10-1976

Some results on arithmetic codes of composite length

Tai-Yang Hwang
Syracuse University

Carlos R.P. Hartmann
Syracuse University, chartman@syr.edu

5-76

SOME RESULTS ON ARITHMETIC CODES OF COMPOSITE LENGTH

Tai-Yang Hwang

Carlos R. P. Hartmann

October 1976

SYSTEMS AND INFORMATION SCIENCE
SYRACUSE UNIVERSITY

SOME RESULTS ON ARITHMETIC CODES OF COMPOSITE LENGTH

Tai-Yang Hwang

Carlos R. P. Hartmann

School of Computer and Information Science

Syracuse University

Syracuse, New York 13210

Tel. (315) 423-2368

This work was supported by the National Science Foundation under Grant ENG75-07709.

ABSTRACT

In this paper we present a new upper bound on the minimum distance of binary cyclic arithmetic codes of composite length. Two new classes of binary cyclic arithmetic codes of composite length are introduced. The error correction capability of these codes are discussed and in some cases the actual minimum distance is found. Decoding algorithms based on majority-logic decision are proposed for these codes.

I. Introduction

Arithmetic codes, first proposed by Diamond [1], are useful for error control in digital computation as well as in data transmission. They are particularly suitable for checking or correcting errors in arithmetic processors. Finding the minimum distance d of an arithmetic code is a major problem. Despite similarities between cyclic arithmetic and cyclic block codes, no general lower bound and, similar to the BCH bound for cyclic codes, exists for arithmetic codes. Thus, in general, the determination of d still relies on computer search. The search for a systematic way of constructing arithmetic codes is another major area of research. Three known classes of arithmetic codes are the high-rate perfect single-error correcting codes [2]-[4], the large-distance low-rate Mandelbaum-Barrows codes [5],[6] and the intermediate-rate intermediate-distance codes [7]. One of the interesting features of the codes introduced in [7] is that they can be decoded using majority-logic decisions.

In this paper we present a new upper bound on d for binary cyclic arithmetic codes of composite length. This bound is quite tight and gives a rather good estimation of the actual minimum distance. We also construct two new classes of binary cyclic arithmetic codes. Many of these codes have intermediate-rate and intermediate-distance and they can be decoded by majority-logic decisions.

In Section II, we present the new upper bound on d. In

Section III, we construct the two new classes of binary cyclic arithmetic codes. The decoding algorithm for these codes are given in Section IV. A discussion of the results is contained in Section V. Numerical examples are given in Appendix A. The conditions for the existence of codes in the classes constructed in Section III are given in Appendix B.

II. Bound on the Minimum Distance of Binary Cyclic Arithmetic Codes of Composite Length

A binary cyclic arithmetic (AN) code of length n is of the form AN, where A is a fixed integer, called the generator of the code, and N = 0,1,...,B-1. B is chosen so that $AB = 2^n-1$, where n is the multiplicative order of 2 modulo A. For a general background on binary cyclic AN-code as well as for the definitions of arithmetic distance and arithmetic weight, the readers are referred to [8]-[10].

The following theorem, which is a generalization of [11, Theorem 1], gives an upper bound on d.

Theorem 1: Let AN be a binary arithmetic code of composite length $n = n_1\ell_1$, $1 < \ell_1 < n$. If B is divisible by either $2^{n_1} + 1$ or by $2^{n_1} - 1$, then $d \le \ell_1$.

Proof: Let $B = B_1(2^{n_1}+1)$. By [12, Lemma 6.3] ℓ_1 is even. Thus,

$$AB_1 = \frac{2^n-1}{2^{n_1}+1} = 2^{(\ell_1-1)n_1} - 2^{(\ell_1-2)n_1} + - \dots + 2^{n_1} - 1$$

is a codeword of arithmetic weight ℓ_1, $W(AB_1) = \ell_1$. Similarly, we can show that $d \le \ell_1$ when $B = B_2(2^{n_1}-1)$.

Q.E.D.

The following example will illustrate the application of Theorem 1.

Example 1: Let $AB = 2^{20}-1$ with $A = 5\cdot31\cdot41$. Thus, $B = 3\cdot5\cdot11$ and n = 20. We note that $GCD(A,2^2-1) = 1$, $GCD(A,2^4-1) \neq 1$ and

$GCD(A, 2^5-1) \neq 1$. Thus, by [11, Theorem 1] $d \leq 10$. We may write $B = 5(2^5+1)$. Thus, by Theorem 1 $d \leq 4$. This code has $d = 4$ [13].

When B is of the form $2^{n_1}+1$ or $2^{n_1}-1$, the exact minimum distance can be determined. This result is given in the following:

<u>Theorem 2</u>: When $B = 2^{n_1}+1$ (or $2^{n_1}-1$), then $d = n/n_1 = \ell_1$.

