

**1200 bit/sn HIZLI BİR DÜŞÜK BİT HIZLI KONUŞMA
KODLAYICI TASARIMI**

Selma ÖZAYDIN

DOKTORA TEZİ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ

T.C. YÜKSEKÖĞRETİM KURULU
DOKÜMANTASYON MERKEZİ

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

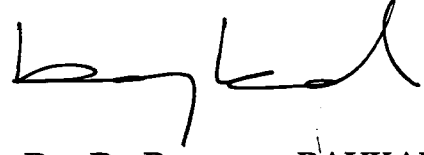
Nisan 2001

ANKARA

11465

✓

Selma ÖZAYDIN tarafından hazırlanan 1200 bit/sn HIZLI BİR DÜŞÜK BİT HIZLI KONUŞMA KODLAYICI TASARIMI adlı bu tezin Doktora tezi olarak uygun olduğunu onaylarım.



Doç.Dr. Buyurman BAYKAL

Tez yöneticisi

Bu çalışma, jürimiz tarafından Elektrik-Elektronik Mühendisliği Anabilim Dalında Doktora tezi olarak kabul edilmiştir.

Başkan : Prof. Dr. Münyen SARITAŞ M. Saritaş

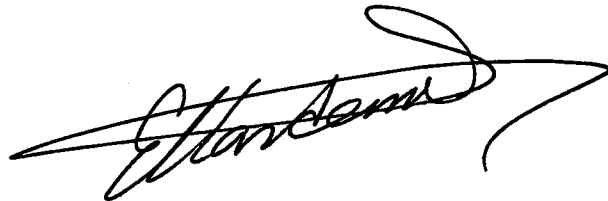
Üye : Doç. Dr. Buyurman Baykal Baykal

Üye : Doç Dr Gözde Bozdağı Akar G. Bozdağı

Üye : Doç. Dr. Tolga Çilbaş T. Çilbaş

Üye : Yrd. Doç. Dr. Erkan Afercan Afercan

Bu tez, Gazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygundur.



Kalite kavramının, bina üretim sürecinde bu şekilde biçimlenmesi gereğine bağlı olarak, yapı hasarları ve konfor koşullarının incelenmesi için sınırlılık alanı olarak tünel kalıp tekniği ile üretilen konut örnekleri seçilerek, üretim sistemi, bileşen analizi ve ısı konfor inceleme başlıkları olarak seçilmiştir. Bunda amaç, üretim tekniğinin, strüktürel sistem, bileşen tasarım ve üretiminin literatür taraması sonucu elde edilen standartlara ve şartnamelere uygun olup olmamasının belirleyiciliğini saptamak olmuştur. Bu yönde yapılan incelemede görülmüştür ki; örnek alan uygulamalarında üretim tekniği olarak seçilen tünel kalıp tekniği, tasarım sürecinde sistemin karakteristiklerine uygun olarak ele alınmaktadır. Yapım hızının ve üretim maliyetinin düşüklüğünün sağlanması yönünden firma bazında avantajlı bir sistem olduğu açıktır. Ancak birinci sınıf bitirme malzemeleri ve işletim sistemlerine yönelik üst düzey yaşam kalitesi öneren pazarlama stratejisi sonucu, kullanıcıya maliyet yükseltilerek sunulmaktadır. Oysa, çoğunlukla tasarım kökenli hataların yapıya yansması, kullanıcının beklentilerinin cevaplanamaması anlamına gelmektedir. Bu hatalar, yapı kabuğu bileşeninin boyutsal düzenlemesi, kompozisyonu ve konstrüksiyonu çerçevesinde binaya/yapıya doğrudan yansımakta, birincil hatalar kadar ikincil ya da dolaylı hasar oluşumlarına da neden olmaktadır. Yapı hasarları, öncelikle bileşenin boyutsal prezisyonu yoluyla derzlerde ortaya çıkmakta, bileşen montajı, derz sızdırmazlık ve dolgu malzemesi seçimi ve derz yapım işçiliğinin etkileriyle katlanarak artabilmektedir. Bu olgu, kabuğa su ve hava sızıntıları oluşması sonucu ısı korunum mekanizmalarını işlevsizleştirmekte, konfor koşullarını olumsuz yönde etkilemektedir. Diğer taraftan, kabuğun kompozisyonu ve konstrüksiyonunda yapı bilimi göstergeleri ve bu göstergelerin standartlara yansıyan ölçütlerine uyulmaması ya da yeterli düzeyde dikkate alınmaması, yapının ısı performansının düşmesinde etken faktördür. Buna bağlı olarak kabuk kesitinde yoğunlaşmanın denetlenememesi, ikincil yapı hasarlarının ortaya çıkışına neden olmuştur.

