

# Error-correcting codes from $k$ -resolving sets

Robert F. Bailey\*

Ismael G. Yero<sup>†</sup>

May 3, 2017

## Abstract

We demonstrate a construction of error-correcting codes from graphs by means of  $k$ -resolving sets, and present a decoding algorithm which makes use of covering designs. Along the way, we determine the  $k$ -metric dimension of grid graphs (i.e. Cartesian products of paths).

**MSC 2010:** 05C12, 94B25 (primary); 05B40, 94B35 (secondary).

**Keywords:** error-correcting code;  $k$ -resolving set;  $k$ -metric dimension; covering design; uncovering; grid graph.

## 1 Introduction

### 1.1 Error-correcting codes

Error-correcting codes are applied to the accurate transmission and storage of data. When information is received by a target, or read from a storage medium, errors may be introduced—for example, due to signal noise, or the medium being damaged—so to alleviate this problem, redundancy is introduced in order that the intended message can still be understood. For an introduction to coding theory, see [20].

Formally, an *error-correcting code* (or simply a *code*) is a collection  $\mathcal{C}$  of vectors, called *codewords*, of given length  $\ell$  over a fixed alphabet. The *Hamming distance* between two codewords  $\mathbf{x} = (x_1, \dots, x_\ell)$ ,  $\mathbf{y} = (y_1, \dots, y_\ell)$  is the number of positions where they differ, i.e.  $|\{i : x_i \neq y_i\}|$ . The *minimum distance* of  $\mathcal{C}$  is the least Hamming distance between any two distinct codewords; if the minimum distance is  $D$ , then the *correction capability* of  $\mathcal{C}$  is  $r = \lfloor (D-1)/2 \rfloor$ . Suppose that a codeword  $\mathbf{x}$  is transmitted via a noisy channel which causes errors to appear, i.e. some symbols are replaced with others. If there are  $r$  errors or fewer, the received word has a unique nearest neighbour in  $\mathcal{C}$ , which is necessarily the transmitted word  $\mathbf{x}$ . For this to be useful in practice, an efficient decoding algorithm is needed to determine the nearest neighbour.

Traditionally, the most familiar error-correcting codes are *linear codes* (i.e. subspaces of vector spaces over finite fields) [20], where the alphabet size is small (such as binary codes, which have an alphabet of size 2). Other classes of codes include *permutation codes* [7], where each codeword is a

---

\*School of Science & Environment (Mathematics), Grenfell Campus, Memorial University of Newfoundland, Corner Brook, NL A2H 6P9, Canada. Email: rbailey@grenfell.mun.ca

<sup>†</sup>Departamento de Matemáticas, Escuela Politécnica Superior de Algeciras, Universidad de Cádiz, Av. Ramón Puyol s/n, 11202 Algeciras, Spain. Email: ismael.gonzalez@uca.es

permutation of  $n$  symbols, so the length and alphabet size are both equal to  $n$ ; codes with larger alphabet sizes have been the subject of more recent attention, in part because of applications such as powerline communications [9] and flash memory devices [24].

## 1.2 $k$ -resolving sets

We consider finite, simple, connected, undirected graphs. The *distance* between two vertices  $u$  and  $v$  of a graph  $G$  is the length of a shortest path between  $u$  and  $v$ , and we denote this by  $d_G(u, v)$ . In recent years, much attention has been paid to the *metric dimension* of graphs: this is the smallest size of a subset of vertices (called a *resolving set*) with the property that the list of distances from any vertex to those in the set uniquely identifies that vertex, and is denoted by  $\dim(G)$ .

These concepts were introduced to graph theory in the 1970s by Harary and Melter [16] and, independently, Slater [23]; however, in the context of arbitrary metric spaces, the concept dates back at least as far as the 1950s [6]. Various applications have been suggested for resolving sets and metric dimension of graphs, including combinatorial optimization [22], pharmaceutical chemistry [8], robot navigation [17] and sonar [23]. For more information, see [3, 8].

The following definition is a natural generalization of the notion of resolving sets.

**Definition 1.** Let  $G = (V, E)$  be a graph. An ordered set of vertices  $(v_1, v_2, \dots, v_\ell)$  is a  *$k$ -resolving set* for  $G$  if, for any distinct vertices  $u, w \in V$ , the lists of distances  $(d_G(u, v_1), d_G(u, v_2), \dots, d_G(u, v_\ell))$  and  $(d_G(w, v_1), d_G(w, v_2), \dots, d_G(w, v_\ell))$  differ in at least  $k$  positions.

In the case  $k = 1$ , we have the usual notion of a resolving set for  $G$ . For  $k > 2$  it is not necessarily the case that an arbitrary graph  $G$  has a  $k$ -resolving set; for example, a complete graph  $K_n$  with  $n \geq 3$  has a 2-resolving set but not a 3-resolving set. If  $G$  has a  $k$ -resolving set, we denote the least size of a  $k$ -resolving set by  $\dim_k(G)$ , the  *$k$ -metric dimension* of  $G$ . A  $k$ -resolving set of size  $\dim_k(G)$  is called a  *$k$ -metric basis* for  $G$ . If  $k$  is the largest integer for which  $G$  has a  $k$ -resolving set, then we say that  $G$  is a  *$k$ -metric dimensional graph*.

The notion of  $k$ -resolving sets in graphs was introduced in [12] and further studied in [13, 14, 25]; it was then extended to more general metric spaces in [5].

## 2 Codes from $k$ -resolving sets

The two themes of this paper are tied together by the following definition.

**Definition 2.** Let  $G$  be a graph with  $n$  vertices and diameter  $d$ , and let  $S = \{v_1, v_2, \dots, v_\ell\}$  be a  $k$ -resolving set for  $G$  of size  $\ell$ . Then the set

$$C(G, S) = \{(d_G(u, v_1), d_G(u, v_2), \dots, d_G(u, v_\ell)) : u \in V\}$$

is called a  $(G, k)$ -code.

