Statistics and Applications {ISSN 2454-7395 (online)}
Special Issue in Memory of Prof. C R Rao
Volume 22, No. 3, 2024 (New Series), pp 227–242
http://www.ssca.org.in/journal

Some Combinatorial Structures and Their Applications in Cryptography

Mausumi Bose
Indian Statistical Institute and St. Xavier's College, Kolkata

Received: 20 June 2024; Revised: 15 August 2024; Accepted: 17 August 2024

Abstract

The science of cryptography makes use of knowledge from several areas of mathematics including number theory, algebraic and combinatorial structures, probability, linear algebra, information theory and others. In this article we give a brief and selected review of some combinatorial structures and highlight their applications in some cryptograhic schemes. Among these structures are the orthogonal arrays, which were introduced by Prof C. R. Rao more than seventy years ago for applications in statistics. Their use in this new field of cryptography is yet another example of the versatility and power of these arrays.

Key words: Block designs; Hadamard matrix; Orthogonal arrays; Error correcting codes; Key predistribution schemes; Visual cryptography.

AMS Subject Classifications: 05B05, 94A60

1. Introduction and preliminaries

Cryptography is an ancient subject, with its early forms traceable to the Pharaonic period of ancient Egypt. The early forms of cryptography were basically some forms of a 'substitution cipher' in which the sender substituted each letter in the plain text of the message by another letter according to some substitution rule g. This substitution rule was known by the receiver and so he could use $d = g^{-1}$ to get the original plain text back from the cipher text. It is known that different versions of this method of cryptography were used by the Romans in Caesar's time, the Indians in ancient times, and in the world wars as the rotor-cipher machines, *e.g.*, the Enigma machine (*cf.* Kahn, 1996).

Over time, these encryption and decryption methods have grown in sophistication and complexity. Currently, cryptography is a field of fundamental importance for protecting the confidentiality and integrity in communication. Various ideas from mathematics and statistics, for instance, number theory, specially prime numbers and finite fields, combinatorics and designs of experiments, sampling methods, results from probability theory, are used to design a variety of cryptographic schemes.

Corresponding Author: Mausumi Bose
Email: mausumi.bose@gmail.com

In this article, we first describe some selected combinatorial structures which are used by statisticians in the context of designs of experiments and then, we give a flavour of the usefulness of these structures in the context of cryptography. This selection is purely subjective and for the sake of brevity, many other interesting uses of these and other combinatorial structures in cryptographic schemes could not be covered.

Section 2 describes the combinatorial structures with examples. Section 3 introduces some areas in cryptography and uses the examples of Section 2 to illustrate how these combinatorial structures are useful in building the schemes for these areas. Throughout, we avoid mathematical details and give references where the details may be found.

2. Some combinatorial structures

In this section we briefly review some of the combinatorial structures which are used in statistics and give examples which are later used in Section 3. For details on these structures we refer to the books by Raghavarao (1971) and Street and Street (1987). More details on the applications of these structures to cryptographic schemes can be found in the books by Stinson (2004), and Stinson and Paterson (2018), and in the references cited.

We shall write 1_n and I_n to denote the unit vector of order n and the identity matrix of order n, respectively. Let $J_{a\times b} = 1_a 1_b'$ and $O_{a\times b}$ be an $a \times b$ null matrix. A finite field of order s will be denoted by GF(s), where s is a prime or prime power. We write Ω to denote a set of s symbols, labeled by $0, 1, \ldots, s-1$.

2.1. Hadamard matrix

Definition 1: An $n \times n$ matrix H_n, with elements -1 and 1 is called a *Hadamard matrix* if $H_n H_n' = nI_n$.

As seen below, H_1 and H_2 exist. For $n > 2$, it is known that H_n exists only if n is an integral multiple if 4. According to the Hadamard conjecture, the converse is also true. Clearly, any two distinct rows of H_n differ in exactly $n/2$ positions. Also, if we multiply all elements in a row (or column) of H_n by -1, the matrix still remains a Hadamard matrix. So, without loss of generality, we can write all Hadamard matrices in the *standard form*, *i.e.*, with all entries in first row and first column being equal to 1.

There are many methods for constructing these matrices, the simplest one is due to Sylvester (1867) who showed that if H_n is a Hadamard matrix, then $H_2 \otimes H_n$ is also a Hadamard matrix of order $2n$ where $\otimes$ denotes Kronecker product. Hence, with $H_1 = (1)$, and $H_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$, a Hadamard matrix of order 2^k can be constructed for every non-negative integer k. These matrices are in standard form and all rows (columns) except the first row (column) have $+1$ in exactly $n/2$ positions and -1 in the remaining $n/2$ positions.

Example 1: H_8 constructed by Sylvester's method is as follows:

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{pmatrix}$$

In statistics, Hadamard matrices are used in constructing designs for experiments, *e.g.*, optimal weighing designs. These matrices have also found applications in signal processing and telecommunication. (*cf.* Yarlagadda and Hershey (1997) and Serberry, Wysocki and Wysocki(2005)). In section 3.2 we highlight the use of these matrices in constructing error-correcting codes.

2.2. Orthogonal arrays

In a series of landmark papers (1946, 1947, 1949) C R Rao proposed some combinatorial structures with applications to statistics, and gave their constructions. Since then, these structures have been widely studied and the entire class of these structures has been called Orthogonal arrays (OAs).

Definition 2: An $M \times k$ array with entries from a set Ω of s symbols is an *orthogonal array* (OA) with M runs, s symbols, strength $t(0 \leq t \leq k)$ and index λ if every $M \times t$ sub-array of this array contains each t-tuple of elements from Ω exactly λ times as a row. Clearly, $M = \lambda s^t$. Such an array will be denoted by $OA(\lambda s^t, k, s, t)$.

