An Optimal Algorithm for the Construction of Optimal Prefix Codes with given Fringe

ALFREDO DE SANTIS*  GIUSEPPE PERSIANO†
Dipartimento di Informatica ed Applicazioni Aiken Comp. Lab.
Università di Salerno Harvard University
84081 Baronissi (Salerno), Italy Cambridge, MA 02138

Abstract

It is well known that the codeword lengths of a maximal prefix code with minimum length among those with a given number of code-words differ by at most one. In this paper we study the length of the optimal maximal prefix code with a given number \( N \) of codewords and the additional constraint that the difference of the lengths of the longest and shortest codeword must be equal to a given parameter \( \Delta \). We call such a code \((N, \Delta)\)-MPC.

The contribution of this paper is twofold. We give an optimal algorithm that, for all \( N \) and \( \Delta \), constructs an \((N, \Delta)\)-MPC of minimum length. Then, we give a lower bound for the length of the optimal \((N, \Delta)\)-MPC for the case \( \Delta \leq N/2 \). The bound we present is tight; that is for each \( N \) and \( \Delta \leq N/2 \) the code constructed by our algorithm achieves the bound with equality.

*Part of this work was done while the author was visiting IBM Research Division, T. J. Watson Research Ctr, Yorktown Heights, NY 10598.
†Partially supported by ONR Grant # N00014-88-C-0613

TH0373-1/91/0000/0297/$01.00 © 1991 IEEE
1 Introduction

Prefix codes constitute an important class of uniquely decipherable codes with a very simple decoding algorithm. Nonetheless, it can be proved that no efficiency is lost by restricting ourselves to prefix codes. Maximal prefix codes (MPC in short) are those prefix codes to which no word can be added and still preserve the prefix-free property. It is well known that the minimum length MPC with \( N \) words has length \( O(N \log N) \). The lengths of the codewords of such an optimal MPC differ by at most 1. On the other hand, the maximum length MPC with \( N \) words has length \( O(N^2) \) in the worst case and the longest word is \( N - 2 \) letters longer than the shortest one.

In this paper we study the gap between these two extreme cases. We fix the parameter \( \Delta \) and study the minimal length MPC with \( N \) codewords and difference between the length of the longest codeword and the length of the shortest codeword \( \Delta \).

Summary of the results. We start by giving an efficient algorithm that, on input \( N \) and \( 2 \leq \Delta \leq N - 2 \), constructs an \((N, \Delta)\)-MPC with minimum length. Then, we present a matching lower bound for the length of an \((N, \Delta)\)-MPC for the case \( \Delta = N/2 \); that is for each \( N \) and \( \Delta = N/2 \) we compute the length of the \((N, \Delta)\)-MPC with minimum length.

2 Notation and terminology.

In this section we set up our notations and terminology.

An alphabet \( \Sigma \) is a finite set of letters. A sequence \( a_1a_2\ldots a_l \) of \( l \) letters of \( \Sigma \) is called a word of length \( l \). We denote the length of a word \( w \) by \( l(w) \) and we say that a word \( w \) is non-empty if \( l(w) > 0 \). A set \( C \) of words of \( \Sigma \) is called a code. We denote by \( C(l) \) the number of codewords of \( C \) of length \( l \). In this paper we will only consider binary codes, that is codes over the alphabet \( \{0, 1\} \).

Let \( C = \{w_1, w_2, \ldots, w_N\} \) be a code with \( N \) codewords. The length, \( \text{LENGTH}(C) \), of \( C \) is defined as

\[
\text{LENGTH}(C) = \sum_{i=1}^{N} l(w_i)
\]

and its fringe, \( \text{FRINGE}(C) \), as

\[
\text{FRINGE}(C) = \max_i l(w_i) - \min_j l(w_j).
\]

