5-2018

Covering Arrays for Equivalence Classes of Words

Joshua Cassels

Anant Godbole
East Tennessee State University

Covering Arrays for Equivalence Classes of Words

Joshua Cassels and Anant Godbole

East Tennessee State University

casselsj@etsu.edu; godbolea@etsu.edu

April 12, 2018

Abstract

Covering arrays for words of length t over a d letter alphabet are $k \times n$ arrays with entries from the alphabet so that for each choice of t columns, each of the d^t t-letter words appears at least once among the rows of the selected columns. We study two schemes in which all words are not considered to be different. In the first case, words are equivalent if they induce the same partition of a t element set. In the second case, words of the same weighted sum are equivalent. In both cases we produce logarithmic upper bounds on the minimum size $k = k(n)$ of a covering array. Most definitive results are for $t = 2, 3, 4$.

1 Introduction

Covering arrays for words of length t over a d letter alphabet are $n \times k$ arrays with entries from the alphabet so that for each choice of t columns, each of the d^t t-letter words appears at least once among the rows of the selected columns. A definitive survey of the field is the one by [3]. A central question in the area is the following: given n, t, and d what is the minimum number $k_0 = k_0(n, t, d)$ of rows so that a $n \times k$ covering array exists? In papers such as [10], [14], the focus was on asymptotics, i.e., finding bounds on $k_0(k, t, d)$ as $n \to \infty$ with t, d being held fixed. For example, the thesis of Roux [**?**], cited in [14] exhibited the fact that for $d = 2$ and $t = 3$, we have

$$k_0(n, 3, 2) \le 7.56 \lg n(1 + o(1)),$$

where lg denotes $\log_2$. In [10], the authors used the Lovász local lemma [1] (denoted here by L^3) to yield the general upper bound

$$k_0(k, t, q) \le (t-1)\frac{\lg n}{\lg\left(\frac{q^t}{q^t-1}\right)}(1 + o(1)),$$

which only yields the bound $10.33 \lg n$ for $t = 3, q = 2$. Borrowing Roux's technique of randomly assigning an equal number of ones and zeros to the n columns, the authors of [10] were then able to match the bound $7.56 \lg n$, also via L^3.

There have been several efforts to improve the bounds from [10] for general values of the parameters. In [6], a technique was used that was intermediate between (i) a straightforward use of the L^3 with nk independent uniform random variables determining the array; and (ii) L^3 in conjunction equal weight columns. Specifically, in [6], columns were tiled with small segments that had equal numbers of each letter of the alphabet. In [15], an effort was made to stick with equal weight columns and conquer the more complicated sums that arose for values of the parameters other than $t = 3, q = 2$. The algorithmic use of the L^3, via a method called *entropy compression*, was adopted in the paper [8]. Almost at the same time, the authors of [12] used alteration to give an improvement of an elementary bound (that uses linearity of expectation) that led to a two-stage construction algorithm. Bounds from the L^3 were improved upon in a different manner in [12], by examining group actions on the set of symbols.

There have been several variations on the basic definition of covering arrays. In [4], and [5], the authors considered the notion of covering arrays of permutations. In [2] and [7], *partial covering arrays* are related to an Erdős-Ko-Rado property. Partial covering arrays are also studied extensively in [13]. In the statistically relevant paper [9], only consecutive sets of t columns are considered. The paper [11] is just one of many in which variable strength covering arrays (where the interactions to be covered in the array are studied by modeling them as facets of an abstract simplicial complex); covering arrays on graphs; and mixed covering arrays (different alphabet sets in different columns). See also the contributed talks in the sessions on Generalizations of Covering Arrays at `https://canadam.math.ca/2011/program/schedule_contributed_mini`

In this paper, we offer two more variations on the definition of covering arrays, and find upper bounds on the size of these arrays using some of the techniques mentioned above. In particular, the L^3, either with or without fixed weight columns, will continue to be used in this paper, together with techniques from [6] and [15]. It would be interesting to see what improvements can be made using entropy compression, or group actions, etc. In both of our schemes, all words are not considered to be different. In the first case words are equivalent if they induce the same partition of a t element set. In the second case, words of the same weight are equivalent. In both cases we produce logarithmic upper bounds on the minimum size $n = n(k)$ of a covering array. Most definitive results are for $t = 2, 3, 4$.