2.5.3.	Melez kodlayıcılar	28
2.5.3.1.	MBE (Multi Band Excitation)	30
2.6.	Kodlayıcı Performansı	32
2.6.1.	Kalite ve bit hızı	32
2.6.2.	Kanal hatalarına dayanıklılık	33
2.6.3.	İletişim gecikmesi	34
2.6.4.	Birlikte çalışabilirlik ve uyumluluk	34
2.6.5.	Ses bandı veri aktarımı	35
2.6.6.	Şifreleme	35
2.6.7.	Karmaşa ve maliyet	36
2.7.	Uluslararası İletişim Birliği	36
3.	DÜŞÜK BİT HIZLI KONUŞMA KODLAMA ARAÇLARI ...	38
3.1.	Giriş	38
3.2.	Kısa Süreli Analiz	39
3.2.1.	Pencere etkisi	39
3.2.2.	Doğrusal öngörü	42
3.2.2.1.	Özilişki metodu	45
3.2.2.2.	Ortak varyans metodu	46
3.2.2.3.	Bant genişliği açma	48
3.2.2.4.	Öngörü derecesi	48
3.3.	Distorsiyon ölçümleri	49
3.3.1.	Zaman düzlemi objektif distorsiyon ölçümleri	50
3.3.1.1.	SNR	50
3.3.1.2.	Kısmi SNR	50
3.3.2.	Frekans düzlemi objektif distorsiyon ölçümleri	51
3.3.2.1.	Logaritmik İzgel Distorsiyon Ölçümü	52
3.3.2.2.	İtakura-Saito distorsiyon ölçümü	53
3.3.2.3.	Ağırlıklı Euclidean LSF mesafe Ölçümü	54

3.4.	Perde Peryodu Belirleme	55
3.4.1.	Zaman düzlemi algoritmalar	56
3.4.1.1.	Özilişki metodu	57
3.4.1.2.	AMDF metodu	59
3.4.2.	Frekans düzlemi algoritmalar	60
3.4.2.1.	Harmonik benzerlik metodu	60
3.4.2.2.	Spektrum benzerlik metodu	62
3.4.3.	Cepstral perde peryodu belirleme	62
3.4.4.	Spektrum düzleştirme	64
3.4.5.	Farklı metodların karşılaştırılması	64
3.5.	Sesli/Sessiz Sinyal Sınıflandırma	65
4.	VEKTÖR NİCEMLEME TEKNİKLERİ	67
4.1.	Giriş	67
4.2.	LPC Alternatif Gösterimleri	67
4.2.1.	Yansıma katsayıları	68
4.2.2.	LAR ve IS katsayıları	69
4.2.3.	Çizgi spektrum frekans (LSF) katsayıları	70
4.2.3.1.	LSF katsayıları hesaplanması	71
4.3.	Vektör Nicemleme	73
4.3.1.	Genelleştirilmiş Lloyd algoritması	75
4.3.2.	Yarı vektör nicemleme	76
4.3.3.	Çok aşamalı vektör nicemleme	77
4.3.3.1.	Özyineli sıralı kod çizelgesi metodu	79
4.3.3.2.	Birleşik kod çizelgesi metodu	80
5.	VEKTÖR VE MATRİS NİCEMLEMEDE GETİRİLEN YENİLİKLER	83
5.1.	Giriş	83

