \leq k$, then

$$\sigma\left(\boxtimes_{i=1}^{k} P_{n_i}\right) \leq \frac{3^k+1}{2}.$$

Proof. From Theorem 6.2.3, $c_{cr}(\boxtimes_{i=1}^{k} P_{n_i}) \leq \frac{3^k-1}{2}$. To show that $\sigma(\boxtimes_{n_i}^{k}) \leq \frac{3^k-1}{2}+1$, by Theorem 2.3.7 it suffices to show that $\frac{3^k-1}{2}$ cops can capture Robert on $\boxtimes_{i=1}^{k} P_{n_i}$ with only one cop pushing Robert. In this proof, we will allow Robert to be more powerful, as if he were a robber in the surrounding variant, for part of the game and then only have one cop push him afterwards. For convenience, we will have the cops play on the graph $\boxtimes_{i=1}^{k} P'_{n_i}$ as defined in the proof of Theorem 5.5.3. The vertices of $\boxtimes_{i=1}^{k} P_{n_i}$ and $\boxtimes_{i=1}^{k} P'_{n_i}$ will be labeled the same as in the proof of Theorem 5.5.3.

Begin by placing all of the cops on the vertex $(0, \ldots, 0)$. Let $(v_1, \ldots, v_k)$ denote the vertex Robert is on. For $2 \leq \ell \leq k$, let A_ℓ denote the set of all vertices adjacent to Robert labelled $(x_1, \ldots, x_{\ell-1}, v_\ell - 1, v_{\ell+1}, v_{\ell+2}, \ldots, v_k)$ where $x_i \in \{v_i - 1, v_i, v_i + 1\}$. For each $2 \leq \ell \leq k$, $|A_\ell| = 3^{\ell-1}$. Thus, $\sum_{\ell=2}^{k} |A_\ell| = \frac{3^k-3}{2}$. Since $c(\boxtimes_{i=1}^{k} P') = 1$, $\frac{3^k-3}{2}$ cops can place themselves on the vertices in $A_2, \ldots, A_k$ in finitely many turns.

While these cops are getting into position, we will allow Robert to have the ability to traverse edges that cops traverse as if Robert was a robber in the surrounding variant. Regardless of whether he has this ability or not, it does not change the cops' ability to set up their strategy. This leaves one cop who will, after finitely many turns, move onto the vertex Robert is on and continue to push him every turn. Once all of the cops are in their positions, there are three possibilities every time Robert moves from $(v_1, \ldots, v_k)$ to $(v'_1, \ldots, v'_k)$:

(i) $v'_1 = v_1 + 1$ and $v'_i = v_i$ for all $2 \leq i \leq k$;

(ii) $v'_1 = v_1 - 1$ and $v'_i = v_i$ for all $2 \leq i \leq k$;

(iii) for some $2 \leq i \leq k$, $v'_i \neq v_i$.

We claim that if (iii) occurs enough times, Robert will be forced onto a vertex of the form $(x, n_2, \ldots, n_k)$ where $1 \leq x \leq n_1$. Because of the cops' placements, we claim that Robert is not able to move back to any vertex he was on in any previous round. Suppose for a contradiction that Robert was able to move to a vertex $(u_1, \ldots, u_k)$ he was at on some previous round. By the way the cops are positioned, at least one of Robert's coordinates increases whenever he moves. Let $u_{j_1}, \ldots, u_{j_m}$ with $j_1 \leq j_2 \leq \cdots \leq j_m$ be the coordinates of Robert's position that increase when Robert first moves off of $(u_1, \ldots, u_k)$. Then for Robert to be able to move back onto $(u_1, \ldots, u_k)$, at some point he would need to decrease u_{j_m} without increasing any of $u_{j_m+1}, \ldots, u_k$ which is not possible due to the positioning of the cops. Therefore every time (iii) occurs, Robert moves to a vertex that he can never go back to. So if (iii) occurs at most $\prod_{i=2}^{k} n_i$ times, he will be forced onto a vertex of the form $(x, n_2, \ldots, n_k)$.

