

## DOĞRUSAL KODLAR

Sevda Sezer

Akdeniz Üniversitesi, Matematik Bölümü, 07058-ANTALYA

Sonlu cisimlerin<sup>1</sup> önemli ve ilginç uygulama alanlarından biri de cebirsel kodlama teorisidir. Haberleşmede meydana gelen bazı problemleri çözmek için ortaya çıkan ve yaklaşık 50 yıllık bir geçmişi olan bu teoride, kısa geçmişine rağmen, önemli gelişmeler kaydedilmiştir.

Bu yazıda, *Kodlama Teori*'sinin tarihsel gelişiminden çok, konu hakkında bilgisi olmayan matematikseverlere, kodlama yönteminin önemi, tanımı ve özellikleri hakkında temel bilgiler verilecektir.

İki kişinin, iki ülkenin veya kısacası iki tarafın "doğru bir şekilde" haberleşmek istediğini düşünelim (burada haberleşmenin gizli olması gerekmiyor). "Doğru bir şekilde" ifadesini kullandım; çünkü, insan hatasından veya parazitten (aşırı ısı, radyasyon v.b.) dolayı bazen bir tarafın mesajı karşı tarafa hatalı bir şekilde ulaşabiliyor. İşte *Kodlama Teorisi* de, haberleşmenin doğru bir şekilde yapılmasını amaçlayan, eğer haberleşmede hata oluşmuşsa bu hatanın nerede olabileceğini inceleyen ve bundan daha da önemlisi hatalı olarak ulaşan mesajdan doğru mesajı elde etmek için çeşitli yöntemler geliştiren bir teoridir.

Günlük yaşantımızda, gerek telefon gerekse karşılıklı konuşmalarımızda, önemli bir mesajı (örneğin; adımızı, soyadımızı v.b bilgiyi) karşımızdaki kişiye ilettiğimizde "Lütfen kodlar mısınız?" sorusuyla hemen hemen hepimiz karşılaşmışızdır. Bu durumda genelde izlenen yol, "karşı tarafa mesajla birlikte bazı ek bilgilerin iletilmesiyle mesajın doğru anlaşılmasını sağlamak" 'tan ibarettir. Örneğin, telefonla önemli bir rezervasyon yaptığımı düşünelim. İlgili kişi benden adımlı kodlamamı istediğinde, eğer ben adımlı harf-harf S-E-V-D-A olarak kodlarsam (söylersem), adım SELDA, SEYDA, ŞEYDA v.b. biçimde yanlış (hatalı) anlaşılabilir. Oysa, ben adımlı

Samsun 'un S 'si, Edirne 'nin E 'si, Van 'ın V 'si, Denizli 'nin D 'si, Antalya 'nın A 'sı

olarak kodlarsam, adımın (mesajımın) doğru anlaşılma olasılığını oldukça arttırmış olurum.

Kodlama Teorisinin önemini şu örnekle daha iyi anlayabiliriz: Diyelim ki, Dünya'dan Mars'a bir uzay aracı gönderiyoruz ve bu aracın bize göndereceği görüntülere göre, aracın Mars'a inip inmeyeceğine karar vereceğiz. Eğer görüntüler iyi olursa aracın Mars'a iniş yapabileceğini, aksi durumda da yörüngede kalması gerektiğini, sırasıyla, 1 ve 0 mesajları (işaretleri) ile ona ileticeğiz. Eğer, uzay aracından gelen görüntüler, aracın Mars'a güvenli bir şekilde iniş yapamayacağını gösterir ve biz buna rağmen yanlışlıkla 0 yerine 1 mesajını gönderirsek ya da 0 mesajı bazı parazitlenmelerden dolayı 1 olarak araca ulaşırsa gerisini siz düşünün artık... Gitti canım keten helva...

Peki bu veya benzeri istenmeyen durumlarla karşılaşmamak için ne yapılabilir? Gerek bizden gerekse dış etkenlerden kaynaklanan hatalar sifira indirgenemeyeceğinden dolayı, demek ki mesaj alış-verişlerinde az da olsa hatalı mesajlarla karşılaşılacaktır. Bu noktadan hareketle kodlama ile ilgili çalışmalarda "hatalı mesajdan doğru mesajı elde edebilme olasılığını arttıran kodlama yöntemlerini bulup, geliştirme" konusu büyük önem kazanmaktadır.

