Existence and Construction of Certain Combinatorial and Algebraic Structures

by

Fatemeh Hasiri

B.Sc., Amirkabir University of Technology, 2018

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

in the
School of Computing Science
Faculty of Applied Sciences

SIMON FRASER UNIVERSITY
Summer 2021

Declaration of Committee

Name:	**Fatemeh Hasiri**
Degree:	**Master of Science (Computing Science)**
Title:	**Existence and Construction of Certain Combinatorial and Algebraic Structures**
Examining Committee:	**Chair:** Qianping Gu Assistant Professor, Computing Science **Igor Shinkar** Supervisor Professor, Computing Science **Valentine Kabanets** Supervisor Professor, Computing Science **Andrei Bulatov** Examiner Assistant Professor, Computing Science

Abstract

Many problems in combinatorics and computer science use algebraic methods in their solutions, even if they do not look inherently algebraic. Polynomial methods [49], rank arguments [16], and spectral graph theory [48] are among the most popular techniques in this area.

In this thesis, we study two different problems in discrete math and theoretical computer science, and show two results using algebraic methods.

In the first part of the thesis, we study the game of Cops and Robbers [10]. Specifically, we present several families of abelian Cayley graphs whose cop number are asymptotically optimal. More precisely, we present several constructions of families of abelian Cayley graphs G on n vertices whose cop number is $\Omega(\sqrt{n})$. This complements the recent result of Bradshaw [12], who proved that for all abelian Cayley graphs on n vertices the cop number is at most $O(\sqrt{n})$.

In the second part of the thesis, we study explicit constructions of tree codes [46, 45]. We focus on applying techniques from linear algebra to prove existence of certain combinatorial objects, whose explicit construction implies tree codes with constant rate. More specifically, we study lower-triangular totally-non-singular matrices. These matrices are the key ingredients in all recent constructions of tree codes [18, 34].

Keywords: Cops and Robbers; Cayley graphs; Meyniel conjecture; Tree codes; Coding theory; Interactive communication

Acknowledgements

I would like to thank my supervisor, Prof Igor Shinkar, for his dedicated guidance and support during my work on this thesis. I have learned a lot about doing research, asking good questions, and attacking problems from his insight and intuition during my master's program.

I am also grateful to all the people in theory labs at SFU who kept up the spirit of learning and curiosity.

I also thank my family for their unconditional love and support. My mother thought me the value of learning and getting deep, and my father thought me the value of being courageous, hard working, and positive.

Contents

List of Figures

Part I

The Game of Cops and Robbers and Meyniel Extremal Cayley Graphs

Abstract

We study the game of *Cops and Robbers*, where cops try to capture a robber on vertices of a graph. Meyniel's conjecture states that for every connected graph G on n vertices, the cop number of G is upper bounded by $O(\sqrt{n})$, i.e., that $O(\sqrt{n})$ cops suffice to catch the robber. We present several families of Abelian Cayley graphs that are Meyniel extremal, i.e., graphs whose cop number is $\Omega(\sqrt{n})$. This proves that the $O(\sqrt{n})$ upper bound for Cayley graphs proved by Bradshaw [12] is tight up to a multiplicative constant. In particular, this shows that Meyniel's conjecture, if true, is tight to a multiplicative constant even for abelian Cayley graphs.

In order to prove the result, we construct Cayley graphs on n vertices with $\Omega(\sqrt{n})$ generators that are $K_{2,3}$-free. In particular, this implies that the Kövári, Sós, and Turán theorem [30], stating that any $K_{2,3}$-free graph of n vertices has at most $O(n^{3/2})$ edges, is tight up to a multiplicative constant even for Abelian Cayley graphs.

Keywords: Cops and Robbers; Cayley graphs; Meyniel conjecture

Chapter 1

Introduction

The game of *cops and robbers* was first introduced by Nowakowski and Winkler [35], and independently by Quilliot [43]. *Cops and Robbers* is a two player game played on an undirected, finite, simple, connected graph $G = (V, E)$. The first player, called the *cops player*, has c cops, and second player, *the robber*, has 1 robber. The game starts with the first player placing each cop in a vertex in G; then, the second player chooses the initial vertex for the robber. The players play in alternate rounds, where in each turn of the cops the first player moves each cop along an edge to an adjacent vertex or keeps it in its current position, and in robber's turn the robber may move along an edge to an adjacent vertex or stay in place. The cops win if after some finite number of rounds, one of the cops *captures* the robber by occupying the same vertex as the robber. Otherwise, if the robber can avoid this situation forever, the robber wins the game. The minimum value of c for which the cops have a winning strategy is called the *cop number of* G, and is denoted by $c(G)$. We say a graph is *k-cop-win* when $c(G) = k$. The game of cops and robbers was initially studies by Nowakowski and Winkler [35], and independently by Quilliot [43] for the case of $c = 1$ cop, and later generalized by Aigner and Fromme [25] to more cops.

It is clear that for an n-vertex graph we have $c(G) \leq n$ since the cops player can place a cop on every vertex of the graph. Another trivial bound can be obtained by the size of minimum dominating set of a graph. A subset of vertices of G is called a *dominating set* if every edge in the graph has an endpoint in that set. The cops can capture the robber in the first round by putting one cop on each vertex of a dominating set. This shows that the cop number of a graph is bounded from above by the size of its minimum dominating set. However, this bound is far from being tight. For example, consider a path of n vertices and $n - 1$ edges. The minimum dominating set of P_n is $\lceil n/3 \rceil$, while its cop number is 1. Indeed, by starting with a cop at any vertex and going towards the robber in each rounds will result in capturing the robber. It is not hard to see that trees are 1-cop-win as well, and cycles are 2-cop-win.

Meyniel's conjecture, mentioned in Frankl's paper [22], states that for any connected n-vertex graph G it holds that $c(G) = O(\sqrt{n})$.

Conjecture 1.1 (Meyniel's Conjecture)**.** *There exist an absolute constant K such that for every graph G of order n it holds that*

$$c(G) \leq K \cdot \sqrt{n}.$$

A weaker conjecture is the following.

Conjecture 1.2 (Weak Meyniel's Conjecture)**.** *There exist $\varepsilon > 0$ and an absolute constant K such that for every graph G of order n it holds that*

$$c(G) \leq K \cdot n^{1-\varepsilon}.$$

Despite considerable attention this problem has received recently, even the weaker conjecture remains open. The best known upper bound, proved independently by [31, 47, 24], is the following.

Theorem 1.3. *The cop number of any graph on n vertices is upper bounded by $n/2^{(1+o(1))\sqrt{\log_2(n)}}$.*

Sharper results are known for special classes of graphs.

Theorem 1.4 ([25])**.** *For any planar graph, $c(G) \leq 3$*

For random graphs [5, 6, 7, 32, 39, 40, 41], Meyniel's conjecture has been proven to be true. The binomial random graph $G(n,p)$ is defined as a graph with vertex set $[n]$ such that each possible edge appears with probability p.

Theorem 1.5 ([40])**.** *Let $\varepsilon > 0$ and suppose that $p(n-1) \geq (1/2+\varepsilon)logn$. Let $G = (V,E) \in G(n,p)$. Then a.a.s*

$$c(G) = O(\sqrt{n})$$

The *diameter* of a graph is the greatest distance between any pairs of vertices of the graph. Hosseini [27] showed that if we bound the diameter we can achieve better bounds.

Theorem 1.6. *If G is a connected graph with order n and diameter d, we have:*

$$c(G) \leq n^{1-\frac{1}{\log(d)+1}+o(1)}$$

For diameter 2, [50] showed that $\sqrt{2n}$ cops is enough. However, they conjecture that it can be reduced to $\sqrt{n}$.

There are sharper bounds for graphs with bounded genus [44, 11], Cayley graphs [13, 12, 23], and more. For a survey of known related results see [10].

There are several works in the literature [39, 2, 8] describing Meyniel extremal families of graphs, i.e., families of graphs whose cop number is $\Omega(\sqrt{n})$ where n is the number of vertices in the graph. The following theorem is the key idea in most of these works.

Theorem 1.7. *Let G be a graph with minimum degree δ and girth $\geq 8t-3$. Then $c(G) > \delta^t$.*

In particular, the case of $t = 1$ corresponds to graphs with no triangles or 4-cycles, in which case we get $c(G) > \delta$.

Our work contributes new examples of Meyniel extremal families. Specifically, we present several Meyniel extremal families of *abelian Cayley* graphs.

Abelian Cayley graphs are very structured, symmetric graphs. More formally, let G be a finite group, and let subset S be a symmetric subset of G, i.e., satisfying the property that if $a \in S$, then $-a \in S$. The Cayley graph associated with (G, S), denoted by $\mathcal{C}(G, S)$, is the graph whose vertices are the elements of G, and there is an edge between g and h if and only if $g - h \in S$. We say that a Cayley graph $\mathcal{C}(G, S)$ is abelian if the underlying group G is abelian.

Frankl [23] proved that for any connected abelian Cayley graphs it holds that $c(\mathcal{C}(G, S)) \leq \lceil (|S| + 1)/2 \rceil$. Recently, Bradshaw [12] showed that the cop number of any connected abelian Cayley graph on n vertices is bounded by $7\sqrt{n}$. Later they improved the bound to almost $0.9424\sqrt{n} + \frac{7}{2}$ [12]. In this work we prove a lower bound that matches Bradshaw's result up to a multiplicative constant. In particular, if Meyniel's conjecture is true, then it is tight to a multiplicative constant even for abelian Cayley graphs.

Finding algorithms to check whether a graph G is k-cop-win is also an interesting direction. If k is fixed and not part of the input of the algorithm, the problem of finding whether the input graph G of size n is k-cop-win or not can be solved in time $O(n^{O(k)})$ and therefore in polynomial time [4][17][9]. The case where k is an input of the algorithm is proved to be NP-hard [21] and EXP-complete [28]. The polynomial algorithm for the fixed k case can be used for every k not larger than the bound in Theorem 1.3, and this gives a subexponential algorithm for computing the cop number of a graph [21].

Chapter 2

Meyniel Extremal Families of Cayley Graphs

In this chapter, we present the main result of this part of the thesis by showing several examples of Meyniel extremal families of abelian Cayley graphs, i.e., abelian Cayley graphs on n vertices whose cop number is $\Omega(\sqrt{n})$.