2 Covering Arrays for Set Partitions

This section will focus on covering arrays for set partitions. The basic definition is as follows, where $B(t)$ denote the unordered Bell numbers, namely the number of partitions of a t-element set into an arbitrary number of parts.

Definition 2.1. *An $k \times n$ array with entries from the alphabet $\{1, 2, \ldots, d\}$ is a covering array for partitions of a set into t or fewer parts if for each choice of t columns each of the $B(t)$ partitions of $[t]$ appears as a word (or word pattern) across the rows of the selected columns.*

Given n, t, and d what is the minimum number $k_0 = k_0(n, t, d)$ of rows so that a $k \times n$ covering array exists for set partitions? This is the key question that we will address in this section. The minimum value of this k_0 can be found manually for small n, which was our first step. The following constructions (which we call t-scattering arrays, where each equivalence class of set partitions can be found) show that for $n = 4$, $d = 2$, and $t = 3$ we need only 5 rows in order to find all partitions (note, with $d = 2$, there are only 5 partitions to find) and for $n = 5$, $d = 2$, and $t = 3$ we only need 7 rows.

A	B	C	D
1	1	1	1
1	2	3	4
1	2	1	2
2	2	1	1
1	2	2	1

Table 1: t-scattering array for $n = 4$, $t = 3$, and $d = 2$

ABC	ABD	ACD	BCD
xyz	xyz	xyz	xyz
x|y|z	x|y|z	x|y|z	x|y|z
y|xz	x|yz	z|xy	y|xz
z|xy	z|xy	x|yz	x|yz
x|yz	y|xz	y|xz	z|xy

Table 2: Verification of Partition Occurrence

A	B	C	D	E
1	1	1	1	1
1	2	3	4	5
1	2	2	1	2
2	1	2	1	2
2	2	1	1	2
2	2	1	1	1
1	1	1	2	2

Table 3: t-scattering array for $n = 5$, $t = 3$ and $d = 2$

ABC	ABD	ABE	ACD	ACE	ADE	BCD	BCE	BDE	CDE
xyz	xyz	xyz	xyz	xyz	xyz	xyz	xyz	xyz	xyz
x|y|z	x|y|z	x|y|z	x|y|z	x|y|z	x|y|z	x|y|z	x|y|z	x|y|z	x|y|z
x|yz	y|xz	x|yz	y|xz	x|yz	z|xy	z|xy	xyz	y|xz	y|xz
y|xz	x|yz	y|xz	xyz	xyz	y|xz	y|xz	x|yz	x|xy	y|xz
z|xy	z|xy	xyz	z|xy	y|xz	y|xz	x|yz	y|xz	y|xz	z|xy
—	—	z|xy	x|yz	x|yz	x|yz	—	x|yz	x|yz	xyz
—	—	—	—	z|xy	—	—	z|xy	—	x|yz

Table 4: Verification of Partition Occurrence

As before, we will seek bounds on $k_0(n,t,d)$ as d and t are fixed, but $n \to \infty$; at times we allow $d \to \infty$ as well. The first proposition (among other results) illustrates the role that d plays; in particular d may be (far) larger than the size t of the set we are trying to partition.

Proposition 2.1. $k_0(n,2,n) = 2$.

Proof. We consider the case for a $k \times n$ scattering array for words of length $t = 3$. There are only two $B(2) = 2$ partitions to find. We fill the first row of our array with all like elements and the second row with the elements 1 through n. Hence, any random choice of two columns will always result in the desired (xy) and $(x)(y)$ sets. □

Lemma 2.2. *No more than 2 elements are needed to optimize* $k_0(n,3,2)$.

