Asymmetric All-or-nothing Transforms

Navid Nasr Esfahani[1], Douglas R. Stinson[1,*]

[1]David R. Cheriton School of Computer Science, University of Waterloo

Received: 9th June 2021 | Revised: 25th October 2021 | Accepted: 25th October 2021

Abstract *A (symmetric) t-all-or-nothing transform is a bijective mapping defined on the set of s-tuples over a specified finite alphabet. It is required that knowledge of all but t outputs leaves any t inputs completely undetermined. There have been numerous papers developing the theory of AONTs as well as presenting various applications of AONTs in cryptography and information security. In this paper, motivated by an application of Karame et al. [10], we initiate a study of* asymmetric all-or-nothing transforms *(or* asymmetric AONTs*). We replace the parameter t by two parameters t_{out} and t_{in}, where $t_{in} \leq t_{out}$. The requirement is that knowledge of all but t_{out} outputs leaves any t_{in} inputs completely undetermined. When $t_{in} < t_{out}$, we refer to the AONT as* asymmetric. *We give several constructions and bounds for various classes of asymmetric AONTs, especially those with $t_{in} = 1$ or $t_{in} = 2$. We pay particular attention to* linear *transforms, where the alphabet is a finite field $\mathbb{F}_q$ and the mapping is linear.*

Keywords: all-or-nothing transform, orthogonal array

2010 Mathematics Subject Classification: 05B30, 94A60

1 INTRODUCTION

In this paper, motivated by an application of Karame et al. [10], we study *asymmetric all-or-nothing transforms*, which we define informally as follows.

Definition 1.1. *Suppose s is a positive integer and $\phi : \Gamma^s \to \Gamma^s$, where Γ is a finite set of size v (called an* alphabet*). Thus ϕ is a function that maps an input s-tuple $\mathbf{x} = (x_1, \ldots, x_s)$ to output s-tuple $\mathbf{y} = (y_1, \ldots, y_s)$. Suppose t_{in} and t_{out} are integers such that $1 \leq t_{in} \leq t_{out} \leq s$.*

*The function ϕ is an (t_{in}, t_{out}, s, v)-*all-or-nothing transform *(or (t_{in}, t_{out}, s, v)-AONT) provided that the following properties are satisfied:*

1. *ϕ is a bijection.*
2. *If any $s - t_{out}$ of the s outputs $y_1, \ldots, y_s$ are fixed, then the values of any t_{in} inputs x_i (for $1 \leq i \leq s$) are completely undetermined.*

Remark 1.1. It is not difficult to see that $t_{in} \leq t_{out}$ if a (t_{in}, t_{out}, s, v)-AONT exists, as follows. If only t_{out} outputs are unknown, then the number of possible values taken on by any subset of the inputs is at most $v^{t_{out}}$. Since a subset of t_{in} inputs must be completely undetermined, we must have $v^{t_{in}} \leq v^{t_{out}}$, or $t_{in} \leq t_{out}$.

All-or-nothing-transforms (AONTs) were invented in 1997 by Rivest [12]. Rivest's work concerned AONTs that are computationally secure. Some early papers on various generalizations of AONTs include [1, 2, 5]. Stinson [13] introduced and studied all-or-nothing transforms in the setting of unconditional security. Further work focussing on the existence of unconditionally secure AONTs can be found in [4, 7, 8, 9, 14, 15]. AONTs have had numerous applications in cryptography and information security; see [6] for an overview.

Rivest's original definition in [12] corresponded to the special case $t_{in} = t_{out} = 1$. Most research since then has involved AONTs where $t_{in} = t_{out} = t$ for some positive integer t. (Such an AONT is often denoted as a (t, s, v)-AONT in the literature.) In such an AONT, knowing all but t outputs leaves any t inputs undetermined. Here we mainly consider AONTs where $t_{in} < t_{out}$. Such an AONT can be thought of *asymmetric* in the sense that the number of missing outputs is greater than the number of inputs about which we are seeking information. In general, asymmetric AONTs are easier to construct than AONTs in which $t_{in} = t_{out}$ because the requirements are weaker.

The first example of asymmetric AONTs in the literature is apparently found in Stinson [13, §2.1]. We present this construction in Example 1.1.

*Corresponding Author: dstinson@uwaterloo.ca.

D.R. Stinson's research is supported by NSERC discovery grant RGPIN-03882.

Example 1.1. For s even, a $(1, 2, s, 2)$-AONT exists as follows. Given s inputs $x_1, \ldots, x_s \in \mathbb{Z}_2$, define

$$r = \sum_{i=1}^{s} x_i$$
$$y_i = r + x_i, \text{ for } 1 \leq i \leq s.$$

This yields the s outputs $y_1, \ldots, y_s$. The inverse transformation is computed as

$$r' = \sum_{i=1}^{s} y_i$$
$$x_i = r' + y_i, \text{ for } 1 \leq i \leq s.$$

Suppose we are given $s - 2$ of the s outputs, so two outputs are missing. It is clear that each input depends on $s - 1$ outputs: x_i is a function of all the y_j's, except for y_i. Thus, if two outputs are missing, then no values can be ruled out for x_i. ■

We note that the construction given in Example 1.1 only works for even s (when s is odd, the mapping is not invertible). A construction for odd values of s will be given later (see Lemma 3.9).

Karame et al. [10] introduced *Bastion*, which is a scheme for securely dispersing a document over multiple cloud storage services. Bastion involves encrypting a plaintext using counter mode and then applying a $(1, 2, s, 2)$-AONT to the resulting ciphertext blocks. The paper [10] considered a threat model where the adversary may have access to the key or use a backdoor to decrypt the ciphertext. To protect against these threats, assuming the adversary cannot access at least two parts, they suggest to divide the ciphertext into multiple parts and store each part on a different server after applying the AONT.

1.1 OUR CONTRIBUTIONS

Our goal in this paper is to develop the basic mathematical theory of asymmetric AONTs. In Section 2, we discuss a combinatorial approach to asymmetric AONTs, and we examine how different combinatorial definitions impact the security of the transforms. We also present some connections with other combinatorial structures such as orthogonal arrays and split orthogonal arrays. Section 3 focusses on existence and bounds for linear asymmetric AONTs. We complete the solution of the existence problem for $t_{in} = 1$, as well as when $t_{in} = 2$ and $t_{out} = s - 1$. Then we turn to cases where $t_{in} \geq 2$. We prove a general necessary condition for existence, and then we consider the case $t_{in} = 2$ in detail. New existence results are obtained from computer searches. Finally, Section 4 is a brief summary.

We note that many of the results in this paper were first presented in the PhD thesis of the first author [6].

