compression of files of information retrieval systems. With this application in mind, bounds on the average codelength of an alphabetical code were studied.

The major results of this correspondence are as follows.

1) A necessary and sufficient condition for the existence of a binary alphabetical code was given.

2) An upper bound for $L_{\text{opt}}$ (the average codelength of the optimal alphabetical code) was given.

This upper bound shows the redundancy of the optimal alphabetical code in comparison with the Huffman code.

Though this correspondence presents a theoretical bound on $L_{\text{opt}}$, the redundancy of the optimal alphabetical code varies with the distribution of probabilities. To verify the efficiency of the code in practice, the author encoded the descriptors (keywords) of the ERIC thesaurus. There are 8696 descriptors (the average length of a descriptor is about 17 characters), and the alphabet size of the source symbols is 39 (26 capital letters, 10 numeric characters, 2 symbols, and a space character). In this preliminary experiment, $L_{\text{diff}}$ and $L_{\text{opt}}$ are 4.254 bits and 4.423 bits, respectively. The redundancy of the optimal alphabetical code is about 5 percent in comparison with the Huffman coding, which shows the usefulness of the alphabetical code.

REFERENCES


Index System and Separability of Constant Weight Gray Codes

A. J. van Zanten

Abstract — A number system is developed for the conversion of natural numbers to the codewords of the Gray code $G(n,k)$ of length $n$ and weight $k$, and vice versa. As an application sharp lower and upper bounds are derived for the value of $l(n,k)$ where $i$ and $j$ are indices of codewords $g_i$ and $g_j$ of $G(n,k)$ such that they differ in precisely $2m$ bits.

Index Terms — Gray codes, constant weight codes, index system, ranking problem, number system, separability.

I. INTRODUCTION

An $n$-bit Gray code is an ordered sequence of all $2^n$ $n$-bit strings (codewords) such that successive codewords differ by the complementation of a single bit. A Gray code is an example of an ordered code. In this correspondence, the term Gray code stands for the so-called binary-reflected Gray code $G(n)$, $n \geq 1$ (cf., e.g., [11]).

Gray codes are used to minimize the number of erroneous $m$-bit strings, when transmitted as analog signals (cf. [1]). In fact when bit strings are Gray-coded a one-level error in the analog signal causes an error in one bit. More generally the minimum analog error required to generate $m$ bit errors is equal to $2^{n/2}$, as was shown by Yuen in [12]. In [3], Cavior proved that the maximum analog error corresponding to $m$-bit errors equals $2^{n-2^{m/2}}$. So one has sharp bounds for the separability of the code $G(n)$.

Apart from the use made of Gray codes in transmitting information, they also play a role in a number of other mathematical disciplines, such as the theory and construction of minimal-change algorithms to produce various combinatorial objects like permutations, combinations and partitions [2], [4], [11], the analysis of odd–even merging [6], and the theory behind some mathematical puzzles [7].

In many of these applications the question arises of converting a natural number (written in its decimal representation) to its Gray code representation or vice versa of converting a Gray codeword to the integer it represents. If we denote a codeword of $G(n)$ by $g_i$ and let the index $i$ run through the ordered set of integers $0, 1, \cdots, 2^n - 1$, these questions are equivalent to asking for nonrecursive rules that describe the bijective mapping between $i$ and $g_i$. As an example of the Gray code $G(n)$ one encoding the descriptors (keywords) of the ERIC thesaurus. There are 8696 descriptors (the average length of a descriptor is about 17 characters), and the alphabet size of the source symbols is 39 (26 capital letters, 10 numeric characters, 2 symbols, and a space character). In this preliminary experiment, $L_{\text{diff}}$ and $L_{\text{opt}}$ are 4.254 bits and 4.423 bits, respectively. The redundancy of the optimal alphabetical code is about 5 percent in comparison with the Huffman coding, which shows the usefulness of the alphabetical code.

