JPL PUBLICATION 77-23

Review of Finite Fields: Applications to Discrete Fourier Transforms and Reed-Solomon Coding

National Aeronautics and Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California 91103

JPL PUBLICATION 77-23

Review of Finite Fields: Applications to Discrete Fourier Transforms and Reed-Solomon Coding

J. S. L. Wong
T. K. Truong
B. Benjauthrit
B. D. L. Mulhall
I. S. Reed

July 15, 1977

National Aeronautics and
Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California 91103

Preface

The work described in this report was performed by the Advanced Engineering Group, Tracking & Data Acquisition Engineering, Jet Propulsion Laboratory.

Dr. J. S. L. Wong is presently with the Hong Kong Polytechnic Institute, Hong Kong. Dr. I. S. Reed is professor of the Electrical Engineering Department at the University of Southern California.

Contents

Review of Finite Fields

Application of Finite Fields to Discrete Fourier Transforms

Application of Finite Fields to Reed-Solomon Coding

Abstract

This report attempts to provide a step-by-step approach to the subject of finite fields. Rigorous proofs and highly theoretical materials are avoided. The simple concepts of groups, rings, and fields are discussed and developed more or less heuristically. Examples are used liberally to illustrate the meaning of definitions and theories. Applications include discrete Fourier transforms and Reed-Solomon coding.

Review of Finite Fields

The ultimate objective of these notes is to show how the finite field can be used to decode Reed-Solomon codes. Before developing these techniques, let us consider some of the structures and properties of finite fields.

1 Group

A set of elements with a binary operation "$\cdot$" is called a group, G, if for any arbitrary elements a, b, and c, which belong to G, the following four postulates are satisfied:

P_1: $a \cdot b \in G$ (closure law)

P_2: $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ (associative law)

P_3: There exists an identity element e in G such that $a \cdot e = e \cdot a = a$ for all $a \in G$ (identity element)

P_4: For $a \in G$, there always exists an inverse element a^{-1} in G such that $aa^{-1} = e$ (inverse element)

1.1 A Commutative Group

A group is called a commutative group if the operation also satisfies the fifth postulate:

P_5: $a \cdot b = b \cdot a$ (commutative law)

1.2 Order of a Group

The order of a group is defined as the number of elements in the group. If the order is infinite, the group is an *infinite group*. Otherwise, the group is a *finite group*.

1.3 Example: An Infinite Group

Let $G = \{\pm 0, \pm 1, \pm 2, \cdots\}$ and let the operation be the arithmetic addition. It is straightforward to verify that G forms an additive infinite commutative group in which 0 is the identity element, and the inverse of a is $-a$.

1.4 Example: A Finite Group

Let $G = \{-1, 1\}$ and let the operation be multiplication. Then, it is straightforward to verify that G is a multiplicative commutative group of order 2, so called because the operation is multiplication, $(-1) \cdot 1 = 1 \cdot (-1)$, and there are only two elements.

1.5 Subgroup

A nonempty subset H of a group G is called a *subgroup* of G, if H itself forms a group under the same operation as in G.

1.6 Example: A Subgroup

In Section 1.3, we verified that $G = \{\pm 0, \pm 1, \pm 2, \cdots\}$ is a group under addition. Let H be the nonempty subset consisting of all multiples of 5, i.e., $H = \{\pm 0, \pm 5, \pm 10, \cdots\}$. It is obvious that H is a subset of G, and H itself forms a group under addition. Hence, H is a subgroup of G.

2 Ring

A nonempty set R with two binary operations is called a ring if in R there are two defined operations, addition (+) and multiplication ($\cdot$), such that any arbitrary elements a, b, and c in R satisfy the following postulates:

P_1: $a + b \in R$ (closed under addition)

P_2: $a + b = b + a$ (commutative law for addition)

P_3: $(a + b) + c = a + (b + c)$ (associative law for addition)

P_4: For every $a \in R$, there exists an element 0 such that $a + 0 = 0 + a = a$ (identity element for addition)

P_5: There exists an element a such that $a + (-a) = 0$ (inverse for addition)

P_6: $a \cdot b \in R$ (closed under multiplication)

P_7: $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ (associative law for multiplication)

P_8: $a\,(b + c) = a \cdot b + a \cdot c$ and $(b + c) \cdot a = b \cdot a + c \cdot a$ (the two distributive laws)

2.1 Commutative Ring

Note that postulates P_1 to P_5 require that R be a commutative group under addition. If R further satisfies the following:

P_9: $a \cdot b = b \cdot a$ (commutative law for multiplication)

it is called a *commutative ring.* If the commutative ring contains an identity or unity element for multiplication such that $a \cdot e = a \cdot e = a$, then the ring is called *a commutative ring with unity (or identity) element.*

2.2 Example: A Commutative Ring With a Unity Element

Consider the set $R = \{0,1,2,3,4,5\}$ (i.e., the elements of R are the integers modulo 6) and the operations addition and multiplication defined by $a + b \pmod 6$ and $a \cdot b \pmod 6$, respectively. It is simple to show that the elements of R satisfy postulates 1 to 9 and that R is a commutative ring with unity element. The modulo arithmetic (see notes below) ensures that the results of the operations remain within the group, i.e., it ensures that the set is closed under addition and multiplication (P_1 and P_6).

Notes on Modulo Arithmetic

In "modulo q arithmetic," one can subtract q or a multiple of q from the result wihtout changing the result, e.g., for $q = 6$, $5 \cdot 5 = 25 \equiv 25 - 4 \cdot 6 \equiv 1 \pmod 6$.

Similarly, one can add q or any multiple of q to the result, e.g., $1 - 5 = -4 \equiv -4 + 6 \equiv 2 \pmod 6$. By so doing, the operations addition and multiplication are warranted to be "closed."

If $a \equiv b \pmod p$, $c \equiv d \pmod p$, and m is any integer, then,

(1) $m \cdot a \equiv m \cdot b \pmod p$

(2) $a \pm c \equiv b \pm d \pmod p$

(3) $a/c \equiv b/d \pmod p$

But $n \cdot a \equiv n \cdot b \pmod p$ need not imply $a \equiv b \pmod p$ unless n and p are relatively prime, for p may not divide $a - b$ if p divides n. For example, $3 \cdot 4 \equiv 3 \cdot 9 \pmod{15}$ but $4 \not\equiv 9 \pmod{15}$; however, $3 \cdot 4 \equiv 3 \cdot 9 \pmod 5$ does imply $4 \equiv 9 \pmod 5$ for $(3,5) = 1$.

3 Field

A field is a commutative ring R with unity element in which every nonzero element has a multiplicative inverse. In other words, a commutative ring with unity element is called a field if the nonzero elements of R form a commutative group under multiplication.

3.1 Order of a Field

The order of a field is defined as the number of elements in the field. If the order is infinite, the field is an infinite field. Otherwise, the field is a finite field.

3.2 Example: A Field

Let F be the set of integers modulo 7, i.e., $F = (0,1,2,3,4,5,6)$. Let the addition and multiplication operations be defined as $a + b \equiv c \pmod 7$ and $a \cdot b \equiv c \pmod 7$, respectively. It is easy to show that F is a commutative ring with unity element 1. Also, every nonzero element has an inverse element, as evidenced by

$1 \cdot 1 = 1 \pmod 7$

$2 \cdot 4 = 1 \pmod 7$

$3 \cdot 5 = 1 \pmod 7$

$4 \cdot 2 = 1 \pmod 7$

$5 \cdot 3 = 1 \pmod 7$

$6 \cdot 6 = 1 \pmod 7$

The nonzero elements (1,2,3,4,5,6) form a multiplicative group.

3.3 Corollary

Let I be a set of integers and let p be a prime; then the set of integers modulo p (i.e., $0,1,2,\cdots p-1$) forms a finite or Galois field of order p, denoted by $GF(p)$. Using $GF(p)$ as a starting point, one can construct extension fields with p^n elements (see Section 5.1). One of the fundamental theorems of field theory states that $GF(p)$ and $GF(p^n)$, where p is prime and n is an integer, are the only possible finite fields (see Ref. 1, page 50).

3.4 Counter Example: To Show That the Set of Integers Modulo 6 Is Not a Field

The ring $R = (0,1,2,3,4,5)$ is not a finite field because some of the nonzero elements in R do not have inverses. For example, 2 does not have an inverse element, as evidenced by multiplying 2 by all elements in R:

$2 \cdot 0 \equiv 0 \pmod 6$

$2 \cdot 1 \equiv 2 \pmod 6$

$2 \cdot 2 \equiv 4 \pmod 6$

$2 \cdot 3 \equiv 0 \pmod 6$

$2 \cdot 4 \equiv 2 \pmod 6$

$2 \cdot 5 \equiv 4 \pmod 6$

3.5 Examples: Finite Rings and Fields

From Sections 2.2 and 3.2, one may generalize that the set of integers modulo any integer forms a finite ring. A set of integers modulo a prime number forms a finite field (see also Section 3.3).

4 Summary of Relationships Between Different Albegraic Structures

Algebraic Structure (Ref. 1)	*Properties*	*Remarks*
Semigroup	One operation, say, "+"; closed; associative	
Monoid	Also with identity element for "+"	
Group	Also with inverse for "+"	
Commutative or Abelian Group	Also commutative for "+"	
Ring	With two operations "+" and "·"; a commutative group under "+"	Therefore, a ring is a commutative group under "+"; a semigroup under "·"; and "+" and "·" possess the distributive property"
	Also closed and associative under "·"; "+" and "·" are distributive	
Commutative Ring	Also commutative for "·"	
Commutative Ring With Unity Element		Therefore, a commutative ring with unity element is a commutative group under "+"; a monoid under "·"
Field	Every nonzero element has multiplicative inverse, i.e., a^{-1} exists and that $aa^{-1} = 1$, where 1 is the identity (or unity) element for multiplication	Therefore, a field is a commutative group under addition and its nonzero elements form a multiplicative group
Finite Field or Galois Field	The number of elements in the field is finite	
Extension Field	$GF(p^m)$ = extension field $GF(p)$ = ground field, where p is prime and m is an integer. It can be shown that $GF(p^m)$ are the only possible finite fields (see Ref. 1)	

5 Ground Field and Extension Field

In Section 3.5, it was concluded that the set of integers modulo a prime number is a finite field, $GF(p)$. $GF(p)$ is referred to as the ground field and $GF(p^n)$ as the extension field of $GF(p)$, where n is an integer and p is a prime.

