{0,1,*} distance problems in combinatorics

{0,1,*} DISTANCE PROBLEMS IN COMBINATORICS

J.H. van Lint
Eindhoven University of Technology
Department of Mathematics and Computing Science
P.O. Box 513
5600 MB Eindhoven, The Netherlands

1 INTRODUCTION

In this survey we shall treat three different combinatorial problems, all of which are concerned with the ternary alphabet {0,1,*}. In each of the problems it is required to construct a certain "code" of length n, i.e., a subset of $\{0,1,*\}^n$. On this set a distance function d is defined by

$$d(\underline{x},\underline{y}) := |\{i \mid 1 \leq i \leq n, \{x_i,y_i\} = \{0,1\}\}| .$$

Notice that the positions where a word has a * do not contribute to the distance to other words. If C is a code then we define $\bar{C}$ to be the code obtained by replacing each 0 in words of C by a 1, and vice versa.
In the following we shall identify the code C with the matrix (which we also call C) which has the codewords of C as its rows.
All three of the problems which we discuss originated from a practical problem, and nearly all the results are fairly recent.

The first problem is the addressing problem for graphs. A connected graph G is given. We wish to give each vertex of G an address from $\{0,1,*\}^n$ such that the distance between two vertices of G is equal to the distance between the corresponding addresses. What can be said about the minimal value N(G) of n for which such an addressing is possible? This problem was introduced by Graham and Pollak (1971). They pointed out that in telephone communication the design of switching functions is based on the assumption that the holding time of a call is long compared to the time needed to set up the call, but for certain types of communication among computers this is not necessarily true.

This introduces the problem of devising a method by which a message, with its destination at its head, can thread its way through a communication network (in our case, the graph G) without waiting for a complete path to be available before starting on its journey. If we use the addressing described above, then at each vertex the message will continue along an edge going to a vertex for which the distance to the destination will decrease. This means that a message will always go along one of the shortest paths to the destination. We shall review some of the theory and lower bounds for N(G) developed by Graham & Pollak. They conjectured that $N(G) \leq |G| - 1$. This conjecture was proved recently by Winkler (1983). The proof will be described, and finally some recent results on the similar problem for directed graphs (Chung et al. 1984) will be given.

Our second problem concerns the construction of so-called associative block designs introduced by Rivest (1974). An ABD(k,w) is a code C of cardinality $b := 2^w$ in $\{0,1,*\}^k$, such that

(i) each row has $k-w$ stars;
(ii) each column has $b(k-w)/k$ stars;
(iii) any two different rows have distance at least one.

The origin of the problem is the following. Consider a file of k-bit binary words. Each word in $\{0,1,*\}^k$ is a partial match query. The partial match retrieval problem is to retrieve from the file all words agreeing with the query in those positions where the query specifies a bit. So-called hash-coding schemes divide a file into b disjoint lists $L_1, L_2, \ldots, L_b$. A record x will be stored in the list with index h(x), where h is the "hash-function" mapping $\{0,1\}^k$ onto $\{1,2,\ldots,b\}$. For a given partial match query some of the lists must be searched. An analysis of the worst-case number of lists searched led to the concept of ABD. In this case h(x) is the index of the unique row of C which has distance 0 to x. The problem can also be formulated in a geometrical way by asking for a packing of the k-dimensional affine space AG(k,2) with $(k-w)$-flats. There are several interesting open problems concerning associative block designs, but it seems that nothing has been done on them since the paper by Brouwer (1976) appeared. Hopefully interest in these problems will be revived.

The third problem which we discuss arose in connection with comma-free codes. A q-ary code D of length n is said to be comma-free if, for every pair of words $(a_1,\ldots,a_n)$ and $(b_1,\ldots,b_n)$ of D, the words $(a_k,a_{k+1},\ldots,a_n,b_1,\ldots,b_{k-1})$, $(k = 2,3,\ldots,n)$, are not in D. Golomb et al. (1958) studied the problem of finding the maximal cardinality of such a code. For even n this led to the subproblem of finding comma-free codes in which every word is a cyclic shift of a word of the form $(a,0,\ldots,0,b,0,0,\ldots,0)$, where a and b are symbols of the alphabet, separated by $\frac{1}{2}n-1$ zeros (cf. Jiggs 1963). A complete reformulation of this problem results in a problem which fits into this survey. Consider a code C in $\{0,1,*\}^n$, such that two different codewords have distance at least 1 and furthermore

$$\forall_{\underline{a},\underline{b}\in C}\ \forall_{i,j}\left[\begin{pmatrix} a_i & a_j \\ b_i & b_j \end{pmatrix} \neq \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\right] .$$

For such a code we can define

$$\underline{a} \to \underline{b} \quad \text{if there is a } j \text{ such that } (a_j,b_j) = (0,1) ,$$

and then the condition implies that the codewords are the vertices of a tournament. We call such a code C a tournament code. The problem we are interested in is studying the function

$$t(k) := \max\{|C| \mid C \text{ a tournament code of length } k\} .$$

The best lower and upper bound for t(k), presently known, will be given.

2 THE ADDRESSING PROBLEM FOR GRAPHS

2.1 Introduction, examples

Let G be a connected graph. We wish to give each vertex of G an "address", where the address is a word in $\{0,1,*\}^n$. We require that the distance between two vertices in the graph is equal to the distance between the corresponding addresses. It is trivial that this can be done if n is sufficiently large. We define

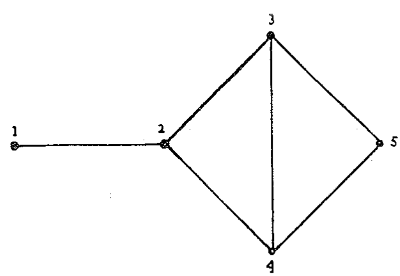
$$N(G) := \text{minimal value of } n \text{ for which an addressing of } G \text{ exists.} \tag{2.1.1}$$

The example K_3 shows that it is not possible to consider the similar problem using the binary alphabet {0,1}. A simple class of graphs for which we can do without the stars consists of the trees. Let G be a tree with vertices $x_0, x_1, \ldots, x_k$, where x_0 is an endpoint. Remove x_0. Suppose we have an addressing for the remaining graph in which x_i has address $\underline{x}_i \in \{0,1,*\}^n$, $(i = 1,2,\ldots,k)$. If we give x_0 the address $(1,\underline{x}_1)$ of length $n+1$, and change each address $\underline{x}_i$ to $(0,\underline{x}_i)$, then we have an addressing for G. This shows that for a tree G, we have $N(G) \leqslant |G| - 1$, and in fact we do not need the symbol *. In Section 2.2 we shall show that $N(G) = |G| - 1$ for a tree.