It follows from the definition that  $C(G, S)$  is an error-correcting code of length  $\ell$ , size  $n$  and minimum Hamming distance at least  $k$ , over the alphabet  $\{0, \dots, d\}$ , which can correct  $r = \lfloor (k-1)/2 \rfloor$  errors. In order for  $r$  to be non-zero, we require that  $k \geq 3$  (otherwise the code has no practical purpose).

For  $C(G, S)$  to be used for error correction, we need a decoding algorithm. Let  $G_j(u)$  denote the subset of vertices of  $G$  at distance  $j$  from  $u$ . Now suppose that  $u \in V$  and  $\mathbf{u} = (d_G(u, v_1), d_G(u, v_2), \dots, d_G(u, v_\ell)) \in C(G, S)$ ; suppose that we transmit  $\mathbf{u}$  and receive the word  $\mathbf{x} = (x_1, x_2, \dots, x_\ell)$ , which is assumed to have at most  $r$  errors.

**Lemma 3.** Let  $u$ ,  $\mathbf{u}$  and  $\mathbf{x}$  be as above, and suppose that  $I$  is an  $(\ell - r)$ -subset of  $\{1, \dots, \ell\}$ .

(i) If the received word  $\mathbf{x}$  contains no errors in the positions indexed by  $I$ , then

$$\bigcap_{i \in I} G_{x_i}(v_i) = \{u\}.$$

(ii) If the received word  $\mathbf{x}$  does contain an error in a position in  $I$ , then

$$\bigcap_{i \in I} G_{x_i}(v_i) = \emptyset.$$

*Proof.* If there are no errors in the positions indexed by  $I$ , then both  $\mathbf{x}$  and  $\mathbf{u}$  contain the same entries in those positions, and thus  $u$  is the unique vertex at those distances from the corresponding entries in the  $k$ -resolving set. If, however, there are errors in those positions, then no such vertex can exist, and thus the intersection is empty.  $\square$

The goal, therefore, when decoding a  $(G, k)$ -code is to find (as quickly as possible) an  $(\ell - r)$ -subset of positions for which the intersection  $\bigcap_{i \in I} G_{x_i}(v_i)$  is non-empty. Successively enumerating the  $(\ell - r)$ -subsets of  $\{1, \dots, \ell\}$  will achieve this, but will be slow in practice. However, if we assume that there are at most  $r' < r$  errors (say  $r' = 2$  or  $r' = 3$ ), we can make use of the following idea.

**Definition 4.** Let  $v, \kappa, \tau$  be integers such that  $v \geq \kappa \geq \tau \geq 0$ . A  $(v, v - \kappa, \tau)$ -*uncovering* is a collection  $\mathcal{U}$  of  $(v - \kappa)$ -subsets of  $\{1, \dots, v\}$  with the property that any  $\tau$ -subset of  $\{1, \dots, v\}$  is disjoint from at least one member of  $\mathcal{U}$ .

If we take the complements of each  $(v - \kappa)$ -subset in  $\mathcal{U}$ , we obtain a  $(v, \kappa, \tau)$ -*covering design*, which are much more widespread in the literature: see the survey by Mills and Mullin [19] for details of these. Uncoverings were introduced by the first author in [1, 2] where they were applied to decoding permutation codes; the same concept was also devised under the name *antiblocking system* by Kroll and Vincenti [18] for a decoding algorithm for linear codes. A further application to network reliability was given in [4].

The best known bound on the minimum size of coverings (and thus uncoverings also) is known as the *Schönheim bound*, proved in [21]. It states that for given  $v, \kappa$  and  $\tau$ , the least size of a  $(v, \kappa, \tau)$ -covering design is

$$L(v, \kappa, \tau) = \left\lceil \frac{v}{\kappa} \left\lceil \frac{v-1}{\kappa-1} \left\lceil \dots \left\lceil \frac{v-\tau+1}{\kappa-\tau+1} \right\rceil \dots \right\rceil \right\rceil \right\rceil.$$

Covering designs meeting this bound are known (or known asymptotically) in many cases: see [10, 19] for tables of results. The database of best-known covering designs [15] is useful for finding uncoverings with small parameters.

**Example 5.** The following is an  $(8, 5, 2)$ -covering design:

4	5	6	7	8
1	2	3	7	8
1	4	5	6	8
2	3	4	5	6

By taking the complements of each block, we obtain an  $(8, 3, 2)$ -uncovering:

$$\begin{array}{ccc} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 2 & 3 & 7 \\ 1 & 7 & 8 \end{array}$$

It can easily be seen that any pair chosen from  $\{1, \dots, 8\}$  is disjoint from at least one row.

## 2.1 A decoding algorithm

Suppose we have a  $(G, k)$ -code of length  $\ell$ , and we wish to correct  $r'$  errors. If  $\mathcal{U}$  is an  $(\ell, \ell - r, r')$ -uncovering, we proceed as follows: for a received word  $\mathbf{x}$ , we consider each  $I \in \mathcal{U}$  and obtain  $\bigcap_{i \in I} G_{x_i}(v_i)$ .

By Lemma 3, this intersection will either be empty or contain the vertex  $u$  corresponding to the transmitted word  $\mathbf{u}$ . If  $\mathbf{u}$  contains at most  $r'$  errors, by the definition of uncovering we know that there exists an  $I \in \mathcal{U}$  disjoint from the error positions; consequently, we are guaranteed to be able to find the transmitted word. To compute  $\bigcap_{i \in I} G_{x_i}(v_i)$ , consider the matrix  $M$  whose rows are indexed by  $V$  and whose columns are indexed by  $S$ , and where the entries are  $M_{uv} = d_G(u, v)$  (so the rows of  $M$  are precisely the codewords). For a given  $I \subseteq S$ , examine the rows of the submatrix to find a row which agrees with  $\mathbf{x}$  in those positions; if such a row exists, by Lemma 3 it must be unique and correspond to the vertex  $u$ .