5.1 Construction of the Extension Field $GF(p^n)$, Given the Ground Field $GF(p)$

Let $p(x)$ be a monic irreducible polynomial (see notes below) of degree n over $GF(p)$, i.e.,

$$p(x) = x^n + a_{n-1}\, x^{n-1} + a_{n-2}\, x^{n-2} + \cdots + a_0\, x^0$$

where

$$a_{n-1}, a_{n-2}, \cdots, a_0 \in GF(p)$$

A root, say, α, of $p(x)$ exists and can be found in the extension field $GF(p^n)$. It can be shown that all elements in $GF(p^n)$ given by (see note below)

$$\{a_{n-1}\, \alpha^{n-1} + a_{n-2}\, \alpha^{n-2} + a_{n-3}\, \alpha^{n-3} + \cdots + a_0\, \alpha^0\} \qquad (1)$$

$$a_{n-1}, a_{n-2}, \cdots, a_0 \in GF(q)$$

satisfy the definition of a field. In general, if p is a prime and n is an integer, there always exists a Galois field of order p^n (Ref. 2).

Thus, according to (1), the elements of $GF(p^n)$ can be obtained by substituting into the polynomial $a_{n-1}\, \alpha^{n-1} + a_{n-2}\, \alpha^{n-2} + a_{n-3}\, \alpha^{n-3} + \cdots + a_0\, \alpha^0$ various values for a_{n-1}, a_{n-2}, etc. Since there are p elements in $GF(p)$, there are "p" different ways to assign a value to a_{n-1}, and similarly to a_{n-2}, a_{n-3}, $\cdots$, a_0. Thus, there are p^n different substitutions, yielding the p^n elements of $GF(p^n)$.

Notes: A monic nth degree polynomial is one in which the coefficient of the highest degree term, x^n, is unity.

An irreducible polynomial is one which cannot be factorized or one which contains no divisors except scalars and scalar multiples of itself.

A = {"a"|"a" has property P} is a mathematical shorthand which is interpreted as "A" is the collection of all elements "a" such that "a" has property P.

5.2 Example: Construct $GF(2^3)$

Rather than establishing the validity of the procedure described in Section 5.1, we shall illustrate it by an example.

Let $p = 2$ and $n = 3$. It is required to construct Galois field $GF(2^3)$.

Since $p(x) = x^3 + x + 1$ is not zero over $GF(2)$ (i.e., for $x = 0$ and $x = 1$, $p(0)$ and $p(1)$ are nonzero), $p(x)$ is said to be irreducible over $GF(2)$. Thus, there exists an $\alpha \in GF(2^3)$ such that $p(\alpha) = \alpha^3 + \alpha + 1 = 0$. Then, the set of elements of $GF(2^3)$ is given by

$$GF(2^3) = \{a_2\, \alpha^2 + a_1\, \alpha + a_0 | a_0, a_1, a_2 \in GF(2)\}$$

Or, expanding this in full, the elements of $GF(2^3) = GF(8)$ are

$$0\alpha^2 + 0\alpha + 0 = 0$$

$$0\alpha^2 + 0\alpha + 1 = 1$$

$$0\alpha^2 + 1\alpha + 0 = \alpha$$

$$0\alpha^2 + 1\alpha + 1 = \alpha + 1$$

$$1\alpha^2 + 0\alpha + 0 = \alpha^2$$

$$1\alpha^2 + 0\alpha + 1 = \alpha^2 + 1$$

$$1\alpha^2 + 1\alpha + 0 = \alpha^2 + \alpha$$

$$1\alpha^2 + 1\alpha + 1 = \alpha^2 + \alpha + 1$$

5.3 Theorem: Existence of Primitive Element(s) and Associated Cyclic Group(s)

There exists a primitive element $\alpha \in GF(p^n)$ that generates the nonzero elements of $GF(p^n)$. The nonzero elements of $GF(p^n)$ form a cyclic group of $p^n - 1$ elements.

5.4 Example

From the example given in Section 5.2, we know that $GF(2^3)$ is formed by

$$GF(2^3) = \{(a_2\, \alpha^2 + a_1\, \alpha + a_0) | a_0, a_1, a_2 \in GF(2)\}$$

where

$$p(\alpha) = \alpha^3 + \alpha^1 + 1 = 0$$

This implies $\alpha^3 = -\alpha - 1 = \alpha + 1$. Let us start from an element $\alpha \in GF(2^3)$. Then,

$$\alpha^2 = \alpha \cdot \alpha = \alpha^2$$

$$\alpha^3 = \alpha^2 \cdot \alpha = -\alpha - 1 = \alpha + 1$$

$$\alpha^4 = \alpha^3 \cdot \alpha = (\alpha + 1)\alpha = \alpha^2 + \alpha$$

$$\alpha^5 = \alpha^4 \cdot \alpha = (\alpha^2 + \alpha)\alpha = \alpha^3 + \alpha^2$$

$$= \alpha + 1 + \alpha^2 = \alpha^2 + \alpha + 1$$

$$\alpha^6 = \alpha^5 \cdot \alpha = (\alpha^2 + \alpha + 1)\alpha$$

$$= \alpha^3 + \alpha^2 + \alpha = \alpha + 1 + \alpha^2 + \alpha = \alpha^2 + 1$$

$$\alpha^7 = \alpha^6 \cdot \alpha = (\alpha^2 + 1)\alpha = \alpha^3 + \alpha$$

$$= \alpha + 1 + \alpha$$

$$= 1$$

Thus, the nonzero elements of $GF(2^3)$ as obtained in Section 5.2 can be written in the form:

$$(\alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^5, \alpha^6, \alpha^7)$$

or

$$(\alpha, \alpha^2, \alpha + 1, \alpha^2 + \alpha, \alpha^2 + \alpha + 1, \alpha^2 + 1, 1)$$

Hence, since α generates all the nonzero elements of $GF(2^3)$, α is a primitive element of $GF(2^3)$ and $\alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^5, \alpha^6, \alpha^7$ form a *cyclic group** of $2^3 - 1$ elements in $GF(2^3)$.

5.5 Example: A Primitive Element Generating a Cyclic Subgroup

Let $GF(p)$ be the ground field. As mentioned in Section 5.3, there always exists a primitive element which generates one cyclic subgroup of order $p - 1$. For example, let the ground field be $GF(7) = (0,1,2,3,4,5,6)$. It can be shown that $\gamma = 3$ is a primitive element, and the cyclic subgroup generated is

$$G_{p-1} = G_6 = (1,2,3,4,5,6)$$

This is so because

$$3^1 = 3$$

$$3^2 = 9 \equiv 2 \pmod 7 = 2$$

$$3^3 = 3 \cdot 3^2 = 3 \cdot 2 = 6$$

$$3^4 = 3 \cdot 6 = 4$$

$$3^5 = 5$$

$$3^6 = 1$$

$G_{p-1} = G_6 = (3,3^2,3^3,3^4,3^5,3^6 = 1)$, where 3 is a primitive element in $GF(7)$. Since 3 generates the cyclic subgroup of order 6 of $GF(7)$, it is *also* called the generator of G_6.

Note: To evalute 5^5, make use of the fact that $5^4 = 2$ is known.

Hence, $5^5 = 5 \cdot 5^4 = 5 \cdot 2 = 10 = 3$ in modulo 7 arithmetic.

5.6 Additional Examples: To Illustrate the Slight Difference Between a Primitive Element and a Generator of a Subgroup

In $GF(7)$, 4 is a generator of a subgroup of order 3, but is *not* a primitive element, as evidenced by the fact that

$$4 = 4$$

$$4^2 = 2$$

$$4^3 = 1$$

Once this point is reached, further multiplication of 4 by itself will not generate any new elements since $4^4 = 4^3 \cdot 4 \equiv 4 \pmod 7$, $4^5 = 4^3 \cdot 4^2 \equiv 2 \pmod 7$, etc. Thus, 4 generates the cyclic subgroup of order 3, $G_3 = \{4,2,1\}$.

*Since $\alpha^7 = 1$, $\alpha^8 = \alpha \cdot \alpha^7 = \alpha$, $\alpha^9 = \alpha^7 \cdot \alpha^2 = \alpha^2$, etc. Thus, after the first 7 elements, *no* new elements will be generated. In coding terminology, if every cyclic shift of a code word gives another, the code is a cyclic code (see Ref. 2, p. 124). By analogy, the group generated by α is referred to as the cyclic group.

However, 5 is a primitive element as evidenced by the fact that:

$5 = 5$

$5^2 = 4$

$5^3 = 6$

$5^4 = 2$

$5^5 = 3$

$5^6 = 1$

To recap, a generator in GF(p) is one which generates a cyclic subgroup of GF(p). The generator whose cyclic group is of order p - 1 is called a primitive element of GF(p). An additional example is given in Section 7.3.

5.7 The Number of Primitive Elements in *GF(p)*

This section briefly shows how the number of primitive elements in a given field $GF(p)$ is calculated (see Ref. 3, pp. 23 and 47). It turns out that the number of primitive elements of $GF(p)$ is given by $\phi(p - 1)$, where $\phi(m)$ is called Euler's ϕ-function and is defined as the number of positive integers less than or equal to m that are relative prime to m.