Next, we claim that (iii) is impossible for Robert to avoid indefinitely. Suppose for a contradiction that he can. If (i) occurs, since Robert is unable to move back to the vertex he was on in the previous round, he is unable to move such that (ii) occurs. Thus, (i) is Robert's only movement option if he wants to avoid (iii). However, Robert's first coordinate v_1 can only decrease finitely many times until $v_1 = 1$. Once this occurs, on Robert's next move (iii) will occur. A similar argument holds if (ii) occurs. Therefore, (iii) is impossible for Robert to avoid indefinitely and, for as long as the game continues, Robert will continually be forced to change at least one of $v_2, \ldots, v_k$ after finitely many turns.

So after finitely many turns, Robert will be on the path induced by the vertices $(1, n_2, \ldots, n_k)$, $(2, n_2, \ldots, n_k)$, $\ldots$, $(n_1, n_2, \ldots, n_k)$. Once here, Robert will be unable to move off of the path due to the cops' positioning. Eventually the one cop that is continuously pushing Robert will force him onto either $(1, n_2, \ldots, n_k)$ or $(n_1, n_2, \ldots, n_k)$ where he will be surrounded. $\square$

6.4 Lexicographic product

Using a similar approach as in Section 6.1, we can obtain a new upper bound on the cheating robot number of the lexicographic product of two graphs.

Theorem 6.4.1. If G and H are graphs, then

$$\delta(G)|V(H)| + \delta(H) \leq c_{cr}(G \bullet H) \leq |V(H)|c_{cr}(G) + \max\{p_{cr}(G), 1\}c_{cr}(H).$$

Proof. The lower bound is obtained by considering $\delta(G \bullet H)$. The upper bound is obtained by using a nearly identical strategy as in the proof of Theorem 6.1.1 where instead of using the bodyguard number, we have a cop on every vertex of a copy of H to prevent Robert from moving onto that copy of H.

Let S be a winning strategy for $c_{cr}(G)$ cops on G such that at most $p_{cr}(G)$ cops push Robert. Let $x_1 \ldots, x_{c_{cr}(G)}$ be the vertices in G that the cops start on when using strategy S. Without loss of generality assume the cops that start on the vertices $x_1, \ldots, x_{p_{cr}(G)}$ are the cops that Robert can force to push him. Let c_i denote the cop starting on x_i.

Suppose we are playing with $|V(H)|c_{cr}(G) + \max\{p_{cr}(G), 1\}c_{cr}(H)$ cops on $G \bullet H$. We begin the game by placing $|V(H)|$ cops on each $\{x_i\}.H$ where $1 \leq i \leq c_{cr}(G)$ and an additional $c_{cr}(H)$ cops on $\{x_j\}.H$ where $1 \leq j \leq p_{cr}(G)$. The $|V(H)|$ cops on each $\{x_i\}.H$ will distribute themselves so that each vertex in $\{x_i\}.H$ is occupied by exactly one cop from the $|V(H)|$ cops while the $c_{cr}(H)$ cops can be placed anywhere within each $\{x_i\}.H$. Let C_i denote the set of all cops starting on $\{x_i\}.H$. Once the cops are set up, if for some $1 \leq i \leq c_{cr}(G)$ the strategy S has the cop c_i move from the vertex v_1 to v_2 where Robert is not on v_2 when playing on G, then the set of cops C_i will move from $\{v_1\}.H$ to $\{v_2\}.H$ such that every vertex of $\{v_2\}.H$ ends up with a cop on it. If, for some $1 \leq i \leq p_{cr}(G)$, the strategy S has the cop c_i move from v_1 to

v_2 where Robert is on v_2 when playing on G, then the $c_{cr}(H)$ cops from C_i will move from $\{v_1\}.H$ to $\{v_2\}.H$ while the $|V(H)|$ cops on $\{v_1\}.H$ remain at $\{v_1\}.H$. Next, the $c_{cr}(H)$ use a winning strategy to capture Robert on $\{v_2\}.H$, forcing him to another copy of H. Once Robert has moved off $\{v_2\}.H$, the $|V(H)|$ cops on $\{v_1\}.H$ move to $\{v_2\}.H$.