Proof. There are $\binom{d}{2}$ ways to form the 3 unique partitions we are looking for, where d is the size of our alphabet. Each of these partitions will appear twice, as $001 = 110$ and so on; thus, there are $\binom{d}{2} \cdot 2 = d(d-1)$ ways to choose a combination of two elements. There are d^3 unique words of length 3 and therefore a probability of $(1-\frac{d(d-1)}{d^3})$ that one of our k_0 rows will not contain one of our partitions. For $d=2$, $(1-\frac{d(d-1)}{d^3})=\frac{3}{4}$. We wish to see if $(1-\frac{d(d-1)}{d^3})>\frac{3}{4}$ for $d>2$.

$$\begin{aligned}
&1-\frac{d(d-1)}{d^3}>\frac{3}{4} \text{ if}\\
&1-\frac{(d-1)}{d^2}>\frac{3}{4} \text{ if}\\
&\frac{d-1}{d^2}<\frac{1}{4} \text{ if}\\
&d^2-4d+4>0 \text{ if}\\
&(d-2)^2>0, \text{ which is always true.}
\end{aligned} \tag{1}$$

Thus to optimize $k_0(n,3,2)$ it is sufficient to use a two-letter alphabet. $\square$

Proposition 2.3. $k_0(n,3,2)\leq 7.23\lg(n)$.

Proof. We consider the case for a $k\times n$ scattering array for words of length $t=3$. From the Bell numbers, $B(3)=5$. We satisfy the partitioning of n elements into exclusively unique sets by filling the first row with $1,2,3,...n$. We can also satisfy the partitioning n elements into the same set by filling our second row with all like elements. Our remaining partitions are xxy, xyx and yxx. Note; we are only interested in finding word patterns. As per Lemma 2.2, we only need to find one instance of these partitions and all of these partitions can be found using only a $d=2$ alphabet.

Let X be defined as the number of 'bad' columns, where 'bad' implies at least one of our word equivalence classes is missing from any random choice of three columns. We wish for the expected value of X, $E(X)$, to be less than one, as this would imply that $P(X=0)>0$.

There are $\binom{n}{3}$ choices of columns in our array. There are three unique partitions and $2^3 = 8$ total words. Two of these are accounted for in the first two rows of our matrix, so we need only find 6. There are $\binom{d}{2} = \binom{2}{2} = 1$ ways to find these partitions, but as each of our partitions has one equivalent representation, this number would be doubled to allow for equivalency. Thus, the probability that one of our partitions will be missing from a random choice of three columns is given by $(1-(\frac{2}{8}))^{k_0} = \left(\frac{3}{4}\right)^{k_0}$. Therefore, $E(X)$ is given by

$$E(X) = \binom{n}{3} \cdot 3 \cdot (\frac{3}{4})^{k_0}$$

.

We wish for this value to be less than 1, therefore

$$\begin{aligned}
\binom{n}{3} \cdot 3 \cdot (\frac{3}{4})^{k_0} &< 1 \text{ if} \\
\frac{n^3}{6} \cdot 3 \cdot (\frac{3}{4})^{k_0} &< 1 \text{ if} \qquad (2) \\
\lg[\frac{n^3}{6} \cdot 3 \cdot (\frac{3}{4})^{k_0}] &< \lg(1) \text{ if}
\end{aligned}$$

$$\begin{aligned}
3\lg(n) + \lg(\frac{3}{6}) - k_0 \cdot \lg(\frac{4}{3}) &< 0 \text{ if} \\
\frac{3\lg(n) + \lg(\frac{1}{2})}{\lg(\frac{4}{3})} &< k_0 \text{ if} \\
\frac{3\lg(n)(1 + \frac{(\lg \frac{1}{2})}{3\lg(n)})}{\lg \frac{4}{3}} &< k_0 \text{ if} \qquad (3) \\
\frac{3\lg(n)(1+o(1))}{\lg \frac{4}{3}} &< k_0 \text{ if} \\
7.23\lg(n)(1+o(1)) &< k_0
\end{aligned}$$

Hence, it is possible that there are no 'bad' columns if $k_0(n,3,2) \leq 7.23\lg(n)$.

