Multiply Constant-Weight Codes and the Reliability of Loop Physically Unclonable Functions

Yeow Meng Chee, *Senior Member, IEEE*, Zouha Cherif, Jean-Luc Danger, Sylvain Guilley, Han Mao Kiah, Jon-Lark Kim, Patrick Solé, and Xiande Zhang

***Abstract*— We introduce the class of multiply constant-weight codes to improve the reliability of certain physically unclonable function response, and extend classical coding methods to construct multiply constant-weight codes from known q-ary and constant-weight codes. We derive analogs of Johnson bounds and give constructions showing these bounds to be asymptotically tight up to a constant factor under certain conditions. We also examine the rates of multiply constant-weight codes and demonstrate that these rates are the same as those of constant-weight codes of corresponding parameters.**

***Index Terms*— Constant-weight codes, doubly constant-weight codes, multiply constant-weight codes, physically unclonable functions.**

I. Introduction

PHYSICALLY unclonable functions (PUFs), introduced by Pappu *et al.* [1], provide innovative low-cost authentication methods that are derived from complex physical characteristics of electronic devices. Recently, PUFs have become an attractive option to provide security in low cost devices such as RFIDs and smart cards [1]–[4]. Reliability and implementation considerations on programmable circuits for the design of Loop PUFs [4] lead to the investigation of a new class of codes called *multiply constant-weight codes* (MCWC).

In an MCWC, each codeword is a binary word of length mn which is partitioned into m equal parts and has weight exactly w in each part [5]. This definition therefore generalizes the class of *constant-weight codes* (where $m = 1$) and a subclass of *doubly constant-weight codes*, introduced by Johnson [6] and Levenshteĭn [7] (where $m = 2$).

In this paper, we consider upper and lower bounds for the possible sizes of MCWCs. Our constructions make use of both classical concatenation techniques [8], [9] and a method due to Zinoviev (for constant-weight codes) [10], that was later independently given by Etzion (for doubly constant-weight codes) [11]. A construction technique using resolvable designs is also examined. For upper bounds, we extend the techniques of Johnson [6] and exhibit that these bounds are asymptotically tight to a constant factor, provided m, w and d are fixed. We also examine the rates of the MCWCs and interestingly, demonstrate that these rates are the same as those of constant-weight codes of length mn and weight mw.

We remark that if the codewords in an MCWC are regarded as m by n arrays, then an MCWC can be regarded as a code over binary matrices, where each matrix has constant row weight w. These codes were studied by Chee *et al.* [12] in an application for power line communications. The relevance of MCWCs for the latter context is an area for future research.

The rest of this article is structured as follows. Section II collects the necessary definitions and notation, and Section III examines an application of MCWCs in the field of PUFs. Section IV deals with constructions and attached lower bounds, while Section V contains the upper bounds. Section VI studies asymptotic versions of the bounds of Section IV and Section V. Some of our results were initially reported in [5] and the present paper contains many new results and generalizations.

II. Definitions and Notation

Let $\mathcal{X}$ be a set of q symbols. A q-ary *code* C of length n over the alphabet $\mathcal{X}$ is a subset of $\mathcal{X}^n$. Elements of C are called *codewords*. Endow the space $\mathcal{X}^n$ with the *Hamming distance* metric. A code C is said to have *distance* d if the (Hamming) distance between any two distinct codewords of C is at least d. A q-ary code of length n and distance d is called an $(n, d)_q$ code.

When $q = 2$, we assume $\mathcal{X} = \mathbb{F}_2$. An $(n, d)_2$ code is simply called an (n, d) code. The (Hamming) *weight* of

Manuscript received November 5, 2013; accepted July 29, 2014. Date of publication September 26, 2014; date of current version October 16, 2014. The work of Y. M. Chee, H. M. Kiah, and X. Zhang was supported in part by the Singapore National Research Foundation under Grant NRF-CRP2-2007-03. The work of Z. Cherif, J.-L. Danger, and S. Guilley was supported in part by Orange Labs and in part by the ENIAC European Project 2010-1 TOISE-Trusted Computing for European Embedded Systems. The work of J.-L. Kim was supported in part by the Basic Research Programme through the Ministry of Education, National Research Foundation of Korea, under Grant NRF-2013R1A1A2005172, and in part by the Sogang University, Seoul, Korea, under Grant 201210058.01. This paper was presented at the 2013 IEEE International Symposium on Information Theory.

Y. M. Chee and X. Zhang are with the Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 637371 (e-mail: ymchee@ntu.edu.sg; xiandezhang@ntu.edu.sg).

Z. Cherif is with the Institut Mines-Telecom, Telecom ParisTech, Paris 75634, France, and also with the Laboratoire Hubert Curien, Université de Lyon, Saint-Étienne 42000, France (e-mail: zouha.cherif@telecom-paristech.fr).

J.-L. Danger and S. Guilley are with the Institut Mines-Telecom, Telecom ParisTech, Paris 75634, France, and also with Secure-IC S.A.S., Rennes 35700, France (e-mail: jean-luc.danger@telecom-paristech.fr; sylvain.guilley@telecom-paristech.fr).

H. M. Kiah was with the Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 637371. He is now with the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Champaign, IL 61801 USA (e-mail: hmkiah@illinois.edu).

J.-L. Kim is with the Department of Mathematics, Sogang University, Seoul 121-742, Korea (e-mail: jlkim@sogang.ac.kr).

P. Solé is with the Institut Mines-Telecom, Telecom ParisTech, Paris 75634, France, and also with the Department of Mathematics, King Abdulaziz University, Jeddah 22254, Saudi Arabia (e-mail: patrick.sole@telecom-paristech.fr).

Communicated by A. Ashikhmin, Associate Editor for Coding Techniques.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIT.2014.2359207

TABLE I
TABLE OF NOTATION

Class of codes	Notation	Optimal size	Asymptotic rate
General q-ary code	$(n,d)_q$ code	$A_q(n,d)$	$\alpha_q(\delta)$
General binary code	(n,d) code	$A(n,d)$	$\alpha(\delta)$
Multiply constant-weight code	MCWC$(w_1,n_1;\ldots;w_m,n_m;d)$	$T(w_1,n_1;\ldots;w_m,n_m;d)$	–
Multiply constant-weight code with $n_1 = \cdots = n_m = n$ and $w_1 = \cdots = w_m = w$	MCWC(m,n,d,w)	$M(m,n,d,w)$	$\mu(\delta,\omega)$
(Usual) constant-weight code with	CWC(n,d,w)	$A(n,d,w)$	$\alpha(\delta,\omega)$
Binary linear code	$[n,k,d]$ code	$B(n,d)$	–
Systematic binary code	–	$S(n,d) = 2^{s(n,d)}$	$\sigma(\delta)$
Systematic constant-weight code	–	$S(n,d,w) = 2^{s(n,d,w)}$	$\sigma(\delta,\omega)$

a codeword $\mathbf{u} \in \mathcal{X}^n$ is given by the number of nonzero coordinates in $\mathbf{u}$. Fix $m, n_1, n_2, \ldots, n_m$ to be positive integers and let $N = n_1 + n_2 + \cdots + n_m$. An $(N,d)_2$ code is said to be of *multiply constant-weight* and denoted by MCWC$(w_1,n_1;w_2,n_2;\ldots;w_m,n_m;d)$, if each codeword has weight w_1 in the first n_1 coordinates, weight w_2 in the next n_2 coordinates, and so on and so forth. When $m = 1$, an MCWC$(n,w;d)$ is a *constant-weight code*, denoted by CWC(n,d,w); when $m = 2$, an MCWC$(w_1,n_1;w_2,n_2;d)$ is a *doubly constant-weight code*.