From Definition 2.2 it follows that an OA of strength t and index λ is also an OA of strength t' $(0 \leq t' < t)$ and index $\lambda s^{t-t'}$. Also, if a Hadamard matrix H_{4n} exists, then writing it in the standard form and then deleting the first column, one can easily obtain an $OA(4n, 4n-1, 2, 2)$.

Rao (1946, 1947) gave a method for obtaining the OA$(s^n, \frac{s^n-1}{s-1}, s, 2)$ whenever $n \geq 2$, over GF(s). For this, we write all n-tuples from GF(s) as rows to get an $s^n \times n$ array with columns $C_1, C_2, \ldots, C_n$. Then the columns of the OA are of the form $\sum_{i=1}^{n} z_i C_i$ where $z_i \, \epsilon \, GF(s)$, z_i not all zero, and the first non-zero z_i is unity. This method of construction gives what are known as linear OAs and is illustrated in Example 2.

Example 2: The following is an OA(8,7,2,2), where the first 3 columns are written first and then the next 4 columns follow as described above.

0	0	0	0	0	0	0
0	0	1	1	1	0	1
0	1	0	0	1	1	1
0	1	1	1	0	1	0
1	0	0	1	1	1	0
1	0	1	0	0	1	1
1	1	0	1	0	0	1
1	1	1	0	1	0	0

Many other methods of construction of orthogonal arrays are available in the literature. Rao (1947) gave bounds for N for OA$(N = \lambda s^t, k, s, t)$ and Bush (1952) gave improved bounds for arrays of index unity. Arrays of index unity are of special interest and as shown in Bush (1952), if $s(\geq 2)$ is a prime power then an OA$(s^t, s+1, s, t)$ of index unity exists whenever $s > t$. Furthermore, an OA$(s^3, s+2, s, 3)$ exists if s is a power of 2 and Example 3 gives such an OA.

Example 3: OA(8,4,2,3) of index unity.

$$\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{array}$$

For a comprehensive exposition on OAs we refer to Hedayat, Stufken and Sloane (1999). In statistics, OAs are used in constructing designs, specially in the context of fractional factorial designs, (*cf.* Mukerjee and Wu (2006)). In Section 3 we describe the use of OAs in constructing codes and threshold schemes.

2.3. Binary block designs

Definition 3: A *block design* is an arrangement of v symbols in b blocks or sets of sizes $k_1, \ldots, k_b$, the i^{th} symbol occurring r_i times in the design, $1 \leq i \leq v$. The *incidence matrix* N of the design is a $v \times b$ matrix, such that its $(i,j)^{th}$ element equals the number of times the i^{th} symbol occurs in the j^{th} block, $1 \leq i \leq v, 1 \leq j \leq b$. For $1 \leq i_1 < i_2 \leq v$, let $\lambda_{i_1 i_2}$ denote the number of blocks containing symbols i_1 and i_2.

The *design is binary* if the symbols in each block are distinct, *i.e.*, N is binary. A binary design with $r_1 = \ldots = r_v = r$ and $k_1 = \ldots = k_b = k$ shall be written as a binary (v, b, r, k) block design.

2.4. Balanced incomplete block designs (BIBDs)

Definition 4: A binary (v, b, r, k) block design with $k < v$ and $\lambda_{i_1 i_2}$ all equal (=λ, say) is called a *Balanced Incomplete Block Design* (BIBD).

We write such a design as BIBD(v, b, r, k, λ). It follows from Definition 2.3 that $b \geq v, vr = bk$ and $r(k-1) = \lambda(v-1)$. A BIBD with $v = b, r = k$ is called a *symmetric BIBD*. BIBDs with $k = 3$ (or *Steiner's triple systems*) have been specially studied and one is shown in Example 4.

Example 4: A BIBD$(v = b = 7, r = k = 3, \lambda = 1)$, with blocks as columns.

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

Definition 5: A BIBD (v, b, r, k, λ) is said to be *resolvable* if its blocks can be partitioned into r sets, each set containing b/r blocks, such that every set contains each treatment exactly once.

Example 5: A resolvable BIBD($v = 9, b = 12, r = 4, k = 3, \lambda = 1$) with blocks grouped into 4 sets of 3 blocks is shown below, blocks within each set written as rows:

1	2	3	1	4	7	1	6	8	1	5	9
4	5	6	2	5	8	2	4	9	2	6	7
7	8	9	3	6	9	3	5	7	3	4	8

Several methods of construction of BIBDs are known, e.g., if a Hadamard matrix H_n exists and is in standard form, then deleting its first row and column and replacing -1 by 0, we get the incidence matrix of a symmetric BIBD with parameters $(v = b = 4t - 1, r = k = 2t - 1, \lambda = t - 1)$. Again, if we delete a block from this BIBD and delete all symbols that occur in this deleted block, we get the residual design as a BIBD$(2t, 4t - 2, 2t - 1, t, t - 1)$.

Example 6: Starting from H_{12}, we can construct a BIBD$(11, 11, 5, 5, 2)$. Then the residual design obtained from this is a BIBD$(6, 10, 5, 3, 2)$.

In statistics, BIBDs are well studied and known to be A-,D-,E-optimal for a full set of orthonormal treatment contrasts in the class of all block designs with v treatments in b blocks of size k each. In Section 3 we illustrate their use in obtaining anonymous threshold schemes and visual cryptographic schemes.

2.5. Pairwise balanced designs (PBDs)

Definition 6: A block design with r_i all equal (= r, say) and $\lambda_{i_1 i_2}$ all equal (= λ, say), is called a *Pairwise Balanced Design*. It will be written as PBD$(v, \{k_1, \ldots, k_p\}, \lambda)$ where $k_1, \ldots, k_p$ are the possible block sizes.

Example 7: A PBD(5, {3, 2}, 2) with blocks shown as columns

1	2	1	2	1	2	3	4	1	1
2	3	3	4	3	3	5	5	2	4
5	4	4	5	5					

In Section 3 we illustrate their use in obtaining visual cryptographic schemes.