## 2.2 Complexity

The matrix  $M$  is a submatrix of the distance matrix of  $G$ , which can be computed in  $O(|V|^3)$  time (for instance, by the Floyd–Warshall algorithm; see [11, §25.2]); however, this need only be done once, prior to the implementation of the code. For a given instance of the decoding problem, where the input is a received word  $\mathbf{x}$  and the output the transmitted word  $\mathbf{u}$ , the matrix  $M$  must be examined at most  $|\mathcal{U}|$  times, and at most  $|I| \cdot |V|$  steps are required each time. Thus the overall complexity of the decoding algorithm is  $O(|\mathcal{U}| \cdot |I| \cdot |V|)$ .

## 3 Some covering designs

Consider a  $(v, \kappa, \tau)$ -covering design where  $v$  is given by a linear function in  $\kappa$ . Then for fixed  $\tau$  there exists a threshold value  $\kappa_0$  such that, beyond this threshold, the Schönheim bound  $L(v, \kappa, \tau)$  remains constant. For example, if  $v = 2\kappa + 3$  we see that for all  $\kappa \geq 9$ , we have  $L(2\kappa + 3, \kappa, 2) = 7$ , while for all  $\kappa \geq 21$ , we have  $L(2\kappa + 3, \kappa, 3) = 15$ . Unfortunately, examples of covering designs which actually achieve this lower bound are rare. However, while it would be desirable to have optimal coverings, from the perspective of complexity a family of coverings of constant size (for a given value of  $\tau$ ) is an acceptable solution. So the following construction (suggested by F. Petrov<sup>1</sup>) is very useful.

**Proposition 6.** *Let  $v = a\kappa + b$  (where  $a \neq 0$  and  $b$  are fixed constants), and let  $\tau$  be a fixed constant. Then, provided  $\kappa$  is sufficiently large, there is a  $(v, \kappa, \tau)$ -covering design of size bounded by a constant dependent only on  $a$  and  $\tau$ .*

<sup>1</sup>Personal communication via mathoverflow.net, February 2016.

*Proof.* Let  $m = \lfloor \kappa/\tau \rfloor$  and  $s = \lceil v/m \rceil$ . Form a partition  $\Pi$  of the set of  $v = a\kappa + b$  points into  $s$  subsets, including as many as possible of size  $m$  and (unless  $\kappa$  divides  $\tau$ ) one of smaller size. By the division algorithm,  $\kappa = m\tau + \rho$ , where  $0 \leq \rho \leq \tau - 1$ . Then we have

$$s = \left\lceil \frac{v}{m} \right\rceil = \left\lceil \frac{a\kappa + b}{m} \right\rceil = \left\lceil \frac{a(m\tau + \rho) + b}{m} \right\rceil = a\tau + \left\lceil \frac{a\rho + b}{m} \right\rceil = a\tau + \left\lceil \frac{\tau(a\rho + b)}{\kappa - \rho} \right\rceil.$$

The quantity  $\frac{\tau(a\rho + b)}{\kappa - \rho}$  is zero if and only if both  $b = 0$  and  $\rho = 0$ ; furthermore, since  $a$ ,  $b$  and  $\tau$  are constants and  $\rho < \tau$ , we have that

$$\lim_{\kappa \rightarrow \infty} \frac{\tau(a\rho + b)}{\kappa - \rho} = 0.$$

Thus, by the definition of a limit, there exists some  $\kappa_1$  such that for all  $\kappa \geq \kappa_1$ ,  $\frac{\tau(a\rho + b)}{\kappa - \rho} \leq 1$ , and therefore  $s \in \{a\tau, a\tau + 1\}$ .

We form a covering design as follows: for any combination of  $\tau$  of the sets in  $\Pi$ , take any  $\kappa$ -subset of points which contains their union. Then this collection of  $\kappa$ -subsets forms the blocks of an  $(a\kappa + b, \kappa, \tau)$ -covering design: the number of blocks is at most  $\binom{a\tau + 1}{\tau}$ ; this depends only on the constants  $a$  and  $\tau$ .  $\square$

**Example 7.** We will use Proposition 6 to construct a  $(23, 10, 2)$ -covering design. The Schönheim bound gives  $L(23, 10, 2) = 7$ , while the best-known covering has size 8 (see [15]); we can obtain a covering of size 10 as follows. First, we partition the set  $\{1, \dots, 23\}$  into  $\lceil 23/(10/2) \rceil = 5$  subsets, four of which have size  $10/2 = 5$  and the remaining set has size 3:

$$1 \ 2 \ 3 \ 4 \ 5 \mid 6 \ 7 \ 8 \ 9 \ 10 \mid 11 \ 12 \ 13 \ 14 \ 15 \mid 16 \ 17 \ 18 \ 19 \ 20 \mid 21 \ 22 \ 23$$

Then we form the blocks of our covering by taking the unions of any pair of these subsets, adding extra points arbitrarily if required. The blocks are the rows of the array below (where \* indicates symbols which can be replaced arbitrarily):

1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	11	12	13	14	15
1	2	3	4	5	16	17	18	19	20
1	2	3	4	5	21	22	23	*	*
6	7	8	9	10	11	12	13	14	15
6	7	8	9	10	16	17	18	19	20
6	7	8	9	10	21	22	23	*	*
11	12	13	14	15	16	17	18	19	20
11	12	13	14	15	21	22	23	*	*
16	17	18	19	20	21	22	23	*	*

In fact, for any sufficiently large value of  $\kappa$ , Proposition 6 will yield a  $(2\kappa + 3, \kappa, 2)$ -covering with  $\binom{5}{2} = 10$  blocks.