REFERENCES


In Section V, we discuss an application of the index system of \( G(n,k) \), analogous to the results of Yuen and Cavior. We derive sharp lower and upper bounds for the value of \( i-j \), where \( i \) and \( j \) are the indices of codewords \( g_i \) and \( g_j \) of \( G(n,k) \) such that they differ in precisely \( 2m \) bits.

II. PRELIMINARIES

The \( n \)-bit Gray code \( G(n) \) is usually denoted as a \( 2^n \times n \)-matrix

\[
G(n) = \begin{bmatrix}
  g_0 \\
  g_1 \\
  \vdots \\
  g_{2^n-1}
\end{bmatrix},
\]

(1)

where

\[
g_i = \begin{bmatrix}
  s_{i-1} \\
  s_{i-2} \\
  \vdots \\
  s_0
\end{bmatrix}
\]

(2)

is the \( i \)-th codeword, \( 0 \leq i \leq 2^n - 1 \), with bits \( g_{ij} \), \( 0 \leq j \leq n - 1 \).

For the definition of \( G(n) \) and for elementary properties we refer to [11, ch. 5]. Among other things it is proved there that, if \((b_0, b_1, \ldots, b_2, b_2) \) is the binary representation of the index \( i \), one has

\[
g_i = b_i + j (\mod 2), \quad 0 \leq j < n,
\]

(3)
or, written more concisely,

\[
g_i = i \oplus \frac{j}{2},
\]

(4)

where \( \oplus \) stands for the exclusive-or-operation.

The inverse mapping is given by

\[
b_j = \sum_{j=0}^{n-1} s_{ij} (\mod 2), \quad 0 \leq j < n.
\]

(5)

In Section IV we shall exploit a property concerning the relative order of two codewords of \( G(n) \), which is an immediate consequence of (5). We formulate this property as a lemma.

**Lemma:** Let \( g_i \) and \( g_j \) be two codewords of \( G(n) \), and let the bit with index \( k \) be the first bit from the left in which these codewords differ or, more specifically,

\[
g_{ik} > g_{jk}, \quad l = k + 1, k + 2, \ldots, n - 1,
\]

Then \( i > j \) if \( \sum_{k=0}^{n-1} s_{ik} \) is even and \( i < j \) if \( \sum_{k=0}^{n-1} s_{ik} \) is odd.

The subcode \( G(n,k) \) is defined as the \( \binom{n}{k} \times n \)-submatrix of \( G(n) \) consisting of all codewords with exactly \( k \) 1-bits, \( 0 < k \leq n \).

Throughout this section we use the following notation.

**Theorem 1:** Let \( k \) be any integer \( \geq 1 \). Any nonnegative integer \( n \), if \( k \) is even, and any positive integer \( n \), if \( k \) is odd, can be uniquely represented as

\[
n = \binom{a_k}{k} + \binom{a_{k-1}}{k-1} + \cdots + \binom{a_1}{1},
\]

(8)

with

\[
a_k > a_{k-1} > \cdots > a_1 \geq 0.
\]

For a proof we refer to [9]. Implicit in this proof is the construction of the digits \( a_k, a_{k-1}, \ldots, a_1 \), respectively. First one chooses \( a_k \) as large as possible such that \( \binom{a_k}{k} \leq n \). Then one chooses \( a_{k-1} \) as large as possible such that \( \binom{a_{k-1}}{k-1} \leq n - \binom{a_k}{k} \), etc. This property provides us with a number system for nonnegative integers, for any fixed value of \( k \), usually called the binomial number system. With respect to this number system (for some fixed value of \( k \)), we write

\[
n = (a_k a_{k-1} \cdots a_1).
\]

(9)