2 COMBINATORIAL DEFINITIONS AND SECURITY PROPERTIES

Definition 1.1 is phrased in terms of security properties, i.e., it specifies information about a subset of inputs that can be deduced if only a certain subset of the outputs is known. (As mentioned in the introduction, we are studying AONTs in the setting of unconditional security.) It is useful to employ a combinatorial description of AONTs in order to analyze them from a mathematical point of view. Combinatorial definitions of AONTs have appeared in numerous papers, beginning in [13]. However, the connections between security definitions and combinatorial definitions turn out to be a bit subtle, as was recently shown by Esfahani and Stinson [9].

First, as noted in [9], there are two possible ways to interpret the security requirement. In the original definition of AONT due to Rivest [12], as well as in Definition 1.1, we only require that the values of any t_{in} inputs are *completely undetermined*, given the values of $s - t_{out}$ outputs. In other words, assuming that every possible input s-tuple occurs with positive probability, the probability that the t_{in} specified inputs take on any specified possible values (given all but t_{out} outputs) is positive. This notion is termed *weak security* in[9].

An alternative notion that is discussed in detail in [9] is that of *perfect security*. Here, we require that the *a posteriori* distribution on any t_{in} inputs, given the values of $s - t_{out}$ outputs, is identical to the *a priori* distribution on the same inputs. Thus, *no information* about any t_{in} inputs is revealed when $s - t_{out}$ outputs are known.

The standard combinatorial definition for (t, t, s, v)-AONT (see, e.g., [4, 7]) involves certain unbiased arrays. We review this definition now and discuss when weak or perfect security can be attained (the security may depend on the probability distribution defined on the input s-tuples). Then we generalize this approach to handle the slightly more complicated case of asymmetric AONTs.

An (N, k, v)-*array* is an N by k array, say A, whose entries are elements chosen from an alphabet Γ of order v. Suppose the k columns of A are labelled by the elements in the set C. Let $D \subseteq C$, and define A_D to be the array

obtained from A by deleting all the columns $c \notin D$. We say that A is *unbiased* with respect to D if the rows of A_D contain every $|D|$-tuple of elements of Γ exactly $N/v^{|D|}$ times. Of course, this requires that N is divisible by $v^{|D|}$.

An AONT, say ϕ, is a bijection from Γ^s to Γ^s, where Γ is a v-set. The *array representation* of ϕ is a $(v^s, 2s, v)$-array, say A, that is constructed as follows: For every input s-tuple $(x_1, \ldots, x_s) \in \Gamma^s$, there is a row of A containing the entries $x_1, \ldots, x_s, y_1, \ldots, y_s$, where $\phi(x_1, \ldots, x_s) = (y_1, \ldots, y_s)$.

Our combinatorial definition of an AONT, Definition 2.1, involves arrays that are unbiased with respect to certain subsets of columns. This definition is an obvious generalization of previous definitions for (t, t, s, v)-AONTs from [4, 7].

Definition 2.1. *A* (t_{in}, t_{out}, s, v)-all-or-nothing transform *is a* $(v^s, 2s, v)$*-array, say* A*, with columns labelled* $1, \ldots, 2s$*, that is unbiased with respect to the following subsets of columns:*

1. $\{1, \ldots, s\}$,
2. $\{s+1, \ldots, 2s\}$, *and*
3. $I \cup J$*, for all* $I \subseteq \{1, \ldots, s\}$ *with* $|I| = t_{in}$ *and all* $J \subseteq \{s+1, \ldots, 2s\}$ *with* $|J| = s - t_{out}$.

We interpret the first s columns of A as indexing the s inputs and the last s columns as indexing the s outputs. Then, as mentioned above, properties 1 and 2 ensure that the array A defines a bijection ϕ. Property 3 guarantees that knowledge of any $s - t_{out}$ outputs does not rule out any possible values for any t_{in} inputs.

The following results concerning (t, t, s, v)-AONTs are from [9].

Theorem 2.1. *Suppose* $\phi : \Gamma^s \to \Gamma^s$ *is a bijection, where* Γ *is an alphabet of size* v*, and suppose* $1 \le t \le s$.

1. *Suppose any input* s*-tuple occurs with positive probability. Then the mapping* ϕ *is a weakly secure AONT if and only if its array representation is a* (t, t, s, v)*-AONT.*
2. *The mapping* ϕ *is a perfectly secure AONT if and only if its array representation is a* (t, t, s, v)*-AONT and every input* s*-tuple occurs with the same probability.*

When we turn to asymmetric AONTs, there is an additional subtlety, namely that we can obtain weak security for combinatorial structures that are weaker than the arrays defined in Definition 2.1. We can characterize asymmetric AONTs achieving weak security in terms of arrays that satisfy covering properties with respect to certain sets of columns. As before, suppose A is an (N, k, v)-array, whose entries are elements chosen from an alphabet Γ of order v and whose columns are labelled by the set C. Also, for $D \subseteq C$, define A_D as before. We say that A is *covering* with respect to a subset of columns $D \subseteq C$ if the rows of A_D contain every $|D|$-tuple of elements of Γ *at least once*.

Remark 2.1. An array that satisfies the covering property for all subsets of t columns is called a *t-covering array*. Such arrays have many important applications, including software testing. See [3, §VI.10] for a brief survey of covering arrays.

We state a few simple observations without proof.

Lemma 2.2. *Suppose* A *is an* (N, k, v)*-array with columns labelled by* C.

1. *If* A *is unbiased or covering with respect to* $D \subseteq C$*, then* $N \ge v^{|D|}$.
2. *If* A *is unbiased with respect to* $D \subseteq C$*, then* A *is covering with respect to* D.
3. *If* $D \subseteq C$ *and* $N = v^{|D|}$*, then* A *is unbiased with respect to* D *if and only if* A *is covering with respect to* D.
4. *If* A *is unbiased or covering with respect to* $D \subseteq C$*, then* A *is unbiased or covering (resp.) with respect to all* $D' \subseteq D$.

Definition 2.2. *A* (t_{in}, t_{out}, s, v)-weak-all-or-nothing transform *is a* $(v^s, 2s, v)$*-array, say* A*, with columns labelled* $1, \ldots, 2s$*, that is covering with respect to the following subsets of columns:*

1. $\{1, \ldots, s\}$,
2. $\{s+1, \ldots, 2s\}$, *and*
3. $I \cup J$*, for all* $I \subseteq \{1, \ldots, s\}$ *with* $|I| = t_{in}$ *and all* $J \subseteq \{s+1, \ldots, 2s\}$ *with* $|J| = s - t_{out}$.