Theorem 2.1 (Meyniel Extremal Families of Cayley Graphs)**.** *There exist graph families that are Meyniel extremal. More specifically, we have graph families satisfying the properties below.*

1. *Let n be a sufficiently large integer, and let $G_1 = \mathbb{Z}_n$ be the additive group modulo n. There exists a set of generator $S_1 \subseteq \mathbb{Z}_n$ of size $|S_1| \geq \sqrt{n/8} - O(n^{0.2625})$ such that the graph $\Gamma_1 = \mathcal{C}(\mathbb{Z}_n, S_1)$ has cop number $c(\Gamma_1) \geq |S_1|/3 \geq \frac{\sqrt{n}}{3\sqrt{8}} - O(n^{0.2625}) \geq 0.1178\sqrt{n} - O(n^{0.2625})$.*
2. *Let p be an odd prime, and let $k \in \mathbb{N}$ be a positive even integer. Consider the abelian group $G_2 = \mathbb{Z}_p^k$ of order $n = p^k$. There exists a set of generators $S_2 \subseteq \mathbb{Z}_p^k$ of size $|S_2| = p^{k/2} + 1$ such that the graph $\Gamma_2 = \mathcal{C}(G_2, S_2)$ has cop number $c(\Gamma_2) \geq |S_2|/3 > \sqrt{n}/3 > 0.3333\sqrt{n}$.*
3. *Let p be an odd prime. Consider the abelian group $G_3 = \mathbb{Z}_5 \times \mathbb{Z}_p \times \mathbb{Z}_p$ of order $n = 5p^2$. There exists a set of generators $S_3 \subseteq G_3$ of size $|S_3| = 2p$ such that the graph $\Gamma_3 = \mathcal{C}(G_3, S_3)$ has cop number $c(\Gamma_3) = \lceil (|S_3| + 1)/2 \rceil = p + 1 > \sqrt{n/5} > 0.4472\sqrt{n}$.*

2.1 Our Methods

We prove our results by presenting a family of Cayley graphs $\mathcal{C}(G, S)$ on $|G| = n$ vertices that are $K_{2,t}$-free for some value of t. This shows an example of a family of abelian Cayley graphs that achieves (up to a multiplicative constant) the bound of Kövári, Sós, and Turán [30] for (a special case of) the Zarankiewicz problem, stating that any $K_{2,3}$-free graph on n vertices has at most $O(n^{1.5})$ edges. Specifically, we describe examples of Cayley graphs on n vertices with a generating set of size $\Omega(\sqrt{n})$ that are $K_{2,3}$-free. Apply the following lemmas on these constructions in order to lower bound their cop number.

Lemma 2.2. *Fix $t \geq 3$. If $G = (V, E)$ is a $K_{2,t}$-free graph of minimum degree δ, then $c(G) \geq \delta/t$.*

Lemma 2.3. *Fix $t \geq 3$. If $G = (V, E)$ be a $\{C_3, K_{2,t}\}$-free graph of minimum degree δ, then $c(G) > (\delta + 1)/(t - 1)$.*

Aigner and Fromme [25] showed that if G does not contain C_3 and C_4 then $c(G) \geq \delta$ holds. Frankl [23] showed that if G does not contain C_3 and $K_{2,3}$ then $c(G) \geq (\delta + 1)/2$. Bonato and Burgess [8] also proved similar results.

Proof of Lemma 2.2. We prove that if the number of cops is less than δ/t, then the robber can avoid the cops forever. Specifically, we prove the following claim.

Claim 2.4. *For every $C \subseteq V$ of size $|C| < \delta/t$ and for every $v \in V \setminus C$ there is some $u \in N(v) \cup \{v\}$ that is not dominated by C, i.e., $u \notin \mathcal{D}(C)$, where $\mathcal{D}(C) = \cup_{c \in C} \mathcal{D}(c)$, and $\mathcal{D}(c) = \{c\} \cup N(c)$ are the vertices at distance at most 1 from c.*

Proof of Claim 2.4. Note that since G is $K_{2,t}$-free, every $c \in C$ dominates at most t neighbours of v, i.e., $|N(v) \cap \mathcal{D}(c)| \leq t$.[1] Thus, the number of vertices in $\{v\} \cup N(v)$ that are dominated by C is at most $|\{v\} \cup (\cup_{c \in C}(N(v) \cap \mathcal{D}(c)))| \leq 1 + t|C|$. Therefore, if $|C| < \delta/t$, then the number of vertices in $\{v\} \cup N(v)$ that are dominated by C is *strictly less* than $1 + \delta \leq 1 + \deg(v)$, and hence there is some $u \in N(v) \cup \{v\}$ that is not dominated by C. □

This implies that (i) in the initial round, given the locations $C \subseteq V$ of the cops, the robber can choose a vertex u so that $u \notin \mathcal{D}(C)$, and hence the cops cannot reach u in the first round; (ii) in the subsequent rounds, given the locations C of the cops, if the robber is in the vertex v then it can move to some $u \in N(v)$ so that $u \notin \mathcal{D}(C)$, and hence the cops capture it in the next round. □

Proof of Lemma 2.3. The proof of Lemma 2.3 is analogous to the above. The only difference is the analogue of Claim 2.4 for $\{C_3, K_{2,t}\}$-free graphs.

Claim 2.5. *For every $C \subseteq V$ of size $|C| \leq \delta/(t - 1)$ and for every $v \in V \setminus C$ there is some $u \in N(v) \cup \{v\}$ that is not dominated by C, i.e., $u \notin \mathcal{D}(C)$.*

Proof of Claim 2.5. Note that since G is $\{C_3, K_{2,t}\}$-free, every $c \in C$ dominates at most $t - 1$ neighbours of v, i.e., $|N(v) \cap \mathcal{D}(c)| \leq t - 1$.[2] Furthermore, since G is C_3-free and $v \notin C$, if $v \in \mathcal{D}(c)$, then c dominates no neighbour of v. Thus, the number of vertices in $\{v\} \cup N(v)$ that are dominated by C is at most $(t - 1)|C|$. Therefore, if $|C| \leq \delta/(t - 1)$, then the number of vertices in $\{v\} \cup N(v)$ dominated by C is at most $\delta \leq \deg(v)$, and hence $\exists u \in N(v) \cup \{v\}$ not dominated by C. □

[1] If c is not a neighbour of v, then it can dominate at most $t - 1$ other neighbours of v. Otherwise it can dominate at most $t - 1$ neighbours of v other than itself.

[2] If c is not a neighbour of v, then it can dominate at most $t - 1$ other neighbours of v. Otherwise it can dominate no neighbour of v other than itself.

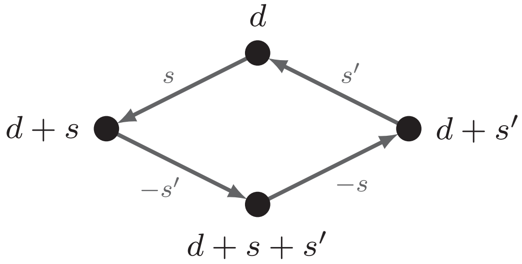

Figure 2.1: A trivial cycle

The rest of the proof is exactly as in the proof of Lemma 2.2. □

We will also need the following observation on Cayley graphs. Let $\Gamma = \mathcal{C}(G, S)$ be a Cayley graph with a symmetric set of generators S. A 4-cycle (or a $K_{2,2}$) in Γ is a collection of 4 edges corresponding to some generators $a, b, c, d \in S$ such that $a + b + c + d = 0$ (the elements are not necessarily distinct). Observe first that any Cayley graph Γ trivially contains a 4-cycle. Indeed, for any $s, s' \in S$ and any $d \in G$ and $d' = d + s + s'$ the vertices $\{d, d'\} \cup \{d + s, d + s'\}$ span a $K_{2,2}$. Such 4-cycles in Γ will be called "trivial", as they correspond to the trivial four tuple of elements in S whose sum is zero, namely, $s + s' + (-s) + (-s') = 0$.

The following observation will be used several times in this paper.

Observation 2.6. *Let $\Gamma = \mathcal{C}(G, S)$ be a Cayley graph with a symmetric set of generators S. If Γ contains no non-trivial 4-cycles, then Γ is $K_{2,3}$-free.*

Proof. Suppose toward contradiction that Γ contains a copy of $K_{2,3}$ with vertices $\{d, d'\}$ on one side, and $\{t_1, t_2, t_3\}$ on the other side. Then S contains the generators $\{s_1 = t_2 - d, s_2 = d' - t_2\}$, and their negations, $\{-s_1, -s_2\}$. Since the 4-cycle (d, t_2, d', t_1) it trivial, it must be the case that the edge (t_1, d) is labeled with $-s_2$ and $t_1 = d + s_2$ Similarly, since the 4-cycle (d, t_2, d', t_3) it trivial, it must be the case that the edge (t_3, d) is labeled with $-s_2$ and $t_3 = d + s_2$ This implies that $t_1 = t_3$ are the same vertex, contradicting the assumption that Γ contains a contains a copy of $K_{2,3}$. See Fig. 2.2. □

We will also need the following simple number theoretic lemma.

Lemma 2.7. *Let $p \geq 3$ be a prime, and let a, b, c, d be integers such that*

$$a + b \equiv c + d \bmod p$$
$$a^2 + b^2 \equiv c^2 + d^2 \bmod p \ .$$

Then either ($a \equiv c \bmod p$ and $b \equiv d \bmod p$) or ($a \equiv d \bmod p$ and $b \equiv c \bmod p$).

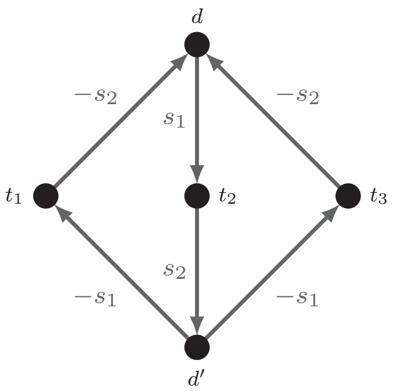

Figure 2.2: Why there can't be any $K_{2,3}$ in graphs with no non-trivial 4-cycle: If the left cycle and the right cycle are both trivial, then $t_1 = t_3$.

Proof. Suppose that $a \not\equiv c \bmod p$, and therefore $b \not\equiv d \bmod p$. Then equation $a^2+b^2 \equiv c^2+d^2 \bmod p$ implies that $(a-c)(a+c) \equiv (d-b)(d+b) \bmod p$, and since $a-c \equiv b-d \not\equiv 0 \bmod p$, it follow that $a+c \equiv b+d \bmod p$. this gives us the following system of equations.

$$a - c \equiv d - b \bmod p$$
$$a + c \equiv d + b \bmod p \ .$$

It is easy to see that all solutions must satisfy $a \equiv d \bmod p$ and $b \equiv c \bmod p$, as required. □

In this section we prove Theorem 2.1 and Theorem 3.1.

2.2 Proof of Meyniel Extremality of Family 1

In this sections we will prove Item 1 of Theorem 2.1.

