

DESIGNING CODES TO FIT YOUR NEEDS:
AN INVESTIGATION INTO THE CONSTRUCTION OF
BCH CODES

Jacob M. Farinholt

submitted in partial fulfillment of the requirements for Honors in
Mathematics at the University of Mary Washington

Fredericksburg, Virginia

April 2009

This thesis by **Jacob M. Farinholt** is accepted in its present form as satisfying the thesis requirement for Honors in Mathematics.

DATE

APPROVED

Keith E. Mellinger, Ph.D.
thesis advisor

J. Larry Lehman, Ph.D.
committee member

M. Gary Collier, Ph.D.
committee member

Contents

1	Introduction to algebraic coding theory	2
1.1	A code and its minimum distance	2
1.2	Codes, polynomials, and fields	3
2	BCH codes and results	4
2.1	The BCH construction	5
2.2	Minimizing irreducible factors	6
2.3	The structure of cyclotomic cosets	8
3	Results for values of δ	11
3.1	A case when $\delta = 9$	11
3.2	A case when $\delta = 11$	12
3.3	A case for arbitrary δ	14
3.4	A generalization to any BCH code	15
4	Some concluding remarks	16
A	Data on Cyclotomic Cosets	17
	References	19

Abstract

In this thesis we look at the construction of a particular subclass of cyclic codes known as the Bose-Chaudhuri-Hocquenghem (BCH) Codes. These codes are constructed with a prescribed minimum distance, which means that the codes can be designed to correct as many errors as are required for the intended application. Our goal is to construct classes of BCH codes in as simple a fashion as possible, and we explain what we mean by this. When we desire our codes to correct as many as t errors, we look at the corresponding polynomial ring and its ideals in order to determine algebraic conditions that would lead to the desired properties. This leads us to number theoretic arguments involving powers of elements in certain finite fields. The results involve the construction of BCH codes with small sets of generators.

1 Introduction to algebraic coding theory

In an age where reliance on communication systems is ever more rapidly growing, it is everyday becoming more necessary to develop methods to communicate more effectively. Suppose two individuals, Alice and Bob, are trying to communicate. When Alice sends Bob a message, the message is first converted into binary n -tuples, that is, binary “codewords” of length n , then transmitted to Bob, who then translates the codewords into the original message. During the transmission process, however, these codewords can be affected by outside elements, that can flip certain bits of a codeword from zeros to ones and vice versa. In turn, the message that Bob receives is often different from the message that Alice sent. Rather than leaving Bob to guess what the message should have said, if the codewords are designed and chosen carefully, the errors can in fact be corrected automatically. The goal of algebraic coding theory is to use algebraic methods to design codes to detect and correct their own errors, so that the receiver can still translate a received message into the original message sent.

1.1 A code and its minimum distance

By definition, a binary linear code, \mathcal{C} , is a collection of vectors that forms a subspace of the vector space V of all binary n -tuples. In this space, addition is defined component wise modulo 2. Since \mathcal{C} is a vector subspace, it can be defined by a collection of basis vectors that form a generator matrix, G .

We can also define the distance between two vectors to be the number of coordinates in which they differ. For example, the distance between (10011) and (10101) is 2, since there are two bits in which the vectors differ. This notion of distance, called *Hamming distance*, actually defines a metric on the vector space. With this in mind, we can then define a sphere of radius r around a vector v to be the set of all vectors u whose distance from v is less than or equal to r , that is, $S_r(v) = \{u \in V \mid d(u, v) \leq r\}$.

If we have a collection of vectors that define a code, we say that the minimum weight, or minimum distance, of the code is the shortest distance between two codewords. This is important when using the popular decoding technique called *maximum likelihood decoding*, as we will see. This technique assumes that when a message is transmitted, the fewest number of errors occurred. In other words, it assumes that when a vector is received, the actual codeword that was sent was the codeword closest to the vector. The following theorem stresses the importance of the minimum distance of a code in relation to maximum likelihood

decoding.

Theorem 1.1. *If d is the minimum weight of a code \mathcal{C} , then \mathcal{C} can correct $t = \lfloor \frac{d-1}{2} \rfloor$ or fewer errors, and conversely.*

A formal proof can be found in [6], but the idea behind the proof is that the spheres of radius $\lfloor \frac{d-1}{2} \rfloor$ around each codeword form the largest possible set of disjoint spheres. From this result, we see that the larger the minimum distance d is, the more errors \mathcal{C} can correct. In other words, to correct the largest number of errors, we want to construct the code so that the codewords are as “spread out” as possible.

Returning to our example of Alice and Bob, if Bob receives a vector that is not a codeword, he assumes that the correct codeword is the one that is closest to it. The only received vectors that cannot be corrected are ones that are not elements of the sphere of radius $\lfloor \frac{d-1}{2} \rfloor$ around any codeword.

1.2 Codes, polynomials, and fields

We start with a brief introduction to the relationship between algebra and coding theory, and we refer the reader to [6] for more details. For a good introduction to the basics of algebra, we refer the reader to [2]. There is a device called a “shift register” that is commonly used for encoding purposes. This device is extremely useful in that it requires no memory storage on a computer. For purposes of this paper, we can simply say that given a vector, a shift register is able to generate any cyclic shift of that vector. For example, suppose we have a vector (1001011). Then if we run it through a certain shift register, we can also obtain the vectors (0010111), (0101110), (1011100), etc. What would be particularly useful is if when a vector is in a code, then so also are all of its cyclic shifts. As a matter of fact, when dealing with a certain class of error-correcting codes called *cyclic codes*, this happens to be the case. In order to explain cyclic codes in much detail, we must first approach the concept of a codeword from a slightly different angle.

Rather than viewing a codeword as a vector in a vector space, we can instead view it as an element of a polynomial ring. We do this by treating each vector as the collection of coefficients of a polynomial. In other words, the vector $(a_n \ a_{n-1} \ \dots \ a_1 \ a_0)$ represents the polynomial $a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$. For example, the vector (10011) represents the polynomial $x^4 + x + 1$. Since the degrees of these polynomials cannot exceed the length of the vectors, we only consider those in the ring $F[x]/f(x)$, where $F = GF(2)$ and $f(x)$ is some polynomial.

