Discrete Mathematics 340 (2017) 201–208

Contents lists available at ScienceDirect

Discrete Mathematics

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

Invertible binary matrices with maximum number of 2-by-2 invertible submatrices

Yiwei Zhang [a], Tao Zhang [b], Xin Wang [b], Gennian Ge [a,c,*]

[a] School of Mathematical Sciences, Capital Normal University, Beijing 100048, China
[b] School of Mathematical Sciences, Zhejiang University, Hangzhou 310027, Zhejiang, China
[c] Beijing Center for Mathematics and Information Interdisciplinary Sciences, Beijing 100048, China

ARTICLE INFO

Article history:
Received 16 January 2016
Received in revised form 28 July 2016
Accepted 18 August 2016

Keywords:
All-or-nothing transforms
Invertible matrices

ABSTRACT

For given positive integers $t \leq s$, what is the maximum number of t-by-t invertible submatrices in an invertible binary matrix of order s? This purely combinatorial problem is posed recently by D'Arco, Esfahani and Stinson. The motivation is related to all-or-nothing transforms (AONTs) suggested by Rivest as a preprocessing for encrypting data with a block cipher, which has been widely applied in cryptography and security. For the case $t = 2$, let $R_2(s)$ denote the maximal proportion of 2-by-2 invertible submatrices of s-by-s invertible matrices. D'Arco, Esfahani and Stinson ask whether $\lim_{s\to\infty} R_2(s)$ exists or not. If it does exist, then their results indicate that the limit is between 0.494 and 0.625. In this paper we completely solve the problem by showing that $\lim_{s\to\infty} R_2(s) = 0.5$.

1. Introduction

The original motivation of this problem traces back to [14], in which Rivest suggested using all-or-nothing transforms as a preprocessing for encrypting data with a block cipher, in the setting of computational security. Later Stinson changed the setting to unconditional security [16] and the generalized version of all-or-nothing transforms is defined in [6] as follows:

Definition 1. Let X be a finite set known as an alphabet. Let s be a positive integer and consider a map $\phi : X^s \to X^s$. For an input s-tuple, say $x = (x_1, \ldots, x_s)$, ϕ maps it to an output s-tuple, say $y = (y_1, \ldots, y_s)$, where $x_i, y_i \in X$ for $1 \leq i \leq s$. The map ϕ is an unconditionally secure *t-all-or-nothing transform* provided that the following properties are satisfied:
- ϕ is a bijection.
- If any $s - t$ out of the s output values $y_1, \ldots, y_s$ are fixed, then any t of the input values x_i $(1 \leq i \leq s)$ are completely undetermined, in an information-theoretic sense.

We will call such a map ϕ as a (t, s, v)-AONT, where $v = |X|$. And when s and v are clear or not relevant, we just call it a t-AONT.

What Rivest defined in [14] corresponds to the special case $t = 1$. 1-AONT can provide a preprocessing called "package transform" for block ciphers. Suppose we want to encrypt plaintexts $(x_1, \ldots, x_s)$. First we apply a 1-AONT to get $(y_1, \ldots, y_s) = \phi(x_1, \ldots, x_s)$. Note that the transform ϕ is not necessarily private. Then we encrypt $(y_1, \ldots, y_s)$ using a block cipher and get the ciphertexts $z_i = e_K(y_i)$ for $1 \leq i \leq s$, where e_K is the encryption function. The receiver can decrypt the ciphertexts and then use the inverse transform ϕ^{-1} to retrieve the plaintexts. However, any adversary needs to decrypt the whole ciphertexts and get the exact values of $(y_1, \ldots, y_s)$ (by means of exhaustive key search, say) in order to determine any

[*] Corresponding author at: School of Mathematical Sciences, Capital Normal University, Beijing 100048, China.
E-mail address: gnge@zju.edu.cn (G. Ge).

http://dx.doi.org/10.1016/j.disc.2016.08.016
0012-365X/© 2016 Published by Elsevier B.V.

one symbol among the plaintexts. In other words, a partial decryption cannot provide any information about each symbol among the plaintexts due to the property of a 1-AONT. In this sense, the application of 1-AONTs gives a certain amount of additional security over block ciphers. Extensions of this technique are studied in [2,7]. AONTs also have various other applications in cryptography and security. For example, it is applied in network coding [4,8], secure data transfer [17], anti-jamming techniques [12], exposure-resilient functions [3], secure distributed cloud storage [11,15], secure secret sharing schemes [13] and query anonymization for location-based services [18].

However, the properties of 1-AONT do not say anything regarding the partial information that might be revealed about more than one of the s input values. Say, it is possible to derive the sum of two input values with only some relatively small number of output values. That is exactly the motivation for the general definition of a t-AONT. Similarly as above, if a t-AONT is applied before using a block cipher, then the adversary will have no information regarding any boolean function of any t symbols among the plaintexts, unless he could do enough decryption to get more than $s-t$ symbols in $(y_1, \ldots, y_s)$.