Fix a prime number $p \geq 5$. For all $a \in \mathbb{N}$ define $s_a = (p^2 + (a^2 \bmod p)p + a) \bmod 8p^2$, where $a^2 \bmod p$ is treated as an integer in $\{0, 1, \ldots, p-1\}$. Note that $p^2 \leq s_a \leq 2p^2 - 2$ for all $0 \leq a \leq p-1$ (where s_a is treated as integer).[3]. Define the sets $S^+ = \{s_a : a \in \{0, 1, \ldots, (p-1)/2\}\}$, $S^- = -S^+$, and let $S = S^+ \cup S^-$.

Lemma 2.8. *The set S satisfies the following properties.*

1. *$s_a \neq s_{a'}$ for all $0 \leq a < a' \leq (p-1)/2$. In particular, $|S| = p+1$.*

2. *For any $s_1, s_2, s_3 \in S$ it holds that $2 \leq |s_1 + s_2 + s_3| < 6p^2$.*

[3] Indeed, for $0 \leq a \leq p-2$ we have $s_a \leq p^2 + (p-1)p + a \leq 2p^2 - 2$, and for $a = p-1$ we have $s_a = p^2 + p + (p-1) \leq 2p^2 - 2$.

3. *Let $s_1 \geq s_2 \geq s_3 \geq s_4$ be elements in S such that $s_1 + s_2 + s_3 + s_4 = 0$. Then $s_1 = -s_4$ and $s_2 = -s_3$.*

Proof. For Item 1 observe that all s_a's are distinct, as they are distinct modulo p, and analogously, all elements in S^- are distinct. It is also clear that S^+ and S^- are disjoint.

For Item 2, let $a_1, a_2, a_3 \in \{0, 1, \ldots, (p-1)/2\}$ be such that $s_i \in \{\pm s_{a_i}\}$ for all $i = 1, 2, 3$. Suppose first that $s_1, s_2, s_3 \in S^+$, i.e., $s_i = s_{a_i}$ for all $i = 1, 2, 3$. Then the sum $s_1 + s_2 + s_3 = s_{a_1} + s_{a_2} + s_{a_3}$ is between $3p^2$ and $3(2p^2 - 2) < 6p^2$. Similarly, if $s_1, s_2, s_3 \in S^-$, then $s_i = -s_{a_i}$ for all $i = 1, 2, 3$, and hence $-6p^2 < -3(2p^2 - 2) \leq s_1 + s_2 + s_3 \leq -3p^2$, as required.

Next, consider the case where two elements are in S^+ and one is in S^-. Then, the sum of the corresponding elements is $s_{a_1} + s_{a_2} - s_{a_3} \geq p^2 + p^2 - (2p^2 - 2) \geq 2$, as required. The case of one element in S^+ and two elements in S^- is similar.

For Item 3 consider the cases based on how many elements s_i's are in S^+ or in S^-.

- If all four elements are in S^+ or all four elements are in S^-, then their sum cannot be zero.
- If three elements are in S^+ and one element is in S^-, then their sum cannot be zero, as $s_1 + s_2 + s_3 + s_4 \geq 3p^2 - (2p^2 - 2) = p^2 + 2 > 0$. Similarly, if three elements are in S^- and one element is in S^+.
- Finally, consider the case where $s_1, s_2 \in S^+$ and $s_3, s_4 \in S^-$. Let $a_1, a_2, a_3, a_4 \in \{0, 1, \ldots, (p-1)/2\}$ be such that $s_1 = s_{a_1}, s_2 = s_{a_2}, s_3 = -s_{a_3}, s_4 = -s_{a_4}$, and hence $s_{a_1} + s_{a_2} = s_{a_3} + s_{a_4}$. Observe that by definition of s_{a_i} this implies

$$a_1 + a_2 \equiv a_3 + a_4 \bmod p$$
$$a_1^2 + a_2^2 \equiv a_3^2 + a_4^2 \bmod p \ .$$

  By Lemma 2.7 all solutions to this system of equations satisfy ($a_1 = a_3, a_2 = a_4$) or ($a_1 = a_4, a_2 = a_3$). Therefore, the assumption $s_1 \geq s_2 \geq s_3 \geq s_4$ implies that $a_1 = a_4$ and $a_2 = a_3$. This completes the proof of Lemma 2.8. □

We are now ready to prove Item 1 of Theorem 2.1. Fix an integer n. Baker, Harman, and Pintz proved in [3] that for all sufficiently large x, there exists a prime between $x - x^{0.525}$ and x. In particular, for $x = \sqrt{n/8}$ there exists a prime p such that $\sqrt{n/8} - (n/8)^{0.2625} \leq p \leq \sqrt{n/8}$.

Let $S_1 = S \cup \{-1, 1\}$ be the set of generators in $\mathbb{Z}_n$, where $S = S^+ \cup S^-$ is as above. Note that $|S_1| \geq |S| = 2p$, and Γ_1 is connected since S_1 is a generating set of $\mathbb{Z}_n$ as $1 \in S_1$.

Claim 2.9. *The Cayley graph $\Gamma_1 = \mathcal{C}(\mathbb{Z}_n, S_1)$ is $\{C_3, K_{2,4}\}$-free.*

Proof. By definition, Γ_1 contains a C_3 if and only if there are three elements in S_1 whose sum is 0 in $\mathbb{Z}_n$. It follows from Lemma 2.8 that he sum of any 3 elements in S is between 2 and $6p^2$, and hence cannot be 0 in $\mathbb{Z}_n$. It is also easy to see there are no $s_1, s_2 \in S$ such that $|s_1 - s_2| = 1$, and hence, Γ_1 in C_3-free.

Next we show that Γ_1 is $K_{2,4}$-free. Recall that a 4-cycle in Γ_1 is a collection of four edges corresponding to four elements $a, b, c, d \in S_1$ such that $a + b + c + d = 0$. Also, recall that a 4-cycle is called "trivial" if the sum is of the form $s + s' + (-s) + (-s') = 0$.

Note that if $s_1 + s_2 + s_3 + s_4 \equiv 0 \bmod n$, then $s_1 + s_2 + s_3 + s_4 = 0$ as an integer, because $|s| < 2p^2$ for all $s \in S_1$ and $n \geq 8p^2$. Therefore, by Lemma 2.8 Item 3 any nontrivial 4-cycle in Γ_1 must contain an edge $(d, d + s)$ such that $s \in \{-1, 1\}$. Furthermore, by Lemma 2.8 Item 2 it follows that any nontrivial 4-cycle in Γ_1 must contain at least two such edges. This implies that Γ_1 is $K_{2,4}$-free. □

By applying Lemma 2.2, we get $c(\Gamma_1) \geq |S_1|/3 = \frac{p}{3} \geq \frac{\sqrt{n}}{3\sqrt{8}} - O(n^{0.2625})$, as required.

2.3 Proof of Meyniel Extremality of Family 2

In this section we prove Item 2 of Theorem 2.1.

For the proof we consider the finite field $\mathrm{GF}(p^k)$, and treat $\mathbb{Z}_p^k$ as the additive group of $\mathrm{GF}(p^k)$. Let $q = p^{k/2}$. Recall that p is an odd prime and k is even, and hence q is an odd prime power. Define the set of generators to be

$$S_2 = \{s \in \mathrm{GF}(p^k) : s^{q+1} = 1\} \ ,$$

where the power s^{q+1} is in the field $\mathrm{GF}(q^2)$. Note that since q is odd, S_2 is, indeed, symmetric as for all $s \in S_2$ we have $(-s)^{q+1} = (-1)^{q+1} \cdot s^{q+1} = 1$, and hence $-s \in S_2$. Also note that $|S_2| = q+1$, since the multiplicative group of $\mathrm{GF}(p^k)$ is a cyclic group of order $p^k - 1 = q^2 - 1$, and hence contains a generating element α of order $q^2 - 1 = (q+1)(q-1)$. Therefore $S_2 = \{\alpha^{(q-1)i} : i \in \{0, 1, 2, \ldots, q\}\}$.

Claim 2.10. *The graph $\Gamma_2 = \mathcal{C}(G_2, S_2)$ is $K_{2,3}$-free. In particular, for all $a_1, b_1, a_2, b_2 \in S_2$ such that $a_1 \neq -b_1$, $a_2 \neq -b_2$, and $\{a_1, b_1\} \neq \{a_2, b_2\}$ it holds that $a_1 + b_1 \neq a_2 + b_2$.*

Proof. If d_1, d_2 are distinct elements of $\mathrm{GF}(q^2)$, then the number of vertices in Γ_2 adjacent to both d_1, and d_2 is equal to the number of solutions of the below system of equations.

$$\begin{aligned}(x - d_1)^{q+1} &= 1 \\ (x - d_2)^{q+1} &= 1 \ ,\end{aligned}$$

or equivalently

$$\begin{aligned}(x - d_1)(x^q - d_1^q) &= 1 \\ (x - d_2)(x^q - d_2^q) &= 1 \ .\end{aligned}$$

This is a special case of system of equations (4) in [29] ($K = \mathrm{GF}(p^k), t = 2, a_{ij} = d_j^{q^{i-1}}, x_i = x^{q^{i-1}}, b_j = 1$). Thus, according to Theorem 3 in [29], the system of equations has at most $t! = 2$ solutions. Therefore, the Cayley graph $\mathcal{C}(G_2, S_2)$ is $K_{2,3}$-free.

For the "in particular" part, note that if we had two distinct pairs $\{a_1, b_1\}$ and $\{a_2, b_2\}$ with $a_1 \neq -b_1$ and $a_2 \neq -b_2$ such that $a_1 + b_1 = a_2 + b_2$, then we would get a copy of $K_{2,3}$ in Γ_2 with the vertices $\{d_1 = 0, d_2 = a_1 + b_1\}$ on one side and $\{a_1, b_1, a_2\}$ on the other side. □

Finally, observe that S_2 is a generating set for $\mathbb{Z}_p^k$. Indeed, by the "in particular" part of Claim 2.10 the set S_2 spans at least $\binom{|S_2|}{2} = \binom{q+1}{2} > q^2/2$ elements of G, as for any pair $a, b \in S_2$ with $a \neq -b$ produces a different sum in G_2. Since the number of elements spanned by S_2 divides q^2, it must be the case that S_2 generates the entire group $\mathbb{Z}_p^k$, and hence $\mathcal{C}(\mathbb{Z}_p^k, S_2)$ is connected.

Using Lemma 2.2, we conclude that $c(\Gamma_2) \geq |S_2|/3 = (q+1)/3 > \sqrt{n}/3$, as required.

2.4 Proof of Meyniel Extremality of Family 3

In this section we prove Item 3 of Theorem 2.1.

Consider the abelian group $G_3 = \mathbb{Z}_5 \times \mathbb{Z}_p \times \mathbb{Z}_p$ of order $n = 5p^2$. Define the set of generators $S_3 = \{(1, a, a^2) : a \in \mathbb{Z}_p\} \cup \{(-1, -a, -a^2) : a \in \mathbb{Z}_p\}$, where a^2 is taken modulo p. Note that S_3 is indeed a symmetric set of size $|S_3| = 2p$.