2.6. Partially balanced incomplete block designs (PBIBDs)

Definition 7: A binary (v, b, r, k) block design with $k < v$ and $\lambda_{i_1 i_2}$ taking only 2 values, (= λ_1, or λ_2, say) for all $1 \leq i_1 < i_2 \leq v$. will be called a *Partially Balanced Incomplete Block Design* (PBIBD) with two associate classes.

Such a design will be written as PBIBD $(v, b, r, k, \lambda_1, \lambda_2)$. There are various association schemes underlying PBIBDs, these schemes determining which pair of symbols occur together λ_1 times and which occur λ_2 times. For simplicity, we do not elaborate on association schemes and refer to Raghavarao (1971), pp 121-127, for details.

Example 8: A PBIBD $(6, 4, 2, 3, 0, 1)$ with blocks as columns

$$\begin{array}{cccc} 1 & 1 & 2 & 3 \\ 2 & 4 & 4 & 5 \\ 3 & 5 & 6 & 6 \end{array}$$

In Section 3 we illustrate their use in obtaining visual cryptographic schemes and also mention their use in key predistibution networks.

3. Applications in cryptography

3.1. Codes and error-correcting codes

Definition 8: Let Ω be a set of elements or symbols. A set of k-tuples of the symbols in Ω, where $k \geq 1$ is an integer, is called a code C over the alphabet Ω. Each k-tuple in C is called a codeword. If $\Omega = \mathrm{GF}(2)$, then C is a binary code.

The *Hamming weight* of a codeword is the number of ones in it. The *Hamming distance* between any two codewords is the number of positions in which they differ. The *distance* of a code C, denoted by d, is the minimal Hamming distance between any two distinct codewords in C. A code with N codewords, each of length k over an alphabet Ω consisting of s elements, and having distance d may be written as a (N, k, d, s) code.

An error-correcting code can correct errors incurred during the transmission of data over noisy channels. A linear block code takes a sequence of m symbols from Ω and encodes it as a sequence of $k(> m)$ symbols. The redundant elements are added to the original message to facilitate recovery of the message. The ability of a code to detect and correct errors is measured by its distance d; a code with distance d can correct up to a maximum of e errors where $e = \lfloor \frac{(d-1)}{2} \rfloor$ and can detect up to $d - 1$ errors.

3.1.1. Hadamard codes

Error-correcting codes obtained from Hadamard matrices have the maximal error correcting ability for a given length of codeword and so these are useful when a message is transmitted over a noisy or unreliable channel. For instance, as described in Serberry *et al.* (2005), these codes were used in the 1960's in the Mariner and Voyager space probes to encode information transmitted back to the earth and due to the powerful error-correction capabilities of these codes, it was possible to decode properly the pictures of Jupiter, Saturn, Uranus, Neptune and their moons.

Hadamard matrices obtained from Sylvetser's method of construction are usually used for obtaining Hadamard codes as they lead to linear codes, but Hadamard matrices constructed by other methods lead to codes too, though not necessarily linear. These latter codes were first studied by Bose and Skrikhande (1959) in connection with symmetrical block code designs.

Let $n = 2^k$. A Hadamard code C is obtained from a Hadamard matrix H_{2^k} by replacing -1 by 1 and 1 by 0. The rows of C are the 2^k codewords, each of length 2^k. As discussed in Section 2.1, all rows of H_n, other than the first row, have $+1$ in exactly $n/2$

positions and any two distinct rows of H_n differ in exactly $n/2$ positions. So, with $n = 2^k$, each non-zero codeword in C has Hamming weight 2^{k-1} and any two codewords in C have Hamming distance equal to 2^{k-1}.

Example 9: Let a message x be represented as a binary vector of length 3. So $m = 3$ and we use a code based on H_8 shown in Example 1, by replacing -1 by 1 and 1 by 0. Table 1 shows the original messages and the corresponding encoded messages given by codewords of length 8 obtained from rows of H_8.

Incidentally, it may be mentioned here that the rows on the right side of Table 1 can also be obtained as a linear transform of the array on the left, over GF(2). Then, on deleting the first column of zeros, we get an orthogonal array OA (8,7,2,2) which is isomorphic to the one shown in Example 2. So an OA (8,7,2,2), with a column of zeros added to it, can also give the same code as in Table 1, but the construction of the code from H_8 is simpler.

Table 1: Original and Encoded messages

Original message	Encoded message
0 0 0	0 0 0 0 0 0 0 0
1 0 0	0 1 0 1 0 1 0 1
0 1 0	0 0 1 1 0 0 1 1
1 1 0	0 1 1 0 0 1 1 0
0 0 1	0 0 0 0 1 1 1 1
1 0 1	0 1 0 1 1 0 1 0
0 1 1	0 0 1 1 1 1 0 0
1 1 1	0 1 1 0 1 0 0 1

For a message x, the encoded message is transmitted and the received message is a vector of length 8, say y, with a possible error, *i.e.*, flipping of 0 and 1. To decode y, we find the Hamming distance between y and the 8 codewords, the message corresponding to the codeword with the least Hamming distance will be the original message. For example, if y = 0 1 0 0 0 1 1 0, then the Hamming distance between y and the 8 codewords in their order of Table 1 are 3,3,5,1,3,3,5,5. The least value 1 corresponds to the codeword 0 1 1 0 0 1 1 0. So the original message is decoded correctly as 1 1 0. Thus, one error can be corrected.

A Hadamard code has a large block length (= 2^k) compared to the message length k. However, it can correct $2^{k-2} - 1$ errors in a 2^k-bit encoded message, which is extremely good.