## 4 Codes from paths and cycles

In this section, we show how two straightforward classes of graphs—namely paths and cycles—may be used to obtain families of  $(G, k)$ -codes to which our decoding algorithm can be applied. While the

parameters for codes obtained from paths and cycles are very similar, the key distinction is that the alphabet size is smaller for codes from cycles than for codes from paths, on account of the diameter of a cycle  $C_n$  being (approximately) half that of a path  $P_n$ .

## 4.1 Paths

Let  $P_n$  denote a path on  $n$  vertices, which has a  $k$ -resolving set for  $k \leq n - 1$ , and for  $k \geq 3$  has  $\dim_k(P_n) = k + 1$ ; see [12] for details. (We remark that  $\dim_k(P_n) = k$  for  $k = 1$  and  $k = 2$ , but this is of no interest from the perspective of error-correction.) A  $(P_n, k)$ -code will therefore have  $n$  codewords over an alphabet of size  $\text{diam}(P_n) = n - 1$ , of length  $\ell = \dim_k(P_n) = k + 1$ , and with minimum distance at least  $k$ , so can correct  $r = \lfloor (k - 1)/2 \rfloor$  errors.

We note that in the extreme case where  $k = n - 1$ , the  $k$ -metric dimension is  $k + 1 = n$ , and thus every vertex is required in a  $k$ -resolving set. This also means that the alphabet size, length of the codewords and number of codewords are all equal, making these codes comparable to Latin squares as permutation codes (but with minimum distance one less).

**Example 8.** Consider a path  $P_5$  on 5 vertices, and let  $k = 4$ . All 5 vertices are needed in a 4-resolving set, and the code obtained is as follows:

Vertex	Codeword
1	0 1 2 3 4
2	1 0 1 2 3
3	2 1 0 1 2
4	3 2 1 0 1
5	4 3 2 1 0

This code has 5 codewords over the alphabet  $\{0, 1, 2, 3, 4\}$ , and has minimum distance 4, so can correct  $\lfloor (4 - 1)/2 \rfloor = 1$  error.

In order to decode a  $(P_n, k)$ -code, we will require an uncovering with parameters  $(k + 1, \ell - \lfloor (k - 1)/2 \rfloor, r')$ , or equivalently a  $(k + 1, \lfloor (k - 1)/2 \rfloor, r')$ -covering design. By letting  $m = \lfloor (k - 1)/2 \rfloor$ , this is either a  $(2m + 2, m, r')$ -covering when  $k$  is odd, or a  $(2m + 3, m, r')$ -covering when  $k$  is even. In either case, for a given value of  $r'$ , we can obtain appropriate covering designs from Proposition 6. For example, for any path  $P_n$  with  $n \geq 23$  may use the uncovering arising from Example 7 to correct two errors.

## 4.2 Cycles

For a cycle  $C_n$  on  $n$  vertices, we must consider the cases where  $n$  is odd or even separately; details of  $k$ -resolvability of cycles were given in [5].

### 4.2.1 Odd cycles

When  $n$  is odd,  $C_n$  has a  $k$ -resolving set for  $k \leq n - 1$  with  $\dim_k(C_n) = k + 1$  for all  $k$ . A  $(C_n, k)$ -code will therefore have  $n$  codewords of length  $\ell = \dim_k(C_n) = k + 1$ , and with minimum distance  $k$ , so can correct  $r = \lfloor (k - 1)/2 \rfloor$  errors; all of these parameters are identical to those for a path  $P_n$ , but this time the codewords are over an alphabet of size  $\text{diam}(C_n) = (n - 1)/2$ . As the alphabet size does not affect the uncovering needed for decoding, any uncovering for a  $(P_n, k)$ -code may also be used for a  $(C_n, k)$ -code when  $n$  is odd.

**Example 9.** Consider a cycle  $C_5$  on 5 vertices (labelled  $0, \dots, 4$ ), and let  $k = 4$ . All 5 vertices are needed in a 4-resolving set, and the code obtained is as follows:

Vertex	Codeword
0	0 1 2 2 1
1	1 0 1 2 2
2	2 1 0 1 2
3	2 2 1 0 1
4	1 2 2 1 0

This has 5 codewords over the alphabet  $\{0, 1, 2\}$ , and has minimum distance 4, so can therefore correct  $\lfloor (4-1)/2 \rfloor = 1$  error.

We note that, because the underlying graph is a cycle, any codeword may be obtained from another by a cyclic permutation.

### 4.2.2 Even cycles

When  $n$  is even, a little more care is required. Letting  $n = 2q$ , we have that  $C_n$  has a  $k$ -resolving set for  $k \leq n-2 = 2q-2$ , and  $\dim_k(C_n) = k+1$  for  $k \leq q-1$ , or  $\dim_k(C_n) = k+2$  for  $q \leq k \leq 2q-2 = n-2$ . Thus for  $k \leq q-1$ , the parameters of a  $(C_n, k)$ -code when  $n$  is even (as well as those of the uncovering needed for decoding) are the same as those from an odd cycle (although with an alphabet of size  $q = n/2$ ). For  $q \leq k \leq 2q-2$ , they are slightly different: we have  $n$  codewords of length  $k+2$  and minimum distance  $k$ . For decoding, we will need a  $(k+2, k+2 - \lfloor (k-1)/2 \rfloor, r')$ -uncovering, or equivalently a  $(k+2, \lfloor (k-1)/2 \rfloor, r')$ -covering design. Letting  $m = \lfloor (k-1)/2 \rfloor$ , we require either a  $(2m+3, m, r')$ -covering or a  $(2m+4, m, r')$ -covering if  $k$  is odd or even respectively; Proposition 6 is applicable in either case.