Let $\Gamma_3 = \mathcal{C}(G_3, S_3)$ be the corresponding Cayley graph. We show below that Γ_3 is $\{C_3, K_{2,3}\}$-free, and hence by Lemma 2.3 we conclude that $c(\Gamma_3) \geq |S_3|/2 = p$, as required.

Claim 2.11. *The graph Γ_3 is connected and $\{C_3, K_{2,3}\}$-free.*

Proof. Observe that Γ_3 has no triangles because there are no three elements in S whose sum is 0 in the first coordinate.

Next we claim that Γ_3 is $K_{2,3}$-free. This is done by proving that Γ_3 contains no non-trivial 4-cycles. Indeed, let $s_1, s_2, s_3, s_4 \in S_3$ be four generators such that $s_1 + s_2 + s_3 + s_4 = 0$ in G_3, By looking at the first coordinate (to $\mathbb{Z}_5$), it must be the case that two of the s_i's are in $\{(1, a, a^2) : a \in \mathbb{Z}_p\}$ and two are in $\{(-1, -a, -a^2) : a \in \mathbb{Z}_p\}$. Assume without loss of generality that $s_1 = (1, a, a^2)$, $s_2 = (1, b, b^2)$, $s_3 = (-1, -c, -c^2)$, $s_4 = (-1, -d, -d^2)$ for some $a, b, c, d \in \mathbb{Z}_p$. Therefore, if $s_1 + s_2 + s_3 + s_4 = 0$, then $a + b \equiv c + d \bmod p$ and $a^2 + b^2 \equiv c^2 + d^2 \bmod p$. Therefore, by Lemma 2.7 we either have ($a = c$ and $b = d$) or ($a = d$ and $b = c$). Therefore, Γ_3 contains only trivial 4-cycles, as required. Therefore, by Observation 2.6 the Cayley graph Γ_3 is $K_{2,3}$-free.

In order to see that Γ_3 is connected, note that the elements spanned by S_3 form a subgroup of G_3, and hence $5p^2$ is divisible by $|\mathrm{span}(S_3)|$. Since Γ_3 contains no non-trivial 4-cycles, it follows that the number of elements spanned by S_3 is $|\mathrm{span}(S_3)| \geq |\{s + s' : s, s' \in S_3, s' \neq s\}| \geq \binom{|S_3|}{2} \geq \binom{2p}{2} = p(2p-1)$, and hence S_3 spans the entire group G_3. □

By Lemma 2.3 the cop number of Γ_3 is $c(\Gamma_3) \geq (|S_3|+1)/2 \geq (2p+1)/2$. On the other hand, according to [23, Theorem 1] we have $c(\Gamma_3) \leq \lceil (|S_3|+1)/2 \rceil = \lceil (2p+1)/2 \rceil = p+1$. Therefore, $c(\Gamma_3) = p+1$.

It is worth noting that all the constructions in Theorem 2.1 are examples of sidon sets in finite groups [1][36].

Chapter 3

Cayley Graphs of any Group with High Cop Number

In this chapter we prove that for *any* abelian group G of order n, such that n is not divisible by 2 or 3, there exists a set of generators $S \subseteq G$ such that the cop number of the corresponding Cayley graph $\mathcal{C}(G,S)$ is lower bounded by $\Omega(n^{1/3})$.

Theorem 3.1. *Let G be any abelian group of order n that contains no elements of order 2 or 3. There exists a symmetric set of generator $S \subseteq G$ of size $|S| = \Omega(n^{1/3})$, such that the Cayley graph $\Gamma = \mathcal{C}(G,S)$ is connected and its cop number is $c(\Gamma) \geq |S|/2 \geq \Omega(n^{1/3})$.*

Let G be an abelian group of order n such that G has no elements of order 2 or 3. We construct a generating set $S \subseteq G$ using Algorithm 1. Before describing the algorithm we make the following notation.

Notation 3.2. *For a subset $S \subseteq G$ let $F_1(S) = \{a+b+c : a,b,c \in S\}$, $F_2(S) = \{a : \exists b,c \in S \text{ s.t. } b+c+a+a = 0\}$, and $F_3(S) = \{a : a+a+a \in S\}$. Define $F_S = F_1(S) \cup F_2(S) \cup F_3(S)$.*

Claim 3.3. *Let $S \subseteq G$ be a symmetric set, and suppose that S has no non-trivial 4-cycles. Then, for any $s^* \in G \setminus F_S$ the set $S \cup \{s^*, -s^*\}$ has no non-trivial 4-cycles.*

Proof. Observe first that $S \subseteq F_1(S)$, as for any $s \in S$ we have $s = s + s + (-s) \in F_1(S)$. In particular $S \subseteq F_S$, and thus if $S \cup \{s^*, -s^*\}$ contains a non-trivial 4-cycle $a+b+c+d = 0$, then at least one of the elements must be in $\{s^*, -s^*\}$.

Note that for any three elements $a,b,c \in S$ we have $a+b+c$ in $F_1(S) \subseteq F_S$, and $s^*, -s^* \notin F_S$. Therefore $S \cup \{s^*, -s^*\}$ does not contains a non-trivial 4-cycle with exactly one element in $\{s^*, -s^*\}$.

Suppose now that two of the elements $\{a,b,c,d\}$ are in $\{s^*, -s^*\}$. Since the 4-cycle is non-trivial, it must be that the two of the elements are equal. Without loss of generality suppose that $a = b = s^*$. But then $s^* \in F_2(S)$, and hence $a+b+c+d = 0$ cannot be a non-trivial 4-cycle with two edges outside S.

Similarly, if three of the elements a, b, c, d belong to $\{s^*, -s^*\}$, we may assume without loss of generality that $a = b = c = s^*$. But this implies that $s^* \in F_3(S)$, and hence $a+b+c+d = 0$ cannot be a non-trivial 4-cycle with three edges outside S.

Finally, since G does not contain elements of order 2, it is impossible that all four elements a, b, c, d belong to $\{s^*, -s^*\}$.

This completes the proof of Claim 3.3 □

We are now ready to describe the algorithm.

Algorithm 1 Constructing a generating set S of a group G

```
$S_0 \leftarrow$ a minimal generating set of $G$
$S \leftarrow S_0 \cup -S_0$
while $G \neq F_S$ do
    Choose an arbitrary element $s \in G \setminus F_S$
    $S \leftarrow S \cup \{-s, s\}$
end while
return $S$
```

For the analysis observe first that in the end of each iteration we have $|F_S| \leq |S|^3 + |S|^2 + |S|$. Indeed, we have (i) $|F_1| \leq |S|^3 = k^3$, (ii) $|F_2| \leq |S|^2 = k^2$, as G has no elements of order 2, and (iii) $|F_3| \leq |S| = k$, as G has no elements of order 3. Therefore, since the algorithm ends when $|F_S| = n$, it follow that the output is a set S of size $\Omega(n^{1/3})$.

Note first that since S contains a generating set of G, the graph $\Gamma = \mathcal{C}(G, S)$ is connected. Also, note that since S_0 is a minimal generating set of G, the set S before the loop contains no non-trivial 4-cycles. Indeed, it is not difficult to see that if G contains no elements of order 2, and $S_0 \cup -S_0$ contains a non-trivial four cycle $a + b + c + d = 0$, then S_0 contains a strict subset generating G.

By Claim 3.3 in each iteration of the algorithm, S does not contain a non-trivial 4-cycles in any iteration, and hence, by Observation 2.6 in the end of the algorithm the graph $\Gamma = \mathcal{C}(G, S)$ is $K_{2,3}$-free. Therefore, by Lemma 2.2 $c(\Gamma) \geq |S|/3 \geq \Omega(n^{1/3})$, as required.

Chapter 4

Final Remarks and Open Problems

We showed in Theorem 2.1 several classes of Meyniel extremal Cayley graphs. Our Theorem 3.1 shows a weaker result for general groups, namely, that any group satisfying certain mild conditions has a Cayley graph of order $\Omega(n^{1/3})$. This raises the following natural question.

Question 4.1. *Is it true that any group G has a Cayley graph that is Meyniel extremal?*

Question 4.2. *Pralat [39] showed a family of graphs on n vertices whose cop number $\geq \sqrt{n/2} \cdot (1 - o(1))$. It would be interesting to find a family of* Cayley *graphs matching these parameters.*

Question 4.3. *Finding bounds on cop number for* non-abelian *Cayley graphs is an open problem. It would be interesting to extend the $O(\sqrt{n}$ upper bound of Bradshaw [12] to non-abelian Cayley graphs.*

Question 4.4. *Improving the coefficient in the best known upper bound of graphs with diameter 2, and the exponential factor of the bound for graphs of diameter 3 and 4 is also a nice direction. The current best known bounds for graphs of diameter $2, 3, 4$ is respectively $\sqrt{2n}, n^{\frac{4}{7}+o(1)}, n^{\frac{3}{5}+o(1)}$ [50][27].*

Question 4.5. *For planar graphs it is well-known that the cop number is at most 3(Theorem 1.4). For directed planar graphs however, there is no good upper bound.*

Question 4.6. *Finding properties of minimal k-cop-win graphs is an important direction. For $k = 3$ it is known to be the Peterson graph, but for $k = 4$ it is open.*

Part II

Good Linear Tree Codes

Abstract

Tree codes are combinatorial structures for encoding information in interactive coding theory. Schulman defined them in [46, 45] and proved that there exist infinite family of tree codes that have both constant distance and constant alphabet. However, giving an explicit construction of tree codes has remained an outstanding open problem since then. Recently, Cohen, Haeupler, and Schulman [18] introduced an explicit construction of a family of tree codes with constant distance but over alphabet of polylogarithmic size. This was a major breakthrough over a two-decade-old construction that has an exponentially larger alphabet of size $poly(n)$. In this work, we show how proving a stronger version of one of the theorems in their paper can lead to an explicit construction of tree codes with constant distance and constant alphabet and we give a probabilistic argument to show that this theorem should be true. We show how totally k-non-singular matrices can be used to construct good tree codes and we give random construction of such matrices.

Keywords: Tree codes; Coding theory; Coding for interactive communication; Totally non-singular matrices

Chapter 5

Introduction

5.1 Interactive Communication

The standard setting of interactive communication is the following. Two parties (Alice and Bob) are given two strings $x, y \in \{0,1\}^n$, and they aim to compute $f(x,y) \in \{0,1\}$ by communicating as few bits as possible. In the setting of *coding for interactive communication* the channel that the parties are communicating over is noisy. The parties goal in this setting is to compute $f(x,y)$ correctly with high probability despite the noise and, again, sending as few bits as possible. The interactive communication is performed by a protocol $\pi = (\pi_A, \pi_B)$, where π_A, π_B are algorithms run by Alice and Bob respectively. At each round, each of them determines the next bit that the party should send, and is a function of the party's input and the message the party has received so far (the *transcript*). For example, in the first round Alice sends $\pi_A(x, 1, \emptyset)$, and Bob sends $\pi_B(y, 1, trans_B)$. After n rounds, Alice and Bob decide on their output and send $\pi_A(x, trans_A), \pi_B(y, trans_B)$ respectively. Tree codes are one of the *coding schemes* that given a noiseless protocol π_0 constructs a noise-resilient protocol π_1 that computes the same function.