Since strategy S results in Robert's capture on G, the above strategy will trap Robert onto some copy of H, say $\{v_f\}.H$. From there, a group of $c_{cr}(H)$ cops can move onto $\{v_f\}.H$ and capture Robert. □

In the case when G and H are paths and cycles, the lower bound and the upper bound in Theorem 6.4.1 are equal. This leads to the following corollary.

Corollary 6.4.2. For any admissible $n, m \in \mathbb{Z}^+$,

$$c_{cr}(P_n \bullet P_m) = m + 1,$$
$$c_{cr}(P_n \bullet C_m) = m + 2,$$
$$c_{cr}(C_n \bullet P_m) = 2m + 1,$$
$$c_{cr}(C_n \bullet C_m) = 2m + 2.$$

Chapter 7

Further Directions

Since Chapter 6 contains the major results of the thesis, we will first discuss the open problems that arise from the results of Chapter 6. In Section 6.1 and Section 6.2, we introduced new upper bounds on the cheating robot number of the strong product of two graphs. Theorem 6.1.1 gives an upper bound on $c_{cr}(G \boxtimes H)$ where G and H are any two graphs. However, there are cases where this bound is not tight. Consider the graph $C_6 \boxtimes C_6$. Theorem 6.1.4 states that $c_{cr}(C_n \boxtimes C_m) = 8$ for all $n, m \geq 2$, and so we know that $c_{cr}(C_6 \boxtimes C_6) = 8$. By Theorem 6.1.1, $c_{cr}(C_6 \boxtimes C_6) \leq 10$. For which graphs G and H is the bound in Theorem 6.1.1 the exact value for $c_{cr}(G \boxtimes H)$? Theorem 6.2.3 gives an upper bound on the cheating robot number of k-dimensional strong grids. Huggan and Nowakowski showed in [23] that when $k = 2$, $c_{cr}(\boxtimes_{i=1}^{k} P_{n_i}) = 4$ for large enough n_1 and n_2. The upper bound in Theorem 6.2.3 attains a value of four when $k = 2$, so the bound is tight for 2-dimensional strong grids. Is the bound in Theorem 6.2.3 tight for all $k \in \mathbb{Z}^+$? We can ask the same questions regarding the bounds on the surrounding number from Theorem 6.3.1 and Theorem 6.3.2.

As mentioned in Chapter 3, Bradhsaw and Hosseini [10] proved that for any connected, planar graph G, $\sigma(G) \leq 7$. As a consequence of the lower bound in Theorem 2.3.7, we have that $c_{cr}(G) \leq 7$ when G is connected and planar. There are infinitely many examples of planar graphs with $c_{cr}(G) \geq 5$ since there are infinitely many planar graphs that have a minimum degree of five [20]. It is unknown if there exists a connected, planar graph with $c_{cr}(G) = 6$ or $c_{cr}(G) = 7$. A graph that can be embedded on the torus such that none of the graph's edges intersect is called *toroidal*. Bradshaw and Hosseini [10] showed that for connected, toroidal graphs, $\sigma(G) \leq 8$.

They also gave an example of a connected, toroidal graph with $\sigma(G) = 7$. It is known that the graph K_7 is toroidal, and so we have an example of a toroidal graph with a cheating robot number of six. What is the largest cheating robot number among all connected, toroidal graphs? If G is a graph with a given genus, what can be said about the cheating robot number of G?

In Section 2.2 we showed that deleting edges or vertices from a graph may reduce its cheating robot number. Since there exist graphs with subgraphs that have a smaller cheating robot number, for example K_n and any tree on n vertices, the cheating robot number is not monotonic with respect to subgraphs. In Section 5.2, we showed that the bodyguard number is also not monotonic with respect to subgraphs. Determining when and how deleting edges or vertices changes either the cheating robot number or the bodyguard number remains an open problem.

