T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

MATRİS KODLAR İLE McELIECE ŞİFRELEME SİSTEMİ

YÜKSEK LİSANS TEZİ

Vedat ŞİAP

Enstitü Anabilim Dalı : MATEMATİK

Tez Danışmanı : Doç. Dr. Refik KESKİN

Haziran 2008

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

MATRİS KODLAR İLE McELIECE ŞİFRELEME SİSTEMİ

YÜKSEK LİSANS TEZİ

Vedat ŞİAP

Enstitü Anabilim Dalı : MATEMATİK

Bu tez 09/06/2008 tarihinde aşağıdaki jüri tarafından oybirliği ile kabul edilmiştir.

Doç. Dr.Refik KESKİN	Doç. Dr.Mehmet ÖZEN	Doç. Dr.Elman ALİYEV
Jüri Başkanı	Üye	Üye

TEŞEKKÜR

Eğitim hayatım süresince beni özveri ile yetiştiren tüm öğretmen ve öğretim üyeleri hocalarıma teşekkürü bir borç bilirim. Bilhassa, tez çalışmamın her aşamasında bilgi ve tecrübeleriyle beni yönlendiren. yardımlarını ve yakın ilgisini esirgemeyen danışmanım Sayın Doç. Dr. Refik KESKİN'e en içten teşekkürlerimi sunarım.

Ayrıca eğitim hayatım boyunca maddi ve manevi desteklerini üzerimden esirgemeyen aileme de teşekkür ediyorum.

Haziran 2008

Vedat ŞİAP

İÇİNDEKİLER

SİMGELER VE KISALTMALAR LİSTESİ

$\mathbb{Z}$: Tamsayılar kümesi

$\mathbb{Z}_m$: Kalan sınıflar kümesi

φ : Euler fonksiyonu

e_k : Şifreleme fonksiyonu

d_k : Şifre çözücü fonksiyon

A^n : n elemanlı sonlu A alfabesi

C : Kod

E : Kodlama fonksiyonu

D : Dekodlama fonksiyonu

d : Minimum uzaklık fonksiyonu

w : Ağırlık fonksiyonu

F_q : q elemanlı sonlu cisim

$V(n,q)$: elemanları F_q dan alınan n - lilerin kümesi

$[n,k]$: n uzunluğunda, k boyutlu bir lineer kod

$[n,k,d]$: n uzunluğunda, k boyutlu, d minimum uzaklığına sahip lineer kod

G : Üreteç matris

H : Kontrol matris

$Mat_{m\times n}$: $m\times n$ tipindeki matrisler kümesi

$A\otimes B$: A ile B matrisinin kronecker çarpımı

ŞEKİLLER LİSTESİ

ÖZET

Anahtar kelimeler: Lineer kodlar, matris kodlar, McEliece şifreleme sistemi

Bilgi çağında yaşadığımız bu günlerde bilginin transferi (internet, cep telefonları, bankacılık vs.) ya da depolanması (CD vs.) aşamasında meydana gelebilecek bilgi zedelenmelerini koruma ve düzeltme amacıyla kodlama kullanılmaktadır. Bu anlamda kullanılan kodlar içinde lineer kodlar önemli bir yer tutmaktadır. Lineer kodlar ailesinin içinden olan matris kodlar zengin bir yapıya sahiptir. Ayrıca, matris kodlar ile hata düzeltme kabiliyetleri artmakta ve bunun sonucunda bilgi daha güvenilir iletilmektedir.

Bu tez altı bölümden oluşmaktadır. Birinci bölümde, tanımlar ve teoremler verilmiştir.

İkinci bölümde, şifreleme ve şifreleme sistemlerinin işleyişi ele alınmıştır.

Üçüncü bölümde, kodlama ile ilgili temel tanım ve teoremler verilmiştir. Ayrıca lineer kodların cebirsel yapıları ve dekodlaması ile ilgili tanım ve teoremler verilmiştir.

Dördüncü bölümde, sonlu cisim üzerinde tanımlanan matris kodlar ile ilgili tanımlar ve matris kodların işleyişi ele alınmıştır.

Beşinci bölümde, McEliece şifreleme sistemi incelenmiş ve matris kodlar, McEliece şifreleme sistemine uygulanmıştır.

Altıncı ve son bölüm, sonuç ve öneriler kısmından oluşmuştur.

THE McELIECE CRYPTOSYSTEM WITH ARRAY CODES

SUMMARY

Key Words: Linear codes, array codes, McEliece cryptosystem

As we live in the information age, coding is used in order to protect or correct the messages in the transferring (via internet, mobile phones, banking, etc.) or the storing (CD,etc.) processes. So, linear codes are important in the transferring or the storing. Due to richness of their structure array codes which are linear are also an important codes. However, the information is then transferred into the source more securely by increasing the error correction capability with array codes.

This thesis consists of six chapters. In the first chapter, some basic definitions of abstract algebra are given.

In the second chapter, cryptology and the process of some classical cryptosystems are discussed.

In the third chapter, some basic definitions and theorems associated with coding theory are given. However, some basic definitions and theorems associated with the algebraic structure and decoding of linear codes are given.

In fourth chapter, the definitions of array codes over finite field are discussed. Moreover, the process of array codes is given.

In the fifth chapter, the McEliece cryptosystem with array codes is given and their applications to the array codes are investigated.

In the sixth and the last chapter, the conclusion and the future works are given.

BÖLÜM 1. GİRİŞ

Bölüm 1 deki tanım ve teoremler için daha geniş bilgi [4] ve [30] nolu kaynaklarda, bölüm 2 ve 3 teki şifreleme ve kodlama teorisi için daha geniş bilgi [3] ve [5] nolu kaynaklarda bulunabilir.

1.1.Tanımlar ve Teoremler

Tanım 1.1.1. [4] m pozitif bir tamsayı olmak üzere eğer $m|a-b$ ise a sayısı b sayısına m modülüne göre kongrudur (denktir) denir ve bu durum $a \equiv b \pmod{m}$ ile gösterilir.

Tanım 1.1.2. [4] $\mathbb{Z}$ deki $\equiv$ denklik bağıntısının belirttiği denklik sınıflarına, m modülüne göre $(\bmod m)$ kalan sınıfları denir ve tüm kalan sınıfları kümesi $\mathbb{Z}_m$ ile gösterilir. $a \in \mathbb{Z}$ nın denklik sınıfı, $\overline{a} = \{x \in \mathbb{Z} \mid m|a-x\}$ dir.

Tanım 1.1.3. [4] $a \in \mathbb{Z}_m$ olsun. $ac = 1$ olacak şekilde bir $c \in \mathbb{Z}_m$ varsa c ye a nın tersi denir ve kısaca $c = a^{-1}$ ile gösterilir.

Tanım 1.1.4. [4] $ax \equiv b \pmod{m}$ şeklindeki bir denkleme bir bilinmeyenli lineer kongrüans denir. Bu denklemi sağlayan x tamsayılarının kümesine de kongrüansın çözüm kümesi denir.

Önerme 1.1.1. [4] $ax \equiv b \pmod{m}$ nin bir çözümü $x_0 \in \mathbb{Z}$ ise $\overline{x_0} \in \mathbb{Z}_m$ sınıfındaki tüm sayılar da bir çözümdür.

Önerme 1.1.2. [4] $(a,m)=1$ ise $ax\equiv b(\bmod m)$ nin çözümü var ve bu $\bmod m$ ye göre tek bir sınıftır.

Örnek 1.1.1. $(2,5)=1$ olduğundan $2x\equiv 1(\bmod 5)$ nin çözümü vardır ve lineer kongrüans denkleminin çözüm kümesi $Ç=\{3,8,13,...\}$ dir.

Önerme 1.1.3. [4] $ax\equiv b(\bmod m)$ nin bir çözümünün olması için gerek ve yeter şart $(a,m)\mid b$ olmasıdır.

Örnek1.1.2. $(6,15)=3\mid 3$ olduğundan $6x\equiv 5(\bmod 15)$ kongrüansının çözümü vardır.

Örnek 1.1.3. $2x\equiv 3\ (\bmod 4)$ denkleminin çözümü yoktur. Çünkü $(2,4)=2$ dir fakat $2\mid 3$ değildir.

Önerme 1.1.4. [4] $\forall b\in\mathbb{Z}_m$ için $ax\equiv b$ (mod m) denkleminin tek türlü bir $x\in\mathbb{Z}_m$ çözümünün olması için gerek ve yeter şart $(a,m)=1$ olmasıdır.

Tanım 1.1.5. [4] $m\geq 1$ olmak üzere $0\leq x\leq m-1$ ve $(x,m)=1$ şartını sağlayan x tamsayılarının sayısı $\varphi(m)$ ile gösterilir. Kısaca,

$$\varphi(m)=\left|\{x\mid 0\leq x\leq m-1,\ (x,m)=1\}\right|$$

şeklinde tanımlı $\varphi(m)$ fonksiyonuna Euler fonksiyonu denir.

Önerme 1.1.5. [4] $m\geq 1$, $n\geq 1$ ve $(m,n)=1$ ise $\varphi(mn)=\varphi(m)\varphi(n)$ dir.

Önerme 1.1.6. [4] $\varphi(1)=1$ ve p asal olmak üzere $\varphi(p)=p-1$, $\varphi(p^{\alpha})=p^{\alpha}-p^{\alpha-1}$ $(\alpha\geq 1)$ ve $\varphi(n)=n\prod_{p\mid n}(1-\frac{1}{p})$ dir.

Teorem 1.1.1. (Euler Teoremi) [4] $(a,m)=1$ ise $a^{\varphi(m)} \equiv 1 \pmod{m}$ dir.

Örnek 1.1.4. $(3,8)=1$ olduğundan $3^{\varphi(8)} \equiv 1 \pmod{8}$ dir. Gerçekten

$$\varphi(8)=\varphi(2^3)=2^3-2^2=4 \text{ ve } 3^{\varphi(8)} \equiv 3^4 \equiv 81 \equiv 1 \pmod{8}$$

dir.

Teorem 1.1.2. (Fermat Teoremi) [4] p asal bir sayı ve $(a,p)=1$ ise $a^{p-1} \equiv 1 \pmod{p}$ dir.

Tanım 1.1.6. [5] $A=(a_{ij})$ matrisi 2×2 tipinde bir matris olsun. $A=(a_{ij})$ matrisinin determinantının değeri

$$\det A = a_{11}a_{22}-a_{12}a_{21}$$

dir.

Teorem 1.1.3. [5] $\det A \neq 0$ olacak şekilde $A=(a_{ij})$ matrisi 2×2 tipinde bir matris olsun. Bu taktirde, A matrisinin tersi

$$A^{-1}=(\det A)^{-1}\begin{pmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{pmatrix}$$

dir.

Tanım 1.1.6. [30] $A=(a_{ij})$ matrisi $m\times n$ tipinde bir matris olsun. $A=(a_{ij})$ matrisinin transpozesi A^T ile gösterilir ve A^T matrisi $n\times m$ tipinde bir matristir.

Yani,

$$\begin{pmatrix} a_{11} & a_{12} & . & . & . & a_{1n} \\ a_{21} & a_{22} & . & . & . & a_{2n} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{m1} & a_{m2} & . & . & . & a_{mn} \end{pmatrix}^{T} = \begin{pmatrix} a_{11} & a_{21} & . & . & . & a_{m1} \\ a_{12} & a_{22} & . & . & . & a_{m2} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{1n} & a_{2n} & . & . & . & a_{mn} \end{pmatrix}$$

dir.

Teorem 1.1.4. [30] A $n \times n$ tipinde bir matris ve tersi A^{-1} olsun. Bu taktirde,

$$A^{-1} = \frac{1}{\det A} adjA$$

dir. Burada $adjA$, A matrisinin adjoint matrisidir.

Tanım 1.1.7. [30] P matrisinin her sütununda ya da satırında bir tane 1 ve geri kalan değerleri 0 olan kare matrise permütasyon matris denir.

Örnek 1.1.5. $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ ve $\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ permütasyon matrislerdir.

Tanım 1.1.8. [30] P permütasyon matris ise P permütasyon matrisinin transpozesi P permütasyon matrisinin tersine eşittir. Yani, $P^T = P^{-1}$ dir.

Tanım 1.1.9. [4] $G \neq \varnothing$ kümesi üzerinde tanımlı bir ikili işlem $(*)$ olsun. $(G,*)$ cebirsel yapısı G1 – G4 aksiyomlarını sağlıyorsa bu cebirsel yapıya grup denir.

G1: $*$, G de bir ikili işlemdir.

G2: $*$ işleminin G de birleşme özelliği vardır. Yani, $\forall a,b \in G$ için $a*(b*c)=(a*b)*c$ dir.

G3: $*$ işleminin G de birim elemanı vardır. Yani, $\forall a \in G$ için $a*e=e*a=a$ olacak şekilde bir $a \in G$ vardır.

G4: $*$ işlemine göre, G deki her elemanın tersi vardır. Yani, $a \in G$ için, $a*a^{-1}=a^{-1}*a=e$ olacak şekilde bir $a^{-1} \in G$ vardır.

Tanım 1.1.10. [4] $R \neq \varnothing$ kümesi üzerinde tanımlı ikili işlem ve $+$ ve $\bullet$ olsun. Aşağıdaki aksiyomları sağlayan $(R, +, \bullet)$ cebirsel yapısına bir halka denir.

H1: $(R, +)$ bir değişmeli gruptur.

H2: $\bullet$ işleminin R de birleşme özelliği vardır.

H3: $\bullet$ işleminin $+$ işlemi üzerinde sağdan ve soldan dağılma özelliği vardır. Yani, $\forall a,b,c \in R$ için $a(b+c)=ab+ac$ ve $(a+b)c=ac+bc$ dir.

Tanım 1.1.11.[4] R birimli ve değişmeli bir halka ve $R-\{0\}=R^*$, ikinci işlem $\bullet$ ye göre bir grup ise R ye bir cisim denir. Sonlu sayıda elemana sahip olan cisme sonlu cisim veya Galois cisim denir.

BÖLÜM 2. ŞİFRELEME

2.1. Şifreleme ve Tarihi

Günümüzde birçok iş alanında giderek artan bilgisayar kullanımı sonucu bilgiler bilgisayarlar aracılığı ile işlenmekte ve işlenen bu bilgiler haberleşme kanalları aracılığı ile iletilmektedir. İletilecek bu bilgiler yetkili olmayan kişiler tarafından kopyalanabilir veya değiştirilebilir. Bu noktada şifreleme (kriptoloji), bilgisayarlar tarafından işlenen bu bilgilerin çeşitli haberleşme kanalları ile iletilmesinde veya korunmasında en ekonomik yoldur.

Şifrelemede temel problem açık mesajları şifrelenmiş mesajlara bir algoritma doğrultusunda dönüştürmeden ibarettir. Bu algoritmanın deşifrelemesinin, zor ve daha güvenilir olması için çok yoğun şifreleme analizine (kriptoanaliz) dayanmalıdır.

Şifreleme (Kriptoloji), Yunanca “cruptos (gizli)” ve “logos (bilim)” anlamına gelen kelimelerin birleşmesi ile oluşur ve dolayısıyla şifreleme ismi günümüzde tam anlamıyla bilginin saklanması ve gizli haberleşmeyi tanımlayan temel kelime olarak kullanılmaktadır.

Bilginin saklanması ve iletilmesi 4000 yıl öncesine dayanır. Eski Mısırlılar şifrelemeyi ilk kullanan kişiler olarak bilinmektedir. Kullandıkları alfabeleri ve yerleştirilen sembollerin yeri okuyanın kafasını karıştırmak ve metnin içerisindeki bilgiyi saklamak için yer değiştirildiği ve ancak diziyi bilen kişinin sadece metni anlayabildiği ve sembollerin şifresini çözebildiği bilinmektedir.

Benzer bir metod Julius Ceaser tarafından kullanılmıştır. Öteleme şifrelemesini kullanarak komutanlarına kendi emirlerini göndermek amacıyla çapı değişken bir

silindirin üzerine sarılmış ince kağıda silindir ekseni doğrultusunda yazı yazarak kullandığı ve bu mesajın ancak aynı çapta bir silindir kullanılarak okunabildiği bilinmektedir.

Şifrelemenin gerçek anlamda kullanımı 18. yüzyılın sonunda başlamaktadır. Bir İngiliz bilimadamının, birbiri ile irtibatlı 36 silindirli bir çubuğun etrafında dönen 29 harfli bir alfabeyi şifreleyebilen dönen tekerleğe benzeyen bir makine yaptığı ve bu makinenin Jefferson Cylinder olarak isimlendirildiği bilinmektedir. Bu makine 1920'lere kadar popülerliğini sürdürmüştür (Menezes, 1996).

Birinci Dünya Savaşı sonunda şifrelemenin önemi oldukça artmış ve bundan dolayı 1919'da ilk örneği görülen ve sonrasında ordu ve devlet kurumları için özel modelleri üretilen Enigma makinesi, İkinci Dünya Savaşı sırasında Nazi Almanyası tarafından gizli mesajların şifrelenmesi ve tekrar çözülmesi amacı ile kullanılan bir şifre makinesi olmuştur. Savaş sonrasında şifreleme sistemlerinin önemi daha da artmıştır. Çünkü müttefik şifreciler tarafından gelen mesajlar çözümlenmiş ve bazı tarihçilere göre Enigma kod sisteminin deşifre olması sayesinde Avrupa'da savaşın 1 yıl daha erken bittiği ileri sürülmüştür. Bunun sonucunda, İkinci Dünya Savaşı ve stratejik planların aktarılmasında kullanılan şifreleme sistemleri ve bunların çözümlenmesinde kullanılan algoritmalar, buluşlar, şifre çözücü makineler bir anlamda bilgisayar biliminin doğmasına neden olmuştur.

1970' lere kadar daha çok askeri alanda kullanıldığı görülen şifreleme sistemleri bu tarihten sonra sivil amaçlı olarak da kullanılmaya başlanmıştır. Bu ihtiyacın doğmasının en büyük sebeplerinden birisi, bilgisayar kullanımının hayatımıza getirdiği kolaylık sonucu elektronik ortamda haberleşme güvenliliğinin önem kazanmasıdır. Örneğin, elektronik ortamda kredi kartları kullanımı, telefon konuşmaları ve para transferi gibi [1], [2].