We shall say that a code \( C \) with \( N \) words has length-configuration \( (L; a_1, a_2, \ldots, a_{\Delta}) \), if \( C \) has \( a_i \) words of length \( L + i \), for \( 0 \leq i \leq \Delta \), \( \sum a_i = N \) and \( a_0, a_{\Delta} \geq 1 \). A code of type \( (L; b_1, a_2, \ldots, a_{\Delta}) \), has an unspecified, but greater than or equal to \( b_1 \), words of length \( L \); more, precisely it has length-configuration \( (L; a_1, a_2, \ldots, a_{\Delta}) \) for some \( a_1 \geq b_1 \); * is a shorthand for \( \geq 0 \).
We say that a code has type \((L; a_1^L, a_2^{L+1}, \ldots, a_k^{L+b_k-1})\) if it has \(a_1\) words of length \(L\), \(a_2\) words of length \(L + 1\), \(L + b_1 - 1\), \(a_3\) words of length \(L + b_1 + 1\), \(L + b_1 + b_2 - 1\), and so on; that is, it has length-configuration 

\[
(L; a_1, a_2, \ldots, a_1, \ldots, a_2, \ldots, a_k).
\]

Two codes with the same length-configuration are considered the same code. In what follows, we will omit \(L\) from a length-configuration when it is either clear from the context or immaterial.

An important property of a code is the unique decipherability. A code \(C\) is uniquely decipherable if every sequence \(w_1w_2\ldots w_k\) of words of \(C\) can be uniquely broken-down into words of \(C\). For example, the code \(\{10, 1, 01\}\) is not uniquely decipherable because the sequence 101 can be parsed in two ways: as 1, 01 and as 10, 1.

Prefix codes constitute an important class of uniquely decipherable codes. Before going any further, we need the concept of a prefix of a word. We say that the word \(p\) is a prefix of the word \(w\) if there exists a word \(s\) for which \(w = ps\). A code \(C\) is said prefix if no code-word is the prefix of another code-word. It is easy to see that a prefix-code is uniquely decipherable. The following is a simple and well-known result, due to Kraft (see [1]), about prefix codes.

**FACT 1.** Let \(l_1, \ldots, l_N\) be a sequence of nonnegative integers. Then, there exists a prefix code with \(N\) code-words of length \(l_1, \ldots, l_N\) iff \(\sum_{i=1}^{N} 2^{-l_i} \leq 1\).

In this paper we study a special class of prefix codes the maximal prefix codes that are prefix codes with the property that if we add any word to the code the resulting code is not prefix. Maximal prefix codes are also called exhaustive (see for example [2]). Maximal prefix codes enjoy the following property which will be extensively used in the sequel.

**FACT 2.** [Kraft Equality] Let \(l_1, \ldots, l_N\) be a sequence of nonnegative integers. Then, there exists a maximal prefix code with \(N\) code-words of length \(l_1, \ldots, l_N\) iff

\[
\sum_{i=1}^{N} 2^{-l_i} = 1.
\]

In this paper we are interested in MPC with \(N\) codewords and given fringe \(\Delta\); we call such a code \((N, \Delta)\)-MPC. In particular, we will study the length of the optimal \((N, \Delta)\)-MPC. To this aim we define the function \(ep(N, \Delta)\) as follows:

\[
ep(N, \Delta) = \min_{C \text{ is an } (N, \Delta)\text{-MPC}} \text{LENGTH}(C).
\]

We restrict our attention to the case \(2 \leq \Delta \leq N - 2\). In fact, as can be immediately seen, if \(\Delta > N - 2\) then there exist no \((N, \Delta)\)-MPC. On the other hand, the cases \(\Delta = 0\) and \(\Delta = 1\) are trivial and are reviewed in Section 3.1.
2.1 Operations on \((N, \Delta)\)-MPC.

In this section we will describe two operations on \((N, \Delta)\)-MPC that we will find useful in the proof of our result.

Expansion of a codeword. This operation expands a codeword \(w\) of length \(l\) into two codewords of length \(l + 1\). That is, the codeword \(w\) is replaced by the two codewords \(w0\) and \(w1\). It is easy to see that this operation preserves the prefix-free and the maximality of the code. This operation can be iterated any number of times as we can in turn expand one (or both) of the two new codewords.

**Example.** Consider the MPC \(C = \{00, 01, 100, 101, 110, 1110, 1111\}\). By expanding twice the codeword 00 we obtain the code
\[
C' = \{000, 0010, 0011, 01, 100, 101, 110, 1110, 1111\}.
\]
It is easy to verify that \(C\) is still an MPC.

