

33514

**YILDIZ TEKNİK ÜNİVERSİTESİ**

**FEN BİLİMLERİ ENSTİTÜSÜ**

**SAYISAL GENLİK  
MODÜLASYONU İÇİN  
ÇOK DÜZEYLİ  
KODLAYICILARIN TASARIMI**

**Elektronik ve Hab.Müh. Ali Yaşar ZORLU**

**F.B.E. Elektronik ve Haberleşme Mühendisliği Anabilim Dalında  
hazırlanan**

**YÜKSEK LİSANS TEZİ**

**Tez Danışmanı : Doç. Dr. Ümit AYGÖLÜ**

**T.C. YÜKSEKÖĞRETİM KURULU  
DOKÜMANASYON MERKEZİ**

**İSTANBUL, 1994**

## İÇİNDEKİLER

. İÇİNDEKİLER	iii
. TEŞEKKÜR	v
. ÖZET	vi
. SUMMARY	vii
1. GİRİŞ	1
2. BANT VERİMLİLİKLİ İŞARETLEŞME	6
2.1. Kafes Kodlamalı Modülasyon (TCM)	8
2.1.1. Kafes Kodlamalı Modülasyonda Kodlama ve İşaret Eşleme	11
2.1.2. TCM İçin Kodçözme Karmaşıklığı	14
2.1.2.1. Delikli Katlamalı Kodlar İçin Kodçözme Karmaşıklığı	15
2.2. Çok Düzeyli Kodlayıcılar	17
2.3. Imai - Hirakawa Tipi Kodlama	18
2.3.1. İşaret Kümesinin Bölünmesine Dayalı İşaret Eşleme	19
2.4. Pottie - Taylor Tipi Kodlama	21
2.5. Çok Düzeyli Kodlayıcılar İçin Serbest Öklit Uzaklığı ve Asimptotik Kodlama Kazancı	22
2.6. Çok Düzeyli Kodlayıcılar İçin Karmaşıklık Hesabı	24
2.7. Çok Aşamalı Kodçözme	25
3. KATLAMALI KODLAYICILAR KULLANILARAK SAYISAL GENLİK MODÜLASYONU İÇİN ÇOK DÜZEYLİ KODLAYICILARIN TASARIMI	28
3.1. Sayısal Genlik Modülasyonu İçin 2 ve 3 Düzeyli IHS Tipi Kodlayıcılara İlişkin Tasarım Örnekleri	29
3.2. IHS Tipi Kodlayıcılar İçin Hata Üst Sınırı	38
3.3. Bilgisayar Benzetim Programının Tanıtılması ve Elde Edilen Sonuçlar	46

4. 4-AM YARDIMIYLA ÇOK DÜZEYLİ HAT KODU TASARIMI	50
4.1. Hat Kodlamasının Kullanılma Nedeni ve Özellikleri	50
4.2. Çoklu 4-AM Hat Kodları	53
4.3. Çoklu 4-AM Kümeleri ve Bu Kümelerin Bölmelenmesi	54
4.4. 2 Düzeyli 4-AM Hat Kodları	56
4.5. Spektral Çözümleme	61
SONUÇ	69
KAYNAKLAR	71
EKLER	74
ÖZGEÇMİŞ	

Bu tezin hazırlanmasında, sürekli yardımlarıyla büyük emeđi geen, deđerli insan, Sayın Hocam Do. Dr. Ümit AYGÖLÜ'ne ve deđerli yardımlarıyla bana yardımcı olan Sayın Hocam Yrd. Do. Dr. Soner ÖZGÜNEL'e ve tüm alıřma arkadaşlarıma teřekkür ederim.

## ÖZET

Bu tezde, çok düzeyli kodlama ve çok aşamalı kodçözme tekniği incelenmiş, bu teknik sayısal genlik modülasyonuna (M-AM) uygulanmıştır.  $M=4$  ve  $M=8$  genlik düzeyleri için 2 ve 3 düzeyli Imai ve Hirakawa tipi kodlayıcılar tasarlanarak, kodlamasız ilişkin modülasyon türlerine göre 3-5 dB arasında, Ungerboeck tipi kafes kodlayıcılara göre ise 2 dB'ye varan kodlama kazançları sağlanmıştır. Diğer yandan kodlamasız duruma göre yaklaşık 5 dB'lik bir kazanç için Ungerboeck tipi kafes kodlayıcıların kodçözme karmaşıklıkları 128 iken, çok düzeyli kodlama ile bu karmaşıklık delikli katlamalı kodlarla 10'a kadar düşürülebilmektedir.

Temelbant veri iletişimde iletilecek işaretlerin güç spektrumlarının, hattın spektral karakteristiklerine uydurulması istenir. Bu amaca yönelik olarak bu tezde, doğru akım bileşeni içermeyen, alçak frekanslarda düşük genlikli spektral bileşenlere sahip, iki düzeyli 4-AM hat kodları tasarlanmıştır. Bu kodların spektral analizleri gerçekleştirilmiş ve literatürde bilinen bir hat koduyla karşılaştırılmıştır.

## SUMMARY

In this thesis, multilevel coding and multistage decoding methods are investigated and applied to the digital amplitude modulation (M-AM). 2 and 3 level Imai-Hirakawa type of coders are designed for  $M=4$  and  $M=8$ , and 3-5dB coding gain respect to uncoded modulation, and up to 2dB gain respect to Ungerboeck type of trellis coders, are obtained. On the other hand, while the decoding complexity of the Ungerboeck type of trellis coders is 128 for 5dB coding gain respect to uncoded case, this complexity is reduced up to 10 by using punctured convolutional codes.

In baseband data communication, the power spectrums of the transmitted signals are to be matched to the spectral characteristics of the channel. For this purpose, in this work, two-level 4-AM line codes are designed, which have spectral components with no dc component and low amplitude in low frequencies. The spectral analysis of these codes are realised and the power spectrums are compared to a line code appeared in literature.

## 1. GİRİŞ

Sayısal iletim sistemlerinde amaç, kaynak tarafından iletilen bilgilerin alıcıya olabildiğince az hata ile aktarılabilmesidir. Bu arada haberleşme sisteminin iletim hızında da bir düşme olmaması istenir.

Kaynak tarafından üretilen işaretler, iletim kanalında gürültü nedeniyle bozulmaya uğrarlar. Ayrıca kanalın izin verdiği bant genişliğinin aşılması, iletilen işaretlerin birbirleriyle girişmesine neden olur. Dolayısıyla, bir haberleşme sisteminin tasarımı için, bilgi taşıyan işaretlerin iletildiği kanalın özelliklerinin çok iyi bilinmesi gerekmektedir.

Kanaldaki gürültü nedeniyle bozulmayı önlemek için kanal kodlaması yöntemi kullanılır. Bu yöntem, iletilecek ikili enformasyon bitlerine fazladan kontrol bitleri ekleyerek kodlama yapma temeline dayanır. İletilecek işaretlerin kanal spektrumuna uygun biçime getirilmesi amacıyla ise hat kodlaması yöntemlerinden yararlanılır.

Kanal kodlamasında,  $k$  bitlik ikili enformasyon dizisine  $n-k$  tane fazladan kontrol biti eklenir. Dolayısıyla  $k$  bitlik bilgi,  $n$  bitlik ikili dizilere dönüştürülerek  $k/n$  oranlı bir kodlama yapılır. Alıcı tarafta ise hata düzeltme kodlarının kullanılmasıyla gürültüden dolayı oluşan bozulma sezilerek düzeltilir. Enformasyon bitlerine fazladan bitler ekleyerek kodlama yapmak, haberleşme sistemine ait simge hızının artması sonucunu doğurur. Bu ise iletim bant genişliğinin artmasına neden olur ki, bu istenmeyen bir durumdur. Kanal sığası olarak bilinen büyüklük, o kanaldan iletilecek en yüksek iletim hızını belirtir ve iletim hızının isteğe göre arttırılması engellenmiş olur. Tersine bir durum iletimin güvenilirliğini tehlikeye düşürür.

Bir haberleşme sisteminde, gürültü nedeniyle oluşan bozulmayı önlemek amacıyla uygulanan klasik hata düzeltme kodlaması, kodlama oranına ters orantılı olarak bant

genişliğini arttırır. Aslında bu yöntem, bant genişliğindeki artmanın yanısıra bir kodlama kazancı da sağlamaktadır. Bu durumda bant genişliğinde bir artma olmaksızın aynı zamanda önemli bir kodlama kazancı da sağlayan bir yöntemin olup olmadığı sorusuyla karşılaşılabılır. İlk kez Ungerboeck [Ungerboeck,1982] tarafından ortaya atılan, kodlama ve modülasyon işlemlerinin tasarım sırasında birlikte düşünüldüğü kafes kodlamalı modülasyon (Trellis Coded Modulation,TCM) yöntemi, klasik hata düzeltme kodlamasında karşılaşılan sorunları önemli ölçüde gidermiştir. Kafes kodlamalı modülasyon yönteminde, enformasyon bitlerine fazladan bitler eklemek yerine modülasyonlu işaret kümesi genişletilir. Kodlama sayesinde iletilecek işaretler arasındaki Öklit uzaklığı da arttırılmış olur. Bu ise gürültünün bozucu etkilerini azaltan bir durumdur.TCM tekniğiyle Ungerboeck tarafından tasarlanan kodlayıcılar ile, bant genişliğinde artma olmadan 2-3 dB'den 6 dB'ye varan kodlama kazançları sağlanabilmektedir.

Haberleşme sistemlerinin tasarımında önemli bir büyüklük de alıcı tarafta kodçözme işlemi sırasında yapılan karşılaştırma sayısıdır. Yapılan tasarımlarda, karmaşıklık sayısı olarak bilinen bu büyüklüğün küçük bir değerde olması, aynı zamanda yüksek kodlama kazancının sağlanması istenir.

Imai ve Hirakawa [Imai et al,1977] tarafından önerilen çok düzeyli kodlama yönteminde, her düzeyde ikili kodlayıcıların kullanıldığı bir sistem oluşturulmuştur. Fakat bu yöntemde kodlayıcı olarak blok kodların kullanılması ve alıcı tarafta sert karar verme yönteminin uygulanması nedeniyle ilk başlarda önemli bir kodlama kazancı elde edilememiştir. Fakat çok düzeyli kodlayıcılar, çok aşamalı kodçözme yönteminin kullanılmasına olanak verebildiklerinden dolayı kodçözme karmaşıklığını azaltırlar.



Yamaguchi ve Imai [Yamaguchi et al,1987] tarafından önerilen çok düzeyli kodlayıcılarda, her düzeyde ikili katlamalı kodlayıcılar kullanılmış ve alıcı tarafta yumuşak kararlı kodçözme yönteminin uygulandığı düşünülmüştür. 8-PSK modülasyonu için, kodlamasız 4-PSK'ya göre 6.02 dB'ye varan kodlama kazançları sağlanmıştır.

Daha sonra Pottie - Taylor [Pottie et al,1989] tarafından, çok düzeyli kodlayıcılar daha genel bir yapıda oluşturulmuştur. Bu yöntemde her düzeyde değişik türden kodlayıcılar kullanılabilir. Imai - Hirakawa tipi kodlayıcılarda, her düzeydeki kodlayıcının çıkışındaki yalnızca bir bit modülasyonlu işaretin belirlenmesinde kullanılırken, Pottie - Taylor tipi kodlayıcıda ise her düzeydeki kodlayıcının çıkışındaki birden fazla bit modülasyonlu işareti belirleyebilmektedir.

Imai - Hirakawa'nın ya da Pottie - Taylor'un önerdiği kodlayıcı sistemlerinde, ikili kod olarak delikli kodlar [Yasuda et al,1984] kullanıldığında, aynı kodlama kazancı için kodçözme karmaşıklığı daha da düşürülebilmektedir.

Bir haberleşme sistemi tasarımı için önemli bir ölçüt de iletilecek modülasyonlu işaretlere ait güç spektrumdur. Özellikle temelbant veri iletiminde, haberleşme sistemine ait frekans spektrumunda düşük frekans bileşenlerinin sifıra yakın olması ve sifır frekans bileşeninin de (dc bileşen) sifır olması istenir. Bu amaçla son yıllarda hat kodlarının, kodlamalı modülasyon tekniğine dayalı tasarımına yönelik çalışmalar yoğunlaşmıştır. Çok sayıda genlik düzeyinin kullanıldığı hat kodu tasarımı yöntemi Calderbank [Calderbank et al,1989] tarafından verilmiştir. Bu kodlar, kodlama hızında bir düşmeye neden olmazken, kodçözme karmaşıklığını oldukça arttırmaktadırlar. Üç düzeyli simgelerin (+, 0, -) kullanıldığı, 4 ve 8 durumlu kafeslere dayalı hat kodlarının tasarımı Aygözü [Aygözü et al,1993] tarafından yapılmış, daha sonra bu kodlar çok düzeyli

yapıda, yine aynı yazarlar tarafından gerçekleştirilmiştir [Aygölü et al,1994].

Kodlanmış işaret dizilerinin güç spektrum yoğunluğunun hesabına ilişkin bir yöntem Cariolaro [Cariolaro et al,1983] tarafından verilmiştir.

#### \* Tezde Yapılan Çalışmalar Ve İzlenen Yol

Bu tezde, çok düzeyli kodlama ve çok aşamalı kodçözme tekniği incelenmiş, bu teknik sayısal genlik modülasyonuna (M-AM) uygulanmıştır. M=4 ve M=8 genlik düzey için 2 ve 3 düzeyli Imai ve Hirakawa tipi kodlayıcılar tasarlanarak, kodlamasız ilişkin modülasyon türlerine göre 3-5 dB arasında, Ungerboeck tipi kafes kodlayıcılara göre ise 2 dB'ye varan kodlama kazançları sağlanmıştır. Diğer yandan kodlamasız duruma göre yaklaşık 5 dB'lik bir kazanç için Ungerboeck tipi kafes kodlayıcıların kodçözme karmaşıklıkları 128 iken, çok düzeyli kodlama ile bu karmaşıklık delikli katlamalı kodlarla 10'a kadar düşürülebilmektedir.

Temelbant veri iletişimde iletilecek işaretlerin güç spektrumlarının, hattın spektral karakteristiklerine uydurulması istenir. Bu amaca yönelik olarak bu tezde, doğru akım bileşeni içermeyen, alçak frekanslarda düşük genlikli spektral bileşenlere sahip iki düzeyli 4-AM hat kodları tasarlanmıştır. Bu kodların spektral analizleri Cariolaro [Cariolaro et al,1983] tarafından geliştirilen bir tekniğe dayalı olarak gerçekleştirilmiş, güç spektrumları literatürde bilinen hat kodlarıyla karşılaştırılmıştır.

Tezin ikinci bölümünde bant verimlilikli işaretleşme ve bu amaçla ortaya atılan kafes kodlamalı modülasyon ve çok düzeyli kodlama / çok aşamalı kodçözme teknikleri genel olarak incelenmiş, bu tekniklerle ilgili temel tanımlar ve ölçütler verilmiştir.

Üçüncü bölümde katlamalı ve delikli katlamalı kodlayıcılar kullanılarak 4-AM ve 8-AM için çok düzeyli kodlayıcı sistemlerin tasarımları yapılmıştır. Tasarlanan kodlardan düşük karmaşıklığa sahip bir örnek sisteme ait bit hata olasılığı için analitik yöntemlerle üst sınır ifadesi bulunarak, çeşitli işaret - gürültü oranları için üst sınır eğrisi elde edilmiştir. Bir bilgisayar benzetim programı yardımıyla, aynı kodlayıcıya ait bit hata olasılığı elde edilmiş ve kodlamasız 2-AM sistemiyle karşılaştırma yapılmıştır.

Tezin dördüncü bölümünde hat kodlaması tekniklerinden yararlanılarak 4-AM için iki düzeyli doğru akım bileşeni içermeyen, alçak frekanslarda düşük spektral bileşenlere sahip hat kodlayıcıları tasarlanmıştır. Tasarlanan kodların güç spektrumları analitik yolla ve benzetim yöntemleriyle yine bu bölümde elde edilmiştir.

Beşinci bölüm, tezde elde edilen sonuçlara ve çeşitli karşılaştırmalara ayrılmıştır. Kullanılan çeşitli bilgisayar programlarının birer örneği ekte sunulmaktadır.

## 2. BANT VERİMLİLİKLİ İŞARETLEŞME

Haberleşme sistemleri, bant genişliği sınırlı veya gücü sınırlı olmalarına göre sınıflandırılırlar. Bant genişliği sınırlı sistemlerde; iletim gücünün arttırılmasına izin verilmesine karşılık, bant genişliğini korumak için spektral açıdan verimli modülasyon yöntemleri kullanılır. Güç sınırlı sistemlerde ise; bant genişliğinin bir miktar artmasına karşılık, güç açısından verimli modülasyon yöntemleri kullanılır.

Bant verimlilikli işaretleşmede amaç, bant genişliğini arttırmaksızın daha düşük hata oranı ile iletimin gerçekleştirilmesidir. Bunu daha iyi kavramak için M-lik işaretleşmenin yapıldığı genel bir haberleşme sistemini düşünelim. Burada M, bu sistemin iletebildiği farklı simge sayısını göstermektedir ve alfabe eleman sayısı olarak bilinir. Eğer iletilecek bir simge m bit ile belirleniyorsa bu sisteme ilişkin alfabe,  $M = 2^m$  elemandan oluşacaktır. Dolayısıyla

$$m = \log_2 M \quad (2.1)$$

bağıntısını yazabiliriz.

$T_s$  zaman aralığı boyunca iletilen her bir simge, elektriksel bir gerilim işaretine karşı düşürülür. Sisteme ait enformasyon bit hızı  $R$ (bit/sn) şu şekilde gösterilebilir.

$$R = \frac{m}{T_s} = \frac{\log_2 M}{T_s} \quad (\text{bit/sn}) \quad (2.2)$$

Bir enformasyon biti için gerekli zaman aralığı ile,  $R$  bit hızı birbirleriyle ters orantılıdır. Benzer şekilde simge başına gerekli zaman da simge hızı ile ters orantılı olacaktır. Bu yüzden (2.2) bağıntısının yardımıyla,  $T_s$  süresi boyunca

iletilen simgeye ait her bir bit için gerekli  $T_b$  süresi ya da simge hızı  $R_s$  şu şekilde belirtilebilir:

$$T_b = \frac{1}{R} = \frac{T_s}{m} = \frac{1}{mR_s} \quad (2.3)$$

(2.1) ve (2.3) bağıntılarını kullanarak simge hızı  $R_s$  değerini  $R$  bit hızı cinsinden şu şekilde yazabiliriz:

$$R_s = \frac{R}{\log_2 M} \quad (2.4)$$

(2.2) ve (2.3) bağıntılarından yararlanarak,  $T_s$  süresi boyunca " $m=\log_2 M$ " tane bit ileten  $W$ (Hz) bant genişlikli bir haberleşme sisteminin bant genişliği verimi:

$$\frac{R}{W} = \frac{\log_2 M}{WT_s} = \frac{1}{WT_b} \quad (\text{bit/sn/Hz}) \quad (2.5)$$

şeklinde tanımlanabilir.

(2.5) bağıntısında,  $WT_b$  çarpanı ne kadar küçük olursa, haberleşme sisteminin bant verimliliği o kadar iyi olur. Burada bir örnek olarak MPSK (M-ary phase shift keying = M'li faz kaydırmalı anahtarlama) modülasyonunu düşünelim. Bu modülasyon türü için gerekli bant genişliği:

$$W = \frac{1}{T_s} = R_s \quad (2.6)$$

bağıntısıyla verilir.

Burada  $T_s$  simge süresi,  $R_s$  ise simge hızıdır. (2.5) ve (2.6) bağıntılarının yardımıyla MPSK modülasyonu için bant genişliği verimi:

$$\frac{R}{W} = \log_2 M \text{ (bit/sn/Hz)} \quad (2.7)$$

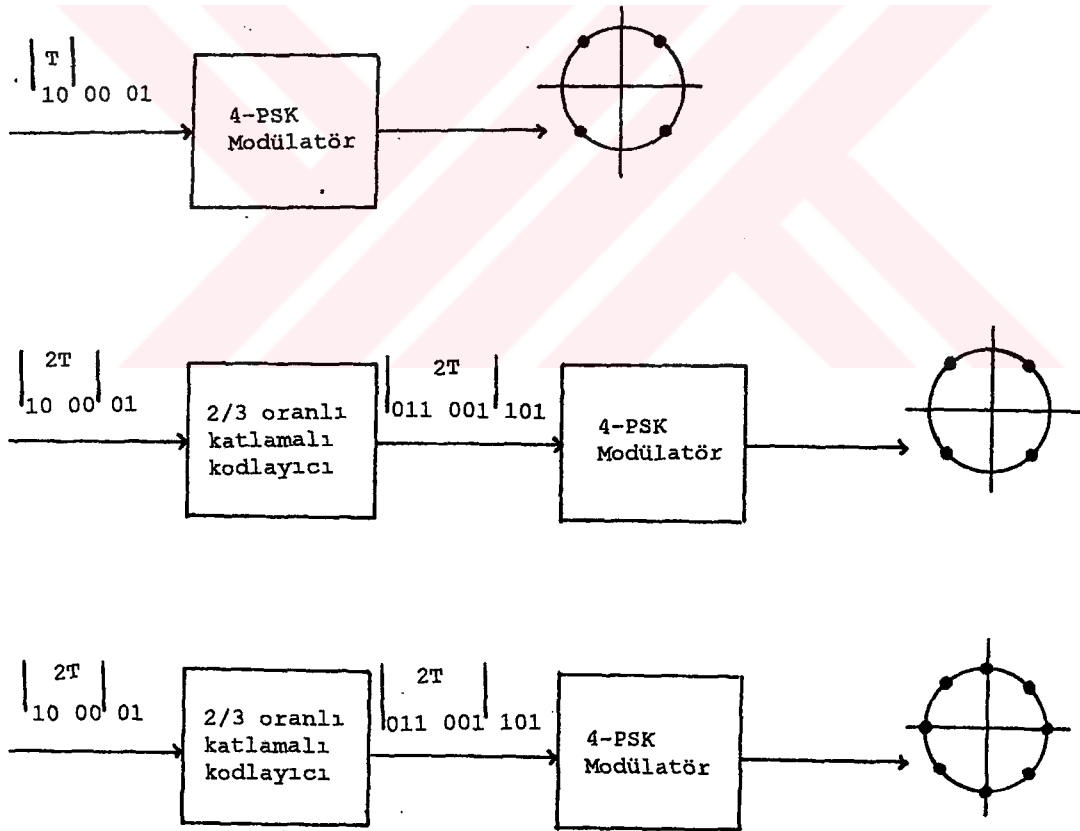
şeklinde elde edilecektir. Burada M'in artmasıyla R/W değerinin de artacağı görülmektedir. Bu da  $E_b/N_0$  (bit başına enerji/gürültü gücü) değerinin artmasına neden olacaktır. Dolayısıyla şu sonucu çıkarabiliriz: MPSK modülasyon türü,  $E_b/N_0$  değerinin artması pahasına bant verimliliğini arttırmaktadır.

Bir haberleşme sistemine klasik hata düzeltme kodlaması işlemi uygulandığında, aynı veri hızı ve aynı modülasyon türü için kodlamasız duruma göre bant genişliği artacaktır. Kodlama; k bitlik enformasyon bit dizisine n-k tane kontrol bitinin eklenmesiyle yapılır. Ancak, kodlama işlemi, kodlama oranına ters orantılı olarak bant genişliğinin artmasına neden olur. Aslında bu yöntemle, bant genişliğindeki artmanın yanısıra bir kodlama kazancı da sağlanmaktadır. Bu durumda, bant genişliğinde bir artma olmaksızın aynı zamanda önemli bir kodlama kazancı sağlayan bir kodun olup olmadığı sorusuyla karşılaşılabılır. Kodlama ve modülasyonun birlikte düşünüldüğü bir yöntem, bu soruya çözüm olarak karşımıza çıkmaktadır. İlk kez Ungerboeck [Ungerboeck,1982] tarafından ortaya atılan, kodlama ve modülasyon işlemlerinin tasarım sırasında birlikte düşünüldüğü yöntem; kafes kodlamalı modülasyon (Trellis Coded Modulation = TCM) olarak bilinmektedir. Bu yöntemin temeli, kullanılacak olan kodlayıcı düzenlerinde işaret kümesine fazladan simgeler eklenmesi, yani bu kümenin genişlemesi ilkesine dayanır. Böylelikle iletilecek işaret dizileri arasındaki Öklit uzaklığı da arttırılabilmektedir. Bu da sonuçta hata olasılığını düşürecektir.

### 2.1. Kafes Kodlamalı Modülasyon (TCM)

Kafes kodlamalı modülasyon (TCM), bant genişliğini veya

iletim gücünü arttırmadan sayısal iletim sisteminin güvenilirliğini arttırmak amacıyla, kodlama ile modülasyonun birlikte düşünüldüğü bir tekniktir. Güç sınırlı ortamda, mümkün olan en düşük güç ile sistem başarımının artırılması istenir. Bu durumda çözüm olarak, hata düzeltme kodlarının kullanılması düşünülebilir ki bu, iletilecek bit dizisine fazladan bitlerin eklenmesi nedeniyle güç verimliliğini artırır, ancak daha yüksek iletim hızına, dolayısıyla daha fazla bant genişliği gereksinimine neden olur. Bant genişliği sınırlı ortamda, bant verimliliğinin artırılması, daha yüksek dereceden modülasyon düzenlerinin seçilmesiyle sağlanabilir (örneğin 4-PSK yerine 8-PSK işaret kümesinin seçilmesi gibi).