Moreover, we can improve upon code C by writing the code as $C = \begin{pmatrix} H_{2^k} \\ -H_{2^k} \end{pmatrix}$ and then replacing -1 by 1 and 1 by 0 as before. So by this method, the code in Example 9 can be improved upon. It is easy to see from Definition 2.1 that such a code C can accommodate $k + 1$ messages while still having block length 2^k and distance 2^{k-1}. This code C is also sometimes called a *Hadamard code* and it is the same as the *first order Reed-Muller code over the binary alphabet.*

3.1.2. Codes from orthogonal arrays

For a (N, k, d, s) code, the Singleton Bound is $N \leq s^{k-d+1}$. Codes for which N attains this bound are called the *maximum distance separable (MDS) codes* as they have the maximum possible distance between codewords. It can be shown that orthogonal arrays with index unity are equivalent to MDS codes, *i.e.*, an (s^{k-d+1}, k, d, s) code is equivalent to an OA(s^t, k, s, t), where $t = k-d+1$. So MDS codes can be obtained from orthogonal arrays of strength unity which were discussed in Section 2.2.

Example 10: When $t \geq 2$ is an integer and s is a prime power, an OA(s^t, s, s, t) exists and this gives an MDS $(s^t, s, s-t+1, s)$ code which is the well known *Reed Solomon code*, being optimal with respect to the Singleton bound.

There is another bound on N called the Sphere-packing Bound and a code for which this bound is satisfied with equality is called a *perfect* code. It can be shown that for s a prime power and an integer $n \geq 2$, if we start from the linear OAs$(s^n, \frac{s^{n-1}}{s-1}, s, 2)$ constructed as in Example 2 and then take the code which is the orthogonal complement of this OA, we will get a perfect $(s^m, \frac{s^{n-1}}{s-1}, 3, s)$ code, where $m = \frac{s^n-1}{s-1} - n$. These codes are known as *Hamming codes*. When $s = 2$, the code is $(2^{2^n-n-1}, 2^n - 1, 3, 2)$.

Example 11: Starting from an OA(8,7,2,2), shown in Example 2, the orthocomplement of this array gives a *binary Hamming code* or a (16,7,3,2) perfect code.

Interestingly, from a binary Hamming code C if we form a matrix A with its columns being the codewords of C with weight 3, then it can be seen that A gives the *incidence matrix of a symmetric BIBD* with 2^{n-1} treatments and blocks of size 3, *i.e.* a *Steiner's triple system*. So the (16,7,3,2) code constructed as above will give the incidence matrix of the BIBD (7, 7, 3, 3,1) as shown in Example 4.

3.2. Threshold schemes

Let a and b be two integers, $2 \leq a \leq b$. Suppose there is a secret K and a set of b participants $\mathcal{P}$. A dealer who does not belong to $\mathcal{P}$, assigns each participant a 'share', *i.e.*, some partial information about K.

The method of assigning these shares is called a (a, b) *threshold scheme* if any a participants can compute the secret K by pooling their shares, and no set of $a-1$ participants can recover K from their shares. The secret K can be chosen from a set of secrets $\mathcal{K}$ and each share is chosen from a specified share set $\mathcal{S}$.

These schemes have many uses, e.g., there may be a set of 5 individuals, each of whom hold a key to a safe but the safe can be opened only if 3 or more persons use their keys together, it cannot be opened by any single person or 2 persons. This is a (3,5) threshold scheme.

An (a, b) threshold scheme is called an *anonymous* (a, b) *threshold scheme* if the participants receive distinct shares and the recovery of the secret can be done by a participants without knowing which participant holds which share.

3.2.1. Threshold (a, b) schemes from orthogonal arrays

A (t, k) threshold scheme can be obtained from an OA$(s^t, k+1, s, t)$. For assigning the shares, the rows of the OA are first rearranged so that they can be grouped in sets of s^{t-1} rows, the last column of every row in the ith group having the symbol i, $0 \leq i \leq s-1$. The scheme will have s secrets each secret corresponding to one group and s shares, each share corresponding to one symbol of the OA. The participants know the OA which is used in the scheme.

If the dealer wants to assign secret i, he chooses one row at random from the ith group and assigns the element in the jth column to the jth participant, $1 \leq j \leq k$. This will be a (t, k) threshold scheme. Since the OA has strength t and index unity, if t participants combine their shares, there will be a unique row of the OA which will match with their t shares in the corresponding t columns, and so the secret will be revealed. This is because, with the knowledge of the OA, participants will be knowing the group that this unique row comes from. There will not be any such unique row if $t-1$ or fewer participants combine their shares, thus making the scheme secure. This scheme is not anonymous since in order to reveal the secret, it must be known which participant held which share.

Example 12: The OA(8,4,2,3) in Example 3 gives a (3,3) threshold scheme where all participants have to get together in order to reveal the secret.

3.2.2. Anonymous (a, b) threshold schemes from resolvable BIBDs

An anonymous $(2, k)$ threshold scheme can be obtained from a resolvable BIBD with $\lambda = 1$. To see this, consider a resolvable BIBD$(v, b, r, k, 1)$. From Definition 2.4, $r = \frac{v-1}{k-1}$ and the b blocks can be divided into r disjoint sets say $L_1, \ldots, L_r$, each set having $\frac{b}{r}$ blocks. The dealer can share r secrets, each secret associated with a set and there will be v shares, each share associated with one symbol. There can be k participants, the resolvable design being known to all participants. Suppose the dealer picks a set L_i as the secret and chooses any random block from this set. He allocates the k symbols in this block to the k participants, each getting one symbol. This will be a $(2, k)$ anonymous threshold scheme as illustrated below.

Example 13: Consider the resolvable BIBD with $\lambda = 1$ in Example 5. It is divided into 4 sets $L_1, \ldots, L_4$, and so, there are 4 secrets. Since $k = 3$ there can be 3 participants. Suppose the secret is L_2 and the dealer chooses the block {2,5,8} and allocates $2, 5$, and 8 to the 3 participants, respectively. Now if participants 2 and 3 get together, their combined share is {5,8} and since the BIBD has $\lambda = 1$, there is a unique block which can have the symbols 5 and 8 together. So they can identify the secret L_2 uniquely, while this identification cannot be done by any single participant. Thus this is a (2,3) anonymous threshold scheme. This scheme is anonymous as it it not necessary to know which participant held which share.