Demek ki, bir mesajın karşı tarafa hatalı gitme olasılığı var. Bu durum üzerinde biraz olasılık hesapları yapmaya ne dersiniz? Belki bu arada "Neden kodlama yöntemlerine ihtiyaç duyulmaktadır?" sorusuna da bir cevap bulmuş oluruz. Bunun için önce bir mesajı (herhangi bir kodlama yöntemi uygulamadan) karşı tarafa gönderelim ve mesajın doğru elde edilebilme olasılığını hesaplayalım. Daha sonra da mesajı bazı ek bilgilerle birlikte (yani, uygun bir kodlama yöntemi uygulayarak)

<sup>1</sup>Sonlu cisim kavramını bilmeyen okurlar, sonlu cisim olarak  $Z_2 = \{0, 1\}$  'i alıp işlemlerini (mod 2) 'ye göre yapabilirler.

karşı tarafa göndereyim ve bu durumda da mesajın doğru elde edilebilme olasılığını hesaplayalım. Sonuçta da bu iki olasılığı karşılaştıralım. Yine yukarıdaki örneği ele alalım ve  $\{0,1\}$  'den oluşan mesaj birimlerimizden herhangi birisinin karşı tarafa "doğru" gitme olasılığı 0.99; "yanlış" gitme olasılığı da 0.01 olsun. 500 birimlik bir mesajı karşı tarafa direkt (herhangi bir yöntem uygulamadan) gönderdiğimizizi düşünelim (mesaj iletiminde, her birimin doğru veya yanlış da olsa karşı tarafa ulaştığı ve herhangi bir birimde meydana gelen bir hatanın diğer birimleri etkilemediği kabul edilmektedir). Bu durumda, 500 birimlik bir mesajın doğru elde edilebilme olasılığı, her birimin "doğru" gitme olasılığının çarpımına eşittir, yani  $(0.99)^{500} \cong 0.0066$  'dir. Görüldüğü gibi bu olasılık hiç de iç açici değil; siz de, mesajların doğru ulaşma olasılığı bu kadar düşük olan bir yöntemi kullanmak istemezsiniz, değil mi?

"Acaba mesajımızın doğru elde edilebilme olasılığını biraz daha arttıracak yöntemler geliştiremez miyiz?" sorusunun cevabı, daha fazla işlem yaparak ve daha fazla zaman harcayarak, "EVELT" tir (yani, zaman ve işlem yükünün artmasıyla sözü edilen olasılığın artması doğru orantılıdır). Bu olasılığı arttırıcı yöntemlerden birisi de *n-li tekrar yöntemi*dir. *n-li* tekrar yönteminde, *n* bir tek tamsayı olmak üzere, mesajdaki her bir birimin *n* defa yanyana (örneğin; 0110 yerine  $\underbrace{0\dots0}_n \underbrace{1\dots1}_n \underbrace{1\dots1}_n \underbrace{0\dots0}_n$ )

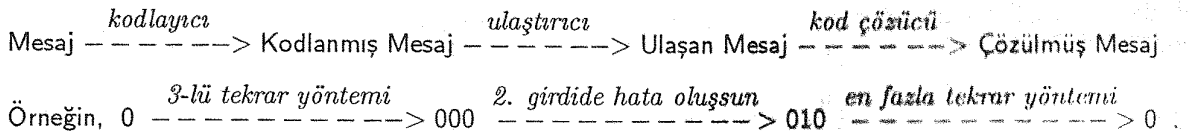
yazılarak karşı tarafa gönderilir. Karşı taraf da elde ettiği mesajı, sırasıyla *n-li* bloklara ayırır, her blokta hangi sayı en fazla tekrarlanmışsa, *n-li* bloğa o sayıyı karşılık getirerek orjinal mesajı elde etmeye çalışır (mesajın ulaşımında, hata sayısı arttıkça olayın meydana gelme olasılığı azalacağından, kodlama yöntemlerinde az hatalı durumlar ele alınır).

Biz örneğimize kaldığımız yerden devam edelim ve 500 birimlik mesajımızı 3-lü tekrar yöntemi ile karşı tarafa göndereyim. Yani, 0 yerine 000 , 1 yerine de 111 yazarak 500 birimlik mesajı 1500 birimlik mesaj olarak karşı tarafa göndereyim. Burada dikkat edilecek nokta 000, 100, 010, 001 üçlü bloklarının her birine 0 'ın (benzer şekilde; 111, 110, 101, 011 üçlü bloklarının herbirine 1 'in) karşılık gelmesidir. Bu nedenle, bir birimin (0 veya 1 'in) doğru elde edilebilme olasılığı, bu üçlülerin meydana gelme olasılıklarının toplamına eşittir. Bu olasılık da,

$$(0.99)^3 + (0.99)^2 \cdot (0.11) + (0.99)^2 \cdot (0.11) + (0.99)^2 \cdot (0.11) \cong 0.9997$$

'dir. Dolayısıyla, 3-lü tekrar yöntemiyle 500 birimlik bir mesajın doğru elde edilebilme olasılığı,  $(0.9997)^{500} \cong 0.86$  olur (tekrar sayısı arttıkça bu olasılığın artacağı açıktır). Kodlama yöntemlerine neden ihtiyaç duyulduğu veya öneminin ne olduğunu sadece 0.0066 ve 0.86 olasılıklarını karşılaştırarak da anlayabiliriz.