Notice what happens to a codeword c in $F[x]/f(x)$ when we multiply it by x . The coefficient for x in c becomes the coefficient of x^2 in the new polynomial. Similarly, the coefficient of x^2 in c becomes the coefficient of x^3 in the new polynomial. We can see that by multiplying the codeword by x , we seem to be shifting each of the coefficients. If we design the ring correctly, then it will always be the case that multiplying a codeword by x corresponds to a complete cyclic shift. When this happens, and when that cyclic shift is also another codeword, we call the collection of cyclic shifts a cyclic code. In other words, cyclic codes have the property that for any codeword $(a_0 \ a_1 \ \dots \ a_{n-1} \ a_n)$, the vector $(a_n \ a_0 \ a_1 \ \dots \ a_{n-1})$ is also a codeword, and this new codeword is created simply by multiplying the first one by x . When discussing cyclic codes, we usually consider those found in the ring $F[x]/(x^n - 1)$, where n is some positive integer. This is because of the following incredible result.

Theorem 1.2. *A set of elements S in $F[x]/(x^n - 1)$ corresponds to a cyclic code \mathcal{C} iff S is an ideal in $F[x]/(x^n - 1)$.*

In this next theorem we find a way for us to describe a cyclic code simply by its generator polynomial, $g(x)$.

Theorem 1.3. *Let \mathcal{C} be an ideal (i.e. a cyclic code of length n) in $F[x]/(x^n - 1)$, and let $g(x)$ be the monic polynomial of smallest degree in \mathcal{C} . Then $g(x)$ is uniquely determined and $\mathcal{C} = \langle g(x) \rangle$*

This result is really more about algebra than it is about coding theory. Theorem 1.2 allows us to work with the algebra directly as it completely represents the code. Thus, we see that a cyclic code is extremely useful in that its construction, through the use of a shift register, requires no storage. Moreover, it is equivalent to an ideal in $F[x]/(x^n - 1)$, and it is generated by a single polynomial, $g(x)$.

2 BCH codes and results

Suppose there is a situation in which it is necessary to have a code that can correct a certain number of errors. An example of such a situation is in the coding of a compact disk. It is very easy for CDs to acquire marks, fingerprints, and scratches, all of which cause errors, or noise, when the disk is being read. In such a case, it is naturally desirable to have some code that can correct a large number of concentrated errors. Another example is found in deep space transmission, where it can take months for a single message to be received. In a situation like this, it is clearly desirable to have a code that can correct a very large number of errors

without the need to retransmit any information. In order to correct a specified number of errors, this code must have a certain minimum distance, δ . We run into a problem here. In general, determining the minimum distance of a code is an NP-hard problem. However, there is a special subclass of cyclic codes discovered by R. C. Bose and D. K. Ray Chaudhuri [1] in 1960, and independently by A. Hocquenghem [3] in 1959 that provides a solution to that problem. Rather than trying to determine the minimum distance of an already constructed code, this subclass of codes, called BCH codes, can be constructed to fit any prescribed minimum distance. We say that a code \mathcal{C} is a BCH code of designed distance δ if $g(x)$, the generator polynomial of \mathcal{C} , is the product of distinct minimal polynomials whose roots contain $\delta - 1$ consecutive roots of $x^n - 1$; say $\alpha^b, \alpha^{b+1}, \dots, \alpha^{b+\delta-2}$, where the powers of α give all roots of $x^n - 1$.

2.1 The BCH construction

To construct a binary BCH code \mathcal{C} of prescribed distance δ , first consider the roots of $x^n - 1$ for some integer $n \geq \delta$, where $x^n - 1$ is considered as a polynomial with coefficients in $GF(2)$. This collection of roots forms a cyclic subgroup of the multiplicative group of $GF(2^m)$ for some m . It is known that such a field exists, so we choose m such that $GF(2^m)$ is the smallest such field. Since the collection of roots is cyclic, all of the roots are generated by a particular root, α , which we call the *primitive n^{th} root of unity*. All of the other roots will simply be powers of α . Since \mathcal{C} is a cyclic code, it is an ideal in $F[x]/(x^n - 1)$ with generator polynomial $g(x)$. Notice that $g(x)$ divides $x^n - 1$. It follows that all of the roots of $g(x)$ will also be roots of $x^n - 1$.

We now present a result about the bound of a BCH code.

Theorem 2.1. *The minimum weight of a BCH code of designed distance δ is at least δ .*

We avoid giving a complete proof of this important result. The idea behind the argument is to show that any $\delta - 1$ columns of the parity check matrix H for the code described in the theorem are independent over $GF(q)$ (in the case of a binary BCH code, $q = 2$). This is done by showing that the determinant of the matrix consisting of any $\delta - 1$ columns of H is a multiple of a very special determinant, called a Vandermonde determinant. It follows from there that the minimum weight must be at least δ .

This theorem tells us that if the roots of $g(x)$ contain $\delta - 1$ consecutive roots of $x^n - 1$, then \mathcal{C} will have a minimum distance of at least δ . This is important, since by Theorem 1.1 it follows that a BCH code of designed distance δ can correct at least $t = \lfloor \frac{\delta-1}{2} \rfloor$ errors.

Notice that if δ is odd, then δ and $\delta + 1$ will both correct at least t errors. For the sake of simplicity, we will always assume that δ is odd. We now see that in order to construct \mathcal{C} so that it can correct t errors, we simply choose monic irreducible polynomials that have at least $\delta - 1$ consecutive roots, and let their product be the generator polynomial for \mathcal{C} . While the BCH construction works for any \mathbf{n} , we will restrict our attention to the case where \mathbf{n} is a power of 2 minus 1. This gives us more structure since there exist finite fields of size 2^k for any k . In particular, if $\mathbf{n} = q^m - 1$, then the code is called a *primitive BCH code*.

2.2 Minimizing irreducible factors

It is important to note that the $\delta - 1$ consecutive roots necessary to generate a BCH code need not start with α . We can construct one with the consecutive roots $\alpha, \alpha^2, \dots, \alpha^{\delta-1}$, and another using the roots $\alpha^5, \alpha^6, \dots, \alpha^{3+\delta}$. We can see that both follow the formula $\alpha^b, \alpha^{b+1}, \dots, \alpha^{b+\delta-2}$ for the same δ . Though they are two different codes, they both have the same bound on minimum distance and hence will both correct at least $\lfloor \frac{\delta-1}{2} \rfloor$ errors.