We note that a (t, t, s, v)-weak-AONT is equivalent to a (t, t, s, v)-AONT. This follows immediately from Lemma 2.2. However, a (t_{in}, t_{out}, s, v)-weak-AONT is not necessarily a (t_{in}, t_{out}, s, v)-AONT if $t_{in} < t_{out}$. Example 2.1 depicts a $(1, 2, 3, 2)$-weak-AONT that is not a $(1, 2, 3, 2)$-AONT.

Example 2.1. We present a $(1, 2, 3, 2)$-weak AONT over the alphabet $\{a, b\}$. The array representation of this AONT is as follows:

x_1	x_2	x_3	y_1	y_2	y_3
a	a	a	a	a	a
a	a	b	b	b	a
a	b	a	b	a	b
a	b	b	b	a	a
b	a	a	a	b	b
b	a	b	a	b	a
b	b	a	a	a	b
b	b	b	b	b	b

This array is biased with respect to various pairs of columns (x_i, y_j). For example, we verify that this array is biased with respect to columns x_1 and y_1. Specifically, the ordered pairs (a, a) and (b, b) each occur once, but the ordered pairs (a, b) and (b, a) each occur three times.

However, for all choices of x_i and y_j, it can be verified that A is covering with respect to the pair of columns (x_i, y_j). ■

The following theorem extends part of Theorem 2.1 to the asymmetric case. Proofs are omitted, as they are essentially the same as the proofs in [9].

Theorem 2.3. *Suppose $\phi : \Gamma^s \to \Gamma^s$ is a bijection, where Γ is an alphabet of size v, and suppose $1 \le t_{in} \le t_{out} \le s$.*

1. *Suppose any input s-tuple occurs with positive probability. Then the mapping ϕ is weakly secure if and only if its array representation is a (t_{in}, t_{out}, s, v)-weak-AONT.*
2. *The mapping ϕ is perfectly secure if its array representation is a (t_{in}, t_{out}, s, v)-AONT and every input s-tuple occurs with the same probability.*

Remark 2.2. *The second part of Theorem 2.1 is "if and only if". However, we do not know if the converse of the second part of Theorem 2.3 is true when $t_{in} < t_{out}$.*

2.1 GENERAL PROPERTIES

In the rest of the paper, we focus on (t_{in}, t_{out}, s, v)-AONTs that satisfy Definition 2.1. These are the AONTs that are unbiased with respect to various subsets of columns. First, we record various general properties about these AONTs. Some of these results are generalizations of previous results pertaining to (t, t, s, v)-AONT, and most of them follow easily from Lemma 2.2.

The following result was shown in [14] for the case $t_{in} = t_{out}$. The generalization to arbitrary $t_{in} \le t_{out}$ is obvious.

Theorem 2.4. *A mapping $\phi : \mathcal{X}^s \to \mathcal{X}^s$ is a (t_{in}, t_{out}, s, v)-AONT if and only if ϕ^{-1} is an $(s-t_{out}, s-t_{in}, s, v)$-AONT.*

Proof. Interchange the first s columns and the last s columns in the array representation of the AONT ϕ. □

An *orthogonal array* $OA(t, k, v)$ is a (v^t, n, v) array, say A, that is unbiased with respect to any t columns. The next theorem generalizes [7, Corollary 35].

Theorem 2.5. *If there exists an $OA(s, 2s, v)$, then there exists a (t_{in}, t_{out}, s, v)-AONT for all t_{in} and t_{out} such that $1 \le t_{in} \le t_{out} \le s$.*

Proof. It suffices to show that an $\mathrm{OA}(s, 2s, v)$ satisfies the conditions of Definition 2.1. This follows immediately from Lemma 2.2 and the observation that

$$1 \le t_{in} + s - t_{out} \le s$$

for all t_{in} and t_{out} such that $1 \le t_{in} \le t_{out} \le s$. □

Levenshtein [11] defined *split orthogonal arrays* (or *SOAs*) as follows. A split orthogonal array $SOA(t_1, t_2; n_1, n_2; v)$ is a $(v^{t_1+t_2}, n_1 + n_2, v)$ array, say A, that satisfies the following properties:

1. the columns of A are partitioned into two sets, of sizes n_1 and n_2, respectively, and
2. A is unbiased with respect to any $t_1 + t_2$ columns in which t_1 columns are chosen from the first set of n_1 columns and t_2 columns are chosen from the second set of n_2 columns.

From the definition of split orthogonal arrays, we can immediately obtain the following theorem.

Theorem 2.6. *Suppose there exists a (t_{in}, t_{out}, s, v)-AONT. Then there exists an $SOA(t_{in}, s - t_{out}, s, s, v)$.*

Proof. Consider the array representation of a (t_{in}, t_{out}, s, q)-AONT. Denote $n_1 = s, n_2 = s, t_1 = t_{in}$ and $t_2 = s - t_{out}$. Fixing any t_2 outputs does not yield any information about any t_1 inputs. Hence, the array is unbiased with respect to any $s - t_{out} + t_{in}$ columns where t_{in} columns are chosen from the first set of s columns and $s - t_{out}$ columns are chosen from the second set of s columns. Therefore the array is an $\mathrm{SOA}(t_{in}, s - t_{out}, s, s, v)$. □

Theorems 2.5 and 2.6 show that, in a certain sense, AONTs (symmetric and asymmetric) are "between" orthogonal arrays and split orthogonal arrays. More precisely, an $\mathrm{OA}(s, 2s, v)$ implies the existence of a (t_{in}, t_{out}, s, v)-AONT (for $1 \le t_{in} \le t_{out} \le s$), which in turn implies the existence of an $\mathrm{SOA}(t_{in}, s - t_{out}, s, s, v)$.

3 LINEAR ASYMMETRIC AONTS

Suppose q is a prime power. If every output of a (t_{in}, t_{out}, s, v)-AONT is an $\mathbb{F}_q$-linear function of the inputs, the AONT is a *linear* (t_{in}, t_{out}, s, q)-AONT. Note that we will write a linear (t_{in}, t_{out}, s, q)-AONT in the form $\mathbf{y} = \mathbf{x}M^{-1}$, where M is an invertible s by s matrix over $\mathbb{F}_q$ (as always, $\mathbf{x}$ is an input s-tuple and $\mathbf{y}$ is an output s-tuple). Of course it holds also that $\mathbf{x} = \mathbf{y}M$.

Remark 3.1. *The $(1, 2, s, 2)$-AONT described in Example 1.1 (for even values of s) is a linear AONT, where M is the s by s matrix with 0's on the diagonal and 1's elsewhere. When s is even, M is invertible and $M^{-1} = M$.*

The following lemma generalizes [4, Lemma 1].