3.3. Visual cryptographic schemes

Visual cryptographic schemes (VCS) are threshold schemes which encode a secret image or text in such a way that the decoding can be done simply by the human eye, without any computations. Naor and Shamir (1994) introduced VCS for black and white

images. In a (k, n) VCS with n participants, a secret image, or text is encrypted into n shares, each share being printed on a transparency sheet. Each participant is given one share, and if k of them stack their shares one on top of another, the secret is discernible visually. If less than k participants stack their shares, the secret is not visible.

During encryption, each pixel of the original image is encrypted into a number of subpixels, say m. This number m is called the *pixel expansion* of the VCS. The clarity with which a reconstructed image is visible is measured by the *relative contrast* of the VCS. The aim is to keep m small and relative contrast high.

For simplicity, we will only elaborate on VCS for black and white images.

Suppose the n participants are labeled as $1, \ldots, n$. Given a Boolean matrix A, let A_i denote its ith row and A_{ij} denote the Boolean 'or' of rows A_i and A_j. Let $w(T)$ be the weight of a Boolean vector T. We assume that the secret image is a collection of black and white pixels, and a black(white) pixel will be represented by 1(0).

Definition 9: A $(2, n)$ VCS, with n participants and pixel expansion m, is defined by two $n \times m$ Boolean basis matrices S^1 and S^0, respectively, for black amd white pixels such that (i) S^1 and S^0 are equal up to a column permutation, *i.e.*, $w(S^1_i) = w(S^0_i)$, $1 \leq i \leq n$, and (ii) $w(S^1_{i_1,i_2}) > w(S^0_{i_1,i_2}), 1 \leq i_1 < i_2 \leq n$.

Let π be a random permutation of $\{1, \ldots, m\}$. While encryption, if a pixel in the secret image is black(white), then π is applied to the columns of $S^1(S^0)$ and row i of the permuted matrix forms the share of the ith participant. Thus each pixel of the image is encrypted and distributed into n shares, each of which consists of m subpixels. The random permutation used in allocating shares together with condition (i) of Definition 3.2 ensures that no single participant can recover the image. Moreover, for any $i_i < i_2$, if shares i_1 and i_2 are stacked together by aligning the subpixels, and the combined share is obtained by taking the Booelan 'or' of these 2 shares, then condition (ii) of Definition 3.3 guarantees that the grey level of a black pixel is darker than that of a white pixel and this makes the recovered image discernible. For any $i_1 < i_2$, the quantity $\xi_{i_1,i_2} = m^{-1}\{w(S^1_{i_1,i_2}) - w(S^0_{i_1,i_2})\}$ is positive in view of (ii), and it is called the relative contrast of the recovery of the image by participants i_1 and i_2.

A VCS is said to be *balanced* if the $\xi_{i_1,i_2}(1 \leq i_1 < i_2 \leq n)$ values are all equal (=ξ, say) and *unbalanced* otherwise. In any *balanced* $(2, n)$ VCS, for given n, the relative contrast ξ is bounded above by $(= \frac{\lfloor n/2 \rfloor \lceil n/2 \rceil}{n(n-1)}) = \xi_0$, say.

3.3.1. Obtaining $(2, n)$ VCS from BIBDs

Blundo, De Santis and Stinson (1999) gave the following three constructions of *balanced* $(2, n)$ VCS, and for all of these the relative contrasts attains the upper bound ξ_0 for given n.

If a BIBD(n, b, r, k, λ) exits, then there exists a balanced $(2, n)$ VCS with $m = b$ and $\xi = \xi_0 = (r - \lambda)/b$ with S^1 as the incidence matrix of the BIBD and $S^0 = [J_{n\times r},\ O_{n\times(b-r)}]$.

Example 14: From the BIBD in Example 4 we have:

$$S^1 = \begin{matrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{matrix} \qquad S^0 = \begin{matrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{matrix}$$

After permuting the columns of S^1 and S^0, when a row of $S^1(S^0)$ is assigned as a subpixel corresponding to a black(white) pixel to a participant, he cannot know what the original pixel was since in both cases he gets a Boolean vector with weight 3. But when any 2 participants combine their shares, 'or' of any 2 rows of $S^1(S^0)$ give a Boolean vector with weight 5(3). So, on recovery, if the original pixel was black, it appears darker to the human eye than if the original pixel was white. So, $\xi = (5-3)/7 = 2/7$.

Suppose n is even. Then a $(2, n)$ VCS exists with optimal ξ and smallest possible pixel expansion ($m = 2n-2$) if there exists a BIBD$(n, 2(n-1), n-1, n/2, n/2-1)$.

Example 15: The BIBD $(6, 10, 5, 3, 2)$ of Example 6 will give a $(2, 6)$ VCS with optimal $\xi = 3/10$ and smallest possible pixel expansion for this value of n as $m = 10$.

3.3.2. Obtaining $(2, n)$ VCS from PBDs

Suppose n is odd. Then there exists a $(2, n)$ VCS with pixel expansion m and optimal ξ if there exists a PBD $(n, \{(n+1)/2, (n-1)/2\}, r - m(n+1)/4n)$ with exactly m blocks, where r is the common replication number of each symbol. As before, S^1 will be the $n \times b$ incidence matrix of the PBD and $S^0 = [J_{n\times m},\ O_{n\times(m-r)}]$.

Example 16: The PBD(5, {3,2},2) of Example 7 is a PBD with parameters as above with $n = 5, m = 10$ and $r = 5$. So this will give a $(2, 5)$ VCS with $m = 10$ and $\xi = 3/10$.