It is very rare that a collection of monic irreducible polynomials in the ring $F[x]/(x^n - 1)$ will produce only the $\delta - 1$ consecutive roots. For example, suppose you want a BCH code of designed distance $\delta = 7$. In the field $F[x]/(x^{15} - 1)$, where $F = GF(2)$, there are the following minimal polynomials: $(x^4 + x^3 + 1)$, $(x^4 + x + 1)$, $(x^4 + x^3 + x^2 + x + 1)$, $(x^2 + x + 1)$, and $(x + 1)$. Since all the roots of $x^{15} - 1$ form the multiplicative cyclic group of $GF(16)$, choose the smallest root of all of the monic irreducible polynomials listed, that is, the primitive \mathbf{n}^{th} root of unity, to be α . All the remaining roots will be powers of α . The monic irreducible polynomials and their roots, as powers of α , are enumerated in Table 1.

<u>Monic irreducible polynomials</u>	<u>Roots as powers of α</u>
$x^4 + x^3 + 1$	$\alpha, \alpha^2, \alpha^4, \alpha^8$
$x^4 + x + 1$	$\alpha^7, \alpha^{14}, \alpha^{13}, \alpha^{11}$
$x^4 + x^3 + x^2 + x + 1$	$\alpha^3, \alpha^6, \alpha^{12}, \alpha^9$
$x^2 + x + 1$	α^5, α^{10}
$x + 1$	1

Table 1: monic irreducible polynomials and their roots

By looking at Table 1, we can determine that $(x^4 + x^3 + 1)(x^4 + x^3 + x^2 + x + 1)(x^2 + x + 1)$ will have the roots $\alpha, \alpha^2, \dots, \alpha^6$, giving us a generator polynomial that will produce a BCH code of designed distance $\delta = 7$. Notice that in addition to those roots, the generator polynomial

also contains the roots α^8 , α^9 , α^{10} , and α^{12} . Since these roots are not consecutive, they will not necessarily affect the minimum distance of the code and can thus be ignored.

In this example we see that starting with the root α , it takes three monic irreducible polynomials in $F[x]/(x^{15} - 1)$ to generate the desired BCH code. If we look at a different polynomial field, however, we may not necessarily obtain the same result. In other words, suppose we are trying to obtain the $\delta - 1 = 6$ consecutive roots α , α^2 , \dots , α^6 of the polynomial $x^n - 1$ for some $n \neq 15$. Then we will be looking for monic irreducible polynomials in $F[x]/(x^n - 1)$ that divide $x^n - 1$ and whose collective roots contain α , α^2 , \dots , α^6 . Naturally, we will have a different set of monic irreducible divisors, and, depending on the value of n , more of them may be necessary in order to generate the desired BCH code.

Our work is motivated by the following problem. By varying the value of n in $F[x]/(x^n - 1)$ we would like to determine the fewest number of monic irreducible polynomials necessary to produce a BCH code of a specific distance δ . In order to answer this question, we must first discuss cyclotomic cosets and their relationship to minimal polynomials.

By definition, a cyclotomic coset is a collection of integers of the form $\{s, ps, p^2s, \dots, p^r s\}$ where each $p^i s$ is reduced modulo $(p^n - 1)$ for some prime p and some integer n . In this collection, s is some positive integer, and r the smallest positive integer such that $p^{r+1}s \equiv s \pmod{p^n - 1}$. We will only be focusing on the case where $p = 2$. A cyclotomic coset can then be equivalently described as the collection of elements m that satisfy

$$m \equiv s \cdot 2^k \pmod{2^n - 1}$$

for all nonnegative integers k , where s and n are some positive integers. All the cyclotomic cosets that satisfy this for a particular $n = 2^n - 1$ are called the *cyclotomic cosets of n* . Notice that s is a solution to the above congruence. If s is the smallest such solution for a given n , we say that the set of solutions to this congruence is the *cyclotomic coset of n generated by s* , denoted C_s . For example, let $n = 4$ so that $2^n - 1 = 15$. In this case the cyclotomic coset of 4 generated by 3 is $\{3, 6, 12, 9\}$. We can easily calculate the other four cyclotomic cosets in this example: $\{1, 2, 4, 8\}$, $\{5, 10\}$, $\{7, 14, 13, 11\}$, and $\{0\}$.

What makes these cyclotomic cosets so important to BCH codes is the following.

Theorem 2.2. *Let α be a root of $x^n - 1$ in the smallest finite field F of characteristic p that contains α and let $m(x)$ be its minimal polynomial. Let β be a primitive n th root of unity in F , and let $\alpha = \beta^s$. If u is the smallest element in the cyclotomic coset of n containing s , then $m(x) = \prod_{i \in C_u} (x - \beta^i)$.*

What this theorem shows us is that the degree of a minimal polynomial $m(x)$ is the size

of a cyclotomic coset. All the elements in the coset correspond to powers of the primitive n^{th} root of unity that are roots of $m(x)$. Recall from our previous example that in the field $F[x]/(x^{15}-1)$ there are five minimal polynomials that contain all 15 primitive roots of $x^{15}-1$. In that example, the minimal divisors of $x^{15}-1$ were calculated, and then their roots were given in Table 1. From that table, we could determine the number of minimal polynomials necessary to generate a BCH code of prescribed distance $\delta = 7$. We now see that we can determine this answer in a much easier manner. If we consider the cyclotomic cosets modulo $15 = 2^4 - 1$, we find that there are three that collectively contain the powers 1 through 6. It follows that there must be three minimal polynomials that produce the roots $\alpha, \alpha^2, \dots, \alpha^6$. The product of these three minimal polynomials will be the generator polynomial for the code. The same generator polynomial will also produce the roots $\alpha^8, \alpha^9, \alpha^{10}$ and α^{12} , but as mentioned earlier, these roots do not matter to us.

Just as the minimal polynomials change as we vary n , so do the cyclotomic cosets modulo $2^n - 1$. Recall our initial question. For a prescribed minimum distance δ , can we vary n in $F[x]/(x^n - 1)$ in order to produce a BCH code of prescribed distance δ using the fewest monic irreducible polynomials possible? We can now see that an equivalent question is the following. By varying n , can we obtain $\delta - 1$ consecutive integers contained in the fewest number of cyclotomic cosets modulo $2^n - 1$? This is helpful in that it allows us to use number theoretic results about cyclotomic cosets to determine important results about BCH codes. For a good introduction to the basics of number theory, we refer the reader to [4].

