Equireplicate Balanced Binary Codes for Oligo Arrays

Noga Alon
Mathematics
Raymond and Beverley Sackler
Faculty of Exact Sciences
Tel Aviv University
Tel Aviv, ISRAEL

Charles J. Colbourn
Computer Science
University of Vermont
Burlington, VT 05405
U.S.A.

Alan C.H. Ling
Mathematics
Michigan Technological University
Houghton, MI 49931
U.S.A.

Martin Tompa
Computer Science & Engineering
University of Washington
Seattle, WA 98195-2350
U.S.A.

Abstract

In the manufacture of oligo arrays for DNA hybridization experiments, manufacturing defects must be detected and their position determined. The design of manufacturing protocols for such oligo arrays leads to a combinatorial problem, requiring certain binary codes which have an additional balance property. Constructions using block designs and packings for these codes, within a range of interest in a practical manufacturing application, are developed. The focus is on equireplicate codes, constant weight codes in which every bit position is a one equally often.

AMS Subject Classification: 05B05.

1 Introduction

Let X be a set of v *elements* or *points*. Let $\mathcal{B}$ be a collection of b subsets of X, called *blocks*. Then $(X, \mathcal{B})$ is a (v, b)*-set system*. The *block sizes* of $(X, \mathcal{B})$ are the cardinalities of the b blocks in $\mathcal{B}$; when all blocks have cardinality k, the set system is k*-uniform*. We often write (v, b, k)*-set system* to denote a k-uniform (v, b)-set system.

In an application to quality control in the manufacture of oligo arrays described in the next section, certain (v, b, k)-set systems are of particular interest. For each point $x \in X$, we define the *replication number* of x to be the number of blocks containing x. The set system is r*-equireplicate* if every point has replication number r. We call a (v, b, k)-set system d*-discriminated* if, for every point $x \in X$, the replication number r_x satisfies $d \leq r_x \leq b - d$; and, for every two distinct points

$x, y \in X$, the number of blocks containing *exactly one* of x and y is at least d. In other words, if λ_{xy} represents the number of blocks containing both x and y, we require that $r_x + r_y - 2\lambda_{xy} \geq d$. A d-discriminated (v, b, k)-set system is henceforth denoted by (v, b, k, d)-*balanced binary code*, or (v, b, k, d)-*bbc* for short.

Table 1 gives an example of a $(28, 14, 10, 5)$-bbc, which is 5-equireplicate. This was constructed using the method described in [3].

0	4	7	9	13	14	18	20	23	27
2	3	4	9	15	16	18	23	25	26
4	6	8	13	15	19	20	21	24	25
0	3	4	5	12	16	19	21	22	27
1	2	6	9	10	13	16	17	21	27
1	5	10	11	13	14	16	19	23	25
3	7	8	9	10	11	12	20	21	25
0	1	2	7	11	15	21	22	23	24
0	2	3	5	8	11	13	17	18	24
0	1	8	9	12	14	15	17	19	26
1	5	6	7	12	18	24	25	26	27
6	11	12	14	15	16	17	18	20	22
3	10	17	19	20	22	23	24	26	27
2	4	5	6	7	8	10	14	22	26

Table 1: A 5-equireplicate (28,14,10,5)-bbc set system

The connection to codes arises as follows. If we form the $b \times v$ incidence matrix of the set system, then each row has weight k and each column has weight at least d and at most $b-d$. Hence each column differs from the all-zero vector and from the all-one vector in at least d positions. Moreover, since two points satisfy $r_x + r_y - 2\lambda_{xy} \geq d$, we have that every two columns have Hamming distance at least d. Hence the code whose words are the columns together with the all-zero and all-one vectors has minimum distance (at least) d. For the example in Table 1, the matrix is given in Table 2.

The fundamental existence question for balanced binary codes is to determine, for a given v and k, a code with a 'small' number b of rows having 'large' discrimination d. (See Section 2 for the motivation.) To make this precise, given v, k, and d, we seek the smallest value of b for which a (v, b, k, d)-bbc exists. We begin by establishing a lower bound on b.

Proposition 1.1 *If a (v, b, k, d)-bbc exists, then* $b \geq \max\left(\left\lceil \frac{vd}{k} \right\rceil, \left\lceil \frac{vd}{v-k} \right\rceil\right)$.

Proof. The incidence matrix of a (v, b, k, d)-bbc contains bk one entries, since each of the b rows contains k ones. Since each of the v columns contains at least d and at most $b - d$ ones, we have:

$$vd \leq bk \leq vb - vd.$$

The bounds follow. □

1000100101000110001010010001
0011100001000001101000010110
0000101010000101000111001100
1001110000001000100101100001
0110001001100100110001000001
0100010000110110100100010100
0001000111111000000011000100
1110000100010001000001111000
1011010010010100011000001000
1100000011001011010100000010
0100011100001000001000001111
0000001000011011111010100000
0001000000100000010110111011
0010111110100010000000100010

Table 2: A 5-equireplicate (28,14,10,5)-bbc incidence matrix

We call a (v, b, k, d)-bbc *optimal* when b realizes the bound in Proposition 1.1. When a (v, b, k, d)-bbc exists, an additional row can easily be appended to form a $(v, b + 1, k, d)$-bbc; in fact, simply duplicating any of the rows produces the extended bbc. It is therefore natural to study the optimal balanced binary codes.

Let $(V, \mathcal{B})$ be a set system. The *complement* of $(V, \mathcal{B})$, denoted by $(V, \overline{\mathcal{B}})$, has the same set V of elements, and the collection of blocks $\overline{\mathcal{B}} = \{V \setminus D : D \in \mathcal{B}\}$.

Lemma 1.2 *The complement of a (v, b, k, d)-bbc is a $(v, b, v - k, d)$-bbc. The complement of an equireplicate bbc is also equireplicate. The complement of an optimal bbc is also optimal.*

The following lemma gives a simple characterization of optimal equireplicate bbc's.

Lemma 1.3 *Suppose B is an equireplicate (v, b, k, d)-bbc with replication number r.*

1. *If $v \geq 2k$, B is optimal if and only if $r = d$.*

2. *If $v < 2k$, B is optimal if and only if $r = b - d$.*

Proof. By Lemma 1.2, assume without loss of generality that $v \geq 2k$. Suppose $r = d$. Since $bk = vr$, both being the number of ones in the incidence matrix of B, we have $b = \frac{vd}{k} = \max\left(\left\lceil \frac{vd}{k} \right\rceil, \left\lceil \frac{vd}{v-k} \right\rceil\right)$, making B optimal. Conversely, suppose B is optimal. Then $b = \left\lceil \frac{vd}{k} \right\rceil \leq \frac{vd+k-1}{k}$. By the definition of discrimination, all replication numbers of B are at least d, so $d \leq r = \frac{bk}{v} \leq d + \frac{k-1}{v} < d + 1$. Since r is integral, $r = d$. $\square$