Lemma 3.1. *Suppose that q is a prime power and M is an invertible s by s matrix with entries from $\mathbb{F}_q$. Suppose $1 \le t_{in} \le t_{out} \le s$. Then the function $\mathbf{y} = \mathbf{x}M^{-1}$ defines a linear (t_{in}, t_{out}, s, q)-AONT if and only if every t_{out} by t_{in} submatrix of M has rank t_{in}.*

Proof. Suppose $I, J \subseteq \{1, \ldots, s\}$, $|I| = t_{in}$, $|J| = t_{out}$. Let $\mathbf{x}' = (x_i : i \in I)$. We have $\mathbf{x}' = \mathbf{y}M'$, where M' is the s by t_{in} matrix formed from M by deleting all columns not in I. Now assume that y_j is fixed for all $j \notin J$ and denote $\mathbf{y}' = (y_j : j \in J)$. Then we can write $\mathbf{x}' = \mathbf{y}'M'' + \mathbf{c}$, where M'' is the t_{out} by t_{in} submatrix of M formed from M by deleting all columns not in I and all rows not in J, and $\mathbf{c}$ is a vector of constants. If M'' is of rank t_{in}, then $\mathbf{x}'$ is completely undetermined, in the sense that $\mathbf{x}'$ takes on all values in $(\mathbb{F}_q)^{t_{in}}$ as $\mathbf{y}'$ varies over $(\mathbb{F}_q)^{t_{out}}$. On the other hand, if $t' = \mathsf{rank}(M'') < t_{in}$, then $\mathbf{x}'$ can take on only $(\mathbb{F}_q)^{t'}$ possible values. □

The following corollaries pertain to the special case where $t_{in} = t_{out} = t$.

Corollary 3.2. *[4] Suppose M is an invertible s by s matrix with entries from $\mathbb{F}_q$. Then $\mathbf{y} = \mathbf{x}M^{-1}$ defines a linear (t, t, s, q)-AONT if and only if every t by t submatrix of M is invertible.*

Corollary 3.3. *Suppose that $\mathbf{y} = \mathbf{x}M^{-1}$ defines a linear (t, t, s, q)-AONT. Then $\mathbf{y} = \mathbf{x}M$ defines a linear $(s - t, s - t, s, q)$-AONT.*

Corollary 3.4. *Suppose M is an invertible s by s matrix with entries from $\mathbb{F}_q$. Then $\mathbf{y} = \mathbf{x}M^{-1}$ defines a linear (t, t, s, q)-AONT if and only if every $s - t$ by $s - t$ submatrix of M^{-1} is invertible.*

Another approach to construct asymmetric AONTs is to use t-AONTs or other asymmetric AONTs. The following results will present various such constructions. First, we generalize [7, Theorem 20].

Lemma 3.5. *If $1 \le t_{in} \le t_{out} < s$, then the existence of a linear (t_{in}, t_{out}, s, q)-AONT implies the existence of a linear $(t_{in}, t_{out}, s - 1, q)$-AONT.*

Proof. Let M be a matrix for a linear (t_{in}, t_{out}, s, q)-AONT. Since M is invertible, if we calculate its determinant using the cofactor expansion of M with respect to its first row, at least one of the $(s - 1) \times (s - 1)$ submatrices is invertible. Also, any $t_{out} \times t_{in}$ submatrix of M, including those in the invertible submatrix, are of rank t_{in}. Hence, the invertible submatrix is a $(t_{in}, t_{out}, s - 1, q)$-AONT. □

Lemma 3.6. *If $1 \le t_{in} \le t_{out} \le s$, then the existence of a linear (t_{in}, t_{out}, s, q)-AONT implies the existence of a linear (t_{in}, t'_{out}, s, q)-AONT for all t'_{out} such that $t_{out} \le t'_{out} \le s$.*

Proof. Consider the matrix representation of the linear $(t_{in}, t_{out}, s, q)-$AONT. Every t'_{out} by t_{in} submatrix is rank t_{in}, because all its $t_{out} \times t_{in}$ submatrices are of rank t_{in}. □

Lemma 3.7. *If $1 \le t_{in} \le t_{out} \le s$, then the existence of a linear (t_{in}, t_{out}, s, q)-AONT implies the existence of a linear (t'_{in}, t_{out}, s, q)-AONT for any t'_{in} such that $1 \le t'_{in} \le t_{in} \le s$.*

Example 3.1. *We observe that existence of a linear* (t_{in}, t_{out}, s, q)*-AONT does not necessarily imply the existence of a linear* (t_{in}, t_{in}, s, q)*-AONT or a linear* (t_{out}, t_{out}, s, q)*-AONT. Consider the linear* $(2, 3, 4, 2)$*-AONT presented by the following matrix*

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

While every 3×2 *submatrix of the matrix above is of rank* 2*, a* $(2, 2, s, 2)$*-AONT does not exist if* $s > 2$*, as was proven by D'Arco et al. [4].*

Additionally, from Corollary 3.3, a linear $(3, 3, 4, 2)$*-AONT would be equivalent to a linear* $(1, 1, 4, 2)$*-AONT. Since it was shown in [13] that an* $(1, 1, 4, 2)$*-AONT does not exist, we conclude that a linear* $(3, 3, 4, 2)$*-AONT does not exist.* ■

The main general construction for linear (t, t, s, q)-AONTs in [4] uses Cauchy matrices. We provide a generalization that applies to asymmetric AONTs.

Theorem 3.8. *Suppose* $q \geq 2s$ *is a prime power and* $1 \leq t_{in} \leq t_{out} \leq s$*. Then there exists a linear* (t_{in}, t_{out}, s, q)*-AONT.*

Proof. In [4, Theorem 2], it was shown that a linear (t, t, s, q)-AONT exists if $q \geq 2s$ is a prime power and $1 \leq t \leq s$. Let $t_{in} = t_{out} = t$ and then apply Lemma 3.7. This shows that there is a a linear (t'_{in}, t_{out}, s, q)-AONT provided that $1 \leq t'_{in} \leq t_{out} \leq s$.

□

3.1 LINEAR $(1, t_{out}, s, q)$-AONT

We noted in Remark 3.1 that there exists a linear $(1, 2, s, 2)$-AONT for all even values of $s \geq 2$. In the next lemma, we show that linear $(1, 2, s, 2)$-AONTs exist for odd values of s.

