TUCS

Tuomo Lehtilä

On Location, Domination and Information Retrieval

Turku Centre *for* Computer Science

TUCS Dissertations
No 257, September 2020

On location, domination and information retrieval

Tuomo Lehtilä

To be presented, with the permission of the Faculty of Science and Engineering of the University of Turku, for public criticism in Arcanum 1 on September 25, 2020, at 12 noon.

University of Turku
Department of Mathematics and Statistics
FI-20014 Turku

2020

Supervisors

Professor Tero Laihonen
Department of Mathematics and Statistics
University of Turku
Finland

Docent Ville Junnila
Department of Mathematics and Statistics
University of Turku
Finland

Reviewers

Doctor Florent Foucaud
LaBRI
Université de Bordeaux
France

Professor Olivier Hudry
LTCI, Télécom ParisTech
Université Paris-Saclay
France

Opponent

Associate Professor Ismael Yero
Department of Mathematics
Universidad de Cádiz
Spain

The originality of this thesis has been checked in accordance with the University of Turku quality assurance system using the *Turnitin OriginalityCheck* service.

ISBN 978-952-12-3981-6
ISSN 1239-1883

Abstract

The thesis is divided into two main branches: identifying and locating-dominating codes, and information retrieval. The former topics are motivated by the aim to locate objects in sensor networks (or other similar applications) and the latter one by the need to retrieve information in memories such as DNA data storage systems. Albeit the underlying applications, the study on these topics mainly belongs to discrete mathematics; more specifically, to the fields of coding and graph theory.

The sensor networks are usually represented by graphs where vertices represent the monitored locations and edges the connections between the locations. Moreover, the locations of the sensors are determined by a code. Furthermore, the desired properties of the sensor network are deeply linked with the properties of the underlying code.

The number of errors in reading the data is abundant in the DNA data storage systems. In particular, there can occur more errors than a reasonable error-correcting code can handle. However, this problem is somewhat offset by the possibility to obtain multiple approximations of the same information from the data storage. Hence, the information retrieval process can be modelled by the Levenshtein's channel model, where a message is sent through multiple noisy channels and multiple outputs are received.

In the first two papers of the thesis, we introduce and study the new concepts of self- and solid-locating-dominating codes as a natural analogy to self-identifying codes with respect to locating-dominating codes. The first paper introduces these new codes and considers them in some graphs such as the Hamming graphs. Then, in the second paper, we broaden our view on the topic by considering graph theoretical questions. We give optimal codes in multiple different graph classes and some more general results using concepts such as the Dilworth number and graph complements. The third paper focuses on the q-ary Hamming spaces. In particular, we disprove a conjecture proposed by Goddard and Wash related to identifying codes. In the fourth paper, we return to self- and solid-locating-dominating codes and give optimal codes in some graph classes and consider their densities in infinite graphs.

In the fifth paper, we consider information retrieval in memories; in particular, the Levenshtein's channel model. In the channel model, we transmit some codeword belonging to the binary Hamming space through multiple identical channels. With the help of multiple different outputs, we give a list of codewords which may have been sent. In the paper, we study the number of channels required to have a rather small (constant) list size when the properties of the channels, the code and the dimension of the Hamming space are fixed. In particular, we give an exact relation between the number of channels and the asymptotic value of the maximum list size.

Tiivistelmä

Väitöskirja käsittelee kahta aihetta: identifioivia ja paikantavia peittokoodeja sekä tiedon noutamista muistista. Ensimmäisen aiheen motivaationa on objektien paikantaminen sensoriverkoista (sekä muut samankaltaiset sovellukset) ja jälkimmäisen tiedonnouto DNA-muisteista. Näiden aiheiden tutkimus kuuluu diskreettiin matematiikkaan, täsmällisemmin koodaus- ja graafiteoriaan.

Sensoriverkkoja kuvataan yleensä graafeilla, joissa solmut esittävät tarkkailtuja kohteita ja viivat yhteyksiä näiden kohteiden välillä. Edelleen sensorien paikat määräytyvät annetun koodin perusteella. Tästä johtuen sensoriverkon halutut ominaisuudet pohjautuvat vahvasti alla olevaan koodiin.

Luettaessa tietoa DNA-muisteista tapahtuvien virheiden määrä saattaa olla erittäin suuri; erityisesti suurempi kuin kiinnitetyn virheitä korjaavan koodin korjauskyky. Toisaalta tilanne ei ole aivan näin ongelmallinen, sillä DNA-muisteista voidaan saada useita eri arvioita muistiin tallennetusta tiedosta. Näistä syistä johtuen tietojen noutamista DNA-muisteista voidaan mallintaa käyttäen Levenshteinin kanavamallia. Kanavamallissa yksi viesti lähetetään useiden häiriöisten kanavien kautta ja näin vastaanotetaan useita viestejä (yksi jokaisesta kanavasta).

Väitöskirjan kahdessa ensimmäisessä julkaisussa esitellään ja tutkitaan uusia paikantavien peittokoodien luokkia, jotka pohjautuvat aiemmin tutkittuihin itse-identifioiviin koodeihin. Ensimmäisessä julkaisussa on esitelty nämä koodiluokat sekä tutkittu niitä joissain graafeissa kuten Hammingin graafeissa. Tämän jälkeen toisessa julkaisussa käsitellään yleisiä graafiteoreettisia kysymyksiä. Julkaisussa esitetään optimaaliset koodit useille graafiperheille sekä joitain yleisempiä tuloksia käyttäen mm. Dilworthin lukua sekä graafikomplementteja. Kolmas julkaisu keskittyy q-arisiin Hammingin avaruuksiin. Erityisesti julkaisussa todistetaan vääräksi Goddardin ja Washin aiemmin esittämä identifioivia koodeja koskeva otaksuma. Neljäs artikkeli käsittelee jo kahdessa ensimmäisessä artikkelissa esiteltyjä paikantavien peittokoodien luokkia. Artikkeli esittää optimaalisia koodeja useille graafiperheille sekä käsittelee äärettömiä graafeja.

Viides artikkeli käsittelee tiedonnoutoa ja erityisesti Levenshteinin kanavamallia. Kanavamallissa binääriseen Hammingin avaruuteen kuuluva koodisana lähetetään useiden identtisten kanavien läpi. Näistä kanavista vastaanotetaan useita eri arvioita lähetetystä koodisanasta ja rakennetaan lista mahdollisesti lähetetyistä sanoista. Artikkelissa tutkitaan kuinka monta kanavaa tarvitaan, jotta tämän listan koko on pieni (vakio), kun kanavien ominaisuudet, koodi ja Hammingin avaruuden dimensio on kiinnitetty. Erityisesti löydetään täsmällinen suhde kanavien lukumäärän ja asymptoottisesti maksimaalisen listan koon välille.

Acknowledgements

First of all, I want to thank my supervisors Professor Tero Laihonen and Docent Ville Junnila for introducing me to the subjects of location and information retrieval. They have presented many questions for me to study, had discussions about the topics and helped me to present my ideas in a much clearer way while also being nice and efficient collaborators.

I would also like to thank my co-author María Luz Puertas Gonzalez for co-operation and innovative ideas. Besides her I extend my gratitudes to the pre-examiners of this thesis Doctor Florent Foucaud and Professor Olivier Hudry, and also to Associate Professor Ismael Yero for agreeing to act as the opponent.

I would also like to thank University of Turku, Turku Centre for Computer Science, Väisälä Fund for supporting the third year of research, the Jenny and Antti Wihuri foundation for supporting me during the fourth year of my research, and the doctoral program MATTI for the financial support for conference costs as well as Doctor Mikhail Barash for technical assistance.

I am also grateful for the nice atmosphere at mathematics department. Besides that I am thankful for my office mates, Markus and Jarkko for guidance at the start and later Esa, Juho, Teemu and Toni for company.

Finally, I want to thank Emilia and my family for all the support they have offered me.

Turku, August 2020 Tuomo Lehtilä

List of original publications

I. Ville Junnila, Tero Laihonen and Tuomo Lehtilä. On regular and new types of codes for location-domination, *Discrete Applied Mathematics*, 247:225–241, 2018.

   https://doi.org/10.1016/j.dam.2018.03.050

II. Ville Junnila, Tero Laihonen, Tuomo Lehtilä and María Luz Puertas. On stronger types of locating-dominating codes, *Discrete Mathematics and Theoretical Computer Science*, 21, 2019.

III. Ville Junnila, Tero Laihonen and Tuomo Lehtilä. On a conjecture regarding identification in Hamming graphs, *The Electronic Journal of Combinatorics*, P2, 2019.

IV. Ville Junnila, Tero Laihonen and Tuomo Lehtilä. New optimal results on codes for location in graphs, *Fundamenta Informaticae*, submitted for publication, 2020.

V. Ville Junnila, Tero Laihonen and Tuomo Lehtilä. On Levenshtein's channel and list size in information retrieval, *IEEE Transactions on Information Theory*, accepted for publication, 2020.

   https://doi.org/10.1109/TIT.2020.3016269

Contents

Part I

Summary

1 Introduction

There exists a vast number of different covering and domination problems. These problems have been widely studied and they can be applied on multiple fields such as: linear algebra and optimization, design and analysis of communication networks, social sciences, bioinformatics, computational complexity, and algorithm design [28, Preface]. In this thesis, we concentrate on two separate but somewhat connected topics: dominating codes in graphs with some additional capabilities to locate vertices and Levenshtein's channel model where based on a large number of approximations we try to deduce what has been sent through multiple noisy channels. We may interpret Levenshtein's channel model also as trying to locate a vertex which is in the proximity of a large but somewhat random set of vertices.

1.1 Domination

Basics

We study undirected simple graphs $G = (V, E)$ on at least two vertices with vertex set V and edge set E. A non-empty subset of the vertex set V is called a *code* and an element of a code is a *codeword*. A *path* $P_n = (V, E)$ is a graph with vertex set $V = \{v_1, \dots, v_n\}$ and edge set $E = \{v_i v_{i+1} \mid 1 \leq i \leq n-1\}$. The *graphical distance*, denoted by $d(u, v)$, between two vertices $u, v \in V$ is equal to the number of edges in any shortest path from u to v. The *open neighbourhood* of the vertex $v \in V$ is denoted by $N_G(v)$, or by $N(v)$ if the context is clear, and it consists of a set of vertices adjacent to v. The *closed neighbourhood* of vertex v is $N_G[v] = N[v] = N(v) \cup \{v\}$. A *ball* of radius r centred at vertex v is denoted by $B_r(v)$ and consists of vertices at distance at most r from v, that is, $B_r(v) = \{u \in V \mid d(v, u) \leq r\}$. The *identifying set* of a vertex v is the set of codewords in the neighbourhood of the vertex v

$$I(G, C; v) = I(v) = C \cap N_G[v]$$

and, for $r \geq 2$, $I_r(G, C; v) = I_r(v) = C \cap B_r(v)$. We generalize these concepts for sets of vertices in a natural manner, that is, for $U \subseteq V$ we have

$$N(U) = \bigcup_{v \in U} N(v)$$

and similarly, we have $N[U] = \bigcup_{v \in U} N[v]$, $I(U) = \bigcup_{v \in U} I(v)$ and $I_r(U) = \bigcup_{v \in U} I_r(v)$. A graph $\overline{G} = (V, E')$ is the *complement* of $G = (V, E)$ if $e \in E'$ if and only if $e \notin E$.

The codes studied in this thesis are mostly dominating codes with some additional requirements. A code $C \subseteq V$ is a *dominating* (2-*dominating*) code (or set) if we have $|I(v)| \geq 1$ (resp. $|I(v)| \geq 2$) for each non-codeword

$v \in V \setminus C$. These abovementioned additional requirements on dominating codes have mostly to do with capabilities of the code to somehow identify a vertex based on its I-set.

A dominating code $C \subseteq V$ is an *identifying* code if we have for every pair of distinct vertices $v, u \in V$:

$$I(v) \neq I(u).$$

Identifying codes have been first introduced by Karpovsky, Chakrabarty and Levitin in [41]. Their basic idea is that every I-set is unique and non-empty and thus, we can identify the vertex just by comparing the I-sets. For example, in [50], identifying codes are applied to locate anomalies in sensor networks as follows: A graph describes some sensor network where vertices are locations and edges inform us about connections between these locations. Now the codewords in the code describe where the sensors are. Now those sensors have an ability to send an alarm if there is an anomaly in the same vertex or in a vertex adjacent to them. However, the alarm does not tell anything else than the existence of the anomaly in the closed neighbourhood. Now, if the sensors are placed at vertices belonging to an identifying code, then we immediately know where the anomaly locates when we receive the information from all of the alarming sensors.

Let us consider, for example, the graph G in Figure 2, where the darkened vertices form the code C. Now, $I(1) = \{1, 2\}$, $I(2) = \{1, 2, 3\}$, $I(3) = \{2, 3\}$, $I(u) = \{1, 3\}$ and $I(v) = \{3\}$. Since each of these sets is unique and non-empty, the code C is identifying.

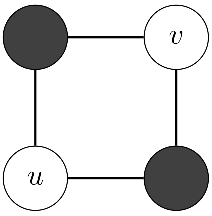


Figure 1: Darkened vertices form a 2-dominating code.

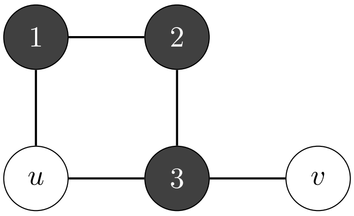


Figure 2: Darkened vertices form an identifying code.

A dominating code $C \subseteq V$ is a *locating-dominating* code if we have for every pair of distinct non-codewords $v, u \in V \setminus C$:

$$I(v) \neq I(u).$$

Locating-dominating codes have been originally studied in the 1980s by Slater [48, 54, 55]. The difference between them and identifying codes is that when comparing I-sets, we do not consider the I-sets of codewords. Thus, it is clear that every identifying code is also a locating-dominating code. Let us consider again the sensor network example as in the case of identifying codes. We notice that to convert the example for locating-dominating codes, we need the additional assumption that each sensor can sense whether the anomaly is in the same vertex with it and can communicate that information (see [56]). Now, if the sensors are located at positions which form a locating-dominating code, then we either know that the anomaly is at a vertex where the sensor is or we know its location by comparing the I-sets. Notice that we may now require less sensors than in the case of identifying codes but we require that sensor have more abilities.

The following definition of self-identifying codes is due to Junnila and Laihonen [38]. A dominating code $C \subseteq V$ is *self-identifying* in G if for any vertex $v \in V$ we have:

$$\bigcap_{c \in I(v)} N[c] = \{v\}. \tag{1}$$

Moreover, in [38, Theorem 6] an equivalent condition has been given. A code $C \subseteq V$ is a self-identifying code in G if for any pair of distinct vertices $u, v \in V$ we have

$$I(u) \setminus I(v) \neq \emptyset. \tag{2}$$

Furthermore, earlier in [32] by Honkala and Laihonen, a code $C \subseteq V$ has been defined as $(r, \leq l)^+$-*identifying code* if for $r > 0$, any two sets $U, U' \subseteq V$, $U \neq U'$ having

$$I_r(U) = I_r(U')$$

implies that $|U|, |U'| \geq l + 1$.

Moreover, in [38], $(1, \leq 1)^+$-identifying codes have been shown to be equivalent with self-identifying codes. From here on, we will call them self-identifying codes. In Figure 3, the darkened vertices form a self-identifying code. We may use Definition (1) to verify this. For example,

$$\bigcap_{c' \in I(a)} N[c'] = N[1] \cap N[3] \cap N[6] = \{a\}$$

and

$$\bigcap_{c' \in I(1)} N[c'] = N[1] \cap N[2] \cap N[6] = \{1\}.$$

Let us again return to the sensor network example. Self-identifying codes have two benefits over identifying codes. First of all, since $\bigcap_{c \in I(v)} N[c] = \{v\}$, we do not have to compare I-sets to identify a vertex. Besides that if we have two or more anomalies in our sensor network, then identifying codes may give an incorrect vertex since we may have $I(v) \subseteq I(u)$ and then $I(\{v, u\}) = I(u)$. For example, consider the graph of Figure 2 and vertices u and 2. Now, $I(u) \subseteq I(2)$ and $I(2) = I(\{2, u\}) = \{1, 2, 3\}$. However, if the sensors locate at codewords of a self-identifying code, then $I(v) \neq I(U)$ for

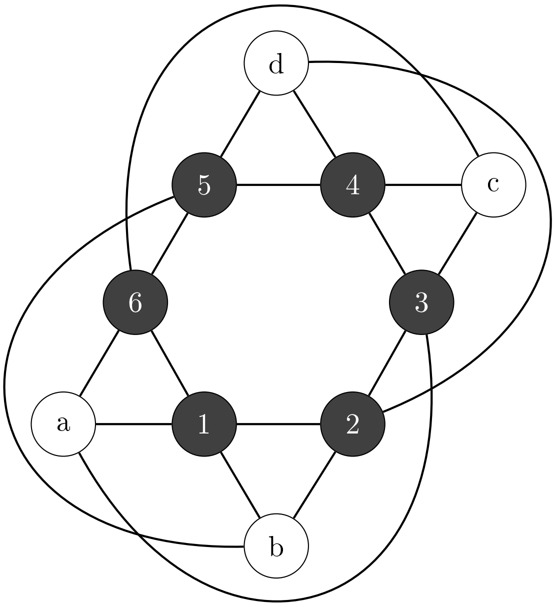


Figure 3: Darkened vertices form a self-identifying code.

any set of vertices $U \neq \{v\}$ and hence, we know the existence of multiple anomalies. For example, if there is an anomaly in vertices a and 1 in the graph of Figure 3, then we have

$$\bigcap_{c' \in I(\{a,1\})} N[c'] = N[1] \cap N[2] \cap N[3] \cap N[6] = \emptyset.$$

A dominating code $C \subseteq V$ has been defined as *self-locating-dominating* in G in Paper I if for every vertex $u \in V \setminus C$ we have

$$\bigcap_{c \in I(u)} N[c] = \{u\}.$$

The definition of self-locating-dominating codes is inspired by the corresponding definition of self-identifying codes (1). However, interestingly, if we consider the other equivalent definition for self-identifying codes (2), then we do not get a definition equivalent with the self-locating-dominating codes. A code $C \subseteq V$ is defined as *solid-locating-dominating* in G in Paper I if for each pair of vertices $u, v \in V \setminus C$ we have

$$I(u) \setminus I(v) \neq \emptyset.$$

In Figure 4, we have illustrated a self-locating-dominating code. The code is self-locating-dominating, since $I(v) = \{2, 3, u\}$ and $N[2] \cap N[3] \cap N[u] = \{v\}$. Moreover, u has to be a codeword in any self-locating-dominating code since otherwise $\{2, u\} \subseteq \bigcap_{c \in N(u)} N[c]$ and hence, $\{2, u\} \subseteq \bigcap_{c \in I(u)} N[c]$. In Figure 5, we have illustrated a solid-locating-dominating code in the same graph. We have $I(v) = \{2, 3\}$ and $I(u) = \{1\}$. Moreover, $I(v) \setminus I(u) = \{2, 3\}$ and $I(u) \setminus I(v) = \{1\}$. Hence, the code is solid-locating-dominating. Observe that u is not in the solid-locating-dominating code, but it must be in every self-locating-dominating code. Therefore, there are differences between these classes of codes. Notice that solid-locating-dominating codes are also dominating as their name suggests. To realize their differences, we have given the following characterization in Paper I, Theorem 8:

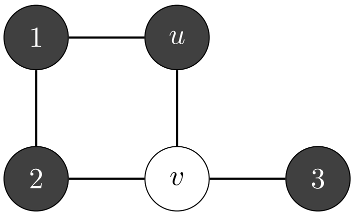


Figure 4: Darkened vertices form a self-locating-dominating code.

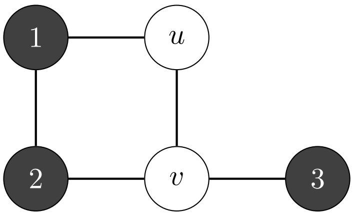


Figure 5: Darkened vertices form a solid-locating-dominating code.

Theorem 1. *A code $C \subseteq V$ is self-locating-dominating if and only if for all distinct vertices $u \in V \setminus C$ and $v \in V$ we have*

$$I(u) \setminus I(v) \neq \emptyset$$

and a code $C' \subseteq V$ is solid-locating-dominating if and only if for all $u \in V \setminus C'$ we have $I(u) \neq \emptyset$ and

$$\left(\bigcap_{c \in I(u)} N[c] \right) \setminus C' = \{u\}.$$

As we can see from these definitions, every self-locating-dominating code is also a solid-locating-dominating code. Let us again consider the sensor network example in the case of locating-dominating codes where each sensor can also signal whether there is an anomaly in the same vertex with it. Similarly, as in the case of self-identifying codes, we do not have to compare I-sets of different vertices in order to locate the anomaly unlike in the case of regular location-domination (see Paper I). Furthermore, we show in Paper I that these codes have also other beneficial properties compared to regular locating-dominating codes. In particular, in the case of solid-locating-dominating codes, we know if there are multiple anomalies as long as they do not occur at codewords. Moreover, in the case of self-locating-dominating codes, we observe the existence of multiple anomalies even when they happen at codeword vertices.

Earlier, in [33] by Honkala, Laihonen and Ranto, locating-dominating codes which locate up to l anomalies have been studied. Although these codes are only briefly mentioned in this thesis, they help to form a bigger picture of the topic. A code $C \subseteq V$ is an *$(r, \leq l)$-locating-dominating code of type A* in G if for $r > 0$ and for every vertex set $X, Y \subseteq V$ of size at most l the two conditions

$$I_r(X) = I_r(Y)$$

and

$$X \cap C = Y \cap C$$

together imply that $X = Y$.

A code $C \subseteq V$ is an *$(r, \leq l)$-locating-dominating code of type B* in G if for $r > 0$ and for every vertex set $X, Y \subseteq V \setminus C$ of size at most l the condition

$$I_r(X) = I_r(Y)$$

implies that $X = Y$. When $l = 2$ and $r = 1$ these two codes have some similarities with self- and solid-locating-dominating codes and from now on we will only consider those cases.

In the following, we consider the relationships between these codes resulting in Figure 6, where we use the following abbreviations: D for dominating codes, $2D$ for 2-dominating codes, ID for identifying codes, SID for self-identifying codes, LD for locating-dominating codes, DLD for solid-locating-dominating codes, SLD for self-locating-dominating codes, LDB for $(1, \leq 2)$-locating-dominating codes of type B and LDA for $(1, \leq 2)$-locating-dominating codes of type A. Moreover, an arrow from X to Y denotes that a code of type X is also a code of type Y. The non-existence of an arrow (or path of multiple arrows) from X to Y means that there exists a code C of type X which is not of type Y. Below, we show the existence and non-existence of the arrows.

It has been shown in [33], that every $(1, \leq 2)$-locating-dominating code of type A is also of type B. Moreover, in Paper I we show that every self-locating-dominating code is also a solid-locating-dominating code and every solid-locating-dominating code is also locating-dominating. Each self-locating-dominating code is shown to be 2-dominating in Lemma 23 of Paper II. In Remark 21 of Paper I, we mention that every $(1, \leq 2)$-locating-dominating code of type B is also a solid-locating-dominating code. Indeed, if $C \subseteq V$ is a $(1, \leq 2)$-locating-dominating code of type B, $u, v \in V \setminus C$ and $I(v) \setminus I(u) = \emptyset$, then $I(v) \subseteq I(u)$ and $I(\{u, v\}) = I(u)$. Thus, we have a contradiction with C being a $(1, \leq 2)$-locating-dominating code of type B and, therefore, C is a solid-locating-dominating code.

Similarly, we can show that every $(1, \leq 2)$-locating-dominating code of type A is also a self-locating-dominating code. Indeed, if $C \subseteq V$ is a $(1, \leq 2)$-locating-dominating code of type A, $v \in V \setminus C$, $u \in V$ and $I(v) \setminus I(u) = \emptyset$, then $I(v) \subseteq I(u)$, $I(u) = I(\{v, u\})$ and $\{u, v\} \cap C = \{u\} \cap C$. Hence, we have a contradiction and C is a self-locating-dominating code. Observe that since we have shown, for example, that $(1, \leq 2)$-locating-dominating codes of type A are also a self-locating-dominating codes and that self-locating-dominating codes are also 2-dominating codes, we do not have to consider whether $(1, \leq 2)$-locating-dominating codes are also 2-dominating codes. The rest of the arrows in Figure 6 trivially follow from the definitions.

Next we will discuss about the missing arrows. Let G be the graph in Figure 3, where the darkened vertices form the code C. Now every I-set consists of three codewords and each I-set is unique. Hence, no I-set is a subset of another I-set and C is a self-identifying code by (2). However, the code C is not a $(1, \leq 2)$-locating-dominating code of type B. Indeed, we have $I(\{a, d\}) = I(\{b, c\}) = C$. Observe that since each self-identifying code is also an identifying code, there are identifying codes which are not $(1, \leq 2)$-locating-dominating codes of type B. Moreover, since $(1, \leq 2)$-locating-dominating codes of type A are $(1, \leq 2)$-locating-dominating codes of type B, there are identifying codes which are not $(1, \leq 2)$-locating-dominating codes of type A. In this way, a single counter example allows us to omit multiple arrows.

As we can see in Figure 1, there exists a graph with a 2-dominating code which is not locating-dominating. Moreover, the darkened vertices form an identifying code in Figure 2. However, those vertices do not form a 2-dominating or solid-locating-dominating code and hence, identifying codes are not always 2-dominating or solid-locating-dominating.

Let us then consider a path of two vertices, the graph P_2, that is, a graph consisting of two adjacent vertices. The only 2-dominating code consists of both vertices. However, to have a $(1, \leq 2)$-locating-dominating code of type B, we only need one of the vertices. Thus, there are $(1, \leq 2)$-locating-dominating codes of type B which are not 2-dominating. Moreover, there are no identifying codes in P_2 and hence, some $(1, \leq 2)$-locating-dominating codes of type A are not identifying codes. Together these observations give Figure 6. For example, the arrow from LDB to SID is missing because we have shown that there exist $(1, \leq 2)$-locating-dominating codes of type B which are not 2-dominating and, hence, cannot be self-identifying.

One of the most studied properties of these codes is how large they have to be in some graph. The smallest possible code in graph G is called *optimal*

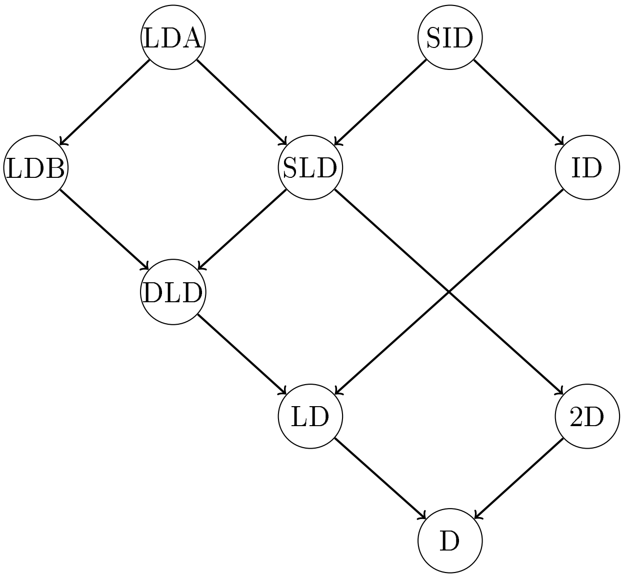


Figure 6: Relations between some different types of dominating codes. An arrow from X to Y denotes that each code of type X in a graph is also a code of type Y.

and its cardinality is denoted by one of $\gamma^{D}(G)$, $\gamma^{2D}(G)$, $\gamma^{LD}(G)$, $\gamma^{SLD}(G)$, $\gamma^{DLD}(G)$, $\gamma^{ID}(G)$, $\gamma^{SID}(G)$, $\gamma^{LDA}(G)$ or $\gamma^{LDB}(G)$ where the abbreviations are as in Figure 6. Hence, Figure 6 gives a bound $\gamma^{X}(G) \geq \gamma^{Y}(G)$ if there is a path from X to Y in Figure 6. Interested readers may find literature on these topics from Lobstein's vast internet bibliography [45].

The *kth* ($k \geq 1$) *power of graph* $G = (V, E)$ is $G^k = (V, E^k)$ where $e = uv \in E^k$ if $d(u, v) \leq k$ in G. Previously, many different dominating codes have been studied using r-radius balls instead of neighbourhoods, that is, for a vertex v, we would consider $I_r(v)$ rather than $I(v)$. For example, a code C in a graph $G = (V, E)$ is *distance-r dominating* (or r-covering [12]) if we have for every $v \in V$ a codeword $c \in C$ such that $d(v, c) \leq r$. Moreover, this notion can be generalized for self-locating-dominating and solid-locating-dominating codes. However, distance-r dominating codes in G are the same as the dominating codes in G^r [27, Proposition 12.2]. Similarly, other variants of dominating codes (discussed previously) with radius $r \geq 1$ can be viewed as codes with radius $r = 1$ in the rth power G^r of the underlying graph G. In this thesis, we mainly consider the case $r = 1$.

There are also other natural ways to locate vertices in a graph. The concept of *resolving set* was introduced by Harary and Melter [26] and independently by Slater [53]. Let $S = \{s_1, ..., s_k\}$ be a non-empty set of vertices. The *distance array* of a vertex $v \in V$ with respect to the set S is defined as $D(v) = (d(v, s_1), ..., d(v, s_k))$. If no two vertices have the same distance array, then the set S is called a resolving set. For recent developments of resolving sets see, for example, [15, 40, 42].

For other earlier related concepts, called *separating systems*, see [51] and [14].

Product graphs and previous results

In this section, we introduce the most relevant graphs and discuss about some previous results related to the thesis. A graph is a *complete graph* on n vertices if every two vertices are adjacent. We denote such a graph with K_n. The *Cartesian product* of two graphs $G_1 = (V_1, E_2)$ and $G_2 = (V_2, E_2)$ is denoted by $G_1 \Box G_2 = (V_1 \times V_2, E)$, where $e = (v_1, u_1)(v_2, u_2) \in E$ if $v_1, v_2 \in V_1$, $u_1, u_2 \in V_2$ and either $v_1 v_2 \in E_1$ and $u_1 = u_2$ or $u_1 u_2 \in E_2$ and $v_1 = v_2$. Let $\mathbb{F}_q$ be a finite field of q elements and $\mathbb{F}_2 = \mathbb{F}$. The Cartesian product of n copies of K_2, that is, $K_2 \Box \cdots \Box K_2$, is called an n dimensional binary hypercube or binary *Hamming space* (*Hamming graph*) of dimension n and denoted by $\mathbb{F}^n$. The binary Hamming space of dimension n is a vector space formed by the set of all n-tuples of $\mathbb{F}$. Now two vertices are adjacent if they differ at exactly one coordinate. Moreover, if we instead have a Cartesian product of n copies of K_q, where $q > 2$ is a prime power, then the resulting graph is called $\mathbb{F}_q^n$, the q-ary Hamming space of dimension n. Now the vector space is formed by the set of all n-tuples of $\mathbb{F}_q$. A vertex in a Hamming space is called a *word*. We denote a word $\mathbf{x} = (x_1, x_2, \ldots, x_n) = x_1 x_2 \cdots x_n$. The Hamming distance between two words $\mathbf{x} = x_1 \cdots x_n$ and $\mathbf{y} = y_1 \cdots y_n$ is $d(x, y) = |\{i \mid x_i \neq y_i\}|$. Note that the Hamming distance is equivalent with the graphical distance in $K_q \Box \cdots \Box K_q$. We denote the cardinality (volume) of an r-radius ball in the binary Hamming space by $V(n, r) = |B_r(\mathbf{x})|$ for any $\mathbf{x} \in \mathbb{F}^n$. Moreover, the *minimum distance* of a code $C \subseteq \mathbb{F}^n$ is $d_{\min}(C) = \min_{\mathbf{x}, \mathbf{y} \in C, \mathbf{x} \neq \mathbf{y}} d(\mathbf{x}, \mathbf{y})$. Furthermore, the code C is called *e-error-correcting* if the minimum distance between any two distinct codewords $\mathbf{c}_1$ and $\mathbf{c}_2$ is $d(\mathbf{c}_1, \mathbf{c}_2) \geq 2e+1$ ([57, p. 34]). Observe that if $C \subseteq \mathbb{F}^n$ is an e-error-correcting code, then we have $|I_e(\mathbf{x})| \leq 1$ for all $\mathbf{x} \in \mathbb{F}^n$.

Different types of identifying and locating-dominating codes have been widely studied in the Cartesian products of complete graphs and especially in Hamming graphs. In [41], Karpovsky, Chakrabarty and Levitin have given the best known lower bound for identifying codes in binary Hamming graphs stating that $\gamma^{ID}(\mathbb{F}^n) \geq \frac{n2^{n+1}}{n(n+1)+2}$ and a general upper bound $\gamma^{ID}(\mathbb{F}^n) \leq n|C|$, where C is a distance-2 dominating code. In [4], Blass, Honkala and Litsyn have given some direct sum constructions for identifying codes in Hamming spaces, for example, if C is an identifying code in $\mathbb{F}^n$, then $C \oplus \mathbb{F}^2$ is an identifying code in $\mathbb{F}^{n+2}$. Moreover, some constructions for small values of n as well as for $n = 2^m - 1$, where $m \geq 4$, are given in the paper.

In [17], Exoo, Laihonen and Ranto have considered such identifying codes in $\mathbb{F}^n$ that every word is covered by at least two codewords. Based on these codes they have given new identifying codes using the so-called $\pi(u)$-*construction* (see e.g. Theorem 26 of Paper I). In addition, some computer search based constructions are given. In [16], Exoo *et al.* have given the general upper bound in Hamming spaces $\gamma^{ID}(\mathbb{F}^{n+1}) \leq (2 + \frac{1}{n+1})\gamma^{ID}(\mathbb{F}^n)$ as well as some other constructions. Moreover, in [46], Moncel has shown that $\gamma^{ID}(\mathbb{F}^n) \leq \gamma^{ID}(\mathbb{F}^{n+1})$. In [8], Charon *et al.* have presented more constructions with the help of computers. In [34], Honkala and Lobstein have studied the asymptotical behaviour of identifying codes in Hamming spaces. Together these papers give good evaluations for values $\gamma^{ID}(\mathbb{F}^n)$ when n is small (see tables in [8]). There exist less results on location-domination in Hamming graphs. In [33], Honkala, Laihonen and Ranto have given a general lower bound for locating-dominating codes in the binary Hamming spaces:

$$\gamma^{LD}(\mathbb{F}^n) \geq \left\lceil \frac{n^2 2^{n+1}}{n^3 + 2n^2 + 3n - 2} \right\rceil .$$

They have also given a general upper bound: $\gamma^{LD}(\mathbb{F}^{n+1}) \leq (2n-1)|C|$, where C is a distance-2 dominating code in $\mathbb{F}^n$. However, $\gamma^{LD}(\mathbb{F}^n) \leq \gamma^{ID}(\mathbb{F}^n)$ is often the best known upper bound.

Identifying codes have also been studied in $K_n \square K_m$. In [25], Gravier, Moncel and Semri give the exact value of optimal codes when $n = m$:

$$\gamma^{ID}(K_n \square K_n) = \left\lfloor \frac{3n}{2} \right\rfloor .$$

Moreover, they show that their construction is the unique optimal code when $n \geq 5$ and n is odd. In [23], Goddard and Wash have given exact values for optimal identifying codes in $K_n \square K_m$:

$$\gamma^{ID}(K_n \square K_m) = \begin{cases} m + \lfloor \frac{n}{2} \rfloor & \text{if } m \leq \frac{3n}{2} \\ 2m - n & \text{if } m \geq \frac{3n}{2}. \end{cases} \tag{3}$$

They also consider identification in Cartesian products of multiple complete graphs. In particular for three equal complete graphs, they give bounds

$$n^2 - n\sqrt{n} \leq \gamma^{ID}(K_n \square K_n \square K_n) \leq n^2 \tag{4}$$

and conjecture that n^2 is the optimal bound. In Paper III, we show that this conjecture does not hold and the lower bound is also improved.

Besides the Cartesian products of complete graphs, we also consider direct products of two complete graphs. The *direct product* of two graphs $G_1 = (V_1, E_2)$ and $G_2 = (V_2, E_2)$ is denoted by $G_1 \times G_2 = (V_1 \times V_2, E)$, where $e = (v_1, u_1)(v_2, u_2) \in E$ if and only if $v_1v_2 \in E_1$ and $u_1u_2 \in E_2$ for $v_1, v_2 \in V_1$ and $u_1, u_2 \in V_2$. Previously identifying codes have been studied in the direct product of two complete graphs in [49] by Rall and Wash. They have found the optimal cardinalities for identifying codes. We have gathered those results below

$$\gamma^{ID}(K_n \times K_m) = \begin{cases} m-1 & \text{if } n \geq 3, m \geq 2n, \\ \lfloor \frac{2m+2n+1}{3} \rfloor & \text{if } 2n-1 \geq m \geq n \geq 6 \\ & \text{and } m \neq 2n-5, \\ 2n-4 & \text{if } m = 2n-5 \geq 7. \end{cases} \tag{5}$$

Interestingly, a direct product of two complete graphs $K_n \times K_m$ is a complement of Cartesian product of two complete graphs $K_n \square K_m$, that is, $\overline{K_n \square K_m} = K_n \times K_m$ ([49]).

Cycles, trees and previous results

Paper II considers many different graphs. Hence, in this section, we introduce some basic graph classes and domination related results on them. A graph $C_n = (V, E)$ is a *cycle* on $n \geq 3$ vertices if $|V| = n$, C_n is connected and each vertex has two neighbours. A graph G is a *tree* if it is connected and does not contain any cycles. Previously optimal locating-dominating codes in paths and cycles have been given by Slater in [55]:

$$\gamma^{LD}(P_n) = \gamma^{LD}(C_n) = \begin{cases} 2k & \text{if } n = 5k \\ 2k+1 & \text{if } n = 5k+1 \text{ or } n = 5k+2 \text{ and} \\ 2k+2 & \text{if } n = 5k+3 \text{ or } n = 5k+4. \end{cases}$$

Optimal identifying codes in paths have been provided by Bertrand *et al.* [2]:

$$\gamma^{ID}(P_n) = \begin{cases} \frac{n+1}{2} & \text{if } n \geq 1 \text{ is odd,} \\ \frac{n}{2}+1 & \text{if } n \geq 4 \text{ is even.} \end{cases}$$

Optimal identifying codes in odd cycles have been presented by Gravier, Moncel and Semri in [24] and optimal codes in even cycles are due to Bertrand *et al.* [2]:

$$\gamma^{ID}(C_n) = \begin{cases} \frac{n+1}{2}+1 & \text{if } n \geq 7 \text{ is odd,} \\ \frac{n}{2} & \text{if } n \geq 6 \text{ is even.} \end{cases}$$

Moreover, locating-dominating codes have been studied in trees with l leaves and s support vertices in [5, 54, 52]. A *leaf* of a tree is a vertex with a single neighbour and a *support vertex* is any vertex adjacent to a leaf. First in [54], Slater has given bounds $\frac{n}{3} < \gamma^{LD}(T) \leq n-1$ for $n \geq 2$ for trees. Then Blidia *et al.* in [5] have improved the bounds to

$$\frac{n+1+l-s}{3} \leq \gamma^{LD}(T) \leq \frac{n+l-s}{2}$$

and in [52] Sewell and Slater have improved the lower bound to

$$\frac{n+1+2(l-s)}{3} \leq \gamma^{LD}(T)$$

and given trees which achieve this bound. For identifying codes in trees Bertrand *et al.* [3] have given bound $\gamma^{ID}(T) \geq \frac{3(n+1)}{7}$ for $n \geq 3$. Furthermore, Blidia *et al.* [5] have improved this bound to

$$3\frac{n+1+l-s}{7} \leq \gamma^{ID}(T).$$

In [1], Auger has constructed a linear algorithm which gives the cardinality of a minimum identifying code in a tree.

Extremal and general graph theoretical results

In this section, we present some extremal results on identifying and locating-dominating codes as well as some more general graph theoretical results. Extremal questions on dominating codes in graphs consider, for example, what is the smallest (or the largest) optimal code in any graph on n vertices.

Karpovsky *et al.* [41] have given a general lower bound for the cardinality of an identifying code in a connected graph G on n vertices. In particular, we have

$$\lceil \log_2(n+1) \rceil \leq \gamma^{ID}(G) \leq n-1.$$

The upper bound is due to Charon *et al.* [10]. Furthermore, in [18], Foucaud *et al.* have classified all graphs attaining this upper bound. For similar results on locating-dominating codes, see [10] by Charon *et al.*

In [29], Hernando *et al.* have shown that

$$|\gamma^{LD}(G) - \gamma^{LD}(\overline{G})| \leq 1.$$

By comparing bounds (3) and (5) when $m = 2n$ we notice that this bound does not hold for identifying codes. However, for identifying codes, there exists a similar but weaker result, see [21] by Foucaud *et al.*

In [7], Cáceres *et al.* have considered realization type theorems for γ^{LD} and some other graph parameters. They have especially shown that if $3 \leq a \leq b \leq 2a-2$, then there always exists a tree T, such that $\gamma^{LD}(T) = b$ and $\eta(T) = a$ where $\eta(T)$ is the cardinality of the minimum metric-locating-dominating set (for the definition see [7]).

Also some other graph parameters are connected to domination type parameters. A set of vertices S in graph G is *independent* if none of the vertices are adjacent. We denote by $\beta(G)$ the cardinality of the largest independent set in G. In [22] Garijo *et al.* have shown that

$$\gamma^{LD}(G) \leq n - \beta(G)$$

when $G = (V, E)$ is *twin-free*, that is, for no two distinct vertices $v, u \in V$ we have $N[v] = N[u]$ or $N(v) = N(u)$, and $|V| \geq 2$. Moreover, in [19] Foucaud *et al.* have given a similar result for identifying codes.

Besides independence number, for example, also $\gamma^{2D}(G)$ has been previously used to bound $\gamma^{ID}(G)$ in [20] by Foucaud and Perarnau and $\gamma^{LD}(G)$ in [22] by Garijo *et al.*

Infinite graphs and previous results

Besides finite graphs, we also study infinite grids. Let $G = (V, E)$, where $V = \mathbb{Z}^2$ and two vertices are adjacent if their Euclidean distance is at most $\sqrt{2}$. We call this graph the *infinite king grid*. In finite graphs optimal codes are defined to be the codes with the smallest possible cardinalities. However, when we consider identifying or other similar codes in infinite graphs, they do not have a finite number of codewords. Hence, the optimality of a code in an infinite graph is defined using the density of the code. Let $V_n = \{(x, y) \mid |x| \leq n, |y| \leq n\}$. Now the *density* of code $C \subseteq V$ is defined as

$$D(C) = \limsup_{n \to \infty} \frac{|V_n \cap C|}{|V_n|}.$$

Similarly to the king grid, we also define a triangular grid. Graph $T = (V', E')$ is the infinite *triangular grid* if $V' = \left\{ i(1,0) + j\left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right) \mid i, j \in \mathbb{Z} \right\}$ and two vertices are adjacent if their Euclidean distance is 1. Let $T_n = \left\{ i(1,0) + j\left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right) \mid |i|, |j| \leq n \right\}$. Now we define the density of a code $C' \subseteq V'$ as

$$D(C') = \limsup_{n \to \infty} \frac{|T_n \cap C'|}{|T_n|}.$$

Previously, identifying codes have been studied in the king grid in [13] by Cohen, Honkala and Lobstein and by Charon, Hudry and Lobstein in [9], where they also consider identification in the triangular grid. Moreover, locating-dominating codes have been investigated in the king grid in [31] by Honkala and Laihonen and in the triangular grid in [30] by Honkala. Furthermore, self-identifying codes have been considered in the king and triangular grids in [32] by Honkala and Laihonen.

In Table 1, optimal densities for some codes in the king and triangular grids have been presented (with the relevant references). Note that the optimal densities for dominating and 2-dominating codes are easy to obtain. We get the densities with simple double counting arguments. In the king grid, there are nine vertices in a closed neighbourhood and every vertex must have at least one codeword in its neighbourhood. Now we use the double counting method on a vertex pair (c, v) where c is a codeword and $c \in I(v)$. Thus, for large n, we have $9|C \cap V_n| \gtrsim |V_n|$[1] and $|C \cap V_n|/|V_n| \gtrsim \frac{1}{9}$. Moreover, a construction achieving this bound is easy to find; any construction where $|I(v)| = 1$ for any vertex v attains the bound. For the triangular grid the bound is achieved using a similar method. The only difference is that there are now seven vertices in a closed neighbourhood of a vertex instead of nine. The bounds for 2-domination are achieved using an analogous method. For example, for the king grid the double counting argument gives (for large n) the inequality $9|C \cap V_n| \gtrsim 2|V_n \setminus C| + |C \cap V_n| = 2|V_n| - |C \cap V_n|$. For the constructions, the goal is to have two codewords in every I-set of a non-codeword and one codeword in every I-set of a codeword. Moreover, such constructions exist and the double counting arguments give the attainable lower bounds. In addition, in [47], Pelto has shown that the optimal locating-dominating codes of types A and B have density of 1/3 in the infinite king grid.

	γ^{ID}	γ^{LD}	γ^{D}	γ^{2D}	γ^{SID}	γ^{SLD}	γ^{DLD}
King	$\frac{2}{9}$ [9, 13]	$\frac{1}{5}$ [31]	$\frac{1}{9}$	$\frac{1}{5}$	$\frac{1}{3}$ [32]	$\frac{1}{3}$ [IV]	$\frac{1}{3}$ [IV]
Triangle	$\frac{1}{4}$ [9]	$\frac{13}{57}$ [30]	$\frac{1}{7}$	$\frac{1}{4}$	$\frac{1}{2}$ [32]	$\frac{1}{4}$ [IV]	$\frac{1}{4}$ [IV]

Table 1: Optimal densities of some dominating codes in the infinite king and triangular grids.

Many of the proofs for different dominating codes in this thesis are innovative in the sense that they do not only use standard methods but rather rely on the structure of the graph (see paper III) or are linked to other concepts — for example, such as independence number (Paper II), Dilworth number (Paper II) or Latin square (Paper III). Some typical methods, which we have not used in this thesis, for evaluating the cardinalities of optimal codes include, for example, *share*. Share of a codeword $c \in C$, $s(c) = \sum_{v \in N[c]} \frac{1}{|I(v)|}$, was introduced in [56] by Slater and it has also been used to show, for example, that the density of an optimal locating-dominating code in the king grid is $\frac{1}{5}$ ([31]). The basic idea behind it is that if C is a dominating code, then $\sum_{c \in C} s(c) = |V|$. Moreover, if there is some positive real number α

---

[1] We consider this for large n. Namely, we may have some vertices in $V_n \setminus V_{n-1}$ which are not dominated by a codeword. However, $V_n \setminus V_{n-1}$ is relatively small compared to V_n and $\frac{|V_{n-1}|}{|V_n|}$ tends to 1 as n tends to infinity. The results we get hold due to the definition of the density and we present them like this for simplicity.

such that for each codeword $c \in C$ we have $s(c) \leq \alpha$, then $\sum_{c \in C} s(c) \leq |C|\alpha$ and thus, $|C| \geq \frac{|V|}{\alpha}$. If we consider for example the graph G of Figure 1 and 2-domination, we have $s(c) \leq 2 \cdot \frac{1}{2} + 1$ for each codeword since each non-codeword has at least two codeword neighbours. Thus, $\gamma^{2D}(G) \geq \frac{4}{2} = 2$.

1.2 Levenshtein's channel

Levenshtein's channel model was first introduced in 2001 by Levenshtein [44] and it has become relevant again due to advanced data storage systems such as DNA storage [6, 11, 35]. Its premise is that the channel we use to transmit the information causes many errors and to combat this the information is sent multiple times giving us multiple approximations of the transmitted information. Together these approximations are used to identify the transmitted information more precisely than we could have identified with only one channel and one approximation.

We especially focus on a situation where we have an e-error-correcting code $C \subseteq \mathbb{F}^n$. A word $\mathbf{x} \in C$ is sent through $N \geq 1$ channels and we get a set Y consisting of N different evaluations $\mathbf{y}_i$ of $\mathbf{x}$, where $1 \leq i \leq N$. Furthermore, each of these output words $\mathbf{y}_i$ is within Hamming distance $t = e + \ell$ of $\mathbf{x}$, that is, $d(\mathbf{y}_i, \mathbf{x}) \leq t$. In other words, there occurs at most t *substitution errors*, where a symbol is substituted by another symbol, in any channel. Based on the set Y, we get a list of codewords which might be transmitted:

$$T(Y) = C \cap \left(\bigcap_{i=1}^{N} B_t(\mathbf{y}_i) \right).$$

The channel model is illustrated in Figure 7. Furthermore, we denote by $\mathcal{L}$ such a value that $|T(Y)| \leq \mathcal{L}$ for every possible set of N output words and any transmitted word $\mathbf{x}$. Thus, roughly saying, as the number of channels N increases the uncertainty $\mathcal{L}$ decreases. Observe that if $\ell \leq 0$, then we have $Y \subseteq B_e(\mathbf{x})$. Hence, as C is an e-error-correcting code, we have $B_e(\mathbf{y}) \cap C = \{\mathbf{x}\}$ for each $\mathbf{y} \in Y$ and thus, we can deduce $\mathbf{x}$ exactly. Also notice that the nature of the base code C has a big role in the value $\mathcal{L}$. For example, if we have only two codewords in C which locate at distance $2e + 1$ from each other, then we have $\mathcal{L} \leq 2$, no matter how many channels we have. However, for application purposes, a code of two codewords is rather useless. For example, in the case of memory storage systems, we store memory units as codewords of an e-error-correcting code to protect the data from e reading errors. Since we usually want to store as much information as possible to the code C, we assume that C is large. We mostly consider the worst possible cases for $\mathcal{L}$. When we consider codewords as memory units, the channels can be understood as, for example, a read head which reads the same unit multiple (say N) times.

Let us, for example, consider a 1-error-correcting code $C = \{00000, 11100, 00111, 11011\} \subseteq \mathbb{F}^5$ and Levenshtein's channel model with $N = 2$ and $\ell = 2$. Now $t = 3$. Let us transmit the codeword $\mathbf{x} = 00000$ through two channels and let us assume that we get output words $Y = \{\mathbf{y}_1, \mathbf{y}_2\}$ where $\mathbf{y}_1 = 11000$ and $\mathbf{y}_2 = 00111$. Now, we have $I_3(\mathbf{y}_1) = C \cap B_3(\mathbf{y}_1) = \{00000, 11100, 11011\}$ and $I_3(\mathbf{y}_2) = \{00000, 00111, 11011\}$. Therefore, we have $T(Y) = I_3(\mathbf{y}_1) \cap I_3(\mathbf{y}_2) = \{00000, 11011\}$.

Previously, Levenshtein [44] has given the number of channels required to know the transmitted codeword $\mathbf{x}$ exactly, that is, to have $\mathcal{L} = 1$. Moreover, Yaakobi and Bruck have shown in [58, 59] the required number of channels N to know that the transmitted information is one of two possible options. However, as the number of channels grows, the problem changes. In [12, p. 36], it has been shown that the cardinality of intersection of four or more balls does not depend only on the distances between the centres of these balls; unlike in the case of two or three balls. Therefore, we use different methods in Paper V.

Levenshtein's original motivation to study this scheme stems from areas such as chemistry and molecular biology. There, in some cases, while transmitting the information we get so many errors that sending the information multiple times becomes a valid option. Levenshtein's channel is also linked to *associative memories* (see [36, 37, 58, 59]). The basic idea of an associative memory is that every memory unit is associated with some other memory units and we try to retrieve the wanted information unit based on these associations. Now, we may consider vertices of a graph as memory units and its edges as associations between memory units. Let us say that we try to retrieve information unit $\mathbf{x}$ and we have a set of *memory* or *input clues* which belong to some set C and are t-associated with $\mathbf{x}$, that is, the set of memory clues is $I_t(\mathbf{x})$ and the set of vertices which is t-associated with

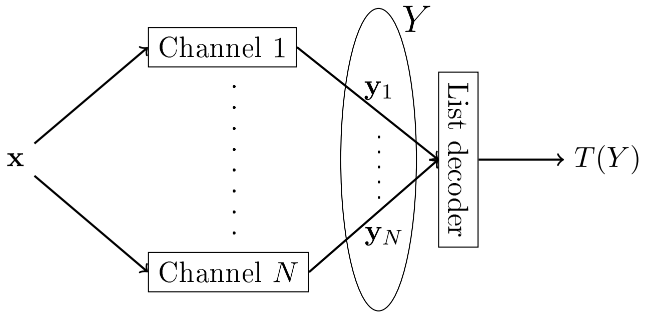


Figure 7: Levenshtein's channel model.

these memory clues is

$$\bigcap_{\mathbf{c}\in I_t(\mathbf{x})} B_t(\mathbf{c}).$$

Notice that if $\bigcap_{\mathbf{c}\in I_t(\mathbf{x})} B_t(\mathbf{c}) = \{\mathbf{x}\}$ for each $\mathbf{x}$, then the memory returns unambiguously the vertex we were searching. Moreover, this is equivalent with the definition of $(t, \leq 1)^+$-identifying codes. Thus, if we have a $(t, \leq 1)^+$-identifying code C and the set $I_t(\mathbf{x})$ as memory clues, then we can successfully retrieve $\mathbf{x}$. However, the above sketch of the problem is a simplification of associative memories since we may require that we use at most m of the memory clues of set $I_t(\mathbf{x})$, where m is some positive integer. More comparison between associative memories, identification and Levenshtein's channel model can be found in [37].

Although the Levenshtein's channel model and associative memories have some apparent similarities, they also have a fundamental difference. In the Levenshtein's channel model we try to store as much information as possible and hence, we wish to have as big code C as possible. While in the case of associative memories we wish to be able to find the information with as few memory clues as possible and thus, we wish to have as small code C as possible. In [59, Theorem 1], Yaakobi and Bruck have shown the existence of a dependency between $\mathcal{L}$, the number of channels N and the number of memory clues. There are also other kinds of approaches, for example, in [43], we have considered Levenshtein's channel model and associative memories using majority voting.

2 Papers of the thesis

In Paper I, we have introduced self- and solid-locating-dominating codes for the first time. First we give optimal self-locating-dominating and solid-locating-dominating codes for Cartesian product of two complete graphs, that is, for graphs $K_n \square K_m$, where n and m are positive integers. Moreover, we also consider self- and solid-locating-dominating codes in the binary Hamming spaces. For self-locating-dominating codes we give a general lower bound and an infinite family of constructions which achieve this bound. For solid-locating-dominating codes we give a general lower bound and show that it is asymptotically correct with an infinite family of codes. In Remark 21 of Paper I, we have pointed out that our new general lower bound for solid-locating-dominating codes in the binary Hamming spaces surprisingly improves the known lower bound for $(1, \leq 2)$-locating-dominating codes of type B in the binary Hamming spaces.

In addition, to the results on self- and solid-locating-dominating codes, we also consider locating-dominating codes. We give optimal locating-dominating codes for $K_n \square K_m$ and a new family of constructions for locating-dominating codes in binary Hamming spaces based on identifying codes which are also 2-dominating. This new method gives some new constructions which are rather close to the best known lower bound of locating-dominating codes. For example, in $\mathbb{F}^{11}$ we get a code of cardinality 320 while the current lower bound is 309.

In the second paper, self-locating-dominating and solid-locating-dominating codes are widely studied under a vast array of graph structures and properties. Using *Sperner's theorem* (see Theorem 10 in Paper II) we give an upper bound for the number of vertices in a graph when the number of codewords in a self- or solid-locating-dominating code is known. For self-locating-dominating codes, we define so-called *forced vertices*, that is, the vertices which are codewords in every code (see [20] for a similar concept), and give an exact condition for their existence using *vicinal preorder.* With this concept, we also classify every such graph G that $\gamma^{SLD}(G) = |V|$. Moreover, with the help of *Dilworth's number* (see Section 2.2 of Paper II) we carry the ideas of vicinal preorder to solid-location-domination. This gives a general lower bound for solid-location-domination and informs us that for a graph G with at least one edge we have $\gamma^{DLD}(G) = |V| - 1$ if and only if G is a *threshold graph* (see page 7 of Paper II). We give a general upper bound (see Theorem 20 in Paper II) for $\gamma^{DLD}(G)$ using independence number when G is connected and $N(u) \not\subseteq N(v)$ for each distinct $u, v \in V$. In particular, we show that $\gamma^{DLD}(G) \leq n - \beta(G)$. We also show a similar result for self-locating-dominating codes.

We also show that the difference of the cardinalities of optimal solid-locating-dominating codes in a graph and its complement is at most one, that is,

$$|\gamma^{DLD}(G) - \gamma^{DLD}(\overline{G})| \leq 1.$$

However, a similar bound does not work for self-locating-dominating codes. In fact, we show that there exists a graph G' such that

$$|\gamma^{SLD}(G') - \gamma^{SLD}(\overline{G'})| \geq \binom{k}{\lfloor \frac{k}{2} \rfloor} - k$$

where $k = \gamma^{SLD}(G')$. Besides these results, we determine all such pairs of integer $a, b \in \mathbb{Z}$ that there exists a graph G with $\gamma^{SLD}(G) = a$ (or $\gamma^{DLD}(G) = a$) and $\gamma^{LD}(G) = b$.

In the case of (specific) graph classes, we give exact values for optimal self- and solid-locating-dominating codes in paths, cycles and ladders ($P_2 \square P_n$). Moreover, we show that in trees a code is self-locating-dominating if and only if it is 2-dominating. Furthermore, we also show that for a tree T the independence number and the cardinality of minimal solid-locating-dominating code are equal

$$\gamma^{DLD}(T) = \beta(T).$$

However, even though the values are the same, there are optimal solid-locating-dominating codes which are not independent sets and there are maximal independent sets which are not solid-locating-dominating codes. In conclusion, the paper uses rather non-standard techniques on the field of location and establishes surprising connections between different parameters.

The third paper considers non-binary Hamming spaces and especially $K_q \square K_q \square K_q$. Originally, Karpovsky *et al.* [41] have given a general lower bound applicable for Cartesian product of n copies of complete graphs K_q

$$\gamma^{ID}(K_q \square \cdots \square K_q) \geq \frac{2q^n}{nq-n+2}$$

which gives $\gamma^{ID}(K_q \square K_q \square K_q) \geq \frac{2q^3}{3q-1} \geq \frac{2q^2}{3}$. In addition, Goddard and Wash have given Estimation (4) and conjectured that $\gamma^{ID}(K_q \square K_q \square K_q) = q^2$. We disprove this conjecture by constructing an infinite family of identifying codes in $K_q \square K_q \square K_q$ with cardinality $q^2 - \frac{q}{4}$ when q is a power of four. Interestingly, we manage to generalize our construction for values of q which are not powers of four by using *Evans' theorem* for extending *Latin squares* (see Theorem 22 of Paper III). Moreover, we also improve the lower bound (4) given by Goddard and Wash. Our lower bound is based on a new approach, which builds on the method of Goddard and Wash [23]. The proof is rather non-standard and the idea is roughly that a "small" code causes problems which require additional codewords causing again more problems. Now, if the code is too small, then we require more codewords to solve all the problems than there are codewords in the code itself. Together, our construction and the lower bound show that

$$\gamma^{ID}(K_q \square K_q \square K_q) = q^2 - \Theta(q).$$

In Section 2 of Paper III, we determine every optimal self-locating-dominating code in $K_q \square K_q \square K_q$ by showing an interesting connection: There is a one-to-one mapping from the set of optimal self-locating-dominating codes in $K_q \square K_q \square K_q$ to $q \times q$ Latin squares. Furthermore, in Section 5, we consider $\mathbb{F}_q^n$ when q is a prime power. We give some families of optimal self-identifying and self-locating-dominating codes for suitable values of n and q. Furthermore, we show that these codes are also identifying and that they are significantly smaller than the previously known constructions.

In the fourth paper, we again consider self- and solid-locating-dominating codes in graphs. We determine the cardinalities and give some constructions of optimal self-locating-dominating, solid-locating-dominating and locating-dominating codes in the direct product of any two complete graphs. For this, we use results on Cartesian product of two complete graphs from Paper

I and results on complements of graphs from Paper II and [29]. We also give an optimal solid-locating-dominating code in the Cartesian product of three identical complete graphs. Finally, we consider self- and solid-locating-dominating codes in infinite grids. We give optimal codes (density wise) for triangular and king grids. Especially, the proof of the lower bound and the construction for solid-locating-dominating code in the infinite king grid are non-standard. The construction contains a single "central" vertex which is in a unique position. Moreover, to prove the lower bound, we consider one-way infinite three vertex wide strips.

Finally, the fifth paper considers Levenshtein's channel model. The results of Paper V have been partially published in [39]. Now we assume that the errors occurring in the channels are substitution errors. We transmit a codeword $\mathbf{x} \in C \subseteq \mathbb{F}^n$ through N identical channels each causing at most $e + \ell$ errors, where C is an e-error-correcting code. We show that if $N \geq V(n, \ell-1)+1$, then list size $\mathcal{L} \leq 2^{\ell}$. Moreover, when n is big enough we get $\mathcal{L} \leq \ell+1$. Intriguingly, the famous *Sauer-Shelah Lemma* (see Theorem 5 of Paper V) proves to be useful attaining these bounds. We show that there exist e-error-correcting codes such that the bound $\mathcal{L} \leq \ell+1$ can be attained. We also give asymptotical results for the list size when C is such an e-error-correcting code that $\mathcal{L}$ is maximal and $V(n, \ell-a-1)+1 \leq N \leq V(n, \ell-a)$. In particular, we show that

$$\mathcal{L} = \Theta(n^a).$$

Although there exist e-error-correcting codes such that the list size $\mathcal{L}$ depends on n when $N \leq V(n, \ell-1)$, we have also constructed rather large e-error-correcting codes such that $\mathcal{L}$ is constant on n when $N \geq V(n, \ell-2)+1$. We also show that if n is large enough and e is large compared to ℓ, then it is very likely that we only need two channels to have small $\mathcal{L}$ (assuming some randomness on errors). The results in Paper V related to the probabilistic case are improved versions of the results in the conference version of Paper V [39].

Bibliography

[1] David Auger. Minimal identifying codes in trees and planar graphs with large girth. *European Journal of Combinatorics*, 31(5):1372–1384, 2010.

[2] Nathalie Bertrand, Irène Charon, Olivier Hudry, and Antoine Lobstein. Identifying and locating-dominating codes on chains and cycles. *European Journal of Combinatorics*, 25(7):969–987, 2004.

[3] Nathalie Bertrand, Irène Charon, Olivier Hudry, and Antoine Lobstein. 1-identifying codes on trees. *Australasian Journal of Combinatorics*, 31:21–36, 2005.

[4] Uri Blass, Iiro Honkala, and Simon Litsyn. Bounds on identifying codes. *Discrete Mathematics*, 241(1-3):119–128, 2001.

[5] Mostafa Blidia, Mustapha Chellali, Frédéric Maffray, Julien Moncel, and Ahmed Semri. Locating-domination and identifying codes in trees. *Australasian Journal of Combinatorics*, 39:219–232, 2007.

[6] James Bornholt, Randolph Lopez, Douglas M. Carmean, Luis Ceze, Georg Seelig, and Karin Strauss. A DNA-based archival storage system. In *Proceedings of ASPLOS*, pages 637–649, 2016.

[7] José Cáceres, Carmen Hernando, Mercè Mora, Ignacio M. Pelayo, and María Luz Puertas. Locating–dominating codes: Bounds and extremal cardinalities. *Applied Mathematics and Computation*, 220:38–45, 2013.

[8] Irène Charon, Gérard Cohen, Olivier Hudry, and Antoine Lobstein. New identifying codes in the binary Hamming space. *European Journal of Combinatorics*, 31(2):491–501, 2010.

[9] Irene Charon, Olivier Hudry, and Antoine Lobstein. Identifying codes with small radius in some infinite regular graphs. *The Electronic Journal of Combinatorics*, 9(1):R11, 2002.

[10] Irène Charon, Olivier Hudry, and Antoine Lobstein. Extremal cardinalities for identifying and locating-dominating codes in graphs. *Discrete mathematics*, 307(3-5):356–366, 2007.

[11] George M. Church, Yuan Gao, and Sriram Kosuri. Next-generation digital information storage in DNA. *Science*, 337(6102):1628–1628, 2012.

[12] Gérard Cohen, Iiro Honkala, Simon Litsyn, and Antoine Lobstein. *Covering codes*, volume 54 of *North-Holland Mathematical Library*. North-Holland Publishing Co., Amsterdam, 1997.

[13] Gérard D. Cohen, Iiro Honkala, and Antoine Lobstein. On codes identifying vertices in the two-dimensional square lattice with diagonals. *IEEE Transactions on Computers*, 50(2):174–176, 2001.

[14] T. J. Dickson. On a problem concerning separating systems of a finite set. *Journal of Combinatorial Theory*, 7(3):191–196, 1969.

[15] Alejandro Estrada-Moreno, Ismael G. Yero, and Juan A. Rodríguez-Velázquez. The k-metric dimension of the lexicographic product of graphs. *Discrete Mathematics*, 339(7):1924–1934, 2016.

[16] Geoffrey Exoo, Ville Junnila, Tero Laihonen, and Sanna Ranto. Upper bounds for binary identifying codes. *Advances in Applied Mathematics*, 42(3):277–289, 2009.

[17] Geoffrey Exoo, Tero Laihonen, and Sanna Ranto. Improved upper bounds on binary identifying codes. *IEEE Transactions on Information Theory*, 53(11):4255–4260, 2007.

[18] Florent Foucaud, Eleonora Guerrini, Matjaž Kovše, Reza Naserasr, Aline Parreau, and Petru Valicov. Extremal graphs for the identifying code problem. *European Journal of Combinatorics*, 32(4):628–638, 2011.

[19] Florent Foucaud, Ralf Klasing, Adrian Kosowski, and André Raspaud. On the size of identifying codes in triangle-free graphs. *Discrete Applied Mathematics*, 160(10-11):1532–1546, 2012.

[20] Florent Foucaud and Guillem Perarnau. Bounds for identifying codes in terms of degree parameters. *The Electronic Journal of Combinatorics*, P32, 2012.

[21] Florent Foucaud, Guillem Perarnau, and Oriol Serra. Random subgraphs make identification affordable. *Journal of Combinatorics*, 8(1):57–77, 2017.

[22] Delia Garijo, Antonio González, and Alberto Márquez. The difference between the metric dimension and the determining number of a graph. *Applied Mathematics and Computation*, 249:487–501, 2014.

[23] Wayne Goddard and Kirsti Wash. ID codes in Cartesian products of cliques. *The Journal of Combinatorial Mathematics and Combinatorial Computing*, 85:97–106, 2013.

[24] Sylvain Gravier, Julien Moncel, and Ahmed Semri. Identifying codes of cycles. *European Journal of Combinatorics*, 27(5):767–776, 2006.

[25] Sylvain Gravier, Julien Moncel, and Ahmed Semri. Identifying codes of Cartesian product of two cliques of the same size. *The Electronic Journal of Combinatorics*, 15(1):4, 2008.

[26] Frank Harary and Robert Melter. On the metric dimension of a graph. *Ars Combinatoria*, 2:191–195, 1976.

[27] Teresa W. Haynes, Stephen T. Hedetniemi, and Peter J. Slater. *Domination in Graphs: Advanced Topics*. Marcel Dekker, New York, 1998.

[28] Teresa W. Haynes, Stephen T. Hedetniemi, and Peter J. Slater. *Fundamentals of Domination in Graphs*. Marcel Dekker, New York, 1998.

[29] Carmen Hernando, Mercè Mora, and Ignacio M. Pelayo. Nordhaus–Gaddum bounds for locating domination. *European Journal of Combinatorics*, 36:1–6, 2014.