(Şekil 2.1) Her T saniyede 2 bitlik veriyi ileten üç farklı haberleşme sistemi

TCM kavramını daha iyi anlamak için şöyle bir örnek düşünülebilir: Her T saniyede iki enformasyon biti ileten bir kaynaktan bilgi işareti gönderen bir sayısal haberleşme sistemi olduğu varsayalım. Bu durumda çeşitli seçenekler düşünülebilir:

a) Her T saniyede bir işaret gönderildiği, kodlamasız 4-PSK modülasyon kullanılabilir. Bu durumda iletilen her bir işaret iki enformasyon biti taşır (Şekil 2.1.a).

b) 2/3 oranlı katlamalı kodlama ile 4-PSK modülasyonu kullanılabilir. Bu durumda her bir işaret 4/3 enformasyon biti taşıyacaktır. Bu nedenle kaynak enformasyon iletim hızının karşılanabilmesi için, bir işaretin 2T/3 süreye sahip olması gerekecektir. Bu da kodlamasız sisteme göre, bant genişliğinin 3/2 katına çıkmasını belirtir (Şekil 2.1.b).

c) İşaret süresinin azalmasını önlemek için, yine 2/3 oranlı bir katlamalı kodlayıcı, ama bu kez 8-PSK modülasyonu kullanılabilir. Bu durumda her bir işaret iki enformasyon biti taşıyacak ve dolayısıyla bant genişliğinde bir artma olmayacaktır (Şekil 2.1.c)

c) seçeneğinde kodlamanın, bant genişliğinde bir artma olmadan yapıldığını görmekteyiz. Burada, yüksek dereceden işaret kümesinin kullanılmasıyla, 4-PSK'ya göre güçten bir kaybın olması beklenebilir. Bu nedenle, 2/3 oranlı kodlayıcının bu kaybı dengelemesi ile, kodlama kazancı sağlanır. Sonuç olarak, bant genişliğinden ödün vermeksizin kodlama kazancı elde edilmiş olmaktadır.

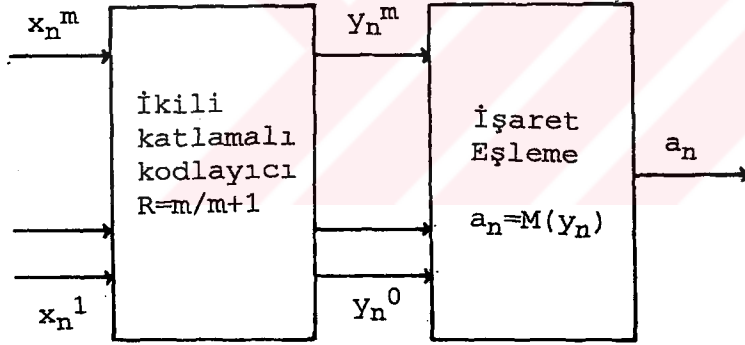
Bu noktadan hareketle, TCM tekniğiyle tasarımda, katlamalı kodlama ve modülasyonun iki ayrı işlem yerine, bir bütün olarak düşünüleceği söylenebilir. Sonuçta, alıcı tarafa ulaşan işaretin ilk önce demodüle edilip, sonra kod çözümü yerine, alıcının demodülasyon ve kodçözmeyi tek adımda birleştirdiği bir işleme uygulanır. Dolayısıyla burada sistem başarımını etkileyen parametre, artık katlamalı kodlayıcının serbest Hamming uzaklığı değil, toplamsal Beyaz Gauus kanalı



üzerinden iletilen işaretler arasındaki serbest Öklit uzaklığı olacaktır. Dolayısıyla TCM'in optimizasyonu Hamming uzaklığına göre değil, Öklit uzaklığına dayanarak gerçekleştirilecektir.

### 2.1.1. Kafes Kodlamalı Modülasyonda Kodlama Ve İşaret Eşleme

Kodlamasız sistemde, her T modülasyon aralığında m veri biti iletilir ve  $2^m$  tane işaretten biri kanala gönderilir. Oysa, kafes modülasyonlu sistemde her T modülasyon aralığında  $2^{m+1}$  tane işaretten biri kanala gönderilmektedir. Dolayısıyla işaret kümesi  $[(2^{m+1})/(2^m)]=2$  kat genişleyecektir ve simgeler birbirlerine yakınlaşmalarına karşın, simge dizileri birbirlerinden uzaklaştırılabileceklerdir. Bu durumda kodlama oranı  $R=m/(m+1)$  olmaktadır. Kodlayıcı yapısı (Şekil 2.2)'de gösterilmiştir.



(Şekil 2.2) Çoklu kodlayıcı yapısı

Kanal işaretleri  $a_n$  ve  $b_n$  arasındaki minimum Öklit uzaklığı  $d(a_n, b_n)$  ile gösterilsin. Bu işaretler arasındaki minimum serbest Öklit uzaklığı şu bağıntı ile verilir:

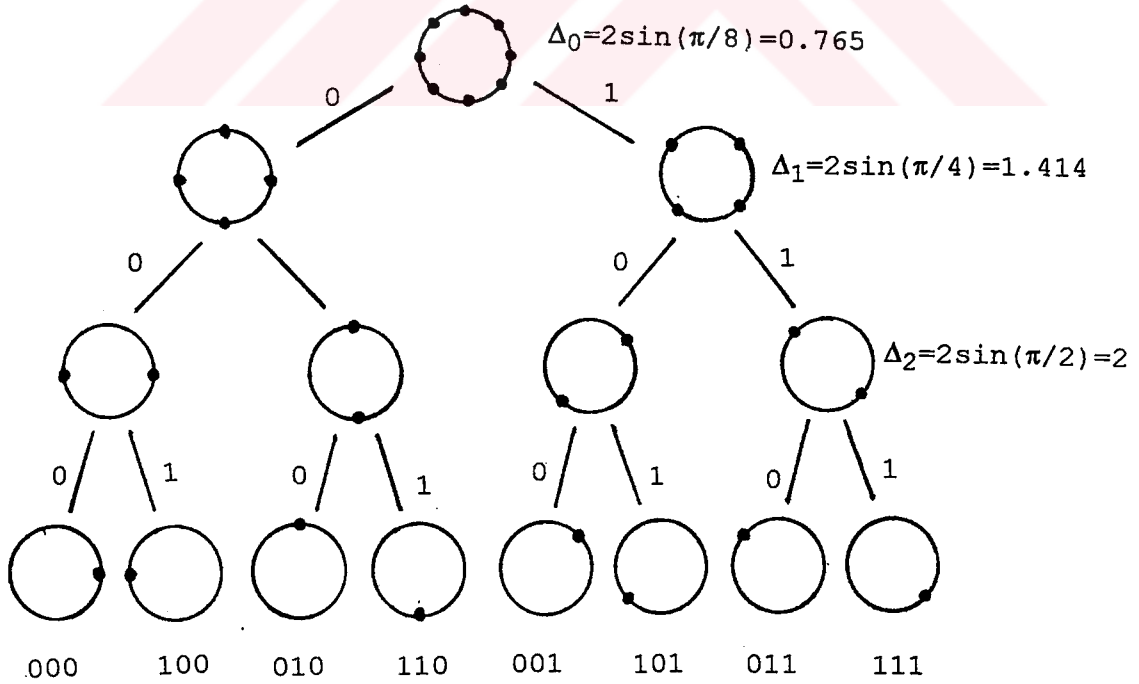
$$d_f = \min_{a_n \neq b_n} \left[ \sum_n d^2(a_n, b_n) \right]^{1/2} \quad (2.8)$$

Kodlayıcı tasarımı, sistemin hata başarımının arttırılması için  $\{a_n\}$  ve  $\{b_n\}$  işaret dizileri arasındaki uzaklığın maksimum olması istenir. Eğer alıcı tarafta en büyük benzerlikli kod çözme kullanılırsa, yüksek işaret - gürültü (SNR) oranında hata olasılığı asimptotik olarak  $\Pr(e)$ 'nin alt sınırına yaklaşacaktır.

$$\Pr(e) \geq N(d_f) \cdot Q(d_f/2\sigma) \quad (2.9)$$

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} \exp(-y^2/2) dy \quad (2.10)$$

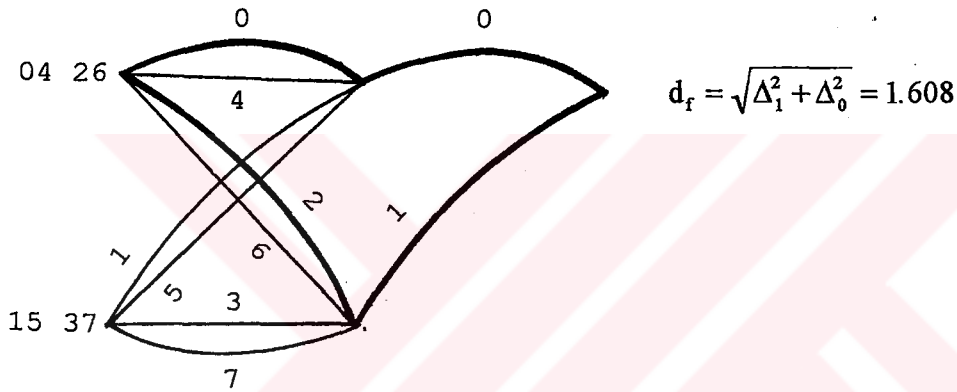
Burada  $Q(\cdot)$ , Gauss hata işlevini belirtir.  $N(d_f)$  ise,  $d_f$  uzaklığına sahip, kodlayıcı durum diyagramında herhangi bir durumdan ayrılıp, bir ya da daha fazla geçişten sonra yeniden birleşen işaret dizi çiftlerinin sayısını göstermektedir.



(Şekil 2.3) 8-PSK işaret kümesinin bölmelenmesi

Serbest uzaklık  $d_f$ 'i maksimum yapmayı amaçlayan TCM yöntemi, küme bölünmesi ile işaret eşleme kuralına dayanır. Bu yöntem, kanal işareti kümesinin, aralarındaki minimum uzaklıkların arttığı alt kümelere bölünmesi şeklinde gerçekleşir ( $\Delta_0 \leq \Delta_1 \leq \Delta_2 \dots$ ). Bu kavram 8-PSK modülasyonu için (Şekil 2.3)'de gösterilmiştir.

Eğer kodlayıcıda her T modülasyon aralığında m tane bit kodlanırsa, bu koda ilişkin durum diagramında, her bir durumdan bir sonraki duruma  $2^m$  tane geçiş olmalıdır.



(Şekil 2.4) 8-PSK modülasyonuna ilişkin 2 durumlu kafes yapı

Kodlayıcı için uygun bir durum geçiş diagramı seçildikten sonra yapılacak iş, serbest uzaklığı maksimum yapacak,  $2^{m+1}$ 'lik kümeden kanal işaretlerinin belirlenmesidir. 2 durumlu 8-PSK modülasyonu için kafes diagramı (Şekil 2.4)'de görülmektedir.

TCM tekniği ile tasarlanmış bir sisteme ait asimptotik kodlama kazancı:

$$AKK = 10 \log \frac{(d_{f,ED}^2 / E_s)_{kodlamalı}}{(d_{f,ED}^2 / E_s)_{kodlamasız}} \quad (2.11)$$

dır.

Burada " $(d_f^2)_{kodlamalı}$ " karşılaştırma yapılacak olan referans sistemin serbest Öklit uzaklığını göstermektedir. Örneğin

kodlamasız 4-PSK için serbest Öklit uzaklığı  $(d_f^2)_{kodlamasız} = 2$ 'dir.  $E_s$  ise sisteme ait olan enerjiyi belirtir. PSK modülasyonu için bu ifade 1'e eşittir.

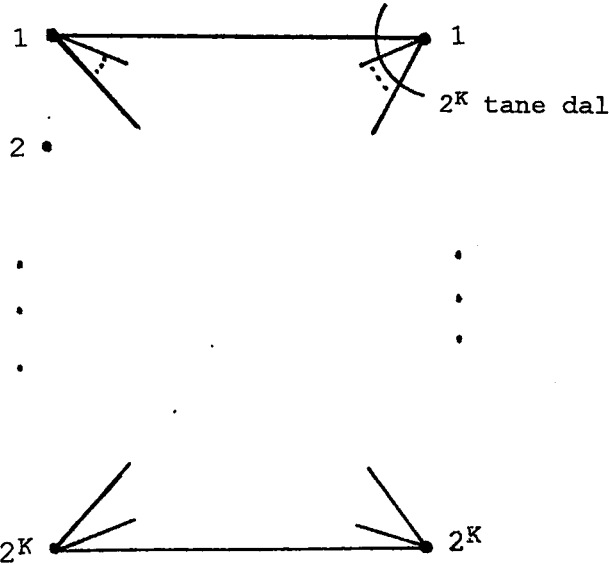
Kafes diagramı (Şekil 2.4)'de gösterilen koda ilişkin asimptotik kodlama kazancı, (2.11) bağıntısına göre

$$10 \log \frac{(1.608)^2}{2} = 1.1 \text{ dB} \text{ olarak bulunur.}$$

TCM tekniğiyle tasarlanmış kodlarla, PSK modülasyonu için 6.02 dB'ye varan kodlama kazançları elde edilebilmektedir [Ungerboeck,1987]. Fakat TCM'in başlıca sakıncası, kodlama kazancı artarken sisteme ait karmaşıklığın da üstel olarak artmasıdır. Bir sonraki alt bölümde TCM için kodçözme karmaşıklığı tanımı verilmektedir.

### 2.1.2. TCM İçin Kodçözme Karmaşıklığı

Bir haberleşme sistemi için alınan kanal işaretinin kodçözme işlemi sırasında yapılan karşılaştırma sayısı, o sistem için karmaşıklık sayısını belirtir.  $k/n$  oranlı Ungerboeck tipi kodlayıcıya ilişkin kafes diyagramı, genel olarak (Şekil 2.5)'deki gibi gösterilebilir.



(Şekil 2.5)  $k/n$  oranlı kodlayıcıya ilişkin kafes yapı

(Şekil 2.5)'den de görüldüğü gibi  $k$  girişli bir kodlayıcıya ilişkin kafes diyagramda, bir duruma  $2^k$  tane dal gelmektedir.

Kodçözme algoritması olarak Viterbi Algoritması [Viterbi,1973] kullanılırsa, herbir durumdaki enküçük yol uzaklığını bulmak için  $2^k-1$  tane karşılaştırma yapmak gerekmektedir. Eğer kodlayıcının bellek elemanı sayısı (constraint length) sayısı  $K$  ise kafes diyagramda  $2^K$  tane durum bulunacak ve kodçözme karmaşıklığı:

$$L=2^K(2^k-1) \quad (2.12)$$

olacaktır. Dolayısıyla yüksek oranlı kodlayıcılar için kodçözme karmaşıklığı, giriş biti sayısına bağlı olarak üstel olarak artmaktadır. Kodçözme karmaşıklığının azaltılması için daha değişik yapıdaki delikli (punctured) katlamalı kodlar kullanılırlar [Yasuda et al,1984].

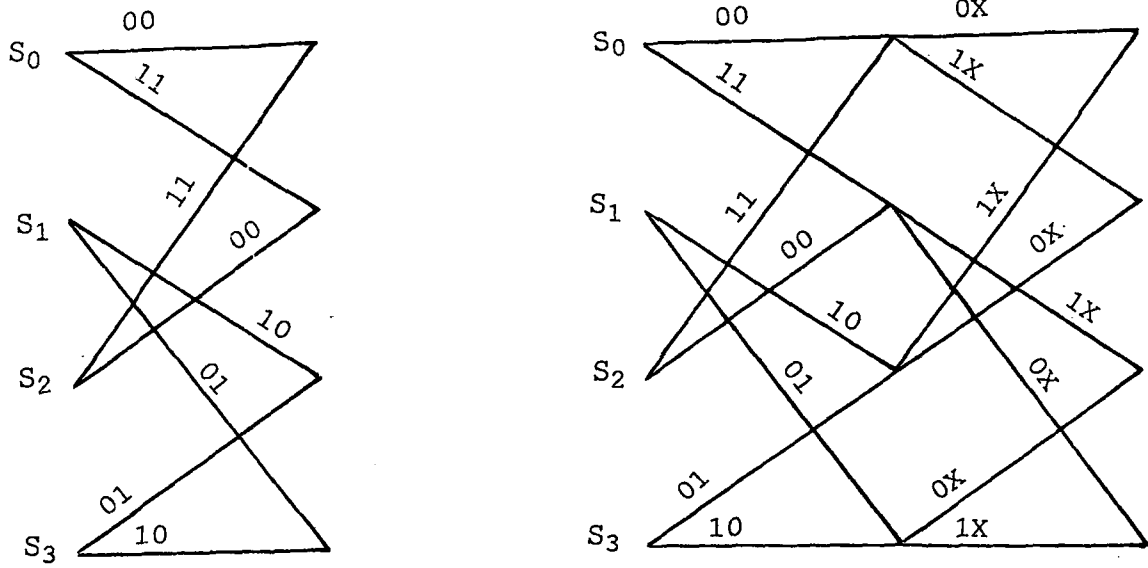
#### 2.1.2.1. Delikli Katlamalı Kodlar İçin Kodçözme Karmaşıklığı

Delikli katlamalı kodlar, düşük oranlı kodlayıcıların çıkışlarındaki bazı bitlerin periyodik olarak silinmesiyle oluşturulurlar. Çıkışlarındaki bitlerin silindiği düşük oranlı kod, 'kaynak kod' olarak isimlendirilir ve genel olarak  $R=1/2$  oranlı kodlardan seçilirler. Kaynak kodun çıkışındaki bitlerden hangilerinin silineceği, delikli koda ait üreteç matrisi  $[P]$  ile belirlenir.

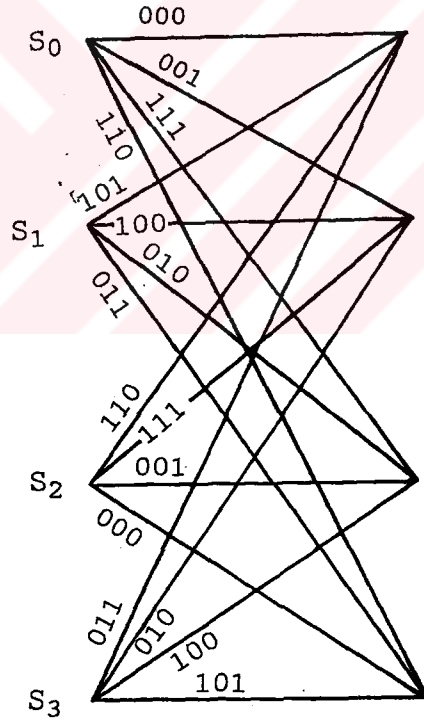
Örnek olarak,  $R=1/2$  oranlı bir kaynak kodu düşünölsün. Buna ilişkin kafes diyagramı (Şekil 2.6)'da gösterilmiştir. Üreteç matrisi,

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

şeklinde olan  $2/3$  oranlı delikli koda ilişkin kafes yapı ise (Şekil 2.7)'de gösterilmiştir.



(Şekil 2.6)  $R=1/2$  oranlı katlamalı kod ile  $2/3$  oranlı delikli katlamalı kodun elde edilmesi



(Şekil 2.7)  $R=2/3$  oranlı delikli katlamalı koda ilişkin kafes diyagram

(Şekil 2.6) ve (Şekil 2.7)'de verilen kafes diyagramlar aslında birbirlerinin eşdeğeridir. Boşluklu katlamalı kodların kafes diyagramlarında her bir duruma 2 dal gelmesi nedeniyle; Viterbi kodçözme algoritması için, her bir

simgenin çözülmesi  $2^1-1=1$  karşılaştırma gerektirir. Eğer kafes diyagramda  $2^K$  tane durum varsa,  $2^{K.1}=2^K$  tane karşılaştırma yapılır. Genel olarak, k/n oranlı delikli katlamalı kodlayıcıda k tane bilgi bitinin çözülmesi için  $2^K.k$  tane karşılaştırma yapılması gerekmektedir. Dolayısıyla, k/n oranlı delikli katlamalı kodlayıcının kodçözme karmaşıklığı:

$$L=2^K k \quad (2.13)$$

bağıntısıyla verilir.

Karmaşıklık sayısının TCM sistemleri için önemli bir sakınca olması ve bu nedenle delikli katlamalı kodlayıcıların kullanılmasının yanısıra, son yıllarda karmaşıklık sayısı açısından daha verimli olan çok düzeyli kodlayıcılar bir ilgi odağı haline gelmiştir.

## 2.2. Çok Düzeyli Kodlayıcılar

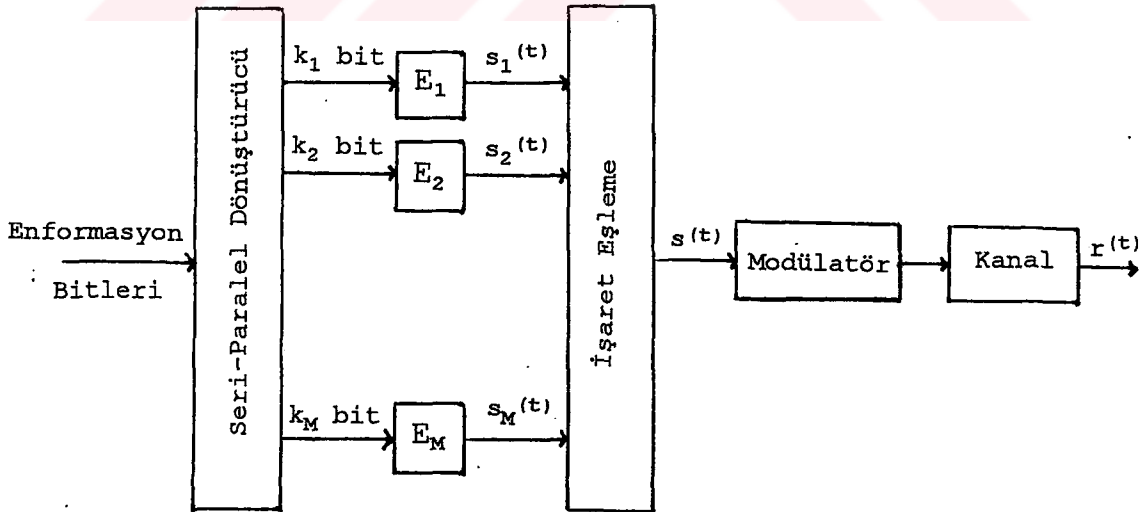
Haberleşme sistemlerinde, bant genişliğinden ödün vermeksizin iletim hızının arttırılması için, tasarım sırasında kodlama ve modülasyonun bir bütün olarak düşünülmesi gerektiği bilinmektedir. Özellikle, yüksek iletim hızı ve yüksek hata başarımına sahip olması nedeniyle çok düzeyli kodlayıcılar son yıllarda ilgi odağı haline gelmiştir. Çok düzeyli kodlayıcıların en önemli üstünlüğü, çok aşamalı kodçözme yönteminin kullanılmasına olanak vermesi ve bunun da kodçözme karmaşıklığını azaltmasıdır.

Çok düzeyli kodlama ve çok aşamalı kodçözme ilk olarak Imai ve Hirakawa [Imai et al,1977] tarafından önerilmiştir. Bu yöntemde, her düzeyde hata düzeltme kodu olarak blok kodlar kullanılmakta ve her düzeydeki kodlayıcının çıkış bitlerinden 1 tanesi kullanılarak iletilecek modülasyonlu işaret belirlenmektedir.

Daha sonra Pottie - Taylor [Pottie et al,1989] tarafından önerilen yöntemle, çok düzeyli kodlayıcılar daha genel bir yapıya sokulmuştur. Pottie-Taylor tipi kodlamada, Imai-Hirakawa tipi kodlama düzeninden farklı olarak, modülasyonlu işaretin belirlenmesinde her düzeydeki kodlayıcının 1'den fazla çıkış biti gözönüne alınabilmektedir. Ayrıca bu tip kodlamada her düzeyde farklı türden ikili kodlar kullanılabilir.

### 2.3. Imai - Hirakawa Tipi (IHS: Imai - Hirakawa Sistemi) Kodlama

IHS tipi kodlayıcı sisteminin genel yapısı (Şekil 2.8)'de gösterilmiştir. Bu sistem, M-düzelili kodlayıcı için M tane kodlayıcı kullanmakta ve her düzeydeki ikili kodlayıcı çıkışlarının birer biti, T aralığında iletilecek modülasyonlu işareti belirlemektedir. Bu işleme, işaret eşleme denilir ve eşleme yapılan işaret kümesi, M düzeyli kodlayıcı için  $2^M$  tane simgeden oluşur.



(Şekil 2.8) M düzeyli IHS tipi kodlayıcı

Kodlama için kullanılan yöntem şu şekilde tanımlanabilir: Çok düzeyli kodlayıcı, herbiri sırasıyla



$R_1, R_2, \dots, R_M$  kodlama oranlarına sahip  $C_1, C_2, \dots, C_M$  kodlarını içeren  $E_1, E_2, \dots, E_M$  kodlayıcılarından oluşmaktadır.  $C_i$  kodu,  $k_i$  enformasyon bitine karşılık  $n_i$  uzunluklu ikili dizi üretir, dolayısıyla kodlama oranı  $R_i = k_i/n_i$  ile belirtilir. Tüm sistemin kodlama oranı ise:

$$R = \sum_{i=1}^M \frac{R_i}{M} \quad (2.14)$$

bağıntısıyla tanımlanır. İletilecek enformasyon dizisi ilk olarak, seri - paralel dönüştürücü yardımıyla, her düzeye  $k_i$  ( $i=1, 2, \dots, M$ ) tane bit gelecek şekilde  $M$  düzeye ayrılır. Daha sonra  $i$ .nci düzeydeki enformasyon biti  $E_i$  kodlayıcısına girer.  $t$  bir tamsayı belirtmek üzere, herhangi bir  $t$  anındaki  $E_i$  kodlayıcısının çıkışı  $s_i^{(t)}$  ile gösterilirse, kodlayıcının son bölümündeki işaret eşleme ile

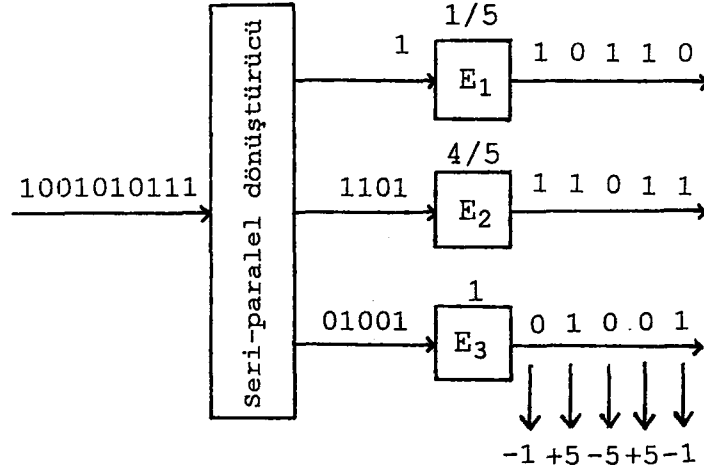
$$s(t) = \sum_{i=1}^M s_i^{(t)} \cdot 2^{i-1} \quad (2.15)$$

çıkış işareti elde edilir. Burada  $s(t)$   $2^M$ 'lik işaretten birini belirtmektedir.