**Example 10.** Consider a cycle  $C_6$  on 6 vertices (labelled  $0, \dots, 5$ ), and let  $k = 4$ . All 6 vertices are needed in a 4-resolving set, and the code obtained is as follows:

Vertex	Codeword
0	0 1 2 3 2 1
1	1 0 1 2 3 2
2	2 1 0 1 2 3
3	3 2 1 0 1 2
4	2 3 2 1 0 1
5	1 2 3 2 1 0

This code has 6 codewords over the alphabet  $\{0, 1, 2, 3\}$ , and has minimum distance 4, so can correct  $\lfloor (4-1)/2 \rfloor = 1$  error.

## 5 Codes from grid graphs

In this section, we will use the family of grid graphs  $P_s \square P_t$ , i.e. the Cartesian product of the paths  $P_s$  and  $P_t$ , to obtain  $(P_s \square P_t, k)$ -codes. As we will show, for any grid graph  $P_s \square P_t$  of order  $st$  and any  $k \in \{1, \dots, s+t-2\}$ , we have that  $\dim_k(P_s \square P_t) = 2k$ . This goes on to provide an interesting infinite family of examples.

## 5.1 The $k$ -metric dimension of grid graphs

Suppose that  $G$  is the grid graph  $P_s \square P_t$ , that  $s \geq t$ , and  $U = \{u_1, u_2, \dots, u_s\}$  and  $V = \{v_1, v_2, \dots, v_t\}$  are the vertex sets of  $P_s$  and  $P_t$ , respectively.

It is clear that, if  $G$  is a  $k$ -metric dimensional graph, then for every positive integer  $k' \leq k$ ,  $G$  also has a  $k'$ -metric basis. Next we present a characterization of  $k$ -metric dimensional graphs, obtained in [12], which will be useful in our work. To do so, we need some additional terminology. Given two vertices  $x, y \in V(G)$ , we say that the set of *distinctive vertices* of  $x, y$  is

$$\mathcal{D}(x, y) = \{z \in V(G) : d_G(x, z) \neq d_G(y, z)\}.$$

**Theorem 11** (Estrada-Moreno *et al.* [12]). *A connected graph  $G$  is  $k$ -metric dimensional if and only if  $k = \min_{x, y \in V(G)} |\mathcal{D}(x, y)|$ .*

To compute the  $k$ -metric dimension of a grid graph, we need first to determine for which values of  $k$  there exists a  $k$ -metric basis. This is answered by our next result.

**Theorem 12.** *For any  $s, t \geq 2$ , the graph  $G = P_s \square P_t$  is  $(s+t-2)$ -metric dimensional.*

*Proof.* First, we consider the vertices  $(u_2, v_1)$  and  $(u_1, v_2)$ . Notice that  $\mathcal{D}_G((u_2, v_1), (u_1, v_2)) = (U \times \{v_1\}) \cup (\{u_1\} \times V) - \{(u_1, v_1)\}$ . Thus,  $|\mathcal{D}_G((u_2, v_1), (u_1, v_2))| = s+t-2$  and  $G$  is  $k$ -metric dimensional for some  $k \leq s+t-2$ .

On the other hand, let  $(u_i, v_j)$  and  $(u_g, v_h)$  be two distinct vertices of  $G$ . We consider the following cases.

Case 1:  $i = g$ . Hence  $j \neq h$  and it follows that  $U \times \{v_j, v_h\} \subseteq \mathcal{D}_G((u_i, v_j), (u_g, v_h))$ . Also,

$$|(\{u_i\} \times V) \cap \mathcal{D}_G((u_i, v_j), (u_g, v_h))| \geq t-1.$$

Thus we have

$$\begin{aligned} |\mathcal{D}_G((u_i, v_j), (u_g, v_h))| &\geq |U \times \{v_j, v_h\}| + |(\{u_i\} \times V) \cap \mathcal{D}_G((u_i, v_j), (u_g, v_h))| - 2 \\ &\geq 2s+t-3 \\ &\geq s+t-1. \end{aligned}$$

Case 2:  $j = h$ . Analogous to Case 1 above, we obtain that  $|\mathcal{D}_G((u_i, v_j), (u_g, v_h))| \geq s+t-1$ .

Case 3:  $i \neq g$  and  $j \neq h$ . We may assume that  $i < g$ . Hence we have one of the following situations.

- If  $j < h$ , then we notice that at most two vertices of the set  $(\{u_1, \dots, u_g\} \times \{v_j\}) \cup (\{u_i, \dots, u_s\} \times \{v_h\}) \cup (\{u_i\} \times \{v_1, \dots, v_h\}) \cup (\{u_g\} \times \{v_j, \dots, v_t\})$  do not belong to the set  $\mathcal{D}_G((u_i, v_j), (u_g, v_h))$  (see Figure 1 for an example with  $i = 3$ ,  $j = 2$ ,  $g = 9$  and  $h = 6$ ).

Consequently,

$$\begin{aligned} |\mathcal{D}_G((u_i, v_j), (u_g, v_h))| &\geq |\{u_1, \dots, u_g\} \times \{v_j\}| + |\{u_i, \dots, u_s\} \times \{v_h\}| + \\ &\quad + |\{u_i\} \times \{v_1, \dots, v_h\}| + |\{u_g\} \times \{v_j, \dots, v_t\}| - 4 \\ &\geq g+s-i+1+h+t-j+1-4 \\ &= s+t+g+h-i-j-2 \\ &\geq s+t+i+j-i-j-2 \quad (\text{since } i < g \text{ and } j < h) \\ &= s+t-2. \end{aligned}$$

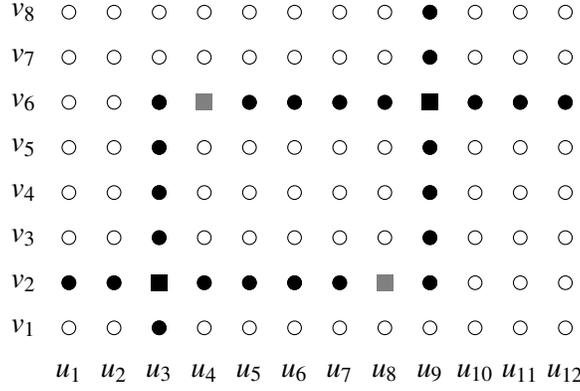


Figure 1: A sketch of the graph  $P_{12} \square P_8$  (edges have not been drawn). Squared bolded vertices have the same distance to the squared grey vertices, but different distance to the circled bolded vertices.