Linear AONTs are of particular interests. Let the alphabet set be $\mathbb{F}_q$, the finite field of order q. A (t, s, q)-AONT with alphabet $\mathbb{F}_q$ is *linear* if each y_i is an $\mathbb{F}_q$-linear function of $(x_1, \ldots, x_n)$. Then it can be represented by an invertible s-by-s matrix with entries from $\mathbb{F}_q$ and the second property of t-AONTs requires that every t-by-t submatrices of M must also be invertible. Matrices satisfying this condition do exist when q is a prime power and $q \geq 2s$ [6]. However, when we are working on $\mathbb{F}_2$, it is easy to see that there is no linear $(1, s, 2)$-AONT for $s > 1$ and no linear $(2, s, 2)$-AONT for $s > 2$. So D'Arco et al. proposed the question on how close one can get to a t-AONT. That is, for given integers $t \leq s$, what is the maximum number of t-by-t invertible submatrices in an invertible binary matrix M of order s? In the sequel, invertibility of a matrix refers to the invertibility over $\mathbb{F}_2$. We follow the notations defined in [6]:

$$\begin{aligned}
&N_t(M) = \text{number of invertible } t\text{-by-}t \text{ submatrices of } M,\\
&R_t(M) = \frac{N_t(M)}{\binom{s}{t}^2},\\
&R_t(s) = \max\{R_t(M) : M \text{ is an } s\text{-by-}s \text{ invertible binary matrix}\}.
\end{aligned}$$

In [6] $R_1(s) = 1 - \frac{s-1}{s^2}$ is shown and upper and lower bounds for $R_2(s)$ are analyzed. A problem is posed in [6] about the existence of the limit $\lim_{s\to\infty} R_2(s)$. If it does exist, then its value is between 0.494 and 0.625. In this paper, we give a complete solution to this problem by showing that $\lim_{s\to\infty} R_2(s) = 0.5$. The rest of the paper is organized as follows. In Section 2 we analyze the upper limit and show that $\overline{\lim}_{s\to\infty} R_2(s) \leq 0.5$. In Section 3 we use some probabilistic tools to derive the lower limit and show that $\underline{\lim}_{s\to\infty} R_2(s) \geq 0.5$. In Section 4 we offer a construction via cyclotomy as an illustrative example. We conclude in Section 5.

2. Upper bound via integer programming

In this section we analyze the upper limit $\overline{\lim}_{s\to\infty} R_2(s)$. Let $R_2'(s) = \max\{R_2(M) : M \text{ is an } s\text{-by-}s \text{ binary matrix}\}$. Then clearly $R_2(s) \leq R_2'(s)$ and then $\overline{\lim}_{s\to\infty} R_2(s) \leq \overline{\lim}_{s\to\infty} R_2'(s)$. Therefore, to give an upper bound of $\overline{\lim}_{s\to\infty} R_2(s)$, it suffices to give an upper bound of $\overline{\lim}_{s\to\infty} R_2'(s)$. That is, within this section we shall study $R_2'(s)$ so the constraint that the matrix itself should be invertible can be ignored for the moment. Note that a 2-by-2 binary matrix is invertible if and only if it is one of the following matrices:

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

Firstly we focus on those matrices containing exactly c entries as "1", for a given integer c. We now analyze how these "1" entries should be distributed in order to get the maximal number of 2-by-2 invertible submatrices. For $1 \leq i \leq n$, let x_i be the weight of each row and let y_i be the weight of each column. For $1 \leq i < j \leq n$, let $z_{i,j}$ be the intersection number of the ith and jth row. That is, $z_{i,j} = |\{k : M_{i,k} = M_{j,k} = 1\}|$. Besides the natural restriction $\sum_{i=1}^n x_i = \sum_{i=1}^n y_i = c$, a standard double counting argument also implies a further restriction that $\sum_{1\leq i<j\leq n} z_{i,j} = \sum_{i=i}^n \binom{y_i}{2}$. Then the number of invertible 2-by-2 submatrices provided by the ith and jth row can be easily calculated as

$$z_{i,j}(x_i - z_{i,j}) + z_{i,j}(x_j - z_{i,j}) + (x_i - z_{i,j})(x_j - z_{i,j}) = x_i x_j - z_{i,j}^2.$$

So we are actually facing an integer programming problem as follows.

$$\begin{aligned}
\text{maximize} :\ & \sum_{1\leq i<j\leq s} x_i x_j - z_{i,j}^2\\
\text{subject to} :\ & \sum_{1\leq i\leq s} x_i = \sum_{1\leq i\leq s} y_i = c,\\
& \sum_{1\leq i<j\leq s} z_{i,j} = \sum_{1\leq i\leq s} \binom{y_i}{2},\\
& x_i, y_i, z_{i,j} \in \mathbb{N}, \quad 1 \leq i < j \leq s,\\
& c \in \mathbb{N}, \quad 0 \leq c \leq s^2.
\end{aligned}$$