2.3 The structure of cyclotomic cosets

Before looking at some specific constructions, we examine the structure of the cyclotomic cosets that are necessary for understanding the generators of BCH codes. Our first result explains how many cosets we can expect for special values of n .

Theorem 2.3. *Suppose $2^n - 1 = p$ for some prime p . Then the number of distinct cyclotomic cosets is $\frac{p-1}{n}$.*

Proof. The set $\{1, 2, \dots, p-1\}$ is a cyclic group, G , under multiplication modulo p , and it has $p-1$ elements. The subgroup $\langle 2 \rangle$ has order n . Then the number of distinct cosets of $\langle 2 \rangle$ in G is

$$|G/\langle 2 \rangle| = \frac{|G|}{|\langle 2 \rangle|} = \frac{p-1}{n}.$$

□

Although each term in a cyclotomic coset is determined by multiplying the generator element by a power of two, since each term is reduced modulo $2^n - 1$ it is still possible for certain terms to be odd. As will become apparent later, it is important to know when terms in a particular cyclotomic coset are odd.

For the remainder of this paper, we will assume that every cyclotomic coset is arranged in the same form. When considering the cyclotomic coset C_s , it will have the form $\{s, 2s, 2^2s, \dots, 2^r s\}$, where each of the terms is reduced modulo $2^n - 1$. In other words, we let s be the term in position 0, $2s$ be in position 1, and in general, $2^i s \pmod{2^n - 1}$ be in position i . We can determine the following.

Theorem 2.4. *Let k_i be the integer representation of the element in C_s in position i . Then using the above terminology we have the following:*

- i) *If $k_i < \frac{2^n - 1}{2}$, then k_{i+1} will be even.*
- ii) *The last term in C_s is represented by the integer $k_r = \frac{s + 2^n - 1}{2}$.*
- iii) *If $\frac{2^n - 1}{2} < k_i < \frac{s + 2^n - 1}{2}$, then k_{i+1} will be odd.*

Proof. We consider each case.

- i) If $k_i < \frac{2^n - 1}{2}$, then it is clear that $2k_i$ will be less than the modulus and will thus be even.
- ii) If the last term is represented by k_r , then it must be the case that $2k_r \equiv s \pmod{2^n - 1}$. But since $2k_r$ is larger than the modulus and less than twice the modulus, it follows that $2k_r - (2^n - 1) = s$, and the result follows.
- iii) If $\frac{2^n - 1}{2} < k_i < \frac{s + 2^n - 1}{2}$, then $2k_i$ will clearly be larger than the modulus but less than twice the modulus. But then k_{i+1} is equal to $2k_i - (2^n - 1)$ and will thus be an odd number.

□

The next theorem gives us particular information about the parity of the last term in a cyclotomic coset.

Theorem 2.5. *The last term in C_s will be even iff $s \equiv 1 \pmod{4}$.*

Proof. By the previous theorem, we know that the last term can be described by the integer $k_r = \frac{s+2^n-1}{2}$. Since s is odd, $s-1$ is even. So we say $s-1 = 2t$, for some integer t . We can then say $k_r = 2^{n-1} + t$. We can clearly see that k_r is even iff t is even, that is, if $4|(s-1)$, or $s \equiv 1 \pmod{4}$. \square

Theorem 2.6. *Consider the cyclotomic coset C_s . If k_r is the integer representation of the last term in C_s , then*

$$k_{r-1} = \begin{cases} \frac{s+2^n-1}{4}, & \text{if } s \equiv 1 \pmod{4}, \\ \frac{s+3(2^n-1)}{4}, & \text{otherwise.} \end{cases}$$

Proof. If $s \equiv 1 \pmod{4}$, then k_r is even, which means that $k_{r-1} = \frac{k_r}{2} = \frac{s+2^n-1}{4}$. If $s \not\equiv 1 \pmod{4}$, then k_r is odd, so we can determine that $k_r = 2k_{r-1} - (2^n - 1)$, and solving for k_{r-1} we obtain the given result. \square

The next theorem was given as a corollary to a theorem in [5]. It is useful as another result about the structure of cyclotomic cosets so we include it here.

Theorem 2.7. *If two consecutive odd integers, $2j+1$ and $2j+3$, are in the same cyclotomic coset modulo 2^n-1 for $n > 5$, then $n \not\equiv 0 \pmod{5}$, and $n \equiv 2u \pmod{5}$, or $n \equiv 3u \pmod{5}$.*

It is also important to understand how many times the values in a particular cyclotomic coset “loop around.” We use this expression to describe when a particular integer value $2^i \cdot s$ is larger than the modulus and therefore gets reduced to a value less than the modulus. The “loop around terms” begin with the first term in the cyclotomic coset in which the integer value is larger than the least residue modulo $2^n - 1$ and continues through the last term in the coset. Our next result shows us how to determine exactly how many loop around terms will be in any cyclotomic coset generated by a particular s .

Theorem 2.8. *If γ is the largest integer such that $2^\gamma < s$, then the last γ terms will “loop around” in the cyclotomic coset C_s .*

Proof. We can rewrite s as $2^\gamma + c$ for some positive integer $c < 2^\gamma$. In C_s , the last term is $s \cdot 2^{n-1} \pmod{2^n - 1}$. We claim that the first term to “loop around,” that is, the first term whose least residue is not itself, is $2^{n-\gamma} \cdot s \pmod{2^n - 1}$. It suffices to show that $2^{n-\gamma} \cdot s > 2^n$ and $2^{n-\gamma-1} \cdot s < 2^n$. We know $s = 2^\gamma + c$, so $2^{n-\gamma} \cdot s = 2^n + 2^{n-\gamma} \cdot c$, clearly larger than 2^n . Also, $2^{n-\gamma-1} \cdot s = 2^{n-1} + 2^{n-\gamma-1} \cdot c$. We can see that $2^{n-\gamma-1} \cdot c < 2^{n-1}$, so $2^{n-\gamma-1} \cdot s = 2^{n-1} + 2^{n-\gamma-1} \cdot c < 2 \cdot 2^{n-1} = 2^n$. \square

We note that in the above proof we assumed that C_s has n terms. When C_s has less than n terms, we let the set repeat itself until it has n terms. All of those repeats will be considered “loop around” terms.