$\square$

Proposition 2.4. $k_0(n,4,3) \leq 36.025\lg(n)$.

Proof. We consider the case for a $k \times n$ scattering array for words of length $t = 4$. From the Bell numbers, $B(4) = 15$. As before, we satisfy the partitioning of n elements into exclusively unique sets by filling the first row with $1, 2, 3, ...n$. We can also satisfy the partitioning n elements into the same set by filling our second row with all like elements. Thus we are left with 13 partitions to find.

Let X be defined as the number of 'bad' columns, where 'bad' implies at least one of our word equivalence classes is missing from any random choice of four columns. We wish for the expected value of X, $E(X)$ to be less than one, as this would imply that $P(X = 0) > 0$.

There are $\binom{n}{4}$ choices of columns in our array.

From inspection, we can see that the bell number $B(n) = \sum_{i=1}^{n} S(n,i)$, where $S(n,i)$ is the Stirling number of the second kind; i.e. the number of ways of obtaining groups of i elements from a set of n. For $S(4,2)$ and $d = 3$ there are $\binom{3}{1} = 3$ ways to chose the element for the first set and thus $\binom{2}{1} = 2$ ways to chose the element for the second set. Thus a total number of 6 ways of constructing each partition.

For $S(4,3)$ and $d = 3$, there are $\binom{3}{1} = 3$ ways to chose the element for the first set and thus $\binom{2}{1} = 2$ ways to chose the element for the second set and $\binom{1}{1} = 1$ ways to choose the element for the third set. Thus, as before, there are 6 ways of constructing each of these partitions. There are $3^4 = 81$ total words and thus the probability that any one of those partitions is missing is given by $(1-\frac{6}{81})^{k_0} = \left(\frac{75}{81}\right)^{k_0}$. Therefore, $E(X)$ is given by

$$E(X) = \binom{n}{4} \cdot 13 \cdot (\frac{75}{81})^{k_0}$$

We wish for this value to be less than 1, therefore

$$\begin{aligned}
&\binom{n}{4} \cdot 13 \cdot (\frac{75}{81})^{k_0} < 1 \text{ if} \\
&= \frac{n^4}{24} \cdot 13 \cdot (\frac{75}{81})^{k_0} < 1 \text{ if} \\
&= \lg[\frac{n^4}{24} \cdot 13 \cdot (\frac{75}{81})^{k_0}] < \lg(1) \text{ if} \\
&= 4\lg(n) + \lg(\frac{13}{24}) - k_0 \cdot \lg(\frac{81}{75}) < 0 \text{ if} \\
&= \frac{4\lg(n) + \lg(\frac{13}{24})}{\lg(\frac{81}{75})} < k_0 \text{ if} \\
&= \frac{4\lg(n)(1 + \frac{\lg(\frac{13}{24})}{4\lg(n)})}{\lg(\frac{81}{75})} < k_0 \text{ if} \\
&= \frac{4\lg(n)(1+o(1))}{\lg(\frac{81}{75})} < k_0 \text{ if} \\
&= 36.036\lg(n)(1+o(1)) < k_0
\end{aligned} \tag{4}$$

Hence, it is possible that there are no 'bad' columns if $k_0(n, 4, 3) \leq 36.036\lg(n)$.

□

Lemma 2.5. *Lovász Local Lemma [1]*

Let $A_1, A_2, \ldots A_n$ be events in an arbitrary probability space. Suppose that each event A_i is mutually independent of a set of all the other events A_j but at most d and that $Pr(A_i) \leq p$ for all $1 \leq i \leq n$ if

$$ep(d+1) \leq 1,$$

then $Pr(\wedge_{i=1}^{n} \bar{A}_i) > 0$.

Here, 'e' is Euler's irrational number, namely $e \approx 2.71828$, 'p' is the probability that at least one of our words is missing from a choice of t, and we let the dependence number $m = d + 1$, which is the number of t columns that are dependent on a fixed column. Where before we wanted our $E(X) < 1$, now we want our $epd < 1$.