[30] Iiro Honkala. An optimal locating-dominating set in the infinite triangular grid. *Discrete Mathematics*, 306(21):2670–2681, 2006.

[31] Iiro Honkala and Tero Laihonen. On locating–dominating sets in infinite grids. *European Journal of Combinatorics*, 27(2):218–227, 2006.

[32] Iiro Honkala and Tero Laihonen. On a new class of identifying codes in graphs. *Information Processing Letters*, 102(2-3):92–98, 2007.

[33] Iiro Honkala, Tero Laihonen, and Sanna Ranto. On locating-dominating codes in binary Hamming spaces. *Discrete Mathematics and Theoretical Computer Science*, 6(2), 2004.

[34] Iiro Honkala and Antoine Lobstein. On identifying codes in binary Hamming spaces. *Journal of Combinatorial Theory, Series A*, 99(2):232–243, 2002.

[35] Michal Horovitz and Eitan Yaakobi. Reconstruction of sequences over non-identical channels. *IEEE Transactions on Information Theory*, 65(2):1267–1286, 2018.

[36] Ville Junnila and Tero Laihonen. Codes for information retrieval with small uncertainty. *IEEE Transactions on Information Theory*, 60(2):976–985, 2013.

[37] Ville Junnila and Tero Laihonen. Information retrieval with varying number of input clues. *IEEE Transactions on Information Theory*, 62(2):625–638, 2015.

[38] Ville Junnila and Tero Laihonen. Tolerant location detection in sensor networks. *Advances in Applied Mathematics*, 112:101938, 2020.

[39] Ville Junnila, Tero Laihonen, and Tuomo Lehtilä. The levenshtein's channel and the list size in information retrieval. In *Proceedings of 2019 IEEE International Symposium on Information Theory (ISIT)*, pages 295–299, 2019.

[40] Cong X. Kang, Ismael G. Yero, and Eunjeong Yi. The fractional strong metric dimension in three graph products. *Discrete Applied Mathematics*, 251:190–203, 2018.

[41] Mark G. Karpovsky, Krishnendu Chakrabarty, and Lev B. Levitin. On a new class of codes for identifying vertices in graphs. *IEEE Transactions on Information Theory*, 44(2):599–611, 1998.

[42] Aleksander Kelenc, Niko Tratnik, and Ismael G. Yero. Uniquely identifying the edges of a graph: the edge metric dimension. *Discrete Applied Mathematics*, 251:204–220, 2018.

[43] Tero Laihonen and Tuomo Lehtilä. Improved codes for list decoding in the levenshtein's channel and information retrieval. In *Proceedings 2017 IEEE International Symposium on Information Theory (ISIT)*, pages 2643–2647, 2017.

[44] Vladimir I. Levenshtein. Efficient reconstruction of sequences. *IEEE Transactions on Information Theory*, 47(1):2–22, 2001.

[45] Antoine Lobstein. Watching systems, identifying, locating-dominating and discriminating codes in graphs: a bibliography. *Published electronically at https://www.lri.fr/%7Elobstein/debutBIBidetlocdom.pdf.*

[46] Julien Moncel. Monotonicity of the minimum cardinality of an identifying code in the hypercube. *Discrete Applied Mathematics*, 154(6):898–899, 2006.

[47] Mikko Pelto. On (r,$\leq$ 2)-locating-dominating codes in the infinite king grid. *Advances in Mathematics of Communications*, 6(1):27–38, 2012.

[48] Douglas F. Rall and Peter J. Slater. On location-domination numbers for certain classes of graphs. *Congressus Numerantium*, 45:97–106, 1984.

[49] Douglas F. Rall and Kirsti Wash. Identifying codes of the direct product of two cliques. *European Journal of Combinatorics*, 36:159–171, 2014.

[50] Saikat Ray, David Starobinski, Ari Trachtenberg, and Rachanee Ungrangsi. Robust location detection with sensor networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1016–1025, 2004.

[51] Alfréd Rényi. On random generating elements of a finite boolean algebra. *Acta Scientiarum Mathematicarum Szeged*, 22:75–81, 1961.

[52] J. Louis Sewell and Peter J. Slater. A sharp lower bound for locating-dominating sets in trees. *Australasian Journal of Combinatorics*, 60(2):136–149, 2014.

[53] Peter J. Slater. Leaves of trees. In *Proceedings of the Sixth Southeastern Conference on Combinatorics, Graph Theory, and Computing (Florida Atlantic Univ., Boca Raton, Fla., 1975). Congressus Numerantium, No. XIV, Winnipeg, Man.*, pages 549–559, 1975.

[54] Peter J. Slater. Domination and location in acyclic graphs. *Networks*, 17(1):55–64, 1987.

[55] Peter J. Slater. Dominating and reference sets in a graph. *Journal of Mathematical and Physical Sciences*, 22(4):445–455, 1988.

[56] Peter J. Slater. Fault-tolerant locating-dominating sets. *Discrete Mathematics*, 249(1-3):179–189, 2002.

[57] Jacobus H. Van Lint. *Introduction to Coding Theory.* Graduate Texts in Mathematics. Springer, expanded third edition, 1999.

[58] Eitan Yaakobi and Jehoshua Bruck. On the uncertainty of information retrieval in associative memories. In *Proceedings of 2012 IEEE International Symposium on Information Theory (ISIT)*, pages 106–110, 2012.

[59] Eitan Yaakobi and Jehoshua Bruck. On the uncertainty of information retrieval in associative memories. *IEEE Transactions on Information Theory*, 65(4):2155–2165, 2018.

Part II

Original publications

Publication I

Ville Junnila, Tero Laihonen and Tuomo Lehtilä. On regular and new types of codes for location-domination, *Discrete Applied Mathematics*, 247:225–241, 2018.
https://doi.org/10.1016/j.dam.2018.03.050

Discrete Applied Mathematics 247 (2018) 225–241

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

On regular and new types of codes for location-domination

Ville Junnila, Tero Laihonen, Tuomo Lehtilä *

Department of Mathematics and Statistics, University of Turku, FI-20014 Turku, Finland

ARTICLE INFO

Article history:
Received 4 May 2017
Received in revised form 12 March 2018
Accepted 21 March 2018
Available online 13 April 2018

Keywords:
Locating-dominating set
Locating-dominating code
Rook's graph
Hamming space
Sensor network

ABSTRACT

Identifying codes and locating-dominating codes have been designed for locating irregularities in sensor networks. In both cases, we can locate only one irregularity and cannot even detect multiple ones. To overcome this issue, self-identifying codes have been introduced which can locate one irregularity and detect multiple ones. In this paper, we define two new classes of locating-dominating codes which have similar properties. These new locating-dominating codes as well as the regular ones are then more closely studied in the rook's graphs and binary Hamming spaces.

In the rook's graphs, we present optimal codes, i.e., codes with the smallest possible cardinalities, for regular location-domination as well as for the two new classes. In the binary Hamming spaces, we present lower bounds and constructions for the new classes of codes; in some cases, the constructions are optimal. Moreover, one of the obtained lower bounds improves the bound of Honkala et al. (2004) on codes for locating multiple irregularities.

Besides studying the new classes of codes, we also present record-breaking constructions for regular locating-dominating codes. In particular, we present a locating-dominating code in the binary Hamming space of length 11 with 320 vertices improving the earlier bound of 352; the best known lower bound for such code is 309 by Honkala et al. (2004).

1. Introduction

Sensor networks are systems designed for environmental monitoring. Various location detection systems such as fire alarm and surveillance systems can be viewed as examples of sensor networks. For location detection, a sensor can be placed in any location of the network. The sensor monitors its neighbourhood (including the location of the sensor itself) and reports possible irregularities such as a fire or an intruder in the neighbouring locations. Based on the reports of the sensors, a central controller attempts to determine the location of a possible irregularity in the network. Usually, the aim is to minimize the number of sensors in the network. More explanation regarding location detection in sensor networks can be found in [4,12,16].

A sensor network can be modelled as a simple and undirected graph $G = (V(G), E(G)) = (V, E)$ as follows: the set of vertices V of the graph represents the locations of the network and the edge set E of the graph represents the connections between the locations. In other words, a sensor can be placed in each vertex of the graph and the sensor placed in the vertex u monitors u itself and the vertices neighbouring u. Besides being simple and undirected, we assume that the graphs in this paper are connected and have order of at least two. In what follows, we present some basic terminology and notation regarding graphs. The *open neighbourhood* of $u \in V$ consists of the vertices adjacent to u and it is denoted by $N(u)$. The *closed neighbourhood* of u is defined as $N[u] = \{u\} \cup N(u)$. A nonempty subset C of V is called a *code* and the elements of the code

---

* Corresponding author.
E-mail addresses: viljun@utu.fi (V. Junnila), terolai@utu.fi (T. Laihonen), tualeh@utu.fi (T. Lehtilä).

https://doi.org/10.1016/j.dam.2018.03.050

are called *codewords*. In this paper, the code C represents the set of locations where the sensors have been placed on. For the set of sensors monitoring a vertex $u \in V$, we use the following notation:

$$I(u) = N[u] \cap C.$$

In order to emphasize the graph G and/or the code C, we sometimes write $I(u) = I(C; u) = I(G, C; u)$. We call $I(u)$ the *identifying set* (or the *I-set*) of u. The notation of identifying set can also be generalized for a subset U of V as follows:

$$I(U) = \bigcup_{u \in U} I(C; u).$$

Here we also use the notation $I(U) = I(C; U) = I(G, C; U)$.

As stated above, a sensor $u \in V$ reports that an irregularity has been detected if there is (at least) one in the closed neighbourhood $N[u]$. In what follows, we divide into two different situations depending on the capability of a sensor to distinguish whether the irregularity has been spotted in the location of the sensor itself or in its (open) neighbourhood. More precisely, we have the following two cases:

(i) In the first case, we assume that a sensor $u \in V$ reports 1 if there is an irregularity in $N[u]$, and otherwise it reports 0.
(ii) In the second case, we assume that a sensor $u \in V$ reports 2 if there is an irregularity in u, it reports 1 if there is one in $N(u)$ (and none in u itself), and otherwise it reports 0.

Assume first that the sensors work as in (i). Notice then that if the sensors in the code C are located in such places that $I(C; u)$ is nonempty and unique for all $u \in V$, then an irregularity in the network can be located by comparing $I(C; u)$ to identifying sets of other vertices. This leads to the following definition of *identifying codes*, which were first introduced by Karpovsky et al. in [11]. For various papers regarding identification and related problems, we refer to the online bibliography [13].

Definition 1. A code $C \subseteq V$ is *identifying* in G if for all distinct $u, v \in V$ we have $I(C; u) \neq \emptyset$ and

$$I(C; u) \neq I(C; v).$$

An identifying code C in a finite graph G with the smallest cardinality is called *optimal* and the number of codewords in an optimal identifying code is denoted by $\gamma^{ID}(G)$.

Let C be an identifying code in G. By the definition, the identifying code C works correctly if there is simultaneously at most one irregularity in the network. However, using the identifying code C, we cannot locate or even detect more than one irregularity in the network. Indeed, for example, consider the graph G in Fig. 1 and the code $C = \{a, b, c\}$ in the graph. Clearly, C is an identifying code in G. However, all the sensors a, b and c are alarming if there is a single irregularity in b, or multiple ones in d, e and f. Hence, no distinction can be made between these two cases. Thus, we might determine a false location and more disturbingly not even notice that something is wrong. To overcome this problem, in [7], self-identifying codes, which are able to locate one irregularity and detect multiple ones, were introduced. (Notice that in the original paper self-identifying codes are called 1^+-identifying.) The formal definition of self-identifying codes is given as follows.

Definition 2. A code $C \subseteq V$ is called *self-identifying* in G if the code C is identifying in G and for all $u \in V$ and $U \subseteq V$ such that $|U| \geq 2$ we have

$$I(C; u) \neq I(C; U).$$

A self-identifying code C in a finite graph G with the smallest cardinality is called *optimal* and the number of codewords in an optimal self-identifying code is denoted by $\gamma^{SID}(G)$.

In addition to [7], self-identifying codes have also been previously discussed in [9,10]. Separately in these papers, two useful characterizations have been presented for self-identifying codes.

Theorem 3 (*[7,9,10]*)**.** *Let C be a code in G. Then the following statements are equivalent:*

(i) *The code C is self-identifying in G.*
(ii) *For all distinct $u, v \in V$, we have $I(C; u) \setminus I(C; v) \neq \emptyset$.*
(iii) *For all $u \in V$, we have $I(C; u) \neq \emptyset$ and*

$$\bigcap_{c \in I(C;u)} N[c] = \{u\}.$$

As stated earlier, self-identifying codes can locate one irregularity and detect multiple ones. Besides that, the characterization (iii) of the previous theorem also gives another useful property for self-identifying codes. Namely, the location of an irregularity can be determined without comparison to other identifying sets, since for all $u \in V$ the neighbourhoods of the codewords in $I(u)$ intersect uniquely in u.

(a) Self-locating-dominating code in G.

(b) Solid-locating-dominating code in G.

Fig. 1. Optimal self-locating-dominating and solid-locating-dominating codes in G.

So far, we have discussed the case where it is assumed that each sensor outputs 1 or 0 depending on whether there is an irregularity in the neighbourhood or not. In what follows, we now focus on the case (ii) where a sensor can also distinguish if the irregularity is on the location of the sensor itself. Then notice that if the sensors in the code C are located in such places that $I(C;u)$ is nonempty and unique for all $u \in V \setminus C$, then an irregularity in the network can be located by comparing $I(C;u)$ to identifying sets of other non-codewords. Indeed, we do not have to worry about vertices in C as an irregularity in such locations is immediately determined by a sensor outputting 2. This leads to the following definition of *locating-dominating codes*, which were first introduced by Slater in [15,17,18].

Definition 4. A code $C \subseteq V$ is *locating-dominating* in G if for all distinct $u, v \in V \setminus C$ we have $I(C;u) \neq \emptyset$ and

$$I(C;u) \neq I(C;v).$$

A locating-dominating code C in a finite graph G with the smallest cardinality is called *optimal* and the number of codewords in an optimal locating-dominating code is denoted by $\gamma^{LD}(G)$.

Comparing the definitions of identifying and locating-dominating codes, we immediately notice their apparent similarities; in the case of identification we require that the identifying sets $I(u)$ are unique for all vertices and in the case of location-domination the same is required for non-codewords. Therefore, as self-identifying codes are a natural specialization of regular identifying codes, it is obvious to consider if something similar could be done for locating-dominating codes. Indeed, the characterizations of Theorem 3 give two natural ways to define new types of locating-dominating codes with similar kind of beneficial properties as self-identifying codes have over regular identifying codes. The definitions of these codes are given as follows.

Definition 5. A code $C \subseteq V$ is *self-locating-dominating* in G if for all $u \in V \setminus C$ we have $I(C;u) \neq \emptyset$ and

$$\bigcap_{c \in I(C;u)} N[c] = \{u\}.$$

A self-locating-dominating code C in a finite graph G with the smallest cardinality is called *optimal* and the number of codewords in an optimal self-locating-dominating code is denoted by $\gamma^{SLD}(G)$.

Definition 6. A code $C \subseteq V$ is *solid-locating-dominating* in G if for all distinct $u, v \in V \setminus C$ we have

$$I(C;u) \setminus I(C;v) \neq \emptyset.$$

A code $C \subseteq V$ is dominating if $I(C;u) \neq \emptyset$ for all $u \in V$. Since G is a connected graph on at least two vertices, a solid-locating-dominating code is also dominating. A solid-locating-dominating code C in a finite graph G with the smallest cardinality is called *optimal* and the number of codewords in an optimal solid-locating-dominating code is denoted by $\gamma^{DLD}(G)$.

The previous definitions are illustrated in the following example. In particular, we show that the given definitions are different. Compare this observation to self-identifying codes for which the analogous requirements are just other characterizations for the codes.

Example 7. Let G be the graph illustrated in Fig. 1. Let C be a self-locating-dominating code in G. Observe first that if $a \notin C$, then $I(C;a) \subseteq \{b, d\}$ and we have

$$\{a, e\} \subseteq \bigcap_{c \in I(C;a)} N[c].$$

This implies a contradiction and therefore the vertex a belongs to C. An analogous argument also holds for the vertices c, d and f. Hence, we have $\{a, c, d, f\} \subseteq C$. Moreover, the code $C_1 = \{a, c, d, f\}$, which is illustrated in Fig. 1(a), is self-locating-dominating in G since for the non-codewords b and e we have $I(C_1;b) = \{a, c\}$ and $N[a] \cap N[c] = \{b\}$, and $I(C_1;e) = \{d, f\}$ and $N[d] \cap N[f] = \{e\}$. Hence, C_1 is an optimal self-locating-dominating code in G and we have $\gamma^{SLD}(G) = 4$.

Let us then consider the code $C_2 = \{a, b, c\}$, which is illustrated in Fig. 1(b). Now we have $I(C_2; d) = \{a\}$, $I(C_2; e) = \{b\}$ and $I(C_2; f) = \{c\}$. Therefore, it is easy to see that C_2 is a solid-locating-dominating code in G. Moreover, it can be shown that there are no solid-locating-dominating codes in G with smaller number of codewords. Thus, C_2 is an optimal solid-locating-dominating code in G and we have $\gamma^{DLD}(G) = 3$.

In the previous example, we showed that the definitions of self-locating-dominating and solid-locating-dominating codes are different. In the following theorem, we present new characterizations for self-locating-dominating and solid-locating-dominating codes. Comparing these characterizations to the original definitions of the codes, the differences of the codes become apparent.

Theorem 8. *Let $G = (V, E)$ be a connected graph on at least two vertices:*

(i) *A code $C \subseteq V$ is self-locating-dominating if and only if for all distinct $u \in V \setminus C$ and $v \in V$ we have*

$$I(C; u) \setminus I(C; v) \neq \emptyset.$$

(ii) *A code $C \subseteq V$ is solid-locating-dominating if and only if for all $u \in V \setminus C$ we have $I(C; u) \neq \emptyset$ and*

$$\left(\bigcap_{c \in I(C;u)} N[c] \right) \setminus C = \{u\}.$$

Proof. (i) Assume first to the contrary that there exist $u \in V \setminus C$ and $v \in V$ such that $I(C; u) \setminus I(C; v) = \emptyset$. This implies that $I(C; u) \subseteq I(C; v)$ and we have a contradiction as

$$\{u, v\} \subseteq \bigcap_{c \in I(C;u)} N[c].$$

On the other hand, if there exists $u \in V \setminus C$ such that

$$\{u, v\} \subseteq \bigcap_{c \in I(C;u)} N[c]$$

for some $v \in V$, then $I(C; u) \setminus I(C; v) = \emptyset$ (a contradiction).

(ii) Assume first to the contrary that there exist $u, v \in V \setminus C$ such that $I(C; u) \setminus I(C; v) = \emptyset$. This implies that $I(C; u) \subseteq I(C; v)$ and we have a contradiction as

$$\{u, v\} \subseteq \left(\bigcap_{c \in I(C;u)} N[c] \right) \setminus C.$$

On the other hand, if there exists $u \in V \setminus C$ such that

$$\{u, v\} \subseteq \left(\bigcap_{c \in I(C;u)} N[c] \right) \setminus C$$

for some $v \in V \setminus C$, then $I(C; u) \setminus I(C; v) = \emptyset$ (a contradiction). □

The previous theorem immediately gives the following corollary.

Corollary 9. *If C is a self-locating-dominating code in G, then C is also solid-locating-dominating in G. Furthermore, we have $\gamma^{DLD}(G) \leq \gamma^{SLD}(G)$.*

As discussed earlier, self-identifying codes have benefits over regular identifying codes; they detect more than one irregularity and locate one irregularity without comparison to other identifying sets. Next we study the same properties concerning self-locating-dominating and solid-locating-dominating codes:

- Let us begin by considering the ability to locate an irregularity without comparison to other identifying sets. For self-locating-dominating codes, this property immediately follows from the definition. Analogously, the property is obtained for solid-locating-dominating codes by Theorem 8(ii).
- Consider then the ability to detect more than one irregularity. Let first C be a self-locating-dominating code in G. If more than one sensor is reporting 2, then we immediately detect that there are multiple irregularities. Hence, we may assume that there is at most one sensor reporting 2. Let U be the set of sensors reporting 1 (U can be empty). Consider then the intersection

$$X = \bigcap_{c \in U} N[c].$$

Here we assume that $X = V$ if the set U of sensors reporting 1 is empty. Now, by the definition of self-locating-dominating codes (as at most one sensor is reporting 2), there are multiple irregularities if and only if the intersection X is empty, or a sensor reporting 2 does not belong to X. Indeed, if $X = \emptyset$ or a sensor reporting 2 does not belong to X, then there are clearly multiple irregularities. On the other hand, if there is an irregularity in a location u with a sensor and at least one without a sensor, then $X = \emptyset$ or $u \notin X$, and if there is no irregularity in a location with a sensor and at least two without a sensor, then $X = \emptyset$. Thus, self-locating-dominating codes can detect multiple irregularities.

On the other hand, solid-locating-dominating codes do not always detect multiple irregularities. For a counterexample, consider the graph G and the solid-locating-dominating code C_2 of Example 7. If the vertex b is reporting 2 and the vertices a and c are reporting 1, then there might be a single irregularity in b or multiple irregularities in b, d and f. However, if it is assumed that the irregularities occur only in the locations without a sensor, then we can detect multiple irregularities using similar arguments as in the case of self-locating-dominating codes.

In the paper our main focus is on the new types of locating-dominating codes. However, we also present some results for regular locating-dominating codes. In Section 2, we consider the different types of locating-dominating codes in the Cartesian product of two complete graphs, which is also called the rook's graph. In particular, we obtain optimal codes for regular location-domination, self-location-domination and solid-location-domination. In Section 3, we consider similar problems in the binary Hamming space (or hypercube) $\mathbb{F}^n$, where n is a positive integer. In particular, we present an infinite family of optimal self-locating-dominating codes and construct regular locating-dominating codes with the smallest known cardinalities; especially proving that $309 \leq \gamma^{LD}(\mathbb{F}^{11}) \leq 320$. Moreover, our bound in Theorem 20 on solid-locating-dominating codes implies an improvement on Honkala et al. bound in [8], see Remark 21.

2. Location-domination in the rook's graphs

In this section, we consider the different locating-dominating codes in the Cartesian product of two complete graphs. The Cartesian product of graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ is $G_1 \square G_2 = (V_1 \times V_2, E)$ where $(x, y)(x', y') \in E$ if and only if $x = x'$ and $yy' \in E_2$, or $y = y'$ and $xx' \in E_1$. If K_n and K_m are two complete graphs of order n and m, respectively, then $K_n \square K_m$ is known as rook's graph and can be viewed as a chess board with m rows and n columns. The closed neighbourhood of a vertex is determined by the movement of a rook in chess. We denote $V(K_n) = \{x_1, \dots, x_n\}$, $V(K_m) = \{y_1, \dots, y_m\}$ and the kth *row* by $R_k = \{(x_i, y_k) \mid i = 1, \dots, n, 1 \leq k \leq m\}$ (resp. the hth *column* by $P_h = \{(x_h, y_i) \mid i = 1, \dots, m, 1 \leq k \leq n\}$).

In our considerations, the columns are ordered from left to right and the rows from bottom to top. However, we will occasionally permute the order of rows and/or columns. If C is a locating-dominating, self-locating-dominating or solid-locating-dominating code, it will also be such code in the graph gained through these permutations since all neighbourhoods remain the same after these permutations. Previously, in [5,6,9] and [10], optimal codes have been respectively found for identification and self-identification in the rook's graphs. These results are combined in the following theorem.

Theorem 10 ([5,6,9]). *Let $G = K_n \square K_m$ be a rook's graph with $m \geq n \geq 2$. We have the following formulas for the sizes of optimal identifying and self-identifying codes in $K_n \square K_m$:*

$$\gamma^{ID}(K_n \square K_m) = \begin{cases} m + \left\lfloor \frac{n}{2} \right\rfloor, & m \leq \frac{3n}{2} \\ 2m - n, & m \geq \frac{3n}{2} \end{cases} \quad \text{and}$$

$$\gamma^{SID}(K_n \square K_m) = 2m, \ m \geq n.$$

In what follows, we are going to find optimal locating-dominating, self-locating-dominating and solid-locating-dominating codes in the rook's graphs. For this purpose, we first introduce the following helpful lemma.

Lemma 11. *For $v = (x_i, y_j) \in V(K_n \square K_m)$ the following statements hold:*

(i) *If $|I(v)| = 2$ and vertices in $I(v)$ are not on the same row or column, then $|\bigcap_{c \in I(v)} N[c]| = 2$.*
(ii) *If $|I(v)| \geq 2$ and vertices in $I(v)$ are on the same row, then $\bigcap_{c \in I(v)} N[c] = R_j$.*
(iii) *If $|I(v)| \geq 2$ and vertices in $I(v)$ are on the same column, then $\bigcap_{c \in I(v)} N[c] = P_i$.*
(iv) *If $|I(v)| = 3$ and all vertices in $I(v)$ are not on the same row or column, then $\bigcap_{c \in I(v)} N[c] = v$.*

Proof. Let $v = (x_i, y_j)$.

(i) If we have $I(v) = \{(x_i, y_{j'}), (x_{i'}, y_j)\}$, when $i \neq i'$ and $j \neq j'$, then $\bigcap_{c \in I(v)} N[c] = \{(x_i, y_j), (x_{i'}, y_{j'})\}$.

(ii) If we have $\{(x_{i'}, y_j), (x_{i''}, y_j)\} \subseteq I(v)$, when $i' \neq i''$, then $\bigcap_{c \in I(v)} N[c] = R_j$.

(iii) If we have $\{(x_i, y_{j'}), (x_i, y_{j''})\} \subseteq I(v)$, when $j' \neq j''$, then $\bigcap_{c \in I(v)} N[c] = P_i$.

(iv) Without loss of generality, we may assume that there are two codewords in the same column as v and one in the same row. Hence we have $I(v) = \{(x_i, y_{j'}), (x_i, y_{j''}), (x_{i'}, y_j)\}$ for some i, j with $i \neq i'$ and $j' \neq j''$. This implies that $\bigcap_{c \in I(v)} N[c] = \{(x_i, y_j)\}$. □

Let us first consider self-locating-dominating codes. We will give optimal cardinality of such codes in the next theorem.

Theorem 12. *Let $G = K_n \square K_m$ be a rook's graph with $m \geq n \geq 1$. We have*

$$\gamma^{SLD}(G) = \begin{cases} m, & m \geq 2n, \text{ or } n = 1, \\ 2n, & 2n > m > n \geq 2, \\ 2n-1, & m = n > 2, \\ 4, & n = m = 2. \end{cases}$$

Proof. Let $V(K_n) = \{x_1, x_2, \ldots, x_n\}$, $V(K_m) = \{y_1, y_2, \ldots, y_m\}$ and $C \subseteq V(G)$ be an optimal self-locating-dominating code in G.

- **Fact 1**: Lemma 11 gives that for each non-codeword $v \in V(G)$ we have $|I(C; v)| \geq 3$ and we know that there are no rows or columns empty of codewords.

Let us first consider the case where $\mathbf{m \geq 2n}$. The fact 1 tells us that there has to be at least one codeword on each row. Hence, we get $\gamma^{SLD}(G) \geq m$. The condition $m \geq 2n$ includes most of the cases under $n = 1$ and the case $n = m = 1$ is clear.

By selecting as our code

$$C_1 = \{(x_i, y_j) \in V(G) \mid i - j \equiv 0 \bmod n\}$$

we get $|C_1| = m$. Since $|C_1| = m \geq 2n$, there is exactly one codeword on each row and, as there are n vertices on each row, there are at least two vertices on each column. Therefore, we have at least three vertices which are not in the same row or column in each I-set of a non-codeword. Now we get from Lemma 11 that C_1 is a self-locating-dominating code.

Let us now consider the case where $\mathbf{2n > m > n}$. If we had $|C| \leq 2n - 1 \leq 2m - 3$, then there would be a column P_i and at least two rows R_j and $R_{j'}$ with only one codeword (or an empty row or column). Now at least one of the vertices (x_i, y_j) and $(x_i, y_{j'})$ is not a codeword. We can assume that $(x_i, y_j) \notin C$, now we have $|I(x_i, y_j)| = 2$ and based on the fact 1 C cannot be a self-locating-dominating code. Thus, we have $\gamma^{SLD}(G) \geq 2n$.

If we choose

$$C_2 = \{(x_i, y_j) \in V(G) \mid j - i = 0 \text{ or } j - i = m - n\},$$

we get $|C_2| = 2n$ and there are two vertices on each column and at least one vertex on each row. Therefore, we have at least three vertices which are not in the same row or column in each I-set of a non-codeword. Thus, based on Lemma 11, C_2 is a self-locating-dominating code. In Fig. 2 code C_2 is illustrated for $K_5 \square K_7$.

Let us consider the case where $\mathbf{m = n > 2}$. If we had $|C| \leq 2n-2 = 2m-2$, then there would be at least two columns and rows with only one codeword (or an empty row or column). Hence, we can again choose a non-codeword v with $|I(v)| = 2$ and fact 1 tells that C cannot be a self-locating-dominating code. Now we have $\gamma^{SLD}(G) \geq 2n - 1$.

If we choose

$$C_3 = \{(x_i, y_j) \in V(G) \mid i - j = 0, i - j = 1 \text{ or } (i, j) = (2, n)\} \setminus \{(x_2, y_1)\},$$

we have two vertices on each row and column except for R_1 and P_1. But since $R_1 \cap P_1 = (x_1, y_1) \in C_3$ each intersection of a row and column with a single codeword belongs to code C_3. Thus, for each non-codeword $v \in V(G)$, we have at least three vertices in $I(v)$ and they are not all in the same row or column. Now Lemma 11 says that C_3 is a self-locating-dominating code. We also have $|C_3| = 2n - 1$. In Fig. 3 code C_3 is illustrated for $K_6 \square K_6$.

Let us finally consider the case $\mathbf{m = n = 2}$. If we have only three codewords in C, then the I-set of the non-codeword contains only two codewords and the intersection of their neighbourhoods contains two words. On the other hand, the whole graph only contains four vertices so we have $\gamma^{SLD}(K_2 \square K_2) = 4$. □

We will see in the next theorem that optimal solid-locating-dominating codes are mostly of the same size as optimal self-locating-dominating codes. However, this is only a superficial similarity. It will be seen in the proof that the structures of solid-locating-dominating codes vary more and there are more of them. For example, the codes in Figs. 4 and 5 are optimal solid-locating-dominating codes for $K_5 \square K_6$ and $K_5 \square K_5$. However, they are not self-locating-dominating codes.

Theorem 13. *Let $G = K_n \square K_m$ be a rook's graph with $m \geq n \geq 1$. We have*

$$\gamma^{DLD}(G) = \begin{cases} m, & m \geq 2n \geq 4 \text{ or } n = 2, \\ 2n, & 2n > m > n > 2, \\ 2n-1, & m = n > 2, \\ m-1, & m > n = 1. \end{cases}$$

Proof. Let $V(K_n) = \{x_1, x_2, \ldots, x_n\}$, $V(K_m) = \{y_1, y_2, \ldots, y_m\}$ and $C \subseteq V(G)$ be an optimal solid-locating-dominating code.

If there is a row R_i such that $R_i \cap C = \emptyset$ and there are no vertices with empty I-sets, then $P_j \subseteq \bigcap_{c \in I(x_j, y_i)} N[c]$ for each j. Thus, $C = V(G) \setminus R_i$.

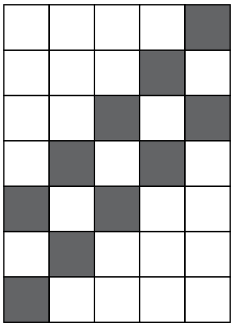


Fig. 2. Optimal self-locating-dominating code for $K_5 \square K_7$.

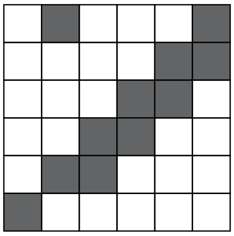


Fig. 3. Optimal self-locating-dominating code for $K_6 \square K_6$.

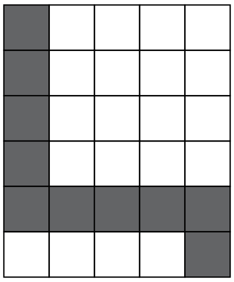


Fig. 4. Optimal solid-locating-dominating code for $K_5 \square K_6$.

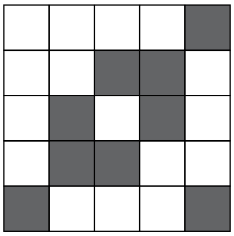


Fig. 5. Optimal solid-locating-dominating code for $K_5 \square K_5$.

- **Fact 2**: The code C, $|C| < m(n-1)$, is not a solid-locating-dominating code if there is a row or a column without any codewords.

Let us first consider the case where $\mathbf{m \geq 2n \geq 4}$. Fact 2 says that there has to be at least one codeword on each row so we have $\gamma^{DLD}(G) \geq m$. We also have $\gamma^{DLD}(G) \leq \gamma^{SLD}(G) = m$ by Theorem 12. Hence, $\gamma^{DLD}(G) = m$.

Let us next consider the case where $\mathbf{2n > m > n > 2}$. Theorem 12 gives an upper bound $\gamma^{DLD}(n) \leq \gamma^{SLD}(n) = 2n$. Let $|C| \leq 2n - 1 \leq 2m - 3$. Without loss of generality we may assume that the rows with a single codeword are consecutive rows and numbered as the first ones in the notation R_i and the same is true for the columns P_i. Denote the rows and columns

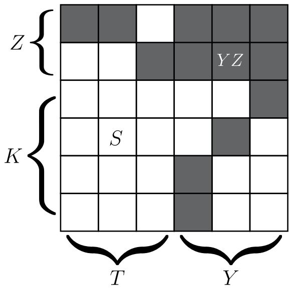


Fig. 6. Part of a solid-locating-dominating code in a rook's graph with $Y = 3, Z = 2, T = 3, K = 4$.

which contain exactly one codeword as follows:

$$\mathcal{K} = \{R_1, \dots, R_K\} \text{ and } \mathcal{T} = \{P_1, \dots, P_T\}.$$

Since there are no empty rows or columns by fact 2 and $|C| \leq 2n-1 \leq 2m-3$, we have $K \geq 3$ and $T \geq 1$. Hence, $K+T \geq 4$.

Let us denote codewords on column P_i as (x_i, y_{s_i}), where $1 \leq i \leq T$ and codewords on row R_j as (x_{h_j}, y_j), where $1 \leq j \leq K$. Let us further denote

$$S = \{(x_i, y_j) \mid 1 \leq i \leq T, 1 \leq j \leq K\}.$$

If we have a codeword, say $(x_t, y_{s_t}) \in S$, then at least one of the vertices (x_1, y_{s_t}) or (x_2, y_{s_t}) is a non-codeword, say (x_1, y_{s_t}). Now $I(x_1, y_{s_t}) = \{(x_t, y_{s_t}), (x_1, y_{s_1})\}$ and $s_1 \neq s_t$. Furthermore, $\bigcap_{c \in I(x_1, y_{s_t})} N[c] = \{(x_t, y_{s_1}), (x_1, y_{s_t})\}$. Since column P_t has only one codeword neither of vertices $(x_t, y_{s_1}), (x_1, y_{s_t})$ belongs to C and C is not a solid-locating-dominating code. Thus, no vertex in S can be a codeword.

For each vertex $(x_i, y_j) \in S$ we have $I(x_i, y_j) = \{(x_{h_j}, y_j), (x_i, y_{s_i})\}$. Hence, in order for C to be a solid-locating-dominating code, vertex (x_{h_j}, y_{s_i}) has to be a codeword if $(x_i, y_j) \in S$. We can assume that codewords (x_{h_j}, y_j) in $\mathcal{K}$ are located in Y different columns and codewords (x_i, y_{s_i}) in $\mathcal{T}$ are located in Z different rows. Now each of the YZ vertices (x_{h_j}, y_{s_i}) has to be a codeword (see Fig. 6).

Let a be a positive integer. Observe that if we have more than two codewords in a row, say $2+a$ codewords, then there are a rows with exactly one codeword since we have $|C| \leq 2n-1$. The same is also true for columns. Hence we have at least $3+T+YZ-2Z$ rows with one codeword due to rows with multiple codewords since we have $T+YZ$ codewords on Z rows. Similarly we see that we have at least $1+K+YZ-2Y$ columns with one codeword due to columns with multiple codewords.

Thus, we get the following inequality which implies a contradiction:

$$K+T \geq 4+(K+YZ-2Y)+(T+YZ-2Z) = 4+K+T+2(YZ-Y-Z) \geq K+T+2.$$

The latter inequality is due to the fact that $YZ-Y-Z \geq -1$, when $Y, Z \geq 1$. Therefore, we have $|C| \geq 2n$.

Let us then consider the case $\mathbf{m} = \mathbf{n} > \mathbf{2}$. If $|C| \leq 2n-2 = 2m-2$, then $K \geq 2, T \geq 2, K+T \geq 4$ and $Y, Z \geq 1$. Now as in the previous case, there are no codewords in S, there is a $Y \times Z$ rectangle filled with codewords and we gain the same contradiction with similar reasoning. Hence the same arguments also apply here and we have $|C| \geq 2n-1$. Since $\gamma^{DLD}(G) \leq \gamma^{SLD}(G) = 2n-1$, we have $\gamma^{DLD}(G) = 2n-1$.

As the next case we consider $\mathbf{n} = \mathbf{2}$. If $|C| < m$, then there is a row without a codeword. On the other hand we can choose $C = P_1$ as our code. Thus $|C| = m$.

Finally as the last case we consider $\mathbf{m} > \mathbf{n} = \mathbf{1}$. If $|C| < m-1$, then there are two non-codewords with I-set equal to C and so C is not a solid-locating-dominating code. On the other hand if $C = V(G) \setminus \{v\}$, then $I(v) = C$ and $I(v)$ is unique as the only I-set of non-codeword. □

Finally, in the following theorem, we construct optimal locating-dominating codes in rook's graphs.

Theorem 14. *Let $G = K_n \square K_m$ be a rook's graph with $m \geq n \geq 1$. We have*

$$\gamma^{LD}(G) = \begin{cases} m-1, & m \geq 2n, \\ \left\lceil \dfrac{2n+2m}{3} \right\rceil - 1, & n \leq m \leq 2n-1. \end{cases}$$

Proof. Let $V(K_n) = \{x_1, x_2, \dots, x_n\}$, $V(K_m) = \{y_1, y_2, \dots, y_m\}$, $\gamma = \gamma^{LD}(K_n \square K_m)$ and $C \subseteq V(G)$ be an optimal locating-dominating code.

We first observe that if there are two rows R_i and R_j without codewords, then vertices (x_1, y_i) and (x_1, y_j) have the same I-set. The case for two columns without codewords is similar. If we have a row R_j and a column P_i without codewords, then $I(x_i, y_j) = \emptyset$.

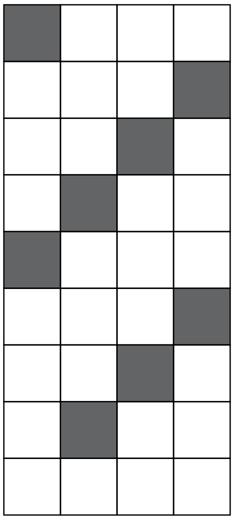


Fig. 7. Optimal locating-dominating code for $K_4 \square K_9$.

- **Fact 3**: The total number of rows and columns without codewords in G is less than two.

Let us consider the case $\mathbf{m} \geq \mathbf{2n}$. If we had $|C| \leq m - 2$, we would have at least two rows empty of codewords. Hence we have $\gamma > m - 2$. We can choose

$$C_1 = \{(x_i, y_j) \mid i - j \equiv 0 \bmod n, j \geq 2\}.$$

The code C_1 is illustrated in Fig. 7 for values $n = 4$ and $m = 9$. Each non-codeword, which is not located in R_1 or P_1, has at least two codewords on the same column and one codeword on the same row. Hence, they are uniquely determined by Lemma 11. If the vertex is located in R_1, then $I(x_i, y_1) = \{(x_i, y_{i+kn}) \mid i + kn \in [2, m], k \in \mathbb{Z}\}$, which is a unique I-set since all other I-sets contain a codeword from two different columns. For each vertex $(x_1, y_j), j > 1$, we have (x_1, y_{n+1}) in its I-set and some codeword from the row R_j. Hence vertices on P_1 have unique I-set. Note that column P_1 has to be considered only when $m = 2n$.

Let us then consider the case $\mathbf{n} \leq \mathbf{m} < \mathbf{2n}$. Let

- s_p denote the number of columns with exactly one codeword,
- s_{p0} denote the number of columns without codewords,
- s_r denote the number of rows with exactly one codeword and
- s_{r0} denote the number of rows without codewords.

We can assume that $s_{p0} + s_{r0} \leq 1$. If we had $s_{p0} + s_{r0} \geq 2$, then we would not have a locating-dominating code by the fact 3.

By counting the size of the code column by column, we get

$$\gamma \geq 0 \cdot s_{p0} + s_p + 2(n - s_p - s_{p0})$$

which gives us

$$s_p \geq 2n - \gamma - 2s_{p0}. \tag{1}$$

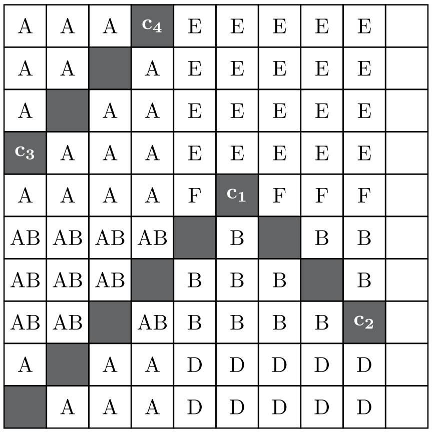
When we count the size of the code row by row, we get similarly

$$s_r \geq 2m - \gamma - 2s_{r0}. \tag{2}$$

If we have two codewords c_1 and c_2 with I-sets $I(c_i) = \{c_i\}$, $1 \leq i \leq 2$, then C is not a locating-dominating code. Let $c_1 = (x_{i_1}, y_{j_1})$ and $c_2 = (x_{i_2}, y_{j_2})$. Now $I(x_{i_2}, y_{j_1}) = I(x_{i_1}, y_{j_2}) = \{c_1, c_2\}$. Hence our s_r rows with exactly one codeword and s_p columns with exactly one codeword share at most one codeword. Now we get from inequalities (1) and (2)

$$\gamma \geq s_p + s_r - 1 \geq 2n + 2m - 2\gamma - 2(s_{p0} + s_{r0}) - 1,$$

$$\gamma \geq \left\lceil \frac{2(m+n)}{3} \right\rceil - 1.$$



A	A	A	c_4	E	E	E	E	E	
A	A	■	A	E	E	E	E	E	
A	■	A	A	E	E	E	E	E	
c_3	A	A	A	E	E	E	E	E	
A	A	A	A	F	c_1	F	F	F	
AB	AB	AB	AB	■	B	■	B	B	
AB	AB	AB	■	B	B	B	■	B	
AB	AB	■	AB	B	B	B	B	c_2	
A	■	A	A	D	D	D	D	D	
■	A	A	A	D	D	D	D	D	

Fig. 8. Optimal locating-dominating code for $K_{10}\square K_{10}$.

We can choose

$C_2 = A_1 \cup A_2 \cup A_3$, where

$$A_1 = \left\{(x_i, y_j) \mid i = j \leq \left\lfloor \frac{m+n}{3} \right\rfloor\right\}, A_2 = \left\{(x_i, y_j) \mid j - i = \left\lfloor \frac{m+n}{3} \right\rfloor\right\} \text{ and}$$

$$A_3 = \left\{(x_i, y_j) \mid i + j = 2\left\lfloor \frac{m+n}{3} \right\rfloor \text{ and } \left\lfloor \frac{m+n}{3} \right\rfloor + 1 \leq i \leq n-1\right\}$$

(and if $n = 2, m = 3$ we can choose $C_2 = P_1$). Now we have $|C_2| = \lfloor\frac{m+n}{3}\rfloor + (n - 1 - \lfloor\frac{m+n}{3}\rfloor) + (m - \lfloor\frac{m+n}{3}\rfloor) = \lceil\frac{2m+2n}{3}\rceil - 1$. In Fig. 8, we have an optimal locating-dominating code for $K_{10}\square K_{10}$. The labelling of areas in what will follow corresponds to that of the figure. Codeword c_1 has coordinates $\left(x_{\lfloor\frac{m+n}{3}\rfloor}, y_{\lfloor\frac{m+n}{3}\rfloor}\right)$, c_2 has coordinates $\left(x_{n-1}, y_{2\lfloor\frac{m+n}{3}\rfloor+1-n}\right)$, c_3 has coordinates $\left(x_1, y_{\lfloor\frac{m+n}{3}\rfloor+1}\right)$ and c_4 has coordinates $\left(x_{m-\lfloor\frac{m+n}{3}\rfloor}, y_m\right)$. The I-set of a vertex on P_n is the set of codewords on the same row as it is. All other I-sets have also vertices from different rows. Hence the vertices on P_n have unique I-set. The non-codewords on columns P_i, $1 \leq i \leq m - \lfloor\frac{m+n}{3}\rfloor$ have at least three codewords in their I-sets of which two are on the same column and at least one on the same row. Thus by Lemma 11 they have unique I-set. Let us denote the set of these vertices by A. The vertices in set B on rows R_j, $2\lfloor\frac{m+n}{3}\rfloor - n + 1 \leq j \leq \lfloor\frac{m+n}{3}\rfloor - 1$, have at least three codewords in their I-sets if they are not on column P_n. Out of these three codewords two are on the same row and at least one on the same column. Hence by Lemma 11 they have unique I-set.

The vertices D on (x_i, y_j), $m - \lfloor\frac{m+n}{3}\rfloor + 1 \leq i \leq n - 1$, $1 \leq j \leq 2\lfloor\frac{m+n}{3}\rfloor - n \leq m - \lfloor\frac{m+n}{3}\rfloor$, have codeword (x_j, y_j) in their I-set and one codeword from a different row. Thus by Lemma 11 there is only one other vertex which has these codewords in its I-set. However, the other vertex would have to be in A and vertices in A have unique I-set. Vertices E; (x_i, y_j), $m - \lfloor\frac{m+n}{3}\rfloor + 1 \leq i \leq n - 1$, $\lfloor\frac{m+n}{3}\rfloor + 1 \leq j \leq m$, have two vertices in their I-set, one of which is from columns P_i, $1 \leq i \leq m - \lfloor\frac{m+n}{3}\rfloor$ and the other one is from a different row. Hence by Lemma 11 the only other vertex that could have the same I-set is in A but vertices in A have unique I-set.

Finally the vertices F (x_i, y_j), $m - \lfloor\frac{m+n}{3}\rfloor + 1 \leq i \leq n - 1, j = \lfloor\frac{m+n}{3}\rfloor$, have two vertices in their I-sets. Codeword $(\lfloor\frac{m+n}{3}\rfloor, \lfloor\frac{m+n}{3}\rfloor)$ and a codeword from a different row R_l with $2\lfloor\frac{m+n}{3}\rfloor - n + 1 \leq l \leq \lfloor\frac{m+n}{3}\rfloor - 1$. The only other vertex that could share this I-set is located on rows R_l, but such vertices had at least three vertices in their I-sets. □

In conclusion, by the previous theorems, we determine the cardinalities of optimal locating-dominating, self-locating-dominating and solid-locating-dominating codes in all graphs $K_m\square K_n$.

3. Location-domination in the binary Hamming spaces

In this section, we consider self-locating-dominating and solid locating-dominating codes in binary Hamming spaces of length n. A binary Hamming space of length n is a graph with the vertex set $\mathbb{F}^n = \{0, 1\}^n$, and two vertices have an edge between them if they differ in exactly one coordinate. Vertices of $\mathbb{F}^n$ are called *words*. The distance $d(x, y)$ is the number of coordinates where words x and y differ. We define $\mathbf{0}$ and $\mathbf{1}$ as the all zero word and respectively all one word. We define e_i as the almost all zero word which has a 1 at i'th coordinate. The *weight* $w(x)$ of a word $x \in \mathbb{F}^n$ is the number of coordinates equal to 1, i.e., $w(x) = d(x, \mathbf{0})$. When we speak about the *cover* of a word x, we mean the cardinality $|I(x)|$. The sizes of optimal self-locating-dominating and solid-locating-dominating codes in $\mathbb{F}^n$ are denoted by $\gamma^{SLD}(\mathbb{F}^n) = \gamma^{SLD}(n)$ and $\gamma^{DLD}(\mathbb{F}^n) = \gamma^{DLD}(n)$, respectively.

In what follows, we first concentrate on the case of self-locating-dominating codes. In particular, we present an infinite family of optimal self-locating-dominating codes in binary Hamming spaces. This result is based on the observation that a code C is self-locating-dominating in $\mathbb{F}^n$ if and only if for each $x \in \mathbb{F}^n \setminus C$ we have $|I(C; x)| \geq 3$ (see Theorem 16). An analogous result for self-identifying codes has been presented in [7]: a code C is self-identifying in $\mathbb{F}^n$ if and only if for each $x \in \mathbb{F}^n$ we have $|I(C; x)| \geq 3$.

The following well-known observation is useful in the following proofs of the paper.

Observation 15. Let $a, b \in \mathbb{F}^n$. We have

$$\big|N[a] \cap N[b]\big| = \begin{cases} 0, & \text{if } d(a,b) \geq 3, \\ 2, & \text{if } d(a,b) = 2, \\ 2, & \text{if } d(a,b) = 1, \\ n+1, & \text{if } a = b. \end{cases}$$

Theorem 16. *A code C is a self-locating-dominating code in $\mathbb{F}^n$ if and only if for each non-codeword w we have $|I(w)| \geq 3$.*

Proof. By Observation 15 we have that if two non-codewords w and w' have at least three common neighbours, then they are the same words. On the other hand if w has only two codewords in its I-set, then there is another word which has those same codewords in its I-set. □

Now we get the lower bound for self-locating-dominating codes.

Theorem 17. *Let $n \geq 3$. We have*

$$\gamma^{SLD}(n) \geq \left\lceil \frac{3 \cdot 2^n}{n+3} \right\rceil .$$

Proof. Let C be a self-locating-dominating code in $\mathbb{F}^n$. Theorem 16 says that each non-codeword w has at least three codewords in $I(w)$. We also have $|I(c)| \geq 1$ for all $c \in C$. Thus by double counting pairs (c, x) such that $c \in C, x \in \mathbb{F}^n$ and $d(c, x) \leq 1$, we get $(n+1)|C| \geq |C| + 3|\mathbb{F}^n \setminus C| = 3 \cdot 2^n - 2|C|$. This gives us

$$|C| \geq \left\lceil \frac{3 \cdot 2^n}{n+3} \right\rceil . \quad \square$$

In the proof of the following theorem, we are going to need some basics of linear codes. For more details, the interested reader is referred to [14]. The binary Hamming space $\mathbb{F}^n$ is a vector space under the normal addition of vectors and multiplication with scalars. We call code C *linear* if it is a subspace of $\mathbb{F}^n$. If C is a linear code, then we call H its *parity-check matrix* if $Hx^T = \mathbf{0}$ if and only if $x \in C$. If $Hy^T = d$, then we get a codeword of C by finding columns of H which form d as their sum and adding e_i to y if i'th column is in the sum. We denote the *covering radius* of C by $R(C) = \max_{x \in \mathbb{F}^n} \min_{c \in C} d(x, c)$. Hence, we have $R(C) = 1$ if each word has a codeword in its closed neighbourhood.

Theorem 18. *Let n and k be positive integers such that $n = 3(2^k - 1)$. We have*

$$\gamma^{SLD}(n) = 2^{3(2^k-1)-k}.$$

Proof. Theorem 17 gives us the lower bound $|C| \geq \frac{3 \cdot 2^n}{n+3} = 2^{3(2^k-1)-k}$.

Let C be a linear code such that its $k \times n$ parity-check matrix H, where $k \in \mathbb{Z}_+$ and $n = 3 \cdot (2^k - 1)$, contains each non-zero column of $\mathbb{F}^k$ three times and no zero columns. We now have $R(C) = 1$ since each non-zero word is a column of H. Furthermore, each non-codeword is covered by three codewords since there are three copies of each non-zero column. The cardinality of the code is $|C| = 2^{n-k} = 2^{3(2^k-1)-k}$ and it is a self-locating-dominating code by Theorem 16. □

Let $C \subseteq \mathbb{F}^n$ and $D \subseteq \mathbb{F}^m$ be codes. The *direct sum* of C and D is defined as $C \oplus D = \{(x, y) \mid x \in C, y \in D\}$. In the following theorem, it is shown that new self-locating-dominating codes can be formed from known ones using a direct sum.

Theorem 19. *If $C \subseteq \mathbb{F}^n$ is a self-locating-dominating code in $\mathbb{F}^n$, then $D = C \oplus \mathbb{F}$ is also a self-locating-dominating code in $\mathbb{F}^{n+1}$.*

Proof. Let $(a, x) \in \mathbb{F}^{n+1}$ where $a \in \mathbb{F}^n$, $x \in \mathbb{F}$ and $a \notin C$. We have $I(D; (a, x)) = \{(c, x) \mid c \in I(C; a)\}$. Since $|I(C; a)| \geq 3$, also $|I(D; (a, x))| \geq 3$. Therefore, D is a self-locating-dominating code. □

Let us then concentrate on solid-locating-dominating codes. We will first give a lower bound such that its ratio to $2\frac{2^n}{n+1}$ approaches 1 as n tends to infinity. After that we will give an infinite sequence of solid-locating-dominating codes with the

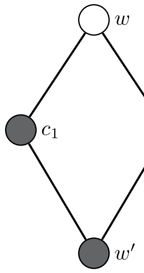


Fig. 9. If $w \in N_2$, then $w' \in C$.

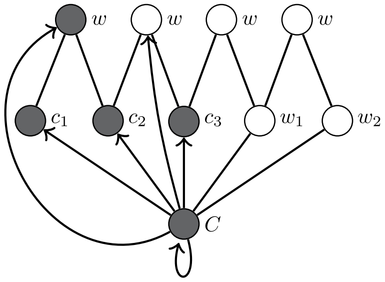


Fig. 10. Rule R2: c gives cover to words pointed by arrows.

same limit. When we compare the sizes of optimal self-locating-dominating and solid-locating-dominating codes we see from Theorems 17 and 18 that optimal solid-locating-dominating codes are essentially smaller. In the following theorem, we first give a lower bound for solid-locating-dominating codes.

Theorem 20. *Let n be an integer such that $n \geq 5$. We have*

$$\gamma^{DLD}(n) \geq \left\lceil \left(1 + \frac{n-1}{n^2+n+2}\right) \frac{2^{n+1}}{n+1} \right\rceil.$$

Proof. Let $C \subseteq \mathbb{F}^n$ be a solid-locating-dominating code. Let us define sets

$$C_i = \{c \in C \mid |I(c)| = i\} \text{ and } N_i = \{v \notin C \mid |I(v)| = i\}.$$

Let us also denote $C_{i^+} = \bigcup_{j=i}^{n+1} C_j$ and $N_{i^+} = \bigcup_{j=i}^{n} N_j$.

Each of the $|C|$ codewords covers $n+1$ words. Hence together they give total of $(n+1)|C|$ cover. On the other hand, if each of the $|\mathbb{F}^n|$ words is covered on average by at least $2+\frac{2n-2}{n^2+n+2}$ codewords, then we have an inequality $(n+1)|C| \geq (2+\frac{2n-2}{n^2+n+2})|\mathbb{F}^n|$ which gives the desired result when we solve $|C|$.

If $w \in N_1$ and $I(w) = \{c\}$, then $c \in C_n$. Otherwise, there would be a non-codeword v such that $I(w) \setminus I(v) = \emptyset$ which would mean that C is not a solid-locating-dominating code. If $w \in N_2$ and $I(w) = \{c_1, c_2\}$, then $N(c_1) \cap N(c_2) = \{w, w'\}$. We have $I(w) \subseteq I(w')$ and this implies that $w' \in C$ as in Fig. 9.

When $n \geq 5$, we have $2 + \frac{2n-2}{n^2+n+2} \leq \frac{9}{4}$ and we will be moving covers from words to words according to following three rules:

R1. If $x \in N_i$, $3 \leq i \leq n$, we will be moving $\frac{1}{4}$ cover from it to each codeword in $N(x)$.
R2. If $c \in C_{3^+}$ and $N(c) \cap N_1 = \emptyset$, then we move $\frac{2n-2}{n^2+n+2}$ cover from c to each codeword in $I(c)$ (including c itself) and each word which is neighbour to two words in $I(c) \setminus \{c\}$ as in Fig. 10.
R3. If $c \in C_n$ and $N(c) \cap N_1 = \{x\}$, then we first move one cover from c to x and then we move $\frac{n-3}{2+\binom{n-1}{2}}$ cover from c to x, c and each word which is neighbour to two words in $I(c) \setminus \{c\}$ as in Fig. 11.

We immediately notice that we never move cover away from a word with two different rules. We will next go through all types of words in the following order: N_{3^+}, C_1, $c \in C_{3^+}$ with $N(c) \cap N_1 = \emptyset$, N_1, $c \in C_n$ for which $N(c) \cap N_1 \neq \emptyset$, N_2 and finally C_2.

Let us first consider words $x \in N_i$, $3 \leq i \leq n$. If we move $\frac{1}{4}$ cover from x to i codewords in its I-set, then x has $i - \frac{i}{4} \geq \frac{9}{4}$ cover left.

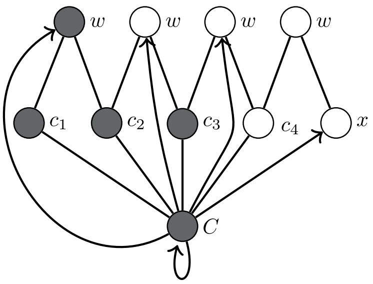


Fig. 11. Rule R3: c gives cover to words pointed by arrows.

If codeword c is in C_1, then each word $x \in N(c)$ is in N_{3+}. Since if $x \in N_1$, then $c \in C_n$ and if $x \in N_2$, then c has a codeword neighbour. Hence c has n neighbours which are in N_{3+} and each of them gives $\frac{1}{4}$ cover to c by R1. Thus c has at least $\frac{9}{4}$ cover since $n \geq 5$.

Let us next consider a codeword $c \in C_{n+1-K}$ such that either $K = 1$ and $N(c) \cap N_1 = \emptyset$ or $K \in \{i \mid 0 \leq i \leq n-2, i \neq 1\}$. We move $\frac{2n-2}{n^2+n+2}$ cover from c to $n+1-K+\binom{n-K}{2}$ words. After that c has

$$\begin{aligned} & n+1-K-\left(n+1-K+\binom{n-K}{2}\right)\frac{2n-2}{n^2+n+2}+\frac{2n-2}{n^2+n+2} \\ & \overset{(*)}{\geq} n+1-K-\left(n+1-K+\binom{n-K}{2}\right)\frac{n-1-K}{n+1-K+\binom{n-K}{2}}+\frac{2n-2}{n^2+n+2} \\ & = 2+\frac{2n-2}{n^2+n+2} \end{aligned} \tag{3}$$

cover. We get the inequality $(*)$ from the fact that $f(x) = \frac{x-1}{x+1+\binom{x}{2}}$ is decreasing when $x \geq 3$ and $f(2) = f(5)$. Hence $f(n-K) = \frac{n-1-K}{n+1-K+\binom{n-K}{2}}$ gets its minimum value when $K = 0$ at $f(n) = \frac{2n-2}{n^2+n+2}$ since $n \geq 5$.

Let $x \in N_1$ and $\{c\} = I(x)$. Hence $c \in C_n$ and x has $1 + 1 + \frac{n-3}{2+\binom{n-1}{2}}$ cover by R3. We have

$$\frac{n-3}{2+\binom{n-1}{2}} \geq \frac{2n-2}{n^2+n+2}, \text{ when } n \geq 5.$$

Hence x has enough cover.

When $c \in C_n$ and $x \in N_1 \cap N(c)$, we move $\frac{n-3}{2+\binom{n-1}{2}}$ cover from c to x, c and $\binom{n-1}{2}$ words neighbouring exactly two of codewords in $N(c)$ and 1 more cover to x in the rule R3. Now c has $n-1-\left(2+\binom{n-1}{2}\right)\frac{n-3}{2+\binom{n-1}{2}}+\frac{n-3}{2+\binom{n-1}{2}} = 2+\frac{n-3}{2+\binom{n-1}{2}}$ cover left which is enough.

Let $x \in N_2$ and $I(x) = \{c_1, c_2\}$. Hence $N(c_1) \cap N(c_2) = \{x, c\}$. Since C is a solid-locating-dominating code, we have that $c \in C$. Thus $|I(c)| \geq 3$ and c gives to x either $\frac{2n-2}{n^2+n+1}$ cover by R2 or $\frac{n-3}{2+\binom{n-1}{2}}$ cover if $|I(c)| = n$ and $N(c) \cap N_1 \neq \emptyset$ by R3. Hence x has at least $2 + \frac{2n-2}{n^2+n+2}$ cover.

Let $c \in C_2$ and $I(c) = \{c, c'\}$. If $c' \in C_2$, then c has $n-1$ non-codewords in N_{3+} in its neighbourhood. These words are in N_{3+}, since they clearly cannot be in N_1 and if $I(w) = \{c, c''\}$ for some $w \in N_2$, then c'' and c have a common codeword in their I-set but this is impossible, since $I(c) = I(c') = \{c, c'\}$ and $c' \neq c''$. Now each non-codeword in $N(c)$ gives c at least $\frac{1}{4}$ cover by R1. If $n \geq 5$, then it will have at least three cover. If $c' \in C_{3+}$ and $N(c') \cap N_1 = \emptyset$, then c' gives $\frac{2n-2}{n^2+n+1}$ cover to c by R2 and c has enough cover. If $c' \in C_n$ and $N(c') \cap N_1 = \{x\}$, then our situation is as in Fig. 12 and c and x have a common non-codeword neighbour v which is in N_{3+} since if $v \in N_1$, then $c \in C_n$ and if $v \in N_2$, then $\bigcap_{y \in I(v)} N[y] = \{v, c''\}$, so $c'' \in C$ and $c'' \neq c'$ so $\{c, c', c''\} \subseteq I(c)$ which is a contradiction. Since c has a neighbour in N_{3+}, it has at least $\frac{9}{4}$ cover by R1.

Now we have considered every word and each of them has at least $2 + \frac{2n-2}{n^2+n+2}$ cover. □

In the following remark, we briefly compare the previously obtained lower bound to one for locating-dominating codes locating multiple irregularities.

Remark 21. In this paper, we have mainly studied locating-dominating codes which can locate one and detect multiple irregularities. Previously, in [8], so called $(1, \leq \ell)$-locating-dominating codes of type B ($(1, \leq \ell)$-LDB codes for short), which can locate up to ℓ irregularities, have been studied. In [8, Theorem 5], the lower bound $\left\lceil \frac{2^{n+1}}{n+1} \right\rceil$ for $(1, \leq 2)$-LDB codes has been achieved. Since it can be shown that every $(1, \leq 2)$-LDB code is also a solid-locating-dominating code, our lower bound in Theorem 20 improves the lower bound for $(1, \leq 2)$-LDB codes in Hamming spaces.

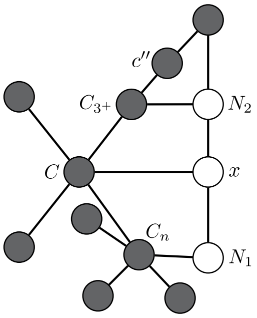


Fig. 12. A solid-locating-dominating code when $x \in N_1$. Darker vertices are in code C.

When $n \geq 5$, the lower bound in Theorem 20 is attained by choosing as codewords all codewords and their neighbours of a code with covering radius two and minimum distance five. Unfortunately, codes like this are only known when $n = 5$ [2, Theorem 11.2.2]. Using this code, the following theorem is obtained.

Theorem 22. *We have* $\gamma^{DLD}(5) = 12$.

Proof. We have $\gamma^{DLD}(5) \geq \lceil (2 + \frac{10-2}{5^2+5+2}) \cdot \frac{2^5}{5+1} \rceil = 12$ by Theorem 20. We can choose as our code C all words of weight 0, 1, 4 or 5 since then each non-codeword is covered by two codewords and the intersection of 1-balls of these two codewords contains a codeword (**1** or **0**). □

In general, we can construct solid-locating-dominating codes from codes with covering radius two.

Theorem 23. *If* $D \subseteq \mathbb{F}^n$ *is a code with* $R(D) = 2$*, then the code*

$$C = \{c \in \mathbb{F}^n \mid c \in N[d], d \in D\}$$

is solid-locating-dominating.

Proof. Since D has covering radius of two, each non-codeword w is covered by at least two codewords $x, y \in C$. These codewords on the other hand have a common codeword d, the one which is also in D. Now $\{w, d\} \subseteq \bigcap_{c \in I(w)} N[c]$ so C is a solid-locating-dominating code. □

In [2, Theorem 4.5.8], Struik has constructed an infinite sequence of codes with covering radius two such that we can build on top of it such a sequence of solid-locating-dominating codes that they converge to our lower bound. We denote the cardinality of a ball with radius 2 in $\mathbb{F}^n$ with $V(n, 2)$.

Theorem 24. *There exists a sequence of solid-locating-dominating codes* $(C_n)_{n=1}^{\infty}$ *such that*

$$\lim_{n \to \infty} \frac{|C_n|}{2\frac{2^n}{n+1}} = 1.$$

Proof. Struik has constructed a sequence of codes $(D_n)_{n=1}^{\infty}$ with covering radius two such that $\frac{|D_n| V(n,2)}{2^n} \xrightarrow{n \to \infty} 1$. If we choose $C_n = \{x \in \mathbb{F}^n \mid a \in D_n, x \in N[a]\}$, this is a solid-locating-dominating code by Theorem 23. We have $|C_n| \leq (n+1)|D_n|$. Hence

$$\frac{|C_n|}{2\frac{2^n}{n+1}} \leq \frac{(n+1)|D_n|}{2\frac{2^n}{n+1}} = \frac{|D_n| V(n, 2)}{2^n} + \frac{|D_n|(\frac{n}{2} - \frac{1}{2})}{2^n} \xrightarrow{n \to \infty} 1.$$

On the other hand we have from Theorem 20 $\frac{|C_n|}{2\frac{2^n}{n+1}} \geq \frac{(1+\frac{n-1}{n^2+n+2})\frac{2^{n+1}}{n+1}}{2\frac{2^n}{n+1}} \xrightarrow{n \to \infty} 1$, which proves the theorem. □

Using direct sum we can construct new solid-locating-dominating codes from existing ones in a similar fashion as with self-locating-dominating codes.