Now let \( l \) be a codeword of \( L(n,k) \) with ones in positions \( b_k, b_{k-1}, \ldots, b_1 \), and with \( n - 1 \geq b_k > b_{k-1} \geq \cdots > b_1 \geq 0 \). We introduce the following classes of codewords:

\[
L_{b_k} = \{0 \cdots 0x \cdots x \},
\]

\[
L_{b_{k-1}} = \{0 \cdots 010 \cdots 0x \cdots x \},
\]

\[
\vdots
\]

\[
L_{b_1} = \{0 \cdots 010 \cdots 010 \cdots 0x \cdots x \}.
\]

(10)

For each codeword of class \( L_{b_i} \), \( k \geq i \geq 1 \), one has to choose precisely \( i \) crossmarked places to fill in \( i \) ones, whereas the remaining places have to be filled in with zeros. It is obvious that the number of codewords in \( L(n,k) \) that precede \( l \) is equal to

\[
|L_{b_k}| + |L_{b_{k-1}}| + \cdots + |L_{b_1}|.
\]

Hence, if the word \( 0^{n-k}1^k \) in \( L(n,k) \) has index \( 0 \) we have for the lexicographic index \( \text{ind}_L(l) \) that

\[
\text{ind}_L(l) = \binom{b_k}{k} + \binom{b_{k-1}}{k-1} + \cdots + \binom{b_1}{1} = (b_k b_{k-1} \cdots b_1).
\]

(11)

**Theorem 3:** Let \( k \) be an integer \( \geq 0 \). Any nonnegative integer \( n \), if \( k \) is even, and any positive integer \( n \), if \( k \) is odd, can be uniquely represented as

\[
n = \binom{a_k}{k} + \binom{a_{k-1}}{k-1} + \cdots + \binom{a_1}{1},
\]

(8)

with

\[
a_k > a_{k-1} > \cdots > a_1 \geq 0.
\]

The inverse mapping is given by

\[
b_k = \sum_{j=0}^{n-1} s_{ik} (\mod 2), \quad 0 \leq k < n.
\]

(5)

In Section IV we shall exploit a property concerning the relative order of two codewords of \( G(n) \), which is an immediate consequence of (5). We formulate this property as a lemma.

**Lemma:** Let \( g_i \) and \( g_j \) be two codewords of \( G(n) \), and let the bit with index \( k \) be the first bit from the left in which these codewords differ or, more specifically,

\[
g_{ik} > g_{jk}, \quad l = k + 1, k + 2, \ldots, n - 1.
\]

Then \( i > j \) if \( \sum_{k=0}^{n-1} s_{ik} \) is even and \( i < j \) if \( \sum_{k=0}^{n-1} s_{ik} \) is odd.

The subcode \( G(n,k) \) is defined as the \( \binom{n}{k} \times n \)-submatrix of \( G(n) \) consisting of all codewords with exactly \( k \) 1-bits, \( 0 < k \leq n \).

**Theorem 1:** Let \( k \) be any integer \( \geq 1 \). Any nonnegative integer \( n \), if \( k \) is even, and any positive integer \( n \), if \( k \) is odd, can be uniquely represented as

\[
n = \binom{a_k}{k} + \binom{a_{k-1}}{k-1} + \cdots + \binom{a_1}{1},
\]

(8)

with

\[
a_k > a_{k-1} > \cdots > a_1 \geq 0.
\]

The inverse mapping is given by

\[
b_k = \sum_{j=0}^{n-1} s_{ik} (\mod 2), \quad 0 \leq k < n.
\]

(5)

In Section IV we shall exploit a property concerning the relative order of two codewords of \( G(n) \), which is an immediate consequence of (5). We formulate this property as a lemma.

**Lemma:** Let \( g_i \) and \( g_j \) be two codewords of \( G(n) \), and let the bit with index \( k \) be the first bit from the left in which these codewords differ or, more specifically,

\[
g_{ik} > g_{jk}, \quad l = k + 1, k + 2, \ldots, n - 1.
\]

Then \( i > j \) if \( \sum_{k=0}^{n-1} s_{ik} \) is even and \( i < j \) if \( \sum_{k=0}^{n-1} s_{ik} \) is odd.