The goal of *interactive coding schemes* is to build a protocol CS_ε that given a noiseless protocol π builds a protocol $CS_\varepsilon(\pi) = \pi_\varepsilon$ that is resilient to a noisy channel of error rate ε. That is, the protocol is resilient to *arbitrary* noise that changes up to ε fraction of the bits sent over the communication channel. In this text, we assume the noise of the protocol is bounded by parameter ε as opposed to other models, including Binary symmetric models, in which every bit is flipped with probability ε.

In general, encoding used in these machines is required to satisfy a set of properties.

1. *Being online:* The encoding of each transmission can only depend on previous transmissions. That's because each party should communicate one transmission at a time.

2. *Distance:* Like the standard coding model, the encoding of two different sequences of transmissions should have high distance with respect to their length so that the parties will be able to eventually decode the message they have received.

Regular error-correcting codes cannot be used in this setting to encode the entire communication because they are not online. Also, they can't be used to encode each transmission. If the adversary corrupts only single transmission, the entire communication will be erroneous.

It is also worth noting that we just study non-adaptive protocols, in which the turn of speech is fixed, and particularly *alternating protocols*, where Alice speaks at odd rounds and Bob speaks at even rounds.

5.2 Tree Codes

Tree codes are combinatorial structures first introduced by Schulman[45] as a building block for interactive coding schemes over noisy channels.

In this section we formally define Tree Codes. In the next section, we describe how they are used in a coding scheme for interactive communication.

Definition 5.1. *Let $\mathbb{F}$ be a field, $n, l \in \mathbb{N}$, and $z, z_1, z_2 \in \mathbb{F}^n$. We define*

- $[n] = \{1, \ldots, n\}$
- $z_{[i,j]} = (z_i z_{i+1} \ldots z_j)$, $\forall i, j \in [n]$
- $split(z_1, z_2) = min_i\{z_{1_i} \neq z_{2_i}\}$
- $\mathsf{weight}(z) = |\{i : z_i \neq 0\}|$
- $\delta(z_1, z_2) = \mathsf{weight}(z_1 - z_2)/n$
- $\delta_{[i,j]}(z_1, z_2) = |\{k \in [i, j] : z_{1_k} \neq z_{2_k}\}|/(j - i + 1)$

Definition 5.2 (Tree code definition without tree)**.** *A function $TC : \Sigma_{in}^n \to \Sigma_{out}^n$ is called a tree code with distance δ if it has following properties:*

- *It is online. i.e. for every i in $[n]$ and every x in Σ_{in}^n, $TC(x)_i$ only depends on $x_1, x_2, ..., x_i$*
- *For every $x, y \in \Sigma_{in}^n$ with $s = split(x, y)$ and every $1 \leq l \leq n - s$, $\delta_{[s,s+l]}(x, y) \leq \delta$.*

Tree codes are defined using trees in the original paper [45], which is equivalent to the Definition 5.2. We will also bring the definition using trees, but we mostly use Definition 5.2.

Definition 5.3 (Tree code definition using tree)**.** *A d-ary tree code of length n over alphabet Σ with distance δ is a rooted d-ary tree of length n with edges labeled with elements of Σ that satisfies the following property:*

- *For every two vertices $u \neq v$ at the same depth k, with the lowest common ancestor $lca(u, v)$ at depth $k - \ell$, let P_u and P_v be the concatenation of the labels on the unique path from $lca(u, v)$ to u and v respectively. Then, $\delta(P_u, P_v) > \delta$.*

The rate of a d-ary tree codes is $1/\log_d(|\Sigma|)$. We say a tree code is asymptotically good if it has non vanishing rate and distance. In the original paper [45] Schulman proved that for any fixed $d \in \mathbb{N}$ and $\delta \in [0,1]$, there exist a d-ary tree code with distance δ over an alphabet of constant size $c(d,\delta)$. In particular,

Theorem 5.4. *[45] For every $\delta < 1$ and $d \in \mathbb{N}$, there exist a d-ary tree code of distance δ over alphabet Σ of size $(cd)^{1/(1-\delta)}$ for some $c < 6$.*

Schulman gave three different proofs for this theorem, all of which probabilistic. In [37], they gave another probabilistic proof whit slightly better probability of success but still constant. Most recently, [19] proved that for $d = 2$, there exist tree codes with positive distance ($\delta = 0.136$) and alphabet of size as small as 4. Furthermore, they showed that for constant distance bounded away from 0, the alphabet size cannot be 3 or less. Using the results of [42] on how lower triangular totally-non-singular matrices can be used to construct tree codes, we present another probabilistic proof of existence of tree codes.

Despite all the attention the tree codes have received since their introduction, no explicit construction of tree codes with constant alphabet and constant distance have been found. By explicit construction we mean an algorithm that given the first i bits of a message m computes the ith bit of the encoding given by the tree code in polynomial time. The trivial construction of constant-distanced tree codes uses alphabet of size 2^n (by encoding the whole path leading to each node). In an unpublished manuscript, Evans, Klugerman, Schulman [20] built a construction of tree codes with alphabet of size polynomial in the depth of the tree. Best rated tree codes are introduced in the recent work of [18] where they build tree codes with alphabet of size $O(\log(n))$. In [14], they give an explicit construction in subexponential time $O(2^\varepsilon)$ but with alphabet size $O(1/\varepsilon)$. There are also candidate constructions based on unproved conjectures in [33][51].

Tree codes were originally introduced as a machinery for coding for interactive communication. However, other applications have been found for it since their introduction by Schulman.

5.3 Tree Codes as a Machinery Used in Coding for Interactive Communication

Tree codes are used as a machinery that allow the parties communicating on a noisy channel to *eventually* decode correctly the message they have received. That is, they detect earlier errors as time goes by. They encode messages in a way that two different messages are encoded the same until they have a difference at some point. As soon as the disagreement occurs, the distance between the encoding of the rest of the messages is guaranteed to be larger than a fixed number.

To be able to design protocols for interactive communication over noisy channels, it is helpful to define the *Pointer Jumping* problem (see Fig. 5.1). In this problem, the input is a complete binary tree T of depth $2n$. Party A has a subset of edges E_A (the blue edges) with exactly one

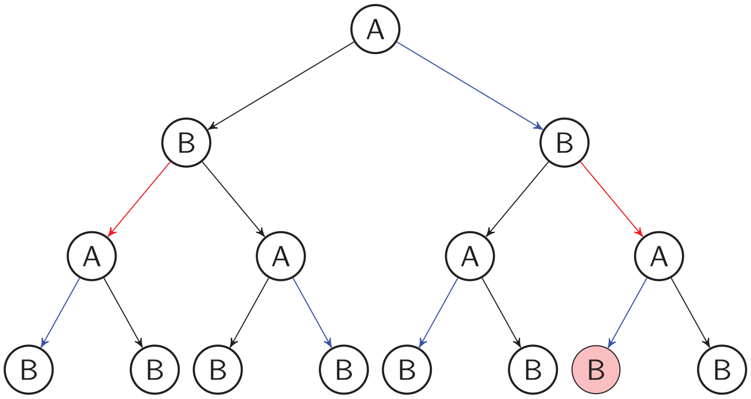

Figure 5.1: Pointer Jumping Problem

edge coming out of every vertex at odd levels, and no edge coming out of vertices at even levels. Similarly, party B has a subset of edges E_B (the red edges) that has exactly one edge coming out of each vertex at even levels, and no edge coming out of vertices at odd levels. The goal is to find the unique path from root to a leaf with edges from $E_A \cup E_B$. Each vertex represents the sequence of bits transmitted so far and each edge of E_A or E_B outgoing from vertex v represents the bit the party would send if communicated messages so far are represented by v. It is easy to see that this is just another interpretation of the interactive communication model.

In his original paper [45], Schulman used tree codes to build a coding scheme that is resilient to error rate of 1/240. In [15], Braverman and Rao improve on Schulman's work to handle larger error rates. Their protocol is resilient to errors that corrupt at most $1/4-\varepsilon$ symbols. These results are for when the alphabet size of the channel is constant. When the channel is binary, [15] built a protocol that is resilient to $1/8-\varepsilon$ fraction of errors at the cost of increasing the communication length by a constant factor. They conjecture that $1/8-\varepsilon$ is the best achievable resilience. However, this is still an open problem.

The idea behind the coding scheme of Braverman and Rao is that they keep track of all the edges they have received so far from the other party, and find the unique rooted path that only crosses those edges and their own edges. Then, they send the unique edge they have that extends this path encoded with a tree code. More formally, let B_i be the set of edges that Bob has sent up to round i, and let B'_i be the set of edges Alice has decoded from the messages of Bob up to round i. Define A_i and A'_i similarly for the edges that Alice sends and Bob decodes up to round i respectively. At this point, if it's Alice's turn, she finds the unique rooted path, P_i that is a subset of $A \cup B'_i$ and communicates the unique edge $e_i \in A$ that extends P_i. For sending e_i, she encodes the entire data she has communicated so far, $a_1 a_2 \ldots a_{i-1} e_i$ by a tree code and sends the last symbol of it. On Bob's turn he sends his next bit using the analogous strategy.

In their coding scheme, Alice simulates algorithm 2 which takes as input a protocol π_0, of length n_0, a noise resilience parameter ε, and an input x with E_x as the set of possible replies by Alice.

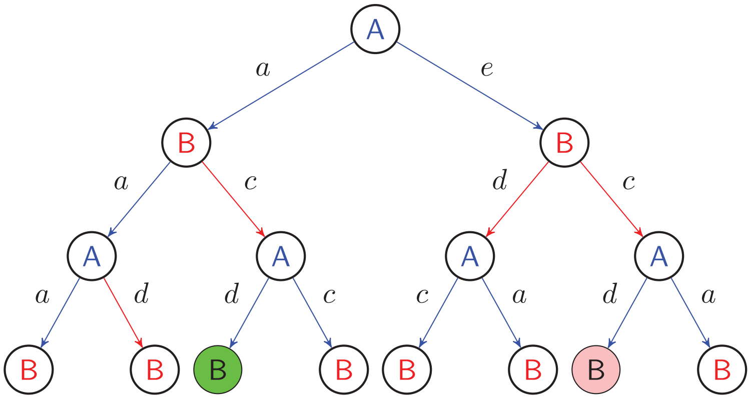

Figure 5.2: There was an error in the first symbol sent by A replacing a with e. Then B responded with c and A responded with d. In the end A thinks they reached the green leaf, but B thinks the reach the red leaf.