Sengupta and Tompa [9] observed that if B_1 is a (v, b_1, k, d_1)-bbc and B_2 is a (v, b_2, k, d_2)-bbc, then $\begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$, the union of the blocks of B_1 and B_2, is a $(v, b_1 + b_2, k, d_1 + d_2)$-bbc; we call this operation *addition*. Unfortunately, the addition of two optimal bbc's need not be optimal. The reason is simple. Since the bound in Proposition 1.1 is the next larger integer, it is possible for the addition of B_1 and B_2 to contain one more row than does an optimal bbc, despite the optimality of B_1 and B_2 individually. Nevertheless, the addition proves to be very useful in limiting the ranges of the discrimination to be examined:

Proposition 1.4 *If B_1 is an optimal equireplicate (v, b_1, k, d_1)-bbc and B_2 is an optimal (v, b_2, k, d_2)-bbc, then $\left[\frac{B_1}{B_2}\right]$ is an optimal $(v, b_1 + b_2, k, d_1 + d_2)$-bbc.*

Proof. By Lemma 1.2, assume without loss of generality that $v \geq 2k$. By Lemma 1.3, then, all replication numbers of B_1 are d_1, so $b_1 k = vd_1$. It follows that $b_1 + b_2 = \frac{vd_1}{k} + \lceil \frac{vd_2}{k} \rceil$. But since $\frac{vd_1}{k}$ is an integer, we have $b_1 + b_2 = \lceil \frac{v(d_1+d_2)}{k} \rceil$, so that the addition is optimal. □

For this reason, the critical ingredients in producing optimal balanced binary codes are those that are equireplicate. In this paper, we provide a number of combinatorial constructions for equireplicate optimal bbc's, primarily within a range of practical interest in the study of the manufacture of oligo arrays. In a companion paper [3], we examine heuristic techniques which we have used for the production of optimal bbc's in the case when replication numbers are not all equal. Combining these techniques yields a powerful existence result for balanced binary codes in the intended application.

An understanding of the application is critical to motivating both the definitions given and to describing the specific bbc's sought. We provide a brief overview of the biotechnology application before pursuing the construction of optimal bbc's. For full details on the application, see Sengupta and Tompa [9].

2 The Quality Control Problem

For this discussion, a *DNA molecule* can be abstracted as a string over the alphabet $\{A, C, G, T\}$. An *oligo array* is a small chip containing approximately 100,000 *spots*, to each of which is attached its own synthesized DNA molecule. Oligo arrays are used to measure how much of each gene product is produced by a given cell type under given conditions. For more information on oligo arrays see, for example, Lipschutz *et al.* [5].

Our application is in the manufacture of oligo arrays rather than their subsequent use. An array is manufactured in a series of steps "labeled" $A, C, G, T, A, C, G, T, A, \ldots$ Initially every spot's DNA molecule is empty. In preparation for any given step, an arbitrary subset of the spots can be *masked*. If the step is labeled σ, only a spot that is unmasked will have σ appended to the end of its DNA molecule. By appropriate construction of the masks, each spot can be designed to contain an arbitrary DNA sequence.

The manufacturing process is subject to two different sorts of faults: (1) several individual spots may fail, and (2) an entire manufacturing step may fail, affecting all spots unmasked during that step. The goal of quality control is to identify any single failed step, even if e individual spots fail, where e is a parameter of the manufacturing process. A small number of spots on the chip can be used for this quality control purpose.

Hubbell and Pevzner [4] first investigated this problem. The clever idea underlying their approach is to manufacture identical DNA molecules at multiple spots, using different schedules of steps. If no step fails, all such spots should behave identically. If some step fails, the spots behaving incorrectly hopefully provide a "signature" that identifies the failed step.

A	C		
		G	T
A			T
	C	G	

A			T
	C	G	
A	C		
		G	T

Figure 1: A pair of 4×4 QC blocks. For ease of visualization, the figure shows blanks instead of zeros, and the manufacturing step's label instead of a one.

The problem Hubbell and Pevzner left open was how to design the quality control molecules and schedules to guarantee such signatures, even in the presence of e faulty spots. Sengupta and Tompa [9] reduced this problem to the design of well discriminated balanced binary codes as described below, and supplied an initial collection of good balanced codes.

First they abstracted the quality control problem as that of designing a *QC matrix* Q, which is a 0-1 matrix with a row for each quality control spot, a column for each manufacturing step, and $Q_{ij} = 1$ if and only if spot i is unmasked during step j. Given the spots that subsequently behave incorrectly as a column vector I, identifying the failed step corresponds roughly to finding the column of Q that resembles I, with up to e exceptions. Although this resembles the familiar error-correcting code problem, what makes it more complicated is that (1) one cannot compare the behaviors of spots with different DNA sequences, and (2) even for the spots with identical sequences, it may not be possible to distinguish between all such spots behaving correctly and all such spots behaving incorrectly.

In terms that are beyond our scope, but are detailed by Sengupta and Tompa [9], the properties of a good QC matrix Q are as follows:

1. The set of DNA molecules manufactured at the quality control spots "hybridize poorly" to themselves and each other.

2. Q has high "separation" sep(Q), which ensures sufficient coverage of each step, and sufficient difference between steps to identify the failed step. Sengupta and Tompa proved that sep$(Q) \geq 2e + 1$ is sufficient to identify any single failed step, even in the presence of e arbitrarily faulty spots.

Sengupta and Tompa designed QC matrices with these properties using a product construction. First they hand crafted some *QC blocks*, which are small QC matrices. An example of a pair of 4×4 QC blocks from their paper is given in Figure 1. They then showed that a certain cross product of any well discriminated balanced binary code and any QC block yields a QC matrix with the desired properties above. More specifically, if B is a (v, b, k, d)-bbc, then alternately replacing the ones in each row of B by the two 4×4 QC blocks of Figure 1, and replacing the zeros in B by 4×4 matrices of zeros, produces a $4b \times 4v$ QC matrix Q for which each DNA molecule has length $2k$, the set of DNA molecules hybridizes poorly, and sep$(Q) = 2d$. An example of this product construction is shown in Figure 2.

This then explains the design problem of Section 1. Since the array manufacturer specifies the number of steps ($4v$) and the molecule lengths ($2k$), and the goal is to minimize the number of quality control spots ($4b$) and maximize separation ($2d$), the resulting balanced binary code design problem is to minimize b and maximize discrimination d for a given v and k. For the current photolithographic process, reasonable ranges for the parameters are $16 \leq 2k \leq 20$, $60 \leq 4v \leq 136$, and $4b$ up to a few hundred.

Although Sengupta and Tompa supplied an initial collection of balanced binary codes, they left open the construction of optimal balanced binary codes for arbitrary choices of v, k, and d. The current paper addresses exactly this problem for the relevant parameter ranges given above. The resulting constructions are summarized in Tables 7 and 8.