<u>Proof</u>: For $B = 2^{n_1}+1$,

$$A = \frac{2^n-1}{2^{n_1}+1} = 2^{(\ell_1-1)n_1} - 2^{(\ell_1-2)n_1} + -\ldots + 2^{n_1} - 1$$

$$= (2^{n_1}-1)(2^{(\ell_1-2)n_1} + 2^{(\ell_1-4)n_1} + \ldots + 2^{2n_1} + 1).$$

It is easily seen that $W(AN) = \ell_1$ for $N = 1$, $N = 2^{n_1}$ and $N = 2^{n_1}-1$. If $0 < N-1 < 2^{n_1}-2$, then

$N-1 = a_{n_1-1}\, 2^{n_1-1} + a_{n_1-2}\, 2^{n_1-2} + \ldots + a_1\, 2 + a_0$ where $a_i = 0$ or 1,

for $i = 0,1,\ldots,n_1-1$. Furthermore not all a_i are 0 or 1. Thus,

$$A(N-1) = (2^{(\ell_1-1)n_1} - 2^{(\ell_1-2)n_1} + -\ldots + 2^{n_1}-1)(a_{n_1-1}2^{n_1-1} + a_{n_1-2}\, 2^{n_1-2} + \ldots + a_1\, 2 + a_0)$$

and

$$AN = (a_{n_1-1}2^{n_1-1} + a_{n_1-2}2^{n_1-2} + \ldots + a_1 2 + a_0)2^{(\ell_1-1)n_1}$$

$$+ ((1-a_{n_1-1})2^{n_1-1} + (1-a_{n_1-2})2^{n_1-2} + \ldots + (1-a_1)2 + (1-a_0))2^{(\ell_1-2)n_1}$$

$$+ (a_{n_1-1}2^{n_1-1} + a_{n_1-2}2^{n_1-2} + \ldots + a_1 2 + a_0)2^{(\ell_1-3)n_1}$$

$$+ ((1-a_{n_1-1})2^{n_1-1} + (1-a_{n_1-2})2^{n_1-2} + \ldots + (1-a_1)2 + (1-a_0))2^{(\ell_1-4)n_1}$$

.
.
.

$$+ (a_{n_1-1}2^{n_1-1} + a_{n_1-2}2^{n_1-2} +\ldots+ a_1 2 + a_0)2^{n_1}$$

$$+ ((1-a_{n_1-1})2^{n_1-1} + (1-a_{n_1-2})2^{n_1-2} +\ldots+ (1-a_1)2 + (1-a_0)).$$

By [7, Lemma 2], $W(AN) \geq \ell_1$. Thus, $d = \ell_1$ when $B = 2^{n_1}+1$. Erosh and Erosh [14] showed that $d = \ell_1$ when $B = 2^{n_1}-1$.

Q.E.D.

Example 2: Let $AB = 2^8-1$ with $A = 3\cdot 17$. Thus $B = 5$ and $n = 8$. By Theorem 2, $d = 4$.

Tables I and II in Appendix A give numerical examples of the application of Theorem 1 and 2, respectively.

III. On the Minimum Distance of Two Classes of Cyclic Arithmetic Codes of Composite Length

In this section we will consider two classes of cyclic AN-codes. The first class, C_1, has a generator of the form

$$A = \frac{2^n-1}{(2^{n_1}+1)(2^{n_2}+1)}$$

and the second class, C_2, of the form

$$A = \frac{2^n-1}{(2^{n_1}+1)(2^{n_2}-1)} ,$$

$n_1 \neq n_2$, $n = \ell_1 n_1 = \ell_2 n_2$ where $1 < \ell_1 < n$ and $1 < \ell_2 < n$. Appendix B gives the conditions for the existence of codes in these classes.

We first consider the class C_1. By [12, Lemma 6.3], ℓ_1 and ℓ_2 are even integers.

Theorem 3: If $n_2 > n_1$ then d of the codes in C_1 is bounded by

$$\min(\ell_2, \ell_1/2) \leq d \leq \ell_2 .$$

Proof: By Theorem 1, $d \leq \ell_2$. To obtain the lower bound we proceed as follows: if $N \equiv 0 \mod(2^{n_1}+1)$, then AN is a nonzero codeword in the AN-code generated by $(2^n-1)/(2^{n_2}+1)$ and by Theorem 2, $W(AN) \geq \ell_2$; if $N \not\equiv 0 \mod(2^{n_1}+1)$, then $AN(2^{n_2}+1) \mod(2^n-1)$ is a nonzero codeword in the AN-code generated by $(2^n-1)/(2^{n_1}+1)$ and by Theorem 2, $W[AN(2^{n_2}+1)] \geq \ell_1$. By the triangle inequality we have

$$W[AN(2^{n_2}+1)] \leq W(AN\cdot 2^{n_2}) + W(AN) ,$$

which implies

$$W[AN(2^{n_2}+1)] \leq 2W(AN) .$$

Thus $W(AN) \geq \ell_1/2$.