Also, a d-ary tree code with distance $\alpha = 1 - \varepsilon$, depth $n = n_0/\varepsilon$, $d = O(n)$, and encoding and decoding algorithms $TCenc_T$ and $TCdec_T$. Bob simulates the symmetric algorithm.

Algorithm 2 The Braverman-Rao simulation

```
$S_A \leftarrow \emptyset$
$recv \leftarrow \emptyset$
for $i = 1$ to $n = n_0/\varepsilon$ do
    $\tilde{S}_B \leftarrow TCdec_T(recv)$
    if $E_x \cup \tilde{S}_B$ has a unique rooted path $P$ then
        $e_i \leftarrow$ the lowest edge in $E_x \cap \tilde{S}_B$ not in $S_A$
        $S_A = S_A \cup e_i$
    else
        $e_i \leftarrow \perp$
    end if
    send the last symbol of $TCenc_T(e_1, e_2, \dots, e_i)$
    receive symbol $r$ from Bob
    $recv \leftarrow recv \circ r$
end for
Output the unique rooted path of length $n_0$ defined by $E_X \cup \tilde{S}_B$
```

Let s_A and r_A be the words that Alice sends and receives respectively. And let s_B and r_B be the words that Bob sends and receive. Also let $\delta_A = \delta(r_A, s_B)$ and $\delta_B = \delta(r_B, s_A)$. For any $r, s \in \Sigma^n$ with split t, let I be the set of all indices $i \in [n]$ where $\delta_{[t,i]}(r, s) \leq \alpha$. They show that if $\delta(r, s) = \beta$, then $|I| \geq (1 - \beta/\alpha)n$. Thus, there will be at least $|I_A| \geq (1 - \frac{2\delta_A}{1-\varepsilon})n$ indices i such that $\delta_{[t,i]}(r_A, s_B) \leq \alpha/2$. Also, $I_B = \{i \in [n] : \delta_{[t,i]}(r_B, s_A) \leq \alpha/2\}$ has size at least $(1 - \frac{2\delta_A}{1-\varepsilon})n$ elements, so in these rounds the parties correctly decode the message received to them by that round. Recall that the protocol has noise resilience parameter ε, so $\delta_A, \delta_B \leq \varepsilon$, and $|I_A|, |I_B| \geq (1 - \frac{2\varepsilon}{1-\varepsilon})n$. This gives that $|I_A \cap I_B| \geq \varepsilon n$, so in at least $\varepsilon n = n_0$, iterations Alice and Bob both have decoded the correct set of edges and communicated the next outgoing edge of the rooted path.

Chapter 6

Tree Code Construction; Random and Explicit

In this section, we show how to construct tree codes using lower-triangular totally k-non-singular matrices. We start with the definition of totally k-non-singular matrices.

Definition 6.1. *Let $k, n \in \mathbb{N}$ and $A \in \mathbb{N}^{n\times n}$, we say A is* k-non-singular*, if for every nonzero vector $v \in \mathbb{Z}^n$ with $|v_i| \leq k$, $Av \neq 0$. And we say A is* totally k-non-singular *(totally non-singular), when every submatrix of A is k-non-singular (non-singular).*

For the tree code application, we will need a restricted variant of the definition above. Specifically, we will need the definition of *lower triangular totally k-non-singular*.

Definition 6.2. *Let $d_1, d_2, n, m \in \mathbb{N}$ with $d_1 \leq n, d_2 \leq m$, and let A be a $n \times m$ matrix. For a set of rows $r_1 < r_2 < \cdots < r_{d_1} \in [n]$ and a set of columns $c_1 < c_2 < \cdots < c_{d_2} \in [m]$ we denote the submatrix formed by these rows and columns by $A_{[r_1,\ldots,r_{d_1}:c_1,\ldots,c_{d_2}]}$.*

We call a lower triangular matrix A, *lower triangular totally k-non-singular* if every submatrix of A whose diagonal is not above the diagonal of A is k-non-singular. The following is the more formal definition.

Definition 6.3. *A matrix $A \in \mathbb{N}^{n\times n}$ is said to be lower triangular totally k-non-singular, if*

1. *A is lower triangular, i.e. $A_{i,j} = 0$ for all $i < j$; and*
2. *for every $s \leq n$, $1 \leq r_1 < \cdots < r_s \leq n$, and $1 \leq c_1 < \cdots < c_s \leq n$ with $c_1 \leq r_1, c_2 \leq r_2, \ldots, c_s \leq r_s$ it holds that $A_{[r_1,\ldots,r_s:c_1,\ldots,c_s]}$ is k-non-singular.*

Lower triangular totally non-singular matrices are defined in a similar way.

Linear tree codes over algebraic fields are defined in similar way to linear error correcting codes.

Definition 6.4 (Linear tree codes)**.** *Let Σ_{in} and Σ_{out} be finitely dimensional vector spaces over fields F_1 and F_2 respectively, and $TC : \Sigma_{in}^n \to \Sigma_{out}^n$ be a tree code. We say TC is linear if and only if $TC(x+y) = TC(x) + TC(y)$ for every $x, y \in \Sigma_{in}^n$*

Finally, given a lower triangular matrix A we can define a tree code as follows.

Definition 6.5. *Let $S \subseteq \mathbb{Z}$, and $A \in \mathbb{Z}^{n\times n}$ be a lower triangular matrix. We define $TC_A : \mathbb{Z}^n \to (\mathbb{Z}^2)^n$ to be the linear tree code such that for every $x \in \mathbb{Z}^n$*

$$TC_A(x) = \begin{bmatrix} x_1, (Ax)_1 \\ x_2, (Ax)_2 \\ \dots \\ x_n, (Ax)_n \end{bmatrix}$$

That is, $TC_A(x)$ computes the matrix-vector multiplication, and concatenates the result with x coordinate-wise. Moreover, we define $TC_A^{(S)} : S^n \to (\mathbb{Z}^2)^n$ to be the TC_A when the input alphabet is reduced to only S.

6.1 Good Tree Codes and Triangular Totally k-non-singular Matrices

Pudlak [42] reduced the problem of constructing asymptotically good tree codes to constructing a lower triangular totally non-singular matrix A over any field $\mathbb{F}$ with polynomially many elements. Here, we show that in this case the condition of being totally nonsingular can be relaxed to k-totally nonsingular with proper k.

Theorem 6.6. *Let $k \in \mathbb{N}$ and $A \in \mathbb{N}^{n\times n}$ be a lower triangular totally $2k$-nonsingular matrix. Set $S = \{-k, -k+1, \dots, 0, \dots, k-1, k\}$, then $TC_A^{(S)}$ has distance greater than $1/2$.*

For every $x, y \in (\mathbb{N}^2)^n$ we define $\tilde{\Delta}(x,y)$ to be the hamming distance of x, y when considered as elements in $\mathbb{N}^{2n}$, and $\tilde{\delta}(x,y) = \tilde{\Delta}/2n$. We define another distance parameter for any tree code TC with output alphabet $\mathbb{N}^2$, $\tilde{\delta}_{TC} := \min(\tilde{\delta}_{[s,s+l]}(TC_A^{(S)}(x), TC_A^{(S)}(y)))$, where min is over all $l \in \mathbb{N}$ and $x, y \in S^n$ with $s = split(TC_A^{(S)}(x), TC_A^{(S)}(y))$ and $l \leq n-s$. It is easy to verify that $\delta_{TC} \geq \tilde{\delta}_{TC}$.

Proof. We prove that $TC_A^{(S)}$ has $\tilde{\delta} > 1/2$.

Since TC_A is linear, it is enough to show that for every $x \in \{-2k, -k+1, \dots, 0, \dots, k-1, 2k\}^n$ with the first non-zero index s, and every $l \in [n-s]$, where $TC_A^{(S)}(x) = y$, if we view $y_{[s,s+l]}$ as an string in $\mathbb{N}^2$, it has at most l zeros. To this end, lets define $C_{x,l}$ and $R_{x,l}$ as follows:

$$C_{x,l} := \{j \in [s+l] : x_j \neq 0\} = \{j_1, j_2, \dots, j_c\}$$
$$R_{x,l} := \{i \in [s, s+l] : (Ax)_i = 0\} = \{i_1, i_2, \dots, i_r\}$$

First we prove that $|R_{x,l}| < |C_{x,l}|$ and since the number of nonzero elements in $TC_A^{(S)}(x)_{[s,s+l]}$ (as a sequence in $\mathbb{N}^{2n}$) is exactly $|C_{x,l}| + (l - |R_{x,l}|)$, we conclude that number of nonzero elements in $TC_A^{(S)}(x)_{[s,s+l]}$ (when viewed as a sequence in $\mathbb{N}^2$) is at least l. Now, we prove that $|R_{x,l}| < |C_{x,l}|$. Assume the opposite is true, and let t be the largest index in $[r]$ such that $i_1 \geq j_1, \ldots, i_t \geq j_t$ and either $i_{t+1} < j_{t+1}$ or $t+1 = r+1$. Note that since $j_1 = s \leq i_1$, such index exists. Now, consider the submatrix $A_S = A[i_1, \ldots, i_t : j_1, \ldots, j_t]$. According to the definition of t, it's diagonal is not above the diagonal of A so should be $2k$-non-singular. However, $Ax_{[j_1,\ldots,j_t]} = 0$. This is a contradiction and concludes that $|R_{x,l}| < |C_{x,l}|$. □

6.2 Random Construction of Totally k-non-singular Matrices

Theorem 6.7. *Let $A \in \mathbb{N}^{n\times n}$ be a random lower triangular matrix that is constructed by putting 0 on entries above the diagonal, and other entries are chosen independently uniformly at random from $\{1, \ldots, T\}$. If $T > kn^3$, then with high probability, A is lower triangular totally k-non-singular.*

Proof of Theorem 6.7. Fix some $d \leq n$, $x \in \mathbb{N}^d$ with $|x_i| \leq k$, and fix $r_1 < r_2 < \cdots < r_d \in [n]$, and $c_1 < c_2 < \cdots < c_d \in [n]$ where $c_i \leq r_i$. It is easy to see that $\forall i \in [d] : Pr[A_{[r_i:c_1,\ldots,c_d]}x = 0] \leq \frac{1}{T}$, so $Pr[A_{[r_1,\ldots,r_d:c_1,\ldots,c_d]}x = 0] \leq (\frac{1}{T})^d$. Using the union bound over all $d < n, x \in \mathbb{N}^d$ with $|x_i| \leq k$, and all $r_1, \ldots, r_d, c_1, \ldots, c_d$ described as above, the probability that there exist a submatrix of A below the lower triangular that is k-non-singular is at most

$$\sum_{d=1}^{n} k^d (\frac{1}{T})^d \binom{n}{d}^2 \leq \sum_{d=1}^{n} (\frac{k}{kn^3})^d \binom{n}{d}^2 \leq \sum_{d=1}^{n} (\frac{1}{n^3})^d \times n^{2d} = \frac{1}{n-1} \to 0$$

□

Corollary 6.8. *There exist tree codes $TC_A : [n]^n \to [n^5]^n$ with distance $\delta > 1/2$ constructed by a matrix A satisfying conditions in Theorem 6.7 with $k = n$.*

In the next chapter, we discuss how assuming Conjecture 7.1 will give us an explicit construction of a binary tree code with positive distance and constant rate. Since the only difference of Corollary 6.8 with Conjecture 7.1 is that it's not explicit, Corollary 6.8 gives us a random construction of asymptotically good binary tree codes. Unlike all previous random constructions, the above method has vanishing failure probability. However, it has the same problem as all previous methods, no method is known to verify in polynomial time that the constructed structure builds a good-distanced tree code (Note that to exhaustively check that all submatrices are non-singular takes time $O(2^n)$.) We don't know if it is possible to check total non-singularity of a matrix in polynomial. A related topic to totally-non-singular matrices is totally (strictly) positive matrices, the class of all matrices that all their minors are (strictly) positive (totally positive and strictly totally positive triangular matrices are defined in the same way as totally non-singular matrices).