An even simpler example is the complete graph K_m. Consider the identity matrix of size $m-1$. Replace the zeros above the diagonal by stars and add a row of zeros. The rows of the resulting matrix pairwise have distance 1. So $N(K_m) \leqslant m-1$. We shall see in Section 2.2 that a shorter addressing is not possible.

As a final example consider the graph of Figure 1.

Figure 1



A possible (but not optimal) addressing is

1	1	1	1	*	*
2	1	0	*	1	*
3	*	0	0	0	1
4	0	0	1	*	*
5	0	0	0	0	0

.

2.2 A lower bound

Graham & Pollak (1971) introduced a correspondence between addressings of a graph and quadratic forms, which led to several lower bounds for N(G). This very nice theory deserves to be better known. We illustrate the correspondence with the graph of Figure 1. To the first column of the addressing we associate the product $(x_1 + x_2)(x_4 + x_5)$. Here x_i is in the first (resp. second) factor if the address of vertex i has a 1 (resp. 0) in the first column. If we do the same for the other columns and then add the terms, we obtain the quatratic form $\Sigma\, d_{ij}\, x_i\, x_j$, where d_{ij} is the distance from vertex i to vertex j in G. Thus an addressing corresponds to writing $\Sigma\, d_{ij}\, x_i\, x_j$ as a sum of n products $(x_{i_1} + \dots + x_{i_k})(x_{j_1} + \dots + x_{j_\ell})$ such that no x_i occurs in both of the factors; (the number of variables is $|G|$). Another way of expressing this is to say that G is partitioned into complete bipartite graphs $K_{k,\ell}$. A trivial way of doing this is to take the first factor to be a monomial. The resulting addressing has length at most $d(|G| - 1)$, where d is the diameter of G.

(2.2.1) Theorem. *Let* n_+ *resp.* n_- *be the number of positive resp. negative eigenvalues of the distance matrix* (d_{ij}) *of the graph* G. *Then* $N(G) \geqslant \max\{n_+, n_-\}$.

Proof. $(x_1 + \dots + x_m)(x_{m+1} + \dots + x_\ell) = \frac{1}{4}\{(x_1 + \dots + x_\ell)^2 - (x_1 + \dots + x_m - x_{m+1} - \dots - x_\ell)^2\}$. An addressing therefore represents $\Sigma\, d_{ij}\, x_i\, x_j$ as the difference of two sums of n squares of linear forms. By Sylvester's law $n \geqslant \max\{n_+, n_-\}$. ∎

(2.2.2) Theorem. $N(K_m) = m-1$.

Proof. In Section 2.1 we saw that $m-1$ is an upper bound. Since $J-I$ of size m is the distance matrix of K_m and this matrix has $n_+ = 1$, $n_- = m-1$, the result follows from the previous theorem. ∎

Several other simple proofs of Theorem 2.2.2 are known (cf. Peck 1984). We shall now show that $N(G)$ is $|G| - 1$ for a tree. Here, one would also expect a simple proof (by induction for example) but the method of Graham & Pollak seems to be the easiest.

(2.2.3) Theorem. *If* G *is a tree with* n *vertices, then* $N(G) = n-1$.

Proof. We first calculate the determinant of the distance matrix (d_{ij}) of G. Number the vertices in such a way that x_n is an endpoint adjacent to x_{n-1}. Subtract row $n-1$ from row n (and similarly for columns). Then all the entries in the new last row and column are 1 except the diagonal element, which is -2. Now, renumber the vertices x_1 to x_{n-1} in such a way that the new vertex with number x_{n-1} is an endpoint of $G\backslash\{x_n\}$ and respeat the procedure. After $n-1$ steps we have the determinant

$$\begin{vmatrix} 0 & 1 & 1 & \cdot & \cdot & \cdot & 1 \\ 1 & -2 & 0 & \cdot & \cdot & \cdot & 0 \\ 1 & 0 & -2 & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & & \cdot & & & \cdot \\ \cdot & \cdot & & & \cdot & & \cdot \\ \cdot & \cdot & & & & \cdot & 0 \\ 1 & 0 & \cdot & \cdot & \cdot & 0 & -2 \end{vmatrix},$$

which is $D_n := (-1)^{n-1}(n-1)2^{n-2}$. So, the determinant of the distance matrix depends only on the number of vertices of the tree. If we number the points of the tree G according to the procedure described above, then the k by k principal minor in the upper left-hand corner of the distance matrix is the distance matrix of the subtree on x_1 to x_k. Therefore the sequence $1,D_1,D_2,\ldots,D_n$, where D_k is the determinant of the k by k minor, is equal to $1,0,-1,4,\ldots,(-1)^{n-1}(n-1)2^{n-2}$. If we call

0 positive, then the sequence has only one occurrence of two consecutive terms of the same sign. By a theorem on quadratic forms (cf. Jones 1950, p. 9) the distance matrix (d_{ij}) has one positive eigenvalue and $n-1$ negative eigenvalues. The result again follows from Theorem 2.2.1. ∎

By the same method Graham & Pollak showed that if G is a cycle on $2\ell+1$ vertices, then $N(G) = 2\ell$. In their paper they describe an algorithm for finding an addressing of a given graph G. In all the examples which they tried, N(G) was never more than $|G| - 1$. This led to the conjecture that $N(G) \leqslant |G| - 1$ for all graphs G. The proof of this conjecture is the topic of the next section.