3 Primal Constructions

In this section, we examine constructions for the bbc set system; to distinguish from later constructions, we call this the *primal* set system. Our constructions begin with a useful connection to balanced incomplete block designs. A t-(v, b, r, k, λ) *design* is a pair $(V, \mathcal{B})$, where V is a set of v *elements*, and $\mathcal{B}$ is a collection of k-element subsets of V called *blocks*. Every t-subset of V appears as a subset of exactly λ of the b blocks in $\mathcal{B}$. It follows that every s-subset for $0 \leq s \leq t$ appears in the same number λ_s of blocks (since the block sizes all equal k). In this notation, $b = \lambda_0$, $r = \lambda_1$, and $\lambda = \lambda_t$. When $t = 2$, a t-design is a *balanced incomplete block design*, or simply a *block design*. The connection to bbc's is immediate:

Theorem 3.1 *When $v > k > 2$, every 2-(v, b, r, k, λ) design is an optimal equireplicate $(v, b, k, \text{min}(r, b - r))$-bbc.*

Proof. The design is a (v, b, k)-set system by construction. To verify that it is $\min(r, b - r)$-discriminated, we observe that the number of blocks containing exactly one of (any) two distinct elements is $2(r - \lambda)$. By Lemma 1.2, we can assume without loss of generality that $v \geq 2k$. Then $2(r - \lambda) \geq r$ since $r = \frac{\lambda(v-1)}{k-1}$. Optimality follows from Lemma 1.3 and the observation that $d = \min(r, b - r) = r$, since $r = bk/v \leq b/2$. □

Corollary 3.2 *There are equireplicate (16,30,8,15)-, (18,34,9,17)-, and (20,38,10,19)-bbc's.*

Proof. There exist 2-(16,30,15,8,7), 2-(18,34,17,9,8), and 2-(20,38,19,10,9) designs (see, e.g., [7]). The first and last are *Hadamard designs* arising from Hadamard matrices; see [1]. □

Block designs have been very extensively studied, and much is known about their existence; see [7] for a table giving known existence results for 'small' values of r. For our application, the conditions on block designs are too stringent. Indeed, in a block design, every two elements have the property that there are *exactly* $2(r - \lambda)$ blocks containing precisely one of them, and the application does not require this type of uniformity. Consequently, block designs provide only a small fraction of the bbc's needed, even among the optimal equireplicate cases. A more serious drawback arises since b is constrained to be at least v by Fisher's inequality (see, for example, [1]). Using addition, however, we are most interested in bbc's with b very small.

We therefore relax the requirements by allowing, for each pair of elements, the number of blocks containing exactly one of them to vary, provided that it remain at least d. Translating to the design vernacular, when the bbc is equireplicate, we are specifying that every pair of elements occur together in at most some number λ of blocks.

A t-(v,k,λ) *packing* $(V,\mathcal{B})$ is a (v,b,k)-set system in which every t-subset of elements occurs together in at most λ of the blocks in $\mathcal{B}$. A 2-(v,k,λ) packing in which $v \geq 2k$ and every element has replication number at least r yields a bbc which is $\min(r, 2(r-\lambda))$-discriminated. See [8] for a survey of packings.

Our first construction produces 2-(v,k,λ) packings with $b = v$. We take, as the set of elements, the integers modulo v, $\mathbb{Z}_v$. We choose a single block, B, containing k elements, and form $\mathcal{B} = \{B+0,\ldots,B+(v-1)\}$, where the translate $B+i = \{x+i \bmod v : x \in B\}$. To determine the index λ of the packing $(\mathbb{Z}_v,\mathcal{B})$, proceed as follows. Each pair $\{i,j\}$ of elements has an associated *difference* modulo v, namely $\min(i-j \bmod v, j-i \bmod v)$. If this difference appears as the difference between two elements of B, then the pair occurs in exactly one translate of these two elements unless the difference is precisely half of v, in which case the pair appears in two translates. Hence to determine the maximum number of times that a pair occurs in the packing, we need only determine how many pairs of elements in B have a specified difference. To handle the case when v is even and the difference examined is $d/2$, we must double the number of occurrences of the difference.

In the construction of bbc's, we may not require the minimum possible value of λ. Indeed, if $v \geq 2k$ and we are to produce a (v,v,k,k)-bbc, we require only that every difference appear at most $\lfloor k/2 \rfloor$ times. A single block of k elements from $\mathbb{Z}_v$ in which every difference is represented at most $\lfloor k/2 \rfloor$ times, except when v is even we require that $v/2$ be represented at most $\lfloor v/4 \rfloor$ times, is a *near difference set*. When v is odd and every difference is represented the same number of times, the block is a *cyclic difference set*, and these have been studied extensively [1].

In Table 3, we present near difference sets for a number of parameters of interest. These solutions were found using a simple backtracking method.

Such bbc's arising from near difference sets can exist only for some of the parameter sets of interest, namely those when $b = v$. We therefore examine a more general method. Again we take $\mathbb{Z}_v$ as the set of elements. We form a number of *base blocks* B_1, B_2, ..., B_ℓ. We can again develop each base block modulo v to form v blocks. For certain base blocks, the v blocks in the development are not all distinct. In these cases, we can choose to include only a subset of the blocks. Suppose, for example, that v and k are both even, and that $B_i = \{b_1,\ldots,b_{k/2}, b_1+(v/2),\ldots,b_{k/2}+(v/2)\}$, with $0 \leq b_i < v/2$ when $1 \leq i \leq k/2$. Then $B_i + (v/2) = B_i$. In this case, we can produce only $v/2$ blocks, a *half orbit*, by including $B_i + j$ for $j = 0,\ldots,(v/2)-1$. In Table 4, we present solutions containing one half orbit and one starter block generating v blocks. To prescribe the block for the half orbit, we give only the elements $b_1,\ldots,b_{k/2}$.

Other relaxations of the stringent block design conditions can be exploited. A $(g,k;\lambda)$-*difference matrix* over $\mathbb{Z}_g$ is a $k \times \lambda g$ array A with entries from $\mathbb{Z}_g$, with the property that for any $1 \leq i < j \leq k$, the collection of differences $\{A_{i,\ell} - A_{j,\ell} \bmod g : 1 \leq \ell \leq \lambda g\}$ contains the g numbers in $\mathbb{Z}_g$ λ times each.