- If  $j > h$ , then a similar procedure gives that  $|\mathcal{D}_G((u_i, v_j), (u_g, v_h))| \geq s + t - 2$ .

As a consequence, we obtain that  $G$  is  $(s + t - 2)$ -metric dimensional.  $\square$

Having established that  $G = P_s \square P_t$  is  $(s + t - 2)$ -metric dimensional, next we obtain its  $k$ -metric dimension for every  $k \in \{1, \dots, s + t - 2\}$ . As an observation regarding the set of distinctive vertices of a pair of vertices  $x, y$ , we notice that if  $S$  is a  $k$ -resolving set for a graph  $G$ , then  $|\mathcal{D}(x, y) \cap S| \geq k$ . This simple fact will be used frequently in our next proof.

**Theorem 13.** For any grid graph  $G = P_s \square P_t$  and every  $k \in \{1, \dots, s + t - 2\}$ ,

$$\dim_k(G) = 2k.$$

*Proof.* If  $k = 1$ , then it is already known (see [8]) that  $\dim_1(G) = \dim(G) = 2$ , so from now on we consider only  $k \geq 2$ . Without loss of generality, we suppose that  $s \geq t$ . Let  $(u_i, v_j)$  and  $(u_g, v_h)$  be two distinct vertices of  $G$ . Next we consider the following three cases according to the value of  $k$ .

Case 1:  $k \leq s$ . Let  $S = \{u_1, \dots, u_k\} \times \{v_1, v_t\}$ . We consider the following subcases.

Subcase 1.1:  $i = g$ . Hence,  $j \neq h$  and for any vertex  $(u_p, v_q) \in S$  it follows  $d_G((u_i, v_j), (u_p, v_q)) \neq d_G((u_g, v_h), (u_p, v_q))$ . Thus,  $(u_i, v_j)$  and  $(u_g, v_h)$  are distinguished by  $2k$  vertices of  $S$ .

Subcase 1.2:  $i \neq g$ . Hence, without loss of generality we assume that  $i < g$ . Moreover, we consider  $j \leq h$ . First notice that  $\mathcal{D}_G((u_i, v_j), (u_g, v_h)) \supseteq \{u_1, \dots, u_i\} \times \{v_1\}$ . If  $i \geq k$ , then clearly  $|\mathcal{D}_G((u_i, v_j), (u_g, v_h)) \cap S| \geq k$ . Hence, we may assume  $i < k$ . If  $g \geq k$ , then there is at most one vertex  $(u_f, v_1) \in \{u_{i+1}, \dots, u_k\} \times \{v_1\}$  such that  $d_G((u_i, v_j), (u_f, v_1)) = d_G((u_g, v_h), (u_f, v_1))$ , which leads to  $|\mathcal{D}_G((u_i, v_j), (u_g, v_h)) \cap (\{u_1, \dots, u_k\} \times \{v_1\})| \geq k - 1$ . On the other hand, since  $i < k$ , there is at least one vertex  $(u_d, v_t) \in \{u_1, \dots, u_k\} \times \{v_t\}$  such that  $d_G((u_i, v_j), (u_d, v_t)) \neq d_G((u_g, v_h), (u_d, v_t))$ . As a consequence,

$$\begin{aligned} |\mathcal{D}_G((u_i, v_j), (u_g, v_h)) \cap S| &= |\mathcal{D}_G((u_i, v_j), (u_g, v_h)) \cap (\{u_1, \dots, u_k\} \times \{v_1\})| + \\ &\quad + |\mathcal{D}_G((u_i, v_j), (u_g, v_h)) \cap (\{u_1, \dots, u_k\} \times \{v_t\})| \\ &\geq k. \end{aligned}$$

We now consider  $g < k$ . Thus,  $\mathcal{D}_G((u_i, v_j), (u_g, v_h)) \supseteq \{u_g, \dots, u_k\} \times \{v_t\}$  and at most one vertex of the set  $\{u_1, \dots, u_g\} \times \{v_1\}$  does not belong to  $\mathcal{D}_G((u_i, v_j), (u_g, v_h))$ . Moreover, at least one vertex of the set  $\{u_i, \dots, u_g\} \times \{v_t\}$  belongs to  $\mathcal{D}_G((u_i, v_j), (u_g, v_h))$ . As a consequence,

$$\begin{aligned} |\mathcal{D}_G((u_i, v_j), (u_g, v_h)) \cap S| &\geq |\{u_g, \dots, u_k\} \times \{v_t\}| + |\mathcal{D}_G((u_i, v_j), (u_g, v_h)) \cap (\{u_1, \dots, u_g\} \times \{v_1\})| + \\ &\quad + |\mathcal{D}_G((u_i, v_j), (u_g, v_h)) \cap (\{u_i, \dots, u_g\} \times \{v_t\})| \\ &\geq k - g + 1 + g - 1 + 1 \\ &= k. \end{aligned}$$

See Figure 2 for examples of the situations above considering  $k = 7$ .