Theorem 25. *If* $C \subseteq \mathbb{F}^n$ *is a solid-locating-dominating code, then code* $D = C \oplus \mathbb{F}$ *is also solid-locating-dominating.*

Proof. Let $(a, x) \in \mathbb{F}^{n+1}$ where $a \in \mathbb{F}^n$, $x \in \mathbb{F}$ and $a \notin C$. We have $I(D; (a, x)) = \{(y, x) \mid y \in I(C; a)\}$. If $|I(C; a)| \geq 3$, also $|I(D; (a, x))| \geq 3$. If $I(C; a) = \{c_1, c_2\}$, then there is a codeword $c_3 \in N(c_1) \cap N(c_2)$. Now we have $I(D; (a, x)) = \{(c_1, x), (c_2, x)\}$ and $(c_3, x) \in N(c_1, x) \cap N(c_2, x)$. If $I(C; a) = \{c\}$, then $|I(C; c)| = n$ and $|I(D; (c, x))| = n + 1$. Since $I(D; (a, x)) = \{(c, x)\}$, D is a solid-locating-dominating code. □

Table 1
Optimal self-locating-dominating and solid-locating-dominating codes in binary Hamming spaces of short lengths.

n	$\gamma^{SLD}(n)$	$\gamma^{DLD}(n)$
1	2	1
2	4	2
3	4	4
4	8	8
5	16	12
6	[22, 28]	[21, 23]

For small lengths n the sizes of optimal self-locating-dominating and solid-locating-dominating codes in $\mathbb{F}^n$ are presented in Table 1. The lower bounds of $\gamma^{DLD}(n)$ for $n \leq 4$ as well as $\gamma^{SLD}(1)$, $\gamma^{SLD}(2)$ and $\gamma^{SLD}(5)$ are achieved with computer search. The lower bounds of $\gamma^{SLD}(3)$ and $\gamma^{SLD}(4)$ are due to the fact $\gamma^{SLD}(n) \geq \gamma^{DLD}(n)$. The rest of the lower bounds are due to Theorems 17 and 20. The upper bound of $\gamma^{SLD}(1)$ comes from the size $|\mathbb{F}| = 2$ and the upper bound of $\gamma^{DLD}(1)$ is gained with the code $C = \{0\}$. The upper bounds of $\gamma^{SLD}(2)$, $\gamma^{SLD}(4)$, $\gamma^{SLD}(5)$ are from Theorem 19 and the upper bounds of $\gamma^{DLD}(2)$, $\gamma^{DLD}(3)$ and $\gamma^{DLD}(4)$ are from Theorem 25. We get $\gamma^{DLD}(5)$ from Theorem 22, $\gamma^{SLD}(3)$ is gained with the code $C = \{x \mid w(x) = 0 \text{ or } w(x) = 2\}$, the upper bound for $\gamma^{SLD}(6)$ with the code $C = \{x \in \mathbb{F}^6 \mid x \in A, w(x) = 1, w(x) = 4 \text{ or } w(x) = 6\}$, where $A = \{(1, 1, 1, 0, 0, 0), (1, 0, 0, 1, 1, 0), (1, 1, 0, 0, 0, 1), (0, 1, 0, 1, 1, 0), (0, 0, 1, 1, 0, 1), (0, 0, 1, 0, 1, 1)\}$ and the upper bound for $\gamma^{DLD}(6)$ with the code $C = \{x \in \mathbb{F}^6 \mid w(x) = 0, 1, 4 \text{ or } 6\}$.

Above, we have discussed self-locating-dominating and solid-locating-dominating codes in binary Hamming spaces. In what follows, we briefly consider regular locating-dominating codes. In particular, for certain lengths, we provide locating-dominating codes with the smallest known cardinalities. Previously, locating-dominating codes in $\mathbb{F}^n$ have been considered, for example, in [3,8]. For future considerations, we first define the mapping $\pi : \mathbb{F}^n \to \mathbb{F}$ as follows:

$$\pi(u) = \begin{cases} 0, & \text{if } w(u) \text{ is even;} \\ 1, & \text{if } w(u) \text{ is odd.} \end{cases}$$

In the following theorem we introduce a novel approach for constructing new locating-dominating codes based on known (suitable) identifying codes.

Theorem 26. *Let C be an identifying code in $\mathbb{F}^n$ such that $|I(C; u)| \geq 2$ for all $u \in \mathbb{F}^n \setminus C$. Then*

$$D = \{(\pi(u), u, u + c) \mid u \in \mathbb{F}^n, c \in C\}$$

is a locating-dominating code in $\mathbb{F}^{2n+1}$.

Proof. Let a be an element of $\mathbb{F}$, u and v be words of $\mathbb{F}^n$ and $x = (a, u, u + v)$ be a word of $\mathbb{F}^{2n+1}$. Assume further that $I(C; v) = \{c_1, c_2, \ldots, c_k\}$ for some positive integer k. Then we have the following observations:

- If $a = \pi(u)$, then we have $I(D; x) = \{(a, u, u + c_1), (a, u, u + c_2), \ldots, (a, u, u + c_k)\}$.
- Assume then that $a \neq \pi(u)$. If v is not a codeword of C, then we have $I(D; x) = \{(a, u + v + c_1, u + v), (a, u + v + c_2, u + v), \ldots, (a, u + v + c_k, u + v)\}$. Indeed, the word $(a, u + v + c_i, u + v)$ belongs to $I(D; x)$ since its distance from x is equal to 1 and $(a, u + v + c_i, u + v) = (a, u + v + c_i, (u + v + c_i) + c_i) \in D$. If v is a codeword of C, say $v = c_1$, then we similarly have $I(D; x) = \{(a + 1, u, u + v), (a, u + v + c_2, u + v), \ldots, (a, u + v + c_k, u + v)\}$.

Assume then that x is not a codeword of D. By the previous observation, we first obtain that $I(D; x) \neq \emptyset$ as $I(C; v) \neq \emptyset$. Furthermore, if $|I(D; x)| \geq 3$, then the identifying set of x immediately identifies the word x by Observation 15. Hence, we may assume that $|I(D; x)| \leq 2$. In what follows, we first suppose that $|I(D; x)| = 2$ implying $|I(C; v)| = 2$.

Assume first that $a = \pi(u)$. Then, by the previous observation, we have $I(D; x) = \{(a, u, u + c_1), (a, u, u + c_2)\}$. Assume then (to the contrary) that there exists $y \in \mathbb{F}^{2n+1} \setminus D$ such that $x \neq y$ and $I(D; x) = I(D; y)$. Since $(a, u, u + v + c_1 + c_2)$ is the unique word in the set $(N[(a, u, u + c_1)] \cap N[(a, u, u + c_2)]) \setminus \{x\}$, we obtain that $y = (a, u, u + v + c_1 + c_2)$. Therefore, as $I(D; x) = I(D; y)$, we have $I(C; v + c_1 + c_2) = I(C; v)$. However, this is a contradiction since $v + c_1 + c_2 \neq v$ and C is an identifying code in $\mathbb{F}^n$.

Assume then that $a \neq \pi(u)$. If v is not a codeword of C, then $I(D; x) = \{(a, u + v + c_1, u + v), (a, u + v + c_2, u + v)\}$ by the previous observation. Assume now (to the contrary) that there exists $y \in \mathbb{F}^{2n+1} \setminus D$ such that $x \neq y$ and $I(D; x) = I(D; y)$. Then we obtain that $y = (a, u + c_1 + c_2, u + v)$ since $(a, u + c_1 + c_2, u + v)$ is the unique word in the set $(N[(a, u + v + c_1, u + v)] \cap N[(a, u + v + c_2, u + v)]) \setminus \{x\}$. Denoting $u' = u + c_1 + c_2$ and $v' = v + c_1 + c_2$, we have $\pi(u) = \pi(u')$, $y = (a, u', u' + v')$ and $I(D; y) = \{(a, u' + v' + c_1, u' + v'), (a, u' + v' + c_2, u' + v')\}$. Therefore, as $I(D; x) = I(D; y)$, it follows that $I(C; v) = I(C; v')$ (a contradiction). Hence, we may assume that v is a codeword of C, say $v = c_1$. Then we have $I(D; x) = \{(a + 1, u, u + v), (a, u + v + c_2, u + v)\}$. Now we obtain that $y = (a + 1, u + v + c_2, u + v)$ since it is the unique word in the set $(N[(a + 1, u, u + v)] \cap N[(a, u + v + c_2, u + v)]) \setminus \{x\}$. Denoting $a' = a + 1$, $u' = u + v + c_2$, $v' = c_2$ and $c_2' = c_1$, we have $y = (a', u', u' + v')$ and $I(D; y) = \{(a' + 1, u', u' + v'), (a', u' + v' + c_2', u' + v')\}$. Therefore, as $I(D; x) = I(D; y)$, it follows that $I(C; v) = \{v, c_2\} = \{v', c_2'\} = I(C; v')$ (a contradiction).

Finally, we assume that $|I(D; x)| = 1$. This implies that $|I(C; v)| = 1$. Hence, as $|I(C; u)| \geq 2$ for all $u \in \mathbb{F}^n \setminus C$, we know that $v \in C$. Then we may assume that $a \neq \pi(u)$ as otherwise $x = (a, u, u+v)$ belongs to D. Now, by the previous observation, we have $I(D; x) = \{(a + 1, u, u + v)\}$. Assume to the contrary that there exists $y = (a', u', u' + v') \in \mathbb{F}^{2n+1} \setminus D$ such that $I(D; x) = I(D; y)$. As above, we obtain that $v' \in C$ and $I(D; y) = \{(a' + 1, u', u' + v')\}$. Therefore, as $I(D; x) = I(D; y)$, we have $a' = a, u' = u, v' = v$ and $x = y$ (a contradiction). Thus, in conclusion, we have shown that D is a locating-dominating code in $\mathbb{F}^{2n+1}$. □

The best known upper bounds on $\gamma^{LD}(\mathbb{F}^n)$ for $1 \leq n \leq 10$ have been presented in [3, Table 3]. For lengths $n > 10$, the smallest known locating-dominating codes are actually identifying codes. (Recall that by the definitions any identifying code is also locating-dominating.) The currently best known upper bounds on $\gamma^{ID}(\mathbb{F}^n)$ can be found in [1]. In the following corollary, we present locating-dominating codes in $\mathbb{F}^n$ with the smallest known cardinalities for the lengths $n = 11$ and $n = 17$. These constructions significantly improve on the known upper bounds $\gamma^{LD}(\mathbb{F}^{11}) \leq \gamma^{ID}(\mathbb{F}^{11}) \leq 352$ and $\gamma^{LD}(\mathbb{F}^{17}) \leq \gamma^{ID}(\mathbb{F}^{17}) \leq 18\,558$.

Corollary 27. *We have* $\gamma^{LD}(\mathbb{F}^{11}) \leq 320$ *and* $\gamma^{LD}(\mathbb{F}^{17}) \leq 16\,384$.

Proof. Let C_1 be a code in $\mathbb{F}^5$ formed by the words of weight 1 and 4. In [11], it has been shown that C_1 is an identifying code in $\mathbb{F}^5$ (with 10 codewords). Moreover, it is straightforward to verify that for all $u \in \mathbb{F}^5 \setminus C_1$ we have $|I(C_1; u)| \geq 2$. Indeed, we have $|I(\mathbf{0})| = 5$ and each word of weight two is covered by exactly two codewords of weight one. By symmetry, analogous observations also hold for the words of weight three and the word $\mathbf{1}$. Therefore, by Theorem 26, the code

$$D_1 = \{(\pi(u), u, u + c) \mid u \in \mathbb{F}^5, c \in C_1\}$$

is locating-dominating in $\mathbb{F}^{11}$. Thus, we have $\gamma^{LD}(\mathbb{F}^{11}) \leq |D_1| = 2^5 \cdot |C_1| = 320$.

Let C_2 be a code in $\mathbb{F}^8$ formed by the binary representations of length 8 of the following integers: 3, 6, 8, 13, 18, 21, 27, 28, 32, 39, 41, 46, 49, 52, 58, 63, 65, 68, 74, 79, 80, 87, 89, 94, 98, 101, 107, 108, 115, 118, 120, 125, 129, 132, 138, 143, 144, 151, 153, 158, 162, 165, 171, 172, 179, 182, 184, 189, 195, 198, 200, 205, 210, 213, 219, 220, 224, 231, 233, 238, 241, 244, 250, 255. It is straightforward to verify that C_2 has 64 codewords, C_2 is an identifying code in $\mathbb{F}^8$ and for all $u \in \mathbb{F}^8 \setminus C_2$ we have $|I(C_2; u)| \geq 2$. Therefore, by Theorem 26, the code

$$D_2 = \{(\pi(u), u, u + c) \mid u \in \mathbb{F}^8, c \in C_2\}$$

is locating-dominating in $\mathbb{F}^{17}$. Thus, we have $\gamma^{LD}(\mathbb{F}^{17}) \leq |D_2| = 2^8 \cdot |C_2| = 16\,384$. □

With the help of the following theorem, which has been shown in [8, Theorem 7], we can construct new improved locating-dominating codes from codes obtained in Corollary 27.

Theorem 28 ([8]). *If* $C \subseteq \mathbb{F}^n$ *is a locating-dominating code, then* $C \oplus \mathbb{F}$ *is also a locating-dominating code.*

The smallest currently known upper bounds for locating-dominating codes of lengths $n = 12$ and $n = 18$ are 684 and 35 604 respectively [1].

Corollary 29. *We have* $\gamma^{LD}(\mathbb{F}^{12}) \leq 640$ *and* $\gamma^{LD}(\mathbb{F}^{18}) \leq 32\,768$.

Proof. The upper bounds follow immediately by applying Theorem 28 on codes obtained in Corollary 27. □

In [8, Theorem 15], a lower bound for $\gamma^{LD}(\mathbb{F}^n)$, which is currently the best known, has been presented. Applying the lower bound on the lengths $n = 11, n = 12, n = 17$ and $n = 18$, we obtain that $\gamma^{LD}(\mathbb{F}^{11}) \geq 309, \gamma^{LD}(\mathbb{F}^{12}) \geq 576, \gamma^{LD}(\mathbb{F}^{17}) \geq 13\,676$ and $\gamma^{LD}(\mathbb{F}^{18}) \geq 26\,006$. Thus, comparing the lower bounds to the constructions of the previous corollaries, we can state the gap between the new upper bound and existing lower bound is significantly smaller than the gap between the previous upper and lower bounds.

Acknowledgement

The third author's research was supported by the University of Turku Graduate School (UTUGS).

References

[1] I. Charon, G. Cohen, O. Hudry, A. Lobstein, New identifying codes in the binary Hamming space, European J. Combin. 31 (2) (2010) 491–501.
[2] G. Cohen, I. Honkala, S. Litsyn, A. Lobstein, Covering Codes, in: North-Holland Mathematical Library, vol. 54, North-Holland Publishing Co., Amsterdam, 1997.
[3] G. Exoo, V. Junnila, T. Laihonen, S. Ranto, Locating vertices using codes, in: Proceedings of the Thirty-Ninth Southeastern International Conference on Combinatorics, Graph Theory and Computing, Vol. 191, 2008, pp. 143–159.
[4] N. Fazlollahi, D. Starobinski, A. Trachtenberg, Connected identifying codes, IEEE Trans. Inform. Theory 58 (7) (2012) 4814–4824.
[5] W. Goddard, K. Wash, ID codes in Cartesian products of cliques, J. Combin. Math. Combin. Comput. 85 (2013) 97–106.
[6] S. Gravier, J. Moncel, A. Semri, Identifying codes of Cartesian product of two cliques of the same size, Electron. J. Combin. 15 (1) (2008) Note 4, 7.

[7] I. Honkala, T. Laihonen, On a new class of identifying codes in graphs, Inform. Process. Lett. 102 (2–3) (2007) 92–98.
[8] I. Honkala, T. Laihonen, S. Ranto, On locating-dominating codes in binary Hamming spaces, Discrete Math. Theor. Comput. Sci. 6 (2) (2004) 265–281.
[9] V. Junnila, T. Laihonen, Collection of codes for tolerant location, in: Proceedings of the Bordeaux Graph Workshop, 2016, pp. 176–179.
[10] V. Junnila, T. Laihonen, Tolerant location detection in sensor networks. submitted for publication, 2016.
[11] M.G. Karpovsky, K. Chakrabarty, L.B. Levitin, On a new class of codes for identifying vertices in graphs, IEEE Trans. Inform. Theory 44 (2) (1998) 599–611.
[12] M. Laifenfeld, A. Trachtenberg, Disjoint identifying-codes for arbitrary graphs, in: Proceedings of International Symposium on Information Theory, 2005. ISIT 2005, 2005, pp. 244–248.
[13] A. Lobstein, Watching systems, identifying, locating-dominating and discriminating codes in graphs, a bibliography. Published electronically at http://perso.enst.fr/~lobstein/debutBIBidetlocdom.pdf.
[14] F. Jessie MacWilliams, Neil J.A. Sloane, The Theory of Error-Correcting Codes, in: North-Holland Mathematical Library, vol. 16, North-Holland Publishing Co., Amsterdam, 1977.
[15] D.F. Rall, P.J. Slater, On location-domination numbers for certain classes of graphs, Congr. Numer. 45 (1984) 97–106.
[16] S. Ray, D. Starobinski, A. Trachtenberg, R. Ungrangsi, Robust location detection with sensor networks, IEEE J. Sel. Areas Commun. 22 (6) (2004) 1016–1025.
[17] P.J. Slater, Domination and location in acyclic graphs, Networks 17 (1) (1987) 55–64.
[18] P.J. Slater, Dominating and reference sets in a graph, J. Math. Phys. Sci. 22 (1988) 445–455.

Publication II

Ville Junnila, Tero Laihonen, Tuomo Lehtilä and María Luz Puertas. On stronger types of locating-dominating codes, *Discrete Mathematics and Theoretical Computer Science*, 21, 2019.

Discrete Mathematics and Theoretical Computer Science DMTCS vol. **21**:1, 2019, #1

On Stronger Types of Locating-Dominating Codes[*]

Ville Junnila[1] Tero Laihonen[1] Tuomo Lehtilä[1†] María Luz Puertas[2‡]

[1] *Department of Mathematics and Statistics, University of Turku, Finland*
[2] *Department of Mathematics, Universidad de Almeria, Spain*

received 22nd Aug. 2018, revised 12th Mar. 2019, accepted 25th Mar. 2019.

Locating-dominating codes in a graph find their application in sensor networks and have been studied extensively over the years. A locating-dominating code can locate one object in a sensor network, but if there is more than one object, it may lead to false conclusions. In this paper, we consider stronger types of locating-dominating codes which can locate one object and detect if there are multiple objects. We study the properties of these codes and provide bounds on the smallest possible size of these codes, for example, with the aid of the Dilworth number and Sperner families. Moreover, these codes are studied in trees and Cartesian products of graphs. We also give the complete realization theorems for the coexistence of the smallest possible size of these codes and the optimal locating-dominating codes in a graph.

Keywords: Dominating set; locating-dominating set; locating-dominating code; Dilworth number; Sperner's Theorem

1 Introduction

Sensor networks are systems designed for environmental monitoring. Various location detection systems such as fire alarm and surveillance systems can be viewed as examples of sensor networks. For location detection, a sensor can be placed in any location of the network. The sensor monitors its neighbourhood (including the location of the sensor itself) and reports possible objects or irregularities such as a fire or an intruder in the neighbouring locations. In the model considered in the paper, it is assumed that a sensor can distinguish whether the irregularity is in the location of the sensor or in the neighbouring locations (as in [21, 24, 25]). Based on the reports of the sensors, a central controller attempts to determine the location of a possible irregularity in the network. Usually, the aim is to minimize the number of sensors in the network. More explanation regarding location detection in sensor networks can be found in [9, 17, 22]. An online bibliography on the topic can be found at [18].

A sensor network can be modelled as a simple and undirected graph $G = (V(G), E(G)) = (V, E)$ as follows: the set of vertices V of the graph represents the locations of the network and the edge set E of the graph represents the connections between the locations. In other words, a sensor can be placed in each vertex of the graph and the sensor placed in the vertex u monitors u itself and the vertices neighbouring u. Moreover, besides being simple and undirected we also assume that the graphs in this paper are finite. In what follows, we present some basic terminology and notation regarding graphs. The *open neighbourhood* of $u \in V$ consists of the vertices adjacent to u and it is denoted by $N(u)$. The *closed neighbourhood* of u is defined as $N[u] = \{u\} \cup N(u)$. The *degree* of a vertex u is the number of vertices in the open neighbourhood $N(u)$ and the *maximum degree* $\Delta(G) = \Delta$ of the graph G is the maximum degree among all the vertices of G. The *distance* between two vertices $d(u, v)$ is the number of edges in any shortest path connecting them. A non-empty subset C of V is called a *code* and the elements of the code are called

---

[*] The paper has been presented in part in the 10th International Colloquium on Graph Theory and Combinatorics (2018, Lyon, France).

[†] Research supported by the University of Turku Graduate School (UTUGS) and the Vilho, Yrjö and Kalle Väisälä Foundation.

[‡] Author partially supported by the grant MTM2015-63791-R (MINECO-FEDER).

ISSN 1365–8050

codewords. In this paper, the code C (usually) represents the set of locations where the sensors have been placed on. For the set of sensors monitoring a vertex $u \in V$, we use the following notation:

$$I(u) = N[u] \cap C.$$

In order to emphasize the graph G and/or the code C, we sometimes write $I(u) = I(C;u) = I(G,C;u)$. We call $I(u)$ the *I-set* or the *identifying set* of u.

As stated above, a sensor $u \in V$ reports that an irregularity has been detected if there is (at least) one in the closed neighbourhood $N[u]$. In the model of the paper, we further assume that a sensor $u \in V$ reports 2 if there is an irregularity in u, it reports 1 if there is one in $N(u)$ (and none in u itself), and otherwise it reports 0. In other words, a sensor can distinguish whether an irregularity is in the location of the sensor or in the neighbouring locations. We say that a set (or a code) C is *dominating* in G if $I(C;u)$ is non-empty for all $u \in V \setminus C$. In other words, an irregularity in the network can be detected (albeit not located). Furthermore, the smallest cardinality of a dominating set in G is called the *domination number* and it is denoted by $\gamma(G)$. Notice then that if the sensors in the code C are located in such places that $I(C;u)$ is non-empty and unique for all $u \in V \setminus C$, then an irregularity in the network can be located by comparing $I(C;u)$ to I-sets of other non-codewords. This leads to the following definition of *locating-dominating codes (or sets)*, which were first introduced by Slater in [21, 24, 25].

Definition 1. A code $C \subseteq V$ is *locating-dominating* in G if for all distinct $u, v \in V \setminus C$ we have $I(C;u) \neq \emptyset$ and

$$I(C;u) \neq I(C;v).$$

A locating-dominating code C in a finite graph G with the smallest cardinality is called *optimal* and the number of codewords in an optimal locating-dominating code is denoted by $\gamma^{LD}(G)$. The value $\gamma^{LD}(G)$ is also called the *location-domination number*.

The previous definition of locating-dominating codes is illustrated in the following example.

Example 2. Let G be the graph illustrated in Figure 1. Consider the code $C = \{b, d, f\}$ in G (see Figure 1). Since the I-sets $I(C;a) = \{b, d\}$, $I(C;c) = \{b, f\}$ and $I(C;e) = \{b, d, f\}$ are all non-empty and different, the code C is locating-dominating in G. Moreover, there do not exist smaller locating-dominating codes in G as using at most two codewords we can form at most three different non-empty I-sets. Therefore, we have $\gamma^{LD}(G) = 3$.

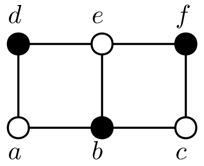
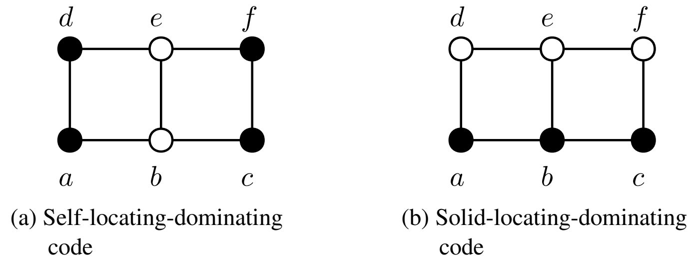


Fig. 1: Optimal locating-dominating code in a graph G

The original concept of locating-dominating codes has some issues in certain types of applications. Firstly, locating-dominating codes might output misleading results if there exist more than one irregularity in the graph. For instance, if in the previous example there exist irregularities in a and c, then the sensors located at b, d and f are reporting 1. Now the system deduces that the irregularity is in e. Hence, a completely false output is given and we do not even notice that something is wrong. Secondly, in order to determine the location of the irregularity, we have to compare the obtained I-set to other such sets. In order to overcome these issues, so called self-locating-dominating and solid-locating-dominating codes have been introduced in [16] motivated by $(1, \leq 1)^+$-identifying or self-identifying codes introduced in [12, 14, 15]. For more detailed discussion on the motivation of self-locating-dominating and solid-locating-dominating codes, the interested reader is referred to [16]. The formal definitions of these codes are given in the following.

Definition 3. A code $C \subseteq V$ is *self-locating-dominating* in G if, for all $u \in V \setminus C$, we have $I(C;u) \neq \emptyset$ and

$$\bigcap_{c \in I(C;u)} N[c] = \{u\}.$$



(a) Self-locating-dominating code

(b) Solid-locating-dominating code

Fig. 2: Optimal self-locating-dominating and solid-locating-dominating codes in a graph G

A self-locating-dominating code C in a finite graph G with the smallest cardinality is called *optimal* and the number of codewords in an optimal self-locating-dominating code is denoted by $\gamma^{SLD}(G)$. The value $\gamma^{SLD}(G)$ is also called the *self-location-domination number*.

Definition 4. A code $C \subseteq V$ is *solid-locating-dominating* in G if $I(C;u) \neq \emptyset$ for every $u \in V \setminus C$ and, for all distinct $u, v \in V \setminus C$, we have

$$I(C;u) \setminus I(C;v) \neq \emptyset.$$

Note that this condition is equivalent with $I(C;u) \not\subseteq I(C;v)$. A solid-locating-dominating code C in a finite graph G with the smallest cardinality is called *optimal* and the number of codewords in an optimal solid-locating-dominating code is denoted by $\gamma^{DLD}(G)$. The value $\gamma^{DLD}(G)$ is also called the *solid-location-domination number*.

By the previous definitions, it is immediate that any self-locating-dominating and solid-locating-dominating code is also locating-dominating (see also Corollary 7) and that every graph contains a self-locating-dominating and solid-locating-dominating code. Indeed, $C = V$ is always a self-locating-dominating and solid-locating-dominating code. The definitions are illustrated in the following example. In particular, we show that the given definitions are indeed different.

Example 5. Let G be a graph illustrated in Figure 2. Let C be a self-locating-dominating code in G. Observe first that if $a \notin C$, then $I(C;a) \subseteq \{b,d\}$ and we have

$$\{a,e\} \subseteq \bigcap_{x \in I(C;a)} N[x].$$

This implies a contradiction and, therefore, the vertex a belongs to C. An analogous argument also holds for the vertices c, d and f. Hence, we have $\{a,c,d,f\} \subseteq C$. Moreover, the code $C_1 = \{a,c,d,f\}$, which is illustrated in Figure 2(a), is self-locating-dominating in G since for the non-codewords b and e we have $I(C_1;b) = \{a,c\}$ and $N[a] \cap N[c] = \{b\}$, and $I(C_1;e) = \{d,f\}$ and $N[d] \cap N[f] = \{e\}$. Hence, C_1 is an optimal self-locating-dominating code in G and we have $\gamma^{SLD}(G) = 4$.

Let us then consider the code $C_2 = \{a,b,c\}$, which is illustrated in Figure 2(b). Now we have $I(C_2;d) = \{a\}$, $I(C_2;e) = \{b\}$ and $I(C_2;f) = \{c\}$. Therefore, it is easy to see that C_2 is a solid-locating-dominating code in G. Moreover, there are no solid-locating-dominating codes in G with smaller number of codewords since even a regular locating-dominating code has always at least 3 codewords by Example 2. Thus, C_2 is an optimal solid-locating-dominating code in G and we have $\gamma^{DLD}(G) = 3$.

In the previous example, we showed that the definitions of self-locating-dominating and solid-locating-dominating codes are different. Furthermore, by comparing Examples 2 and 5, we notice that the new codes are also different from the original locating-dominating codes. In the following theorem, we present new characterizations for self-locating-dominating and solid-locating-dominating codes. Comparing these characterizations to the original definitions of the codes, the differences of the codes become apparent. We omit the proof of the following theorem, because it is proved in [16] for connected graphs and it is easily modified for non-connected ones.

Theorem 6. *Let G be a graph on at least two vertices.*

(i) A code $C \subseteq V$ is self-locating-dominating if and only if, for all distinct $u \in V \setminus C$ and $v \in V$, we have $I(C;u) \setminus I(C;v) \neq \emptyset$.

(ii) A code $C \subseteq V$ is solid-locating-dominating if and only if, for all $u \in V \setminus C$, we have $I(C;u) \neq \emptyset$ and

$$\left(\bigcap_{c \in I(C;u)} N[c] \right) \setminus C = \{u\}.$$

By comparing Definition 4 and Theorem 6 (i) we notice that the only difference is that in $I(C;u) \setminus I(C;v)$ vertex v can be a codeword when we consider self-location-domination. Similarly, when we compare Definition 3 and Theorem 6 (ii) we notice that the only difference is that in the case of solid-location-domination we omit codewords from the intersection.

The previous theorem together with the definition of solid-locating-dominating codes and the previous observation immediately gives the following corollary.

Corollary 7. *The following facts hold for all graphs G.*

- *If C is a self-locating-dominating code in G, then C is also solid-locating-dominating in G.*
- *If C is a solid-locating-dominating code in G, then C is also locating-dominating in G.*

Thus, we have $\gamma^{LD}(G) \leq \gamma^{DLD}(G) \leq \gamma^{SLD}(G)$.

As stated earlier, self-locating-dominating and solid-locating-dominating codes have benefits over regular locating-dominating codes; they detect more than one irregularity and locate one irregularity without comparison to other I-sets — for more details, see [16].

Previously, when self-locating-dominating and solid-locating-dominating codes have been studied, in [16], the optimal values for $\gamma^{SLD}(K_n \square K_m)$ and $\gamma^{DLD}(K_n \square K_m)$ have been found. Also the a general lower bound for $\gamma^{SLD}(\mathbb{F}_2^n)$ has been given and an infinite family of constructions attaining this bound is presented for suitable values of n. Moreover, a general lower bound for $\gamma^{DLD}(\mathbb{F}_2^n)$ is given and this bound is shown to be asymptotically tight as n grows.

In what follows, the structure of the paper is briefly discussed. In Section 2, we first show some general bounds and properties for self- and solid-locating-dominating codes; in particular, we utilize the Dilworth number and Sperner families. Then, in Section 3, we consider the codes in trees and determine self-location-domination and solid-location-domination numbers with the help of other graph parameters. In Section 4, we consider Cartesian products and give some general bounds for them which are shown to be achieved in the case of ladders and some rook's graphs. Finally, in Section 5, we study the existence of graphs when we are given the location-domination number and the self-location-domination or the solid-location-domination number associated with them.

2 Basics

In this section, we present some basic results regarding self-locating-dominating and solid-locating-dominating codes. In particular, we give various lower and upper bounds for such codes. We first begin by giving results which do not take advantage of any properties or parameters of the graph such as the maximum degree or the independence number. Then, in Section 2.1, we use the Sperner's Theorem to gain new bounds. Later, in Section 2.2, we apply the Dilworth number. Finally, in Section 2.3, we use independence number and consider complements of graphs.

In the following theorem, we begin by giving a simple upper bound for solid-locating-dominating codes in graphs. It is clear that the discrete graph D_n, with n vertices and no edges, satisfies $\gamma^{DLD}(D_n) = n$, because $V(D_n)$ is its unique dominating set. We now focus on graphs with at least one edge.

Theorem 8. *If $G = (V,E)$ is a graph with order n and size $m \geq 1$, then the code $V \setminus \{u\}$ is solid-locating-dominating in G for any non-isolated vertex $u \in V$. Thus, we have*

$$\gamma^{DLD}(G) \leq n - 1.$$

Proof: Let $u \in V$ be a non-isolated vertex of G, i.e., $N(u) \cap (V \setminus \{u\}) \neq \emptyset$. By the definition, it is immediate that $V \setminus \{u\}$ is a solid-locating-dominating code of G. This further implies that $\gamma^{DLD}(G) \leq |V| - 1$. □

The result of the previous theorem can also be interpreted as follows: in the particular case of graphs with no isolated vertices, none of the vertices of a graph is forced to be in all the solid-locating-dominating codes of the graph and hence, the same is also true for locating-dominating codes by Corollary 7. However, this is not the case with self-locating-dominating codes. Hence, for future considerations, we define the concept of forced codewords as follows: a vertex u of G is said to be a *forced codeword* regarding self-location-domination if u belongs to all self-locating-dominating codes in G. In the following theorem, we give a simple characterization for forced codewords and show that such vertices indeed exist.

Theorem 9. *Let $G = (V, E)$ be a graph. If $|V| = 1$, then the single vertex of the graph G is a forced codeword. Assuming $|V| \geq 2$, a vertex $u \in V$ is a forced codeword regarding self-location-domination if and only if for some vertex $v \in V$ other than u we have $N(u) \subseteq N[v]$.*

Proof: Let C be a self-locating-dominating code in G and u be a vertex of V. If $|V| = 1$, then due to the domination the single vertex is a forced codeword. Assume now that $|V| \geq 2$. Suppose further that there exists another vertex $v \in V$ such that $u \neq v$ and $N(u) \subseteq N[v]$. If $N(u) = \emptyset$, then again the domination yields that u is a forced codeword. Suppose that $N(u) \neq \emptyset$. This implies that if $u \notin C$, then

$$\{u, v\} \subseteq \bigcap_{c \in N(u)} N[c] \subseteq \bigcap_{c \in I(C;u)} N[c].$$

Therefore, as the previous intersection does not consist of a single vertex, the vertex u belongs to C and is a forced codeword.

Suppose then to the contrary that for any vertex $v \in V$ other than u we have $N(u) \not\subseteq N[v]$, i.e., $N(u) \setminus N[v] \neq \emptyset$. Now choosing $C = V \setminus \{u\} (\neq \emptyset)$, we have $I(C; u) = N(u)$ and

$$\bigcap_{c \in I(C;u)} N[c] = \{u\}.$$

Therefore, by the definition, C is a self-locating-dominating code in G. Thus, u is not a forced codeword and we have a contradiction with the supposition. This concludes the proof of the theorem. □

By the previous theorem, we immediately observe that there exist graphs such that all the vertices are forced codewords. For example, the complete graphs and the complete bipartite graphs, where both independent sets of the partition have at least two vertices, are such extreme graphs.

2.1 Results based on Sperner's Theorem

One of the fundamental results on locating-dominating codes by Slater [24] says that if G is a graph with n vertices and $\gamma^{LD}(G) = k$, then $n \leq k + 2^k - 1$. This result is based on the simple fact that using k codewords at most $2^k - 1$ distinct, non-empty I-sets can be formed. In what follows, we present an analogous result for self-locating-dominating and solid-locating-dominating codes. However, here it is not enough that all the I-sets are non-empty and unique, but we further require that none of the I-sets is included in another one. For this purpose, we present Sperner's theorem, which considers the maximum number of subsets of a finite set such that none of the subsets is included in another subset. Sperner's theorem has originally been presented in [26], and for more recent developments regarding the Sperner theory, we refer to [8].

Theorem 10 (Sperner's theorem [26])**.** *Let N be a set of k elements and let $\mathcal{F}$ be a family of subsets of N such that no member of $\mathcal{F}$ is included in another member of $\mathcal{F}$, i.e., for all distinct $X, Y \in \mathcal{F}$ we have $X \setminus Y \neq \emptyset$. Then we have*

$$|\mathcal{F}| \leq \binom{k}{\lfloor \frac{k}{2} \rfloor}.$$

Moreover, the equality holds if and only if $\mathcal{F} = \{X \subseteq N \mid |X| = k/2\}$ when n is even, and $\mathcal{F} = \{X \subseteq N \mid |X| = (k-1)/2\}$ or $\mathcal{F} = \{X \subseteq N \mid |X| = (k+1)/2\}$ when n is odd.

A family of subsets satisfying the conditions of the previous theorem is called a *Sperner family*. In the following theorem, we apply Sperner's theorem to obtain an upper bound on the order of a graph based on the number of codewords in a solid-locating-dominating (or self-locating-dominating) code.

Theorem 11. *Let G be a graph with n vertices and C be a solid-locating-dominating code in G with k codewords. Then we have the following upper bound on the order of G:*

$$n \leq k + \binom{k}{\lfloor \frac{k}{2} \rfloor}.$$

Proof: Let C be a solid-locating-dominating code in G with k codewords. By the definition, for any distinct $u, v \in V \setminus C$, we have $I(C;u) \setminus I(C;v) \neq \emptyset$. Therefore, the I-sets of non-codewords of G form a Sperner family of subsets of C. Thus, by Sperner's theorem, we obtain that

$$|V \setminus C| = n - k \leq \binom{k}{\lfloor \frac{k}{2} \rfloor}.$$

Hence, the claim immediately follows. □

Observe that the previous theorem also holds for self-locating-dominating codes due to Corollary 7. Furthermore, the upper bound of the theorem can be attained even for self-locating-dominating codes as is shown in the following example.

Example 12. Let k be a positive integer and ℓ be an integer such that $\ell = \binom{k}{\lfloor k/2 \rfloor}$. Consider then a bipartite graph G with the vertex set $U \cup V$, where $U = \{u_1, u_2, \ldots, u_k\}$ and $V = \{v_1, v_2, \ldots, v_\ell\}$. There are no edges within the sets U and V, and the edges between the two sets are defined as follows. Let $\mathcal{F}$ be a (maximum) Sperner family of U attaining the upper bound of Theorem 10 with each subset of $\mathcal{F}$ having $\lfloor k/2 \rfloor$ elements. Recall that the number of subsets in $\mathcal{F}$ is ℓ. Denoting the subsets of $\mathcal{F}$ by $F_1, F_2, \ldots, F_\ell$, we define the edges of each v_i as follows: v_i is adjacent to the vertices of F_i.

Now the code $C = U$ is self-locating-dominating in G. Indeed, the I-sets of the non-codewords in V form a Sperner family and, hence, the characterization (i) of Theorem 6 is satisfied. Thus, C is a self-locating-dominating code in G with k codewords and G is a graph with $k + \binom{k}{\lfloor k/2 \rfloor}$ vertices.

In the following immediate corollary of Theorem 11, we give a lower bound on the minimum size of solid-locating-dominating and self-locating-dominating codes based on the order of a graph. Notice also that the obtained lower bounds can be attained by the construction given in the previous example.

Corollary 13. *Let G be a graph with n vertices and let k be the smallest integer such that $n \leq k + \binom{k}{\lfloor k/2 \rfloor}$. Then we have*

$$\gamma^{SLD}(G) \geq \gamma^{DLD}(G) \geq k.$$

2.2 Results using the Dilworth Number

In what follows, we are going to present some results on self-location-domination and solid-location-domination based on certain properties or parameters of graphs. For this purpose, we first present some definitions and notation. Let u and v be distinct vertices of G. We say that u and v are *false twins* if $N(u) = N(v)$ and that u and v are *true twins* if $N[u] = N[v]$. Furthermore, we say that u and v are *twins* if they are false or true twins. Then a graph is called *twin-free* if there does not exist a pair of twin vertices.

The characterization of forced codewords regarding self-location-domination in Theorem 9 motivates us to recall the following definition from [10]. For a graph G, the *vicinal preorder* $\lesssim$ is defined on $V(G)$ as follows:

$$x \lesssim y \text{ if and only if } N(x) \subseteq N[y].$$

In other words, a vertex x is a forced codeword if and only if there exists a vertex y such that $x \lesssim y$ by Theorem 9. It is easy to see that $\lesssim$ is in fact a preorder, that is, a reflexive and transitive relation. We use the following notation:

- $x \sim y$ for ($x \lesssim y$ and $y \lesssim x$),

- $x < y$ for ($x \lesssim y$ and not $y \lesssim x$),

A *chain* is a subset $B \subseteq V(G)$ such that for any two elements x and y of B, $x \lesssim y$ or $y \lesssim x$ must hold. An *antichain* is a subset $A \subseteq V(G)$ such that for any $x, y \in A$, $x \lesssim y$ implies $x = y$. A vertex x is *maximal* if there is no vertex y satisfying $x < y$. The existence of at least a single maximal vertex in the vicinal preorder is guaranteed in every *finite* graph.

Lemma 14. *Let G be a graph of order $n \geq 2$. Then the following statements hold.*

1. *If $x, y \in V(G)$ are neighbours, then $x \sim y$ if and only if x and y are true twins. On the other hand if x and y are not neighbours, then $x \sim y$ if and only x and y are false twins.*

2. *A vertex $x \in V(G)$ is a forced codeword if and only if there exists $y \neq x$ such that $x \lesssim y$.*

As a consequence, we obtain the following properties of extreme graphs, for the self-location-domination number.

Corollary 15. *Let G be a graph of order $n \geq 2$. Then $\gamma^{SLD}(G) = n$ if and only if every maximal vertex in the vicinal preorder has a twin.*

Proof: Suppose that $\gamma^{SLD}(G) = n$, then every vertex of G is a forced codeword. In particular, if x is a maximal vertex in the vicinal preorder, then there exists $y \neq x$ such that $x \lesssim y$. By the maximality of x, we obtain that $x \sim y$ and therefore, x and y are twin vertices. Suppose now that every maximal vertex has a twin, so maximal vertices are forced codewords. Let $u \in V(G)$ be a non-maximal vertex. Consequently, there exists $v \in V(G)$ such that $u \lesssim v$ and u is a forced codeword. Therefore, every vertex in G is a forced codeword and $\gamma^{SLD}(G) = n$, as desired. □

Some graphs satisfying the conditions of the previous corollary are, for example, graphs with at least two vertices with full degree, that is, vertices which are connected to all other vertices. By the previous corollary, we immediately obtain the following result.

Corollary 16. *If G is a twin-free graph of order $n \geq 2$, then we have $\gamma^{SLD}(G) \leq n - 1$.*

In order to characterize graphs having the greatest solid-location-domination number, we will use the Dilworth number, whose definition we quote from [10]. The *Dilworth number* $\nabla(G)$ of a graph G is the minimum number of chains of the vicinal preorder covering $V(G)$. According to the well-known theorem of Dilworth (see [7]), $\nabla(G)$ is equal to the cardinality of the maximum size antichains in the vicinal preorder. In the following results, we describe the relationship between the Dilworth number and the solid-location-domination number.

Lemma 17. *Let G be a graph and C be a solid-locating-dominating code. Then $V(G) \setminus C$ is an antichain of the vicinal preorder.*

Proof: Let $x, y \in V(G) \setminus C$ and suppose that $x \lesssim y$. Hence $N(x) \subseteq N[y]$. Because $x, y \notin C$, we obtain that $I(x) = N[x] \cap C = N(x) \cap C \subseteq N[y] \cap C = I(y)$. Therefore, $x = y$ by the definition of a solid-locating-dominating code. □

Using the previous result, we obtain the following lower bound.

Corollary 18. *Let G be a graph with n vertices. Then $n - \nabla(G) \leq \gamma^{DLD}(G)$.*

Proof: Let C be an optimal solid-locating-dominating code of G. Then the set $V(G) \setminus C$ is an antichain of the vicinal preorder of G and, therefore,

$$n - \gamma^{DLD}(G) = |V(G)| - |C| = |V(G) \setminus C| \leq \nabla(G).$$

□

This lower bound for the solid-location-domination number will allow us to characterize graphs where this parameter reaches its maximum value $n - 1$ among graphs with at least one edge. Recall that a graph is a *threshold graph* [6] if it can be constructed from the empty graph by repeatedly adding either an isolated vertex or a universal vertex (sometimes also called a dominating vertex), i.e., a vertex adjacent to all the existing vertices. It is well known that the following statements are equivalent [19]:

- G is a threshold graph,
- $\nabla(G) = 1$,
- the vicinal preorder in $V(G)$ is total, that is, $V(G)$ is a chain of the vicinal preorder.

In the following proposition, we characterize all the graphs G attaining the maximum solid-location-domination number of $n-1$ (when we have at least one edge in a graph).

Proposition 19. *Let G be a graph of order n and size $m \geq 1$. Then $\gamma^{DLD}(G) = n-1$ if and only if G is a threshold graph.*

Proof: Theorem 8 gives that $\gamma^{DLD}(G) \leq n-1$. If G is a threshold graph, then $\nabla(G) = 1$ and $n-1 = n-\nabla(G) \leq \gamma^{DLD}(G)$. Hence, $\gamma^{DLD}(G) = n-1$.

Suppose now that $\gamma^{DLD}(G) = n-1$ and let $x, y \in V(G)$ be such that $x \neq y$. We will show that $x \lesssim y$ or $y \lesssim x$. Denote $C = V(G) \setminus \{x, y\}$. Observe that C is not a solid-locating-dominating code as $\gamma^{DLD}(G) = n-1$. If $I(C; x) = I(x) = \emptyset$, then $N(x) \subseteq N[y]$ and $x \lesssim y$. Analogously $I(y) = \emptyset$ implies $y \lesssim x$. Assume now that $I(x) \neq \emptyset$ and $I(y) \neq \emptyset$. Because C is not solid-locating-dominating, we obtain $I(x) \subseteq I(y)$ or $I(y) \subseteq I(x)$. We may assume without loss of generality that $I(x) \subseteq I(y)$. Now we have $N(x) \setminus \{y\} = N[x] \cap (V(G) \setminus \{x, y\}) = I(x) \subseteq I(y) = N[y] \cap (V(G) \setminus \{x, y\}) = N(y) \setminus \{x\} \subseteq N(y)$. Therefore $N(x) \subseteq N(y) \cup \{y\}$ or equivalently $x \lesssim y$. For every pair of vertices $x, y \in V(G)$, we have obtained that $x \lesssim y$ or $y \lesssim x$. This means that the vicinal preorder is total or equivalently that G is a threshold graph. □

2.3 Independent Sets and Complements

In what follows, we present upper bounds on the self-location-domination and solid-location-domination numbers based on the independence number and the maximum degree of the graph. Recall that a set $S \subseteq V(G)$ is *independent* in G if no two vertices in S are adjacent. Furthermore, the *independence number* $\beta(G)$ of G is the maximum size of an independent set in G. Moreover, a set S is called 3*-distance-independent* if we have $d(v, u) \geq 3$ for each pair of vertices $v, u \in S$. We denote the maximal size of 3-distance-independent set in G with $\beta_2(G)$. Now we are ready to present the following theorem.

Theorem 20. *Let $G = (V, E)$ be a connected graph on $n \geq 2$ vertices with maximum degree Δ.*

(i) Then we have

$$\gamma^{DLD}(G) \leq n - \beta_2(G) \leq \left\lfloor n\left(1 - \frac{1}{\Delta^2+1}\right)\right\rfloor.$$

(ii) If G has the additional property that $N(u) \not\subseteq N(v)$ for all distinct vertices $u, v \in V$, then

$$\gamma^{DLD}(G) \leq n - \beta(G) \leq \left\lfloor n\left(1 - \frac{1}{\Delta+1}\right)\right\rfloor.$$

(iii) If G has the property that $N(u) \not\subseteq N[v]$ for all distinct vertices $u, v \in V$, then

$$\gamma^{SLD}(G) \leq n - \beta(G) \leq \left\lfloor n\left(1 - \frac{1}{\Delta+1}\right)\right\rfloor.$$

Proof: (i) Let us first consider a set $S \subseteq V$ which is obtained in the following way. Let $T_1 = V$. We choose first any $u_1 \in T_1$ and then we set $T_2 = T_1 \setminus \cup_{v \in N(u_1)} N[v]$. Next we choose $u_2 \in T_2$ and set $T_3 = T_2 \setminus \cup_{v \in N(u_2)} N[v]$. We continue this way by choosing $u_i \in T_i$ and defining $T_{i+1} = T_i \setminus \cup_{v \in N(u_i)} N[v]$ until $T_{i+1} = \emptyset$. Now we denote $S = \{u_1, u_2, \dots\}$ (this is a finite set). Since the maximum degree equals Δ, we know that on each round we remove from T_i at most $\Delta^2 + 1$ vertices. Therefore,

$$|S| \geq \frac{n}{\Delta^2+1}.$$

Next we show that the code $C = V \setminus S$ is solid-locating-dominating. Observe that the distance between two vertices in S (that is, the non-codewords in V) is at least three and hence, S is 3-distance-independent. Consequently, $I(u) \setminus I(v) = N(u) \neq \emptyset$ for any distinct non-codewords u and v (if $|S| = 1$ we are immediately done). Thus,

$$\gamma^{DLD}(G) \leq |C| = n - |S| \leq n\left(1 - \frac{1}{\Delta^2 + 1}\right).$$

(ii) In this case, let S be an independent set in G with $|S| = \beta(G)$. In what follows, we show that the code $C = V \setminus S$ is solid-locating-dominating. Let u and v be any non-codewords. If $d(u, v) \geq 3$, then clearly $I(u) \setminus I(v) \neq \emptyset$ as above. Since S is an independent set, it suffices to assume then that $d(u, v) = 2$. We need to show that $I(u) \setminus I(v) \neq \emptyset$. Notice that now $I(u) = N(u)$ and $I(v) = N(v)$. If $I(u) \setminus I(v) = \emptyset$, then $N(u) \setminus N(v) = \emptyset$, which contradicts the property of the graph. Therefore, we have

$$\gamma^{DLD}(G) \leq n - \beta(G).$$

Furthermore, it is shown in [1, page 278] that

$$|S| = \beta(G) \geq \frac{n}{\Delta + 1}.$$

(iii) Let S be as in Case (ii) and $C = V \setminus S$. Take any $u \notin C$, that is, $u \in S$. Again $I(u) = N(u)$. We need to show that

$$\bigcap_{c \in I(u)} N[c] = \{u\}.$$

Assume to the contrary that the intersection contains another vertex, say $v \in V$, besides u. But this implies that $N(u) \subseteq N[v]$ which is not possible. Therefore, the assertion follows. □

The constraints $N(u) \nsubseteq N(v)$ and $N(u) \nsubseteq N[v]$ for all distinct vertices $u, v \in V$ have their purpose in the cases (ii) and (iii) of the previous theorem. For example, if G is a star on n vertices and v, v' are two distinct pendant vertices, then $N(v) \subseteq N(v')$. Moreover, we have $\beta(G) = \gamma^{SLD}(G) = \gamma^{DLD}(G) = n-1$ while $n - \beta(G) = 1$. Observe also that the bound of (i) is now attained since we have $\beta_2(G) = 1$ and $\gamma^{DLD}(G) = n - 1 = n - \beta_2(G)$.

The bounds (ii) and (iii) of Theorem 20 can be attained, for example, when $G = C_t$ is a cycle on $t \geq 5$ vertices. In these cases, we have $\beta(G) = \left\lfloor \frac{t}{2} \right\rfloor$. This implies that $\gamma^{DLD}(G) \leq \gamma^{SLD}(G) \leq \left\lceil \frac{t}{2} \right\rceil$ by the previous theorem. Moreover, let C be a solid-locating-dominating code in a cycle C_t where $t \geq 5$ and let us consider four consecutive vertices $P = \{v_1, v_2, v_3, v_4\}$ of the cycle, where $v_i v_{i+1} \in E$ ($i \in \{1, 2, 3\}$). If v_1 is the only codeword in P, then $I(v_3) = \emptyset$. If v_2 is the only codeword in P, then $I(v_3) \subseteq I(v_1)$. The cases with v_3 and v_4 being the only codewords are symmetric. Hence, we have at least two codewords among every four consecutive vertices and there are t different sets consisting of four consecutive vertices. On the other hand, each codeword belongs to four different sets of consecutive vertices. Therefore, by a double counting argument, we obtain that $4|C| \geq 2t$ and hence, $\gamma^{DLD}(C_t) \geq \left\lceil \frac{t}{2} \right\rceil$. Thus, in conclusion, we have $\gamma^{DLD}(G) = \gamma^{SLD}(G) = \left\lceil \frac{t}{2} \right\rceil$ and the bounds (ii) and (iii) are attained.

We conclude the section by considering self-location-domination and solid-location-domination numbers in a graph and its complement. It has been shown in [11] that in a graph and its complement the (regular) location-domination number always differs by at most one. In the following theorem, we show that a similar result also holds for solid-location-domination number. However, later in Remark 22, it is shown that an analogous result *does not* hold for self-location-domination number.

Theorem 21. *Let G be a graph on at least two vertices and $\overline{G}$ be its complement. We have $|\gamma^{DLD}(G) - \gamma^{DLD}(\overline{G})| \leq 1$ and the optimal codes are of different cardinality if and only if G is a complete or discrete graph.*

Proof: Let C be an optimal solid-locating-dominating code in G and $v \in V(G) \setminus C$. Suppose that $I(G, C; w) \neq C$ for each vertex $w \in V(G) \setminus C$. Hence, $I(\overline{G}, C; w) \neq \emptyset$ for each vertex $w \in V(G) \setminus C$. We have $I(\overline{G}, C; v) = C \setminus I(G, C; v)$ and $I(G, C; v) = C \setminus I(\overline{G}, C; v)$. If there exists a vertex $u \in V(G) \setminus C$

such that $I(\overline{G},C;u) \subseteq I(\overline{G},C;v)$, then $C \setminus I(G,C;u) \subseteq C \setminus I(G,C;v)$ and hence, $I(G,C;v) \subseteq I(G,C;u)$ which is a contradiction. Therefore, C is also a solid-locating-dominating code for $\overline{G}$ and similarly we get that if C' is a solid-locating-dominating code for $\overline{G}$ with no non-codewords adjacent to all codewords, then it is also a solid-locating-dominating code in G.

Let us then suppose that there is a vertex v such that $I(G,C;v) = C$ and $v \in V(G) \setminus C$. We immediately notice that we then have only one non-codeword since if we had another non-codeword u, we would have $I(G,C;u) \subseteq I(G,C;v)$. Furthermore, in $\overline{G}$ we have $N[v] = \{v\}$, vertex v is a codeword and thus, there are no vertices in $V(\overline{G})$ which would contain all codewords in their neighbourhoods. Hence, if we have $\gamma^{DLD}(\overline{G}) \leq |V| - 2$, then by the previous considerations we have $\gamma^{DLD}(G) \leq |V| - 2$ which is a contradiction. Therefore, we may assume that $\gamma^{DLD}(\overline{G}) \geq |V| - 1$. Furthermore, the only graph for which we have $\gamma^{DLD}(\overline{G}) = |V|$ is the discrete graph by Theorem 8 and in that case G is the complete graph. □

In the following remark, it is shown that an analogous result to the previous theorem does not hold for self-locating-dominating codes; in other words, the difference of the self-location-domination number of the graph and its complement can be arbitrarily large.

Remark 22. Consider the graph $G = (V \cup U, E)$ of Example 12 with $k \geq 4$. Form a new graph $G' = (V \cup U, E')$ based on G by adding edges between each pair of distinct vertices of U (the subgraph graph induced by U is now a clique with k vertices). Then each vertex of V is a forced codeword of a self-locating-dominating code by Theorem 9. On the other hand, V is a self-locating-dominating code in G' by the characterization (i) of Theorem 6. Indeed, for any distinct vertices $u \in U$ and $w \in U \cup V$ there exists a vertex $v \in V$ such that $u \in N(v)$ and $w \notin N(v)$ (recall that the open neighbourhoods of the vertices in V form a *maximum* Sperner family). Thus, we have $\gamma^{SLD}(G') = |V| = \binom{k}{\lfloor k/2 \rfloor}$.

Consider then the complement graph $\overline{G'}$. Now the subgraph induced by V is a clique and the intersections $N(v) \cap U$ of all the vertices $v \in V$ form a (maximum) Sperner family with $|N(v) \cap U| = \lceil k/2 \rceil$. Hence, as V induces a clique, all the vertices of U are forced codewords (by Theorem 9). On the other hand, as in Example 12, it can be shown that U is a self-locating-dominating code in $\overline{G'}$. Thus, we have $\gamma^{SLD}(\overline{G'}) = |U| = k$. Therefore, in conclusion, we have shown that $|\gamma^{SLD}(G') - \gamma^{SLD}(\overline{G'})| = \binom{k}{\lfloor k/2 \rfloor} - k$.

3 Trees

In this section, we study both the self-location-domination and the solid-location-domination number in *trees*. We recall the following definition from [2]. A 2-*dominating set* in a graph G is a dominating set S that dominates every vertex of $V \setminus S$ at least twice, i.e., $|I(S;u)| \geq 2$ for all $u \in V \setminus S$. The 2-*domination number* of G, which is the minimum cardinality of a 2-dominating set of G, is denoted by $\gamma_2(G)$. In addition to the 2-domination number, also the independence number $\beta(G)$ will play a role in this section. In general, both parameters are non-comparable, i.e., there are graphs where either of these values can be larger than the other one. However, we have $\beta(T) \leq \gamma_2(T)$ for every tree T by [2]. We will prove that, in the case of trees, self-locating dominating codes are precisely the 2-dominating sets, and therefore the associated parameters also agree. We will also show that solid-location-domination number equals independence number in trees, in spite of associated sets *are not agreeing* in general.

First of all, we focus on the relationship between self-locating-dominating codes and 2-dominating sets. However, we require the concept of *girth* of a graph G, that is, the length of shortest cycle in G. The graphs without cycles are considered to have an infinite girth.

Lemma 23. *Let G be a graph.*

(i) Every self-locating-dominating code in G is a 2-dominating set.

(ii) If the girth of G is at least 5, then every 2-dominating set of G is a self-locating-dominating code.

Proof: (i) Let C be a self-locating-dominating code. If there exists $u \in V(G) \setminus C$ such that $I(u) = \{v\}$, then $v \in \bigcap_{c \in I(u)} N[c]$, which is not possible. Hence, C is a 2-dominating set.

(ii) Let G be a graph with girth at least 5, C be a 2-dominating set and u belong to $V(G) \setminus C$. By the hypothesis, there exist $c_1, c_2 \in I(u)$, $c_1 \neq c_2$, and since G contains no cycles of length three, we know that c_1 is not a neighbour of c_2. Suppose that there exists $v \neq u$ such that $v \in \bigcap_{c \in I(u)} N[c]$, so $v \in N[c_1] \cap N[c_2]$. Moreover $v \neq c_1, c_2$, because c_1c_2 is not an edge of G. Again because G has no triangles, we obtain that u is not a neighbour of v and therefore the vertex subset $\{u, c_1, v, c_2\}$ induces a 4-cycle, a contradiction. $\square$

The following corollary is an immediate consequence of the previous lemma and, in particular, it can be applied to every tree.

Corollary 24. *Let G be a graph with girth at least 5. Then $\gamma^{SLD}(G) = \gamma_2(G)$.*

In what follows, we briefly discuss the previous requirement stating that the girth of the graph is at least 5. Let us consider a graph $G = (V, E)$ where $V = K \cup P$, $K = \{v_1, v_2\}$, $P = \{u_1, \ldots, u_p\}$, $p \geq 2$, and we have $E = \{v_iu_j \mid v_i \in K, u_j \in P\}$. The graph G has girth 4, it has a 2-dominating set K and the unique self-locating-dominating code C consists of whole V. Therefore, the requirement of girth at least 5 is not only needed but removing it may cause an arbitrarily large difference between $\gamma_2(G)$ and $\gamma^{SLD}(G)$.

A particular case of trees are paths, where 2-dominating numbers are known [20]. Therefore, using the above corollary, we obtain:

Corollary 25. *Let $n \geq 2$ and P_n be a path. Then we have*

$$\gamma^{SLD}(P_n) = \gamma_2(P_n) = \left\lceil \frac{n+1}{2} \right\rceil.$$

We now study the behaviour of the solid-location-domination number in trees, and we prove that it agrees with the independence number. We will need the following notation. A vertex in a tree T is a *leaf* if it is of degree one and a vertex is a *support vertex* if there is at least one leaf in its neighbourhood. If u is a support vertex, then L_u will denote the set of leaves attached to it. In the following lemma we recall a result from [2].

Lemma 26 (Lemma 3 of [2])**.** *Let T be a tree and let u be a support vertex in T such that $|N(u) \setminus L_u| = 1$. If $T' = T - (L_u \cup \{u\})$, then $\beta(T') = \beta(T) - |L_u|$.*

A similar result can be proved for the solid-location-domination number, as we show in the following lemma.

Lemma 27. *Let T be a tree and let u be a support vertex in T such that $|N(u) \setminus L_u| = 1$. If $T' = T - (L_u \cup \{u\})$, then $\gamma^{DLD}(T') = \gamma^{DLD}(T) - |L_u|$.*

Proof: Denote by v the unique non-leaf neighbour of u and let C be an optimal solid-locating-dominating code in T. If $u \notin C$, then clearly $L_u \subseteq C$, to keep the domination. If $u, v \in C$ then, by minimality of C, there exists exactly one vertex in $L_u \setminus C$. And if $u \in C$ and $v \notin C$, then $L_u \subseteq C$, by definition of solid-locating-dominating code and, in this case, we define $C^* = (C \setminus \{u\}) \cup \{v\}$, which can be straightforwardly shown to be an optimal solid-locating-dominating code in T with $|C^*| = |C|$.

In all the cases, we have an optimal solid-locating-dominating code C in T such that $|C \cap (L_u \cup \{u\})| = |L_u|$. Note that in all cases $C' = C \setminus (L_u \cup \{u\})$ is a solid-locating-dominating code of $T' = T \setminus (L_u \cup \{u\})$. Hence, we have

$$\gamma^{DLD}(T') \leq |C'| = |C| - |L_u| = \gamma^{DLD}(T) - |L_u|.$$

Suppose that $\gamma^{DLD}(T') < \gamma^{DLD}(T) - |L_u|$ and let C'' be a solid-locating-dominating code in T' with $|C''| = \gamma^{DLD}(T') < \gamma^{DLD}(T) - |L_u|$. If $v \notin C''$, define $D = C'' \cup L_u$. If $v \in C''$, pick a leaf $x \in L_u$ and define $D = C'' \cup (L_u \setminus \{x\}) \cup \{u\}$. In both cases, we obtain a solid-locating-dominating code D of T that satisfies $|D| = |C''| + |L_u| < (\gamma^{DLD}(T) - |L_u|) + |L_u| = \gamma^{DLD}(T)$, which is a contradiction. $\square$

We can now prove the following result that gives the desired equality between the solid-location-domination number and the independence number in trees.

Proposition 28. *Let T be a tree. Then $\gamma^{DLD}(T) = \beta(T)$.*

Proof: If $T = K_{1,n-1}$ is a star with n vertices, then it is clear that $\gamma^{DLD}(T) = \beta(T) = n - 1$. Assume now that T is not a star. We proceed by induction on $n = |V(T)|$. The result is trivially true if $n = 1$ or $n = 2$. Let $n \geq 3$ be an integer and assume that the statement is true for trees with at most $n - 1$ vertices.

Since T is not a star, there exists a support vertex u such that $|N(u) \setminus L_u| = 1$. By Lemma 27, the tree $T' = T - (L_u \cup \{u\})$ satisfies $\gamma^{DLD}(T') = \gamma^{DLD}(T) - |L_u|$. The inductive hypothesis gives that $\gamma^{DLD}(T') = \beta(T')$ and, by Lemma 26, we know that $\beta(T') = \beta(T) - |L_u|$. Therefore $\gamma^{DLD}(T) = \gamma^{DLD}(T') + |L_u| = \beta(T') + |L_u| = \beta(T)$, as desired. $\square$

In the particular case of paths, independence number is known ([13, Lemma 4]), and therefore:

Corollary 29. *We have*

$$\gamma^{DLD}(P_n) = \beta(P_n) = \left\lceil \frac{n}{2} \right\rceil .$$

In the above proposition, we have shown that $\gamma^{DLD}(T) = \beta(T)$. Previously, the independence number has been extensively studied, and due to [3], it is known that $\beta(G) \geq (n + \ell(G) - s(G))/2$. This lower bound immediately gives the following corollary. Observe that the lower bound can be attained by any path with even number of vertices (by the previous corollary). Moreover, this bound has been studied together with location-domination number, independence number and 2-domination number in [4].

Corollary 30. *Let T be tree of order n, with $\ell(T)$ leaves and $s(T)$ support vertices. Then $\gamma^{DLD}(T) \geq (n + \ell(T) - s(T))/2$.*

We have proved that in every tree, self-locating-dominating codes are exactly 2-dominating sets and this gives the equality between associated parameters as shown in Corollary 24. However, the equality between solid-location-domination number and independence number in trees as shown in Proposition 28, does not imply any general relationship between minimum solid-locating-dominating codes and maximum independent sets.

The path with six vertices satisfies $\gamma^{DLD}(P_6) = \beta(P_6) = 3$. In Figure 3(a), we show all maximum independent sets in P_6 (squared vertices). In Figure 3(b), we show all minimum solid-locating-dominating codes (squared vertices) in the same graph. In this case, none of the maximum independent sets is solid-locating-dominating and none of the minimum solid-locating-dominating codes is independent. However, occasionally the optimal solid-locating-dominating may also be an independent set like in the case of P_3 with the middle vertex as the non-codeword.

(a) Maximum independent sets in P_6 are denoted by the squared vertices.

(b) Minimum solid-locating-dominating codes in P_6 are denoted by the squared vertices.

Fig. 3: Maximum independent sets and minimum solid-locating-dominating codes are different.

4 Cartesian products and ladders

In this section, we consider self-location-domination and solid-location-domination in the *Cartesian product of graphs*. The Cartesian product of graphs $G = (V(G), E(G))$ and $H = (V(H), E(H))$ is $G \square H = (V(G) \times V(H), E)$ where $(u, v)(u', v') \in E$ if and only if $u = u'$ and $vv' \in E(H)$ or $uu' \in E(G)$ and $v = v'$. We begin by presenting a theorem which gives lower and upper bounds for the self-location-domination and the solid-location-domination numbers for Cartesian products. Then we proceed by studying these numbers more closely in the Cartesian products $P_n \square P_2$, where P_k denotes a path with k vertices. Using these results concerning $P_n \square P_2$ and some other previously known ones for the Cartesian product of two complete graphs (see [16]), we are able to show that most of the obtained lower and upper bounds can be attained.

Theorem 31. *We have*

(i) $\max\{\gamma^{SLD}(G),\gamma^{SLD}(H)\} \leq \gamma^{SLD}(G\Box H) \leq \min\{|V(H)|\gamma^{SLD}(G),|V(G)|\gamma^{SLD}(H)\}$ *and*

(ii) $\max\{\gamma^{DLD}(G),\gamma^{DLD}(H)\} \leq \gamma^{DLD}(G\Box H) \leq \min\{|V(H)|\gamma^{DLD}(G),|V(G)|\gamma^{DLD}(H)\}$.

Proof: (i) Let us first show the upper bound on $\gamma^{SLD}(G\Box H)$. Without loss of generality, we may assume that $|V(H)|\gamma^{SLD}(G) \leq |V(G)|\gamma^{SLD}(H)$. Let C be a self-locating-dominating code in G attaining $\gamma^{SLD}(G)$. Denote $D = \{(c,v) \mid c \in C, v \in V(H)\}$. Clearly, $|D| = |V(H)|\gamma^{SLD}(G)$. We will show that D is self-locating-dominating in $G\Box H$. We denote, for any $h \in V(H)$, the set $\{(u,h) \mid u \in V(G)\}$ by L_h and we call it a *layer*. Observe that for any vertices (u_1,h) and (u_2,h) $(u_1 \neq u_2)$ in the same layer L_h we have $N[(u_1,h)] \cap N[(u_2,h)] \subseteq L_h$. Let x be any non-codeword in $G\Box H$, say $x = (u,h) \in L_h$ for some h and $u \in V(G) \setminus C$. Now the codewords in $I(G\Box H;x)$ all belong to L_h. Since C is self-locating-dominating in G, we get (by the previous observation) that

$$\bigcap_{c \in I(G\Box H;x)} N[c] = \{x\}.$$

Next we consider the lower bound on $\gamma^{SLD}(G\Box H)$. Without loss of generality, say $\gamma^{SLD}(G) \geq \gamma^{SLD}(H)$. Let D be a self-locating-dominating code in $G\Box H$ of cardinality $\gamma^{SLD}(G\Box H)$. Denote by $C(\subseteq V(G))$ the set which is obtained by collecting all the first coordinates from D. We claim that C is self-locating-dominating in G. Let $u \in V(G)$ be a non-codeword with respect to C. This implies that the vertices (u,h) are non-codewords with respect to D for all $h \in V(H)$ and hence, $I((u,h)) \subseteq L_h$. Since D is self-locating-dominating, we know that for any layer L_h the neighbourhoods of the codewords in $I(G\Box H, D;(u,h))$ intersect uniquely in (u,h). Because the first coordinates of the codewords in $I(G\Box H, D;(u,h))$ belong to $I(G,C;u)$, we obtain

$$\bigcap_{c \in I(G,C;u)} N[c] = \{u\}.$$

Thus C is self-locating-dominating and the claim follows by noticing that $|C| \geq \gamma^{SLD}(G)$.