Şifrelemenin tarihi hakkında geniş bilgi için "Şifrelerin Matematiği: Kriptografi, ODTÜ Geliştirme Vakfı Yayıncılık" adlı kitaba bakılabilir.

2.2. Şifreleme Sistemlerinin Temel İlkeleri

Şifreleme sistemlerinin aşağıdaki özelliklere sahip olması gerekir [3]:

1. Güvenlik derecesi: Bu aslında zor ölçülebilen bir unsurdur. Genellikle bilgiyi ele geçirme amaçlı olarak bilinen en iyi yöntemlerin kesin sonuç alınıncaya dek uygulanmasındaki işlem sayısı olarak verilir. Tipik bir sistemin güvenlik derecesi bu en yüksek sayıdaki işlem ataklarından (saldırılardan) daha çok işlem yapılmasını gerektirir. Buna bazen işlem unsuru denir.

2. Fonksiyonellik: Şifreleme sisteminin güvenliğini sağlayan kısımları birbirleriyle bütünleşmiş bir yapıda olmalıdır. Sistemin tüm kısımları iletilen çeşitli türdeki bilgileri güvenli bir şekilde çözümleyebilmelidir.

3. İşlem yöntemleri: Şifreleme sisteminin temel yapılarının, uygulama sırasında değişik girişlerle değişik şekillerde çalışması tipik karakteristikler olarak farklılık göstermesidir.

4. Başarım: Bir şifreleme algoritmasının bir saniyede şifreleyebileceği bit sayısıdır.

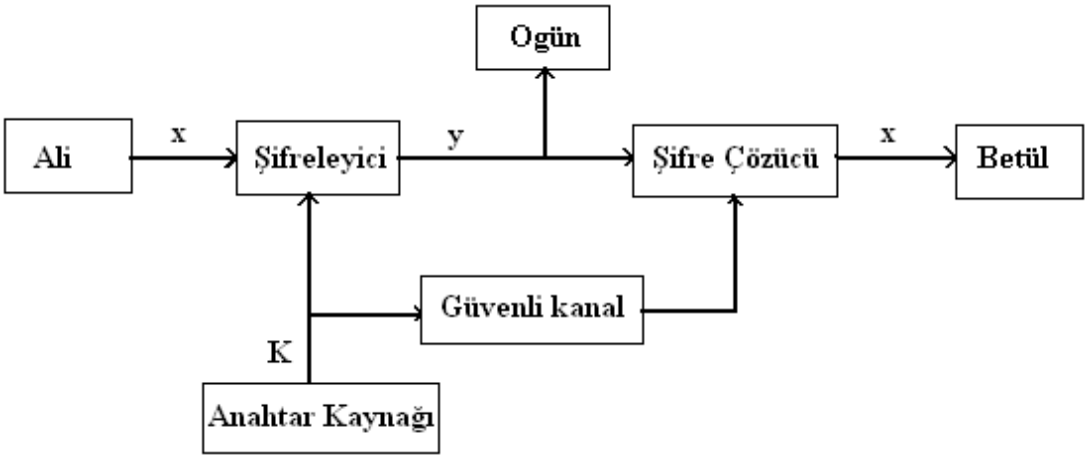
5. Uygulamada kolaylık: Temel bir şifreleme sisteminin yapısının zor durumlarda uygulanabilirliği önemlidir. Bu yapılar karmaşık bir yazılım ya da donanım ortamını içerebilir. Sistemin yazılım veya donanım bölümüyle ilgili karmaşıklık derecesi işlem gücünü etkiler.

2.3. Birkaç Temel Klasik Şifreleme Sistemi

Bu kısımda şifrelemenin ilk dönemlerinde kullanılan klasik şifrelemelere örnek teşkil edecek birkaç klasik şifreleme sistemi ele alınacaktır. Şu anda bu şifrelemelerin klasik olmasının en büyük sebebi şifrelemenin ilk dönemlerinde bilgisayar olmamasıdır. Bilgisayar teknolojisinin gelişmesi ve bunun sonucu olarak şifreleme analiz metodlarının bilgisayar aracılığıyla kullanılması, klasik şifreleme sistemlerinin

güvenirliliğini ortadan kaldırmış ve bunların yerine daha güvenilir şifreleme sistemleri geliştirilmiştir.

Şifrelemenin temel amacı, güvenli olmayan bir kanal aracılığıyla, saklı olan bilgiyi almaya çalışan kişinin de bulunduğu bu ortamda iki kişinin iletişimine olanak sağlamaktır. Bu kısımda iletişim kurmaya çalışan iki kişi Ali ve Betül, iki kişi arasındaki saklı olan bilgiye izinsiz ulaşmaya çalışan kişi de Ogün olsun. Güvenli olmayan kanal ise telefon hattı veya bilgisayar ağı olarak düşünülebilir. Ali adlı kişinin Betül adlı kişiye göndermek istediği bilgiye açıkmetin(plaintext) denir. Ali elindeki açıkmetni önceden belirlenmiş anahtarı(key) kullanarak şifreler ve anahtar kullanılarak şifrelenmiş şifrelimetni(ciphertext) Betül adlı kişiye bu kanaldan gönderir. Ogün de dinleyici olarak bulunduğu kanalda şifrelimetni görse bile açıkmetnin ne olduğunu belirleyemez. Ama Betül deşifreleme anahtarını bildiği için şifrelimetni çözer ve açıkmetni elde eder. Şekil 2.1 bu ilişkiyi göstermektedir [5].



Şekil 2.1 İletişim Kanalı

Klasik şifreleme sistemleri örneklerinde Türkçe alfabesi üzerinde şifreleme yapılacağından alfabedeki harfler ile modül 29 arasında Şekil 2.2'deki gibi bir ilişki kurulabilir.

A↔0, B↔1, C↔2, Ç↔3, . . . , Z↔28

A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K
0	1	2	3	4	5	6	7	8	9	10	11	12	13

L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28

Şekil 2.2 Türkçe Alfabenin Modül 29 ile olan ilişkisi

2.3.1. Öteleme (Shift) Şifrelemesi

$\mathbb{Z}_m$, Tanım 1.1.9 dan dolayı toplama ve çıkarma işlemlerine göre halka olduğu görülür. $0 \leq K \leq 28$ için $e_k(x) = x + K (\mathrm{mod}\, 29)$ şeklinde şifreleme fonksiyonu alınabilir. Bu fonksiyon öteleme fonksiyon olarak adlandırılır. Bu şekilde yapılan şifrelemeye de öteleme(shift) şifrelemesi denir. Eğer şifreleme fonksiyonunda özel olarak $K = 3$ alınırsa şifreleme sistemi, Julius Caesar'ın kullandığı Caesar şifrelemesi olarak adlandırılır. Şifreleme fonksiyonu

$$e_k(x) = x + K (\mathrm{mod}\, 29)$$

olduğundan şifreyi çözücü fonksiyon ise,

$$d_k(x) = x - K (\mathrm{mod}\, 29)$$

dır. Çünkü,

$$d_k(e_k(x)) = d_k(x + K) = x + K - K = x (\mathrm{mod}\, 29)$$

olmalıdır. Dolayısıyla öteleme(shift) şifrelemesi için, $0 \leq K \leq 28$ ve $x, y \in \mathbb{Z}_{29}$ olmak üzere;

$$e_k(x) = x + K (\mathrm{mod}\, 29) \quad \text{ve} \quad d_k(x) = x - K (\mathrm{mod}\, 29)$$

dir [5].

Örnek 2.3.1.1. Caesar şifrelemesi kullanılarak

"SAKARYAÜNİVERSİTESİ"

açıkmetni aşağıdaki şekilde şifrelenir:

Açıkmetnin her bir harfine karşılık gelen sayı değerleri Şekil 2.2 kullanılarak aşağıdaki şekilde yazılır:

S	A	K	A	R	Y	A	$Ü$	N	$İ$	V	E	R	S	$İ$	T	E	S	$İ$
$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
21	0	13	0	20	27	0	25	16	11	26	5	20	21	11	23	5	21	11

Caesar şifrelemesinde $K = 3$ olduğu için her bir harfin sayı değerini üç öteleyerek ya da her bir harfin sayı değerine üç ekleyerek modül 29 işlemine göre karşılık gelen sayı değerleri Şekil 2.2'den bulunarak şifrelimetin elde edilir. Yani,

	S	A	K	A	R	Y	A	$Ü$	N	$İ$	V	E	R	S	$İ$	T	E	S	$İ$
	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
	21	0	13	0	20	27	0	25	16	11	26	5	20	21	11	23	5	21	11
$K=3\Rightarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
	<u>24</u>	<u>3</u>	<u>16</u>	<u>3</u>	<u>23</u>	<u>1</u>	<u>3</u>	<u>28</u>	<u>19</u>	<u>14</u>	<u>0</u>	<u>8</u>	<u>23</u>	<u>24</u>	<u>14</u>	<u>26</u>	<u>8</u>	<u>24</u>	<u>14</u>
	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
	U	$Ç$	N	$Ç$	T	B	$Ç$	Z	P	L	A	$Ğ$	T	U	L	V	$Ğ$	U	L

bulunur. Altı çizili olan sayılar, modül 29 işlemine göre elde edilen kalanlardır.

Açıkmetin: "SAKARYAÜNİVERSİTESİ"

Şifrelimetin: "UÇNÇTBÇZPLAĞTULVĞUL"

elde edilir. Şifrelimetni çözmek için şifrelimetindeki harflere karşılık gelen sayı değerlerinden modül 29 işlemine göre herbirinden üç geri öteleyerek ya da üç çıkararak karşılık gelen sayı değerleri Şekil 2.2'de karşılığı bulunarak açıkmetin

elde edilir. Örneğin, şifrelimetinde bulunan U ve Ç harflerine karşılık gelen sayı değerleri sırasıyla 24 ve 3 sayılarıdır. Bu sayı değerlerinden, $K=3$ olduğundan üç geri öteleyerek ya da üç çıkararak karşılık gelen sayı değerleri Şekil 2.2 kullanılarak bulunur. Öteleme sonucu elde edilen 21 ve 0 sayılarına karşılık gelen harfler sırasıyla S ve A harfleridir. Dolayısıyla deşifreleme aynı şekilde diğer harflere uygulanarak yapılır.

Öteleme(shift) şifrelemesinde şifreyi kırmak için aşağıdaki yöntemler gözönünde bulundurulabilir [5]:

1) Sadece Şifrelimetin: Kişi sadece şifrelimetne sahiptir. Anahtar için 29 ihtimal vardır.

2) Bilinen Açıkmetin: Açıkmetinden sadece bir harf ve bu harfe karşılık gelen şifrelimetindeki harf biliniyorsa, anahtar bulunabilir. Örneğin, açıkmetinden sadece bilinen harf T ve T harfine karşılık gelen şifrelimetindeki harf ise D olsun. T ve D harflerinin sayısal değerleri sırasıyla Şekil 2.2'den 23 ve 4 tür. Bu taktirde, T harfinden D harfine 10 sayı ötelenerek ulaşılır ya da diğer bir ifadeyle $K \equiv 4-23 \equiv -19 \equiv 10 \pmod{29}$ bulunur.

3) Seçili Açıkmetin: Açıkmetinde A harfi seçilsin. Şifrelimetin anahtarı verir. Örneğin, eğer şifrelimetin H ise anahtar $K=9$ bulunur.

4) Seçili Şifrelimetin: Şifrelimetinde A harfi seçilsin. Örneğin, eğer açıkmetin H ise anahtar $K \equiv 0-9 \equiv -9 \equiv 20 \pmod{29}$ bulunur ya da H harfinden A harfine 20 sayı öteleme yapıldığından $K=20$ bulunur.

Örnek 2.3.1.2. "ŞGBJŞGBÖR" şifrelimetin $K=7$ anahtarı kullanılarak şu şekilde deşifrelenir:

Şifre çözücü fonksiyon $d_k(x) = x - 7 \pmod{29}$ şeklindedir. Yani, açıkmetnin sayı değerlerinden yedi sayı geri ötelendiği görülür. Şifrelimetnin her bir harfine karşılık gelen sayı değerleri Şekil 2.2 kullanılarak yazılır. Buna göre,

Ş	G	B	J	Ş	G	B	Ö	R
↓	↓	↓	↓	↓	↓	↓	↓	↓
22	7	1	12	22	7	1	18	20

değerleri bulunur. Sayısal değerler sırasıyla şifre çözücü fonksiyonda yerlerine yazılırsa;

	Ş	G	B	J	Ş	G	B	Ö	R
	↓	↓	↓	↓	↓	↓	↓	↓	↓
	22	7	1	12	22	7	1	18	20
$K = 7 \Rightarrow$	↓	↓	↓	↓	↓	↓	↓	↓	↓
	<u>15</u>	<u>0</u>	<u>23</u>	<u>5</u>	<u>15</u>	<u>0</u>	<u>23</u>	<u>11</u>	<u>13</u>
	↓	↓	↓	↓	↓	↓	↓	↓	↓
	M	A	T	E	M	A	T	İ	K

elde edilir. Altı çizili olan sayılar, modül 29 işlemine göre elde edilen kalanlardır.

Şifrelimetin: "ŞGBJŞGBÖR"

Açıkmetin: "MATEMATİK"

olur.

2.3.2. Afin(Affine) Şifrelemesi

$a, b \in \mathbb{Z}_{29}$ ve $ebob(a, 29) = 1$ olmak üzere şifreleme fonksiyonu;

$$e_k(x) = ax + b \pmod{29}$$

şeklinde tanımlanır. Bu fonksiyona afin fonksiyon denir [5]. ($a=1$ alındığında öteleme şifrelemesi elde edilir.)

Şifrelemenin mümkün olabilmesi için, afin fonksiyonun birebir olup olmadığına bakmak gerekir. Diğer bir deyişle, her $y \in \mathbb{Z}_{29}$ için,

$$ax+b \equiv y \pmod{29}$$

kongrüansı x için tek bir çözüme sahip olmalıdır. Bu kongrüans

$$ax \equiv y-b \pmod{29}$$

kongrüansına denktir. Bu durumda, $\mathbb{Z}_{29}$ üzerinde y değişirken aynı zamanda $y-b$ de değişmektedir. Dolayısıyla, $ax \equiv y \pmod{29}$ kongrüansında çalışmak yeterlidir.

Kabul edelimki $ebob(a,29)=d>1$ olsun. Bu taktirde $ax \equiv 0 \pmod{29}$ kongrüansı $\mathbb{Z}_{29}$ üzerinde en azından iki farklı çözüme sahiptir. Bunlar $x=0$ ve $x=\frac{29}{d}$ dir. Bu durumda $e_k(x)=ax+b \pmod{29}$ birebir fonksiyon değildir ve bu nedenle de geçerli bir şifreleme fonksiyonu yoktur. Dolayısıyla bu kongrüansın tek bir çözümünün olması için $ebob(a,29)=1$ olmalıdır [5].

Çarpımın birleşim özelliğinden,

$$a^{-1}(ax) \equiv (a^{-1}a)x \equiv 1x \equiv x$$

elde edilir. Sonuç olarak, $x \equiv a^{-1}(y-b) \pmod{29}$ yazılabilir. Bu ise şifre çözücü fonksiyondur.

$\kappa = \{(a,b) \in \mathbb{Z}_{29} \times \mathbb{Z}_{29} : ebob(a,29)=1\}$ ve $K=(a,b) \in \kappa$ olmak üzere;

$$e_k(x) = ax + b \pmod{29} \text{ ve } d_k(y) = a^{-1}(y-b) \pmod{29}$$

şifreleme ve şifre çözücü fonksiyonları olur.

Örnek 2.3.2.1. $K = (5,3)$ anahtarı ile "MATEMATİK" açıkmetni afin şifrelemesine göre aşağıdaki gibi şifrelenir:

mod 29 işlemine göre $5^{-1} = 6$ olduğundan, şifreleme ve şifre çözücü fonksiyonlar

$$e_k(x) = 5x + 3 \pmod{29} \text{ ve } d_k(y) = 6(y-3) = 6y - 18 \pmod{29}$$

olarak elde edilir. Açıkmetnin her bir harfine karşılık gelen sayı değerleri Şekil 2.2 kullanılarak aşağıdaki şekilde yazılır:

M	A	T	E	M	A	T	$İ$	K
$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
15	0	23	5	15	0	23	11	13

Bu sayısal değerler sırasıyla şifreleme fonksiyonu olan $e_k(x) = 5x + 3 \pmod{29}$ fonksiyonunda yerine yazılırsa;

	M	A	T	E	M	A	T	$İ$	K
	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
	15	0	23	5	15	0	23	11	13
$K=(5,3)\Rightarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
	$\underline{20}$	$\underline{3}$	$\underline{2}$	$\underline{28}$	$\underline{20}$	$\underline{3}$	$\underline{2}$	$\underline{0}$	$\underline{10}$
	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
	R	$Ç$	C	Z	R	$Ç$	C	A	I

elde edilir. Altı çizili olan sayılar, modül 29 işlemine göre elde edilen kalanlardır. Dolayısıyla afin şifrelemesine göre alınan $K = (5,3)$ anahtarı için,

Açıkmetin: “MATEMATİK”

Şifrelimetin: “RÇCZRÇCAI”

olur. Benzer şekilde elde edilen şifrelimetin $d_k(y)=6y-18(\mathrm{mod}\,29)$ şifre çözücü fonksiyonda yerine yazılırsa açıkmetin elde edilir.

Şifreleme metodunun anahtarı (a,b) ikilisidir. a için 28 mümkün durum vardır ve b için $ebob(a,29)=1$ olmak üzere 29 mümkün durum vardır. Anahtarı bulmak için 28∗29=812 seçenek vardır.

Afin şifrelemesinde şifreyi kırmak için aşağıdaki yöntemler gözönünde bulundurulabilir [5]:

1) Sadece Şifrelimetin: Anahtar için 812 seçeneği incelemek öteleme şifrelemesinden daha geniş kapsamlı bir tarama yapmayı gerektirir ve bu uzun olabilir; ancak bilgisayar aracılığıyla daha kısa sürede tarama yapılabilir. Burada harflere atanılan değerler dikkate alınarak anahtarı hesaplamak mümkün olabilir.