Q.E.D.

Example 3: Let $AB = 2^{60}-1$ with $B = (2^{15}+1)(2^{10}+1)$. Thus, $A = (2^{60}-1)/(2^{15}+1)(2^{10}+1)$ and $n = 60$. By Theorem 3, $3 \leq d \leq 4$.

Next, we consider the class C_2. By [12, Lemma 6.3], ℓ_1 is even.

Theorem 4: For codes in the class C_2 the following hold:

(a) If $n_2 > n_1$, then $\min(\ell_2, \ell_1/2) \leq d \leq \ell_2$.

(b) If $n_2 < n_1$, then $d = \ell_1$.

Proof: If $n_2 > n_1$, then the proof is analogous to the proof of Theorem 3. If $n_2 < n_1$, then the proof is analogous to the proof of [7, Theorem 1].

Q.E.D.

Example 4: Let $AB = 2^{60}-1$ with $B = (2^{10}+1)(2^{15}-1)$. Thus, $A = (2^{60}-1)/(2^{10}+1)(2^{15}-1)$ and $n = 60$. By Part (a) of Theorem 4, $3 \leq d \leq 4$.

Example 5: Let $AB = 2^{72}-1$ with $B = (2^{12}+1)(2^{9}-1)$. Thus, $A = (2^{72}-1)/(2^{12}+1)(2^{9}-1)$ and $n = 72$. By Part (b) of Theorem 4, $d = 6$.

Tables III and IV in Appendix A give numerical examples of the application of Theorems 3 and 4.

IV. Decoding Class C_1 and Class C_2 Codes

In this section we will present decoding algorithms for the codes of Classes C_1 and C_2. Their decoding algorithms depend on the decoding of the codes of length $n = n_1\ell_1$ generated by $A_0 = (2^n-1)/(2^{n_1}+1)$, which by Theorem 2 has minimum distance ℓ_1.

Suppose $R = A_0N+E$, $0 \leq N \leq 2^{n_1}$, is a corrupted codeword, and the arithmetic weight of the error pattern is $W(E) = t \leq \lfloor(\ell_1-1)/2\rfloor$, where $\lfloor x\rfloor$ denotes the largest integer less than or equal to x. As the first step of decoding we note that N is equal to zero if and only if $W(R) = W(E) \leq \lfloor(\ell_1-1)/2\rfloor$. Thus, N = 0 can be uniquely identified. When $0 < N \leq 2^{n_1}$, the decoding will be based on the result of the following theorem:

<u>Theorem 5</u>: The binary form of a codeword A_0N, $0 < N \leq 2^{n_1}$, is the following:

$$\begin{aligned}
A_0N = {} & (a_{n_1-1}2^{n_1-1} + a_{n_1-2}2^{n_1-2} + \ldots + a_1 2 + a_0)2^{(\ell_1-1)n_1} \\
& + ((1-a_{n_1-1})2^{n_1-1} + (1-a_{n_1-2})2^{n_1-2} + \ldots + (1-a_1)2 + (1-a_0))2^{(\ell_1-2)n_1} \\
& + (a_{n_1-1}2^{n_1-1} + a_{n_1-2}2^{n_1-2} + \ldots + a_1 2 + a_0)2^{(\ell_1-3)n_1} \\
& + ((1-a_{n_1-1})2^{n_1-1} + (1-a_{n_1-2})2^{n_1-2} + \ldots + (1-a_1)2 + (1-a_0))2^{(\ell_1-4)n_1} \\
& \vdots \\
& + (a_{n_1-1}2^{n_1-1} + a_{n_1-2}2^{n_1-2} + \ldots + a_1 2 + a_0)2^{n_1} \\
& + ((1-a_{n_1-1})2^{n_1-1} + (1-a_{n_1-2})2^{n_1-2} + \ldots + (1-a_1)2 + (1-a_0)),
\end{aligned}$$

where a_i is 0 or 1 for $i = 0,1,\ldots,n_1-1$ and
$N-1 = a_{n_1-1}2^{n_1-1} + a_{n_1-2}2^{n_1-2} + \ldots + a_1 2 + a_0$.

The proof of Theorem 5 is similar to the proof of Theorem 2.

Since the carry propagation caused by an error stops whenever a digit 0 is reached and the borrow propagation caused by an error stops whenever a digit 1 is reached, then, by Theorem 5, a single error can never corrupt more than n_1-1 consecutive digits in the binary form of A_0N when $N \neq 2^i$, $i = 0,1,\ldots,n_1$; and a single error can never corrupt more than n_1+1 consecutive digits in the binary form of A_0N when $N = 2^i$, $i = 0,1,\ldots,n_1$.