(ii) We can again assume without loss of generality that $|V(H)|\gamma^{DLD}(G) \leq |V(G)|\gamma^{DLD}(H)$. Let C be a solid-locating-dominating code in G attaining $\gamma^{DLD}(G)$ and denote again $D = \{(c,v) \mid c \in C, v \in V(H)\}$. In order to verify that D is solid-locating-dominating in $G\Box H$, we show that $I(D;x) \setminus I(D;y)$ is non-empty for any distinct non-codewords $x,y \in V(G\Box H)$. Denote $x = (u,h)$ and $y = (v,h')$ for some $u,v \in V(G)$ and $h,h' \in V(H)$. If $h = h'$, then we are done, since C is solid-locating-dominating. If $h \neq h'$, then the claim follows from the fact that $I(D;x)$ contains a codeword in the layer L_h and $I(D;y)$ cannot contain that codeword (since x and y are non-codewords).

The proof of the lower bound is again similar — let D be a solid-locating-dominating code in $G\Box H$ of cardinality $\gamma^{DLD}(G\Box H)$ and C be a set obtained from its first coordinates. Now let $u \in V(G)$ and $v \in V(G)$ be non-codewords with respect to C. This implies that the vertices (u,h) and (v,h') are non-codewords with respect to D for all $h,h' \in H$. Since D is solid-locating-dominating, we must have that $I(D;(u,h)) \setminus I(D;(v,h))$ contains a codeword (c,h) of D in the layer L_h. Therefore, $c \in I(C;u) \setminus I(C;v)$ in G and we are done. □

Remark 32. Due to Corollary 15, for any complete graph K_m, we have $\gamma^{SLD}(K_m) = m$. Moreover, it has been shown in [16] that $\gamma^{SLD}(K_m \Box K_n) = m$ for $m \geq 2n$. Therefore, the lower bound of Case (i) of the previous theorem can be attained.

In what follows, we focus on the self-location-domination and solid-location-domination numbers in the Cartesian product of paths P_n and P_2. Using these results, we are able to show that the upper bounds in Cases (i) and (ii) can be attained.

The Cartesian product of the paths P_n and P_2 will be called the *ladder (graph)* of length n. Furthermore, we use the following notation for the vertex sets of P_n and P_2: $V(P_n) = \{v_1, v_2, v_3, \ldots v_n\}$ and $V(P_2) = \{1,2\}$, and so the vertex set of the Cartesian product $P_n \Box P_2$ is $V(P_n \Box P_2) = \{(v_i,j)\colon 1 \leq i \leq n,\ 1 \leq j \leq 2\}$ (see Figure 4).

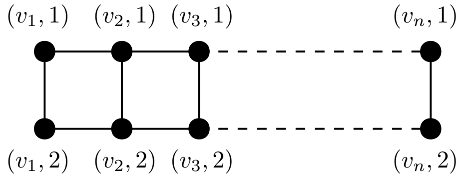
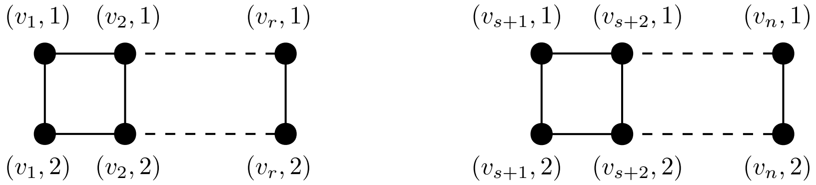


Fig. 4: The ladder $P_n\square P_2$.

The following notation will be useful in this section. Let $1 \le r \le n$ be an integer. Now $P_r\square P_2$ is the subgraph of $P_n\square P_2$ induced by the vertex set $\{(v_i,j)\colon 1 \le i \le r,\ 1 \le j \le 2\}$ (see Figure 5(a)), which is a ladder of length r. On the other hand, for an integer $0 \le s \le n-1$, $P_{n-s}\square P_2$ is the subgraph of $P_n\square P_2$ induced by $\{(v_i,j)\colon s+1 \le i \le n,\ 1 \le j \le 2\}$ (see Figure 5(b)), which is a ladder of length $n-s$.



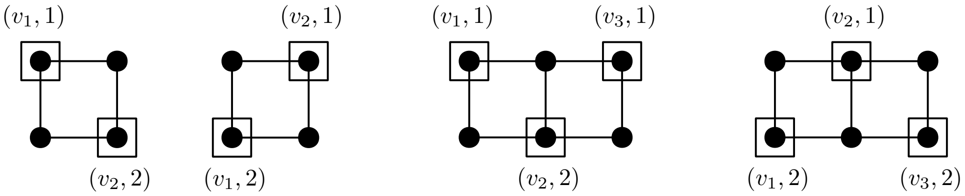
(a) Subgraph $P_r\square P_2$. (b) Subgraph $P_{n-s}\square P_2$.

Fig. 5: Induced subgraphs in $P_n\square P_2$.

We begin by computing the self-location-domination number of ladders. To this end, we will use the relationship between self-locating-dominating codes and 2-dominating sets that we showed in Lemma 23. It is known that $\gamma_2(P_n\square P_2) = n$ (see [20, 23]) for $n \ge 2$ and $\gamma_2(P_1\square P_2) = 2$. In the next lemma, we prove an additional property of optimal 2-dominating sets that will be useful to our purpose.

Lemma 33. *Let $n \ge 2$ be an integer and let $C \subseteq V(P_n\square P_2)$ be a 2-dominating set such that $|C| = n$. Then we have $\{(v_1,1),(v_1,2)\} \nsubseteq C$.*

Proof: There are exactly two 2-dominating sets in $P_2\square P_2$ with two vertices (see Figure 6(a)) and exactly two 2-dominating sets in $P_3\square P_2$ with three vertices (see Figure 6(b)). Therefore, the statement is clearly true for $n = 2$ and $n = 3$.



(a) 2-dominating sets with two vertices in $P_2\square P_2$. (b) 2-dominating sets with three vertices in $P_3\square P_2$.

Fig. 6: Squared vertices are in the 2-dominating set.

We now proceed by induction on n. Assume the statement is true for $k < n$, $n \ge 4$, and let C be a 2-dominating set of $P_n\square P_2$ such that $|C| = n$. Suppose to the contrary that $\{(v_1,1),(v_1,2)\} \subseteq C$, then, because C has n elements, there exists $i \in \{2,\dots,n\}$ such that $(v_i,1),(v_i,2) \notin C$. Note that $i < n$, because C is 2-dominating. Now $\{(v_{i-1},1),(v_{i-1},2),(v_{i+1},1),(v_{i+1},2)\} \subseteq C$ to keep the 2-domination.

Consider the induced subgraphs $G_1 = P_{i-1}\square P_2$ and $G_2 = P_{n-i}\square P_2$. It is clear that $C \cap V(G_1)$ and $C \cap V(G_2)$ are 2-dominating sets in G_1 and G_2, respectively. Moreover, they satisfy $(v_{i-1},1),(v_{i-1},2) \in C \cap V(G_1)$ and $(v_{i+1},1),(v_{i+1},2) \in C \cap V(G_2)$.

Suppose that $i-1 \geq 2$ and $n-i \geq 2$. By the inductive hypothesis $|C \cap V(G_1)| \geq (i-1)+1 = i$ and $|C \cap V(G_2)| \geq (n-i)+1$. Therefore, $|C| \geq n+1$, which is a contradiction. Assume now that $i-1 = 1$ and $n-i = n-2$. In this case $|C \cap V(G_1)| = 2$ and, by the inductive hypothesis, $|C \cap V(G_2)| \geq (n-2)+1 = n-1$. Again $|C| \geq n+1$, a contradiction. The remaining case, $i-1 = n-2$ and $n-i = 1$, is similar to the previous one. Therefore, $\{(v_1,1),(v_1,2)\} \nsubseteq C$ as desired. □

This property gives that self-locating-dominating codes of ladders $P_n \Box P_2$ are non-optimal 2-dominating sets for $n \geq 2$.

Lemma 34. *Let C be a self-locating-dominating code in $P_n \Box P_2$ with $n \geq 2$. Then $(v_1,1),(v_1,2)$, $(v_n,1),(v_n,2) \in C$ and $|C| \geq n+1$.*

Proof: By Lemma 23, C is a 2-dominating set in $P_n \Box P_2$. Hence, we have $|C| \geq n$. Suppose that $(v_1,1) \notin C$. Now $(v_1,2),(v_2,1) \in C$ and $(v_2,2) \in \bigcap_{c \in I((v_1,1))} N[c]$, which is not possible for a self-locating-dominating code. So $(v_1,1) \in C$ and analogously $(v_1,2),(v_n,1),(v_n,2) \in C$. Using Lemma 33, we obtain that $|C| \geq n+1$. □

We can now compute the exact self-location-domination numbers of ladders.

Theorem 35. *Let $n \geq 2$ be an integer. Then*

$$\gamma^{SLD}(P_n \Box P_2) = \begin{cases} n+1 & \text{if } n \text{ is odd;} \\ n+2 & \text{if } n \text{ is even.} \end{cases}$$

Proof: If $n = 2k+1$, $k \geq 1$, then the set $\{(v_{2i+1},1),(v_{2i+1},2) \colon 0 \leq i \leq k\}$ (see Figure 7) is a self-locating-dominating code with $2(k+1) = n+1$ vertices. Thus, Lemma 34 gives $\gamma^{SLD}(P_{2k+1} \Box P_2) = (2k+1)+1$.

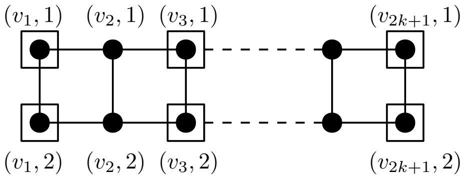


Fig. 7: Squared vertices form an optimal self-locating-dominating code in $P_{2k+1} \Box P_2$.

Assume now that $n = 2k$, $k \geq 1$. We will prove that $\gamma^{SLD}(P_{2k} \Box P_2) \geq 2k+2$, by induction on k. Clearly, $\gamma^{SLD}(P_2 \Box P_2) = 4$ (by the proof of Lemma 34). Assume that the statement is true for $r < k$, $k \geq 2$, and let $C \subseteq V(P_{2k} \Box P_2)$ be a self-locating-dominating code. Suppose that $\{(v_i,1),(v_i,2)\} \cap C \neq \emptyset$ for every $i \in \{1,2,\ldots,2k\}$. By Lemma 34, $(v_1,1),(v_1,2),(v_{2k},1)$, $(v_{2k},2) \in C$. So $|C| \geq 2k+2$.

Suppose now that there exists $i \in \{2,\ldots,2k-1\}$ such that $(v_i,1),(v_i,2) \notin C$. Since C is also a 2-dominating set, we obtain that $\{(v_{i-1},1),(v_{i-1},2),(v_{i+1},1),(v_{i+1},2)\} \subseteq C$. Consider the induced subgraphs $G_1 = P_{i-1} \Box P_2$ and $G_2 = P_{2k-i} \Box P_2$, with self-locating-dominating codes $C \cap V(G_1)$ and $C \cap V(G_2)$, respectively. Note that $(i-1)+(2k-i)$ is odd. Let us assume that $i-1$ is odd and $2k-i$ is even (the other case is analogous). For $i-1$ odd, we know that $|C \cap V(G_1)| \geq (i-1)+1$ (this holds also for $i = 2$) and when $2k-i$ is an even number, by the inductive hypothesis, $|C \cap V(G_2)| \geq (2k-i)+2$. This gives $|C| \geq 2k+2$.

Finally the set $\{(v_{2i+1},1),(v_{2i+1},2) \colon 0 \leq i \leq k-1\} \cup \{(v_{2k},1),(v_{2k},2)\}$ (see Figure 8) is a self-locating-dominating code of $P_{2k} \Box P_2$ with $2k+2$ vertices and therefore $\gamma^{SLD}(P_{2k} \Box P_2) = 2k+2$. □

Remark 36. Recall that we have $\gamma^{SLD}(P_n) = \gamma_2(P_n) = \lceil (n+1)/2 \rceil$ by Corollary 25. Notice then that, whether n is even, with $n = 2k$, or odd, with $n = 2k+1$, we have shown that $\gamma^{SLD}(P_n \Box P_2) = 2k+2 = 2(k+1) = |V(P_2)|\gamma^{SLD}(P_n)$. Therefore, the upper bound of Case (i) of Theorem 31 is attained for both even and odd n.

We now focus on the solid-location-domination number of ladders.

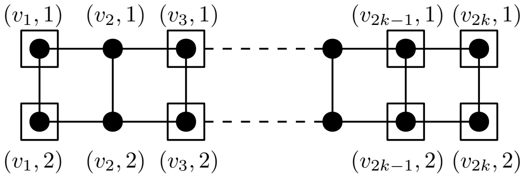


Fig. 8: Squared vertices form an optimal self-locating-dominating code in $P_{2k}\square P_2$.

Proposition 37. *Let $n \geq 1$ be an integer and $C \subseteq V(P_n\square P_2)$ be a solid-locating-dominating code. Then $|C| \geq n$.*

Proof: The statement is clearly true for $n = 1$ and $n = 2$. Assume now that $n \geq 3$ and the claim is true for every $k < n$. Let $C \subseteq V(P_n\square P_2)$ be a solid-locating-dominating code and suppose to the contrary that $|C| < n$. Then there exists $i \in \{1, 2, \ldots, n\}$ such that $(v_i, 1), (v_i, 2) \notin C$.

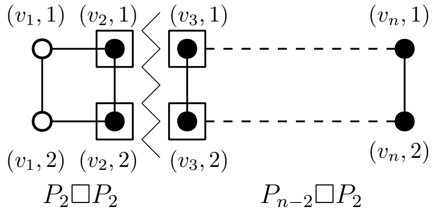


Fig. 9: Squared vertices are in C and white vertices are not in C.

Firstly suppose that $(v_1, 1), (v_1, 2) \notin C$, then using the fact that C is a solid-locating-dominating code we obtain that $(v_2, 1), (v_2, 2), (v_3, 1), (v_3, 2) \in C$. Consider the induced subgraphs $G_1 = P_2\square P_2$ and $G_2 = P_{n-2}\square P_2$. Clearly both $C \cap V(G_1)$ and $C \cap V(G_2)$ are solid-locating-dominating codes in G_1 and G_2 respectively (see Figure 9). Moreover, $|C \cap V(G_1)| = 2$ and by the inductive hypothesis $|C \cap V(G_2)| \geq n-2$. Hence, we have $|C| \geq n$ (a contradiction). Therefore, we may assume that $1 < i$ and, with the same reasoning, that $i < n$.

Assume now that $(v_{i+1}, 1), (v_{i+1}, 2) \notin C$. Consequently, the vertices $(v_{i-1}, 1)$, $(v_{i-1}, 2)$, $(v_{i+2}, 1)$ and $(v_{i+2}, 2)$ belong to C. Consider the induced subgraphs $G_1 = P_i\square P_2$ and $G_2 = P_{n-i}\square P_2$ with solid-locating-dominating codes $C \cap V(G_1)$ and $C \cap V(G_2)$, respectively (see Figure 10(a)). By the inductive hypothesis $|C \cap V(G_1)| \geq i$ and $|C \cap V(G_2)| \geq n - i$, so $|C| \geq n$, which is a contradiction. A similar argument can be used if $(v_{i-1}, 1), (v_{i-1}, 2) \notin C$.

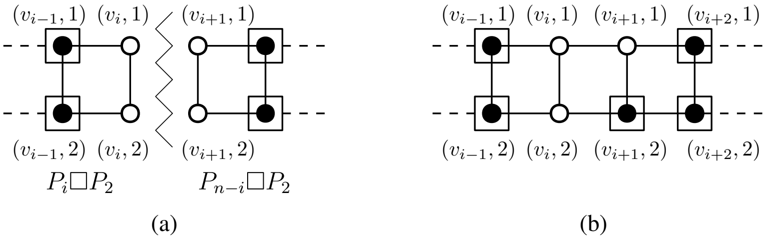


Fig. 10: Squared vertices are in C and white vertices are not in C.

Suppose that $(v_{i+1}, 1) \notin C$ and $(v_{i+1}, 2) \in C$. Clearly, $i + 2 \leq n$ and $(v_{i+2}, 1) \in C$ because otherwise $I((v_{i+1}, 1)) \subseteq I((v_i, 2))$. Moreover, $(v_{i+2}, 2) \in C$ because otherwise $I((v_{i+1}, 1)) \subseteq I((v_{i+2}, 2))$. Furthermore, $(v_{i-1}, 1) \in C$ since $I((v_i, 1)) \neq \emptyset$ and $(v_{i-1}, 2) \in C$ as otherwise $I((v_i, 1)) \subseteq I((v_{i-1}, 2))$. Hence, there are two pairs of codewords $\{(v_{i-1}, 1), (v_{i-1}, 2)\}$ and $\{(v_{i+2}, 1), (v_{i+2}, 2)\}$ such that the pair of non-codewords $\{(v_i, 1), (v_i, 2)\}$ is between them (see Figure 10(b)).

We have stated that there cannot be a pair of non-codewords at the beginning or the end of the ladder. Furthermore, we may assume that there are no consecutive pairs of non-codewords and, if there is a pair of non-codewords $(v_i, 1), (v_i, 2)$, then there are two pairs of codewords $\{(v_j, 1), (v_j, 2)\}$ and $\{(v_{j'}, 1), (v_{j'}, 2)\}$, $j < i < j'$, such that if for some i', $j < i' < j'$, we have $(v_{i'}, 1), (v_{i'}, 2) \notin C$, then $i' = i$. Hence, the number of vertex pairs such that $\{(v_j, 1), (v_j, 2)\} \subseteq C$ is greater or equal to the number of vertex pairs $(v_i, 1), (v_i, 2) \notin C$ and thus, we have $|C| \geq n$. □

We can finally determine the exact solid-location-domination numbers of all ladders.

Corollary 38. *If* $n \geq 1$ *is an integer, then* $\gamma^{DLD}(P_n \square P_2) = n$.

Proof: The set $C = \{(v_i, 1) \colon 1 \leq i \leq n\}$ is a solid-locating-dominating code in $P_n \square P_2$ with n elements since $I((v_i, 2)) = \{(v_i, 1)\}$. So, $\gamma^{DLD}(P_n \square P_2) \leq n$. The reverse inequality comes from Proposition 37. □

Remark 39. Recall that we have $\gamma^{DLD}(P_n) = \beta(P_n) = \lceil n/2 \rceil$ by Corollary 29. Notice that, if n is an even number, then $\gamma^{DLD}(P_n \square P_2) = 2 \cdot \frac{n}{2} = |V(P_2)|\gamma^{DLD}(P_n)$. Therefore, the upper bound shown in Case (ii) of Theorem 31 is attained in this case.

5 Realization theorems

In this section, we consider location-domination, self-location-domination and solid-location-domination numbers; in particular, we study what are the values the location-domination number can simultaneously have with the self-location-domination or the solid-location-domination number in a graph. Similar types of questions have been previously studied in [5] regarding various values such as domination number, location-domination number and metric dimension. In the following theorem, we characterize which values of location-domination and self-location-domination numbers can be simultaneously achieved in a graph.

Theorem 40. *Let* a *and* b *be positive integers. Then there exists a graph* G *such that* $a = \gamma^{LD}(G)$ *and* $b = \gamma^{SLD}(G)$ *if and only if we have*

$$0 \leq b - a \leq 2^a - 1.$$

Proof: We cannot have $a > b$ since each self-locating-dominating code is also locating-dominating. We also cannot have $b > a + 2^a - 1$ since we can have at most $a + 2^a - 1$ vertices in a graph with locating-dominating code of cardinality a by [25]. Hence, $b - a$ is in the claimed interval. Based on the difference $b - a$, the proof divides into the following cases: (i) $a = b$, (ii) $b - a = 1$, (iii) $a = 2$ and $b = 4$ or $b = 5$, (iv) $2 \leq b - a \leq 2^a - 2$ and $a \geq 3$, and (v) the extremal case $a \geq 3$ and $b - a = 2^a - 1$.

(i) Let us first study the case $a = b$. We can now consider the discrete graph G, that is, the graph with no edges, of order a and we have $\gamma^{SLD}(G) = \gamma^{LD}(G)$.

(ii) Let us then study the case $b = a + 1$. We can consider graph G of order b with one edge and we have $\gamma^{SLD}(G) = \gamma^{LD}(G) + 1$.

(iii) Let us then study the case $a = 2$ and $b = 4$. We immediately notice that these numbers are realized in the graph of Figure 11(a). The case $a = 2$ and $b = 5$ is given in the graph of Figure 11(b).

(a) $\gamma^{LD}(G) = 2, \gamma^{SLD}(G) = 4$.

(b) $\gamma^{LD}(G) = 2, \gamma^{SLD}(G) = 5$.

Fig. 11: Black vertices are in an optimal locating-dominating code and squared vertices are in an optimal self-locating-dominating code.

(iv) Let us then study the case $2 \leq b - a \leq 2^a - 2$ and $a \geq 3$. There is an integer $k \in \mathbb{Z}$ such that $2^{k-1} - 2 < b - a \leq 2^k - 2$. Notice that $2 \leq k \leq a$. Let us have $K = \{v_i \mid 1 \leq i \leq k\}$,

$K' = \{v_i \mid k+1 \leq i \leq a\}$, $P = \{u_i \mid 1 \leq i \leq b+1-a\}$ and $V = K \cup K' \cup P$. We have $|V| = b+1$. Let us connect u_1 to each vertex in V and each vertex u_i, $2 \leq i \leq k+1$ to a single vertex v_j, $j = i-1$. Indeed, the latter edges are possible since for any $k \geq 3$ we have $|P| \geq 2^{k-1} - 1 + 1 \geq k+1$ and if $k = 2$, then $b - a = 2$ and $|P| \geq 3$. Let us further connect each other vertex in P to some proper non-empty subset of vertices in K in such a manner that no two vertices in P have the same neighbourhood in K. This choice of non-identical neighbourhoods is possible since we have $|P| \leq 2^{|K|} - 1 = 2^k - 1$. A graph G with $k = 4$ and $|V| = 16$ is shown in the Figure 12.

Let us first consider self-location-domination in G. There are b forced codewords in a self-locating-dominating code since we have $N(v) \subseteq N[u_1]$ for each $v \in V$. Hence, we have $\gamma^{SLD}(G) \geq b$ and $V \setminus \{u_1\}$ is a self-locating-dominating code since $I(u_2) \cap I(u_3) = \{u_1\}$ and thus, $\gamma^{SLD}(G) = b$.

Let us then consider location-domination in G. We can choose each vertex in $K \cup K'$ as a codeword and have a locating-dominating code of size a. Let us show that $\gamma^{LD}(G) \geq a$. We have $N[u_{j+1}] = \{u_1, u_{j+1}, v_j\}$ for each $1 \leq j \leq k$. Hence, if v_j and v'_j in K are non-codewords, then at least two of vertices u_{j+1}, $u_{j'+1}$ and u_1 are codewords. Furthermore, by the same idea, if we have t non-codewords in K, then there are at least t codewords in P. Hence, we have at least k codewords in $K \cup P$. Since we have $N(v_i) = \{u_1\}$ for each vertex v_i in K', we have at least $|K'|$ codewords in $K' \cup \{u_1\}$. If u_1 is a non-codeword, then it is immediate that we have $|K| + |K'| = a$ codewords in G. Hence, we may assume that u_1 is a codeword. If all the vertices $v_{k+1}, v_{k+2}, \ldots, v_a$ are codewords, then we are again immediately done. Moreover, by the previous observations at most one of the vertices can be a non-codeword. Hence, we may assume that there exists a unique non-codeword v_y, $y \geq k+1$. Now we have $I(v_y) = \{u_1\}$. Therefore, for any $1 \leq j \leq k$ at least one of v_j and u_{j+1} is a codeword as otherwise $I(u_{j+1}) = \{u_1\} = I(v_y)$ (a contradiction). Thus, there exist $|K| = k$ codewords in $K \cup P \setminus \{u_1\}$. Hence, in all the cases, we have $\gamma^{LD}(G) \geq a$.

(v) Let us finally study the extremal case $a \geq 3$ and $b - a = 2^a - 1$. Let us consider graph $G = (V, E)$. Let us have $K = \{v_i \mid 1 \leq i \leq a\}$, $P = \{u_i \mid 1 \leq i \leq b - a\}$ and $V = K \cup P$. Let us connect

1. v_1 to v_i for each $2 \leq i \leq a$
2. u_1 to v_i for each $1 \leq i \leq a$
3. u_i, $2 \leq i \leq a$, to v_1 and v_i
4. u_i, $a+1 \leq i \leq b-a$, to some non-empty subset of vertices of K in such a manner that no two vertices of P have the same neighbourhood in K. This is possible since $|P| \leq 2^{|K|} - 1$.
5. u_1 to u_i, $2 \leq i \leq b-a$, if $u_i \in N(v_1)$
6. u_i, $2 \leq i \leq a$, to u_j, $a+1 \leq j \leq b-a$, if $u_j \in N(v_i)$.

Since we have $2^a + a - 1$ vertices in the graph, we have $\gamma^{LD}(G) \geq a$. On the other hand, we can choose K as an optimal locating-dominating code, since if $u, u' \in P$, then $N(u) \cap K \neq N(u') \cap K$ and $|K| = a$. In order to prove that $\gamma^{SLD}(G) = b = |V|$, we have to show that each vertex of the graph is a forced codeword. By Theorem 9, it suffices to show that for each vertex $v \in V$ there exists another vertex v' such that $N(v) \subseteq N[v']$. Indeed, it is straightforward to verify that $N[v_i] = N[u_i]$ for $1 \leq i \leq a$ and $N(u_i) \subseteq N[u_1]$ for $a+1 \leq i \leq b-a$. Thus, the claim follows. $\square$

In the following theorem, we proceed by characterizing which values of location-domination and solid-location-domination numbers can be simultaneously achieved in a graph.

Theorem 41. *Let a and b be positive integers. Then there exists a graph G such that $a = \gamma^{LD}(G)$ and $b = \gamma^{DLD}(G)$ if and only if we have*

$$0 \leq b - a \leq 2^a - 1 - \binom{a}{\lceil \frac{a}{2} \rceil}.$$

Proof: Let us have a locating-dominating code C_{LD} of cardinality a in $G = (V, E)$. Then all the $|V| - a$ non-codewords $u \in V$ have different and non-empty sets $I(C_{LD}; u)$. Hence, we have $|V| \leq 2^a - 1 + a$.

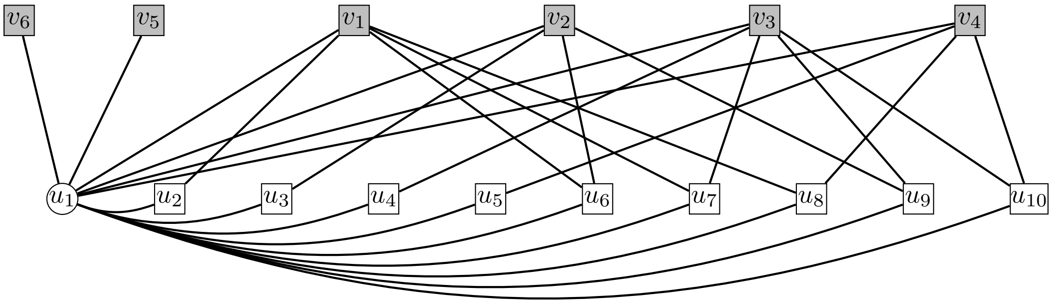


Fig. 12: Graph G with $k = 4$, $\gamma^{LD}(G) = 6$ and $\gamma^{SLD}(G) = 15$. Squared vertices are codewords in an optimal self-locating-dominating code and gray vertices are codewords in an optimal locating-dominating code.

Denote $R = \left\{v \in V \mid |N[v] \cap C_{LD}| = \left\lceil \frac{a}{2} \right\rceil, v \notin C_{LD}\right\}$. Clearly, $0 \leq |R| \leq \binom{a}{\lceil \frac{a}{2} \rceil}$ and there are at most $2^a - 1 + a - \binom{a}{\lceil \frac{a}{2} \rceil}$ vertices in $V \setminus R$ (as the I-sets of the non-codewords have to be non-empty and unique). Denote then $D = V \setminus R$. Now for any distinct pair of vertices $v, w \in R$ we have $I(D; v) \not\subseteq I(D; w)$ because $C_{LD} \subseteq D$ and $|I(C_{LD}; v)| = |I(C_{LD}; w)| = \lceil a/2 \rceil$. Therefore, we have $b = \gamma^{DLD}(G) \leq |D| \leq 2^a - 1 + a - \binom{a}{\lceil \frac{a}{2} \rceil}$. Thus, we obtain that $b - a \leq 2^a - 1 - \binom{a}{\lceil \frac{a}{2} \rceil}$.

Let us first consider the situation $a = b$ and a graph $G = (V, E)$. We notice that this is possible by assuming that G is the star with a pendant vertices. Let us then assume that $a < b$ and let us have $k \in \mathbb{Z}_+$ such that $2^{k-1} - 1 - \binom{k-1}{\lceil \frac{k-1}{2} \rceil} < b - a \leq 2^k - 1 - \binom{k}{\lceil \frac{k}{2} \rceil}$. Since $2^k - 1 - \binom{k}{\lceil \frac{k}{2} \rceil}$ is an increasing function on k when $k > 0$ and it gains value 1 when $k = 2$, each value of difference $b - a$ is linked to a unique value of k. Since the function gives 0 when $k = 1$, we can assume that $k \geq 2$. Let us have $K = \{v_i \mid 1 \leq i \leq k\}$, $K' = \{v_i \mid k + 1 \leq i \leq a\}$, $P = \left\{u_i \mid 1 \leq i \leq b - a + \binom{k}{\lceil \frac{k}{2} \rceil}\right\}$ and $V = K \cup K' \cup P$. We have $|V| = b + \binom{k}{\lceil \frac{k}{2} \rceil}$. Let us connect

1. v_i to v_j for each $1 \leq i < j \leq a$ forming the complete graph K_a,
2. u_1 to v_i for each $1 \leq i \leq a$,
3. each u_i, $2 \leq i \leq \binom{k}{\lceil \frac{k}{2} \rceil} + 1$, to $\left\lfloor \frac{k}{2} \right\rfloor$ vertices in K in such a manner that all of these vertices have different open neighbourhoods,
4. each u_i, $\binom{k}{\lceil \frac{k}{2} \rceil} + 2 \leq i \leq \binom{k}{\lceil \frac{k}{2} \rceil} + k + 1$, to vertex v_j with $j = i - \binom{k}{\lceil \frac{k}{2} \rceil} - 1$ when $k \geq 4$ (and no vertices are connected if $2 \leq k \leq 3$) and
5. each other vertex $u_i \in P$ to some non-empty subset of vertices in K in such a manner that no two vertices of P have the same neighbourhood in K.

Denote the graph constructed above by G. Step 3 is possible since $\binom{k}{\lceil \frac{k}{2} \rceil} = \binom{k}{\lfloor \frac{k}{2} \rfloor}$ and $|P| \geq \binom{k}{\lceil \frac{k}{2} \rceil} + 1$. Furthermore, because $k \geq 2$, u_1 has different neighbourhood in K (compared to the vertices of Step 3). Step 4 is possible since

$$|P| \geq 2^{k-1} - 1 - \binom{k-1}{\lceil \frac{k-1}{2} \rceil} + \binom{k}{\lceil \frac{k}{2} \rceil} + 1 = \sum_{i=0}^{k-1} \binom{k-1}{i} - \binom{k-1}{\lceil \frac{k-1}{2} \rceil} + \binom{k}{\lceil \frac{k}{2} \rceil} \geq \binom{k}{\lceil \frac{k}{2} \rceil} + k + 1$$

when $k \geq 4$. However, when $k = 2$ or $k = 3$, we have $\left\lfloor \frac{k}{2} \right\rfloor = 1$ and thus, by step 3, also in these cases for each vertex $v \in K$ there exists a vertex of P such that it has only v in its neighbourhood. Step 5 is possible since we have $|P| \leq 2^{|K|} - 1 = 2^k - 1$.

Let us first show that the location-domination number of G is equal to a. We can now choose $K \cup K'$ as a locating-dominating code of size a since each vertex in P has its open neighbourhood with a unique and non-empty intersection with K. Furthermore, each vertex in $K' \cup \{u_1\}$ has the same closed neighbourhood and hence, we can have at most one non-codeword among them. Since each vertex $v_j \in K$ neighbours a vertex u_i such that $N(u_i) = \{v_j\}$ for some j and i, we have v_j or u_i in the code and hence, we have at least $|K| + |K'| = a$ codewords. Thus, $\gamma^{LD}(G) = a$.

Let us then show that $\gamma^{DLD}(G) = b$. We can choose as our solid-locating-dominating code C the vertex set containing all vertices except for u_i for $2 \leq i \leq \binom{k}{\lceil \frac{k}{2} \rceil} + 1$. We have $|C| = b$ (as $|V| = b + \binom{k}{\lceil k/2 \rceil}$) and if $v, v' \notin C$, then $|N(v) \cap K| = |N(v') \cap K| = \lfloor \frac{k}{2} \rfloor$ and $N(v) \cap K \neq N(v') \cap K$ and hence, C is a solid-locating-dominating code. In order to show that $\gamma^{DLD}(G) \geq b$, let X be a solid-locating-dominating code in G. Observe that if a vertex $v_h \in K' \cup K$ does not belong to X, then each vertex in P is in the code since for $u_i \in P$ we have $N(u_i) \subseteq K' \cup K \subseteq N[v_h]$. Moreover, since we have $N[v'] \subseteq N[v]$ when $v' \in K'$ and $v \in K' \cup K$, we can have at most one non-codeword in K' and only if there are no non-codewords in K. Therefore, we may assume that $K \cup K' \subseteq X$ since otherwise we would have at least $|P| + |K'| = b + \binom{k}{\lceil \frac{k}{2} \rceil} - k \geq b$ codewords because $\binom{k}{\lceil \frac{k}{2} \rceil} \geq k$. Furthermore, due to the Sperner's theorem and the independence of the set P, we can choose at most $\binom{k}{\lceil \frac{k}{2} \rceil}$ vertices in P in such a manner that none of their neighbourhoods is contained within another. This gives $\gamma^{DLD}(G) \geq b$. □

References

[1] C. Berge. *Graphs and Hypergraphs.* North-Holland, Amsterdam, 1976.

[2] M. Blidia, M. Chellali, and O. Favaron. Independence and 2-domination in trees. *Australas. J. Combin.*, 33:317–327, 2005.

[3] M. Blidia, M. Chellali, O. Favaron, and N. Meddah. On k-independence in graphs with emphasis on trees. *Discrete Math.*, 307(17-18):2209–2216, 2007.

[4] M. Blidia, O. Favaron, and R. Lounes. Locating-domination, 2-domination and independence in trees. *Australas. J. Combin.*, 42:309–316, 2008.

[5] J. Cáceres, C. Hernando, M. Mora, I. M. Pelayo, and M. L. Puertas. Locating-dominating codes: bounds and extremal cardinalities. *Appl. Math. Comput.*, 220:38–45, 2013.

[6] V. Chvátal and P. Hammer. Aggregation of inequalities in integer programming. *Technical Report, Comp. Sci. Dept. Stanford Univ., Stanford, California*, STAN-CS-75-518, 1975.

[7] R. P. Dilworth. A decomposition theorem for partially ordered sets. *Ann. of Math. (2)*, 51:161–166, 1950.

[8] K. Engel. *Sperner theory*, volume 65 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 1997.

[9] N. Fazlollahi, D. Starobinski, and A. Trachtenberg. Connected identifying codes. *IEEE Trans. Inform. Theory*, 58(7):4814–4824, 2012.

[10] S. Földes and P. Hammer. The Dilworth number of a graph. *Ann. Discrete Math.*, 2:211–219, 1978. Algorithmic aspects of combinatorics (Conf., Vancouver Island, B.C., 1976).

[11] C. Hernando, M. Mora, and I. M. Pelayo. Nordhaus-Gaddum bounds for locating domination. *European J. Combin.*, 36:1–6, 2014.

[12] I. Honkala, T. Laihonen. On a new class of identifying codes in graphs. *Inform. Process. Lett.*, 102:92–98, 2007.

[13] M. Jou and J. Lin. Independence numbers in trees. *Open J. Discrete Math.*, 5, 27-31, 2015.

[14] V. Junnila, T. Laihonen. Collection of codes for tolerant location. *Proceedings of the Bordeaux Graph Workshop*, 2016, pp. 176–179.

[15] V. Junnila, T. Laihonen. Tolerant location detection in sensor networks. Submitted for publication, 2016.

[16] V. Junnila, T. Laihonen, and T. Lehtilä. On regular and new types of codes for location-domination. *Discrete Appl. Math.*, 247:225–241, 2018.

[17] M. Laifenfeld and A. Trachtenberg. Disjoint identifying-codes for arbitrary graphs. In *Proceedings of International Symposium on Information Theory, 2005. ISIT 2005*:244–248, 2005.

[18] A. Lobstein. Watching systems, identifying, locating-dominating and discriminating codes in graphs, a bibliography. Published electronically at

`https://www.lri.fr/~lobstein/debutBIBidetlocdom.pdf`

[19] N. V. R. Mahadev and U. N. Peled. *Threshold graphs and related topics*, Volume 56 of *Ann. Discrete Math.*, North Holland, page 241, 1995.

[20] J. J. Mohana and I. Kelkarb. Restrained 2-domination number of complete grid graphs. *Int. J. Appl. Math. Comput.*, 4(4), 2012.

[21] D. F. Rall and P. J. Slater. On location-domination numbers for certain classes of graphs. *Congr. Numer.*, 45:97–106, 1984.

[22] S. Ray, D. Starobinski, A. Trachtenberg, and R. Ungrangsi. Robust location detection with sensor networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1016–1025, August 2004.

[23] R. Shaheen, S. Mahfud, and K. Almanea. On the 2-domination number of complete grid graphs. *Open J. Discrete Math.*, 7:32–50, 2017.

[24] P. J. Slater. Domination and location in acyclic graphs. *Networks*, 17(1):55–64, 1987.

[25] P. J. Slater. Dominating and reference sets in a graph. *J. Math. Phys. Sci.*, 22:445–455, 1988.

[26] E. Sperner. Ein Satz über Untermengen einer endlichen Menge. *Math. Z.*, 27(1):544–548, 1928.

Publication III

Ville Junnila, Tero Laihonen and Tuomo Lehtilä. On a conjecture regarding identification in Hamming graphs, *The Electronic Journal of Combinatorics*, P2, 2019.

On a Conjecture Regarding Identification in Hamming Graphs*

Ville Junnila, Tero Laihonen and Tuomo Lehtilä†

Department of Mathematics and Statistics
University of Turku, FI-20014 Turku, Finland

`viljun@utu.fi, terolai@utu.fi, tualeh@utu.fi`

Submitted: Apr 25, 2018; Accepted: May 29, 2019; Published: Jun 21, 2019

Abstract

In 2013, Goddard and Wash studied identifying codes in the Hamming graphs K_q^n. They stated, for instance, that $\gamma^{ID}(K_q^n) \leqslant q^{n-1}$ for any q and $n \geqslant 3$. Moreover, they conjectured that $\gamma^{ID}(K_q^3) = q^2$. In this article, we show that $\gamma^{ID}(K_q^3) \leqslant q^2 - q/4$ when q is a power of four, which disproves the conjecture. Goddard and Wash also gave the lower bound $\gamma^{ID}(K_q^3) \geqslant q^2 - q\sqrt{q}$. We improve this bound to $\gamma^{ID}(K_q^3) \geqslant q^2 - \frac{3}{2}q$. Moreover, we improve the above mentioned bound $\gamma^{ID}(K_q^n) \leqslant q^{n-1}$ to $\gamma^{ID}(K_q^n) \leqslant q^{n-k}$ for $n = 3\frac{q^k-1}{q-1}$ and to $\gamma^{ID}(K_q^n) \leqslant 3q^{n-k}$ for $n = \frac{q^k-1}{q-1}$, when q is a prime power. For these bounds, we utilize two classes of closely related codes, namely, the self-identifying and the self-locating-dominating codes. In addition, we show that the self-locating-dominating codes satisfy the result $\gamma^{SLD}(K_q^3) = q^2$ related to the above conjecture.

Mathematics Subject Classifications: 94B05, 94B25, 94B65, 05B15, 05C69

Keywords: Hamming graph; identifying code; linear codes over finite fields; Latin square; location-domination

1 Introduction

Sensor networks are systems consisting of sensors and links between them. As a monitoring tool they may be used for example in surveillance or to oversee arrays of processors. The basic idea is that a sensor is placed at some node of a network and then it monitors its

---

[*] A shortened version [15] of the paper has been presented at the 10th International Colloquium on Graph Theory and combinatorics, Lyon, 2018.

[†] Research supported by the University of Turku Graduate School (UTUGS) and the Vilho, Yrjö and Kalle Väisälä Foundation.

surroundings reporting on possible anomalies or irregularities. Based on these reports the central unit will deduce the location of the anomaly. The goal is to minimize the number of sensors in networks with certain structures. More on location in sensor networks can be found in [7, 18, 23].

A simple, undirected and connected graph $G = (V, E)$ is utilized to model the sensor network. The set of vertices adjacent to a vertex v is called the *open neighbourhood* of v, denoted by $N(v)$, and the set $N(v) \cup \{v\} = N[v]$ is called the *closed neighbourhood* of v. The vertices represent the possible locations of the sensors and the edge set determines the surrounding area of a sensor. In other words, a sensor placed on vertex u monitors locations $N[u]$ for irregularities.

A nonempty subset C of a vertex set V of a graph is called a *code* and its elements are called *codewords.* We define the *identifying set* or *I-set* of a vertex u as

$$I(u) = N[u] \cap C$$

or if the code or graph is unclear we may use the notation $I(C; u)$ or $I(G, C; u)$. The I-set can also be defined for a *set* of vertices. That is, for $U \subseteq V$, we define

$$I(U) = \bigcup_{v \in U} I(v).$$

In this paper, the code can be understood as the locations of the sensors within our sensor network and the I-set of u as the set of sensors which oversee the location u. The set of vertices C is called a *dominating set* if $I(v) \neq \varnothing$ for each vertex $v \in V$ and the minimum size of a dominating set in a graph G is called the *domination number* $\gamma(G)$. Hence, if sensors are placed at vertices which form a dominating set, then each location is monitored by a sensor and an irregularity is always detected. However, if the sensors report only that there is an irregularity within the area they monitor, then we need stronger condition than just a dominating set for locating the irregularity. For this purpose Karpovsky, Chakrabarty and Levitin defined *identifying codes* in [17]. More on identifying codes can be found at [20] and for recent development, see [1],[8] and[12].

Definition 1. A code $C \subseteq V$ is *identifying* in a graph G if C is a dominating set and

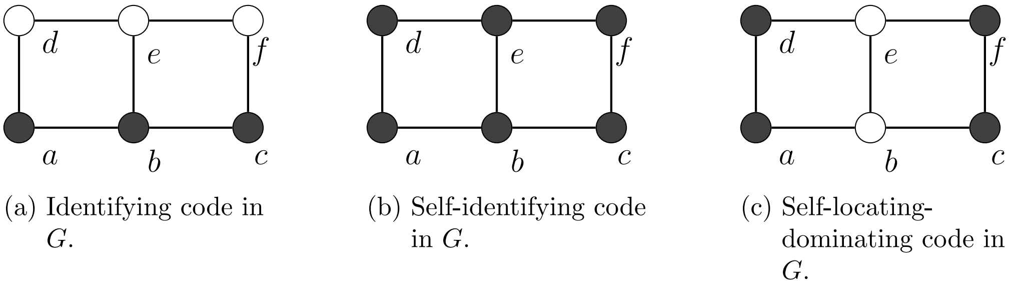
$$I(u) \neq I(v)$$

for each pair of distinct vertices $u, v \in V$. An identifying code C of minimum cardinality in a finite graph G is called *optimal* and its cardinality is denote with $\gamma^{ID}(G)$.

The previous definition of identifying code is illustrated in the following example.

Example 2. Let us consider graph G of Figure 1a and the code $C = \{a, b, c\}$. We have $I(d) = \{a\}$, $I(e) = \{b\}$, $I(f) = \{c\}$, $I(a) = \{a, b\}$, $I(b) = \{a, b, c\}$ and $I(c) = \{b, c\}$. Hence, each I-set is non-empty and unique and, therefore, the code C is an identifying code. Moreover, there are no smaller identifying codes in G since using at most two codewords we can form at most three different nonempty subsets of the code. Hence, C is an optimal identifying code in G and $\gamma^{ID}(G) = 3$.

Notice that there are some possible problems with identifying codes if more than one irregularity may occur in the sensor network. For instance, in the previous example, we have $I(b) = I(\{d, e, f\})$. Hence, if there were irregularities in the vertices d, e and f, then we would mistakenly deduce that an irregularity is in the vertex b. Moreover, we would not even notice that something went wrong. To overcome this problem, in [11], so called self-identifying codes, which are able to locate one irregularity and detect multiple ones, have been introduced. (Notice that in the original paper self-identifying codes are called 1^+-identifying.) The formal definition of self-identifying codes is given as follows.



(a) Identifying code in G.

(b) Self-identifying code in G.

(c) Self-locating-dominating code in G.

Figure 1: Optimal identifying, self-identifying and self-locating-dominating codes in G.

Definition 3. A code $C \subseteq V$ is called *self-identifying* in G if the code C is identifying in G and for all $u \in V$ and $U \subseteq V$ such that $|U| \geqslant 2$ we have

$$I(C; u) \neq I(C; U).$$

A self-identifying code C in a finite graph G with the smallest cardinality is called *optimal* and the number of codewords in an optimal self-identifying code is denoted by $\gamma^{SID}(G)$.

In addition to [11], self-identifying codes have also been previously discussed in [13, 14]. In these papers, two useful characterizations have been presented for self-identifying codes. These characterizations are presented in the following theorem.

Theorem 4 ([11, 13, 14])**.** *Let C be a code in G. Then the following statements are equivalent:*

(i) The code C is self-identifying in G.

(ii) For all distinct $u, v \in V$, we have $I(C; u) \setminus I(C; v) \neq \varnothing$.

(iii) For all $u \in V$, we have $I(C; u) \neq \varnothing$ and

$$\bigcap_{c \in I(C;u)} N[c] = \{u\}.$$

The previous definition of self-identifying codes is illustrated in the following example.

Example 5. Let G be the graph in Figure 1b and let C be a self-identifying code in G. If we now have $|I(a)| = 1$, then $\bigcap_{c \in I(a)} N[c] \neq \{a\}$ contradicting the fact that C is self-identifying due to Theorem 4(iii). Furthermore, if we have $I(a) = \{b, d\}$, then $I(a) \subseteq I(e)$ which contradicts with Theorem 4(ii). Finally, if $I(a) = \{a, b\}$ or $I(a) = \{a, d\}$, then respectively $I(a) \subseteq I(b)$ or $I(a) \subseteq I(d)$ (a contradiction). Hence, we must have $I(a) = \{a, b, d\}$ if C is self-identifying. Analogously, we get $I(f) = \{c, e, f\}$. Therefore, $C = V$ and, indeed, V is a self-identifying code in G.

For the situations when the sensor can distinguish whether the anomaly is in the open neighbourhood of the sensor or in the location of the sensor itself, we have *locating-dominating* codes which were introduced by Slater in [21, 24, 25] (for recent developments, see [2] and [20]). More precisely, a code $C \subseteq V$ is locating-dominating in G if the identifying sets $I(C; u)$ are nonempty and unique for all $u \in V \setminus C$. Inspired by self-identifying codes, we may analogously define so called self-locating-dominating codes, which have been introduced and motivated in [16].

Definition 6. A code $C \subseteq V$ is *self-locating-dominating* in G if for each vertex $u \in V \setminus C$ we have $I(u) \neq \varnothing$ and

$$\bigcap_{c \in I(u)} N[c] = \{u\}.$$

A self-locating-dominating code C in a finite graph G with the smallest cardinality is called *optimal* and the number of codewords in an optimal self-locating-dominating code is denoted by $\gamma^{SLD}(G)$.

In the following theorem, we show that self-locating-dominating codes have a characterization analogous to the one of self-identifying codes. By comparing Definition 6 and Theorem 7 to Theorem 4, we can see that they are almost the same except that only non-codewords are considered in the context of self-location-domination.

Theorem 7 ([16])**.** *A code $C \subseteq V$ is self-locating-dominating in G if and only if for each vertex $u \in V \setminus C$ and $v \in V$ $(u \neq v)$ we have*

$$I(u) \setminus I(v) \neq \varnothing.$$

The previous definition of self-locating-dominating codes is illustrated in the following example.

Example 8. Let G be the graph in Figure 1c and let C be a self-locating-dominating code in G. Necessarily, the vertex a belongs to C since otherwise

$$I(a) \setminus I(e) = \varnothing.$$

Similar reasoning also applies to the vertices c, d and f. Hence, we have $\{a, c, d, f\} \subseteq C$. On the other hand, we have $N[a] \cap N[c] = \{b\}$ and $N[d] \cap N[f] = \{e\}$. Therefore, by the definition, $C = \{a, c, d, f\}$ is an optimal self-locating-dominating code in G.

A graph is called a *complete graph* on q vertices, denoted by K_q, if each pair of vertices of the graph is adjacent. The *Cartesian product* of two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ is defined as $G_1 \square G_2 = (V_1 \times V_2, E)$, where E is a set of edges such that $(u_1, u_2)(v_1, v_2) \in E$ if and only if $u_1 = v_1$ and $u_2v_2 \in E_2$, or $u_2 = v_2$ and $u_1v_1 \in E_1$. The Cartesian product $K_q \square K_q \square \cdots \square K_q$ of n copies of K_q is denoted by K_q^n.

Identifying codes have been extensively studied, for example, in the binary hypercubes K_2^n (see the many articles listed in [20]), and [10, 22] for other Cartesian products. In 2008, Gravier, Moncel and Semri [5] investigated identification in K_q^2. Goddard and Wash [4] studied identification in the more general case of K_q^n and they, in particular, gave a conjecture for the cardinality of an optimal identifying code in $K_q^3 = K_q \square K_q \square K_q$.

Conjecture 9 ([4])**.** For all $q \geqslant 1$, $\gamma^{ID}(K_q \square K_q \square K_q) = q^2$.

In [4], Goddard and Wash prove that $\gamma^{ID}(K_q^3) \leqslant q^2$ for all $q \geqslant 1$. Moreover, by an exhaustive computer search, they show that $\gamma^{ID}(K_q^3) = q^2$ when $q = 3$. Furthermore, they provide a lower bound stating that $\gamma^{ID}(K_q^3) \geqslant q^2 - q\sqrt{q}$. Recall that the domination number $\gamma(K_q^3) = \left\lceil \frac{q^2}{2} \right\rceil$ (see [3]).

In this paper, we first show a one-to-one correspondence between Latin squares and optimal self-locating-dominating codes in K_q^3 in Section 2. Then based on this observation we see that the bound q^2 in Conjecture 9 holds for *self-locating-dominating codes*. In Section 3, we show for identifying codes that $\gamma^{ID}(K_q^3) \leqslant q^2 - q/4$ when q is a power of four. The approach is based on the recursive use of suitable Latin squares. This result disproves the Conjecture 9. We also give constructions of identifying codes for values of q other than the powers of four. After that we improve the lower bound of identifying codes in K_q^3 from $q^2 - q\sqrt{q}$ to $q^2 - \frac{3}{2}q$. Finally, in Section 5, we consider identifying codes in K_q^n, $n > 3$. In [4], it has been shown that $\gamma^{ID}(K_q^n) \leqslant q^{n-1}$. In Section 5, we significantly improve this upper bound when q is a prime power using suitable linear codes over finite fields as well as self-identifying codes and self-locating-dominating codes.

2 Self-location-domination in K_q^3

In this section, we examine self-locating-dominating codes in K_q^3. The vertices of the graph are denoted by (x, y, z), where $1 \leqslant x, y, z \leqslant q$, i.e., the vertex set $V = \{(x, y, z) \mid (x, y, z) \in \mathbb{Z}^3, 1 \leqslant x, y, z \leqslant q\}$. Hence, K_q^3 can be viewed as a cube in $\mathbb{Z}^3$ consisting of coordinates (x, y, z). A *pipe* is defined as a set of vertices which fixes two of the three coordinates. For example, the set $\{(1, 2, z) \mid 1 \leqslant z \leqslant q\}$ is a pipe in K_q^3. A pipe that fixes y- and z-coordinates is called a *row*, a pipe fixing x- and z-coordinates is called a *column* and a pipe fixing x- and y-coordinates is called a *tower*. Two vertices are neighbours in K_q^3 if and only if they belong to the same pipe. Hence, we have $|N(v)| = 3q - 2$.

We can represent a code in $K_q \square K_m \square K_l$ by taking a two dimensional $q \times m$ grid and placing the z-coordinates of the codewords to the positions with their x- and y-coordinates. The tower $(1, 1, z)$ is considered to be at the top left corner. Some codes and their representations are illustrated in Example 17. Moreover, as we will see in Theorem

12, this representation of K_q^3 has connection to $q \times q$ *Latin square*, which is a $q \times q$ array filled with numbers from 1 to q in such a way that each number occurs exactly once in each row or column.

Lemma 10. *Let C be a code in K_q^3 and v be a vertex of K_q^3.*

(i) If a vertex v has two codewords in its I-set and they do not locate within a single pipe, then there is exactly one other vertex which has those two codewords in its I-set.

(ii) The I-set $I(v)$ is not a subset of any other I-set if and only if there are at least three codewords in $I(v)$ and they do not locate within a single pipe.

Proof. Let C be a code in K_q^3. (i) Let us have $I(u) = \{c_1, c_2\}$, where $u = (x, y, z), c_1 = (x_1, y_1, z_1)$ and $c_2 = (x_2, y_2, z_2)$. Since c_1 and c_2 do not belong to the same pipe, we can without loss of generality assume that $x_1 \neq x_2$, $y_1 \neq y_2$, $x = x_1$, $y = y_2$ and $z = z_1 = z_2$. Now we have $N[c_1] \cap N[c_2] = \{u, (x_2, y_1, z)\}$.

(ii) Let us first show that if we have less than three codewords in $I(u)$ or the codewords locate within a single pipe, then $I(u)$ is a subset of another I-set. Let us have $I(u) = \{c_1\}$ and $v \in N(c_1)$. Then $I(u) \subseteq I(v)$. If we have $|I(u)| = 2$ and the codewords do not locate within the same pipe, then the case is the same as in (i). If we have $I(u) = \{c_1, \ldots, c_n\}$ and $I(u) \subseteq P$ for some pipe P, then $I(u) \subseteq P \subseteq I(c_1)$. Let us then assume that $\{c_1, c_2, c_3\} \in I(u)$, c_1, c_2 do not belong to the same pipe and $N[c_1] \cap N[c_2] = \{w, u\}$. Hence, u and w do not belong to the same pipe and by (i) we have $N[w] \cap N[u] = \{c_1, c_2\}$. Therefore, $c_3 \notin I(w)$ and $I(u)$ is not a subset of any other I-set. ☐

In the following two theorems we show that the bound q^2 in Conjecture 9 is true for *self-locating-dominating codes.*

Theorem 11. *We have*

$$\gamma^{SLD}(K_q^3) \geqslant q^2.$$

Proof. Let C be a self-locating-dominating code in K_q^3. By Lemma 10, each non-codeword has to have at least three codewords in its I-set and each codeword has at least one codeword in its I-set. Hence, by double counting pairs (c, x) where $c \in C$ and $x \in N[c]$, we get

$$(3q - 2)|C| \geqslant 3(q^3 - |C|) + |C| \iff |C| \geqslant q^2. \qquad \square$$

In the following theorem, we show with the aid of Lemma 10 that each optimal self-locating-dominating code in K_q^3 can be represented as a Latin square (and vice versa).

Theorem 12. *There is a one-to-one correspondence between optimal self-locating-dominating codes in K_q^3 and $q \times q$ Latin squares.*

Proof. Let L be a $q \times q$ Latin square. Consider the Latin square L as a code C in K_q^3 as in if there is the value z in array slot (x, y), then $(x, y, z) \in C$. It is immediate that $|C| = q^2$ as L is a Latin square. Observe that now each non-codeword is covered by exactly three codewords not belonging to a same pipe since there is exactly one codeword in each tower, column and row intersecting with the non-codeword. Hence, by Lemma 10(ii), C is a self-locating-dominating code and we have $\gamma^{SLD}(K_q^3) = q^2$ due to Theorem 11.

Let C be an optimal self-locating-dominating code of cardinality q^2 in K_q^3. By Lemma 10, each non-codeword is covered by at least three codewords and each codeword is covered by at least one codeword. Hence, by double counting pairs (x, c) where $c \in C$ and $c \in N[x]$ we get the inequality

$$3q^3 - 2q^2 = (3q-2)|C| \geqslant |C| + 3|V(K_q^3) \setminus C| = 3q^3 - 2q^2$$

and since both sides are equal, each codeword is covered by exactly one codeword (itself) and each non-codeword is covered by exactly three codewords. If there is a tower without codewords, then some tower has two codewords and there is a codeword which is covered by two codewords (a contradiction). So, there is exactly one codeword in each tower. Similarly, we may also show that there is exactly one codeword in each row and column. If we now represent this code using a $q \times q$ grid, we get a Latin square, since there is a number from 1 to q in each box of the Latin square and the same number never occurs twice in the same row or column. □

Corollary 13. *We have*

$$\gamma^{SLD}(K_q^3) = q^2.$$

3 Constructions for identification in K_q^3

In this section, we consider identification in K_q^3 and present a bound $\gamma^{ID}(K_q^3) \leqslant q^2 - q/4$ when q is a power of four, giving an infinite family of counterexamples to Conjecture 9. First we give a construction for an identifying code in K_4^3 with cardinality 15 and then we use that identifying code and suitable Latin squares to recursively construct the infinite family.

Goddard and Wash [4] have shown the following results.

Theorem 14 ([4])**.** *For $m > 2l$ and $l > 2q$, we have*

$$\gamma^{ID}(K_q \square K_l \square K_m) = q(m-1).$$

Moreover, for the complete graphs of equal order, we have

$$q^2 - q\sqrt{q} \leqslant \gamma^{ID}(K_q^3) \leqslant q^2.$$

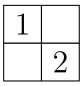
Due to the recursive nature of our construction we first define an operation which combines two codes in K_q^3 and K_m^3 into a code in K_{qm}^3.

Definition 15. Let $C_1 \subseteq \{(x, y, z) \mid 1 \leqslant x, y, z \leqslant q\}$ and $C_2 \subseteq \{(x, y, z) \mid 1 \leqslant x, y, z \leqslant m\}$ be codes in K_q^3 and K_m^3 respectively. Define $\mathrm{Ext}(C_1, C_2) = \{(x, y, z, a, b, c) \mid (x, y, z) \in C_1, (a, b, c) \in C_2\}$, which is the Cartesian product of the sets C_1 and C_2.

The sextuple produced by $\mathrm{Ext}(C_1, C_2)$ (and also other sextuples) can be interpreted in the following way.

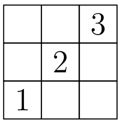
Observation 16. *We can interpret each vertex* $(v, u, w) \in K^3_{qm} = \{(x', y', z') \mid 1 \leqslant x', y', z' \leqslant qm\}$ *as* (x, y, z, a, b, c) *where* $1 \leqslant x, y, z \leqslant q$ *and* $1 \leqslant a, b, c \leqslant m$ *by having* $v = x + q(a-1)$, $u = y + q(b-1)$ *and* $w = z + q(c-1)$. *In other words,* $\mathrm{Ext}(C_1, C_2)$ *can be interpreted as a code in* K^3_{qm}. *Furthermore, since each pipe fixes two out of three coordinates in a triple, a pipe fixes four out of six coordinates in a sextuple.*

Example 17. Let $C_1 = \{(1,1,1), (2,2,2)\}$ and $C_2 = \{(1,3,1), (2,2,2), (3,1,3)\}$ be codes in K^3_2 and K^3_3, respectively (see Figures 2 and 3). Recall that the vertex $(1,1,z)$ is represented by the top left box and the z coordinate corresponds to the number in that box. Then we have $\mathrm{Ext}(C_1, C_2) = \{(1,1,1,1,3,1),\ (1,1,1,2,2,2),\ (1,1,1,3,1,3),\ (2,2,2,1,3,1),\ (2,2,2,2,2,2),\ (2,2,2,3,1,3)\}$. Furthermore, we can consider this as a code in K^3_6 as seen in Figure 4.



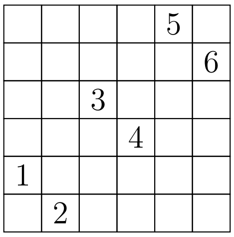
1	
	2

Figure 2: Code C_1 in K^3_2.



		3
	2	
1		

Figure 3: Code C_2 in K^3_3.



				5	
					6
		3			
			4		
1					
	2				

Figure 4: Code $\mathrm{Ext}(C_1, C_2)$ in K^3_6.

Goddard and Wash [4] give the following construction for identifying codes of cardinality q^2 in the graph K^3_q.

Lemma 18. *The code* $C_q = \{(a, b, c) \mid a + b + c \equiv 0 \pmod{q}\}$ *is identifying in* K^3_q *with the additional property that each pipe in* K^3_q *has exactly one codeword.*

Proof. The code C_q is shown to be identifying in [4]. Furthermore, there is exactly one codeword in each pipe since if we fix two of the three coordinates, then for exactly one value of the third coordinate the equation $a + b + c \equiv 0 \pmod{q}$ is satisfied. □

By presenting the previous construction in a grid, we can consider it as a Latin square and hence, we can get the properties mentioned in the previous lemma also in that way. The identifying code in K^3_4 of the following theorem is of cardinality 15. The code is presented in Figure 5.

Theorem 19. *The code* $C^1 = \{(2,1,3),\ (2,1,4),\ (3,1,1),\ (4,1,2),\ (1,2,2),\ (1,2,4),\ (2,2,4),\ (3,2,2),\ (1,3,1),\ (2,3,2),\ (3,3,3),\ (4,3,3),\ (2,4,4),\ (4,4,1),\ (4,4,3)\}$ *is identifying in* K^3_4.

	3, 4	1	2
2, 4	4	2	
1	2	3	3
	4		1, 3

Figure 5: Identifying code C^1 in K_4^3.

	3	4	2
4		1	3
2	4		1
3	1	2	

Figure 6: Identifying code C_L in $K_4^3[V(K_4^3) \setminus Di]$.

1	3	4	2
4	**2**	1	3
2	4	**3**	1
3	1	2	**4**

Figure 7: Code C^t where bolded numbers represent cubes with the code C^{t-1} and other numbers represent cubes with the code $C_{4^{t-1}}$.

Proof. By examining Table 1 in the appendix, we notice that each I-set is nonempty and unique. □

In what follows, we call the set $Di = \{(j, j, j) \mid 1 \leqslant j \leqslant 4\}$ as the *diagonal.* We also need another code, C_L, to produce the infinite family of codes of the desired cardinality. The code is presented in Figure 6 for the graph $K_4^3[V(K_4^3) \setminus Di]$, that is, for the graph K_4^3 with its diagonal vertices Di deleted. Note that the empty squares of the array of Figure 6 is easy to fill in such a way that a Latin square is obtained.

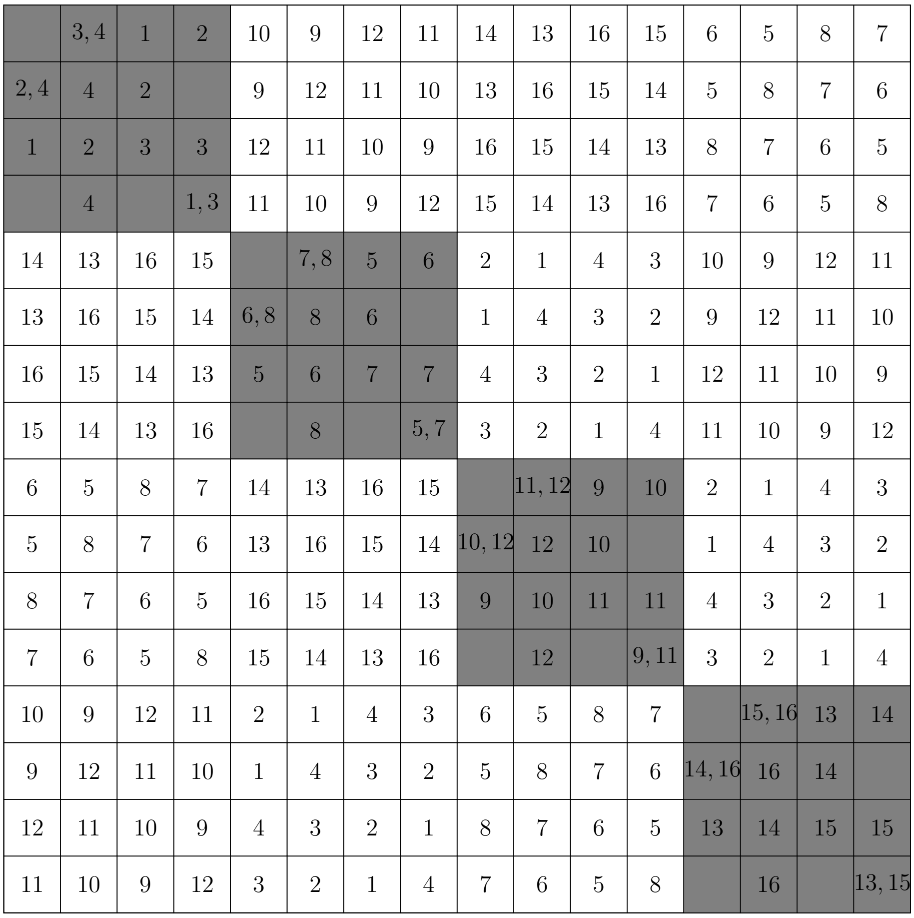
Lemma 20. *The code* $C_L = \{(2, 1, 3), (3, 1, 4), (4, 1, 2), (1, 2, 4), (3, 2, 1), (4, 2, 3), (1, 3, 2), (2, 3, 4), (4, 3, 1), (1, 4, 3), (2, 4, 1), (3, 4, 2)\}$ *is identifying in* $K_4^3[V(K_4^3) \setminus Di]$ *and for each codeword* $c \in C_L$ *we have* $I(c) = \{c\}$.

Proof. By examining Table 2 in the appendix, we notice that each I-set is nonempty and unique. Furthermore, by checking the highlighted vertices we notice that we have $I(c) = \{c\}$ for each codeword $c \in C_L$. □

With the help of the codes C_q, C_L, C^1 and Di, we can construct a family of identifying codes of cardinality $q^2 - \frac{q}{4}$ in K_q^3 for $q = 4^t$, $t \in \mathbb{Z}$ and $t > 0$ as described in the following theorem.

Theorem 21. *The code* $C^t = \mathrm{Ext}(C_{q/4}, C_L) \cup \mathrm{Ext}(C^{t-1}, Di)$ *is identifying in* K_q^3, *where* $q = 4^t$ *and* $t \geqslant 2$, *of cardinality* $q^2 - \frac{q}{4}$.