IHS tipi kodlayıcıda modülasyonlu işaret, küme bölmelenmesi yardımıyla işaret eşleme yöntemine göre elde edilir. Şimdi bu yöntemi inceleyelim.

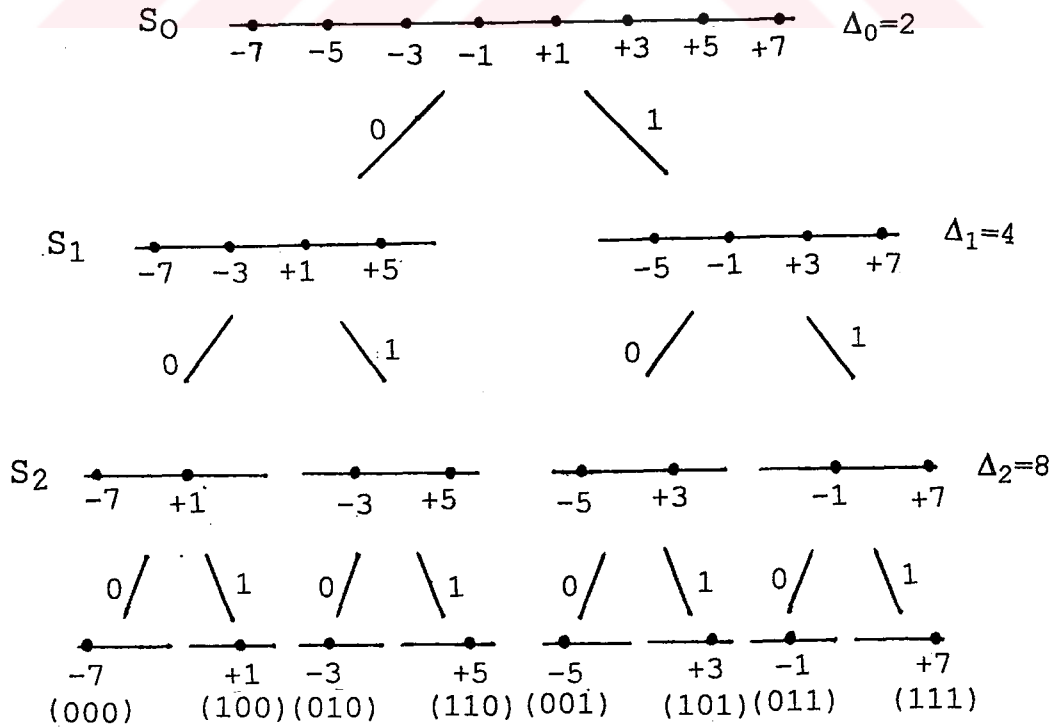
### 2.3.1. İşaret Kümesinin Bölünmesine Dayalı İşaret Eşleme

Kodlayıcı çıkışında iletilecek olan modülasyonlu işaretlerin kümesi  $S_0$  olsun.  $M$  düzeyli kodlayıcı için, bu  $S_0$  kümesi  $2^M$  adet simgeden oluşacaktır.  $S_0$  kümesinin,  $2^{M-1}$  elemanlı iki alt kümeye bölüdüğü düşünölsün (bu iki alt kümenin birleşimi yine  $S_0$ 'ı oluşturacaktır) ve bu bölünme  $S_0/S_1$  şeklinde gösterilsin.  $M$  düzeyli bir kodlayıcı yapısı için bu bölünme işlemi  $S_0/S_1/S_2/\dots/S_M$  şeklinde devam eder.



(Şekil 2.9) 3 düzeyli IHS tipi kodlayıcı

$S_{i-1}/S_i$  küme bölünmesi ile  $C_i(S_{i-1}/S_i; E_i)$  kodu arasında bir ilişki kurulabilir. Burda  $E_i$  kodlayıcıyı göstermektedir.  $i$ .nci düzeydeki ( $i=1,2,\dots,M$ )  $E_i$  kodlayıcısının çıkış bitlerinin 0 veya 1 olmasına göre  $S_i$  alt kümelerinden biri belirlenir. En. son düzeydeki  $E_M$  kodlayıcısı iletilecek modülasyonlu işareti belirler.



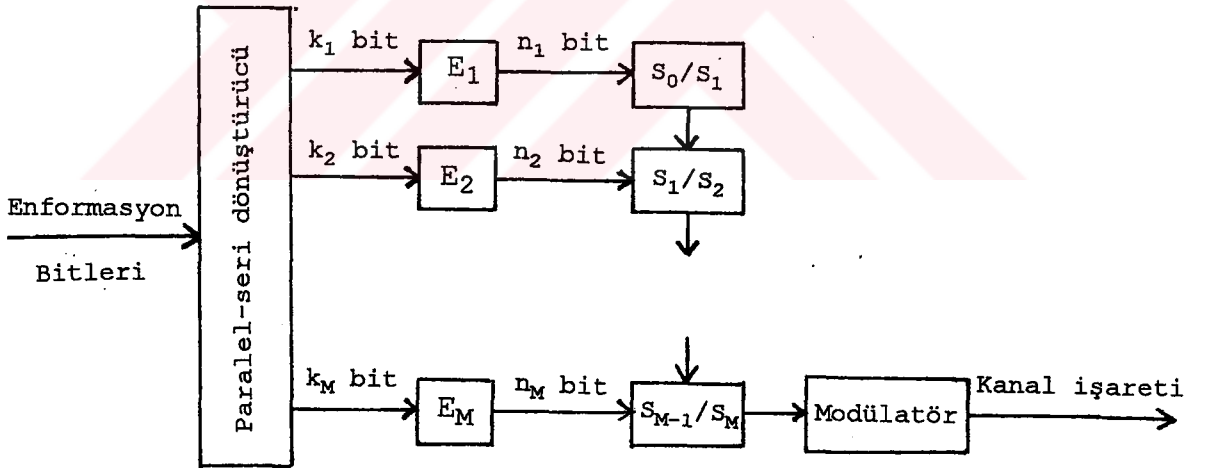
(Şekil 2.10) 8-AM işaret kümesinin bölmelenmesi

Bu bölünme zinciri içindeki herhangi bir  $S_i$  için,  $S_i$ 'ye ait alt kümeler içindeki işaretler arasındaki uzaklıkların en küçüğü  $\Delta_i$  olarak belirtilir ve i.nci alt küme uzaklığı olarak bilinir. Çok düzeyli kodlamada amaç, her işaretleme aralığında işaret kümesi bölünürken,  $\Delta_i$  uzaklıklarının da artmasını sağlamaktır. Yani her düzeydeki alt küme uzaklıklarının  $\Delta_0 \leq \Delta_1 \leq \Delta_2 \leq \dots \leq \Delta_M$  şeklinde giderek artması gerekmektedir.

(Şekil 2.9), 3 düzeyli bir örnek kodlayıcı yapısını, (Şekil 2.10) ise 8-AM işaret kümesinin bölmelenmesini göstermektedir.

#### 2.4. Pottie - Taylor Tipi Kodlama

(Şekil 2.11), Pottie - Taylor tipi kodlayıcı düzenini göstermektedir.



(Şekil 2.11) Pottie - Taylor tipi M-düzeyle kodlayıcı yapısı

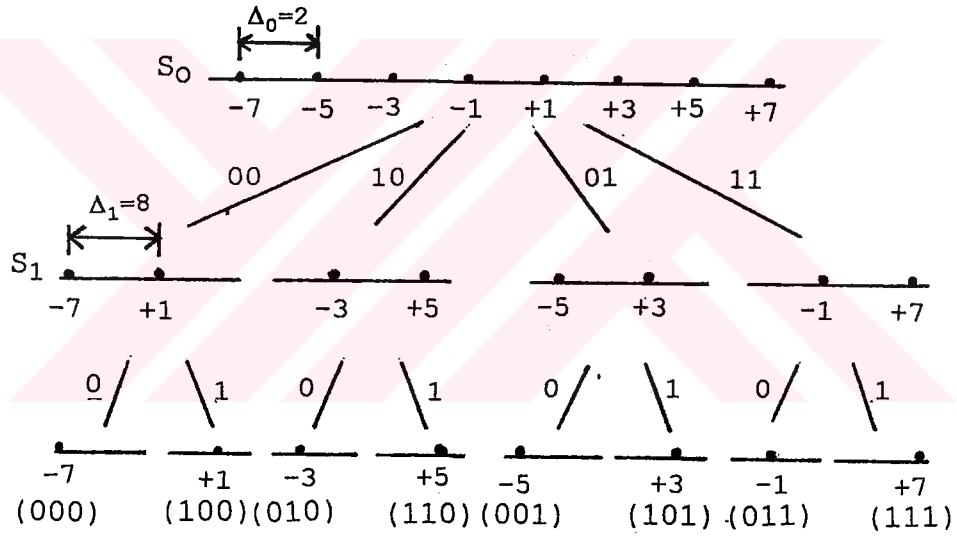
Bu düzen de IHS tipi kodlayıcıya benzemektedir. Enformasyon bitleri seri - paralel dönüştürücü ile  $k_i$  ( $i=1,2,\dots,M$ ) uzunluklu bitler şeklinde M düzeye ayrılırlar. Her düzeydeki  $R_i=k_i/n_i$  kodlama oranına sahip  $E_i$  kodlayıcısı bir  $C_i$  kodunu kullanır. Tüm sisteme ait kodlama oranı ise:

$$R = \frac{\sum_{i=1}^M k_i}{\sum_{i=1}^M n_i} \quad (2.16)$$

bağıntısıyla tanımlanır.

IHS tipi kodlayıcıdan farklı olarak bu tür kodlayıcı yapıda, modülasyonlu işaret belirlenirken, her düzeydeki kodlayıcının birden fazla biti etkin olabilmektedir. Modülasyonlu işaret, IHS sisteminde olduğu gibi, küme bölmelenmesi yardımıyla işaret eşleme yöntemine göre elde edilmektedir.

2-düzeyle Pottie - Taylor tipi kodlayıcı için 8-AM işaret kümesinin bölmelenmesi (Şekil 2.12)'de gösterilmiştir. Burada, 1. düzeydeki kodlayıcının  $R_1=1/2$  oranlı, 2. düzeyde ise kodlama yapılmadığı dolayısıyla  $R_2=1$  olduğu düşünülmüştür.



(Şekil 2.12) Pottie-Taylor tipi kodlayıcı için 8-AM işaret kümesinin bölmelenmesi

## 2.5. Çok Düzeyli Kodlayıcılar İçin Serbest Öklit Uzaklığı ve Asimptotik Kodlama Kazancı

Serbest Öklit uzaklığı, katlamalı kodlayıcılar ile elde edilen kodlanmış işaret dizileri arasındaki Öklit uzaklıklarının en küçüğü olarak tanımlanabilir. Çok düzeyli kodlayıcılarda, her düzeydeki  $C_i$  kodunun bir katlamalı kod olduğu gözönünde bulundurulursa, bu kodlara ilişkin serbest

Hamming uzaklıklarından ( $d_{H(i)}$ ) yararlanarak tüm sistemin serbest Öklit uzaklığını bulabiliriz.

$C_i$  koduna ilişkin serbest Hamming uzaklığı ( $d_{H(i)}$ ) ve küme bölmeleyerek eşleme yöntemindeki her bölmelemede en küçük alt küme uzaklıkları  $\Delta_{i-1}$  ( $i=1,2,\dots,M$ ) ile gösterilmek üzere,  $i$ .nci düzeydeki kodlayıcıya ilişkin serbest Öklit uzaklığı:

$$d_{f,ED(i)} = \sqrt{d_{H(i)} \cdot \Delta_{i-1}} \quad (2.17)$$

bağıntısı ile elde edilir.

Tüm sisteme ait serbest Öklit uzaklığı ise:

$$d_{f,ED} = \min_i (d_{f,ED(i)}) \quad (2.18)$$

bağıntısı ile tanımlanır.

Çok düzeyli kodlayıcılarda işaretler, küme bölmeleyerek eşleme yöntemine göre belirlendiği için, her bölmelemedeki alt küme uzaklığı  $\Delta_i$ , bir öncekine eşit veya büyük olacaktır. ( $\Delta_0 < \Delta_1 < \Delta_2 < \dots < \Delta_M$ ). Bu durum (2.9) bağıntısı ile birlikte düşünüldüğünde, daha alt düzeylerde karmaşıklığı daha az olan kodların kullanılabilmesi ortaya çıkmaktadır.

Çok düzeyli kodlayıcıya ilişkin asimptotik kodlama kazancı:

$$AKK = 10 \log \frac{(d_{f,ED}^2 / E_b)_{kodlamalı}}{(d_{f,ED}^2 / E_b)_{kodlamasız}} \quad (2.19)$$

ile verilir.  $(d_{f,ED}^2)_{kodlamalı}$  ve  $(d_{f,ED}^2)_{kodlamasız}$ , sırasıyla çok düzeyli kodlayıcının ve referans olarak alınan ilişkin kodlamasız yapının serbest Öklit uzaklıklarını göstermektedir.  $(E_b)_{kodlamalı}$  ve  $(E_b)_{kodlamasız}$  ise sırasıyla kodlamalı ve kodlamasız sistemlere ilişkin enerjiyi belirtmektedir.

## 2.6. Çok Düzeyli Kodlayıcılar İçin Karmaşıklık Hesabı

Kodçözme algoritması olarak Viterbi Algoritması kullanılması durumunda, alınan bir kanal işaretini çözmek için, her bir durumdaki en küçük yol uzunluğunu bulmak amacıyla yapılan karşılaştırma sayısı, kullanılan kodlayıcı için karmaşıklığı belirtir.

$R_i = k_i/n_i$  oranlı bir katlamalı kodlayıcı yapısında  $K_i$  tane bellek elemanı bulunuyorsa, bu koda ilişkin karmaşıklık:

$$L_i = 2^{K_i} (2^{k_i} - 1) r_i / n_i \quad (2.20)$$

bağıntısıyla verilir. Burada  $r_i$  bir kodsözcüğünü belirtmek için kodlayıcı çıkışından alınan bit sayısını belirtmektedir.

IHS tipi kodlayıcılarda bir kodsözcüğünün belirlenmesinde her kodlayıcının yalnız bir biti etkin olduğundan  $r_i = 1$  olacak ve karmaşıklık:

$$L_i = 2^{K_i} (2^{k_i} - 1) / n_i \quad (2.21)$$

şeklinde olacaktır.

Pottie - Taylor tipi kodlayıcılarda ise, bir kodsözcüğü için her kodlayıcının tüm çıkışları etkin olduğundan  $r_i = n_i$ 'dir. Karmaşıklık ise:

$$L_i = 2^{K_i} (2^{k_i} - 1) \quad (2.22)$$

bağıntısı ile verilir.

Kodlayıcılar için önemli bir büyüklük olan karmaşıklık sayısı, büyük  $R_i$  değerleri için oldukça artacaktır. Karmaşıklığı azaltmak içinse, bir önceki bölümde de anlatıldığı gibi, delikli katlamalı kodlayıcılar kullanılırlar [Yasuda et al, 1984]. IHS tipi kodlayıcı için

i.nci düzeyde  $n_{i-1}/n_i$  oranlı delikli katlamalı kodlayıcı kullanılması durumunda karmaşıklık

$$L_i = 2^{K_i} (n_i - 1) / n_i \quad (2.23)$$

şeklinde belirtilir.

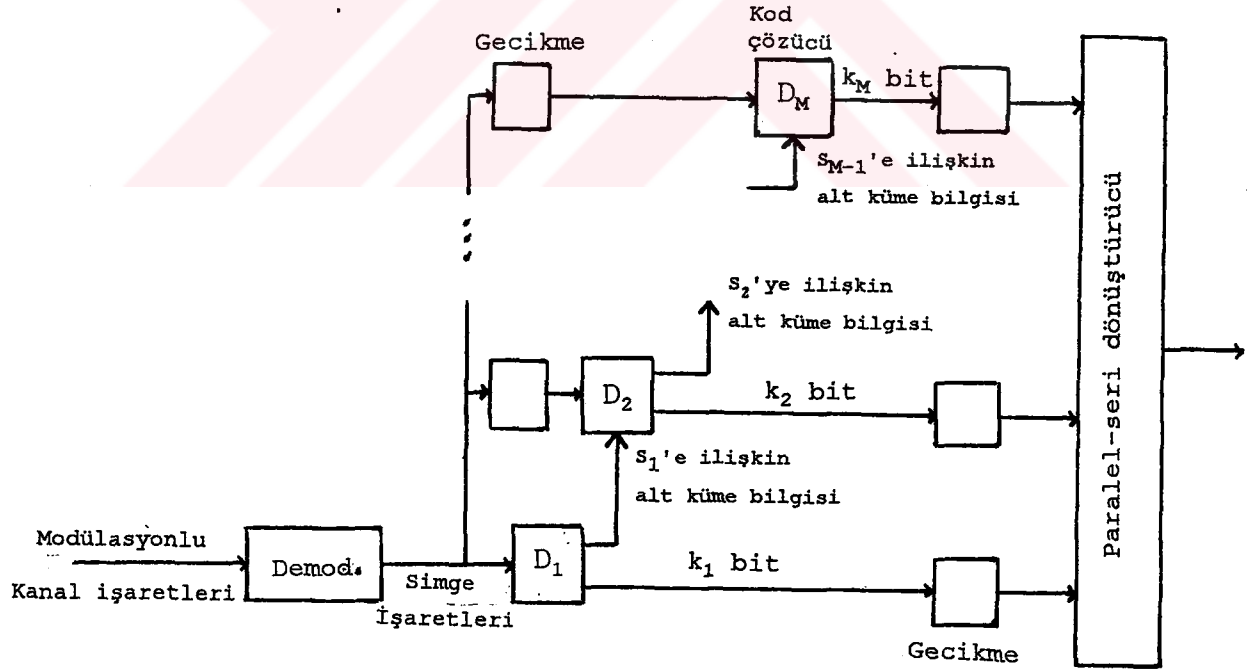
M düzeyli bir kodlayıcı için tüm sisteme ait karmaşıklık ise:

$$L = \sum_{i=1}^M L_i \quad (2.24)$$

eşitliğiyle tanımlanır.

Burada hemen belirtmek gerekir ki, eğer bir düzeyde kodlama yapılmıyorsa o düzey için karmaşıklık 0 olacaktır.

## 2.7. Çok Aşamalı Kod Çözme



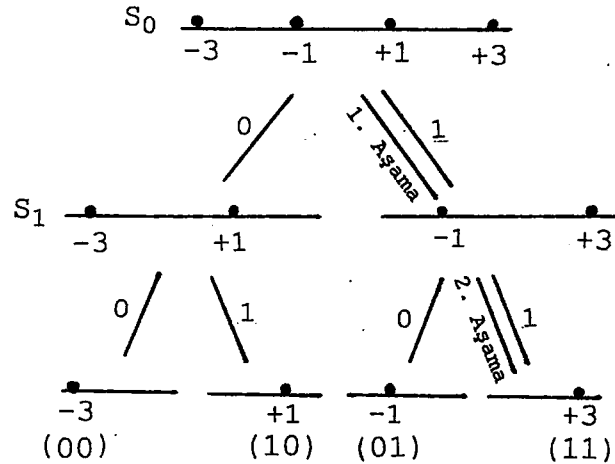
(Şekil 2.13) IHS tipi M-aşamalı kodçözücü yapısı

Çok düzeyli kodlayıcılar için alıcı tarafta, ilk kez Imai ve Hirakawa tarafından [Imai et al,1977] önerilen çok

aşamalı bir kodçözme yöntemi kullanılır. Bu yöntemle ilişkin bir kodçözücü yapısı (IHS) sistemi için (Şekil 2.13)'de gösterilmiştir.

$D_i$  ( $i=1,2,\dots,M$ ) ile gösterilen kodçözücü  $C_i$  koduna uygun bir yapıda olup, her  $D_i$  kodçözücüsü  $C_i$  kodunu çözer. Kodçözme işlemi şu mantığa göre yapılmaktadır:

Her işaretleme aralığı için,  $D_1$  kodçözücüsü alınan işaretin,  $S_1$ 'e ait alt kümelerden hangisine en yakın olduğunu belirleyerek,  $S_1$ 'e ilişkin alt küme bilgisini verir. Böylece işaret kümesinde ilgilenilen eleman sayısı yarıya indirilmiş olur.  $D_2$  kodçözücüsü ise  $D_1$ 'in verdiği karara ve alınan işarete bakarak  $S_2$ 'ye ilişkin alt küme bilgisini belirler. En sonunda da  $D_M$  kodçözücüsü,  $D_{M-1}$  kodçözücüsünün vermiş olduğu karara ve alınan işarete bakarak hangi işaretin gönderildiğini belirler. Her aşamadaki  $D_i$  kodçözücüleri, uygun kodçözme algoritmaları kullanarak (Viterbi Algoritması gibi),  $E_i$  kodlayıcısına ilişkin giriş ve çıkış bitlerini elde eder. Daha sonra paralel - seri dönüştürücü yardımıyla enformasyon bit dizileri belirlenir.



(Şekil 2.14) 4-AM işaret kümesi üzerindeki kodçözme aşamaları

(Şekil 2.14), 2 aşamalı bir kodçözücü için her aşamayı işaret kümesi üzerinde göstermektedir.



2 aşamalı bir kod çözücü için;  $D_1$  kod çözücüsü,  $E_1$  kodlayıcısının çıkışının 0 ya da 1 olduğuna karar verir. Dolayısıyla işaret kümesinde  $S_1$ 'e ait alt küme belirlenir. Böylece işaret kümesindeki eleman sayısı da yarıya inmiş olur.  $D_2$  kodçözücüsü ise, belirlenen bu karara ve alınan işarete bakarak  $E_2$  kodlayıcısının çıkış bitlerini belirler. Dolayısıyla 1. aşamada hangi alt küme belirlenmişse, 2. aşamada bu alt kümenin elemanlarından biri seçilerek gönderilen işaret bulunmuş olur. (Şekil 2.14)'de  $D_1$  kodçözücüsünün  $(-1,+3)$  elemanlarından oluşan alt kümeyi ve  $D_2$  kodçözücüsünün de  $(-1)$  işaretini seçtiği bir durum görülmektedir.

Burada hemen belirtmek gerekir ki, çok aşamalı kodçözme yönteminde ilk aşamada verilen alt küme bilgisine ilişkin kararların yanlış olması, daha sonraki aşamalarda da yanlış kararlar verilmesine neden olacaktır. Dolayısıyla bu tür bir sorunun çözümü için; kodlayıcı kısmında üst düzeylere alt düzeydekilere göre hata düzeltme yeteneği açısından daha güçlü kodlar yerleştirilir. Bu şekilde, hata başarımı daha iyi olan sistemler tasarlanabilmektedir.

### 3. KATLAMALI KODLAYICILAR KULLANILARAK SAYISAL GENLİK MODÜLASYONU İÇİN ÇOK DÜZEYLİ KODLAYICILARIN TASARIMI

Ungerboeck [Ungerboeck,1982] tarafından önerilen, kodlama ile modülasyonun tasarım sırasında birlikte düşünüldüğü ve modülasyonlu işaret kümesinin 2 kat genişletilip, küme bölmeleyerek eşleme yöntemi ile ikili sayılara eşleştirildiği TCM tekniği sayesinde bant verimliliğinden bir kayıp olmadan haberleşmenin olanaklı olduğu görülmüştür. 8-AM için Ungerboeck tipi kodlama ile kodlamasız 4-AM'e göre 3.3-5.8 dB kodlama kazançları elde edilmiştir [Ungerboeck,1987]. Bu yöntemde önemli olan bir nokta, kodlama kazancının atmasıyla birlikte kodçözme karmaşıklığının, kodlayıcıdaki bellek elemanı sayısına bağlı olarak üstel biçimde artmasıdır. Ungerboeck tarafından, katlamalı kodlayıcılar kullanılarak 4-AM ve 8-AM için elde edilen kodlayıcılar sırasıyla (Tablo 3.1.a) ve (Tablo 3.1.b)'de verilmiştir.

(Tablo 3.1.a) R=1/2 oranlı, 4-AM  
Ungerboeck tipi kodlar

No	K	AKK(dB)	L
Ung-1	2	2.5	4
Ung-2	3	3.0	8
Ung-3	4	3.4	16
Ung-4	5	4.2	32
Ung-5	6	4.5	64
Ung-6	7	5.1	128

(Tablo 3.1.b) R=1/2 oranlı, 8-AM  
Ungerboeck tipi kodlar

No	K	AKK(dB)	L
Ung-7	2	3.3	12
Ung-8	3	3.8	24
Ung-9	4	4.2	48
Ung-10	5	5.0	96
Ung-11	6	5.2	192
Ung-12	7	5.8	384

Tablolardaki asimptotik kodlama kazancı ve karmaşıklık değerleri sırasıyla (2.11) ve (2.12) bağıntılarından yararlanarak elde edilmiştir.

(Tablo 3.1.a) ve (Tablo 3.1.b)'den de görüldüğü gibi Ungerboeck tipi kodlama yönteminde kodlama kazancı artarken, kodçözme karmaşıklığı da çok büyük oranda artmaktadır.