2) Bilinen Açıkmetin: Açıkmetinden iki harfi, bu harflere şifrelimetinde karşılık gelen harfleri ve bu harflere atanılan değerleri bilmek anahtarı çözmede yeterli olabilir. Örneğin, açıkmetin MA ile başlasın ve şifreli metinde karşılığı RÇ olsun. Bu taktirde, açıkmetinde M ve A harflerine karşılık gelen sayısal değerler Şekil 2.2’den sırasıyla 15 ve 0; şifrelimetindeki R ve Ç harflerine karşılık gelen sayısal değerler ise Şekil 2.2’den sırasıyla 20 ve 3 tür. Bu değerler şifreleme fonksiyonunda yazılırsa,

$$15a+b=20(\mathrm{mod}\,29) \qquad \text{ve} \qquad 0a+b=3(\mathrm{mod}\,29)$$

denklemleri elde edilir. Buradan $b=3$ ve $a=5$ bulunur. $ebob(5,29)=1$ olduğundan $K=(5,3)$ anahtarı elde edilir.

3) Seçili Açıkmetin: Açıkmetinde AB harfleri alınsın. Şifrelimetindeki ilk karakter, $a.0+b=b$ olur. İkinci karakter $a+b$ olur. Bu şekilde anahtar bulunur.

4) Seçili Şifrelimetin: AB şifrelimetinden seçilsin. Bu, $x=a_1y+b_1$ şeklindeki şifre çözücü fonksiyonu sağlar. Dolayısıyla denklem y için çözülebilir ve şifreleme fonksiyonu elde edilir.

2.3.3. Vigenere Şifrelemesi

16. yüzyılda yaşamış olan Blaise de Vigenere öteleme şifrelemesini daha genelleştirerek bu klasik şifrelemeye kendi ismini vermiştir. Burada sırasıyla öteleme şifrelemesinde olduğu gibi benzer biçimde şifreleme fonksiyonu ve şifre çözücü fonksiyonu,

$K=(k_1,k_2,...,k_m)$ anahtarı için,

$$e_k(x_1,x_2,...,x_m)=(x_1+k_1,x_2+k_2,...,x_m+k_m)(\mathrm{mod}\,29)$$

ve

$$d_k(y_1,y_2,...,y_m)=(y_1-k_1,y_2-k_2,...,y_m-k_m)(\mathrm{mod}\,29)$$

şeklindedir. Öteleme şifrelemesinde, açıkmetindeki her harf aynı sayı değeriyle ötelenirken; Vigenere şifrelemesinde, açıkmetindeki her harf farklı sayı değeriyle ötelenir.

Örnek 2.3.3.1. $K=(21,13,5,4,17,8)$ anahtarı ile "SAKARYAÜNİVERSİTESİ" açıkmetni Vigenere şifrelemesi ile aşağıdaki gibi şifrelenir:

Açıkmetnin her bir harfine karşılık gelen sayı değerleri Şekil 2.2 kullanılarak şu şekilde yazılır:

S	A	K	A	R	Y	A	Ü	N	İ	V	E	R	S	İ	T	E	S	İ
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
21	0	13	0	20	27	0	25	16	11	26	5	20	21	11	23	5	21	11

$K=(21,13,5,4,17,8)$ anahtarı, altılı gruplar halinde elde edilen sayısal değerlere eklenir.

S	A	K	A	R	Y	A	Ü	N	İ	V	E	R	S	İ	T	E	S	İ
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
21	0	13	0	20	27	0	25	16	11	26	5	20	21	11	23	5	21	11
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
21	13	5	4	17	8	21	13	5	4	17	8	21	13	5	4	17	8	21
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
<u>13</u>	<u>13</u>	<u>18</u>	<u>4</u>	<u>8</u>	<u>6</u>	<u>21</u>	<u>9</u>	<u>21</u>	<u>15</u>	<u>14</u>	<u>13</u>	<u>12</u>	<u>5</u>	<u>16</u>	<u>27</u>	<u>22</u>	<u>0</u>	<u>3</u>
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
K	K	Ö	D	Ğ	F	S	H	S	M	L	K	J	E	N	Y	Ş	A	Ç

Altı çizili olan sayılar, modül 29 işlemine göre elde edilen kalanlardır.

Açık metin: “SAKARYAÜNİVERSİTESİ”

Şifrelimetin: “KKÖDĞFSHSMLKJENYŞAÇ”

olur.

2.3.4. Hill Şifrelemesi

Bu şifreleme 1929’da Lester S.Hill tarafından inşa edildi. Şifrelenmiş metindeki her eleman, açıkmetnin n tane elemanın lineer birleşimi şeklinde alınır. Açıkmetnin elemanları olarak $x=(x_1,x_2,...,x_n)$ ve şifrelimetnin elemanları olarak da $y=(y_1,y_2,...,y_n)$ alınırsa $n\times n$ biçiminde K anahtar matrisi elde edilir. Açıkmetni şifrelemek için;

$$e_k(x) = xK \pmod{29}$$

denklemini çözmek gerekir. Şifre çözücü fonksiyonu elde etmek için K matrisinin tersinin olması gerekir. Ancak matrisin tersinin olması için $\det(K) \neq 0$ olmalıdır. Bu durumu sağlayan şifre çözücü fonksiyon,

$$d_k(y) = yK^{-1} \pmod{29}$$

şeklindedir. Deşifreleme işleminin yapılabilmesi için

$$(\det K, 29) = 1 \text{ ve } KK^{-1} \equiv K^{-1}K \equiv 1 \pmod{29}$$

şartları sağlanmalıdır [5].

Örneğin, $n = 3$ olması halinde $x = (x_1, x_2, x_3)$ ve $y = (y_1, y_2, y_3)$ şeklinde vektörler ve 3×3 tipinde bir K anahtar matrisi elde edilir. Açıkmetni şifrelemek için;

$$(y_1, y_2, y_3) = (x_1, x_2, x_3)[K]_{3\times 3} \pmod{29}$$

denklemini çözmek gerekir. Şifrelimetni deşifrelemek için;

$$(x_1, x_2, x_3) = (y_1, y_2, y_3)[K]_{3\times 3}^{-1} \pmod{29}$$

denklemini çözmek gerekir.

Örnek 2.3.4.1. $K = \begin{pmatrix} 11 & 8 \\ 3 & 7 \end{pmatrix}$ anahtar matrisi ile "MATE" açıkmetni Hill şifrelemesine göre aşağıdaki gibi şifrelenir:

Matris 2×2 tipinde olduğundan şifreleme ikişerli gruplar halinde yapılır. Eğer matris 3×3 tipinde olursa şifreleme üçerli gruplar halinde yapılır. Bu durumda

MATE açıkmetni, MA ve TE olarak ikişerli gruplara ayrılır. Şekil 2.2'den M, A, T, E harflerinin sayısal değerleri sırasıyla 15, 0, 23 ve 5 tir. Dolayısıyla ikişerli gruplar olarak

$$(M,A)=(15,0) \text{ ve } (T,E)=(23,5)$$

yazılır. Şifreleme fonksiyonu,

$$e_k(x)=(y_1,y_2)=(x_1,x_2)\begin{pmatrix}11 & 8\\ 3 & 7\end{pmatrix}$$

olduğundan sırasıyla ikişerli gruplara ayrılan harflerin sayısal değerleri şifreleme fonksiyonunda aşağıdaki gibi yerine yazılır:

$$(y_1,y_2)=(15,0)\begin{pmatrix}11 & 8\\ 3 & 7\end{pmatrix}=(20,4)=(R,D)(\text{mod}\,29)$$

$$(y_1,y_2)=(23,5)\begin{pmatrix}11 & 8\\ 3 & 7\end{pmatrix}=(7,16)=(G,N)(\text{mod}\,29)$$

Açıkmetin: "MATE"

Şifrelimetin: "RDGN"

olur.

BÖLÜM 3. KODLAMA TEORİSİ

3.1. Kodlama ve Tarihi

1940'ların sonlarına doğru C.E.Shannon, R.W.Hamming ve J.C.Golay'ın yazdıkları makaleler yeni bilim sahalarının oluşmasına neden oldu. İlk önce C.E.Shannon makalesinde [6] iletişimin sınırları üzerine dayanan temel bir teori yayınladı. C.E.Shannon, bu makalesinde bit hızının kanal kapasitesi altında olmak şartıyla, herhangi bir kanal üzerinde kodları kullanarak düşük hata olasılığı elde etmenin mümkün olacağını gösterdi. Bununla birlikte, bunun nasıl başarılabileceğini göstermedi. Shannon'ın makalesi [6] en azından iki araştırma sahasının ortaya çıkmasına sebep oldu. Bunlardan biri başlıca performans üzerine dayalı sınırlarla ilgilenen bilgi teorisi (Information Theory) ve diğeri kodları kullanarak iyi iletişimler kurulması için metodlar geliştiren kodlama teorisi (Coding Theory) oldu.

Kodlama teorisi C.E.Shannon ve R.W.Hamming ile başladı. R.W.Hamming [7], 1950'de tek hata düzelten ikili kodların bir sınıfınının inşasını yapıp bunu makalesinde yayınladı. Bu kodlar, C.E.Shannon tarafından 1948'deki makalesinde söz edildi. J.C.Golay [8], C.E.Shannon'ın makalesi aracılığıyla R.W.Hamming'in keşfini öğrendi. 1949'da J.C.Golay yapının genelleştirilmiş halini yayınladı. Hem R.W.Hamming'in orjinal ikili kodları hem de J.C.Golay'ın genelleştirilmeleri, şu anda Hamming kodları olarak bilinir. Daha önemlisi, J.C.Golay birden çok hata düzelten kodların iki yapısını verdi. Bu kodlar da kendi ismi ile Golay kodlar olarak bilinir.

R.W.Hamming ve J.C.Golay tarafından yapılan keşifler hem mühendisler hem de matematikçiler tarafından araştırma sahaları oluşturdu. Mühendisler ilk önce gelişmiş bilgi iletişimi için yeni olanakları kullanmak istedi. Diğer taraftan, matematikçiler ise

daha çok kodların cebirsel yapılarına ilgi duydu. Dolayısıyla kodlama teorisi bu şekilde gelişti.

Modern iletişimin gereksinimleri sonucu bugün çok sayıda kodlar inşa edildi ve bunun sonucu olarak da bu kodların taşınması sırasında oluşabilecek hataların düzeltilmesi gerekliliğinden hata düzelten kodlar teorisi de oldukça gelişti. Bu teori sayesinde modern telekomünikasyon ve uzay iletişiminde verilerin hızlı ve doğru bir şekilde iletilmesi sağlandı [9].

3.2. Kodlama ile Bilgi Aktarımı

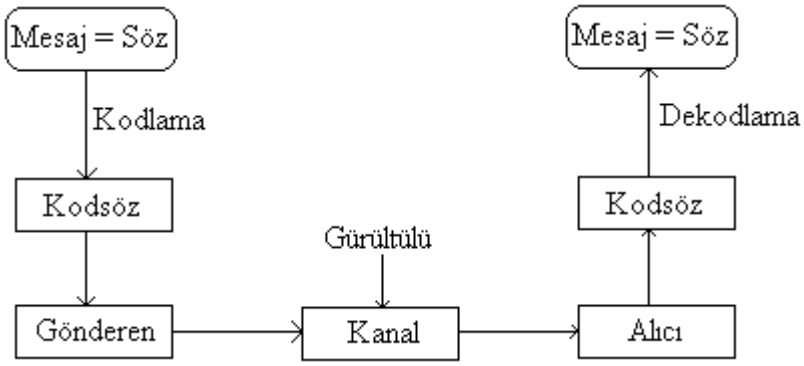
A sonlu bir alfabe olmak üzere,

$$f : A^k \to A^n , \quad (x_1,...,x_k) \to (x_1, x_2,..., x_n)$$

şeklinde k uzunluğundaki bir sonlu sıralı k-lıyı bir sıralı n-liye resmeden f fonksiyonuna kodlama denir. Ancak, bilgi aktarımı esnasında değişik nedenlerden dolayı (manyetik alan,silinme, vs.) meydana gelebilecek bilgi kayıplarını tekrar düzeltme ve aktarılan k uzunluğundaki bilgiyi doğrulayabilmek önemlidir. Bu amaçla hata düzelten kodlar yardımıyla, k bilgi bileşenine (information bits) bağlı $n-k$ ek bileşenler (parity check bits) tanımlanarak gönderilen mesaj düzeltilmeye çalışılır. Burada $(x_1, x_2,..., x_k)$ mesaj, söz veya bilgi vektörü; $(x_1, x_2,..., x_n)$ kodsöz olarak adlandırılır.

A^n in herhangi bir C alt kümesine n-li blok kod denir. Eğer $C \subset A^n$ nin M tane elemanı varsa, C ye n uzunluğunda, M elemanlı bir kod denir ve C ye kısaca bir (n, M)- kodu denir [10].

Bilgi aktarımı, Şekil 3.1'deki şema ile özetlenmektedir .



Şekil 3.1. Kodlama ile Bilgi Aktarımı

3.3. Kodlama Teorisinde Temel Kavramlar

Dijital ortamlarda yapılan iletişimlerde kullanılan alfabe 0 ve 1 lerden oluştuğu için bu bölümdeki işlemler $\mathcal{B} = \mathbb{Z}_2$ cismi üzerinde olacaktır.

Tanım 3.3.1. [11] $\mathcal{B} = \{0,1\}$ ve n bir pozitif tamsayı olmak üzere,

$$\mathcal{B}^n = \{x_1 x_2 ... x_n : i = 1,2,3,...,n \text{ için } x_i \in \mathcal{B}\}$$

olsun. $\mathcal{B}^n$ kümesi üzerinde toplama işlemi,

$$x = x_1 x_2 ... x_n$$
$$y = y_1 y_2 ... y_n$$

olmak üzere

$$x + y = z_1 z_2 ... z_n$$

şeklindedir.

Burada $i = 1,2,3,...,n$ için,

$$z_i = (x_i + y_i) \mod 2$$

olarak tanımlanır.

Tanım 3.3.2. [11] $m \leq n$ olmak üzere,

$$E : \mathcal{B}^m \rightarrow \mathcal{B}^n$$

ile tanımlanmış 1-1 fonksiyonuna kodlama fonksiyonu ve

$$D : \mathcal{B}^n \rightarrow \mathcal{B}^m$$

şeklindeki fonksiyonuna ise dekodlama fonksiyonu denir.

Tanım 3.3.3. [11] $E : \mathcal{B}^m \rightarrow \mathcal{B}^n$ kodlama fonksiyonun tanım kümesi sözlerin kümesi, görüntü kümesi kodsözlerin kümesi olarak adlandırılır.

Örnek 3.3.1. $E : \mathcal{B}^3 \rightarrow \mathcal{B}^4$ kodlama fonksiyonunda C kodu aşağıdaki gibi oluşturulabilir:

$$\mathcal{B}^3 = \{000, 001, 010, 100, 011, 110, 111, 101\}$$

$$\mathcal{B}^4 = \{0000, 0001, 0010, 0100, 1000, 1100, 1010, 1001, 0110, 0101, 0011, 1110, 0111, 1101, 1011, 1111\}$$

kümeleri için E kodlama fonksiyonu şu şekilde tanımlanabilir:

$$E : \mathcal{B}^3 \rightarrow \mathcal{B}^4$$

000	$\rightarrow$	0000	011	$\rightarrow$	0111
001	$\rightarrow$	0011	110	$\rightarrow$	1100
010	$\rightarrow$	0101	111	$\rightarrow$	1111
100	$\rightarrow$	1001	101	$\rightarrow$	1010.

Bu durumda C kodsözlerin kümesi

$$C = \{0000, 0011, 0101, 1001, 0111, 1100, 1111, 1010\}$$

şeklindedir.

Tanım 3.3.4. [11] $x, y \in \mathcal{B}^n$ ise,

$$d(x_i, y_i) = \begin{cases} 0, & \text{eğer } x_i = y_i \text{ ise} \\ 1, & \text{eğer } x_i \neq y_i \text{ ise} \end{cases}$$

olmak üzere

$$d(x, y) = \sum_{i=1}^{n} d(x_i, y_i)$$

şeklinde tanımlanan fonksiyona uzaklık fonksiyonu ve $d(x, y)$ ye de x ile y arasındaki uzaklık denir. Kısaca aynı pozisyonda bulunan farklı değerlerin toplamı uzaklığı verecektir. Ayrıca bu şekilde tanımlanan uzaklık fonksiyonu ile $\mathcal{B}^n$ bir metrik uzay olur.

Örnek 3.3.2. x ve y kodsözleri aşağıdaki gibi alınsın:

$$x = 10000101$$
$$y = 01101011.$$

Bu taktirde iki kodsöz arasındaki uzaklık, aynı pozisyonda bulunan farklı elemanların sayılarının toplamı olduğundan,

$$d(x, y) = 6$$

dır.

Tanım 3.3.5. [11] $x \in \mathcal{B}^n$ olmak üzere x kodsözlerinin ağırlığı x dizisinin sıfırdan farklı bileşenlerinin sayısıdır. Yani,

$$w(x) = \sum_{i=1}^{n} x_i$$

şeklinde tanımlanan fonksiyona ağırlık fonksiyonu denir ve x kodsözün ağırlığı ise $w(x)$ ile gösterilir.

Örnek 3.3.3. Aşağıda verilen x ve y kodsözlerinin sırasıyla ağırlıkları,

$$x = 10101011 \Rightarrow w(x) = 5$$
$$y = 00110011 \Rightarrow w(y) = 4$$

şeklindedir.

Tanım 3.3.6. [12] C kodunun elemanları olan kodsözler arasındaki uzaklıkların en küçüğüne C kodunun minimum uzaklığı denir.

C kodunun minimum uzaklığı $d(C)$ ile gösterilirse,

$$d(C) = \min\left\{ d(x,y) \middle| x, y \in C, x \neq y \right\}$$

dir.