Proposition 2.6. *$k_0(n, 3, 2) \leq 4.8188 \lg(n)$.*

Proof. We wish to make an improvement on our previous bound for $k_0(n, 3, 2)$ by employing the Lovász Local Lemma.

We consider the case for a $k \times n$ scattering array for words of length $t = 3$. From the Bell numbers, $B(3) = 5$. We satisfy the partitioning of n elements into exclusively unique sets by filling the first row with $1, 2, 3 ... n$. We can also satisfy the partitioning n elements into the same set by filling our second row with all like elements, leaving us with 3 partitions to find.

From Lovász, we wish for our $e \cdot p \cdot m < 1$ where $e \approx 2.71828$, p is the probability that any one of our partition sets is missing from a choice of 3 columns and m is the dependence number.

The probability that any one of our partition sets is missing from a choice of t columns was found in Proposition 2.1 to be $3 \cdot (\frac{3}{4})^{k_0}$. For our m value, we chose any one from of our set of three columns to be part of the intersection. We then choose two columns from the remaining $n - 3$ columns to fill the pair. Conversely, we can choose two columns from our set of three and one more from the remaining $n-3$ columns. This gives us

$$\begin{aligned}
m &= \binom{3}{1}\binom{n-3}{2} + \binom{3}{2}\binom{n-3}{1} \\
&= \frac{3!}{1!2!}\frac{(n-3)!}{2!(n-5)!} + \frac{3!}{2!1!}\frac{(n-3)!}{1!(n-4)!} \\
&= \frac{3(n-3)(n-4)}{2} + 3(n-3) \\
&= \frac{3(n^2-7n+12)}{2} + \frac{6n}{2} - \frac{18}{2} \\
&= \frac{3n^2}{2} - \frac{n}{2} - \frac{6}{2} \\
&\leq \frac{3n^2}{2}. \text{ Thus,}
\end{aligned}$$

$$e \cdot p \cdot m < 1 \text{ if}$$

$$e \cdot 3 \cdot (\frac{3}{4})^{k_0} \cdot \frac{3n^2}{2} < 1.$$

Employing logarithms, we have

$$\begin{aligned}
&\lg[e \cdot 3 \cdot (\frac{3}{4})^{k_0} \cdot \frac{3n^2}{2} < 1] \text{ if} \\
&\lg(e) + 2\lg(3n) + \lg(\frac{3}{2}) - k_0 \lg(\frac{4}{3}) < 0 \text{ if} \\
&2\lg(3n) + \lg(\frac{3}{2}) < k_0 \lg(\frac{4}{3}).
\end{aligned}$$

Rearranging in terms of k_0, we have

$$\begin{aligned}
&\frac{2\lg(3n) + \lg(\frac{3}{2})}{\lg(\frac{4}{3})} < k_0 \text{ if} \\
&\frac{2\lg(3n)(1 + \frac{\lg(\frac{3}{2})}{2\lg(3n)})}{\lg(\frac{4}{3})} < k_0 \text{ if} \\
&\frac{2\lg(3n)(1+o(1))}{\lg(\frac{4}{3})} < k_0 \text{ if} \\
&\frac{2\lg(3n)}{\lg(\frac{4}{3})} = \frac{2\lg(3) + 2\lg(n)}{\lg(\frac{4}{3})} < \frac{2\lg(n)}{\lg\frac{4}{3}} < k_0 \text{ if} \\
&4.8188\lg(n) < k_0. \qquad (5)
\end{aligned}$$

Thus, $k_0(n,3,2) \leq 4.8188 \lg(n)$. $\square$

Proposition 2.7. $k_0(n,4,3) \leq 27.019 \lg(n)$

Proof. We wish to make an improvement on our previous bound for $k_0(n,4,3)$ by employing the Lovász Local Lemma.