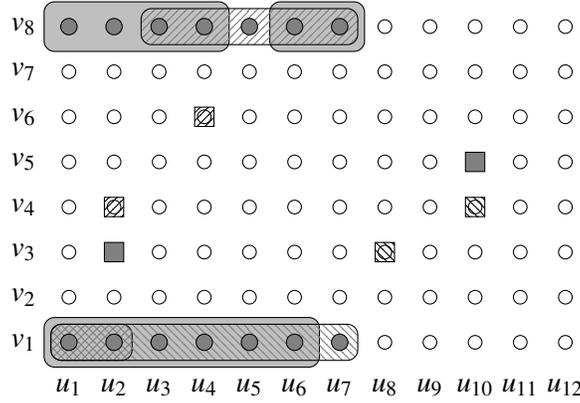


Figure 2: A sketch of the graph  $P_{12} \square P_8$  (edges have not been drawn). Each pair of squared vertices with identically filled shapes is recognized by vertices in the rounded rectangles with identically filled areas, respectively.

Finally, if  $j > h$ , then a symmetric procedure gives an analogous result and we observe that  $S$  is a  $k$ -resolving set for  $G$ .

Case 2:  $s < k \leq s + t - 2$ . Note that  $s \geq 3$ , since the case  $s = 2$  would lead to the graph  $C_4$ , which is 2-metric dimensional. Assume that  $k = s + \alpha$  for some integer  $\alpha \geq 1$ . Let  $S = (U \times \{v_1, v_t\}) \cup (\{u_1, u_s\} \times \{v_2, \dots, v_{\alpha+1}\})$ . According to Case 1, the set  $S_1 = U \times \{v_1, v_t\} \subset S$  is a  $s$ -resolving set of  $G$ , which means that any pair of vertices of  $G$  is recognized by at least  $s$  vertices of  $S_1$ . On the other hand, also by Case 1 (in a symmetric version), the set  $S_2 = \{u_1, u_s\} \times \{v_2, \dots, v_{\alpha+1}\} \subset S$  is an  $(\alpha + 1)$ -resolving set of  $G$ , which similarly means any pair of vertices of  $G$  is recognized by at least  $\alpha + 1$  vertices of  $S_2$ . Since only two vertices, *i.e.*  $(u_1, v_1)$  and  $(u_s, v_1)$ , belong to both sets  $S_1$  and  $S_2$ , it must happen that at least  $s + \alpha + 1 - 2$  vertices of  $S$  recognize each pair of vertices of  $G$ . Suppose that there is a pair of vertices  $(u_i, v_j)$  and  $(u_g, v_h)$  such that they are distinguished by exactly  $s + \alpha - 1$  vertices of  $S$ . Thus, they are distinguished by  $(u_1, v_1)$  and  $(u_s, v_1)$ , by  $s - 2$  vertices in  $S_1 \setminus \{(u_1, v_1), (u_s, v_1)\}$  and by  $\alpha - 1$  vertices in  $S_2 \setminus \{(u_1, v_1), (u_s, v_1)\}$ . However, we can now notice that if  $(u_i, v_j)$  and  $(u_g, v_h)$  are distinguished by  $(u_1, v_1)$  and  $(u_s, v_1)$ , then  $(u_i, v_j)$  and  $(u_g, v_h)$  are not distinguished by at most one vertex in  $U \times \{v_1\}$  and by at most one vertex in  $U \times \{v_t\}$ , that is,  $2s - 2 > s$  (since  $s \geq 3$ ) vertices of  $S$ , which is a contradiction. Thus, each pair of vertices of  $G$  is distinguished by at least  $s + \alpha = k$  vertices of  $S$ . Therefore,  $S$  is a  $k$ -resolving set of  $G$ . As a consequence of both cases, we obtain that  $\dim_k(G) \leq 2k$ .

We now want to show that no smaller  $k$ -resolving set can exist. Let  $S'$  be a  $k$ -metric basis for  $G$ . We consider the following vertices:  $(u_1, v_2)$ ,  $(u_2, v_1)$ ,  $(u_s, v_{t-1})$  and  $(u_{s-1}, v_t)$ . Since

$$|A| = |\mathcal{D}_G((u_1, v_2), (u_2, v_1)) \cap S'| = |(U \times \{v_1\}) \cup (\{u_1\} \times V) - \{(u_1, v_1)\} \cap S'| \geq k$$

and

$$|B| = |\mathcal{D}_G((u_s, v_{t-1}), (u_{s-1}, v_t)) \cap S'| = |(U \times \{v_t\}) \cup (\{u_s\} \times V) - \{(u_s, v_t)\} \cap S'| \geq k$$

we have that

$$|S'| \geq |A| + |B| - 2 \geq 2k - 2.$$

Now suppose that  $|S'| = 2k - 2$ . Thus, it follows that  $|A| = k$ ,  $|B| = k$ ,  $(u_s, v_1), (u_1, v_t) \in S$  and  $S = A \cup B$ . Thus  $|(\{u_s\} \times \{v_2, \dots, v_{t-1}\}) \cup (\{u_2, \dots, u_{s-1}\} \times \{v_t\}) \cap S'| = k - 1$  and  $|(\{u_1\} \times \{v_2, \dots, v_{t-1}\}) \cup (\{u_2, \dots, u_{s-1}\} \times \{v_1\}) \cap S'| = k - 1$ . We now consider the vertices  $(u_1, v_1)$ ,  $(u_2, v_2)$ ,  $(u_{s-1}, v_{t-1})$  and  $(u_s, v_t)$ . Hence,

$$|\mathcal{D}_G((u_1, v_1), (u_2, v_2)) \cap S'| \leq |((\{u_s\} \times \{v_2, \dots, v_{t-1}\}) \cup (\{u_2, \dots, u_{s-1}\} \times \{v_t\})) \cap S'| = k - 1$$

and

$$|\mathcal{D}_G((u_{s-1}, v_{t-1}), (u_s, v_t)) \cap S'| \leq |(\{u_1\} \times \{v_2, \dots, v_{t-1}\}) \cup (\{u_2, \dots, u_{s-1}\} \times \{v_1\}) \cap S'| = k - 1,$$

which is a contradiction. So  $|S'| \geq 2k - 1$ . If we suppose that  $|S'| = 2k - 1$ , then an analogous procedure to the one above gives a contradiction again. Therefore, we have that  $|S'| \geq 2k$  and the proof is complete.  $\square$