3 Results for values of δ

We are now ready to apply knowledge of the cyclotomic cosets to some specific values of δ . Our goal is to completely understand the number of irreducible factors necessary for a prescribed δ and a fixed value of $n = 2^n - 1$.

3.1 A case when $\delta = 9$

The fact that more than one BCH code can be constructed to have the same prescribed distance makes the general answer to this question much more complicated. To simplify this problem slightly, we will only consider BCH codes generated by polynomials that contain the consecutive roots $\alpha, \alpha^2, \dots, \alpha^{\delta-1}$ (although we will return to the more general construction in Section 3.4). Suppose we want to generate a BCH code of designed distance $\delta = 9$. This will produce a code that can correct at least $\frac{9-1}{2} = 4$ errors. We want to determine the fewest number of monic irreducible polynomials necessary to produce the roots $\alpha, \alpha^2, \dots, \alpha^8$ of the polynomial $x^n - 1$. The answer to this question is expressed as a theorem whose proof will rely on the following lemmas. In order to obtain α^5 and/or α^7 as roots, we first need to determine which cyclotomic cosets contain 5 and/or 7. Our first lemma will help us to determine if 5 and/or 7 lie in the same cyclotomic coset as 3. The second lemma will determine when 5 and 7 are in the same cyclotomic coset.

Lemma 3.1. *An integer a is an element of the cyclotomic coset generated by 3 if and only if one of the following is true:*

- 1) $a = 3 \cdot 2^k$ for some nonnegative integer k , or
- 2) $(2^n - 1) \mid (2a - 3)$ for some nonnegative integer n .

Proof. The definition of the cyclotomic coset generated by 3, C_3 , is the collection of elements a that satisfy the congruence $3 \cdot 2^k \equiv a \pmod{2^n - 1}$. The first case, then, is true by definition. The only way some number not in the set described by the first case can be an element of C_3 is if for some k , the least residue of $3 \cdot 2^k$ modulo $2^n - 1$ does not equal $3 \cdot 2^m$ for some m . By Theorem 2.8 this is only the case when $k = n - 1$. In other words, such an a is in C_3

only if $3 \cdot 2^{n-1} \equiv a \pmod{2^n - 1}$ for some n . Multiplying both sides of the congruence by 2, we see that $3 \equiv 2a \pmod{2^n - 1}$. It follows from subtracting 3 from both sides that a is in C_3 only if $2^n - 1$ divides $2a - 3$. \square

It follows from this lemma that for $n > 3$, C_3 will never contain 5 or 7.

Lemma 3.2. *An integer a is an element of the cyclotomic coset generated by 5, C_5 if and only if either $a = 5 \cdot 2^k$ for some nonnegative integer k , or*

- 1) *if a is odd, then $(2^n - 1) | (4a - 5)$ for some nonnegative integer n , or*
- 2) *if a is even, then $(2^n - 1) | (2a - 5)$ for some nonnegative integer n .*

Proof. If $a = 5 \cdot 2^k$, then a is in C_5 by definition. If $a \neq 5 \cdot 2^k$ for some k , then a can only be in C_5 if for some k , the least residue of $5 \cdot 2^k$ modulo $2^n - 1$ does not equal $5 \cdot 2^m$ for some integer m . By Theorems 2.6 and 2.8 we can determine that if a is odd, this is only the case when $k = n - 2$, and if a is even, this is only the case when $k = n - 1$. Supposing a is odd, we are considering the congruence $5 \cdot 2^{n-2} \equiv a \pmod{2^n - 1}$. Multiplying both sides by 4, and then subtracting 5, it follows that a is in C_5 only if $2^n - 1$ divides $4a - 5$. Now supposing a is even, we are considering the congruence $5 \cdot 2^{n-1} \equiv a \pmod{2^n - 1}$. Multiplying both sides by 2, then subtracting 5, it follows that a is in C_5 only if $2^n - 1$ divides $2a - 5$. \square

It follows from this lemma that for $n > 1$, 7 will never be in C_5 . We now use these results to form our main result of this section.

Theorem 3.3. *Any BCH Code of length $2^n - 1$ where $n > 3$ and of designed distance $\delta = 9$ (using $\alpha, \alpha^2, \dots, \alpha^8$ as the sequence of roots) must be generated by at least 4 monic irreducible polynomials.*

Proof. First note that $\alpha, \alpha^2, \alpha^4$, and α^8 are all roots of the same polynomial. By Lemma 3.1, the irreducible polynomial for α^3 does not have α^5 nor α^7 as a root. Similarly, by Lemma 3.2, the irreducible polynomial for α^5 does not have α^7 as a root. Hence, the smallest polynomial $f(x)$ that has $\alpha, \alpha^3, \alpha^5$ and α^7 as roots, is comprised of 4 irreducible factors. The remaining roots, $\alpha^2, \alpha^4, \alpha^6$, and α^8 , are all roots of $f(x)$. \square

3.2 A case when $\delta = 11$

Suppose we are in a situation requiring a code that can correct at least five errors. The simplest BCH code would be one of designed distance $\delta = 11$ (since $\frac{11-1}{2} = 5$). Again, we

will restrict our attention to the BCH code generated by the polynomials that produce the roots $\alpha, \alpha^2, \dots, \alpha^{\delta-1}$ of the polynomial $x^n - 1$. We already know from the previous case that for $n > 3$, α^3, α^5 , and α^7 are all produced by different polynomials. By looking at cyclotomic cosets, we know that if $n > 3$, then the polynomial that generates α also generates α^2, α^4 , and α^8 . Additionally, the polynomial that generates α^3 also generates α^6 and the polynomial that generates α^5 also generates α^{10} . The only root that we still need is α^9 .

By Lemma 3.1 we can determine that C_3 contains 9 only in the cases where $n = 2$ and $n = 4$. By Lemma 3.2 we can determine that C_5 contains 9 only in the case where $n = 5$. The following lemma will give us results about the cyclotomic coset generated by 7, allowing us to determine when and if it contains 9.