For example, consider $GF(7)$ when $p = 7$ and $p - 1 = 6$. Here, $\phi(6)$ is 2 since there are only two numbers which are relative prime to 6, namely, 1 and 5. It has been demonstrated (Sections 5.5 and 5.6) heuristically that 3 and 5 are only *two* primitive elements in $GF(7)$.

6 Definition of the Order of an Element

Let γ be a nonzero element in $GF(p)$, and let d be the smallest integer such that $\gamma^d = 1$. The order of γ is then d.

6.1 Example: Primitive Element and Order of an Element

It was shown in Section 5.5 that 5 is an element of $GF(7)$ of order 6 for $5^6 = 1$. Also, since the cyclic group of 5 is of order $7 - 1 = 6$, 5 is a primitive element of $GF(7)$. Further, since $2^3 = 8 \equiv 1 \pmod 7$, 2 is an element of order 3.

7 Theorem on Cyclic Subgroups of a Given Field *GF(p)*

The following theorem enables one to obtain all of the cyclic subgroups of a given field $GF(p)$. It states:

Let $GF(p)$ be a finite field. If d divides $p - 1$, then $GF(p)$ has an element γ of order d. This element γ is a generator of cyclic subgroup $G_d \subset GF(p)$, where $G_d = (\gamma, \gamma^2, \cdots, \gamma^d = 1)$. (Note that the symbol $\subset$ means G_d is a subset of $GF(p)$, or G_d is contained in $GF(p)$). Stated differently, the theorem says that the order of every element must divide $p - 1$, where p is the order of the group.

7.1 Example: Application of the Theorem Given in Section 7

Consider $GF(7) = (0,1,2,3,4,5,6)$. Since $p = 7$, $p - 1 = 6$. Consider $d = 3$; since $(p - 1)/d = 6/3 = 2$, d divides $p - 1$, and one can conclude from the above theorem that there exists a subgroup G_3 of $GF(7)$, i.e., G_3 is a subset of the elements of $GF(7)$.

7.2 Plausible "Proof" of the Theorem Given in Section 7

Consider $GF(p)$ which has a primitive element γ such that

$$\gamma^{p-1} = 1 \tag{2}$$

Also, let d divide $p - 1$ such that

$$\frac{p-1}{d} = m \quad \text{or} \quad p - 1 = dm \tag{3}$$

Substituting Eq. (3) into Eq. (2), we have

$$\gamma^{p-1} = \gamma^{dm} = 1 \quad \text{or} \quad (\gamma^m)^d = 1$$

Thus, there always exists an element γ^m of order d. Also, the element γ^m will generate the cyclic subgroup $\{(\gamma^m)^1, (\gamma^m)^2, (\gamma^m)^3, \cdots, (\gamma^m)^d\}$ of order d (since there are d elements). Hence, the theorem in Section 7 is "proved."

To further illustrate the use of the theorem, consider the nonzero elements of $GF(7)$ denoted by G_{p-1}:

$$G_6 = (1,2,3,4,5,6)$$

$$= (3,3^2,3^3,3^4,3^5,3^6 = 1)$$

For $d = 3$, since $p - 1/d = 6/3 = 2$, $\gamma^6 = \gamma^{3 \cdot 2} = (\gamma^2)^3 = 1$, one can conclude that there exists a primitive element $\gamma^2 = 3^2$ of order 3 that generates the elements $((3^2)^1, (3^2)^2, (3^2)^3 = 1)$, which constitute the subgroup G_3. Thus,

$$G_d = G_3 = ((3^2)^1, (3^2)^2, (3^2)^3 = 1)$$

$$= (2, 2^2, 2^3 = 1) \text{ since } 3^2 = 2$$

Here, $\gamma = 2$ is called the element of the cyclic subgroup G_3, and since $\gamma^d = 2^3 = 1$, the order of γ is 3.

7.3 Distinction Between a Primitive Element α and a Generator γ

Let α denote the primitive element that generates *all* of the nonzero elements $(\alpha^1, \alpha^2, \cdots, \alpha^{p-1})$ of $GF(p)$. The order of α is $p - 1$. Further, let γ denote the generator of a subgroup G_d of $GF(p)$. If d divides $p - 1$, it was shown in Section 7.2 that there exists an element γ such that $\gamma^d = 1$, and such that γ will generate *some* of the elements of $GF(p)$ but *all* of the elements of G_d, i.e., $G_d = (\gamma, \gamma^2, \cdots, \gamma^d)$, where G_d is a subgroup of $GF(p)$. Here, d is the order of the element γ, and also the order of the subgroup G_d.

7.4 Different Terminology for γ

Element γ has many names. It is referred to as:

(1) The generator of G_d, since $G_d = (\gamma, \gamma^2, \cdots, \gamma^d)$.

(2) The "dth root of unity," since $\gamma^d = 1$ and, hence, $\gamma = \sqrt[d]{1}$.

(3) An element of order d, since d is the smallest positive integer such that $\gamma^d = 1$.

7.5 Different Terminology for d

Integer d has many names. It is referred to as:

(1) The order of the subgroup G_d, since G_d has d elements.

(2) The order of the element γ, since $\gamma^d = 1$.

(3) The transform length, in the context of the study of "DFT over finite fields" (see Section 8).

Application of Finite Fields to Discrete Fourier Transforms

Finite fields have applications in many areas of modern studies. Among them is the area of Fourier transforms (Refs. 4-11). We shall now describe how finite fields are applied to the evaluation of Fourier transforms.

8 The Discrete Fourier Transform Over a Finite Field *GF(p)*

The discrete Fourier transform (DFT) of an integer sequence $a_0, a_1, a_n, \cdots, a_{d-1}$ is defined with respect to G_d, where G_d is a subgroup of $GF(p)$ (Refs. 4 and 5). The order d of the subgroup G_d determines the number of members in sequence a_n and is referred to as the transform length. Thus, the DFT of sequence a_n over G_d of $GF(p)$ is defined as

$$A_k = \sum_{n=0}^{d-1} a_n \gamma^{nk}, \quad 0 \leqslant k \leqslant d-1 \tag{4}$$

where

$a_0, a_1, a_2, \cdots, a_{d-1}$ is the given integer sequence whose DFT is desired, and $a_n \in GF(p)$.

γ is an element of $GF(p)$ and is the generator of G_d such that $G_d = (\gamma^1, \gamma^2, \cdots, \gamma^d = 1)$.

d is the order of the element γ and also the order of the cyclic subgroup G_d generated by γ, such that $\gamma^d = 1$, and there are d elements in G_d. Here, d is referred to as the transform length and determines the number of members in sequence a_n and in its transformed sequence A_k.

$A_0, A_1, \cdots, A_{d-1}$ is the transform of $a_0, a_1, \cdots, a_{d-1}$, and $A_k \in GF(p)$.

It can be shown (Ref. 7) that the inverse transform is given by

$$a_n = (d)^{-1} \sum_{k=0}^{d-1} A_k \gamma^{-nk}, \text{ for } 0 \leqslant n \leqslant d-1 \tag{5}$$

where

$A_0, A_1, \cdots, A_{d-1}$ is the given transform, the inverse transform of which is desired.

(d) is the residue of d modulo p, for $d \leqslant p$, $(d) = d$.

$(d)^{-1}$ is the inverse element of d.

If d is a power of two, it is well known (Refs. 6 and 7) that the fast Fourier transform (FFT) algorithm can be utilized to realize the needed transforms.

8.1 Choice of γ and d

Considering how the transform and the inverse transform are computed (see Eqs. 4 and 5), it is advantageous to choose $\gamma = 2$ or power of 2 and the order of 2 is a power of 2 because multiplication in these cases means "shifting" in actual logic implementation, and the most efficient FFT algorithm can be used to yield a fast transform.

Also, it is advantageous to choose d to be a power of 2. This is because fast Fourier transform techniques can then be

applied. The reader is referred to Ref. 6 for further elucidation on this point.

8.2 Finite Field With p Equal to a Fermat Prime

A Fermat prime, F_n, is a prime number defined by

$$F_n = 2^{2^n} + 1, \quad \text{for} \quad n = 1,2,3,4$$

We shall consider $GF(p)$, where $p = F_n$. This is because the values of γ and d resulting from $GF(F_n)$ will have the desired properties discussed in Section 8 (see Section 8.3).

8.3 To Show That $GF(F_n)$ Has an Element $\gamma=2$ and That the Order of γ Is a Power of 2

Let $\gamma \in GF(F_n)$ and d be a power of 2. There exists a theorem (see Ref. 7) which states that if

$$\gamma^{d/2} \equiv -1 \pmod{p} \tag{6}$$

then γ is an element of order d.

Making use of this theorem, choosing $p = F_n$, since

$$F_n = 2^{2^n} + 1 \tag{7}$$

$$2^{2^n} \equiv -1 \pmod{F_n} \tag{8}$$

Comparing Eqs. (6) and (8), we have

$$\gamma^{d/2} = 2^{2^n}$$

where

$$\gamma = 2$$

and

$$d = 2^{n+1}$$

Hence, $GF(F_n)$ will have an element equal to 2, whose order is $d = 2^{n+1}$. Also, γ will generate a cyclic subgroup G_d of $GF(F_n)$, where

$$G_d = G_{2^{n+1}} = (2^1, 2^2, \cdots, 2^{2^{n+1}} = 1)$$

8.4 Example: A Finite Field With $\gamma=2$, and d Being a Power of 2

Consider $n = 2$, such that $F_n = 2^{2^2} + 1 = 17$. The theorem predicts that there exists an element $\gamma = 2$ of order $2^{n+1} = 2^{2+1} = 8$, that this is so is evident by

$$\gamma^d = 2^8 = 256$$

Since 256/17 = 15 with a residue of 1,

$$256 \equiv 1 \pmod{17}$$

that is,

$$\gamma^d = 2^8 \equiv 1 \pmod{17}$$

Here, γ (= 2) has order d (= 8) which is a power of 2. Also,

$$G_d = G_{2^3} = G_8 = (2^1, 2^2, 2^3, \cdots, 2^7, 2^8 = 1)$$

8.5 Example: To Obtain the DFT Over $GF(5)$ of the Sequence $(a_0=1, a_1=1, a_2=3, a_3=1)$

Since $2^{2^n} + 1 = 5$ for $n = 1, 5$ is a Fermat prime. We expect $\gamma = 2$ to be a generator, and that it will have order $d = 2^{n+1} = 2^{1+1} = 4$. Also, γ generates a cyclic subgroup $G_d = G_4 \subset GF(5)$.