There is a rich literature in this topic, and it is proved that testing that a matrix is strictly totally positive (also triangular strictly totally positive) can be done in $O(n^2)$ time [26](see also [38] theorem 2.8). Strictly totally positive is a stronger condition than total non-singularity and is a sufficient property to build constant distance tree codes. However, all known integer strictly totally positive matrix structures have elements of exponential order which is not good enough if we want to have tree codes with constant alphabet size. In [26], they prove that a lower triangular matrix is strictly totally positive if and only if all minors of submatrices of consecutive leading columns and consecutive rows are positive. This reduces the complexity of verifying that a triangular matrix is strictly totally positive from $O(2^n)$ to $O(n^2)$. It is an interesting problem to find a similar test for totally non-singular matrices.

Question 6.9. *Given a lower triangular matrix, is it possible to check in poly time if it is a totally non-singular matrix?*

6.3 Random Construction of tree codes of polynomial alphabet size without using totally k-non-singular matrices

Building totally k-non-singular matrices are not the only way to generate random construction of good tree codes of polynomial alphabet size. In the following theorem we give another random construction of tree codes without using such matrices.

Theorem 6.10. *For every $n \in \mathbb{N}$ there exist an $n \times n$ lower-triangular matrix M with elements before the diagonal in range $\{1, 2, ..., n^9\}$ that is a generating matrix of code $c : \mathbb{Z}^n \to \mathbb{Z}^n$ with the following property:*

For every $z \in \{-2n, -2n+1, ..., 0, , ..., 2n\}^n$ with $s = min_i\{z_i \neq 0\}$, and every l with $s + l \leq n$ $\mathsf{weight}_{s,s+l}(Mz) \geq l/2$.

Proof of Theorem 6.10. Let M be a lower triangular $n \times n$ random matrix where the elements in row $i \leq n$ not above the main diagonal are chosen uniformly at random from $[n^9]$. Let c be the tree code with generating matrix M and let $\delta^* = min_{l \in [n-s], z \in \{-2n, -2n-1, ..., 0, , ..., 2n\}} \mathsf{weight}(Mz_{s,s+l})$, where s is the index of the first non-zero element in Mz. Let's calculate the probability that $\delta^* < 1/2$.

Fix $z \in \{-2n, ..., 0, ..., 2n\}^n$, and $s, l \in [n]$ such that $s + l \leq n$. The probability that $(Mz)_i = 0$ is at most $1/n^9$ for each $i \in [n]$. Let X_i be a Boolean random variable that indicates whether $Mz_i = 0$ or not, and let $X = \sum_{i=s}^{s+l} X_i$. The probability that each $X_i = 1$ is at most $1/n^9$ so the expected value of X is at most l/n^9. We use Chernoff bound to give an upper bound on the probability that at least $l/2$ of the elements of $Mz_{[s,s+l]}$ is 0.

$$
\begin{aligned}
Pr[X > l/2] &= Pr[X - l/n^9 > l/2 - l/n^9] \\
&= Pr[X - l/n^9 > l/n^9(n^9/2 - 1)] \\
&= Pr[X - \mu > \mu(n^9/2 - 1)] \\
&\leq (\frac{e^{n^9/2-1}}{(n^9/2)^{n^9/2}})^{\mu} \\
&\leq (\frac{e^{n^9/2}}{(n^9/2)^{n^9/2}})^{l/n^9} \\
&\leq (\frac{e}{(n^9/2)})^{l/2} \\
&\leq \frac{(2e)^{l/2}}{(n^{4.5})^{l}} \\
&\leq \frac{1}{n^{4l}}
\end{aligned}
$$

Using union bound over s, l and z's, the probability that $\delta^* < l/2$ is at most

$$
\sum_{s=1}^{n} \sum_{l=1}^{n-s} \frac{1}{n^{4l}} \times n^l = \sum_{s=1}^{n} \sum_{l=1}^{n-s} \frac{1}{n^{3l}} \leq \frac{n^2}{n^3} = \frac{1}{n} \to 0
$$

□

Chapter 7

Alphabet Reduction Machine

In this section, we show that how assuming Conjecture 7.1 will give us a binary tree code with constant alphabet and constant distance (Theorem 7.3). The proof here is very similar to the proof in [18] but with different parameters.

Conjecture 7.1. *For every constant $\delta \geq 0$ there exist a function $p : \mathbb{Z} \to \mathbb{Z}$ such that $p(n) = O(log(n))$ and an explicit tree code $TC_{\mathbb{Z}} : \mathbb{Z}^{\mathbb{N}} \to \mathbb{Z}^{\mathbb{N}}$ with distance δ, such that $\forall t \in \mathbb{N}$ and $z \in \mathbb{Z}^{\mathbb{N}}, |TC_{\mathbb{Z}}(z)_t| \leq 2^{p(t)} \cdot max(z_0^2, ..., z_t^2)$*

Corollary 7.2. *For every integer $n \geq 1$ there exist a function $p : \mathbb{Z} \to \mathbb{Z}$ such that $p(n) = O(log(n))$ and an explicit tree code $TC_{p(n)} : (\{0,1\}^{p(n)})^n \to (\{0,1\}^{3p(n)})^n$ with distance $1/2$.*

Proof of Corollary 7.2. Let $TC_{\mathbb{Z}} : \mathbb{Z}^{\mathbb{N}} \to \mathbb{Z}^{\mathbb{N}}$ be a tree code from Conjecture 7.1 with distance δ. If we bound the domain to $(\{0,1\}^{p(n)})^n$, we will have each $z_i^2 \leq 2^{2p(n)}$, and since $t \leq n$, we will have $|TC_{\mathbb{Z}}(z)_t| \leq 2^{p(n)} \times 2^{2p(n)}$ which is equivalent to the range being $(\{0,1\}^{3p(n)})^n$, so the restriction of $TC_{\mathbb{Z}}$ to the domain $(\{0,1\}^{p(n)})^n$ gives the function $TC_l : (\{0,1\}^{p(n)})^n \to (\{0,1\}^{3p(n)})^n$ with distance $1/2$. □

$0^{c_2 p(n)}$	$ECC(TC_{p(n)}(m)_1)$	$\cdots$	$ECC(TC_{p(n)}(m)_{l-1})$
$TC_e(m_1))_{[1,p(n)]}$	$TC_e(m_2))_{[1,p(n)]}$	$\cdots$	$TC_e(m_l))_{[1,p(n)]}$
$0^{c_1 p(n)}$	$TC_e(m_1 m_2))_{[p(n),2p(n)]}$	$\cdots$	$TC_e(m_{l-1} m_l))_{[p(n),2p(n)]}$

Figure 7.1: $TC(m)$. The number of blocks is $l = n/p(n)$.

Theorem 7.3. *Assuming conjecture Conjecture 7.1, there exist a constant $c \in \mathbb{N}$ and an algorithm A, where for every $n \in \mathbb{N}$ algorithm A computes a constant-distanced tree code $TC : \{0,1\}^n \to [c]^n$ in polynomial time.*

Proof of Theorem 7.3. Let $p : \mathbb{Z} \to \mathbb{Z}$ be the function in Conjecture 7.1. To be able to use the tree code in Corollary 7.2, we partition every $m \in \{0,1\}^n$ to blocks of length $p(n)$ in other words, we write it as $m = (m_1, ..., m_{n/p(n)})$ where $m_i \in \{0,1\}^{p(n)}$. The idea is to apply tree code of Corollary 7.2 to m, interpreted as an element of $(\{0,1\}^{p(n)})^{n/p(n)}$. However, this will just guarantee distance of the the code as elements of $(\{0,1\}^{3p(n)})^{n/p(n)}$ and that only means there is enough different blocks in the code of two different strings, not enough different bits. To overcome this issue, we also use an error correcting block code on each block of the output code to guarantee distance between corresponding blocks that are different. We also use tree codes with constant distance and alphabet of length $p(n)$ to gaurantee distance within a block. So the building blocks in the construction of TC are the following:

1. let $TC_{p(n)} : (\{0,1\}^{p(n)})^{n/p(n)} \to (\{0,1\}^{3p(n)})^{n/p(n)}$ be the tree code from Corollary 7.2. Recall that $TC_{p(n)}$ has distance 1/2.
2. Let $TC_e : (\{0,1\})^{2p(n)} \to (\{0,1\}^{c_1})^{2p(n)}$ be a tree code with distance 1/2 that we find using brute force algorithm (Note that since $p(n)$ is of order $log(n)$ the brute force takes polynomial time.)
3. Let $ECC : \{0,1\}^{3p(n)} \to (\{0,1\}^{c_2})^{p(n)}$ be an error correcting block code with distance 5/6. By Lemma 3.2 in [18], c_2 is constant.