2.3 An upper bound

In this section we describe the proof by Winkler (1983) that $N(G) \leqslant |G| - 1$ for all graphs G. Consider the graph G of Figure 2 (which is the same graph Winkler uses as example).

Figure 2

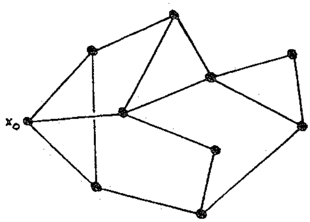
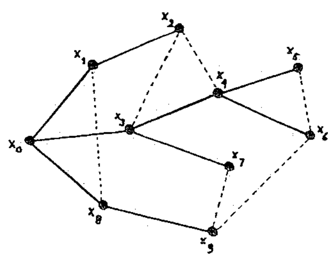


Figure 3



Fix some vertex as x_0. The first step in the algorithm is to find a spanning tree T of G, which preserves the distance to x_0. This is done by a breadth-first search. The second step is to number the remaining vertices by a depth-first search of T. This results in Figure 3 (edges of G\T are dotted).

Next, we need some notation and definitions. Let $|G| = n+1$.

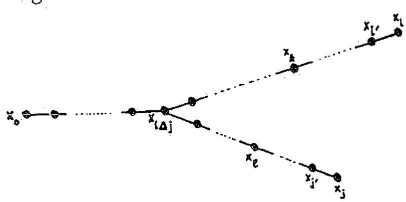
$$\text{For } i \leqslant n \text{ let } P(i) := \{j \mid x_j \text{ is on the path from } x_0 \text{ to } x_i \text{ in } T\} . \quad (2.3.1)$$

(E.g. $P(6) = \{0,3,4,6\}$ in Figure 3.)

$$i \,\Delta\, j := \max(P(i) \cap P(j)) . \quad (2.3.2)$$

The general situation in T is described by Figure 4. Note that $i < j \Leftrightarrow k < \ell$.

Figure 4



$$\text{For } i \leqslant n \text{ define } i' := \max(P(i) \setminus \{i\}) . \quad (2.3.3)$$

(E.g. $5' = 4$, $7' = 3$ in Figure 3.)

$$\text{Define } i \sim j \text{ iff } P(i) \subset P(j) \text{ or } P(j) \subset P(i) . \quad (2.3.4)$$

Let d_G and d_T be the distance functions in G and T respectively. We define the <u>discrepancy</u> function $c(i,j)$ by

$$c(i,j) := d_T(x_i,x_j) - d_G(x_i,x_j) . \quad (2.3.5)$$

(E.g. in Figure 3 we have $c(6,9) = 4$.)

(2.3.6) <u>Lemma</u>. (i) $c(i,j) = c(j,i) \geqslant 0$;

(ii) *if* $i \sim j$, *then* $c(i,j) = 0$;

(iii) *if* $i \not\sim j$, *then* $c(i,j') \leqslant c(i,j) \leqslant c(i,j') + 2$.

Proof. (i) is trivial; (ii) follows from the definition of T, since $d_G(x_i,x_j) \geq |d_G(x_j,x_0) - d_G(x_i,x_0)| = d_T(x_i,x_j)$; (iii) follows from $|d_G(x_i,x_j) - d_G(x_i,x_j')| \leq 1$ and the fact that $d_T(x_i,x_j) = 1 + d_T(x_i,x_j')$. ∎

For $0 \leq i \leq n$ the vertex x_i is given the address $\underline{a}_i \in \{0,1,*\}^n$, where $\underline{a}_i = (a_i(1), a_i(2), \ldots, a_i(n))$ and

$$a_i(j) := \begin{cases} 1 \text{ if } j \in P(i)\,; \\ * \text{ if } (a)\ c(i,j) - c(i,j') = 2, \text{ or} \\ \quad\ \ (b)\ c(i,j) - c(i,j') = 1,\ i<j,\ c(i,j) \text{ even or} \\ \quad\ \ (c)\ c(i,j) - c(i,j') = 1,\ i>j,\ c(i,j) \text{ odd}\,; \\ 0 \text{ otherwise}. \end{cases} \tag{2.3.7}$$

(2.3.8) Theorem. $d(\underline{a}_i,\underline{a}_k) = d_G(x_i,x_k)$.

Proof. Since the case $i = k$ is trivial, we may assume $i < k$. Suppose that $i \sim k$. Then $d_G(x_i,x_k) = |P(k)\backslash P(i)|$. The values of j s.t. $j \in P(k)\backslash P(i)$ are exactly the ones where $a_k(j) = 1 \neq a_i(j)$. However, for these j we see that $c(i,j) = 0$. So, in this case we are done. It remains to consider the case $i \not\sim k$. The key observation is the following. Let $n_1 \leq n_2 \leq \ldots \leq n_\ell$ be an increasing sequence of integers s.t. $|n_{i+1} - n_i| \leq 2$. If m is an even integer between n_1 and n_ℓ which does not occur in the sequence, then there is an i such that $n_i = m-1$, $n_{i+1} = m+1$. Now consider the sequence

$$c(i,k) \geq c(i,k') \geq c(i,k'') \geq \ldots \geq c(i,i \Delta k) = 0\,.$$

By (2.3.7) and the observation above, $a_i(j) = *$ and $a_k(j) = 1$ exactly as many times as there are even integers between $c(i,i\Delta k)$ and $c(i,k)$. Similarly $a_k(j) = *$ and $a_i(j) = 1$ as many times as there are odd integers between $c(i,i\Delta k)$ and $c(i,k)$. Therefore (by (2.3.7))

$$d(\underline{a}_i,\underline{a}_k) = |P(k)\backslash P(i)| + |P(i)\backslash P(k)| - c(i,k) = $$
$$= d_T(x_i,x_k) - c(i,k) = d_G(x_i,x_k)\,.$$ ∎

2.4 Directed graphs

The analogous problem for directed graphs has not been settled. Of course, for the problem to make sense we must consider a digraph G (with vertices $x_1, x_2, \ldots, x_n$) such that for any pair of vertices x_i, x_j there is a directed path from x_i to x_j (i.e., a strongly connected digraph). We must modify our definition of distance of addresses (since distance is no longer symmetric in G). If x_i has address $\underline{a}_i \in \{0,1,*\}^n$, $(i = 1,2,\ldots,n)$, then we require that the distance from x_i to x_j in G is equal to the number of coordinate places k such that $a_i(k) = 0$, $a_j(k) = 1$. The number N(G) is defined as before.