Contraction of two codewords. This operation is the inverse of an expansion. It replaces two codewords of length \(l + 1\) \(w1\) and \(w0\) into the codeword \(w\) of length \(l\). Again, this operation preserves the maximality and the prefix-freeness of the code. As before, this operation can be iterated. That is, we can contract \(2^a\) codewords of length \(l\) into one codeword of length \(l - a\).

**Example.** Consider the MPC \(C = \{000, 001, 010, 011, 100, 1010, 1011, 11\}\). By contracting the \(2^a\) codeword of length 3, 000, 001, 010, 011 into one of length 1 we obtain the code \(C' = \{0, 100, 1010, 1011, 11\}\) which is still an MPC.

3 The Algorithm

In this section we present and prove the correctness of an algorithm that, on input \(N\) and \(\Delta\) constructs the \((N, \Delta)\)-MPC with minimum length.

3.1 The cases \(\Delta = 0, 1\).

Before attacking the general case, let us briefly discuss the cases \(\Delta = 1\) and \(\Delta = 0\).

The case \(\Delta = 0\) is easily settled: an \((N, 0)\)-MPC exists iff \(N\) is a power of 2, in which case the \(N\) codewords are the \(N\) words of \(\log N\) and the code is unique. On the other hand, an \((N, 1)\)-MPC exists iff \(N\) is not a power of 2, in which case it is unique. The codewords of the unique \((N, 1)\)-MPC have lengths \([\log N] - 1\) and \([\log N]\) and its length is \(N([\log N] + 1) - 2^{\log N}\). The following property of \((N, 1)\)-MPC can be easily proved.

**Lemma 1.** If the \((N, 1)\)-MPC has \(c > 0\) codewords of length \([\log N] - 1\) then \(N + c\) is a power of 2 and, as \(c < N\) none of the integers in the interval \([N, N + c - 1]\) is a power of 2.
3.2 The general case

The two following definitions will be useful for the description of our algorithm.

**Definition 1.** For \(-1 \leq a \leq \Delta - 2\), we define an \((N, \Delta, a)\)-MPC in the following way.

An \((N, \Delta, -1)\)-MPC is an \((N, \Delta)\)-MPC with length-configuration \((+, +, 1^{\Delta-2}, 2)\).

For \(0 \leq a \leq \Delta - 3\), an \((N, \Delta, a)\)-MPC is an \((N, \Delta)\)-MPC with length-configuration \((1, 0^a, +, +, 1^{\Delta-(a+3)}, 2)\).

An \((N, \Delta, \Delta - 2)\)-MPC is an \((N, \Delta)\)-MPC with length-configuration \((1, 0^{\Delta-2}, +, +)\).

**Definition 2.** For \(N \geq 4\) and \(2 \leq \Delta \leq N - 2\), we define the function \(F(a, N, \Delta)\) as follows:

\[
F(a, N, \Delta) = N \left( \lfloor \log_X(a, N, \Delta) \rfloor + 1 \right) - 2 \log_X(a, N, \Delta) + \frac{2^{2^a + a + 1} + \Delta(\Delta - 3)}{2} - a\Delta - 2,
\]

where

\[
X(a, N, \Delta) = N - a + 2^{a+1} + a + 1.
\]

The Algorithm

**Input:** \(N\) and \(\Delta\).

**Step 1.** Compute \(\alpha\) as the integer part of the (unique) solution to the equation

\[
x + 2^{\alpha+1} = \Delta.
\]

**Step 2.** Let \(\beta\) be the largest integer \(-1 \leq \beta \leq \alpha\) such that the interval

\[
[X(\beta, N, \Delta), X(\beta, N, \Delta) + 2^{\beta+1} - 1]
\]

does not contain any power of 2.

**Step 3.** Let \(\gamma\) be the smallest integer \(\alpha \leq \gamma \leq \Delta - 2\) such that the interval

\[
[X(\gamma, N, \Delta), X(\gamma, N, \Delta) + 2^{\gamma+1} - 1]
\]

does not contain any power of 2.

**Step 4.** If \(F(\beta, N, \Delta) \leq F(\gamma, N, \Delta)\) then Output the \((N, \Delta, \beta)\)-tree

Else Output the \((N, \Delta, \gamma)\)-tree.