Bant verimliliğinden kayıp olmaksızın, Ungerboeck'in elde ettiği kodlama kazançlarının sağlandığı, bunun yanısıra kodçözme karmaşıklığının daha düşük olarak elde edildiği yöntemlerden biri de 2. bölümde de sözü edilen ve ilk kez Imai ve Hirakawa [Imai et al,1977] tarafından önerilen çok düzeyli kodlama tekniğidir. Bir sonraki alt bölümde bu yönteme göre tasarlanmış kodlayıcılar incelenmiştir.

### 3.1. Sayısal Genlik Modülasyonu İçin 2 ve 3 Düzeyli IHS Tipi Kodlayıcılara İlişkin Tasarım Örnekleri

Literatürde var olan çeşitli katlamalı kodlayıcıların seçilmesiyle çok düzeyli kodlayıcılar oluşturulabilir. Bu bölümde 4-AM ve 8-AM için sırasıyla 1/2 ve 2/3 oranlarında 2 ve 3 düzeyli kodlayıcılar tasarlanmıştır.

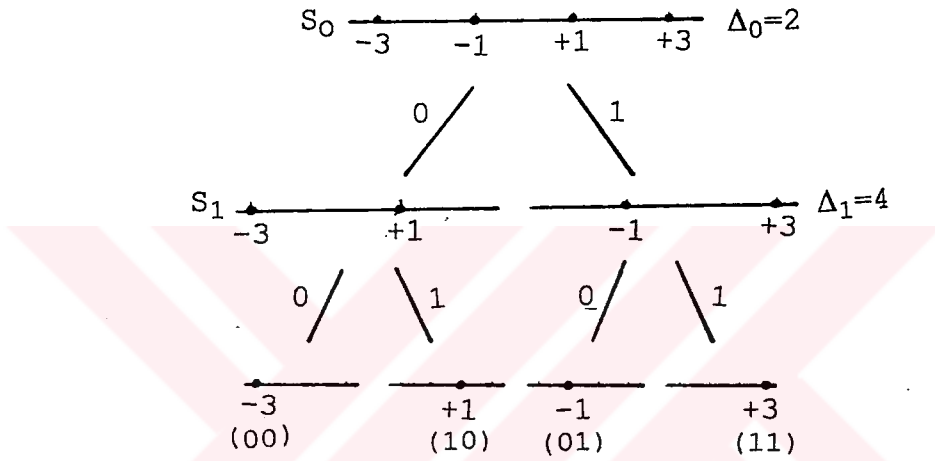
Literatürdeki [Daut et al,1982 ] katlamalı kodlayıcılar kullanılarak oluşturulan 2 ve 3 düzeyli kodlayıcılar sırasıyla (Tablo 3.2.a) ve (Tablo 3.3.a)'da gösterilmiştir. Karmaşıklık sayısının düşürülmesi amacıyla delikli katlamalı kodlayıcıların [Yasuda et al,1984] kullanıldığı 2 ve 3 düzeyli kodlayıcılar da (Tablo 3.2.b) ve (Tablo 3.3.b)'de gösterilmiştir.

Tablolardaki her bir kodlayıcı sistemine ilişkin asimptotik kodlama kazancı ifadesi, (2.19) bağıntısıyla elde edilmiştir.

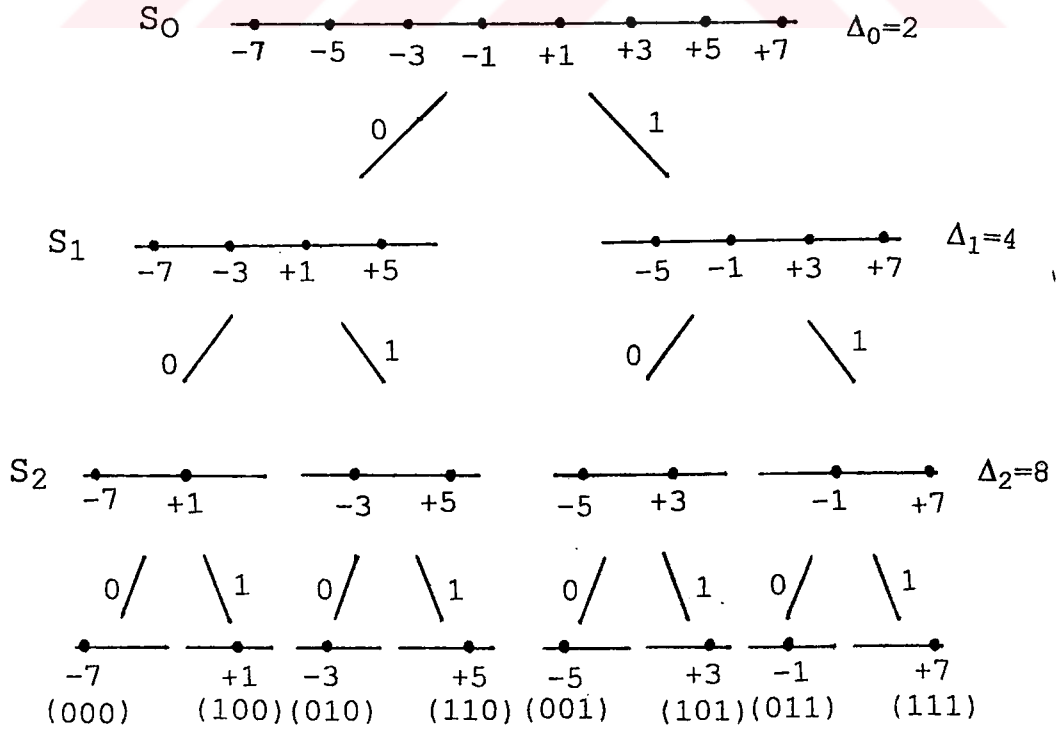
Kodlayıcı kazancı hesaplanırken, karesel Öklit uzaklığı  $d_{f,ED}^2$  değeri, bit başına enerji miktarı olan  $E_b$ 'ye göre normalize edilmiştir. 4-AM için simge enerjisi  $E_s = ((-3)^2 + (-1)^2 + (1)^2 + (3)^2)/4 = 5$ 'tir. 2 düzeyli kodlayıcı için  $E_b = E_s$

olduğundan  $E_b=5$  olarak elde edilir. 8-AM için simge enerjisi ise,  $E_s = ((-7)^2 + (-5)^2 + (-3)^2 + (-1)^2 + (1)^2 + (3)^2 + (5)^2 + (7)^2)/8=21$ 'dir. 3 düzeyli kodlayıcı için  $E_b=E_s/2$  olduğundan  $E_b=10.5$  olarak elde edilir.

4-AM ve 8-AM için küme bölmelenmesi ve alt küme uzaklıkları  $\Delta_i$ 'ler sırasıyla (Şekil 3.1) ve (Şekil 3.2)'de gösterilmiştir.

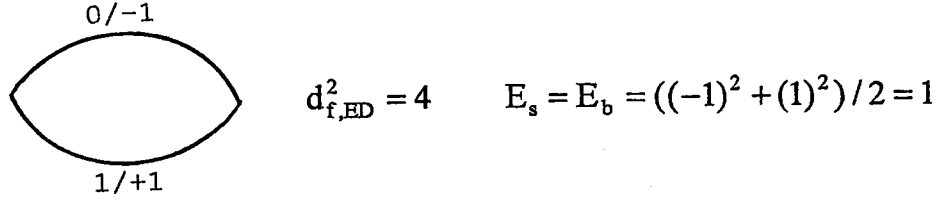


(Şekil 3.1) 4-AM işaret kümesinin bölmelenmesi

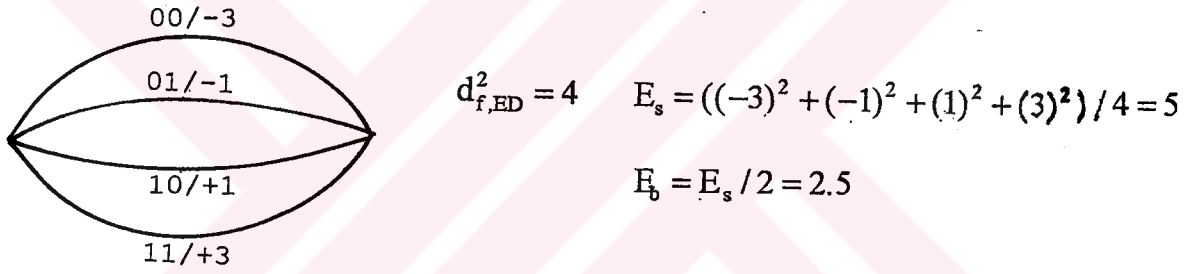


(Şekil 3.2) 8-AM işaret kümesinin bölmelenmesi

4-AM için 2 düzeyli kodlayıcının kodlama kazancı kodlamasız 2-AM'e göre ve 8-AM için 3 düzeyli kodlayıcının kodlama kazancı da kodlamasız 4-AM'e göre hesaplanmıştır. Kodlamasız 2-AM ve 4-AM için tek durumlu kafes yapıları sırasıyla (Şekil 3.3) ve (Şekil 3.4)'de gösterilmiştir.



(Şekil 3.3) Kodlamasız 2-AM modülasyon yapısı



(Şekil 3.4) Kodlamasız 4-AM modülasyon yapısı

Tablolardaki karmaşıklık sayısı (L), kullanılan kodlayıcıların türlerine göre (2.21), (2.23) ve (2.24) bağıntılarından yararlanarak elde edilmiştir.

(Tablo 3.2.a) R=1/2 oranlı, 2-düzeyle IHS tipi kodlayıcılar

No	R <sub>i</sub>	K <sub>i</sub>	d <sub>H</sub> (m)	d <sub>f.ED</sub>	AKK(dB)	L <sub>i</sub>	L
IHS-1	1/2	3	6	4.899	0.792	4	6
	1/2	2	5			2	
IHS-2	1/2	4	7	5.292	1.462	8	10
	1/2	2	5			2	
IHS-3	1/2	5	8	5.657	2.041	16	18
	1/2	2	5			2	
IHS-4	1/2	6	10	6.325	3.011	32	34
	1/2	2	5			2	
IHS-5	1/4	2	10	6.325	3.011	1	15
	3/4	3	4			14	
IHS-6	1/4	3	13	7.211	4.150	2	16
	3/4	3	4			14	
IHS-7	1/4	4	16	8	5.051	4	18
	3/4	3	4			14	
IHS-8	1/4	5	18	8.485	5.563	8	64
	3/4	5	5			56	
IHS-9	1/3	2	8	5.657	2.040	1.33	5.33
	2/3	2	3			4	
IHS-10	1/3	3	10	6.325	3.011	2.67	6.67
	2/3	2	3			4	
IHS-11	1/3	4	12	6.928	3.802	5.33	9.33
	2/3	2	3			4	
IHS-12	1/3	5	13	7.211	4.150	10.67	18.67
	2/3	3	4			8	

(Tablo3.2.b) R=1/2 oranlı, son düzeyde delikli katlamalı kod kullanılan  
2 -düzeyli IHS tipi kodlayıcılar

No	R <sub>i</sub>	K <sub>i</sub>	d <sub>H(i)</sub>	d <sub>f,ED</sub>	AKK(dB)	L <sub>i</sub>	L
IHS-13	1/2	3	6	4.899	0.792	4	6
	1/2	2	5			2	
IHS-14	1/2	4	7	5.292	1.462	8	10
	1/2	2	5			2	
IHS-15	1/2	5	8	5.657	2.041	16	18
	1/2	2	5			2	
IHS-16	1/2	6	10	6.325	3.011	32	34
	1/2	2	5			2	
IHS-17	1/5	2	13	5.657	2.041	0.8	4
	4/5(p)	2	2			3.2	
IHS-18	1/5	2	13	6.928	3.802	0.8	7.2
	4/5(p)	3	3			6.4	
IHS-19	1/5	2	13	7.211	4.150	0.8	26.4
	4/5(p)	5	4			25.6	
IHS-20	1/4	3	13	6.928	3.802	2	5
	3/4(p)	2	3			3	
IHS-21	1/4	3	13	7.211	4.150	2	8
	3/4(p)	3	4			6	
IHS-22	1/4	4	16	8	5.051	4	10
	3/4(p)	3	4			6	
IHS-23	1/4	5	18	8.485	5.563	8	56
	3/4(p)	6	5			48	
IHS-24	1/4	6	20	8.944	6.021	16	64
	3/4(p)	6	5			48	
IHS-25	1/3	2	8	5.657	2.041	1.33	4
	2/3(p)	2	3			2.67	
IHS-26	1/3	3	10	6.325	3.010	2.67	5.34
	2/3(p)	2	3			2.67	
IHS-27	1/3	4	12	6.928	3.802	5.33	8
	2/3(p)	2	3			2.67	

(Tablo 3.3.a) R=2/3 oranlı, 3-düzeyle IHS tipi kodlayıcılar

No	$R_i$	$K_i$	$d_{H(i)}$	$d_{f,ED}$	AKK(dB)	$L_i$	L
IHS-28	1/5	2	13	5.657	2.799	0.8	6.8
	4/5	1	2			6	
	1	-	1			-	
IHS-29	2/5	4	10	6.325	3.768	9.6	15.2
	3/5	2	4			5.6	
	1	-	1			-	
IHS-30	2/5	5	11	6.633	4.181	19.2	24.8
	3/5	2	4			5.6	
	1	-	1			-	
IHS-31	1/2	8	12	6.928	4.559	128	138.5
	3/4	2	3			7	
	3/4	1	2			3.5	
IHS-32	1/2	10	14	7.483	5.228	512	529.5
	3/4	3	4			14	
	3/4	1	2			3.5	
IHS-33	1/4	3	13	7.211	4.907	2	287.75
	7/8	4	4			254	
	7/8	1	2			31.75	
IHS-34	1/4	4	16	8	5.809	4	289.75
	7/8	4	4			254	
	7/8	1	2			31.75	



(Tablo 3.3.b) R=2/3 oranlı, son düzeyde delikli katlamalı kod kullanılan  
3-düzeyle IHS tipi kodlayıcılar

No	R <sub>i</sub>	K <sub>i</sub>	d <sub>H(i)</sub>	d <sub>f,ED</sub>	AKK(dB)	L <sub>i</sub>	L
IHS-35	1/5	2	13	5.657	2.799	0.8	4
	4/5(p)	2	2			3.2	
	1	-	1			-	
IHS-36	1/5	2	13	6.928	4.559	0.8	7.2
	4/5(p)	3	3			6.4	
	1	-	1			-	
IHS-37	1/5	2	13	7.211	4.907	0.8	26.4
	4/5(p)	5	4			25.6	
	1	-	1			-	
IHS-38	1/5	3	16	8	5.809	1.6	27.2
	4/5(p)	5	4			25.6	
	1	-	1			-	
IHS-39	1/2	8	12	6.928	4.559	128	134
	3/4(p)	2	3			3	
	3/4(p)	2	3			3	
IHS-40	1/2	10	14	7.483	5.229	512	521
	3/4(p)	3	4			6	
	3/4(p)	2	3			3	
IHS-41	1/4	3	13	6.928	4.559	2	19.5
	7/8(p)	4	3			14	
	7/8(p)	2	2			3.5	
IHS-42	1/4	3	13	7.211	4.907	2	117.5
	7/8(p)	7	4			112	
	7/8(p)	2	2			3.5	

(Tablo 3.4)'de, 4-AM için, (Tablo 3.2.a)'da ve (Tablo 3.2.b)'deki tasarlanan sistemlerle Ungerboeck tarafından elde edilen (Tablo 3.1.a)'daki sistemler karşılaştırılmaktadır.

(Tablo 3.4) 4-AM için, Ungerboeck tipi ve 2-düzeyle IHS tipi kodlayıcıların karşılaştırılması

Kodlayıcı	AKK(dB)	L
IHS-10	3.011	6.67
IHS-22(p)	3.011	5.34
Ung-1	3.0	8
IHS-11	3.802	9.33
IHS-18(p)	3.802	7.2
Ung-3	3.4	16
IHS-12	4.15	18.67
IHS-21(p)	4.15	8
Ung-4	4.2	32
IHS-7	5.051	18
IHS-22(p)	5.051	10
Ung-6	5.1	128

(Tablo 3.4)'den de görüldüğü gibi aynı kodlama kazançları için, IHS tipi kodlayıcıların karmaşıklığı, Ungerboeck tipi kodlayıcılara göre oldukça düşüktür. Örneğin Ung-3, 16 karmaşıklıkla 3.4 dB kazanç sağlamakta, buna karşın IHS-11, 3.802 dB'lik kazancı 9.33 karmaşıklıkla elde etmektedir. Delikli kod kullanılması durumunda IHS-18 ile 3.802 dB'lik kazanç 7.2 karmaşıklıkla elde edilmektedir. Benzer şekilde Ung-4, 32 karmaşıklıkla 4.2 dB'lik kazanç

sağlarken, IHS-12, 18.67 karmaşıklıkla 4.15 dB kazanç elde etmektedir.

(Tablo 3.5)'de ise, 8-AM için, (Tablo 3.3.a) ve (Tablo 3.3.b)'deki tasarlanan sistemlerden bazılarıyla, Ungerboeck tarafından elde edilen (Tablo 3.1.b)'deki sistemler karşılaştırılmaktadır.

(Tablo 3.5) 8-AM için, Ungerboeck tipi ve 2-düzeyle IHS-tipi kodlayıcıların karşılaştırılması

Kodlayıcı	AKK(dB)	L
IHS-29	3.768	15.2
Ung-8	3.8	24
IHS-30	4.181	24.8
IHS-36(p)	4.559	7.2
Ung-9	4.2	48
IHS-37(p)	4.907	26.4
Ung-10	5.0	96
IHS-38(p)	5.809	27.2
Ung-12	5.8	384

(Tablo 3.5)'den de görüldüğü gibi, 8-AM için, IHS tipi kodlayıcılar, aynı kodlama kazancını, Ungerboeck tipi kodlayıcılara göre daha düşük karmaşıklıkla gerçekleştirebilmektedirler. Örneğin IHS-29 15.2 karmaşıklıkla 3.768 dB'lik kazanç sağlarken Ung-8 3.8 dB'lik kazancı 24 karmaşıklıkla elde etmektedir. Benzer şekilde IHS-30 24.8 karmaşıklıkla 4.181 dB kazanç sağlarken, Ung-9 48 karmaşıklıkla 4.2 dB kazanç elde etmektedir. IHS tipi kodlayıcılar için delikli katlamalı kodlar kullanılması durumunda kodçözme karmaşıklıklarının daha da düşürüldüğü görülmektedir.

(Tablo 3.4) ve (Tablo3.5)'deki karşılaştırmalara bakarak; IHS tipi kodlama yöntemiyle, uygun kodlar seçilerek, yüksek kodlama kazancı ve düşük kodçözme karmaşıklığına sahip verimli sistemler oluşturulabileceği sonucu söylenebilir.

### 3.2. IHS Tipi Kodlayıcılar İçin Hata Üst Sınırı

Çok düzeyli katlamalı kodlara ilişkin hata üst sınırı için bir çalışma ilk kez Kofman, Zehavi ve Shamaı [Kofman et al,1990] tarafından yapılmıştır.

M düzeyli bir kodlayıcıda her düzeydeki  $C_i$  koduna ilişkin kodlama oranı  $R_i$  ve ortalama bir hata olasılığı  $P_{b(i)}$  ise, tüm kodlayıcı sistemine ilişkin hata olasılığı üst sınırı  $P_b$ :

$$P_b = \frac{\sum_{i=1}^M R_i P_{b(i)}}{\sum_{i=1}^M R_i} \quad (3.1)$$

eşitliği ile tanımlanır.

Kanaldan iletilen kodsözcüklerinin,  $\sigma^2=N_0/2$  varyanslı Gauss gürültüsü ile bozulduğu düşünülürse i.nci düzeydeki katlamalı kodlayıcıya ilişkin ortalama bit hata olasılığı üst sınırı:

$$P_{b(i)} \leq \frac{1}{k_i} Q \left( \sqrt{\frac{d_{f,ED(i)}^2 E_b}{2N_0}} \right) \exp \left( \frac{d_{f,ED(i)}^2 E_b}{4N_0} \right) \frac{\mathcal{A}_i(D, I)}{\mathcal{A}} \quad (3.2)$$

$$I = 1$$

$$D = a \cdot \exp \left( \frac{-\Delta_{i-1}^2}{4N_0} \right)$$

bağıntısı ile tanımlanır.

D ifadesindeki a çarpanı; eğer  $C_i$  koduna ilişkin işaret kümesinde herhangi bir elemana en yakın  $\Delta_{i-1}$  uzaklıklı 2 eleman varsa 2 olur. Eğer  $\Delta_{i-1}$  uzaklıklı tek eleman varsa 1 olacaktır.

$C_i$  koduna ilişkin Öklit uzaklığı  $d_{f,ED(i)}$  (2.8) bağıntısı ile elde edilir.  $T_i(D,I)$ ,  $C_i$  kodunun üretme işlevidir ve  $Q(\cdot)$  ise Gauss hata işlevini göstermektedir.

$$Q(x) = \int_x^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\gamma^2/2} d\gamma \quad (3.3)$$

Hata üst sınır ifadesi elde edilirken, her düzeydeki kodçözücünün bir sonraki kodçözücüye aktardığı yan bilgilerin (alt küme bilgileri) doğru olarak aktarıldığı varsayılmıştır.

2 düzeyli 4-AM modülasyonu kullanan IHS tipi bir kodlayıcı için yukarıda tanımlanan bağıntılar, aşağıda görüldüğü gibi elde edilecektir.

Kodlayıcı olarak, (Tablo 3.2.a)'da verilen IHS-9 kodlayıcısı ele alınırsa;  $R_1=1/3$  ve  $R_2=2/3$  için:

$$P_b = \frac{P_{b1} \cdot 1/3 + P_{b2} \cdot 2/3}{(1/3) + (2/3)} = P_{b1} \cdot 1/3 + P_{b2} \cdot 2/3 \quad (3.4)$$

olup 1. ve 2. düzeylerde bit hata olasılığı üst sınırları:

$$P_{b1} \leq Q\left(\sqrt{\frac{32 \cdot E_b}{2N_0}}\right) \exp\left(\frac{32 \cdot E_b}{4N_0}\right) \frac{\partial T_1(D,I)}{\partial I} \quad (3.5)$$

$$\left. \begin{array}{l} I=1 \\ D=2 \exp\left(\frac{-\Delta_0^2 E_b}{4N_0}\right) \end{array} \right|$$

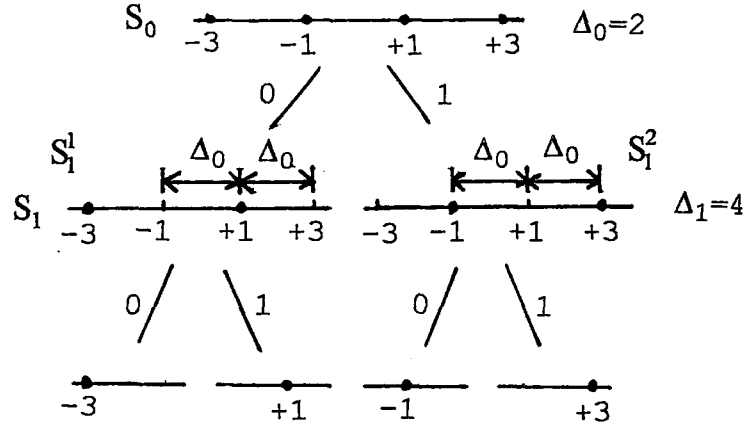
$$P_{b2} \leq Q\left(\sqrt{\frac{48 \cdot E_b}{2N_0}}\right) \exp\left(\frac{48 \cdot E_b}{4N_0}\right) \frac{\partial T_2(D,I)}{\partial I} \quad (3.6)$$

$$\left. \begin{array}{l} I=1 \\ D=\exp\left(\frac{-\Delta_1^2 E_b}{4N_0}\right) \end{array} \right|$$

biçiminde elde edilir.

(3.5) bağıntısında  $D$  değerinin 2 çarpanı almasının nedeni  $C_1$  koduna ilişkin işaret kümesinde herhangi bir

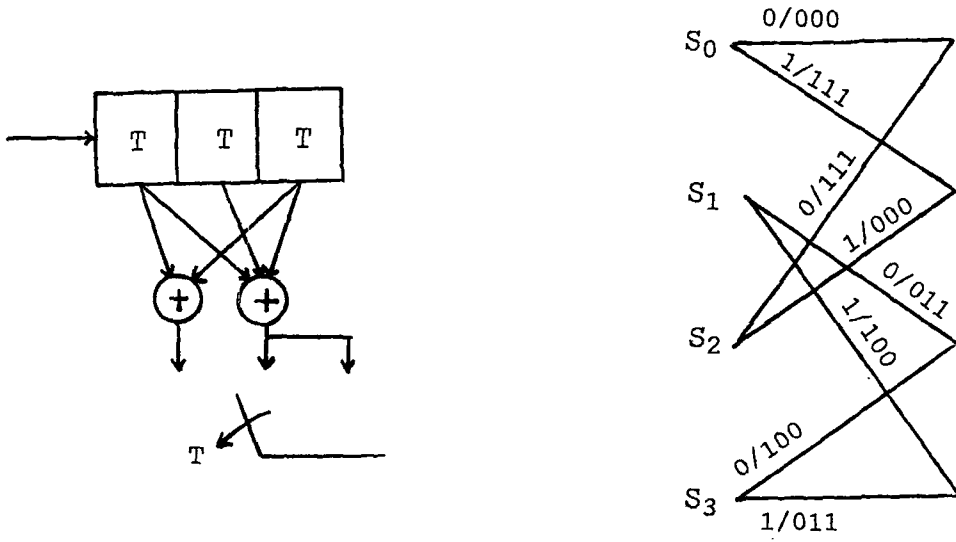
elemana (-1 veya +1)  $\Delta_0=2$  uzaklıklı 2 elemanın bulunmasıdır. (Şekil 3.5) bu durumu göstermektedir.