As in the case of undirected graphs, it is not difficult to show that $N(G) \leqslant n(n-1)$ (by a trivial addressing scheme). We give one theorem which shows that the situation is really much more difficult than in the previous sections. This theorem is due to Chung et al. (1984).

(2.4.1) Theorem. *There is a constant* $c > 0$ *such that* $N(G) > cn^{3/2}$ *if* G *is a cycle on n vertices.*

Proof. Let G be a cycle on n vertices and suppose that the code $C \subset \{0,1,*\}^{\ell}$ is an addressing. We define $S_i := \{k \mid (c_{i+1,k}, c_{ik}) = (0,1)\}$. Since $d_{i+1,i} = n-1$, we have $|S_i| = n-1$ for $1 \leqslant i \leqslant n$. Similarly $d_{i+1,i+j} = j-1$ implies that $|S_i \cap S_{i+j}| \leqslant j-1$. Therefore we have, for $m < n$,

$$\ell \geqslant |S_1 \cup S_2 \cup \ldots \cup S_m| \geqslant \sum_{i \leqslant m} |S_i| - \sum_{i<j\leqslant m} |S_i \cap S_j| \geqslant$$

$$\geqslant m(n-1) - \sum_{i=1}^{m-2} i(m-1-i) = m(n-1) - \frac{1}{6}m^3 + \frac{1}{2}m^2 - \frac{1}{3}m .$$

The best choice for m is $m = \lfloor \sqrt{2n} \rfloor$. This yields the required result with $c = \frac{2}{3}\sqrt{2}$. ∎

In the same paper mentioned above, Chung et al. show that a cycle on n vertices can be addressed with an address of length $k \leqslant cn^{5/3}(\log n)^{1/3}$.

We close this section by returning to the practical origin of our mathematical problem. We have shown that it is difficult for a message to thread its way through a directed circuit!

3 ASSOCIATIVE BLOCK DESIGNS

3.1 Introduction

We recall that a code C in $\{0,1,*\}^k$ is identified with the $|C|$ by k matrix which has the words of the code as its rows.

(3.1.1) Definition. An ABD(k,w) is a rectangular matrix with $b = 2^w$ rows and k columns, with entries from {0,1,*}, such that

(i) each row has $k-w$ stars,

(ii) each column has $b(k-w)/k$ stars,

(iii) any two rows have distance at least 1.

We can formulate (i) and (ii) as follows: the stars form a 1-design. Condition (iii) implies that the subsets obtained from the rows by replacing stars by 0 or 1 in all possible ways are disjoint $(k-w)$-flats in $AG(k,2) = \{0,1\}^k$. We formulate this as follows:

$$\text{Each word in } \{0,1\}^k \text{ is represented by exactly one row of } C\,. \tag{3.1.2}$$

(This means that the word has distance 0 to the row of C.)
The concept of an associative block design was introduced by Rivest (1974), who gave some restrictions on the parameters (k,w) and some constructions. Several other restrictions and constructions were found by the combinatorial theory seminar of the Mathematical Center in Amsterdam. A report on this work was written by Brouwer (1976). In the following we survey the most important results of these two papers. But first we give two examples.

Consider the code C of length 4 given by

*	0	0	0
0	*	1	0
0	0	*	1
0	1	0	*

.

The rows of C represent all the words of weight < 2 in $\{0,1\}^4$ and also those of weight 2 with a 0 in the first position.

Therefore the rows of $\bar{C}$ represent the words of weight 2 with a 1 in the first position and all the words of weight > 2. So C and $\bar{C}$ together form an ABD(4,3). A different example is obtained if, instead of $\bar{C}$, we take the matrix obtained from C by only complementing the first row and column. (In both cases, (3.1.1) (i) and (ii) are clearly satisfied.)

3.2 Restrictions on parameters

From the construction of the examples in Section 3.1 it is obvious that they have as many zeros as ones in each column. This is always the case.

(3.2.1) Theorem. *If an* ABD(k,w) *exists, then*

(i) *it has exactly* bw/(2k) *zeros and* bw/(2k) *ones in each column;*

(ii) *for each* $\underline{x} \in \{0,1\}^k$ *it has exactly* $\binom{w}{u}$ *rows which agree with* $\underline{x}$ *in* u *positions;*

(iii) *the parameters satisfy*

$$w^2 \geqslant 2k(1 - \frac{1}{b}) \quad .$$

Proof. Let C be the ABD(k,w).

(i) A row of C with a star (resp. a 0) in column j represents 2^{k-w-1} words (resp. 2^{k-w} words) in $\{0,1\}^k$ with a 0 in position j. It follows from (3.1.2) and (3.1.1) that each column has bw/(2k) zeros.

(ii) Let $\underline{x} \in \{0,1\}^k$. Denote by n_i the number of rows of C which agree with $\underline{x}$ in i places. There are $\binom{k}{\ell}$ words in $\{0,1\}^k$ which agree with $\underline{x}$ in exactly ℓ positions. Therefore $\binom{k}{\ell} = \Sigma\, n_i \binom{k-w}{\ell-i}$, i.e.,

$$(1+x)^k = (1+x)^{k-w} \cdot \Sigma\, n_i x^i \ .$$

This shows that $n_i = \binom{w}{i}$.

(iii) The sum of the distances between pairs of rows of C is $k(\frac{bw}{2k})^2$ by (i). By (3.1.1) (iii) this sum is at least $\binom{b}{2}$. ∎

Note that (3.2.1) (i) implies that an ABD(k,w) cannot exist if k does not divide $w \cdot 2^{w-1}$.