When $w_1 = w_2 = \cdots = w_m = w$ and $n_1 = n_2 = \cdots = n_m = n$, we simply denote this multiply constant-weight code of length $N = mn$ by MCWC(m,n,d,w). Unless specified otherwise, a multiply constant code refers to an MCWC(m,n,d,w) in this paper.

Example 2.1: The code $\{0101, 0110, 1010\}$ is an MCWC$(1,2;1,2;2)$ or an MCWC$(2,2,1,2)$ with $m = 2$, $n_1 = n_2 = n = 2$, $w_1 = w_2 = w = 1$ and $d = 2$. We also observe that it is also a CWC$(4,2,2)$. On the other hand, the code $\{0101, 0110, 1010, 0011\}$ is *not* multiply constant-weight, but it is also a CWC$(4,2,2)$.

The largest size of an $(n,d)_q$ code is denoted by $A_q(n,d)$. When $q = 2$, this size is simply denoted by $A(n,d)$. The largest of size of an MCWC$(w_1,n_1;w_2,n_2;\ldots;w_m,n_m;d)$ is given by $T(w_1,n_1;w_2,n_2;\ldots;w_m,n_m;d)$; the largest of size of an MCWC(m,n,d,w) is given by $M(m,n,d,w)$; and the largest of size of a CWC(n,d,w) is given by $A(n,d,w)$.

In this paper, we are mainly interested in determining $M(m,n,d,w)$. Observe that by definition,

$$M(1,n,d,w) = A(n,d,w),$$
$$M(2,n,d,w) = T(w,n;w,n;d).$$

Moreover, the functions $A(n,d,w)$ and $T(w,n;w,n;d)$ have been well studied (see [6], [11], [13]–[15]). Online tables of the lower bounds for $A(n,d,w)$ can be found at [16] while upper bounds for $A(n,d,w)$ and $T(w,n;w,n;d)$ can be found at [17].

In this paper, we are mainly interested in building multiply constant-weight codes from known q-ary codes and constant-weight codes. One such class of codes is the class of binary *linear codes*. A binary linear code of length n, dimension k and distance d is called a linear $[n,k,d]$ code and we denote the largest quantity 2^k of a binary linear $[n,k,d]$ code by $B(n,d)$.

Unfortunately, an MCWC cannot be linear and hence, we look at possible generalization of linearity. A possible generalization is given by the notion of systematic codes. A code of size 2^k is said to be *systematic* if there is a set I of k coordinates such that the code when restricted to the coordinate set I is exactly $\mathbb{F}_2^k$. The largest sizes of a systematic (n,d) code and a systematic CWC(n,d,w) are denoted by $S(n,d) = 2^{s(n,d)}$ and $S(n,d,w) = 2^{s(n,d,w)}$ respectively. We remark that systematic constant-weight codes have been studied in [18] and [19]. A summary of the notation are provided in Table I.

Finally, as mentioned in the introduction, a codeword in an MCWC(m,n,d,w) can be regarded as a binary m by n matrix with constant row weight w. Throughout the rest of this paper, we shall regard a codeword in an MCWC as either a word of length mn or an m by n matrix.

III. APPLICATION TO LOOP PUFS

The need of an MCWC arises from the generation of some type of PUFs in trusted electronic circuits. In this section, we demonstrate the relevance of MCWC in the implementation of Loop PUF on Field Programmable Gate Array (FPGA) and in enhancing the reliability of PUF response. First, we present the principle behind Loop PUF.

A. Loop PUF Principle

In general, the PUF provides a unique signature to a device without the need for the user to program an internal memory [1]. This signature allows the user to build lightweight authentication protocols or even protect a master key in cryptographic implementations. Such a key can be used for standard cryptographic protocols, or for internal cryptography (*e.g.*, memory encryption). Essentially, the PUF takes advantage of technological process variations to differentiate between two devices. For instance, consider two delay lines with the same structure. In theory, the propagation time is the same for both two delay lines. However, actual measurements of the propagation time differ between the delay lines due to imbalances between the physical elements. Furthermore, as these measurements cannot be predicted accurately, they are well suited for cryptographic purposes.

Here, we consider the Loop PUF [4] that is a set of n identical delay lines laid out on a programmable circuit. The delay lines form a loop that oscillates as a single ring oscillator

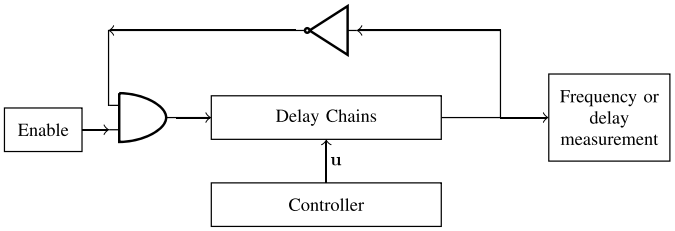

Fig. 1. Loop PUF structure.

when closed by an inverter (see Figure 1) and this setup enhances the accuracy of delay measurements. Furthermore, each delay line is a series of m delay elements and the delay of the ith element of the jth line is controlled by the (i, j)-th bit of some control word $\mathbf{u}$ of length mn. Hence, corresponding to a control word $\mathbf{u}$, we have a delay measurement, denoted by $\mathsf{D}(\mathbf{u})$.

For expository purposes, we illustrate how a general binary code can be used in conjunction with the Loop PUF [4] to generate a set of Challenge-Response pairs for authentication purposes. For other cryptographic applications, we refer the interested reader to [1]–[4].

Given a binary code C of length mn, the set of Challenge-Response pairs is given by

$$\left\{\Big((\mathbf{u}, \mathbf{v}), \mathsf{sign}(\mathsf{D}(\mathbf{u}) - \mathsf{D}(\mathbf{v}))\Big) : \mathbf{u} \neq \mathbf{v}, \mathbf{u}, \mathbf{v} \in C\right\}.$$

In other words, each challenge is an ordered pair of distinct codewords $(\mathbf{u}, \mathbf{v})$ from the binary code and the corresponding response is the sign of the delay difference between the pair of codewords.[1]

For the set of Challenge-Response pairs to be used for authentication, it is important that we are unable to infer the sign of the delay difference with only knowledge of $\mathbf{u}$ and $\mathbf{v}$. To achieve this *unpredictability* of response, we show that C needs to be an MCWC in Section III-B. On the other hand, it is also important that the measured response (or the sign of delay difference) remains the same despite environmental noise. The *reliability* of response is then shown to be associated with the minimum distance of the code C in Section III-C. Therefore, MCWCs are needed to satisfy both requirements of unpredictability and reliability.

B. MCWC to Achieve Unpredictability on FGPAs

Programmable circuits, like FPGAs, have a hierarchical layout. It is thus convenient to organize the PUF with two levels, namely with a structure of n clusters of m cells each.[2] For this technology, it is rather easy to copy/paste exactly the logic of one cluster to generate all of them, in an indistinguishable manner (logically, not physically). Thus the Loop PUF can be easily constructed from a set of n clusters of m cells just by replicating the base cluster of m cells. As the routing inside a cluster between the m elements is not constrained, the PUF designer can easily port this structure to any FPGA family.