Lemma 3.9. *There is a linear* $(1, 2, s, 2)$*-AONT for any odd value of* $s \geq 3$*.*

Proof. Suppose $s \geq 3$ is odd. Let M be the s by s matrix whose first subdiagonal consists of 0's, but all other entries are 1's. For example, when $s = 5$, we have

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \end{pmatrix}.$$

The matrix M is invertible and its inverse is an s by s matrix with a right top submatrix that is an identity matrix, and 1's occur along the last row and first column. For example, when $s = 5$, we have

$$M^{-1} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

Further, any 2 by 1 submatrix of M has rank 1 because there is at most one occurrence of 0 in each column of M. □

Recall that $\mathbf{y} = \mathbf{x}M^{-1}$ and $\mathbf{x} = \mathbf{y}M$. Given s inputs $x_1, \ldots, x_s \in \mathbb{Z}_2$, the above-discussed transform can be computed as follows:

$$y_1 = \sum_{i=1}^{s} x_i.$$

$$y_i = x_{i-1} + x_s, \text{ for } 2 \leq i \leq s.$$

This yields the s outputs $y_1, \ldots, y_s$. The inverse transform is computed as

$$x_s = \sum_{i=1}^{s} y_i$$

$$x_i = x_s + y_{i+1}, \text{ for } 2 \leq i \leq s.$$

Thus, computation of the transform or its inverse requires $2s - 2$ addition operations in $\mathbb{Z}_2$ (i.e., exclusive-ors).

Theorem 3.10. *Suppose q is a prime power and $1 \leq t_{out} \leq s$. Then there is a linear $(1, t_{out}, s, q)$-AONT unless $q = 2$ and $t_{out} = 1$. Further, there does not exist any $(1, 1, s, 2)$-AONT.*

Proof. When $q > 2$, it was shown in [13, Corollary 2.3] that there exists a linear $(1, 1, s, q)$-AONT for all $s \geq 1$. Applying Lemma 3.6, there exists a linear $(1, t_{out}, s, q)$-AONT for all prime powers $q > 2$ and all t_{out} and s such that $1 \leq t_{out} \leq s$.

We have also noted in Remark 3.1 that there exists a linear $(1, 2, s, 2)$-AONT for all even values of $s \geq 2$. Applying Lemma 3.6, there exists a linear $(1, t_{out}, s, 2)$-AONT for all t_{out} and s such that s is even and $2 \leq t_{out} \leq s$. From Lemmas 3.6 and 3.9, there exists a linear $(1, t_{out}, s, 2)$-AONT for all t_{out} and s such that s is odd and $2 \leq t_{out} \leq s$.

Finally, it was shown in [13] that there does not exist any $(1, 1, s, 2)$-AONT. □

3.2 LINEAR $(2, s-1, s, 2)$-AONT

In this section, we consider linear $(2, s-1, s, 2)$-AONTs.

For even values of $s \geq 4$, we use the $(1, 2, s, 2)$-AONT from Remark 3.1. This AONT is based on the s by s matrix M with 0's on the diagonal and 1's elsewhere. We have already noted that this matrix is invertible. To show that it gives rise to a $(2, s-1, s, 2)$-AONT, we need to show that any $s-1$ by 2 submatrix has rank 2. It can be observed that any choice of $s-1$ rows and two columns will contain at least $s-3 \geq 1$ occurrences of the row $(1, 1)$ and at least one copy of the row $(0, 1)$ or $(1, 0)$. Therefore, we have proven the following.

Lemma 3.11. *For any even integer $s \geq 4$, there exists a linear $(2, s-1, s, 2)$-AONT.*

Now we turn to odd values of s.

Lemma 3.12. *For any odd integer $s \geq 5$, there is a linear $(2, s-1, s, 2)$-AONT.*

Proof. For an odd integer $s \geq 5$, define the s by s matrix B_s to have 1's in the entries on the main diagonal, the last row and the last column, and 0's elsewhere.

For example, the matrix B_5 is as follows:

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

Suppose we subtract rows $1, \ldots, s-1$ of B_s from row s. Then we obtain an upper triangular matrix with 1's on the main diagonal. This proves that B_s is invertible.

Now we prove that any $s-1$ by 2 submatrix has rank two. First, consider columns i and s, where $1 \leq i \leq s-1$. The s rows of this submatrix contain two copies of $(1, 1)$ and $s-2$ copies of $(0, 1)$. Therefore, any $s-1$ rows still contain at least one copy of $(1, 1)$ and at least one copy of $(0, 1)$. This means that the $s-1$ by 2 submatrix has rank 2.

Next, we consider columns i and j, where $1 \leq i < j \leq s-1$. The s rows of this submatrix contain one copy of each of $(0, 1)$, $(1, 0)$ and $(1, 1)$. Therefore, any $s-1$ rows still contain at least two of the three pairs $(0, 1)$, $(1, 0)$ and $(1, 1)$. This means that the $s-1$ by 2 submatrix has rank 2. □

Theorem 3.13. *There is a linear $(2, s-1, s, 2)$-AONT if and only if $s \geq 4$.*

Proof. If a (t_{in}, t_{out}, s, q)-AONT exists, we must have $t_{in} \leq t_{out}$. Hence, $s \geq 3$ if a $(2, s-1, s, 2)$-AONT exists. D'Arco et al. [4] proved that a linear $(2, 2, 3, 2)$-AONT does not exist. For $s \geq 4$, Lemmas 3.11 and 3.12 show that a linear $(2, s-1, s, 2)$-AONT exists. □

3.3 LINEAR (t_{in}, t_{out}, s, q)-AONT WITH $t_{in} \geq 2$

In this section, we study linear (t_{in}, t_{out}, s, q)-AONTs with $t_{in} \geq 2$. We first prove a general upper bound on s as a function of t_{in}, t_{out} and q. Then we consider the case $t_{in} = 2$ in detail.

Theorem 3.14. *Suppose there exists a linear (t_{in}, t_{out}, s, q)-AONT with $2 \leq t_{in} \leq t_{out}$. Then the following bound holds:*

$$s \leq \frac{(t_{out} - 1)(q^{t_{in}} - 1)}{(t_{in} - 1)(q - 1)}.$$

Proof. Fix any t_{in} columns of the matrix M and consider the resulting submatrix M'. Recall that any t_{out} by t_{in} submatrix of M must have rank t_{in}.

There are $q^{t_{in}}$ possible t_{in}-tuples for any given row of M'. We can replace an all-zero t_{in}-tuple with any other t_{in}-tuple, and it does not decrease the rank of any t_{out} by t_{in} submatrix in M'. Hence, we can assume that there is no all-zero t_{in}-tuple among the rows of M'. Therefore, there are $q^{t_{in}} - 1$ possible rows in M'.