Let $A_0N = \sum_{j=0}^{n-1} b_j 2^j$, b_j is 0 or 1 for $j = 0,1,\ldots,n-1$.

If a single error does not corrupt more than n_1 digits, the binary coefficient b_k, $0 \leq k < n_1$, can be correctly estimated by taking the majority vote on the coefficients b_k, $1-b_{k+n_1}$, b_{k+2n_1}, $1-b_{k+3n_1}$, $\ldots, b_{k+(\ell_1-2)n_1}$ and $1-b_{k+(\ell_1-1)n_1}$ whenever $W(E) \leq (\ell_1-2)/2$ [7]. Thus, if $N \neq 2^i$, $i = 0,1,\ldots,n_1$ and $W(E) \leq (\ell_1-2)/2$ we would, using the above majority decision, correctly estimate A_0N. If $N = 2^i$, $i = 0,1,\ldots,n_1$, a single error can corrupt n_1+1 consecutive digits, this can contribute to at most two wrong votes in the majority decision. However, by noting the following facts:

(a) a carry propagation caused by an error which corrupts n_1+1 digits will introduce a subsequence of the form $F_1 = (10 \ldots 0)$ with at least n_1+1 consecutive 0's; and

(b) a borrow propagation caused by an error which corrupts n_1+1 digits, will introduce a subsequence of the form $F_2 = (01 \ldots 1)$ with at least n_1+1 consecutive 1's,

we can remove the effect of n_1+1 corrupted consecutive digits by applying the following operation on the binary representation of R:

Operation 1: If there is a subsequence of the form F_1, then change it to $F_1' = (10\ldots010\ldots0)$ by changing the (n_1+1)th bit of F_1 from 0 to 1. If there is a subsequence of the form F_2, then change it to $F_2' = (01\ldots101\ldots1)$ by changing the (n_1+1)th bit of F_2 from 1 to 0.

Thus, if $N = 2^i$, in the modified binary representation of R, each error will contribute to at most one wrong vote in the majority decision.

If $N \neq 2^i$, for $i = 0,1,\ldots,n_1$, Operation 1 will not change the majority decision since in this case it needs at least two errors to introduce a subsequence of the form F_1 or F_2.

In summary, the decoding of the A_0N code can be described as follows:

(a) If $W(R) \leq \lfloor(\ell_1-1)/2\rfloor$, decode $N = 0$, otherwise go to (b)

(b) If form F_1 or F_2 appears in the binary representation of R apply Operation 1; otherwise go to (c)

(c) the binary coefficients of N are determined by majority-logic decisions.

The decoding scheme described above for the A_0N codes can be used in the decoding of codes of Class C_1 and C_2. Let res(x) denote the residue of x modulo 2^n-1. Let $R = AN + E$ be the received word while AN is the codeword sent.

Decoding algorithm for Class C_1 Codes $(n_2 \geq 2n_1)$

(a) $N = 0$ if and only if $W(R) \leq \lfloor(\ell_1-1)/2\rfloor$; otherwise

(b) decode $res\{R(2^{n_2}+1)\}$, which is a corrupted word in the A_0N-code, to get $E' = res\{R(2^{n_2}+1)\} - res(A_0N)$;

(c) decode $res\{E'/(2^{n_2}+1)\}$, which is a corrupted word in the $A_0'N$-code, where $A_0' = (2^n-1)/(2^{n_2}+1)$, to get E.

Decoding algorithms for Class C_2 Codes

1. If $\ell_1 > \ell_2$,

   (a) $N = 0$ if and only if $W(R) \leq \lfloor(d-1)/2\rfloor$, where $d = \min(\ell_2, \ell_1/2)$; otherwise go to (b)

   (b) decode $res\{R(2^{n_2}-1)\}$, which is a corrupted word in the A_0N-code, to get $E' = res\{R(2^{n_2}-1)\} - res(A_0N)$;

   (c) decode $res\{E'/(2^{n_2}-1)\}$, which is a corrupted word in the A_0^*N-code, where $A_0^* = (2^n-1)/(2^{n_2}-1)$, to get E.

2. If $\ell_2 \geq 2\ell_1-1$

   (a) $N = 0$ if and only if $W(R) \leq \lfloor(\ell_1-1)/2\rfloor$; otherwise go to (b)

   (b) decode $res\{R(2^{n_1}+1)\}$, which is a corrupted word in the A_0^*N-code where $A_0^* = (2^n-1)/(2^{n_2}-1)$, to get $E' = res\{R(2^{n_1}+1)\} - res(A_0^*N)$ [7];

(c) decode $\text{res}\{E'/(2^{n_1}+1)\}$, which is a corrupted word in the A_0N-code, to get E.