In the construction of ABD(4,3) in Section 3.1 the star pattern in the lower half is equal to the pattern in the upper half. It was observed by P. van Emde Boas that this is also true in general.

(3.2.2) Theorem. *Let* $w > 0$. *Let* C *be an* ABD(k,w). *In the* 1*-design formed by the stars of* C *each block occurs an even number of times.*

Proof. The argument is the same as in (3.2.1) (i). Count the words in $\{0,1\}^k$ which have zeros in the positions of the stars of a given row of C. Each row with a different star pattern represents an even number of these; the row itself only one. ∎

Brouwer (1974) gave other necessary conditions by counting more complicated configurations. He also obtained the following strengthening of (3.2.1) (iii).

(3.2.2) Theorem. *Let* C *be an* ABD(k,w) *with* $w > 3$.

(i) *If two rows of* C *agree in all but one position, then*

$$\binom{w}{2} \geqslant k \ ;$$

(ii) *otherwise* $w^2 > 2k$.

Proof.

(i) Suppose $\underline{c}_1$ and $\underline{c}_2$ are two rows of C which differ only in position one. Then all the other rows of C must differ from $\underline{c}_1$ in some other position. So, by (3.1.1) (i) and (3.2.1) (i), we find

$$b-2 \leqslant (w-1) \cdot \frac{bw}{2k} .$$

To prove the assertion, we must show that the right-hand side cannot be equal to $b-2$ or $b-1$. In both cases equality would imply that $2^{w-1} | k$ which contradicts (3.2.1) (iii) unless $w = 4$. The value $w = 4$ is excluded by substitution.

(ii) Consider two rows of C which have the same star pattern. By hypothesis they differ in more than one position. Again, count the sum of the distances of all the rows from one of this pair. This sum is at least $2 + (b-2) = b$ and by (3.2.1) (i) it is equal to $w \cdot (bw)/(2k)$. So $w^2 \geqslant 2k$. We must show that equality does not hold. By the argument above, equality would imply that rows with the same star pattern occur in pairs which have distance 2, and furthermore all the other rows have distance 1 to each row of such a pair. W.l.o.g. such a pair would be

$$(* * \ldots * 0\,0 \ldots 0\,0\,0) \quad \text{and} \quad (* * \ldots * 0\,0 \ldots 0\,1\,1) .$$

The $bw/(kw) - 1$ rows ending in a 1 would have to end in 0 1, for otherwise they would have distance 0 to the second row or distance > 1 to the first row. Similarly there would be $bw/(2k) - 1$ rows ending in 1 0. Since we now have rows with distance 2, we find that $bw/(2k) - 1 = 1$. Therefore $2^w = 2w$, which is impossible if $w \geqslant 3$. ∎

(3.2.4) <u>Corollary</u>. *An* ABD(8,4) *does not exist.*

Using these results it is easy to find all ABD(k,w) with $w \leqslant 4$. Of course, $w = 0$ is trivial. For $w = 1$, 2 or 4 we must have $k = w$ (no stars). If $w = 3$ then $k = 3$ (no stars) or $k = 4$ and then there are two types, both given in Section 3.1. It is not known whether an ABD(8,5) or an ABD(10,5) exists.

3.3 Constructions

We shall present a few construction methods to make new ABD's out of a given one. Brouwer (1974) remarks that all known ABD's can be constructed in this way from the trivial ABD(k,0) and ABD(k,k) and the two examples of Section 3.1.

(3.3.1) Theorem. *If an* $ABD(k_i,w_i)$ *exists* $(i = 1,2)$, *then an* $ABD(k_1k_2,w_1w_2)$ *exists.*

Proof. We may suppose $w_2 = 0$. Partition the rows of $ABD(k_2,w_2)$ into two classes R_0 and R_1 of the same size. In $ABD(k_1,w_1)$ we replace each star by a row of k_2 stars, each 0 by a row from R_0 and each 1 by a row from R_1 in all possible ways. A trivial calculation shows that the resulting design is an $ABD(k_1k_2,w_1w_2)$. ∎

(3.3.2) Corollary. *An* $ABD(4^t,3^t)$ *exists for* $t \geq 1$.

For the second construction method we need a theorem on 1-designs. If A is the b by v matrix of a 1-design with row sum k and if b = v, then by a well-known theorem of Birkhoff we can delete ones from A in such a way that a new 1-design with row sum $k-1$ is obtained. This theorem was generalized by A. Schrijver in the following way.

(3.3.3) Theorem. *Let* A *be the* b *by* v *incidence matrix of a 1-design with row sum* k *and column sum* r *(so* $bk = vr$*). If* $k_0 \leq k$ *and* $r_0 \leq r$ *and* $bk_0 = vr_0$, *then* A *is the sum of two (0,1)-matrices* A_1 *and* A_2, *where* A_1 *is the incidence matrix of a 1-design with row sum* k_0.

Proof. Consider the following graph (Figure 5).

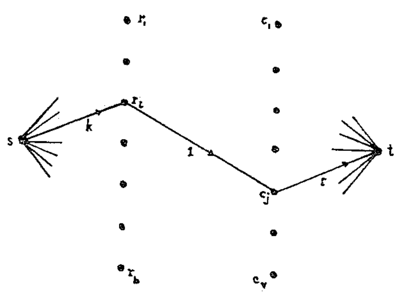


Figure 5

The vertex s is joined to r_i $(1 \leq i \leq b)$, t is joined to c_j $(1 \leq j \leq v)$, r_i is joined to c_j iff $a_{ij} = 1$. We consider this as a network in which the edges from s have capacity k, edges to t have capacity r, and all other edges have capacity 1. There is a flow with value bk = vr through this network (by definition). Now, reduce the capacities of the edges from s to k_0 and the capacities of the edges to t to r_0. Reduce all flows by the factor k_0/k. We again have a maximal flow. The total flow and the capacities are all integers, so there is a maximal flow in which all flows are integers. The matrix A_1 has ones in the positions (i,j) for which the flow in the edge (r_i,c_j) is 1. ∎

For the proof of the following theorem we introduce a new symbol –. A word with c symbols – represents the 2^c words obtained by replacing each – by 0 or 1 in all possible ways.