Proof. Let C^t be a code recursively defined as $C^t = \mathrm{Ext}(C_{q/4}, C_L) \cup \mathrm{Ext}(C^{t-1}, Di)$. In other words, the code C^t can be intuitively interpreted as follows. The cube K_q^3 can be considered as K_4^3 where each vertex is replaced with a *subcube* $K_{q/4}^3$. More precisely, the last three digits of the sextuple notation give the vertex which has been replaced with a $K_{q/4}$ subcube and the first three coordinates give the location within the subcube. Furthermore, the code C^t can be considered as a union of codes $C_{q/4}$ placed into the



	3, 4	1	2	10	9	12	11	14	13	16	15	6	5	8	7
2, 4	4	2		9	12	11	10	13	16	15	14	5	8	7	6
1	2	3	3	12	11	10	9	16	15	14	13	8	7	6	5
	4		1, 3	11	10	9	12	15	14	13	16	7	6	5	8
14	13	16	15		7, 8	5	6	2	1	4	3	10	9	12	11
13	16	15	14	6, 8	8	6		1	4	3	2	9	12	11	10
16	15	14	13	5	6	7	7	4	3	2	1	12	11	10	9
15	14	13	16		8		5, 7	3	2	1	4	11	10	9	12
6	5	8	7	14	13	16	15		11, 12	9	10	2	1	4	3
5	8	7	6	13	16	15	14	10, 12	12	10		1	4	3	2
8	7	6	5	16	15	14	13	9	10	11	11	4	3	2	1
7	6	5	8	15	14	13	16		12		9, 11	3	2	1	4
10	9	12	11	2	1	4	3	6	5	8	7		15, 16	13	14
9	12	11	10	1	4	3	2	5	8	7	6	14, 16	16	14	
12	11	10	9	4	3	2	1	8	7	6	5	13	14	15	15
11	10	9	12	3	2	1	4	7	6	5	8		16		13, 15

Figure 8: An identifying code of size $16^2 - 4$ in K_{16}^3.

subcubes given by the code C_L and codes C^{t-1} placed into the subcubes given by the code Di (see Figure 7). Furthermore, the code C^2 is illustrated in Figure 8.

Since codes C_L and Di are separate in the graph K_4^3 and $|C^1| = 4^2 - \frac{4}{4}$, we can use induction on the cardinality $|C^t|$ and thus, have

$$|C^t| = |C_L| \cdot |C_{q/4}| + |Di| \cdot |C^{t-1}| = 12 \cdot \left(\frac{q}{4}\right)^2 + 4|C^{t-1}| = q^2 - \frac{q}{4}.$$

The basic idea of the code C^t is that codes C_L and Di identify the three latter coordinates of the sextuple (x, y, z, a, b, c) and the codes $C_{q/4}$ and C^{t-1} identify the first three coordinates. When $t = 2$, C^1 is an identifying code in K_4^3 (by Theorem 19). Let us now make an induction hypothesis that C^{t-1} is an identifying code in $K_{q/4}^3$ for $t \geqslant 2$.

Let us first show that if for $v = (x, y, z, a, b, c)$ and $w = (x', y', z', a', b', c')$ we have $(a, b, c) \neq (a', b', c')$, then $I(C^t; v) \neq I(C^t; w)$. Let us assume first that $(a, b, c), (a', b', c') \notin Di$. First notice that by Lemma 18 each pipe which goes through a subcube with the code $C_{q/4}$ intersects with exactly one codeword. Hence, if $c = (x'', y'', z'', a'', b'', c'') \in I(C^t; v)$ and $c \notin \operatorname{Ext}(C^{t-1}, Di)$, then $(a'', b'', c'') \in C_L$. Thus, if $I(C^t; v) = I(C^t; w)$, then $I(C_L; (a, b, c)) = I(C_L; (a', b', c'))$ which is not possible since C_L is an identifying code.

Moreover, if $(a,b,c) \in Di$, then $I(C^t; v) \subseteq \mathrm{Ext}(C^{t-1}, \{(a,b,c)\})$ since $N[Di] \cap C_L = \varnothing$ by Table 18 and $K_4^3[Di]$ is a discrete graph.

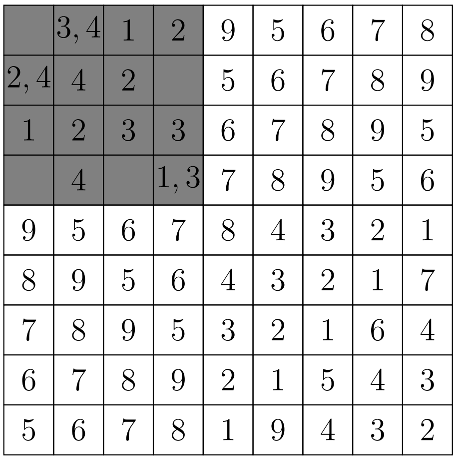
We now divide the proof into cases based on the location of $v = (x,y,z,a,b,c)$. Let us consider the case where v is such that $(a,b,c) \in C_L$. Moreover, let $w = (x', y', z', a', b', c')$ and let us assume that $I(C^t; v) = I(C^t; w)$. By the previous deduction, we have $(a,b,c) = (a', b', c')$. Moreover, $I(C_{q/4}; (x,y,z)) = I(C_{q/4}; (x', y', z'))$ since $I(C^t; v) = I(C^t; w)$ and $(a,b,c) = (a', b', c')$. Since $C_{q/4}$ is an identifying code, we have $(x,y,z) = (x', y', z')$ and hence, $v = w$.

Let us then consider the case where $(a,b,c) \notin C_L \cup Di$. We have $|I(\mathrm{Ext}(C_{q/4}, C_L); v)| \geqslant 2$ since only codewords have one codeword in their I-sets in C_L. If $|I(\mathrm{Ext}(C_{q/4}, C_L); v)| \geqslant 3$, then everything is clear due to Lemma 10(ii) since in the codes $C_{q/4}$ and C_L there are no pipes with multiple codewords. Hence, we may assume that $|I(\mathrm{Ext}(C_{q/4}, C_L); v)| = 2$. Recall that no pipe in K_q^3 has two codewords which are in $\mathrm{Ext}(C_{q/4}, C_L)$ and by Lemma 10(i) there is exactly one vertex w such that $I(\mathrm{Ext}(C_{q/4}, C_L); v) \subsetneq I(\mathrm{Ext}(C_{q/4}, C_L); w)$. We may assume that the codewords in $I(\mathrm{Ext}(C_{q/4}, C_L); v)$ locate in the subcubes which are placed at coordinates (a', b, c) and (a, b', c). However, now w locates in the subcube at coordinates (a', b', c) which is not possible.

Finally, we have the case $(a,b,c) \in Di$. We immediately notice that if $c \in I(C^t; v)$, then c is of the form $(x', y', z', a, b, c) \in \mathrm{Ext}(C^{t-1}, Di)$. Furthermore, the code C^{t-1} is an identifying code in $K_{q/4}^3$ and hence, the vertices within the same diagonal subcube have different I-sets than v and thus, the I-set of v is unique. □

So far, we have given constructions for identifying codes in K_q^3 with $q = 4^t$. However, we can further use these codes to construct new identifying codes for other values of q. For this purpose, we use Latin squares and Evans' Theorem.

Theorem 22 (Evans' Theorem [6]). *Any $q \times q$ Latin square can be extended into an $r \times r$ Latin square if $r \geqslant 2q$.*



	3, 4	1	2	9	5	6	7	8
2, 4	4	2		5	6	7	8	9
1	2	3	3	6	7	8	9	5
	4		1, 3	7	8	9	5	6
9	5	6	7	8	4	3	2	1
8	9	5	6	4	3	2	1	7
7	8	9	5	3	2	1	6	4
6	7	8	9	2	1	5	4	3
5	6	7	8	1	9	4	3	2

Figure 9: Identifying code of size 80 in K_9^3.

Theorem 23. *Let C be an identifying code in K_q^3 of cardinality m. If $r \geqslant 2q$, then we have an identifying code in K_r^3 of cardinality $r^2 - q^2 + m$.*

Proof. Let C be an identifying code in K_q^3 of cardinality m and $r \geqslant 2q$. Let us consider a $q \times q$ Latin square L' with values from 1 to q. According to Theorem 22, we can extend the Latin square L' into an $r \times r$ Latin square L. Let us assume that the Latin square L' locates in the coordinates (x, y, z), where $x, y, z \leqslant q$. Moreover, we use notation $(x, y, z) \in L$ if there is value z at the location (x, y) in the Latin square. Let us have

$$C' = C \cup \{(x, y, z) \mid \max\{x, y\} \geqslant q + 1 \text{ and } (x, y, z) \in L\}.$$

The code C' is illustrated in Figure 9 when $q = 4$, $r = 9$ and the original code $C = C^1$. We have $|C'| = r^2 + m - q^2$. Moreover, we have the following two observations on the structure of the code.

Observation 1: Each pipe P with at least one of the two fixed coordinates greater than q, has exactly one codeword in it. Indeed, since P is a pipe with at least one fixed coordinate greater than q, it does not intersect with L' and hence, it does intersect with $L \setminus L'$. Note that since L is a Latin square, each pipe intersects with exactly one vertex in L.

Observation 2: Vertex (x, y, z) does not belong to C' if exactly one of the three coordinates is greater than q. Indeed, if a vertex (x, y, z) with exactly one coordinate greater than q is a codeword, then the pipe which intersects with (x, y, z) and L' (there is such a pipe) contradicts against the structure of the Latin square L.

Let us show that C' is an identifying code by dividing the proof into cases based on the location of the vertex $v = (x, y, z) \in V(K_q^3)$ and whether v is a codeword or not. Let us first consider the case where at least two of the coordinates (x, y, z) of the vertex v are greater than q and v is a non-codeword. Hence, there is exactly one codeword in each pipe intersecting with v by Observation 1, $|I(v)| = 3$ and the codewords in $I(v)$ do not locate within a single pipe. Therefore, v is now uniquely identified by Lemma 10.

Let us then consider the case where exactly one of the coordinates (x, y, z) is greater than q. Now v is a non-codeword by Observation 2. We have $|I(v)| \geqslant 2$ since two of the pipes going through v fix the coordinate which is greater than q. Hence, if $|I(v)| > 2$, then the I-set is unique. On the other hand, if $|I(v)| = 2$, then there is another vertex w which has those two codewords in its I-set. Now, if $I(v) = \{c, c'\}$, then exactly one coordinate of c is less than $q + 1$ and the same is true for c' due to Observation 2 and since the codewords in $I(v)$ locate in the pipes with a fixed coordinate greater than q. Thus, it is straightforward to verify that each coordinate of w is greater than q. Hence, $|I(w)| \geqslant 3$ and $I(v)$ is unique.

Let us then consider the case where at least two of the coordinates (x, y, z) are greater than q and v is a codeword. We have $I(v) = \{v\}$ and each neighbour of v has at least two codewords in its I-set as we have seen in the two previous cases.

Finally, we have the case $x, y, z \leqslant q$. Now the vertex v is identified by the code C since C is an identifying code, each vertex with a coordinate greater than q has codewords in its I-set which do not belong to C by the previous considerations and $I(v) \subseteq C$ by Observation 2. $\square$

By considering the identifying code C^t and Theorem 23, we get the following corollary which gives an identifying code in K_q^3 for all $q \geqslant 8$ of cardinality less than q^2.

Corollary 24. *If* $2 \cdot 4^t \leqslant q \leqslant 2 \cdot 4^{t+1} - 1$, *then we have*

$$\gamma^{ID}(K_q^3) \leqslant q^2 - 4^{t-1}.$$

4 Lower bound for identification in K_q^3

With our construction and the lower bound of Goddard and Wash, we now know that $q^2 - q\sqrt{q} \leqslant \gamma^{ID}(K_q^3) \leqslant q^2 - \frac{q}{4}$ when $q = 4^t$, $t \in \mathbb{Z}$, $t > 0$. In this section, we improve the lower bound to $\gamma^{ID}(K_q^3) \geqslant q^2 - \frac{3}{2}q$. The standard techniques for obtaining lower bounds for identifying codes in graphs are based on the covering properties of balls or symmetric differences (see [20]). For K_q^3 these methods are not powerful enough, so we provide a new approach, which builds on the method of Goddard and Wash [4].

Definition 25. Let C be a code in K_q^3 and let i be an integer such that $1 \leqslant i \leqslant q$. Define an *$x_i$-layer* of the graph K_q^3, denoted by D_i^1, as the set of vertices which fixes the coordinate $x = i$, i.e., $D_i^1 = \{(i, y, z) \mid 1 \leqslant y, z \leqslant q\}$. Analogously, we define a *y_i-layer* D_i^2 and a *z_i-layer* D_i^3. Let j be an integer such that $1 \leqslant j \leqslant 3$. Define then $X_i^j = \{v \in D_i^j \mid I(C; v) \cap D_i^j = \varnothing\}$ and $Y_i^j = \{v \in D_i^j \mid I(C; v) \cap D_i^j = \{v\}\}$. Furthermore, we use the following notation: $X^j = \bigcup_{i=1}^q X_i^j$, $Y^j = \bigcup_{i=1}^q Y_i^j$ and $X = \bigcup_{j=1}^3 X^j$, $Y = \bigcup_{j=1}^3 Y^j$ and $C_i^j = C \cap D_i^j$. A codeword which does not belong to Y is called a *corner*, and a *fellow* is a codeword belonging to Y such that it has another codeword in its open neighbourhood.

Lemma 26. *Let C be an identifying code in K_q^3. For a pipe P, the following statements hold:*

(i) The pipe P does not contain multiple fellows.

(ii) The pipe P does not contain a corner, a fellow and a vertex $v \in X$.

(iii) The pipe P does not contain a codeword and vertices $v, v' \in X$.

Proof. (i) Let c and c' be fellows in P. Therefore, as $c' \in I(c)$ and $c \in I(c')$, we have a contradiction; $I(c) = I(c')$. (ii) Let c be a corner, c' a fellow and v a vertex of X in P. Hence, since $c \in I(c')$ and $c \in I(v)$, we have $I(c') \subseteq P$ and $I(v) \subseteq P$. This implies that $I(c) = I(v)$. (iii) Let c be a codeword and $v, v' \in X$ in P. Similarly, we again have $I(v) \subseteq P$ and $I(v') \subseteq P$ and, thus, $I(v) = I(v')$. $\square$

We can show that for each vertex $x \in X$, there is a corner that is linked to the vertex x. Later, in the proof of Lemma 29, we show that each corner is linked to at most three vertices of X.

Lemma 27. *Let C be an identifying code in K_q^3. If $x \in X$, then there exists a codeword $c \in I(x)$ such that c is a corner or a fellow with a corner in $I(c)$.*

Proof. Let C be an identifying code in K_q^3 and let $x \in X$. Since C is an identifying code, we have $I(x) \neq \varnothing$. Let us say that $c \in I(x)$. We can assume that c is not a corner since otherwise we are immediately done. Furthermore, if $I(c) = \{c\}$, then $I(c) = I(x)$. We can now assume that there exists another codeword $c' \in I(c)$ ($c' \neq c$). Now since $c \in Y$ and $|I(c)| \geqslant 2$, c is a fellow. Moreover, by Lemma 26(i), c' is not a fellow and therefore it is not in Y. Thus, c' is a corner. ☐

Definition 28. Let C be an identifying code in K_q^3 and $c \in C$ be a corner. If there are two codewords $c', c'' \in I(c) \cap D_i^j$ which do not belong to the same pipe, then we say that c is a corner of the layer D_i^j. Furthermore, we denote by k_i^j the total number of corners of the layer D_i^j.

We have shown that there is a corner for each vertex in X. We will further show that each corner can be associated to at most three vertices of X.

Lemma 29. *Let C be an identifying code in K_q^3. Then we have*

$$|X| \leqslant 3 \sum_{j=1}^{3} \sum_{i=1}^{q} k_i^j.$$

Proof. Let $c \in C$ be a corner. By Lemma 26, we have in total at most three fellows and vertices of X in $N[c]$. Moreover, if c' is a fellow, then $|N(c') \cap X| \leqslant 1$ since if $v, v' \in N(c') \cap X$, then $I(v) = I(v')$. Hence, for each corner c there are at most three vertices in X such that they are in the neighbourhood of c or in the neighbourhood of a fellow $c' \in I(c)$.

With the aid of Lemma 27, we notice that for each vertex $x \in X$ there exists in $N(x)$ a corner or a fellow with a corner in its neighbourhood. Thus, we have $|X| \leqslant 3\,|\{c \mid c \text{ is a corner}\}|$. Moreover, each corner is counted in the sum $\sum_{j=1}^{3} \sum_{i=1}^{q} k_i^j$ and hence, we have $|X| \leqslant 3 \sum_{j=1}^{3} \sum_{i=1}^{q} k_i^j$. ☐

To approximate the cardinality of each X_i^j, we need the domination number of $K_q \square K_q$. Later, this result is used to approximate the number of vertices of X in a layer. The following lemma is Exercise 1.12 in [9].

Lemma 30 ([9])**.** *For each positive integer q, we have*

$$\gamma(K_q \square K_q) = q.$$

Definition 31. Let C be a code in K_q^3 and $M_i^j \subseteq D_i^j$ be a minimum dominating set of D_i^j such that $C_i^j \subseteq M_i^j$. Then we denote $f_i^j = |M_i^j| - q$. Note that q is the domination number of $K_q \square K_q$. Let us further denote $a_i^j = q - |C_i^j|$.

Note that we need $q + f_i^j$ codewords to dominate the layer D_i^j. Hence, value f_i^j can be understood as a measurement of how much the structure of the code within a layer increases the cardinality of X_i^j. Indeed, observe that if a non-codeword in the layer D_i^j is not dominated by C_i^j, then it belongs to X_i^j and there is a row and a column within the layer D_i^j without codewords. Moreover, observe that we have only $q - a_i^j$ codewords in the layer D_i^j. Hence, there are at least $q + f_i^j - (q - a_i^j) = f_i^j + a_i^j$ rows and columns which do not have a codeword when $f_i^j + a_i^j \geqslant 0$. Thus, we have $|X_i^j| \geqslant (a_i^j + f_i^j)^2$. The previous observations are illustrated in Figure 10. Furthermore, the number of corners in a layer is connected with f_i^j as explained in the following lemma.

Lemma 32. *Let C be an identifying code in K_q^3. Then $2f_i^j \geqslant k_i^j$ for each i and j.*

Proof. Let C be an identifying code in K_q^3, and consider a layer D_i^j for some $1 \leqslant i \leqslant q$ and $1 \leqslant j \leqslant 3$. Since the vertices of D_i^j can be viewed as a graph K_q^2, we can consider pipes locating within it as rows and columns. There are q rows and q columns in D_i^j and a subset of D_i^j is dominating if and only if it intersects with all rows or all columns. Indeed, if there are a row R and a column S which do not contain any codewords, then the vertex in their intersection is not dominated. Let us now assume that C_i^j intersects with n rows and h columns ($n \geqslant h$) and that M_i^j is a minimal dominating set of D_i^j such that $C_i^j \subseteq M_i^j$.

We have $|M_i^j| = |C_i^j| + (q - n)$ since C_i^j dominates n out of q rows and $n \geqslant h$. Thus, we have

$$f_i^j = |M_i^j| - q = |C_i^j| - n.$$

If we have one or two corners of the layer D_i^j in a row, then that row has at least two codewords in it and deleting a corner still preserves at least one codeword in that row. If a row has $m \geqslant 3$ corners in it, then it has at least m codewords and hence, deleting $m - 1$ corners still maintains a codeword in the row. Therefore, we may delete at least half of the corners in such a way that there still are codewords in n rows. Hence, we have $n \leqslant |C_i^j| - \frac{1}{2}k_i^j$ and thus, we have

$$f_i^j \geqslant \frac{1}{2}k_i^j. \qquad \square$$

From Lemmas 29 and 32 we get following corollary.

Corollary 33. *If C is an identifying code in K_q^3, then we have*

$$|X| \leqslant 6 \sum_{j=1}^{3} \sum_{i=1}^{q} f_i^j.$$

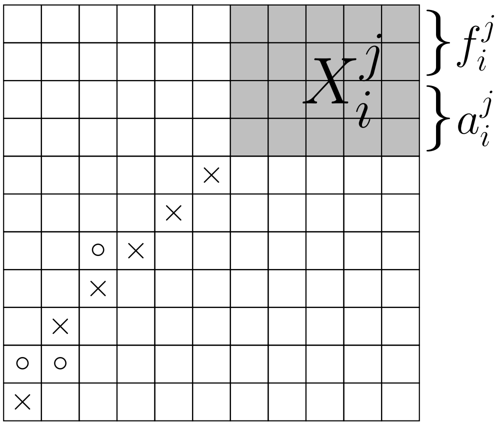


Figure 10: A code within a layer of K_{12}^3 with 3 corners marked by ∘, 7 other codewords marked by ×, $f = 2$ and $|X| = 20$.

Now we can prove the new lower bound for $\gamma^{ID}(K_q^3)$. The proof is based on the idea that each vertex in X requires corners, corners increase the values f_i^j and the values f_i^j increase the cardinality of X. Figure 10 shows how corners and the number of codewords affect the size of X.

Theorem 34. *We have*

$$\gamma^{ID}(K_q^3) \geqslant q^2 - \frac{3}{2}q.$$

Proof. Let C be an identifying code in K_q^3 of an optimal size $\gamma^{ID}(K_q^3)$. We have,

$$|C| = \frac{1}{3}\sum_{j=1}^{3}\sum_{i=1}^{q}|C_i^j| = \frac{1}{3}\sum_{j=1}^{3}\sum_{i=1}^{q}(q - a_i^j) = q^2 - \frac{1}{3}\sum_{j=1}^{3}\sum_{i=1}^{q}a_i^j. \tag{1}$$

Since there exists an identifying code with size q^2 by [4], we have $\frac{1}{3}\sum_{j=1}^{3}\sum_{i=1}^{q}a_i^j \geqslant 0$. Let M_i^j be a minimum dominating set in D_i^j such that $C_i^j \subseteq M_i^j$. Notice that $f_i^j \geqslant 0$ and a_i^j can be negative but $a_i^j \geqslant -f_i^j$ since $f_i^j = |M_i^j| - q \geqslant |C_i^j| - q = -a_i^j$. Hence, we have $f_i^j + a_i^j \geqslant 0$.

Now we can give an approximation for X:

$$|X| \geqslant \sum_{j=1}^{3}\sum_{i=1}^{q}(a_i^j + f_i^j)^2. \tag{2}$$

We can do this approximation since there are at least $|M_i^j| - |C_i^j|$ rows and columns without codewords in D_i^j. Hence, we have $|X_i^j| \geqslant \big(|M_i^j| - |C_i^j|\big)^2 = \big((q + f_i^j) - (q - a_i^j)\big)^2 = \big(f_i^j + a_i^j\big)^2$. Furthermore, by Corollary 33, we have

$$6\sum_{j=1}^{3}\sum_{i=1}^{q}f_i^j \geqslant |X| \geqslant \sum_{j=1}^{3}\sum_{i=1}^{q}(a_i^j + f_i^j)^2. \tag{3}$$

Now we can give a lower bound for $|C|$:

$$\begin{aligned}
|C| &\overset{(1)}{=} q^2 - \frac{1}{3}\sum_{j=1}^{3}\sum_{i=1}^{q} a_i^j \\
&= q^2 - \frac{1}{3}\left(\sum_{j=1}^{3}\sum_{i=1}^{q} a_i^j + \sum_{j=1}^{3}\sum_{i=1}^{q} f_i^j - \sum_{j=1}^{3}\sum_{i=1}^{q} f_i^j\right) \\
&= q^2 - \frac{1}{3}\left(\sum_{j=1}^{3}\sum_{i=1}^{q} \left(a_i^j + f_i^j\right) - \sum_{j=1}^{3}\sum_{i=1}^{q} f_i^j\right) \\
&\overset{(3)}{\geqslant} q^2 - \frac{1}{3}\left(\sum_{j=1}^{3}\sum_{i=1}^{q} \left(a_i^j + f_i^j\right) - \frac{\sum_{j=1}^{3}\sum_{i=1}^{q}\left(a_i^j + f_i^j\right)^2}{6}\right) \\
&\overset{(*)}{\geqslant} q^2 - \frac{1}{3}\left(3qA - \frac{3qA^2}{6}\right) \\
&= q^2 - q\left(A - \frac{A^2}{6}\right) \\
&\overset{(**)}{\geqslant} q^2 - \frac{3}{2}q.
\end{aligned}$$

We get the inequality $(*)$ by using Lagrange's method. We can minimize the value of sum $\sum_{j=1}^{3}\sum_{i=1}^{q}\left(a_i^j + f_i^j\right)^2$ while retaining the value of sum $\sum_{j=1}^{3}\sum_{i=1}^{q}\left(a_i^j + f_i^j\right)$. The minimum of sum $\sum_{j=1}^{3}\sum_{i=1}^{q}\left(a_i^j + f_i^j\right)^2$ equals to $3qA^2$ where $A = \frac{\sum_{j=1}^{3}\sum_{i=1}^{q}\left(a_i^j + f_i^j\right)}{3q}$ that is the average value of $a_i^j + f_i^j$. For inequality $(**)$ we notice that $A - \frac{A^2}{6}$ gains its maximum value at $A = 3$. □

5 Results in K_q^n when $n > 3$

In this section, we consider identifying, self-identifying and self-locating-dominating codes in K_q^n, when $n > 3$ and $q > 2$. Goddard and Wash [4] showed that $\gamma^{ID}(K_q^n) \leqslant q^{n-1}$. In what follows, we first give optimal self-identifying and self-locating-dominating codes in K_q^n for certain values of n and q. Then based on these codes we are able to significantly improve the bound $\gamma^{ID}(K_q^n) \leqslant q^{n-1}$ when q is a *prime power*.

For later use, we first begin by introducing some notation and preliminary results based on the classical book [19] of coding theory. For the rest of the section, we assume that q is a prime power. Then there exists a finite field with q elements, and we denote this field by $\mathbb{F}_q$. The set of all n-tuples of $\mathbb{F}_q$ forms a vector space $\mathbb{F}_q^n$. The vector space $\mathbb{F}_q^n$ can be considered as a graph by defining two vertices of $\mathbb{F}_q^n$ to be adjacent if they differ in exactly one coordinate. This graph is called the q-ary hypercube or the q-ary Hamming space. A vertex of a q-ary Hamming space is called a *word*. For two words u and v of $\mathbb{F}_q^n$, the *Hamming distance* is defined as the usual (geodesic) distance $d(u, v)$ of the graph, i.e., the (Hamming) distance is the number of coordinate places in which u and v differ. It is easy to see that the q-ary hypercube is isomorphic to the Cartesian product of n copies of K_q. Denote the all-zero word of $\mathbb{F}_q^n$ by $\mathbf{0}$. A word with one at the ith coordinate place and zero in other coordinates is denoted by e_i.

A linear subspace of $\mathbb{F}_q^n$ is called a *q-ary linear code*. Let C be a linear code in $\mathbb{F}_q^n$ with dimension d. Then there exists an $(n-d) \times n$ matrix H such that $Hu^T = \mathbf{0}$ if and only if $u \in C$. Now H is called the *parity check matrix* of C. Observe that an equivalence relation in $\mathbb{F}_q^n \setminus \{\mathbf{0}\}$ is obtained by defining for all $u, v \in \mathbb{F}_q^n \setminus \{\mathbf{0}\}$, $u \sim v$ if and only if $u = \lambda v$ for some $\lambda \in \mathbb{F}_q^* = \mathbb{F}_q \setminus \{0\}$. Now each equivalence class consists of $q-1$ words of $\mathbb{F}_q^n \setminus \{\mathbf{0}\}$. Assuming k is a positive integer, we form a $k \times (q^k-1)/(q-1)$ matrix H over $\mathbb{F}_q$ by taking as its columns one element from each equivalence class of $\mathbb{F}_q^k \setminus \{\mathbf{0}\}$. Concerning H as a parity check matrix, we obtain a linear code C of length n and dimension $n-k$ such that $|I(C;u)| = 1$ for all $u \in \mathbb{F}_q^n$, i.e., the Hamming distance between any two codewords of C is at least three. The linear code C is called the *Hamming code* of length n and it consists of q^{n-k} codewords.

Let us first begin by presenting a lemma which proves useful in later discussions.

Lemma 35. *Let C be a code in $\mathbb{F}_q^n$.*

(i) For two distinct codewords c_1 and c_2 of C, we have

$$|N[c_1] \cap N[c_2]| = \begin{cases} q, & \text{if } d(c_1,c_2) = 1; \\ 2, & \text{if } d(c_1,c_2) = 2; \\ 0, & \text{if } d(c_1,c_2) > 2. \end{cases}$$

(ii) For three distinct codewords c_1, c_2 and c_3 of C such that there exists a pair of them with the distance equal to 2 and there exists $u \in \mathbb{F}_q^n$ satisfying $c_1, c_2, c_3 \in N[u]$, we obtain that $N[c_1] \cap N[c_2] \cap N[c_3] = \{u\}$.

Proof. (i) Let $c_1, c_2 \in C$ be such that $c_1 \neq c_2$. If $d(c_1,c_2) > 2$, then it is immediate that $N[c_1] \cap N[c_2] = \varnothing$. Furthermore, if $d(c_1,c_2) = 1$, then c_1 and c_2 differ in exactly one coordinate place and, hence, the intersection $N[c_1] \cap N[c_2]$ consists of all the words having same values in the rest of the $n-1$ coordinate. Therefore, we have $|N[c_1] \cap N[c_2]| = q$. Finally, suppose that $d(c_1,c_2) = 2$. Without loss of generality, we may assume that $c_1 = c_2 + \lambda_1 e_1 + \lambda_2 e_2$ for some $\lambda_i \in \mathbb{F}_q (i = 1,2)$. Hence, we have $N[c_1] \cap N[c_2] = \{c_1 + \lambda_1 e_1, c_1 + \lambda_2 e_2\}$ and $|N[c_1] \cap N[c_2]| = 2$.

(ii) Let c_1, c_2 and c_3 be distinct codewords of C such that the distance between two of them is equal to two and there exists $u \in \mathbb{F}_q^n$ satisfying $c_1, c_2, c_3 \in N[u]$. Without loss of generality, we may assume that $d(c_1,c_2) = 2$. By the first case, we obtain that $N[c_1] \cap N[c_2] = \{u, v\}$ and $d(u,v) = 2$ for some $v \in \mathbb{F}_q^n$. Now, without loss of generality, we may assume that $c_1 = u + \lambda_1 e_1$ and $c_2 = u + \lambda_2 e_2$, where $\lambda_1, \lambda_2 \in \mathbb{F}_q$. Therefore, we have $v = u + \lambda_1 e_1 + \lambda_2 e_2$. Hence, we are immediately done if $u = c_3$ as $d(c_3, v) \geqslant 2$. Otherwise, $c_3 = u + \lambda_i' e_i$ ($\lambda_i' \in \mathbb{F}_q$) with c_3 being distinct from c_1 and c_2 and the claim follows as $d(c_3, v) \geqslant 2$. Thus, in all cases, we obtain that $N[c_1] \cap N[c_2] \cap N[c_3] = \{u\}$. □

In what follows, we introduce an approach to construct identifying codes based on self-identifying codes in $\mathbb{F}_q^n$. We first begin by presenting a characterization for self-identifying codes in $\mathbb{F}_q^n$.

Theorem 36. *A code C is self-identifying in $\mathbb{F}_q^n$ if and only if for each word $u \in \mathbb{F}_q^n$ we have $|I(C;u)| \geqslant 3$ and there exist $c_1, c_2 \in I(C;u)$ such that $d(c_1, c_2) = 2$.*

Proof. Assume that C is a self-identifying code in $\mathbb{F}_q^n$. Suppose first that there exists $u \in \mathbb{F}_q^n$ such that $I(C;u) = \{c_1, c_2\}$, where $c_1, c_2 \in C$. By Lemma 35, we obtain that the intersection $N[c_1] \cap N[c_2]$ contains at least two vertices. This contradicts with the characterization of Theorem 4. Similarly, there does not exist a word $u \in \mathbb{F}_q^n$ which is covered by exactly one codeword of C. Hence, we may assume that $I(C;u)$ contains at least 3 codewords. Suppose then that there does not exist a pair of codewords c and c' in $I(C;u)$ such that $d(c,c') = 2$. Hence, the pairwise distance of any two codewords in $I(C;u)$ is one, i.e., the codewords differ in only one coordinate. This implies that the intersection

$$\bigcap_{c \in I(C;u)} N[c]$$

contains q words contradicting with the assumption that C is a self-identifying code. Thus, the claim follows.

Assume then that $C \subseteq \mathbb{F}_q^n$ is a code such that for any $u \in \mathbb{F}_q^n$ we have $|I(C;u)| \geqslant 3$ and there exist $c_1, c_2 \in I(C;u)$ with $d(c_1, c_2) = 2$. By Lemma 35(ii), we immediately obtain that for any $v \in \mathbb{F}_q^n$ we have

$$\bigcap_{c \in I(C;v)} N[c] = \{v\}.$$

Thus, C is a self-locating-dominating code in $\mathbb{F}_q^n$. □

For the next theorem, we recall the following notation: for any word $u \in \mathbb{F}_q^n$ and code $C \subseteq \mathbb{F}_q^n$,

$$u + C = \{u + c \mid c \in C\}.$$

In the following theorem, we present an infinite family of optimal self-identifying codes in $\mathbb{F}_q^n$.

Theorem 37. *Let q be a prime power and let n and k be integers such that $n = (q^k - 1)/(q-1)$. If C is a Hamming code in $\mathbb{F}_q^n$, then $C \cup (e_1 + C) \cup (e_2 + C)$ is an optimal self-identifying code in $\mathbb{F}_q^n$ with cardinality $3q^{n-k}$.*

Proof. Let C be a Hamming code in $\mathbb{F}_q^n$ and denote the code $C \cup (e_1 + C) \cup (e_2 + C)$ by C'. Since e_1, e_2 and $e_2 - e_1$ do not belong to C, the code C' is formed by the Hamming code C and two of its distinct cosets. Hence, it is immediate that each word of $\mathbb{F}_q^n$ is covered by exactly three codewords. Therefore, the claim clearly follows if for all $u \in \mathbb{F}_q^n$ there exist $c_1, c_2 \in I(C';u)$ such that $d(c_1, c_2) = 2$. Suppose to the contrary that there exists a word $v \in \mathbb{F}_q^n$ such that the pairwise distance of any two codewords of $I(C;v)$ is one. By the construction of C', we have $I(C';v) = \{c_1, c_2, c_3\}$, where $c_1 \in C$, $c_2 \in e_1 + C$ and

$c_3 \in e_2 + C$. Therefore, as $d(c_1, c_2) = 1$ and $d(c_1, c_3) = 1$, we respectively have $c_2 = c_1 + e_1$ and $c_3 = c_1 + e_2$. However, then a contradiction follows since $d(c_2, c_3) = 2$. Thus, the code C' is self-identifying in $\mathbb{F}_q^n$ with $|C'| = 3q^{n-k}$ (as $|C| = q^{n-k}$).

On the other hand, if D is an arbitrary self-identifying code in $\mathbb{F}_q^n$, then each word of $\mathbb{F}_q^n$ is covered by at least three codewords of D by Theorem 36. Therefore, using a double counting argument similar to the proof of Theorem 11, we obtain that $(n(q-1)+1)|D| \geqslant 3q^n$ which further implies $|D| \geqslant 3q^{n-k}$. Therefore, the self-identifying code C' in $\mathbb{F}_q^n$ is optimal as $|C'| = 3q^{n-k}$. This concludes the proof of the claim. □

For the following theorem, we recall the notation of the *direct sum*: for any codes $C_1 \subseteq \mathbb{F}_q^{n_1}$ and $C_2 \subseteq \mathbb{F}_q^{n_2}$, where n_1 and n_2 are positive integers, we denote

$$C_1 \oplus C_2 = \{(c_1, c_2) \mid c_1 \in C_1, c_2 \in C_2\}.$$

In the following theorem, we present a simple method of constructing self-identifying codes in $\mathbb{F}_q^{n+1}$ based on the ones in $\mathbb{F}_q^n$.

Theorem 38. *If C is a self-identifying code in $\mathbb{F}_q^n$, then $C \oplus \mathbb{F}_q$ is a self-identifying code in $\mathbb{F}_q^{n+1}$.*

Proof. Let C be a self-identifying code in $\mathbb{F}_q^n$ and $u = (u_1, u_2)$ be a word of $\mathbb{F}_q^{n+1}$ such that $u_1 \in \mathbb{F}_q^n$ and $u_2 \in \mathbb{F}_q$. By Theorem 36, the word u_1 is covered by at least three codewords c_1, c_2 and c_3 of C with the additional property that the distance between two of them is equal to two. Hence, the word $u \in \mathbb{F}_q^{n+1}$ is covered at least by the codewords (c_1, u_2), (c_2, u_2) and (c_3, u_2) of $C \oplus \mathbb{F}_q$, and the codewords satisfy the additional property that the distance between two of them is equal to two. Therefore, by Theorem 36, the code $C \oplus \mathbb{F}_q$ is self-identifying in $\mathbb{F}_q^{n+1}$. □

Recall that each self-identifying code is always identifying. Therefore, by the previous theorems, we have $\gamma^{ID}(\mathbb{F}_q^n) \leqslant 3q^{n-k}$ for integers n, k and ℓ such that $n = (q^k-1)/(q-1)+\ell$, where q is a prime power. This significantly improves over the previous upper bound $\gamma^{ID}(\mathbb{F}_q^n) \leqslant q^{n-1}$ by Goddard and Wash [4]; however, recall that they do not require that q is a prime power. In what follows, we introduce another way to construct identifying codes based on self-locating-dominating codes in $\mathbb{F}_q^n$. We first begin by presenting a characterization for self-locating-dominating codes in $\mathbb{F}_q^n$.

Theorem 39. *A code C is self-locating-dominating in $\mathbb{F}_q^n$ if and only if for each word $u \in \mathbb{F}_q^n \setminus C$ we have $|I(C; u)| \geqslant 3$ and there exist $c_1, c_2 \in I(C; u)$ such that $d(c_1, c_2) = 2$.*

Proof. Recall that a code C is self-locating-dominating if for all $u \in \mathbb{F}_q^n \setminus C$

$$\bigcap_{c \in I(C;u)} N[c] = \{u\}.$$

By Theorem 4, a code is self-identifying if and only if the same condition is satisfied for all words $u \in \mathbb{F}_q^n$. Hence, the claim follows by an argument analogous to the proof of Theorem 36. □

In the following theorem, we present an infinite family of optimal self-locating-dominating codes in $\mathbb{F}_q^n$.

Theorem 40. *Let q be a prime power and let n and k be integers such that $n = 3(q^k - 1)/(q-1)$. Assume that H is a $k \times n$ parity check matrix formed from the $k \times (n/3)$ parity check matrix of the Hamming code by repeating each column three times. Now the code C corresponding to the parity check matrix H is an optimal self-locating-dominating code in $\mathbb{F}_q^n$ with cardinality q^{n-k}.*

Proof. Let u be a word of $\mathbb{F}_q^n$. Now we obtain the following observations:

- Suppose that $Hu^T = x \in \mathbb{F}_q^k$ and $x \neq \mathbf{0}$. Due to the construction of the parity check matrix H, there exist exactly three columns h_{i_1}, h_{i_2} and h_{i_3} of H such that $x = \lambda h_{i_1} = \lambda h_{i_2} = \lambda h_{i_3}$ for some $\lambda \in \mathbb{F}_q$. Hence, there exist exactly three words λe_{i_1}, λe_{i_2} and λe_{i_3} of weight one in $\mathbb{F}_q^n$ such that the indices i_j are all different and $H(u + \lambda e_{i_j})^T = \mathbf{0}$, i.e., $u + \lambda e_{i_j}$ belongs to C. Therefore, the word u is covered by exactly three codewords of C in $\mathbb{F}_q^n$. Moreover, the distance between any of these codewords is equal to two.

- If $Hu^T = \mathbf{0} \in \mathbb{F}_q^k$, then analogously to the previous case we can observe that $u \in C$ is covered by exactly one codeword of C in $\mathbb{F}_q^n$; namely, by itself.

Thus, by the previous observations, we know that $I(C; u) = \{u\}$ if $u \in C$ and for non-codewords $u \in \mathbb{F}_q^n \setminus C$ we have $|I(C; u)| = 3$ with the additional property that the distance between any two codewords of $I(C; u)$ is equal to two. Therefore, by Theorem 39, the code C is self-locating-dominating in $\mathbb{F}_q^n$. Moreover, it is easy to calculate that $|C| = q^{n-k}$.

On the other hand, if D is an arbitrary self-locating-dominating code in $\mathbb{F}_q^n$, then each word of $\mathbb{F}_q^n \setminus D$ is covered by at least three codewords of D by Theorem 39. Therefore, using a double counting argument similar to the proof of Theorem 37, we obtain that $(n(q-1)+1)|D| \geqslant 3(q^n - |D|) + |D|$ which further implies $|D| \geqslant q^{n-k}$. Therefore, the self-locating-dominating code C in $\mathbb{F}_q^n$ is optimal as $|C| = q^{n-k}$. This concludes the proof of the claim. □

In a similar way, we can also construct self-locating-dominating codes (albeit not optimal) for other lengths n.

Theorem 41. *Let q be a prime power and let n, k and ℓ be integers such that $n = 3(q^k - 1)/(q-1) + \ell$. Assume that H is a $k \times n$ parity check matrix formed from the $k \times ((n-\ell)/3)$ parity check matrix of the Hamming code by repeating the first column $\ell + 3$ times and each other column three times. Now the code C corresponding to the parity check matrix H is self-locating-dominating in $\mathbb{F}_q^n$ with cardinality q^{n-k}.*

Proof. The proof of the claim is similar to the one of Theorem 40. □

Observe that the codes $C \subseteq \mathbb{F}_q^n$ constructed in Theorems 40 and 41 are such that for each codeword $c \in V$ we have $I(c) = \{c\}$ and for non-codewords $u \in \mathbb{F}_q^n \setminus C$ we have

$$\bigcap_{c \in I(u)} N[c] = \{u\}.$$

Therefore, all the constructed self-locating-dominating codes are also identifying in $\mathbb{F}_q^n$. Hence, we have $\gamma^{ID}(\mathbb{F}_q^n) \leqslant q^{n-k}$ for all integers n, k and ℓ such that $n = 3(q^k-1)/(q-1) + \ell$, where q is a prime power. Thus, using the constructions based on the self-locating-dominating codes, we are able to significantly improve the previous upper bound $\gamma^{ID}(\mathbb{F}_q^n) \leqslant q^{n-1}$ by [4] (recall again that in [4] it is not required that q is a prime power).

In [4], it is stated that the best known lower bound for identifying codes in $\mathbb{F}_q^n$ is the following one by Karpovsky *et al.* [17].

Theorem 42 ([17])**.** *We have*

$$\gamma^{ID}(\mathbb{F}_q^n) \geqslant \frac{2q^n}{nq-n+2}.$$

Assume that q is a prime power and n and k are integers such that $n = 3(q^k-1)/(q-1)$. As stated above, we now have $\gamma^{ID}(\mathbb{F}_q^n) \leqslant q^{n-k}$. Now the previous lower bound can be written as follows:

$$\gamma^{ID}(\mathbb{F}_q^n) \geqslant \frac{2q^n}{nq-n+2} = \frac{2q^n}{3q^k-1} \geqslant \frac{2q^n}{3q^k} = \frac{2}{3}q^{n-k}.$$

Hence, comparing the previous lower and upper bounds, it can be seen that they are of the same order $\Theta(q^{n-k})$. Analogously, it can be shown that the (self-)identifying codes obtained in Theorem 37 for lengths $n = (q^k-1)/(q-1)$ are also rather small compared to the lower bound above.

References

[1] N. Bousquet, A. Lagoutte, Z. Li, A. Parreau and S. Thomassé. *Identifying codes in hereditary classes of graphs and VC-dimension. SIAM J. Discrete Math.*, 29(4):2047–2064, 2015.

[2] J. Cáceres, C. Hernando, M. Mora, I. M. Pelayo and M. L. Puertas. *Locating-dominating codes: bounds and extremal cardinalities. Appl. Math. Comput.*, 220:38–45, 2013.

[3] G. Cohen, I. Honkala, S. Litsyn and A. Lobstein. *Covering codes*, volume 54 of *North-Holland Mathematical Library.* North-Holland Publishing Co., Amsterdam, 1997.

[4] W. Goddard and K. Wash. ID codes in Cartesian products of cliques. *J. Combin. Math. Combin. Comput.*, 85:97–106, 2013.

[5] S. Gravier, J. Moncel, and A. Semri. Identifying codes of Cartesian product of two cliques of the same size. *Electron. J. Combin.*, 15(1):#N4, 2008.

[6] T. Evans. Embedding Incomplete Latin Squares. *The American Mathematical Monthly*, number 10, volume 67:958–961, 1960.

[7] N. Fazlollahi, D. Starobinski, and A. Trachtenberg. Connected identifying codes. *IEEE Trans. Inform. Theory*, 58(7):4814–4824, 2012.

[8] S. Gravier, A. Parreau, S. Rottey, L. Storme and É. Vandomme. *Identifying codes in vertex-transitive graphs and strongly regular graphs. Electron. J. Combin.*, 22(4):Paper 4.6, 2015.

[9] T. Haynes, S. Hedetniemi and P. Slater. *Fundamentals of Domination in Graphs*, volume 208 of *Pure and Applied Mathematics*. Marcel Dekker, Inc., New York, 1998.

[10] S. Hedetniemi. On identifying codes in the Cartesian product of a path and a complete graph. *J. Comb. Optim.*, 31(4):1405–1416, 2016.

[11] I. Honkala and T. Laihonen. On a new class of identifying codes in graphs. *Inform. Process. Lett.*, 102(2-3):92–98, 2007.

[12] O. Hudry and A. Lobstein. More results on the complexity of identifying problems in graphs. *Theoret. Comput. Sci.*, 626:1–12, 2016.

[13] V. Junnila and T. Laihonen. Collection of codes for tolerant location. In *Proceedings of the Bordeaux Graph Workshop*, pages 176–179, 2016.

[14] V. Junnila and T. Laihonen. Tolerant location detection in sensor networks. Submitted, 2016.

[15] V. Junnila, T. Laihonen, and T. Lehtilä. On a Conjecture regarding Identification in the Hamming Graphs. *10th International Colloquium on Graph Theory and combinatorics, 2018*.

[16] V. Junnila, T. Laihonen, and T. Lehtilä. On regular and new types of codes for location-domination. *Discrete Appl. Math.*, 247:225–241, 2018.

[17] M. G. Karpovsky, K. Chakrabarty, and L. B. Levitin. On a new class of codes for identifying vertices in graphs. *IEEE Trans. Inform. Theory*, 44(2):599–611, 1998.

[18] M. Laifenfeld and A. Trachtenberg. Disjoint identifying-codes for arbitrary graphs. In *Proceedings of International Symposium on Information Theory, 2005. ISIT 2005*, pages 244–248, 2005.

[19] J. H. van Lint. *Introduction to coding theory*, volume 86 of *Graduate Texts in Mathematics*. Springer-Verlag, Berlin, 1999.

[20] A. Lobstein. Watching systems, identifying, locating-dominating and discriminating codes in graphs, a bibliography. Published electronically at `https://www.lri.fr/~lobstein/debutBIBidetlocdom.pdf`

[21] D. F. Rall and P. J. Slater. On location-domination numbers for certain classes of graphs. *Congr. Numer.*, 45:97–106, 1984.

[22] D. F. Rall and K. Wash. On Minimum Identifying Codes in Some Cartesian Product Graphs. *Graphs Combin.*, 33(4):1037–1053, 2017.

[23] S. Ray, D. Starobinski, A. Trachtenberg, and R. Ungrangsi. Robust location detection with sensor networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1016–1025, 2004.

[24] P. J. Slater. Domination and location in acyclic graphs. *Networks*, 17(1):55–64, 1987.

[25] P. J. Slater. Dominating and reference sets in a graph. *J. Math. Phys. Sci.*, 22:445–455, 1988.

Appendix

Below we give the tables which show that the code C^1 in Theorem 19 and the code C_L in Theorem 20 are identifying codes.

$I(1,1,1)$	$\{(3,1,1),(1,3,1)\}$	$I(2,1,1)$	$\{(3,1,1),(2,1,3),(2,1,4)\}$
$I(3,1,1)$	$\{(3,1,1)\}$	$I(4,1,1)$	$\{(3,1,1),(4,4,1),(4,1,2)\}$
$I(1,2,1)$	$\{(1,3,1),(1,2,2),(1,2,4)\}$	$I(2,2,1)$	$\{(2,2,4)\}$
$I(3,2,1)$	$\{(3,1,1),(3,2,2)\}$	$I(4,2,1)$	$\{(4,4,1)\}$
$I(1,3,1)$	$\{(1,3,1)\}$	$I(2,3,1)$	$\{(1,3,1),(2,3,2)\}$
$I(3,3,1)$	$\{(1,3,1),(3,3,3),(3,1,1)\}$	$I(4,3,1)$	$\{(1,3,1),(4,3,3),(4,4,1)\}$
$I(1,4,1)$	$\{(1,3,1),(4,4,1)\}$	$I(2,4,1)$	$\{(2,4,4),(4,4,1)\}$
$I(3,4,1)$	$\{(3,1,1),(4,4,1)\}$	$I(4,4,1)$	$\{(4,4,1),(4,4,3)\}$
$I(1,1,2)$	$\{(1,2,2),(4,1,2)\}$	$I(2,1,2)$	$\{(2,1,3),(2,1,4),(2,3,2),(4,1,2)\}$
$I(3,1,2)$	$\{(3,1,1),(3,2,2),(4,1,2)\}$	$I(4,1,2)$	$\{(4,1,2)\}$
$I(1,2,2)$	$\{(1,2,2),(1,2,4),(3,2,2)\}$	$I(2,2,2)$	$\{(1,2,2),(2,2,4),(2,3,2),(3,2,2)\}$
$I(3,2,2)$	$\{(1,2,2),(3,2,2)\}$	$I(4,2,2)$	$\{(1,2,2),(3,2,2),(4,1,2)\}$
$I(1,3,2)$	$\{(1,2,2),(1,3,1),(2,3,2)\}$	$I(2,3,2)$	$\{(2,3,2)\}$
$I(3,3,2)$	$\{(2,3,2),(3,2,2),(3,3,3)\}$	$I(4,3,2)$	$\{(2,3,2),(4,1,2),(4,3,3)\}$
$I(1,4,2)$	$\{(1,2,2)\}$	$I(2,4,2)$	$\{(2,3,2),(2,4,4)\}$
$I(3,4,2)$	$\{(3,2,2)\}$	$I(4,4,2)$	$\{(4,1,2),(4,4,1),(4,4,3)\}$
$I(1,1,3)$	$\{(2,1,3)\}$	$I(2,1,3)$	$\{(2,1,3),(2,1,4)\}$
$I(3,1,3)$	$\{(2,1,3),(3,1,1),(3,3,3)\}$	$I(4,1,3)$	$\{(2,1,3),(4,1,2),(4,3,3),(4,4,3)\}$
$I(1,2,3)$	$\{(1,2,2),(1,2,4)\}$	$I(2,2,3)$	$\{(2,1,3),(2,2,4)\}$
$I(3,2,3)$	$\{(3,2,2),(3,3,3)\}$	$I(4,2,3)$	$\{(4,3,3),(4,4,3)\}$
$I(1,3,3)$	$\{(1,3,1),(3,3,3),(4,3,3)\}$	$I(2,3,3)$	$\{(2,1,3),(2,3,2),(3,3,3),(4,3,3)\}$
$I(3,3,3)$	$\{(3,3,3),(4,3,3)\}$	$I(4,3,3)$	$\{(3,3,3),(4,3,3),(4,4,3)\}$
$I(1,4,3)$	$\{(4,4,3)\}$	$I(2,4,3)$	$\{(2,1,3),(2,4,4),(4,4,3)\}$
$I(3,4,3)$	$\{(3,3,3),(4,4,3)\}$	$I(4,4,3)$	$\{(4,3,3),(4,4,1),(4,4,3)\}$
$I(1,1,4)$	$\{(1,2,4),(2,1,4)\}$	$I(2,1,4)$	$\{(2,1,3),(2,1,4),(2,2,4),(2,4,4)\}$
$I(3,1,4)$	$\{(2,1,4),(3,1,1)\}$	$I(4,1,4)$	$\{(2,1,4),(4,1,2)\}$
$I(1,2,4)$	$\{(1,2,2),(1,2,4),(2,2,4)\}$	$I(2,2,4)$	$\{(1,2,4),(2,1,4),(2,2,4),(2,4,4)\}$
$I(3,2,4)$	$\{(1,2,4),(2,2,4),(3,2,2)\}$	$I(4,2,4)$	$\{(1,2,4),(2,2,4)\}$
$I(1,3,4)$	$\{(1,2,4),(1,3,1)\}$	$I(2,3,4)$	$\{(2,1,4),(2,2,4),(2,3,2),(2,4,4)\}$
$I(3,3,4)$	$\{(3,3,3)\}$	$I(4,3,4)$	$\{(4,3,3)\}$
$I(1,4,4)$	$\{(1,2,4),(2,4,4)\}$	$I(2,4,4)$	$\{(2,1,4),(2,2,4),(2,4,4)\}$
$I(3,4,4)$	$\{(2,4,4)\}$	$I(4,4,4)$	$\{(2,4,4),(4,4,1),(4,4,3)\}$

Table 1: I-sets of code C^1.

$I(1,1,1)$	$\varnothing$	$I(2,1,1)$	$\{(2,1,3),(2,4,1)\}$
$I(3,1,1)$	$\{(3,1,4),(3,2,1)\}$	$I(4,1,1)$	$\{(4,1,2),(4,3,1)\}$
$I(1,2,1)$	$\{(1,2,4),(3,2,1)\}$	$I(2,2,1)$	$\{(2,4,1),(3,2,1)\}$
$I(3,2,1)$	$\{(3,2,1)\}$	$I(4,2,1)$	$\{(3,2,1),(4,2,3),(4,3,1)\}$
$I(1,3,1)$	$\{(1,3,2),(4,3,1)\}$	$I(2,3,1)$	$\{(2,3,4),(2,4,1),(4,3,1)\}$
$I(3,3,1)$	$\{(3,2,1),(4,3,1)\}$	$I(4,3,1)$	$\{(4,3,1)\}$
$I(1,4,1)$	$\{(1,4,3),(2,4,1)\}$	$I(2,4,1)$	$\{(2,4,1)\}$
$I(3,4,1)$	$\{(2,4,1),(3,2,1),(3,4,2)\}$	$I(4,4,1)$	$\{(2,4,1),(4,3,1)\}$
$I(1,1,2)$	$\{(1,3,2),(4,1,2)\}$	$I(2,1,2)$	$\{(2,1,3),(4,1,2)\}$
$I(3,1,2)$	$\{(3,1,4),(3,4,2),(4,1,2)\}$	$I(4,1,2)$	$\{(4,1,2)\}$
$I(1,2,2)$	$\{(1,2,4),(1,3,2)\}$	$I(2,2,2)$	$\varnothing$
$I(3,2,2)$	$\{(3,2,1),(3,4,2)\}$	$I(4,2,2)$	$\{(4,1,2),(4,2,3)\}$
$I(1,3,2)$	$\{(1,3,2)\}$	$I(2,3,2)$	$\{(1,3,2),(2,3,4)\}$
$I(3,3,2)$	$\{(1,3,2),(3,4,2)\}$	$I(4,3,2)$	$\{(1,3,2),(4,1,2),(4,3,1)\}$
$I(1,4,2)$	$\{(1,3,2),(1,4,3),(3,4,2)\}$	$I(2,4,2)$	$\{(2,4,1),(3,4,2)\}$
$I(3,4,2)$	$\{(3,4,2)\}$	$I(4,4,2)$	$\{(3,4,2),(4,1,2)\}$
$I(1,1,3)$	$\{(1,4,3),(2,1,3)\}$	$I(2,1,3)$	$\{(2,1,3)\}$
$I(3,1,3)$	$\{(2,1,3),(3,1,4)\}$	$I(4,1,3)$	$\{(2,1,3),(4,1,2),(4,2,3)\}$
$I(1,2,3)$	$\{(1,2,4),(1,4,3),(4,2,3)\}$	$I(2,2,3)$	$\{(2,1,3),(4,2,3)\}$
$I(3,2,3)$	$\{(3,2,1),(4,2,3)\}$	$I(4,2,3)$	$\{(4,2,3)\}$
$I(1,3,3)$	$\{(1,3,2),(1,4,3)\}$	$I(2,3,3)$	$\{(2,1,3),(2,3,4)\}$
$I(3,3,3)$	$\varnothing$	$I(4,3,3)$	$\{(4,2,3),(4,3,1)\}$
$I(1,4,3)$	$\{(1,4,3)\}$	$I(2,4,3)$	$\{(1,4,3),(2,1,3),(2,4,1)\}$
$I(3,4,3)$	$\{(1,4,3),(3,4,2)\}$	$I(4,4,3)$	$\{(1,4,3),(4,2,3)\}$
$I(1,1,4)$	$\{(1,2,4),(3,1,4)\}$	$I(2,1,4)$	$\{(2,1,3),(2,3,4),(3,1,4)\}$
$I(3,1,4)$	$\{(3,1,4)\}$	$I(4,1,4)$	$\{(3,1,4),(4,1,2)\}$
$I(1,2,4)$	$\{(1,2,4)\}$	$I(2,2,4)$	$\{(1,2,4),(2,3,4)\}$
$I(3,2,4)$	$\{(1,2,4),(3,1,4),(3,2,1)\}$	$I(4,2,4)$	$\{(1,2,4),(4,2,3)\}$
$I(1,3,4)$	$\{(1,2,4),(1,3,2),(2,3,4)\}$	$I(2,3,4)$	$\{(2,3,4)\}$
$I(3,3,4)$	$\{(2,3,4),(3,1,4)\}$	$I(4,3,4)$	$\{(2,3,4),(4,3,1)\}$
$I(1,4,4)$	$\{(1,2,4),(1,4,3)\}$	$I(2,4,4)$	$\{(2,3,4),(2,4,1)\}$
$I(3,4,4)$	$\{(3,1,4),(3,4,2)\}$	$I(4,4,4)$	$\varnothing$

Table 2: I-sets of code C_L, the codewords are highlighted. The empty I-sets belong to the diagonal vertices.

Publication IV

Ville Junnila, Tero Laihonen and Tuomo Lehtilä. New optimal results on codes for location in graphs, *Fundamenta Informaticae*, submitted for publication, 2020.

New Optimal Results on Codes for Location in Graphs[*]

Ville Junnila, **Tero Laihonen** and **Tuomo Lehtilä**[†]
Department of Mathematics and Statistics
University of Turku, FI-20014 Turku, Finland
viljun@utu.fi, terolai@utu.fi and tualeh@utu.fi

Abstract

In this paper, we broaden the understanding of the recently introduced concepts of solid-locating-dominating and self-locating-dominating codes in various graphs. In particular, we present the optimal, i.e., smallest possible, codes in the infinite triangular and king grids. Furthermore, we give optimal locating-dominating, self-locating-dominating and solid-locating-dominating codes in the direct product $K_n \times K_m$ of complete graphs. We also present optimal solid-locating-dominating codes for the Hamming graphs $K_q \square K_q \square K_q$ with $q \geq 2$.

Keywords: Location-domination; solid-location-domination; self-location-domination; king grid; direct product; Hamming graph

1 Introduction

Sensor networks consist of sensors monitoring various places and connections between these places (see [11]). A sensor network is modeled as a simple and undirected graph $G = (V(G), E(G)) = (V, E)$. In this context, a sensor can be placed on a vertex v and its closed neighbourhood $N[v]$ represents the set of locations that the sensor monitors. Besides assuming that graphs are simple and undirected, we also assume that they are connected and have cardinality at least two. In the following, we present some terminology and notation. The *closed neighbourhood* of v is defined $N[v] = N(v) \cup \{v\}$, where $N(v)$ is the *open neighbourhood* of v, that is, the set of vertices adjacent to v. A *code* C is a nonempty subset of V and its elements are *codewords*. The codeword $c \in C$ *covers* a vertex $v \in V$ if $v \in N[c]$. We denote the set of codewords covering v in G by

$$I(G, C; v) = I(G; v) = I(C; v) = I(v) = N[v] \cap C.$$

The set $I(v)$ is called an *identifying set* or an *I-set*. We say that a code $C \subseteq V$ is *dominating* in G if $I(C; u) \neq \emptyset$ for all $u \in V$. If the sensors are placed at the locations corresponding to the codewords, then each vertex is monitored by the sensors located in $I(v)$. More explanation regarding location detection in the sensor networks can be found in [1, 10, 14].

Let us now define *identifying codes*, which were first introduced by Karpovsky *et al.* in [9]. For numerous papers regarding identifying codes and related topics, the interested reader is referred to the online bibliography [11].

Definition 1. A code $C \subseteq V$ is *identifying* in G if for all distinct $u, v \in V$ we have $I(C; u) \neq \emptyset$ and

$$I(C; u) \neq I(C; v).$$

An identifying code C in a finite graph G with the smallest cardinality is called *optimal* and the number of codewords in an optimal identifying code is denoted by $\gamma^{ID}(G)$.

---

[*] An extended abstract [5] of the paper has been presented at the Fifth Russian Finnish Symposium on Discrete Mathematics.

[†] Research supported by the University of Turku Graduate School (UTUGS), the Vilho, Yrjö and Kalle Väisälä Foundation, and the Jenny and Antti Wihuri Foundation.

Identifying codes require unique I-sets for codewords as well as for non-codewords. However, if we omit the requirement of unique I-sets for codewords, then we obtain the following definition of *locating-dominating codes*, which were first introduced by Slater in [12, 15, 16].

Definition 2. A code $C \subseteq V$ is *locating-dominating* in G if for all distinct $u, v \in V \setminus C$ we have $I(C; u) \neq \emptyset$ and

$$I(C; u) \neq I(C; v).$$

Notice that an identifying code in G is also locating-dominating (by the definitions). In [6], self-locating-dominating and solid-locating-dominating codes have been introduced and, in [7, 8], they have been further studied. The definitions of these codes are given as follows.

Definition 3. Let $C \subseteq V$ be a code in G.

(i) We say that $C \subseteq V$ is *self-locating-dominating code* in G if for all $u \in V \setminus C$ we have $I(C; u) \neq \emptyset$ and

$$\bigcap_{c \in I(C;u)} N[c] = \{u\}.$$

(ii) We say that $C \subseteq V$ is *solid-locating-dominating code* in G if for all distinct $u, v \in V \setminus C$ we have

$$I(C; u) \setminus I(C; v) \neq \emptyset.$$

Observe that since G is a connected graph on at least two vertices, a self-locating-dominating and solid-locating-dominating code is always dominating. Analogously to identifying codes, in a finite graph G, we say that dominating, locating-dominating, self-locating-dominating and solid-locating-dominating codes with the smallest cardinalities are *optimal* and we denote the cardinality of an optimal code by $\gamma(G)$, $\gamma^{LD}(G)$, $\gamma^{SLD}(G)$ and $\gamma^{DLD}(G)$, respectively.

In the following theorem, we offer characterizations of self-locating-dominating and solid-dominating codes for easier comparison of them.

Theorem 4 ([6])**.** *Let $G = (V, E)$ be a connected graph on at least two vertices:*

(i) A code $C \subseteq V$ is self-locating-dominating if and only if for all distinct $u \in V \setminus C$ and $v \in V$ we have

$$I(C; u) \setminus I(C; v) \neq \emptyset.$$

(ii) A code $C \subseteq V$ is solid-locating-dominating if and only if for all $u \in V \setminus C$ we have $I(C; u) \neq \emptyset$ and

$$\left(\bigcap_{c \in I(C;u)} N[c] \right) \setminus C = \{u\}.$$

Based on the previous theorem, we obtain the following corollary.

Corollary 5. *If C is a self-locating-dominating or solid-locating-dominating code in G, then C is also solid-locating-dominating or locating-dominating in G, respectively. Furthermore, for a finite graph G, we have*

$$\gamma^{LD}(G) \leq \gamma^{DLD}(G) \leq \gamma^{SLD}(G).$$

The structure of the paper is described as follows. First, in Section 2, we obtain optimal self-locating-dominating and solid-locating-dominating codes in the infinite triangular and king grids, i.e., the smallest possible codes regarding their density (a concept defined later). Regarding the triangular grid, the proofs are rather simple and straightforward, but they serve as nice introductory examples to the concepts of solid-location-domination and self-location-domination. However, the case with the king grid is more interesting; in particular, the proof of the lower bound for solid-location-domination is based on global arguments instead of only local ones, which are more usual in domination type problems. Then, in Section 3, we give optimal locating-dominating, self-locating-dominating and solid-locating-dominating codes in the direct product $K_n \times K_m$ of complete graphs, where $2 \leq n \leq m$. Finally, in Section 4, we present optimal solid-locating-dominating codes for graphs $K_q \square K_q \square K_q$ with $q \geq 2$.

2 Triangular and king grids

In this section, we consider solid-location-domination and self-location-domination in the so called infinite triangular and king grids, which are widely studied graphs in the field of domination (see [11]). As defined in the introduction, for finite graphs, the optimality of a code has been defined using the minimum cardinality. However, this method is not valid for the infinite graphs of this section. Hence, we need to use the usual concept of *density* of a code (see various papers concerning infinite grids in [11]). Let us first consider the *infinite triangular grid*.

Definition 6. Let $G = (V, E)$ be a graph with the vertex set

$$V = \left\{ i(1,0) + j\left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right) \mid i, j \in \mathbb{Z} \right\}$$

and two vertices are defined to be adjacent if their Euclidean distance is equal to one. The obtained graph G is called the *infinite triangular grid* and it is illustrated in Figure 1. We further denote $v(i,j) = i(1,0) + j\left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right)$. Let R_n be the subgraph of G induced by the vertex set $V_n = \{v(i,j) \mid |i|, |j| \leq n\}$. The *density* of a code in G is now defined as follows:

$$D(C) = \limsup_{n\to\infty} \frac{|C \cap V_n|}{|V_n|}$$

We say that a code is *optimal* if there exists no other code with smaller density.

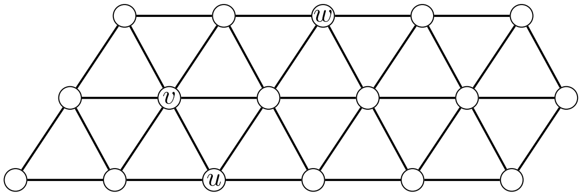


Figure 1: Triangular grid with the vertices $v = v(0,0)$, $u = v(1,-1)$ and $w = v(1,1)$.

In the following theorem, optimal self-locating-dominating and solid-locating-dominating codes are given in the triangular grid. The methods used in the proof are rather typical for domination type of problems. However, we present the proof for completeness and as an introductory example.

Theorem 7. *Let $G = (V, E)$ be the triangular grid. The code*

$$C = \{v(i,j) \mid i, j \equiv 0 \pmod 2\}$$

is self-locating-dominating in G and, therefore, also solid-locating-dominating. The density of the code C is equal to $1/4$ and there exists no self-locating-dominating or solid-locating-dominating code with smaller density, i.e., the code is optimal in both cases.

Proof. Let us first show that the code C is self-locating-dominating in the triangular grid G. The proof now divides into the following cases depending on the parity of i and j in $v(i,j)$:

- If i is odd and j is even, then $I(v(i,j)) = \{v(i-1,j), v(i+1,j)\}$ and $N[v(i-1,j)] \cap N[v(i+1,j)] = \{v(i,j)\}$.
- Analogously, if i is even and j is odd, then $I(v(i,j)) = \{v(i,j-1), v(i,j+1)\}$ and $N[v(i,j-1)] \cap N[v(i,j+1)] = \{v(i,j)\}$.