Tanım 3.3.7. [11] Kodlamada gönderilen kodsöz x ve buna karşılık olarak alınan kodsöz ise x^* ise e hata vektörü olmak üzere,

$$x^* = x + e$$

şeklindedir. e hata vektörünün ağırlığı kodsözde kaç hata meydana geldiğini gösterir.

Örnek 3.3.4. $x^* = 1001011$ ve $x = 1001101$ ise $e = 0000110$ dir. e hata vektörünün ağırlığı 2 olduğundan kodsözde 2 hata meydana gelmiştir.

Alıcı sadece kanal aracılığıyla gelen x^* değerini bilmektedir. Dolayısıyla e hata vektörünü bilmediğinden yukarıdaki işlemle x değerini elde edemez. İşte kodlamanın amacı da buradaki e hata vektörünü eğer tespit edilebiliyorsa tespit etmek ve düzeltip düzeltilemeyeceğine karar vermektir. Şimdi bir kodlamada hangi hataların tespit edilip düzeltileceğine dair birkaç teorem verelim.

Bir hatanın tespit edilebilmesi için koddan aldığımız elemanın hata vektörü ile işlemi sonucu elde edilen kodsözün tekrar kodun elemanı olması gerekir. Aksi taktirde hata tespit edilemez.

Teorem 3.3.1. [3] Kodlamada C kodunun minimum uzaklığı en az $t+1$ ise C kodu en fazla t hata tespit edebilir. Yani, $d(C) \geq t+1$ ise C kodu en fazla t hata tespit edebilir.

Teorem 3.3.2. [3] Kodlamada C kodunun minimum uzaklığı en az $2t+1$ ise C kodu en fazla t hata düzeltebilir.

Örnek 3.3.5. $E : \mathcal{B}^2 \rightarrow \mathcal{B}^5$ kodlamasında kodsözlerin eşleşmesi aşağıdaki şekilde olsun. $\mathcal{B}^2$ nin eleman sayısı $2^2 = 4$ olacağından $\mathcal{B}^2$ nin elemanları

$$\begin{aligned} 00 &\rightarrow 00111 \\ 01 &\rightarrow 01111 \\ 10 &\rightarrow 10111 \\ 11 &\rightarrow 11111 \end{aligned}$$

şeklinde eşlenebilir. Bütün kodsözler ikişerli olarak birbiriyle karşılaştırıldıklarında birbirlerine uzaklıkları 3, 4 ve 5 tir. Dolayısıyla minimum uzaklık 3 tür. Yani, $d(C)=3$ tür. Teorem 3.3.1'den, C kodunun minimum uzaklığı $d(C)=3\geq t+1\Rightarrow 2\geq t$ olacak şekilde en fazla $t=2$ hataya sahip olan kodsözleri tespit edebilir. En fazla $t=2$ hataya sahip olan kodsözler

00000	00011	10010
00001	00101	01100
00010	10001	10100
00100	01001	11000
01000	00110	
10000	01010	

kodsözleridir.

Teorem 3.3.2'den, minimum uzaklığı 3 olan C kodu, $2t+1=3\Rightarrow t=1$ değerinden en fazla 1 hataya sahip kodsözleri düzeltebilir. En fazla $t=1$ hataya sahip olan kodsözler

00000	00100
01000	10000
00001	00010

kodsözleridir.

3.4. Lineer Kodlar

$V(n,q)=F_q^n$, n uzunluğunda ve q elemanlı sonlu bir cisim üzerinde tanımlanmış bir vektör uzayı olsun.

Tanım 3.4.1. [10] $V(n,q)$ vektör uzayının alt vektör uzayına lineer kod denir. Eğer C kodunun boyutu k ise o zaman C koduna bir $[n,k]$- kodu denir. Aynı zamanda C kodunun minimum uzaklığı d ise C koduna bir $[n,k,d]$- kodu denir.

Örnek 3.4.1. C_1 ve C_2 kodlarını aşağıdaki şekilde olsun:

$$C_1 = \{11000, 01110, 10011, 00101\} \quad \text{ve} \quad C_2 = \{00000, 10110, 01011, 11101\}.$$

C_1 deki bütün kodsözlerin toplamı gözönüne alındığında, $11000 + 01110 = 10110 \notin C_1$ olduğundan C_1 lineer bir kod değildir. C_2 deki bütün kodsözler toplamı gözönüne alındığında,

$$\begin{aligned}
&00000 + 10110 = 10110 \in C_2 \\
&00000 + 01011 = 01011 \in C_2 \\
&00000 + 11101 = 11101 \in C_2 \\
&10110 + 01011 = 11101 \in C_2 \\
&10110 + 11101 = 01011 \in C_2 \\
&01011 + 11101 = 10110 \in C_2
\end{aligned}$$

C_2 kodunun lineer bir kod olduğu görülür.

Tanım 3.4.2. [10] C kodunun elemanları olan kodsözler arasındaki ağırlıkların en küçüğüne C kodunun minimum ağırlığı denir ve $w(C)$ ile gösterilir. Yani,

$$w(C) = \min\left\{ w(x, y) \middle| x, y \in C, x \neq y \right\}$$

dir.

Teorem 3.4.1. [10] C lineer bir kod ise $d(C) = w(C)$ dir. Yani, lineer bir kodun minimum uzaklığı minimum ağırlığına eşittir.

Genel olarak bir (n, M) - kodun minimum uzaklığını bulmak için $\binom{M}{2}$ tane uzaklık hesap etmek gerekir. Ancak kod lineer ise sadece $(M-1)$ tane kodsözün ağırlığına bakmak kodun minimum uzaklığını bulmak için yeterlidir. Örneğin; C kodunun

eleman sayısı 4 ise C kodunun minimum uzaklığını bulmak için C kodundaki her kodsözün birbiriyle olan uzaklıklarını hesaplamak için $\binom{4}{2}=6$ işlem yapmak gerekirken, eğer bu lineer kod ise $4-1=3$ tane işlem yapmak gerekir. Dolayısıyla kodun eleman sayıları büyük olduğu düşünüldüğünde kodun lineer olması oldukça kolaylık sağlar.

Örnek 3.4.2. $C=\{00000,10110,01011,11101\}$ lineer kodunun uzaklığı iki şekilde bulunabilir. Birincisi; kodun minimum uzaklığı hesaplanır. Bunun için kodsözlerin birbirine olan uzaklıkları bulunur. Buradan,

$$d_1(00000,10110)=3 \qquad d_4(10110,01011)=4$$
$$d_2(00000,01011)=3 \qquad d_5(10110,11101)=3$$
$$d_3(00000,11101)=4 \qquad d_6(01011,11101)=3$$

elde edilir. Dolayısıyla kodun minimum uzaklığı,

$$d(C)=\min\{d_1,d_2,d_3,d_4,d_5,d_6\}=3$$

olur. İkincisi; C lineer kod olduğundan Teorem 3.4.1'e göre kodun minimum uzaklığı kodun minimum ağırlığına eşittir. Bunun için;

$$w_1(10110)=3$$
$$w_2(01011)=3$$
$$w_3(11101)=4$$

bulunur. Buradan,

$$w(C)=\min\{w_1,w_2,w_3\}=3=d(C)$$

elde edilir. Dolayısıyla her iki durumda minimum uzaklık 3 tür. Görülür ki kodun lineer olması daha hızlı bir şekilde minimum uzaklığın hesaplanmasını sağlar.

Tanım 3.4.3. [10] C bir lineer $[n,k]$- kodu olsun. Satırları C kodunun bir baz vektörlerinden oluşan $k \times n$ boyutlu G matrisine, C kodunun üreteç matrisi denir.

Eğer G matrisi C kodunun üreteç matrisi ise C kodunun kodsözleri, G matrisinin satırlarının lineer birleşimidir. Yani,

$$C = \{xG | x \in V(k,q)\}$$

şeklindedir. Bu kodlamada kolaylık sağlar. Eğer mesajlar k uzunluğunda ve q- lu sözler olarak temsil edilirse, x sözü xG kodsözü olarak kodlanır.

Elementer satır işlemleri (satırların yerlerini değiştirmek, satırları sıfırdan farklı bir skaler ile çarpmak ve skalerle çarpılmış bir satırı diğerine eklemek) bir matrise uygulanırsa, bu matrisin satır uzayı (satırların lineer birleşimlerinden oluşan vektör uzayı) değişmez.

Teorem 3.4.2. [10] C bir lineer $[n,k]$- kodu olsun. Herhangi bir k koordinat yeri verilsin. Bu yerler üzerinde C lineer koduna denk olan bir sistematik kod vardır. $G = (I_k | A)$ biçimindeki bir üreteç matrisine standart formdadır denir.

Burada I_k , k boyutundaki birim matrisi temsil eder. Her lineer kodun, standart formda bir üreteç matrisi vardır. $k \times n$ tipindeki bir üreteç matris standart formda ise bu kod ilk k koordinatlarında sistematiktir. Bu ise kodlama ve dekodlama işlemlerini kolaylaştırır.

Örnek 3.4.3. C lineer kodu aşağıdaki gibi olsun:

$$\text{C}=\{00000,10110,01011,11101\}.$$

C lineer kodu için muhtemel üreteç matrisi

$$G=\begin{pmatrix}1 & 0 & 1 & 1 & 0\\ 0 & 1 & 0 & 1 & 1\end{pmatrix}$$

şeklinde alınabilir. $G=\left(I_k\middle|A\right)$ biçiminde olduğundan G üreteç matrisi standart formdadır. $V\left(2,2\right)$ nin sözleri,

$$xG=\left(x_1,x_2\right)\begin{pmatrix}1 & 0 & 1 & 1 & 0\\ 0 & 1 & 0 & 1 & 1\end{pmatrix}=\left(x_1,x_2,x_1,x_1+x_2,x_2\right)$$

şeklinde kodlanır. Bu kodsözler $\left(x_1,x_2,x_1,x_1+x_2,x_2\right)$ şeklindedir. Gerçekten de, $V\left(2,2\right)$ nin sözleri; $\left(0,0\right)$, $\left(1,0\right)$, $\left(0,1\right)$ ve $\left(1,1\right)$ dir. Bu sözlere karşılık gelen kodsözler şu şekilde eşlenir:

$$\begin{aligned}\left(0,0\right) &\rightarrow \left(x_1,x_2,x_1,x_1+x_2,x_2\right)=\left(0,0,0,0,0\right)\\ \left(1,0\right) &\rightarrow \left(x_1,x_2,x_1,x_1+x_2,x_2\right)=\left(1,0,1,1,0\right)\\ \left(0,1\right) &\rightarrow \left(x_1,x_2,x_1,x_1+x_2,x_2\right)=\left(0,1,0,1,1\right)\\ \left(1,1\right) &\rightarrow \left(x_1,x_2,x_1,x_1+x_2,x_2\right)=\left(1,1,1,0,1\right).\end{aligned}$$

Elde edilen $\left(0,0,0,0,0\right)$, $\left(1,0,1,1,0\right)$, $\left(0,1,0,1,1\right)$, $\left(1,1,1,0,1\right)$ kodsözleri C lineer kodunun elemanlarıdır.

G matrisi standart formda olduğu için orjinal mesaj ya da söz, kodlanmış kodsözlerin ilk $k=2$ koordinatlarında bulunur. Bu kodlama sistematik bir kodlamadır.

Tanım 3.4.4. [10] C bir lineer $\left[n,k\right]$- kodu olsun.

$$C^{\perp}=\left\{y\in V\left(n,q\right)\middle|\left\langle y,x\right\rangle=0,\forall x\in C\right\}$$

şeklinde tanımlanan $C^{\perp}$ koduna lineer kodun duali denir.

Teorem 3.4.3. [10] 1. C bir lineer $[n,k]$- kodu olsun. Bu taktirde, C lineer kodun duali de $[n,n-k]$- kodudur.

2. C lineer kod ise $\left(C^{\perp}\right)^{\perp}=C$ dir.

Tanım 3.4.5. [10] C kodunun üreteç matrisi $G=\left(I_k \middle| A\right)$ olmak üzere; $GH^T=0$ şartını sağlayan $H=\left(-A^T \middle| I_{n-k}\right)$ matrisine C kodunun kontrol matrisi denir.

Gerçekten,

$$GH^T=\left(I_k \middle| A\right)\begin{pmatrix}-A\\ I_{n-k}\end{pmatrix}=-A+A=0$$

olduğu görülür. Bu ise H matrisinin satırları ile G matrisinin satırlarının birbirine dik olduklarını gösterir. Ayrıca, $rank\left(H\right)=n-k=boy\left(C^{\perp}\right)$ olduğundan, H matrisi $C^{\perp}$ kodunun bir üreteç matrisidir.

Örnek 3.4.4. C lineer kodu aşağıdaki gibi olsun:

$$C=\{00000,10110,01011,11101\}.$$

C lineer kodu için muhtemel üreteç matrisi

$$G=\begin{pmatrix}1 & 0 & 1 & 1 & 0\\ 0 & 1 & 0 & 1 & 1\end{pmatrix}$$

ise G üreteç matrisi standart formdadır. Yani,

$$G=\begin{pmatrix}1 & 0 & 1 & 1 & 0\\ 0 & 1 & 0 & 1 & 1\end{pmatrix}=\left(I_2 \middle| A\right)$$

dır. Burada $I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ ve $A = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$ dır. C kodunun kontrol matrisi, $H = \left(-A^T \middle| I_{n-k}\right)$ şeklinde olacağından $H = \left(-A^T \middle| I_3\right)$ dır. Buradan,

$$H = \left(-A^T \middle| I_3\right) = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

elde edilir. ($\mathbb{Z}_2$ üzerinde $-A^T = A^T$ dır.)

H kontrol matrisi $C^{\perp}$ kodunun üreteç matrisi olduğundan, H kontrol matrisinin satırlarının lineer toplamı $C^{\perp}$ kodunun elemanları olan kodsözleri verir. Bunlar,

$$C^{\perp} = \{00000, 10100, 11010, 01001, 01110, 11101, 10011, 00111\}$$

dır.

3.5. Lineer Kodların Dekodlaması

Tanım 3.5.1. [10] C kodu, kontrol matrisi H olan bir lineer kod olsun. Herhangi bir $x \in C$ için, Hx^T ifadesine x kodsözünün sendromu denir ve bu sendrom s ile gösterilir. Dolayısıyla, $x \in C$ olması için gerek ve yeter şart x kodsözünün sendromunun sıfır olmasıdır. Yani, alınan x kodsözü orjinal kodsöz (hatasız giden kodsöz) ise $Hx^T = 0$ dır.

Tanım 3.5.2. [3] Sendrom dekodlama işlemi aşağıdaki gibi tanımlanır:

1. Eğer sendrom değeri sıfır ise o zaman alınan kodsöz, orjinal kodsözdür.

2. Eğer sendrom değeri H kontrol matrisinin i inci sütunu ise alınan kodsözün i inci bileşeninde hata vardır.

3. Eğer sendrom değeri ne sıfır ne de H kontrol matrisinin bir sütunu değilse alınan kodsözde en az iki hata vardır.

Örnek 3.5.1. $C=\{00000,10110,01011,11101\}$ lineer kodunun üreteç ve kontrol matrislerini aşağıdaki gibi olsun:

$$G=\begin{pmatrix}1&0&1&1&0\\0&1&0&1&1\end{pmatrix} \quad \text{ve} \quad H=\begin{pmatrix}1&0&1&0&0\\1&1&0&1&0\\0&1&0&0&1\end{pmatrix}.$$

1. Alınan kodsöz $r=10110$ olsun. Alınan kodsözde hata olup olmadığını anlamak için sendrom hesaplanır. Dolayısıyla,

$$s=Hr^T=\begin{pmatrix}1&0&1&0&0\\1&1&0&1&0\\0&1&0&0&1\end{pmatrix}\begin{pmatrix}1\\0\\1\\1\\0\end{pmatrix}=(000)$$

olur. Buradan sendrom $s=(000)$ bulunur. Bu da alınan kodsözün hata içermediğini yani alınan kodsözün orjinal kodsöz olduğunu gösterir.

2. Alınan kodsöz $r=10111$ olsun. Bu taktirde sendrom,

$$s=Hr^T=\begin{pmatrix}1&0&1&0&0\\1&1&0&1&0\\0&1&0&0&1\end{pmatrix}\begin{pmatrix}1\\0\\1\\1\\1\end{pmatrix}=(001)$$

olur. Elde edilen $s=(001)$ sendromu alınan kodsözün hatalı olduğunu gösterir. Sendrom değeri, H kontrol matrisinin 5 inci sütunu olduğundan alınan vektörün 5

inci bileşeninde hata olduğunu gösterir. Buradan alınan vektörün 5 inci bileşeni düzeltilerek orjinal kodsöz olan $x = 10110$ elde edilir.

3. Alınan kodsöz $r = 11010$ olsun. Bu taktirde sendrom

$$s = Hr^T = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} = (111)$$

olur. Elde edilen $s = (111)$ sendromu alınan kodsözün hatalı olduğunu gösterir. Sendrom değeri $s = (111)$, H kontrol matrisinin herhangi bir sütununu vermediğinden alınan kodsözde en az 2 hata vardır.

BÖLÜM 4. MATRİS (ARRAY) KODLAR

4.1. Matris Uzayı ve Matris Kodlar

Tanım 4.1.1. F_q, q elemanlı sonlu bir cisim olmak üzere,

$$Mat_{m\times n}\left(F_q\right)=\left\{A=\left(a_{ij}\right)\middle| a_{ij}\in F_q,\ 1\le i\le m,\ 1\le j\le n\right\}$$

şeklinde tanımlanan vektör uzayına matris uzayı denir.

Tanım 4.1.2. [13] $C\subset Mat_{m\times n}\left(F_q\right)$ ile verilen C lineer koduna (veya alt vektör uzayına) matris (array) kod denir.