Aşağıdaki şema ile, Kodlama Teorisinde bir mesajın karşı tarafa gönderilip çözümlünceye kadar hangi aşamalardan geçtiği açıklanmaktadır:



Aşağıda kodlama yöntemi, kod ve kodlanmış mesajların matematiksel tanımına yer verilmiştir:

**Tanım 1 :**  $\mathbb{F}$  sonlu bir cisim ve  $\mathcal{M}$  tüm mesajlarımızdan oluşan küme olsun.  $\mathcal{M}$  'nin  $k$ -boyutlu bir  $\mathbb{F}$ -vektör uzayı olduğunu kabul edelim.  $\mathcal{M}$  'yi

$$\mathbb{F}^n := \{ (a_1, \dots, a_n) : a_i \in \mathbb{F}, i = 1, \dots, n \}$$

'nin  $k$ -boyutlu bir  $\mathcal{C}$  altvektör uzayına dönüştüren bir  $\mathbb{F}$ -doğrusal dönüşüme *kodlama yöntemi*,  $\mathcal{C}$  'ye bir  $[n, k]$  *doğrusal kod*,  $\mathcal{C}$  'nin elemanlarına da *kodlanmış kelimeler (mesajlar)* denir. Ayrıca, eğer  $\mathbb{F} = \mathbb{Z}_2 (= \{0,1\})$  ise  $\mathcal{C}$  'ye *binary (ikili birime dayanan) kod* denir (bu tür kodlar en çok kullanılan kodlardır).

$\mathcal{C}$   $k$ -boyutlu bir  $\mathbb{F}$ -vektör uzayı olduğu için,  $\mathcal{C}$  'nin eleman sayısının  $|\mathcal{C}| = |\mathbb{F}|^k$  olacağı açıktır.

Bundan sonra  $\mathbb{F}$  bir sonlu cisim,  $\mathcal{C}$  de  $[n, k]$  kod olarak düşünülecektir.

**Örnek 1 :**  $\mathbb{F} = \mathbb{Z}_2$ ,

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

ve

$$f : \mathcal{M} \longrightarrow \mathbb{F}^7, f(a_1, a_2, a_3, a_4) = (a_1, a_2, a_3, a_4, a_1 + a_2 + a_3, a_1 + a_3 + a_4, a_2 + a_3 + a_4)$$

olmak üzere aşağıdaki tablo elde edilir:

$\mathcal{M}$	$f$	$\mathcal{C}$
0000	→	0000000
0001	→	0001011
0010	→	0010111
0100	→	0100101
1000	→	1000110
1100	→	1100011
1010	→	1010001
1001	→	1001101
0110	→	0110010
0101	→	0101110
0011	→	0011100
1110	→	1110100
1101	→	1101100
1011	→	1011010
0111	→	0111001
1111	→	1111111

Bu şekilde oluşturulan  $\mathcal{C}$  bir  $[7, 4]$  koddur, bu kod *Hamming*  $[7, 4]$  kodu olarak adlandırılır.

Yandaki örnekte eğer bir kodlanmış kelime karşı tarafa 0111111 olarak ulaşırsa, bu kelime  $\mathcal{C}$  'nin elemanı olmadığı için bir yerlerde hata olduğu anlaşılır ve yapılan incelemelerde bunun 1111111 olarak gelmesi gerektiği sonucuna varılır. Çünkü; bu durumda sadece bir hata (ilk girdide) meydana gelmiştir, diğer durumlarda ise en az iki hatanın (yapılmış olması gerekir (daha önce de açıklandığı gibi, az sayıda hata yapma olasılığı çok sayıda hata yapma olasılığından fazla olduğu için, mesaj alış-verişlerinde olasılığı fazla olan durum incelenir, diğer durum gözardı edilir.) Görüldüğü gibi, bazı kodlama yöntemlerinde, kodlanmış bir mesaj hatalı olarak ulaşsa bile bunun düzeltilebilme imkanı bulunabilmektedir.