The subcode \( G(n,k) \) is defined as the \( \binom{n}{k} \times n \)-submatrix of \( G(n) \) consisting of all codewords with exactly \( k \) 1-bits, \( 0 < k \leq n \).

**Theorem 1:** Let \( k \) be any integer \( \geq 1 \). Any nonnegative integer \( n \), if \( k \) is even, and any positive integer \( n \), if \( k \) is odd, can be uniquely represented as

\[
n = \binom{a_k}{k} + \binom{a_{k-1}}{k-1} + \cdots + \binom{a_1}{1},
\]

(8)

with

\[
a_k > a_{k-1} > \cdots > a_1 \geq 0.
\]

The inverse mapping is given by

\[
b_k = \sum_{j=0}^{n-1} s_{ik} (\mod 2), \quad 0 \leq k < n.
\]

(5)
Choose $a_k$ as small as possible such that $\binom{n}{k} > n$. Then

$$0 < n_1 = \left(\frac{a_k}{k}\right) - n \leq \left(\frac{a_k}{k}\right) - \left(\frac{a_k - 1}{k - 1}\right) = \frac{a_k - 1}{k - 1}.$$  

Choose $a_{k-1}$ as small as possible such that $\binom{n}{k-1} > n_1$. Then $a_{k-1} < a_k$ and

$$0 \leq n_2 = \left(\frac{a_{k-1}}{k-1}\right) - n_1 < \left(\frac{a_{k-1}}{k-1}\right) - \left(\frac{a_{k-1} - 1}{k-2}\right) = \frac{a_{k-1} - 1}{k-2}.$$  

Choose $a_{k-2}$ as small as possible such that $\binom{n}{k-2} > n_2$. Then $a_{k-2} < a_{k-1}$. Continue with

$$0 < n_3 = \left(\frac{a_{k-2}}{k-2}\right) - n_2 = - n + \left(\frac{a_k}{k}\right) - \left(\frac{a_{k-1}}{k-1}\right) + \left(\frac{a_{k-2}}{k-2}\right)$$

in the same fashion until one has

$$0 < n_{k-1} = \left(\frac{a_{2}}{2}\right) - n_{k-2} = - n + \left(\frac{a_k}{k}\right) - \left(\frac{a_{k-1}}{k-1}\right) + \cdots + \left(\frac{a_2}{2}\right).$$

Choose $a_1 = n_{k-1}$, Then we have

$$n = \left(\frac{a_k}{k}\right) - \left(\frac{a_{k-1}}{k-1}\right) + \left(\frac{a_{k-2}}{k-2}\right) - \cdots - \left(\frac{a_2}{2}\right),$$

with

$$a_k > a_{k-1} > a_{k-2} > \cdots > a_1 \geq 1.$$  

Let $k$ be odd and $n > 0$. Choose $a_k$ as small as possible such that $\binom{n}{k} \geq n$. Then

$$0 \leq n_1 = \left(\frac{a_k}{k}\right) - n < \left(\frac{a_k}{k}\right) - \left(\frac{a_k - 1}{k-1}\right) = \frac{a_k - 1}{k-1}.$$  

Continue with choosing the $a_{i}, k-1 \geq i \geq 1$, as small as possible such that $\binom{n}{i} \geq n_{i-1}$ if $i$ is odd, and $\binom{n}{i} > n_{i-2}$ if $i$ is even, as in the $k$ is even case. Since $n > 0$, we finally have $n_{k-1} > 0$ and so we can choose $a_1 = n_{k-1}$, with $a_1 \geq 1$.

We end up with

$$n = \left(\frac{a_k}{k}\right) - \left(\frac{a_{k-1}}{k-1}\right) + \left(\frac{a_{k-2}}{k-2}\right) - \cdots + \left(\frac{a_2}{2}\right),$$

and

$$a_k > a_{k-1} > a_{k-2} > \cdots > a_1 \geq 1.$$  

Hence, in all cases we have proved the existence of a representation as stated in the theorem.