We shall proceed to obtain the DFT over G_4, which is a cyclic subgroup of $GF(5)$. Repeating Eq. (4),

$$A_k = \sum_{n=0}^{d-1} a_n \gamma^{nk}, \quad 0 \leqslant k \leqslant d-1$$

for $d = 4$, $\gamma = 2$, $A_k = a_0 + a_1 2^k + a_2 2^{2k} + a_3 2^{3k}$.

Given $a_0 = 1, a_1 = 1, a_2 = 3, a_3 = 1$, the transform of this integer sequence is

$$A_0 = 1 + 1 + 3 + 1 = 6 \equiv 1 \pmod{5}$$

$$A_1 = 1 + 1 \cdot 2 + 3 \cdot 2^2 + 1 \cdot 2^3 \equiv 3 \pmod{5}$$

$$A_2 = 1 + 1 \cdot 2^2 + 3 \cdot 2^4 + 1 \cdot 2^6 \equiv 2 \pmod{5}$$

$$A_3 = 1 + 1 \cdot 2^3 + 3 \cdot 2^6 + 1 \cdot 2^9 \equiv 3 \pmod{5}$$

For the above computation, we make use of the fact that $\gamma^d = 2^4 = 1$ to simplify the arithmetic.

8.6 Example: To Obtain the DFT^{-1} (Inverse Transform) of the Sequence (A_0=1, A_1=3, A_2=2, A_3=3)

It is obvious that if the modulo arithmetic is performed correctly, the inverse transform should be $a_0 = 1, a_1 = 1, a_2 = 3, a_3 = 1$.

Before substituting in Eq. (5) which is

$$a_n = (d)^{-1} \sum_{k=0}^{d-1} A_k \gamma^{-nk}, \text{ for } 0 \leqslant n \leqslant d-1$$

it is necessary to evaluate $(d)^{-1}$, which is the inverse of d. For $\gamma = 2, d = 2^2 = 4$,

$$(d)^{-1} = 2^{-2}$$

Since

$$(d) \cdot (d)^{-1} = (2^2) \cdot (2^{-2}) = 1$$

and

$$(d)^{-1} = 2^{-2} = 2^{-2}2^4 \equiv 4 \equiv -1 \pmod 5$$

then

$$(d)^{-1} \equiv -1 \pmod 5$$

Note: Since $\gamma^d = 1$, $2^4 = 1$. Also, $4 = -1$ in modulo 5 arithmetic.

Thus,

$$a_n = -1 (A_0 + A_1 \cdot 2^{-n} + A_2 \cdot 2^{-2n} + A_3 \cdot 2^{-3n})$$

$$= -1 (1 + 3 \cdot 2^{-n} + 2 \cdot 2^{-2n} + 3 \cdot 2^{-3n}),$$

for $n = 0, 1, 2, 3$

or

$$a_0 = -(1 + 3 + 2 + 3) = -4 = 1$$

$$a_1 = -(1 + 3 \cdot 2^{-1} + 2 \cdot 2^{-2} + 3 \cdot 2^{-3})$$

$$= -(1 + 3 \cdot 2^3 + 2 \cdot 2^2 + 3 \cdot 2^1) = 1$$

$$a_2 = -(1 + 3 \cdot 2^{-2} + 2 \cdot 2^{-2 \cdot 2} + 3 \cdot 2^{-3 \cdot 2})$$

$$= -(1 + 3 \cdot 2^2 + 2 + 3 \cdot 2^2) = 3$$

$$a_3 = -(1 + 3 \cdot 2^{-3} + 2 \cdot 2^{-2 \cdot 3} + 3 \cdot 2^{-3 \cdot 3})$$

$$= -(1 + 3 \cdot 2 + 2 \cdot 2^2 + 3 \cdot 2^3) = 1$$

Hence,

$$(a_0, a_1, a_2, a_3) = (1, 1, 3, 1)$$

Note: For the above computation, again, we use the fact that $2^{-n} = 2^{-n} \cdot 2^4$. This is because $\gamma^d = 2^4 \equiv 1 \pmod 5$.

9 Convolution Over $GF(F_n)$

Let $a_0, a_1, \cdots, a_d$ and $b_0, b_1, \cdots, b_d$ be two sequences of integers (where d will later be referred to as the transform length). The discrete convolution is defined as

$$c_p = \sum_{n=0}^{d-1} a_n b_{(p-n)}, \quad p = 0, 1, 2, \cdots, d-1 \qquad (9)$$

where $(p - n)$ denotes the residue of $p - n$ modulo d.

9.1 Example: Direct Evaluation of the Convolution

Let $a_0 = 1, a_1 = 1, a_2 = 0, a_3 = 0$ and $b_0 = 1, b_1 = 1, b_2 = 0, b_3 = 0$.

Compute the discrete convolution,

$$c_p = \sum_{n=0}^{4-1} a_n b_{(p-n)}, \text{ where } d = 4$$

or

$$c_p = a_0 b_{(p-0)} + a_1 b_{(p-1)} + a_2 b_{(p-2)} + a_3 b_{(p-3)}, \qquad (10)$$

for $p = 0, 1, 2, 3$

or

$$c_0 = a_0 b_{(0)} + a_1 b_{(-1)} + a_2 b_{(-2)} + a_3 b_{(-3)}$$

$$= a_0 b_0 + a_1 b_3 + a_2 b_2 + a_3 b_1$$

(Note that $b_{-n} = b_{-n+d}$; hence, $b_{-3} = b_1$)

$$= 1 \cdot 1 + 1 \cdot 0 + 0 \cdot 0 + 0 \cdot 1 = 1$$

$$c_1 = a_0 b_{(1)} + a_1 b_0 + a_2 b_{(-1)} + a_3 b_{(-2)}$$

$$= a_0 b_1 + a_1 b_0 + a_2 b_3 + a_3 b_2$$

$$= 1 \cdot 1 + 1 \cdot 1 + 0 \cdot 0 + 0 \cdot 0 = 2$$

$$c_2 = a_0 b_2 + a_1 b_{(1)} + a_2 b_0 + a_3 b_{(-1)}$$

$$= a_0 b_2 + a_1 b_1 + a_2 b_0 + a_3 b_3$$

$$= 1 \cdot 0 + 1 \cdot 1 + 0 \cdot 1 + 0 \cdot 0 = 1$$

$$c_3 = a_0 b_3 + a_1 b_2 + a_2 b_1 + a_3 b_0$$

$$= 1 \cdot 0 + 1 \cdot 0 + 0 \cdot 1 + 0 \cdot 1 = 0$$

Hence,

$$c_0 = 1, c_1 = 2, c_2 = 1, c_3 = 0$$

9.2 Evaluation of the Convolution by the Discrete Frontier Transform Method

The discrete convolution of sequences a_n and b_n can be computed by DFT over $GF(F_n)$. To do this, we compute the discrete Fourier transform over $GF(F_n)$ of sequences a_n and b_n, respectively, i.e.,

$$A_k = \sum_{n=0}^{d-1} a_n \gamma^{nk}$$

and

$$B_k = \sum_{n=0}^{d-1} b_n \gamma^{nk}$$

Let

$$C_k = A_k \cdot B_k$$

It can be shown that the discrete convolution over $GF(F_n)$ can be obtained by taking the inverse transform of C_k, i.e.,

$$c_p = (d)^{-1} \sum_{k=0}^{d-1} C_k \gamma^{-kn} = a_n b_{(p-n)}$$

9.3 Determination of the Dynamic Range

In order to avoid overflow (i.e., to avoid c_p assuming integer values outside of that allowed by $GF(F_n)$), it is necessary to keep

$$-\frac{F_n - 1}{2} \leqslant c_p \leqslant \frac{F_n - 1}{2}$$

To achieve this, limits have to be imposed on sequences a_n and b_n. Since

$$|c_p| \leqslant \frac{F_n - 1}{2}$$

$$\left|\sum_{n=0}^{d-1} a_n b_{(p-n)}\right| \leqslant \sum_{n=0}^{d-1} |a_n| \cdot |b_{(p-n)}| \leqslant \frac{F_n - 1}{2}$$

Let

$$|a_n| \leqslant A, |b_n| \leqslant B, \text{ for } n = 0, 1, \cdots, d-1$$

Thus,

$$\left|\sum_{n=0}^{d-1} a_n b_{(p-n)}\right| \leqslant d \cdot A \cdot B \leqslant \frac{F_n - 1}{2}$$

If $A = B$, then

$$A = \left[\sqrt{\frac{F_n - 1}{2d}}\right]$$

where $[x]$ denotes the greatest integer less the x, and A is called the dynamic range.