[1]Here, we consider a response that consists of *one* bit. A different control strategy can be used to extract a response with more bits and this is described in [4].

[2]Logic Array Block (LABs) for ALTERA and Configurable Logic Blocks (CLBs) for XILINX.

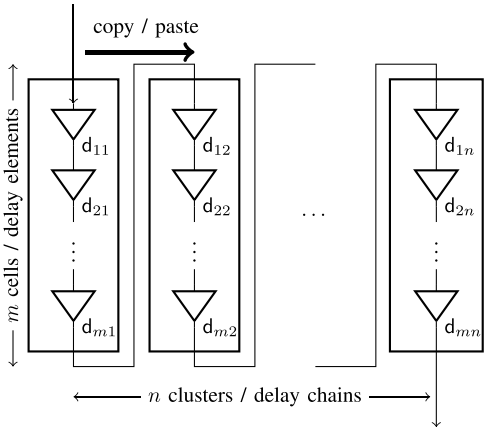

Fig. 2. Delay chain layout.

Consider an MCWC$(n, w_1; \cdots; n, w_m; d)$ and choose a control word $\mathbf{u} = (u_{ij})_{1\le i\le m, 1\le j\le n}$. Let $\mathsf{d}_{ij}(u_{ij})$ be the resulting delay of the ith delay element in the jth line and hence, the total measured delay $\mathsf{D}(\mathbf{u})$ due to $\mathbf{u}$ is given by $\sum_{i=1}^{m}\sum_{j=1}^{n}\mathsf{d}_{ij}(u_{ij})$.

Ideally, $\mathsf{d}_{ij}(u_{ij}) = \mu + \epsilon_{ij}(u_{ij})$, where ϵ_{ij} is a small timing variation on the jth delay element on the ith line caused by technological dispersion and μ is the average delay that is independent of the position on the circuit. However, the latter is not true due to manufacturing constraints. In particular, a designer has no control about the routing within an FPGA cluster and hence, it is hardly possible to get balanced delay elements within a cluster. But fortunately due to copy / paste operation, the internal routing of a cluster can be faithfully reproduced from one cluster to another (see Figure 2).

In other words, we have

$$\mathsf{d}_{ij}(u_{ij}) = \mu_i(u_{ij}) + \epsilon_{ij}(u_{ij}),$$

where ϵ_{ij} is a small timing variation and μ_i is the average delay dependent on the controlled bit and the position of the delay element. We compute the total delay due to $\mathbf{u}$, and we have

$$\begin{aligned}\mathsf{D}(\mathbf{u}) &= \sum_{i=1}^{m}\sum_{j=1}^{n}\mathsf{d}_{ij}(u_{ij}) \\ &= \sum_{i=1}^{m}\sum_{j=1}^{n}\mu_i(u_{ij}) + \epsilon_{ij}(u_{ij}) \\ &= \left(\sum_{i=1}^{m}(n - w_i)\mu_i(0) + w_i\mu_i(1)\right) \\ &\quad + \left(\sum_{i=1}^{m}\sum_{j=1}^{n}\epsilon_{ij}(u_{ij})\right). \end{aligned} \tag{1}$$

The last equality follows from the fact that $\mathbf{u}$ belongs to an MCWC$(n, w_1; n, w_2; \cdots; n, w_m; d)$. Furthermore, we observe that all codewords from the MCWC have the same expected response. Therefore, the delay difference between any pair of control words from the MCWC has expectation zero and the sign of the difference is dependent only

on ϵ_{ij}'s. In other words, the response depends entirely on the unpredictable physical characteristics of the individual delay elements.

C. Hamming Distance to Improve the PUF Reliability

The PUF response is very sensitive to environmental noise as the ϵ_{ij} can be very low in comparison to the delays. Hence it is necessary to choose pairs of control words which offer the largest possible difference between their delays.

From (1), we see that

$$\begin{aligned}\mathsf{D}(\mathbf{u}) - \mathsf{D}(\mathbf{v}) &= \sum_{i=1}^{m}\sum_{j=1}^{n} \epsilon_{ij}(u_{ij}) - \epsilon_{ij}(v_{ij}) \\ &= \sum_{u_{ij}\neq v_{ij}} \epsilon_{ij}(u_{ij}) - \epsilon_{ij}(v_{ij}).\end{aligned}$$

Therefore, the greater the Hamming distance between $\mathbf{u}$ and $\mathbf{v}$, the greater the delay difference $\mathsf{D}(\mathbf{u}) - \mathsf{D}(\mathbf{v})$. Hence, by choosing a code of high distance, we improve the reliability of the PUF response.

The arguments in this section demonstrate the relevance of MCWC in the design of reliable Loop PUF. In the remaining of the paper, we examine the possible lower and upper bounds for optimal MCWCs, focusing our attention to the case where $w_1 = w_2 = \cdots = w_m = w$.

IV. Lower Bounds

A. Coding Constructions

In this section, we study constructions of MCWCs using known unrestricted codes. Our first construction is based on *concatenation*.

Proposition 4.1: Let $q \leq A(n, d_1, w)$. We have

$$M(m, n, d_1 d_2, w) \geq A_q(m, d_2).$$

Proof: Consider a concatenation scheme [8], [9] where the *outer code* C is an $(m, d_2)_q$ code of size $A_q(m, d_2)$ over $\mathcal{X}$ and the *inner code* D is a CWC(n, d_1, w) of size q. Let $\phi : \mathcal{X} \to D$ be an injective map. For each codeword $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_m)$ in C, we construct the binary codeword $(\phi(\mathbf{u}_1), \phi(\mathbf{u}_2), \ldots, \phi(\mathbf{u}_m))$. Then the resulting code is an MCWC$(m, n, d_1 d_2, w)$ of size $A_q(m, d_2)$. ∎

A special case of concatenation is the *product code construction*. Recall that if C and D are two binary linear codes then their product $C \otimes D$ is the code of length nm consisting of m by n arrays whose rows belong to C and columns belong to D. If C and D are linear $[m, k, d]$ and $[n, l, e]$ codes, then the code $C \otimes D$ has parameters $[nm, kl, de]$ [9, Lemma 2.8].

We generalize this construction by relaxing certain requirements.[3] In particular, we require only the rows of our arrays to be in C, while not all the columns need to be in D.

Formally, consider a systematic CWC(n, d_1, w) C of size 2^{k_1} and a systematic (m, d_2) code D of size 2^{k_2}. Given a binary k_2 by k_1 matrix $\mathbf{M}$, we replace each column of length k_2 of $\mathbf{M}$ with its corresponding codeword in D to obtain a binary m by k_1 matrix $\mathbf{M}'$. Next replace each row of length k_1 of $\mathbf{M}'$ with its corresponding codeword in C. This results in a binary m by n matrix with constant row weight w. In particular, each row of the matrix belongs to the constant-weight code C while the first k_1 columns belong to the code D. Hence, the collection of all $2^{k_1 k_2}$ matrices from this construction results in an MCWC$(m, n, d_1 d_2, w)$. We call this construction a *pseudo-product code construction*.

We remark that as with product construction, the pseudo-product code construction is a special case of concatenation. In addition, the pseudo-product construction coincides with the construction given by Amrani [20, Definition 1]. The following follows immediately from the pseudo-product code construction.