**Running Time.** For Step 1 we observe that \(\alpha\) is equal either to \(\lfloor \log \Delta \rfloor - 2\) or to \(\lfloor \log \Delta \rfloor - 1\). Therefore, Step 1 can be performed with a constant number of arithmetical operations. Step 2 requires \(O(\alpha) = O(\log \Delta)\) time. Step 3 takes \(O(\Delta)\) time. The running time of the algorithm is dominated by the time required to output the code that is \(O(N)\) and is thus optimal.
3.3 Correctness of the Algorithm

In this section we shall prove that the code output by the algorithm is indeed the minimum length \((N, \Delta)\)-MPC. To this aim we characterize the structure of an \((N, \Delta)\)-MPC and show that the minimum length \((N, \Delta)\)-MPC is an \((N, \Delta, a)\)-MPC for some \(a\).

We start with the following two lemmas.

Lemma 2. Let \(C\) be an \((N, \Delta)\)-MPC with minimum length and let \(L\) be the length of its shortest codeword. Then there exist at most two different integers \(L < l_1 < l_2 < L + \Delta\) for which \(C(l_1), C(l_2) \geq 2\). Moreover, if two such integers exists, then \(l_2 = l_1 + 1\).

**Proof.** Assume, by sake of contradiction, that \(C\) is an \((N, \Delta)\)-MPC of minimum length and length-configuration \((L; a_0, a_1, \ldots, a_A)\) where \(a_i, a_0 \geq 2\), for \(0 < l_1 + 1 < l_2 < \Delta\). In this case we will construct an \((N, \Delta)\)-MPC \(C'\) with shorter length by “replacing” two codewords of length \(l_2\) with two codewords of length \(l_1 + 1\).

More formally, consider the code \(C'\) with length configuration \((L; a_0', a_1', \ldots, a_A')\) where

\[
a_j' = \begin{cases} 
  a_j - 1, & \text{if } j = l_1; \\
  a_{j+1} + 2, & \text{if } j = l_1 + 1; \\
  a_{j-1} + 1, & \text{if } j = l_2 - 1; \\
  a_{l_2} - 2, & \text{if } j = l_2; \\
  a_j, & \text{otherwise}.
\end{cases}
\]

It is easy to verify that \(C'\) is still an \((N, \Delta)\)-MPC. Moreover, \(C'\) has length

\[
\text{LENGTH}(C') = \text{LENGTH}(C) + l_1 + 1 - l_2 < \text{LENGTH}(C).
\]

Thus \(C\) cannot have minimum length.

Now suppose that there exist \(l_1 < l_2 < l_3\) such that \(C(l_1), C(l_2), C(l_3) \geq 2\). Then it is immediate that \(l_3 \neq l_1 + 1\) and therefore by the previous case \(C\) cannot have minimum length. \(\Box\)

Lemma 3. Let \(C\) be an \((N, \Delta)\)-MPC with minimum length and let \(L\) be the length of its shortest codeword. Then \(C(L + \Delta) = 2\).

**Proof.** Using the same argument as before, it can be readily seen that \(C(L + \Delta) \leq 2\). The lemma follows from the observation that for any maximal prefix code \(P\) with longest codeword of length \(M\), \(P(M)\) is an even integer. \(\Box\)

We are now ready to prove the following

**Theorem 1.** If \(C\) is an \((N, \Delta)\)-MPC with minimum length then \(C\) is an \((N, \Delta, a)\)-MPC for some \(-1 \leq a \leq \Delta - 2\).

**Proof.** In view of the previous lemmas an \((N, \Delta)\)-MPC with minimum length must be of type either \((\leq 1^a, *, *, \leq 1^a-\Delta+2)\), for some \(0 \leq a \leq \Delta - 2\), or \((\leq 1^a, *, *, * )\).