Örnek 4.1.1. $C\subset Mat_{2\times 2}\left(F_3\right)$ matris kodu, $v_1=\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}$ ve $v_2=\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}$ matris kodsözleri tarafından üretilen kod olsun:

$$C=\left\langle\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix},\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}\right\rangle.$$

Bu taktirde C matris kodunun kodsözleri, v_1 ve v_2 kodsözlerin lineer toplamları olmak üzere aşağıdaki şekildedir:

$$C=\left\{\alpha_1 v_1+\alpha_2 v_2:\alpha_1,\alpha_2\in F_3; v_1,v_2\in Mat_{2\times 2}\left(F_3\right)\right\}.$$

C matris kodunun elemanları aşağıdaki şekilde elde edilir:

α_1	α_2	C matris kodunun elemanları $(\alpha_1 v_1 + \alpha_2 v_2)$
0	0	$0\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}+0\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}=\begin{pmatrix}0 & 0\\ 0 & 0\end{pmatrix}$
1	0	$1\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}+0\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}=\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}$
0	1	$0\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}+1\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}=\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}$
1	1	$1\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}+1\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}=\begin{pmatrix}2 & 1\\ 0 & 2\end{pmatrix}$
2	0	$2\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}+0\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}=\begin{pmatrix}2 & 0\\ 0 & 2\end{pmatrix}$
0	2	$0\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}+2\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}=\begin{pmatrix}2 & 2\\ 0 & 2\end{pmatrix}$
2	1	$2\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}+1\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}=\begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}$
1	2	$1\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}+2\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}=\begin{pmatrix}0 & 2\\ 0 & 0\end{pmatrix}$
2	2	$2\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}+2\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}=\begin{pmatrix}1 & 2\\ 0 & 1\end{pmatrix}.$

Elde edilen C matris kodunun elemanları;

$$C=\left\{\begin{pmatrix}0 & 0\\ 0 & 0\end{pmatrix},\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix},\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix},\begin{pmatrix}2 & 1\\ 0 & 2\end{pmatrix},\begin{pmatrix}2 & 0\\ 0 & 2\end{pmatrix},\begin{pmatrix}2 & 2\\ 0 & 2\end{pmatrix},\begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix},\begin{pmatrix}1 & 2\\ 0 & 1\end{pmatrix}\right\}$$

şeklindedir.

Herhangi bir C matris kodu, $Mat_{m\times n}(F_q)$ matris uzayının bir alt kümesidir. Lineer olan C matris kodu, $Mat_{m\times n}(F_q)$ matris uzayının F_q-lineer alt uzayıdır. Dolayısıyla $Mat_{m\times n}(F_q)$ matris uzayı ile F_q^{mn} uzayı izomorftur. Bu nedenle $Mat_{m\times n}(F_q)$

uzayındaki her matris, matrisin ilk satırını ardından ikinci satırını ve benzer biçimde diğer satırlarını ardarda yazarak, $1\times mn$ vektörü olarak gösterilebilir. Benzer biçimde F_q^{mn} deki her vektör, n koordinatlı m gruplara ayrılarak $m\times n$ tipindeki matrislerle gösterilebilir [14].

Örnek 4.1.2. Z_2^6 deki $x=100110$ kodsözü, $Mat_{3\times 2}(Z_2)$ matris uzayında $\begin{pmatrix}1&0\\0&1\\1&0\end{pmatrix}$ matrisine karşılık gelir.

Örnek 4.1.3. $Mat_{3\times 4}(Z_2)$ matris uzayından alınan $A=\begin{pmatrix}1&0&1&0\\0&1&1&0\\1&1&0&0\end{pmatrix}$ matrisi, Z_2^{12} de $x=101001101100$ kodsözüne karşılık gelir.

Tanım 4.1.3. [13] $A=\left(a_{ij}\right)_{\substack{1\le i\le m\\1\le j\le n}}$ matrisi için, yatay sendrom $h=(h_1,h_2,...,h_m)$ olmak üzere,

$$h_i=\sum_{l=1}^{n}a_{il},\ 1\le i\le m$$

olarak tanımlanır.

Tanım 4.1.4. [13] $A=\left(a_{ij}\right)_{\substack{1\le i\le m\\1\le j\le n}}$ matrisi için, dikey sendrom $v=(v_1,v_2,...,v_n)$ olmak üzere,

$$v_j=\sum_{l=1}^{m}a_{lj},\ 1\le j\le n$$

olarak tanımlanır.

Örnek 4.1.4.

$$A=\begin{pmatrix}1 & 0 & 1 & 0\\ 0 & 1 & 1 & 0\\ 1 & 1 & 0 & 0\end{pmatrix}_{3\times 4}\in Mat_{3\times 4}\left(Z_2\right)$$

matrisi için,

$$h_1=1+0+1+0=0 \mod 2$$
$$h_2=0+1+1+0=0 \mod 2$$
$$h_3=1+1+0+0=0 \mod 2$$

olmak üzere, yatay sendrom $h=(0,0,0)$ dır. A matrisi için dikey sendrom ise

$$v_1=1+0+1=0 \mod 2$$
$$v_2=0+1+1=0 \mod 2$$
$$v_3=1+1+0=0 \mod 2$$
$$v_4=0+0+0=0 \mod 2$$

olmak üzere, $v=(0,0,0,0)$ dır.

Tanım 4.1.5. Yatay ve dikey sendromlar sıfır olacak şekilde, matrisin satırlarına veya sütunlarına eklenen bitlere parite kontrol bitleri (parity check bits) denir. Matrise eklenen sütun sayısı tek ise bu koda tek parite kontrol sütununa sahip bir matris kodudur denir.

Örnek 4.1.5. Örnek 4.1.4 teki $A\in Mat_{3\times 4}\left(Z_2\right)$ matrisi için parite kontrol bitleri aşağıda altı çizili olarak gösterilmiştir:

$$A=\begin{pmatrix}1 & 0 & 1 & \underline{0}\\ 0 & 1 & 1 & \underline{0}\\ \underline{1} & \underline{1} & \underline{0} & \underline{0}\end{pmatrix}_{3\times 4}.$$

Elias [15] tarafından inşa edilen matris kodlar, ardışık ve rasgele hata kontrol uygulamalarında kullanılır. Bu kodlar, basit bir yapı ve düşük karmaşık uygulamaya sahiptir. Matris kodların en çok bilinen şekillerinden ikisi; satır ve sütun matris kod (row and column array code) ve genelleştirilmiş matris kod (generalised array code) dur [16].

4.2. Satır ve Sütun Matris Kod (Blok Matris Kod)

İki boyutlu matris kodların en basit şekli satır ve sütun matris kodudur. Bu kodlar şekil olarak kare ve dikdörtgen şeklinde olabilir [17],[18]. Bu tip kodlar bilgisayar hafızasında bilgi iletim sistemlerinde yaygın bir şekilde kullanılır.

C_1 satır kodu, $(n_1, k_1, d_1 = 2)$ parametrelerine ve C_2 sütun kodu, $(n_2, k_2, d_2 = 2)$ parametrelerine sahip ise C matris kodu, $(n_1 n_2, k_1 k_2, d_1 d_2 = 4)$ parametrelerine sahiptir [18].

Satır ve sütun matris kodlamasında, $k = k_1 \times k_2$ boyutlu bilgi sembolleri k_1 satır ve k_2 sütun olmak üzere $k_1 \times k_2$ tipindeki matrise yerleştirilir. Satırlar ve sütunlar üzerine tek parite kontrol bit işlemi uygulanır. Bunun sonucunda $n_1 \times n_2$ tipinde bir matris elde edilir. Dolayısıyla C matris kodu için $(n = n_1 n_2, k = k_1 k_2, d = d_1 d_2 = 4)$ parametreleri elde edilir [18].

Örnek 4.2.1. $x = 100110101010$ bilgi vektörünü kodlamak istediğimizi kabul edelim. Bunun için $k_1 = 2$ ve $k_2 = 6$ olacak şekilde $k_1 \times k_2 = 2 \times 6$ tipindeki matrise x bilgi vektörü aşağıdaki şekilde yerleştirilir:

$$A = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}_{2\times 6}.$$

A matrisinin satırlarına ve sütunlarına tek parite kontrol bit işlemi uygulanırsa A matrisi aşağıdaki şekilde $n = n_1 \times n_2 = 3 \times 7$ tipindeki bir matrise dönüşür. Altı çizili semboller parite kontrol bitleridir.

$$B = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 & \underline{1} \\ 1 & 0 & 1 & 0 & 1 & 0 & \underline{1} \\ \underline{0} & \underline{0} & \underline{1} & \underline{1} & \underline{0} & \underline{0} & \underline{0} \end{pmatrix}_{3\times 7} .$$

Dolayısıyla B matrisi 3×7 tipinde bir matris kod olur.

Tanım 4.2.1. [19] Kronecker Çarpım (Tensor Çarpım)

A, $m\times n$ tipinde bir matris ve B, $p\times q$ tipinde bir matris olmak üzere; $A\otimes B$ kronecker çarpımı, $mp\times nq$ tipinde bir matristir. Bu matris aşağıdaki şekilde bulunur:

$$A\otimes B = \begin{pmatrix} a_{11}B & . & . & . & a_{1n}B \\ . & . & & & . \\ . & & . & & . \\ . & & & . & . \\ a_{m1}B & . & . & . & a_{mn}B \end{pmatrix}.$$

Daha açık bir ifadeyle,

$$A\otimes B=\begin{pmatrix} a_{11}b_{11} & a_{11}b_{12} & \cdots & a_{11}b_{1q} & \cdots & a_{1n}b_{11} & a_{1n}b_{12} & \cdots & a_{1n}b_{1q} \\ a_{11}b_{21} & a_{11}b_{22} & \cdots & a_{11}b_{2q} & \cdots & a_{1n}b_{21} & a_{1n}b_{22} & \cdots & a_{1n}b_{2q} \\ \vdots & \vdots & & \vdots & & \vdots & \vdots & & \vdots \\ a_{11}b_{p1} & a_{11}b_{p2} & \cdots & a_{11}b_{pq} & \cdots & a_{1n}b_{p1} & a_{1n}b_{p2} & \cdots & a_{1n}b_{pq} \\ \vdots & \vdots & & \vdots & & \vdots & \vdots & & \vdots \\ a_{m1}b_{11} & a_{m1}b_{12} & \cdots & a_{m1}b_{1q} & \cdots & a_{mn}b_{11} & a_{mn}b_{12} & \cdots & a_{mn}b_{1q} \\ a_{m1}b_{21} & a_{m1}b_{22} & \cdots & a_{m1}b_{2q} & \cdots & a_{mn}b_{21} & a_{mn}b_{22} & \cdots & a_{mn}b_{2q} \\ \vdots & \vdots & & \vdots & & \vdots & \vdots & & \vdots \\ a_{m1}b_{p1} & a_{m1}b_{p2} & \cdots & a_{m1}b_{pq} & \cdots & a_{mn}b_{p1} & a_{mn}b_{p2} & \cdots & a_{mn}b_{pq} \end{pmatrix}$$

olarak yazılır.

Örnek 4.2.2.

$$A=\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \text{ ve } B=\begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix}$$

olmak üzere, kronecker çarpım

$$A\otimes B=\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}\otimes\begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix}=\begin{pmatrix} 1.0 & 1.5 & 2.0 & 2.5 \\ 1.6 & 1.7 & 2.6 & 2.7 \\ 3.0 & 3.5 & 4.0 & 4.5 \\ 3.6 & 3.7 & 4.6 & 4.7 \end{pmatrix}=\begin{pmatrix} 0 & 5 & 0 & 10 \\ 6 & 7 & 12 & 14 \\ 0 & 15 & 0 & 20 \\ 18 & 21 & 24 & 28 \end{pmatrix}$$

olur.

Örnek 4.2.3. C_1, tek parite kontrollü $(n_1=3,k_1=2,d_1=2)$ parametrelerine sahip bir satır kodu ve C_2 de, tek parite kontrollü $(n_2=3,k_2=2,d_2=2)$ parametrelerine sahip bir sütun kodu olsun.

$(n_1 = 3, k_1 = 2, d_1 = 2)$ satır kodunun tek parite kontrollü üreteç matrisi G_1 aşağıdaki şekilde alınsın:

$$G_1 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}.$$

$(n_2 = 3, k_2 = 2, d_2 = 2)$ sütun kodunun tek parite kontrollü üreteç matrisi G_2 de aşağıdaki şekilde alınsın:

$$G_2 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}.$$

Bu taktirde, C satır ve sütun matris kodunun üreteç matrisi olan G, G_1 ve G_2 üreteç matrislerinin kronecker çarpımı olur. Yani,

$$G = G_1 \otimes G_2 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1.G_2 & 0.G_2 & 1.G_2 \\ 0.G_2 & 1.G_2 & 1.G_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{pmatrix}$$

olur.

$x = 0001$ bilgi vektörü ve y kodsöz olmak üzere; $x = 0001$ bilgi vektörü G üreteç matrisi ile kodlanırsa,

$$y = xG = (0001) \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{pmatrix} = (000011011)$$

olarak bulunur. $y = 000011011$ kodsözü, matris kod olarak aşağıdaki şekilde yazılır. Altı çizili olan semboller parite kontrol sembollerini göstermektedir.

$$y=\begin{pmatrix}0 & 0 & \underline{0}\\ 0 & 1 & \underline{1}\\ \underline{0} & \underline{1} & \underline{1}\end{pmatrix}=(000011011)$$

ise y kodsözü satır satır iletilir.

4.3. Genelleştirilmiş Matris Kod

Tanım 4.3.1. [20] Genelleştirilmiş matris kod, satır ve sütun altkodlarının farklı sayıda bilgi ve parite kontrol sembollerine sahip olduğu bir matris koddur.

(n_0,k_0,d_0) genelleştirilmiş matris kodunu tasarlamak için aşağıdaki adımlar izlenir [8], [9], [10] :

$$\begin{bmatrix}(n_1,k_1,d_1)\\ (n_1,k_1,d_1)\\ .\\ .\\ .\\ (n_1,k_1,d_1)\end{bmatrix}_{n_2\times n_1} \qquad \begin{bmatrix}0 & . & . & . & 0 & (n_1-k_1,k')\\ 0 & . & . & . & 0 & (n_1-k_1,k')\\ . & & & & . & .\\ . & & & & . & .\\ . & & & & . & .\\ 0 & . & . & . & 0 & (n_1-k_1,k')\end{bmatrix}_{n_2\times n_1} \qquad \begin{bmatrix}0 & 0 & . & . & . & 0\\ 0 & 0 & . & . & . & 0\\ . & . & . & . & . & .\\ . & . & . & . & . & .\\ . & . & . & . & . & .\\ & & B=(n_1,1,n_1) & & & \end{bmatrix}_{n_2\times n_1}$$

(A) C_1 kodu (B) C_2 kodu (C) C_3 kodu

Şekil 4.1 Genelleştirilmiş Matris Kod İnşası

1. n_0 asal sayı ise $n=n_0+1$; n_0 asal sayı değilse $n=n_0$ alınır.

2. $R_1=(n_1,k_1,d_1)$ satır kodu ve $n=n_2\times n_1$ tipinde tek parite kontrol sütun koduna sahip bir C_1 lineer kodu tasarlanır (Şekil 4.1 (A)). Burada, $d_1=\left[\!\left[\frac{d_0}{2}\right]\!\right]$ olmak üzere; $[\![.]\!]$ tamdeğer fonksiyonunu göstermektedir.

3. $n = n_2 \times n_1$ tipinde ek bir C_2 lineer kodu tasarlanır. C_2 lineer kodu tasarlanırken, ilk satırı k_1 tane sıfır ve $\left(n_1 - k_1, k'\right)$ koddan oluşur. Diğer $\left(n_2 - 1\right)$ tane satır ise ilk satırın tekrarıdır (Şekil 4.1 (B)).

4. C_2 lineer kodu tasarlandıktan sonra geriye kullanılacak bilgi sembolü kaldıysa, $n = n_2 \times n_1$ tipinde ek bir C_3 lineer kodu tasarlanır. C_3 lineer kodu tasarlanırken , ilk $\left(n_2 - 1\right)$ tane satırın hepsi sıfır ve son satır $B = \left(n_1, 1, n_1\right)$ olacak şekilde tekrarlı satır kodundan oluşur (Şekil 4.1 (C)).

5. $C = \left(n_0, k_0, d_0\right)$ genelleştirilmiş matris kodu, tasarlanan C_1 , C_2 ve C_3 lineer kodlarının modül 2 işlemine göre toplanması sonucu elde edilir. Yani,

$$C = C_1 \oplus C_2 \oplus C_3 \pmod 2$$

bulunur.

Örnek 4.3.1. [20] $C = \left(16, 11, 4\right)$ genelleştirilmiş matris kodu aşağıdaki şekilde inşa edilir:

$$\left(n_0, k_0, d_0\right) = \left(16, 11, 4\right) \Rightarrow n_0 = 16,\ k_0 = 11,\ d_0 = 4$$

tür.

1. $n_0 = 16$ asal sayı olmadığından $n = n_0 = 16$ alınır.

2. $d_1 = \left[\!\left[\frac{d_0}{2} \right]\!\right]$ olduğundan $d_1 = \left[\!\left[\frac{d_0}{2} \right]\!\right] = \left[\!\left[\frac{4}{2} \right]\!\right] = 2 \Rightarrow d_1 = 2$ elde edilir. $n = 16$ olduğundan, $n = n_2 \times n_1$ olacak şekilde 4×4 tipinde tek parite kontrol sütun koduna ve $R_1 = \left(n_1, k_1, d_1\right) = \left(4, 3, 2\right)$ satır koduna sahip C_1 lineer kodu alınabilir.

Gönderilecek bilgi vektörü $(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11})$ olduğundan C_1 lineer kodunda en fazla bu bilgi sembollerinden 9 tanesi kullanılabilir. Yani;

$$C_1 = \begin{pmatrix} x_1 & x_2 & x_3 & p_1 \\ x_4 & x_5 & x_6 & p_2 \\ x_7 & x_8 & x_9 & p_3 \\ c_1 & c_2 & c_3 & p_4 \end{pmatrix}$$

şeklindedir.