Proposition 4.2: We have

$$\begin{aligned}M(m, n, d_1 d_2, w) &\geq 2^{s(n,d_1,w)s(m,d_2)} \\ &\geq B(m, d_2)^{s(n,d_1,w)}.\end{aligned}$$

Example 4.1: Consider the systematic constant-weight code $\{0011, 0101, 1010, 1100\}$ of distance two. Taking its pseudo-product with a binary linear $[6, 2, 4]$ code yields a lower bound of $2^{2\cdot 2} = 16$ on $M(6, 4, 8, 2)$.

We give a simple but robust construction technique for systematic constant-weight codes due to Böinck and van Tilborg.

Proposition 4.3 (Böinck and van Tilborg [18, Construction 4.1]): We have

$$S(2n, 2d, n) \geq S(n, d) \geq B(n, d).$$

Proof: Let C be a systematic code of size $S(n, d)$. Construct a constant-weight code by the rule

$$D = \{(x, \overline{x}) | \, x \in C\},$$

where the bar denotes complementation. The code D hence has twice the distance of C and is systematic because C is. ∎

Example 4.2: Observe that $B(2^{m-1}, 2^{m-2}) = 2^m$ follows from the Plotkin bound and the Reed Muller code $RM(1, m-1)$ [21, Ch. 13]. Proposition 4.3 therefore yields $S(2^m, 2^{m-1}, 2^{m-1}) \geq 2^m$.

We extend the code construction in Proposition 4.3 by appending each codeword with a codeword from a suitable constant-weight code.

Proposition 4.4: If $2^k \leq A(n, d, w)$ we have

$$s(n + 2k, d + 2, w + k) \geq k.$$

Proof: Let C be a constant-weight code of size $A(n, d, w)$. Let $\phi : \mathbb{F}_2^k \to C$ be an injective map. Let

$$D = \{(x, \overline{x}, \phi(x)) | \, x \in \mathbb{F}_2^k\},$$

where $\overline{x}$ denotes the addition of the all one vector to x. The code D is systematic with information set the first k coordinates and has the required parameters. ∎

The next construction generalizes a construction by Zinoviev [10] (see also [22]) and by Etzion [11, Th. 16] to construct multiply constant-weight codes from q-ary codes.

Proposition 4.5: We have

$$M(m, qw, 2d, w) \geq A_q(mw, d).$$

[3] See Appendix for a discussion on the necessity of this relaxation.

Proof: Consider an $(mw, d)_q$ code of size $A_q(mw, d)$ over the alphabet $\mathcal{X}$. We extend each word of length mw to a word of length qmw by replacing each symbol with a binary word of length q. Specifically, replace each symbol in the codeword with the following characteristic function $\phi : \mathcal{X} \to \{0,1\}^{\mathcal{X}}$,

$$\phi(x)_y = \begin{cases} 1, & \text{if } x = y, \\ 0, & \text{otherwise.} \end{cases}$$

We check that the new binary word of length qmw comprises m parts each of weight w.

It remains to check that the distance of the code is 2d. Observe that for any pair of distinct symbols $x, y \in \mathcal{X}$, the distance between $\phi(x)$ and $\phi(y)$ is two. Hence, since the distance between two q-ary codewords is at least d, the distance between the corresponding binary codewords is at least $2d$. ∎

When q is a prime power and $q \geq mw - 1$, there exists a q-ary Reed Solomon code of length mw and distance d. Hence, $A_q(mw, d) \geq q^{mw-d+1}$ and the following corollary is immediate.

Corollary 4.1: If q is a prime power and $q \geq mw - 1$, then

$$M(m, qw, 2d, w) \geq q^{mw-d+1}.$$

On the other hand, when $w = 1$, we observe that we are able to reverse the construction so as to construct an n-ary codeword of length m from an m by n matrix with constant row weight one. Hence, the following corollary is immediate.

Corollary 4.2: We have

$$M(m, n, 2d, 1) = A_n(m, d).$$

B. Designs Constructions

Here, we consider a construction from designs, in particular, resolvable t-designs.

A t-$(v, k, 1)$ design, or t-design, is a pair $(X, \mathcal{B})$ such that $|X| = v$ and $\mathcal{B}$ is a collection of k-subsets of X, called *blocks*, with the property that every t-subset of X is contained in exactly one block. A t-design $(X, \mathcal{B})$ is *resolvable* if the blocks in $\mathcal{B}$ can be partitioned into *parallel classes*, each of which is a partition of X.

Suppose $(X, \mathcal{B})$ is a resolvable t-$(v, k, 1)$ design with $M = k\binom{v}{t}/v\binom{k}{t}$ parallel classes. Let $m = v/k$ and $n = v$. For each parallel class, we construct a binary m by n matrix, where the support of each row is given by a corresponding block. Hence, we form a binary m by n matrix with constant row weight k. Since every pair of blocks intersect at most in $t - 1$ places, the distance between every pair of binary matrices is at least $2m(k - t + 1)$. Hence, we obtain an MCWC$(m, n, 2m(k - t + 1), k)$ of size M. We summarize the construction in the following proposition.

Proposition 4.6: Suppose there exists a resolvable t-$(v, k, 1)$ design. Then

$$M\left(\frac{v}{k}, n, 2(k - t + 1)\frac{v}{k}, k\right) \geq \frac{k\binom{v}{t}}{v\binom{k}{t}}.$$

Existence results for resolvable 2-$(v, k, 1)$ design are surveyed by Abel *et al.* [23, Table 7.35]. When $t \geq 3$, existence results are given by Laue [24] (see also [25]–[27]).

V. Upper Bounds

Trivially, an MCWC(m, n, d, w) is a CWC(mn, d, mw). Hence, we have our first upper bound.

Proposition 5.1: We have

$$M(m, n, d, w) \leq A(nm, d, mw).$$

Next, we extend the techniques of Johnson [6] to obtain the following recursive bounds on $T(w_1, n_1; w_2, n_2; \ldots; w_m, n_m; d)$. Let $i \in [m]$. From [6, eq. (15), (16), and (19)], we have

$$T(w_1, n_1; \ldots; w_m, n_m; d) \leq \left\lfloor \frac{n_i}{w_i} T(w_1, n_1; \ldots; w_i - 1, n_i - 1; \ldots; w_m, n_m; d) \right\rfloor, \quad (2)$$

$$T(w_1, n_1; \ldots; w_m, n_m; d) \leq \left\lfloor \frac{n_i}{n_i - w_i} T(w_1, n_1; \ldots; w_i, n_i - 1; \ldots; w_m, n_m; d) \right\rfloor, \quad (3)$$

$$T(w_1, n_1; \ldots; w_m, n_m; d) \leq \left\lfloor \frac{u}{w_1^2/n_1 + w_2^2/n_2 + \cdots + w_m^2/n_m - \lambda} \right\rfloor, \quad (4)$$

where $d = 2u$ and $\lambda = w_1 + w_2 + \cdots + w_m - u$. Since $M(m, n, d, w) = T(w, n; w, n; \ldots; w, n; d)$, we apply the recursive bound (2) iteratively for m times to have

$$\begin{aligned} &M(m, n, d, w) \\ &= T(w, n; w, n; \ldots; w, n; d) \\ &\leq \left\lfloor \frac{n}{w} T(w - 1, n - 1; w, n; \ldots; w, n; d) \right\rfloor \\ &\leq \left\lfloor \frac{n^2}{w^2} T(w - 1, n - 1; w - 1, n - 1; \ldots; w, n; d) \right\rfloor \\ &\leq \cdots \\ &\leq \left\lfloor \frac{n^m}{w^m} T(w - 1, n - 1; w - 1, n - 1; \ldots; w - 1, n - 1; d) \right\rfloor \\ &= \left\lfloor \frac{n^m}{w^m} M(m, n - 1, d, w - 1) \right\rfloor. \end{aligned}$$

Similarly, we obtain the following recursive upper bounds from (3) and (4).