In Section 2.3, we showed that for any graph G, $c_{cr}(G) \leq \sigma(G) \leq c_{cr}(G) + p_{cr}(G)$. Since we know $p_{cr}(G) \leq c_{cr}(G)$ by Theorem 2.3.1, we have $0 \leq \sigma(G) - c_{cr}(G) \leq c_{cr}(G)$. In [23], it was asked whether $0 \leq \sigma(G) - c_{cr}(G) \leq 1$ holds for any graph G. One way of answering this question in the affirmative, by a previous discussion, would be to prove that $p_{cr}(G) \leq 1$ for all graphs. It was proven in [23] that for any outerplanar graph G, $0 \leq \sigma(G) - c_{cr}(G) \leq 1$. A *chord* in a cycle C contained in a graph G is an edge $uv \in E(G)$ where $u, v \in V(C)$ but $uv \notin E(C)$. A *chordal graph* is a graph where every cycle on at least four vertices contained in the graph has a chord. Trees and complete graphs are two examples of chordal graphs. The *clique number* of a graph G, denoted $\omega(G)$, is the largest n such that K_n is an induced subgraph of G. It was shown in [11] that if G is a chordal graph then $\omega(G) - 1 \leq \sigma(G) \leq \omega(G)$. Since any graph containing a k-core has cheating robot number at least k by Theorem 2.1.1, any graph of clique number m has cheating robot number at least $m - 1$. Therefore, for any chordal graph G, $\omega(G) - 1 \leq c_{cr}(G) \leq \sigma(G) \leq \omega(G)$. So for chordal graphs, it is true that the cheating robot number and the surrounding number differ by at most one.

Clarke, Finbow and Mullen in [13] introduce a new variant of Cops and Robber with a nearly identical ruleset to Surrounding Cops and Robbers where the only difference is that on every robber turn, the robber must move to an adjacent vertex. In other words, the robber is *active* since they will never occupy the same vertex on two consecutive robber turns. The fewest number of cops needed to win this game on the graph G is called the *active surrounding number* and is denoted $\sigma_a(G)$. The authors [13] proved that for any connected graph G, $\frac{\sigma_a(G)}{2} \leq c_{cr}(G) \leq \sigma_a(G) + 1$. By making use of the double subdivision operation, they also demonstrate that there exist graphs with $c_{cr}(G) = \sigma_a(G)+1$. However, it remains open whether there exists a graph with $c_{cr}(G) < \sigma_a(G)$.It was also proven in [13] that $\sigma_a(G) \leq \sigma(G) \leq \sigma_a(G)+1$. Therefore, if it were proven that there does not exist a connected graph such that $c_{cr}(G) < \sigma_a(G)$, then we would have $\sigma_a(G) \leq c_{cr}(G) \leq \sigma(G) \leq \sigma_a(G)+1$. This would be another way of proving that $0 \leq \sigma(G) - c_{cr}(G) \leq 1$ holds for all graphs.

Containment, studied in [15, 37], is a variation of Cops and Robber where the cops play on the edges of a graph while the robber plays on the vertices. If $uv, xy \in E(G)$, then the cops are allowed to move from the edge uv to the edge xy if either $x \in uv$ or $y \in uv$. The robber moves the same way as in Cops and Robber, with the only exception being that he cannot traverse an edge that is occupied by a cop. The cops win if in finitely many turns, they can occupy all edges incident to the vertex occupied by the robber. The *containability number* of a graph G, denoted $\xi(G)$, is the fewest number of cops needed to win Containment. In the original paper on Surrounding Cops and Robbers [11], the authors asked if $\sigma(G) \leq \xi(G)$. Here, we ask the same question for the cheating robot number. Is it true that $c_{cr}(G) \leq \xi(G)$? Jungeblut, Schneider and Ueckerdt in a conference paper [27] studied the behaviour of four different variations of Cops and Robber. One is Surrounding Cops and Robbers, one is Containment, one is a variation on Surrounding Cops and Robbers where the robber is allowed to end his turn on a cop vertex, and one is a variation on Containment where the robber is allowed to traverse edges occupied by cops. Besides Surrounding Cops and Robbers, the cheating robot variant has yet to be compared to any of the three other games in that list.