Proposition 3.3 *There is an equireplicate (27,21,9,7)-bbc and an equireplicate (30,21,10,7)-bbc.*

Proof. There is a (3,9;3)-difference matrix; see [2], for example. Choose any seven of its columns, and append the fourteen further columns obtained by developing the columns under addition modulo 3. Treat the resulting set of 21 columns as blocks of a packing on the 27 points (i,σ), where i indicates the row, and σ the symbol from $\mathbb{Z}_3$. The resulting packing has $\lambda = 3$, and hence is a 2-(27,9,3) packing on 21 blocks which is equireplicate. Hence an equireplicate (27,21,9,7)-bbc

Figure 2: The product of a (19,19,9,9,4) 2-design and the pair of 4×4 QC blocks of Figure 1, resulting in a 76×76 QC matrix Q with minimum separation sep$(Q) = 18$.

v	k	d	Block	v	k	d	Block
9	8	1	0 1 2 3 4 5 6 7	10	8	2	0 1 2 3 4 5 6 7
10	9	1	0 1 2 3 4 5 6 7 8	11	8	3	0 1 2 3 4 5 6 8
11	9	2	0 1 2 3 4 5 6 7 8	11	10	1	0 1 2 3 4 5 6 7 8 9
12	9	3	0 1 2 3 4 5 6 7 9	12	10	2	0 1 2 3 4 5 6 7 8 9
13	8	5	0 1 2 3 4 5 8 10	13	9	4	0 1 2 3 4 5 7 9 10
13	10	3	0 1 2 3 4 5 6 7 8 10	14	8	6	0 1 2 3 4 5 7 10
14	9	5	0 1 2 3 4 5 6 9 11	15	8	7	0 1 2 3 5 7 8 11
15	9	6	0 1 2 3 4 5 6 8 11	15	10	5	0 1 2 3 4 5 6 7 10 12
16	8	8	0 1 2 3 4 7 9 12	16	9	7	0 1 2 3 4 6 7 11 13
16	10	6	0 1 2 3 4 5 6 7 9 12	17	8	8	0 1 2 3 4 6 9 13
17	9	8	0 1 2 3 4 5 8 10 13	17	10	7	0 1 2 3 4 5 7 8 11 13
19	8	8	0 1 2 3 4 6 9 13	19	9	9	0 1 2 3 5 7 12 13 16
19	10	9	0 1 2 3 5 7 12 13 15 16	20	9	9	0 1 2 3 4 7 9 12 16
20	10	10	0 1 2 3 4 6 8 11 14 15	21	8	8	0 1 2 3 5 8 12 16
21	9	9	0 1 2 3 4 7 9 13 18	21	10	10	0 1 2 3 4 5 8 10 13 17
22	9	9	0 1 2 3 4 6 9 13 17	22	10	10	0 1 2 3 4 5 8 10 13 17
23	8	8	0 1 2 3 5 8 12 16	23	9	9	0 1 2 3 4 6 9 13 17
23	10	10	0 1 2 3 4 5 7 10 14 18	24	9	9	0 1 2 3 4 6 9 13 17
24	10	10	0 1 2 3 5 6 11 13 17 20	25	8	8	0 1 2 3 5 8 12 16
25	9	9	0 1 2 3 4 6 9 13 17	26	9	9	0 1 2 4 6 11 12 20 23
26	10	10	0 1 2 3 4 7 9 12 16 20	27	8	8	0 1 2 3 5 8 12 16
27	10	10	0 1 2 3 4 6 9 13 17 22	28	9	9	0 1 2 3 5 8 12 16 21
29	8	8	0 1 2 3 5 8 12 16	29	9	9	0 1 2 3 5 8 12 16 22
29	10	10	0 1 2 3 4 6 9 13 17 23	30	9	9	0 1 2 3 5 8 12 16 21
31	8	8	0 1 2 4 7 12 16 25	31	9	9	0 1 2 3 5 8 12 16 21
31	10	10	0 1 2 3 4 6 9 13 17 22	32	9	9	0 1 2 3 5 8 12 16 22
33	8	8	0 1 2 4 7 11 19 24	33	9	9	0 1 2 3 5 8 12 16 21
33	10	10	0 1 2 3 5 8 12 18 22 27	34	9	9	0 1 2 3 5 8 12 16 21

Table 3: Near Difference Sets

v	k	d	Half Orbit	Full Orbit
10	8	3	0 1 2 3	0 1 2 3 4 5 6 7
12	10	3	0 1 2 3 4	0 1 2 3 4 5 6 7 8 9
14	8	9	0 1 2 4	0 1 2 3 4 6 7 12
16	10	9	0 1 2 3 4	0 1 2 3 4 5 7 8 10 14
22	10	15	0 1 2 3 5	0 1 2 3 5 7 10 15 18 19
24	10	15	0 1 2 4 9	0 1 2 3 6 7 9 11 17 20
26	10	15	0 1 2 4 7	0 1 2 3 4 7 10 12 18 22

Table 4: One and a Half Orbits

exists. Now this 2-(27,9,3) packing can, by construction, be partitioned into seven sets of three blocks each, so that each set contains three mutually disjoint blocks. Let $P_1, \ldots, P_7$ be such a partition of the blocks. Add three new elements a, b, c to the packing. Add a to each block in P_1 and P_2 and to the first block in P_7; add b to each block in P_3 and P_4 and to the second block in P_7; add c to the remaining 7 blocks. The result is a 2-(30,10,3) packing; it is equireplicate with replication number 7, and hence yields an equireplicate (30,21,10,7)-bbc. □

4 Dual Constructions

Since we are primarily interested in cases in which $b < v$, it is natural to consider the dual set system. The *dual set system* of a set system $(V, \mathcal{B})$ is a set system $(X, \mathcal{D})$ in which $X = \{x_B : B \in \mathcal{B}\}$ and $\mathcal{D} = \{D_y : y \in V\}$, where $D_y = \{x_B : y \in B \in \mathcal{B}\}$. The dual of a (v, b, k)-set system with replication numbers $r_1, \ldots, r_v$ is a (b, v)-set system with v blocks of sizes $r_1, \ldots, r_v$ and having constant replication number k. Indeed when the (v, b, k)-set system is equireplicate with replication number r, its dual is a (b, v, r)-set system which has constant replication number k. The dual of the set system in Table 1 is given in Table 5.

0	3	7	8	9	4	5	7	9	10	1	4	7	8	13
1	3	6	8	12	0	1	2	3	13	3	5	8	10	13
2	4	10	11	13	0	6	7	10	13	2	6	8	9	13
0	1	4	6	9	4	5	6	12	13	5	6	7	8	11
3	6	9	10	11	0	2	4	5	8	0	5	9	11	13
1	2	7	9	11	1	3	4	5	11	4	8	9	11	12
0	1	8	10	11	2	3	5	9	12	0	2	6	11	12
2	3	4	6	7	3	7	11	12	13	0	1	5	7	12
2	7	8	10	12	1	2	5	6	10	1	9	10	12	13
0	3	4	10	12										

Table 5: Dual Set System of (28,14,10,5)-bbc

The discrimination of the primal is reflected in the dual in a somewhat different manner than in the primal. Two blocks of the dual sharing μ elements result in a discrimination d of the primal satisfying $d \leq 2r - 2\mu$; hence maximizing d amounts to minimizing μ, the intersection size of two blocks, since r is fixed. Translating this into design vernacular, we establish that:

Theorem 4.1 *A t-$(b, r, 1)$ packing on v blocks with replication number k yields an equireplicate $(v, b, k, min(r, b - r, 2(r - t + 1)))$-bbc with replication number r.*

Proof. The dual of a t-$(b, r, 1)$ packing on v blocks with replication number k is a (v, b, k)-set system with replication number r in which every pair of elements occurs in at most $t - 1$ blocks together. □

Hence our goal is to produce t-$(b, r, 1)$ packings with $t \leq r/2 + 1$. One potential benefit of this dual approach when $b < v$ is that we can examine constructions over $\mathbb{Z}_b$ rather than the larger $\mathbb{Z}_v$. We illustrate this by producing a number of 4-equireplicate $(2m, m, 8, 4)$-bbc's.