For any two nonzero t_{in}-tuples, say a and b, define $a \sim b$ if there is a nonzero element $\alpha \in \mathbb{F}_q$ such that $a = \alpha b$. Clearly $\sim$ is an equivalence relation, and there are $(q^{t_{in}} - 1)/(q - 1)$ equivalence classes, each of size $q - 1$.

Suppose the equivalence classes of rows are denoted by $\mathcal{E}_i$. Further, suppose there are a_i rows from $\mathcal{E}_i$ in M', for $1 \le i \le (q^{t_{in}} - 1)/(q - 1)$. The sum of the a_i's is equal to s and hence the average value of an a_i is

$$\overline{a} = \frac{s(q-1)}{q^{t_{in}} - 1}.$$

Let L denote the sum of the $t_{in} - 1$ largest a_i's. It is clear that

$$L \ge (t_{in} - 1)\overline{a} = \frac{s(t_{in} - 1)(q-1)}{q^{t_{in}} - 1}.$$

Also, because the t_{out} rows of M' cannot come from fewer than t_{in} equivalence classes, we have

$$L \le t_{out} - 1.$$

Hence, combining the two inequalities, we see that

$$s \le \frac{(t_{out} - 1)(q^{t_{in}} - 1)}{(t_{in} - 1)(q-1)}.$$

□

We now look at the case $t_{in} = 2$ in more detail.

Theorem 3.15. *Suppose there exists a linear $(2, t_{out}, s, q)$-AONT with $2 \le t_{out}$. Then the following bound holds:*

$$s \le \max\{1 + (t_{out} - 2)(q+1), 2 + (t_{out} - 1)(q-1)\}.$$

Proof. Consider an s by 2 submatrix M' and let a_0 be the number of $(0, 0)$ rows in this submatrix. We divide the proof into two cases.

case (1)

Suppose $a_0 \ge 1$. We claim that M' contains at most $t_{out} - a_0 - 1$ rows from any one equivalence class $\mathcal{E}_i$, where equivalence classes are as defined in the proof of Theorem 3.14. This follows because $t_{out} - a_0$ rows from one equivalence class, together with the a_0 rows of 0's, would result in M' having rank 1. Excluding the rows of 0's, there are $q + 1$ possible equivalence classes of rows, so

$$s \le a_0 + (t_{out} - a_0 - 1)(q+1) \le 1 + (t_{out} - 2)(q+1).$$

case (2)

If we are not in case (1), then $a_0 = 0$ for *every* s by 2 submatrix M'. There can be at most one 0 in each row of M, so there are at most s occurrences of 0 in M. Therefore, there must be two columns in M that contain a total of at most two 0's. We focus on this particular s by 2 submatrix M'.

Let the number of 0's in M' be denoted by a; we have noted that $a \le 2$. In the $s - a$ rows that do not contain a 0, there are at most $t_{out} - 1$ rows from any equivalence class $\mathcal{E}_i$. Note that we have excluded two $\mathcal{E}_i$'s, i.e., $(*, 0)$ and $(0, *)$, so

$$s \le a + (t_{out} - 1)(q-1) \le 2 + (t_{out} - 1)(q-1).$$

Since one of the above two cases must hold, we have

$$s \le \max\{1 + (t_{out} - 2)(q+1), 2 + (t_{out} - 1)(q-1)\}.$$

□

Table 1: Examples of bounds from Theorems 3.14 and 3.15.

t_{in}	q	t_{out}	Upper bound on $S(t_{in}, t_{out}, q)$	Justification
2	2	2	$t_{out} + 1$	Theorem 3.15
2	2	≥ 3	$3t_{out} - 5$	Theorem 3.15
2	3	2, 3	$2t_{out}$	Theorem 3.15
2	3	≥ 4	$4t_{out} - 7$	Theorem 3.15
2	4	2, 3	$3t_{out} - 1$	Theorem 3.15
2	4	≥ 4	$5t_{out} - 9$	Theorem 3.15
3	3	any	$\frac{13(t_{out}-1)}{2}$	Theorem 3.14
3	4	any	$20(t_{out} - 1)$	Theorem 3.14
3	5	any	$\frac{121(t_{out}-1)}{2}$	Theorem 3.14

We note that

$$1 + (t_{out} - 2)(q + 1) < (t_{out} - 1)(q + 1)$$

and

$$2 + (t_{out} - 1)(q - 1) < (t_{out} - 1)(q + 1),$$

so

$$\max\{1 + (t_{out} - 2)(q + 1), 2 + (t_{out} - 1)(q - 1)\} < (t_{out} - 1)(q + 1).$$

Hence the bound from Theorem 3.15 improves Theorem 3.14 when $t_{in} = 2$.

For positive integers t_{in} and t_{out}, where $1 \leq t_{in} \leq t_{out}$, and a prime power q, define

$$S(t_{in}, t_{out}, q) = \max\{s : \text{a linear } (t_{in}, t_{out}, s, q)\text{-AONT exists}\}.$$

Note that $S(t_{in}, t_{out}, q) \geq t_{out}$ because the t_{out} by t_{out} identity matrix is a $(t_{in}, t_{out}, t_{out}, q)$-AONT.

Theorem 3.16. *Suppose $1 \leq t_{in} \leq t_{out}$ and q is a prime power. Then there exists a (t_{in}, t_{out}, s, q)-AONT for $t_{out} \leq s \leq S(t_{in}, t_{out}, q)$.*

Proof. This is an immediate consequence of Lemma 3.5. □

We mainly consider cases where $2 \leq t_{in} < t_{out}$. However, before proceeding, we recall some previous results concerning the special case $t_{in} = t_{out} = 2$. Theorems 3.14 and 3.15 both assert that $S(2, 2, q) \leq q + 1$. However, the stronger result that $S(2, 2, q) \leq q$ was previously shown in [7, Theorem 14]. There are also some known lower bounds on $S(2, 2, q)$, which are recorded in the following theorem.

Theorem 3.17. *Suppose q is a prime power. Then the following bounds hold.*

1. $\lfloor q/2 \rfloor \leq S(2, 2, q) \leq q$.
2. $q - 1 \leq S(2, 2, q) \leq q$ *if* $q = 2^n - 1$ *is prime, for some integer n.*
3. $S(2, 2, q) = q$ *if q is prime.*

Proof. 1. and 2. are shown in [7], while 3. is proven in [14]. □

The cases when $t_{in} < t_{out}$ have not received previous study in the literature. Theorems 3.14 and 3.15 provide upper bounds on $S(t_{in}, t_{out}, q)$. We evaluate some of these upper bounds for specific families of parameters in Table 1.