Example 6 illustrates the decoding algorithm for an arithmetic code generated by $A_0 = (2^n-1)/(2^{n_1}+1)$, while Example 7 illustrates the decoding algorithm for an arithmetic code in Class C_2.

Example 6: Suppose $A_0 = (2^{24}-1)/(2^4+1)$, then $B = 2^4+1$, $n = 24$ and $n_1 = 4$. By Theorem 2, d = 6 and this code is capable of correcting any double errors. We have

$$A_0 = 2^{19}+2^{18}+2^{17}+2^{16}+2^{11}+2^{10}+2^9+2^8+2^3+2^2+2^1+1$$
$$= (00001111000011110000 1111) .$$

If a double error $E = 2^8 - 2^5$ is added to the codeword $3A_0$, then the corrupted word is $R = 3A_0 + E$ with binary representation

$$R = (001011010010111000001101) .$$

In this case there is a subsequence of the form F_1 in R with $n_1+1 = 5$ consecutive 0's. By applying Operation 1, the modified binary representation of R is R' = (001011010010111000101101).

We divide R' into 6 block and complement all the digits in positions 4i+j where i = 1,3,5 and j = 0,1,2,3. Then R' becomes

1101, 1101, 1101, 1110, 1101, 1101 .

We check that

digits with position		majority value
2^{4k+3}, $0 \le k \le 5$	1,1,1,1,1,1	1
2^{4k+2}, $0 \le k \le 5$	1,1,1,1,1,1	1
2^{4k+1}, $0 \le k \le 5$	0,0,0,1,0,0	0
2^{4k+0}, $0 \le k \le 5$	1,1,1,0,1,1	1 .

Hence

$A_0N = (0010,1101,0010,1101,0010,1101)$. By subtracting A_0N from R, we obtain

$$E = (0000,0000,0000,001(-1),00(-1)0,0000) = 2^8 - 2^5 .$$

Example 7: Suppose $A = (2^{60}-1)/(2^6+1)(2^{10}-1)$, then $B = (2^6+1)(2^{10}-1)$, $n = 60$, $n_1 = 6$ and $n_2 = 10$. By Part (a) of Theorem 4, $5 \le d \le 6$ and this code is capable of correcting any double errors. We have

$$A = \frac{2^{60}-1}{(2^6+1)(2^{10}-1)}$$

$$= 2^{43}+2^{42}+2^{41}+2^{40}+2^{39}+2^{38}+2^{34}+2^{31}+2^{30}+2^{29}+2^{27}+2^{26}+2^{24}+2^{22}$$
$$+2^{19}+2^{17}+2^{16}+2^{12}+2^9+2^8+2^7+2^6+1$$

$$= (000000000000000011111100010011101101010010110001001111000001) .$$

It a double error $E = -2^{53}+2^{11}$ is added to the codeword 32A, then the corrupted word is $R = 32A+E$. To decode, we first calculate the residue of $R(2^{10}-1) \bmod (2^{60}-1)$ which is

100000,000000,011111,100000,011111,

100000,100111,011111,111111,011000 . (1)

It is found there are one subsequence with more than $n_1 = 6$ consecutive 0's and one subsequence with more than n_1 consecutive 1's. Applying

Operation 1 to (1) yields

100000,100000,011111,100000,011111,

100000,100111,011111,011111,011000 . (2)

We complement all the digits in positions $6i+j$ where $i = 1,3,5,7,9$ and $j = 0,1,2,3,4,5$. Then (2) becomes

011111,100000,100000,100000,100000,

100000,011000,011111,100000,011000 .

We check that

digits with positions			majority value
2^{6k+5},	$0 \le k \le 9$	0,1,1,1,1,1,0,0,1,0	1
2^{6k+4},	$0 \le k \le 9$	1,0,0,0,0,0,1,1,0,1	0
2^{6k+3},	$0 \le k \le 9$	1,0,0,0,0,0,1,1,0,1	0
2^{6k+2},	$0 \le k \le 9$	1,0,0,0,0,0,0,1,0,0	0
2^{6k+1},	$0 \le k \le 9$	1,0,0,0,0,0,0,1,0,0	0
2^{6k+0},	$0 \le k \le 9$	1,0,0,0,0,0,0,1,0,0	0 .

Hence the majority decision of $\text{res}(R(2^{10}-1))$ yields a codeword $\text{res}(A_0N)$ = (011111,100000,011111,100000,011111,100000,011111,100000, 011111,100000). The error is now $E' = \text{res}(R(2^{10}-1)) - \text{res}(A_0N)$, which has the form

000000,100000,000000,000000,000000,

000000,001000,000000,(-1)00000,00(-1)000 .