Proposition 5.2: We have

$$M(m, n, d, w) \leq \left\lfloor \frac{n^m}{w^m} M(m, n - 1, d, w - 1) \right\rfloor, \quad (5)$$

$$M(m, n, d, w) \leq \left\lfloor \frac{n^m}{(n - w)^m} M(m, n - 1, d, w) \right\rfloor, \quad (6)$$

$$M(m, n, d, w) \leq \left\lfloor \frac{d/2}{mw^2/n - (mw - d/2)} \right\rfloor. \quad (7)$$

Suppose $s = mw - d/2 + 1 \leq m$. Applying (2) for s iterations, we have $M(m, n, d, w) \leq \frac{n^s}{w^s} T(w - 1, n - 1; \ldots; w - 1, n - 1; w, n; \ldots; w, n; d)$ and $T(w - 1, n - 1; \ldots; w - 1, n - 1; w, n; \ldots, w, n; d)$ is trivially one. Hence, we obtain the next upper bound.

Proposition 5.3: If $mw - d/2 + 1 \leq m$, then

$$M(m, n, d, w) \leq \left(\frac{n}{w}\right)^{mw - d/2 + 1}. \quad (8)$$

We remark that when $w = 1$, Proposition 5.3 reduces to the classical Singleton bound.

Given m, d, w, let i be the smallest integer such that $m(w-i)-d/2+1 \leq m$. Then i iterative applications of (5), followed by an application of (8), yields the following corollary.

Corollary 5.1: Given m, d, w, let i be the smallest integer such that $m(w-i)-d/2+1 \leq m$ and $t = m(w-i)-d/2+1$. Then we have

$$\begin{aligned} &M(m,n,d,w) \\ &\leq \left\lfloor \frac{n^m}{w^m} \left\lfloor \frac{(n-1)^m}{(w-1)^m} \cdots \left\lfloor \frac{(n-i+1)^m}{(w-i+1)^m} \left\lfloor \frac{(n-i)^t}{(w-i)^t} \right\rfloor \right\rfloor \cdots \right\rfloor \right\rfloor \\ &\leq \frac{n^{mw-d/2+1}}{(w-i)^{mw-d/2+1}}. \end{aligned}$$

When the m, d and w are fixed, we establish tightness of the bound given by Corollary 5.1.

Corollary 5.2: Fix m, d and w. Let $s = mw-d/2+1$ and i be the smallest integer such that $m(w-i)-d/2+1 \leq m$.

Consider $M(m,n,d,w)$ as a function of n. We have

$$1 \leq \limsup_{n\to\infty} \frac{M(m,n,d,w)}{n^s/w^s} \leq \frac{w^s}{(w-i)^s}. \tag{9}$$

In addition, when $s \leq m$, $n/w \geq mw-1$ and n/w is a prime power, we have

$$M(m,n,d,w) = \frac{n^s}{w^s}.$$

Proof: When $n/w \geq mw-1$ and n/w is a prime power, setting $q = n/w$ in Corollary 4.2 establishes that

$$M(m,n,d,w) \geq \left(\frac{n}{w}\right)^{mw-d/2+1} = \frac{n^s}{w^s}.$$

Hence, $\limsup_{n\to\infty} M(m,n,d,w)/(n^s/w^s) \geq 1$. The other inequality of (9) follows from Corollary 5.1. In addition, when $s \leq m$, we have $M(m,n,d,w) \leq (n/w)^s$ from Proposition 5.3 and this yields the value of $M(m,n,d,w)$. ∎

VI. Asymptotics

In this section, we consider the asymptotic rate of $M(m,n,d,w)$ when m is large, n is a function of m, $d = \lfloor \delta nm \rfloor$ and $w = \lfloor \omega n \rfloor$ for $0 < \delta, \omega < 1$. Specifically, we determine the value $\mu(\delta,\omega)$, where

$$\mu(\delta,\omega) := \limsup_{m\to\infty} \frac{\log_2 M(m,n,\lfloor \delta mn \rfloor, \lfloor \omega n \rfloor)}{mn}.$$

In the following discussion, we make use of the following better known exponents.

$$\begin{aligned} \alpha_q(\delta) &:= \limsup_{n\to\infty} \frac{\log_q A(n,\lfloor \delta n \rfloor)}{n}, \\ \alpha(\delta) &:= \limsup_{n\to\infty} \frac{\log_2 A(n,\lfloor \delta n \rfloor)}{n}, \\ \alpha(\delta,\omega) &:= \limsup_{n\to\infty} \frac{\log_2 A(n,\lfloor \delta n \rfloor, \lfloor \omega n \rfloor)}{n}, \\ \sigma(\delta) &:= \limsup_{n\to\infty} \frac{\log_2 S(n,\lfloor \delta n \rfloor)}{n}, \\ \sigma(\delta,\omega) &:= \limsup_{n\to\infty} \frac{\log_2 S(n,\lfloor \delta n \rfloor, \lfloor \omega n \rfloor)}{n}. \end{aligned}$$

First, we reduce the problem of determining $\mu(\delta,\omega)$ to problem of determining $\alpha(\delta,\omega)$.

Lemma 6.1: We have

$$A(nm,d,mw) \leq \frac{\binom{mn}{mw}}{\binom{n}{w}^m} M(m,n,d,w).$$

Lemma 6.1 is analogous to Elias-Bassalygo [21, Th. 33, Ch. 17] by regarding the set of m by n matrices with constant row weight w as a subset of the set of words of length mn with constant-weight mw. As the proof requires some graph theoretical techniques, its proof is deferred to Section VI-B.

As a consequence, we have that the asymptotic exponent of $M(m,n,d,w)$ is equal to the asymptotic exponent of $A(mn,d,mw)$.

Proposition 6.1: We have

$$\mu(\delta,\omega) = \alpha(\delta,\omega).$$

Proof: Observe that

$$\lim_{n\to\infty} \log \frac{\binom{mn}{mw}}{\binom{n}{w}^m} = mnH(\omega) - mnH(\omega) = 0.$$

Then applying limits on n, m and taking logarithms for Lemma 6.1, we have $\alpha(\delta,\omega) \leq \mu(\delta,\omega)$.

On the other hand, asymptotic version of Proposition 5.1 yields $\alpha(\delta,\omega) \geq \mu(\delta,\omega)$ and the proof is complete. ∎

Unfortunately, the value of $\alpha(\delta,\omega)$ is in general not known. Estimates of $\alpha(\delta,\omega)$ are provided by McEliece et al. [28] and Ericson and Zinoviev [29]. In the following subsection, we focus on the case where $\omega = \frac{1}{2}$ and evaluate the asymptotic behavior of the constructions given in Section IV-A.