To prove the uniqueness of this representation we assume that

$$n = \left(\frac{b_k}{k}\right) = \left(\frac{b_{k-1}}{k-1}\right) + \left(\frac{b_{k-2}}{k-2}\right) + \cdots + \left(\frac{b_1}{1}\right),$$

with

$$b_k > b_{k-1} > \cdots > b_1 \geq 1.$$  

is any representation of $n$ satisfying the requirements of the theorem. Then we shall show that $b_k$ is the smallest integer such that $\binom{n}{k} \geq n$, if $k$ is odd, and $\binom{n}{k} > n$, if $k$ is even.

Assume that this is not the case. From the assumption it follows that

$$\left(\frac{b_k}{k}\right) - \left(\frac{b_{k-1}}{k-1}\right) + \left(\frac{b_{k-2}}{k-2}\right) + \cdots + \left(\frac{b_1}{1}\right) > \left(\frac{b_{k-1}}{k-1}\right) = \left(\frac{b_k}{k}\right)$$

or

$$\left(\frac{b_{k-1}}{k-1}\right) - \left(\frac{b_{k-2}}{k-2}\right) + \cdots + \left(\frac{b_1}{1}\right) \geq \left(\frac{b_k}{k}\right) - \left(\frac{b_{k-1}}{k-1}\right) = \left(\frac{b_{k-1}}{k-1}\right).$$

However,

$$\left(\frac{b_k}{k-1}\right) - \left(\frac{b_{k-2}}{k-2}\right) + \cdots + \left(\frac{b_1}{1}\right) \leq \left(\frac{b_{k-1}}{k-1}\right) + \left(\frac{b_{k-2}}{k-2}\right) + \cdots + \left(\frac{b_1}{1}\right) \leq \left(\frac{b_k}{k-1}\right) - \left(\frac{b_{k-2}}{k-2}\right) + \cdots + \left(\frac{b_1}{1}\right).$$

If $b_k > b_{k-1} + 2$, the last expression is less than $\binom{n}{k-1}$ and we have a contradiction. The remaining case is when $b_k = b_{k-1} + 1$. Since now

$$\left(\frac{b_k}{k-1}\right) - \left(\frac{b_{k-1}}{k-1}\right) = \left(\frac{b_k}{k}\right),$$

the assumption yields

$$\left(\frac{b_k}{k-1}\right) - \left(\frac{b_{k-1}}{k-1}\right) + \cdots + \left(\frac{b_1}{1}\right) \leq 0.$$  

For odd $k$, this is obviously a contradiction because $b_k > b_{k-1} + 2 > \cdots > b_1$. However, then we have $n - \binom{n}{k}$ and $b_k$ is the smallest possible integer such that $\binom{n}{k} > n$. We conclude that in all cases $b_k = a_k$.

Similarly we can show that $b_k = a_k$, $k > 1$.

Hence, the representation derived in the first part of the proof is unique.

The contents of Theorem 1 allow us to represent the positive integers in a unique way, for any fixed value of $k$. Moreover, if $k$ is even, we can represent $0$ as well. We shall call this type of representation the alternating binomial number system (for the chosen $k$-value) and we shall write

$$n = (a_k a_{k-1} \cdots a_1).$$

We remark that implicit in the proof of Theorem 1, there is an algorithm to determine the digits $a_k, a_{k-1}, \cdots, a_1$.