Kodlama Teorisinde aşağıdaki kavramlar büyük önem taşımaktadır:

**Tanım 2 :**  $\mathbb{F}$  bir cisim,  $a = (a_1, \dots, a_n)$ ,  $b = (b_1, \dots, b_n) \in \mathbb{F}^n$  için

$$d(a, b) := |\{i : a_i \neq b_i, i = 1, \dots, n\}|, w(a) := |\{i : a_i \neq 0\}|$$

olmak üzere,  $d(a, b)$  'ye  $a$  ve  $b$  'nin *Hamming*<sup>2</sup> uzaklığı,  $w(a)$  'ya da  $a$  'nın *Hamming ağırlığı* denir. Ayrıca,  $\mathcal{C}$  bir kod olmak üzere

$$d(\mathcal{C}) := \min\{d(a, b) : a, b \in \mathcal{C}, a \neq b\} = \min\{w(c) : c \in \mathcal{C}, c \neq (0, \dots, 0)\}$$

olarak tanımlanan  $d(\mathcal{C})$  sayısına  $\mathcal{C}$  'nin *minimum uzaklığı* (veya *ağırlığı*) denir.

**Örnek 2 :** Hamming  $[7, 4]$  kodunda,  $a = (0001011)$ ,  $b = (0111001)$ ,  $u = (1111111)$ ,  $v = (0000000)$  olsun. Bu durumda,  $d(a, a) = 0$ ,  $d(a, b) = 3$ ,  $d(a, u) = 4$ ,  $d(a, v) = 3$ ,  $d(b, u) = 3$ ,  $d(u, v) = 7$ ,  $w(a) = 3$ ,  $w(b) = 4$ ,  $w(u) = 7$ ,  $w(v) = 0$  ve  $d(\mathcal{C}) = \min\{3, 4, 7\} = 3$  tür.

Şimdi de Hamming uzaklığı ve Hamming ağırlığı ile ilgili bazı özellikleri inceleyelim:

<sup>2</sup>Richard W. Hamming (1915-1998), 1942 'de İllinois Üniversitesi Matematik Bölümünde doktorasını yaptı. 1950 yılında yayınlanan hata-arama ve hata-düzeltilme kodları ile ilgili makalesi bilgi teknolojilerinde bir dönüm noktası olmuştur. Gerek bilgisayar gerekse haberleşme alanında, 75 'e yakın makalesi, yarım düzineden fazla da kitabı bulunmaktadır.

**Özellikler :**  $C$  bir  $[n,k]$  kod;  $d$ ,  $w$  daha önce tanımlandığı gibi (sırasıyla, Hamming uzaklığı ve Hamming ağırlığı) ve  $u = (u_1, \dots, u_n)$ ,  $v = (v_1, \dots, v_n)$ ,  $s = (s_1, \dots, s_n) \in \mathbb{F}^n$  olsun. Bu durumda,

$$(1) \quad d(u, v) = d(v, u) = w(v - u) = w(u - v),$$

$$(2) \quad d(u, v) \leq d(u, s) + d(v, s)$$

'dir.

**İspat (1) :**

$$\begin{aligned} d(u, v) &= |\{i : u_i \neq v_i\}| \\ &= |\{i : v_i \neq u_i\}|, \quad (= d(v, u)) \\ &= |\{i : v_i - u_i \neq 0\}|, \quad (= w(v - u)) \\ &= |\{i : u_i - v_i \neq 0\}|, \quad (= w(u - v)) \end{aligned}$$

'dir.

(2) : Sözkonusu eşitsizliği karşılıklı girdileri inceleyerek ispatlayabiliriz. Eğer uygun bir  $j \in \{1, \dots, n\}$  için  $u_j \neq v_j$  ise (aksi durumda eşitsizliğin sağlanacağı açıktır) hem  $u_j = s_j$  hem de  $v_j = s_j$  olamaz. Çünkü, bu durumda  $u_j = s_j = v_j$  olmalıdır ki, bu başlangıçtaki koşulumuz olan  $u_j \neq v_j$  ile çelişir. Ayrıca, diğer durumlar için eşitsizlik sağlanacağından dolayı,  $d(u, v) \leq d(u, s) + d(v, s)$  olmalıdır.