A. Asymptotics for $\omega = \frac{1}{2}$

The next result follows from the best known upper bound on $\alpha(\delta,\omega)$ due to McEliece *et al.*

Proposition 6.2 (McEliece et al. [28, eq. (2.16)]): We have $\mu(\delta,\omega) \leq g(u^2)$, with $g(x) = H((1-\sqrt{1-x})/2)$, and

$$u = -\delta + \sqrt{\delta^2 - 2\delta + 4\omega(1-\omega)}.$$

In particular,

$$\mu(\delta,1/2) \leq H(1/2 - \sqrt{\delta(1-\delta)}). \tag{10}$$

Our first construction is based on Proposition 4.1, using geometric Goppa codes as outer codes. In particular, fix q to be a prime power and a square, and fix $0 \leq \delta \leq 1 - \frac{1}{\sqrt{q}-1}$. Tsfasman *et al.* [30] exhibited the existence of a family of geometric codes with relative distance δ and rate

$$\alpha_q(\delta) \geq 1 - \delta - \frac{1}{\sqrt{q}-1}.$$

Suppose we pick a CWC$(n,d,n/2)$ of size q as the inner code. For the outer code, we pick a Goppa $(m, \lfloor \delta mn/d \rfloor)_q$ code of rate at least $1 - n\delta/d - 1/(\sqrt{q}-1)$. Applying Proposition 4.1, we obtain an MCWC$(m,n,\lfloor \delta mn \rfloor, n/2)$ of size at least

$$q^{m(1-n\delta/d-1/(\sqrt{q}-1))}$$

Taking logarithm, we have our first lower bound for $\mu(\delta, 1/2)$.

Theorem 6.1: If there exists a CWC$(n, d, n/2)$ of size q, then for $\delta \leq d/n(1 - 1/(\sqrt{q} - 1))$,

$$\mu(\delta, 1/2) \geq \frac{\log q}{d}\left(\frac{d}{n}\left(1 - \frac{1}{\sqrt{q} - 1}\right) - \delta\right).$$

Searching through the online table of lower bounds for $A(n, d, w)$ [16], we pick the following constant-weight codes as inner codes:

(i) a CWC$(12, 4, 6)$ of size 11^2,
(ii) a CWC$(28, 14, 14)$ of size 7^2,
(iii) a CWC$(28, 4, 14)$ of size 1237^2.

Applying Theorem 6.1, we have

$$\mu(\delta, 1/2) \geq \frac{\log 11}{6}\left(\frac{3}{10} - \delta\right), \tag{11}$$

$$\mu(\delta, 1/2) \geq \frac{\log 7}{14}\left(\frac{5}{12} - \delta\right), \tag{12}$$

$$\mu(\delta, 1/2) \geq \frac{\log 1237}{14}\left(\frac{1235}{8652} - \delta\right). \tag{13}$$

Our next construction makes use of the pseudo-product code construction given by Proposition 4.2. The asymptotic version of this proposition is as follows.

Proposition 6.3: We have

$$\mu(\delta, \omega) \geq \sigma(\delta_1, \omega)\sigma(\delta_2),$$

where $0 < \delta_1, \delta_2 < 1$ with $\delta = \delta_1\delta_2$.

Theorem 6.2: We have for $\delta \leq 1/4$,

$$\mu(\delta, 1/2) \geq (1 - H(\sqrt{\delta}))^2/2. \tag{14}$$

Proof: By applying Varshamov-Gilbert (VG) bound [21, Th. 30, Ch. 17] to systematic codes, we get

$$\sigma(\delta_2) \geq 1 - H(\delta_2).$$

Combining VG bound for linear codes with Proposition 4.3 we get

$$\sigma(\delta_1, 1/2) \geq (1 - H(\delta_1))/2.$$

Using Proposition 6.3 with $\delta_1 = \delta_2 = \sqrt{\delta}$, the result follows. ∎

Our final construction follows from setting $q = 2$ in Proposition 4.5.

Theorem 6.3: We have for $\delta \leq 1/2$,

$$\mu(\delta, 1/2) \geq 1 - H(\delta). \tag{15}$$

Proof: Setting $q = 2$ in Proposition 4.5 and applying VG bound, we have

$$M(m, 2w, 2d, w) \geq A(mw, d) \geq 2^{mw(1-H(d/mw))}.$$

Taking logarithms, we obtain (15). ∎

Coincidentally, (15) can be obtained directly by observing that $\mu(\delta, 1/2) = \alpha(\delta, 1/2) = \alpha(\delta)$.

We summarize all the constructions given in this subsection in Figure 3. The top graph compares the lower bounds resulting from Theorem 6.1 with various constant-weight codes as inner codes, while the bottom graph compares the lower bounds resulting from Theorem 6.1, Theorem 6.2 and Theorem 6.3. We observe that the construction given by Proposition 4.5 (or Theorem 6.3) provides the best lower bound.

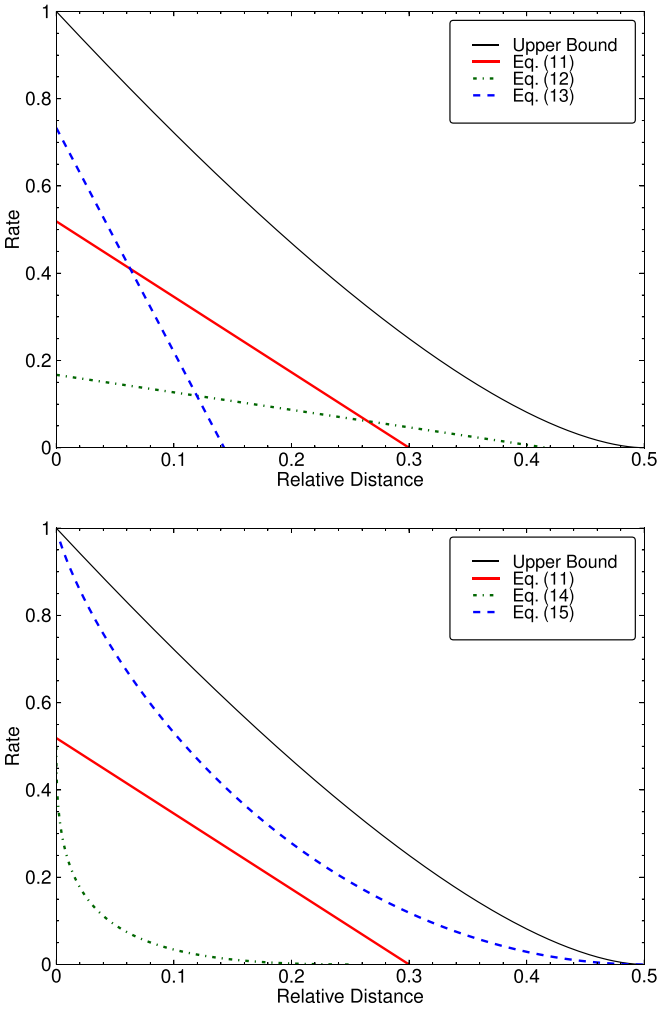

Fig. 3. Upper and lower bounds for $\omega = 1/2$.

B. Proof of Lemma 6.1

El Rouayheb and Georghiades [31] generalized the methods of Elias-Bassalygo using graph theoretical methods. Below we introduce certain concepts necessary for the proof of Lemma 6.1.

Given two graphs $G = (V_G, E_G)$ and $H = (V_H, E_H)$, a mapping $\phi : V_G \to V_H$ is called a *graph homomorphism* if u, v are adjacent in G implies that $\phi(u), \phi(v)$ are adjacent in H. When $G = H$ and ϕ is a bijection, then ϕ is called an *automorphism* of G. Observe that the set of all automorphisms of G is a group under composition; it is called the *automorphism group* of G. A graph is then *vertex transitive* if the action of its automorphism group on its vertex set is transitive.

Given a graph G, a subset X of the vertices is said to be *independent* if every pair of vertices in X is not adjacent in G.