Therefore, if

$$-A \leqslant a_n, b_n \leqslant A, \text{ for } n = 0,1,2, \cdots, d-1$$

then

$$-\frac{F_n - 1}{2} \leqslant c_p \leqslant \frac{F_n - 1}{2}$$

9.4 Example: Computation of Convolution of DFT and Determination of the Dynamic Range

Let $a_0 = 1, a_1 = 1, a_2 = 0, a_3 = 0$ and $b_0 = 1, b_1 = 1, b_2 = 0, b_3 = 0$. Compute the discrete convolution by using DFT over $GF(2^2 + 1)$.

Since $F_1 = 2^{2^1} + 1 = 5$, the dynamic range is

$$A = \left[\sqrt{\frac{F_n - 1}{2 \cdot 2}}\right] = \left[\sqrt{\frac{4}{4}}\right] = 1$$

Note: Although $d = 4$, we use an "effective $d = 2$." This is because the two given sequences a_n and b_n possess zeros in the last two terms. It is obvious, by studying Eq. (10), for example, that zeros in a_n and b_n will reduce the number of terms. In general, if the number of nonzero elements in a_n and b_n are n_a and n_b, the "effective d" is equal to the larger of n_a and n_b.

Hence,

$$-1 \leqslant a_n, b_n \leqslant 1, \text{ for } n = 0,1,2,3 \qquad (11)$$

and

$$-\frac{5 - 1}{2} \leqslant c_p \leqslant \frac{5 - 1}{2} \qquad (12)$$

The given sequences a_n and b_n are seen to satisfy the constraint (11).

The DFT over $GF(5)$ of a_n are

$$A_k = \sum_{n=0}^{4-1} a_n 2^{nk} = a_0 + a_1 2^k + a_2 2^{2k} + a_3 2^{3k}$$

where 2 is an element of order 4 in $GF(5)$. It follows that $A_k = 1 + 1 \cdot 2^k$ and $A_0 = 1 + 1 \cdot 2^0 = 2, A_1 = 1 + 2 = 3, A_2 = 1 - 1 = 0, A_3 = 1 + 3 = 4$.

Similarly, the DFT of sequence b_n is

$$B_0 = 2, B_1 = 3, B_2 = 0, B_3 = 4$$

But

$$C_k = A_k \cdot B_k$$

Therefore, the DFT of the convolution modulo 5 is

$$C_0 = 4, \quad C_1 = 4, \quad C_2 = 0, \quad C_3 = 1$$

The inverse transform of C_k is

$$c_k = (4)^{-1} \cdot \sum_{n=0}^{4-1} C_k 2^{-nk}$$

$$= -1(C_0 + C_1 2^{-k} + C_2 2^{-2k} + C_3 2^{-3k})$$

$$= -(4 + 4 \cdot 2^{-k} + 2^{-3k}), \text{ for } k = 0, 1, 2, 3$$

or

$$c_0 = -(4 + 4 + 1) = 1$$

$$c_1 = -(4 + 4 \cdot 2^{-1} + 2^{-3}) = -(1 + 2) = 2$$

$$c_2 = -(4 + 4 \cdot 2^{-2} + 2^{-6}) = -(4 + 1 + 2^2) = 1$$

$$c_3 = -(4 + 4 \cdot 2^{-3} + 2^{-9}) = -(4 + 4 \cdot 2 + 2^3)$$

$$= -(4 + 3 + 3) = 0$$

It is seen that the values of c_n's remain within the dynamic range specified by (12).

Note: In the above computation, the usual tricks in finite field arithmetic have been used, namely,

(1) Adding to or subtracting from the result by "multiples of $5 \equiv 0 \pmod 5$."

(2) Multiplying the result by $\gamma^d = 2^4 \equiv 1 \pmod 5$ to get an appropriate finite field element.

This is done "to bring the result of the computation to within the field."

10 Arithmetic Operations Needed to Compute DFT Over $GF(F_n)$

The arithmetic operations needed are:

(1) Negation modulo F_n

(2) Integer addition modulo F_n

(3) Multiplication modulo F_n

(4) Multiplication by power of 2

We shall illustrate the algorithms by examples.

Consider $GF(F_1) = GF(2^2 + 1)$; a 3-bit word length is required to represent all the $2^2 + 1$ elements. In general, for $GF(F_n) = GF(2^{2^n} + 1)$, $2^n + 1$ bits are required to represent all of the $2^{2^n} + 1$ elements. (Note that if there are 2^{2^n} elements, 2^n bits are required, and there is no unused state. Since there are $2^{2^n} + 1$ elements, the extra element calls for one additional bit, with $2^{2^n+1} - 2^{2^n} - 1$ unused states.)

10.1 Negation Modulo F_n

The problem can be stated as follows: Given an element "a" in $GF(F_n)$, what is the procedure (or algorithm) for computing "$-a$"?

Solution: Consider the specific example $a = 2$ such that $2 \in GF(5)$. -2 can be computed by subtracting 2 from 5, i.e., $-2 = 5 - 2 = 3$.

Hence, the algorithm is: Given $a \in GF(F_n)$, to obtain $-a$, use the property that $-a = F_n - a$.

10.2 Addition Modulo F_n

The problem can be stated as follows: Given elements $a,b \in GF(F_n)$, what is the algorithm for computing $(a + b)$ modulo F_n?

Solution: Consider the specific example $a = 3$, $b = 4$, $F_n = F_1 = 5$.

In modulo arithmetic $3 + 4 = 7 \equiv 2 \pmod 5$. The algorithm may be illustrated as follows:

	2^3	2^2	2^1	2^0
$a = 3$		0	1	1
$+b = 4$		1	0	0
$c = 7$		(1)	1	1
		-		1
$c = 7 \equiv 2 \pmod 5$		0	1	0

Thus, the algorithm is: if the 2^2th bit is a 1 (and the 2^1th bit and 2^0th bit are both not equal to zero), discard the 1 in the 2^2-bit position, and subtract 1 from the 2^0-bit position. The above parenthesized condition helps to exclude the case of 100 when a modulo operation is not needed.

Justification for the algorithm: When the number is 5 or more, "discarding the 1 in the 2^2-bit position and subtracting 1 from the 2^0-bit position" is equivalent to subtracting $2^2 = 4$ and then $2^0 = 1$ from the result, i.e., subtracting a total of 5.

Generalized algorithm: To perform modulo F_n addition, let $m = n + 1$. If the 2^m-bit position is a 1 (and at least one other bit position is a 1), discard the 1 in the 2^m-bit position, and subtract 1 from the 2^0-bit position.

10.3 Multiplication Modulo F_n

The problem can be stated as follows: Given $a,b \in GF(F_n)$, what is the algorithm for computing $a \cdot b$ modulo F_n?

Solution: As before, perform the binary multiplication and do a modulo F_n arithmetic. For $F_n = 5$, discard the 1 in the 2^2-bit position and subtract 1 from the least significant position, e.g.,

		2^2	2^1	2^0
$a = 3$		0	1	1
$b = 2$		0	1	0
$a \cdot b$	0	(1)	1	0
				-1
$a \cdot b \pmod 5 = 1$		0	0	1

10.4 Multiplication by Powers of 2 in Modulo F_n Arithmetic

The problem can be stated as follows: Given that "a" belongs to $GF(F_n)$ and any integer m, what is the algorithm for computing $a \cdot 2^m$ modulo F_n?

Solution: Let $a = 3, m = 2, n = 1$

	2^4	2^3	2^2	2^1	2^0
$a = 3$:	0	0	0	1	1
	0	(1)	(1)	0	0
-				1	1
-				1	1

$a \cdot 2^m$ is equivalent to shifting left by m positions

The explanation is as follows: Since

$2^2 \equiv -1 \pmod 5$

$2^3 \equiv -2 \pmod 5$

discarding the 1 in the 2^2th position and minus 1 is equivalent to taking a modulo 5. Similarly, discarding a 1 in the 2^3th position and minus 2 is also equivalent to taking a modulo 5.

11 Extension of the Dynamic Range of c_n* Using the Chinese Remainder Theorem

Section 9-3 shows that to avoid "overflow," i.e., to avoid $|c_n|$'s exceeding $(F_n - 1)/2$, $|a_n|$'s and $|b_n|$'s are kept below the value

$$A = \left[\sqrt{\frac{F_n - 1}{2d}}\right]$$

Conversely, if $|a_n|$'s and $|b_n|$'s exceed A, the $|c_n|$'s will exceed its dynamic range $(F_n - 1)/2$. In order to preserve precision, it is often necessary to extend the dynamic range for the c_n's.

The method for increasing the dynamic range for the c_n's (Ref. 8) is as follows: Obtain the convolutions of the a_n's and b_n's twice – once over the finite field $GF(F_n)$ and once over the finite field $GF(F_m)$, where $m \neq n$. It can be shown that the convolution of the two sequences $(C_0)_1, (C_1)_1, \cdots, (C_d)_1$ and $(C_0)_2, (C_1)_2, \cdots, (C_d)_2$ over the ring $R(F_n \cdot F_m)$ denoted by $C_0, C_1, \cdots, C_d$ can be computed using the Chinese Remainder Theorem (see Ref. 3, p. 31). The dynamic range is now from "zero to $F_n \cdot F_m - 1$."

*c_n is the convolution of two sequences a_n and b_n.

11.1 The Chinese Remainder Theorem

This theorem provides an efficient method for solving a certain kind of problem, e.g., find x given that the remainders are 1 and 2 when x is divided by 3 and 4, or find all integers that have remainders 1 or 2 when they are divided by each of 3, 4, and 5 (see Ref. 3, p. 31).