IV. THE INDEX SYSTEM FOR $G(n,k)$

Let $g$ be a codeword of $G(n,k)$ with ones in positions $b_k, b_{k-1}, \cdots, b_1$, and $n = 1 \geq b_k > b_{k-1} > \cdots > b_1 \geq 0$. We introduce the following classes of codewords

$$G_{b_k} = \{0 \cdots 0 x \cdots \cdots \cdots x\},$$

$$G_{b_{k-1}} = \{0 \cdots 010 \cdots 0 \cdots \cdots \cdots x\},$$

$$\cdots,$$

$$G_{b_1} = \{0 \cdots 010 \cdots 0 \cdots 0 x \cdots \cdots \cdots x\}.$$

The argument follows that given for the classes $L_{i,k}$ in Section III. One has to choose $i$ crossmarked places in the codewords of $G_{b_k}$ to fill in $i$ ones, $k \geq i \geq 1$. Since all codewords of $G(n,k)$ are also words of $G(n)$ and since their relative order does not change when we restrict ourselves to the subcode $G(n,k)$, we can apply the lemma of Section II. This proves that the number of codewords of $G(n,k)$ preceding $g$ is equal to

$$|G_{b_k}| + |G_{b_{k-1}}| + \cdots + |G_{b_1}| + \epsilon_k.$$  

(13)
Here \( e_k = 0 \) if \( k \) is even and \( e_k = -1 \) if \( k \) is odd, since otherwise the codeword \( g \) itself would be counted as a word preceding \( g \). It follows that, if the word \( 0^a \cdot 1^b \in G(n, k) \) has index \( 0 \), the Gray index \( \text{ind}_G(g) \) satisfies

\[
\text{ind}_G(g) = \left( \binom{b_1 + 1}{k} - \binom{b_1 + 1}{k-1} \right) + \cdots + \left( \binom{b_1 + 1}{1} + e_k \right) = (b_k + b_{k-1} + 1 \cdot \cdots b_1 + 1) + e_k. \tag{14}
\]

The inverse problem of converting an index \( n \) to the corresponding codeword of \( G(n, k) \) amounts to expressing \( n - e_k \) in the alternating binary system by means of the construction of the digits \( a_k, a_{k-1}, \ldots, a_1 \) in the proof of Theorem 1. The positions \( b_k, b_{k-1}, \ldots, b_1 \) of the \( k \) nonzero entries in the codeword then follow immediately by taking \( b_i = a_i - 1, k \geq i \geq 1 \).

**Example:** In the following, all codewords of the code \( G(6, 4) \) are listed arranged in Gray order:

\[
\begin{align*}
001111 & \quad 110011 \quad 111001 \\
011011 & \quad 110110 \quad 101110 \\
011101 & \quad 111100 \quad 101011 \\
010111 & \quad 111010 \quad 101011 \\
\end{align*}
\]

According to (14), the index of the word 110101 is equal to

\[
(6531)_2 = \left( \binom{4}{0} - \binom{3}{1} + \binom{2}{2} - \binom{1}{1} \right) = 7.
\]

Conversely, suppose one wants to know the codeword with index 11 in \( G(6, 4) \). First we choose \( a_4 \) as small as possible such that \( \binom{a_4}{4} > 11 \). We find \( a_4 = 6 \). Next we choose \( a_3 \) as small as possible such that

\[
\left( \binom{a_3}{3} - \binom{a_3}{2} \right) - \binom{a_3}{1} = 11
\]

and find \( a_3 = 4 \). Since

\[
\binom{a_3}{3} - \binom{a_3}{2} + \binom{a_3}{1} = 0
\]

it now follows immediately that \( a_3 = 2 \) and \( a_2 = 1 \) (remember that always \( a_2 \geq i, k \geq i \geq 1 \), as a consequence of the inequalities that have to be satisfied by the \( a_i \)). So 11 = (6421)_4, that corresponds to the codeword 101011.

By a similar argument, we could derive the index of \( g \) in \( G(6n) \). Instead of the binomial coefficients in (14), we would have powers of 2 since the number of nonzero entries is not fixed any more in a class \( G_{kl} \). Some elementary manipulations with sequences of powers of 2 would then lead to the expression (5).

### V. Bounds for Distances in \( G(n, k) \)

In this section we present tight lower and upper bounds for the value of \( |i - j| \), where \( i \) and \( j \) are the indexes of \( g_i \) and \( g_j \), which have a Hamming distance of \( 2m \) (cf. Section II).