A characterization for copwin graphs was found by Quilliot [41] and independently by Nowakowski and Winkler [35]. Decades later, Clarke and MacGillivray [14] characterized graphs with a cop number of at least k for each $k \in \mathbb{Z}^+$. Let $k \in \mathbb{Z}^+$ and G be a graph on n vertices. Currently, the only known characterization for $c_{cr}(G) = k$ is when $k = 1$ which was found by Huggan and Nowakowski [23]. What can be said about graphs with $c_{cr}(G) = k$ where $k \geq 2$? For Bodyguards and Presidents, we gave characterizations for when $B(G) = 1$, $B(G) = 2$ and $B(G) = n - 1$. What characterizations exist for graphs with $B(G) = k$ where $3 \leq k \leq n - 2$?

One natural variation of Bodyguards and Presidents would be to have the bodyguards play on the edges of the graph while the president plays on the vertices. How would this game compare to the original Bodyguards and Presidents? Would it be possible to obtain new results for the Cops and Robber variants where the cops play on the edges in the same way the original Bodyguards and Presidents allowed us to obtain results for the cheating robot variant and Surrounding Cops and Robbers?

Let G be a connected graph on $n \geq 3$ vertices and e edges. Since G is connected, $n-1 \leq e \leq \binom{n}{2}$. What are the range of values that $B(G)$ can attain? It is easy to construct G so that $B(G) = n-1$ since we can first make one of the vertices of G adjacent to all of the other vertices using $n-1$ edges. Since G contains a universal vertex, $B(G) = n-1$ by Theorem 5.2.3. Therefore, the more difficult question is what is the smallest bodyguard number G could have?

The *Handshaking Lemma* is a well-known result in graph theory that states that for any graph G, $\sum_{v \in V(G)} \deg(v) = 2|E(G)|$. We can obtain a lower bound on $B(G)$ by making use of the Handshaking Lemma. By the Handshaking Lemma and Lemma 5.2.1 we have

$$2e = \sum_{v \in V(G)} \deg(v) \leq n \cdot \Delta(G) \leq n \cdot B(G).$$

Therefore,

$$B(G) \geq \left\lceil \frac{2e}{n} \right\rceil.$$

As an example, consider $e = n-1$. Note that on a graph with n vertices, it is known that the fewest number of edges needed to make G connected is $n-1$. Then by our lower bound, $B(G) \geq \left\lceil \frac{2n-2}{n} \right\rceil = 2$. We know that $B(P_n) = 2$, and so in this case our lower bound is tight. For which $n \leq e \leq \binom{n}{2}$ can we find a graph with bodyguard number $\left\lceil \frac{2e}{n} \right\rceil$?

This line of questioning leads naturally into asking about the bodyguard number of random graphs. Consider a graph on n vertices randomly generated by giving each possible edge a probability $p \in [0,1]$ of being in the graph. Let G be a graph randomly generated in this way. In this case, $B(G)$ is now a random variable. Given n and p, what is the expected value of $B(G)$? For more on random graphs, see Chapter 7 of [5].

In Section 2.3 we introduced the push number of a graph. The push number was used in Theorem 2.3.7 to give an upper bound of the surrounding number. We also used the push number to obtain an upper bound on the cheating robot number of the strong product of two graphs in Theorem 6.1.1. What other bounds on the cheating robot number can be obtained by making use of the push number? By Theorem 2.3.1 we know that for any graph G, $0 \leq p_{cr}(G) \leq c_{cr}(G)$. However, we have yet to find an example where $p_{cr}(G) > 1$. Is it true that $0 \leq p_{cr}(G) \leq 1$ for all graphs?

While we have yet to find an upper bound on the push number that we know is tight, we can ask questions regarding characterizations for graphs with different push numbers. Lemma 2.3.3 gives a property such that graphs that satisfy that property have $p_{cr}(G) = 0$. Theorem 2.3.5 gives a property such that graphs that satisfy that property have $p_{cr}(G) > 0$. If a graph has a push number of zero, what can be said about that graph? Which graphs have a nonzero push number?