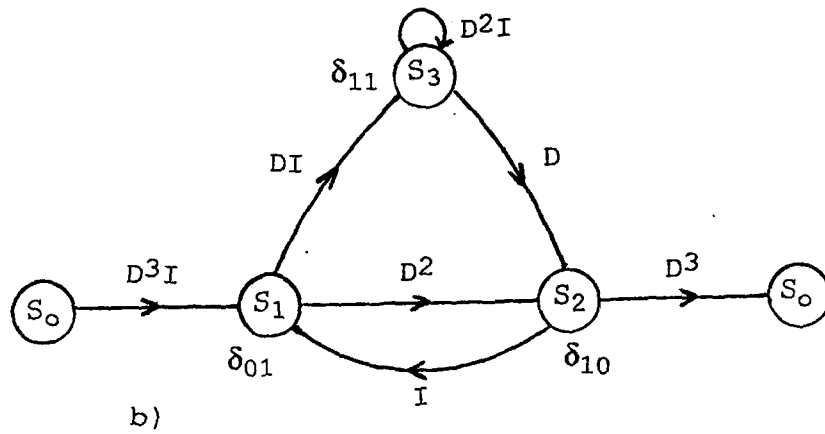
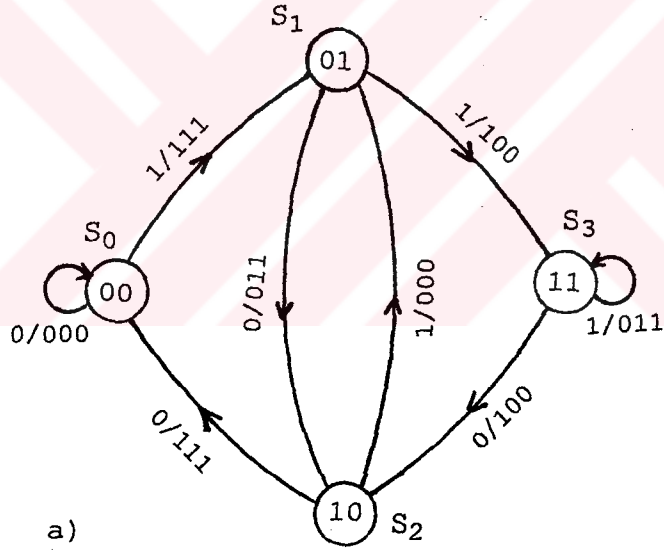
(Şekil 3.5) Kafes üzerinde  $C_1$  kodu için  $\Delta_0=2$  uzaklıklı elemanlar

(Şekil 3.5)'den de görüldüğü gibi kodlayıcı tarafından iletilen simge  $S_1^1$  kümesindeki (+1) elemanı olursa, kodçözücü bu (+1) elemanı yerine ona en yakın olan  $\Delta_0=2$  uzaklıklı  $S_1^2$  kümesindeki (-1) veya (+3) elemanlarından birine karar vererek hata yapabilir. Dolayısıyla  $S_1^1$  kümesinden bir eleman gönderildiğinde,  $S_1^2$  kümesinden bir eleman 2 farklı şekilde seçilerek hata yapılabilir. Sonuç olarak 1. düzeydeki kodlayıcı için bir bitin iletilmesinde 2 farklı hata yapılabilecektir. Son düzeyde ise herhangi bir elemana  $\Delta_1=4$  uzaklıklı yalnızca bir tane eleman vardır ve bu kodlayıcının bir bit için yapacağı hata tek biçimde ortaya çıkacaktır.

1. düzeydeki  $R_1=1/3$  oranlı katlamalı kodlayıcı ve bu kodlayıcıya ilişkin kafes yapı sırasıyla (Şekil 3.6.a) ve (Şekil 3.6.b)'de gösterilmiştir. Yine bu kodlayıcıya ilişkin durum diyagramı ve geliştirilmiş durum diyagramı (Şekil 3.7.a) ve (Şekil 3.7.b)'de görülmektedir.



(Şekil 3.6) a)  $C_1$  koduna ilişkin kodlayıcı yapısı  
b)  $C_1$  koduna ilişkin kafes yapı



(Şekil 3.7) a)  $C_1$  koduna ilişkin durum diyagramı

b)  $C_1$  koduna ilişkin geliştirilmiş durum diyagramı

$C_1$  koduna ilişkin  $T_1(D, I)$  üretme işlevi aşağıda anlatıldığı gibi elde edilmiştir.

Geliştirilmiş durum diyagramına bakarak şu eşitlikler yazılabilir:

$$\delta_{01} = D^3 I + 0 \cdot \delta_{01} + I \delta_{10} + 0 \delta_{11} \quad (3.7)$$

$$\delta_{10} = 0 + D^2 \delta_{01} + 0 \cdot \delta_{10} + D \delta_{11} \quad (3.8)$$

$$\delta_{11} = 0 + D I \delta_{01} + 0 \cdot \delta_{10} + D^2 I \delta_{11} \quad (3.9)$$

Üretme işlevi ise:

$$T_1(D, I) = D^3 \delta_{10} \quad (3.10)$$

biçiminde elde edilir.

Yukarıdaki denklem takımı matris formda yazılacak olursa şu eşitliği elde edebiliriz:

$$\begin{bmatrix} \delta_{01} \\ \delta_{10} \\ \delta_{11} \end{bmatrix} = \begin{bmatrix} 0 & I & 0 \\ D^2 & 0 & D \\ DI & 0 & D^2 I \end{bmatrix} \cdot \begin{bmatrix} \delta_{01} \\ \delta_{10} \\ \delta_{11} \end{bmatrix} + \begin{bmatrix} D^3 I \\ 0 \\ 0 \end{bmatrix} \quad (3.11)$$

A                      B                      C

(3.11) eşitliğindeki A'yı U birim matrisi ile çarpıp B'yi sol tarafa geçirelim:

$$\begin{bmatrix} 1 & -I & 0 \\ -D^2 & 1 & -D \\ -DI & 0 & 1-D^2 I \end{bmatrix} \cdot \begin{bmatrix} \delta_{01} \\ \delta_{10} \\ \delta_{11} \end{bmatrix} = \begin{bmatrix} D^3 I \\ 0 \\ 0 \end{bmatrix} \quad (3.12)$$

Sonuç olarak;



$$\begin{bmatrix} \delta_{01} \\ \delta_{10} \\ \delta_{11} \end{bmatrix} = \begin{bmatrix} 1 & -I & 0 \\ -D^2 & 0 & -D \\ -DI & 0 & 1-D^2 \end{bmatrix}^{-1} \begin{bmatrix} D^3 I \\ 0 \\ 0 \end{bmatrix} \quad (3.13)$$

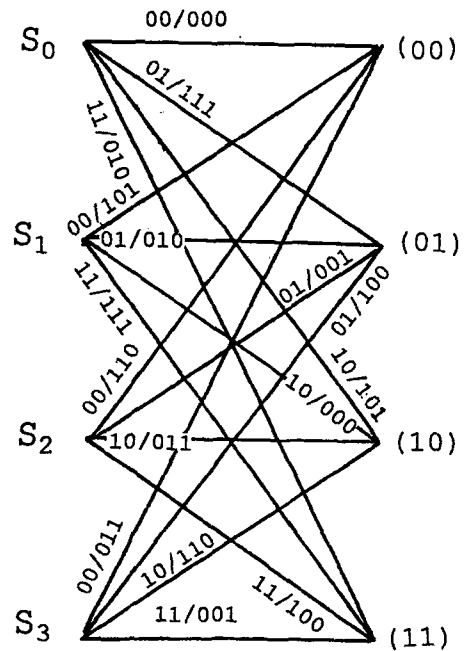
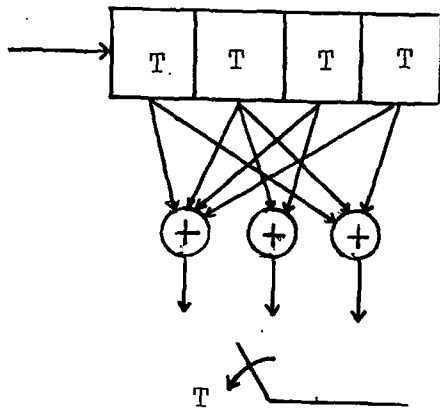
elde edilir.  $T_1(D, I)$  üretme işlevi ise:

$$T_1(D, I) = [ 0 \quad D^3 \quad 0 ] \cdot \begin{bmatrix} 1 & -I & 0 \\ -D^2 & 1 & -D \\ -DI & 0 & 1-D^2 \end{bmatrix}^{-1} \begin{bmatrix} D^3 I \\ 0 \\ 0 \end{bmatrix} \quad (3.14)$$

bağıntısı ile verilebilir.

$\frac{\partial T_1(D, I)}{\partial I}$  türevi ise , türev tanımından yararlanarak;

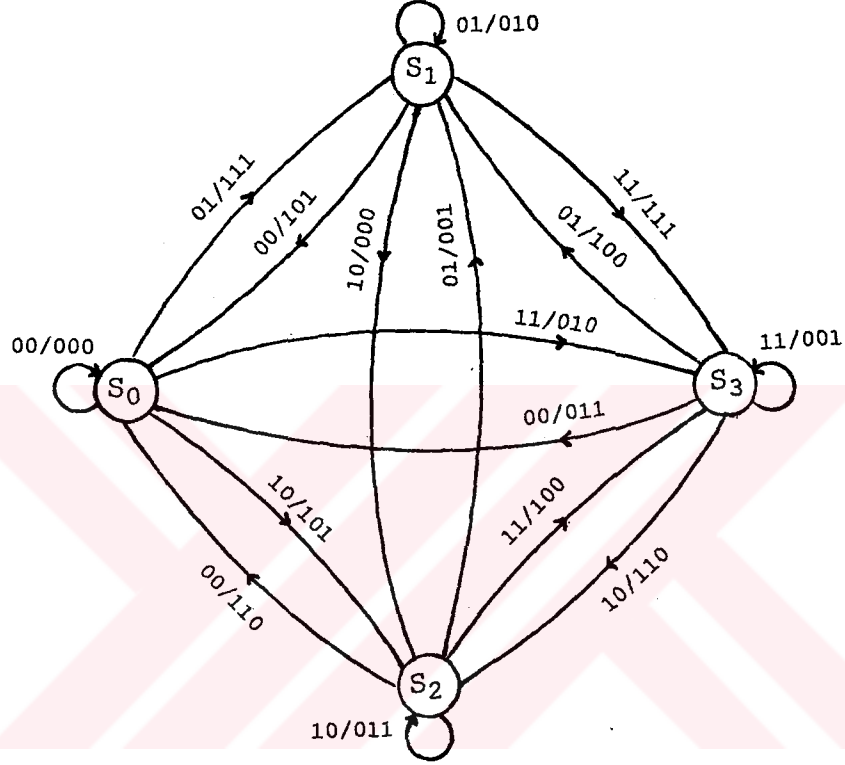
$\frac{T_1(D, 1.00001) - T_1(D, 1)}{0.00001}$  yaklaşıklıkıyla bulunabilir.



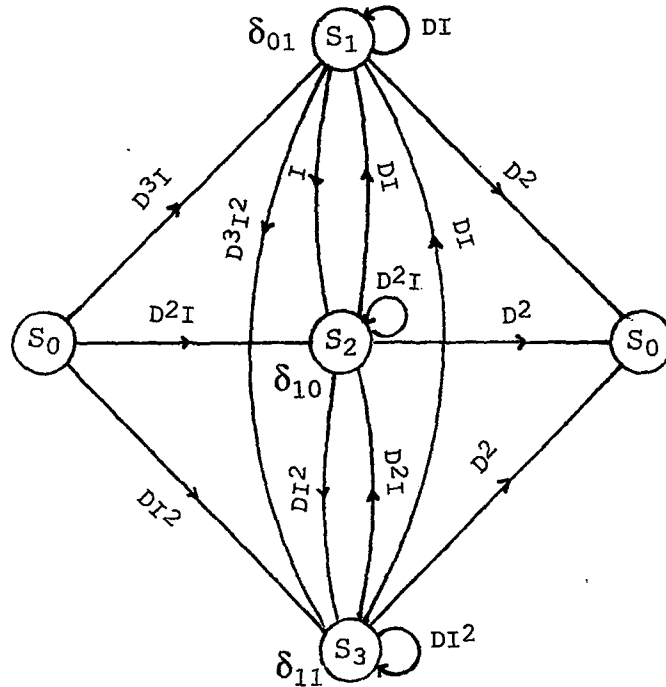
(Şekil 3.8) a)  $C_2$  koduna ilişkin kodlayıcı yapısı

b)  $C_2$  koduna ilişkin kafes yapısı

2. düzeydeki  $R_2=2/3$  oranlı katlamalı kodlayıcı ve bu kodlayıcıya ilişkin kafes yapı ise sırasıyla (Şekil 3.8.a) ve (Şekil 3.8.b)'de görülmektedir. Yine bu kodlayıcıya ilişkin durum diyagramı ile geliştirilmiş durum diyagramı (Şekil 3.9.a) ve (Şekil 3.9.b)'de gösterilmiştir.



(Şekil 3.9.a)  $C_2$  koduna ilişkin durum diyagramı



(Şekil 3.9.b)  $C_2$  koduna ilişkin geliştirilmiş durum diyagramı

$C_2$  koduna ilişkin  $T_2(D, I)$  üretme işlevi de 1. düzeydeki  $T_1(D, I)$ 'yı bulurken izlediğimiz yoldan elde edilir. 2. düzey için (Şekil 3.9.b)' ye bakarak şu eşitlikleri yazabiliriz:

$$\delta_{01} = D^3I + DI\delta_{01} + DI\delta_{10} + DI\delta_{11} \quad (3.15)$$

$$\delta_{10} = D^2I + I\delta_{01} + D^2I\delta_{10} + D^2I\delta_{11} \quad (3.16)$$

$$\delta_{11} = DI^2 + D^3I^2\delta_{01} + DI^2\delta_{10} + DI^2\delta_{11} \quad (3.17)$$

Bu denklem takımını matris formda yazalım:

$$\begin{matrix} A \\ \begin{bmatrix} \delta_{01} \\ \delta_{10} \\ \delta_{11} \end{bmatrix} \end{matrix} = \begin{matrix} B \\ \begin{bmatrix} DI & DI & DI \\ I & D^2I & D^2I \\ D^3I^2 & DI^2 & DI^2 \end{bmatrix} \end{matrix} \cdot \begin{matrix} \\ \begin{bmatrix} \delta_{01} \\ \delta_{10} \\ \delta_{11} \end{bmatrix} \end{matrix} + \begin{matrix} C \\ \begin{bmatrix} D^3I \\ D^2I \\ DI^2 \end{bmatrix} \end{matrix} \quad (3.18)$$

A'yı U birim matrisi ile çarpıp B'yi sol tarafa geçirelim:

$$\begin{bmatrix} 1-DI & -DI & -DI \\ -I & 1-D^2I & -D^2I \\ -D^3I^2 & -DI^2 & 1-DI^2 \end{bmatrix} \cdot \begin{bmatrix} \delta_{01} \\ \delta_{10} \\ \delta_{11} \end{bmatrix} = \begin{bmatrix} D^3I \\ D^2I \\ DI^2 \end{bmatrix} \quad (3.19)$$

Sonuç olarak:

$$\begin{bmatrix} \delta_{01} \\ \delta_{10} \\ \delta_{11} \end{bmatrix} = \begin{bmatrix} 1-DI & -DI & -DI \\ -I & 1-D^2I & -D^2I \\ -D^3I^2 & -DI^2 & 1-DI^2 \end{bmatrix}^{-1} \cdot \begin{bmatrix} D^3I \\ D^2I \\ DI^2 \end{bmatrix} \quad (3.20)$$

yazılır.  $T_2(D, I)$  üretme işlevi ise:

$$T_2(D, I) = D^2\delta_{01} + D^2\delta_{10} + D^2\delta_{11}$$

bağıntısıyla elde edilir. O halde;

$$T_2(D, I) = \begin{bmatrix} D^2 & D^2 & D^2 \end{bmatrix} \cdot \begin{bmatrix} 1-DI & -DI & -DI \\ -I & 1-D^2I & -D^2I \\ -D^3I^2 & -DI^2 & 1-DI^2 \end{bmatrix}^{-1} \cdot \begin{bmatrix} D^3I \\ D^2I \\ DI^2 \end{bmatrix}$$

olarak elde edilebilir.

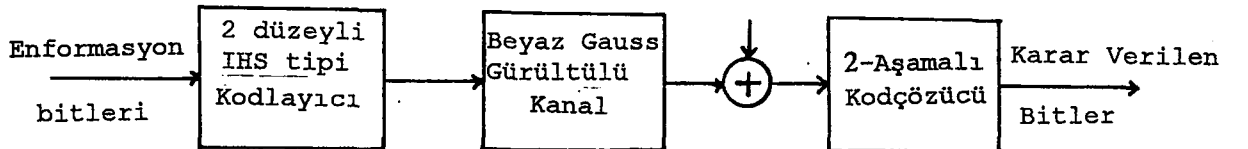
$$\frac{\partial T_2(D, I)}{\partial I} \text{ türevi, türev tanımından hareketle } \frac{T_2(D, 1.00001) - T_2(D, 1)}{0.00001}$$

yaklaşıklığından elde edilebilir.

2 düzeyli bu kodlayıcı için hata üst sınır değeri bir bilgisayar programı yardımıyla hesaplanmış olup, sonuçlar (Şekil 3.11)'de verilmiştir.

### 3.3. Bilgisayar Benzetim Programının Tanıtılması Ve Elde Edilen Sonuçlar

Bir önceki altbölümde hata üst sınırı elde edilen 2 düzeyli kodlayıcı için bit hata olasılığına ilişkin benzetim programı [Ek 1 ]'de verilmiştir. Benzetimin yapıldığı haberleşme sistemini genel olarak (Şekil 3.10)'da görüldüğü gibi düşünebiliriz.



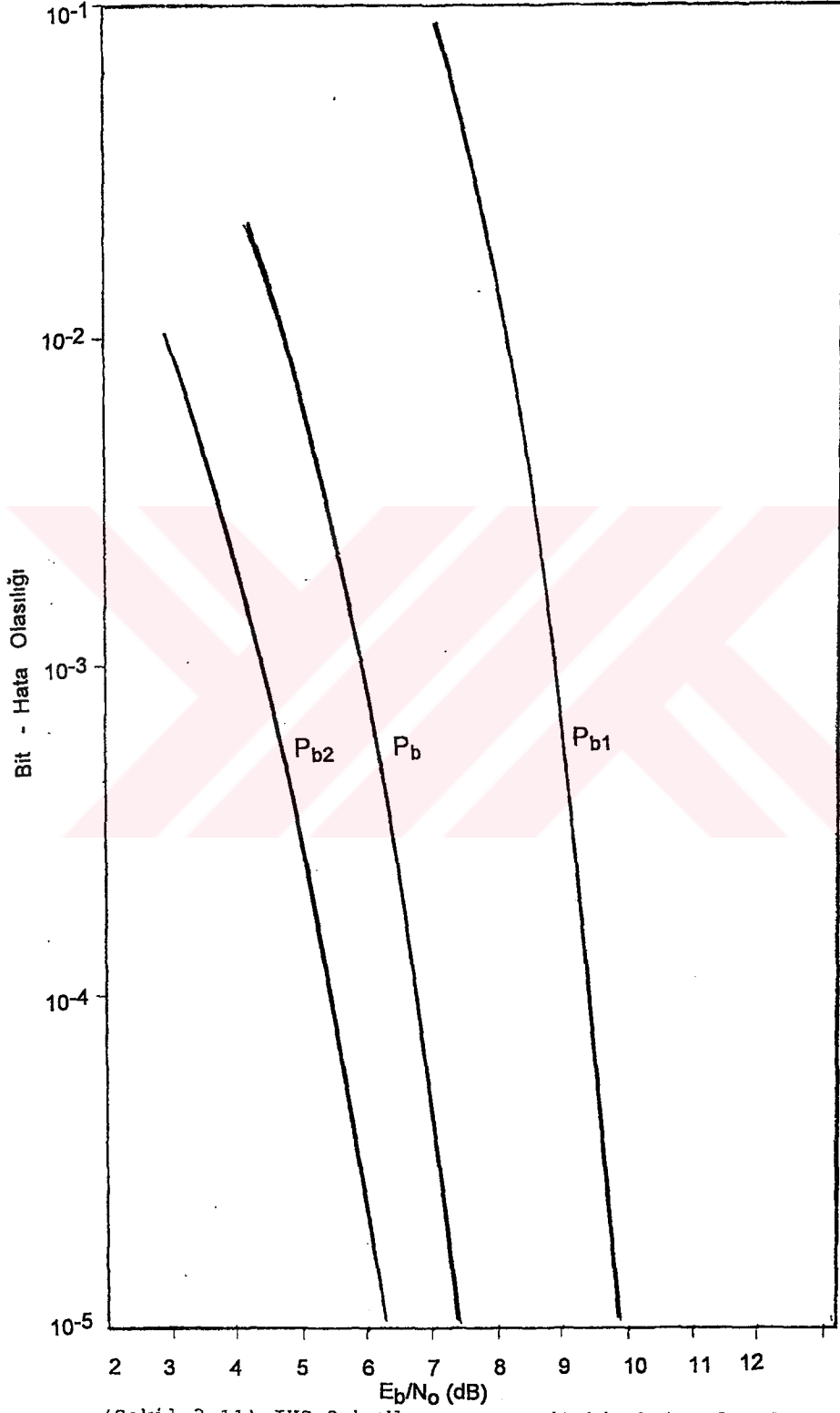
(Şekil 3.10) Bilgisayar benzetim programının uygulandığı sayısal haberleşme sistemi

Kanaldan iletilen simgelerin  $\sigma^2=N_0/2$  varyanslı Gauss gürültüsü ile bozulduğu düşünölmüştür.

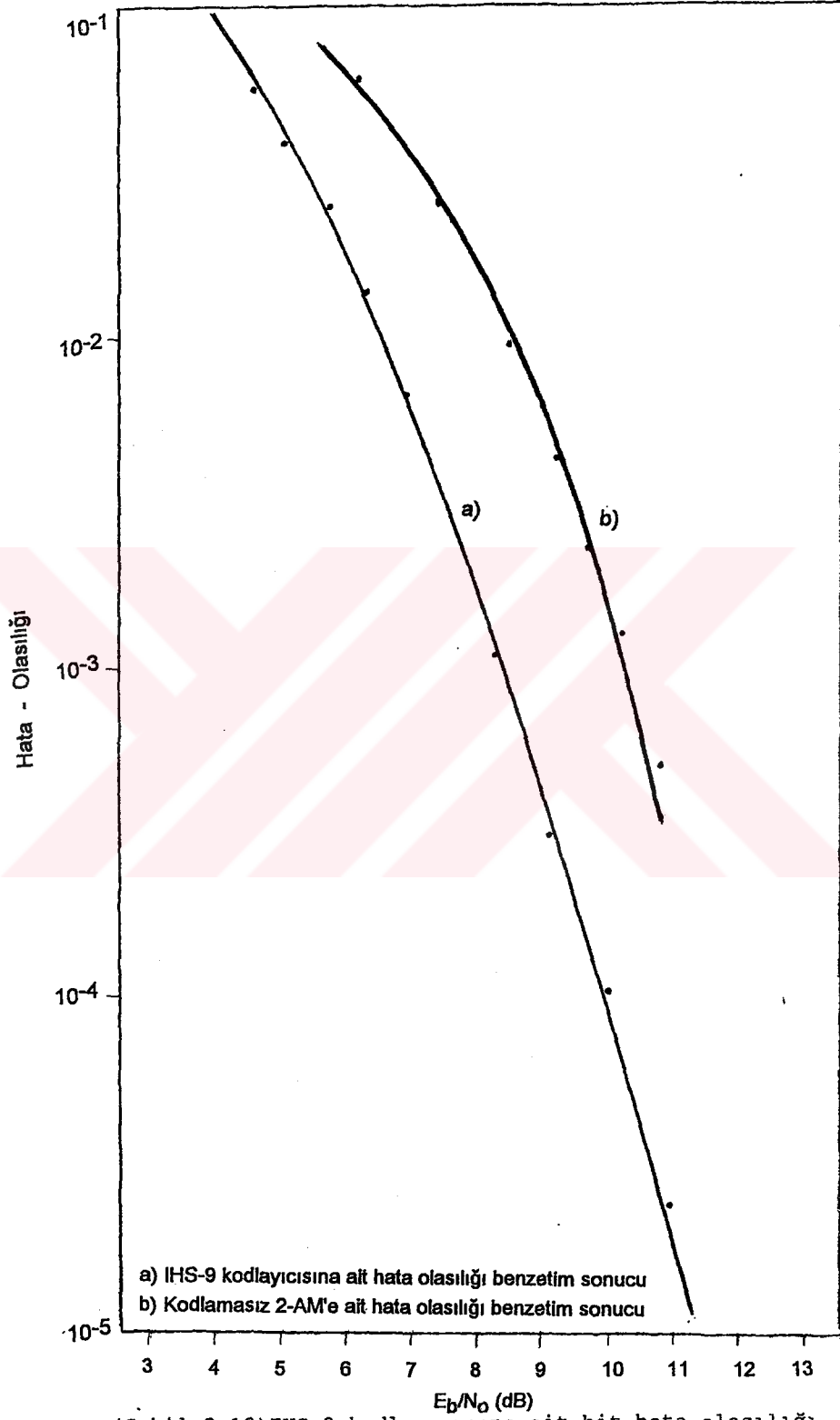
Alıcı taraftaki çok aşamalı kodçözücü, karar verme algoritması olarak Viterbi Algoritmasını kullanmaktadır ve bu algoritmaya ilişkin adım sayısı 10 olarak alınmaktadır.

Benzetim programında, her bir deneme için 10 adımda iletilen bitlerin kaç tanesinin doğru olarak alındığı bulunmuştur. Bu işlem 500 deneme için tekrarlanarak, doğru alınan toplam bit sayısı, gönderilen toplam bit sayısına bölünmüş ve bit hata olasılığı elde edilmiştir. Bu işlem çeşitli gürültü varyans değerleri için tekrarlanarak (Şekil 3.12)'de görölen benzetim sonucu elde edilmiştir.