Lemma 3.4. *An integer a is an element of the cyclotomic coset generated by 7, C_7 if and only if one of the following is true:*

- 1) $a = 7 \cdot 2^k$ for some nonnegative integer k ,
- 2) $(2^n - 1) | (4a - 7)$ for some nonnegative integer n , or
- 3) $(2^n - 1) | (2a - 7)$ for some nonnegative integer n .

Proof. If $a = 7 \cdot 2^k$, then a is in C_7 by definition. As in the previous lemmas, if $a \neq 7 \cdot 2^k$ for some k , then a can only be in C_7 if for some k , the least residue of $7 \cdot 2^k$ modulo $2^n - 1$ does not equal $7 \cdot 2^m$ for some integer m . Theorem 2.8 shows that this is only the case for the last two terms, that is, when $k = n - 2$ and $k = n - 1$. Supposing first that $k = n - 2$, we are considering the congruence $7 \cdot 2^{n-2} \equiv a \pmod{2^n - 1}$. Multiplying both sides by 4, and then subtracting 7, it follows that in this case a is in C_7 only if $2^n - 1$ divides $4a - 7$. Now supposing that $k = n - 1$, we are considering the congruence $7 \cdot 2^{n-1} \equiv a \pmod{2^n - 1}$. Multiplying both sides by 2, then subtracting 7, it follows that in this case a is in C_7 only if $2^n - 1$ divides $2a - 7$. \square

From this lemma, we can conclude that for $n > 2$, the cyclotomic coset generated by 7 will never contain 9. We can now compile this information to give us a result about a BCH code of prescribed distance $\delta = 11$.

Theorem 3.5. *Any BCH Code of length $2^n - 1$, where $n > 5$, and of designed distance $\delta = 11$ (using $\alpha, \alpha^2, \dots, \alpha^{10}$ as the sequence of roots) must be generated by at least 5 monic irreducible polynomials.*

3.3 A case for arbitrary δ

We just saw two examples of constructions of BCH codes of particular designed distances δ . Suppose we wanted to generate a BCH code of a larger δ . The steps for determining the number of monic irreducible polynomials necessary would be similar to the previous two examples. For example, if $\delta = 13$, we would already know the constraints on n for the cyclotomic cosets up to C_9 . We would then need only determine the constraints on n such that cyclotomic cosets generated by a number smaller than $\delta - 2 = 11$ would contain 11 (we already know that 12 will be in C_3 , 10 will be in C_5 , and in general, any even number will be an element of a cyclotomic coset generated by a smaller number). As seen in previous examples, when n is large enough, 11 must generate its own cyclotomic coset, and hence an additional monic irreducible polynomial will be necessary. When $\delta = 15$ we can again predict that when n is large enough, you will need to add yet another monic irreducible polynomial.

We will now generalize this method for an arbitrary δ . In order to do this, we must first generalize the lemmas used in the previous examples. We see this in the following theorem.

Theorem 3.6. *Suppose d is odd and that $d = 2^\gamma + s$, where γ is as large as possible. Then an integer $a > d$ is an element of the cyclotomic coset generated by d modulo $2^n - 1$, C_d , iff one of the following is true:*

- 1) $a = d \cdot 2^k$ for some positive integer $k < n$, or
- 2) $(2^n - 1) | (a \cdot 2^m - d)$, for some $m \in \{1, 2, \dots, \gamma\}$.

Proof. Case 1) is true by definition. If $a \neq d \cdot 2^k$ for some k , then a can only be in C_d if it is a loop around term. Only the last γ terms are loop around terms from Theorem 2.8, so this will only be the case if $k = n - m$, for some $m \in \{1, 2, \dots, \gamma\}$. Then consider the congruence $d \cdot 2^{n-m} \equiv a \pmod{2^n - 1}$. Multiplying both sides by 2^m and then subtracting d from both sides, it follows that a is in C_d only if $2^n - 1$ divides $2^m \cdot a - d$. \square

This theorem gives us the requirements for an element to be in a particular cyclotomic coset. Notice that if n is large enough, an odd a will not be the element of any cyclotomic coset generated by a number smaller than itself, so each odd integer up through $\delta - 2$ will generate its own cyclotomic coset. What this means in terms of BCH codes is that if n is large enough, then we will need a separate monic irreducible polynomial in order to generate each odd-powered root. Suppose we want to generate a BCH code of prescribed distance δ . Our next result will give us a sufficient condition for n to be “large enough.”

Theorem 3.7. *Let γ be the largest integer such that $2^\gamma < (\delta - 4)$. If $2^n > (\delta - 2)(2^\gamma - 1) + 3$, then each positive odd integer less than δ will generate its own cyclotomic coset modulo $2^n - 1$.*

Proof. Considering Theorem 3.6, if $2^n - 1$ is larger than $(a \cdot 2^m - d)$ for all possible odd values of a and d and all m , then it cannot divide any of them, and hence each odd integer a will generate its own cyclotomic coset. By definition, d is always the smallest number in the cyclotomic coset generated by d . Hence, any value $a \neq d$ that could be an element of C_d must be larger than d . Since $2^n - 1$ must be larger than every $(a \cdot 2^m - d)$, we want to consider the case when $(a \cdot 2^m - d)$ is as large as possible. Clearly, the largest odd integer smaller than δ is $\delta - 2$ since we always assume δ is odd. Thus we let $a = \delta - 2$. Since m is determined by d , in order to have m as large as possible, d must be as large as possible. The largest value of d is $a - 2$, that is, $\delta - 4$. Since m is determined by d , the largest value for m is γ , where γ is the largest integer such that $2^\gamma < d$. Thus, $(a \cdot 2^m - d)$ is largest when $a = \delta - 2$, $d = \delta - 4$, and γ is the largest integer such that $2^\gamma < (\delta - 4)$. Notice that it is clear that $(\delta - 2) \cdot 2^\gamma$ is always larger than d . Plugging these values in and simplifying, we obtain the desired result. \square

Notice that there are $\frac{\delta-1}{2}$ odd powers less than δ . What the above theorem shows us about BCH codes is given in this next corollary.