3.3.3. Obtaining $(2, n)$ VCS from PBIBDs

Adhikary and Bose (2004) and Adhikary, Bose, Kumar and Roy (2005) used *Latin squares* and PBIBDs to show that one can get *unbalanced* $(2, n)$ VCS where the relative contrast for some pairs of participants are more than the optimal bound of ξ for the balanced case. Moreover, given v, since PBIBDs require only partial balance, they have fewer blocks than BIBDs, and hence lead to VCS with smaller pixel expansion (m) than those from BIBDs. We illustrate their method with PBIBDs below:

Example 17: The PBIBD $(6, 4, 2, 3, 0, 1)$ in Example 8 gives $(2, 6)$ VCS with S^1 as the incidence matrix of this design and S^0 as shown below. This is an unbalanced $(2, 6)$ VCS and it can be checked that for some pairs of symbols the relative contrasts are $\xi_{1,6} = \xi_{2,5} =$

$\xi_{3,4} = 1/2$ while for other pairs it is $1/4$.

$$S^1 = \begin{matrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{matrix} \qquad S^0 = \begin{matrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{matrix}$$

Given n, we may choose a suitable PBIBD from the tables of Clatworthy (1973) with small b and large λ_1 to get a $(2, n)$ VCS with $n = v$ and pixel expansion $m = b$. If a PBIBD with $v = n$ is not available, we can choose a PBIBD with $v > n$, construct S^1 from its incidence matrix and then delete $v - n$ suitable rows from it to get S^1 for the $(2, n)$ VCS.

3.3.4. Optimal $(2, n)$ VCS through general binary block designs

We now consider the scenario where both the given n and the allowable m are held fixed and then we optimize with respect to the relative contrasts. As seen above, unbalanced VCS play a crucial role in optimizing over the ξ_{i_1,i_2}, but given n and m, there often does not exist any VCS that maximizes each ξ_{i_1,i_2} separately.

Let $\mathcal{V}(n, m)$ be the class of all $(2, n)$ VCS, balanced or unbalanced, with n participants and pixel expansion m. Then, in the spirit of *A-and E-optimality* in statistical design theory, (*cf.* Shah and Sinha (1989)), Bose and Mukerjee (2006) introduced the notion of *optimal* VCS that maximize the average, say $\bar{\xi}$, or the minimum, say ξ_{min}, of the ξ_{i_1,i_2}, $1 \le i_1 < i_2 \le n$, over $\mathcal{V}(n, m)$.

Such optimal VCS were called *Type I optimal* and *Type II optimal*, respectively. A VCS which is both Type I and Type II optimal will be called Type III optimal. Indeed, given n, m, a VCS, say V^* which is Type I optimal, is also admissible in the sense that there cannot exist another VCS, say V, with same n, m such that each ξ_{i_1,i_2} under V is greater than or equal to the corresponding ξ_{i_1,i_2} under V^*, the inequality being strict for some $i_1 < i_2$.

Bose and Mukerjee showed that the following binary block designs lead to Type III optimal $(2, n)$ VCS:

(i) Suppose m is odd. Let $n = m$. If there exists a binary (v, b, r, k) block design with $v = n = m$ such that $r = (m-1)/2$, $k = (m-1)/2$, and no two of the $\lambda_{ij} (1 \le i < j \le n)$ differ by more than unity, then with basis matrices S^1 as the incidence matrix of this design and $S^0 = [J_{n\times(m-1)/2},\ O_{n\times(m+1)/2)}]$ we get an optimal Type III $(2, n)$ VCS.

(ii) Suppose m is even. If there exists a binary block design with $v = n$, $b = m$, $r_1 = \ldots = r_v = m/2$, $k_j = (n - \delta)/2$ $(1 \le j \le m/2)$, $k_j = (n + \delta)/2$ $(m/2 \le j \le m)$, where $\delta = 1$ if n is odd and $= 0$ if n is even, and no two of the $\lambda_{ij} (1 \le i < j \le n)$ differ by more than unity, then with basis matrices S^1 as the incidence matrix of this design and $S^0 = [J_{n\times m/2},\ O_{n\times m/2)}]$ we get an optimal Type III $(2, n)$ VCS.

There are several broad classes of block designs which satisfy the conditions in (i) and (ii) above. These include *BIBDs, PBIBDs, symmetrical unequal block designs* and *regular*

graph designs with appropriately chosen parameters. We give examples based on BIBD and PBIBD below. For more examples and results we refer to Bose and Mukerjee (2006).

Example 18: For $n = m = 4t - 1$, the BIBD $(v = b = 4t - 1, r = k = 2t - 1, \lambda = t - 1)$ discussed before Example 6, gives a Type III optimal VCS in $\mathcal{V}(4t - 1, 4t - 1)$. Again, the residual BIBD $(2t, 4t - 2, 2t - 1, t, t - 1)$ obtained from the earlier BIBD gives a Type III optimal VCS in $\mathcal{V}(2t, 4t - 2)$. Again, if we delete the last rows of the matrices S^1 and S^0 from those used for the optimal VCS from the residual design, we get a Type III optimal VCS in $\mathcal{V}(2t - 1, 4t - 2)$. So the designs in Example 6 gives optimal Type III VCS.

Example 19: For $n = m = 2t$, where $2 \leq t \leq 12$, $t \neq 7$, as shown in Table 1 of Bose and Mukerjee (2006), we can find an initial block T of cardinality t, such that among the ordered differences (mod n) arising out of the elements of T, each of $1, 2, \ldots, n - 1$ occurs either ρ or $\rho+1$ times, where $\rho = \lfloor t(t - 1)/n - 1) \rfloor$. Upon development of T we get a regular graph design which leads to a Type III optimal VCS in $\mathcal{V}(2t, 2t), 2 \leq t \leq 12, t \neq 7$. When $t = 2$ or 3, this design is also a PBIBD. Moreover, if we delete the last rows of S^1 and S^0 of the optimal VCS in $\mathcal{V}(2t, 2t)$ as obtained above, then we get a Type III optimal VCS in $\mathcal{V}(2t - 1, 2t)$.