Theorem 4.2 *A 4-equireplicate $(2m, m, 8, 4)$-bbc exists for all $m \geq 10$.*

Proof. The dual set system is constructed with elements in $\mathbb{Z}_m$, and has two base blocks which are developed modulo m. We need only ensure that the result is a 3-$(m, 4, 1)$ packing. When $m = 10$, use the base blocks {0,1,2,6} and {0,2,4,7}; when $m \geq 11$, use the base blocks {0,1,2,7} and {0,1,3,5}. The proof is completed by verifying that no translate of a triple in either base block appears as a translate of a different triple or as a different translate of this triple. □

In a similar vein, other bbc's are easily produced from 3-$(b, 5, 1)$ packings:

v	b	k	d	Dual Base Blocks in $\mathbb{Z}_b$
30	15	10	5	{0,1,4,11,14}, {0,2,7,8,13}
32	16	10	5	{0,2,8,14,15}, {0,3,7,11,14}
34	17	10	5	{0,3,10,12,14}, {0,4,12,15,16}

The dual solutions thus far presented all have the property that v is an integral multiple of b. We can vary the construction to admit other solutions. Suppose, for example, that we are to produce a $(25, 10, 10, 4)$-bbc. Its dual is a $(10, 25, 4)$-set system which is 10-equireplicate and forms a 3-(10,4,1) packing. Two base blocks, {0,1,2,6} and {0,2,4,7}, generate 20 blocks in $\mathbb{Z}_{10}$. A third base block $\{0, 1, 3, 4\}$ is used, but in its development, we only include translates obtained by adding the *even* integers. Since this last base block contains two even and two odd numbers, this development ensures that the resulting packing is 10-equireplicate.

In general, by selecting certain translates out of one orbit of a base block, we can vary k and b in the construction. We give some further examples of constructions of this type next, subscripting one block with the integers to be added in forming its translates. The first three employ packings with $t = 3$, while the last five employ packings with $t = 4$.

v	b	k	d	Dual Base Blocks in $\mathbb{Z}_b$
27	12	9	4	{0,1,3,5}, {0,1,2,7}, $\{0,3,6,9\}_{0,1,2}$
30	12	10	4	{0,1,3,5}, {0,1,2,7}, $\{0,2,6,8\}_{0,1,2,3,4,5}$
32	20	8	5	{0,1,8,14,17}, $\{0,2,11,18,19\}_{0,1,2,5,6,7,10,11,12,15,16,17}$
15	10	9	4	{2,3,5,6,7,9}, $\{3,4,5,6,8,9\}_{0,2,4,6,8}$
28	21	8	6	{0,1,4,9,18,20}, $\{0,1,7,8,14,15\}_{0,1,2,3,4,5,6}$
33	22	9	6	{0,1,6,7,10,15}, $\{0,1,3,11,12,14\}_{0,1,2,3,4,5,6,7,8,9,10}$
24	21	8	7	{0,1,2,4,6,7,14}, $\{0,3,6,9,12,15,18\}_{0,1,2}$
32	28	8	7	{0,1,2,4,7,11,17}, $\{0,4,8,12,16,20,24\}_{0,1,2,3}$

In a number of cases, we have not been able to find (dual) solutions which are cyclic modulo b. In some of these situations, we have resorted to using a smaller group.

Theorem 4.3 *There is a* $(3m, 2m, 9, 6)$*-bbc for all* $m \geq 7$*.*

Proof. We form the dual of the required bbc on the element set $\mathbb{Z}_m \times \{0, 1\}$. We begin with three base blocks {(0,0),(1,0),(3,0),(0,1),(1,1),(3,1)}, {(2,0),(4,0),(5,0),(6,0),(0,1),(3,1)}, and {(0,0),(3,0),(2,1),(4,1),(5,1),(6,1)}. Each gives m blocks of the dual by adding the nonzero elements of $\mathbb{Z}_m$ in turn to the first coordinates of each element. It is easily verified that the result is a 3-$(2m, 6, 1)$ packing which is 9-equireplicate. □

Theorem 4.4 *There is a* $(4m, 3m, 8, 6)$*-bbc and a* $(5m, 3m, 10, 6)$*-bbc for all* $m \geq 5$*.*

Proof. We form the dual of the required bbc on the element set $\mathbb{Z}_m \times \{0, 1, 2\}$. We begin with five base blocks:

{(0,0),(1,0),(2,0),(3,0),(4,1),(4,2)},
{(0,1),(1,1),(2,1),(3,1),(4,0),(4,2)},
{(0,2),(1,2),(2,2),(3,2),(4,1),(4,0)},
{(0,0),(1,0),(0,1),(1,1),(0,2),(1,2)},
{(0,0),(2,0),(0,1),(2,1),(0,2),(2,2)}.

Each gives m blocks of the dual by adding the nonzero elements of $\mathbb{Z}_m$ in turn to the first coordinates of each element. It is easily verified that the result is a 4-$(3m, 6, 1)$ packing, and yields a $(5m, 3m, 10, 6)$-bbc. Deleting the last base block and its translates yields a $(4m, 3m, 8, 6)$-bbc. □

Proposition 4.5 *There exists a (24,15,8,5)-bbc and a (27,15,9,5)-bbc.*

Proof. The point set for the dual in each case is the fifteen points $\mathbb{Z}_{12} \cup \{a, b, c\}$. Start with the blocks obtained by developing $\{0, 1, 2, 4, 9\}$ and $\{0, 1, 5\}$ modulo 12. Then the translates of $\{0, 1, 5\}$ can be partitioned into three parallel classes of four blocks each. For the three parallel classes in turn, add the points $\{a, b\}$, $\{a, c\}$, and $\{b, c\}$, respectively to each block of the parallel class. The result is the dual of the (24,15,8,5)-bbc.

To this dual, add the three distinct translates of $\{0, 3, 6, 9\}$ modulo 12, placing a, b, and c respectively in one of the three translates. This is the dual of the (27,15,9,5)-bbc. □

For small values of d, a direct construction can be quite simple:

Proposition 4.6 *There are (12,6,8,2)-, (12,8,9,2)-, (14,7,10,2)-, and (15,6,10,2)-bbc's.*

Proof. Start with a k-regular graph on n vertices, for $(k, n) =$ (4,6), (3,8), (4,7), or (5,6), respectively. The complement of this set system forms the dual of the required bbc. □

Similarly, the complement of the blocks of a 2-(9,12,4,3,1) design forms the dual of a (12,9,8,3)-bbc.