3. $k_1 = 3$ ve $n_1 = 4$ olduğundan, tasarlanılacak 4×4 tipindeki C_2 lineer kodunun ilk satırı $k_1 = 3$ tane sıfır ve $(n_1 - k_1, k') = (4-3,1) = (1,1)$ koddan oluşur. Diğer $(n_2 - 1) = (4-1) = 3$ tane satır ilk satırın tekrarıdır. Dolayısıyla C_2 lineer kodu aşağıdaki şekilde tasarlanır:

$$C_2 = \begin{pmatrix} 0 & 0 & 0 & x_{10} \\ 0 & 0 & 0 & x_{10} \\ 0 & 0 & 0 & x_{10} \\ 0 & 0 & 0 & x_{10} \end{pmatrix}.$$

4. x_{11} bilgi sembolü C_1 ve C_2 lineer kodlarında yer almadığından ek bir 4×4 tipinde C_3 lineer koduna ihtiyaç vardır. Bu lineer kod, ilk $(n_2 - 1) = (4-1) = 3$ tane satırı sıfır ve son satırı $B = (n_1, 1, n_1) = (4,1,4)$ şeklinde tekrarlı satır kodu olacak şekilde tasarlanır. Yani,

$$C_3 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ x_{11} & x_{11} & x_{11} & x_{11} \end{pmatrix}$$

şeklindedir.

5. $C=(16,11,4)$ genelleştirilmiş matris kodunu elde etmek için tasarlanan C_1, C_2 ve C_3 lineer kodları modül 2 işlemine göre toplanarak aşağıdaki şekilde hesaplanır:

$$C=C_1\oplus C_2\oplus C_3=\begin{pmatrix}x_1 & x_2 & x_3 & p_1\\ x_4 & x_5 & x_6 & p_2\\ x_7 & x_8 & x_9 & p_3\\ c_1 & c_2 & c_3 & p_4\end{pmatrix}\oplus\begin{pmatrix}0 & 0 & 0 & x_{10}\\ 0 & 0 & 0 & x_{10}\\ 0 & 0 & 0 & x_{10}\\ 0 & 0 & 0 & x_{10}\end{pmatrix}\oplus\begin{pmatrix}0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\\ x_{11} & x_{11} & x_{11} & x_{11}\end{pmatrix}$$

$$=\begin{pmatrix}x_1 & x_2 & x_3 & (p_1\oplus x_{10})\\ x_4 & x_5 & x_6 & (p_2\oplus x_{10})\\ x_7 & x_8 & x_9 & (p_3\oplus x_{10})\\ (c_1\oplus x_{11}) & (c_2\oplus x_{11}) & (c_3\oplus x_{11}) & (p_4\oplus x_{10}\oplus x_{11})\end{pmatrix}.$$

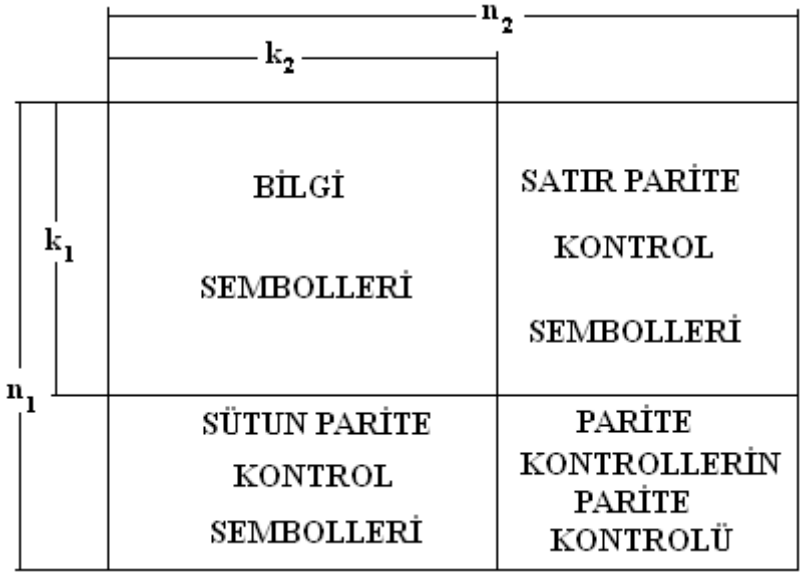
Bu işlemler sonucu $C=(16,11,4)$ genelleştirilmiş matris kodu elde edilir. Bu matris kod aynı zamanda aşağıdaki şekilde de yazılabilir:

$$C=(x_1,x_2,x_3,p_1\oplus x_{10},x_4,x_5,x_6,p_2\oplus x_{10},x_7,x_8,x_9,p_3\oplus x_{10},c_1\oplus x_{11},c_2\oplus x_{11},c_3\oplus x_{11},p_4\oplus x_{10}\oplus x_{11}).$$

4.4. Çarpım Kod (Product Code)

Kodları birleştirerek yeni kodlar elde etmek için birçok metod vardır. Bu metodlardan biri de, iki kodu birleştirmek için iki kodun kronecker çarpımını oluşturmaktır. Kronecker çarpım ile iki kodun çarpımı sonucu yeni bir kod olan çarpım kodu elde edilir [23].

Tanım 4.4.1. [23] C_1 ve C_2 sırasıyla, sonlu cisim üzerinde $[n_1,k_1,d_1]$ ve $[n_2,k_2,d_2]$ lineer kod olsunlar. $C_1\otimes C_2$ kronecker çarpımı, $[n_1n_2,k_1k_2,d_1d_2]$ çarpım kodudur ve $C_1\otimes C_2$ çarpım kodunun kodsözleri aşağıdaki gibi inşa edilen $n_1\times n_2$ tipindeki matrislerden oluşur (Şekil 4.2).



Şekil 4.2 $C_1 \otimes C_2$ çarpım kodunun kodsözleri

Şekil 4.2'de sol üst bölümünde $k_1 k_2$ tane bilgi sembolü vardır. İlk k_2 tane sütun C_1 lineer kodunun kodsözleri ve ilk k_1 tane satır ise C_2 lineer kodunun kodsözlerinden oluşur. Dolayısıyla $C_1 \otimes C_2$ çarpım kodunun sütunları, C_1 lineer kodunun kodsözleri ve $C_1 \otimes C_2$ çarpım kodunun satırları, C_2 lineer kodunun kodsözleridir [23], [24].

Önerme 4.4.1. [23] C_1 ve C_2 lineer kodlarının üreteç matrisleri sırasıyla G_1 ve G_2 olsun. Bu taktirde $G_1 \otimes G_2$ kronecker çarpımı, $C_1 \otimes C_2$ çarpım kodunun üreteç matrisidir.

Çarpım kodlar, 1954 yılında Elias [15] tarafından inşa edilen iki veya daha fazla kodun birleşmesi sonucu elde edilen kodlardır. Şekil 4.2'den de görüleceği gibi çarpım kod yapısı oldukça kolaydır. Buna bağlı olarak, iki yada daha fazla kısa blok kodlar kullanılarak çok uzun blok kodlar elde etmek için verimli bir kod yapısı vardır. Aynı zamanda küçük minimum uzaklıklara sahip olan kısa blok kodları birleştirerek büyük minimum uzaklığa sahip uzun blok kodlar elde edilir [24].

Matris kodlar, çarpım kodlarının genelleştirilmiş halidir ve blok matris kodlarının üst sınıfı olarak bilinir [13].

Önerme 4.4.2. [25] C_1 ve C_2 lineer kodlarının minimum uzaklıkları sırasıyla d_1 ve d_2 , hata düzeltme kabiliyetleri sırasıyla t_1 ve t_2 ise $C_1 \otimes C_2$ çarpım kodunun minimum uzaklığı $d_1 d_2$ ve hata düzeltme kabiliyeti $2t_1t_2 + t_1 + t_2$ dir.

Örnek 4.4.1. C_1 ve C_2 lineer kodlarının minimum uzaklıkları $d_1 = d_2 = 3$ ise hata düzeltme kabiliyetleri $t_1 = t_2 = 1$ olur. $C_1 \otimes C_2$ çarpım kodunun minimum uzaklığı $d_1 d_2 = 3.3 = 9$ ve $2t_1t_2 + t_1 + t_2 = 2.1.1 + 1 + 1 = 4$ hata düzeltme kabiliyetine sahiptir.

Önerme 4.4.2 den görüleceği gibi, kısa uzunluklu lineer kodlardan elde edilecek uzun lineer kodların minimum uzaklığı tekrar bir hesaplama yapmadan kolayca elde edilir. Aksi taktirde elde edilecek çarpım kodunun minimum uzaklığını hesaplamak için uğraş gerekir. Bu da uzun blok kodlarına sahip veya çok sayıda blok koda sahip olan çarpım kod için çok fazla iş gücü gerektirir.

Örnek 4.4.2. $C_1 = \{000,101,011,110\}$ olacak şekilde $[3,2,2]$ lineer kod ve $C_2 = \{000,101,011,110\}$ olacak şekilde $[3,2,2]$ lineer kod olsunlar. C_1 ve C_2 lineer kodlarının üreteç matrisleri sırasıyla aşağıdaki gibi olsun:

$$G_1 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \text{ ve } G_2 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}.$$

C_1 ve C_2 lineer kodlarının üreteç matrisleri G_1 ve G_2 olduğundan $C_1 \otimes C_2$ çarpım kodunun üreteç matrisi $G_1 \otimes G_2$ dir. Çarpım kodunun üreteç matrisi G , kronecker çarpım kullanılarak şu şekilde elde edilir:

$$G = G_1 \otimes G_2 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1.G_2 & 0.G_2 & 1.G_2 \\ 0.G_2 & 1.G_2 & 1.G_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

Dolayısıyla $C_1 \otimes C_2$ çarpım kodu, $[3,2,2] \otimes [3,2,2] = [9,4,4]$ lineer kodudur.

$C_1 \otimes C_2$ çarpım kodunun kodsözleri $n_1 = 3$ ve $n_2 = 3$ olduğundan Şekil 4.2'ye göre $n_1 \times n_2 = 3 \times 3$ tipinde olacaktır. Bu taktirde, G üreteç matrisinin satırları matris kod yardımıyla aşağıdaki şekilde elde edilir.

G üreteç matrisinin satırları G_i, $i = 1,2,3,4$ olmak üzere;

$$G_1 = (101000101) \rightarrow \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} = P_1 , \qquad G_3 = (000011011) \rightarrow \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix} = P_3$$

$$G_2 = (011000011) \rightarrow \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix} = P_2 , \qquad G_4 = (000011011) \rightarrow \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix} = P_4$$

P_i , $i = 1,2,3,4$ matris kodları $C_1 \otimes C_2$ çarpım kodunun kodsözleridir. $C_1 \otimes C_2$ çarpım kodunun kodsöz sayısı $2^4 = 16$ dır. Diğer 12 tane kodsöz, P_i, $i = 1,2,3,4$ matris kodlarının lineer birleşimidir. $C_1 \otimes C_2$ çarpım kodunun kodsözleri kümesi C ise C kodu şu şekilde elde edilir:

$$C = \langle P_1, P_2, P_3, P_4 \rangle = \left\langle \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix} \right\rangle.$$

Bu taktirde, P_i , $i=1,2,...,16$ matris kodları çarpım kodunun kodsözleri olmak üzere;

$$P_1=\begin{pmatrix}1&0&1\\0&0&0\\1&0&1\end{pmatrix} \qquad P_9=\begin{pmatrix}0&1&1\\1&1&0\\1&0&1\end{pmatrix}$$

$$P_2=\begin{pmatrix}0&1&1\\0&0&0\\0&1&1\end{pmatrix} \qquad P_{10}=\begin{pmatrix}1&0&1\\0&1&1\\1&1&0\end{pmatrix}$$

$$P_3=\begin{pmatrix}0&0&0\\1&0&1\\1&0&1\end{pmatrix} \qquad P_{11}=\begin{pmatrix}1&0&1\\1&0&1\\0&0&0\end{pmatrix}$$

$$P_4=\begin{pmatrix}0&0&0\\0&1&1\\0&1&1\end{pmatrix} \qquad P_{12}=\begin{pmatrix}1&0&1\\1&1&0\\0&1&1\end{pmatrix}$$

$$P_5=\begin{pmatrix}0&0&0\\0&0&0\\0&0&0\end{pmatrix} \qquad P_{13}=\begin{pmatrix}1&1&0\\0&0&0\\1&1&0\end{pmatrix}$$

$$P_6=\begin{pmatrix}0&0&0\\1&1&0\\1&1&0\end{pmatrix} \qquad P_{14}=\begin{pmatrix}1&1&0\\0&1&1\\1&0&1\end{pmatrix}$$

$$P_7=\begin{pmatrix}0&1&1\\0&1&1\\0&0&0\end{pmatrix} \qquad P_{15}=\begin{pmatrix}1&1&0\\0&1&1\\1&0&1\end{pmatrix}$$

$$P_8=\begin{pmatrix}0&1&1\\1&0&1\\1&1&0\end{pmatrix} \qquad P_{16}=\begin{pmatrix}1&1&0\\1&1&0\\0&0&0\end{pmatrix}$$

şeklindedir. $C=\left\{P_i \middle| i=1,2,...,16\right\}$ olacak şekilde çarpım kodunun kodsözler kümesi bulunur. P_i , $i=1,2,...,16$ kodsözlerinin herbirinin sol üst kısmında $k_1=2$ ve $k_2=2$ olduğundan $k_1\times k_2=2\times 2$ tipindeki matris bilgi sembollerini içermektedir. Aynı zamanda herbir kodsözün satırları, C_1 lineer kodunun kodsözleri ve herbir kodsözün sütunları, C_2 lineer kodunun kodsözleridir. Örneğin; P_1 kodsözü için altı

çizili semboller bilgi sembollerini, üstü çizili semboller ise parite kontrol sembollerini göstermektedir.

$$P_1 = \begin{pmatrix} \bar{1} & \bar{0} & \underline{1} \\ \bar{0} & \bar{0} & \underline{0} \\ \underline{1} & \underline{0} & \underline{1} \end{pmatrix} \begin{matrix} \rightarrow ç_1 \\ \rightarrow ç_1 \\ \rightarrow ç_1 \end{matrix}$$

$$\begin{matrix} \downarrow & \downarrow & \downarrow \\ ç_2 & ç_2 & ç_2 \end{matrix}$$

BÖLÜM 5. MATRİS KOD İLE McELİECE ŞİFRELEME SİSTEMİ

5.1. Simetrik (Symmetric) ve Açık (Public) Anahtarlı Şifreleme Sistemleri

Şifreleme ve deşifreleme metodları iki kategoriye ayrılır: Bunlar simetrik ve açık anahtarlı sistemlerdir.

Simetrik anahtarlı şifreleme sistemlerinde, şifreleme ve deşifreleme için kullanılan anahtarlar hem mesajı gönderen hem de mesaji alan kişiler tarafından bilinir. Örneğin, şifreleme anahtarı kişiler tarafından paylaşılır ve deşifreleme anahtarı da kolaylıkla şifreleme anahtarı aracılığıyla hesaplanır. Birçok durumda, şifreleme ve deşifreleme anahtarı aynıdır. 1970 öncesinde kullanılan klasik şifreleme sistemlerinin hepsi simetrik şifreleme sistemleri iken, şu anda kullanılan DES (Data Encryption Standard) ve Rijndael (AES) şifreleme sistemleri de simetrik şifreleme sistemleridir.

Açık anahtarlı şifreleme sistemi fikri ilk önce Diffie – Hellman [26] tarafından 1970'lerde kullanılmıştır. Bu fikir sayesinde şifrelemede büyük adımlar atılmıştır. Açık anahtarlı şifreleme sistemlerinde, mesajı gönderen ve alan kişiler biraraya gelmeden birbirinden uzak olacak şekilde güvenli bir şekilde iletişim kurmak ister. Biraraya gelmeden bir anahtar üzerinde anlaşmak zor görünür. Bu durumda mesajı gönderen kişi, açık kanallar üzerinden mesajı alacak kişiye anahtarı gönderemez. Bu problemin çözümü Diffie – Hellman ikilisi tarafından yapılmıştır. Problemin dayandığı çözüm, şifreleme anahtarı mesajı alacak kişi tarafından tasarlanır ve herkesin göreceği şekilde ortama verilir. Herkesin açık anahtarı görmesinden dolayı mesajı gönderecek kişi açık anahtarı alır ve bu anahtar sayesinde mesajı şifreler. Bu sistemde herkesin açık anahtarı bilmesi, mesajı alacak kişilerdeki bilgiyi bilmeden deşifreleme anahtarını bulması oldukça zordur. Yani, mesajı alan kişi anahtarı tasarlarken kullandığı bilgileri bilmeden mesajı çözmek oldukça zordur.

Açık anahtar ile şifreleme yapma fikri, matematiksel olmayan bir yolla şu şekilde düşünülebilir:

Mesajı gönderen ve alan kişi sırasıyla Ali ve Betül olsun. Bu kişilerin sadece kendilerinin bildiği özel anahtarlarının olduğu kabul edilsin. Gönderilecek bilginin kutu içerisinde iletildiği düşünülsün. Betül, Ali'ye kutuyu gönderir. Kutu açık anahtar olarak düşünülebilir. Ali kutu içerisine mesajı koyar ve kendine özel anahtar ile kutuyu şifreler. Kutuyu tekrar Betül'e gönderir fakat Betül Ali'nin anahtarını bilmediğinden kutuyu açamaz. Gelen kutuya kendi özel anahtarını ekleyerek şifreler ve tekrar Ali'ye gönderir. Ali kendine özel anahtarı kutunun üzerinden alır ve Betül'e kutuyu tekrar gönderir. Betül de artık kutunun üzerinde sadece kendi özel anahtarı kaldığından kutuyu açabilir ve mesajı alır. Açık anahtarlı şifreleme sistemlerinde de bu fikir matematiksel olarak uygulanarak kullanılmıştır.

Açık anahtarlı şifreleme sistemleri, simetrik anahtarlı şifreleme sistemlerine göre oldukça güvenlidir. Çünkü simetrik şifreleme sistemlerinde anahtar iletiminde üçüncü bir kişi anahtara ulaşabilir ama açık anahtarlı şifreleme sistemlerinde anahtarı bilmek şifreyi çözmek için yeterli değildir. Genelde açık anahtarlı şifreleme sistemlerinde, bilginin iletilmesi için ihtiyaç duyulan işlem miktarı oldukça büyüktür. Örneğin; DES veya Rijndael simetrik şifreleme sistemlerinde bu işlem miktarı daha azdır. Bu yönüyle simetrik şifreleme sistemleri daha avantajlıdır [3].