2 düzeyli kodlayıcıya ait, benzetim sonucu ile elde edilen bit hata olasılığı, kodlamasız 2-AM'e ait hata olasılığı ile karşılaştırılmıştır. (Şekil 3.12)



(Şekil 3.11) IHS-9 kodlayıcısına ait bit hata olasılığı üst sınırı



(Şekil 3.12) IHS-9 kodlayıcısına ait bit-hata olasılığı benzetim sonucu ile kodlamasız 2-AM bit hata olasılığı benzetim sonucunun karşılaştırılması

#### 4. 4-AM YARDIMIYLA ÇOK DÜZEYLİ HAT KODU TASARIMI

##### 4.1. Hat Kodlamasının Kullanılma Nedeni Ve Özellikleri

Daha önceden yapılmış olan çalışmalar, yüksek iletim hızı için iletim kanalı olarak metalik kablolardan yararlanan Tümüleşik Hizmetler Sayısal Şebekelerinde (Integrated Services Digital Network = ISDN), ya da bazı optik Yerel Alan Şebekelerinde (Local Area Network = LAN) ikili kod olarak çoklu hat kodlarının (line code) kullanılması gerektiğini göstermiştir. Bu yöntem, alçak geçiren frekans karakteristiğine sahip iletim kanalları üzerinde oluşan simgelerarası girişimi ve işaret zayıflamasını azaltmak için kullanılabilir.

Değişik iletim kanalları için kafes yapıda hat kodlarının kullanıldığı çeşitli yapılar özellikle son yıllarda ilgi odağı haline gelmektedir [Calderbank,1989], [Aygölü,1994].

Bir çift tel üzerinden iletilecek olan ikili sözcüklerin, doğru akım bileşenlerinin olması, iletim için elverişli olmayan bir durumdur. Ayrıca bu işaretlerin bant genişliklerinin de küçük olması istenir. Bu iki durum, hat kodlaması yardımıyla gerçekleştirilebilir.

Hat kodlaması, kaynak tarafından üretilen sayısal bilginin kanaldan en iyi biçimde iletilmesi için gerekli işlemleri gerçekleştirir.

Sayısal temelbant iletişim kanalı için kullanılan bir hat kodunun göstermesi gereken en önemli özellik şu şekilde açıklanabilir: Metalik ya da optik kablolar üzerinden güç beslemesini mümkün kılmak ve dc bağlaşmanın olmaması nedeniyle oluşan bozulmayı sınırlandırmak için, kodlanmış işaret dizilerine ait güç spektral yoğunluğunun sıfır frekans bileşeninin olmaması ve alçak frekans bileşenlerinin düşük olması istenir. Sıfır frekans (dc) bileşeninin



olmaması için, iletilen kodsimgelerinin akan sayısal toplamının (running digital sum = RDS) belirli simetrik  $(-a, a)$  değerleri arasında kalması gerek ve yeter koşuldur.

Kod simgelerine ilişkin akan sayısal toplam (AST):

$$AST(k) = AST(0) + \sum_{n=1}^k c_n \quad c_n: \text{ kod simgesi} \quad (4.1)$$

bağıntısıyla tanımlanır.

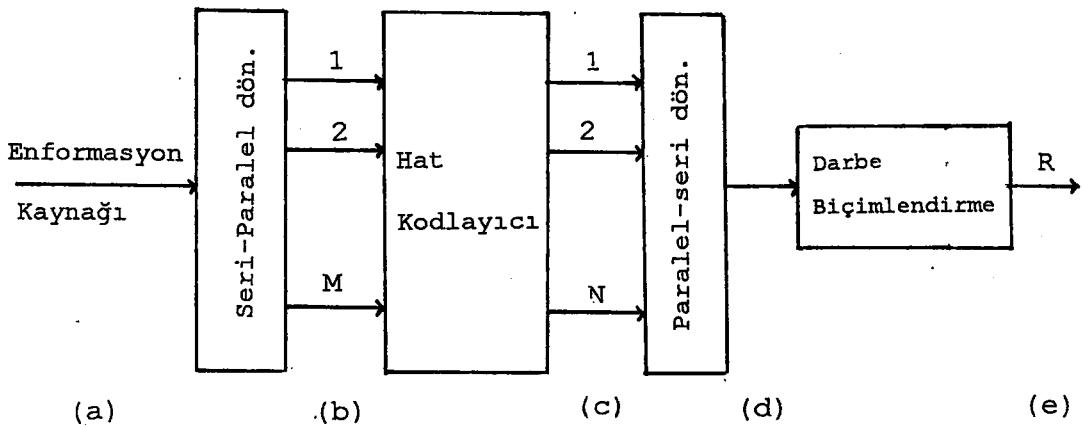
$$Z_{k,l} = \sum_k^l c_n \quad (4.2)$$

olmak üzere sayısal toplam değişimi (digital sum variation = DSV):

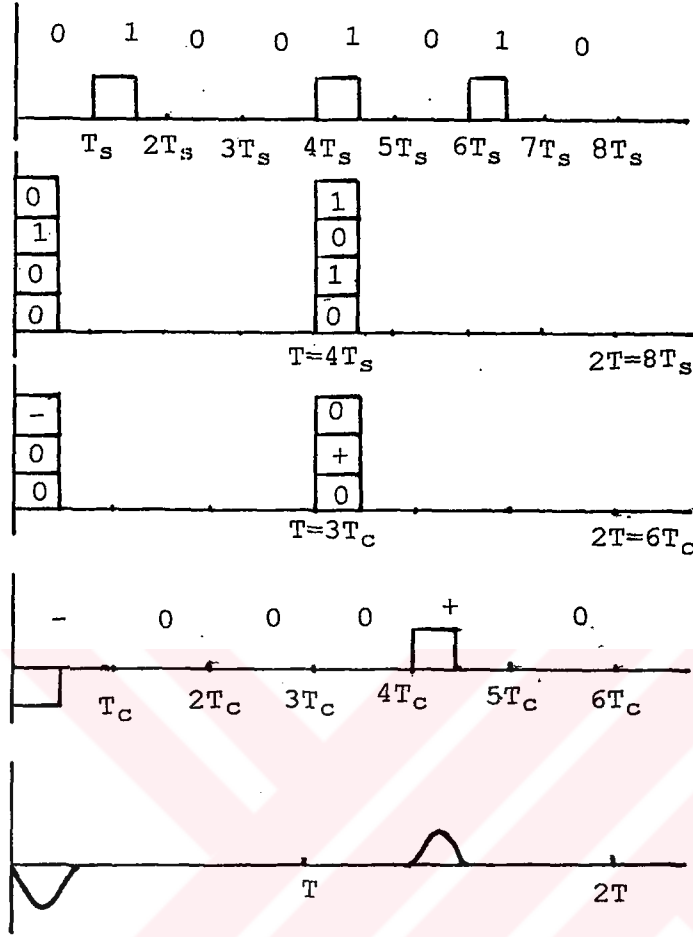
$$DSV = \max_{k,l} Z_{k,l} \quad (4.3)$$

olarak tanımlanır. Hat kodundaki bozulmanın küçük olması için, DSV değerinin olabildiğince küçük olması gerekmektedir.

(Şekil 4.1.a) genel bir hat kodu yapısını göstermektedir [Cariolaro et al, 1983].



(Şekil 4.1.a) Hat kodlayıcı yapısının genel şekli



(Şekil 4.1.b) Bir hat kodlayıcısına ilişkin işaretlerin gösterilimi

a) Enformasyon kaynağı tarafından üretilen ikili bitler kaynak simge dizisi olarak adlandırılır. Her bir simge  $T_s$  süresi boyunca iletilir.

b) Kaynak sözcüğü dizisi:  $M$  uzunluklu bloklara ayrılmış data bitleri, kaynak sözcüğü dizisini oluşturur.  $T$  saniyelik bir işaretleşme aralığı için  $T=MT_s$  eşitliği geçerli olacaktır.

c) Kod sözcüğü dizisi:  $M$  uzunluklu kaynak sözcüğü dizisine karşılık, hat kodlayıcısı tarafından elde edilen  $N$  uzunluklu dizi, kod sözcüğü dizisini oluşturur. Her bir kod sözcüğü  $NT_c$  süresince iletiliyorsa  $T=MT_s=NT_c$  eşitliği düşünülebilir.

d) Kod simge dizisi: Kod sözcüğü dizisindeki her bir bileşen kod simgesini oluşturur.  $T_C$ , her bir simgenin iletilmesi için gerekli süreyi göstermektedir

e) Çıkış işareti: Kod simgeleri, bir darbe şekliyle biçimlendirilerek çıkış işareti elde edilir.

Yukarıda anlatılar (Şekil 4.1.b)'de gösterilmiştir.

#### 4.2. Çoklu 4-AM Hat Kodları

Çoklu 4-AM hat kodları için,  $(-3, -1, +1, +3)$  elemanlarının kullanıldığı 2 uzunluklu kod sözcüklerinden oluşan 8'li bir kod sözcüğü kümesinin ( $S_0$ ) oluşturulması ve kod sözcükleri n-uzunluklu ise bu kümede  $2^{n+1}$  tane elemanın bulunması gerekmektedir.

Tasarlanan kodlayıcılarda, alfabedeki kod sözcüklerinin dördünün akan sayısal toplamı değeri (AST) 0'a eşit, ikisinin AST değeri +2 (ya da +4)'ye eşit, diğer ikisinin de -2 (ya da -4)'ye eşittir.

İkili kodlayıcı olarak M-düzeyle bir kodlayıcı kullanılıyorsa, kod sözcüğü alfabesi  $S_0$ , M düzeyli altkümelere bölünür. Bu bölünme sırasında alt küme uzaklıklarının  $\Delta_0 < \Delta_1 < \Delta_2 < \dots < \Delta_{M-1}$  özelliğini sağlaması da gerekmektedir. Her düzeydeki bölünme ile ikili bir kod (burada bir katlamalı kod olacaktır) arasında ilişki kurulabilir. Alt kümeler için karesel Öklit uzaklığı, yani i. düzeydeki kodlayıcıya ait karesel serbest Öklit uzaklığı:

$$d_{f,ED}^2(i) = \Delta_{i-1}^2 d_{Hi} \quad (i=1, 2, \dots, M-1) \quad (4.4)$$

şeklinde tanımlanır. Bağlantıda  $d_{Hi}$ , i. düzeydeki ikili kodlayıcının serbest Hamming uzaklığını belirtmektedir.

Son düzey için gerekli Öklit uzaklığı, bu düzeydeki  $R=1/2$  oranlı kodlayıcının serbest Hamming uzaklığına bağlı değildir. Bu kodlayıcı için serbest Öklit uzaklığı  $d_{f,ED}(M)$

şeklinde tanımlanırsa, M düzeyli kodlayıcının karesel Öklit uzaklığı:

$$d_{f,ED}^2 = \min_{1 \leq i \leq M-1} (d_{f,ED(i)}^2, d_{f,ED(M)}^2) \quad (4.5)$$

bağıntısı yardımıyla bulunur.

Son düzeyde kullanılacak  $R=1/2$  oranlı katlamalı kodlayıcı AST değeri  $(+2, -2)$  arasında değişen hat kod sözcüklerini sırayla seçmektedir. İletilecek her kod sözcüğü, son düzeydeki ikili kodlayıcının 2 giriş biti ile belirlenir. Daha önceden de değinildiği gibi bu kodlama düzeyinde, Hamming uzaklığı ile Öklit uzaklığı arasında bir ilişki yoktur.

$R=1/2$  oranlı kodlayıcının kullanılması, sistemin toplam kodlama oranında bir azalma meydana getirmektedir. Bu da tüm sisteme ait asimptotik kodlama kazancında (AKK) bir kayıp oluşmasına neden olur. Asimptotik kodlama kazancı:

$$AKK = 10 \log \frac{(d_{f,ED}^2 / E_b)_{kodlamalı}}{(d_{f,ED}^2 / E_b)_{kodlamasız}} - 10 \log \frac{R_{uc}}{R_c} \quad (4.6)$$

bağıntısı ile tanımlanır.

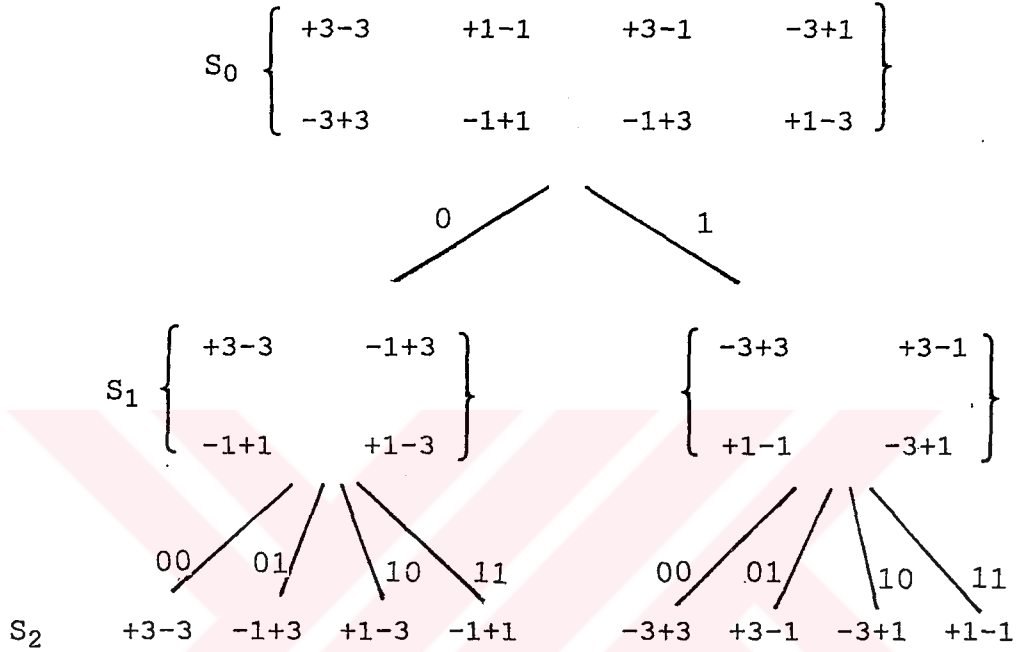
Burada kodlamalı ve kodlamasız durumdaki karesel serbest Öklit uzaklıkları  $(d_{f,ED}^2)$ , ortalama bit enerjisi  $E_b$  ile normalize edilmiştir.  $R_c$  ve  $R_{uc}$  ise sırasıyla kodlamalı ve kodlamasız durumdaki bir kodsözcüğünün iletilmesi için gerekli enformasyon bit sayısını belirtmektedir ve birimi [bit/kodsözcüğü]'dür.

#### 4.3. Çoklu 4-AM Kümeleri Ve Bu Kümelerin Bölmelenmesi

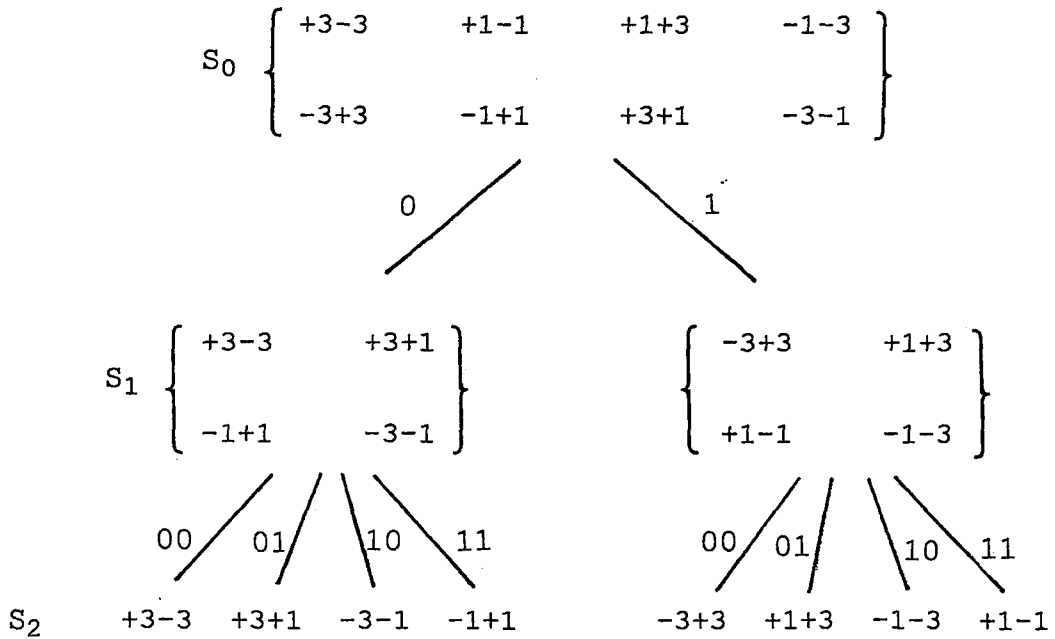
Akan sayısal toplam değerleri (AST)  $(+2, -2)$  ve  $(+4, -4)$  arasında sınırlı kalan, farklı iki 4-AM küme bölmelenmesi sırasıyla (Şekil 4.2) ve (Şekil 4.3)'de verilmiştir. 2

uzunluklu bu kod sözcükleri en iyi yapıyı sağlayacak şekilde seçilmişlerdir.

Bu küme bölmelemelerine uygun iki düzeyli hat kodlayıcı yapıları bir sonraki alt bölümde incelenmiştir.



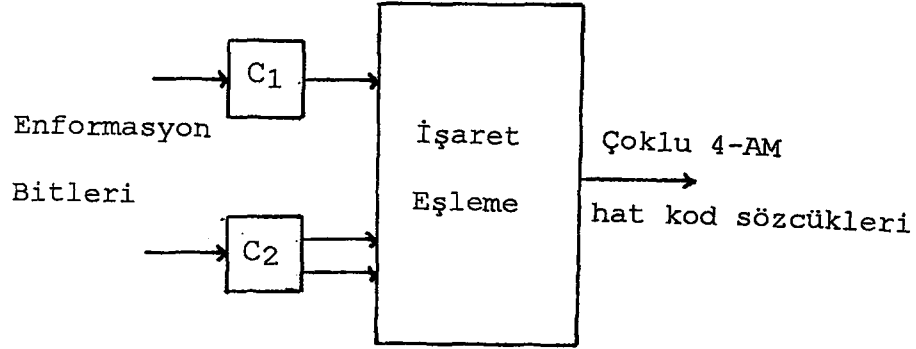
(Şekil 4.2) AST değeri (+2,-2) arasında değişen çoklu 4-AM kümesi ve bölmelenmesi



(Şekil 4.3) AST değeri (+4,-4) arasında değişen çoklu 4-AM kümesi ve bölmelenmesi

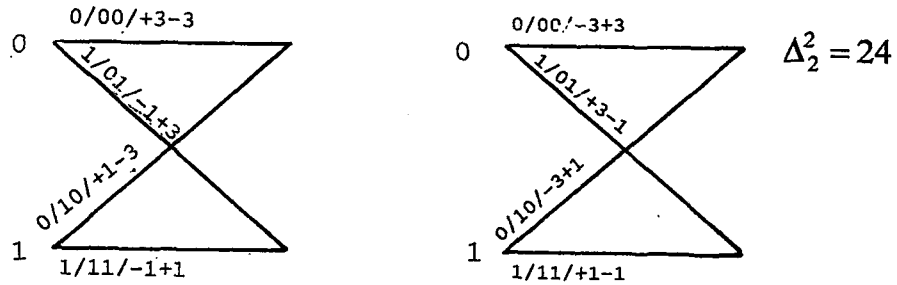
#### 4.4. 2 Düzeyli 4-AM Hat Kodları

(Şekil 4.5)'de 2 düzeyli hat kodlayıcısının genel yapısı görülmektedir.

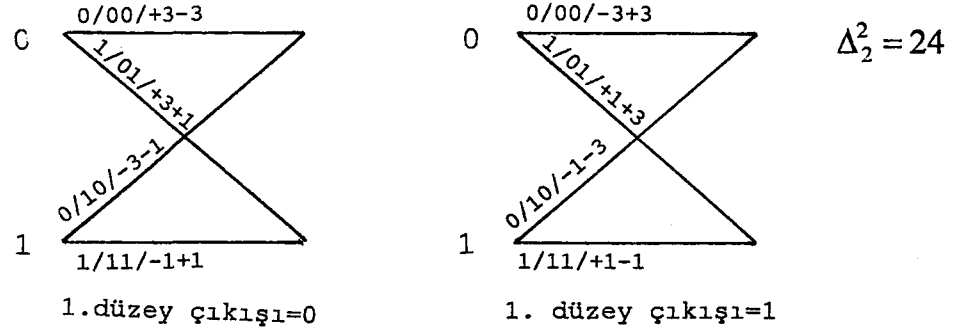


(Şekil 4.5) 2 düzeyli hat kodlayıcı yapısı

2. düzeyde kullanılan  $R=1/2$  oranlı  $C_2$  kodlayıcısına ilişkin iki durumlu kafes yapısı, AST değeri  $(+2, -2)$  arasında değişen sistem için (Şekil 4.6)'da,  $(+4, -4)$  arasında değişen sistem için (Şekil 4.7)'de gösterilmiştir. Bu düzeyde kullanılan kodlayıcı Pottie - Taylor tipinde olup, çıkışındaki iki bit ile kanaldan iletilecek kod sözcüğü belirlenir.

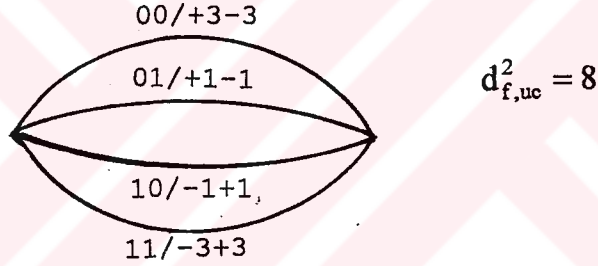


(Şekil 4.6) AST  $(+2, -2)$  için 2.düzyen kodlayıcı kafes yapısı



(Şekil 4.7) AST (+4, -4) için 2.düzyer kodlayıcı kafes yapısı

Tasarlanan kodlar için kodlama kazançları, kodlamasız duruma göre hesaplanmıştır. (Şekil 4.8). referans olarak alınan kodlamasız durumdaki tek durumlu kafes yapısını göstermektedir.



(Şekil 4.8) Kodlamasız durumdaki modülasyon yapısı

Kodlamasız durumda, iki bitlik girişe bir tane kodsözcüğü karşılık düştüğünden, iletim oranı  $R_{uc}=2(\text{bit/sözcük})$ 'tür. Aynı zamanda, bit başına enerji  $E_b=E_s/2$ 'dir.

$$E_s = \frac{((+3)^2 + (-3)^2 + (+1)^2 + (-1)^2) \cdot 2}{4} = 10 \text{ ise } E_b = 5 \text{ tir.}$$

1. düzeydeki çeşitli kodlama oranı ( $R_1$ ) değerlerine sahip katlamalı kodlayıcılar için oluşturulan ve AST değerleri (+2, -2) ile (+4, -4) arasında sınırlı kalan 2-düzyerli kodlayıcılar sırasıyla (Tablo 4.1.a) ve (Tablo 4.2.a)'de verilmiştir. Aynı düzeydeki kodlayıcılar için

delikli katlamalı kodlar [Yasuda et al,1984] kullanılırsa (Tablo 4.1.b) ve (Tablo 4.2.b)'de verilen yapılar oluşturulabilir.

(Tablo 4.1.a) AST değeri (+2,-2) arasında sınırlı kalan 2-düzeyle, 4-AM hat kodlayıcılar

No	$R_i$	Rkodlama (bit/kodkelimesi)	$\Delta_{i-1}^2$	$d_H$	$d_{f,ED}^2$	$K_i$	AKK(dB)	$L_i$	L
1	2/3 1/2	5/3	4 24	3 -	12	2 1	0.177	4 2	6
2	2/3 1/2	5/3	4 24	4 -	16	3 1	1.427	8 2	10
3	2/3 1/2	5/3	4 24	5 -	20	4 1	2.396	16 2	18
4	2/3 1/2	5/3	4 24	6 -	24	5 1	3.188	32 2	34
5	3/4 1/2	7/4	4 24	3 -	12	2 1	0.601	7 2	9
6	3/4 1/2	7/4	4 24	4 -	16	3 1	1.850	14 2	16
7	3/4 1/2	7/4	4 24	5 -	20	5 1	2.820	56 2	58
8	5/6 1/2	11/6	4 24	3 -	12	3 1	1.005	41.3 2	43.3



(Tablo 4.1.b) AST değeri (+2,-2) arasında sınırlı kalan,1.düzeyde delikli katlamalı kod kullanılan,2-düzeyle, 4-AM hat kodlayıcılar

No	$R_i$	Rkodlama (bit/kodkelimesi)	$\Delta_{i-1}^2$	dH	$d_{f,ED}^2$	$K_i$	AKK(dB)	$L_i$	L
1	$\frac{2}{3}(p)$ $\frac{1}{2}$	5/3	4 24	3 -	12	2 1	0.177	2.67 2	4.67
2	$\frac{2}{3}(p)$ $\frac{1}{2}$	5/3	4 24	4 -	16	3 1	1.427	5.33 2	7.33
3	$\frac{2}{3}(p)$ $\frac{1}{2}$	5/3	4 24	6 -	24	5 1	3.188	21.3 2	23.3
4	$\frac{3}{4}(p)$ $\frac{1}{2}$	7/4	4 24	3 -	12	2 1	0.601	3 2	5
5	$\frac{3}{4}(p)$ $\frac{1}{2}$	7/4	4 24	4 -	16	3 1	1.850	6 2	8
6	$\frac{3}{4}(p)$ $\frac{1}{2}$	7/4	4 24	5 -	20	6 1	2.820	24 2	26
7	$\frac{3}{4}(p)$ $\frac{1}{2}$	7/4	4 24	6 -	24	7 1	3.611	96 2	98
8	$\frac{5}{6}(p)$ $\frac{1}{2}$	11/6	4 24	3 -	12	3 1	1.005	6.67 2	8.67

(Tablo 4.2.a) AST değeri (+4,-4) arasında sınırlı kalan, 2-düzeyle, 4-AM hat kodlayıcılar

No	$R_i$	Rkodlama (bit/kodkelimesi)	$\Delta_{i-1}^2$	$d_H$	$d_{f,ED}^2$	$K_i$	AKK(dB)	$L_i$	L
1	2/3	5/3	8	2	16	1	1.427	2	4
	1/2		24					-	
2	2/3	5/3	8	3	24	2	3.188	4	6
	1/2		24					-	
3	3/4	7/4	8	3	24	2	3.611	7	9
	1/2		24					-	
4	5/6	11/6	8	3	24	3	4.015	41.3	43.3
	1/2		24					-	
5	7/8	15/8	8	3	24	3	4.211	127	129
	1/2		24					-	

(Tablo 4.2.b) AST değeri (+4,-4) arasında sınırlı kalan, 1.düzeyle delikli katlamalı kod kullanılan,2-düzeyle, 4-AM hat kodlayıcılar

No	$R_i$	Rkodlama (bit/kodkelimesi)	$\Delta_{i-1}^2$	$d_H$	$d_{f,ED}^2$	$K_i$	AKK(dB)	$L_i$	L
1	2/3(p)	5/3	8	3	24	2	3.188	2.67	4.67
	1/2		24					-	
2	3/4(p)	7/4	8	3	24	2	3.611	3	5
	1/2		24					-	
3	5/6(p)	11/6	8	3	24	3	4.015	6.7	8.7
	1/2		24					-	
4	7/8(p)	15/8	8	3	24	4	4.211	14	16
	1/2		24					-	

(Tablo 4.1.a) ve (Tablo 4.2.a)'daki kodlayıcılara ilişkin karmaşıklık:

$$L_i = 2^{K_i} (2^{k_i} - 1) r_i / n_i \quad (4.7)$$

bağıntısı yardımıyla bulunur. Burada  $r_i$ , bir kodsözcüğünün belirlenmesi için gerekli bit sayısını belirtmektedir.