The theorem states: Let $p_1, p_2, p_3, \cdots, p_k$ be integers which are relative prime in pairs (i.e., taking any two numbers in the list, say, p_i and p_j, there is no common factor between p_i and p_j other than 1, e.g., $p_i = 8, p_j = 9$ are relative prime, although 8 and 9 themselves are not prime numbers.) Also, let

$$p = p_1 p_2 p_3 \cdots p_k = p_1 m_1 = p_2 m_2 = \cdots = p_k m_k$$

If

$$x \equiv c_1 \pmod{p_1}$$

$$x \equiv c_2 \pmod{p_2}$$

$$\vdots$$

$$x \equiv c_k \pmod{p_k}$$

the solution for x, which lies in the range

$$0 \leq x < p_1 p_2 \cdots p_k$$

is

$$x = \sum_{i=1}^{k} c_i m_i m_i^{-1}$$

where m_j^{-1} satisfies the relation $m_j m_j^{-1} \equiv 1$ modulo p_j for $j = 1, 2, \cdots, k$.

11.2 Example: Application of the Chinese Remainder Theorem

Given $x \equiv 2$ modulo 3 and $x \equiv 1$ modulo 5, find $x \equiv a$ modulo $(3 \cdot 5)$.

Solution: By the Chinese Remainder Theorem,

$$x = \sum_{i=1}^{2} c_i m_i m_i^{-1} = c_1 m_1 m_1^{-1} + c_2 m_2 m_2^{-1}$$

$$= 2m_1 m_1^{-1} + m_2 m_2^{-1}$$

For $m_1 = 5$ and $p_1 = 3$, since

$$m_1 m_1^{-1} \equiv 1 \pmod{p_1}$$

so that

$$5m_1^{-1} \equiv 1 \pmod{3}$$

we get

$$m_1^{-1} = 5^{-1} \equiv 2 \pmod{3}$$

using the Fermat theorem in Section 11.4. Similarly,

$$m_2^{-1} = 3^{-1} \equiv 2 \pmod{5}$$

Hence,

$$x = 2 \cdot 5 \cdot 2 + 1 \cdot 3 \cdot 2 = 26 \equiv 11 \pmod{3 \cdot 5}$$

11.3 Example: To Illustrate How the Dynamic Range of the Convolution of Two Sequences Can be Extended

Problem: Let $a_0 = 4$, $a_1 = 0$, $b_0 = 2$, $b_1 = 0$ be two given sequences. Compute the convolutions of a_n and b_n using DFT over $GF(3)$ and $GF(5)$.

Solution: Let us first compute the convolutions c_0, c_1 directly without using DFT. From

$$c_p = \sum_{n=0}^{d-1} a_n b_{(p-n)}$$

we have

$$c_0 = a_0 b_0 + a_1 b_{-1} = 8, \text{ for } d = 2, p = 0$$
(note $b_{-1} = b_1$, see Section 9.1)

$$c_1 = a_0 b_1 + a_1 b_0 = 0, \text{ for } d = 2, p = 1$$

It is seen that $0 \leqslant c_0, c_1 \leqslant 14$, i.e., the dynamic range of the c_p's is from 0 to 14. To obtain the convolution using DFT over $GF(3)$ only or using DFT over $GF(5)$ only would not be good enough since the required dynamic range exceeds that provided by $GF(3)$ and $GF(5)$.

In order to get the required dynamic range, one should use DFT over a Galois field of order 15; but 15 is not prime and no such Galois field exists. To overcome this problem, we obtain two convolutions, one over $GF(3)$ and another one over $GF(5)$, and then use the Chinese Remainder Theorem to compute c_0 and c_1. The direct sum of Galois fields $GF(3)$ and $GF(5)$ is isomorphic to the ring $R(15)$.*

It can be shown that the convolutions over $GF(3)$ are:

$$(c_0)_3 \equiv 2 \pmod{3}$$

$$(c_1)_3 \equiv 0 \pmod{3}$$

and the convolutions over $GF(5)$ are:

$$(c_0)_5 \equiv 3 \pmod{5}$$

$$(c_1)_5 \equiv 0 \pmod{5}$$

Using the Chinese Remainder Theorem,

$$c_0 = 2 \cdot 5 \cdot 2 + 3 \cdot 3 \cdot 2 = 38 \equiv 8 \pmod{15}$$

$$c_1 = 0 \cdot 5 \cdot 2 + 0 \cdot 3 \cdot 2 \equiv 0 \pmod{15}$$

Since the last step uses "modulo 15" arithmetic, and 15 is not prime, we say that we have performed a convolution using DFT defined over the ring $R(15)$.

*This follows from a theorem given in Ref. 12 which states: Let q_i be any prime and $q = q_1 \cdot q_2 \cdots q_r$. Further, suppose $d | q_i - 1$ for all i. Then, a d-point transform on ring $R(q)$ and its inverse transforms exist. The inverse of the above is also true.

11.4 Fermat's Theorem

For every integer a and a prime p, if p is relatively prime to a, then $a^{p-1} \equiv 1 \pmod{p}$ or $a^{-1} \equiv a^{p-2} \pmod{p}$.

If $a \in GF(p)$, it can be shown that $a^{-1} = a^{p-2}$. Thus, given 5 in $GF(3)$, $5^{-1} = 5^{3-2} = 5 \equiv 2 \pmod{3}$.

Proof: If $a \in GF(p)$,

$$a^{p-1} \equiv 1 \pmod{p}$$

$$aa^{p-2} \equiv 1 \pmod{p}$$

But

$$aa^{-1} \equiv 1 \pmod{p}$$

Hence,

$$a^{-1} \equiv a^{p-2} \pmod{p}$$

Application of Finite Fields to Reed-Solomon Coding

Another important area of application of finite fields is coding. Since Reed-Solomon (RS) codes are of increasing importance in modern deep space telecommunication, this section is devoted to the study of these codes, employing finite fields and fast Fourier transforms.

12 Application of Fermat Theoretic Transform to the Decoding of Reed-Solomon Code

12.1 Reed-Solomon Code

We shall summarize some of the basic ideas pertaining to the Reed-Solomon code (Ref. 9).

(1) The RS code is a block code (as opposed to being a convolutional code).

(2) An RS codeword will consist of I information or message symbols, together with P parity or check symbols. The word length is $N = I + P$.

(3) The symbols in an RS codeword are usually not binary, i.e., each symbol is represented by more than one bit. In fact, a favorite choice is to use 8-bit symbols. This is related to the fact that most computers have word length of 8 bits or multiples of 8 bits.

(4) A multi-bit symbol is the information unit in an RS code. Each symbol may be corrupted at a single bit-position or by a burst of bit-errors affecting many bit positions. In the latter case if the corrupted symbol is corrected, the RS code is seen to be correcting a burst of bit-errors. This suggests that the RS code has the "built-in potential" of correcting burst errors.

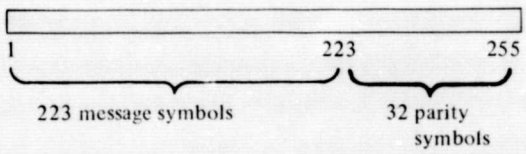
(5) In order to be able to correct "t" symbol errors, the minimum distance of the codewords "D" is given by $D = 2t + 1$. For multi-bit symbol sequences, the "distance" between two symbol sequences equals the number of symbol positions at which the two sequences differ for example, the distance between the sequences 2,4,3,4 and 2,0,3,4 is one. Note that each symbol is denoted by a decimal representation.

If the minimum distance of an RS code is D, and the word length is N, then, the number of message symbols I in a word is given by

$$I = N - (D - 1)$$

Combining with the formula given in paragraph (2), above, $P = D - 1$.

An example of the structure of a code word in a practical RS code is as follows:



Each symbol consists of 8 bits. Thus, each codeword has 255 symbols, or $255 \cdot 8$ bits, consisting of $223 \cdot 8$ message bits and $32 \cdot 8$ check bits. This code is capable of correcting 16 symbol errors.

12.2 Relationships Between RS Decoding and Finite Field $GF(F_n)$

It will be seen later that one of the steps in the decoding of the RS code is "the computation of the syndromes," and this step is identical to obtaining the DFT of a sequence defined in Section 8.

Also, this step is very time-consuming, and RS decoding will be greatly accelerated if the discrete Fourier transform techniques applied to obtain the DFT of a sequence (as described in Sections 8.1 to 8.4) is applied to "the computation of the syndromes." Section 8.2 concludes that in order to apply discrete Fourier transform techniques, the elements of the sequence must belong to $GF(F_n)$ when $F_n = 2^{2^n} + 1$ such that

- There exists a generator element $\gamma = 2$.
- The order of the element d (given by $\gamma^d = 1$) is a power of 2.

It follows from the above discussion that certain "parameters" of the RS code must be related to the "parameters" of $GF(F_n)$ if the DFT techniques are to be applicable. The required relationships are summarized as follows:

(1) The symbols used in the RS code must be elements of $GF(F_n)$, where $F_n = 2^{2^n} + 1$.

(2) It is convenient to choose the codeword length N to be equal to the order of the element $\gamma = 2$, i.e., choose $N = d = 2^{n+1}$. This is because $GF(2^{2^n} + 1)$ will always have a generator element γ equal to 2 (see Section 8.2). Also, since $2^{2^n} + 1 = F_n$,

$$2^{2^n} \equiv -1 \pmod{F_n}$$

$$(2^{2^n})^2 \equiv 1 \pmod{F_n}$$

$$2^{2^{n+1}} \equiv 1 \pmod{F_n}$$

and, finally,

$$d = 2^{n+1}$$

Note that in DFT terminology, d is also the transform length (see Section 7.3).

(3) The number of bits needed to represent a symbol is determined by the number of different symbols used, which, by (1), is equal to $2^{2^n} + 1$. Therefore, for $F_n = 2^{2^n} + 1$, the number of bits/symbol is $2^n + 1$ (see also Section 10).

12.3 To Construct an RS Code of Wordlength Equal to 8 Symbols, and Capable of Correcting 2 Symbol Errors

It follows from 12.2(2) that $N = d = 8 = 2^{n+1}$. Hence, we shall choose $n = 2$, i.e, $GF(F_2) = GF(2^{2^2} + 1) = GF(17)$. $GF(17)$ will have a generator $\gamma = 2$ such that $\gamma^d = 2^8 \equiv 1$ modulo 17.