The *independence number* of G, denoted by $\alpha(G)$, the maximum size of an independent set in G. The following theorem gives the relation between the independence numbers of two graphs that are related by a graph homomorphism (see also [32, Ch. 7]).

Theorem 6.4 (El Rouayheb and Georghiades [31, Th. 4]): If H is vertex transitive and there is a graph homomorphism from G to H, then

$$\alpha(H) \leq \frac{V(H)}{V(G)}\alpha(G).$$

Therefore, Lemma 6.1 is a straightforward application of Theorem 6.4. Let G be the graph whose vertices are the m by n arrays with constant row weight w and two vertices are adjacent if the distance between the corresponding arrays is less than d. It is then not difficult to observe that an independent set in G corresponds to a multiply constant-weight code of distance d and hence, $\alpha(G) = M(m, n, d, w)$.

Similarly, let H be the graph whose vertices are codewords of length mn with constant row weight mw and two vertices are adjacent if the distance between the corresponding arrays is less than d. We also have $\alpha(H) = A(mn, d, mw)$.

Finally, observe that G is a subgraph of H and hence, we have a graph homomorphism from G to H. Since H is vertex transitive, we apply Theorem 6.4 to obtain Lemma 6.1.

VII. Conclusion

Motivated by PUFs, we introduced a new class of codes, called multiply constant-weight codes, that generalizes constant-weight codes and doubly constant-weight codes. Using known q-ary codes and constant-weight codes as ingredients, we construct families of multiply constant-weight codes. We also provide analogues of the Johnson bound and show that the bound is asymptotically tight up to a constant factor, assuming certain conditions. We then demonstrate that the asymptotic rates of multiply constant-weight codes and constant-weight codes are the same. An analysis of the asymptotic rates of our code constructions are also given.

Finally, we remark that the tabulating the estimates of $M(m, n, d, w)$ for modest values of the four parameters is a worthwhile project. In addition, the function $S(n, d, w)$ is also worth tabulating and has other applications [18], [19].

Appendix

On Pseudo-Product Construction

We discuss the necessity to relax the requirements for the pseudo-product construction described in Section IV-A. In particular, we demonstrate that the pseudo-production construction cannot guarantee that all columns in any matrix codeword belongs to the code D.

Indeed, consider the following systematic codes given in Example 4.2,

$$C = \{0011, 0101, 1010, 1100\},$$
$$D = \{000000, 101011, 010111, 111100\}.$$

Then one possible codeword from the pseudo-product construction is given by

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix},$$

and we see that the last two columns do not belong to D. This arises mainly from the fact that replacing the columns before the rows result in a matrix codeword different from one obtained by replacing the rows before the columns.

In general, more structure is required on both codes C and D to fulfill the requirements of the usual product construction and this is discussed in detail by Chee *et al.* [33].

Acknowledgement

The authors thank Dr. Son Hoang Dau for pointing out relevant literature and Dr. Punarbasu Purkayastha for helpful discussions. The authors are also grateful to the anonymous reviewer for helpful suggestions.

References

[1] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, "Physical one-way functions," *Science*, vol. 297, no. 5589, pp. 2026–2030, Sep. 2002.
[2] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, "Silicon physical random functions," in *Proc. 9th ACM Conf. Comput. Commun. Secur.*, 2002, pp. 148–160.
[3] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *Proc. 44th ACM/IEEE Ann. Design Autom. Conf.*, Jun. 2007, pp. 9–14.
[4] Z. Cherif, J.-L. Danger, S. Guilley, and L. Bossuet, "An easy-to-design PUF based on a single oscillator: The loop PUF," in *Proc. 15th Euromicro Conf. Digit. Syst. Design*, Izmir, Turkey, Sep. 2012, pp. 156–162.
[5] Z. Cherif, J.-L. Danger, S. Guilley, J.-L. Kim, and P. Solé, "Multiply constant weight codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Istanbul, Turkey, Jul. 2013, pp. 306–310.
[6] S. Johnson, "Upper bounds for constant weight error correcting codes," *Discrete Math.*, vol. 3, nos. 1–3, pp. 109–124, 1972.
[7] V. I. Levenshteĭn, "Upper-bound estimates for fixed-weight codes," *Problems Inf. Transmiss.*, vol. 7, no. 4, pp. 281–287, 1971.
[8] G. D. Forney, Jr., *Concatenated Codes*. Cambridge, MA, USA: MIT Press, 1966.
[9] I. Dumer, "Concatenated codes and their multilevel generalizations," in *Handbook of Coding Theory*, vol. 2. Amsterdam, The Netherlands: North Holland, 1998, pp. 1911–1988.
[10] V. A. Zinoviev, "Cascade equal-weight codes and maximal packings," *Problems Control Inf. Theory*, vol. 12, no. 1, pp. 3–10, 1983.
[11] T. Etzion, "Optimal doubly constant weight codes," *J. Combinat. Designs*, vol. 16, no. 2, pp. 137–151, 2008.
[12] Y. M. Chee, H. M. Kiah, and P. Purkayastha, "Matrix codes and multitone frequency shift keying for power line communications," in *Proc. IEEE Int. Symp. Inf. Theory*, Istanbul, Turkey, Jul. 2013, pp. 2870–2874.
[13] A. Brouwer, J. B. Shearer, N. I. A. Sloane, and W. D. Smith, "A new table of constant weight codes," *IEEE Trans. Inf. Theory*, vol. 36, no. 6, pp. 1334–1380, Nov. 1990.
[14] E. Agrell, A. Vardy, and K. Zeger, "Upper bounds for constant-weight codes," *IEEE Trans. Inf. Theory*, vol. 46, no. 7, pp. 2373–2395, Nov. 2000.
[15] D. H. Smith, L. A. Hughes, and S. Perkins, "A new table of constant weight codes of length greater than 28," *Electron. J. Combinat.*, vol. 13, no. 1, p. 18, May 2006, Art. ID A2.
[16] A. E. Brouwer. (Nov. 1, 2013). *Bounds for Binary Constant Weight Codes*. [Online]. Available: http://www.tue.nl/~aeb/codes/Andw.html