5.2.	Kod Çizelgesi Oluşturma ve Eğitim	83
5.2.1.	Eğitim giriş dizisi dizaynı	83
5.2.2.	Başlangıç kod çizelgesi dizaynı	85
5.3.	Artık LSF Vektör Hesaplanması	86
5.4.	İndirgenmiş Birleşik Kod Çizelgesi Metodu	89
5.4.1.	LSF ve artık sinyal metodları ile kod çizelgesi hazırlanması ...	91
5.5.	Matris Nicemleme	95
5.5.1.	Artık çok aşamalı matris nicemleme metodu	96
5.5.2.	Artık çok aşamalı matris nicemleme metodu ile kod çizelgesi hazırlanması	99
5.6.	Kod Çizelgesi Arama Karmaşası	103
6.	MATRİS NİCEMLEME TABANLI KONUŞMA KODLAYICI ...	
		105
6.1.	Giriş	105
6.2.	MELP Kodlayıcı Sistemi	105
6.2.1.	2400 bps MELP kodlayıcı analiz yapısı	107
6.2.1.1.	Perde tahmini	107
6.2.1.2.	Sesli/sessiz sinyal analizi	108
6.2.1.3.	LPC analiz	109
6.2.1.4.	LSF parametrelerinin hesaplanması	109
6.2.1.5.	Fourier genliklerin hesabı	110
6.2.1.6.	Kazanç hesabı	111
6.2.2.	Parametrelerin nicemlenmesi	112
6.2.2.1.	Doğrusal nicemleme	113
6.2.3.	Kodlayıcı sentezleme ve aradeğerleme	113
6.2.3.1.	Sesli ses sentezi	114
6.2.3.2.	Sessiz ses sentezi	114
6.2.3.3.	Aradeğerleme	115

6.3.	Matris Nicemleyici Tabanlı Konuşma Kodlayıcı (MQBC)	117
6.3.1.	Perde peryodu	119
6.3.2.	Sesli/sessiz sinyal analizi	120
6.3.2.1.	Frekans bantlarına ayırma ile sesli/sessiz sinyal analizi	121
6.3.2.2.	Doğrusal ayırıcı ile sesli/sessiz sinyal analizi	122
6.3.3.	LPC analiz	127
6.3.4.	LSF parametrelerinin hesaplanması	127
6.3.5.	Kazanç hesabı	129
6.3.6.	Parametrelerin nicemlenmesi	130
6.3.7.	Kodlayıcı sentez ve aradeğerleme	131
6.3.8.	MQBC kodlayıcı perofrmans kriterleri	132
7.	FARKLI KODLAYICILAR ve PERFORMANS ÖLÇÜMLERİ	138
7.1.	2000 bit/sn MELP Konuşma Kodlayıcı	138
7.2.	1600 bit/sn Matris Nicemleme Tabanlı Konuşma Kodlayıcı	138
7.3.	Kodlayıcı Subjektif Kalite Ölçümleri	139
7.3.1.	Ses kalitesi ölçümü	140
7.3.2.	Ses anlaşılrlığı ölçümü	141
7.3.2.1.	İngilizce DRT test tablosu	142
7.3.2.2.	Türkçe DRT test tablosu.	143
7.4.	MELP Kodlayıcılar ve MQBC Kodlayıcılar için Subjektif Testler	143
7.4.1.	İngilizce cümleler ile MOS test sonuçları.	144
7.4.2.	Türkçe cümleler ile MOS ve DRT test sonuçları	145
7.5.	Performans.	147
7.6.	Birlikte Çalışabilirlik	148
7.7.	1200 bit/sn MF_MELP Konuşma Kodlayıcı	150

7.8.	1200 bit/sn MQBC ve 1200 bit/sn MF_MELP Kodlayıcılar Arasındaki Farklar	154
7.9.	MQBC Kodlayıcıda Kazanç ve Perde Peryodu için Matris Nicemleme Metodu	155
7.10.	MSMQ Kod çizelgesi Arama ve Kodlayıcı Karmaşası	155
8.	SONUÇ VE ÖNERİLER	157
KAYNAKLAR		162
EKLER		171



1200 bit/sn HIZLI BİR DÜŞÜK BİT HIZLI KONUŞMA KODLAYICI TASARIMI

(Doktora Tezi)