It follows from 12.2(1) that the symbols in the RS code will be the elements of $GF(17)$. Also, the number of bits/symbol will be, from 12.2(3), $2^n + 1$ or 5.

In order to correct two errors, the minimum distance of the code word is $D = 2t + 1 = 2 \cdot 2 + 1 = 5$. The number of message symbols/codeword is $I = N - (D - 1) = 8 - (5 - 1) = 4$. The number of check symbols is $P = N - I = 8 - 4 = 4$.

To construct an RS code with minimum distance D, we first define a generator polynomial as follows:

$$g(Z) = \prod_{i=1}^{D-1} (Z - 2^i) = \prod_{i=1}^{5-1} (Z - 2^i)$$

$$= (Z - 2)(Z - 2^2)(Z - 2^3)(Z - 2^4)$$

$$\equiv (Z - 2)(Z - 4)(Z - 8)(Z + 1)$$

$$\equiv Z^4 - 13Z^3 + 8Z^2 - 8Z - 64$$

$$\equiv Z^4 + 4Z^3 + 8Z^2 - 8Z + 4 \pmod{17}$$

Assume the message symbols to be $1,2,3,2, \in GF(17)$. Let us form $f(Z) = Z^7 + 2Z^6 + 3Z^5 + 2Z^4$ of degree $N - 1 = 7$, using the message symbols as coefficients. In order to generate a "code word in a polynomial $C(Z)$," which is a multiple of $g(Z)$, we proceed as follows:

$$f(Z) = q(Z)\,g(Z) + R(Z)$$

where

$q(Z)$ = quotient polynomial

$g(Z)$ = generator polynomial

$R(Z)$ = residue polynomial

$$C(Z) = q(Z)\,g(Z) = f(Z) - R(Z)$$

$R(Z)$ is obtained by long division of the form:

$$\begin{array}{r} q(Z) \\ g(Z)\ \overline{)\ f(Z)} \\ \vdots \\ \hline R(Z) \end{array}$$

$$C(Z) = q(Z)\,g(Z)$$

Thus,

$$\begin{array}{r} Z^3 - 2Z^2 + 3Z - 3 \\ Z^4 + 4Z^3 + 8Z^2 - 8Z + 4\ \overline{)\ Z^7 + 2Z^6 + 3Z^5 + 2Z^4} \\ Z^7 + 4Z^6 + 8Z^5 - 8Z^4 + 4Z^3 \\ \hline - 2Z^6 - 5Z^5 + 10Z^4 - 4Z^3 \\ - 2Z^6 - 8Z^5 - 16Z^4 + 16Z^3 - 8Z^2 \\ \hline 3Z^5 + 9Z^4 - 3Z^3 + 8Z^2 \\ 3Z^5 + 12Z^4 + 24Z^3 - 24Z^2 + 12Z \\ \hline - 3Z^4 - 10Z^3 - 2Z^2 - 12Z \\ - 3Z^4 - 12Z^3 - 24Z^2 + 24Z - 12 \\ \hline R(Z) = + 2Z^3 + 5Z^2 - 2Z + 12 \end{array}$$

The encoded codeword is

$$C(Z) = q(Z) \cdot g(Z) = f(Z) - R(Z)$$

$$= Z^7 + 2Z^6 + 3Z^5 + 2Z^4 - 2Z^3 - 5Z^2 + 2Z + 5$$

The codewords have the properties

$$C(\gamma^i) = q(\gamma^i)\,g(\gamma^i)$$

$$C(2^i) = q(2^i)\,g(2^i) = q(2^i)\,0 = 0, \text{ for } i = 1,2,3,4$$

This results from the structure of the generating polynomial that

$$g(Z) = (Z - 2^1)(Z - 2^2)(Z - 2^3)(Z - 2^4)$$

Thus, it can be shown that

$$C(2^1) = 0$$

$$C(2^2) = 0$$

$$C(2^3) = 0$$

$$C(2^4) = 0$$

$$C(2^5) = -16 \equiv 1 \pmod{17}, \text{ etc.}$$

It can be shown also that if there are errors *in the received codewords:*

$$r(2^1) \neq 0$$

$$r(2^2) \neq 0$$

$$r(2^3) \neq 0$$

$$r(2^4) \neq 0$$

Suppose 2 errors exist in the received codeword at the positions underlined below:

$$r(Z) = 5Z^0 + 2Z^1 + \underline{9Z^2} + 15Z^3 + 2Z^4 + \underline{1Z^5} + 2Z^6 + Z^7$$

or, written differently,

$$(r_0, r_1, r_2, \cdots, r_7) = (5,2,9,15,2,1,2,1)$$

$$= (5,2,12-3,15,2,3-2,2,1)$$

The error pattern is

$$(0,0,-3,0,0,-2,0,0)$$

or

$$(0,0,14,0,0,15,0,0)$$

The received pattern can be rewritten as

$$(C_0, C_1, C_2, \cdots, C_7) + (e_0, e_1, \cdots e_7)$$

$$(5,2,12,15,2,3,2,1) + (0,0,14,0,0,15,0,0)$$

where $(e_0, e_1, \cdots, e_7)$ is an error pattern, and $(C_0, C_1, \cdots, C_7)$ are the uncorrupted symbols.

Now the syndromes S_k for $r(Z) = (r_0, r_1, \cdots, r_7)$, where $r(Z) = 5Z^0 + 2Z^1 + 9Z^2 + 15Z^3 + 2Z^4 + Z^5 + 2Z^6 + Z^7$, can be computed by defining

$$S_k = \sum_{i=0}^{N-1} r_i \gamma^{ki} = \sum_{i=0}^{8-1} r_i 2^{ki} \qquad (13)$$

for $k = 1,2, \cdots, D-1 = 2t$ (i.e., $k = 1,2,3,4$) and $\gamma = 2$.

Since $r(Z) = (r_0, r_1, \cdots, r_7)$, the received symbols are known; the syndromes S_1, S_2, S_3, and S_4 can be calculated from Eq. (13). Specifically, for $r(2) = (5,2,9,15,2,1,2,1)$, Eq. (13) yields (S_1, S_2, S_3, S_4) to be $(-8, -5, 11, -1)$.

Actually, the way Eq. (13) was defined implicitly spells out the relationship between the syndromes and the symbol error pattern, for from Eq. (13),

$$S_k = \sum_{i=0}^{8-1} (C_i + e_i) 2^{ki} \qquad (14)$$

$$= \sum_{i=0}^{8-1} C_i 2^{ki} + \sum_{i=0}^{8-1} e_i 2^{ki}$$

where $e(Z) = (e_0, e_1, \cdots, e_7)$ is the symbol error pattern. But $C(Z)$ is a multiple of $q(Z)$, and, consequently,

$$C(\gamma^k) = C(2^k) = 0, \text{ for } k = 1,2,3,4$$

so that

$$\sum_{i=0}^{8-1} C_i (2^k)^i = 0, \quad \text{for } k = 1,2,3,4$$

Thus, Eq. (14) becomes

$$S_k = \sum_{i=0}^{8-1} e_i (2^k)^i = E_k, \text{ for } k = 1,2,3,4 \qquad (15)$$

Equation (15) reveals that the syndrome S_k is in fact the DFT of the error pattern, i.e., $S_k = E_k$ = DFT of e_j.

The problem in decoding the RS code is to try to determine the values of e_i, $i = 0,1,2, \cdots, 7$. Since at present e_i are not known, we let Y_i and X_i be the ith error amplitude and the ith error location, respectively. Thus, the syndrome in Eq. (15) can be re-expressed as

$$S_k = \sum_{i=1}^{8-1} Y_i X_i^k, \text{ for } k = 1,2,3,4$$

However, as we see above, $(e_0, e_1, \cdots, e_7)$ are all zero except in the location $i_1, i_2, \cdots, i_t$, where t is the maximum number of symbol errors that can be corrected. This is to say that

$$S_k = \sum_{i=1}^{t} Y_i X_i^k, \text{ for } k = 1,2,3,4$$

$$= \sum_{i=1}^{t=2} Y_i X_i^k = Y_1 X_1^k + Y_2 X_2^k$$

(in our example)

Hence,

$$S_k = e_0 (2^k)^0 + e_1 (2^k)^1 + \cdots + e_7 (2^k)^7,$$
$$\text{for } k = 1,2,3,4$$

$$= 0(2^k)^0 + 0(2^k)^1 + 14(2^k)^2 + \cdots + 15(2^k)^5 + \cdots + 0(2^k)^7,$$

$$(e_0, e_1, \cdots, e_7) = (0,0,14,0,0,15,0,0)$$

or

$$S_k = 14(2^k)^2 + 15(2^k)^5, \text{ for } k = 1,2,3,4$$

Since from the above S_1, S_2, S_3, S_4 are -8, -5, 11, -1, we have

$$Y_1 X_1^{\ 1} + Y_2 X_2^{\ 1} = S_1 = -8$$

$$Y_1 X_1^{\ 2} + Y_2 X_2^{\ 2} = S_2 = -5$$

$$Y_1 X_1^{\ 3} + Y_2 X_2^{\ 3} = S_3 = 11$$

$$Y_1 X_1^{\ 4} + Y_2 X_2^{\ 4} = S_4 = -1$$

However, rather than solving the four nonlinear equations directly, it will be simpler to obtain the transform of the error pattern. After simple calculations, take the inverse discrete Fourier transform of the result to achieve the error pattern. This method is now described.