**Theorem 2:** Let \( g_i \) and \( g_j \) be codewords of \( G(n, k) \), \( n > k > 0 \), such that \( d(g_i, g_j) = 2m \), \( 0 < m \leq \min(k, n-k) \).

1. The value of \( |i - j| \) is minimal for the pair of codewords

\[
g_j = g_i = 0^n \cdot a_i = \underbrace{m \cdot 1101011010011}_{m},
\]

2. The value of \( |i - j| \) is maximal for the pair of codewords

\[
g_j = 0^n \cdot a_i = \underbrace{m \cdot 01010110011}_{m},
\]

We only give the outlines of a proof. Let \( g_i \) and \( g_j \) be codewords as indicated in Theorem 2. If \( g_i = g_j \), we say that \( g_i \) and \( g_j \) have the \( l \)th bit in common. Our proof now consists of the following steps.

- a) The value of \( |i - j| \) does not increase if one shifts common bits to the left in \( g_i \) and \( g_j \).
- b) Let \( k = m \) and \( n = 2m \). If \( j > i \) and \( j - i \) is minimal, then the codewords have the form \( g_i = 10^g \), and \( g_j = 01^g \), with \( g_i, g_j \in G(2m - 2, m - 1) \) and \( d(g_i, g_j) = 2m - 2 \).
- c) Let \( k = m \) and \( n = 2m \). If \( j - i \) is maximal, then the codewords have the form \( g_i = 10^g \), and \( g_j = 01^g \), with \( g_i, g_j \in G(2m - 2, m - 1) \) and \( d(g_i, g_j) = 2m - 2 \).
- d) Using b) and c) and applying induction to \( m \), we can now prove that Theorem 2 is true for \( G(2m, m) \), \( m \geq 0 \).
- e) If \( g_i \) and \( g_j \) are of the type \( g_i = f0^g, g_j = f1^g \), and \( g_i = 0f^g, g_j = 1f^g \), then \( |i - j| > |i - j| \).
- f) If \( j > i \) and \( g_i = 1g \), and \( g_j = 0g \), then \( j - i \) increases if one shifts common 0-bits in \( g_i \) and in \( g_j \) to the left and common 1-bits to the right.
- g) Part 2) of Theorem 2 now follows by using d), e), and f), and applying induction to \( m \).

The calculations necessary to prove a)-c), e), and f) are straightforward and only elementary properties of binomial coefficients are involved. However, they are lengthy. For this reason they are omitted here and we refer to [13] for the details.

We remark that, instead of \( 0^n \cdot a_i = \underbrace{m \cdot 1101011010011}_{m} \) in part 1) of Theorem 2, we could have taken any other common subword of length \( n - 2m \) with \( k = m \) ones.

**Corollary:** Let \( g_i \) and \( g_j \) be codewords of \( G(n, k) \), \( n > k > 0 \) and let \( d(g_i, g_j) = 2m \), \( 0 < m \leq \min(k, n-k) \).

1. The minimal value of \( |i - j| \) is equal to

\[
\sum_{l=1}^{m-1} \left( \frac{2l}{l-1} \right).
\]

2. The maximal value of \( |i - j| \) is equal to

\[
\sum_{l=1}^{m-1} \left( \frac{2l}{l-1} \right) - \frac{n}{k}.
\]

**Proof:** Assume, without restriction of the generality, that \( j > i \).

a) From Part 1 of Theorem 2 and from Section IV, it follows immediately that, if \( j - i \) is minimal, we have

\[
\begin{align*}
\frac{2m}{m} & - \frac{m - 1}{m} - \cdots - \frac{2m - 2}{m - 1} + \frac{2m - 3}{m - 1} \\
& = \frac{2m - 4}{m - 2} - \frac{2m - 6}{m - 3} + \cdots + \frac{2}{1} + \frac{1}{1} \\
& = \frac{2m - 1}{m - 1} - \cdots - \frac{2m - 2}{m - 1} + \frac{2m - 4}{m - 2} \cdots + \frac{2}{1} + 1 \\
& = \frac{2m - 2}{m - 2} - \cdots - \frac{2m - 4}{m - 2} + \cdots + \frac{2}{1} + \frac{1}{1}.
\end{align*}
\]