Let us denote by \(m\) the largest of the (at most) two lengths \(l\) for which \(C(l) \geq 2\). The because of Kraft equality if \(l > m\) then \(C(l) > 0\). Therefore the only possible types are \((\leq 1^a, *, *, 1^a-\Delta+2)\) for \(a = 0\) we have an \((N, \Delta, -1)\)-MPC and \((\leq 1^a, *, *, * )\).
Now, let us consider \( C(m) \) and \( C(m - 1) \). If \( C(m) = C(m - 1) = 1 \) then, by Kraft, for all \( l < m - 1 \) \( C(l) = 1 \) and thus we have an \((N, \Delta, -1)\)-MPC. Suppose now that at least one of \( C(m - 1) \) and \( C(m) \) is greater than 1. In this case, we see that for \( L < l < m - 1 \) \( C(l) = 0 \). For otherwise, using the same trick as before, we could construct an \((N, \Delta)\)-MPC with shorter length by replacing two codewords of length \( m - 1 \) with two codewords of length \( l + 1 \) thus getting an \((N, \Delta)\)-MPC with shorter length. Therefore, an \((N, \Delta)\)-MPC with minimum length must be either of type \((1, 0^a, *, 1, a - 2, 2)\), for some \( 0 \leq a \leq \Delta - 3 \), or of type \((*, *, 1, 0^a - 2, 2)\), or of type \((1, 0^a - 2, *, *)\), thus proving the theorem.

Next we give a necessary and sufficient condition for the existence of an \((N, A, a)\)-MPC.

**Lemma 4.** For all \( N, 2 \leq \Delta \leq N - 2 \) and \( 0 \leq a \leq \Delta - 3 \), there exists an \((N, \Delta, a)\)-MPC if and only if the interval \([X(a, N, \Delta), X(a, N, \Delta) + 2^{a+1} - 1]\) does not contain any power of 2. If an \((N, \Delta, a)\)-MPC exists, it is unique and has length \( F(a, N, \Delta) \).

**Proof.** The if part can be argued as follows. As the interval \([X(a, N, \Delta), X(a, N, \Delta) + 2^{a+1} - 1]\) does not contain any power of 2, the unique \((X(a, N, \Delta), 1)\)-MPC has at least \( 2^{a+1} \) codeword of length \( \lceil \log X(a, N, \Delta) \rceil - 1 \). We can thus construct an \((N, \Delta, a)\)-MPC in the following way:

First, contract \( 2^{a+1} \) codewords of length \( \lceil \log X(a, N, \Delta) \rceil - 1 \) into one codeword of length \( \lceil \log X(a, N, \Delta) \rceil - a - 1 \) and then, expand one codeword of length \( \lceil \log X(a, N, \Delta) \rceil - a - 1 \) codewords one for each length \( \lceil \log X(a, N, \Delta) \rceil + 1, \ldots, \lceil \log X(a, N, \Delta) \rceil + \Delta - a - 1 \) and 2 of length \( \lceil \log X(a, N, \Delta) \rceil + \Delta - a - 2 \). The code so obtained has \( X(a, N, \Delta) - 2^{a+1} + 1 - 1 + \Delta - a - 2 \) = \( N \) codewords. This construction allows us also to compute the length of the \((N, \Delta, a)\)-MPC that a tedious computation proves to be \( F(a, N, \Delta) \).

**Example.** Let us consider the \((9, 5, 1)\)-MPC with length-configuration \((1; 1, 0, 2, 3, 1, 2)\). This code is obtained from the \((10, 1)\)-MPC with length-configuration \((3; 6, 4)\) in the following way. First, by contracting \( 4 (= 2^{a+1}) \) codewords of length 3 into one of length 1. Then, by expanding one codeword of length 4 into 1 of length 5 and 2 of length 6.

The only part can be proved by contradiction. Suppose that, for some \( 0 \leq c < 2^{a+1}, X(a, N, \Delta) + c \) is a power of 2 and that there exists an \((N, \Delta, a)\)-MPC. Then, by reversing the above construction we could construct an \((X(a, N, \Delta), 1)\)-MPC with at least \( 2^{a+1} > c \) codewords of length \( \lceil \log X(a, N, \Delta) \rceil \). This contradicts Lemma 1. Finally we prove that the \((N, \Delta, a)\)-MPC is unique. Suppose, on the sake of contradiction, that, for some \( N \) and \( \Delta \), there exist two different (non-isomorphic) \((N, \Delta, a)\)-MPCs. Then, by reversing the above transformation, we would be able to construct two (non-isomorphic) \((X(a, N, \Delta), 1)\)-MPC. Contradiction.