We consider the case for a $k \times n$ scattering array for words of length $t = 4$. From the Bell numbers, $B(4) = 15$. We satisfy the partitioning of n elements into exclusively unique sets by filling the first row with $1, 2, 3, ...n$. We can also satisfy the partitioning n elements into the same set by filling our second row with all like elements, leaving us with 13 partitions to find.

From Lovasz, we wish for our $e \cdot p \cdot m < 1$ where $e \approx 2.71828$, p is the probability that any one of our partition sets is missing from a choice of 4 columns and m is the dependence number.

The probability that any one of our partition sets is missing from a choice of t columns was found in Proposition 2.2 to be $13 \cdot (\frac{75}{81})^{k_0}$. For our m value, we chose any one from of our set of four columns to be part of the intersection. We then choose two columns from the remaining $n-4$ columns to fill the pair. Conversely, we can choose two columns from our set of four to be part of the intersection and two from the remaining $n-4$ columns, or choose 3 columns from our set of four and one from the remaining $n-4$ columns. This gives us

$$\begin{aligned} m &= \binom{4}{1}\binom{(n-4)}{3} + \binom{4}{2}\binom{(n-4)}{2} + \binom{4}{3}\binom{(n-4)}{1} \\ &= \frac{4!}{1!3!}\frac{(n-4)!}{3!(n-7!)} + \frac{4!}{2!2!}\frac{(n-4)!}{2!(n-6)!} + \frac{4!}{3!1!}\frac{(n-4)!}{1!(n-5!)} \\ &= \frac{4}{6}(n-4)(n-5)(n-6) + \frac{18}{6}(n-4)(n-5) + \frac{24}{6}(n-4) \end{aligned} \tag{6}$$

$$= \frac{4}{6}(n^3 - 15n^2 + 74n - 120) + \frac{18}{6}(n^2 - 9n + 20) + \frac{24}{6}(n-4)$$
$$= \frac{4}{6}n^3 - \frac{42}{6}n^2 + \frac{158}{6}n + \frac{936}{6} \tag{7}$$
$$= m \leq \frac{4n^3}{6}.$$

Therefore,

$$e \cdot p \cdot m < 1 \text{ if}$$
$$e \cdot \frac{4n^3}{6} \cdot 13(\frac{75}{81})^{k_0} < 1.$$

Employing logarithms, we have

$$lg[e \cdot \frac{4n^3}{6} \cdot 13(\frac{75}{81})^{k_0} < 1] \text{ if}$$
$$lg(e) + lg\frac{4n^3}{6} + lg(13) + k_0 lg(\frac{75}{81}) < 0 \text{ if}$$
$$3lg(4n) - k_0 lg(\frac{81}{75}) < 0.$$

Rearranging in terms of k_0, we have

$$\frac{3lg(4) + 3lg(n)}{lg\frac{81}{75}} < k_0 \text{ if}$$
$$\frac{3lg(n)(1 + o(1))}{lg\frac{81}{75}} < k_0 \text{ if}$$
$$27.1953lg(n) < k_0. \tag{8}$$

Hence, it is possible that there are no 'bad' columns if $k_0(n, 4, 3) \leq 27.019 \lg(n)$.

□

3 Covering Arrays for Weight-Equivalent Words

This section will focus on covering arrays for words when words with the same weight are equivalent, and we only need to find a single word of a given weight.

Definition 3.1. *An $k \times n$ array with entries from the alphabet $\{1, 2, \ldots, d\}$ is a covering array for weight-equivalent words of length t over $[d]$ if for each choice of t columns a word of each weight between t and dt appears at least once across the rows of the selected columns.*

Given n, t, and d what is the minimum number $k_w = k_w(n, t, d)$ of rows so that a $k \times n$ covering array exists for weight-equivalent words? This is the key question that we will address in this section. The methods of finding these bounds are very similar to the propositions in Section 2, except now instead of finding equivalent partitions (i.e. $110 = 001$) we're looking for partitions of equal weight (i.e. $100 = 010 = 001$).

Proposition 3.1. $k_w(n, 3, 2) \leq 2.95 \lg(n)$.