## 5.2 Decoding

For the grid graphs  $G = P_s \square P_t$  described above, and for any  $k \leq s + t - 2$ , we obtain  $(G, k)$ -codes with  $n = st$  codewords of length  $\ell = 2k$  over an alphabet of size  $s + t + 1$ . Such a  $(G, k)$ -code has correction capability  $r = \lfloor (k - 1)/2 \rfloor$ . To decode  $r' < r$  errors using the algorithm described above will require a  $(2k, 2k - r, r')$ -uncovering, or equivalently a  $(2k, r, r')$ -covering design. Depending on the parity of  $k$ , we therefore require either: (i) if  $k = 2m$  is even, a  $(4m, m - 1, r')$ -covering design; or (ii) if  $k = 2m + 1$  is odd, a  $(4m + 2, m, r')$ -covering design. Proposition 6 provides the coverings (and thus uncoverings) we require. Given that these uncoverings have constant size, the complexity of the decoding algorithm in this case is  $O(kn)$ .

## 6 Conclusion

The main achievement of this paper was to obtain a new application of  $k$ -resolving sets in graphs to coding theory, by obtaining a new method of constructing error-correcting codes. We considered three families of graphs, namely paths, cycles and grid graphs; naturally, there are many more graph families which could be investigated with this application in mind.

We conclude by mentioning that, while 1-resolving sets have many applications, for the application of  $k$ -resolving sets to error-correcting codes we require that  $k \geq 3$ ; however, 2-resolving sets could potentially be applied to the detection of errors.

## References

- [1] R. F. Bailey, Uncoverings-by-bases for base-transitive permutation groups, *Des. Codes Cryptogr.* **41** (2006), 153–176.
- [2] R. F. Bailey, Error-correcting codes from permutation groups, *Discrete Math.* **309** (2009), 4253–4265.
- [3] R. F. Bailey and P. J. Cameron, Base size, metric dimension and other invariants of groups and graphs, *Bull. London Math. Soc.* **43** (2011), 209–242.
- [4] R. F. Bailey and B. Stevens, Uncoverings on graphs and network reliability, *Australas. J. Combin.* **50** (2011), 219–231.
- [5] A. F. Beardon and J. A. Rodríguez-Velázquez, On the  $k$ -metric dimension of metric spaces, preprint; arXiv:1603.04049.
- [6] L. M. Blumenthal, *Theory and Applications of Distance Geometry*, Clarendon Press, Oxford, 1953.
- [7] P. J. Cameron, Permutation codes, *European J. Combin.* **31** (2010), 482–490.
- [8] G. Chartrand, L. Eroh, M. A. Johnson and O. R. Oellermann, Resolvability in graphs and the metric dimension of a graph, *Discrete Appl. Math.* **105** (2000), 99–113.
- [9] W. Chu, C. J. Colbourn and P. Dukes, Constructions for permutation codes in powerline communications, *Des. Codes Cryptogr.* **32** (2004), 51–64.
- [10] C. J. Colbourn and J. H. Dinitz (editors), *Handbook of Combinatorial Designs* (second edition), CRC Press, Boca Raton, 2007.
- [11] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, *Introduction to Algorithms* (second edition), MIT Press, Cambridge/McGraw-Hill, Boston, 2001.
- [12] A. Estrada-Moreno, J. A. Rodríguez-Velázquez and I. G. Yero, The  $k$ -metric dimension of a graph, *Appl. Math. Inf. Sci.* **9** (2015), 2829–2840.
- [13] A. Estrada-Moreno, I. G. Yero and J. A. Rodríguez-Velázquez, The  $k$ -metric dimension of corona product graphs, *Bull. Malaysian Math. Sci. Soc.* **39:1** (2016), 135–156.
- [14] A. Estrada-Moreno, I. G. Yero and J. A. Rodríguez-Velázquez, The  $k$ -metric dimension of the lexicographic product of graphs, *Discrete Math.* **339** (2016), 1924–1934.
- [15] D. M. Gordon, La Jolla Covering Repository, <http://www.ccrwest.org/cover.html>.
- [16] F. Harary and R. A. Melter, On the metric dimension of a graph, *Ars Combin.*, **2** (1976), 191–195.
- [17] S. Khuller, B. Raghavachari and A. Rosenfeld, Landmarks in graphs, *Discrete Appl. Math.* **70** (1996), 217–229.
- [18] H.-J. Kroll and R. Vincenti, Antiblocking systems and PD-sets, *Discrete Math.* **308** (2008), 401–407.

- [19] W. H. Mills and R. C. Mullin, Coverings and packings, in *Contemporary Design Theory: A collection of surveys*, (eds J. H. Dinitz and D. R. Stinson), John Wiley & Sons, New York, 1992.
- [20] V. S. Pless, *Introduction to the Theory of Error-Correcting Codes* (third edition), John Wiley & Sons, New York, 1998.
- [21] J. Schönheim, On coverings, *Pacific J. Math* **14** (1964), 1405–1411.
- [22] A. Sebő and E. Tannier, On metric generators of graphs, *Math. Oper. Res.* **29** (2004), 383–393.
- [23] P. J. Slater, Leaves of trees, *Congr. Numer.* **14** (1975), 549–568.
- [24] I. Tamo and M. Schwartz, Correcting limited-magnitude errors in the rank-modulation scheme, *IEEE Trans. Inform. Theory* **56** (2010), 2551–2560.
- [25] I. G. Yero, A. Estrada-Moreno and J. A. Rodríguez-Velázquez, Computing the  $k$ -metric dimension of graphs, *Applied Mathematics and Computation* **300** (2017), 60–69.