We define $TC(m) = (tc_1^{(3)}, tc_2^{(3)}, ..., tc_n^{(3)})$, where $tc_1^{(3)} = (\{0\}^{c_2 p(n)}, TC_e(m_1), \{0\}^{c_1 p(n)})$, and for $i \in \{2, ..., n/p(n)\}$:

$$ec_i = (ECC(TC_{p(n)}(m)_{i-1})) \tag{7.1}$$

$$tc_i^{(1)} = (TC_e(m_i))_{[1,p(n)]}) \tag{7.2}$$

$$tc_i^{(2)} = (TC_e(m_{i-1}m_i))_{[p(n),2p(n)]}) \tag{7.3}$$

$$tc_i^{(3)} = (ec_i, tc_i^1, tc_i^2) \tag{7.4}$$

To analyze the distance, suppose $x, y \in \{0,1\}^n$ and $s = split(x,y)$, $d \leq n - s$. Let S be the block that the split occurs, and t be the index of the split in that block. i.e. $S = \lceil s/p(n) \rceil$, and t is the remainder of s mod $p(n)$. Also, let $d_1 = min(d, p(n) - t)$, $d_2 = \lfloor \frac{d-d_1}{p(n)} \rfloor$, and $d_3 = d - d_1 - d_2 p(n)$. Note that $dist(x,y)_{|s,s+d_1]} \leq \frac{d_1}{2}$ due to $tc_S^{(1)}$, and $dist(x,y)_{[Sp(n),(S+d_2)p(n)]} \leq \frac{d_2}{2} \times \frac{5p(n)}{6} = \frac{5d_2 p(n)}{12}$ due to $TC_{p(n)}$ and ECC. Suppose $d_2 > 0$, since $d_3 \leq p(n)$, $d = d_1 + d_2 p(n) + d_3 \leq 2d_1 + 2d_2 p(n)$, so

$$dist(x,y)_{s,s+d} \geq \frac{d_1}{2} + \frac{5d_2 p(n)}{12} \geq \frac{5(d_1 + d_2 p(n))}{12} \geq \frac{5}{12} \times \frac{d}{2} = \frac{5d}{24} \tag{7.5}$$

And if $d_2 = 0$, we have $dist(x, y)_{[s,s+d_1+d_3]} \geq \frac{d_1+d_3}{2} = \frac{d}{2}$ due to $tc^{(2)}$, so in any case $dist(x, y)_{s,s+d} \geq \frac{5d}{24}$. □

Chapter 8

Open Problems and Future Work

- Is there any explicit Binary tree code with constant distance and constant alphabet size?
- Is there any polynomial-time test for checking if a lower-triangular matrix is totally-non-singular? Is this problem NP-Complete? Note that if there is a polynomial-time test for checking if a lower-triangular matrix is totally-non-singular, then we could construct good tree codes by taking a random lower-triangular matrix and verifying that it is indeed totally-non-singular.
- Is there any $n \times n$ lower triangular totally positive matrix with elements of order $O(poly(n))$? Is there any random algorithm for generating them? Such randomized algorithm can be useful since checking if a matrix is totally positive can be done in polynomial time.

Bibliography

[1] L. Babai and V. Sós. Sidon sets in groups and induced subgraphs of cayley graphs. *Eur. J. Comb.*, 6:101–114, 1985.

[2] William Baird and Anthony Bonato. Meyniel's conjecture on the cop number: A survey. *Journal of Combinatorics*, 3(2):225–238, 2012.

[3] R. Baker, G. Harman, and J. Pintz. The difference between consecutive primes, ii. *Proceedings of the London Mathematical Society. Third Series*, 83:532–562, 2001.

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

[5] B. Bollobás, G Kun, and I. Leader. Cops and robbers in a random graph. *Journal of Combinatorial Theory, Series B*, 103(2):226–236, 2013.

[6] A. Bonato, P. Prałat, and C. Wang. Pursuit-evasion in models of complex networks. *Internet Mathematics*, 4(4):419–436, 2007.

[7] A. Bonato, P. Prałat, and C. Wang. Vertex pursuit games in stochastic network models. In *Combinatorial and Algorithmic Aspects of Networking*, pages 46–56, 2007.

[8] Anthony Bonato and Andrea Burgess. Cops and robbers on graphs based on designs. *Journal of Combinatorial Designs*, 21(9):404–418, 2013.

[9] Anthony Bonato and Ehsan Chiniforooshan. Pursuit and evasion from a distance: algorithms and bounds. In *2009 Proceedings of the Workshop on Analytic Algorithmics and Combinatorics (ANALCO)*, 2009.

[10] Anthony Bonato and Richard Nowakowski. *The Game of Cops and Robbers on Graphs*. American Mathematical Society, 09 2011.

[11] Nathan Bowler, Joshua Erde, Florian Lehner, and Max Pitz. Bounding the cop number of a graph by its genus. 2019.

[12] Peter Bradshaw. A proof of the meyniel conjecture for abelian cayley graphs. *Discrete Mathematics*, 343(1):111546, 2020.

[13] Peter Bradshaw, Seyyed Aliasghar Hosseini, and Jérémie Turcotte. Cops and robbers on directed and undirected abelian cayley graphs. *European Journal of Combinatorics*, 97:103383, 2021.

[14] Mark Braverman. Towards deterministic tree code constructions. *Electron. Colloquium Comput. Complex.*, 18:64, 2011.

[15] Mark Braverman and Anup Rao. Toward coding for maximum errors in interactive communication. *IEEE Transactions on Information Theory*, 60:7248–7255, 2014.

[16] Boris Bukh. Rank arguments, 2014. Lecture notes on Algebraic Methods in Combinatorics.

[17] Nancy E. Clarke and Gary MacGillivray. Characterizations of k-copwin graphs. *Discrete Mathematics*, 312(8):1421–1425, 2012.

[18] Gil Cohen, Bernhard Haeupler, and Leonard J. Schulman. Explicit binary tree codes with polylogarithmic size alphabet. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2018, page 535–544, New York, NY, USA, 2018. Association for Computing Machinery.

[19] Gil Cohen and Shahar Samocha. Palette-alternating tree codes. In *Proceedings of the 35th Computational Complexity Conference*, CCC '20, Dagstuhl, DEU, 2020. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[20] W. Evans, M. Klugerman, and Schulman L. J. Postscript of 21 september 2003 to coding for interactive communication. http://users.cms.caltech.edu/ schulman/Papers/intercodingpostscript.txt.

[21] Fedor Fomin, Petr Golovach, Jan Kratochvíl, N. Nisse, and Karol Suchan. Pursuing a fast robber on a graph. *Theoretical Computer Science*, 411:1167–1181, 02 2010.

[22] P. Frankl. Cops and robbers in graphs with large girth and cayley graphs. *Discrete Applied Mathematics*, 17:301–305, 1987.

[23] P. Frankl. On a pursuit game on cayley graphs. *Combinatorica*, 7(1):67–70, 1987.

[24] Alan Frieze, Michael Krivelevich, and Po-Shen Loh. Variations on cops and robbers. *Journal of Graph Theory*, 69(4):383–402, 2012.

[25] M. Fromme and M. Aigner. A game of cops and robbers. *Discrete Appl. Math*, 8:1–12, 1984.

[26] M. Gasca and J. Peña. Total positivity and neville elimination. *Linear Algebra and its Applications*, 165:25–44, 1992.

[27] Seyyed Aliasghar Hosseini, Fiachra Knox, and Bojan Mohar. Cops and robbers on graphs of bounded diameter. *SIAM Journal on Discrete Mathematics*, 34(2):1375–1384, Jan 2020.

[28] William B. Kinnersley. Cops and robbers is exptime-complete. *Journal of Combinatorial Theory, Series B*, 111:201–220, Mar 2015.

[29] János Kollár, Lajos Rónyai, and Tibor Szabó. Norm-graphs and bipartite turán numbers. *Combinatorica*, 16(3):399–406, 1996.

[30] T. Kövári, V. T. Sós, and P. Turán. On a problem of Zarankiewicz. *Colloquium Mathematicae*, 3:50–57, 1954.

[31] Linyuan Lu and Xing Peng. On meyniel's conjecture of the cop number. *Journal of Graph Theory*, 71(2):192–205, 2012.

[32] Tomasz Łuczak and Paweł Prałat. Chasing robbers on random graphs: Zigzag theorem. *Random Structures & Algorithms*, 37(4):516–524, 2010.

[33] Cristopher Moore and Leonard J. Schulman. Tree codes and a conjecture on exponential sums. In *Proceedings of the 5th Conference on Innovations in Theoretical Computer Science*, ITCS '14, page 145–154, 2014.

[34] Anand Kumar Narayanan and Matthew Weidner. On decoding cohen-haeupler-schulman tree codes. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, SODA 20, pages 1337–1356. SIAM, 2020.

[35] R. Nowakowski and P. Winkler. Vertex-to-vertex pursuit in a graph. *Discrete Mathematics*, 43(2-3):235–239, 1983.

[36] Kevin O'Bryant. A complete annotated bibliography of work related to sidon sequences. *Electron J Comb*, 11, 08 2004.

[37] Marcin Peczarski. An improvement of the tree code construction. *Inf. Process. Lett.*, 99(3):92–95, 2006.

[38] A. Pinkus and P. Allan. *Totally Positive Matrices*. Cambridge Tracts in Mathematics. Cambridge University Press, 2010.

[39] Pawel Pralat. When does a random graph have constant cop number? *Australasian J. Combinatorics*, 46:285–296, 2010.

[40] Paweł Prałat and Nicholas Wormald. Meyniel's conjecture holds for random graphs. *Random Structures & Algorithms*, 48(2):396–421, 2016.

[41] Paweł Prałat and Nicholas Wormald. Meyniel's conjecture holds for random d-regular graphs. *Random Structures & Algorithms*, 55(3):719–741, 2019.

[42] Pavel Pudlák. Linear tree codes and the problem of explicit constructions. *Linear Algebra and its Applications*, 490, 10 2013.

[43] A. Quilliot. Problemes de jeux, de point fixe, de connectivité et de représentation sur des graphes, des ensembles ordonnés et des hypergraphes. *These d'Etat, Université de Paris VI*, pages 131–145, 1983.

[44] A. Quilliot. A short note about pursuit games played on a graph with a given genus. *Journal of Combinatorial Theory, Series B*, 38(1):89–92, 1985.

[45] Leonard Schulman. Coding for interactive communication. *Information Theory, IEEE Transactions on*, 42:1745 – 1756, 12 1996.

[46] Leonard J. Schulman. Deterministic coding for interactive communication. In *Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing*, STOC 93, page 747–756, New York, NY, USA, 1993. Association for Computing Machinery.

[47] A. Scott and B. Sudakov. A bound for the cops and robbers problem. *SIAM Journal on Discrete Mathematics*, 25(3):1438–1442, 2011.

[48] Daniel A. Speilman. *Spectral and Algebraic Graph Theory*. 2019.

[49] Terence Tao. Algebraic combinatorial geometry: the polynomial method in arithmetic combinatorics, incidence combinatorics, and number theory. *EMS Surveys in Mathematical Sciences*, 1, 10 2013.

[50] Zsolt Adam Wagner. Cops and robbers on diameter two graphs. *Discrete Mathematics*, 338(3):107–109, Mar 2015.

[51] Inbar Ben Yaacov, Gil Cohen, and Anand Kumar Narayanan. Candidate tree codes via pascal determinant cubes. *Electron. Colloquium Comput. Complex.*, 27:141, 2020.