The actual error E is congruent to $E'/(2^{10}-1) \bmod (2^{60}-1)$

$$E(2^{10}-1) \equiv E' = 2^{53}+2^{21}-2^{11}-2^{3} \quad \bmod (2^{60}-1)$$

$$E \equiv 2^{43}+2^{33}+2^{23}+2^{13}+2^{11}+2^{3} \quad \bmod (2^{60}-1)/(2^{10}-1)$$

which has the binary form

0000000000,0000001000,0000001000,0000001000,

0000001010,0000001000 . (3)

Again the majority scheme on (3) yields a block 0000001000. Repeating this block six times, we have

0000001000,0000001000,0000001000,0000001000,

0000001000,0000001000 . (4)

The binary integer (4) is a codeword generated by $(2^{60}-1)/(2^{10}-1)$. Subtracting (4) from (3), we get the actual error E.

000000(-1)000,0000000000,0000000000,0000000000

0000000010,0000000000 .

Hence, the error pattern E is $-2^{53}+2^{11}$.

V. Discussion

In this paper we have presented a new upper bound on the minimum distance of cyclic arithmetic codes of composite length. This upper bound is quite tight and gives a good estimation of the minimum distance. Two new classes of codes of composite length $n = \ell_1 n_1 = \ell_2 n_2$ have been introduced. The error correction capability of these codes are discussed and in some cases the actual minimum distance is found. Since n_1 and n_2 need not be relatively prime, some of these new codes have better information rate than the comparable codes found in [7]. Decoding algorithms for these codes have also been provided. They are based on majority-logic decision, and are similar to the decoding algorithm proposed in [7].

Appendix A

In this appendix we will present numerical examples of the application of Theorems 1,2,3, and 4. The symbols for the tables are the following:

n	code length
A	generator of the code
B	number of codewords
d	actual minimum distance
d_{u_1}	upper bound on d given by Theorem 1
d_{u_2}	upper bound on d given by [11, Theorem 1]
R	is the code rate $(R = (\log_2 B)/n)$.

Table I gives numerical examples of the application of Theorem 1. The d of these codes were obtained by a computer search [13].

Table II gives numerical examples of the application of Theorem 2.

Table III and IV give numerical examples of the application of Theorems 3 and 4. In Table III we give upper and lower bounds on d while in Table IV the actual minimum distance is given.

TABLE I

n	A	B	d	d_{u_1}	d_{u_2}
12	5·7	3·3·13	3	4	6
16	3·257	5·17	4	4	16
18	3·7·19	3·3·73	4	6	18
20	5·31	3·5·11·41	3	4	10
20	5·31·41	3·5·11	4	4	10
20	5·11·31	3·5·41	4	5	10
20	3·5·31·41	5·11	6	10	20
24	3·3·17	5·7·13·241	3	4	8
24	7·17	3·3·5·13·241	3	4	6
24	3·3·241	5·7·13·17	4	4	8
24	7·17·241	3·3·5·13	4	4	6
24	5·7·241	3·3·13·17	5	6	12
24	3·3·5·241	7·13·17	5	6	8
24	3·3·13·241	5·7·17	6	6	8
24	5·7·17·241	3·3·13	6	8	12
28	5·127	3·29·43·113	3	4	14
28	29·113·127	3·5·43	4	4	7
30	7·31·331	9·11·151	6	6	15
30	7·31·151·331	9·11	6	6	15
32	17·65537	3·5·257	4	4	8
32	5·65537	3·17·257	4	4	16
32	3·65537	5·17·257	4	4	32
32	5·257·65537	3·17	8	8	16
32	3·257·65537	5·17	8	8	32
36	13·73	3·3·3·5·7·19·37·109	4	4	6
36	3·19·37·109	3·3·5·7·13·73	3	3	12
36	5·73	3·3·3·7·13·19·37·109	3	4	6
36	3·13·19·73	3·3·5·7·37·109	4	6	12

TABLE II

n	A	B	d
8	3·17	5	4
12	5·7·13	3·3	4
16	3·17·257	5	8
16	3·5·257	17	4
18	3·7·19·73	3·3	6
20	5·5·31·41	3·11	4
24	5·7·13·17·241	3·3	8
24	3·3·5·7·13·241	17	6
24	3·3·7·17·241	5·13	4
28	3·29·43·113·127	5	14
28	5·29·113·127	3·43	4
30	7·11·31·151·331	3·3	10
30	3·7·31·151·331	3·11	6
32	3·5·257·65537	17	8
32	3·5·17·65537	257	4
36	3·5·7·13·19·37·73·109	3·3	12
36	3·3·3·7·19·37·73·109	5·13	6
36	5·7·13·37·73·109	3·3·3·19	4