[17] E. Agrell. *Erik Agrell's Tables of Binary Block Codes*. [Online]. Available: http://webfiles.portal.chalmers.se/s2/research/kit/bounds/
[18] F. J. H. Böinck and H. C. A. Van Tilborg, "Constructions and bounds for systematic tEC/AUED codes," *IEEE Trans. Inf. Theory*, vol. 36, no. 6, pp. 1381–1390, Nov. 1990.
[19] M.-C. Lin, "Constant weight codes for correcting symmetric errors and detecting unidirectional errors," *IEEE Trans. Comput.*, vol. 42, no. 11, pp. 1294–1302, Nov. 1993.
[20] O. Amrani, "Nonlinear codes: The product construction," *IEEE Trans. Commun.*, vol. 55, no. 10, pp. 1845–1851, Oct. 2007.
[21] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam, The Netherlands: North Holland, 1977.
[22] W. Kautz and R. Singleton, "Nonrandom binary superimposed codes," *IEEE Trans. Inf. Theory*, vol. 10, no. 4, pp. 363–377, Oct. 1964.
[23] R. J. R. Abel, G. Ge, and J. Yin, "Resolvable and near-resolvable designs," in *The CRC Handbook of Combinatorial Designs*, C. J. Colbourn and J. H. Dinitz, Eds. Boca Raton, FL, USA: CRC Press, 2007, pp. 124–132.
[24] R. Laue, "Resolvable t-designs," *Designs, Codes, Cryptogr.*, vol. 32, nos. 1–3, pp. 277–301, 2004.
[25] E. S. Kramer, S. S. Magliveras, and D. M. Mesner, "Some resolutions of $S(5, 8, 24)$," *J. Combinat. Theory, A*, vol. 29, no. 2, pp. 166–173, 1980.
[26] T. van Trung, "Construction of 3-designs using parallelism," *J. Geometry*, vol. 67, nos. 1–2, pp. 223–235, 2000.
[27] T. van Trung, "Recursive constructions for 3-designs and resolvable 3-designs," *J. Statist. Planning Inference*, vol. 95, nos. 1–2, pp. 341–358, 2001.
[28] R. J. McEliece, E. R. Rodemich, H. Rumsey, and L. Welch, "New upper bounds on the rate of a code via the Delsarte–MacWilliams inequalities," *IEEE Trans. Inf. Theory*, vol. 23, no. 2, pp. 157–166, Mar. 1977.
[29] T. Ericson and V. Zinoviev, "An improvement of the Gilbert bound for constant weight codes (Corresp.)," *IEEE Trans. Inf. Theory*, vol. 33, no. 5, pp. 721–723, Sep. 1987.
[30] M. A. Tsfasman, S. G. Vlădutx, and T. Zink, "Modular curves, Shimura curves, and Goppa codes, better than Varshamov–Gilbert bound," *Math. Nachrichten*, vol. 109, no. 1, pp. 21–28, 1982.
[31] S. El Rouayheb and C. N. Georghiades, "Graph theoretic methods in coding theory," in *Classical, Semi-Classical and Quantum Noise*, L. Cohen, H. V. Poor, and M. O. Scully, Eds., 1st ed. New York, NY, USA: Springer-Verlag, 2012, pp. 53–62.
[32] C. Godsil and G. F. Royle, *Algebraic Graph Theory*. New York, NY, USA: Springer-Verlag, 2001.
[33] Y. M. Chee, H. M. Kiah, P. Purkayastha, and P. Solé, "Product construction of affine codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Honolulu, HI, USA, Jun./Jul. 2014, pp. 1441–1445.

Yeow Meng Chee (SM'08) received the B.Math. degree in computer science and combinatorics and optimization and the M.Math. and Ph.D. degrees in computer science from the University of Waterloo, Waterloo, ON, Canada, in 1988, 1989, and 1996, respectively.

Currently, he is a Professor at the Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore. Prior to this, he was Program Director of Interactive Digital Media R&D in the Media Development Authority of Singapore, Postdoctoral Fellow at the University of Waterloo and IBM's Zürich Research Laboratory, General Manager of the Singapore Computer Emergency Response Team, and Deputy Director of Strategic Programs at the Infocomm Development Authority, Singapore.

His research interest lies in the interplay between combinatorics and computer science/engineering, particularly combinatorial design theory, coding theory, extremal set systems, and electronic design automation.

Zouha Cherif is a research associate at Université Paris XI. She received an M.S. degree in telecommunication from Ecole Superieur des Communications de Tunis, Tunisia in 2010. She carried out her Ph.D research on the modeling and characterization of physically unclonable functions at Telecom ParisTech and Telecom Saint-Etienne, and received a Ph.D degree in 2014. Her research interests are in hardware security of embedded systems.

Jean-Luc Danger (M'–) is full professor at TELECOM-ParisTech where he is the head of the digital system group. His research interests are trusted computing, randomness generation, secure prototyping in FPGA and ASIC. Jean-Luc has co-authored more than one hundred research papers, and more than fifteen patents. He is member of the IACR and senior member of the CryptArchi club. In 2010, he co-founded the Secure-IC S.A.S. company as a spin-off of TELECOM-ParisTech.

Sylvain Guilley (M'–) is full professor at TELECOM-ParisTech. His team works on provable security of electronic circuits and embedded systems. His own research interests are trusted computing, cyber-physical security, secure prototyping in FPGA and ASIC, and formal/mathematical methods. Sylvain has co-authored more than one hundred research papers, and more than ten patents. He is member of the IACR and senior member of the CryptArchi club. He is alumni from Ecole Polytechnique and TELECOM-ParisTech. In 2010, he has co-founded with Jean-Luc Danger, Laurent Sauvage, Hassan Triqui and Philippe Nguyen the Secure-IC S.A.S. company as a spin-off of TELECOM-ParisTech. Since 2012, he organizes the PROOFS workshop, which brings together researchers whose objective is to increase the trust in the security of embedded systems.

Han Mao Kiah received the B.Sc. (Hon) and Ph.D. degrees in mathematics from the National University of Singapore and Nanyang Technological University, Singapore in 2006 and 2014, respectively. Currently, he is a Postdoctoral Research Associate at the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, IL, USA. His research interest lies in the application of combinatorics to engineering problems in information theory. In particular, his interests include combinatorial design theory, coding theory, and enumerative combinatorics.

Jon-Lark Kim (S'01–A'03) received the B.S.degree in mathematics from POSTECH, Pohang, Korea, in 1993, the M.S. degree in mathematics from Seoul National University, Seoul, Korea, in 1997, and the Ph.D. degree in mathematics from the University of Illinois at Chicago, in 2002. From 2002 to 2005, he was with the Department of Mathematics at the University of Nebraska-Lincoln as a Research Assistant Professor. From 2005 to 2012 he was with the Department of Mathematics at the University of Louisville, KY as an Assistant Professor and an Associate Professor. Currently, he is an Associate Professor at the Department of Mathematics, Sogang University, Korea from 2012. He was awarded a 2004 Kirkman Medal of the Institute of Combinatorics and its Applications. He is a member of the Editorial Board of *Designs, Codes, and Cryptography* (2011-current), *International Journal of Information and Coding Theory* (2009-2013), and *Journal of Algebra, Combinatorics, Discrete Structures and Applications* (2014-current). His areas of interest include Coding Theory and its interaction with Algebra, Combinatorics, Number Theory, Cryptography, and Industrial Mathematics.

Patrick Solé received the Ingenieur and the Docteur Ingénieur degrees from Ecole Nationale Supérieure des Télécommunications, Paris, France in 1984 and 1987, respectively, and the Habilitation á Diriger des Recherches degree from Université de Nice, Sophia-Antipolis, France, in 1993.

He has held visiting positions at Syracuse University, Syracuse, NY, during 1987–1989, Macquarie University, Sydney, Australia, during 1994–1996, and at Université des Sciences et Techniques de Lille, Lille, France, during 1999-2000. He has been a permanent member of Centre National de la Recherche Scientifique since 1989, and with the rank of Directeur de Recherche since1996. He has been with the CNRS lab of Telecom ParisTech, the LTCI since 2009. Since 2011 he has a joint affiliation with the Math Dept of King Abdelaziz University, Jeddah, Saudi Arabia. His research interests include coding theory, and cryptography. Dr Solé is the co-recipient of the best paper award for Information Theory in 1995 and served as an associate editor from 1999 till 2003.

Xiande Zhang received the Ph.D. degree in mathematics from Zhejiang University, Hangzhou, Zhejiang, P. R. China in 2009. After that, she held postdoctoral positions in Nanyang Technological University and Monash University. Currently, she is a Research Fellow at the Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore. Her research interests include combinatorial design theory, coding theory, cryptography, and their interactions.