**Similarly, we can prove the following.**

**Lemma 5.** There exists an \((N, \Delta, -1)\)-MPC if and only if \( X(-1, N, \Delta) \) is not a power of 2. Moreover, if it exists, the \((N, \Delta, -1)\)-MPC is unique and has length \( F(-1, N, \Delta) \).
Lemma 6. There exists an \((N, \Delta, \Delta - 2)\)-MPC if and only if the interval 
\[X(\Delta - 2, N, \Delta), X(\Delta - 2, N, \Delta) + 2^{\Delta-1} - 1\]
does not contain any power of 2. If it exists, the \((N, \Delta, \Delta - 2)\)-MPC is unique and has length \(F(\Delta - 2, N, \Delta)\).

In the next lemma we study how the function \(F(\cdot, \cdot, \cdot)\) behaves as \(a\) varies between \(-1\) and \(\Delta - 2\) and \(N\) and \(\Delta\) are fixed.

Lemma 7. Let \(N \geq 4\), \(2 \leq \Delta \leq N - 2\), and \(0 \leq a \leq \Delta - 2\). If the \((N, \Delta, a - 1)\)-MPC exists then
\[
F(a, N, \Delta) \begin{cases} 
\leq F(a - 1, N, \Delta), & \text{if } \Delta \leq a + 2^{a+1}; \\
\geq F(a - 1, N, \Delta), & \text{if } \Delta \geq a + 2^{a+1}.
\end{cases}
\]

Proof. From the definition of \(F(a, N, \Delta)\), one gets that the difference \(F(a - 1, N, \Delta) - F(a, N, \Delta)\) is equal to 
\[
N(\log X(a - 1, N, \Delta) - \log X(a, N, \Delta) + 2\log X(a, N, \Delta) - 2\log X(a - 1, N, \Delta) + \Delta - 2^{a+1} - a.
\]

By hypothesis the \((N, \Delta, a - 1)\)-MPC exists. Therefore the interval 
\[X(a - 1, N, \Delta), X(a - 1, N, \Delta) + 2^{a} - 1\]
does not contain any power of 2. Recalling the definition of \(X(a, N, \Delta)\), we can rewrite the above interval as 
\[N - \Delta + 2^{a+1} + a, N - \Delta + 2^{a+1} + a - 1].
\]
If \(N - \Delta + 2^{a+1} + a\) is a power of 2 then \(\log X(a - 1, N, \Delta) = \log(N - \Delta + 2^{a+1} + a)\) and \(\log X(a, N, \Delta) = \log(N - \Delta + 2^{a+1} + a) + 1\) and thus
\[
F(a - 1, N, \Delta) - F(a, N, \Delta) = -N + N - \Delta + 2^{a+1} + a\Delta - 2^{a+1} - a.
\]
If \(N - \Delta + 2^{a+1} + a = X(a, N, \Delta) - 1\) is not a power of 2, then \(\log X(a, N, \Delta) = \log X(a - 1, N, \Delta)\). Therefore
\[
F(a - 1, N, \Delta) - F(a, N, \Delta) = \Delta - (2^{a+1} + a) \geq 0.
\]
Thus, if \(\Delta \leq a + 2^{a+1}\) then \(F(a, N, \Delta) \leq F(a - 1, N, \Delta)\) while if \(\Delta \geq a + 2^{a+1}\) then \(F(a, N, \Delta) \geq F(a - 1, N, \Delta)\).

Because of the above lemma, the length of the \((N, \Delta, a)\)-MPC does not increase with \(a\) if \(a \leq \alpha\) and does not decrease if \(a \geq \alpha\), where \(\alpha\) is the integer part of the solution to the equation \(\alpha + 2^{\alpha+1} = \Delta\). Therefore, if the \((N, \Delta, \alpha)\)-MPC exists then it is the \((N, \Delta)\)-MPC with minimum length. Otherwise, because of the behaviour of the function \(F(a, N, \Delta)\) there are only two possible candidates left: the \((N, \Delta, \beta)\)-MPC and the \((N, \Delta, \gamma)\)-MPC where \(\beta\) is the largest integer less than \(\alpha\) for which the \((N, \Delta, \beta)\)-MPC exists and \(\gamma\) is the smallest integer for which the \((N, \Delta, \gamma)\)-MPC exists. We can thus conclude that our algorithm is correct.
4 The lower bound

In this section we give a matching lower bound for all $N$ and $\Delta \leq N/2$; that is we compute exactly $\epsilon p(N, \Delta)$ for all $N$'s and $\Delta \leq N/2$.