The idea of a push number can be generalized to the original game of Cops and Robber. In the cheating robot variant, a push occurs when a cop moves to the vertex Robert is on, forcing him to move or else he is captured next turn. In Cops and Robber, to get the robber into a situation where he is forced to move off his current vertex or else he will lose by the end of the next cop move, a cop needs to move into the robber's neighbourhood. Suppose $c(G)$ cops are playing Cops and Robber on the graph G. Let $p_c(G)$ be the fewest number of cops needed to enter the robber's neighbourhood such that when they do move into the robber's neighbourhood, the cops cannot guarantee capture by the end of their next turn. Is it true that $0 \leq p_c(G) \leq 1$ for any graph?

If instead of playing with the fewest number of cops needed to win we play with more cops, it is possible to decrease the number of pushes. For example, on a graph on n vertices, playing either Cops and Robber or the cheating robot variant with $n-1$ cops allows the cops to win without any pushes. Given a graph G, how many cops are needed to win either Cops and Robber or the cheating robot variant such that none of the cops push the robber?

Bibliography

[1] M. Aigner and M. Fromme, A game of cops and robbers, *Discrete Appl. Math.* **8** (1984) 1–11.

[2] S. S. Akhtar, S. Das, and H. Gahlawat, Cops and Robber on butterflies, grids, and AT-free graphs, *Discrete Appl. Math.* **345** (2024) 231–245.

[3] L. W. Beineke and R. C. Vandell, Decycling graphs, *J. Graph Theory* **25** (1997), 59–77.

[4] A. Berarducci and B. Intrigila, On the cop number of a graph, *Advances in Appl. Math.* **14** (1993) 389–403.

[5] B. Bollobás, *Modern Graph Theory*, Graduate Texts in Mathematics **184**, Springer, New York, NY, 1998.

[6] A. Bonato, E. Chiniforooshan, and P. Prałat, Cops and Robbers from a distance, *Theoretical Comp. Sci.* **411** (2010), 3834–3844.

[7] A. Bonato and G. MacGillivray, Characterizations and algorithms for generalized cops and robbers games, *Contrib. Discrete Math.* **12** (2017) 110–122.

[8] A. Bonato and R. J. Nowakowski, *The Game of Cops and Robbers on Graphs*, Vol. 61 of Student Mathematical Library, American Mathematical Society, Providence, R.I., 2011.

[9] N. Bowler, J. Erde, F. Lehner, and M. Pitz, Bounding the cop number of a graph by its genus, *SIAM J. Discrete Math.* **35** (2021) 2459-–2489.

[10] P. Bradshaw and S. A. Hosseini, Surrounding cops and robbers on graphs of bounded genus, preprint arXiv:1909.09916, (2019).

[11] A. C. Burgess, R. A. Cameron, N. E. Clarke, P. Danziger, S. Finbow, C. W. Jones, and D. A. Pike, Cops that surround a robber, *Discrete Appl. Math.* **285** (2020), 552–566.

[12] N. E. Clarke, *Constrained Cops and Robber*, Ph.D. Thesis, Dalhousie University, 2002.

[13] N. E. Clarke, S. Finbow, and T. Mullen, Surrounding an active robber, preprint (2023).

[14] N. E. Clarke and G. MacGillivray, Characterizations of k-copwin graphs, *Discrete Math.* **312** (2012) 1421–1425.

[15] D. Cryster, N. Komarov, and J. Mackey, Containment: a variation of cops and robber, *Graphs Combin.* **36** (2020) 591–605.

[16] F. V. Fomin, P. A. Golovach, J. Kratochvíl, N. Nisse, and K. Suchan, Pursuing a fast robber on a graph, *Theoretical Comp. Sci.* **411** (2010) 1167–1181.

[17] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-completeness*, W. H. Freeman, San Francisco, CA, 1979.

[18] A. S. Goldstein and E. M. Reingold, The complexity of pursuit on a graph, *Theoretical Comp. Sci.* **143** (1995) 93–112.