Aşağıdaki teorem sayesinde, bir kodun düzeltilebilme (hatalı ulaşılan mesajdan doğru mesajı elde etme) kapasitesini belirleyebiliriz:

**Teorem 2 :**  $C$  bir kod olmak üzere, eğer sıfırdan farklı her  $c \in C$  için  $w(c) \geq 2t + 1$  olacak biçimde bir  $t \in \mathbb{N}$  varsa,  $C$  'nin herhangi bir elemanında meydana gelen  $t$  veya  $t$  'den daha az hata düzeltilebilir (yani, doğru) mesaj elde edilebilir. (Buradaki  $t$  sayısına  $C$  'nin *düzeltilme kapasitesi* denir).

**İspat :** Eğer,  $v \in \mathbb{F}^n$  ve  $d(v, c) \leq t$  olacak biçimde bir  $c \in C$  varsa  $c$  'nin tek türlü belirli (yani,  $c' \in C \setminus \{c\}$  için  $d(v, c') > t$ ) olduğunu göstermek ispat için yeterlidir (Neden?).  $v \in \mathbb{F}^n$ ,  $t$  veya  $t$  'den daha az hatayla elimize ulaşsın. Bu durumda, öyle bir  $c \in C$  vardır ki,  $d(v, c) \leq t$  olur. Diğer yandan,  $c' \in C \setminus \{c\}$  için  $c - c' \neq 0$  olduğundan

$$2t + 1 \leq w(c - c') = d(c, c') \leq d(v, c) + d(v, c') \leq t + d(v, c') \implies d(v, c') > t$$

eşitsizliğini elde ederiz. Bu da, bize  $v$  'nin  $c$  olarak düzeltilmesi gerektiği sonucunu verir.  $\square$

Teorem 2 'yi Hamming [7,4] koduna uygulamak istersek; sıfırdan farklı her  $c \in C$  için  $w(c) \geq 2 \cdot 1 + 1 = 3$  olduğundan, teorem gereği, kodlanmış mesajların ulaşımında meydana gelen 1 hata düzeltilebilir, yani  $C$  'nin düzeltilebilme kapasitesi 1 'dir. Örneğin,  $v = 1001011$  alınıp ( $v \notin C$  olduğuna dikkat edelim), her  $c \in C$  için  $d(v, c)$  'ler hesaplandığında,  $c = (0001011)$  için  $d(v, c) = 1$  ve her  $c' \in C \setminus \{c\}$  için de  $d(v, c') > 1$  olduğundan,  $v$  'nin  $c$  olarak düzeltilmesi gerektiği sonucuna varılır.

Dikkat edecek olursak, Hamming [7,4] 'deki kodlanmış mesajların ilk 4 girdisi asıl mesajdan oluşmakta, diğer 3 girdisi ise fazladan bilgi içermektedir. Bu özelliği sağlayan dönüşümleri (dolayısıyla, kodları) inceleyelim:

**Tanım 3 :**  $C$  bir  $[n, k]$  kod olmak üzere, satırları  $C$  'nin taban elemanlarından oluşan  $k \times n$  matrise  $C$  'nin bir *üreteç matrisi* denir ve  $G$  ile gösterilir.

$G$ ,  $C$  'nin bir üreteç matrisi olmak üzere  $C = \{a.G : a \in \mathbb{F}^k\}$  olacağı açıktır.

**Örnek 3 :** Mesajlarımızın kümesi  $\mathcal{M} = \{00, 01, 02, 10, 11, 12, 20, 21, 22\}$  olmak üzere  $\mathbb{F} = \mathbb{Z}_3$

üzerinden

$$\mathcal{G} = \begin{bmatrix} 1 & 0 & 2 & 1 \\ 0 & 1 & 2 & 2 \end{bmatrix}$$

ile bir  $[4,2]$  kod elde etmek istersek mesajlar ve bunlara karşılık gelen kodlanmış kelimeler tablodaki gibi olur:

$\mathcal{M}$	$\mathcal{G}$	$\mathcal{C}$
00	→	0000
01	→	0122
02	→	0211
10	→	1021
11	→	1110
12	→	1202
20	→	2012
21	→	2101
22	→	2220

Burada görüldüğü gibi, sıfırdan farklı her  $c \in \mathcal{C}$  için  $w(c) \geq 2.1+1$  olduğundan, bu kodun düzeltilebilme kapasitesi 1 'dir.