In view of the above lemma we have that if the $(N, \Delta, \alpha)$-MPC exists, that is if the interval $[X(\alpha, N, \Delta), X(\alpha, N, \Delta) + 2^{\alpha+1} - 1]$ does not contain any power of 2, then $\epsilon p(N, \Delta) = F(\alpha, \Delta, N)$.

Let us now discuss the case in which the $(N, \Delta, \alpha)$-MPC does not exist.

Lemma 8. For all $N$'s and $\Delta \leq N/2$, if the $(N, \Delta, \alpha)$-MPC does not exist then both the $(N, \Delta, \alpha - 1)$-MPC and the $(N, \Delta, \alpha + 1)$-MPC exists.

Proof. As the $(N, \Delta, \alpha)$-MPC does not exist, the interval

$$[X(\alpha, N, \Delta), X(\alpha, N, \Delta) + 2^{\alpha+1} - 1]$$

contains a power of 2.

Now we first prove that, if $\Delta \leq N/2$, then the $(N, \Delta, \alpha + 1)$-MPC exists. As the interval $[X(\alpha, N, \Delta), X(\alpha, N, \Delta) + 2^{\alpha+1} - 1]$ contains a power of 2, the next power of 2 lies in the interval $[2 \cdot X(\alpha, N, \Delta), 2 \cdot (X(\alpha, N, \Delta) + 2^{\alpha+1} - 1)]$. But $X(\alpha + 1, N, \Delta) > X(\alpha, N, \Delta) + 2^{\alpha+1} - 1$ always and $X(\alpha + 1, N, \Delta) + 2^{\alpha+2} - 1 < 2 \cdot X(\alpha, N, \Delta)$ in the case $\Delta \leq N/2$. Therefore the interval $[X(\alpha + 1, N, \Delta), X(\alpha + 1, N, \Delta) + 2^{\alpha+2} - 1]$ does not contain any power of 2; that is, the $(N, \Delta, \alpha + 1)$-MPC exists.

A similar reasoning proves that if the $(N, \Delta, \alpha)$-MPC does not exist and $\Delta \leq N/2$ then the $(N, \Delta, \alpha - 1)$-MPC exists.

We can thus conclude that, when $\Delta \leq N/2$ and the $(N, \Delta, \alpha)$-MPC does not exist, the minimum length is obtained either by the $(N, \Delta, \alpha - 1)$-MPC or by the $(N, \Delta, \alpha + 1)$-MPC. Easy computations show that

$$F(\alpha - 1, \Delta, N) - F(\alpha + 1, \Delta, N) = 2\Delta - 2\alpha - 32^{\alpha+1} - 1 + 2^{\alpha+1} - N,$$

where $c = [\log X(\alpha, N, \Delta)]$.

We have thus proved the following.

Theorem 2. For all $N$'s and $\Delta \leq N/2$,

$$\epsilon p(N, \Delta) = \begin{cases} F(\alpha, \Delta, N) & \text{if } [X(\alpha, N, \Delta), X(\alpha, N, \Delta) + 2^{\alpha+1} - 1] \\
& \text{does not contain any power of 2;} \\
F(\alpha + 1, \Delta, N) & \text{if } 2\Delta \geq N - 2^{\alpha+2} + 3 \cdot 2^{\alpha+1} + 1; \\
F(\alpha - 1, \Delta, N) & \text{otherwise,}
\end{cases}$$

where $\alpha$ is the integer part of the (unique) solution of the equation $x + 2^{\alpha+1} = \Delta$ and $c = [\log X(\alpha, N, \Delta)]$ and $X(\alpha, N, \Delta) = N - \Delta + 2^{\alpha+1} + \alpha + 1$. 

References