3.3.5. Optimal (k, n) VCS from block designs

Bose and Mukerjee (2010) studied (k, n) VCS and gave conditions for their existence and also methods for getting optimal (k, n)VCS. For simplicity, we only give two examples with $k = 3$, one obtained from BIBD and another from PBIBD.

For a binary $(v = n, b, r, k)$ block design let λ_{i_1,i_2,i_3} denote the number of blocks containing symbols i_1, i_2, i_3, and $\xi(i_1, i_2, i_3)$ be the relative contrast for the recovery of the image by the 3 participants $i_1, i_2, i_3, 1 \leq i_2 < i_2 < i_3 \leq n$. We call a $(3, n)$ VCS optimal if it maximizes the average of $\xi(i_1, i_2, i_3)$ over $1 \leq i_2 < i_2 < i_3 \leq n$. Given a design with incidence matrix N, we take $S^1 = [J_{n\times(b-2r)}\ N]$ and $S^0 = [O_{n\times(b-2r)}\ \bar{N}]$ where $\bar{N}$ is obtained from N by interchanging its elements 1 and 0.

Example 20: Let $n = 13$. Then the BIBD $(13, 26, 6, 3, 1)$ (given in Takeuchi (1962)) will lead to an optimal (unbalanced) $(3, 13)$ VCS. This will have pixel expansion $m = 2(b - r) = 40$ and $\xi(i_1, i_2, i_3) = 5/40$ if i_1, i_2, i_3 occur together in a block and $3/40$ otherwise. It also maximizes the minimum possible value of $\xi(i_1, i_2, i_3)$ among all $(3, 13)$VCS with $m = 40$.

Example 21: Let $n = 20$. The PBIBD $(20, 16, 4, 5, 0, 1)$ (=design SR58 in Clatworthy (1973) tables) will lead to an optimal (unbalanced) $(3, 20)$ VCS. This will have pixel expansion $m = 2(b - r) = 24$ and the smallest value of $\xi(i_1, i_2, i_3) = 1/24$. It also maximizes the minimum possible value of $\xi(i_1, i_2, i_3)$ among all $(3, 20)$ VCS with $m = 24$.

It may also be noted that the two VCS in Examples 20 and 21 substantially reduce the pixel expansion compared to the corresponding *balanced* $(3, n)$ VCS as in Blundo et. al (2003), which have $m = 440$ and $m = 23256$, respectively.

3.4. Key predistribution schemes for distributed sensor networks using block designs

Key predistribution schemes (KPS) is another area of cryptography where block designs have been effectively used to get good schemes. These schemes are used in various applications, for instance, in a military operation, where sensor nodes with secret keys installed in them, may be distributed in a random manner over a sensitive area and, once deployed, these nodes are required to communicate with each other through secure keys in order to gather and relay information.

The two main metrics for a KPS are the *network size*, *i.e.*, number of nodes (n) and the *key storage* k, *i.e.*, the number of keys stored per node. Any two nodes within a neighbourhood can communicate with each other if they have $q(\geq 1)$ common keys, where q is the *intersection threshold* of the network. If two nodes do not have q keys in common then they can still communicate through multiple secure links if there is a sequence of one or more intermediate nodes connecting them such that every pair of adjacent nodes in this sequence share q common keys.

If some nodes are captured in an attack, all keys in them are lost but the remaining nodes can still communicate (*i.e.*, be resilient) using the remaining keys. For more details on the applications, the security framework and models for these distributed sensor networks (DSNs) we refer *e.g.*, to Roman et al. (2005) and Du *et al.* (2005) and Martin (2009).

Key assignment schemes based on *combinatorial designs* is specially useful since using the combinatorial structures of the underlying designs, one can study the connectivity and resiliency properties of the scheme, and also carry out shared-key discovery and path-key establishment in a structured manner. Camtepe and Yener (2004) used *finite projective planes* and *generalized quadrangles* and Dong et al. (2008) used *3-designs* to construct KPS with $q = 1$. Lee and Stinson (2008) used *transversal designs* to construct KPS and give schemes separately for $q = 1$ and for $q = 2$.

Bose, Dey and Mukerjee (2013) suggested one general construction method for KPS for any given q and by varying the choices of the designs, this resulted in KPS for networks with varying numbers of nodes, key-pool sizes and numbers of keys per node, thus providing more flexibility in choosing a scheme suitable for the requirements of a situation. The designs used were *BIBDs, PBIBDs based on the group-divisible, Latin square and triangular association schemes,* and suitable duals of these designs. This method works for general q and can cover a wide variety of values of n.

The complexity of the KPS scheme and its various metrics leads to involved algebra and so we refrain from elaborating further in this area.

4. Conclusion

In this article, an endeavour has been made to highlight the fact that combinatorial designs have a wide applicability in various areas of cryptography.

We have mainly focused on Hadamard matrices, orthogonal arrays, pairwise balanced designs, balanced incomplete block designs and partially balanced incomplete block designs and their applications in various areas of cryptography. There are many other topics that

could not be covered, *e.g.* a combinatorial structure used often in the context of threshold access structures is the perfect hash family (PHF). Long *et al.* (2006) and Martin and Ng (2007) used generalized cumulative arrays focusing on the situation where all participants have the same probability of being selected for activation. Bose and Mukerjee (2014) gave a method where an unequal probability scheme given by PHFs leads to better levels of group and participant anonymity, and also showed that BIBDs can be used to get schemes in this context too.

The Anti-collusion Digital Fingerprinting Codes is another area of cryptography where combinatorial designs have been used effectively, for instance, Trappe *et al.*(2003) used BIBDs, Kang *et al.*(2006) used PBIBDs, Yagi *et al.*(2009) used finite geometries, Li *et al.*(2009) used OAs, Bose and Mukerjee (2010) used partially cover-free families.