(3.3.4) Theorem. *Let* $w > 0$. *Suppose an* ABD(k,w) *exists, where* $k = k_0 \cdot 2^\ell$ (k_0 *odd*). *Then* ABD$(k, w + ik_0)$ *exists for* $0 \leq i \leq (k-w)/k_0$.

Proof. It is sufficient to consider i = 1. Let A be the incidence matrix of the 1-design formed by the stars in ABD(k,w). By Theorem 3.3.3 $A = A_1 + A_2$, where A_1 is the incidence matrix of a 1-design with k_0 stars in each row, $2^{w-\ell}$ stars in each column. The stars corresponding to A_1 are replaced by the symbol –. This produces the required ABD$(k, w + k_0)$. ∎

(3.3.5) Theorem. *If* ABD(k,w) *exists and* $\alpha \geq 1$ *is a number such that* αk *and* αw *are integers, then* ABD$(\alpha k, \alpha w)$ *exists.*

Proof. It is sufficient to show that ABD$(k+\ell, w+m)$ exists for $(k+\ell)/(w+m) = k/w$ and $(\ell, m) = 1$. Again, let $k = k_0 \cdot 2^e$ (k_0 odd). From (3.1.1) (ii) we know that $k_0 | w$. Therefore $w\ell = mk$ and $(\ell,m) = 1$ imply that ℓ is a power of 2. Consider the ℓ by ℓ circulant matrix with a row of $\ell - m$ stars and m minus signs as first row. Since ℓ divides b, we can adjoin b/ℓ copies of this circulant to the matrix C of the ABD. The columns which have been added all have $\frac{b}{\ell}(\ell - m) = \frac{b}{k}(k-w)$ stars. So, we have obtained the required ABD$(k+\ell, w+m)$. ∎

We give an example of a construction based on these theorems. From Corollary 3.3.2 we have ABD(64,27). Theorem 3.3.4 shows that ABD(64,w) exists for $27 \leqslant w \leqslant 64$, in particular for $w = 32$. Then Theorem 3.3.5 guarantees the existence of ABD(2w,w) for $w \geqslant 32$.

Section 3 contains no information that was not known about ten years ago. Hopefully some reader will get sufficiently interested in the subject to add some new knowledge on ABD's to combinatorics.

4 {0,1,*}-TOURNAMENT CODES

4.1 Introduction

We repeat the definition of a {0,1,*} tournament code C of length k.

(4.1.1) Definition. A code C of length k over the alphabet {0,1,*} is called a tournament code if, for any two distinct codewords $\underline{a}$, $\underline{b}$, exactly one of the following two conditions is true:

(i) $\exists_j((a_j,b_j) = (0,1))$,
(ii) $\exists_j((a_j,b_j) = (1,0))$.

If (i) holds we shall say $\underline{a} \to \underline{b}$; (this defines the tournament).

(4.1.2) Definition. The maximal value of $|C|$ over all tournament codes of length k is called t(k).

As usual C also denotes the matrix which is a list of the codewords. If $|C| = t(k)$, we call the code optimal.

(4.1.3) Lemma. *For every* $k \in \mathbb{N}$ *there is an optimal code* C *of length* k *with* $\underline{0} \in C$, $\underline{1} \in C$.

Proof. If C is optimal and $\underline{0} \notin C$ then clearly C must contain a word with distance 0 to $\underline{0}$. Replace this word by $\underline{0}$ to obtain a new optimal code. Similarly for $\underline{1}$. ∎

The following lemma is trivial.

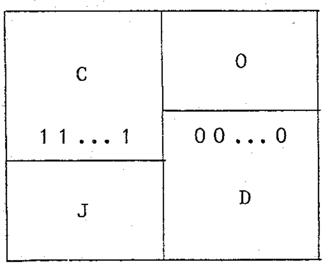
(4.1.4) Lemma. *If* C *is optimal then* $\bar{C}$ *is optimal.*

Clearly t(k) is strictly increasing.

In order to find lower bounds for t(k), we describe a construction which produces a long tournament code from two shorter ones.

(4.1.5) Theorem. $t(k+\ell) \geqslant t(k) + t(\ell) - 1$.

Proof. Let C be optimal of length k and let $\underline{0}$ be the top row of C and $\underline{1}$ the bottom row. Similarly with D for length ℓ. Consider the code



.

This is a tournament code of length $k+\ell$ and cardinality $t(k) + t(\ell) - 1$. ∎

(4.1.6) Corollary. $t(nk) \geqslant 1 + n\{t(k) - 1\}$.

This shows that $\lim_{k\to\infty} k^{-1}t(k)$ exists (possibly ∞). For a while it was believed that this limit was 2 until Golomb & Tang (1983) found that $t(7) = 16$. In fact the limit is ∞ as we shall see in Section 4.3.

4.2 An upper bound

It was first shown by R.L. Graham (unpublished) that there is a $c > 0$ such that $t(k) < k^{c \log k}$ for $k > 1$. Below we shall give a simple proof of this bound. The proof is due to C.L.M. van Pul (unpublished).

Let K be a {0,1,*} tournament code of length k. (We may assume $\underline{0} \in K$, $\underline{1} \in K$.) By permuting rows and columns K can be put in the following "standard form".

$$K = \left[\begin{array}{c|c|c} & \longleftarrow \ell \longrightarrow & \longleftarrow k-\ell-1 \longrightarrow \\ 0 & & \\ 0 & & \\ \vdots & A & B \\ 0 & & \\ \hline 1 & & \\ 1 & & \\ \vdots & C & D \\ 1 & & \\ \hline * & & \\ * & & \\ \vdots & E & F \\ * & & \end{array}\right]$$

Here, every column of A contains a 1 but no column of B contains a 1. From 4.1.1 and the first column of K we see that no column of C has a 0. This shows that K or $\bar{K}$ has a standard form with $\ell \leqslant \lfloor\frac{k-1}{2}\rfloor$.