We can also obtain lower bounds on $S(2, t_{out}, q)$, for specific choices of t_{out} and q, from computer searches. The results of our searches are presented in Examples A.1 to A.15. Table 2 lists upper and lower bounds on $S(2, t_{out}, q)$, for some fixed values of t_{out}, and q. There are four cases where we can report exact values of $S(2, t_{out}, q)$. When $(t_{out}, q) = (3, 2)$ and $(3, 3)$, we have found examples that meet the upper bounds from Theorem 3.15. For $(t_{out}, q) = (4, 2)$ and $(5, 2)$, the searches were run to completion and the exact values of $S(2, t_{out}, q)$ turn out to be strictly less than the bounds obtained from Theorem 3.15, which are $S(2, 4, 2) \leq 7$ and $S(2, 5, 2) \leq 10$.

Table 2: Upper and lower bounds on $S(2, t_{out}, q)$

t_{out}	q	lower bound	reference	upper bound	reference
3	2	4	Example 3.1	4	Theorem 3.15
4	2	5	Example A.1	5	exhaustive search
5	2	8	Example A.2	8	exhaustive search
6	2	10	Example A.3	13	Theorem 3.15
7	2	12	Example A.4	16	Theorem 3.15
8	2	13	Example A.5	19	Theorem 3.15
3	3	6	Example A.6	6	Theorem 3.15
4	3	8	Example A.7	9	Theorem 3.15
5	3	9	Example A.8	13	Theorem 3.15
6	3	13	Example A.9	17	Theorem 3.15
3	4	6	Example A.10	8	Theorem 3.15
4	4	9	Example A.11	11	Theorem 3.15
5	4	11	Example A.12	16	Theorem 3.15
3	5	8	Example A.13	10	Theorem 3.15
4	5	10	Example A.14	14	Theorem 3.15
3	7	8	Example A.15	14	Theorem 3.15

4 DISCUSSION

There are many open problems involving asymmetric AONTs. It would certainly be of interest to find improved necessary conditions and general constructions. The first cases are when $t_{in} = 2$. A starting point would be to close the gaps in the bounds reported in Table 2.

As mentioned in Remark 2.2, it is unknown if the converse of part 2 of Theorem 2.3 is true when $t_{in} < t_{out}$. We feel that this question is worthy of further study.

ACKNOWLEDGMENT

We thank Bill Martin for helpful discussions.

REFERENCES

[1] Victor Boyko. “On the security properties of OAEP as an all-or-nothing transform”. In: *Advances in Cryptology – CRYPTO’ 99,* Lecture Notes in Computer Science **1666**. Springer. 1999, pp. 503–518.

[2] Ran Canetti et al. “Exposure-resilient functions and all-or-nothing transforms”. In: *Advances in Cryptology – EUROCRYPT 2000,* Lecture Notes in Computer Science **1807**. Springer. 2000, pp. 453–469.

[3] Charles J. Colbourn and Jeffrey H. Dinitz. *Handbook of Combinatorial Designs*. 2nd. Chapman & Hall/CRC, 2006. ISBN: 1584885068.

[4] Paolo D’Arco, Navid Nasr Esfahani, and Douglas R. Stinson. “All or nothing at all”. In: *The Electronic Journal of Combinatorics* 23.4 (2016), Paper P4.10.

[5] Anand Desai. “The security of all-or-nothing encryption protecting against exhaustive key search”. In: *Annual International Cryptology Conference – CRYPTO 2000,* Lecture Notes in Computer Science **1880**. Springer. 2000, pp. 359–375.

[6] Navid Nasr Esfahani. “Generalizations of All-or-Nothing Transforms and their Application in Secure Distributed Storage”. PhD thesis. 2021. URL: `http://hdl.handle.net/10012/16739`.

[7] Navid Nasr Esfahani, Ian Goldberg, and Douglas R. Stinson. “Some results on the existence of t-all-or-nothing transforms over arbitrary alphabets”. In: *IEEE Transactions on Information Theory* 64.4 (2017), pp. 3136–3143.

[8] Navid Nasr Esfahani and Douglas R. Stinson. “Computational results on invertible matrices with the maximum number of invertible 2× 2 submatrices”. In: *Australasian Journal of Combinatorics* 69.1 (2017), pp. 130–144.

[9] Navid Nasr Esfahani and Douglas R. Stinson. “On Security Properties of All-or-nothing Transforms”. In: (2021). arXiv: `2103.05697`.

[10] Ghassan O. Karame et al. “Securing Cloud Data under Key Exposure”. In: *IEEE Transactions on Cloud Computing* 7 (2019), pp. 838–849.

[11] Vladimir Levenshtein. “Split orthogonal arrays and maximum independent resilient systems of functions”. In: *Designs, Codes and Cryptography* 12 (1997), pp. 131–160.

[12] Ronald L. Rivest. “All-or-nothing encryption and the package transform”. In: *Fast Software Encryption,* Lecture Notes in Computer Science **1267**. Springer. 1997, pp. 210–218.

[13] Douglas R. Stinson. “Something about all or nothing (transforms)”. In: *Designs, Codes and Cryptography* 22.2 (2001), pp. 133–138.

[14] Xin Wang, Jie Cui, and Lijun Ji. “Linear $(2, p, p)$-AONTs exist for all primes p”. In: *Designs, Codes and Cryptography* 87.10 (2019), pp. 2185–2197.

[15] Yiwei Zhang et al. “Invertible binary matrices with maximum number of 2-by-2 invertible submatrices”. In: *Discrete Mathematics* 340.2 (2017), pp. 201–208.

A APPENDIX

Example A.1. *A linear* $(2, 4, 5, 2)$*-AONT:*

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

Example A.2. *A linear* $(2, 5, 8, 2)$*-AONT:*

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

Example A.3. *A linear* $(2, 6, 10, 2)$*-AONT:*

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

Example A.4. *A linear* $(2, 7, 12, 2)$*-AONT:*

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

Example A.5. *A linear* $(2, 8, 13, 2)$*-AONT:*

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

Example A.6. *A linear* $(2, 3, 6, 3)$*-AONT:*

$$\begin{pmatrix}
0 & 1 & 1 & 1 & 1 & 1 \\
1 & 0 & 1 & 1 & 2 & 2 \\
1 & 1 & 0 & 2 & 1 & 2 \\
1 & 1 & 2 & 0 & 2 & 1 \\
1 & 2 & 1 & 2 & 0 & 1 \\
1 & 2 & 2 & 1 & 1 & 0
\end{pmatrix}.$$