TABLE III

n	B	$d_{\leq}$	$d_{\geq}$	R
60	$(2^{15}-1)(2^{10}+1)$	4	3	0.4166
60	$(2^{15}+1)(2^{10}+1)$	4	3	0.4167
60	$(2^{10}+1)(2^{6}-1)$	6	5	0.2663
60	$(2^{10}-1)(2^{6}+1)$	6	5	0.2670
72	$(2^{18}-1)(2^{12}+1)$	4	3	0.4166
72	$(2^{18}+1)(2^{12}+1)$	4	3	0.4166
72	$(2^{12}+1)(2^{9}+1)$	6	4	0.2917
84	$(2^{21}-1)(2^{14}+1)$	4	3	0.4166
84	$(2^{21}+1)(2^{14}+1)$	4	3	0.4166
120	$(2^{30}-1)(2^{20}+1)$	4	3	0.4166
120	$(2^{30}+1)(2^{20}+1)$	4	3	0.4166
120	$(2^{20}-1)(2^{12}+1)$	6	5	0.266
120	$(2^{20}+1)(2^{15}+1)$	6	4	0.2916
120	$(2^{15}-1)(2^{12}+1)$	8	5	0.2250
120	$(2^{15}+1)(2^{12}+1)$	8	5	0.2250
120	$(2^{12}+1)(2^{10}-1)$	10	6	0.1833
120	$(2^{12}+1)(2^{10}+1)$	10	6	0.1833

TABLE IV

n	B	d=	R
60	$(2^{15}-1)(2^{6}+1)$	4	0.3503
60	$(2^{15}+1)(2^{6}+1)$	4	0.3503
72	$(2^{12}+1)(2^{9}-1)$	6	0.2916
84	$(2^{21}-1)(2^{6}+1)$	4	0.3217
84	$(2^{21}-1)(2^{6}+1)$	4	0.3217
84	$(2^{14}-1)(2^{6}+1)$	6	0.2383
84	$(2^{14}+1)(2^{6}-1)$	6	0.2378
120	$(2^{30}+1)(2^{12}+1)$	4	0.3500
120	$(2^{30}+1)(2^{15}-1)$	4	0.3750
120	$(2^{30}+1)(2^{15}+1)$	4	0.3750
120	$(2^{20}+1)(2^{12}-1)$	6	0.2666
120	$(2^{20}+1)(2^{15}-1)$	6	0.2916
120	$(2^{15}+1)(2^{4}+1)$	8	0.1590
120	$(2^{15}-1)(2^{6}+1)$	8	0.1751
120	$(2^{12}+1)(2^{5}-1)$	10	0.1412
120	$(2^{12}+1)(2^{5}+1)$	10	0.1420
120	$(2^{10}+1)(2^{6}-1)$	12	0.1331

Appendix B

In this section we will present conditions for the existence of codes in Classes C_1 and C_2.

Let $k > 1$ be an odd positive integer. $\bar{e}(k)$ is defined as the least positive integer such that $2^{\bar{e}(k)}+1$ is divisible by k, if one does exist. $e(k)$ is the exponent of k.

At this point we are required to prove the following technical lemmas:

Lemma B1: $e(k) = 2\bar{e}(k)$.

Proof: Since k divides $2^{\bar{e}(k)}+1$, $k|2^{\bar{e}(k)}+1$, then $k|2^{2\bar{e}(k)}-1$. Thus $e(k)|2\bar{e}(k)$. Assume $e(k)$ is odd, then $e(k)|\bar{e}(k)$. Thus $k|2^{\bar{e}(k)}-1$ which is a contradiction since $2^{\bar{e}(k)}+1$ and $2^{\bar{e}(k)}-1$ are relatively prime. So we can conclude that $e(k)$ is even. Let $e(k) = 2x$, so $x|\bar{e}(k)$. Assume $\bar{e}(k) = mx$, $m > 1$. Since $k|(2^x+1)(2^x-1)$ and $k|2^x+1$ there exists $k_1 \neq 1$ such that $k_1|k$ and $k_1|2^x-1$. Thus, $k_1|2^{\bar{e}(k)}-1$ which is a contradiction.

Q.E.D.

Lemma B2: $k|2^y+1$ if and only if $\bar{e}(k)|y$ and $y/\bar{e}(k)$ is odd.

Proof: Since $(2^y+1) - (2^{\bar{e}(k)}+1) = 2^{\bar{e}(k)}(2^{y-\bar{e}(k)}-1)$, $k|2^y+1$ if and only if $k|2^{y-\bar{e}(k)}-1$, i.e., if and only if $e(k)|y-\bar{e}(k)$. By Lemma B1, $e(k) = 2\bar{e}(k)$. So, $k|2^y+1$ if and only if $\bar{e}(k)|y$ and $y/\bar{e}(k)$ is odd.