(4.2.1) Theorem. $t(k) \leqslant t(k-1) + t(\lfloor\frac{k-1}{2}\rfloor)$.

Proof. Let K be optimal and in standard form with $\ell \leqslant \lfloor\frac{k-1}{2}\rfloor$. By the definition of B the matrix A is a tournament code of length ℓ. So A has at most $t(\lfloor\frac{k-1}{2}\rfloor)$ rows. Clearly $\binom{C\ D}{E\ F}$ is a tournament code of length $k-1$. The result follows. ∎

(4.2.2) Corollary. $t(k) \leqslant \lceil\frac{k+1}{2}\rceil\ t(\lfloor\frac{k-1}{2}\rfloor)$.

Proof. This follows by repeated application of Theorem 4.2.1 and the fact that t is an increasing function. ∎

(4.2.3) Theorem. $t(k) < k^{\log k}$ *for* $k > 2$ (*logarithm to base* 2).

Proof. Trivially t(1) = 2 and t(2) = 3. Application of Theorem 4.2.1 shows that the assertion of 4.2.3 is true for $k < 7$. Suppose $k \geq 7$. We use induction. By Corollary 4.2.2 we have

$$t(k) \leq \lceil \frac{k+1}{2} \rceil \, t(\lfloor \frac{k-1}{2} \rfloor) \leq 2 \lfloor \frac{k-1}{2} \rfloor \left(\lfloor \frac{k-1}{2} \rfloor^{\log \lfloor \frac{k-1}{2} \rfloor} \right)$$

$$\leq 2 (\frac{k-1}{2})^{\log(k-1)} = \frac{2}{k-1} \cdot (k-1)^{\log(k-1)} < k^{\log k} . \blacksquare$$

4.3 A lower bound

Until recently the best known lower bound for t(k) was based on Corollary 4.1.6 and the following example.

(4.3.1) Example. Consider the following words of length 7:

$\underline{a} = (1\,0\,0\,*\,0\,*\,*)$,

$\underline{b} = (1\,*\,1\,1\,0\,*\,*)$.

Number the positions from 0 to 6 (mod 7). Note that if $(a_i, a_{i+k}) = (1,0)$, then $k \in \{1,2,4\}$ and the same holds for $\underline{b}$. This implies that if $(a_i, a_{i+j}) = (0,1)$, then $j \in \{6,5,3\}$ and the same holds for $\underline{b}$. Also note that these values of k and j actually occur. Consider two words $\underline{c}$, $\underline{d}$ which are both cyclic shifts of $\underline{a}$ or $\underline{b}$. If $\begin{pmatrix} c_i & c_j \\ d_i & d_j \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, then $j - i \in \{1,2,4\} \cap \{6,5,3\}$ which is absurd. So, $\underline{c}$ and $\underline{d}$ satisfy the condition that $\underline{c} \to \underline{d}$ and $\underline{d} \to \underline{c}$ are not both true. We claim that $d(\underline{c},\underline{d}) \geq 1$. If $\underline{c}$ and $\underline{d}$ are both cyclic shifts of $\underline{a}$ (or of $\underline{b}$) then this follows from the fact that $\{1,2,4\} \cup \{6,5,3\}$ contains all possible non-zero shifts. Now consider $\underline{a}$ and $\underline{b}$. If $(a_i, b_{i+k}) = (1,0)$, then $i = 0$, $k = 4$. If $(a_i, b_{i+k}) = (0,1)$ then we have the possibilities $k \in \{6,1,2\}$, $k \in \{5,0,1\}$, $k \in \{3,5,6\}$, all of them occurring. Since all values of k in $\{0,1,\ldots,6\}$ occur, we see that $\underline{c}$ and $\underline{d}$ indeed have distance ≥ 1. It follows that $\underline{0}$, $\underline{1}$ and all cyclic shifts of $\underline{a}$ and $\underline{b}$ form a tournament code of length 7 with 16 codewords. With Corollary 4.1.6 this gave the bound $t(7k) \geq 15k + 1$.

Recently three of my students (F. Abels, W. Janse, J. Verbakel) found three words of length 13 which behave in the same way as the words $\underline{a}$, $\underline{b}$ of Example 4.3.1. Therefore they could show

$$t(13) \geqslant 41 \ . \qquad (4.3.2)$$

It turns out that these two examples are the first two in an infinite sequence which was found by Collins et al. (1984). The essence of the idea is given in Example 4.3.1.

(4.3.3) Theorem. *For* $n \in \mathbb{N}$ *we have*

$$t(n^2+n+1) \geqslant n(n^2+n+1) + 2 \ .$$

Proof. We shall construct a code C consisting of $\underline{0}$, $\underline{1}$ and all the cyclic shifts of the words of a set $\{\underline{a}^0, \underline{a}^1, \dots, \underline{a}^{n-1}\}$. To define these words we number the positions with the integers mod n^2+n+1, starting with -1 (i.e., for the front position). The positions $\neq -1$ will have their number written in the $(n+1)$-ary system. So (x,y) denotes position $(n+1)x + y$. Therefore $0 \leqslant x \leqslant n-1$, $0 \leqslant y \leqslant n$. The definition of the words $\underline{a}^i$ is as follows:

(i) For each i take $a^i_{-1} = 1$,

(ii) $\underline{a}^i$ has $\begin{cases} 0 & \text{in position } (x,y) \text{ if } x \geqslant i,\ x+y \leqslant n-1, \\ 1 & \text{in position } (x,y) \text{ if } x \leqslant i-1,\ x+y \geqslant n-1, \\ * & \text{otherwise}. \end{cases}$

Just as in Example 4.3.1 the easy part is showing that (4.1.1) (i) and (ii) cannot both hold. We calculate the values of k for which there is a j such that $(a^i_j, a^i_{j+k}) = (1,0)$. Let $k = (\xi,\eta)$.