Selma ÖZAYDIN

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

ÖZET

4 kbit/sn altındaki bit hızlarında konuşma kodlama, telsiz ve kablolu haberleşme, uydu haberleşmesi, ve sayısal ses depolama sistemleri gibi uygulamalarda yaygın olarak kullanılmaktadır. Böyle servisler, konuşma sinyalini aktarmadan yada depolamadan önce etkin olarak kodlamak için yüksek kaliteli düşük bit hızlı kodlayıcıların kullanımını gerektirirler. Böyle kodlayıcıların çoğu, doğrusal öngörülü kodlamaya (linear predictive coding-LPC) dayalı algoritmalar kullanırlar. Bu tezde, LPC algoritmasına dayalı 1.2 kbit/sn hızında kaliteli konuşma kodlama için dizayn edilmiş yeni bir konuşma sıkıştırma tekniği sunulmuştur. Bir 1.2 kbit/sn MBC (matrix based coder) kodlayıcı, C programlama dili kullanılarak tasarlanmıştır. Performans sonuçları, 2.4 kbit/sn MELP (mixed excitation linear prediction) kodlayıcı ve 2.4 kbps LPC kodlayıcı karşılaştırma için kullanılarak, subjektif dinleme testleri ile kontrol edilmiştir. 1.2 kbit/sn MBC kodlayıcı, yukarıdaki 2.4 kbit/sn kodlama şemaları ile karşılaştırıldığında yüksek anlaşılabilirlik ve doğruluk ile iyi kalitede ses vermektedir. Çizgi spektrum frekans (line spectral frequency-LSF) parametrelerinin kodlanmasında bit hızını azaltmak amacıyla yeni bir metot geliştirilmiştir. Bitişik pencerelerin LSF vektörleri arasındaki ilişki kullanılarak bir artık sinyal vektör bulunmuştur. Nicemleme bit hızını azaltmak için artık çok aşamalı matris nicemleme metoduna dayalı yeni bir kodlama modeli kullanılmıştır. İki bitişik pencerenin LSF parametreleri yüksek kalite etkinlik elde etmek için gruplanmış ve birlikte nicemlenmiştir. Bu metodun etkinliği ve sınırlamaları her bir dizayn edilen kod çizelgesi için izgel distorsiyon ve aykırı değer hesaplamaları yapılarak araştırılmıştır. Sonuçlar göstermiştir ki, R-MSMQ tekniği kullanılarak izgel distorsiyon 18 bit/pencere için 1 dB seviyelerine kadar düşmektedir. Doğrusal sınıflandırıcı metodu ve perde periyodu kodlama tablosu ile, seslilik bilgisi aktarmak için sadece 1 bit kullanılmıştır.

Bilim Kodu : 0609

Anahtar Kelimeler : Sayısal konuşma kodlama, düşük bit hızı, vektör nicemleme, matris nicemleme, artık sinyal

Sayfa Adedi : 200

Tez Yöneticisi : Doç.Dr. Buyurman BAYKAL

**IMPLEMENTATION OF A 1200 bps LOW BIT RATE SPEECH
CODING ALGORITHM
(Phd. Thesis)**

Selma ÖZAYDIN

**GAZİ UNIVERSITY
INSTITUTE OF SCIENCE AND TECHNOLOGY**

ABSTRACT

Speech coding at bit rates below 4 kbps is widely used in applications such as digital wireless and wireline communications, satellite communications and digital voice storage systems. Such services require the use of high quality low bit rate coders to efficiently code the speech signal before transmission or storage. The majority of such coders employ algorithms that are based on linear predictive coding (LPC). In this thesis, we present a new speech compression technique designed for good quality speech coding at 1.2 kbps based on the LPC algorithm. A 1.2 kbps matrix based coder (MBC) has been simulated using the C programming language. We have verified our performance results using subjective listening tests, in which 2.4 kbps MELP (mixed excitation linear prediction) coder and 2.4 kbps LPC coder has been used for comparison. 1.2 kbps MBC coder has produced good quality coded speech with a high degree of intelligibility and naturalness when compared to the coding schemes above operating at 2.4 kbps. We explore a method to decrease the bit rate of LSF (line spectral frequency) parameters coding. We found a LSF residual vector using the correlation between adjacent frame LSF vectors. A new coding model based on the residual multi stage matrix quantization (R-MSMQ) method is used to decrease the bit rate of quantization. LSF parameters of two adjacent frames are grouped and jointly quantized in order to obtain a high quality efficiency. The effectiveness and limitations of this method are investigated by calculating the spectral distortion and outliers for each designed codebook. Results indicate that codebook spectral distortion can be decreased as low as 1 dB at 18 bits/frame with R-MSMQ technique. With using the linear discriminant voicing classification method and pitch period coding table, only 1 bit is used to code voicing information.