Q.E.D.

Let $n = \ell_1 n_1 = \ell_2 n_2$, $1 < \ell_1 < n$, $1 < \ell_2 < n$, $m_1 = n_1/g$ and $m_2 = n_2/g$ where $g = GCD(n_1,n_2)$.

We are now in the position to prove the next two theorems which are the main results of this section.

<u>Theorem B1</u>: If ℓ_1,ℓ_2 and $m_1 m_2$ are even integers, then $(2^{n_1}+1)(2^{n_2}+1)\,|\,2^n-1$.

<u>Proof</u>: By [12, Lemma 6.3], $2^{n_1}+1\,|\,2^n-1$ and $2^{n_2}+1\,|\,2^n-1$. Now we will show that $GCD(2^{n_1}+1,2^{n_2}+1) = 1$. Assume $a > 1$ is a common factor of $2^{n_1}+1$ and $2^{n_2}+1$. By Lemma B2, $n_1 = v_1\bar{e}(a)$, $n_2 = v_2\bar{e}(a)$ with v_1 and v_2 odd integers. Thus, $\bar{e}(a)\,|\,g$. So m_1 divides v_1 and m_2 divides v_2. So, m_1 and m_2 are odd which is a contradiction.

Q.E.D.

<u>Theorem B2</u>: If ℓ_1 is even and m_2 is odd, then $(2^{n_1}+1)(2^{n_2}-1)\,|\,2^n-1$.

<u>Proof</u>: By [12, Lemma 6.3], $2^{n_1}+1\,|\,2^n-1$. It is simple to show that $2^{n_2}-1\,|\,2^n-1$. Now we will show that $GCD(2^{n_1}+1,2^{n_2}-1) = 1$. Assume $a > 1$ is a common factor of $2^{n_1}+1$ and $2^{n_2}-1$. Then, by Lemma B2, $n_1 = v_1\bar{e}(a)$ with v_1 odd. By [12, Lemma 6.1], $n_2 = v_2 e(a)$. By Lemma B1, $e(a) = 2\bar{e}(a)$. Since m_2 is odd, g must be divisible by $2\bar{e}(a)$. Thus, v_1 is even which is a contradiction.

Q.E.D.

References

[1] J.M. Diamond, "Checking codes for digital computers", Proceedings of IRE, vol. 43, pp. 478-488, April 1955.

[2] D.T. Brown, "Error detecting and correcting binary codes for arithmetic operations", IRE Trans. Electron. Comput., vol. EC-9, pp. 333-337, Sept. 1960.

[3] J.L. Massey, "Survey of residue coding for arithmetic errors", Int. Computation Cent. Bull. (UNESCO Rome, Italy), vol. 3, no. 4, pp. 3-17, Oct. 1964.

[4] W.W. Peterson, Error-Correcting Codes. Cambridge, Mass: M.I.T. Press, 1961.

[5] D. Mandelbaum, "Arithmetic codes with large distance", IEEE Trans. Inform. Theory, vol. IT-13, pp. 237-242, Apr. 1967.

[6] J.T. Barrows, Jr., "A new method for constructing multiple error correcting linear residue codes", Coordinated Science Lab., Univ. of Ill., Urbana, Ill., Rep. R-277, Jan. 1966.

[7] C.L. Chen, R.T. Chien and C.K. Liu, "On majority-logic-decodable arithmetic codes", IEEE Trans. on Inform. Theory, vol. IT-19, pp. 678-682, Sept. 1973.

[8] J.L. Massey and O.N. Garcia, "Error-correcting codes in computer arithmetic", in Advances in Information Systems Science, vol. 4. New York: Plenum, 1972, pp. 273-326.

[9] W.W. Peterson and E.J. Weldon, Jr., Error-Correcting Codes. 2nd ed. Cambridge, Mass.: M.I.T. Press, 1972.

[10] T.R.N. Rao, Error Coding for Arithmetic Processors. New York: Academic Press, 1974.

[11] C.R.P. Hartmann and K.K. Tzeng, "A bound for arithmetic codes of composite length", IEEE Trans. on Inform. Theory, vol. IT-18, pp. 308, March 1972.

[12] T. Kasami, S. Lin and W.W. Peterson, "Some results on cyclic codes which are invariant under the affine group and their applications", Inform. and Control, vol. 11, pp. 475-496, 1968.

[13] R.T. Chien and S.J. Hong, "On root-distance relation for arithmetic codes", Coordinated Sci. Lab., Univ. Ill., Urbana, Ill., Rep. R-440, Oct. 1969.

[14] I.L. Erosh and S.L. Erosh, "Arithmetic codes with correction of multiple errors", Probl. Peredach. Inform., vol. 3, pp. 72-80, 1968.