(Tablo 4.1.b) ve (Tablo 4.2.b)'deki kodlayıcılarda ise 1. düzeydeki kodlar delikli katlamalı kodlar seçildiğinden, bu düzey için karmaşıklık:

$$L_1 = 2^{K_i} (k_i / n_i) \quad (4.8)$$

bağıntısı yardımıyla bulunur.

#### 4.5. Spektral Çözümleme

Bir hattın iletilecek olan işaretlerin sıfır frekans bileşeni içermemesi ve alçak frekans bölgesinde de güç yoğunluğunun az olması amacıyla hat kodlayıcılarının kullanılması gerektiğine değinilmişti. Bu bölümde, kodlanmış işaret dizisinin güç spektral yoğunluğunun nasıl bulunduğu incelenecek ve tasarlanan kodlar için spektrumlar elde edilecektir.

Bir hat kodlayıcısı tarafından üretilen kod sözcükleri, genel olarak  $T$  zaman aralıklarında, değişik standart yapılarıdaki  $g(t)$  darbe şekilleriyle iletirler. İletilen bu işaretler:

$$x(t) = \sum_{n=-\infty}^{\infty} u_n g(t - nT) \quad (4.9)$$

biçiminde gösterilebilirler.

(4.9)'da belirtilen  $x(t)$ 'ye ait güç spektrumunun hesaplanması için Cariolaro [Cariolaro et al, 1983]

tarafından geliştirilen yöntemden yararlanacağız. Bu amaçla önce önemli bazı tanımları vermek uygun olacaktır.

Durum geçiş matrisi  $[E_u]$ : Kodlayıcının girişindeki  $M$  bitlik giriş dizisi  $K=2^M$  tane farklı giriş dizisi tanımlar.  $u=1,2,\dots,K$  olmak üzere  $u$ . giriş için kodlayıcı durum diyagramında  $S_i$  durumundan  $S_j$  durumuna geçiş varsa matrisin  $[E_u]_{i,j}$  elemanı 1'e, eğer geçiş yoksa 0'a eşit olur.  $N$  durumlu bir kodlayıcı için  $E_u$  matrisinin boyutu  $N \times N$  olur.

Kodlayıcı çıkış matrisi  $[\Gamma_u]$ : Kodlayıcı çıkış matrisinin  $i$ . satırı ( $i=1,2,\dots,N$ ),  $u$ . giriş için  $i$ . durumun çıkışındaki kod sözcüklerinden oluşur.

$M$  uzunluklu herhangi bir  $u$ . giriş dizisinin üretilme olasılığı  $q_u$  ile gösterilir. Bundan sonra, kodlayıcı girişindeki 0 ve 1'lerin istatistiksel bağımsız olarak üretildiği durumu düşüneceğiz. Yani 0 ve 1 üretme olasılıkları 0.5'e eşit olacaktır. Dolayısıyla  $M$  uzunluklu bir giriş dizisinin üretilme olasılığı  $q_u=(0.5)^M$  olacaktır.

Durum Geçiş Matrisi  $[P]$ : Durum geçiş matrisinin  $[P]_{i,j}$  elemanı  $S_i$  durumundan  $S_j$  durumuna geçiş olasılığı değerine eşittir ve boyutu  $N \times N$ 'dir.

Durum olasılık vektörü ( $P$ ):  $N$  boyutlu bir satır vektör olan  $p$ 'nin  $i$ . elemanı, herhangi bir anda  $S_i$  durumunda olma olasılığı değerine eşittir.

$$p = pP \quad (4.10)$$

eşitliğinin çözümü bize  $p$  durum olasılık vektörünü verir. Unutulmaması gereken bir nokta da  $\sum_i p_i = 1$  olduğudur.

Geçiş olasılık matrisi, limitte  $P^\infty$  ile gösterilmek üzere :

$$P^\infty = \lim_{n \rightarrow \infty} P^n = 1.p \quad (4.11)$$

bağıntısı ile tanımlanır. Burada  $1$ , tüm elemanları 1 olan  $N$  boyutlu bir sütun vektördür. Dolayısıyla  $P$  matrisi,  $N \times N$

boyutlu ve tüm satırları  $p$  satır vektöründen oluşan bir matristir.

$(P - P^\infty)^k$  matris işlevi mutlak toplanabilir olup;

$$B(z) = [zI - (P - P^\infty)]^{-1} \quad \{z = \exp(j2\pi ft)\} \quad (4.12)$$

bağıntısı tanımlanır. Burada  $I$ ,  $N \times N$  boyutlu birim matrisi göstermektedir.

Şimdi de güç spektral yoğunluğunun hesabında önemli rol oynayan dört tane büyüklüğü tanımlayalım:

Birinci büyüklük, kodsözcüklerinin karesel ortalama değer büyüklüğüdür:

$$C_0 = \sum_{u=1}^K q_u \Gamma_u' D \Gamma_u \quad (4.13)$$

Burada  $D$ ,  $N \times N$  boyutlu bir köşegen matris olup  $D \triangleq (P_1, P_2, \dots, P_N)$  şeklinde gösterilebilir.

İkinci büyüklük ise

$$C_1 = \sum_{u=1}^K q_u \Gamma_u' D E_u (I - P^\infty) \quad (4.14)$$

bağıntısıyla tanımlanır.

Üçüncü büyüklük ise  $\Gamma_u$  matrislerinin ortalama değerini belirten  $C_2$  büyüklüğüdür ve

$$C_2 = \sum_{u=1}^K q_u \Gamma_u \quad (4.15)$$

bağıntısıyla verilir.

Son olarak da, kod sözcüğü ortalama değerini veren  $m_c$  değeri

$$m_c = \sum_{u=1}^K q_u p \Gamma_u \quad (4.16)$$

eşitliğiyle verilir.

Sonuç olarak kod sözcük dizilerine ait güç spektral yoğunluğunun sürekli bileşeni  $W_c^{(c)}(f)$ :

$$W_c^{(c)}(f) = T[Y(z) + Y'(z^{-1})] \quad (4.17)$$

şeklinde elde edilir. Burada;

$$Y(z) = \frac{1}{2}(C_0 - m_c' m_c) + C_1 B(z) C_2 \quad (4.18)$$

olarak tanımlıdır.

Artık, kod simgelerine ilişkin güç spektral yoğunluğu ifadesini elde edebiliriz:

$$W_c^{(c)}(f) = V^*(f) W_c^{(c)}(f) V(f) \quad (4.19)$$

bağıntısıyla kod simgelerine ilişkin güç spektral yoğunluğunun sürekli bileşeni tanımlanır. Burada  $V(f)$  ifadesi, (Şekil 4.1)'de görülen ve çıkışında kod simgelerinin üretildiği paralel - seri dönüştürücüye ait frekans yanıtını göstermektedir ve

$$V(f) = \frac{1}{N} [1, \exp(-j2\pi f T_c), \dots, \exp(-j2\pi f (N-1) T_c)] \quad (4.20)$$

eşitliğiyle tanımlanır.

Burada hemen belirtmek gerekir ki, eğer işaretleme aralığı  $T$  ise kodsözcüğündeki her bir simge için gerekli süre  $T_c = (1/N)T$ 'dir.

(Tablo 4.1.a)'daki 1 numaralı kodlayıcı ve (Tablo 4.2.a)'daki 2 numaralı kodlayıcı yapısı için elde edilen spektral çözümleme sonuçları (Şekil 4.9)'da gösterilmiştir.

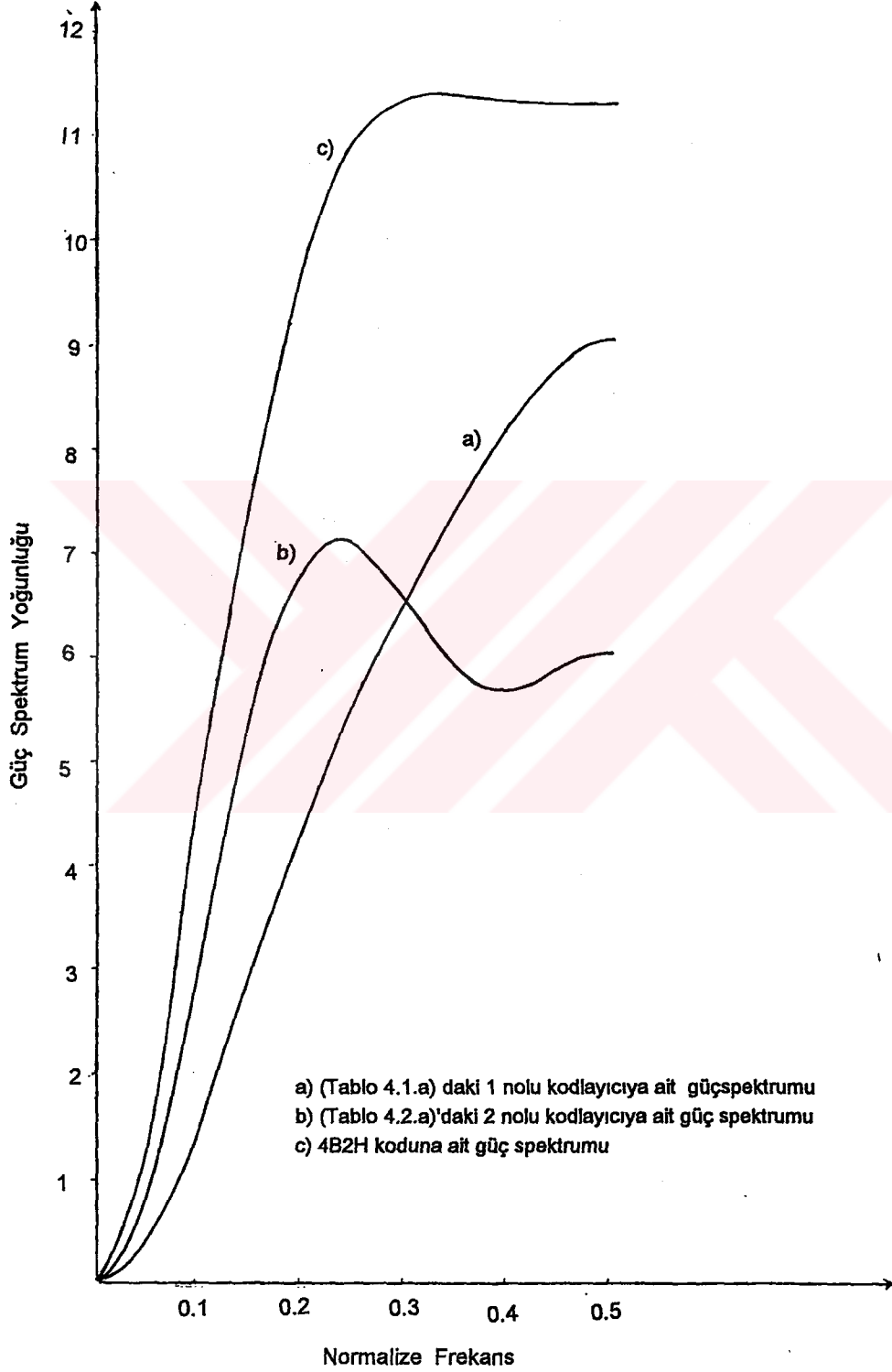
Çok düzeyli olarak tasarlanan hat kodlayıcılarının spektral analizinde gerekli olan durum geçiş matrisi, kodlayıcı çıkış matrisi gibi yukarıda tanımlanan matrislerin elde edilmesi için, her düzeydeki kodlayıcıların tek bir

kafes yapısı oluşturulmalıdır. Örneğin 1. düzeyde 4 durumlu ve 2. düzeyde 1 durumlu katlamalı kodlayıcı için eşdeğer kafes yapı  $4.2=8$  durumlu olacaktır.

Spektral çözümlemesi yapılan örnek kodlayıcılardan (Tablo 4.1.a)'daki 1 numaralı kodlayıcı için tek bir kafes yapı oluşturulmuş ve buna ait durum geçiş matrisleri ile kodlayıcı çıkış matrisleri [Ek 2]'de verilmiştir.

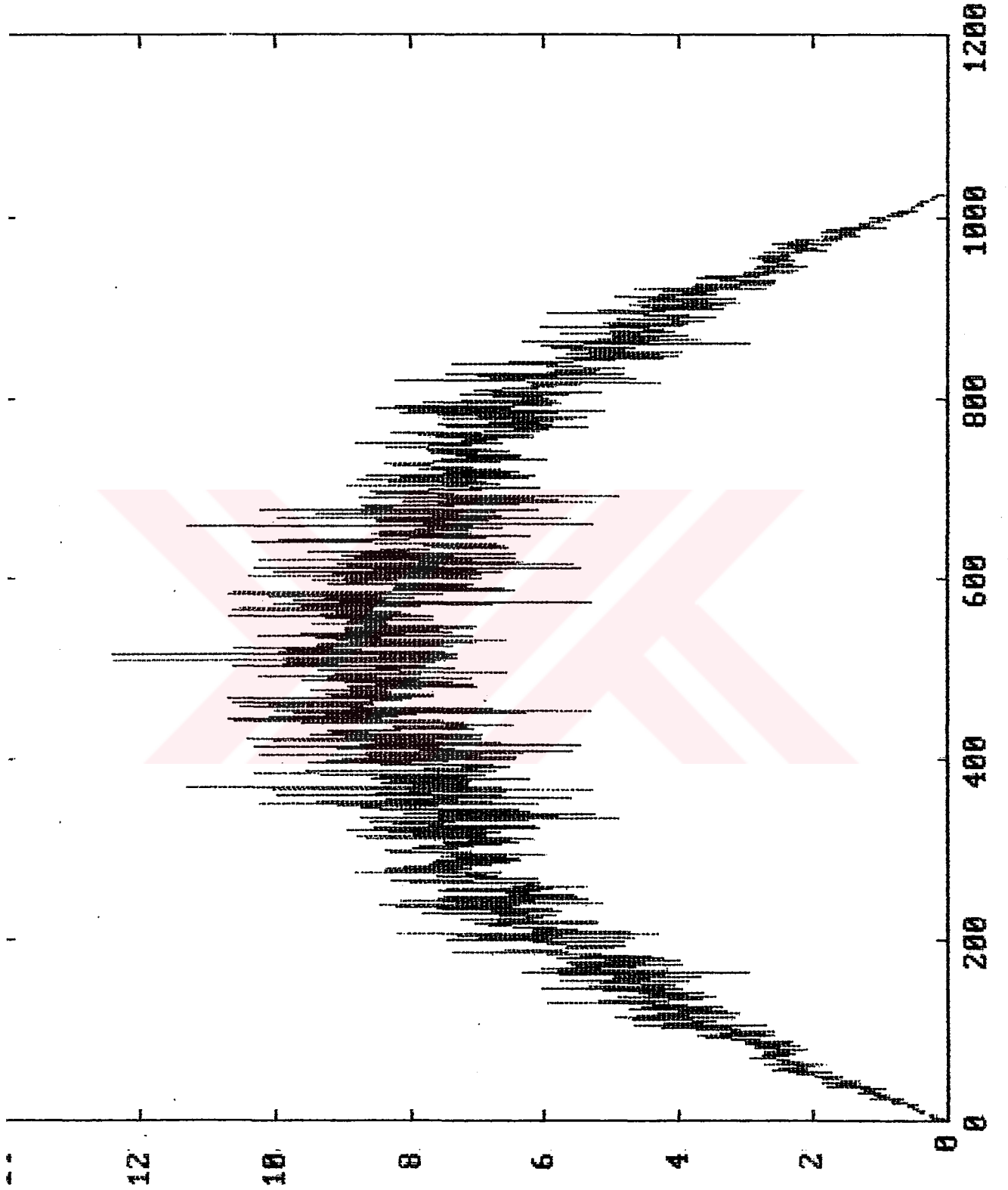
Sonuçlardan da görüleceği gibi, tasarlanan kodlayıcıların doğru akım bileşeni sıfır ve alçak frekans bölgesindeki spektral bileşenleri de oldukça zayıflamış olarak elde edilmiştir. (Şekil 4.9)'da aynı zamanda Botha [Botha et al,1993] tarafından elde edilen '4B2H' koduna ait güç spektrumu da verilmiştir. Görüldüğü gibi, tasarlanan kodlayıcıların güç spektrumlarının genlik değerleri alçak frekans bölgesinde, '4B2H' koduna ait güç spektrumu genlik değerlerinden daha küçük olarak elde edilmiştir.

Spektral çözümlemesi yapılan kodlayıcının 8 durumlu ve bir durumdan diğerine 4 paralel geçiş olmak üzere 32 çıkışlı olduğu gözönünde bulundurulursa, karmaşıklığı daha fazla olan kodlayıcılar için, durum geçiş, durum olasılık ve kodlayıcı çıkış matrislerinin elde edilmesi de o ölçüde zorlaşacaktır. Bu nedenle bir kodlayıcının spektral çözümlemesi için daha basit olan şöyle bir yol izlenebilir: Üretilen rastgele girişlere karşılık kodlayıcının ürettiği  $N$  adet çıkış, yine  $N$  elemanlı bir  $x(n)$  dizisine atanır. Daha sonra bu dizinin  $N$  noktalı hızlı Fourier dönüşümü (fast Fourier transform=FFT) ile  $X(f)$  elde edilir. Bu işlem bir çok kez tekrarlanarak, elde edilen sonuçların ortalama değeri, bize bu kodlayıcının spektral özelliği açısından bir fikir verebilir. Yukarıda çözümlenmeleri elde edilen kodlayıcıların, son olarak anlatılan benzetim yöntemiyle bulunan spektral gösterimleri, sırasıyla (Şekil 4.10) ve (Şekil 4.11)'de verilmiştir.

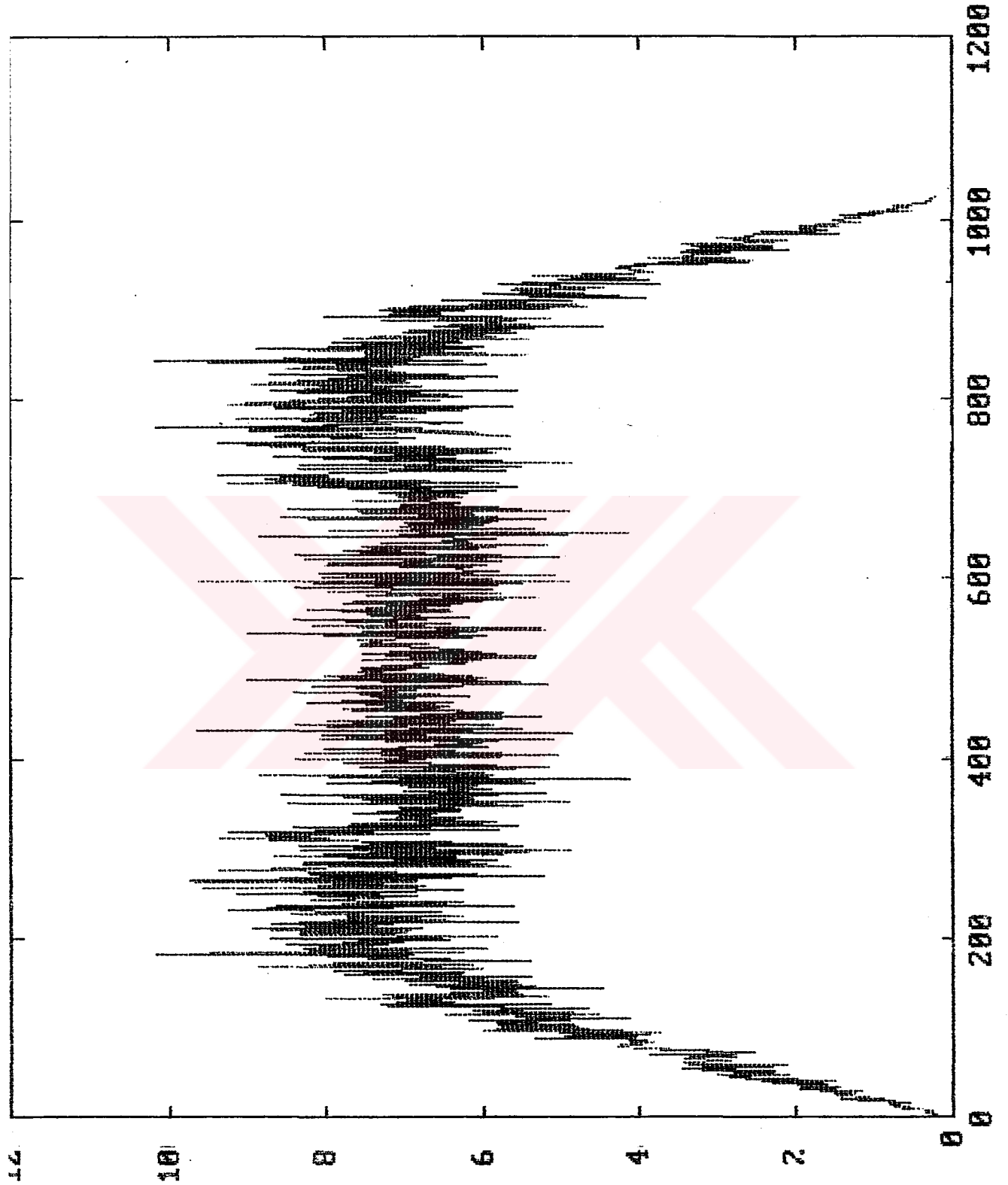


(Şekil 4.9) Tasarlanan 4-AM hat kodlayıcılarına ve 4B2H kodlayıcısına ilişkin güç spektrumları





(Şekil 4.10) (Tablo 4.1.a)'daki 1 nolu kodlayıcının benzetim yoluyla elde edilen güç spektrumu



(Şekil 4.11) (Tablo 4.2.a)'daki 2 nolu kodlayıcının benzetim yoluyla elde edilen güç spektrumu

## SONUÇ

Çok düzeyli kodlama ve çok aşamalı kodçözme tekniğine dayalı olarak, sayısal genlik modülasyonu (M-AM, M=4 ve M=8) için 2 ve 3 düzeyli, Imai - Hirakawa tipi kodlayıcılar tasarlanmıştır. Tasarlanan kodlayıcılar, kodlamasız ilişkin modülasyon türlerine göre 3-5 dB arasında, Ungerboeck tipi kafes kodlayıcılara göre ise 2 dB'ye varan kodlama kazançları sağlamaktadırlar.

4-AM modülasyon türü için tasarlanan (Tablo 3.2.a)'daki IHS-12 kodlayıcısı, 4.15dB'lik kodlama kazancını, karmaşıklık değeri 18.67 iken sağlarken, delikli katlamalı kodlayıcı kullanılması durumunda aynı kazanç değeri karmaşıklık değeri 8 iken sağlanmaktadır. (Tablo 3.1.a)'daki Ungerboeck tipi Ung-4 kodlayıcısı ise 4.2 dB'lik kazancı karmaşıklık değeri 32 iken elde etmektedir.

8-AM modülasyon türü için tasarlanan (Tablo 3.3.b)'deki IHS-37 kodlayıcısı, 4.907 dB'lik kazancı 26.4 karmaşıklık değerinde sağlarken, (Tablo 3.1.b)'deki Ung-10, 5.0 dB'lik kodlama kazancını 96 karmaşıklık değerinde sağlamaktadır. Yine aynı modülasyon türü için, IHS-38 ile 5.809 dB'lik kazanç 27.2 karmaşıklık değeriyle elde edilirken, Ung-12 aynı değerdeki kodlama kazancını 384 karmaşıklık değeriyle elde etmektedir.

Çok düzeyli kodlamanın, çok aşamalı kod çözme yönteminin kullanılmasına olanak vermesinden dolayı, düşük kod çözme karmaşıklığı ile, önemli miktarda kodlama kazancı sağlanabilmektedir.

Temelbant veri iletişimde, bir çift tel üzerinden iletilecek olan ikili sözcüklerin, doğru akım bileşenlerinin olması, iletim için elverişli olmayan bir durumdur. Ayrıca bu işaretlerin bant genişliklerinin de küçük olması istenir. Bu özellikler, hat kodlayıcılarının kullanılması ile sağlanabilir. Bu amaca yönelik olarak, doğru akım bileşeni


içermeyen ve alçak frekans bölgesinde düşük genlikli spektral bileşenlere sahip 2 düzeyli 4-AM hat kodları tasarlanmış ve güç spektrumları elde edilmiştir.