Corollary 3.8. *A BCH code of prescribed distance δ that uses $\alpha, \alpha^2, \dots, \alpha^{\delta-1}$ as the sequence of roots is generated by at most $\frac{\delta-1}{2}$ monic irreducible polynomials.*

3.4 A generalization to any BCH code

As we just saw, when n is large enough to meet the bound in Theorem 3.7, and we use the roots $\alpha, \alpha^2, \dots, \alpha^{\delta-1}$ to construct a BCH code of prescribed distance δ , we are required to use $\frac{\delta-1}{2}$ monic irreducible polynomials. This happens to be the number of errors the code is guaranteed to correct. In other words, we can say that if we want to construct a BCH code that can correct at least t errors, then using the roots $\alpha, \alpha^2, \dots, \alpha^{\delta-1}$, we will need at most t monic irreducible polynomials.

This result naturally leads us to another question. Can we extend this result to a BCH code generated using an arbitrary collection of roots? In other words, is the maximum number of monic irreducible polynomials necessary to generate a BCH code of prescribed distance δ dependent upon which sequence of roots we choose? The answer to this question is yes. Notice that when considering a BCH code generated by the roots $\alpha, \alpha^2, \dots, \alpha^{\delta-1}$, that is, the powers 1 through $\delta - 1$ as elements of cyclotomic cosets, we have the convenience

of knowing that a cyclotomic coset containing one of the necessary even powers will have as its generator one of the necessary odd powers of α . We do not in general have this luxury. An even power in any other consecutive set is often generated by a smaller number not in the set.

For example, suppose we are considering the powers 8 through 13 (for $\delta = 7$), and that n is large enough that each odd power requires its own cyclotomic coset. In addition to needing separate cyclotomic cosets for the powers 9, 11, and 13, we also need separate cyclotomic cosets for 8, 10, and 12. While using the powers 1 through 6 would only require three cyclotomic cosets, using the powers 8 through 13 requires six. The reason for this is simply because the generators for the cyclotomic cosets containing the even powers are not in the collection 8 through 13. We can see from this example that in the worst case scenario, we could need a separate cyclotomic coset for each power, that is, $\delta - 1$ cyclotomic cosets. Because of this we can conclude the following about BCH codes in general.

Theorem 3.9. *Any BCH code of prescribed distance δ will be generated by at most $\delta - 1$ monic irreducible polynomials.*

Since a BCH code of prescribed distance δ can correct at least $t = \frac{\delta-1}{2}$ errors, we could equivalently say that a BCH code designed to correct at least t errors will require at most $2t$ monic irreducible polynomials.

4 Some concluding remarks

In conclusion, we know that a BCH code of prescribed distance δ is constructed using $\delta - 1$ consecutive roots of monic irreducible polynomials in the polynomial ring $F[x]/(x^n - 1)$. Each polynomial contains multiple roots, and we know that two roots are generated by the same polynomial if their powers are elements of the same cyclotomic coset. We found that the number of polynomials necessary to generate the consecutive roots is dependant upon the size of the field (i.e., the value of $n = 2^n - 1$). Using the roots $\alpha, \alpha^2, \dots, \alpha^{\delta-1}$ to construct the BCH code, we determined that if the field is large enough (i.e., the power of n meets the requirement in Theorem 3.7), then each odd-powered root needed will be generated by a separate polynomial. If it is smaller than this, then it is sometimes possible for a single polynomial to generate multiple odd-powered roots. Since all of the even-powered roots needed are generated by one of these polynomials, it follows that when the field is large enough, the number of polynomials necessary to generate the BCH code is equivalent to the number of odd-powered roots, as well as the number of errors that the code will correct. If

we use an arbitrary collection of roots, we could need up to twice as many monic irreducible polynomials.

We can conclude, then, that the simplest construction of a BCH code, that is, the construction that requires the fewest number of monic irreducible polynomials, is a BCH code constructed using the roots $\alpha, \alpha^2, \dots, \alpha^{\delta-1}$.

Recall that throughout this paper we restricted our attention to BCH codes constructed in the polynomial ring $F[x]/(x^n - 1)$ where n is a power of 2 minus 1. While this is helpful in that it gives us more structure since there exist finite fields of size 2^k for any k , it is not necessary. It would be interesting to see if we can determine similar results for the case when n is a power of some other prime minus 1, or even when n is arbitrary.

A Data on Cyclotomic Cosets

Here we see several tables containing all possible (distinct) cyclotomic cosets, given a particular modulus, along with the sets of roots of $x^n - 1$ to which they correspond.

<u>Cosets of $2^3 - 1 = 7$</u>	<u>Roots as powers of α</u>
{1, 2, 4}	$\alpha, \alpha^2, \alpha^4$
{3, 6, 5}	$\alpha^3, \alpha^6, \alpha^5$

<u>Cosets of $2^4 - 1 = 15$</u>	<u>Roots as powers of α</u>
{1, 2, 4, 8}	$\alpha, \alpha^2, \alpha^4, \alpha^8$
{3, 6, 12, 9}	$\alpha^3, \alpha^6, \alpha^{12}, \alpha^9$
{5, 10}	α^5, α^{10}
{7, 14, 13, 11}	$\alpha^7, \alpha^{14}, \alpha^{13}, \alpha^{11}$

<u>Cosets of $2^5 - 1 = 31$</u>	<u>Roots as powers of α</u>
{1, 2, 4, 8, 16}	$\alpha, \alpha^2, \alpha^4, \alpha^8, \alpha^{16}$
{3, 6, 12, 24, 17}	$\alpha^3, \alpha^6, \alpha^{12}, \alpha^{24}, \alpha^{17}$
{5, 10, 20, 9, 18}	$\alpha^5, \alpha^{10}, \alpha^{20}, \alpha^9, \alpha^{18}$
{7, 14, 28, 25, 19}	$\alpha^7, \alpha^{14}, \alpha^{28}, \alpha^{25}, \alpha^{19}$
{11, 22, 13, 26, 21}	$\alpha^{11}, \alpha^{22}, \alpha^{13}, \alpha^{26}, \alpha^{21}$
{15, 30, 29, 27, 23}	$\alpha^{15}, \alpha^{30}, \alpha^{29}, \alpha^{27}, \alpha^{23}$

