

# Six New Constant Weight Binary Codes

*Hui Kheng Aw*  
Kimberly-Clark Corporation  
2100 Winchester Road  
Neenah, Wisconsin 54956  
USA

*Yeow Meng Chee*  
12 Jambol Place  
Singapore 119339  
ym@ymchee.com

*Alan C. H. Ling*  
Department of Computer Science  
University of Vermont  
Burlington, Vermont 05405  
USA

## Abstract

We give six improved bounds on  $A(n, d, w)$ , the maximum cardinality of a binary code of length  $n$  with minimum distance  $d$  and constant weight  $w$ .

## 1 Introduction

A *binary code of length  $n$*  is any set  $C \subseteq \{0, 1\}^n$ . The elements of  $C$  are called *codewords*.  $C$  is said to have *minimum distance  $d$*  and *constant weight  $w$*  if the Hamming distance between any two distinct codewords is at least  $d$  and  $\|u\|^2 = w$  for all  $u \in C$ . For simplicity, we refer to a binary code of length  $n$ , minimum distance  $d$ , and constant weight  $w$  as an  $(n, d, w)$ -code. We also assume without loss of generality that  $d$  is even. Define  $A(n, d, w)$  to be the maximum cardinality of an  $(n, d, w)$ -code, that is,

$$A(n, d, w) = \max\{|C| : C \text{ is an } (n, d, w)\text{-code}\}.$$

The function  $A(n, d, w)$  is fundamental in the theory of error-correcting codes [2]. Unfortunately, the exact determination of  $A(n, d, w)$  is difficult. Most efforts have therefore focused on establishing good bounds for  $A(n, d, w)$ . The function  $A(n, d, w)$  is also widely studied in combinatorial design theory, under the guise of packing designs [3].

In this paper, we give some improved lower bounds on  $A(n, d, w)$ .

## 2 Results

### 2.1 A Cyclic (30, 8, 5)-Code

The set of all distinct cyclic shifts of the two vectors

$$\begin{aligned} &100000100000100000100000100000 \\ &110100000010001000000000000000 \end{aligned}$$

is a (30, 8, 5)-code with 36 codewords.

### 2.2 Length-Reduction Heuristic

We represent a binary code  $\mathcal{C}$  by a  $\{0, 1\}$ -matrix  $M(\mathcal{C})$  whose columns are the codewords of  $\mathcal{C}$ . Let  $\mathcal{M}_{n,m}(d, w)$  be the set of all  $n \times m$   $\{0, 1\}$ -matrices  $M$  with constant column sum  $w$ , such that the Hamming distance between any two distinct columns  $\mathbf{u}$  and  $\mathbf{v}$  of  $M$  is at least  $d$ . So for an  $(n, d, w)$ -code  $\mathcal{C}$ , we have  $M(\mathcal{C}) \in \mathcal{M}_{n,|\mathcal{C}|}(d, w)$ . For a positive integer  $i$ ,  $M \in \{0, 1\}^{n \times m}$ , and  $\mathbf{u} \in \{0, 1\}^n$ , we denote by  $M_i(\mathbf{u})$  the matrix obtained by replacing the  $i$ th column of  $M$  by  $\mathbf{u}$ . We also denote by  $\tilde{M}$  the matrix obtained from  $M$  by deleting its last row.

The **length-reduction heuristic** works as follows. The inputs are  $n$ ,  $m$ ,  $d$ , and  $w$ , where  $n \geq w \geq d/2$ . We begin with  $M \in \mathcal{M}_{N,m}(d, w)$ , for some  $N$ . At each stage of the heuristic, we generate a random integer  $i$  and a random element  $\mathbf{u} \in \{0, 1\}^n$  whose last component is zero. If  $M_i(\mathbf{u}) \in \mathcal{M}_{n,m}(d, w)$ , then we replace  $M$  by  $M_i(\mathbf{u})$ . It could happen at this point that the last row of  $M$  is a zero vector. If this is the case, we replace  $M$  by  $\tilde{M}$ , and repeat the process. We stop when  $M$  has only  $n$  rows.

Let

$$I_{m,w} = \begin{bmatrix} \mathbf{1}_w & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{1}_w & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{1}_w \end{bmatrix},$$

where  $\mathbf{1}_w$  is the  $w$ -dimensional column vector of all ones. Clearly,  $I_{m,w} \in \mathcal{M}_{mw,m}(d,w)$  for any  $d \leq 2w$ . For our experiments, the initial choice of  $M$  is  $I_{m,w}$ .

### length-reduction heuristic( $n, m, d, w$ )

**Step 1:**  $M = I_{m,w}$  and  $N = mw$ .

**Step 2:** Repeat Step 3 to Step 5 until  $N = n$ .

**Step 3:** Randomly choose  $i \in \{1, 2, \dots, m\}$  and  $\mathbf{u} \in \{0, 1\}^N$ .

**Step 4:** If  $M_i(\mathbf{u}) \in \mathcal{M}_{N,m}(d,w)$ , then  $M = M_i(\mathbf{u})$ .

**Step 5:** If the last row of  $M$  is the zero vector, then  $M = \bar{M}$  and set  $N = N - 1$ .

It is easy to see that when the heuristic terminates, we have  $M$  as the matrix of an  $(n, d, w)$ -code of cardinality  $m$ .

Most algorithms and heuristics for constructing constant weight binary codes attempts to pack as many codewords into an  $(n, d, w)$ -code as possible, given  $n, d$ , and  $w$ . Here, the **length-reduction** heuristic takes the alternative approach of minimizing the length  $n$  of an  $(n, d, w)$ -code, given  $d, w$ , and its cardinality.

## 2.3 New Bounds

The **length-reduction** heuristic has been used to produce five new lower bounds on  $A(n, d, w)$ .

**Theorem 1.**  $A(18, 6, 5) \geq 69$ ,  $A(27, 8, 5) \geq 31$ ,  $A(29, 8, 5) \geq 34$ ,  $A(33, 8, 5) \geq 44$  and  $A(34, 8, 5) \geq 47$ .

*Proof.* The matrices in Appendix A represent the necessary  $(n, d, w)$ -codes for providing the lower bounds on  $A(n, d, w)$ .  $\square$

### 3 Conclusion

The following table summarizes the results obtained in this paper.

$n$	$d$	$w$	best lower bound on $A(n, d, w)$ previously known [1, 4, 5]	lower bound on $A(n, d, w)$ obtained in this paper
18	6	5	68	69
27	8	5	30	31
29	8	5	33	34
30	8	5	33	36
33	8	5	43	44
34	8	5	43	47

### References

- [1] A. E. Brouwer, J. B. Shearer, N. J. A. Sloane, and W. D. Smith, A new table of constant weight codes, *IEEE Trans. Inform. Theory* **36** (1990) 1334–1380.
- [2] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland, Amsterdam 1977.
- [3] W. H. Mills and R. C. Mullin, Coverings and packings, in: *Contemporary Design Theory: A Collection of Surveys* (J. H. Dinitz, D. R. Stinson; eds.) John Wiley & Sons, New York, 1992, 371–399.
- [4] E. M. Rains and N. J. A. Sloane, Table of constant weight binary codes, URL=<http://www.research.att.com/~njas/codes/Andw/>.
- [5] J. Yin and A. M. Assaf, Constructions of optimal packing designs, *J. Combin. Designs* **6** (1998) 245–260.