We employ some constructions from Hadamard designs. A *Hadamard 3-design* is a 3-$(4n, 2n, n-1)$ design [1]. Such a design has $8n-2$ blocks, and they occur in $4n-1$ complementary pairs. Deleting one point of a 3-$(4n, 2n, n-1)$ design produces a 2-$(4n-1, 2n-1, n-1)$ design which has $4n-1$ blocks and replication number $2n-1$. Hence the 2-design is *symmetric*, and consequently every two blocks of the 2-design intersect in $n-1$ elements. The 3-design can be recovered from the 2-design by including the complements of the blocks of the 2-design, and including the blocks with a single new element which is adjoined to each. From this construction, the 3-design is an $(n+1)$-$(4n, 2n, 1)$ packing. Deleting blocks retains this packing property, but more importantly deleting complementary pairs of blocks retains the property that the packing is equireplicate. Indeed if we select j complementary pairs of blocks, the replication number is j; when $j \geq 2n$, the packing leads to a $(2j, 4n, j, 2n)$-bbc. Using Hadamard designs for $n \in \{3, 4, 5\}$, we obtain the following:

Proposition 4.7 *There exist (16,12,8,6)-, (16,20,8,10)-, (18,12,9,6)-, (18,16,9,8)-, (18,20,9,10)-, (20,12,10,6)-, and (20,16,10,8)-bbc's.*

We also need one specific construction:

Proposition 4.8 *There exist (18,9,8,4)- and (18,9,10,4)-bbc's.*

Proof. The second is the complement of the first. To construct the dual of the first, we begin with nine points $\{(i,j) : i,j \in \mathbb{Z}_3\}$. We include all nine blocks of the form $\{(i,k),(i,\ell),(j,k),(j,\ell)\}$ with $i,j,k,\ell \in \mathbb{Z}_3$, $i \neq j$, and $k \neq \ell$. We then add all nine blocks of the form $\{(i,j),(i,k),(a,\ell),(b,\ell)\}$ when $\{i,a,b\} = \{j,k,\ell\} = \mathbb{Z}_3$. This is a 3-(9,4,1) packing with 18 blocks, having constant replication number eight. □

In Table 6, the dual of a (16,38,8,19)-bbc is presented. The method used to obtain this solution is of independent interest, and is described in [3].

0	4	6	7	8	9	10	11	13	14	17	19	21	27	28	29	32	35	37
0	2	4	5	8	9	15	16	17	18	19	20	22	25	27	29	31	32	34
0	1	5	6	14	16	18	19	20	23	24	26	27	28	30	31	34	35	37
0	3	7	8	12	13	16	18	20	21	26	28	29	30	31	32	33	35	36
1	2	5	6	7	9	11	13	20	21	22	24	25	30	31	32	34	35	36
1	2	3	4	5	10	13	14	15	16	20	21	24	27	29	30	32	33	37
0	1	3	4	5	7	11	12	14	17	19	20	22	23	32	33	34	36	37
2	5	6	7	8	9	12	14	15	16	19	23	25	29	30	33	35	36	37
1	2	3	4	6	12	13	15	17	23	25	26	28	29	31	32	34	35	37
2	3	6	7	10	11	15	16	17	18	20	21	22	23	27	28	31	36	37
3	4	5	6	9	10	11	12	18	19	22	24	26	27	29	31	33	35	36
0	1	2	8	10	11	12	13	15	17	19	24	27	28	30	31	33	34	36
9	10	11	13	14	15	18	19	20	21	22	23	25	26	28	29	30	33	34
1	3	5	7	8	9	10	11	12	14	15	16	17	18	24	25	26	28	32
0	2	3	4	7	8	10	12	14	21	22	23	24	25	26	27	30	34	35
0	1	4	6	8	9	13	16	17	18	21	22	23	24	25	26	33	36	37

Table 6: Dual of a (16,38,8,19)-bbc

5 Nonexistence Results

We have presented a large collection of constructions for optimal equireplicate bbc's, focussing on those with smaller discriminations in order to use addition to produce those with larger discrimination. However, not all bbc's exist; in fact, those with low discrimination appear to be the least likely to exist. We do not restrict to equireplicate bbc's in this section. We establish a preliminary result for small discrimination:

Theorem 5.1 *An optimal* $(v, b, k, 1)$*-bbc exists only when* $v = k + 1$ *or* $k = 1$.

Proof. If $v \geq 2k$, the dual set system has $b = \lceil v/k \rceil$ points, and has at least $v - k + 1$ blocks of size 1. When $k > 1$, some block is repeated and hence the discrimination is 0. If $v < 2k$, the dual set system has $\lceil v/(v - k) \rceil$ points, and has at least $k + 1$ blocks of size $v - 1$. Its complement therefore has $k + 1$ blocks of size 1, and hence contains a repeated block unless $v - k = 1$. □

When the discrimination is two, the analysis is slightly more complex. We describe one concrete example, and then give much briefer arguments thereafter. Let us establish that a (31,8,8,2)-bbc does not exist. If one were to exist, its dual has eight points. It has 31 blocks, and each must have size at least two (and at most six). There are $64 = 8 \cdot 8$ occurrences of points in blocks. Hence there are either 30 blocks of size two and one of size four, or there are 29 blocks of size two and two of size three. In this case, since $\binom{8}{2} = 28$, there must be a *repeated* block of size two. But then the bbc has two identical columns, and its discrimination is zero, a contradiction. In general, the nonexistence results all arise from an analysis of the cases that can arise, showing that each cannot have the required discrimination.

Theorem 5.2 *An optimal* $(v, b, k, 2)$*-bbc exists only if:*

1. $v \leq 13$, $v \in \{29, 30\}$, *or* $v \geq 33$ *when* $k = 8$*;*

2. $v \leq 14$, $v \in \{32, 37, 38\}$, *or* $v \geq 41$ *when* $k = 9$*;*

3. $v \leq 16$, $v \in \{46, 47\}$, *or* $v \geq 51$ *when* $k = 10$.

Proof. First we suppose that $v \geq 2k$. Then $b = \lceil \frac{2v}{k} \rceil$. The dual of the required bbc therefore has bk occurrences of elements distributed across v blocks, each having size at least two. It follows that 'most' blocks have size equal to two. If the dual has a block of size three, then no block of size two can share both elements with the block of size three. To maximize the number of blocks in the dual, we therefore construct the dual with the largest possible number of blocks of size four, and the remaining blocks of size two.

Consider the case when $k = 8$. Write $v = 4s + \alpha$ with $\alpha \in \{1, 2, 3, 4\}$. Then $b = 2s + 1$, and $bk = 16s + 8$. It follows that the number of blocks of size two in the dual, when no blocks of size three are chosen, is at least $4s - 4 + 2\alpha$. Now requiring that $4s - 4 + 2\alpha \leq \binom{b}{2}$, we obtain that $s(s - 7) \geq 4\alpha - 8$. Hence $s \geq 7$ when $\alpha \in \{1, 2\}$ and $s \geq 8$ when $\alpha \in \{3, 4\}$. When $k = 9$ or $k = 10$, the analysis is similar and is omitted.