Example A.7. *A linear* $(2, 4, 8, 3)$*-AONT:*

$$\begin{pmatrix}
0 & 0 & 0 & 1 & 1 & 1 & 1 & 2 \\
0 & 1 & 1 & 0 & 1 & 2 & 2 & 0 \\
0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 2 \\
1 & 0 & 1 & 1 & 0 & 1 & 2 & 0 \\
1 & 1 & 0 & 2 & 2 & 0 & 1 & 0 \\
1 & 1 & 2 & 0 & 1 & 1 & 2 & 1 \\
1 & 2 & 0 & 1 & 0 & 2 & 1 & 1
\end{pmatrix}.$$

Example A.8. *A linear* $(2, 5, 9, 3)$*-AONT:*

$$\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 2 \\
0 & 1 & 1 & 1 & 2 & 0 & 2 & 2 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 2 & 2 & 1 \\
1 & 0 & 1 & 1 & 0 & 2 & 1 & 2 & 0 \\
1 & 1 & 0 & 2 & 0 & 0 & 1 & 2 & 1 \\
1 & 1 & 2 & 0 & 1 & 1 & 2 & 1 & 0 \\
1 & 2 & 0 & 1 & 2 & 1 & 1 & 2 & 0
\end{pmatrix}.$$

Example A.9. *A linear* $(2, 6, 13, 3)$*-AONT:*

$$\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 2 \\
0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 2 & 2 \\
0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 2 & 2 & 0 \\
0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\
0 & 1 & 1 & 1 & 2 & 1 & 2 & 1 & 0 & 0 & 1 & 2 & 2 \\
1 & 0 & 1 & 0 & 2 & 1 & 1 & 2 & 0 & 2 & 0 & 1 & 2 \\
1 & 0 & 1 & 1 & 0 & 0 & 2 & 0 & 1 & 2 & 2 & 1 & 2 \\
1 & 0 & 1 & 1 & 1 & 2 & 0 & 2 & 2 & 0 & 1 & 2 & 0 \\
1 & 1 & 0 & 2 & 0 & 2 & 0 & 1 & 1 & 2 & 2 & 2 & 2 \\
1 & 1 & 2 & 0 & 1 & 1 & 0 & 0 & 2 & 1 & 2 & 1 & 2 \\
1 & 1 & 2 & 0 & 2 & 0 & 2 & 2 & 1 & 0 & 1 & 1 & 1 \\
1 & 2 & 0 & 1 & 2 & 0 & 1 & 1 & 0 & 1 & 2 & 2 & 0 \\
1 & 2 & 0 & 2 & 1 & 1 & 2 & 0 & 1 & 0 & 0 & 2 & 1
\end{pmatrix}.$$

Example A.10. *A linear* $(2,3,6,4)$*-AONT:*

$$\begin{pmatrix} 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 2 \\ 1 & 0 & 0 & 1 & 2 & 1 \\ 1 & 1 & 1 & 0 & 1 & 3 \\ 1 & 2 & 3 & 0 & 1 & 2 \\ 1 & 3 & 2 & 1 & 0 & 2 \end{pmatrix}.$$

Example A.11. *A linear* $(2,4,9,4)$*-AONT:*

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 2 \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 & 2 & 1 \\ 1 & 0 & 1 & 2 & 0 & 1 & 2 & 3 & 1 \\ 1 & 0 & 1 & 2 & 1 & 0 & 3 & 1 & 2 \\ 1 & 1 & 0 & 3 & 0 & 2 & 1 & 1 & 1 \\ 1 & 1 & 2 & 0 & 3 & 3 & 2 & 1 & 3 \\ 1 & 2 & 0 & 3 & 1 & 0 & 1 & 3 & 2 \\ 1 & 2 & 3 & 0 & 2 & 1 & 1 & 3 & 3 \end{pmatrix}.$$

Example A.12. *A linear* $(2,5,11,4)$*-AONT:*

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 2 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 2 & 3 \\ 1 & 0 & 1 & 2 & 0 & 1 & 0 & 1 & 2 & 3 & 1 \\ 1 & 0 & 1 & 2 & 1 & 0 & 1 & 1 & 3 & 2 & 1 \\ 1 & 1 & 0 & 3 & 0 & 1 & 1 & 2 & 3 & 1 & 3 \\ 1 & 1 & 2 & 0 & 3 & 0 & 2 & 1 & 3 & 3 & 2 \\ 1 & 2 & 0 & 3 & 3 & 0 & 0 & 1 & 2 & 3 & 3 \\ 1 & 2 & 3 & 0 & 2 & 1 & 0 & 2 & 1 & 3 & 3 \\ 1 & 3 & 2 & 1 & 1 & 2 & 1 & 0 & 3 & 2 & 3 \end{pmatrix}.$$

Example A.13. *A linear* $(2,3,8,5)$*-AONT:*

$$\begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 2 & 2 & 4 \\ 1 & 0 & 1 & 2 & 3 & 2 & 4 & 1 \\ 1 & 0 & 1 & 2 & 4 & 3 & 1 & 2 \\ 1 & 1 & 0 & 3 & 2 & 1 & 3 & 4 \\ 1 & 1 & 0 & 3 & 4 & 4 & 2 & 2 \\ 1 & 2 & 3 & 0 & 1 & 3 & 2 & 1 \\ 1 & 2 & 3 & 0 & 1 & 4 & 1 & 4 \end{pmatrix}.$$

Example A.14. *A linear* $(2,4,10,5)$*-AONT:*

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 4 & 1 & 0 & 1 & 2 \\ 1 & 0 & 1 & 2 & 3 & 1 & 0 & 1 & 4 & 2 \\ 1 & 0 & 1 & 2 & 3 & 4 & 1 & 0 & 2 & 1 \\ 1 & 1 & 0 & 3 & 4 & 2 & 0 & 1 & 2 & 2 \\ 1 & 1 & 0 & 3 & 4 & 3 & 2 & 2 & 1 & 4 \\ 1 & 2 & 3 & 0 & 1 & 1 & 1 & 4 & 3 & 4 \\ 1 & 2 & 3 & 0 & 1 & 4 & 4 & 3 & 1 & 2 \\ 1 & 3 & 4 & 1 & 0 & 2 & 2 & 2 & 2 & 3 \end{pmatrix}$$

Example A.15. *A linear* $(2, 3, 8, 7)$*-AONT:*

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 2 & 0 & 1 & 2 & 3 \\ 1 & 1 & 0 & 3 & 1 & 1 & 2 & 4 \\ 1 & 2 & 3 & 5 & 3 & 0 & 1 & 3 \\ 1 & 3 & 4 & 6 & 6 & 0 & 1 & 2 \\ 1 & 4 & 5 & 0 & 5 & 2 & 5 & 1 \\ 1 & 5 & 6 & 4 & 2 & 3 & 0 & 1 \end{pmatrix}.$$