From the previous discussion, since $S_k = E_k$ for $k = 1,2,3,4$, some of the transforms of the error pattern $e(Z)$ are known at this stage. The rest of the transforms, i.e., E_0, E_5, E_6, E_7, can be computed from those already known, i.e., E_1, E_2, E_3, E_4. To do this, let us define a generating function as

$$E(x) = E_1 x^{-1} + E_2 x^{-2} + E_3 x^{-3} + \cdots = \sum_{k=1}^{\infty} E_k x^{-k} \tag{16}$$

in which it is noted that $E_8 = E_0$, $E_9 = E_1$, etc. Since

$$S_k = E_k = \sum_{n=0}^{8-1} e_n 2^{nk} = \sum_{i=1}^{2} Y_i X_i^k \tag{17}$$

substituting Eq. (16) into Eq. (17) gives

$$E(x) = \sum_{k=1}^{\infty} \left(\sum_{i=1}^{2} Y_i X_i^k \right) x^{-k}$$

$$= \sum_{i=1}^{2} Y_i \sum_{k=1}^{\infty} (X_i x^{-1})^k$$

$$= \sum_{i=1}^{2} Y_i \frac{X_i x^{-1}}{1 - X_i x^{-1}}$$

the last step being obtained by the usual technique of summing a geometrical series. Thus,

$$E(x) = \sum_{i=1}^{2} Y_i \frac{X_i}{x - X_i} = \frac{P(x)}{\sigma(x)}$$

where

$$\sigma(x) = \prod_{i=1}^{2} (x - X_i)$$

$$= x^2 - (X_1 + X_2) x + X_1 X_2$$

$$= x^2 - \sigma_1 x + \sigma_2$$

and $\sigma(x)$ is called the "error location polynomial" since its roots help to locate the errors:

$$E(x) = -8x^{-1} - 5x^{-2} + 11x^{-3} - x^{-4} + ?x^{-5} + ?x^{6} \tag{18}$$

It can be shown that $\sigma(x)$ can be obtained from Eq. (18) by the "continued fraction method" developed by Prof. L. R. Welch and R. Scholtz of USC (Ref. 10) as an alternative method to the Berlekamp algorithm, which solves the same problem. The "continued fraction method" is illustrated in Table 1.

From Table 1, one observes that

$$R_3 = 0 + ?x^{-3} + \cdots$$

$$\sigma(x) = \sigma_3(x) = (x - x_1)(x - x_2)$$

$$= (2x + 3)(6x - 4) = x^2 - 2x + 9$$

Hence,

$$\sigma_1 = 2, \quad \sigma_2 = 9$$

Table 1. The continued fraction method

S	$R_{S-2}(x) = q_S(x) R_{S-1}(x) + R_S(x)$	$q_S(x)$	$R_S(x)$	$\sigma_S(x) = q_S(x)\sigma_{S-1}(x) + \sigma_{S-2}(x)$
-1				1
0				0
1	$1 \overline{)\, -8x^{-1} - 5x^{-2} + 11x^{-3} - x^{-4} + Xx^{-5} + \cdots}$ $0x^{-1} + 0x^{-2} + 0x^{-3} + 0x^{-4} + \cdots$ $-8x^{-1} - 5x^{-2} + 11x^{-3} - x^{-4} + Xx^{-5} + \cdots$	0	$-8x^{-1} - 5x^{-2} + 11x^{-3} - x^{-4} + Xx^{-5} + \cdots$	$\sigma_1(x) = 0 \cdot 0 + 1 = 1$
2	$-8x^{-1} - 5x^{-2} + 11x^{-3} - x^{-4} + Xx^{-5} + \cdots$ $\overline{)\, 1 - 10x^{-1} + 5x^{-2} - 2x^{-3} + Xx^{-4} + \cdots}$ $10x^{-1} - 5x^{-2} + 2x^{-3} + Xx^{-4} + \cdots$ $10x^{-1} - 15x^{-2} - x^{-3} + \cdots$ $10x^{-2} + 3x^{-3} + Xx^{-4} + \cdots$	$2x + 3$	$10x^{-2} + 3x^{-3} + Xx^{-4} + \cdots$	$\sigma_2(x) = (2x + 3) \cdot 1 + 0$ $= 2x + 3$
3	$-7x^{-2} + 3x^{-3} + Xx^{-4} + \cdots$ $\overline{)\, -8x^{-1} - 5x^{-2} + 11x^{-3} - x^{-4} + Xx^{-5} + \cdots}$ $-8x^{-1} + x^{-2} + Xx^{-3} + \cdots$ $-6x^{-2} + Xx^{-3} + \cdots$ $-6x^{-2}$ $0x^{-2} + Xx^{-3}$	$6x - 4$	$0 + Xx^{-3} + \cdots$	$\sigma_3 = (6x - 4)(2x + 3) + 1$ $= x^2 - 2x + 9$

and

$$\sigma(X_i) = X_i^2 - 2X_i + 9 = 0, \text{ for } i = 1,2$$

Multiplying the above equation by $Y_i X_i^j$ gives

$$Y_i X_i^{2+j} - 2Y_i X_i^{1+j} + 9Y_i X_i^j = 0, \text{ for } i = 1,2 \qquad (19)$$

Substituting

$$E_k = \sum_{i=1}^{2} Y_i X_i^k$$

into Eq. (19) yields

$$E_{2+j} - 2E_{1+j} + 9E_j = 0, \quad \text{for } j = 1,2,3$$

Using this recursive formula, E_0, E_5, E_6, E_7 can be computed from E_1, E_2, E_3, E_4. Thus (see Eq. 15),

$$E_5 = 2E_4 - 9E_3 = 2(-1) - 9(11) \equiv 1 \pmod{17}$$

Similarly,

$$E_6 \equiv 11 \pmod{17}$$

$$E_7 \equiv 13 \pmod{17}$$

$$E_8 \equiv 12 \pmod{17} = E_0$$

Thus,

$$E(Z) = (12, -8, -5, 11, -1, 1, 11, 13)$$

Since

$$E_k = \sum_{n=0}^{8-1} e_n 2^{nk}, \text{ for } k = 0,1,2,\cdots,7$$

the inverse DFT of E_k is defined by

$$e_n = (8)^{-1} \sum_{k=0}^{8-1} E_k 2^{-nk}, \text{ for } n = 0,1,2,\cdots,7$$

$$= (-2)(E_0 2^0 + E_1 2^{-n} + \cdots + E_7 2^{-7n})$$

Since $E_0, \cdots, E_7$ are now known, $e_0, \cdots, e_7$ can be solved.

It can be shown, as expected, that for this example

$$(e_0, e_1, \cdots, e_7) = (0,0,14,0,0,15,0,0)$$

Since the received codeword is (5,2,9,15,2,1,2,1) and the error pattern is (0,0,14,0,0,15,0,0), the corrected coded sequence is (5,2,9,15,2,1,2,1) - (0,0,14,0,0,15,0,0) = (5,2,12,15,2,3,2,1).

To recapitulate, the decoding of Reed-Solomon codes using the transform over $GF(F_n)$ is composed of the following three steps (Ref. 11):

(1) Compute the DFT over $GF(F_n)$ of the received code N-tuple; i.e.,

$$S_k = \sum_{m=0}^{N-1} r_m \gamma^{mk}$$

where $r_m \in GF(F_n)$, and γ is an element of order N.

(2) Use continued fractions to determine σ_i from the known $S_j = E_j$ for $i = 1,2, \cdots, t$ and $j = 1,2, \cdots, 2t$. Then compute the remaining transform errors E_j.

(3) Compute the inverse of the transform over $GF(F_n)$ of $S_k - E_k$ to obtain the corrected code.

References

1. Booth, T. L., *Sequential Machines and Automata Theory*, John Wiley & Sons, Inc., New York, 1967.

2. Berlekamp, E. R., *Algebraic Coding Theory*, McGraw-Hill Book Co., Inc., New York, 1968.

3. Niven, I., and Zuckerman, H. S., *An Introduction to the Theory of Numbers*, John Wiley & Sons, Inc., New York, 1972.

4. Rader, C. M., "Discrete Convolution via Mersenne Transforms," *IEEE Trans. Computers*, Vol. C-21, No. 12, Dec. 1972.

5. Agarwal, R. C., and Burrus, C. S., "Number Theoretic Transforms to Implement Fast Digital Convolution," *Proc. IEEE*, Vol. 63, No. 4, Apr. 1975.

6. Brigham, E. O., *The Fast Fourier Transform*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1974.

7. Reed, I. S., and Truong, T. K., "The Use of Finite Field to Compute Convolutions," *IEEE Trans. Inform. Theory*, Vol IT-21, pp. 208-213, Mar. 1975.

8. Reed, I. S., and Truong, T. K., "Complex Integer Convolutions over a Direct Sum of Galois Fields," *IEEE Trans. Inform. Theory*, Vol. IT-21, Nov. 1975.

9. Reed, I. S., and Solomon, G., "Polynomial Codes over Certain Finite Fields," *SIAM J. Appl. Math.*, Vol. 8, pp. 300-304, June 1960.

10. Welch, L. R., Reed, I. S., and Truong, T. K., "The Fast Decoding of Reed-Solomon Codes Using Fermat Theoretic Transforms and Continued Fractions," in *The Deep Space Network Progress Report 42-36*, pp. 63-74, Jet Propulsion Laboratory, Pasadena, Calif., Dec. 15, 1976.

11. Reed, I. S., Welch, L. R., and Truong, T. K., "The Fast Decoding of Reed-Solomon Codes Using Number Theoretic Transforms," in *The Deep Space Network Progress Report 42-35*, pp. 64-78, Jet Propulsion Laboratory, Pasadena, Calif. Oct. 15, 1976.

12. Reed, I. S., "The Use of Finite Fields and Rings to Compute Convolutions," Laboratory Technical Memorandum No. 246-0016, Lincoln Laboratory, MIT, Oct. 1973.