When $v < 2k$, we use the fact that a $(v, b, k, 2)$-bbc is equivalent to a $(v, b, v - k, 2)$-bbc. The remaining cases have $b = 5$ but require more than 10 blocks of size two in the dual of the complementary bbc. □

The restrictions when the discrimination is three are more severe. In this case, the dual has $\lceil \frac{3v}{k} \rceil$ points, and its v blocks are almost all of size three. However, two blocks of size three are permitted to intersect in only one element. This establishes easily that when $k \in \{8, 9, 10\}$ and $2k \leq v \leq 34$, no optimal $(v, b, k, 3)$-bbc exists. When $v < 2k$, a similar argument excludes $v \in \{13, 14, 15\}$ when $k = 8$; $v \in \{15, 16, 17\}$ when $k = 9$; and $v \in \{15, 16, 17, 18, 19\}$ when $k = 10$.

Turning to discrimination four, the blocks of size four in the dual form a packing in which every 3-subset appears in at most one block. Using this fact, we can conclude that no optimal $(v, b, k, 4)$-bbc exists when $v \in \{15, 16, 17\}$ and $k = 8$; $v \in \{17, 18, 19, 20\}$ and $k = 9$; or $v \in \{19, 20, 21, 22\}$ and $k = 10$. For example, when $(v, k) \in \{(16, 8), (18, 9), (20, 10)\}$, the dual is a 3-(8,4,1) packing with 16, 18, or 20 blocks; but the maximum packing has only 14 blocks.

For discrimination five, the blocks of size five again form a packing in which every 3-subset appears in at most one block. When $(v, k) \in \{(17, 8), (19, 9), (21, 10), (22, 10)\}$, the dual has 11 points and has at least 14 blocks of size five. Consider then the *derived* design obtained by choosing a point containing the maximum number of blocks of size five, selecting all blocks of size five containing this point, and then deleting the point from each. This is a 2-(10,4,1) packing, which must have at least 7 blocks by construction. But no 2-(10,4,1) packing with 7 blocks exists. By complementation, we also eliminate the cases when $(v, k) = (17, 9)$ or (19,10). A similar argument shows that no (24,12,10,5)-bbc or (26,13,10,5)-bbc exists. A complete exhaustive search by backtracking established the nonexistence of a (19,12,8,5)-bbc.

The astute reader will have observed that fewer negative results arise for even discrimination than for odd, and that as the discrimination increases, the negative results are sparser. Indeed, in Tables 7 and 8 there are very few negative results for $d > 5$. It is, however, possible to prove such results. We give examples in the following two theorems.

Theorem 5.3 *A* $(2k, 2d, k, d)$*-bbc does not exist when* d *is odd and* $d < 2k - 1$.

Proof. Such a bbc is a 2-$(2k, 2d, \lfloor d/2 \rfloor)$ packing. Hence we require that $\lfloor d/2 \rfloor \cdot \binom{2k}{2} \geq 2d \cdot \binom{k}{2}$. Letting $d = 2s + 1$, we require that $s(2k - 1) \geq (2s + 1)(k - 1)$. Simplifying, $2ks - s \geq 2ks + k - 2s - 1$, i.e., $s \geq k - 1$, or $d \geq 2k - 1$. □

For even values of d there is also a nonexistence result.

Theorem 5.4 *A* $(2k, 2d, k, d)$*-bbc does not exist when* $d < k/2$.

Proof. The columns of such a bbc are $2k$ binary vectors of length $2d$ so that the Hamming distance between any pair is at least d. By the pigeonhole principle k of them share the same first coordinate, giving a set of k vectors of length $2d - 1$ so that the Hamming distance between every pair is at least d. Since in each coordinate there are at most $\lfloor k^2/4 \rfloor$ pairs of these vectors that differ in this coordinate, and the sum of distances between all pairs of these vectors is at least $\binom{k}{2}d$, it follows that

$$(2d - 1)k^2/4 \geq (2d - 1)\lfloor k^2/4 \rfloor \geq \binom{k}{2}d,$$

implying that $d \geq k/2$, as needed. □

Similar nonexistence results can be derived for other values of v and k, provided $v \geq 2k$ and $v - 2k$ is small, using the Plotkin bound (see, for example, [6, pp. 41-43]). Since our focus here is on cases in the range of practical interest, we do not include a detailed study of these results.

6 Existence of optimal bbc's

We summarize the existence results for equireplicate optimal bbc's in the range of primary interest for the oligo array application. We can assume that addition is applied to all of the basic designs produced. Then it is an easy matter to verify that all but a handful of parameter sets are settled. When $k \in \{8, 9, 10\}$ and $k < v \leq 34$, we have established existence or nonexistence in all but five cases, namely when (v, k, d) is one of (16,8,17), (18,9,19), (18,9,21), (20,10,21), or (20,10,23).

In [3], we develop a hillclimbing method which is remarkably successful at producing bbc's, even optimal ones. Indeed, when the bbc is not equireplicate, we succeeded in producing a large number of base designs. In Tables 7 and 8, we give a statement of the current result for all parameter sets with $k \in \{8, 9, 10\}$ and $k < v \leq 34$. The encoding is as follows: `+` denotes the existence of an optimal equireplicate bbc, which is described in this paper; `?` denotes an unsettled equireplicate case; `.` denotes a parameter set for which nonexistence of any optimal bbc has been established; `Y` denotes a non-equireplicate optimal bbc, found using the algorithm from [3]; and `o` denotes an unsettled non-equireplicate case. The majority of entries are obtained by addition of bbc's with smaller discrimination; a construction of this type is denoted by `I`, for 'implied'. Note that sometimes an optimal bbc can be implied by the addition of two nonequireplicate optimal bbc's.

We present the status only for $1 \leq d \leq 40$, but it can easily be established that existence is implied for all $d \geq 40$ for all parameter sets in our range, using addition.

The practical consequence of this is that, for large discrimination, the problem appears to become easier. However, only through the direct and computational constructions for small discrimination have we been able to establish such a strong existence result.

7 Concluding Remarks

Optimal balanced binary codes appear, at first glance, to require strong balance conditions leading to designs. Indeed, when $v = 2k$, the conditions are quite severe and do require the pair-balance condition of balanced incomplete block designs. However, when v is not near $2k$, the packing conditions that are required appear to be much less restrictive than do the conditions on block sizes and replication numbers. This is the primary reason that the approach here of constructing the required packings directly appears more fruitful than the approach of starting with block designs and applying simple transformations.

One might expect that the non-equireplicate cases would be easier in view of the increased flexibility in choosing replication numbers. In [3], we exploit this flexibility to develop an heuristic search technique that is very successful.

While we have focussed in this paper on cases in the range of practical interest, we expect that similar conclusions and techniques arise more generally in the existence of bbc's.