<u>Cosets of $2^6 - 1 = 63$</u>	<u>Roots as powers of α</u>
{1, 2, 4, 8, 16, 32}	$\alpha, \alpha^2, \alpha^4, \alpha^8, \alpha^{16}, \alpha^{32}$
{3, 6, 12, 24, 48, 33}	$\alpha^3, \alpha^6, \alpha^{12}, \alpha^{24}, \alpha^{48}, \alpha^{33}$
{5, 10, 20, 40, 17, 34}	$\alpha^5, \alpha^{10}, \alpha^{20}, \alpha^{40}, \alpha^{17}, \alpha^{34}$
{7, 14, 28, 56, 49, 35}	$\alpha^7, \alpha^{14}, \alpha^{28}, \alpha^{56}, \alpha^{49}, \alpha^{35}$
{9, 18, 36}	$\alpha^9, \alpha^{18}, \alpha^{36}$
{11, 22, 44, 25, 50, 37}	$\alpha^{11}, \alpha^{22}, \alpha^{44}, \alpha^{25}, \alpha^{50}, \alpha^{37}$
{13, 26, 52, 41, 19, 38}	$\alpha^{13}, \alpha^{26}, \alpha^{52}, \alpha^{41}, \alpha^{19}, \alpha^{38}$
{15, 30, 60, 57, 51, 39}	$\alpha^{15}, \alpha^{30}, \alpha^{60}, \alpha^{57}, \alpha^{51}, \alpha^{39}$
{21, 42}	α^{21}, α^{42}
{23, 46, 29, 58, 53, 43}	$\alpha^{23}, \alpha^{46}, \alpha^{29}, \alpha^{58}, \alpha^{53}, \alpha^{43}$
{27, 54, 45}	$\alpha^{27}, \alpha^{54}, \alpha^{45}$
{31, 62, 61, 59, 55, 47}	$\alpha^{31}, \alpha^{62}, \alpha^{61}, \alpha^{59}, \alpha^{55}, \alpha^{47}$

Cosets of $2^7 - 1 = 127$

{1, 2, 4, 8, 16, 32, 64}
{3, 6, 12, 24, 48, 96, 65}
{5, 10, 20, 40, 80, 33, 66}
{7, 14, 28, 56, 112, 97, 67}
{9, 18, 36, 72, 17, 34, 68}
{11, 22, 44, 88, 49, 98, 69}
{13, 26, 52, 104, 81, 35, 70}
{15, 30, 60, 120, 113, 99, 71}
{21, 42, 84, 41, 82, 37, 74}
{23, 46, 92, 57, 114, 101, 75}
{27, 54, 108, 89, 51, 102, 77}
{29, 58, 116, 105, 83, 39, 78}
{31, 62, 124, 121, 115, 103, 79}
{43, 86, 45, 90, 53, 106, 85}
{47, 94, 61, 122, 117, 107, 87}
{55, 110, 93, 59, 118, 109, 91}
{63, 126, 125, 123, 119, 111, 95}

Roots as powers of α

$\alpha, \alpha^2, \alpha^4, \alpha^8, \alpha^{16}, \alpha^{32}, \alpha^{64}$
 $\alpha^3, \alpha^6, \alpha^{12}, \alpha^{24}, \alpha^{48}, \alpha^{96}, \alpha^{65}$
 $\alpha^5, \alpha^{10}, \alpha^{20}, \alpha^{40}, \alpha^{80}, \alpha^{33}, \alpha^{66}$
 $\alpha^7, \alpha^{14}, \alpha^{28}, \alpha^{56}, \alpha^{112}, \alpha^{97}, \alpha^{67}$
 $\alpha^9, \alpha^{18}, \alpha^{36}, \alpha^{72}, \alpha^{17}, \alpha^{34}, \alpha^{68}$
 $\alpha^{11}, \alpha^{22}, \alpha^{44}, \alpha^{88}, \alpha^{49}, \alpha^{98}, \alpha^{69}$
 $\alpha^{13}, \alpha^{26}, \alpha^{52}, \alpha^{104}, \alpha^{81}, \alpha^{35}, \alpha^{70}$
 $\alpha^{15}, \alpha^{30}, \alpha^{60}, \alpha^{120}, \alpha^{113}, \alpha^{99}, \alpha^{71}$
 $\alpha^{21}, \alpha^{42}, \alpha^{84}, \alpha^{41}, \alpha^{82}, \alpha^{37}, \alpha^{74}$
 $\alpha^{23}, \alpha^{46}, \alpha^{92}, \alpha^{57}, \alpha^{114}, \alpha^{101}, \alpha^{75}$
 $\alpha^{27}, \alpha^{54}, \alpha^{108}, \alpha^{89}, \alpha^{51}, \alpha^{102}, \alpha^{77}$
 $\alpha^{29}, \alpha^{58}, \alpha^{116}, \alpha^{105}, \alpha^{83}, \alpha^{39}, \alpha^{78}$
 $\alpha^{31}, \alpha^{62}, \alpha^{124}, \alpha^{121}, \alpha^{115}, \alpha^{103}, \alpha^{79}$
 $\alpha^{43}, \alpha^{86}, \alpha^{45}, \alpha^{90}, \alpha^{53}, \alpha^{106}, \alpha^{85}$
 $\alpha^{47}, \alpha^{94}, \alpha^{61}, \alpha^{122}, \alpha^{117}, \alpha^{107}, \alpha^{87}$
 $\alpha^{55}, \alpha^{110}, \alpha^{93}, \alpha^{59}, \alpha^{118}, \alpha^{109}, \alpha^{91}$
 $\alpha^{63}, \alpha^{126}, \alpha^{125}, \alpha^{123}, \alpha^{119}, \alpha^{111}, \alpha^{95}$

References

- [1] R. C. Bose and D. K. Ray-Chaudhuri, On a class of error-correcting binary group codes, *Info. and Control*, **3** (1960) 68–79, 279–290.
- [2] Durbin, J. *Modern Algebra: An Introduction*, fifth edition, John Wiley & Sons, Inc., 2005.
- [3] A. Hocquenghem, Codes correcteurs d'erreurs, *Chiffres* (Paris) **2** (1959) 147–156.
- [4] Lehman, L. *Number Theory*, Course Pack (Unpublished), 2008.
- [5] D. Mandelbaum, Two Applications of Cyclotomic Cosets to certain BCH Codes, *IEEE Transactions on Information Theory*, Vol. IT-26, No. 6, November 1980.
- [6] Pless, V. *Introduction to the Theory of Error-Correcting Codes*, third edition, Wiley-Interscience Series in Discrete Mathematics, 1998.