**KAYNAKLAR**

- [1] Ü. Aygözü and E. Panayırıcı, "New Ternary Line Codes Based on Trellis Structure", IEEE Trans Commun. vol.41, no.6, pp.864-873, June 1993.
- [2] Ü. Aygözü and E. Panayırıcı, "Multilevel Ternary Line Codes with Trellis Structure", IEE Proc-Commun vol.141, no.1, pp.7-12, Feb.1994.
- [3] S. Benedetto, E. Biglieri and V. Castellani "Digital Transmission Theory", Ch.9, Prentice Hall, 1987.
- [4] E. Biglieri, "Ungerboeck Codes Do Not Shape the Signal Power Spectrum", IEEE Trans. Inform. Theory, vol.32, no.4, pp.595-596, July 1986.
- [5] E. Biglieri, D. Divsalar, P.J. Mclane and M.K.Simon "Introduction to TCM with Applications", Macmillan, 1991.
- [6] L. Botha, H.C. Ferreira and I. Broere, "Multilevel Sequences and Line Codes", IEE Proc.1, vol.140, No.4, PP.255-261, August 1993.
- [7] A.R. Calderbank and J.E. Mazo, "Baseband Line Codes via Spectral Factorization", IEEE J. Select. Areas Commun. vol.7, no.6, pp.914-928, August 1989.
- [8] G. Cariolaro, G. Pierobon and G. Tronca, "Analysis of Codes and Spectral Calculations", Int. J. Electronics, vol.55, no.1, pp.35-79, 1983.

- [9] D.G. Daut, J.W. Modestino and L.D. Wismer, "New Short Constraint Length Convolutional Code Constructions for Selected Rational Rates", IEEE Trans. Inform. Theory, vol.28, no.5, pp.794-800, Sep.1982.
- [10] H.Imai and S. Hirakawa, "A New Multilevel Coding Method Using Error-Correcting Codes", IEEE Trans. Inform. Theory, vol.23,no.3, pp.371-377,May 1977.
- [11] Y. Kofman, E. Zehavi and S. Shamai, "Analysis of a Multilevel Coded Modulation System",EE Pub.no.751, Technion, Israel,1990.
- [12] G.J.Pottie and D.P. Taylor, "Multilevel Codes Based on Partitioning", IEEE Trans. Inform. Theory vol.35 no.1, pp.87-98, Jan 1989.
- [13] J.G. Proakis, "Digital Communications",Ch.5,MacGraw Hill, 1983.
- [14] B. Sklar, "Defining, Designing, and Evaluating Digital Communication Systems", IEEE Comm. Magazine, Nov. 1993.
- [15] G. Ungerboeck, "Channel Coding with Multilevel/Phase Signals", IEEE Trans. Inform. Theory, vol.28 no.1, pp.55-67, Jan.1982.
- [16] G. Ungerboeck, "Trellis-Coded Modulation with Redundant Signal Sets Part I", IEEE Commun. Magazine, vol.25, no.2, pp.5-11, Feb.1987.

- [17] K. Yamaguchi and H. Imai, "Highly Reliable Multilevel Channel Coding System Using Binary Convolutional Codes", *Electronics Letters*, vol.23 no.18, pp.939-941, August 1987.
- [18] Y. Yasuda, K.Kashiki and Y. Hirata, "High-Rate Punctured Convolutional Codes for Soft Decision Viterbi Decoding", *IEEE Trans. Commun.* vol.32, no.3, pp.315-319, March 1984.
- [19] E. Zehavi and J.K. Wolf, "On the Performance Evaluation of Trellis Codes", *IEEE Trans. Inform. Theory*, vol.33, no.2, pp.196-202, March 1987.
- 

## EK-1

(Bilgisayar Benzetim Programı)

```

Program Multistage (input,output);
uses crt;
Var
M1,M2,N11,N21,N12,N22,N13,N23,N3,N14,N24,TUR,I1,I2, integer;
J1,J2,P1,P2,L,L1,L2,ALTK,EL,TR,H,AD,K,KZ1,KZ2,KE:integer;
C1,C2,E11,E12,E21,E22,M11,K11,K12,M21,K21,K22:integer;
K22,MY1,MY2,PE1,K1,K2,DNG,BHO:integer;
V,Z,R1,R2,F1,F,A1,A2,THATA,KON1,KON2:real;
U1: Array [1..8,1..8,1..8] of integer;
U2: Array [1..8,1..8,1..8] of integer;
S1: Array [1..8,1..8,1..9] of integer;
S2: Array [1..8,1..8,1..9] of integer;
X1: Array [1..14,1..8] of integer;
X2: Array [1..14,1..8] of integer;
T: Array [1..14,1..8] of integer;
W1: Array [1..8,1..8,1..9] of real;
W2: Array [1..8,1..8,1..9] of real;
R: Array [1..14,1..8] of real;
LMIN1: Array [1..24,1..9] of real;
LMIN2: Array [1..24,1..9] of real;
NSUR1: Array [1..24,1..9] of integer;
NSUR2: Array [1..24,1..9] of integer;
D11: Array [1..30] of integer;
D12: Array [1..30] of integer;
D21: Array [1..30] of integer;
D22: Array [1..30] of integer;
G: Array [1..14,1..8] of real;
Y1: Array [1..8,1..8] of real;
Y2: Array [1..8,1..8] of real;
SS: Array [0..15,1..10] of integer;
ALT: Array [1..14,1..8,1..8] of integer;
DALT: Array [1..14,1..8] of integer;
KKD: Array [1..8,1..8,0..12,1..8] of integer;
PE: Array[1..200] of real;
SPE:Array[1..10] of real;
BT11,BT12,BT21,BT22:Array[1..10,1..5] of integer;
LABEL J2700,J2800,J2900,J2910,J3500,J3600,J3710;
LABEL J3810,J3900,J3901,J4000,J7000;J7100,J7550;
LABEL J7600,J8000,J8100,J8300,J8800,J8550,J8600;
LABEL J9000,J9100,J9300,J9800,J10000,J10010,J10900,J11020;
Begin
clrscr;
RANDOMIZE;

{...KODLAYICI BİLGİLERİ GİRİLECEK...}
WRITE ('1.Duzey Adım Uzunlugu=');READ(M1);
WRITE ('2.Duzey Adım Uzunlugu=');READ(M2);

```



```

WRITE ('2.Duzey Adım Uzunlugu=');READ(M2);
WRITE ('1.Duzey Durum Sayısı=');READ(N11);
WRITE ('2.Duzey Durum Sayısı=');READ(N21);
WRITE ('1. DUZEY DURUM CIKIS SAYISI=');READ(N12);
WRITE ('2. DUZEY DURUM CIKIS SAYISI=');READ(N22);
WRITE ('1.DUZEY KOD KELIMESI SAYISI=');READ(N13);
WRITE ('2.DUZEY KOD KELIMESI SAYISI=');READ(N23);
WRITE ('BIRLESIK KOD KELIMESI SAYISI=');READ(N3);
WRITE ('1.DUZEY GIRIS DIJIT SAYISI=');READ(N14);
WRITE ('2.DUZEY GIRIS DIJIT SAYISI=');READ(N24);
WRITE ('TUR SAYISI='); READ(TUR);
FOR I1:=1 TO N11 DO
  BEGIN
  WRITELN ('1. DUZEY',I1,'.NCI DURUM VERILERI');
  FOR J1:=1 TO N12 DO
    BEGIN
    FOR P1:= 1 TO N14 DO
      BEGIN
      WRITE ('U1(',I1,',',',',J1,',',',',P1,')=');READLN(U1[I1,J1,P1]);
      END;
      WRITE (J1,'NCI CIKIS ILE GITTIGI DURUM NO=');
      READLN (S1[I1,J1,1]);
      FOR L1:=1 TO N13 DO
        BEGIN
        WRITE (L1,'NCI DIGIT=');READLN (S1[I1,J1,L1+1]);
        END;
      END;
    END;
  FOR I2:=1 TO N21 DO
    BEGIN
    WRITELN ('2. DUZEY',I2,'.NCI DURUM VERILERI');
    FOR J2:=1 TO N22 DO
      BEGIN
      FOR P2:=1 TO N24 DO
        BEGIN
        WRITE ('U2(',I2,',',',',J2,',',',',P2,')=');
        READLN (U2[I2,J2,P2]);
        END;
        WRITE (J2,'NCI CIKIS ILE GITTIGI DURUM NO=');
        READLN (S2[I2,J2,1]);
        FOR L2:=1 TO N23 DO
          BEGIN
          WRITE (L2,'NCI DIGIT=');READLN (S2[I2,J2,L2+1]);
          END;
        END;
      END;
    END;
  WRITE ('DENEME SAYISI=');READLN(H);
  v:=0.4;
  FOR DNG:=1 TO 10 DO
    BEGIN
    v:=v+0.05;

```

```

THATA:=0;
CLRSCR;
FOR TR:=1 TO TUR DO
BEGIN
BHO:=0;
FOR AD:=1 TO H DO
  BEGIN
    I1:=1; I2:=1;
    FOR K:=1 TO M1 DO
      BEGIN
        FOR P1:=1 TO N14 DO
          BEGIN
            Z:=RANDOM;
            IF Z >= 0.5 THEN begin X1[K,P1]:=1; GOTO J2800;end;
            X1[K,P1]:=0;
J2800: END;
            FOR P2:=1 TO N24 DO
              BEGIN
                Z:= RANDOM;
                IF Z >= 0.5 THEN begin X2[K,P2]:=1; GOTO J2910;end;
                X2[K,P2]:=0;
J2910: END;
                FOR J1:=1 TO N12 DO
                  BEGIN
                    FOR P1:=1 TO N14 DO
                      BEGIN
                        IF X1[K,P1] <> U1[I1,J1,P1] THEN GOTO J3500;
                        END;
                      GOTO J3600;
J3500: END;
J3600: FOR J2:=1 TO N22 DO
                    BEGIN
                      FOR P2:=1 TO N24 DO
                        BEGIN
                          IF X2[K,P2] <> U2[I2,J2,P2] THEN GOTO J3710;
                          END;
                          GOTO J3810;
J3710: END;
J3810: FOR L:=1 TO N3 DO
                    BEGIN
                      IF S1[I1,J1,1+L]=0 THEN GOTO J3900;
                      GOTO J3901;
J3900: IF S2[I2,J2,1+L]=0 THEN begin T[K,L]:=-3; GOTO
J4000;end;
                      T[K,L]:=1; GOTO J4000;
J3901: IF S2[I2,J2,1+L]=0 THEN begin T[K,L]:=-1; GOTO
J4000;end;
                      T[K,L]:=3;
J4000: END;
                    D11[K]:=I1; D21[K]:=I2;
                    I1:= S1[I1,J1,1];

```

```

I2:= S2[I2,J2,1];
FOR P1:=1 TO N14 DO
  BEGIN
    BT11[K,P1]:=U1[I1,J1,P1];
  END;
FOR P2:=1 TO N24 DO
  BEGIN
    BT21[K,P2]:=U2[I2,J2,P2];
  END;
END;
KZ1:= M1+1; KZ2:= M2+1;
D11[KZ1]:=I1; D21[KZ2]:=I2;
FOR K:=1 TO M1 DO
  BEGIN
    FOR L:=1 TO N3 DO
      BEGIN
        R1:=RANDOM;
        R2:=RANDOM;
        R[K,L]:=T[K,L]+V*SQRT(-2*LN(R1))*COS(6.28318*R2);
      END;
    END;
    {...MULTISTAGE KOD COZME....}
FOR K:=1 TO M1 DO
  BEGIN
    KE:=K-1;
    F1:=999999;
    FOR C1:=1 TO N11 DO
      BEGIN
        LMIN1[K,C1]:=99999;
        FOR I1:=1 TO N11 DO
          BEGIN
            FOR J1:=1 TO N12 DO
              BEGIN
                IF S1[I1,J1,1] <> C1 THEN GOTO J8000;
                Y1[I1,J1]:=0;
                FOR L:=1 TO N3 DO
                  BEGIN
                    KON1=R[K,L]-SS[S1[I1,J1,L+1],1];
                    KON2=R[K,L]-SS[S1[I1,J1,L+1],2];
                    IF ABS(KON1)>ABS(KON2) THEN GOTO J7000;
                    W1[I1,J1,L+1]:=SQR(ABS(R[K,L]-SS[S1[I1,J1,L+1],1]));
                    GOTO J7100;
                  J7000: W1[I1,J1,L+1]:=SQR(ABS(R[K,L]-SS[S1[I1,J1,L+1],2]));
                  J7100: Y1[I1,J1]:= Y1[I1,J1]+W1[I1,J1,L+1];
                  END;
                IF KE=0 THEN GOTO J7550;
                A1:=Y1[I1,J1]+LMIN1[KE,I1];
                GOTO J7600;
              J7550: A1:=Y1[I1,J1];
              J7600: IF A1 >= LMIN1[K,C1] THEN GOTO J8100;
                LMIN1[K,C1]:=A1;

```

```

        NSUR1[K,C1]:=I1;
        FOR L:=1 TO N3 DO
            BEGIN
                ALT[K,C1,L]:=S1[I1,J1,L+1];
            END;
        GOTO J8100;
J8000:  END;
J8100:  IF K=1 THEN GOTO J8300;
        END;
J8300:  END;
        END;
        FOR C1:=1 TO N11 DO
            BEGIN
                K:=M1;
                IF LMIN1[K,C1] >= F1 THEN GOTO J8800;
                F1:= LMIN1[K,C1];
                D12[KZ1]:=C1;
                D12[K]:=NSUR1[K,C1];
                FOR L:=1 TO N3 DO
                    BEGIN
                        DALT[K,L]:=ALT[K,C1,L];
                    END;
J8800:  END;
                E11:=D12[K];
                M11:=M1-1;
                FOR K11:=1 TO M11 DO
                    BEGIN
                        K12:=M1-K11;
                        D12[K12]:=NSUR1[K12,E11];
                        FOR L:=1 TO N3 DO
                            BEGIN
                                DALT[K12,L]:=ALT[K12,E11,L];
                            END;
                        E11:=D12[K12];
                    END;
                {.... 1.DUZEY KOD COZME SONU ....}
                {      2.DUZEY KOD COZME      }
FOR K:=1 TO M2 DO
    BEGIN
        KE:=K-1;
        F:=999999;
        FOR C2:=1 TO N21 DO
            BEGIN
                LMIN2[K,C2]:=99999;
                FOR I2:=1 TO N21 DO
                    BEGIN
                        FOR J2:=1 TO N22 DO
                            BEGIN
                                IF S2[I2,J2,1] <> C2 THEN GOTO J9000;
                                Y2[I2,J2]:=0;
                            END;
                        FOR L:=1 TO N3 DO

```

```

      BEGIN
      W2[I2,J2,L+1]:=SQR(ABS(R[K,L]-
      KKD[I2,J2,DALT[K,L],L]));
      Y2[I2,J2]:=Y2[I2,J2]+W2[I2,J2,L+1];
      END;
      IF KE=0 THEN GOTO J8550;
      A2:=-Y2[I2,J2]+LMIN2[KE,I2];
      GOTO J8600;
J8550: A2:=Y2[I2,J2];
J8600: IF A2 >= LMIN2[K,C2] THEN GOTO J9100;
      LMIN2[K,C2]:=A2;
      NSUR2[K,C2]:=I2;
      GOTO J9100;
J9000: END;
J9100: IF K=1 THEN GOTO J9300;
      END;
J9300: END;
      END;
      FOR C2:=1 TO N11 DO
      BEGIN
      K:=M2;
      IF LMIN2[K,C2] >= F THEN GOTO J9800;
      F:=LMIN2[K,C2];
      D22[KZ2]:=C2;
      D22[K]:=NSUR2[K,C2];
J9800: END;
      E21:=D22[K];
      M21:=M2-1;
      FOR K21:=1 TO M21 DO
      BEGIN
      K22:=M2-K21;
      D22[K22]:=NSUR2[K22,E21];
      E21:=D22[K22];
      END;
      { KARAR VERILİYOR }
      FOR K1:=1 TO M1 DO
      BEGIN
      FOR I1:=1 TO N11 DO
      BEGIN
      FOR J1:=1 TO N12 DO
      BEGIN
      IF D12[K1+1] <> S1[I1,J1,1] THEN GOTO J10000;
      FOR P1:=1 TO N14 DO
      BT12[K1,P1]:=U1[I1,J1,P1];
J10000: END;
      END;
      END;
      FOR K2:=1 TO M2 DO
      BEGIN
      FOR I2:=1 TO N21 DO
      BEGIN

```

```
FOR J2:=1 TO N22 DO
BEGIN
IF D22[K2+1] <> S2[I2,J2,1] THEN GOTO J10010;
FOR P2:=1 TO N24 DO
BT22[K2,P2]:=U2[I2,J2,P2];
J10010: END;
END;
END;
FOR K1:=1 TO M1 DO
BEGIN
FOR P1:=1 TO N14 DO
IF BT11[K1,P1] <> BT12[K1,P1] THEN BHO:=BHO+1;
END;
FOR K2:=1 TO M2 DO
BEGIN
FOR P2:=1 TO N24 DO
IF BT21[K2,P2] <> BT22[K2,P2] THEN BHO:=BHO+1;
END;
END;
THATA:=THATA+BHO/(H*M1*(N14+N24));
WRITELN(TR, '.HATA OLASILIGI=', THATA);
END;
SPE[DNG]:=THATA/TUR;
WRITE ('SONUC HATA OLASILIGI=', SPE[DNG]);
END;
END.
```

## EK-2

{spektrum çözümlemesinde elde edilen  
 $E_i$  ve  $\Gamma_i$  matrisleri}

e1 =e3=e5=e7

1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0

» e2

e2 =e2=e4=e6

0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0

»

e9 =e11=e13=e15

0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	0	0	0	0	0

» e10

e10 =e12=e14=e16

0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	0

»

e17 =e19=e21=e23

0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	1	0	0	0

» e18

e18 =e20=e22=e24

0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0

»

e25 =e27=e29=e31

0	0	0	0	0	0	1	0
0	0	0	0	0	0	1	0
0	0	0	0	0	0	1	0
0	0	0	0	0	0	1	0
0	0	0	0	0	0	1	0
0	0	0	0	0	0	1	0
0	0	0	0	0	0	1	0
0	0	0	0	0	0	1	0

» e26

e26 =e28=e30=e32

0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1

»





g5 =

-1	3	1	-3	3	-3
-1	1	1	-3	3	-3
3	-1	1	-3	-3	3
1	-1	1	-3	-3	3
3	-1	-3	1	3	-3
1	-1	-3	1	3	-3
-1	3	-3	1	-3	3
-1	1	-3	1	-3	3

» g6

g6 =

-1	3	1	-3	-1	3
-1	1	1	-3	-1	3
3	-1	1	-3	3	-1
1	-1	1	-3	3	-1
3	-1	-3	1	-1	3
1	-1	-3	1	-1	3
-1	3	-3	1	3	-1
-1	1	-3	1	3	-1

»

g7 =

-1	3	-1	1	1	-3
-1	1	-1	1	1	-3
3	-1	-1	1	-3	1
1	-1	-1	1	-3	1
3	-1	1	-1	1	-3
1	-1	1	-1	1	-3
-1	3	1	-1	-3	1
-1	1	1	-1	-3	1

» g8

g8 =

-1	3	-1	1	-1	1
-1	1	-1	1	-1	1
3	-1	-1	1	1	-1
1	-1	-1	1	1	-1
3	-1	1	-1	-1	1
1	-1	1	-1	-1	1
-1	3	1	-1	1	-1
-1	1	1	-1	1	-1

»

g9 =

-	0	0	0	0	0
-	0	1	0	0	0
-	0	0	1	0	0
-	0	0	0	1	0
-	0	0	0	0	1
-	0	0	0	0	0

» g10

g10 =

-	0	0	0	0	1
-	0	1	0	0	1
-	0	0	1	0	0
-	0	0	0	1	0
-	0	0	0	0	1
-	0	0	0	0	0

»

g11 =

-	0	0	-1	-0	1
-	0	1	-1	-0	1
-	0	0	-1	1	-0
-	1	-0	-1	1	-0
-	0	-0	-1	-0	1
-	1	-0	0	-0	1
-	0	0	-1	1	-0
-	0	1	-1	1	-0

» g12

g12 =

-	0	0	-1	1	-1
-	0	1	-1	1	-1
-	0	0	-1	-1	1
-	1	-0	-1	-1	1
-	0	-0	-1	1	-1
-	1	-0	0	1	-1
-	0	-1	0	-1	1
-	0	1	0	-1	1

»

g13 =

3	-1	-3	1	-3	3
1	-1	-3	1	-3	3
-1	3	-3	1	3	-1
-1	1	-3	1	3	-1
-1	3	1	-3	-3	3
-1	1	1	-3	-3	3
3	-1	1	-3	3	-1
1	-1	1	-3	3	-1

» g14

g14 =

3	-1	-3	1	3	-1
1	-1	-3	1	3	-1
-1	3	-3	1	-1	3
-1	1	-3	1	-1	3
-1	3	1	-3	3	-1
-1	1	1	-3	3	-1
3	-1	1	-3	-1	3
1	-1	1	-3	-1	3

»

g15 =

3	-1	1	-1	-3	1
1	-1	1	-1	-3	1
-1	3	1	-1	1	-3
-1	1	1	-1	1	-3
-1	3	-1	1	-3	1
-1	1	-1	1	-3	1
3	-1	-1	1	1	-3
1	-1	-1	1	1	-3

» g16

g16 =

3	-1	1	-1	1	-1
1	-1	1	-1	1	-1
-1	3	1	-1	-1	1
-1	1	1	-1	-1	1
-1	3	-1	1	1	-1
-1	1	-1	1	1	-1
3	-1	-1	1	-1	1
1	-1	-1	1	-1	1

»

q17 =

1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1

» q18

q18 =

1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1

»

q19 =

1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1

» q20

q20 =

1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1

»

g21 =

3	-1	1	-3	-1	3
1	-1	1	-3	-1	3
-1	3	1	-3	3	-1
-1	1	1	-3	3	-1
-1	3	-3	1	-1	3
-1	1	-3	1	-1	3
3	-1	-3	1	3	-1
1	-1	-3	1	3	-1

» g22

g22 =

3	-1	1	-3	3	-1
1	-1	1	-3	3	-1
-1	3	1	-3	-1	3
-1	1	1	-3	-1	3
-1	3	-3	1	3	-1
-1	1	-3	1	3	-1
3	-1	-3	1	-1	3
1	-1	-3	1	-1	3

»

g23 =

3	-1	-1	1	-3	1
1	-1	-1	1	-3	1
-1	3	-1	1	1	-3
-1	1	-1	1	1	-3
-1	3	1	-1	-3	1
-1	1	1	-1	-3	1
3	-1	1	-1	1	-3
1	-1	1	-1	1	-3

» g24

g24 =

3	-1	-1	1	1	-1
1	-1	-1	1	1	-1
-1	3	-1	1	-1	1
-1	1	-1	1	-1	1
-1	3	1	-1	1	-1
-1	1	1	-1	1	-1
3	-1	1	-1	-1	1
1	-1	1	-1	-1	1

»

g25 =

0	-0	-1	0	0	0
1	-0	-0	-1	-1	-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

» g26

g26 =

0	-0	-1	0	-1	0
1	-0	-0	-1	-1	-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

»

g27 =

0	-0	0	-1	1	-0
1	-0	0	-1	1	-0
-0	0	0	-1	-0	1
-0	0	-1	0	1	-0
-0	1	-1	0	1	-0
1	-0	-1	0	-0	1

» g28

g28 =

0	-0	0	-1	-1	1
1	-0	0	-1	-1	1
-0	0	0	-1	1	-1
-0	0	-1	-1	1	-1
-0	1	-1	0	-1	1
1	-0	-1	0	1	-1

»

g29 =

-1	3	-3	1	3	-3
-1	1	-3	1	3	-3
3	-1	-3	1	-3	3
1	-1	-3	1	3	-3
3	-1	1	-3	3	-3
1	-1	1	-3	3	-3
-1	3	1	-3	-3	3
-1	1	1	-3	-3	3

» g30

g30 =

-1	3	-3	1	-1	3
-1	1	-3	1	-1	3
3	-1	-3	1	3	-1
1	-1	-3	1	3	-1
3	-1	1	-3	-1	3
1	-1	1	-3	-1	3
-1	3	1	-3	3	-1
-1	1	1	-3	3	-1

»

g31 =

-1	3	1	-1	1	-3
-1	1	1	-1	1	-3
3	-1	1	-1	-3	1
1	-1	1	-1	-3	1
3	-1	-1	1	1	-3
1	-1	-1	1	1	-3
-1	3	-1	1	-3	1
-1	1	-1	1	-3	1

» g32

g32 =

-1	3	1	-1	-1	1
-1	1	1	-1	-1	1
3	-1	1	-1	1	-1
1	-1	1	-1	1	-1
3	-1	-1	1	-1	1
1	-1	-1	1	-1	1
-1	3	-1	1	1	-1
-1	1	-1	1	1	-1

»



## ÖZGEÇMİŞ

1970 yılında Adıyaman'ın Besni ilçesinde doğdu. 1987 yılında Akşehir Lisesi'nde orta öğrenimini tamamladıktan sonra aynı yıl, Yıldız Üniversitesi Mühendislik Fakültesi Elektronik ve Haberleşme Mühendisliği Bölümü'nde yüksek öğrenimine başladı. 1991 yılında bu bölümden mezun oldu ve aynı üniversitenin Fen Bilimleri Enstitü'sü Haberleşme Bilim Dalı'nda yüksek lisans öğrenimine başladı. 1993 yılında Elektronik ve Haberleşme Mühendisliği Bölümünde araştırma görevlisi olarak çalışmaya başladı. Halen aynı göreve devam etmektedir.

ÖZGEÇMİŞ KURULU  
KARAGÖZÜMÜSÜ