[19] G. Hahn and G. MacGillivray, A characterization of k-cop-win graphs and digraphs, *Discrete Math.* **306** (2006) 2492–2497.

[20] M. Hasheminezhad, B. D. McKay, and T. Reeves, Recursive generation of simple planar 5-regular graphs and pentangulations, *J. Graph Algorithms Appl.* **15** (2011) 417–436.

[21] A. Hatcher, *Algebraic Topology*, Cambridge University Press, Cambridge, 2002.

[22] M. A. Huggan, Studies in alternating and simultaneous combinatorial game theory, Ph.D. thesis, Dalhousie University (2019).

[23] M. A. Huggan and R. J. Nowakowski, Cops and an insightful robber, *Discrete Appl. Math.* **295** (2021) 112–119.

[24] M. A. Huggan, R. J. Nowakowski, Cheating Robot Games, *Internat. J. Game Theory* (2023) https://doi.org/10.1007/s00182-023-00862-3.

[25] W. Imrich and S. Klavzar, *Product graphs, Structure and Recognition*, Wiley-Interscience Series in Discrete Mathematics and Optimization, Wiley-Interscience, New York, 2000.

[26] K. Ito, T. Sueishi, Y. Yamakawa, and M. Ishikawa, Tracking and recognition of a human hand in dynamic motion for Janken (rock-paper-scissors) robot, *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, IEEE, (2016), pp. 891–896.

[27] P. Jungeblut, S. Schneider, and T. Ueckerdt, Cops and robber—when capturing is not surrounding, *Lecture Notes in Comp. Sci.* **14093** (2023) 403–416.

[28] A. Keramatipour and B. Bahrak, Zombie number of the Cartesian product of graphs, *Discrete Appl. Math.* **289** (2021) 545–549.

[29] W. B. Kinnersley, Cops and Robbers is EXPTIME-complete, *J. Combin. Theory Ser. B* **111** (2015) 201–220.

[30] F. Lehner, On the cop number of toroidal graphs, *J. Combin. Theory Ser. B* **151** (2021) 250-–262.

[31] M. Mamino, On the computational complexity of a game of cops and robbers, *Theoretical Comp. Sci.* **477** (2013) 48–56.

[32] M. Maamoun and H. Meyniel, On a game of policemen and robber, *Discrete Math.* **17** (1987) 307–309.

[33] S. Neufeld, *The game of Cops and Robber*, M.Sc. Thesis, Dalhousie University, 1990.

[34] S. Neufeld and R. J. Nowakowski, A game of cops and robbers played on products of graphs, *Discrete Math.* **186** (1998) 253–268.

[35] R. J. Nowakowski and P. Winkler, Vertex-to-vertex pursuit in a graph, *Discrete Math.* **43** (1983) 235—239.

[36] OEIS Foundation Inc. (2022), Entry A027914 in The On-Line Encyclopedia of Integer Sequences, https://oeis.org/A027914.

[37] P. Prałat, Containment game played on random graphs: another zig-zag theorem, *Electron. J. Combin.* **22** (2015) #P2.32, 14pp.

[38] B. S. W. Schroeder, The copnumber of a graph is bounded by $\lfloor \frac{3}{2}\text{genus}(G) \rfloor + 3$, *Categorical Perspectives*, Trends Math., Birkhäuser, Boston, MA, 2001, 243–263.

[39] E. Short, J. Hart, M. Vu, and B. Scassellati, No fair!! An interaction with a cheating robot, *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, IEEE, 2010, pp. 219–226.

[40] R. Tošić, On cops and robber games, *Studia Scientarum Mathematicarum Hungarica* **23** (1988) 225–229.

[41] A. Quilliot, Jeux et pointes fixes sur les graphes, *Thèse de 3ème cycle*, Université de Paris VI, 1978, 131–145.

[42] D. B. West, *Introduction to graph theory*, 2nd edition, Prentice Hall, 2001.

[43] World Rock Paper Scissors Association (2021), The Official Rules of Rock Paper Scissors, https://wrpsa.com/the-official-rules-of-rock-paper-scissors/.