Science code	: 0609
Key Words	: Digital speech coding, low bit rate, vector quantization, matrix quantization, residual signal
Page number	: 200
Adviser	: Assoc. Prof. Buyurman BAYKAL

TEŞEKKÜR

Tez çalışması boyunca çok değerli görüş ve tecrübeleriyle beni yönlendiren ve bu çalışmanın sonuçlanmasında büyük emeği olan değerli hocam Sayın Doç. Dr. Buyurman BAYKAL'a çok teşekkür ederim. Bu tez çalışmasının belirli bir döneminde beni üniversitesi'nde ağırlayarak çalışmanın ilerlemesinde büyük katkıları olan Sayın Prof. Dr. Ahmet KONDOZ'a çok teşekkür ederim. Doktora çalışmasının ders aşamasında danışmanlığımı yürüten ve tez çalışması sırasında bana istediğim çalışma fırsatlarını tanıyan Sayın Prof. Dr. Müzeyyen SARITAŞ'a çok teşekkür ederim. Tez çalışmasının ilk dönemlerinde bölümlerinin ve üniversitelerinin bütün imkanlarından yararlanmamı sağlayan Başkent Üniversitesi Elektrik-Elektronik Mühendisliği Bölümü değerli Öğretim Üyelerine çok teşekkür ederim. Türkçe akustik sesbilim konusunda değerli bilgi ve tecrübelerini esirgemeyen Sayın Prof. Dr. Mehmet DEMİREZEN'e, vektör nicemleme konusundaki bilgi ve tecrübelerinden faydalandığım Sayın Wilf LeBLANC'a, ses kodlama ile ilgili görüşlerinden faydalandığım Sayın Stephane VILETTE'e, manevi desteği ve bütün yardımlarından dolayı sevgili kardeşim Erhan ÖZAYDIN'a, Türkçe ses kayıtları için yardımlarından dolayı Sayın Nil ve Özgür YÜKSEL çiftine, tez çalışması sırasında manevi destekleri ve dinleme testleri için yardımlarından dolayı Gazi Üniversitesi Elektrik-Elektronik Mühendisliği Bölümü Araştırma Görevlisi olan sevgili arkadaşlarıma, dinleme testleri için yardımlarından dolayı Savunma Sanayii Müsteşarlığı'ndaki değerli arkadaşlarıma çok teşekkür ederim. En son olarak (ama en az değil), sevgi, sabır ve fedakârlıklarıyla bu günlere ulaşmamı sağlayan sevgili aileme çok teşekkür ederim.