b) The proof is analogous to the proof of Part a).
Part 1) of the Corollary is analogous to Yuen's lower bound for \(|i - j|\) where \(i\) and \(j\) are the indexes of codewords \(g_i\) and \(g_j\) of \(G(n, m)\), such that \(d(g_i, g_j) = m\) (cf. [12]). Part 2) of the Corollary is analogous to the upper bound for \(|i - j|\) as given by Cavior in [3].

Remark: The binomial coefficient occurring in Part 1) of the Corollary is close to the Catalan number \(C_i = \left(\begin{array}{c} 2i \\ i \end{array}\right) + 1\). In fact we have
\[
\left(\begin{array}{c} 2i \\ i \end{array}\right) = IC_i,
\]

These results indicate that, for parameters near \(10^5\), errors in the tabulated values are in the order of \(10^{-7}\). This is much too large to be accounted for by accumulated roundoff error. With \(NX\) and \(N\) near \(10^7\) the error is more reasonable, in the order of \(10^{-10}\), but still larger than expected. The problem is in the large parameter calculations of the two exponents \(A^i\) in Figs. 1-4. \(A\) is calculated as a difference between \(M\) and \(Y\), in one case, and \(K\) and \(N\) in the other. It turns out that, for parameters in the order of \(10^6\), each pair of terms is large and about equal so that \(A\) is a small difference of two large numbers. The resulting loss in significant digits noticeably affects the accuracy in the final answers in these cases. This problem can be largely overcome by combining terms differently. We can replace the original terms used to calculate \(A\),
\[
A = M \ln(y) - (y + C)
\]
where, with \(z = M + 1\), we have
\[
C = (z - 1/2) \ln(z) - z + \ln(\sqrt{2 \pi}) + J(z)
\]
and
\[
J(z) = \frac{1}{12z^2 + 5z + 42z + 53z^2 + z},
\]
by the following rearrangement,
\[
A = \left\{z + \frac{1}{2} \left[ \frac{1 - y/z}{1 + \ln(y)} \right] + \frac{1}{2} \ln(2\pi y) - J(z) \right\} - \frac{1}{2} \ln(2\pi y) - J(z).
\]
This substantially reduces the loss in significant digits for \(A\). Alternatively one could compute \(A\) using quadruple precision for even more accurate results. The errors with the adjusted calculations for \(A\) are in the order of \(10^{-15}\) for parameters near \(10^6\) and \(10^{-14}\) for parameters near \(10^7\). Using quadruple precision for the calculation of \(A\), we obtain yet smaller errors, in the order of \(10^{-15}\) or better even for parameters as large as \(10^9\). This level of error is the limit of accuracy with the double precision arithmetic used throughout (except for the calculation of \(A\)). Since there is little or no noticeable effect on the error when parameters are below \(10^7\) and virtually all cases of practical interest would have values below this, there is little practical reason why one should implement these changes if the earlier version is already installed.

Some corrections are as follows. Line 3 of Fig. 4 should read
\[
Y_s = \frac{1}{2} \left[ \left( N - \frac{1}{2} \right) + \sqrt{\frac{8}{5} \ln[4P_{FA}(1-P_{FA})]} \right]
\]
\[
+ \sqrt{\left( N - \frac{1}{2} \right)^2}
\]

Note on "The Calculation of the Probability of Detection and the Generalized Marcum Q-Function"

David A. Shnidman

In the above paper, computational results for \(P_s(X, Y)\) are given in Table I. Professor Carl W. Heisler provided me with corresponding results using steepest descent integration [1].

REFERENCES


IEEE Log Number 9143033.

REFERENCES