Açık anahtarlı şifreleme sistemlerinden biri, cebirsel kodlama teorisine dayanan McEliece [27] şifreleme sistemidir. McEliece şifreleme sistemi, kodların bir sınıfı olan Goppa kodlarını [28] kullanmaktadır. Goppa kodların kullanılmasının en önemli sebebi; genel lineer kodlar hızlı ve verimli bir dekodlamaya sahip değil iken, Goppa kodlar dekodlama için hızlı ve verimli bir dekodlama algoritmasına sahiptir. Aynı zamanda bu kodları üretmek kolaydır ve aynı parametrelere sahip birbirine denk olmayan çok sayıda Goppa kodları vardır. Aynı parametrelere sahip denk olmayan bir çok Goppa kod olması dekodlama işlemini zorlaştırarak dışarıdan gelecek saldırılara karşı dekodlamayı verimli hale getirir. McEliece şifreleme sistemi çok hızlı bilgi transferine izin vererek oldukça güvenli bir iletişim sağlar. Bu yüzden bu tip bir şifreleme sistemi çok kullanıcılı iletişim ağları için ideal bir şifreleme

sistemidir. Örneğin; McEliece şifreleme sistemi, NASA (National Aeronautics and Space Administration) tarafından uzaydan elde edinilmiş bilgilerin yayılımı için kullanılmıştır.

McEliece şifreleme sisteminde, 50 hata düzeltebilen 1000 uzunluklu bir kod için bilgisayar iletişimlerinde şifreleme ve deşifrelemenin pratik olması için çok büyük miktarda hesaplama gerekir. Bu hesaplamanın daha düşük olması için Rao tarafından McEliece şifreleme sistemi, simetrik bir şifreleme sistemi olarak alınır [32]. Açık anahtarlı McElice şifrelemesine göre tek farkı, anahtarın gizli olmasıdır. Bu sayede daha basit hata düzelten kodlarla daha iyi güvenlik sağlanır [33].

McEliece şifreleme sisteminin algoritması aşağıdaki şekildedir:

5.2. McEliece Şifreleme Sistemi Algoritması

m bir pozitif tamsayı olmak üzere 2^m elemanlı sonlu cisim üzerinde t inci dereceden her indirgenemez polinoma karşılık gelen $n = 2^m$ uzunluklu, $k \geq n - mt$ boyutlu ve en fazla t hata düzeltme kabiliyetine sahip ikili bir Goppa kod vardır [5].

$n = 2^m$, $k \geq n - mt$ ve $d = 2t + 1$ parametrelerine sahip $[n, k, d]$ C Goppa kodunun üreteç matrisi G olsun. S, $k \times k$ tipinde terslenebilir bir matris ve P, $n \times n$ tipinde permütasyon matris olmak üzere şifreleme ve deşifreleme algoritması şu şekildedir:

5.2.1. Şifreleme Algoritması

Mesajı gönderen ve alan kişiler sırasıyla Ali ve Betül olsun. S, G ve P matrisleri Betül adlı kişinin gizli anahtarlarıdır. Yani, gizli anahtar matrisler sadece Betül tarafından bilinir. Betül anahtar matrisleri kullanarak,

$$G' = SGP$$

olacak biçimde $G^{'}$ matrisini hesaplar. $G^{'}$ matrisi açık anahtardır. Yani, bu açık anahtara herkes erişebilir. Herkes bu matrise erişebildiğinden Ali adlı kişi açık anahtar matrisini mesajı şifrelemek için kullanır. Bu şifrelemeyi yaparken aşağıdaki şifreleme fonksiyonunu kullanır [27]:

$K=\left(G,S,P,G^{'}\right)$ anahtarı için $e_k\left(x,e\right)$ yi

$$e_k\left(x,e\right)=y=xG^{'}+e$$

olarak hesaplar. Burada, x mesaj, y şifrelimetin ve e , t ağırlığına sahip rasgele seçilen bir hata vektörüdür. Ali hesapladığı mesajı gönderir.

5.2.2. Deşifreleme Algoritması

Betül, Ali adlı kişiden gelen y şifrelimetninden x mesajını elde etmek için aşağıdaki işlemleri uygular [27]:

1. Gizli anahtarlarından biri olan P anahtar matrisini kullanarak $y_1=yP^{-1}$ eşitliğini hesaplar. Buradan,

$$y_1=yP^{-1}=\left(xG^{'}+e\right)P^{-1}=\left(xSGP+e\right)P^{-1}=\left(xSGPP^{-1}+eP^{-1}\right)=xSG+e_1$$

bulunur.

2. $x_1\in C$ olmak üzere, $y_1=x_1+e_1$ eşitliğini elde ederek y_1 ifadesini dekodlar.

3. Gizli anahtarlarından biri olan G anahtar matrisini kullanarak $x_0G=x_1$ olacak şekilde x_0 değerini hesaplar.

4. Diğer gizli anahtarlardan biri olan S anahtar matrisini kullanarak $x=x_0S^{-1}$ hesaplar ve x mesajını elde edilir. Böylece şifrelimetinden açıkmetin elde edilir.

Dekodlamayı zorlaştırmak için kodun minimum uzaklığı, mümkün olabilecek en büyük değer seçilmelidir. McElice R.J. dekodlamayı zorlaştırmak için makalesinde uygun parametreler olarak $[1024,524,101]$ parametrelerini önerir [27]. Burada $n=1024$, $k=524$ ve $d=101$ parametrelerine sahip olan kod, $t=50$ hata düzeltme kabiliyetine sahiptir. $[1024,524,101]$ parametrelerini sahip olan kodu dekodlamak için gerekli olan iş miktarı $2^{80.7}$ dir [31]. Carlisle M. Adams ve Henk Meijer, McEliece şifrelemesinde dekodlamayı daha zor hale getirmek için $[1024,654,75]$ parametrelerini önerir [31]. Bu parametrelere sahip olan kod, $t=37$ hata düzeltme kabiliyetine sahiptir. Ayrıca bu parametrelere sahip olan kodun dekodlanması için gerekli olan iş miktarı $2^{84.1}$ dir [31]. Dolayısıyla, McElice R.J.'nin parametrelerine göre Carlisle ve Henk'in önerdiği parametreler güvenlik için daha iyidir fakat bu güvenliği sağlamak için yapılan işlemler, kodun hata düzeltme kabiliyetini azaltır.

Örnek 5.2.1. C Goppa kodu $[23,12,7]$ lineer kodu olsun. C Goppa kodunun üreteç matrisi G olsun. Betül adlı kişinin gizli anahtar matrisleri olan S , P ve G aşağıdaki şekilde alınsın: (Goppa kod elde etmek zor olduğundan $[23,12,7]$ parametrelerine sahip Goppa kodun üreteç matrisi [29] nolu kaynaktan alınmıştır)

$$S=\begin{pmatrix}
1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&1\\
0&0&0&0&0&0&0&0&0&0&1&0
\end{pmatrix}$$

$$P = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}$$

$$G = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0
\end{pmatrix}$$

Mesajı alacak kişi olan Betül, gizli anahtarları ile $G' = SGP$ olacak şekilde G' açık anahtarını hesaplar. G' açık anahtarı,

$$G' = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0
\end{pmatrix}$$

olarak bulunur. G' açık anahtar olduğundan mesajı gönderecek kişi olan Ali, G' açık anahtarına erişebilir. Ali bu açık anahtar ile mesajını iletir. Ali'nin ileteceği mesaj $x = (1,0,0,1,0,0,0,1,1,0,1,1)$ alınsın. Ali'nin bu mesajı şifrelemesi için e hata vektörüne ihtiyacı vardır. e hata vektörü belirlenirken kodun minimum uzaklığına ihtiyaç vardır. Kodun minimum uzaklığı $d = 7$ olduğundan bu kod $d = 2t + 1 = 7 \Rightarrow t = 3$ eşitliğinden 3 hata düzeltme kabiliyetine sahiptir. Dolayısıyla e hata vektörü 3 ağırlığına sahip rasgele seçilen bir vektördür. Bu taktirde,

$$e = (1,0,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)$$

olacak şekilde alınsın. Ali x mesajını şifreleme fonksiyonu $y = xG' + e$ eşitliğini kullanarak x mesajını şifreler ve y şifrelimetnini aşağıdaki gibi elde eder:

$$y = (0,0,0,0,1,1,1,0,1,1,1,1,1,0,0,0,1,1,0,1,0,1,1).$$

y şifrelimetnini alan Betül, x mesajını elde etmek için şu işlemleri yapar:

1. P gizli anahtarını kullanarak $y_1 = yP^{-1}$ eşitliğini hesaplar. (P permütasyon matris olduğundan, $P^{-1} = P^T$ dir [30].) Buradan;

$$y_1 = (0,0,0,1,0,1,1,1,0,1,1,1,1,0,0,0,1,1,0,1,0,1,1)$$

olarak bulunur.

2. $x_1 \in C$ ve $e_1 = eP^{-1}$ olmak üzere, $y_1 = x_1 + e_1$ olacak şekilde y_1 ifadesini dekodlar. Buradan e_1 ve x_1 aşağıdaki gibi elde edilir:

$$e_1 = (1,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)$$
$$x_1 = (1,0,0,1,0,0,0,1,0,1,1,1,1,0,0,0,1,1,0,1,0,1,1) \quad .$$

3. Gizli anahtarlarından biri olan G anahtar matrisini kullanarak $x_0 G = x_1$ olacak şekilde x_0 ifadesini hesaplar. Buradan x_0 değeri,

$$x_0 = (1,0,0,1,0,0,0,1,0,1,1,1)$$

olarak bulunur.

4. Diğer gizli anahtarlardan biri olan S anahtar matrisini kullanarak $x = x_0 S^{-1}$ değerini hesaplar ve x mesajını elde eder.

$$S^{-1}=\begin{pmatrix}1&0&0&0&0&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0&0&0&0&0\\0&0&0&1&0&0&0&0&0&0&0&0\\0&0&0&0&0&1&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&1&0&0\\0&0&0&0&0&0&0&1&0&0&0&0\\0&0&0&0&0&0&1&0&0&0&0&0\\0&0&0&0&0&0&0&0&1&0&0&0\\0&0&0&0&0&0&0&0&0&0&0&1\\0&0&0&0&0&0&0&0&0&0&1&0\end{pmatrix}$$

olduğundan,

$$x_0S^{-1}=(1,0,0,1,0,0,0,1,0,1,1,1)\begin{pmatrix}1&0&0&0&0&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0&0&0&0&0\\0&0&0&1&0&0&0&0&0&0&0&0\\0&0&0&0&0&1&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&1&0&0\\0&0&0&0&0&0&0&1&0&0&0&0\\0&0&0&0&0&0&1&0&0&0&0&0\\0&0&0&0&0&0&0&0&1&0&0&0\\0&0&0&0&0&0&0&0&0&0&0&1\\0&0&0&0&0&0&0&0&0&0&1&0\end{pmatrix}$$

$$=(1,0,0,1,0,0,0,1,1,0,1,1)=x$$

mesajını elde eder. Böylece Betül mesajı çözer.

5.3. Matris Kodların McEliece Şifreleme Sistemine Uygulanması

Bu kısımda, bölüm 4 te ele alınan matris kodların McEliece şifrelemesine uygulanması ele alınacaktır. Bu sayede, matris kodlarla McEliece şifrelemesinde dekodlama yaparken daha fazla hata tespiti sağlanabilecektir.

Örnek 5.3.1. McEliece şifreleme sisteminde kullanılan üreteç matrisi, $C=(8,4,4)$ genelleştirilmiş matris kod olarak alınsın. Şifrelemeyi yapmadan önce $C=(8,4,4)$ genelleştirilmiş matris kodunu inşa edelim. $C=(8,4,4)$ genelleştirilmiş matris kodu aşağıdaki şekilde inşa edilir:

$$(n_0,k_0,d_0)=(8,4,4)\Rightarrow n_0=8,\ k_0=4,\ d_0=4$$

tür.

1. $n_0=8$ asal sayı olmadığından $n=n_0=8$ alınır.

2. $d_1=\left[\!\left[\frac{d_0}{2}\right]\!\right]$ olduğundan $d_1=\left[\!\left[\frac{d_0}{2}\right]\!\right]=\left[\!\left[\frac{4}{2}\right]\!\right]=2$ elde edilir. $n=8$ olduğundan, $n=n_2\times n_1$ olacak şekilde 4×2 tipinde tek parite kontrol sütun koduna ve $R_1=(n_1,k_1,d_1)=(2,1,2)$ satır koduna sahip C_1 lineer kodu alınabilir. Gönderilecek bilgi vektörü (x_1,x_2,x_3,x_4) olduğundan C_1 lineer kodunda en fazla bu bilgi sembollerinden 3 tanesi kullanılabilir. Yani;

$$C_1=\begin{pmatrix} x_1 & p_1\\ x_2 & p_2\\ x_3 & p_3\\ p_4 & p_4 \end{pmatrix}$$

şeklindedir.

3. $k_1 = 1$ ve $n_1 = 2$ olduğundan, tasarlanılacak 4×2 tipindeki C_2 lineer kodunun ilk satırı $k_1 = 1$ tane sıfır ve $\left(n_1 - k_1, k'\right) = (2-1,1) = (1,1)$ koddan oluşur. Diğer $(n_2 - 1) = (4-1) = 3$ tane satır ilk satırın tekrarıdır. Dolayısıyla C_2 lineer kodu aşağıdaki şekilde tasarlanır:

$$C_2 = \begin{pmatrix} 0 & x_4 \\ 0 & x_4 \\ 0 & x_4 \\ 0 & x_4 \end{pmatrix}$$

4. x_1, x_2, x_3 ve x_4 bilgi sembollerinin dışında başka kullanılacak bilgi sembolü kalmadığından ek bir lineer kod tasarlanmaya gerek yoktur.

5. $C = (8,4,4)$ genelleştirilmiş matris kodunu elde etmek için tasarlanan C_1 ve C_2 lineer kodları modül 2 işlemine göre toplanarak aşağıdaki şekilde hesaplanır:

$$C = C_1 \oplus C_2 = \begin{pmatrix} x_1 & p_1 \\ x_2 & p_2 \\ x_3 & p_3 \\ p_4 & p_4 \end{pmatrix} \oplus \begin{pmatrix} 0 & x_4 \\ 0 & x_4 \\ 0 & x_4 \\ 0 & x_4 \end{pmatrix} = \begin{pmatrix} x_1 & (p_1 \oplus x_4) \\ x_2 & (p_2 \oplus x_4) \\ x_3 & (p_3 \oplus x_4) \\ p_4 & (p_4 \oplus x_4) \end{pmatrix}.$$

Bu işlemler sonucu $C = (8,4,4)$ genelleştirilmiş matris kodu elde edilir. Bu matris kod olduğundan yatay ve dikey sendromlar sıfıra eşit olmalıdır. Yani,

$$\begin{aligned} & x_1 + (p_1 \oplus x_4) = 0 \\ & x_2 + (p_2 \oplus x_4) = 0 \\ & x_3 + (p_3 \oplus x_4) = 0 \\ & p_4 + (p_4 \oplus x_4) = 0 \\ & x_1 + x_2 + x_3 + p_4 = 0 \\ & (p_1 \oplus x_4) + (p_2 \oplus x_4) + (p_3 \oplus x_4) + (p_4 \oplus x_4) = 0 \end{aligned}$$

eşitlikleri sağlanır. Aynı zamanda matris kod aşağıdaki şekilde de yazılabilir:

$$C = \left(x_1, (p_1 \oplus x_4), x_2, (p_2 \oplus x_4), x_3, (p_3 \oplus x_4), p_4, (p_4 \oplus x_4)\right).$$

$C = (8,4,4)$ genelleştirilmiş matris kodunda gönderilen mesaj $x = (x_1, x_2, x_3, x_4)$ ise buna karşılık gelen kodsöz aşağıdaki şekildedir:

$$c = \left(x_1, (p_1 \oplus x_4), x_2, (p_2 \oplus x_4), x_3, (p_3 \oplus x_4), p_4, (p_4 \oplus x_4)\right).$$

Aynı zamanda bu kodsöz, C matris kod olduğundan aşağıdaki şekilde de yazılabilir:

$$c = \begin{pmatrix} x_1 & (p_1 \oplus x_4) \\ x_2 & (p_2 \oplus x_4) \\ x_3 & (p_3 \oplus x_4) \\ p_4 & (p_4 \oplus x_4) \end{pmatrix}.$$

Bu kodsözde yatay ve dikey sendromlar sıfır olacağından bu kodsöz satır satır iletildiğinde hata meydana gelirse hangi satırda hata olduğu tespit edilebilir. Bu da matris kodunda hata tespitinin ne kadar avantajlı olduğunu gösterir [13].

Bir kodun üreteç matrisi, $c = mG$ eşitliğini sağlayacağından $C = (8,4,4)$ genelleştirilmiş matris kodu için üreteç matrisi aşağıdaki şekildedir:

$$G = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

$C = (8,4,4)$ genelleştirilmiş matris kodunun üreteç matrisi olan G, genelleştirilmiş Hamming kod olarak da bilinir.