- Finally, if i and j are both odd, then $I(v(i,j)) = \{v(i-1,j+1), v(i+1,j-1)\}$ and $N[v(i-1,j+1)] \cap N[v(i+1,j-1)] = \{v(i,j)\}$.

Thus, as $v(i,j)$ is a codeword for even i and j, the code C is self-locating-dominating in G. Furthermore, we have $D(C) = \frac{1}{4}$ since $v(i,j)$ is a codeword if and only if i and j are both even. Notice that C is also a solid-locating-dominating code.

For the lower bound, assume that C' is a solid-locating-dominating code in G. Immediately, by the definition of solid-locating-dominating codes, we know that $|I(C';u)| \geq 2$ for any non-codeword u. Therefore, by counting in two ways the pairs (u,c), where $c \in C' \cap V_n$ and $u \in N[c] \cap V_{n-1}$, we obtain that $7|C' \cap V_n| \geq |C' \cap V_{n-1}| + 2(|V_{n-1}| - |C' \cap V_{n-1}|) \geq 2|V_{n-1}| - |C' \cap V_{n-1}| \geq 2|V_{n-1}| - |C' \cap V_n|$, which is equivalent to $|C' \cap V_n| \geq |V_{n-1}|/4$. Thus, we may estimate the density of C' as follows:

$$D(C') = \limsup_{n\to\infty} \frac{|C' \cap V_n|}{|V_n|} \geq \limsup_{n\to\infty} \frac{|V_{n-1}|/4}{|V_n|} = \frac{1}{4}.$$

□

Next we consider the more interesting problems of solid-location-domination and self-location-domination in the infinite king grid. Let us first begin by defining the grid and the density of a code in it.

Definition 8. Let $G = (V,E)$ be a graph with $V = \mathbb{Z}^2$ and for the vertices $v = (v_1, v_2) \in V$ and $u = (u_1, u_2) \in V$ we have $vu \in E$ if and only if $|v_1 - u_1| \leq 1$ and $|v_2 - u_2| \leq 1$. The obtained graph G is called the *infinite king grid*. Further let V_n be a subset of V such that $V_n = \{(x,y) \mid |x| \leq n, |y| \leq n\}$. The *density* of a code $C \subseteq V = \mathbb{Z}^2$ is now defined as

$$D(C) = \limsup_{n\to\infty} \frac{|C \cap V_n|}{|V_n|}.$$

We say that a code is *optimal* if there exists no other code with smaller density.

In what follows, we first consider solid-location-domination in the king grid. In the following theorem, we present a solid-locating-dominating code in the king grid with density 1/3. Later, in Theorem 11, it is shown that the code is optimal.

Theorem 9. *Let $G = (V,E)$ be the king grid. The code*

$$C = \big\{(x,y) \in \mathbb{Z}^2 \mid |x| + |y| \equiv 0 \pmod 3\big\}$$

is solid-locating-dominating in G and its density is 1/3.

Proof. Let $C = \big\{(x,y) \in \mathbb{Z}^2 \mid |x| + |y| \equiv 0 \mod 3\big\}$ be a code in G (illustrated in Figure 2). By the definition, it is immediate that the density of C is equal to 1/3. In order to show that C is a solid-locating-dominating code in G, we prove that the condition of Theorem 4(ii) holds for every non-codeword of G. Let $u = (x,y) \in \mathbb{Z}^2$ be a vertex not belonging to C. Suppose first that $x = 0$ and $y > 0$. Now, if $y \equiv 1 \pmod 3$, then $I(u) = \{u + (0,-1), u + (-1,1), u + (1,1)\}$ and $N[u + (0,-1)] \cap N[u + (-1,1)] \cap N[u + (1,1)] = \{u\}$, else $y \equiv 2 \pmod 3$ implying $I(u) = \{u + (-1,0), u + (1,0), u + (0,1)\}$ and $(N[u + (-1,0)] \cap N[u + (1,0)] \cap N[u + (0,1)]) \setminus C = \{u\}$. Thus, the required condition is met. The case with $y < 0$ is analogous. Moreover, the case with $y = 0$ is symmetrical to the one with $x = 0$. Hence, we may assume that $x \neq 0$ and $y \neq 0$.

Suppose then that $x \geq 1$ and $y \geq 1$. Now we have either $I(u) = \{u + (0,-1), u + (-1,0), u + (1,1)\}$ or $I(u) = \{u + (0,1), u + (1,0), u + (-1,-1)\}$. In both cases, we obtain that $\bigcap_{c \in I(u)} N[c] = \{u\}$ and the condition is satisfied. The other (three) cases with $x \leq -1$ or $y \leq -1$ can be handled analogously. Thus, in conclusion, C is a solid-locating-dominating code in G. □

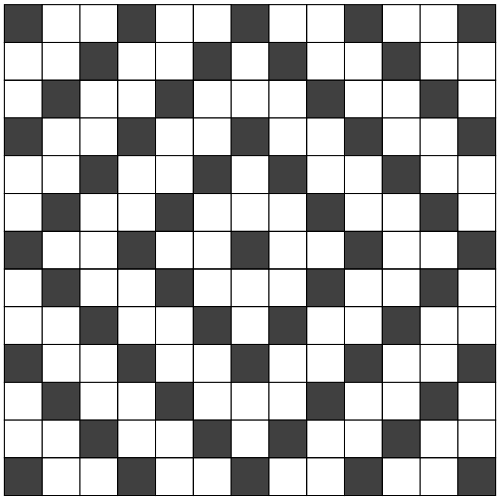


Figure 2: The darkened squares form a solid-locating-dominating code of density $\frac{1}{3}$ in the king grid.

Usually, the best known constructions for domination type codes in infinite grids are formed by a repetition of a finite pattern. However, this is not the case with the code C of the previous theorem. Another observation is that the codeword $c = (0,0)$ has a special role as a sort of center of the code. In particular, the density of the code (or more precisely the ratio $|C \cap V_n|/|V_n|$) in the close proximity of c is less than 1/3. Consider now the lower bound on the density of a solid-locating-dominating code. Usually, the lower bounds are obtained by locally studying the symmetric difference of closed neighbourhoods of vertices or the domination properties of vertices (such as the concept of share [17] or the common technique used in the proof of Theorem 7). However, in order to deal with the special type of codewords c, we develop a new technique of more global nature. For this purpose, we first present the following lemma on a forbidden pattern of non-codewords.

Lemma 10. *Let $G = (V,E)$ be the king grid and $C \subseteq V$ be a solid-locating-dominating code in G. Then $T = \{(i,j),(i,j+1),(i,j+2),(i+1,j+2),(i-1,j+2)\}$ and any formation obtained from T by a rotation of $\pi/2$, π or $3\pi/2$ radians around the origin contains a codeword of C.*

Proof. Assume that the set $T = \{(i,j),(i,j+1),(i,j+2),(i+1,j+2),(i-1,j+2)\}$ contains no codewords of C. Then a contradiction with the definition follows since $I(i,j+1) \setminus I(i,j) = \emptyset$. The other cases obtained from T by a rotation are proved analogously. $\square$

In the following theorem, we prove that the solid-locating-dominating code of Theorem 9 is optimal, i.e., there is no code with density smaller than 1/3. The proof is based on the idea of studying one-way infinite strips of vertices of width 3 and showing that the density of codewords in these strips is at least 1/3.

Theorem 11. *If $G = (V,E)$ is the king grid and $C \subseteq V$ is a solid-locating-dominating code in G, then the density $D(C) \geq \frac{1}{3}$.*

Proof. Let S^j be a subgraph of G induced by the vertex set $V_j' = \{(x,y) \mid 1 \leq x \leq 3, 1 \leq y \leq j\}$. Recall first the definition $V_n = \{(x,y) \mid |x| \leq n, |y| \leq n\}$. Observe now that we may fit into the first quadrant $\{(x,y) \mid 1 \leq x \leq n, 1 \leq y \leq n\}$ of V_n $\lfloor n/3 \rfloor$ graphs isomorphic to S^n. Similarly, the other three quadrants of V_n can each contain $\lfloor n/3 \rfloor$ graphs isomorphic to S^n. Thus, in total, $4\lfloor n/3 \rfloor$ graphs isomorphic to S_n can be fitted into V_n.

Let C be a solid-locating-dominating code in G. In the final part of the proof, we show that any subgraph of G isomorphic to S^n contains at least $n-3$ codewords. Assuming this is the case, the density of C can be estimated as follows:

$$D(C) = \limsup_{n\to\infty} \frac{|C \cap V_n|}{|V_n|} \geq \limsup_{n\to\infty} \frac{4\lfloor \frac{n}{3} \rfloor \cdot (n-3)}{(2n+1)^2} \geq \limsup_{n\to\infty} \frac{4(n-3)^2}{3(2n+1)^2} = \frac{1}{3}.$$

It remains to be shown that any subgraph of G isomorphic to S^n contains at least $n-3$ codewords. By symmetry, it is enough to show that $|C \cap V'_n| \geq n-3$. In what follows, we consider more closely the number of codewords in a row $S_i = \{(j,i) \mid 1 \leq j \leq 3\}$ of V'_n. For this purpose, the following set of rules for rearranging the codewords inside V'_n are introduced:

Rule 1.1: If $S_i \cap C = \emptyset$, $1 \leq i \leq n-1$ and $\{(1,i+1),(3,i+1)\} \subseteq C$, then one codeword is moved from S_{i+1} to S_i. The rule is illustrated in Figure 3.

Rule 1.2: If $S_i \cap C = \emptyset$, $2 \leq i$ and $\{(1,i-1),(3,i-1)\} \subseteq C$, then one codeword is moved from S_{i-1} to S_i. The rule can be viewed as a reflected version of Rule 1.1.

Rule 2.1: If $S_i \cap C = \emptyset$, $2 \leq i$ and $\{(1,i-1),(2,i-1)\} = C \cap S_{i-1}$, then one codeword is moved from S_{i-1} to S_i. The rule is illustrated in Figure 4.

Rule 2.2: If $S_i \cap C = \emptyset$, $2 \leq i$ and $\{(2,i-1),(3,i-1)\} = C \cap S_{i-1}$, then one codeword is moved from S_{i-1} to S_i. The rule can be viewed as a reflected version of Rule 2.1.

Rule 3.1: If $S_i \cap C = \emptyset$, $3 \leq i$, $S_{i-1} \cap C = \{(1,i-1)\}$ and $\{(2,i-2),(3,i-2)\} \subseteq S_{i-2} \cap C$, then one codeword is moved from S_{i-2} to S_i. The rule is illustrated in Figure 5.

Rule 3.2: If $S_i \cap C = \emptyset$, $3 \leq i$, $S_{i-1} \cap C = \{(3,i-1)\}$ and $\{(2,i-2),(1,i-2)\} \subseteq S_{i-2} \cap C$, then one codeword is moved from S_{i-2} to S_i. The rule can be viewed as a reflected version of Rule 3.1.

Rule 4.1: If $S_i \cap C = \emptyset$, $3 \leq i$, $S_{i-1} \cap C = \{(1,i-1)\}$ and $\{(1,i-2),(2,i-2)\} = S_{i-2} \cap C$, then one codeword is moved from S_{i-2} to S_i. The rule is illustrated in Figure 6.

Rule 4.2: If $S_i \cap C = \emptyset$, $3 \leq i$, $S_{i-1} \cap C = \{(3,i-1)\}$ and $\{(2,i-2),(3,i-2)\} = S_{i-2} \cap C$, then one codeword is moved from S_{i-2} to S_i. The rule can be viewed as a reflected version of Rule 4.1.

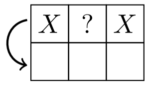


Figure 3:
Rule 1.1

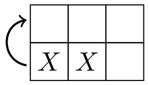


Figure 4:
Rule 2.1

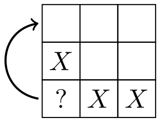


Figure 5:
Rule 3.1

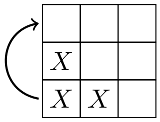


Figure 6:
Rule 4.1

Denote the code obtained after simultaneously applying the previous rules by C'. Notice that the rearrangement C' of C is not completely determined by the previous rules and that this is not actually needed as in the following we are only interested on the number of codewords in the rows of V'_n. In other words, when a codeword is moved from a row we can choose any of the codewords and move it to replace any non-codeword of the target row. In what follows, we show that each row which has given away codewords still contains at least one and each row which originally did not contain any codeword has received at least one except possibly the rows S_1, S_2 and S_n.

We immediately notice that the rules move codewords only from the rows with at least two codewords. Each type of row with at least two codewords is examined as follows:

- $C \cap S_i = \{(j,i) \mid 1 \leq j \leq 3\}$: Rules $1.1, 1.2, 3.1$ and 3.2 can be applied on rows with three codewords. Among these, Rules 3.1 and 3.2 cannot be applied at the same time and Rule 1.2 cannot be applied together with Rules 3.1 or 3.2. Hence, we apply at most two rules on a row with three codewords and that row has at least one codeword left in the code C'.

- $C \cap S_i = \{(j,i) \mid 1 \leq j \leq 2\}$: Rules $2.1, 3.2$ and 4.1 can be applied on this types of rows. We cannot apply Rule 2.1 at the same time as 3.2 or 4.1 since 2.1 requires that $S_{i+1} \cap C = \emptyset$ and Rules 3.2 and 4.1 require that $|S_{i+1} \cap C| = 1$. Furthermore, we cannot apply Rules 3.2 and 4.1 at the same time since they require the codeword on the row S_{i+1} to locate at different places. Hence, C' is left with at least one codeword.

- $C \cap S_i = \{(j,i) \mid 2 \leq j \leq 3\}$: This case is symmetrical to the previous one (now the rules to be considered are 2.2, 3.1 and 4.2).

- $C \cap S_i = \{(j,i) \mid j \neq 2\}$: We can only apply Rules 1.1 and 1.2 on these types of rows and both of them only when $i \geq 2$. However, if both of the rules are used, then $C \cap S_{i-1} = C \cap S_{i+1} = \emptyset$ and a contradiction with Lemma 10 follows. Hence, at most one rule is used and $|C' \cap S_i| \geq 1$.

Let us then show that we have $|C' \cap S_i| \geq 1$ for each i such that $C \cap S_i = \emptyset$ and $3 \leq i \leq n-1$. In the following cases, we assume that $S_i \cap C = \emptyset$ and the cases are categorized by considering the different formations of the row S_{i-1}.

- $S_{i-1} \cap C = \emptyset$: Considering different orientations and positions of the formation T in Lemma 10, we have $S_{i+1} \subseteq C$. Hence, due to Rule 1.1, one codeword from S_{i+1} is moved to S_i and we obtain $|C' \cap S_i| \geq 1$.

- $S_{i-1} \cap C = \{(1, i-1)\}$: By Lemma 10, we have $(2, i-2) \in C$. Notice that if $(1, i-2)$ and $(3, i-2)$ do not belong to C, then a contradiction with the definition of solid-locating-dominating codes follows since we have $I(2, i-1) \subseteq I(1, i-2)$ for non-codewords $(2, i-1)$ and $(1, i-2)$. Hence, at least one of the vertices $(1, i-2)$ and $(3, i-2)$ belongs to C. Therefore, either Rule 3.1 or 4.1 can be applied (to the row S_{i-2}) and we have $|C' \cap S_i| \geq 1$.

- $S_{i-1} \cap C = \{(3, i-1)\}$: This case is symmetrical to the previous one. Here we just use either Rule 3.2 or 4.2.

- $S_{i-1} \cap C = \{(2, i-1)\}$: By Lemma 10, we have $\{(1, i+1), (3, i+1)\} \subseteq C$. Hence, due to Rule 1.1, we have $|C' \cap S_i| \geq 1$.

- $S_{i-1} \cap C = \{(1, i-1), (2, i-1)\}$: Due to Rule 2.1, we have $|C' \cap S_i| \geq 1$.

- $S_{i-1} \cap C = \{(2, i-1), (3, i-1)\}$: Due to Rule 2.2, we have $|C' \cap S_i| \geq 1$.

- $S_{i-1} \cap C = \{(1, i-1), (3, i-1)\}$: Due to Rule 1.2, we have $|C' \cap S_i| \geq 1$.

- $S_{i-1} \cap C = \{(1, i-1), (2, i-1), (3, i-1)\}$: Due to Rule 1.2, we have $|C' \cap S_i| \geq 1$.

Thus, in conclusion, we have shown that for $3 \leq i \leq n-1$ we have $|C' \cap S_i| \geq 1$. Therefore, as the rules rearrange codewords only inside V_n', we have $|C \cap V_n'| \geq |C' \cap V_n'| \geq n-3$. This concludes the proof of the lower bound $D(C) \geq 1/3$. □

In the previous theorems, we have shown that the density of an optimal solid-locating-dominating code in the king grid is 1/3. Recall that a self-locating-dominating code is always solid-locating-dominating. Hence, by the previous lower bound, we also know that there exists no self-locating-dominating code in the king grid with density smaller than 1/3. However, the construction given for the solid-location-domination does not work for self-location-domination. For example, we have $I(2,0) = \{(2,-1), (2,1), (3,0)\}$ and $N[(2,-1)] \cap N[(2,1)] \cap N[(3,0)] = \{(2,0), (3,0)\}$ contradicting with the definition of self-locating-dominating codes (see Figure 2). In the following theorem, we present a self-locating-dominating code in the king grid with the density 1/3. Notice that this code is also solid-locating-dominating.

Theorem 12. *Let $G = (V, E)$ be the king grid. The code*

$$C = \{(x, y) \in \mathbb{Z}^2 \mid x - y \equiv 0 \pmod 3\}$$

is self-locating-dominating in G and its density is $1/3$.

Proof. The density $D(C) = 1/3$ since in each row every third vertex is a codeword. Furthermore, C is a self-locating-dominating code since each non-codeword v is covered either by the set of three codewords $\{v + (1, 0), v + (0, -1), v + (-1, 1)\}$ or $\{v + (-1, 0), v + (0, 1), v + (1, -1)\}$, and in both cases the closed neighbourhoods of the codewords intersect uniquely in the vertex v. $\square$

3 Direct product of complete graphs

A graph is called a *complete graph* on q vertices, denoted by K_q, if each pair of vertices of the graph is adjacent. The vertex set $V(K_q)$ is denoted by $\{1, 2, \dots, q\}$. The *Cartesian product* of two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ is defined as $G_1 \Box G_2 = (V_1 \times V_2, E)$, where E is a set of edges such that $(u_1, u_2)(v_1, v_2) \in E$ if and only if $u_1 = v_1$ and $u_2v_2 \in E_2$, or $u_2 = v_2$ and $u_1v_1 \in E_1$. The *direct product* of two graphs G_1 and G_2 is defined as $G_1 \times G_2 = (V_1 \times V_2, E)$, where $E = \{(u_1, u_2)(v_v, v_2) \mid u_1v_1 \in E_1 \text{ and } u_2v_2 \in E_2\}$. A *complement* of a graph $G = (V, E)$ is the graph $\overline{G} = (V, E')$ with the edge set E' being such that $uv \in E'$ if and only if $uv \notin E$.

In this section, we give optimal locating-dominating, self-locating-dominating and solid-locating-dominating codes in the direct product $K_n \times K_m$, where $2 \leq n \leq m$. For location-domination and solid-location-domination, the results heavily depend on the exact values of $\gamma^{LD}(K_n \Box K_m)$ and $\gamma^{DLD}(K_n \Box K_m)$, which have been determined in [6]. In the graphs $K_n \times K_m$ and $K_n \Box K_m$, the jth *row* (of $V(K_n) \times V(K_m)$) is denoted by R_j and it consists of the vertices $(1, j), (2, j), \dots, (n, j)$. Analogously, the ith *column* is denoted by P_i and it consists of the vertices $(i, 1), (i, 2), \dots, (i, m)$. Now we are ready to present the following observations:

- In the Cartesian product $K_n \Box K_m$, the closed neighbourhood $N[(i, j)] = N[i, j]$ consists of the row R_j and the column P_i. Therefore, as the closed neighbourhood of a vertex resembles the movements of a rook in a chessboard, $K_n \Box K_m$ is also sometimes called the *rook's graph.*
- In the direct product $K_n \times K_m$, we have $N((i, j)) = N(i, j) = V(K_n \Box K_m) \setminus (R_j \cup P_i)$.

Due to the previous observations, we know that $\overline{K_n \Box K_m} = K_n \times K_m$.

Recall that identification is a topic closely related to the various location-domination type problems. Previously, in [13], the identifying codes have been studied in the direct product $K_n \times K_m$ of complete graphs by Goddard and Wash. More precisely, they determined the exact values of $\gamma^{ID}(K_n \times K_m)$ for all m and n.

In what follows, we determine the exact values of $\gamma^{LD}(K_n \times K_m)$ for all m and n. For this purpose, we first present the following result concerning location-domination in the Cartesian product $K_n \Box K_m$ of complete graphs given in [6].

Theorem 13 ([6], Theorem 14)**.** *Let m and n be integers such that $2 \leq n \leq m$. Now we have*

$$\gamma^{LD}(K_n \Box K_m) = \begin{cases} m - 1, & 2n \leq m, \\ \left\lceil \frac{2n+2m}{3} \right\rceil - 1, & n \leq m \leq 2n - 1. \end{cases}$$

There is a strong connection between the values of $\gamma^{LD}(K_n \Box K_m)$ and $\gamma^{LD}(K_n \times K_m)$ as explained in the following. In [4], it has been shown that $|\gamma^{LD}(G) - \gamma^{LD}(\overline{G})| \leq 1$. Therefore, as $\overline{K_n \times K_m} = K_n \Box K_m$, we obtain that $\gamma^{LD}(K_n \Box K_m) - 1 \leq \gamma^{LD}(K_n \times K_m) \leq \gamma^{LD}(K_n \Box K_m) + 1$. This result is further sharpened in the following lemma.

Lemma 14. *For $2 \leq n \leq m$ and $(n, m) \neq (2, 4)$, we have*

$$\gamma^{LD}(K_n \Box K_m) - 1 \leq \gamma^{LD}(K_n \times K_m) \leq \gamma^{LD}(K_n \Box K_m).$$

If $\gamma^{LD}(K_n \times K_m) = \gamma^{LD}(K_n \Box K_m) - 1$, then the optimal locating-dominating code C in $K_n \times K_m$ has a non-codeword v such that $I(v) = C$.

Proof. First denote $G = K_n \square K_m$ and $H = K_n \times K_m$. The lower bound of the claim is immediate by the result preceding the lemma. For the upper bound, let C be an optimal locating-dominating code in G. The code C can also be viewed as a code in H. If we have $I(H;u) = I(H;v)$ for some non-codewords u and v, then a contradiction follows since $I(G;u) = C \setminus I(H;u) = C \setminus I(H;v) = I(G;v)$. Hence, we have $I(H;u) \neq I(H;v)$ for all distinct non-codewords u and v. Moreover, if $I(G;v) \neq C$ for each non-codeword v, then we also have $I(H;v) \neq \emptyset$, and the upper bound follows since C is a locating-dominating code in H.

Hence, we may assume that $I(G;v) = C$ for some non-codeword v. This implies that $C \subseteq P_i \cup R_j$ for some i, j. There exists at most one non-codeword in $P_i \setminus \{v\}$ since otherwise there are at least two non-codewords with the same I-set. Similarly, there exists at most one non-codeword in $R_j \setminus \{v\}$. Furthermore, if both $P_i \setminus \{v\}$ and $R_j \setminus \{v\}$ contain a non-codeword, then there exists a vertex with an empty I-set. Thus, in conclusion, there exists at most two non-codewords in $P_i \cup R_j$ and, hence, we have $|C| \geq n + m - 3$. Dividing into the following cases depending on n and m, we next show that $|C| \geq n + m - 3 > \gamma^{LD}(G)$ in majority of the cases of the lemma:

- If $n \geq 3$ and $m \geq 2n$, then we have $\gamma^{LD}(G) = m - 1 < n + m - 3 \leq |C|$ (by Theorem 13).
- If $n \geq 4$, $n \leq m \leq 2n-1$ and $(n, m) \neq (4, 4)$, then $\gamma^{LD}(G) = \lceil 2(n+m)/3 \rceil - 1 < n+m-3 \leq |C|$ (by Theorem 13).

Thus, if $n \geq 3$ and $m \geq 2n$, or $n \geq 4$, $n \leq m \leq 2n - 1$ and $(n, m) \neq (4, 4)$, then a contradiction with the optimality of C follows. Hence, in these cases, we have $\gamma^{LD}(H) \leq \gamma^{LD}(G)$.

The rest of the cases are covered in the following:

- If $n = 2$ and $2 \leq m \leq 3$, then $C = P_1$ is an optimal locating-dominating code in G with the property that for any non-codeword v we have $I(G;v) \neq C$. Similarly, if $n = 2$ and $m \geq 5$, then $C = \{(2,1),(2,2)\} \cup P_1 \setminus \{(1,i) \mid i \leq 3\}$ is an optimal locating-dominating code in G with the property that for no vertex v we have $I(G;v) = C$. Thus, in both cases, the code C is also locating-dominating in H by the first paragraph of the proof.
- If $n = m = 3$, then $C = \{(1,1),(1,2),(2,1)\}$ is a locating-dominating code in H with $\gamma^{LD}(G) = 3$ codewords.
- If $n = 3$ and $4 \leq m \leq 5$ or $(n,m) = (4,4)$, then $\{(1,1),(1,3),(2,2),(2,4)\}$, $\{(1,1),(1,3),(2,2),(2,4),(3,5)\}$ and $\{(1,1),(1,3),(2,2),(2,4),(3,1)\}$ obtained from the proof of [6, Theorem 14] are optimal locating-dominating codes in $K_3 \square K_4$, $K_3 \square K_5$ and $K_4 \square K_4$, respectively. Therefore, since there does not exist a non-codeword covering all the codewords (in the Cartesian product) in any of the cases, the codes are also locating-dominating in $K_3 \times K_4$, $K_3 \times K_5$ and $K_4 \times K_4$ (by the first paragraph of the proof), respectively.

Let then C' be a locating-dominating code in H. Similarly as above, we get that if $I(H;v) \neq C'$ for each non-codeword v, then C' is also a locating-dominating code in G. Therefore, if $\gamma^{LD}(H) = \gamma^{LD}(G) - 1$, then there exist a non-codeword v such that $I(H;v) = C'$. Thus, the last claim of the lemma follows. □

Now with the help of the previous lemma and Theorem 13, we determine the exact values of $\gamma^{LD}(K_m \times K_n)$ in the following theorem.

Theorem 15. *For $2 \leq n \leq m$ we have*

$$\gamma^{LD}(K_n \times K_m) = \begin{cases} m-1, & 2n \leq m \text{ and } (n,m) \neq (2,4), \\ \left\lceil \frac{2n+2m-1}{3} \right\rceil - 1, & 2 < n \leq m < 2n \text{ and } (m,n) \neq (4,4) \\ m, & n = 2, m \leq 4, \\ 5, & n = 4, m = 4. \end{cases}$$

			■	B_1	B_1				
			B_1	■	B_1				
			B_1	B_1	■				
			B_1	B_1	■				
			B_1	■	B_1				
			■	B_1	B_1				
B_2	B_2	■	B_1B_2	B_1B_2	B_1B_2	B_2	B_2	■	
B_2	■	B_2	B_1B_2	B_1B_2	B_1B_2	B_2	■	B_2	
■	B_2	B_2	B_1B_2	B_1B_2	B_1B_2	■	B_2	B_2	

Figure 7: Optimal locating-dominating code for $K_{10} \times K_{10}$. Dark boxes are codewords.

Proof. Let C be a locating-dominating code in $K_n \times K_m$. We cannot have $R_i \cap C = R_j \cap C = \emptyset$ for $i \neq j$ since otherwise, for example, $I(C;(1,i)) = I(C;(1,j))$. Similarly, there exists at most one column without codewords of C. Thus, we have $\gamma^{LD}(K_n \times K_m) \geq m-1$. Therefore, if $m \geq 2n$ and $(n,m) \neq (2,4)$, then by the previous lemma we have $m-1 \leq \gamma^{LD}(K_n \times K_m) \leq \gamma^{LD}(K_n \square K_m) = m-1$, i.e., $\gamma^{LD}(K_n \times K_m) = m-1$.

Assume then that $2 < n \leq m \leq 2n-1$ and $n+m \equiv 0,1 \pmod 3$. In what follows, we show that now $|C| \geq \gamma^{LD}(K_n \square K_m)$. By the previous lemma, we know that if there is no non-codeword u such that $I(K_n \times K_m, C; u) = C$, i.e., there does not exist a row and column without codewords, then $|C| = \gamma^{LD}(K_n \square K_m)$. Hence, we may now assume that there exist a row and a column without codewords. Without loss of generality, we may assume that they are P_n and R_m. Observe that C can now also be viewed as a code in $K_{n-1} \square K_{m-1}$ and that C is locating-dominating in $K_{n-1} \square K_{m-1}$ with the following additional properties: (i) each column has at least one codeword, (ii) each row has at least one codeword and (iii) no codeword $(i,j) \in C$ is such that $(P_i \cup R_j) \cap C = \{(i,j)\}$, i.e., no codeword of C is isolated. Indeed, the properties (i) and (ii) follow immediately by the first paragraph of the proof and if $(i,j) \in C$ is a codeword violating the property (iii), then we have $I(K_n \times K_m, (n,j)) = I(K_n \times K_m; (i,m)) = C \setminus \{(i,j)\}$ (a contradiction). Now we are ready to prove a lower bound on $|C|$ as in [6, Theorem 14]. Denote the number of columns and rows with exactly one codeword in $K_n \square K_m$ by s_p and s_r, respectively. Now we obtain that $|C| \geq s_p + 2(n-1-s_p) = 2(n-1) - s_p$ and $|C| \geq s_r + 2(m-1-s_r) = 2(m-1) - s_r$ (by the properties (i) and (ii)). This further implies that $s_p \geq 2(n-1) - |C|$ and $s_r \geq 2(m-1) - |C|$. By the property (iii), we now obtain that $|C| \geq s_p + s_r \geq 2(n-1) + 2(m-1) - 2|C|$. Thus, we have $|C| \geq \lceil (2m+2n-1)/3 \rceil - 1$. Hence, as $n+m \equiv 0,1 \pmod 3$, we have $|C| \geq \lceil (2m+2n-1)/3 \rceil - 1 = \lceil (2m+2n)/3 \rceil - 1 = \gamma^{LD}(K_n \square K_m)$. Thus, by the upper bound of the previous lemma, we obtain that $\gamma^{LD}(K_n \times K_m) = \gamma^{LD}(K_n \square K_m)$ if $2 < n \leq m \leq 2n-1$ and $n+m \equiv 0,1 \pmod 3$.

Assume then that $2 < n \leq m \leq 2n-1$, $n+m \equiv 2 \pmod 3$ and $(n,m) \neq (4,4)$. In what follows, we show that the lower bound of Lemma 14 is attained, i.e., $\gamma^{LD}(K_n \times K_m) = \gamma^{LD}(K_n \square K_m) - 1$. Denote $n' = n-1$ and $m' = m-1$ and observe that $n' + m'$ is divisible by three. Let $C' =$

$A_1 \cup A_2 \cup A_3$ be a code in $K_n \times K_m$ with

$$
\begin{aligned}
A_1 &= \left\{ (i,i) \mid 1 \le i \le \frac{n'+m'}{3} \right\}, \\
A_2 &= \left\{ (j,i) \mid \frac{n'+m'}{3} + 1 \le i \le m', j = 2\frac{n'+m'}{3} + 1 - i \right\} \text{ and} \\
A_3 &= \left\{ (j,i) \mid 1 \le i \le \frac{2n'-m'}{3}, j = i + \frac{n'+m'}{3} \right\}.
\end{aligned}
$$

The code C' is illustrated in Figure 7. By straightforward counting , we get $|C'| = |A_1| + |A_2| + |A_3| = m' + \frac{2n'-m'}{3} = \frac{2n'+2m'}{3} = \frac{2n+2m-1}{3} - 1 = \gamma^{LD}(K_n \Box K_m) - 1$. In what follows, we first show that C' is almost a locating-dominating code in $K_n \Box K_m$ with the exception that $I(C'; (n,m)) = \emptyset$.

Denote the sets of non-codewords (j,i) with $(2n'+2m')/3 - m' + 1 \le j \le (n'+m')/3$ and $i \le (2n'+2m')/3 - m'$ by B_1 and B_2, respectively. It is straightforward to verify that each non-codeword $u \in B_1 \cup B_2$ has at least three codewords in $I(K_n \Box K_m, C'; u)$ and the codewords of $I(K_n \Box K_m, C'; u)$ do not lie on a single row or column. This implies that $\bigcap_{c \in I(C';u)} N[c] = \{u\}$ for any $u \in B_1 \cup B_2$, i.e., there is no other vertex containing $I(C'; u)$ in its I-set. Thus, each non-codeword in $B_1 \cup B_2$ has a unique nonempty I-set. Consider then a non-codeword $v = (j,i)$ with $i > (2n'+2m')/3 - m'$ and $j < (2n'+2m')/3 - m + 1'$. By the construction of C', we have $|I(C'; v)| = 2$. Now there exists a codeword $(j,j) \in I(v)$ since $j \le (2n'+2m')/3 - m'$. Furthermore, there exists a codeword $c \in I(j,j) \cap A_3$. Hence, if there exists a non-codeword w such that $I(C'; v) = I(C'; w)$, then $w \in B_2$ and a contradiction follows as $|I(C'; w)| \ge 3$. Thus, the I-set of v is nonempty and unique. Similarly, it can be shown that $I(C'; (j,i))$ is nonempty and unique for $i > (2n'+2m')/3 - m'$ and $j > (n'+m')/3$.

Consider then non-codewords $u = (j,m)$ and $v = (n,i)$ with $1 \le j \le n-1$ and $1 \le i \le m-1$. We immediately obtain that $I(C'; (j,m)) = P_j \cap C'$ and $I(C'; (n,i)) = R_i \cap C'$. These I-sets are nonempty since each row and column contains a codeword. These I-sets are also different from the ones of non-codewords inside $K_{n'} \Box K_{m'}$ which contain at least two codewords in different rows and columns. It is also impossible to have $I(C'; u) = I(C'; v)$ since each codeword has another one in the same row or column. Thus, u and v have nonempty and unique I-sets. Thus, in conclusion, we have shown that $I(C'; u)$ is nonempty and unique for all non-codewords u in $K_n \Box K_m$ except (n,m) (for which we have $I(C'; (n,m)) = \emptyset$). Furthermore, there does not exist a non-codeword v such that $I(C'; v) = C'$. Therefore, as in the proof of Lemma 14, we obtain that C' is a locating-dominating code in $K_n \times K_m$. Thus, we have $\gamma^{LD}(K_n \times K_m) = \gamma^{LD}(K_n \Box K_m) - 1$.

Now majority of the cases have been considered, and we only have some special cases left. Concluding the proof, these cases are solved as follows:

- Assume that $n = 2$ and $m \le 4$. It is easy to see that $C = P_1$ is a locating-dominating code in $K_n \times K_m$. For the lower bound, first recall that $K_n \times K_m$ has at most one row without codewords (by the first paragraph of the proof). Therefore, if C is a locating-dominating code in $K_n \times K_m$ with $|C| \le m - 1$, then all the codewords lie on different rows. Hence, in all the cases, there exist a non-codeword with an empty I-set. Thus, we have $\gamma^{LD}(K_n \times K_m) = m$.

- Assume that $n = m = 4$. By Lemma 14, we immediately have $4 \le \gamma^{LD}(K_4 \times K_4) \le 5$. Let C be a locating-dominating code in $K_4 \times K_4$. As in the second paragraph of the proof, it can be shown that either $|C| \ge \gamma^{LD}(K_4 \Box K_4) = 5$ (and we are done), or C is locating-dominating in $K_3 \times K_3$ with the additional properties (i), (ii) and (iii). In the latter case, due to (i), (ii) and (iii), there exist a row and a column of $K_3 \times K_3$ with two codewords such that their intersection is a non-codeword u. Hence, a contradiction follows since $I(K_4 \times K_4, C; u) = \emptyset$. Thus, we have $\gamma^{LD}(K_4 \times K_4) = 5$.

□

Let us next briefly consider solid-location-domination. The following result has been shown in [6].

Theorem 16 ([6])**.** *For all integers m and n such that $m \geq n \geq 1$, we have*

$$\gamma^{DLD}(K_n \Box K_m) = \begin{cases} m, & 4 \leq 2n \leq m \text{ or } n = 2, \\ 2n, & 2 < n < m < 2n, \\ 2n-1, & 2 < m = n. \end{cases}$$

In the following theorem, we show that the cardinalities of optimal solid-locating-dominating codes are same for $K_n \times K_m$ and $K_n \Box K_m$.

Theorem 17. *For all integers m and n such that $m \geq n \geq 2$, we have*

$$\gamma^{DLD}(K_n \times K_m) = \gamma^{DLD}(K_n \Box K_m).$$

Proof. By [8, Theorem 21], we have $\gamma^{DLD}(G) = \gamma^{DLD}(\overline{G})$ if G is not a discrete or a complete graph. Therefore, as this is the case for $G = K_n \times K_m$, we have $\gamma^{DLD}(K_n \times K_m) = \gamma^{DLD}(\overline{K_n \times K_m}) = \gamma^{DLD}(K_n \Box K_m)$. $\square$

Let us then consider self-location-domination. Unlike location-domination [4, Theorem 7] and solid-location-domination [8, Theorem 21], the optimal cardinality of a self-locating-dominating code in G does not depend on the one of the complement graph $\overline{G}$. In the following theorem, we first give the result presented in [6] regarding $\gamma^{SLD}(K_n \Box K_m)$.

Theorem 18 ([6])**.** *For all integers m and n such that $m \geq n \geq 2$, we have*

$$\gamma^{SLD}(K_n \Box K_m) = \begin{cases} m, & 2n \leq m, \\ 2n, & 2 \leq n < m < 2n, \\ 2n-1, & 2 < m = n, \\ 4, & n = m = 2. \end{cases}$$

In the following theorem, we determine the exact values of $\gamma^{SLD}(K_n \times K_m)$ for all values of m and n. Notice that $\gamma^{SLD}(K_n \Box K_m) = \gamma^{SLD}(K_n \times K_m)$ if and only if $n = m$, $m = n + 1 > 3$, or $n = 2$ and $m \geq 4$.

Theorem 19. *For all integers m and n such that $m \geq n \geq 2$, we have*

$$\gamma^{SLD}(K_n \times K_m) = \begin{cases} m + n - 1, & n > 2, \\ m, & n = 2, m > 2, \\ 4, & n = m = 2. \end{cases}$$

Proof. Let C be a self-locating-dominating code in $K_n \times K_m$. Notice first that if $n = m = 2$, then $K_2 \times K_2$ is isomorphic to a forest of two paths of length two and, therefore, $\gamma^{SLD}(K_2 \times K_2) = 4$. Hence, we may assume that $(n, m) \neq (2, 2)$. Observe then that if a column P_i contains no codewords, i.e., $P_i \cap C = \emptyset$, then $C = V \setminus P_i$. Indeed, for any vertices $(i, j) \in P_i$ and $(h, j) \in V$ with $i \neq h$, we have $I(h, j) \subseteq I(i, j)$ and the claim $C = V \setminus P_i$ follows by Theorem 4. Analogously, it can be shown that if $R_i \cap C = \emptyset$, then $C = V \setminus R_i$. Suppose now that $n = 2$ and $m > 2$. If each row contains a codeword, then we immediately have $|C| \geq m$. Otherwise, there exists a row without codewords and, by the previous observation, we have $|C| \geq 2m - 2 \geq m$. Hence, we obtain that $|C| \geq m$. Furthermore, P_1 is a self-locating-dominating code in $K_2 \times K_m$ with m codewords. Thus, in conclusion, we have $\gamma^{SLD}(K_2 \times K_m) = m$.

Assume that $n > 2$. By the previous observations, we know that if there exists a row or a column without codewords, then $|C| \geq \min\{mn - m, mn - n\} = mn - m \geq m + n - 1$. Hence, we may assume that each row and column contains a codeword of C. Furthermore, if each row contains at least 2 codewords, then $|C| \geq 2m \geq m + n - 1$. Hence, we may assume that there exists a row R_i with exactly one codeword, i.e., $R_i \cap C = \{(j, i)\}$ for some j. Hence, as $I(j, h) \subseteq I(j, i)$ for any $h \neq i$, we have $P_j \subseteq C$. Therefore, as each column different from P_j also contains a codeword, we obtain that $|C| \geq m+n-1$. Thus, we have $\gamma^{SLD}(K_n \times K_m) \geq m+n-1$. Finally, this lower bound can be attained with a code $C' = \{(i,j) \mid i = 1 \text{ or } j = 1\}$. Indeed, for any $i, j > 1$, we have $I(1,1) = \{(1,1)\}$, $I(1,j) = \{(1,j)\} \cup (R_1 \setminus \{(1,1)\})$, $I(j,1) = \{(j,1)\} \cup (P_1 \setminus \{(1,1)\})$ and $I(i,j) = C' \setminus \{(1,j),(i,1)\}$. Therefore, we have $I(v) \not\subseteq I(u)$ for any vertex u and non-codeword v. Thus, by Theorem 4, C' is a self-locating-dominating code in $K_n \times K_m$, and we have $\gamma^{SLD}(K_n \times K_m) = n+m-1$. □

4 On certain type of Hamming graphs

The Cartesian product $K_q \square K_q \square \cdots \square K_q$ of n copies of K_q is denoted by K_q^n and called a Hamming graph. Goddard and Wash [2] studied identification in the case of K_q^n and they, in particular, bounded the cardinality of an optimal identifying code to $q^2 - q\sqrt{q} \leq \gamma^{ID}(K_q^3) \leq q^2$. In [7], we further improved this bound to $q^2 - \frac{3}{2}q \leq \gamma^{ID}(K_q^3) \leq q^2 - 4^{t-1}$ where $2 \cdot 4^t \leq q \leq 2 \cdot 4^{t+1} - 1$ or $q = 4^t$, and we also showed that $\gamma^{SLD}(K_q^3) = q^2$. In this section, we show that also $\gamma^{DLD}(K_q^3) = q^2$.

The following lemma is presented as Exercise 1.12 in [3].

Lemma 20 ([3]). *For each positive integer q, we have*

$$\gamma(K_q \square K_q) = q.$$

In the following we present some terminology and notations we use. More information about them and their usefulness can be found in [7].

- The *pipe* $P^i(a,b) \subseteq V(K_q^3)$ is a set of vertices fixing all but the ith coordinate which varies between 1 and q. The fixed coordinates are a and b where a is the value of left fixed coordinate in the representation (x,y,z). For example $P^3(a,b) = \{(a,b,i) \mid 1 \leq i \leq q\}$.
- The *layer* $L_j^i \subseteq V(K_q^3)$ is a set of vertices fixing the ith coordinate as j. For example, the layer L_j^1 consists of pipes $P^i(1,j)$ for $i = 1,2$ and $1 \leq j \leq q$.
- $C_j^i \subseteq L_j^i$ denotes the set of codewords in layer L_j^i, that is, for code $C \subseteq V(K_q^3)$ we have $C_j^i = C \cap L_j^i$.
- $X_j^i \subseteq L_j^i$ denotes such non-codewords v in L_j^i that $I(C_j^i; v) = \emptyset$ and $X^i = \bigcup_{j=1}^q X_j^i$.
- Let us denote $a_j^i = q - |C_j^i|$.
- $M_j^i \subseteq L_j^i$ denotes the minimum dominating set of induced subgraph $K_q^3[L_j^i]$ such that $C_j^i \subseteq M_j^i$. Note that $K_q^3[L_j^i] \simeq K_q \square K_q$ and hence, $|M_j^i| \geq q$.
- Let us denote $f_j^i = |M_j^i| - q$. Note that $|X_j^i| \geq (f_j^i + a_j^i)^2$ and $f_j^i + a_j^i \geq 0$ since $f_j^i = |M_j^i| - q \geq |C_j^i| - q = -a_j^i$, ([7, page 11 and 13]).

Lemma 21. *Let $C \subseteq V(K_q^3)$ and let $K_t \square K_t$ be a subgraph of $K_q^3[C_j^i]$ for some i, j. Then we have $f_j^i \geq t^2 - t$.*

Proof. We have $C_j^i \subseteq M_j^i$. Besides the vertices of C_j^i inducing graph $K_t \square K_t$, there are $(q-t)^2$ vertices which are not dominated by these vertices. Moreover, we require at least $q-t$ vertices to dominate them. Hence, we have $|M_j^i| \geq t^2 + (q-t)$ and thus, $f_j^i \geq t^2 - t$. □

Lemma 22 ([7], Lemma 10). *Let C be a code in K_q^3 and v be a vertex of K_q^3.*

- *If a vertex v has two codewords in its I-set and they do not locate within a single pipe, then there is exactly one other vertex which has those two codewords in its I-set.*
- *The I-set $I(v)$ is not a subset of any other I-set if and only if there are at least three codewords in $I(v)$ and they do not locate within a single pipe.*

Theorem 23. *We have for $q \geq 2$*

$$\gamma^{DLD}(K_q^3) = q^2.$$

Proof. We have shown in [7] that $\gamma^{SLD}(K_q^3) = q^2$. Hence, we have $\gamma^{DLD}(K_q^3) \leq q^2$ by Corollary 5. Let us assume that C is an optimal solid-locating-dominating code in $V(K_q^3)$ with $|C| < q^2$. Since $|C| < q^2$, we have a layer, say L_1^3, with at most $q-1$ codewords and hence, we have $|X_1^3| \geq 1$ by Lemma 20. Let us assume that $(1,1,1) \in X_1^3$. Now, we have $(i,1,1) \notin C$ for any i and the same is true for $(1,j,1)$ for any j. Moreover, if we have $(1,1,h) \notin C$, then $I(1,1,1) \subseteq I(1,1,h)$, a contradiction. Therefore, for each non-codeword in X_j^i we have a pipe with $q-1$ codewords. Let us denote a pipe with $q-1$ codewords as $P_C^i(a,b)$ where i denotes the direction of the pipe and (a,b) denotes the coordinates in which the pipe intersects with the layer. Note that if $(a,b,z) \in X_z^3$ and $(a,b,z') \in X_{z'}^3$, then $z = z'$.

Let us first note that we have

$$|\{P_C^i(a,b) \mid 1 \leq a,b \leq q\}| \leq q+1 \tag{1}$$

for any fixed $i \in \{1,2,3\}$. Otherwise, we would have $|C| \geq (q+2)(q-1) = q^2+q-2 > q^2-1$. Let us then consider the case where we have only $q-t$, $t \geq 2$, codewords in a layer, say L_1^3. Then we have $|X_1^3| \geq t^2$ and these vertices (or some subset of them) induce subgraph $K_t \square K_t$ on K_q^3. Therefore, we have at least t^2 copies of codeword pipes $P_C^3(a,b)$ and without loss of generality, we may assume that values (a,b) form the set $\{(i,j) \mid 1 \leq i,j \leq t\}$. Thus, some subset of the vertices in C_j^3, for any fixed j such that $2 \leq j \leq q$, form an induced subgraph $K_t \square K_t$. Therefore, we have $f_j^3 \geq t^2 - t$ for any $2 \leq j \leq q$ by Lemma 21. Thus, we have

$$|X^3| \geq t^2 + \sum_{j=2}^{q}(f_j^3 + a_j^3)^2 \geq t^2 + \sum_{j=2}^{q} f_j^3 + \sum_{j=2}^{q} a_j^3 \geq t^2 - t + (q-1)(t^2-t) + 1 = q(t^2-t)+1 \geq 2q+1.$$

Note that $\sum_{j=2}^{q} a_j^3 \geq 1-t$ and if $(a,b,j) \in X_j^3$, then $(a,b,i) \notin X_i^3$ for each $i \neq j$. However, this is a contradiction with (1). Therefore, we have $|C_j^i| \geq q-1$ for any i,j.

Let us then consider the case where $|C_1^3| = q-1$ and C_1^3 induces a discrete graph. Then for any non-codeword $v = (a,b,1)$, we have $|N(v) \cap C_1^3| \leq 2$ and the codewords in $N(v) \cap C_1^3$ do not locate within the same pipe. Therefore, by Lemma 22, we have another non-codeword $w \in L_1^3$ such that $N(v) \cap C_1^3 \subseteq N(w)$. Furthermore, this means that there is a codeword in $P^3(a,b)$. Since this is true for any non-codeword and $|L_1^3| = q^2$, we have $|C| \geq q^2$, a contradiction.

Let us then consider the case $|C_1^3| = q-1$ for $q \geq 3$ and assume that some codewords in C_1^3 are neighbours. We may assume that $(1,1,1),(1,2,1) \in C_1^3$. Moreover, we may assume that $(q,q,1) \in X_1^3$. Since there are at least two codewords in the pipe $P^2(1,1)$ and there are $q-1$ codewords in C_1^3, we have at least two pipes $P^2(a,1)$ and $P^2(q,1)$ such that they contain no codewords. Therefore, we have $(a,q,1) \in X_1^1$. Moreover, we have codeword pipes $P_C^3(q,q)$ and $P_C^3(a,q)$. Now, we can consider layers L_q^1 and L_a^1. Let us first consider the layer L_q^1. First of all, it contains the codeword pipe $P_C^3(q,q)$ and since the pipe $P^2(q,1)$ contains no codewords, there has to be at least one codeword in every pipe $P^3(q,i)$ where $1 \leq i \leq q-1$. Indeed, otherwise we would have $q-1$ codewords in some pipe $P_C^1(i,q)$, $2 \leq i \leq q$, a contradiction with pipes $P^2(a,1)$ and $P^2(q,1)$ containing no codewords. Therefore, we have $|C_q^1| \geq 2q-2$. Furthermore, we get similarly $|C_a^1| \geq 2q-2$. However, now we have

$$|C| \geq 2(2q-2) + \sum_{i=1, i\neq a}^{q-1} |C_i^1| \geq 2(2q-2) + (q-2)(q-1) = q^2+q-2 > q^2-1,$$

a contradiction. $\square$

5 Acknowledgement

The authors would like to thank María Luz Puertas Gonzalez for fruitful discussions on the topic.

References

[1] N. Fazlollahi, D. Starobinski, and A. Trachtenberg. Connected identifying codes. *IEEE Trans. Inform. Theory*, 58(7):4814–4824, 2012.

[2] W. Goddard and K. Wash. ID codes in Cartesian products of cliques. *J. Combin. Math. Combin. Comput.*, 85:97–106, 2013.

[3] T. Haynes, S. Hedetniemi and P. Slater. *Fundamentals of Domination in Graphs*, volume 208 of *Pure and Applied Mathematics*. Marcel Dekker, Inc., New York, 1998.

[4] C. Hernando, M. Mora, and I. M. Pelayo. Nordhaus-Gaddum bounds for locating domination. *European J. Combin.*, 36:1–6, 2014.

[5] V. Junnila, T. Laihonen and T. Lehtilä. New results on codes for location in graphs. In *Proceedings of Russian Finnish Symposium on Discrete Mathematics 2019*, pages 105–116, 2019.

[6] V. Junnila, T. Laihonen, and T. Lehtilä. On regular and new types of codes for location-domination. *Discrete Appl. Math.*, 247:225–241, 2018.

[7] V. Junnila, T. Laihonen and T. Lehtilä. On a conjecture regarding identification in Hamming graphs. *Electron. J. Combin.*, 26(1): Research Paper P.2.45, 2019.

[8] V. Junnila, T. Laihonen. T. Lehtilä and M. Puertas. On stronger types of locating-dominating codes. *Discrete Math. Theor. Comput. Sci.*, 21(1): Research Paper 1, 2019.

[9] M. G. Karpovsky, K. Chakrabarty, and L. B. Levitin. On a new class of codes for identifying vertices in graphs. *IEEE Trans. Inform. Theory*, 44(2):599–611, 1998.

[10] M. Laifenfeld and A. Trachtenberg. Disjoint identifying-codes for arbitrary graphs. In *Proceedings of International Symposium on Information Theory, 2005. ISIT 2005*, pages 244–248, 2005.

[11] A. Lobstein. Watching systems, identifying, locating-dominating and discriminating codes in graphs, a bibliography. Published electronically at `https://www.lri.fr/∼lobstein/debutBIBidetlocdom.pdf`.

[12] D. F. Rall and P. J. Slater. On location-domination numbers for certain classes of graphs. *Congr. Numer.*, 45:97–106, 1984.

[13] D. Rall and K. Wash. Identifying codes of the direct product of two cliques *European J. Combin.*, 36:159–171, 2014.

[14] S. Ray, D. Starobinski, A. Trachtenberg, and R. Ungrangsi. Robust location detection with sensor networks. *IEEE J. Sel. Areas Commun.*, 22(6):1016–1025, August 2004.

[15] P. J. Slater. Domination and location in acyclic graphs. *Networks*, 17(1):55–64, 1987.

[16] P. J. Slater. Dominating and reference sets in a graph. *J. Math. Phys. Sci.*, 22:445–455, 1988.

[17] P. J. Slater. Fault-tolerant locating-dominating sets. *Discrete Math.*, 249(1–3):179–189, 2002.

Publication V

Ville Junnila, Tero Laihonen and Tuomo Lehtilä. On Levenshtein's channel and list size in information retrieval, *IEEE Transactions on Information Theory*, accepted for publication, 2020.

https://doi.org/10.1109/TIT.2020.3016269

On Levenshtein's Channel and List Size in Information Retrieval

Ville Junnila, Tero Laihonen and Tuomo Lehtilä

Abstract

The Levenshtein's channel model for substitution errors is relevant in information retrieval where information is received through many noisy channels. In each of the channels there can occur at most t errors and the decoder tries to recover the information with the aid of the channel outputs. Recently, Yaakobi and Bruck considered the problem where the decoder provides a list instead of a unique output. If the underlying code $C \subseteq \mathbb{F}_2^n$ has error-correcting capability e, we write $t = e + \ell$, ($\ell \geq 1$). In this paper, we provide new (constant) bounds on the size of the list. In particular, we give using Sauer-Shelah lemma the upper bound $\ell + 1$ on the list size for large enough n provided that we have sufficient number of channels. We also show that the bound $\ell + 1$ is the best possible. Most of our other new results rely on constant weight codes.

Index Terms

Levenshtein's Channel, Information Retrieval, Substitution Errors, List Decoding, Sauer-Shelah Lemma.

I. INTRODUCTION

In this paper, we consider the *Levenshtein's channel model* of substitution errors introduced in [2] for sequences reconstruction problems. The original motivation came from biology and chemistry where the usual redundancy method of error correction is not feasible. Recently, it was pointed out that this channel model is very relevant to information retrieval in advanced storage technologies where the stored information is either a single copy, which is read by many read heads, or the stored information has several copies [3], [4]. As mentioned in [3], this model is specifically applicable to DNA data storage systems [5]. There DNA strands give us a large number of erroneous copies of the information and we try to recover the information with the aid of these strands. For various related sequences reconstruction problems (like the deletion and insertion errors) see, for example, [2], [6], [7].

Let us first introduce some notation. We denote the set $\{1, 2, \ldots, n\}$ by $[1, n]$. Let $\mathbb{F} = \mathbb{F}_2$ be a finite field of 2 elements, and denote the Hamming space by $\mathbb{F}^n$. The *support* of a word $\mathbf{x} = x_1 \ldots x_n \in \mathbb{F}^n$ is defined by $\text{supp}(\mathbf{x}) = \{i \mid x_i \neq 0\}$. We denote the all-zero word $\mathbf{0} = 00 \ldots 0 \in \mathbb{F}^n$ and $\mathbf{e}_i \in \mathbb{F}^n$ is a word with 1 in the ith coordinate and zeros elsewhere. The *Hamming weight* $w(\mathbf{x})$ of $\mathbf{x} \in \mathbb{F}^n$ equals $|\text{supp}(\mathbf{x})|$. The *Hamming distance* is defined as $d(\mathbf{x}, \mathbf{y}) = w(\mathbf{x} + \mathbf{y})$ for $\mathbf{x}, \mathbf{y} \in \mathbb{F}^n$. We denote the *Hamming ball* of radius t centered at $\mathbf{x} \in \mathbb{F}^n$ by $B_t(\mathbf{x}) = \{\mathbf{y} \in \mathbb{F}^n \mid d(\mathbf{x}, \mathbf{y}) \leq t\}$ and the cardinality of the ball by $V(n, t) = \sum_{i=0}^{t} \binom{n}{i}$. A nonempty subset of $\mathbb{F}^n$ is called a *code* and its elements are called *codewords*. The *minimum distance* of a code $C \subseteq \mathbb{F}^n$ is defined as $d_{\min}(C) = \min_{\mathbf{c}_1, \mathbf{c}_2 \in C, \mathbf{c}_1 \neq \mathbf{c}_2} d(\mathbf{c}_1, \mathbf{c}_2)$. Thus, the code has the error-correcting capability $e = e(C) = \lfloor (d_{\min}(C) - 1)/2 \rfloor$.

Let us consider now the channel model in more detail. A codeword $\mathbf{x} \in C \subseteq \mathbb{F}_2^n$ is transmitted through N channels where at most t substitution errors can occur in each of them — in other words, we get N estimations of a stored information unit. (In the model, it is assumed that all the outputs from the channels are different from each other.) This is illustrated in Fig. 1. It is also assumed that $t > e(C)$, that is, there can be more errors than the code C can cope if it is considered only as an error-correcting code. We denote $t = e(C) + \ell = e + \ell$ for $\ell \geq 1$. For a recent generalization of the problem, see [3].

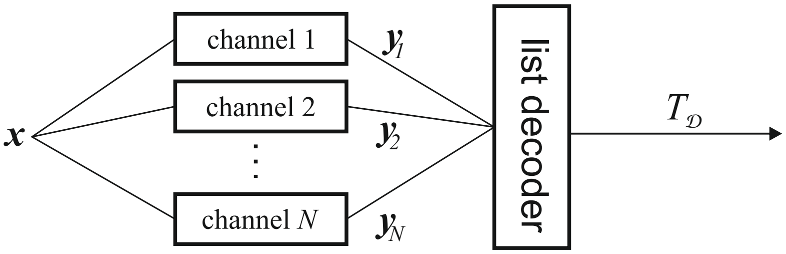


Fig. 1. The Levenshtein's channel model.

This paper was presented in part at 2019 IEEE International Symposium on Information Theory (ISIT2019) (see [1]).

T. Lehtilä was supported by the University of Turku Graduate School (UTUGS) and the Vilho, Yrjö and Kalle Väisälä Foundation.

The authors are with the Department of Mathematics and Statistics, University of Turku, Turku FI-20014, Finland (e-mail: viljun@utu.fi; terolai@utu.fi; tualeh@utu.fi)

Based on the N different outputs $Y = \{\mathbf{y}_1, \dots, \mathbf{y}_N\}$ of the channels, we should be able to recover $\mathbf{x}$. Clearly, if $t \leq e$, then only one channel is enough. In [8], [9], the authors consider the situation where instead of always recovering $\mathbf{x}$ uniquely, we obtain sometimes a short list of possibilities for $\mathbf{x}$. In other words, based on the different output words $\mathbf{y}_1, \dots, \mathbf{y}_N$ and the code C, the list decoder $\mathcal{D}$ gives an estimation $T_\mathcal{D} = T_\mathcal{D}(Y) = \{\mathbf{x}_1, \dots, \mathbf{x}_{|T_\mathcal{D}|}\}$ on the transmitted word $\mathbf{x}$. We denote by $\mathcal{L}_\mathcal{D}$ the maximum cardinality of the list $T_\mathcal{D}(Y)$ over all possible sets Y of output words. The decoder is *successful* if the transmitted word $\mathbf{x}$ belongs to $T_\mathcal{D}$. In this paper, we concentrate on the minimal value of $\mathcal{L}_\mathcal{D}$ over all successful decoders $\mathcal{D}$, that is, on the value $\mathcal{L} = \min_{\mathcal{D} \text{ is successful}}\{\mathcal{L}_\mathcal{D}\}$. Hence, we have

$$\mathcal{L} = \max\{|C \cap (\bigcap_{\mathbf{y} \in Y} B_t(\mathbf{y}))| \mid Y \text{ is a set of } N \text{ output words}\}.$$

We also denote

$$T = T(Y) = C \cap (\bigcap_{\mathbf{y} \in Y} B_t(\mathbf{y})).$$

The value of $\mathcal{L}$ is studied for example, in [8]–[13]. Naturally, we would like to have as small $\mathcal{L}$ as possible. Notice that $\mathcal{L}$ depends on e, ℓ, n, C and N where C is an e-error-correcting code.

There is also another closely related problem of *information retrieval in associative memory* introduced by Yaakobi and Bruck [8], [9]. In their model, an associative memory is given as a (simple and undirected) graph $G = (V, E)$. A vertex in the graph corresponds to a stored information unit and if two information units are associated, then there is an edge between them. Moreover, two vertices are called *t-associated*, if the distance between them is at most t. An unknown information unit $x \in V$ is retrieved from the associative memory using *input clues* provided by an information seeker. The input clues are t-associated to x and also belong to a code $C \subseteq V$ serving as a reference set. The reference set should be such that given enough input clues, the sought information unit x can be found unambiguously (or with some small uncertainty). Naturally, we want the maximum number m of input clues, which are needed to retrieve any information unit from the memory, to be as small as possible. The two parameters $\mathcal{L}$ and m are closely related (see, for instance, [10]).

The structure of the paper is as follows. In Section II, we show some upper and lower bounds on $\mathcal{L}$ for an e-error-correcting code when $t = e + \ell$. We also show that there exist e-error-correcting codes such that $\mathcal{L}$ is not a constant (i.e., depends on n) if the number of channels $N \leq V(n, \ell - 1)$. In Section III, we give an upper bound $\mathcal{L} \leq \ell + 1$ for an e-error-correcting code when n is large enough and $N \geq V(n, \ell - 1) + 1$. Moreover, in Theorem 8, we show that there exist codes which attain this upper bound. Section IV considers a case with at least two distant output words in Y when $e \geq 2\ell - 1$. We show that having distant output words is a reasonable assumption and in that case $\mathcal{L}$ is rather small (we may even reach $|T| \leq 2$). Finally, in Section V, we consider the case with less than $V(n, \ell - 1) + 1$ channels. We especially show that if $V(n, \ell - a - 1) + 1 \leq N \leq V(n, \ell - a)$ where $0 \leq a \leq \ell - 1$ and if C is an e-error-correcting code such that $\mathcal{L}$ is maximal, then $\mathcal{L} = \Theta(n^a)$.

II. ELEMENTARY BOUNDS ON $\mathcal{L}$

For the rest of the section, let C be an e-error-correcting code in $\mathbb{F}^n$ and $t = e + \ell$ be the maximum number of errors that might occur during the transmission. We will first consider upper bounds on $\mathcal{L}$ and then the lower bounds. The basic idea on estimating the maximum length $\mathcal{L}$ of the decoded list is the following: given the output words of the channels, we analyse the number of codewords of C that locate in the intersection of Hamming balls of radius t centered at the output words. As expected, the length $\mathcal{L}$ of the decoded list in the Levenshtein's channel model strongly depends on the number of channels. In particular, as N increases, $\mathcal{L}$ decreases and vice versa. We discuss more about the dependency between N and $\mathcal{L}$ in Section V.

In Remark 9, we show that if the number of channels N is at most $V(n, \ell - 1)$, then the maximum length $\mathcal{L}$ of the decoded list might depend on n. Hence, in this paper, we focus on the case $N \geq V(n, \ell - 1) + 1$ and later prove that $\mathcal{L} \leq 2^\ell$ if $N \geq V(n, \ell - 1) + 1$. Previously, in [2] and [8], the Levenshtein's channel model has been considered for $\mathcal{L} = 1$ and $\mathcal{L} = 2$, respectively. However, in both cases, the number of channels is larger than $N = V(n, \ell - 1) + 1$ focused on this paper.

The results on the number of required channels in the cases $\mathcal{L} = 1$ and $\mathcal{L} = 2$ are obtained by analysing cardinalities of two and three intersecting Hamming balls centered at the codewords of C, respectively. However, contrary to the cases with two or three balls, if the intersection of four or more balls is considered, then the size of the intersection no longer depends on the distances of the centers of the balls (see [14, p. 36]). Hence, in this paper, another approach is used. For this purpose, observe first that each Hamming ball of radius e contains at most one codeword of C. Thus, if the intersection of code C and the balls of radius t centered at the output words of Y can be covered by k balls of radius e, then a list of length (at most) k can be obtained. This approach is reformulated in the following lemma.

Lemma 1. *If for any $Y = \{\mathbf{y}_1, \dots, \mathbf{y}_N\}$ we have $C \cap (\bigcap_{i=1}^N B_t(\mathbf{y}_i)) \subseteq \bigcup_{i=1}^k B_e(\beta_i)$ for some words $\beta_i \in \mathbb{F}^n$ ($i = 1, \dots, k$), then $\mathcal{L} \leq k$*

Notice that the previous lemma also gives a decoding algorithm. Indeed, if the words β_i are known, then each ball $B_e(\beta_i)$ contains at most one codeword and the decoding can be done by adding these codewords to the list T.

A. Upper bounds on $\mathcal{L}$

Now we are ready to examine the actual upper bounds on $\mathcal{L}$. The first upper bound is based on the following theorem by Kleitman [15].

Theorem 2. *If r is a positive integer, $n \geq 2r+1$ and S is a subset of $\mathbb{F}^n$ such that $d(\mathbf{x}, \mathbf{y}) \leq 2r$ for any distinct $\mathbf{x}, \mathbf{y} \in \mathbb{F}^n$, then $|S| \leq V(n, r)$.*

The following result is an immediate corollary of the previous theorem.

Corollary 3. *If $n \geq 2\ell - 1$ and the number of channels $N \geq V(n, \ell-1)+1$, then there exist two output words $\mathbf{y}_1$ and $\mathbf{y}_2$ such that $d(\mathbf{y}_1, \mathbf{y}_2) \geq 2\ell - 1$.*

In the following theorem, we show that the maximum length $\mathcal{L}$ of the decoded list is at most $\binom{2\ell}{\ell}$. This result and its proof can be seen as reformulations of a result by Yaakobi and Bruck [8, Algorithm 18].

Theorem 4. *Let $n \geq 2\ell - 1$ and C be an e-error-correcting code in $\mathbb{F}^n$. If $t = e + \ell$ and $N \geq V(n, \ell - 1)+1$, then we have*

$$\mathcal{L} \leq \binom{2\ell}{\ell}.$$

Proof. Assume that $N \geq V(n, \ell-1)+1$. By Corollary 3, we have two outputs $\mathbf{y}_0, \mathbf{y} \in Y$ such that $d(\mathbf{y}_0, \mathbf{y}) \geq 2\ell - 1$. Without loss of generality, we may assume that $\mathbf{y}_0 = \mathbf{0}$. Since $w(\mathbf{y}) \geq 2\ell - 1$, there exists a set $A \subseteq \text{supp}(\mathbf{y})$ with $2\ell - 1$ elements. Now either $\mathbf{y}_0$ or $\mathbf{y}$ differs from the input word $\mathbf{x}$ in at least ℓ coordinates in A. Suppose first that this is the case with the word $\mathbf{y}_0$, i.e., $d(\mathbf{y}_0, \mathbf{x}) \geq \ell$ or $w(\mathbf{x}) \geq \ell$. Denote by $\beta_i \in \mathbb{F}^n$, $1 \leq i \leq \binom{2\ell-1}{\ell}$, all the words of weight ℓ with the support belonging to A. Since $d(\mathbf{y}_0, \mathbf{x}) \geq \ell$, we have $d(\beta_i, \mathbf{x}) \leq e$ for some β_i. Analogously, in the case where $\mathbf{x}$ differs from $\mathbf{y}$ in at least ℓ coordinates of A, we may choose $\beta_i \in \mathbb{F}^n$, $\binom{2\ell-1}{\ell}+1 \leq i \leq 2\binom{2\ell-1}{\ell}$, to be all the words of weight $\ell - 1$ with the support belonging to A. Again, for some β_i, we have $d(\beta_i, \mathbf{x}) \leq e$. Therefore, by Lemma 1, we obtain $\mathcal{L} \leq \binom{2\ell-1}{\ell-1} + \binom{2\ell-1}{\ell} = \binom{2\ell}{\ell}$ and the claim follows. □

In order to improve the previous upper bound (to 2^ℓ), we present the well-known Sauer-Shelah lemma ([16], [17]). Let $\mathcal{F}$ be a family of subsets of $[1, n]$, where n is a positive integer. We say that a subset S of $[1, n]$ is *shattered* by $\mathcal{F}$ if for any subset $E \subseteq S$ there exists a set $F \in \mathcal{F}$ such that $F \cap S = E$. The Sauer-Shelah lemma states that if $|\mathcal{F}| > \sum_{i=1}^{k} \binom{n}{i}$, then $\mathcal{F}$ shatters a subset of size (at least) k. Since the subsets of $[1, n]$ can naturally be interpreted as words of $\mathbb{F}^n$, the Sauer-Shelah lemma can be reformulated as follows.