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 2.1. Amerikan İngilizcesi'ndeki sesbirimlerin gösterimi	12
Çizelge 2.2. Türkçe'de fonemlerin gösterimi.	14
Çizelge 2.3. Türkçe'de ünlü ve ünsüz ses grupları	14
Çizelge 2.4. Sayısal kodlama tekniklerinin sınıflandırılması	15
Çizelge 2.5. 2400 bit/sn LPC10e kodlayıcı bit yerleşimi.	20
Çizelge 4.1. GLA kod çizelgesi hazırlama algoritması.	76
Çizelge 4.2. Lloyd (GLA) algoritması akış çizelgesi.	76
Çizelge 4.3. Özyineli sıralı kod çizelgesi hazırlama algoritması.	80
Çizelge 4.4. Birleşik dizayn ÇAVN metodu algoritması.	82
Çizelge 5.1 Oluşturulan kod çizelgeleri için kullanılan başlangıç kod çizelgesi hazırlama algoritması	86
Çizelge 5.2. Birleşik dizayn LSF kodlama tekniği ile değişik bit seviyelerinde hazırlanmış vektör kod çizelgeleri.	92
Çizelge 5.3. Birleşik dizayn artık LSF kodlama tekniği ile değişik bit seviyelerinde hazırlanmış vektör kod çizelgeleri	92
Çizelge 5.4. Özyineli sıralı LSF kodlama tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri.	94
Çizelge 5.5. Birleşik dizayn Nandkumar artık LSF vektör dizayn tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri.	94
Çizelge 5.6. Birleşik dizayn LSF vektör kodlama tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri.	100
Çizelge 5.7. Birleşik dizayn LSF MATRİS kodlama tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri.	100
Çizelge 5.8. Özyineli sıralı dizayn LSF Nandkumar MATRİS kodlama tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri	100
Çizelge 5.9. Özyineli sıralı dizayn LSF MATRİS kodlama tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri.	101

Çizelge**Sayfa**

Çizelge 5.10. Özyineli sıralı dizayn artık LSF MATRİS kodlama tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri.	101
Çizelge 6.1. MELP kodlayıcı için 7-bit indeks çizelgesi	109
Çizelge 6.2. 2400 bit/sn MELP kodlayıcıda 22.5 msn analiz penceresi için parametre yerleşimi.	113
Çizelge 6.3. MQBC kodlayıcı için 6-bit indeks çizelgesi	120
Çizelge 6.4. MELP kodlayıcı ve MQBC kodlayıcı için bazı analiz parametreleri .	127
Çizelge 6.5. 25 msn MQBC pencereleri için parametre yerleşimi	130
Çizelge 6.6. Sesli/sessiz durumu ve perde peryodunun birleşik nicemlenmesi ...	130
Çizelge 7.1. 2000 bit/sn MELP kodlayıcıda 25 msn analiz penceresi için parametre yerleşimi	138
Çizelge 7.2. 25 msn 1600 bit/sn MQBC pencereleri için parametre yerleşimi ..	139
Çizelge 7.3. MOS ve DMOS test ölçek değişimleri	141
Çizelge 7.4. The 192 Stimulus Words of the DRT_English	142
Çizelge 7.5. The 192 Stimulus Words of the DRT_Turkish	143
Çizelge 7.6. MOS test sonuçları	144
Çizelge 7.7. Değişik kodlayıcılar için İngilizce MOS test sonuçları	145
Çizelge 7.8. Değişik kodlayıcılar için Türkçe MOS test sonuçları	146
Çizelge 7.9. Değişik kodlayıcılar ile Türkçe DRT test sonuçları için on dinleyicinin işaretlediği toplam hata sayısı	147
Çizelge 7.10. Değişik kodlayıcılar ile on dinleyici için Türkçe DRT % test sonuçları	147
Çizelge 7.11. 2400 bit/sn MELP kodlayıcı ve 1200 bit/sn MQBC parametre yerleşimi	149
Çizelge 7.12. MELP kodlayıcı sesli/sessiz sinyallerin MQBC kodlayıcı sesli/sessiz sinyallerine dönüştürülmesi	149
Çizelge 7.13. MELP kodlayıcı sesli/sessiz sinyallerin MQBC kodlayıcı sesli/sessiz sinyallerine dönüştürülmesi	150
Çizelge 7.14. Perde ve sesli/sessiz kararların birlikte nicemlenmesi	152
Çizelge 7.15. LSF nicemleme için bit yerleşimi	152

Çizelge**Sayfa**

Çizelge 7.16. 1.2 kbit/sn MF_MELP kodlayıcı için 67.5 msn süper penceredeki bit yerleşimi	153
Çizelge 7.17. DRT test sonuçları	153
Çizelge 7.18. DAM test sonuçları	154
Çizelge 7.19. Farklı kodlayıcıların her pencerede harcadığı ortalama süre	156



ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 1.1. Bir ses aktarım/depolama blok şeması.	2
Şekil 1.2. Standart ses kodlayıcı örnekleri ve bit sayıları.	4
Şekil 2.1. Örnekleme ve nicemleme işlemi.	9
Şekil 2.2. Ses üretim yolu (vocal tract) organlarını içeren insan gırtlak yapısı	10
Şekil 2.3. DPCM kodlayıcı ve çözücü blok şeması.	16
Şekil 2.4. Ses üretimi için temel kaynak filtre (Fant) modeli.	19
Şekil 2.5. Bir LPC10e kodlayıcı/sentezleyici blok şeması.	21
Şekil 2.6. Kanal kodlayıcı analiz ve sentez blok şeması.	22
Şekil 2.7. İki bant analiz/sentez filtre bankası.	23
Şekil 2.8 Homomorphic kodlayıcı analiz ve sentez blok şeması.	24
Şekil 2.9. Gecikmeli perde bilgisi gösteren bir ses spektrumu.	24
Şekil 2.10. Bir AM/FM kodlayıcı/çözücü blok şeması.	26
Şekil 2.11. Bir tek kanal faz kodlayıcı analiz ve sentez şeması.	28
Şekil 2.12. Bir melez kodlayıcı/sentezleyici blok şeması.	29
Şekil 2.13. Bir MBE kodlayıcı ve çözücü blok şeması.	31
Şekil 2.14. Konuşma kodlayıcı türlerinin subjektif kalite karşılaştırılması.	33
Şekil 2.15. Bit hızına karşı işleme gecikmesi	34
Şekil 3.1. Değişik analiz pencerelerinin zaman düzlemi gösterimleri.	41
Şekil 3.2. Değişik analiz pencerelerinin frekans düzlemi gösterimleri.	41
Şekil 3.3. Sesli ve sessiz ses bölümlerinin zaman ve frekans çizimleri	43
Şekil 3.4. LPC filtre derecesine karşı LPC tahmin kazancı ve LPC zarf çizimleri .	49
Şekil 3.5. Orijinal sinyal, LPC ve artık perde zaman düzlemi çizimleri	56
Şekil 3.6. Orijinal sinyal ve normalize edilmiş sinyal için sesin özilişkisi	58
Şekil 3.7. Orijinal sesli ses sinyali için özilişki ve AMDF fonksiyonu	60
Şekil 3.8. Harmonik pik benzerlik metodu	61

Şekil	Sayfa
Şekil 3.9. Spektrum benzerlik perde belirleme metodunda kullanılan orijinal ve sentetik ses spektrumu.	62
Şekil 3.10. Merkez kırpıcıların giriş-çıkış karakteristiği.	64
Şekil 4.1. İki boyutlu uzay örneği ($N = 2$)	74
Şekil 4.2. Vektör nicemleme modeli.	75
Şekil 4.3. Çok aşamalı vektör nicemleyici (m-ÇAVN)	78
Şekil 5.1. Kod çizelgesi dizayn işlemi blok gösterimi.	84
Şekil 5.2. Eğitim giriş dizisi için kullanılan ses dosyaları yerleşimi.	84
Şekil 5.3. Artık LSF kodlama şeması blok şeması.	87
Şekil 5.4. Artık sinyal metodunda, optimum ilişki katsayısı (α) için SD grafiği ...	88
Şekil 5.5. Artık sinyal metodunda, optimum ilişki katsayısı (α) için % aykırı değer grafiği	88
Şekil 5.6. Birleşik dizayn LSF kodlama ve artık LSF kodlama teknikleri ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri için SD ve %aykırı değer grafikleri	93
Şekil 5.7. Özyineli sıralı LSF kodlama ve Nandkumar artık LSF kodlama teknikleri ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri için SD ve %aykırı değer grafikleri	94
Şekil 5.8. Artık ÇAMAN kodlayıcı ve kod çözücü şeması	93
Şekil 5.9. ÇAVN ve Artık ÇAMAN kod çizelgeleri bit hızları ve izgel distorsiyon (SD) arasındaki bağlantı grafiği	97
Şekil 5.10. ÇAVN ve Artık ÇAMAN kod çizelgeleri bit hızları ve % aykırı değer arasındaki bağlantı grafiği	102
Şekil 6.1. MELP kodlayıcı analiz şeması.	107
Şekil 6.2. MELP sentezleyici blok şeması.	114
Şekil 6.3. MQBC kodlayıcı analiz şeması.	119
Şekil 6.4. Ses spektrumunun sesli/sessiz analiz sonuçları.	126
Şekil 6.5. MQBC kodlayıcıda kullanılan kazanç interpolasyon algoritması	132