McEliece şifrelemesi için gönderilen mesaj, terslenebilir S matrisi ve permütasyon matrisi P sırasıyla aşağıdaki gibi olsun:

$$m=(x_1,x_2,x_3,x_4)=(1010)$$

$$S=\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{pmatrix},\ P=\begin{pmatrix}1&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&1&0&0&0&0\\0&0&0&0&1&0&0&0\\0&0&0&0&0&1&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&1\end{pmatrix}.$$

Mesajı alacak kişi, gizli anahtarları olan S, G ve P ile $G^{'}=SGP$ olacak şekilde $G^{'}$ açık anahtarını hesaplar. $G^{'}$ açık anahtarı,

$$G^{'}=\begin{pmatrix}1&1&0&0&0&0&1&1\\0&0&1&1&0&0&1&1\\0&0&0&0&1&1&1&1\\0&1&0&1&0&1&0&1\end{pmatrix}$$

olur. $G^{'}$ açık anahtar olduğundan mesajı gönderecek kişi, $G^{'}$ açık anahtarına erişebilir. Bu açık anahtar ile mesajını iletir. $t=1$ ağırlıklı hata vektörü aşağıdaki gibi alınsın:

$$e=(01000000).$$

Mesajı gönderecek kişi, $m=(x_1,x_2,x_3,x_4)=(1010)$ mesajını şifreleme fonksiyonu $c=mG^{'}+e$ eşitliğini kullanarak m mesajını şifreler ve c şifrelimetnini diğer bir ifadeyle c kodsözünü elde eder. Burada

$$mG^{'} = (1010)\begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} = (11001100)$$

bulunur. C matris kod olduğundan elde edilen $c_1 = (11001100)$ kodsözü aşağıdaki şekilde yazılabilir:

$$c_1 = \begin{pmatrix} 1 & 1 \\ 0 & 0 \\ 1 & 1 \\ 0 & 0 \end{pmatrix}.$$

Benzer biçimde hata vektörü, $e = \begin{pmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$ şeklinde yazılabilir. Buradan,

$$c = mG^{'} + e = c_1 + e = \begin{pmatrix} 1 & 1 \\ 0 & 0 \\ 1 & 1 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 1 & 1 \\ 0 & 0 \end{pmatrix}$$

bulunur ve c kodsözü satır satır iletilir. Kodsözde dikey ve yatay sendromlar sıfıra eşit olacağından satır satır iletimde ilk satırda hata tespit edilir. Kodsöz bütün satırlarıyla iletidildiğinde ikinci sütunda da hata tespit edilir. Sonuç olarak hata, ilk satır ve ikinci sütunda olduğu tespit edilir. Dolayısıyla c kodsözünün ilk satırı olan (10) yerine (11) gelmelidir. Buradan orjinal kodsözün yada asıl gönderilmek istenen c_1 kodsözünün $\begin{pmatrix} 1 & 1 \\ 0 & 0 \\ 1 & 1 \\ 0 & 0 \end{pmatrix}$ olduğu tespit edilir. Burada e hata vektörü mesajı alacak kişi tarafından bilinmemektedir. Dolayısıyla matris kodlarının en büyük avantajının hata

tespitinde kolaylık sağlaması olduğu görülür. Deşifreleme Örnek 5.2.1'deki gibi yapılır.

Tanım 5.3.1. [34] m ve n parametrelerine sahip doğrudan eşleme $DM_{m,n}(.)$ aşağıdaki gibi tanımlansın:

$i=0,1,...,m-1$ ve $j=0,1,...,n-1$ olmak üzere, $a_{i,j}=v_{in+j+1}$ olacak şekilde $V=(v_1,...,v_{mn})$ vektörünü $A=\begin{pmatrix} a_{0,0} & . & . & . & a_{0,n-1} \\ . & & . & & . \\ a_{m-1,0} & . & . & . & a_{m-1,n-1} \end{pmatrix}_{m\times n}$ matrisine dönüştüren eşlemedir.

Tanım 5.3.2. [34] $i=1,2,...,mn$ olmak üzere, e_i modül q kongrüans sınıfının elemanları olsun. $E=(e_1,...,e_{mn})$ hata vektörü aşağıdaki iki şartı sağlıyorsa ayrılamaz hata vektörü olarak adlandırılır ve bu $BSE_q(m,n)$ ile gösterilir.

1) E hata vektörünün sıfırdan farklı bileşenleri, modül q kongrüans sınıfındaki sıfırdan ve birbirinden farklı elemanlardır.

2) $DM_{m,n}(E)$ doğrudan eşlemesinin satır ve sütunlarında en fazla sıfırdan farklı bir eleman vardır.

Örnek 5.3.3. $q=5$, $m=3$, $n=4$ ve $E=(0,0,1,0,4,0,0,0,0,0,0,3)$ olsun. Bu taktirde, $DM_{m,n}(E)$ doğrudan eşlemesi aşağıdaki şekildedir:

$$DM_{m,n}(E)=\begin{pmatrix} 0 & 0 & 1 & 0 \\ 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix}_{3\times 4}.$$

E hata vektörü için Tanım 5.3.2'deki iki şart sağlandığından E hata vektörü ayrılamaz hata vektörüdür.

Önerme 5.3.1. [34] $BSE_q(m,n)$ ayrılamaz hata vektörünün ağırlığı

$$w = \min\left(q-1, \min(m,n)\right)$$

dir.

Teorem 5.3.1. [34] C_1 , tek parite kontrollü $(n_1 = r+1, k_1 = r, d_1 = 2)$ parametrelerine sahip bir satır kodu ve C_2 de, tek parite kontrollü $(n_2 = s+1, k_2 = s, d_2 = 2)$ parametrelerine sahip bir sütun kodu olsun. C_1 ve C_2 kodları, sonlu cisim üzerinde $(n = (r+1)(s+1), k = rs, d = 4)$ parametrelerine sahip C çarpım kodunun sırasıyla satır ve sütun kodları olmak üzere; C çarpım kodu en fazla w ağırlıklı bir $BSE_q(r+1, s+1)$ ayrılamaz hata vektörünü düzeltebilir.

Örnek 4.3.4. $\mathbb{Z}_5$ üzerinde C_1 , tek parite kontrollü $(n_1 = 3, k_1 = 2, d_1 = 2)$ parametrelerine sahip bir satır kodu ve C_2 de, tek parite kontrollü $(n_2 = 4, k_2 = 3, d_2 = 2)$ parametrelerine sahip bir sütun kodu olsun.

$(n_1 = 3, k_1 = 2, d_1 = 2)$ satır kodunun tek parite kontrollü üreteç matrisi G_1 aşağıdaki şekilde alınsın:

$$G_1 = \begin{pmatrix} 1 & 0 & 4 \\ 0 & 1 & 4 \end{pmatrix}.$$

$(n_2 = 4, k_2 = 3, d_2 = 2)$ sütun kodunun tek parite kontrollü üreteç matrisi G_2 de şu şekilde alınsın:

$$G_2 = \begin{pmatrix} 1 & 0 & 0 & 4 \\ 0 & 1 & 0 & 4 \\ 0 & 0 & 1 & 4 \end{pmatrix}.$$

Bu taktirde, C çarpım kodunun üreteç matrisi olan G, G_1 ve G_2 üreteç matrislerinin kronecker çarpımı olur. Yani,

$$G = G_1 \otimes G_2 = \begin{pmatrix} 1 & 0 & 4 \\ 0 & 1 & 4 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 & 0 & 4 \\ 0 & 1 & 0 & 4 \\ 0 & 0 & 1 & 4 \end{pmatrix} = \begin{pmatrix} 1.G_2 & 0.G_2 & 4.G_2 \\ 0.G_2 & 1.G_2 & 4.G_2 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 1 \\ 0 & 1 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 1 \\ 0 & 0 & 1 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 4 & 4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 4 & 0 & 4 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 0 & 0 & 4 & 1 \end{pmatrix}$$

olur.

McEliece şifrelemesi için gönderilen mesaj, terslenebilir S matrisi ve permütasyon matrisi P sırasıyla aşağıdaki gibi olsun:

$$m = (x_1, x_2, x_3, x_4, x_5, x_6) = (3,2,0,1,4,2)$$

$$S = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad P = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Mesajı alacak kişi, gizli anahtarları olan S, G ve P ile $G^{'} = SGP$ olacak şekilde $G^{'}$ açık anahtarını hesaplar. $G^{'}$ açık anahtarı,

$$G^{'} = \begin{pmatrix} 1 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 1 \\ 0 & 1 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 1 \\ 0 & 0 & 1 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 4 & 4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 4 & 0 & 4 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 0 & 0 & 4 & 1 \end{pmatrix}$$

olarak bulunur. $G^{'}$ açık anahtar olduğundan mesajı gönderecek kişi, $G^{'}$ açık anahtarına erişebilir. Bu açık anahtar ile mesajını iletir. Önerme 5.3.1'den ayrılamaz hata vektörünün ağırlığı 3 tür. Dolayısıyla $t = 3$ ağırlıklı hata vektörü Tanım 5.3.2 deki iki şartı sağlayacak şekilde aşağıdaki gibi alınsın:

$$e = (001040000003).$$

Mesajı gönderecek kişi, $m = (x_1, x_2, x_3, x_4, x_5, x_6) = (320142)$ mesajını şifreleme fonksiyonu $c = mG^{'} + e$ eşitliğini kullanarak m mesajını şifreler ve c şifrelimetnini diğer bir ifadeyle c kodsözünü elde eder. Burada

$$mG^{'} = (320142)\begin{pmatrix} 1 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 1 \\ 0 & 1 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 1 \\ 0 & 0 & 1 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 4 & 4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 4 & 0 & 4 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 0 & 0 & 4 & 1 \end{pmatrix} = (320014231432)$$

dır. C çarpım kodu, matris kodun özel hali olduğundan elde edilen $c_1 = (320014231432)$ kodsözü şu şekilde yazılabilir:

$$c_1 = \begin{pmatrix} 3 & 2 & 0 & 0 \\ 1 & 4 & 2 & 3 \\ 1 & 4 & 3 & 2 \end{pmatrix}.$$

Benzer biçimde hata vektörü, $e = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix}$ şeklinde yazılabilir. Buradan,

$$c = mG' + e = c_1 + e = \begin{pmatrix} 3 & 2 & 0 & 0 \\ 1 & 4 & 2 & 3 \\ 1 & 4 & 3 & 2 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 1 & 0 \\ 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix} = \begin{pmatrix} 3 & 2 & 1 & 0 \\ 0 & 4 & 2 & 3 \\ 1 & 4 & 3 & 0 \end{pmatrix}$$

olur ve c kodsözü satır satır iletilir. C çarpım kod olduğundan kodsözde yatay ve dikey sendromlar sıfıra eşit olacağından satır satır iletimde ilk satırda 1 hata sembolü, ikinci satırda 4 hata sembolü ve üçüncü satırda 3 hata sembolü tespit edilir. Kodsöz bütün satırlarıyla iletidildiğinde birinci sütunda 4 hata sembolü, ikinci sütunda hatanın olmadığı, üçüncü sütunda 1 hata sembolü ve dördüncü sütunda 3 hata sembolü tespit edilir. Tanım 5.3.2'de gözönüne alınarak mesajı alan kişi hata vektörünü kolayca bulabilir. Mesajı alan kişi hata vektörünü aşağıdaki gibi tespit eder:

$$e = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix}.$$

Kanalda meydana gelen hataları ve orjinal kodsözü mesajı alan kişi bilmediğinden matris kodları sayesinde hatanın nerede olduğu kolayca tespit edilir. Hata vektörünü tespit eden kişi elde etmek istediği orjinal kodsözü bulur. Elde edilmek istenen kodsözde aşağıdaki gibi bulunur:

$$c = c_1 + e = c_1 + \begin{pmatrix} 0 & 0 & 1 & 0 \\ 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix} = \begin{pmatrix} 3 & 2 & 1 & 0 \\ 0 & 4 & 2 & 3 \\ 1 & 4 & 3 & 0 \end{pmatrix} \Rightarrow c_1 = \begin{pmatrix} 3 & 2 & 0 & 0 \\ 1 & 4 & 2 & 3 \\ 1 & 4 & 3 & 2 \end{pmatrix}.$$

Matris kodlarının özel yapısı olan çarpım kodlarının en büyük avantajının hata tespitinde kolaylık sağlaması olduğu görülür. Deşifreleme Örnek 5.2.1'deki gibi yapılır.

BÖLÜM 6. SONUÇ VE ÖNERİLER

Lineer kodların bir ailesi olan matris kodlar ele alınıp matris kodlar açık anahtarlı şifreleme sistemlerinden biri olan McEliece şifreleme sistemine uygulanmıştır. Bu sayede McEliece şifreleme sisteminde meydana gelebilecek hataların, matris kodlar aracılığıyla daha kolay bir şekilde tespit edilebileceği ve düzeltilebileceği gözlenmiştir.

Beşinci bölümde yapılan matris kodların McEliece şifreleme sistemine uygulanması değişik şifreleme sistemleri üzerinde de uygulanabilir. Aynı zamanda matris kodlar sıradan veya ardışık hata tespit etmekte kolaylık sağladığından ardışık hata düzelten kodlar teorisi için de hem McEliece şifreleme sistemi hem de değişik şifreleme sistemleri incelenebilir.

KAYNAKLAR

[1] ÖZTÜRK M., Security Aspects in Digital Communications, Yüksek Lisans, Orta Doğu Teknik Üniversitesi, sayfa 1-13, Eylül 2002.

[2] APOHAN A.M., Cryptography, Yüksek Lisans, Orta Doğu Teknik Üniversitesi, sayfa 1-7, Temmuz 1993.

[3] TRAPPE W., WASHINGTON L.C., Introduction to Cryptography with Coding Theory, Prentice Hall, 2002.

[4] ÇALLIALP F., Örneklerle Soyut Cebir, Sakarya Üniversitesi Yayınları No:16, Sakarya, 1995.

[5] STINSON D.R., Cryptography Theory and Practice, CRC Press LLC, 1995.

[6] SHANNON C.E., A Mathematical Theory of Communication, The Bell System Technical Journal, 27:379-423 and 623-656, July and October 1948.

[7] HAMMING R.W., Error detecting and error correcting codes, The Bell System Technical Journal, 29(2): 147-160, 1950.

[8] GOLAY M.J.E., Notes on digital coding, Proc. I.R.E., 37:657, 1949.

[9] OLOFSSON M., Lectures notes on Error Control Coding, Linköpings Universitet, 2005.

[10] ROMAN S., Coding and Information Theory, Graduate Text in Mathematics, Springer Verlag, 1992.

[11] CHAPMAN H., Coding Theory. St Edmundsbury Press, 12-105, Great Britian, 1996.

[12] RAYMOND H., A first course in coding theory, Oxford Press, 1996.

[13] BLAUM M., FARRELL P.G., VAN TILBORG H.C.A., Array codes, in:V. Pless, W. Cary Huffman (Eds.), Handbook of Coding Theory, vol. II, Elsevier, North-Holland, pp. 1855–1909, 1998.

[14] SAPNA J., Campopiano-type bounds in non-Hamming array coding,

ScienceDirect, Linear Algebra and its Applications 420, pp.135-159, 2007.

[15] ELIAS, P., Error free coding, IEEE Transaction, IT-4, pp.29-37, 1954.

[16] HOSANY M.A., SOYJAUDAH K.M.S., Statistical time channel evaluation of array codes in block coded phase modulation, Proceedings of the International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL'06).

[17] HONARY B., SOYJAUDAH K.M.S., RAMSAWOCK G., YU WAI MAN Y.K.L., Modified trellis decoding of simple product codes and their error performance in Gaussian channels, Proceedings IEEE CMTC Globecom'95, pp. 22-26, 1995.

[18] HONARY B., MARKARIAN G., Trellis decoding of block codes: A practical approach, Kluwer Academic Publisher, 1997.

[19] ZHE-XIAN-WAN, Lectures on finite fields and Galois rings, World Scientific Publishing, pp.242, 2003.

[20] HONARY B., MARKARIAN G., FARRELL P.G., Generalised array codes and their trellis structure, Electronics Letters, vol. 29, No.6, pp.242-245, 1993.

[21] YUAN D., GAO C., ZHANG L., Gac-based trellis decoding of block codes in Rayleigh fading channel, IEEE Transaction, pp.1449-1452, 2000.

[22] YUAN D., GAO C., ZHANG L., ZHIGANG, CAO, Generalized array codes for wireless image communication in presence of fading, IEEE Transaction, pp.408-411, 2001.

[23] MACWILLIAMS F.J. and SLOANE N.J., The theory of correcting codes, North Holland Pub. Co., 1977.

[24] RAMESH M.P., Near-Optimum decoding of product codes: Block turbo codes, IEEE Transaction on Communications, vol.46, no.8, August 1998.

[25] PETER S., Error Control Coding, John Wiley&Sons, Ltd, 2002.

[26] DIFFIE W. and HELLMAN M. E., New directions in cryptography, IEEE Transaction on Information Theory 22, pp. 644 – 654, 1976.

[27] McELIECE R.J., A public key cryptosystem based on algebraic coding theory, DSN Progress Report 42-44, pp.114-116, 1978.

[28] BERLEKAMP E.R., Goppa codes, IEEE Transaction on Information Theory, vol. IT-19, No.5, September 1973.

[29] GARY L.M., MUMMERT C., Algebraic coding theory on finite fields and applications, American Mathematical Society, 2007.

[30] GILBERT S., Introduction to linear algebra, Third edition, Wellesley Cambridge Press, 2003-03-01.

[31] ADAMS C.M., MEIJER H., Security-Related comments regarding McEliece's public-key cryptosystem, IEEE Transcantions on Information Theory, vol.35, No.2, March 1989.

[32] RAO T.R.N., Cryptosystems using algebraic codes, in Proc. Int. Conf. Computer Systems and Signal Processing, Bangalore, India, Dec.1984.

[33] RAO T.R.N., NAM K.H., Private-key algebraic-code encryptions, IEEE Transactions on Information Theory, vol.35, No.4, July 1989.

[34] HUNG-MIN S., SHIUH-PYNG S., On private-key cryptosystems based on product codes, Springer-Verlag Berlin Heidelberg, 1998.

ÖZGEÇMİŞ

Vedat Şiap, 20.04.1981'de Makedonya'da doğdu. İlkokul ve lise eğitimini Bayrampaşa'da, ortaokul eğitimini Eyüp'te tamamladı. 1999 yılında Yabancı dil ağırlıklı Rıfat Canayakın Lisesi'nden mezun oldu. 1999 yılında başladığı Gaziosmanpaşa Üniversitesi Matematik bölümünü 2003 yılında bölüm üçüncüsü olarak bitirdi. 2003 – 2006 yılları arasında özel bir dersanede Matematik öğretmeni olarak çalıştı. 2005 yılında Sakarya Üniversitesi Matematik Anabilim Dalı Matematik Bölümü Yüksek Lisans Programına girdi. 2006 yılında Japonya Milli Eğitim Bakanlığı bursu olan Monbukagakusho Yüksek Lisans ve Doktora bursu kazandı.