**Örne 4 :** Hamming  $[7,4]$  kodu için bir üreteç matris

$$\mathcal{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

'dir. Burada  $\mathcal{G}$ ,  $I_4$   $4 \times 4$  birim matris,  $A$  da girdileri  $\mathbb{F}$  'nin elemanlarından oluşan  $k \times (n-k)$  matris olmak üzere,  $\mathcal{G} = [I_k \ A]$  biçimindedir. Eğer,  $\mathcal{G}$  bu örnekte olduğu gibi,  $\mathcal{G} = [I_k \ A]$  formunda ise  $\mathcal{G}$  'ye  $\mathcal{C}$  'nin standart üreteç matrisi (veya standart kodlama matrisi),  $\mathcal{C}$  'ye de sistematik kod adı verilir. Ayrıca, sistematik bir koddaki herhangi bir elemanın (yani, bir kodlanmış kelimenin) asıl mesajdan oluşan ilk  $k$ -birimlik kısmı *bilgi kısmı*; geri kalan  $(n-k)$ -birimlik kısmı da *kontrol kısmı* olarak adlandırılır.

Her doğrusal kod bir sistematik kod olarak düşünülebileceğinden [2] ve asıl mesajları içerdiği için sistematik kodlarla çalışmak çok daha avantajlı olacağından dolayı, bu tür kodlar Kodlama Teorisinde önemli bir yer teşkil etmektedir.

Elimizde, mesajlarımızı kodlayabilecek uygun ( $\mathcal{G}$  matrisi yardımıyla) bir metot olmasına karşılık maalesef, kodlanmış mesajların hatalı ulaşması durumunda, bunları her zaman için çözebilecek bir yöntem bulunmamaktadır (1 hatanın düzeltilebileceği bir yöntem hariç).  $\mathcal{H}$ ,  $[I_k \ A]$  formunda olmak üzere,

$$\mathcal{H} = \begin{bmatrix} -A \\ I_{n-k} \end{bmatrix}_{n \times (n-k)} = \begin{bmatrix} -a_{11} & -a_{12} & \dots & -a_{1(n-k)} \\ \vdots & \vdots & \dots & \vdots \\ -a_{k1} & -a_{k2} & \dots & -a_{k(n-k)} \\ 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}_{n \times (n-k)}$$

matrisine  $\mathcal{C}$  'nin kontrol matrisi denir. Eğer ulaşan kodlanmış mesajda 1 tane hata yapılmışsa,  $\mathcal{H}$  matrisi yardımıyla bu hatanın hangi girdide meydana geldiği ve doğru mesajın ne olması gerektiği hemen belirlenebilir:

Diyelim ki;  $w$  mesajı elimize ulaştı,

1. **Adım** :  $w \in \mathcal{C}$  ise, mesajın doğru geldiğini kabul ederiz.

2. **Adım** :  $w \notin \mathcal{C}$  ise,  $w \cdot \mathcal{H}$  'yi hesaplarız. Elde ettiğimiz sonuç  $\mathcal{H}$  'nin  $i$ . satırının  $s$  katına ( $s \in \mathbb{F} \setminus \{0\}$ ) eşit ve  $c := w - (00\dots 0 \underbrace{s}_{i.girdi} 0\dots 0) \in \mathcal{C}$  ise,  $w$  'nin  $i$ . girdisinde bir hata olduğunu anlarız. Bu nedenle  $w$  yerine  $c$  'nin ulaşması gerektiği sonucuna varırız ( $\mathcal{H}$  'nin herhangi iki satırının doğrusal bağımsız olduğunu kabul ediyoruz). Özel olarak  $\mathbb{F} = \mathbb{Z}_2$ ,  $w$  'de 1 hata oluşmuş ve  $w \cdot \mathcal{H}$ ,  $\mathcal{H}$  'nin  $i$ . satırına eşitse,  $w$  'nin  $i$ . girdisinde hata oluşmuş demektir. Eğer,  $i$ . girdi 1 ise 0 ile; 0 ise de 1 ile değiştirilmelidir.

3. **Adım** : Eğer 1. ve 2. adımdaki koşullar sağlanmıyorsa ulaşımda en az 2 hata yapılmış demektir. Bu durumda bu yöntemle doğru bir çözüm elde edilemeyebilir.

**Örnek 5** : Örnek 4 'deki Hamming [7,4] koduna karşılık gelen kontrol matrisi

$$\mathcal{H} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

'dir. Elimize  $w = (0000110)$  mesajı ulaştığında,  $w \notin \mathcal{C}$  olduğundan,  $w \cdot \mathcal{H}$  'yi hesaplırsak;  $w \cdot \mathcal{H} = (110)$  'ı elde ederiz.  $(110)$ ,  $\mathcal{H}$  'nin 1. satırı olduğu için  $s=1$  ve  $c = w - (s0\dots 0) = (1000110) \in \mathcal{C}$  olduğundan;  $w$  'nin,  $(1000110)$  'ın 1. girdisinde 1 hatanın oluşmasıyla elimize ulaştığı sonucuna varırız.