To conclude, our aim in this article is to highlight that there are various areas of cryptography where combinatorial designs and structures give effective and efficient schemes. We hope that this article will generate sufficient interest among statisticians who are already familiar with these structures, to take up research in this area.

Acknowledgements

This work has been supported by the National Board for Higher Mathematics, Dept of Atomic Energy, Govt of India. The author sincerely thanks the two referees for their valuable comments which have led to an improvement in the presentation of the paper.

References

Adhikari, A. and Bose, M. (2004). Construction of new visual threshold schemes using combinatorial designs. *IEICE Transactions*, **E87-A**, 1198-1202.

Adhikari, A., Bose, M., Kumar, D., and Roy, B. (2007). Applications of partially balanced incomplete block designs in developing (2,n) Visual cryptographic Schemes. *IEICE Transactions*, **E90-A**, 949-951.

Blundo, C., De Santis, A., and Stinson, D. R. (1999). On the contrast in visual cryptography schemes. *Journal of Cryptology*, **12**, 261-289.

Bose, M., Dey, A., and Mukerjee, R. (2013). Key predistribution schemes distributed sensor networks via block designs. *Designs, Codes and Cryptography*, **467**, 111-136.

Bose, M. and Mukerjee, R. (2006). Optimal $(2, n)$ visual cryptographic schemes. *Designs, Codes and Cryptography*, **40**, 255-267.

Bose, M. and Mukerjee, R. (2010). Optimal (k, n) visual cryptographic schemes for general k. *Designs, Codes and Cryptography*, **55**, 19-35.

Bose, M. and Mukerjee, R. (2013). Union distinct families of sets, with an application to cryptography. *Ars Combinatoria,* **110** 179-192.

Bose, M. and Mukerjee, R. (2014). An unequal probability scheme for improving anonymity in shared key operations. *Journal of Statistical Theory and Practice*, **8**, 100-112.

Bose, R. C. and Shrikhande, S. S. (1959). A note on result in the theory of code construction. *Information and Control*, **2**, 183–194.

Bush, K. A. (1952). Orthogonal arrays of index unity. *Annals of Mathematical Statistics*, **23**, 416-434.

Clatworthy, W. H. (1973). *Tables of Two-associate Partially Balanced Designs.* National Bureau of Standards, Applied Maths, series no **63**, Washingto D.C.

Climato, S., Prisco, R. D., and Santis, A. D. (2005). Optimal coloured threshold visual cryptography schemes. *Designs Codes and Cryptography*, **35**, 311-335.

Hedayat, A. S., Stufken, J., and Sloane, N. J. A. (1999). *Orthogonal Arrays: Theory and Applications.* Springer-Verlag, New York.

Ibrahim, D. R., Teh, J. S., and Abdullah, R. (2021). An overview of visual cryptography techniques. *Multimedia Tools and Applications*, **80**, 31927-31952.

Kahn, D. (1996). *The Codebreakers: The Comprehensive History of Secret Communication from Ancient Times to the Internet.* Scribner, New York.

Kang, I., Arce, G., and Lee, H. K. (2011). Color Extended Visual Cryptography Using Error Diffusion. *IEEE Transactions on Image Processing*, **20**, 132-145.

Kang, I., Sinha, K., and Lee, H. K. (2006). New digital fingerprint code scheme using group-divisible design. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Science*, **E89-A**, 3732-3735.

Mukerjee, R. and Wu, C. F. J. (2006). *A Modern Theory of Factorial Design.* Springer, New York.

Naor, M. and Shamir, A. (1994). Visual cryptography. Advances in Cryptology, Eurocrypt'94. Lecture Notes in Computer Science, **950**, 1-12, Springer-Verlag.

Plotkin M. (1960). Binary codes with specified minimum distance, *IRE Transactions*, **IT-6** 445–450.

Raghavarao, D. (1971). *Construction and Combinatorial Problems in Design of Experiments.* New York, Wiley.

Rao, C. R. (1946). Hypercubes of strength 'd' leading to confounded designs in factorial experiments. *Bulletin of Calcutta Mathematical Society*, **38**, 67-78.

Rao, C. R. (1947). Factorial experiments derivable from combinatorial arrangements of arrays. *Journal of Royal Statistical Society*, **9**, 128-139.

Rao, C. R. (1949). On a class of arrangements. *Proceedings of Edinburgh Mathematical Society*, **8**, 119-125.

Serberry J, Wysocky B. J., and Wysocki T. A. (2005). On some applications of Hadamard matrices. *Metrika*, **62**, 221-239.

Shah, K. and Sinha, B.K. (1989). *Theory of Optimal Designs.* Springer-Verlag, New York.

Stinson, D. R. (2004). *Combinatorial Designs: Constructions and Analysis,* Springer, ISBN 978-0-387-95487-5.

Stinson, D. R. and Paterson, M. B. (2018). *Cryptography: Theory and Practice*, 4th ed. CRC Press.

Street, A. P. and Street, D. J. (1987). *Combinatorics of Experimental Design.* Oxford. Clarendon Press.

Takeuchi, K. (1962). A table of difference sets generating balanced incomplete block designs. *Review of the International Statistical Institute*, **30**, 361-366.

Trappe, W., Wu, M, Wang, Z. J., and Liu, K. J. R. (2003). Anti-collusion finger-printing for multimedia. *IEEE Transactions on Signal Processing*, **51**, 1069-1087.

Yagi, H., Matsushima, T., and Hirasawa, S. (2007). Improved collusion-secure codes for digital fringerprinting based on finite geometries. *IEEE International Conference on Systems, Man and Cybernetics*, 948-953.

Yarlagadda, R. K. and Hershey, J. E. (1997). *Hadamard Matrix Analysis and Synthesis with Applications to Communications and Signal: Image Processing.* Kluwer.