Theorem 5 ([16], [17])**.** *If $Y \subseteq \mathbb{F}^n$ is a set containing at least $\sum_{i=0}^{k-1} \binom{n}{i} + 1$ words, then there exists a set S of k coordinates such that for any word $\mathbf{w} \in \mathbb{F}^n$ with supp$(\mathbf{w}) \subseteq S$ there exists a word $\mathbf{s} \in Y$ satisfying supp$(\mathbf{w})$ = supp$(\mathbf{s}) \cap S$. Here we say that the set S of coordinates is* shattered *by Y.*

In the following theorem, we show that $\mathcal{L} \leq 2^\ell$.

Theorem 6. *Let $n \geq \ell$ and C be an e-error-correcting code in $\mathbb{F}^n$. If $t = e + \ell$ and $N \geq V(n, \ell - 1) + 1$, then we have*

$$\mathcal{L} \leq 2^\ell.$$

Proof. Let Y be the set of output words and $\mathbf{x}$ be the input word. Assume that $N \geq V(n, \ell-1)+1$. Now, by Theorem 5, there exists a set S of ℓ coordinates which is shattered by Y. Without loss of generality, we may assume that $S = [1, \ell]$. Let $\mathbf{s}$ be a word such that $\text{supp}(\mathbf{s}) = S$ and $Y' = \{\mathbf{y}_1, \ldots, \mathbf{y}_{2^\ell}\}$ be a subset of Y such that Y' shatters S. Define then $\beta_i = \mathbf{s} + \mathbf{y}_i \in \mathbb{F}^n$ for $1 \leq i \leq 2^\ell$. By the choice of Y', there exists a word $\mathbf{y}_i \in Y'$ such that $\text{supp}(\mathbf{y}_i) \cap S = \text{supp}(\mathbf{x} + \mathbf{s}) \cap S$, i.e., $\mathbf{y}_i$ and $\mathbf{x}$ differ in ℓ coordinate places of S. Hence, we obtain $d(\beta_i, \mathbf{x}) \leq e$ for $\beta_i = \mathbf{s} + \mathbf{y}_i$. Therefore, by Lemma 1, we have $\mathcal{L} \leq 2^\ell$ and the claim follows. □

Observe that when $\ell = 1$ and $N \geq 2$ or $\ell = 2$ and $N \geq n + 2$, we have $\mathcal{L} \leq 2$ or $\mathcal{L} \leq 4$, respectively. Later, in Theorem 8, we show that the first upper bound is tight and then, in Remark 18, we show that we can in some circumstances attain the upper bound $\mathcal{L} \leq 4$.

B. Lower bounds on $\mathcal{L}$

In the following, we concentrate on the lower bounds on $\mathcal{L}$. Here the main idea of the proofs is to find a (worst possible) set of output words to maximize the possible input words that could have been transmitted. In the following theorem, we give a lower bound on the list size when the number of channels is bounded from above.

Theorem 7. *For an e-error-correcting code $C \subseteq \mathbb{F}^n$ and radius $t = e + \ell$, we have*

$$\mathcal{L} \geq \frac{|C|(V(n, t-a+1) - \binom{n-a}{t-a+1})}{2^n}$$

if there exist at most $N \leq V(n, a-1)+1$ channels, $a \leq \ell$ and $n \geq t+1$.

Proof. Let us first consider the words of $B_{a-1}(\mathbf{0})$. The intersection of the balls with radius t centered at these words gives

$$\bigcap_{\mathbf{b} \in B_{a-1}(\mathbf{0})} B_t(\mathbf{b}) = B_{t-a+1}(\mathbf{0}).$$

Denote by $\mathbf{z}$ the word with $\text{supp}(\mathbf{z}) = [1, a]$. Let us denote by S the intersection $B_t(\mathbf{z}) \cap B_{t-a+1}(\mathbf{0})$. It is straightforward to verify that $|S| = V(n, t-a+1) - \binom{n-a}{t-a+1}$. Next we show that $\mathcal{L} \geq |C||S|/2^n$ which gives the assertion. It is straightforward to verify that

$$\sum_{\mathbf{u} \in \mathbb{F}^n} |(\mathbf{u}+S) \cap C| = |C||S|.$$

Therefore, there exists $\mathbf{u} \in \mathbb{F}^n$ such that $|(\mathbf{u}+S) \cap C| \geq |C||S|/2^n$. Let $\mathbf{c} \in C$ be a codeword in $\mathbf{u}+S$. If we transmit $\mathbf{c}$ through the N channels with at most t errors occuring in each one, then we can receive a set of output words Y which is a subset of $B_{a-1}(\mathbf{u}) \cup \{\mathbf{z}+\mathbf{u}\}$. Therefore, we get $\mathcal{L} \geq |C||S|/2^n$. ☐

Next we give a lower bound on the list size when the number of channels $N \leq V(n, \ell-1)+1$. In other words, we show that there exists an e-error-correcting code for which $\mathcal{L} \geq \ell+1$. Later, in Section III, it is shown that the lower bound can be attained for any e-error-correcting code if $N \leq V(n, \ell-1)+1$ and n is large enough.

Theorem 8. *Let $t = e+\ell$. There exists an e-error-correcting code $C \subseteq \mathbb{F}^n$ such that $\mathcal{L} \geq \ell+1$ if $n \geq \ell+\ell e+e$ and the number of channels satisfies $N \leq V(n, \ell-1)+1$.*

Proof. Let us consider a code C which consists of the codewords $\mathbf{c}_i$ $(i = 1, \dots, \ell)$ satisfying

$$\text{supp}(\mathbf{c}_i) = \{i, \ell+e(i-1)+1, \dots, \ell+e(i-1)+e\}$$

together with the word $\mathbf{c}_{\ell+1}$ where $\text{supp}(\mathbf{c}_{\ell+1}) = [n-e+1, n]$. Observe that $w(\mathbf{c}_1) = \cdots = w(\mathbf{c}_\ell) = e+1$ and $w(\mathbf{c}_{\ell+1}) = e$. Since the supports of these $\ell+1$ codewords are disjoint, they form a code with minimum distance $2e+1$.

Let $\mathbf{z} \in \mathbb{F}^n$ be a word such that $\text{supp}(\mathbf{z}) = [1, \ell]$. Assume that the set Y of the N received output words from the channels is a subset of $B_{\ell-1}(\mathbf{0}) \cup \{\mathbf{z}\}$. It is easy to see that the codewords $\mathbf{c}_i$ are included in $B_t(\mathbf{z})$ and, by the proof of Theorem 7 (with $a = \ell$), all the codewords $\mathbf{c}_i$ also belong to the intersection of the balls of radius t centered at the output words of $B_{\ell-1}(\mathbf{0})$. Consequently, for the code C, we obtain $\mathcal{L} \geq \ell+1$. ☐

Notice that previous theorem is not just an example suitable for small codes. In fact, if n is large enough, we may take any e-error-correcting code, remove every codeword in some $(3e+1)$-radius ball and insert the code C inside it in such a way that the all-zero word in the proof of previous theorem corresponds to the central word $\mathbf{w}$ of the $(3e+1)$-radius ball. Then it is easy to see that with set of output words Y corresponding to the one in the proof of previous theorem, we get $\mathcal{L} \geq \ell+1$.

In the following remark, we show that the list size can depend on n if the number of channels is at most $V(n, \ell-1)$.

Remark 9. Let C be an e-error-correcting code in $\mathbb{F}^n$. Assume that the number of channels $N \leq V(n, \ell-1)$ and that all the output words locate inside $B_{\ell-1}(\mathbf{0})$. By the proof of Theorem 7, we know that

$$B_{e+1}(\mathbf{0}) = B_{t-\ell+1}(\mathbf{0}) = \bigcap_{\mathbf{b} \in B_{\ell-1}(\mathbf{0})} B_t(\mathbf{b}).$$

In particular, all the words of weight $e+1$ belong to the intersection. By [18, p. 525], there exists a code with constant weight $e+1$ and minimum distance $2e+2$ with $\lfloor n/(e+1) \rfloor$ words. This implies that $\mathcal{L} \geq \lfloor n/(e+1) \rfloor$ and, hence, the list size depends on n when e is constant.

III. OPTIMAL UPPER BOUND $\mathcal{L} \leq \ell+1$ FOR LARGE ENOUGH n

In this section, we first give bound $\mathcal{L} \leq \ell+1$ when we have $N \geq V(n, \ell-1)+1$ and n is exponentially dependant on e and ℓ. Then we improve it for cases with n polynomially depending on e and ℓ. Notice that these results are improved versions of the results in the conference version of this paper [1].

In the following lemma, we show that if n is large enough and $N \geq \sum_{i=0}^{\ell-1} \binom{n}{i} + 1$, then there exists an output word $\mathbf{y} \in Y$ such that $\mathbf{y}$ differs from the transmitted codeword $\mathbf{x}$ in at least $\ell-1$ coordinate places outside $\overline{D} \subseteq [1, n]$ (with small size compared to n).

Lemma 10. *Assume that $Y \subseteq \mathbb{F}^n$, $|Y| = N \geq \sum_{i=0}^{\ell-1} \binom{n}{i} + 1$, C is an e-error-correcting code and b is a positive integer. Then for any codeword $\mathbf{x} \in T(Y)$ and for any set $\overline{D} \subseteq [1, n]$ with $|\overline{D}| = b$, there exists a word $\mathbf{y} \in Y$ such that $|supp(\mathbf{x}+\mathbf{y}) \setminus \overline{D}| \geq \ell-1$ if*

$$n \geq \ell-2+(\ell-1)^2 2^b.$$

Proof. Let $\overline{D} \subseteq [1,n]$ and $|\overline{D}| = b$ for some fixed b. Without loss of generality, we may assume that $\mathbf{x} = \mathbf{0}$. Suppose to the contrary that there does not exist a word $\mathbf{y} \in Y$ such that $|\mathrm{supp}(\mathbf{x}+\mathbf{y}) \setminus \overline{D}| = |\mathrm{supp}(\mathbf{y}) \setminus \overline{D}| \geq \ell - 1$, i.e., $|\mathrm{supp}(\mathbf{y}) \setminus \overline{D}| < \ell - 1$ for all $\mathbf{y} \in Y$. This implies that the number of words in Y is at most

$$\begin{aligned}
&\sum_{j=0}^{\ell-2} \sum_{i=0}^{\min\{b,t-j\}} \binom{b}{i}\binom{n-b}{j} \\
\leq &\sum_{j=0}^{\ell-2} \sum_{i=0}^{b} \binom{b}{i}\binom{n-b}{j} \\
= &2^b \sum_{j=0}^{\ell-2} \binom{n-b}{j} \\
\leq &(\ell-1)2^b \binom{n}{\ell-2} \\
= &(\ell-1)\binom{n}{\ell-1}\frac{\ell-1}{n-\ell+2}2^b \\
\leq &\binom{n}{\ell-1},
\end{aligned}$$

when $n \geq \ell - 2 + (\ell-1)^2 2^b$. This contradicts with the assumption that $N = |Y| \geq \sum_{i=0}^{\ell-1}\binom{n}{i} + 1$. Thus, the claim follows. □

The following lemma is a critical part in showing that $\mathcal{L} \leq \ell + 1$ when n is large enough. In particular, we show that if there exists a word $\mathbf{w}$ close to every codeword in T, then the cardinality of T is rather small. Moreover, we later verify the existence of such a word $\mathbf{w}$.

Lemma 11. *Let the set of outputs Y consist of $N \geq \sum_{i=0}^{\ell-1}\binom{n}{i} + 1$ words and C be an e-error-correcting code. Further let h be an integer and $\mathbf{w} \in \mathbb{F}^n$ be a word such that $0 \leq h \leq \ell$ and $d(\mathbf{w}, \mathbf{c}) \leq e + h$ for each $\mathbf{c} \in T(Y)$. Then we have*

$$|T(Y)| \leq \sum_{i=0}^{h} \binom{\ell}{i}.$$

Proof. Let $\mathbf{x}$ be the transmitted codeword. Assume that $\mathbf{w} \in \mathbb{F}^n$ is a word satisfying $d(\mathbf{w}, \mathbf{c}) \leq e + h$ for every $\mathbf{c} \in T(Y)$; in particular, $d(\mathbf{w}, \mathbf{x}) \leq e + h$. Without loss of generality, we may assume that $\mathbf{w} = \mathbf{0}$. Since $N \geq \sum_{i=0}^{\ell-1}\binom{n}{i} + 1$, by Theorem 5, there exists a set $S \subseteq [1,n]$ of ℓ coordinates which are shattered by a subset $Y' = \{\mathbf{y}_1, \ldots, \mathbf{y}_{2^\ell}\} \subseteq Y$. Let $\mathbf{s}$ denote a word such that $\mathrm{supp}(\mathbf{s}) = S$. The proof now divides into two cases based on the number of different coordinates between $\mathbf{x}$ and $\mathbf{w}$ in S, that is, $|\mathrm{supp}(\mathbf{x}) \cap S|$.

Assume first that $|\mathrm{supp}(\mathbf{x}) \cap S| \leq h - 1$. Define $\overline{Y} = \{\overline{\mathbf{y}} \in Y' \mid |\mathrm{supp}(\overline{\mathbf{y}}) \cap S| \geq \ell - (h-1)\} \subseteq Y'$ and $\mathcal{B}_1 = \{\beta = \overline{\mathbf{y}} + \mathbf{s} \mid \overline{\mathbf{y}} \in \overline{Y}\}$. Notice that $|\overline{Y}| = |\mathcal{B}_1| = \sum_{i=\ell-(h-1)}^{\ell}\binom{\ell}{i} = \sum_{i=0}^{h-1}\binom{\ell}{i}$. Since $|\mathrm{supp}(\mathbf{x}) \cap S| \leq h-1$, there exist words $\overline{\mathbf{y}} \in \overline{Y}$ and $\beta = \overline{\mathbf{y}} + \mathbf{s} \in \mathcal{B}_1$ such that $\mathrm{supp}(\overline{\mathbf{y}} + \mathbf{x}) \cap S = S$, that is, $\mathbf{x}$ and $\overline{\mathbf{y}}$ differ in every coordinate of S. Therefore, we have $d(\mathbf{x}, \beta) = d(\mathbf{x}, \overline{\mathbf{y}} + \mathbf{s}) = d(\mathbf{x}, \overline{\mathbf{y}}) - \ell \leq t - \ell \leq e$.

Let us then assume that $|\mathrm{supp}(\mathbf{x}) \cap S| \geq h$. Define $\mathcal{B}_2 = \{\beta \in \mathbb{F}^n \mid w(\beta) = h \text{ and } \mathrm{supp}(\beta) \subseteq S\}$. Notice that $|\mathcal{B}_2| = \binom{\ell}{h}$. Now there exists a word $\beta \in \mathcal{B}_2$ such that $\mathrm{supp}(\beta) \subseteq \mathrm{supp}(\mathbf{x})$. Hence, we have $d(\mathbf{x}, \beta) = |\mathrm{supp}(\mathbf{x})| - h \leq (e+h) - h \leq e$. Therefore, we obtain that

$$\mathbf{x} \in \bigcup_{\beta \in \mathcal{B}_1 \cup \mathcal{B}_2} B_e(\beta)$$

and the claim follows by Lemma 1 since $|\mathcal{B}_1 \cup \mathcal{B}_2| = |\mathcal{B}_1| + |\mathcal{B}_2| = \sum_{i=0}^{h}\binom{\ell}{i}$. □

The following corollary is immediately obtained by choosing in the previous lemma $h = 1$.

Corollary 12. *Let the set of output words Y consist of $N \geq \sum_{i=0}^{\ell-1}\binom{n}{i} + 1$ words, C be an e-error-correcting code and let there exist a word $\mathbf{w} \in \mathbb{F}^n$ such that $d(\mathbf{w}, \mathbf{c}) \leq e + 1$ for each $\mathbf{c} \in T(Y)$. Then we have*

$$|T(Y)| \leq \ell + 1.$$

In the following lemma, we show that if n is large enough and $N \geq \sum_{i=0}^{\ell-1}\binom{n}{i} + 1$, then the pairwise distances of codewords in T are rather small.

Lemma 13. *Let $n \geq \ell - 2 + (\ell-1)^2 2^{2t}$, C be an e-error-correcting code and $|Y| = N \geq \sum_{i=0}^{\ell-1}\binom{n}{i} + 1$. Then we have $d(\mathbf{c}_1, \mathbf{c}_2) \leq 2e + 2$ for any two $\mathbf{c}_1, \mathbf{c}_2 \in T(Y)$.*

Proof. Let $\mathbf{c}_1$ and $\mathbf{c}_2$ be codewords in $T(Y)$. Without loss of generality, we may assume that $\mathbf{c}_1 = \mathbf{0}$. In order to show that $d(\mathbf{c}_1, \mathbf{c}_2) \leq 2e+2$, we suppose to the contrary that $d(\mathbf{c}_1, \mathbf{c}_2) \geq 2e+3$, i.e., $w(\mathbf{c}_2) \geq 2e+3$. Since $\mathbf{c}_1, \mathbf{c}_2 \in T(Y)$, we have $w(\mathbf{c}_2) = d(\mathbf{c}_1, \mathbf{c}_2) \leq 2t$. Hence, there exists a set $\overline{D} \subseteq [1,n]$ such that $|\overline{D}| = 2t$ and $\text{supp}(\mathbf{c}_2) \subseteq \overline{D}$.

Since $n \geq \ell - 2 + (\ell-1)^2 2^{2t}$, by Lemma 10, there exists an output $\mathbf{y} \in Y$ such that $|\text{supp}(\mathbf{y}) \setminus \text{supp}(\mathbf{c}_2)| \geq \ell - 1$. Since $w(\mathbf{y}) = d(\mathbf{y}, \mathbf{c}_1) \leq t$, we have $|\text{supp}(\mathbf{c}_2) \cap \text{supp}(\mathbf{y})| \leq e+1$. This further implies that

$$d(\mathbf{c}_2, \mathbf{y}) \geq (w(\mathbf{c}_2) - |\text{supp}(\mathbf{c}_2) \cap \text{supp}(\mathbf{y})|) + \ell - 1 \geq (2e+3-(e+1)) + \ell - 1 \geq t+1.$$

This leads to a contradiction, and the claim follows. ☐

We have shown that if there exists a word $\mathbf{w}$ such that it is close to every codeword in T, then $|T|$ is small. Moreover, we have shown that every codeword in T is pairwise close to each other. Therefore, it seems rather natural suggestion for such a word $\mathbf{w}$ to indeed exist. The proof of the following theorem is based on this idea.

Theorem 14. *Let* $n \geq \ell - 2 + (\ell-1)^2 2^b$, $b = \max\{2t, 4e+4\}$, $|Y| = N \geq \sum_{i=0}^{\ell-1} \binom{n}{i} + 1$ *and* C *be an* e*-error-correcting code. Then we have*

$$\mathcal{L} \leq \ell + 1.$$

Proof. Observe first that the cases $\ell = 0$ and $\ell = 1$ follow from Theorem 6 since $2^0 = 0+1$ and $2^1 = 1+1$. Therefore, we may assume that $\ell \geq 2$. Hence, there exist codewords $\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2 \in T(Y)$ (as we are immediately done if $|T(Y)| \leq 2$). Without loss of generality, we may assume that $\mathbf{c}_0 = \mathbf{0}$. Thus, we have $w(\mathbf{c}_1) = d(\mathbf{c}_0, \mathbf{c}_1)$ and $w(\mathbf{c}_2) = d(\mathbf{c}_0, \mathbf{c}_2)$.

Observe that the pairwise distances of any codewords of C are at least $2e+1$. Therefore, by Lemma 13, we have $2e+1 \leq d(\mathbf{c}_i, \mathbf{c}_j) \leq 2e+2$ for any distinct $i, j \in \{0,1,2\}$. Since $d(\mathbf{c}_0, \mathbf{c}_1) + d(\mathbf{c}_1, \mathbf{c}_2) + d(\mathbf{c}_2, \mathbf{c}_0) = 2\sum_{i=0}^{2} w(\mathbf{c}_i) - 2|\text{supp}(\mathbf{c}_0 \cap \mathbf{c}_1)| - 2|\text{supp}(\mathbf{c}_1 \cap \mathbf{c}_2)| - 2|\text{supp}(\mathbf{c}_2 \cap \mathbf{c}_0)|$, the sum of the distances is even. Hence, we have two possibilities for the distances: either each of them equals $2e+2$ or exactly one of them equals $2e+2$.

Consider first the latter case. Without loss of generality, we may suppose that $d(\mathbf{c}_1, \mathbf{c}_2) = 2e+2$ and $w(\mathbf{c}_1) = w(\mathbf{c}_2) = 2e+1$. Denote now $A = \text{supp}(\mathbf{c}_1) \cap \text{supp}(\mathbf{c}_2)$ and let $\mathbf{w}$ be a word such that $\text{supp}(\mathbf{w}) = A$. It is now immediate that $|A| = e$, $|\text{supp}(\mathbf{c}_1) \cup \text{supp}(\mathbf{c}_2)| = 3e+2$ and $|\text{supp}(\mathbf{c}_i) \setminus A| = e+1$ for each $i \in \{1,2\}$. Notice that $d(\mathbf{w}, \mathbf{c}_i) \leq e+1$ for $i \in \{0,1,2\}$. Let $\mathbf{c}_3$ be an arbitrary codeword in $T(Y)$. As above, we obtain that $2e+1 \leq d(\mathbf{c}_3, \mathbf{c}_i) \leq 2e+2$ for any $i \in \{0,1,2\}$. Moreover, if $|\text{supp}(\mathbf{c}_3) \setminus (\text{supp}(\mathbf{c}_1) \cup \text{supp}(\mathbf{c}_2))| \geq e+2$, then a contradiction follows as $d(\mathbf{c}_1, \mathbf{c}_3) \geq 2e+3$. Thus, denoting $\overline{D} = \text{supp}(\mathbf{c}_1) \cup \text{supp}(\mathbf{c}_2) \cup \text{supp}(\mathbf{c}_3)$, we have $|\overline{D}| \leq 4e+3$.

By Lemma 10, there exists an output word $\mathbf{y} \in Y$ such that $|\text{supp}(\mathbf{y}) \setminus \overline{D}| \geq \ell - 1$. (Observe that $\mathbf{y}$ depends on the choice of $\mathbf{c}_3$.) Notice first that $|\text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_i)| \geq e$ for $i \in \{1,2\}$ since otherwise $d(\mathbf{y}, \mathbf{c}_i) \geq |\text{supp}(\mathbf{c}_i) \setminus \text{supp}(\mathbf{y})| + (\ell-1) \geq 2e+1-(e-1)+(\ell-1) = t+1$ (a contradiction). Furthermore, as $w(\mathbf{y}) = d(\mathbf{y}, \mathbf{c}_0) \leq t$, we have $|\text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_i)| \leq |\text{supp}(\mathbf{y}) \cap \overline{D}| \leq e+1$ for $i \in \{1,2\}$. Moreover, if $|\text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_1)| = e+1$, then $d(\mathbf{y}, \mathbf{c}_2) = |\text{supp}(\mathbf{y}) \setminus \text{supp}(\mathbf{c}_2)| + |\text{supp}(\mathbf{c}_2) \setminus \text{supp}(\mathbf{y})| \geq \ell + e + 1 > t$ (a contradiction). Hence, using analogous arguments to $\mathbf{c}_2$, we obtain that

$$|\text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_1)| = |\text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_2)| = e.$$

Now it can be shown that $A = \text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_1) = \text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_2)$. Indeed, suppose to the contrary that $\text{supp}(\mathbf{c}_1) \cap (\text{supp}(\mathbf{y}) \setminus A) \neq \emptyset$ or $\text{supp}(\mathbf{c}_2) \cap (\text{supp}(\mathbf{y}) \setminus A) \neq \emptyset$. Now a contradiction follows since $d(\mathbf{y}, \mathbf{c}_2) \geq 1 + |\text{supp}(\mathbf{c}_2) \setminus \text{supp}(\mathbf{y})| + (\ell-1) \geq 1 + (e+1) + (\ell-1) = t+1$ or $d(\mathbf{y}, \mathbf{c}_1) \geq t+1$, respectively. Furthermore,

$$|\text{supp}(\mathbf{y}) \setminus \overline{D}| = \ell - 1$$

since otherwise $d(\mathbf{y}, \mathbf{c}_1) \geq t+1$ (a contradiction). Thus, we have $w(\mathbf{y}) = t-1$. Therefore, we have $A = \text{supp}(\mathbf{y}) \cap \overline{D}$. Moreover, we have $|\text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_3)| \leq |\text{supp}(\mathbf{y}) \cap \overline{D}| \leq e$. Hence, we have $w(\mathbf{c}_3) = 2e+1$ (as $2e+1 \leq w(\mathbf{c}_3) \leq 2e+2$). This implies that $|\text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_3)| = e$. Hence, as $A = \text{supp}(\mathbf{y}) \cap \overline{D}$, we obtain that $A = \text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_3)$. Therefore, $A \subseteq \text{supp}(\mathbf{c}_3)$ and $d(\mathbf{w}, \mathbf{c}_3) \leq e+1$ concluding the first case.

The case with $w(\mathbf{c}_1) = w(\mathbf{c}_2) = 2e+2$ is similar to the previous one. As above, we denote $A = \text{supp}(\mathbf{c}_1) \cap \text{supp}(\mathbf{c}_2)$ and let $\mathbf{w}$ be a word such that $\text{supp}(\mathbf{w}) = A$. Similarly, we obtain that $|A| = e+1$, $|\text{supp}(\mathbf{c}_1) \cup \text{supp}(\mathbf{c}_2)| = 3e+3$ and $|\overline{D}| = |\text{supp}(\mathbf{c}_1) \cup \text{supp}(\mathbf{c}_2) \cup \text{supp}(\mathbf{c}_3)| \leq 4e+4$. Hence, by Lemma 10, there exists an output word $\mathbf{y} \in Y$ such that $|\text{supp}(\mathbf{y}) \setminus \overline{D}| \geq \ell - 1$. Since $w(\mathbf{y}) = d(\mathbf{y}, \mathbf{c}_0) \leq t$, we have $|\text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_i)| = e+1$ for $i \in \{1,2\}$ and $w(\mathbf{y}) = t$. As above, it can be shown that $A = \text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_1) = \text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_2)$ and further that $A = \text{supp}(\mathbf{y}) \cap \overline{D}$. Thus, $A = \text{supp}(\mathbf{y}) \cap \text{supp}(\mathbf{c}_3)$ and $d(\mathbf{w}, \mathbf{c}_3) \leq e+1$ concluding the second case.

The claim now follows by Corollary 12. ☐

In the previous theorem, we have shown that $\mathcal{L} \leq \ell + 1$ when n depends exponentially on e and ℓ. The proof is based on Lemma 10, in which we use rather rough estimations. In what follows, we significantly improve the previous theorem by showing that it is enough to require n to depend only polynomially on e and ℓ. We first present an improved version of Lemma 10. The proof of the improved lemma is rather technical and, therefore, its proof is postponed to Appendix.

Lemma 15. *Let $b \geq 3t$ be an integer with $t = e + \ell$ and C_1 be an e-error-correcting code. Assume that $n \geq (\ell-1)^2(b - e + (e+1)(b - 3e - 2e^2 + eb + \binom{b-2e-1}{2})) + \ell - 2$, $|Y| = N \geq \sum_{i=0}^{\ell-1}\binom{n}{i} + 1$, $|T(Y)| \geq 3$ and $\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2 \in T(Y)$. If now $\overline{D} \subseteq [1,n]$ is a set such that $|\overline{D}| = b$ and supp$(\mathbf{c}_0 + \mathbf{c}_1) \cup$ supp$(\mathbf{c}_0 + \mathbf{c}_2) \cup$ supp$(\mathbf{c}_1 + \mathbf{c}_2) \subseteq \overline{D}$, then for any word $\mathbf{w} \in \mathbb{F}^n$ we have supp$(\mathbf{w} + \mathbf{c}_0) \setminus \overline{D} =$ supp$(\mathbf{w} + \mathbf{c}_1) \setminus \overline{D} =$ supp$(\mathbf{w} + \mathbf{c}_2) \setminus \overline{D}$ and there exists an output word $\mathbf{y} \in Y$ such that $|$supp$(\mathbf{y} + \mathbf{c}_0) \setminus \overline{D}| \geq \ell - 1$.*

Proof. See Appendix. ☐

Using the previous lemma, we show a result similar to Lemma 13.

Lemma 16. *Let $n \geq (\ell-1)^2(2t + \ell + (e+1)(3\ell - 2e^2 + 3et + \binom{t+2\ell-1}{2})) + \ell - 2$, $|Y| = N \geq \sum_{i=0}^{\ell-1}\binom{n}{i} + 1$, C be an e-error-correcting code and $|T(Y)| \geq 3$. Then we have $d(\mathbf{c}_1, \mathbf{c}_2) \leq 2e + 2$ for any two $\mathbf{c}_1, \mathbf{c}_2 \in T(Y)$.*

Proof. The proof is similar to the one of Lemma 13. Let $\mathbf{c}_1$ and $\mathbf{c}_2$ be distinct codewords in $T(Y)$ ($|T(Y)| \geq 3$). Without loss of generality, we may assume that $\mathbf{c}_1 = \mathbf{0}$. In order to show that $d(\mathbf{c}_1, \mathbf{c}_2) \leq 2e + 2$, we suppose to the contrary that $d(\mathbf{c}_1, \mathbf{c}_2) \geq 2e + 3$, i.e., $w(\mathbf{c}_2) \geq 2e + 3$. Since $|T(Y)| \geq 3$, there exists another codeword $\mathbf{c}_3 \in T(Y)$. Now we have $|\text{supp}(\mathbf{c}_1 + \mathbf{c}_2) \cup \text{supp}(\mathbf{c}_1 + \mathbf{c}_3) \cup \text{supp}(\mathbf{c}_2 + \mathbf{c}_3)| = |\text{supp}(\mathbf{c}_2) \cup \text{supp}(\mathbf{c}_3) \cup \text{supp}(\mathbf{c}_2 + \mathbf{c}_3)| = |\text{supp}(\mathbf{c}_2) \cup \text{supp}(\mathbf{c}_3)| \leq 3t$ since $d(\mathbf{c}_i, \mathbf{c}_j) \leq 2t$. Hence, there exists a set $\overline{D} \subseteq [1,n]$ such that $|\overline{D}| = b = 3t$ and $\text{supp}(\mathbf{c}_2) \subseteq \text{supp}(\mathbf{c}_1 + \mathbf{c}_2) \cup \text{supp}(\mathbf{c}_1 + \mathbf{c}_3) \cup \text{supp}(\mathbf{c}_2 + \mathbf{c}_3) \subseteq \overline{D}$.

Since $n \geq (\ell-1)^2(2t+\ell+(e+1)(3\ell-2e^2+3et+\binom{t+2\ell-1}{2}))+\ell-2 = (\ell-1)^2(b-e+(e+1)(b-3e-2e^2+eb+\binom{b-2e-1}{2}))+\ell-2$, by Lemma 15, there exists an output word $\mathbf{y} \in Y$ such that $|\text{supp}(\mathbf{y}) \setminus \text{supp}(\mathbf{c}_2)| \geq |\text{supp}(\mathbf{y}) \setminus \overline{D}| \geq \ell - 1$. Since $w(\mathbf{y}) = d(\mathbf{y}, \mathbf{c}_1) \leq t$, we have $|\text{supp}(\mathbf{c}_2) \cap \text{supp}(\mathbf{y})| \leq e + 1$. This further implies that

$$d(\mathbf{c}_2, \mathbf{y}) \geq (w(\mathbf{c}_2) - |\text{supp}(\mathbf{c}_2) \cap \text{supp}(\mathbf{y})|) + \ell - 1 \geq (2e + 3 - (e+1)) + \ell - 1 \geq t + 1.$$

This leads to a contradiction, and the claim follows. ☐

The following theorem is an improved version of Theorem 14.

Theorem 17. *Let $n \geq (\ell-1)^2(b-e+(e+1)(b-3e-2e^2+eb+\binom{b-2e-1}{2}))+\ell-2$, $b = \max\{3t, 4e+4\}$, $|Y| = N \geq \sum_{i=0}^{\ell-1}\binom{n}{i}+1$ and C be an e-error-correcting code. Then we have*

$$\mathcal{L} \leq \ell + 1.$$

Proof. The proof is similar to the one of Theorem 14 with the following small modifications. Lemma 13 is replaced by Lemma 16. Furthermore, instead of Lemma 10, we apply Lemma 15. Here, the additional requirement of Lemma 15 is satisfied as $\text{supp}(\mathbf{c}_0 + \mathbf{c}_1) \cup \text{supp}(\mathbf{c}_0 + \mathbf{c}_2) \cup \text{supp}(\mathbf{c}_1 + \mathbf{c}_2) \subseteq \overline{D} = \text{supp}(\mathbf{c}_0) \cup \text{supp}(\mathbf{c}_1) \cup \text{supp}(\mathbf{c}_2)$. ☐

In the following remark, we show that in order to have $\mathcal{L} \leq \ell + 1$ when C is an e-error-correcting and $N = \sum_{i=0}^{\ell-1}\binom{n}{\ell} + 1$ some restrictions are needed on the values n, ℓ and e.

TABLE I

A POSSIBLE SET OF EIGHT OUTPUT WORDS WITH LIST SIZE $|T| = 2^\ell$.

							$d(\mathbf{c}_1,*)$	$d(\mathbf{c}_2,*)$	$d(\mathbf{c}_3,*)$	$d(\mathbf{c}_0,*)$
$\mathbf{c}_1$	0	1	1	1	0	0	0	4	4	3
$\mathbf{c}_2$	1	0	1	0	1	0	4	0	4	3
$\mathbf{c}_3$	1	1	0	0	0	1	4	4	0	3
$\mathbf{c}_0$	0	0	0	0	0	0	3	3	3	0
$\mathbf{y}_0$	0	0	0	0	0	0	3	3	3	0
$\mathbf{y}_1$	1	1	1	0	0	0	2	2	2	3
$\mathbf{y}_2$	0	1	1	0	0	0	1	3	3	2
$\mathbf{y}_3$	1	0	1	0	0	0	3	1	3	2
$\mathbf{y}_4$	1	1	0	0	0	0	3	3	1	2
$\mathbf{y}_5$	1	0	0	1	0	0	3	3	3	2
$\mathbf{y}_6$	0	1	0	0	1	0	3	3	3	2
$\mathbf{y}_7$	0	0	1	0	0	1	3	3	3	2

Remark 18. In what follows, we give a couple of examples of e-error-correcting codes such that $\mathcal{L} > \ell + 1$ when the number of channels $N \leq \sum_{i=0}^{\ell-1}\binom{n}{\ell} + 1$. Consider first a code $C = \{\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3, \mathbf{c}_0\} \subseteq \mathbb{F}^6$ and a set of outputs $Y = \{\mathbf{y}_0, \ldots, \mathbf{y}_7\} \subseteq \mathbb{F}^6$ given in Table I. By the table, we observe that C is a code with minimum distance 3 and, hence, it is a 1-error-correcting code. Assuming $\mathbf{x} = \mathbf{c}_0 = \mathbf{0}$ is the transmitted word, we notice by the table that $Y \subseteq B_3(\mathbf{x})$. Thus, with $e = 1$, $\ell = 2$ and $t = 3$, the set Y is a possible set of outputs words for $\mathbf{x}$. Therefore, as $\{\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3, \mathbf{c}_0\} \subseteq \bigcap_{j=0}^{7} B_3(\mathbf{y}_j)$, we have $|T| = 4 > \ell + 1 = 3$ with $|Y| = \sum_{i=0}^{\ell-1}\binom{n}{\ell} + 1 = 8$. Hence, $\mathcal{L} > \ell + 1$.

Another example is the extreme situation with $e = 0$, $C = \mathbb{F}^n$, $\ell = n$ and $N = \sum_{i=0}^{\ell-1}\binom{n}{\ell} + 1 = 2^n$. In this case, we have $B_t(\mathbf{y}) = \mathbb{F}^n$ for every $\mathbf{y} \in Y$. Therefore, we obtain that $\bigcap_{\mathbf{y}\in Y} B_t(\mathbf{y}) = \mathbb{F}^n = C$. Thus, $\mathcal{L} = 2^\ell > \ell + 1$ with $N = \sum_{i=0}^{\ell-1}\binom{n}{i} + 1$.

Notice also that the previous examples attain the upper bound $\mathcal{L} \leq 2^\ell$ of Theorem 6.

IV. SMALL LIST SIZE WITH DISTANT OUTPUT WORDS

Throughout the section, we assume that the errors occurring in the transmission are uniformly and (almost) randomly distributed with the exception that no two output words are identical. We have previously assumed that we have at least $\sum_{i=0}^{\ell-1}\binom{n}{i} + 1$ channels. However, as we will see in this section, it is very likely that such a large number of channels is unnecessary to get small list size if n is large and e is sufficiently large compared to ℓ. Moreover, we will provide a method which is very likely to give $|T| \leq 2$ when $e \geq 4\ell - 2$. Notice that these results are improvements on the results in the conference version of this paper [1].

Let C be an e-error-correcting code and $\mathbf{x} \in C$ be the transmitted word. In the following theorem, we see that if we have an output word $\mathbf{y}$ in the vicinity of $\mathbf{x}$, then there cannot be any other codewords in $B_t(\mathbf{y})$, thus, giving us exact knowledge about the transmitted word.

Theorem 19. *Let C be an e-error-correcting code in $\mathbb{F}^n$ and $\mathbf{x} \in C$. If $t = e + \ell$ and $d(\mathbf{x}, \mathbf{y}) \leq e - \ell$ for some output word $\mathbf{y} \in Y$, then we have $B_t(\mathbf{y}) \cap C = \{\mathbf{x}\}$ and $\mathbf{x}$ is the transmitted word.*

Proof. Let $\mathbf{x} \in C$ and $d(\mathbf{x}, \mathbf{y}) \leq e - \ell$ for some $\mathbf{y} \in Y$. Furthermore, we have $d(\mathbf{x}, \mathbf{c}) \geq 2e + 1$ for every codeword $\mathbf{c} \in C$, $\mathbf{c} \neq \mathbf{x}$. Hence, if we have $d(\mathbf{x}, \mathbf{y}) \leq e - \ell$, then the triangular inequality gives us

$$2e + 1 \leq d(\mathbf{x}, \mathbf{c}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{c}) \leq e - \ell + d(\mathbf{y}, \mathbf{c}).$$

Thus, we have $d(\mathbf{y}, \mathbf{c}) \geq t + 1$ for each $\mathbf{c} \in C$, $\mathbf{c} \neq \mathbf{x}$. Therefore, $B_t(\mathbf{y}) \cap C = \{\mathbf{x}\}$. Moreover, there happens at most t errors in each channel and hence, the transmitted word is in $B_t(\mathbf{y}) \cap C$. Thus, $\mathbf{x}$ is the transmitted word and the claim follows. ☐

Now we are going to show that if n is large, then we very likely have two pairwise distant output words.

Theorem 20. *Let C be an e-error-correcting code in $\mathbb{F}^n$ and $\mathbf{x} \in C$ be the transmitted word. If $t = e + \ell$ and $\mathbf{y}_1, \mathbf{y}_2 \in Y$ are output words such that $d(\mathbf{y}_i, \mathbf{x}) \geq e - \ell + 1$ for $i = 1, 2$, then the probability that $d(\mathbf{y}_1, \mathbf{y}_2) \geq 2e - 2\ell + 2$ tends to 1 as n tends to infinity.*

Proof. Let us assume, without loss of generality, that $\mathbf{x} = \mathbf{0}$, $w(\mathbf{y}_1) = e - \ell + a_1$ and $w(\mathbf{y}_2) = e - \ell + a_2$ where $1 \leq a_2 \leq a_1 \leq 2\ell$. We may further assume that $\text{supp}(\mathbf{y}_1) = [1, e - \ell + a_1]$. Observe that if $|\text{supp}(\mathbf{y}_1) \cap \text{supp}(\mathbf{y}_2)| \leq \frac{a_1+a_2}{2} - 1$, then we have $d(\mathbf{y}_1, \mathbf{y}_2) = w(\mathbf{y}_1) + w(\mathbf{y}_2) - 2|\text{supp}(\mathbf{y}_1) \cap \text{supp}(\mathbf{y}_2)| \geq w(\mathbf{y}_1) + w(\mathbf{y}_2) - (a_1 + a_2 - 2) = 2e - 2\ell + 2$. Notice that $\frac{a_1+a_2}{2} - 1 \geq 0$ since $a_1 \geq a_2 \geq 1$. Let us denote by $P_n(a_1, a_2)$ the probability that $d(\mathbf{y}_1, \mathbf{y}_2) \geq 2e - 2\ell + 2$, i.e., $\text{supp}(\mathbf{y}_1) \cap \text{supp}(\mathbf{y}_2)| \leq (a_1 + a_2)/2 - 1$. Now, we have

$$P_n(a_1, a_2) \geq \sum_{i=0}^{\lfloor \frac{a_1+a_2}{2} - 1 \rfloor} \frac{\binom{e-\ell+a_1}{i}\binom{n-e+\ell-a_1}{e-\ell+a_2-i}}{\binom{n}{e-\ell+a_2}} \geq \frac{\binom{n-e+\ell-a_1}{e-\ell+a_2}}{\binom{n}{e-\ell+a_2}} = \prod_{i=0}^{e-\ell+a_2-1} \frac{n - e + \ell - a_1 - i}{n - i} \overset{n\to\infty}{\longrightarrow} 1.$$

Since $P_n(a_1, a_2) \overset{n\to\infty}{\longrightarrow} 1$ for each possible value of a_1 and a_2, the claim follows. ☐

Now, based on Theorems 19 and 20, we obtain that if n is large, $e \geq \ell$ and we have at least two output words in Y, then we either have $|T(Y)| = 1$ or we are very likely to have two output words which are far away from each other. Furthermore, we only consider two output words in this section. However, if we have more output words, say m, and none of them is close to the transmitted word $\mathbf{x}$, then the likelihood that at least two of them are distant is naturally greater than we would have with only two output words. More precisely, the probability is greater than $1 - \prod_{i=2}^{m}(1 - P_n(a_1, a_i))$.

Note that quite modest n is enough for this approach to work; especially, if we have multiple channels. For example, assuming $n = 250$, $e = 10$ and $\ell = 3$, we have $P(a_1, a_2) \geq 0.768$ if $a_1 = a_2 = 1$ and $N = 2$. However, if we have $a_1 = a_2 = 4$ and $N = 2$, then $P(a_1, a_2) \geq 0.999$ or if $a_i = 1$ for $i \in [1, N]$, then $1 - \prod_{i=2}^{N}(1 - P_n(a_1, a_i)) \geq 1 - 0.232^{N-1}$.

In what follows, we use known results for codes with a given minimum distance to obtain upper bounds on the outputted list of codewords. As usual, we denote by $A(n, d)$ the maximal cardinality among all codes in $\mathbb{F}^n$ with minimum distance at least d. Similarly, we denote by $A(n, d, w)$ the maximal cardinality among all *constant weight codes* in $\mathbb{F}^n$, in which each codeword has weight w and of which minimum distance is d. The maximum cardinalities $A(n, d)$ and $A(n, d, w)$ have been widely studied. In what follows, we first present some useful results regarding them. In the following theorem, the well-known Plotkin bound on $A(n, d)$ is given.

Theorem 21 (Plotkin bound [19]). *If $n < 2d + 1$ and d is odd, then*

$$A(n, d) \leq 2 \left\lfloor \frac{d + 1}{2d + 1 - n} \right\rfloor.$$

In the following theorem, we give some useful bounds on $A(n,d,w)$ from [20]. Inequality (i) immediately follows from the definitions of $A(n,d)$ and $A(n,d,w)$. Inequalities (ii), (iii) and (iv) have been show in [20, Theorem 8], [20, Corollary 5] and [20, Theorem 12], respectively.

Theorem 22 ([20]). *We have*

(i) $A(n,d,w) \leq A(n,d)$,

(ii) $A(n,2\delta-1,w) = A(n,2\delta,w)$,

(iii) $A(n,2\delta,w) \leq \left\lfloor \frac{\delta}{b} \right\rfloor$, *if* $b \geq \frac{\delta}{n}$ *where* $b = \delta - \frac{w(n-w)}{n}$ *and*

(iv) $A(n,2\delta,w) \leq \frac{\binom{n}{k}}{\binom{w}{k}}$ *where* $k = w-\delta+1$.

In the following theorem, we establish an upper bound for $|T(Y)|$ using $A(n,d)$ and $A(n,d,w)$ when we have two remote output words.

Theorem 23. *Let* $C \subseteq \mathbb{F}^n$ *be an* e*-error-correcting code in* $\mathbb{F}^n$ *and* $\mathbf{y}_0$ *and* $\mathbf{y}$ *be words of* Y *such that* $d(\mathbf{y}_0,\mathbf{y}) = 2e-2\ell+2+a$ *and* $0 \leq a \leq 4\ell-2$. *If* $t = e+\ell \geq 3\ell-1$, *then we have*

$$|T| \leq A\left(2e-2\ell+2+a, 2e-4\ell+3+2\left\lceil\frac{a}{2}\right\rceil\right)$$

and

$$|T| \leq A\left(2e-2\ell+2+a, 2e-4\ell+3+a, e-\ell+1+\left\lfloor\frac{a}{2}\right\rfloor\right).$$

Proof. Without loss of generality, we may assume that $\mathbf{y}_0 = \mathbf{0}$, $w(\mathbf{y}) = 2e-2\ell+2+a$ and $\operatorname{supp}(\mathbf{y}) = [1, w(\mathbf{y})]$. Notice that since the all-zero word is received as an output word, we may restrict our investigation to codewords with weight at most t. For a codeword $\mathbf{c} \in C \cap B_t(\mathbf{y}_0) \cap B_t(\mathbf{y})$, we use the following notation: $S_\mathbf{c} = \operatorname{supp}(\mathbf{c}) \cap \operatorname{supp}(\mathbf{y})$, $A_\mathbf{c} = \operatorname{supp}(\mathbf{c}) \setminus \operatorname{supp}(\mathbf{y})$ and $\mathbf{y}_\mathbf{c}$ is a word such that $\operatorname{supp}(\mathbf{y}_\mathbf{c}) = S_\mathbf{c}$. Moreover, we denote $V_\mathbf{c} = \left\lfloor \left| w(\mathbf{y}_\mathbf{c}) - \frac{w(\mathbf{y})}{2} \right| \right\rfloor$. In other words, if $w(\mathbf{y})$ is even, then $V_\mathbf{c}$ gives the difference of $w(\mathbf{y}_\mathbf{c})$ and $w(\mathbf{y})/2$, and if $w(\mathbf{y})$ is odd, then it gives the difference of $w(\mathbf{y}_\mathbf{c})$ and $\lfloor w(\mathbf{y})/2 \rfloor$ or $\lceil w(\mathbf{y})/2 \rceil$ whichever is closer. In order to show that $\max\{d(\mathbf{y}_0,\mathbf{y}_\mathbf{c}), d(\mathbf{y},\mathbf{y}_\mathbf{c})\} = \max\{w(\mathbf{y}_\mathbf{c}), w(\mathbf{y})-w(\mathbf{y}_\mathbf{c})\} = \left\lceil \frac{w(\mathbf{y})}{2} \right\rceil + V_\mathbf{c}$, we need to study the following two cases:

- If $w(\mathbf{y}_\mathbf{c}) \geq w(\mathbf{y})/2$, then $w(\mathbf{y}) - w(\mathbf{y}_\mathbf{c}) \leq w(\mathbf{y}) - w(\mathbf{y})/2 \leq w(\mathbf{y}_\mathbf{c})$ and
$$\max\{w(\mathbf{y}_\mathbf{c}), w(\mathbf{y})-w(\mathbf{y}_\mathbf{c})\} = w(\mathbf{y}_\mathbf{c}) = \lceil w(\mathbf{y})/2 \rceil + V_\mathbf{c}.$$
- If $w(\mathbf{y}_\mathbf{c}) < w(\mathbf{y})/2$, then $w(\mathbf{y}) - w(\mathbf{y}_\mathbf{c}) > w(\mathbf{y}) - w(\mathbf{y})/2 > w(\mathbf{y}_\mathbf{c})$ and
$$\max\{w(\mathbf{y}_\mathbf{c}), w(\mathbf{y})-w(\mathbf{y}_\mathbf{c})\} = w(\mathbf{y}) - w(\mathbf{y}_\mathbf{c}) = w(\mathbf{y}) - (\lfloor w(\mathbf{y})/2 \rfloor - V_\mathbf{c}) = \lceil w(\mathbf{y})/2 \rceil + V_\mathbf{c}.$$

Similarly, it can be shown that $\min\{d(\mathbf{y}_0,\mathbf{y}_\mathbf{c}), d(\mathbf{y},\mathbf{y}_\mathbf{c})\} = \min\{w(\mathbf{y}_\mathbf{c}), w(\mathbf{y})-w(\mathbf{y}_\mathbf{c})\} = \left\lfloor \frac{w(\mathbf{y})}{2} \right\rfloor - V_\mathbf{c}$. Moreover, we have $d(\mathbf{y}_0,\mathbf{c}) = |A_\mathbf{c}| + d(\mathbf{y}_0,\mathbf{y}_\mathbf{c})$ and $d(\mathbf{y},\mathbf{c}) = |A_\mathbf{c}| + d(\mathbf{y},\mathbf{y}_\mathbf{c})$. Furthermore, since $\max\{d(\mathbf{y}_0,\mathbf{c}), d(\mathbf{y},\mathbf{c})\} \leq t$, we have

$$|A_\mathbf{c}| \leq t - \left\lceil \frac{w(\mathbf{y})}{2} \right\rceil - V_c. \tag{1}$$

Assume then that $\mathbf{c}_1, \mathbf{c}_2 \in C \cap B_t(\mathbf{y}_0) \cap B_t(\mathbf{y})$ and $\mathbf{c}_1 \neq \mathbf{c}_2$. We may approximate the distance of $\mathbf{c}_1$ and $\mathbf{c}_2$ in following way:

$$d(\mathbf{c}_1,\mathbf{c}_2) \leq |A_{\mathbf{c}_1}| + |A_{\mathbf{c}_2}| + d(\mathbf{y}_{\mathbf{c}_1},\mathbf{y}_{\mathbf{c}_2}). \tag{2}$$

Now, by estimating the right side of Inequality (2) with Inequality (1) and the left side of Inequality (2) by recalling $d(\mathbf{c}_1,\mathbf{c}_2) \geq 2e+1$, we get the following lower bound for $d(\mathbf{y}_{\mathbf{c}_1},\mathbf{y}_{\mathbf{c}_2})$ (as $w(\mathbf{y}) = 2e-2\ell+2+a$):

$$d(\mathbf{y}_{\mathbf{c}_1},\mathbf{y}_{\mathbf{c}_2}) \geq 2\left\lceil \frac{w(\mathbf{y})}{2} \right\rceil + 1 - 2\ell + V_{\mathbf{c}_1} + V_{\mathbf{c}_2} = 2e-4\ell+3+2\lceil a/2 \rceil + V_{\mathbf{c}_1} + V_{\mathbf{c}_2}. \tag{3}$$

Observe that when $e \geq 2\ell-1$ this lower bound is positive and $\mathbf{y}_{\mathbf{c}_1}$ and $\mathbf{y}_{\mathbf{c}_2}$ are distinct.

By Inequality (3), each pair of codewords in $B_t(\mathbf{y}_0) \cap B_t(\mathbf{y})$ differ in at least $2e-4\ell+3+2\left\lceil\frac{a}{2}\right\rceil$ coordinate positions of $\operatorname{supp}(\mathbf{y})$ (as $V_{\mathbf{c}_1}, V_{\mathbf{c}_2} \geq 0$). Thus, the words $\mathbf{y}_\mathbf{c}$ form a code with minimum distance $2e-4\ell+3+2\left\lceil\frac{a}{2}\right\rceil$ in $\mathbb{F}^{w(\mathbf{y})}$. Hence, we have $|T| \leq A\left(w(\mathbf{y}), 2e-4\ell+3+2\left\lceil\frac{a}{2}\right\rceil\right)$. This gives the first bound of the theorem. However, the bound does not take into account the values $V_{\mathbf{c}_1}$ and $V_{\mathbf{c}_2}$ in Inequality (3). In what follows, we try to improve the previous bound by making use of $V_{\mathbf{c}_1}$ and $V_{\mathbf{c}_2}$.

Let us define $C' = \{\mathbf{c} \in \mathbb{F}^{w(\mathbf{y})} \mid \mathbf{c}' \in C \cap B_t(\mathbf{y}_0) \cap B_t(\mathbf{y}), \operatorname{supp}(\mathbf{c}) = S_{\mathbf{c}'}\}$, that is, the code $C' \subseteq \mathbb{F}^{w(\mathbf{y})}$ is formed by taking each codeword in $B_t(\mathbf{y}_0) \cap B_t(\mathbf{y})$ and then restricting their support to $\operatorname{supp}(\mathbf{y})$. Therefore, as $d(\mathbf{y}_{\mathbf{c}_1},\mathbf{y}_{\mathbf{c}_2}) > 0$ by Inequality (3), we have $|C'| = |C \cap B_t(\mathbf{y}_0) \cap B_t(\mathbf{y})|$. The proof now divides into two cases depending on the parity of $w(\mathbf{y})$.

Suppose first that $w(\mathbf{y})$ is even, that is, a is even. Based on C', form a new code D as follows:

- If $\mathbf{c} \in C'$ and $w(\mathbf{c}) = w(\mathbf{y})/2$, then add $\mathbf{c}$ to D.
- If $\mathbf{c} \in C'$ and $w(\mathbf{c}) > w(\mathbf{y})/2$, then delete $V_{\mathbf{c}}$ elements from the support $\text{supp}(\mathbf{c})$ and add the resulting word of weight $w(\mathbf{y})/2$ to D.
- If $\mathbf{c} \in C'$ and $w(\mathbf{c}) < w(\mathbf{y})/2$, then add $V_{\mathbf{c}}$ elements to the support $\text{supp}(\mathbf{c})$ and add the resulting word of weight $w(\mathbf{y})/2$ to D.

Assume that $\mathbf{c}_1'$ and $\mathbf{c}_2'$ are codewords of D and that they have been respectively formed from the codewords $\mathbf{c}_1$ and $\mathbf{c}_2$ of C'. By Inequality (3), we obtain that $d(\mathbf{c}_1', \mathbf{c}_2') \geq d(\mathbf{c}_1, \mathbf{c}_2) - V_{\mathbf{c}_1} - V_{\mathbf{c}_2} \geq 2e - 4\ell + 3 + a > 0$ when $e \geq 2\ell - 1$. Thus, D is a code with minimum distance (at least) $w(\mathbf{y}) + 1 - 2\ell$ and $|C'| = |D|$. Therefore, we have $|T| \leq |C'| = |D| \leq A\left(w(\mathbf{y}), 2e - 4\ell + 3 + a, \frac{w(\mathbf{y})}{2}\right) = A\left(w(\mathbf{y}), 2e - 4\ell + 3 + a, e - \ell + 1 + \lfloor \frac{a}{2} \rfloor\right)$.

Suppose then that $w(\mathbf{y})$ is odd, that is, a is odd. As in the previous case, form a code D based on C' as follows:

- If $\mathbf{c} \in C'$ and $w(\mathbf{c}) \geq \lceil w(\mathbf{y})/2 \rceil$, then delete $V_{\mathbf{c}} + 1$ elements from the support $\text{supp}(\mathbf{c})$ and add the resulting word of weight $\lfloor w(\mathbf{y})/2 \rfloor$ to D.
- If $\mathbf{c} \in C'$ and $w(\mathbf{c}) \leq \lfloor w(\mathbf{y})/2 \rfloor$, then delete $V_{\mathbf{c}}$ elements from the support $\text{supp}(\mathbf{c})$ and add the resulting word of weight $\lfloor w(\mathbf{y})/2 \rfloor$ to D.

Thus, the resulting code D contains words of weight $\lfloor w(\mathbf{y})/2 \rfloor$. Assume that $\mathbf{c}_1'$ and $\mathbf{c}_2'$ are codewords of D and that they have been respectively formed from the codewords $\mathbf{c}_1$ and $\mathbf{c}_2$ of C'. By Inequality (3) and recalling the additional element deleted in the former case of the construction of D, we obtain that $d(\mathbf{c}_1', \mathbf{c}_2') \geq d(\mathbf{c}_1, \mathbf{c}_2) - V_{\mathbf{c}_1} - V_{\mathbf{c}_2} - 2 \geq 2e - 4\ell + 2 + a > 0$ when $e \geq 2\ell - 1$. Thus, D is a code with minimum distance (at least) $2e - 4\ell + 2 + a$ and $|C'| = |D|$. Therefore, we have $|T| \leq |C'| = |D| \leq A\left(w(\mathbf{y}), 2e - 4\ell + 2 + a, \lfloor \frac{w(\mathbf{y})}{2} \rfloor\right) = A\left(w(\mathbf{y}), 2e - 4\ell + 3 + a, e - \ell + 1 + \lfloor \frac{a}{2} \rfloor\right)$ (where the last equality is due to Theorem 22(ii)). □

Notice that in the proof of the previous theorem, in the case of odd a, we actually have a *two-weight code*, that is, a code where every codeword has either weight w_1 or w_2. Then, in order to obtain a constant weight code, the two-weight code is slightly modified. Hence, it might be possible to gain a slight improvement on the bound by investigating two-weight codes. Observe that in the proof of Theorem 4 we have actually considered a two-weight code (the set of words β_i).

In what follows, we give a few corollaries of the previous theorem. For this purpose, we first make the following simple observation: if k, k' and m are nonnegative integers such that $k \geq k'$, then

$$\frac{k}{k'} = \frac{k + m(k/k')}{k' + m} \geq \frac{k + m}{k' + m}. \tag{4}$$

Now we are ready to present the first corollary.

Corollary 24. *If $e \geq 3\ell - 2$, $C \subseteq \mathbb{F}^n$ is an e-error-correcting code and $d(\mathbf{y}_0, \mathbf{y}) \geq 2e - 2\ell + 2$ with $\mathbf{y}, \mathbf{y}_0 \in Y$, then we have*

$$|T| \leq 2 \left\lfloor \frac{2e - 4\ell + 4}{2e - 6\ell + 5} \right\rfloor.$$

Proof. Let a be an integer such that $d(\mathbf{y}_0, \mathbf{y}) = 2e - 2\ell + 2 + a$ and $0 \leq a \leq 4\ell - 2$. By Theorem 23, we have $|T| \leq A(2e - 2\ell + 2 + a, 2e - 4\ell + 3 + 2\lceil a/2 \rceil)$. Since $e \geq 3\ell - 2$, it can be straightforwardly verified that the requirement $n < 2d + 1$ of the Plotkin bound is satisfied. Now the proof divides into the following two cases depending on the parity of a:

- Suppose that a is even. By the Plotkin bound and Observation (4), we obtain that
$$|T| \leq A(2e - 2\ell + 2 + a, 2e - 4\ell + 3 + a) \leq 2 \left\lfloor \frac{2e - 4\ell + 4 + a}{2e - 6\ell + 5 + a} \right\rfloor \leq 2 \left\lfloor \frac{2e - 4\ell + 4}{2e - 6\ell + 5} \right\rfloor.$$
- Suppose that a is odd. By the Plotkin bound and Observation (4), we obtain that
$$|T| \leq A(2e - 2\ell + 2 + a, 2e - 4\ell + 4 + a) \leq 2 \left\lfloor \frac{2e - 4\ell + 5 + a}{2e - 6\ell + 7 + a} \right\rfloor \leq 2 \left\lfloor \frac{2e - 4\ell + 4}{2e - 6\ell + 5} \right\rfloor.$$

Thus, the claim follows. □

When $e \geq 4\ell - 2$, the previous corollary implies the following result.

Corollary 25. *If $e \geq 4\ell - 2$, $C \subseteq \mathbb{F}^n$ is an e-error-correcting code and $d(\mathbf{y}_0, \mathbf{y}) \geq 2e - 2\ell + 2$ with $\mathbf{y}_0, \mathbf{y} \in Y$, then we have*

$$|T| \leq 2.$$

Proof. Since $e \geq 4\ell - 2$, we obtain by the previous corollary and Observation (4) that

$$|T| \leq 2 \left\lfloor \frac{2e - 4\ell + 4}{2e - 6\ell + 5} \right\rfloor \leq 2 \left\lfloor \frac{2(4\ell - 2) - 4\ell + 4}{2(4\ell - 2) - 6\ell + 5} \right\rfloor = 2 \left\lfloor \frac{4\ell}{2\ell + 1} \right\rfloor = 2.$$

Hence, the claim follows. □

The previous corollaries have been obtained by applying the Plotkin bound to Theorem 23. In some cases, this can be improved by considering constant weight codes and Theorem 22(iii).

Corollary 26. *If $e \geq 3\ell - 2$, $C \subseteq \mathbb{F}^n$ is an e-error-correcting code and $d(\mathbf{y}_0, \mathbf{y}) \geq 2e - 2\ell + 2$ with $\mathbf{y}, \mathbf{y}_0 \in Y$, then we have*

$$|T| \leq 2\left\lfloor \frac{\ell+1}{2} \right\rfloor$$

if a is odd and

$$|T| \leq 2\ell$$

if a is even.

Proof. Let a be an integer such that $d(\mathbf{y}_0, \mathbf{y}) = 2e - 2\ell + 2 + a$ and $0 \leq a \leq 4\ell - 2$. Based on the parity of a, the proof divides into the following cases:

- Suppose that a is odd. By Theorem 23, we have
$$|T| \leq A(2e - 2\ell + 2 + a, 2e - 4\ell + 3 + 2\lceil a/2 \rceil) = A(2e - 2\ell + 2 + a, 2e - 4\ell + 4 + a).$$
Further, by the Plotkin bound and Observation (4), we obtain that
$$|T| \leq 2\left\lfloor \frac{2e - 4\ell + 5 + a}{2e - 6\ell + 7 + a} \right\rfloor \leq 2\left\lfloor \frac{2(3\ell-2) - 4\ell + 5 + a}{2(3\ell-2) - 6\ell + 7 + a} \right\rfloor \leq 2\left\lfloor \frac{2\ell+2}{4} \right\rfloor = 2\left\lfloor \frac{\ell+1}{2} \right\rfloor.$$
- Suppose that a is even. By Theorems 23 and 22(ii), we have
$$|T| \leq A(2e-2\ell+2+a, 2e-4\ell+3+a, e-\ell+1+a/2) = A(2e-2\ell+2+a, 2e-4\ell+4+a, e-\ell+1+a/2) = A(n, 2\delta, w),$$
where $n = 2w$, $w = e - \ell + 1 + a/2$ and $\delta = e - 2\ell + 2 + a/2$. In order to apply Theorem 22(iii), we observe that
$$b = \delta - \frac{w(n-w)}{n} = \delta - \frac{w}{2} = \frac{2e - 6\ell + 6 + a}{4} \geq \frac{2(3\ell-2) - 6\ell + 6 + a}{4} \geq \frac{1}{2}$$
and
$$\frac{\delta}{n} = \frac{1}{2} - \frac{\ell - 1}{n} \leq \frac{1}{2}.$$
Therefore, as $b \geq \delta/n$, we obtain by Theorem 22(iii) and Observation (4) that
$$|T| \leq A(n, 2\delta, w) = \left\lfloor \frac{\delta}{b} \right\rfloor = 1 + \left\lfloor \frac{e - \ell + 1 + a/2}{e - 3\ell + 3 + a/2} \right\rfloor \leq 1 + \left\lfloor \frac{(3\ell-2) - \ell + 1}{(3\ell-2) - 3\ell + 3} \right\rfloor = 2\ell.$$

Thus, the claim follows. □

In the three corollaries above, we have considered the cases with $e \geq 3\ell - 2$. However, Theorem 23 already holds for $e \geq 2\ell - 1$. In the following corollary, we complete this gap.

Corollary 27. *If $e \geq 2\ell - 1$, $C \subseteq \mathbb{F}^n$ is an e-error-correcting code and $d(\mathbf{y}_0, \mathbf{y}) = 2e - 2\ell + 2 + a$ with $\mathbf{y}, \mathbf{y}_0 \in Y$ and $0 \leq a \leq 4\ell - 2$, then we have*

$$|T| \leq \frac{\binom{2e-2\ell+2+a}{\ell}}{\binom{e-\ell+1+\lfloor \frac{a}{2} \rfloor}{\ell}}.$$

Proof. Let a be an integer such that $d(\mathbf{y}_0, \mathbf{y}) = 2e - 2\ell + 2 + a$ and $0 \leq a \leq 4\ell - 2$. Theorem 23 gives $|T| \leq A\left(2e - 2\ell + 2 + a, 2e - 4\ell + 3 + a, e - \ell + 1 + \lfloor a/2 \rfloor\right)$. The proof now divides into the following two cases depending on the parity of a:

- Suppose that a is even. By Theorem 22(ii), we have $|T| \leq A(2e-2\ell+2+a, 2e-4\ell+4+a, e-\ell+1+a/2) = A(n, 2\delta, w)$, where $n = 2e - 2\ell + 2 + a$, $\delta = e - 2\ell + 2 + a/2$ and $w = e - \ell + 1 + a/2$. Now $k = w - \delta + 1 = \ell$. Hence, by Theorem 22$(iv)$, we obtain that
$$|T| \leq \frac{\binom{2e-2\ell+2+a}{\ell}}{\binom{e-\ell+1+\lfloor a/2 \rfloor}{\ell}}.$$
- Suppose that a is odd. Now we have $|T| \leq A(2e - 2\ell + 2 + a, 2e - 4\ell + 3 + a, e - \ell + 1 + (a-1)/2) = A(n, 2\delta, w)$, where $n = 2e - 2\ell + 2 + a$, $\delta = e - 2\ell + 2 + (a-1)/2$ and $w = e - \ell + 1 + (a-1)/2$. Now $k = w - \delta + 1 = \ell$. Hence, by Theorem 22$(iv)$, we obtain that
$$|T| \leq \frac{\binom{2e-2\ell+2+a}{\ell}}{\binom{e-\ell+1+\lfloor a/2 \rfloor}{\ell}}.$$

Thus, the claim follows. □

Notice that if $e = 2\ell - 1$, then we are very likely to have two output words with distance at least 2ℓ by Theorem 20 (or we have $|T| = 1$). Earlier, in Corollary 3, we have shown that if $N \geq \sum_{i=0}^{\ell-1} \binom{n}{i} + 1$, then we have two output words with distance at least $2\ell - 1$. In Theorem 4, this is applied to give the upper bound $|T| \leq \binom{2\ell}{\ell}$. Observe that Corollary 27 also gives upper bound $\binom{2\ell}{\ell}$ when $e = 2\ell - 1$ and $a = 0$.