Kabul edelim ki,  $c = (1001100)$  kodlanmış mesajı elimize  $w = (1001010)$  olarak (en az iki hatayla) ulaşsın.  $w \notin \mathcal{C}$  olduğundan,  $w \cdot \mathcal{H}$  'yi hesaplırsak  $\mathcal{H}$  'nin 3. satırını, yani  $(111)$  'i, elde ederiz. Bu nedenle  $s = 1$  olmalıdır. Fakat, diğer yandan  $c = (1001010) - (0010000) = (1011010) \notin \mathcal{C}$  olduğundan, en az iki hatanın yapılması durumunda bu yöntemle doğru bir sonuç elde edemeyeceğimizi görmüş oluruz.

Bundan sonra, " $\mathcal{C}$  bir  $[n, k, d]$  kod" denilince,  $\mathcal{C}$  'nin  $\mathbb{F}^n$  'nin  $k$ -boyutlu bir  $\mathbb{F}$ -altvektör uzayı olduğu ve bu kodun minimum uzaklığının (veya ağırlığının)  $d$  olduğu anlaşılacaktır. Daha önce Teorem 1 'de bir kodun düzeltilebilme kapasitesinin  $w$  ile (dolayısıyla  $d$  ile) doğru orantılı olduğunu gördük. Şimdi de, "Acaba, bir  $[n, k, d]$  kodunda  $d$ ,  $n$  ve  $k$  arasında ne gibi bir ilişki var?" , " $d$  'nin  $n$  ve  $k$  'ya bağlı olarak alt ve üst sınırlarını belirleyebilir miyiz?" sorularına cevap aramaya çalışalım:

**Önerme 1 (Singleton Sınırı)** :  $\mathcal{C}$  bir  $[n, k, d]$  kod ise,  $k + d \leq n + 1$  'dir.

**İspat** :  $S := \{(a_1, \dots, a_n) \in \mathbb{F}^n : \text{her } i \geq d \text{ için } a_i = 0\}$  olarak tanımlanırsa;  $S$ ,  $\mathbb{F}^n$  'nin  $(d-1)$ -boyutlu bir alt vektör uzayı olur. Ayrıca, dikkat edilecek olursa, her  $a \in S$  için  $w(a) \leq d-1$  ve  $S \cap \mathcal{C} = \{(0, \dots, 0)\}$  'dir (Çünkü,  $0 \neq b \in S \cap \mathcal{C} \implies \begin{cases} b \in S \implies w(b) < d \\ b \in \mathcal{C} \implies w(b) \geq d \end{cases}$  çelişkisi elde edilir.) Bu nedenle,

$$k + (d-1) = \text{boyut } \mathcal{C} + \text{boyut } S = \text{boyut } (\mathcal{C} + S) + \text{boyut } (\mathcal{C} \cap S) = \text{boyut } (\mathcal{C} + S) \leq n \implies k + d \leq n + 1$$

'dir.  $\square$

$k + d = n + 1$  olan  $[n, k, d]$  kodları *maksimum uzaklıkla ayrılabilen kodlar* olarak adlandırılır. Kodlama Teorisindeki önemli kodlardan birisi de maksimum uzaklıkla ayrılabilen kodlardır. Çünkü, bu tür kodlar,  $n$  ve  $k$  verildiğinde  $d$  'si (dolayısıyla, düzeltilebilme kapasitesi) en fazla olan kodlardır.

Reed Solomon<sup>3</sup> kodları bu tür kodlara bir örnek teşkil eder:

**Reed Solomon Kodları :**  $|\mathbb{F}| = q$ ,  $n = q - 1$  ve  $\beta \in \mathbb{F} \setminus \{0\}$  'in bir ilkel kökü (yani,  $\mathbb{F} \setminus \{0\} = \{\beta, \beta^2, \dots, \beta^{n-1}\} = \langle \beta \rangle$ ) olsun.  $1 \leq k \leq n$  olmak üzere,

$$\mathcal{M}_k := \{ g \in \mathbb{F}[x] : \text{derece } g \leq k - 1 \}$$

kümesi  $\mathbb{F}$  'nin  $k$ -boyutlu bir altvektör uzayıdır.