Proof. In Section 2, we filled our first two rows with like and unique elements respectively to account for the individual partitions of every elements in the same set and every element in its own set. Now, using a two-letter $(0, 1)$ alphabet, we fill the first row with 0s to allow for words of weight 0 and the second row with 1s to allow for words of weight 3. Thus, we need only find two different words; a word of weight 1, and a word of weight 2. There are three ways to find each word, namely

$$100 = 010 = 100,$$

$$110 = 101 = 011.$$

and $d^t = 2^3 = 8$ total possible words. Therefore, the probability that one of our two words will be missing from a random choice of 3 columns is given by $(1 - \frac{3}{8})^{k_w} = (\frac{5}{8})^{k_w}$. Employing Lovasz, with $m = n^2$ we find that our expected value is given by

$$\begin{aligned}
&e \cdot n^2 \cdot 2(\frac{5}{8})^{k_w} < 1 \text{ if} \\
&lg[e \cdot n^2 \cdot 2(\frac{5}{8})^{k_w}] < lg(1) \text{ if} \\
&lg(e) \cdot lg(n^2) \cdot lg((\frac{5}{8})^{k_w}) < 0 \text{ if} \\
&2lg(n) - k_w lg(\frac{8}{5}) < 0 \text{ if} \\
&\frac{2lg(n)}{lg(\frac{8}{5})} < k_w \text{ if} \\
&2.95lg(n) < k_w.
\end{aligned} \tag{9}$$

□

Hence, it is possible that there are no 'bad' columns if $k_w(n, 3, 2) \leq 2.95 \lg(n)$.

Proposition 3.2. *$k_w(n, 4, 2) \leq 7.23 \lg(n)$.*

Proof. Using a two-letter $(0, 1)$ alphabet, we fill the first row with 0s to allow for words of weight 0 and the second row with 1s to allow for words of weight 4. Thus, we need find four different words; a word of weight 1, weight 2, and weight 3. There are four ways to find a word of weight one, six ways to find a word of weight two, and four ways to find a word of weight 3, namely

$$1000 = 0010 = 0100 = 1000,$$

$$0011 = 0110 = 1100 = 1001 = 0110 = 1010 = 0101,$$

$$0111 = 1011 = 1101 = 1110.$$

There are $d^t = 2^4 = 16$ total possible words. Therefore, the probability that one of our three words will be missing from a random choice of 4 columns is less than, or equal to $(1 - \frac{4}{12})^{k_w} = (\frac{12}{16})^{k_w}$. Employing Lovász, with $m = n^3$ we find

$$\begin{aligned}
&e \cdot n^3 \cdot 2(\frac{12}{16})^{k_w} < 1 \text{ if} \\
&lg[e \cdot n^3 \cdot 2(\frac{3}{4})^{k_w}] < lg(1) \text{ if} \\
&lg(e) \cdot lg(n^3) \cdot lg((\frac{3}{4})^{k_w}) < 0 \text{ if} \\
&3lg(n) - k_w lg(\frac{4}{3}) < 0 \text{ if} \\
&\frac{3lg(n)}{lg(\frac{4}{3})} < k_w \text{ if} \\
&7.23lg(n) < k_w.
\end{aligned} \tag{10}$$

□

Hence, it is possible that there are no 'bad' columns if $k_w(n, 4, 2) \leq 7.23 \lg(n)$.

Proposition 3.3. $k_w(n, 3, 3) \leq 11.77 \lg(n)$.