V. LESS THAN $V(n, \ell - 1) + 1$ CHANNELS

In this section, we investigate some cases with $N \leq V(n, \ell - 1)$. In the following, we consider the asymptotic behaviour of $\mathcal{L}$ for different values of N when e and ℓ are constants and $C \subseteq \mathbb{F}^n$ is such an e-error-correcting code that $\mathcal{L}$ is maximal. First we give an upper bound on $\mathcal{L}$ and then a lower bound.

Lemma 28. *Let $N \geq V(n, \ell - a - 1) + 1$ where $0 \leq a \leq \ell - 1$. Then for any e-error-correcting code $C \subseteq \mathbb{F}^n$, we have*

$$\mathcal{L} \leq 2^{\ell - a} \sum_{i=0}^{a} \binom{n - e - \ell + a}{i}.$$

Proof. Let $\mathbf{x}$ be the input word and Y be the set of output words. By Theorem 5 there exists a set $S \subseteq [1, n]$ of size $\ell - a$ which is shattered by some set $Y' \subseteq Y$ such that $|Y'| = 2^{\ell - a}$. Without loss of generality, we may assume that $S = [1, \ell - a]$. Moreover, let $\mathbf{s}$ be such a word that $\operatorname{supp}(\mathbf{s}) = S$. Furthermore, as in the proof of Theorem 6, for each $\mathbf{y}_i \in Y'$, $1 \leq i \leq 2^{\ell - a}$, let $\beta_i = \mathbf{y}_i + \mathbf{s}$.

Let β_j, $j \in [1, 2^{\ell - a}]$, be such a word that $\operatorname{supp}(\mathbf{x}) \cap S = \operatorname{supp}(\beta_j) \cap S$. Since $d(\mathbf{x}, \mathbf{y}_j) \leq e + \ell$, we have $d(\mathbf{x}, \beta_j) \leq e + a$. Notice that if $d(\mathbf{x}, \beta_j) = e + a' \leq e + a$, then $\mathbf{x}$ and β_j differ in $e + a'$ coordinates in the set $[\ell - a + 1, n]$. Hence, we may consider words $\beta_j + \mathbf{w}_h$ where $0 \leq h \leq \sum_{i=0}^{a} \binom{n - e - \ell + a}{i}$, $\operatorname{supp}(\mathbf{w}_h) \in [\ell - a + 1, n - e]$ and $0 \leq w(\mathbf{w}_h) \leq a$. Since $d(\mathbf{x}, \beta_j) = e + a' \leq e + a$ one of the words $\mathbf{w}_h$, say $\mathbf{w}_{h'}$, corresponds to a' differences between $\mathbf{x}$ and β_j (or to 0 differences if a' is negative), i.e., $\operatorname{supp}(\mathbf{w}_{h'}) \subseteq \operatorname{supp}(\mathbf{x} + \beta_j)$ and $w(\mathbf{w}_{h'}) = a'$. Hence, we have $d(\mathbf{x}, \beta_j + \mathbf{w}_{h'}) \leq e$. ☐

Lemma 29. *Let $N \leq V(n, \ell - a)$ where $0 \leq a \leq \ell$ and $n \geq 2e + a$. Then there exists such an e-error-correcting code $C \subseteq \mathbb{F}^n$ that*

$$\mathcal{L} \geq \frac{\binom{n}{e+a}}{\sum_{i=0}^{e} \binom{e+a}{i} \binom{n-e-a}{i}} \geq \frac{n^a}{(e+a)^a \sum_{i=0}^{e} \binom{e+a}{i}}.$$

Proof. Let $S = \{\mathbf{w} \in \mathbb{F}^n \mid w(\mathbf{w}) \leq \ell - a\}$ and $Y \subseteq S$. We immediately notice that if $w(\mathbf{c}) \leq e + a$, then $\mathbf{c} \in \bigcap_{\mathbf{y} \in Y} B_t(\mathbf{y})$. Let us now consider a maximal e-error-correcting code C with constant weight $e + a$. By [21, Theorem 7] (Gilbert bound for constant weight codes) and Theorem 22(ii), we have $\mathcal{L} \geq |C| = A(n, 2e+1, e+a) = A(n, 2e+2, e+a) \geq \frac{\binom{n}{e+a}}{\sum_{i=0}^{e} \binom{e+a}{i} \binom{n-e-a}{i}}$. Furthermore, we may estimate

$$\begin{aligned}
&\frac{\binom{n}{e+a}}{\sum_{i=0}^{e} \binom{e+a}{i} \binom{n-e-a}{i}} \\
\geq &\frac{\binom{n}{e+a}}{\binom{n-a}{e} \sum_{i=0}^{e} \binom{e+a}{i}} \\
= &\frac{n! e! (n-e-a)!}{(n-a)! (e+a)! (n-e-a)! \sum_{i=0}^{e} \binom{e+a}{i}} \\
\geq &\frac{n^a}{(e+a)^a \sum_{i=0}^{e} \binom{e+a}{i}}.
\end{aligned}$$

The last inequality is due to Observation (4). ☐

In the following theorem, we give an asymptotic estimate for $\mathcal{L}$ with exact dependency on N.

Theorem 30. *Let $V(n, \ell - a - 1) + 1 \leq N \leq V(n, \ell - a)$ where $0 \leq a \leq \ell - 1$. Moreover, let $C \subseteq \mathbb{F}^n$ be such an e-error-correcting code that $\mathcal{L}$ is maximal. Then we have*

$$\mathcal{L} = \Theta(n^a).$$

Proof. Let $V(n, \ell - a - 1) + 1 \leq N \leq V(n, \ell - a)$. By Lemma 28 we have $\mathcal{L} \leq 2^{\ell - a} \sum_{i=0}^{a} \binom{n - e - \ell + a}{i} \leq 2^{\ell - a} (a+1) \frac{n^a}{a!}$ (for $n \geq 2a$). Since e, ℓ and a are constants, the claim follows by Lemma 29. ☐

Although we have shown that there exist such e-error-correcting codes that $\mathcal{L}$ is rather large when we have less than $V(n, \ell - 1) + 1$ channels, we may also construct such rather large e-error-correcting codes that $\mathcal{L}$ is constant on n when $N \geq V(n, \ell - 2) + 1$.

Theorem 31. *For any $t = e+\ell$, there exist e-error-correcting codes $C \subseteq \mathbb{F}_2^n$ of length $n = 2^m - 1$, where $m > \log_2(e+1)!+3$, and of size at least $2^{2^m-(e+1)m}$ with*

$$\mathcal{L} \leq 2^\ell$$

and

$$N \geq V(n, \ell-2)+1.$$

Proof. Let us first consider a primitive narrow-sense BCH code C with designed distance $2(e+1)+1$, that is, error-correcting capability at least $e+1$ [18, p. 203]. Let α be a primitive element of the finite field $\mathbb{F}_{2^m}$. It is also a primitive nth root of unity of the field. The BCH code is defined by

$$C = \{c(x) \in R \mid c(\alpha) = c(\alpha^3) = \cdots = c(\alpha^{2e+1}) = 0\}$$

where the ring $R = \mathbb{F}_2[x]/\langle x^n - 1\rangle$. Hence, $C = \{\mathbf{x} \in \mathbb{F}_2^n \mid H\mathbf{x}^T = 0\}$ where

$$H = \begin{pmatrix} 1 & \alpha & \dots & \alpha^{n-1} \\ 1 & \alpha^3 & \dots & \alpha^{3(n-1)} \\ \vdots & & & \vdots \\ 1 & \alpha^{2e+1} & \dots & \alpha^{(2e+1)(n-1)} \end{pmatrix}.$$

Let now $\{\gamma_1, \dots, \gamma_m\}$ be a basis of the field extension $\mathbb{F}_{2^m}$ over $\mathbb{F}_2$. Thus, we can write the elements α^i in H with the aid of the basis as column vectors. Consequently, we obtain an $(e+1)m \times n$-matrix H_2 with entries in $\mathbb{F}_2$ such that $C = \{\mathbf{x} \in \mathbb{F}_2^n \mid H_2\mathbf{x}^T = 0\}$.

Let us write $t = (e+1) + (\ell-1)$. Due to Theorem 6, we know that for any set of at least $V(n, \ell-2)+1$ outputs $Y = \{\mathbf{y}_1, \dots, \mathbf{y}_N\}$ we have

$$|\bigcap_{\mathbf{y}\in Y} B_t(\mathbf{y}) \cap C| \leq 2^{\ell-1} \tag{5}$$

for radius t. Notice that the error-correcting capability of C can be better than $e+1$, but in that case we get a code with even better parameters by writing $t = (e+i) + (\ell-i)$.

Since $m > 2\log_2(2e+1)$, the rows of H_2 are linearly independent [18, p. 263]. Let us delete suitable linearly independent rows among the rows in the matrix H_2 which correspond to the row

$$R = (1, \alpha^{2e+1}, \dots, \alpha^{(2e+1)(n-1)})$$

of the matrix H. We delete the smallest number of rows, say p rows ($p \leq m$), in such a way that the obtained matrix H' gives us a code $C' = \{\mathbf{x} \in \mathbb{F}_2^n \mid H'\mathbf{x}^T = 0\}$ with error-correcting capability exactly e. Notice that the error-correcting capability of C' is at least e. Indeed, since $m > \log_2(e+1)!+1$, we know [18, p. 259] that the code corresponding to H without the row R has error-correcting capability exactly e and the corresponding rows in H' remain intact. Let C_2 be the code which is obtained just before C', that is, using the matrix where we have deleted only $p-1$ rows from H_2. Now the error-correcting capability of C_2 is at least $e+1$, so it has a list size at most $2^{\ell-1}$ according to (5). Due to the fact that the code C' consists of C_2 and one of its cosets, the code C' has list size at most $2 \cdot 2^{\ell-1} = 2^\ell$. For the cardinality we have [18, p. 203]

$$|C'| \geq 2^{2^m - m(e+1)}.$$

□

We may improve the previous theorem for suitable values of e and ℓ by using Theorem 17.

Theorem 32. *For any $t = e+\ell$ such that $e \geq \ell$ and $e \geq 7$, there exist e-error-correcting codes $C \subseteq \mathbb{F}_2^n$ of length $n = 2^m - 1$, where $m > \log_2(e+1)!+3$, and of size at least $2^{2^m-(e+1)m}$ with*

$$\mathcal{L} \leq 2\ell$$

and

$$N \geq V(n, \ell-2)+1.$$

Proof. Consider the $(e+1)$-error-correcting code and Inequality (5) in the proof of the previous theorem together with Theorem 17 instead of Theorem 6. Since $e \geq 7$ and $e \geq \ell$, we have $3t \leq 6e$ and $4(e+1)+4 \leq 6e$. Therefore, we may choose $b = 6e$ in Theorem 17. Together with the notation $t = (e+1)+(\ell-1)$, we get

$$(\ell-2)^2(5e-1+(e+2)(12e^2-9e+1))+\ell-2$$

for the lower bound of n in Theorem 17. Moreover, since $e \geq 7$, we have

$$n > 2^{\log_2(e+1)!+3} - 1 = 8 \cdot (e+1)! - 1 \geq (\ell-2)^2(5e-1+(e+2)(12e^2-9e+1))+\ell-2.$$

Hence, n satisfies the requirements in Theorem 17 and we may modify Inequality (5) to

$$|\bigcap_{\mathbf{y}\in Y} B_t(\mathbf{y}) \cap C| \leq \ell$$

and thus, the code C' in the proof of Theorem 31 has list size at most $2 \cdot \ell$ instead of $2 \cdot 2^{\ell-1}$. ☐

Naturally, we can get corresponding results for shorter lengths than $n = 2^m - 1$ by applying the shortening method [18, p. 29] to the code C' in the proof above provided that the minimum distance of the code does not increase.

Example 33. Consider first a 2-error-correcting primitive and narrow-sense BCH code C_1 of length $n = 15$. By Theorem 6, we know that for $t = 4$ (so $\ell = 2$) using at least $N = 17$ channels C provides us a code with list size $\mathcal{L} \leq 4$. With the method of Theorem 31 we get a code C' with $\mathcal{L} \leq 4$ for $t = 4$ when we have only $N = 2$ channels! Notice that although here m does not satisfy $m > \log_2(2+1)! + 3$, we have one linearly independent row to delete from H_2 to get C'. The price we pay for this is that C_1 has 128 codewords and C' has 64.

APPENDIX

PROOF OF LEMMA 15

Here, in the appendix, we give the rather technical proof of Lemma 15. For the proof, we first present an auxiliary result.

Lemma 34. *Let $b \leq n$ be a positive integer and $\mathbf{c}_0$, $\mathbf{c}_1$, $\mathbf{c}_2$, $\mathbf{c}'_0$, $\mathbf{c}'_1$ and $\mathbf{c}'_2$ be words of $\mathbb{F}^n$ such that $d(\mathbf{c}_i, \mathbf{c}_j) \geq d(\mathbf{c}'_i, \mathbf{c}'_j)$ for all i, j. Further, let $\overline{D}$ and $\overline{D'}$ be such subsets of $[1, n]$ that $|\overline{D}| = |\overline{D'}| = b$, supp$(\mathbf{c}_i + \mathbf{c}_j) \subseteq \overline{D}$ and supp$(\mathbf{c}'_i + \mathbf{c}'_j) \subseteq \overline{D'}$ for $i, j \in \{0, 1, 2\}$. If*

$$S = \{\mathbf{w} \in \mathbb{F}^n \mid \mathbf{w} \in \bigcap_{i=0}^{2} B_t(\mathbf{c}_i) \text{ and } |\mathit{supp}(\mathbf{w} + \mathbf{c}_0) \setminus \overline{D}| < \ell - 1\}$$

and

$$S' = \{\mathbf{w} \in \mathbb{F}^n \mid \mathbf{w} \in \bigcap_{i=0}^{2} B_t(\mathbf{c}'_i) \text{ and } |\mathit{supp}(\mathbf{w} + \mathbf{c}'_0) \setminus \overline{D'}| < \ell - 1\},$$

then we have $|S| \leq |S'|$.

Proof. Since only the cardinalities of S and S' are considered and the distances between $\mathbf{c}_i$'s and the distances between $\mathbf{c}'_i$'s do not depend on each other, we may assume without loss of generality that $\mathbf{c}_0 = \mathbf{c}'_0 = \mathbf{0}$ and $\overline{D} = \overline{D'} = [1, b]$. Notice that supp$(\mathbf{w} + \mathbf{c}_0) \setminus \overline{D} =$ supp$(\mathbf{w} + \mathbf{c}_1) \setminus \overline{D} =$ supp$(\mathbf{w} + \mathbf{c}_2) \setminus \overline{D}$ for any $\mathbf{w} \in \mathbb{F}^n$. Hence, $\mathbf{c}_0$ and $\mathbf{c}'_0$ could be replaced in the definitions of S and S' by any $\mathbf{c}_i$ and $\mathbf{c}'_i$ with $i \in \{1, 2\}$, respectively. Consider first intersections among the coordinates in $\overline{D}$. For this purpose, let $\mathbf{c}_{Di}$ and $\mathbf{c}'_{Di}$ be words of $\mathbb{F}^b$ such that supp$(\mathbf{c}_{Di}) =$ supp$(\mathbf{c}_i)$ and supp$(\mathbf{c}'_{Di}) =$ supp$(\mathbf{c}'_i)$ for $i \in \{0, 1, 2\}$. Notice that $\mathbf{c}_{Di}$ and $\mathbf{c}'_{Di}$ preserve the distances, that is, $d(\mathbf{c}_{Di}, \mathbf{c}_{Dj}) = d(\mathbf{c}_i, \mathbf{c}_j)$ and $d(\mathbf{c}'_{Di}, \mathbf{c}'_{Dj}) = d(\mathbf{c}'_i, \mathbf{c}'_j)$. Then denote $S_h^b = B_h(\mathbb{F}^b; \mathbf{c}_{D0}) \cap B_h(\mathbb{F}^b; \mathbf{c}_{D1}) \cap B_h(\mathbb{F}^b; \mathbf{c}_{D2})$ and $S'^b_h = B_h(\mathbb{F}^b; \mathbf{c}'_{D0}) \cap B_h(\mathbb{F}^b; \mathbf{c}'_{D1}) \cap B_h(\mathbb{F}^b; \mathbf{c}'_{D2})$. Now we have $|{S'_h}^b| \geq |{S_h}^b|$ by [14, Theorem 2.4.10].

Now, we are ready to determine $|S|$ and $|S'|$. Notice that $|S^b_{t-i}|$ is equal to the number of words $\mathbf{y} \in S$ such that supp$(\mathbf{y}) \setminus \overline{D}$ is fixed and contains $i \leq \ell - 2$ elements. Therefore, we obtain that $|S| = \sum_{i=0}^{\ell-2} \binom{n-b}{i} |S^b_{t-i}|$. Similarly, we get $|S'| = \sum_{i=0}^{\ell-2} \binom{n-b}{i} |S'^b_{t-i}|$. Thus, as $|{S_h}^b| \leq |{S'_h}^b|$, we have $|S| \leq |S'|$. ☐

Now we are ready to present the proof of Lemma 15.

Proof. Without loss of generality, we may assume that $\mathbf{c}_0 = \mathbf{0}$ and $\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2 \in C_1 \cap T(Y)$ where C_1 is an e-error-correcting code. We may choose three such codewords since $|T(Y)| \geq 3$ by assumption. Furthermore, let $D = [b+1, n]$, $\overline{D} = [1, b]$ with $b \geq 3t$ and supp$(\mathbf{c}_i) \subseteq \overline{D}$, $i \in \{1, 2\}$. Notice that $|$supp$(\mathbf{c}_1) \cup$ supp$(\mathbf{c}_2)| \leq 3t$ since $d(\mathbf{c}_i, \mathbf{c}_j) \leq 2t$ for $i, j \in \{0, 1, 2\}$. To show that $|$supp$(\mathbf{y}) \setminus \overline{D}| \geq \ell - 1$, we prove that $N > |\{\mathbf{w} \in \mathbb{F}^n \mid \mathbf{w} \in \bigcap_{i=0}^{2} B_t(\mathbf{c}_i)$ and $|$supp$(\mathbf{w}) \setminus \overline{D}| < \ell - 1\}|$. Due to Lemma 34, we may assume that distances $d(\mathbf{c}_i, \mathbf{c}_j) \in \{2e+1, 2e+2\}$ for $i \neq j$, $i, j \in \{0, 1, 2\}$ and only $d(\mathbf{c}_1, \mathbf{c}_2) = 2e + 2$.

Let

$$\begin{aligned} A &= \text{supp}(\mathbf{c}_1) \cap \text{supp}(\mathbf{c}_2), \\ B &= \text{supp}(\mathbf{c}_1) \setminus \text{supp}(\mathbf{c}_2), \\ C' &= \text{supp}(\mathbf{c}_2) \setminus \text{supp}(\mathbf{c}_1) \end{aligned}$$

and

$$E = \overline{D} \setminus (\text{supp}(\mathbf{c}_1) \cup \text{supp}(\mathbf{c}_2)).$$

Notice that $A \cup B \cup C'$ contains every coordinate in which the words $\mathbf{c}_0$, $\mathbf{c}_1$ and $\mathbf{c}_2$ differ.

We have $|A| = e$, $|B| = |C'| = e+1$, $|E| = b - |A| - |B| - |C'|$, $|D| = n - b$. Moreover, let us denote with S the set $S = \{\mathbf{w} \in \mathbb{F}^n \mid \mathbf{w} \in \bigcap_{i=0}^{2} B_t(\mathbf{c}_i) \text{ and } |\text{supp}(\mathbf{w}) \setminus \overline{D}| < \ell - 1\}$. Let $\mathbf{y} \in S$ and $|\text{supp}(\mathbf{y}) \cap A| = i_1$, $|\text{supp}(\mathbf{y}) \cap B| = i_2$, $|\text{supp}(\mathbf{y}) \cap C'| = i_3'$, $|\text{supp}(\mathbf{y}) \cap D| = i_4$ and $|\text{supp}(\mathbf{y}) \cap E| = i_5$. Next we will approximate the cardinality of S. Our goal is to show that $|S| < V(n, \ell-1) + 1 \leq N$. Immediately, by the definition of S, we have

$$i_1 + i_2 + i_3' + i_4 + i_5 \leq t \text{ and } i_4 \leq \ell - 2. \tag{6}$$

Moreover, $d(\mathbf{y}, \mathbf{c}_i) \leq t$ for $i \in \{1,2\}$. Thus, $i_4 + i_5 + (|A| - i_1) + (|B| - i_2) + i_3' \leq t$ and $i_4 + i_5 + (|A| - i_1) + (|C'| - i_3') + i_2 \leq t$. By summing these two inequalities together and recalling that $|A| = e$, $|B| = |C'| = e+1$ and $t = e + \ell$, we get

$$e + 1 + i_4 + i_5 - \ell \leq i_1. \tag{7}$$

Notice that $i_1 \leq |A| = e$ and hence,

$$i_5 \leq \ell - 1 - i_4.$$

As we can see from the two inequalities $d(\mathbf{y}, \mathbf{c}_i) \leq t$ for $i \in \{1,2\}$, the value $|i_2 - i_3'|$ can be bounded from above. More precisely, $i_4 + i_5 + (|A| - i_1) + (e+1) + |i_2 - i_3'| \leq t$ and hence, $|i_2 - i_3'| \leq \ell - i_4 - i_5 - (|A| - i_1) - 1$. Furthermore, by (6) and because $i_3' \geq i_2 - |i_2 - i_3'|$, we have $t - i_1 - i_4 - i_5 \geq i_2 + i_3' \geq 2i_2 - |i_2 - i_3'| \geq 2i_2 + i_4 + i_5 + e + 1 - \ell - i_1$. Hence,

$$i_2 \leq \ell - 1 - i_4 - i_5. \tag{8}$$

We can bound i_3' from below using $i_3' \geq i_2 - |i_2 - i_3'|$ and from above using inequality $i_3' \leq i_2 + |i_2 - i_3'|$. Based on these inequalities we get

$$i_2 - (\ell - i_4 - i_5 - (|A| - i_1) - 1) \leq i_3' \leq i_2 + (\ell - i_4 - i_5 - (|A| - i_1) - 1).$$

Now we are ready to approximate $|S|$. Notice that we consider $\binom{p}{q} = 0$ for $q < 0$. Hence,

$$\begin{aligned}
|S| &\leq \sum_{i_4=0}^{\ell-2} \sum_{i_5=0}^{\ell-1-i_4} \sum_{i_1=i_4+i_5+e+1-\ell}^{e} \sum_{i_2=0}^{\ell-1-i_4-i_5} \sum_{i_3'=i_2+i_4+i_5+e+1-\ell-i_1}^{i_1+i_2+\ell-e-1-i_4-i_5} \binom{|D|}{i_4}\binom{|E|}{i_5}\binom{|A|}{i_1}\binom{|B|}{i_2}\binom{|C'|}{i_3'} \\
&\overset{(i)}{\leq} \sum_{i_4=0}^{\ell-2} \sum_{i_1=i_4+e+1-\ell}^{e} \sum_{i_2=0}^{\ell-1-i_4} \sum_{i_5=0}^{\ell-1-i_4} \sum_{i_3'=i_2+i_4+i_5+e+1-\ell-i_1}^{i_1+i_2+\ell-e-1-i_4-i_5} \binom{|D|}{i_4}\binom{|A|}{i_1}\binom{|B|}{i_2}\binom{|E|}{i_5}\binom{|C'|}{i_3'} \\
&\overset{(ii)}{\leq} \sum_{i_4=0}^{\ell-2} \sum_{i_1=i_4+e+1-\ell}^{e} \sum_{i_2=0}^{\ell-1-i_4} \sum_{i_3=i_2+i_4+e+1-\ell-i_1}^{i_1+i_2+\ell-e-1-i_4} \binom{|D|}{i_4}\binom{|A|}{i_1}\binom{|B|}{i_2}\binom{|C|}{i_3}.
\end{aligned}$$

To approximate $|S|$, we first omit the restrictions set by i_5 for i_1 and i_2 in (i). After that, in (ii), we denote $C = C' \cup E$ and combine the binomial sums considering i_3' and i_5. We can do this because on the left hand side we choose i_5 elements from E and i_3' elements from C' while on the right hand side we choose $i_3' + i_5$ elements from $C' \cup E$. To further estimate the binomial sum, we partition it into smaller pieces using the notations

$$G(i_4) = \sum_{i_1=i_4+e+1-\ell}^{e} \sum_{i_2=0}^{\ell-1-i_4} \sum_{i_3=i_2+i_4+e+1-\ell-i_1}^{i_1+i_2+\ell-e-1-i_4} \binom{|A|}{i_1}\binom{|B|}{i_2}\binom{|C|}{i_3} \tag{9}$$

and

$$g(i_4) = \binom{|D|}{i_4} G(i_4). \tag{10}$$

Hence,

$$|S| \leq \sum_{i_4=0}^{\ell-2} g(i_4) = \sum_{i_4=0}^{\ell-2} \binom{|D|}{i_4} G(i_4). \tag{11}$$

The goal of the proof is to first show that $g(i_4) \geq g(i_4 - 1)$ for $i_4 \leq \ell - 2$ and then calculate the value $G(\ell-2)$. Together these will give $|S| \leq (\ell-1)\binom{|D|}{\ell-2} G(\ell-2)$ which we can show to be less than N when n is large enough.

Moreover, let us denote:

$$\begin{aligned}
f'(i_4) =& \binom{|A|}{i_4+e-\ell} \sum_{i_2=0}^{\ell-i_4} \binom{|B|}{i_2}\binom{|C|}{i_2},\\
h'(i_4) =& \sum_{i_1=i_4+e+1-\ell}^{e} \binom{|A|}{i_1} \sum_{i_2=0}^{\ell-1-i_4} \binom{|B|}{i_2} \left(\binom{|C|}{i_1+i_2+\ell-e-i_4} + \binom{|C|}{i_2+i_4+e-\ell-i_1} \right)
\end{aligned}$$

and

$$s'(i_4) = \binom{|B|}{\ell-i_4} \sum_{i_1=i_4+e+1-\ell}^{e} \binom{|A|}{i_1} \sum_{i_3=e-i_1}^{i_1+2\ell-e-2i_4} \binom{|C|}{i_3}.$$

By comparing the sums $G(i_4)$ and $G(i_4-1)$, we notice that the sum $G(i_4-1)$ contains every term in $G(i_4)$. In addition, it may contain three different types of extremal terms; one where i_1 is one smaller than it is possible in $G(i_4)$, one where i_3 is either one greater or one smaller than in $G(i_4)$ and one where i_2 is one greater than in $G(i_4)$. Therefore, we have

$$G(i_4-1) \leq G(i_4) + f'(i_4) + h'(i_4) + s'(i_4). \tag{12}$$

The partial sum $f'(i_4)$ contains the terms with $i_1 = i_4+e-\ell$. Notice that in this case $|i_2-i_3| = 0$ and thus, $i_2 = i_3$. The partial sum $h'(i_4)$ considers the cases where $i_1 \geq i_4+e+1-\ell$, $i_2 \leq \ell-1-i_4$ and $i_3 \in \{i_1+i_2+\ell-e-i_4, i_2+i_4+e-\ell-i_1\}$. Finally, the partial sum $s'(i_4)$ consists of the cases with $i_1 \geq i_4+e+1-\ell$, $i_2 = \ell-i_4$ and $i_3 \in [e-i_1, i_1+2\ell-e-2i_4]$. Moreover, denote

$$\begin{aligned}
f(i_4) =& \binom{|A|}{i_4+e+1-\ell} \sum_{i_2=0}^{\ell-1-i_4} \binom{|B|}{i_2}\binom{|C|}{i_2},\\
h(i_4) =& \sum_{i_1=i_4+e+1-\ell}^{e} \binom{|A|}{i_1} \sum_{i_2=0}^{\ell-1-i_4} \binom{|B|}{i_2} \left(\binom{|C|}{i_1+i_2+\ell-e-1-i_4} + \binom{|C|}{i_2+i_4+e+1-\ell-i_1} \right)
\end{aligned}$$

and

$$s(i_4) = \binom{|B|}{\ell-1-i_4} \sum_{i_1=i_4+e+1-\ell}^{e} \binom{|A|}{i_1} \sum_{i_3=e-i_1}^{i_1+2\ell-e-2-2i_4} \binom{|C|}{i_3}.$$

Notice that $f(i_4)$, $h(i_4)$ and $s(i_4)$ are partial sums of $G(i_4)$. Moreover, we may use them to approximate the values $f'(i_4)$, $h'(i_4)$ and $s'(i_4)$. To estimate these sums, we use the identity

$$\binom{n}{a+1} = \frac{n-a}{a+1}\binom{n}{a}. \tag{13}$$

Let us first consider f'. We have

$$\begin{aligned}
f'(i_4) &= \binom{|A|}{i_4+e-\ell} \sum_{i_2=0}^{\ell-i_4} \binom{|B|}{i_2}\binom{|C|}{i_2}\\
&\overset{(13)}{=} \frac{i_4+e+1-\ell}{\ell-i_4} \binom{|A|}{i_4+e+1-\ell}\\
&\quad \cdot \left(\sum_{i_2=0}^{\ell-i_4-1} \binom{|B|}{i_2}\binom{|C|}{i_2} + \frac{i_4+|B|+1-\ell}{\ell-i_4} \cdot \frac{i_4+|C|+1-\ell}{\ell-i_4} \binom{|B|}{\ell-i_4-1}\binom{|C|}{\ell-i_4-1} \right)\\
&\overset{(iii)}{\leq} \frac{i_4+e+1-\ell}{\ell-i_4} \left(1 + \frac{i_4+|B|+1-\ell}{\ell-i_4} \cdot \frac{i_4+|C|+1-\ell}{\ell-i_4} \right) f(i_4)\\
&\overset{(iv)}{\leq} \frac{e(|B||C|+4)}{8} G(i_4).
\end{aligned}$$

In Step (iii), we estimate $\binom{|B|}{\ell-i_4-1}\binom{|C|}{\ell-i_4-1} \leq \sum_{i_2=0}^{\ell-i_4-1} \binom{|B|}{i_2}\binom{|C|}{i_2}$. In Step (iv), we approximate $i_4 \leq \ell-2$, $f(i_4) \leq G(i_4)$ and disregard some small negative constants. Now we consider h':

$$\begin{aligned}
h'(i_4) &\overset{(13)}{=} \sum_{i_1=i_4+e+1-\ell}^{e} \binom{|A|}{i_1} \sum_{i_2=0}^{\ell-1-i_4} \binom{|B|}{i_2} \left(\frac{|C|+i_4+e+1-\ell-i_1-i_2}{i_1+i_2+\ell-e-i_4} \binom{|C|}{i_1+i_2+\ell-e-1-i_4} \right. \\
&\quad \left. + \;\frac{i_2+i_4+e+1-\ell-i_1}{|C|+i_1+\ell-e-i_2-i_4} \binom{|C|}{i_2+i_4+e+1-\ell-i_1} \right) \Bigg) \\
&\overset{(v)}{\leq} \sum_{i_1=i_4+e+1-\ell}^{e} \binom{|A|}{i_1} \frac{|C|+i_4+e+1-\ell-i_1}{i_1+\ell-e-i_4} \sum_{i_2=0}^{\ell-1-i_4} \binom{|B|}{i_2} \\
&\quad \cdot \left(\binom{|C|}{i_1+i_2+\ell-e-1-i_4} + \binom{|C|}{i_2+i_4+e+1-\ell-i_1} \right) \Bigg) \\
&\overset{(vi)}{\leq} |C| h(i_4) \\
&\leq |C| G(i_4).
\end{aligned}$$

For Inequality (v), we first show that $i_2 \leq |C| - i_2$. Indeed, $|C| - i_2 = b - 2e - 1 - i_2 \geq t + \ell > i_2$ since $b \geq 3t$ and $\ell - 1 \geq i_2$ by (8). Therefore, we may estimate $\frac{i_2+i_4+e+1-\ell-i_1}{|C|+i_1+\ell-e-i_2-i_4} \leq \frac{|C|+i_4+e+1-\ell-i_1-i_2}{i_1+i_2+\ell-e-i_4}$ since $|C|+i_1+\ell-e-i_2-i_4 \geq i_1+i_2+\ell-e-i_4$ and $i_2+i_4+e+1-\ell-i_1 \leq |C|+i_4+e+1-\ell-i_1-i_2$. Observe that $i_1+i_2+\ell-e-i_4 \geq 1$ because $i_2 \geq 0$ and $i_1 \geq i_4+e+1-\ell$. Furthermore, the latter fraction gets its maximal value with respect to i_2 when $i_2 = 0$. Moreover, in Step (vi) we set $i_1 = i_4+e+1-\ell$, so that, $\frac{|C|+i_4+e+1-\ell-i_1}{i_1+\ell-e-i_4}$ gets its maximal value. Next we concentrate on $s'(i_4)$:

$$\begin{aligned}
s'(i_4) &\overset{(13)}{=} \frac{|B|+i_4+1-\ell}{\ell-i_4} \binom{|B|}{\ell-i_4-1} \sum_{i_1=i_4+e+1-\ell}^{e} \binom{|A|}{i_1} \sum_{i_3=e-i_1}^{i_1+2\ell-e-2i_4} \binom{|C|}{i_3} \\
&= \frac{|B|+i_4+1-\ell}{\ell-i_4} \binom{|B|}{\ell-i_4-1} \sum_{i_1=i_4+e+1-\ell}^{e} \binom{|A|}{i_1} \left(\sum_{i_3=e-i_1}^{i_1+2\ell-e-2-2i_4} \binom{|C|}{i_3} \right. \\
&\quad \left. + \;\frac{|C|+2i_4+e+2-2\ell-i_1}{i_1+2\ell-e-1-2i_4} \left(\frac{|C|+2i_4+e+1-2\ell-i_1}{i_1+2\ell-e-2i_4} + 1 \right) \binom{|C|}{i_1+2\ell-e-2-2i_4} \right) \\
&\overset{(vii)}{\leq} \frac{|B|-1}{2} \left(1 + \frac{|C|-1}{2} \left(1 + \frac{|C|-2}{3} \right) \right) s(i_4) \\
&< \frac{|B|(|C|+1)^2}{12} G(i_4).
\end{aligned}$$

In Step (vii) we first estimate $i_1 \leq i_4+e+1-\ell$ and then $i_4 \leq \ell-2$ in the fractional multipliers to maximize them.

Now we are ready to show that $g(i_4-1) \leq g(i_4)$, when n is large enough, by combining the previous upper bounds for $f'(i_4)$, $h'(i_4)$ and $s'(i_4)$ with Equation (12). Now,

$$\begin{aligned}
g(i_4-1) =& G(i_4-1) \binom{|D|}{i_4-1} \\
&\leq \left(1 + \frac{e(|B||C|+4)}{8} + |C| + \frac{|B|(|C|+1)^2}{12} \right) G(i_4) \binom{|D|}{i_4} \frac{i_4}{|D|-i_4+1} \\
&= \left(1 + \frac{e(|B||C|+4)}{8} + |C| + \frac{|B|(|C|+1)^2}{12} \right) \frac{i_4}{|D|-i_4+1} g(i_4).
\end{aligned}$$

Thus, $g(i_4-1) \leq g(i_4)$ when

$$\begin{aligned}
|D| - i_4 + 1 \geq& i_4 \cdot \left(1 + \frac{e(|B||C|+4)}{8} + |C| + \frac{|B|(|C|+1)^2}{12} \right) \\
n \geq& i_4 \cdot \left(1 + \frac{e(|B||C|+4)}{8} + |C| + \frac{|B|(|C|+1)^2}{12} \right) + i_4 + b - 1.
\end{aligned}$$

Furthermore, $g(\ell-2) \geq g(j)$ for $\ell-3 \geq j \geq 0$ when

$$n \geq \left(1 + \frac{e(|B||C|+4)}{8} + |C| + \frac{|B|(|C|+1)^2}{12} \right) (\ell-2) + (\ell-2) - 1 + b. \tag{14}$$

For $i_4 = \ell-2$, Equation (9) gives: $G(\ell-2) = |C|+|B|(1+|C|+e|C|+\binom{|C|}{2})+e+1 = |C|+|B|(2+|C|+e|C|+\binom{|C|}{2})$. Based on this value and the inequality $g(i_4-1) \leq g(i_4)$ when n is large enough, we may estimate $|S|$ using Inequality (11):

$$\begin{aligned}|S| \leq &\sum_{i_4=0}^{\ell-2} g(i_4) \leq (\ell-1)g(\ell-2)\\ =&(\ell-1)\binom{|D|}{\ell-2}G(\ell-2)\\ <&(\ell-1)\binom{n}{\ell-2}G(\ell-2)\\ =&\frac{(\ell-1)^2}{n-\ell+2}\binom{n}{\ell-1}\left(|C|+|B|\left(2+|C|+e|C|+\binom{|C|}{2}\right)\right).\end{aligned}$$

Hence, $|S| < \binom{n}{l-1} < |N|$ when $n-\ell+2 \geq (\ell-1)^2\left(|C|+|B|\left(2+|C|+e|C|+\binom{|C|}{2}\right)\right)$, that is, $n \geq (\ell-1)^2\left(|C|+|B|\left(2+|C|+e|C|+\binom{|C|}{2}\right)\right)+\ell-2 = (\ell-1)^2(b-e+(e+1)(b-3e-2e^2+eb+\binom{b-2e-1}{2}))+\ell-2$. Notice that this value is greater than the one in Inequality (14), since

$$\begin{aligned}&(\ell-1)^2\left(|C|+|B|\left(2+|C|+e|C|+\binom{|C|}{2}\right)\right)+\ell-2\\ \geq\ &(\ell-2)\left(|C|+|B|\left(2+|C|+e|C|+\binom{|C|}{2}\right)\right)+\ell-3+b\\ =\ &\left(2|B|+\frac{4e|B||C|+4e|B||C|}{8}+|C|+6|B|\frac{|C|^2+|C|}{12}\right)(\ell-2)+\ell-3+b\\ \overset{(viii)}{>}&\left(1+\frac{e(|B||C|+4)}{8}+|C|+\frac{|B|(|C|+1)^2}{12}\right)(\ell-2)+\ell-3+b.\end{aligned}$$

In Step $(viii)$, we have $2|B|>1$, $\frac{4e|B||C|+4e|B||C|}{8} > \frac{e(|B||C|+4)}{8}$ and $6|B|\frac{|C|^2+|C|}{12} = 3|B|\frac{2|C|^2+2|C|}{12} > |B|\frac{|C|^2+2|C|+1}{12}$.

Since $|S| < N$, there is an output word $\mathbf{y} \in Y$ such that $\mathbf{y} \notin S$. ☐

REFERENCES

[1] V. Junnila, T. Laihonen, and T. Lehtilä, "The Levenshtein's channel and the list size in information retrieval," in *Proceedings of 2019 IEEE International Symposium on Information Theory*, 2019, pp. 295–299.

[2] V. I. Levenshtein, "Efficient reconstruction of sequences," *IEEE Trans. Inform. Theory*, vol. 47, no. 1, pp. 2–22, 2001.

[3] M. Horovitz and E. Yaakobi, "Reconstruction of sequences over non-identical channels," *IEEE Trans. Inform. Theory*, vol. 65, no. 2, pp. 1267–1286, 2018.

[4] E. Yaakobi, J. Bruck, and P. H. Siegel, "Constructions and decoding of cyclic codes over *b*-symbol read channels," *IEEE Trans. Inform. Theory*, vol. 62, no. 4, pp. 1541–1551, 2016.

[5] J. Bornholt, R. Lopez, D. M. Carmean, L. Ceze, G. Seelig, and K. Strauss, "A DNA-based archival storage system," *ACM SIGARCH Computer Architecture News*, vol. 44, no. 2, pp. 637–649, 2016.

[6] V. Levenshtein, E. Konstantinova, E. Konstantinov, and S. Molodtsov, "Reconstruction of a graph from 2-vicinities of its vertices," *Discrete Applied Mathematics*, vol. 156, pp. 1399–1406, 2008.

[7] R. Gabrys and E. Yaakobi, "Sequence reconstruction over the deletion channel," *IEEE Trans. Inform. Theory*, vol. 64, no. 4, pp. 2924–2931, 2018.

[8] E. Yaakobi and J. Bruck, "On the uncertainty of information retrieval in associative memories," *IEEE Trans. Inform. Theory*, vol. 65, no. 4, pp. 2155–2165, 2018.

[9] ——, "On the uncertainty of information retrieval in associative memories," in *Proceedings of 2012 IEEE International Symposium on Information Theory*, 2012, pp. 106–110.

[10] V. Junnila and T. Laihonen, "Information retrieval with varying number of input clues," *IEEE Trans. Inform. Theory*, vol. 62, no. 2, pp. 625–638, 2016.

[11] ——, "Codes for information retrieval with small uncertainty," *IEEE Trans. Inform. Theory*, vol. 60, no. 2, pp. 976–985, 2014.

[12] T. Laihonen and T. Lehtilä, "Improved codes for list decoding in the Levenshtein's channel and information retrieval," in *Proceedings of 2017 IEEE International Symposium on Information Theory*, 2017, pp. 2643–2647.

[13] T. Laihonen, "On t-revealing codes in binary Hamming spaces," *Information and Computation*, vol. 268, 2019.

[14] G. Cohen, I. Honkala, S. Litsyn, and A. Lobstein, *Covering codes*, ser. North-Holland Mathematical Library. Amsterdam: North-Holland Publishing Co., 1997, vol. 54.

[15] D. J. Kleitman, "On a combinatorial conjecture of Erdös," *Journal of Combinatorial Theory*, vol. 1, no. 2, pp. 209–214, 1966.

[16] N. Sauer, "On the density of families of sets," *Journal of Combinatorial Theory, Series A*, vol. 13, no. 1, pp. 145–147, 1972.

[17] S. Shelah, "A combinatorial problem; stability and order for models and theories in infinitary languages," *Pacific Journal of Mathematics*, vol. 41, no. 1, pp. 247–261, 1972.

[18] F. J. MacWilliams and N. J. A. Sloane, *The theory of error-correcting codes*, ser. North-Holland Mathematical Library. Amsterdam: North-Holland Publishing Co., 1977, vol. 16.

[19] M. Plotkin, "Binary codes with specified minimum distance," *IRE Trans. Inform. Theory*, vol. 6, no. 4, pp. 445–450, 1960.

[20] E. Agrell, A. Vardy, and K. Zeger, "Upper bounds for constant-weight codes," *IEEE Trans. Inform. Theory*, vol. 46, no. 7, pp. 2373–2395, 2000.

[21] R. Graham and N. Sloane, "Lower bounds for constant weight codes," *IEEE Trans. Inform. Theory*, vol. 26, no. 1, pp. 37–43, 1980.

Turku Centre for Computer Science
TUCS Dissertations

1. **Marjo Lipponen**, On Primitive Solutions of the Post Correspondence Problem
2. **Timo Käkölä**, Dual Information Systems in Hyperknowledge Organizations
3. **Ville Leppänen**, Studies on the Realization of PRAM
4. **Cunsheng Ding**, Cryptographic Counter Generators
5. **Sami Viitanen**, Some New Global Optimization Algorithms
6. **Tapio Salakoski**, Representative Classification of Protein Structures
7. **Thomas Långbacka**, An Interactive Environment Supporting the Development of Formally Correct Programs
8. **Thomas Finne**, A Decision Support System for Improving Information Security
9. **Valeria Mihalache**, Cooperation, Communication, Control. Investigations on Grammar Systems.
10. **Marina Waldén**, Formal Reasoning About Distributed Algorithms
11. **Tero Laihonen**, Estimates on the Covering Radius When the Dual Distance is Known
12. **Lucian Ilie**, Decision Problems on Orders of Words
13. **Jukkapekka Hekanaho**, An Evolutionary Approach to Concept Learning
14. **Jouni Järvinen**, Knowledge Representation and Rough Sets
15. **Tomi Pasanen**, In-Place Algorithms for Sorting Problems
16. **Mika Johnsson**, Operational and Tactical Level Optimization in Printed Circuit Board Assembly
17. **Mats Aspnäs**, Multiprocessor Architecture and Programming: The Hathi-2 System
18. **Anna Mikhajlova**, Ensuring Correctness of Object and Component Systems
19. **Vesa Torvinen**, Construction and Evaluation of the Labour Game Method
20. **Jorma Boberg**, Cluster Analysis. A Mathematical Approach with Applications to Protein Structures
21. **Leonid Mikhajlov**, Software Reuse Mechanisms and Techniques: Safety Versus Flexibility
22. **Timo Kaukoranta**, Iterative and Hierarchical Methods for Codebook Generation in Vector Quantization
23. **Gábor Magyar**, On Solution Approaches for Some Industrially Motivated Combinatorial Optimization Problems
24. **Linas Laibinis**, Mechanised Formal Reasoning About Modular Programs
25. **Shuhua Liu**, Improving Executive Support in Strategic Scanning with Software Agent Systems
26. **Jaakko Järvi**, New Techniques in Generic Programming – C++ is more Intentional than Intended
27. **Jan-Christian Lehtinen**, Reproducing Kernel Splines in the Analysis of Medical Data
28. **Martin Büchi**, Safe Language Mechanisms for Modularization and Concurrency
29. **Elena Troubitsyna**, Stepwise Development of Dependable Systems
30. **Janne Näppi**, Computer-Assisted Diagnosis of Breast Calcifications
31. **Jianming Liang**, Dynamic Chest Images Analysis
32. **Tiberiu Seceleanu**, Systematic Design of Synchronous Digital Circuits
33. **Tero Aittokallio**, Characterization and Modelling of the Cardiorespiratory System in Sleep-Disordered Breathing
34. **Ivan Porres**, Modeling and Analyzing Software Behavior in UML
35. **Mauno Rönkkö**, Stepwise Development of Hybrid Systems
36. **Jouni Smed**, Production Planning in Printed Circuit Board Assembly
37. **Vesa Halava**, The Post Correspondence Problem for Market Morphisms
38. **Ion Petre**, Commutation Problems on Sets of Words and Formal Power Series
39. **Vladimir Kvassov**, Information Technology and the Productivity of Managerial Work
40. **Frank Tétard**, Managers, Fragmentation of Working Time, and Information Systems

41. **Jan Manuch**, Defect Theorems and Infinite Words
42. **Kalle Ranto**, Z_4-Goethals Codes, Decoding and Designs
43. **Arto Lepistö**, On Relations Between Local and Global Periodicity
44. **Mika Hirvensalo**, Studies on Boolean Functions Related to Quantum Computing
45. **Pentti Virtanen**, Measuring and Improving Component-Based Software Development
46. **Adekunle Okunoye**, Knowledge Management and Global Diversity – A Framework to Support Organisations in Developing Countries
47. **Antonina Kloptchenko**, Text Mining Based on the Prototype Matching Method
48. **Juha Kivijärvi**, Optimization Methods for Clustering
49. **Rimvydas Rukšėnas**, Formal Development of Concurrent Components
50. **Dirk Nowotka**, Periodicity and Unbordered Factors of Words
51. **Attila Gyenesei**, Discovering Frequent Fuzzy Patterns in Relations of Quantitative Attributes
52. **Petteri Kaitovaara**, Packaging of IT Services – Conceptual and Empirical Studies
53. **Petri Rosendahl**, Niho Type Cross-Correlation Functions and Related Equations
54. **Péter Majlender**, A Normative Approach to Possibility Theory and Soft Decision Support
55. **Seppo Virtanen**, A Framework for Rapid Design and Evaluation of Protocol Processors
56. **Tomas Eklund**, The Self-Organizing Map in Financial Benchmarking
57. **Mikael Collan**, Giga-Investments: Modelling the Valuation of Very Large Industrial Real Investments
58. **Dag Björklund**, A Kernel Language for Unified Code Synthesis
59. **Shengnan Han**, Understanding User Adoption of Mobile Technology: Focusing on Physicians in Finland
60. **Irina Georgescu**, Rational Choice and Revealed Preference: A Fuzzy Approach
61. **Ping Yan**, Limit Cycles for Generalized Liénard-Type and Lotka-Volterra Systems
62. **Joonas Lehtinen**, Coding of Wavelet-Transformed Images
63. **Tommi Meskanen,** On the NTRU Cryptosystem
64. **Saeed Salehi**, Varieties of Tree Languages
65. **Jukka Arvo**, Efficient Algorithms for Hardware-Accelerated Shadow Computation
66. **Mika Hirvikorpi**, On the Tactical Level Production Planning in Flexible Manufacturing Systems
67. **Adrian Costea**, Computational Intelligence Methods for Quantitative Data Mining
68. **Cristina Seceleanu**, A Methodology for Constructing Correct Reactive Systems
69. **Luigia Petre**, Modeling with Action Systems
70. **Lu Yan**, Systematic Design of Ubiquitous Systems
71. **Mehran Gomari**, On the Generalization Ability of Bayesian Neural Networks
72. **Ville Harkke**, Knowledge Freedom for Medical Professionals – An Evaluation Study of a Mobile Information System for Physicians in Finland
73. **Marius Cosmin Codrea**, Pattern Analysis of Chlorophyll Fluorescence Signals
74. **Aiying Rong**, Cogeneration Planning Under the Deregulated Power Market and Emissions Trading Scheme
75. **Chihab BenMoussa**, Supporting the Sales Force through Mobile Information and Communication Technologies: Focusing on the Pharmaceutical Sales Force
76. **Jussi Salmi**, Improving Data Analysis in Proteomics
77. **Orieta Celiku**, Mechanized Reasoning for Dually-Nondeterministic and Probabilistic Programs
78. **Kaj-Mikael Björk**, Supply Chain Efficiency with Some Forest Industry Improvements
79. **Viorel Preoteasa**, Program Variables – The Core of Mechanical Reasoning about Imperative Programs
80. **Jonne Poikonen**, Absolute Value Extraction and Order Statistic Filtering for a Mixed-Mode Array Image Processor
81. **Luka Milovanov**, Agile Software Development in an Academic Environment
82. **Francisco Augusto Alcaraz Garcia**, Real Options, Default Risk and Soft Applications
83. **Kai K. Kimppa**, Problems with the Justification of Intellectual Property Rights in Relation to Software and Other Digitally Distributable Media
84. **Dragoş Truşcan**, Model Driven Development of Programmable Architectures
85. **Eugen Czeizler**, The Inverse Neighborhood Problem and Applications of Welch Sets in Automata Theory

86. **Sanna Ranto**, Identifying and Locating-Dominating Codes in Binary Hamming Spaces
87. **Tuomas Hakkarainen**, On the Computation of the Class Numbers of Real Abelian Fields
88. **Elena Czeizler**, Intricacies of Word Equations
89. **Marcus Alanen**, A Metamodeling Framework for Software Engineering
90. **Filip Ginter**, Towards Information Extraction in the Biomedical Domain: Methods and Resources
91. **Jarkko Paavola**, Signature Ensembles and Receiver Structures for Oversaturated Synchronous DS-CDMA Systems
92. **Arho Virkki**, The Human Respiratory System: Modelling, Analysis and Control
93. **Olli Luoma**, Efficient Methods for Storing and Querying XML Data with Relational Databases
94. **Dubravka Ilić**, Formal Reasoning about Dependability in Model-Driven Development
95. **Kim Solin**, Abstract Algebra of Program Refinement
96. **Tomi Westerlund**, Time Aware Modelling and Analysis of Systems-on-Chip
97. **Kalle Saari**, On the Frequency and Periodicity of Infinite Words
98. **Tomi Kärki**, Similarity Relations on Words: Relational Codes and Periods
99. **Markus M. Mäkelä**, Essays on Software Product Development: A Strategic Management Viewpoint
100. **Roope Vehkalahti**, Class Field Theoretic Methods in the Design of Lattice Signal Constellations
101. **Anne-Maria Ernvall-Hytönen**, On Short Exponential Sums Involving Fourier Coefficients of Holomorphic Cusp Forms
102. **Chang Li**, Parallelism and Complexity in Gene Assembly
103. **Tapio Pahikkala**, New Kernel Functions and Learning Methods for Text and Data Mining
104. **Denis Shestakov**, Search Interfaces on the Web: Querying and Characterizing
105. **Sampo Pyysalo**, A Dependency Parsing Approach to Biomedical Text Mining
106. **Anna Sell**, Mobile Digital Calendars in Knowledge Work
107. **Dorina Marghescu**, Evaluating Multidimensional Visualization Techniques in Data Mining Tasks
108. **Tero Säntti**, A Co-Processor Approach for Efficient Java Execution in Embedded Systems
109. **Kari Salonen**, Setup Optimization in High-Mix Surface Mount PCB Assembly
110. **Pontus Boström**, Formal Design and Verification of Systems Using Domain-Specific Languages
111. **Camilla J. Hollanti**, Order-Theoretic Mehtods for Space-Time Coding: Symmetric and Asymmetric Designs
112. **Heidi Himmanen**, On Transmission System Design for Wireless Broadcasting
113. **Sébastien Lafond**, Simulation of Embedded Systems for Energy Consumption Estimation
114. **Evgeni Tsivtsivadze**, Learning Preferences with Kernel-Based Methods
115. **Petri Salmela**, On Commutation and Conjugacy of Rational Languages and the Fixed Point Method
116. **Siamak Taati**, Conservation Laws in Cellular Automata
117. **Vladimir Rogojin**, Gene Assembly in Stichotrichous Ciliates: Elementary Operations, Parallelism and Computation
118. **Alexey Dudkov**, Chip and Signature Interleaving in DS CDMA Systems
119. **Janne Savela**, Role of Selected Spectral Attributes in the Perception of Synthetic Vowels
120. **Kristian Nybom**, Low-Density Parity-Check Codes for Wireless Datacast Networks
121. **Johanna Tuominen**, Formal Power Analysis of Systems-on-Chip
122. **Teijo Lehtonen**, On Fault Tolerance Methods for Networks-on-Chip
123. **Eeva Suvitie**, On Inner Products Involving Holomorphic Cusp Forms and Maass Forms
124. **Linda Mannila**, Teaching Mathematics and Programming – New Approaches with Empirical Evaluation
125. **Hanna Suominen**, Machine Learning and Clinical Text: Supporting Health Information Flow
126. **Tuomo Saarni**, Segmental Durations of Speech
127. **Johannes Eriksson**, Tool-Supported Invariant-Based Programming

128. **Tero Jokela**, Design and Analysis of Forward Error Control Coding and Signaling for Guaranteeing QoS in Wireless Broadcast Systems
129. **Ville Lukkarila**, On Undecidable Dynamical Properties of Reversible One-Dimensional Cellular Automata
130. **Qaisar Ahmad Malik**, Combining Model-Based Testing and Stepwise Formal Development
131. **Mikko-Jussi Laakso**, Promoting Programming Learning: Engagement, Automatic Assessment with Immediate Feedback in Visualizations
132. **Riikka Vuokko**, A Practice Perspective on Organizational Implementation of Information Technology
133. **Jeanette Heidenberg**, Towards Increased Productivity and Quality in Software Development Using Agile, Lean and Collaborative Approaches
134. **Yong Liu**, Solving the Puzzle of Mobile Learning Adoption
135. **Stina Ojala**, Towards an Integrative Information Society: Studies on Individuality in Speech and Sign
136. **Matteo Brunelli**, Some Advances in Mathematical Models for Preference Relations
137. **Ville Junnila**, On Identifying and Locating-Dominating Codes
138. **Andrzej Mizera**, Methods for Construction and Analysis of Computational Models in Systems Biology. Applications to the Modelling of the Heat Shock Response and the Self-Assembly of Intermediate Filaments.
139. **Csaba Ráduly-Baka**, Algorithmic Solutions for Combinatorial Problems in Resource Management of Manufacturing Environments
140. **Jari Kyngäs**, Solving Challenging Real-World Scheduling Problems
141. **Arho Suominen**, Notes on Emerging Technologies
142. **József Mezei**, A Quantitative View on Fuzzy Numbers
143. **Marta Olszewska**, On the Impact of Rigorous Approaches on the Quality of Development
144. **Antti Airola**, Kernel-Based Ranking: Methods for Learning and Performace Estimation
145. **Aleksi Saarela**, Word Equations and Related Topics: Independence, Decidability and Characterizations
146. **Lasse Bergroth**, Kahden merkkijonon pisimmän yhteisen alijonon ongelma ja sen ratkaiseminen
147. **Thomas Canhao Xu**, Hardware/Software Co-Design for Multicore Architectures
148. **Tuomas Mäkilä**, Software Development Process Modeling – Developers Perspective to Contemporary Modeling Techniques
149. **Shahrokh Nikou**, Opening the Black-Box of IT Artifacts: Looking into Mobile Service Characteristics and Individual Perception
150. **Alessandro Buoni**, Fraud Detection in the Banking Sector: A Multi-Agent Approach
151. **Mats Neovius**, Trustworthy Context Dependency in Ubiquitous Systems
152. **Fredrik Degerlund**, Scheduling of Guarded Command Based Models
153. **Amir-Mohammad Rahmani-Sane**, Exploration and Design of Power-Efficient Networked Many-Core Systems
154. **Ville Rantala**, On Dynamic Monitoring Methods for Networks-on-Chip
155. **Mikko Pelto**, On Identifying and Locating-Dominating Codes in the Infinite King Grid
156. **Anton Tarasyuk**, Formal Development and Quantitative Verification of Dependable Systems
157. **Muhammad Mohsin Saleemi**, Towards Combining Interactive Mobile TV and Smart Spaces: Architectures, Tools and Application Development
158. **Tommi J. M. Lehtinen**, Numbers and Languages
159. **Peter Sarlin**, Mapping Financial Stability
160. **Alexander Wei Yin**, On Energy Efficient Computing Platforms
161. **Mikołaj Olszewski**, Scaling Up Stepwise Feature Introduction to Construction of Large Software Systems
162. **Maryam Kamali**, Reusable Formal Architectures for Networked Systems
163. **Zhiyuan Yao**, Visual Customer Segmentation and Behavior Analysis – A SOM-Based Approach
164. **Timo Jolivet**, Combinatorics of Pisot Substitutions
165. **Rajeev Kumar Kanth**, Analysis and Life Cycle Assessment of Printed Antennas for Sustainable Wireless Systems
166. **Khalid Latif**, Design Space Exploration for MPSoC Architectures

167. **Bo Yang**, Towards Optimal Application Mapping for Energy-Efficient Many-Core Platforms
168. **Ali Hanzala Khan**, Consistency of UML Based Designs Using Ontology Reasoners
169. **Sonja Leskinen**, m-Equine: IS Support for the Horse Industry
170. **Fareed Ahmed Jokhio**, Video Transcoding in a Distributed Cloud Computing Environment
171. **Moazzam Fareed Niazi**, A Model-Based Development and Verification Framework for Distributed System-on-Chip Architecture
172. **Mari Huova**, Combinatorics on Words: New Aspects on Avoidability, Defect Effect, Equations and Palindromes
173. **Ville Timonen**, Scalable Algorithms for Height Field Illumination
174. **Henri Korvela**, Virtual Communities – A Virtual Treasure Trove for End-User Developers
175. **Kameswar Rao Vaddina**, Thermal-Aware Networked Many-Core Systems
176. **Janne Lahtiranta**, New and Emerging Challenges of the ICT-Mediated Health and Well-Being Services
177. **Irum Rauf**, Design and Validation of Stateful Composite RESTful Web Services
178. **Jari Björne**, Biomedical Event Extraction with Machine Learning
179. **Katri Haverinen**, Natural Language Processing Resources for Finnish: Corpus Development in the General and Clinical Domains
180. **Ville Salo**, Subshifts with Simple Cellular Automata
181. **Johan Ersfolk**, Scheduling Dynamic Dataflow Graphs
182. **Hongyan Liu**, On Advancing Business Intelligence in the Electricity Retail Market
183. **Adnan Ashraf**, Cost-Efficient Virtual Machine Management: Provisioning, Admission Control, and Consolidation
184. **Muhammad Nazrul Islam**, Design and Evaluation of Web Interface Signs to Improve Web Usability: A Semiotic Framework
185. **Johannes Tuikkala**, Algorithmic Techniques in Gene Expression Processing: From Imputation to Visualization
186. **Natalia Díaz Rodríguez**, Semantic and Fuzzy Modelling for Human Behaviour Recognition in Smart Spaces. A Case Study on Ambient Assisted Living
187. **Mikko Pänkäälä**, Potential and Challenges of Analog Reconfigurable Computation in Modern and Future CMOS
188. **Sami Hyrynsalmi**, Letters from the War of Ecosystems – An Analysis of Independent Software Vendors in Mobile Application Marketplaces
189. **Seppo Pulkkinen**, Efficient Optimization Algorithms for Nonlinear Data Analysis
190. **Sami Pyöttiälä**, Optimization and Measuring Techniques for Collect-and-Place Machines in Printed Circuit Board Industry
191. **Syed Mohammad Asad Hassan Jafri**, Virtual Runtime Application Partitions for Resource Management in Massively Parallel Architectures
192. **Toni Ernvall**, On Distributed Storage Codes
193. **Yuliya Prokhorova**, Rigorous Development of Safety-Critical Systems
194. **Olli Lahdenoja**, Local Binary Patterns in Focal-Plane Processing – Analysis and Applications
195. **Annika H. Holmbom**, Visual Analytics for Behavioral and Niche Market Segmentation
196. **Sergey Ostroumov**, Agent-Based Management System for Many-Core Platforms: Rigorous Design and Efficient Implementation
197. **Espen Suenson**, How Computer Programmers Work – Understanding Software Development in Practise
198. **Tuomas Poikela**, Readout Architectures for Hybrid Pixel Detector Readout Chips
199. **Bogdan Iancu**, Quantitative Refinement of Reaction-Based Biomodels
200. **Ilkka Törmä**, Structural and Computational Existence Results for Multidimensional Subshifts
201. **Sebastian Okser**, Scalable Feature Selection Applications for Genome-Wide Association Studies of Complex Diseases
202. **Fredrik Abbors**, Model-Based Testing of Software Systems: Functionality and Performance
203. **Inna Pereverzeva**, Formal Development of Resilient Distributed Systems
204. **Mikhail Barash**, Defining Contexts in Context-Free Grammars
205. **Sepinoud Azimi**, Computational Models for and from Biology: Simple Gene Assembly and Reaction Systems
206. **Petter Sandvik**, Formal Modelling for Digital Media Distribution

207. **Jongyun Moon**, Hydrogen Sensor Application of Anodic Titanium Oxide Nanostructures
208. **Simon Holmbacka**, Energy Aware Software for Many-Core Systems
209. **Charalampos Zinoviadis**, Hierarchy and Expansiveness in Two-Dimensional Subshifts of Finite Type
210. **Mika Murtojärvi**, Efficient Algorithms for Coastal Geographic Problems
211. **Sami Mäkelä**, Cohesion Metrics for Improving Software Quality
212. **Eyal Eshet**, Examining Human-Centered Design Practice in the Mobile Apps Era
213. **Jetro Vesti**, Rich Words and Balanced Words
214. **Jarkko Peltomäki**, Privileged Words and Sturmian Words
215. **Fahimeh Farahnakian**, Energy and Performance Management of Virtual Machines: Provisioning, Placement and Consolidation
216. **Diana-Elena Gratie**, Refinement of Biomodels Using Petri Nets
217. **Harri Merisaari**, Algorithmic Analysis Techniques for Molecular Imaging
218. **Stefan Grönroos**, Efficient and Low-Cost Software Defined Radio on Commodity Hardware
219. **Noora Nieminen**, Garbling Schemes and Applications
220. **Ville Taajamaa**, O-CDIO: Engineering Education Framework with Embedded Design Thinking Methods
221. **Johannes Holvitie**, Technical Debt in Software Development – Examining Premises and Overcoming Implementation for Efficient Management
222. **Tewodros Deneke**, Proactive Management of Video Transcoding Services
223. **Kashif Javed**, Model-Driven Development and Verification of Fault Tolerant Systems
224. **Pekka Naula**, Sparse Predictive Modeling – A Cost-Effective Perspective
225. **Antti Hakkala**, On Security and Privacy for Networked Information Society – Observations and Solutions for Security Engineering and Trust Building in Advanced Societal Processes
226. **Anne-Maarit Majanoja**, Selective Outsourcing in Global IT Services – Operational Level Challenges and Opportunities
227. **Samuel Rönnqvist**, Knowledge-Lean Text Mining
228. **Mohammad-Hashem Hahgbayan**, Energy-Efficient and Reliable Computing in Dark Silicon Era
229. **Charmi Panchal**, Qualitative Methods for Modeling Biochemical Systems and Datasets: The Logicome and the Reaction Systems Approaches
230. **Erkki Kaila**, Utilizing Educational Technology in Computer Science and Programming Courses: Theory and Practice
231. **Fredrik Robertsén**, The Lattice Boltzmann Method, a Petaflop and Beyond
232. **Jonne Pohjankukka**, Machine Learning Approaches for Natural Resource Data
233. **Paavo Nevalainen**, Geometric Data Understanding: Deriving Case-Specific Features
234. **Michal Szabados**, An Algebraic Approach to Nivat's Conjecture
235. **Tuan Nguyen Gia**, Design for Energy-Efficient and Reliable Fog-Assisted Healthcare IoT Systems
236. **Anil Kanduri**, Adaptive Knobs for Resource Efficient Computing
237. **Veronika Suni**, Computational Methods and Tools for Protein Phosphorylation Analysis
238. **Behailu Negash**, Interoperating Networked Embedded Systems to Compose the Web of Things
239. **Kalle Rindell**, Development of Secure Software: Rationale, Standards and Practices
240. **Jurka Rahikkala**, On Top Management Support for Software Cost Estimation
241. **Markus A. Whiteland**, On the k-Abelian Equivalence Relation of Finite Words
242. **Mojgan Kamali**, Formal Analysis of Network Routing Protocols
243. **Jesús Carabaño Bravo**, A Compiler Approach to Map Algebra for Raster Spatial Modeling
244. **Amin Majd**, Distributed and Lightweight Meta-heuristic Optimization Method for Complex Problems
245. **Ali Farooq**, In Quest of Information Security in Higher Education Institutions: Security Awareness, Concerns, and Behaviour of Students
246. **Juho Heimonen**, Knowledge Representation and Text Mining in Biomedical, Healthcare, and Political Domains

247. **Sanaz Rahimi Moosavi**, Towards End-to-End Security in Internet of Things based Healthcare
248. **Mingzhe Jiang**, Automatic Pain Assessment by Learning from Multiple Biopotentials
249. **Johan Kopra**, Cellular Automata with Complicated Dynamics
250. **Iman Azimi**, Personalized Data Analytics for Internet-of-Things-based Health Monitoring
251. **Jaakko Helminen**, Systems Action Design Research: Delineation of an Application to Develop Hybrid Local Climate Services
252. **Aung Pyae**, The Use of Digital Games to Enhance the Physical Exercise Activity of the Elderly: A Case of Finland
253. **Woubishet Zewdu Taffese**, Data-Driven Method for Enhanced Corrosion Assessment of Reinforced Concrete Structures
254. **Etienne Moutot**, Around the Domino Problem – Combinatorial Structures and Algebraic Tools
255. **Joonatan Jalonen**, On Some One-Sided Dynamics of Cellular Automata
256. **Outi Montonen**, On Multiobjective Optimization from the Nonsmooth Perspective
257. **Tuomo Lehtilä**, On Location, Domination and Information Retrieval

TURKU CENTRE *for* COMPUTER SCIENCE

http://www. tucs.fi
tucs@abo.fi

University of Turku
Faculty of Science and Engineering
- Department of Future Technologies
- Department of Mathematics and Statistics

Turku School of Economics
- Institute of Information Systems Science

Åbo Akademi University
Faculty of Science and Engineering
- Computer Engineering
- Computer Science

Faculty of Social Sciences, Business and Economics
- Information Systems

ISBN 978-952-12-3981-6
ISSN 1239-1883

Tuomo Lehtilä

On Location, Domination and Information Retrieval

Tuomo Lehtilä

On Location, Domination and Information Retrieval

Tuomo Lehtilä

On Location, Domination and Information Retrieval