$$f : \mathcal{M}_k \longrightarrow \mathbb{F}^n, \quad f(g) = (g(\beta), g(\beta^2), \dots, g(\beta^n))$$

olarak tanımlanırsa;  $f$ ,  $\mathbb{F}$ -doğrusal ve  $\ker f = \{0\}$  olduğundan (çünkü,  $\text{derece } g \leq k - 1$  için  $\beta, \dots, \beta^n$ ,  $g$  'nin  $n$  farklı kökü olamaz, dolayısıyla  $\ker f$  sıfırdan ibarettir) bire-bir bir dönüşümdür. Bu nedenle

$$\mathcal{C}_k := \{ (g(\beta), g(\beta^2), \dots, g(\beta^n)) : g \in \mathcal{M}_k \} \subseteq \mathbb{F}^n$$

olarak tanımlandığında,  $\mathcal{C}_k$  bir  $[n, k]$  kod olur. İşte bu koda *Reed Solomon Kodu* adı verilir. Burada,  $0 \neq c \in \mathcal{C}_k$  için

$$w(c) = n - |\{ i : g(\beta^i) = 0, i = 1, \dots, n \}| \geq n - \text{derece } g \geq n - (k - 1)$$

olduğundan,  $d = \min\{ w(c) : c \in \mathcal{C}_k, c \neq 0 \} \geq n - (k - 1)$  dir. Böylece,  $d + k \geq n + 1$  ve  $d + k \leq n + 1$  (Singleton Sınırı) eşitsizliklerinden  $d + k = n + 1$  elde edilir ki bu,  $\mathcal{C}_k$  'nın maksimum uzaklıkla ayrılabilen bir kod olduğunu gösterir. Reed Solomon kodlarını bir örnekle pekiştirelim:

**Örnek 6 :**  $\mathbb{F} = \mathbb{Z}_5$ ,  $n = 4$  (ve dolayısıyla  $k = 2$ ) alalım.  $\mathbb{F} \setminus \{0\} = \{1, 2, 3, 4\} = \langle 2 \rangle$  olduğundan,

$$\mathcal{M}_2 = \{ g \in \mathbb{Z}_5[x] : \text{derece } g \leq 1 \} = \{ a + bx : a, b \in \mathbb{Z}_5 \}$$

$$= \{ 0, 1, 2, 3, 4, x, 2x, 3x, 4x, 1+x, 1+2x, 1+3x, 1+4x, 2+x, 2+2x, \\ 2+3x, 2+4x, 3+x, 3+2x, 3+3x, 3+4x, 4+x, 4+2x, 4+3x, 4+4x \}$$

ve

$$\mathcal{C}_2 = \{ (f(\beta), f(\beta^2), f(\beta^3), f(\beta^4)) : f \in \mathcal{M}_2 \}$$

$$= \{ (0, 0, 0, 0), (1, 1, 1, 1), (2, 2, 2, 2), (3, 3, 3, 3), (4, 4, 4, 4), \\ (2, 4, 3, 1), (4, 3, 1, 2), (1, 2, 4, 3), (3, 1, 2, 4), (3, 0, 4, 2), \\ (0, 4, 2, 3), (2, 3, 0, 4), (4, 2, 3, 0), (4, 1, 0, 3), (1, 0, 3, 4), \\ (3, 4, 1, 0), (0, 3, 4, 1), (0, 2, 1, 4), (2, 1, 4, 0), (4, 0, 2, 1), \\ (1, 4, 0, 2), (1, 3, 2, 0), (3, 2, 0, 1), (0, 1, 3, 2), (2, 0, 1, 3) \}$$

olur. Burada,  $d = d(\mathcal{C}) = 3$  olduğu kolayca görülebilir ( $n + 1 = k + d \implies 4 + 1 - 2 = d \implies d = 3$ ). Bu nedenle,  $\mathcal{C}$  'nin düzeltilebilir kapasitesi 1 'dir.

Burada Kodlama Teorisindeki kısa gezintimize, istemeyerek de olsa, son veriyoruz. Diğer kodlama yöntemlerini incelemek için, yeni gezintilerde buluşmak dileğiyle...

## KAYNAKLAR

[ 1 ] Gallian, J. ; Contemporary Abstract Algebra; D. C. Heath and Company, 1990.

[ 2 ] Lint, J. H. ; Introduction to Coding Theory; Springer-Verlag, Heidelberg, 1992.

[ 3 ] Stichtenoth, H. ; Algebraic Function Fields and Codes; Springer-Verlag, Heidelberg, 1993.

<sup>3</sup>Reed Solomon, 1998 yılında Cornell Üniversitesinde doktorasını yaptı. Kendi adıyla anılan kodlarla ilgili bir çok yayını bulunmaktadır. Halen Wisconsin Üniversitesinde Matematik Bölümünde misalî profesör olarak görev yapmaktadır.