Proof. Using a three-letter $(0, 1, 2)$ alphabet, we fill the first row with 0s to allow for words of weight 0 and the second row with 2s to allow for words of weight 6. Thus, we need find five different words; a word of weight 1, weight 2, and weight 3, weight 4 and weight 5. There are three ways to find a word of weight one, six ways to find a word of weight two, seven ways to find a word of weight 3, six ways to find a word of weight 4, and three ways to find a word of weight 5, namely

$$100 = 010 = 001,$$

$$110 = 101 = 011 = 200 = 020 = 002,$$

$$111 = 120 = 102 = 012 = 210 = 021 = 201,$$

$$112 = 121 = 211 = 022 = 202 = 220,$$

$$122 = 212 = 221.$$

There are $d^t = 3^3 = 27$ total possible words. Therefore, the probability that one of our five words will be missing from a random choice of 3 columns is less than, or equal to $(1 - \frac{3}{27})^{k_w} = (\frac{24}{27})^{k_w}$. Employing Lovász, with $m = n^2$ we find that our expected value is given by

$$\begin{aligned}
E(X) &= e \cdot n^2 \cdot 2(\frac{24}{27})^{k_w} < 1 \\
&= lg[e \cdot n^2 \cdot 2(\frac{24}{27})^{k_w}] < lg(1) \\
&= lg(e) \cdot lg(n^2) \cdot lg((\frac{24}{27})^{k_w}) < 0 \\
&= 2lg(n) - k_w lg(\frac{27}{24}) < 0 \\
&= \frac{2lg(n)}{lg(\frac{27}{24})} < 0 \\
&= 11.77lg(n) < 0.
\end{aligned} \tag{11}$$

□

Hence, it is possible that there are no 'bad' columns if $k_w(n, 3, 3) \leq 11.77 \lg(n)$.

4 Open Questions

(i) What are some exact values that one might find via constructions?

(ii) Why do fixed weight columns appear to do no better in some cases, but play a critical role in improvements in other cases?

(iii) What are some applications of our schema, beyond those noted in the beginning of Section 3? What other equivalence classes of words might we consider?

References

[1] N. Alon and J. Spencer (1992). *The Probabilistic Method.* Wiley, New York.

[2] P. Carey and A. Godbole (2010). “Partial covering arrays and a generalized Erdös-Ko-Rado property," *J. Combinatorial Designs* **18**, 155–166.

[3] C. Colbourn (2004), “Combinatorial aspects of covering arrays," *Le Matematiche (Catania)* **58**, 121–167.

[4] Y. M. Chee, C. Colbourn, D. Horsley, and J. Zhou (2013). “Sequence covering arrays", *SIAM Journal on Discrete Mathematics* **27**, 1844–1861.

[5] , S. deGraaf, A. Godbole, Z. Koch, and K. Lan (2017+). “t-scrambling permutations and t-covering arrays," Preprint.

[6] M. Donders and A. Godbole (2013). “t-covering arrays generated by a tiling probability model," *Congressus Numerantium* **218**, 111– 116.

[7] P. Erdős, P. Frankl, and Z. Furedi (1982). "Families of finite sets in which no set is covered by the union of two others," *J. Combin. Theory Ser. A* **33**, 158–166.

[8] N. Francetić and B. Stevens (2017). "Asymptotic size of covering arrays: an application of entropy compression," *J. Combinatorial Designs* **25**, 243–257.

[9] A. Godbole, M. Koutras, and F. Milienos (2011). "Binary consecutive covering arrays," *Annals of the Institute of Statistical Mathematics* **63**, 559–584.

[10] A. Godbole, D. Skipper, and R. Sunley (1996). "t-covering arrays: upper bounds and Poisson approximations," *Combinatorics, Probability and Computing* **5**, 105–118.

[11] S. Raaphorst, L. Moura, B. Stevens (2017+), "Variable strength covering arrays," To appear.

[12] K. Sarkar and C. Colbourn (2017). "Upper bounds on the size of covering arrays", *SIAM Journal on Discrete Mathematics* **31**, 1277–1293.

[13] K. Sarkar, C. Colbourn, A. De Bonis, and U. Vaccaro (2017+). "Partial Covering Arrays: Algorithms and Asymptotics," *Theory of Computing Systems*, to appear.

[14] Sloane, N. J. A. (1993). "Covering arrays and intersecting codes," *Journal of Combinatorial Designs* **1**, 51–63.

[15] R. Yuan, Z. Koch, and A. Godbole (2015). “Covering array bounds using analytical techniques," *Congressus Numerantium* **222**, 65–73.