Acknowledgments

Thanks to Jeff Dinitz, Vic Klee, Don Kreher, Esther Lamken, and Rimli Sengupta for helpful suggestions. Research of the authors is supported by a USA-Israel BSF grant (Alon), ARO grant

v	k	Existence for discrimination d, $1 \le d \le 40$			
		0000000001	1111111112	2222222223	3333333334
		1234567890	1234567890	1234567890	1234567890
9	8	+IIIIIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
10	8	.++IIIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
10	9	+IIIIIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
11	8	.Y+YIIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
11	9	.+YIIIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
11	10	+IIIIIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
12	8	.++IIIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
12	9	.++IIIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
12	10	.++IIIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
13	8	.Y.Y+YIYII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
13	9	.YY+IIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
13	10	.Y+IIIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
14	8	...YY+YY+I	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
14	9	.YYI+YIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
14	10	.+YIIIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
15	8	....oY+YYY	YIIIIIIIII	IIIIIIIIII	IIIIIIIIII
15	9	...+Y+YIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
15	10	.+.I+IIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
16	8	.....+.+.+	.I.I+I?I+I	IIIIIIIIII	IIIIIIIIII
16	9	...YYY+YYY	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
16	10	.Y.YY+YI+I	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
17	8	.....Yo+YY	YIYIYIIIII	IIIIIIIIII	IIIIIIIIII
17	9	.....Yo+YY	YIYIYIIIII	IIIIIIIIII	IIIIIIIIII
17	10	...YYY+IYI	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
18	8	...+YYoIII	YIIIIIIIII	IIIIIIIIII	IIIIIIIIII
18	9	.....+.+.+	.I.I.I+I?I	?IIIIIIIII	IIIIIIIIII
18	10	...+YYoIII	YIIIIIIIII	IIIIIIIIII	IIIIIIIIII
19	8	...Y.YY+YY	IIYIIIIIII	IIIIIIIIII	IIIIIIIIII
19	9	.....ooY+Y	YYYYYIIIII	IIIIIIIIII	IIIIIIIIII
19	10	.....ooY+Y	YYYYYIIIII	IIIIIIIIII	IIIIIIIIII
20	8	...+Y+YIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
20	9	....oYoY+Y	YYYIIIIIII	IIIIIIIIII	IIIIIIIIII
20	10	.....+.+.+	.I.I.I.I+I	?I?IIIIIII	IIIIIIIIII
21	8	...YYYY+YI	YIIIIIIIII	IIIIIIIIII	IIIIIIIIII
21	9	...Yo+YY+I	YIIIIIIIII	IIIIIIIIII	IIIIIIIIII
21	10	.....oooo+	YYYYYYYYYI	IIIIIIIIII	IIIIIIIIII
22	8	...+YYYIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
22	9	...YYYYI+I	YIIIIIIIII	IIIIIIIIII	IIIIIIIIII
22	10	.....YoYo+	YYYY+IYIYI	IIIIIIIIII	IIIIIIIIII

Table 7: Existence of optimal bbc's, I

v	k	Existence for discrimination d, $1 \le d \le 40$			
		0000000001	1111111112	2222222223	3333333334
		1234567890	1234567890	1234567890	1234567890
23	8	...YYYY+YY	YIIIIIIIII	IIIIIIIIII	IIIIIIIIII
23	9	...YoYYY+Y	IYIIIIIIII	IIIIIIIIII	IIIIIIIIII
23	10	...YoYYYY+	YIYIIIIIII	IIIIIIIIII	IIIIIIIIII
24	8	...++++III	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
24	9	...YY+YI+I	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
24	10	...Y.YoIY+	YYII+IYIII	IIIIIIIIII	IIIIIIIIII
25	8	...YYYY+II	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
25	9	...YoYYY+Y	YIIYIIIIII	IIIIIIIIII	IIIIIIIIII
25	10	...+o+YIYI	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
26	8	...+YYYIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
26	9	...YYYYI+Y	YYIIIIIIII	IIIIIIIIII	IIIIIIIIII
26	10	...Y.YYYY+	YIYI+IIIII	IIIIIIIIII	IIIIIIIIII
27	8	...YYYY+II	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
27	9	...++++III	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
27	10	...YYYYII+	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
28	8	...+Y+YIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
28	9	...YYYYY+I	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
28	10	...Y+YYYII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
29	8	.Y.YYYY+YI	YIIIIIIIII	IIIIIIIIII	IIIIIIIIII
29	9	...YYYYI+I	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
29	10	...YYYYII+	YYYIIIIIII	IIIIIIIIII	IIIIIIIIII
30	8	.Y.+YIIIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
30	9	...YY+YY+I	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
30	10	...++++III	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
31	8	...YYYY+YY	YIIIIIIIII	IIIIIIIIII	IIIIIIIIII
31	9	...YYYYI+I	YIIIIIIIII	IIIIIIIIII	IIIIIIIIII
31	10	...YYYYYY+	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
32	8	...++++III	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
32	9	.Y.YYYYY+I	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
32	10	...Y+YYIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
33	8	.Y.YYYY+II	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
33	9	...YY+YI+I	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
33	10	...YYYYYY+	IIYIIIIIII	IIIIIIIIII	IIIIIIIIII
34	8	.Y.+YIYIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
34	9	...YYYYY+I	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII
34	10	...Y+YYIII	IIIIIIIIII	IIIIIIIIII	IIIIIIIIII

Table 8: Existence of optimal bbc's, II

DAAG55-98-1-0272 (Colbourn), DOE grant DE-FG02-00ER45828 (Colbourn), and NSF grant DBI-9974498 (Tompa).

References

[1] T. Beth, D. Jungnickel and H. Lenz, *Design Theory*, Cambridge University Press, Cambridge, 1986.

[2] C.J. Colbourn and W. de Launey, Difference matrices, in *CRC Handbook of Combinatorial Designs*, (C.J. Colbourn and J.H. Dinitz, editors), CRC Press, Boca Raton FL, 1996, pp. 279-289.

[3] C.J. Colbourn, A.C.H. Ling, and M. Tompa, A hillclimbing method for balanced binary codes for oligo arrays, preprint, 2001.

[4] E. Hubbell and P.A. Pevzner, Fidelity probes for DNA arrays, in *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology*, Heidelberg, Germany, August 1999, pp. 113-117.

[5] R.J. Lipshutz, S.P.A. Fodor, T.R. Gingeras, and D.J. Lockhart, High density synthetic oligonucleotide arrays, *Nature Genetics Supplement 21*, 1999, pp. 20-24.

[6] F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error-Correcting Codes*, North Holland, Amsterdam, 1977.

[7] R.A. Mathon and A. Rosa, Balanced incomplete block designs, in *CRC Handbook of Combinatorial Designs*, (C.J. Colbourn and J.H. Dinitz, editors), CRC Press, Boca Raton FL, 1996, pp. 3-41.

[8] W.H. Mills and R.C. Mullin, Coverings and packings, in *Contemporary Design Theory: A Collection of Surveys* (J. H. Dinitz and D. R. Stinson, Eds.), 1992, pp. 371-399, John Wiley & Sons, New York.

[9] R. Sengupta and M. Tompa, Quality control in manufacturing oligo arrays: a combinatorial design approach, in *Pacific Symposium on Biocomputing*, Mauna Lani, Hawaii, January 2001, 348-359. Also Technical Report 2000-08-03, Department of Computer Science and Engineering, University of Washington, November 2000, `ftp://ftp.cs.washington.edu/tr/2000/08/UW-CSE-00-08-03.PS.Z`.