(a) If $j = -1$, then $i \leqslant \xi \leqslant n-1$, $1 \leqslant \eta \leqslant n$, $\xi + \eta \leqslant n$;

(b) If $j \geqslant 0$, then by our definition we find the pairs $(\xi,\eta) = (x,y) - (x',y')$ satisfying $0 \leqslant \xi \leqslant n-2$, $1 \leqslant \eta \leqslant n$, $\xi + \eta \leqslant n$.

So the set of values k which occur is $K = \{(\xi,\eta) \mid 0 \leqslant \xi \leqslant n-1,$

$1 \leqslant \eta \leqslant n$, $\xi + \eta \leqslant n$. Now look at a pair (a^i_j, a^i_{j+k}) which is (0,1). The possible values of k are found by subtracting the elements of K from $n^2 + n + 1 = (n,1)$. There are two cases:

(a) If $\eta = 1$, then $(n,1) - (\xi,\eta) = (n-\xi, 0)$,

(b) If $\eta > 1$, then $(n,1) - (\xi,\eta) = (n-1-\xi,\ n+2-\eta)$.

It is easily seen that these numbers are the non-zero elements of the complement of K w.r.t. $\{0,1,\dots,n^2+n\}$. Exactly the same argument as we used in Example 4.3.1 shows that if $\underline{c}$ is a cyclic shift of any $\underline{a}^i$ and $\underline{d}$ is a cyclic shift of any $\underline{a}^j$, then $\underline{c} \to \underline{d}$ and $\underline{d} \to \underline{c}$ cannot both hold.

We must now show that for such words $\underline{c}$, $\underline{d}$ one of the relations $\underline{c} \to \underline{d}$ and $\underline{d} \to \underline{c}$ holds. In Example 4.3.1 we saw that this amounts to showing the following. Let $i \leqslant j$. Determine the set K_1 of integers k such that a pair $(a^i_\ell, a^j_{\ell+k}) = (1,0)$ exists. Do the same for (0,1) and call the set of values of k which occur K_2. We must show that if $i = j$, then $K_1 \cup K_2$ contains all non-zero integers mod n^2+n+1 and if $i < j$ it contains these and zero. This is tedious but it is elementary arithmetic.

Just as in Example 4.3.1 the pairs (1,0) are relatively scarce. If $(a^i_\ell, a^j_{\ell+k}) = (1,0)$, then

(a) $\ell = -1$ and $k = (\xi,\eta)$, where $j \leqslant \xi \leqslant n-1$, $1 \leqslant \eta$, $\xi+\eta \leqslant n$,

(b) $k = (\xi,\eta) = (x',y') - (x,y)$, with $x' \geqslant j$, $x \leqslant i-1$, $x'+y' \leqslant n-1$, $x+y \geqslant n-1$, i.e., $n-2 \geqslant \xi \geqslant j-i$, $1 \leqslant \eta \leqslant n$, $\xi+\eta \leqslant n$.

Combining these we find $K_1 = \{(\xi,\eta) \mid j-i \leqslant \xi \leqslant n-1,\ 1 \leqslant \eta \leqslant n,\ \xi+\eta \leqslant n\}$. This situation was easy because the zeros of $\underline{a}^j$ are to the right of the ones of $\underline{a}^i$. For K_2 we have to be more careful. So, we consider $(a^i_\ell, a^j_{\ell+k}) = (0,1)$ and start by assuming that the 0 is to the left of the 1. This is case (i)

(i) we find all $k = (x'-x, y'-y) = (\xi,\eta)$ with $\xi \leqslant j-i-1$. Note that 0 is in this set if $j > i$ and that otherwise this set is empty.

In the remaining two cases the 0 in $\underline{a}^i$ is to the right of the 1 in $\underline{a}^j$. It is easier to calculate k', where $(a^j_\ell, a^i_{\ell+k'}) = (1,0)$. Then

(ii) if $\ell = -1$ we find $k' = (\xi,\eta)$, $i \leqslant \xi \leqslant n-1$, $1 \leqslant \eta \leqslant n$, $\xi+\eta \leqslant n$,

(iii) otherwise $k' = (\xi,\eta) = (x',y') - (x,y) = (x'-x-1, y'-y+n+1)$, so $0 \leqslant \xi \leqslant n-2$, $1 \leqslant \eta$, $\xi+\eta \leqslant n$.

The values of k are $n^2+n+1-k'$. Combining with (i) we find that indeed $K_1 \cup K_2 = \{0,1,\dots,n^2+n\}$ if $i < j$ and $K_1 \cup K_2 = \{1,2,\dots,n^2+n\}$ if $i = j$. ∎

There is a tremendous gap between the upper bound of Section 4.1.2 and the lower bound of Theorem 4.3.3. The upper bound is probably not too good but improving it does not look easy. Probably the best problem to settle first is the question how large a cyclic tournament code can be.

REFERENCES

Brouwer. A.E. (1976), On associative block designs. In Colloquia Mathematica Societatis János Bolyai 18, Combinatorics.

Chung, F.R.K., Graham, R.L. & Winkler, P.M. (1984), On the addressing problem for directed graphs, (preprint).

Collins, K., Shor, P. & Stembridge, J. (1984), Private communication.

Golomb, S.W., Gordon, B. & Welch, L.R. (1958), Comma-free codes. Can. J. Math., 10, 202 - 209.

Golomb, S.W. & Tang, B. (1983), A new result on comma-free codes of even word length, (preprint).

Graham, R.L. & Pollak, H.O. (1971), On the addressing problem for loop switching. Bell System Tech. J., 50, 2495 - 2519.

Jiggs, B.H. (1963), Recent results in comma-free codes. Can. J. Math., 15, 178 - 187.

Jones, B.W. (1950), The Arithmetic Theory of Quadratic Forms. Carus Math. Monograph 10 (MAA).

Peck, G.W. (1984), A new proof of a theorem of Graham and Pollak. Discrete Math., 49, 327 - 328.

Rivest, R.L. (1974), On hash-coding algorithms for partial-match retrieval. Proc. of the 15-th Annual Symposium on Switching and Automata Theory, 95 - 103.

Winkler, P.M. (1983), Proof of the squashed cube conjecture. Combinatorica, 3, 135 - 139.