Şekil**Sayfa**

Şekil 6.6. Nicemlenmiş ve nicemlenmemiş LSF vektörlerinin 25 bit MELP kod çizelgesi, 21-bit artık LSF kod çizelgesi ve 18 bit matris kod çizelgesi için ilk 4 LSF vektörlerinin grafikleri.	134
Şekil 6.7. Nicemlenmiş LSF vektörlerinin 25 bit MELP kod çizelgesi, 21-bit artık LSF kod çizelgesi, 18 bit matris kod çizelgesi için izgel distorsiyon grafikleri.	135
Şekil 6.8. Eğitim giriş dizisi içinde yer alan bir ses verisi için 25 bit MELP kod çizelgesi, 21-bit artık kod çizelgesi ve 18 bit artık matris kod çizelgesi için SD grafikleri.	136
Şekil 6.9. Test giriş dizisi içinde bir ses verisi için 25 bit MELP kod çizelgesi, 21 bit artık kod çizelgesi ve 18 bit artık Matris kod çizelgesi için SD grafiği.	137

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Kısaltmalar	Açıklama
AC	Autocorrelation
ADM	Adaptive Delta Modulation
ADPCM	Adaptive Differential Pulse Code Modulation
AMDF	Average Magnitude Difference Method
AM-FM	Amplitude Modulation – Frequency Modulation
ASE	Adaptive Spectral Enhancement
ATC	Adaptive Transform Coding
BER	Bit Error Rate
CCITT	The International Telephone and Telegraph Consultative Committee
CELP	Codebook Excited Linear Prediction
CVSDM	Continuous Variable Slope Delta Modulation
ÇAMAN	Çok Aşamalı Matris Nicemleme
ÇAVN	Çok Aşamalı Vektör Nicemleme
DAM	Diagnostic Acceptability Measure
DM	Delta Modulation
DMOS	Degraded Mean Opinion Score
DPCM	Differential Pulse code modulation
DRT	Diagnostic Rhyme Test
DSP	Digital Spectral Filter
DTMF	Dual Tone Multi Frequency
ESA	Enhancement Spectral Filter
FEC	Forward Error Correction
GLA	Generalized Lloyd Algorithm
GSM	Group Speciale Mobile
IFFT	Inverse Fast Fourier Transform

Kısaltmalar	Açıklama
IS	Inverse Sinus
ITU	International Telecommunication Union
IMBE	Improved Multi Band Excitation
LPF	Low Pulse Filter
LAR	Log Area Ratio
LP	Linear Prediction
LPC	Linear Prediction Coding
LSF	Line Spectral Frequency
LTP	Long Term Prediction
MDA	Multi Band Demodulation
MELP	Mixed Excitation Linear Prediction
MBE	Multi Band Excitation
MQBC	Matris Nicemleme Tabanlı Konuşma Kodlayıcı
MOS	Mean Opinion Measure
MSMQ	Multi Stage Matrix Quantization
MSVQ	Multi Stage Vector Qantization
PCM	Pulse Code Modulation
PDF	Probability
PSTN	Public Switched Telephone Network
RELp	Residual Excited Linear Prediction
RMS	Root Mean Square
RPE	Regular Pulse Excitation
SBC	Subband Coding
SD	Spectral Distortion
SNR	Signal to Noise Ratio
STFT	Short Time Fourier Transform
VSELP	Vector sum Excited Linear Prediction
VQ	Vector Quantization
V/U	Voiced/Unvoiced
VLSI	Very Large Scale Integration

Kısaltmalar	Açıklama
WMSE	Weighted Mean Square Error
YVN	Yarık Vektör Nicemleme



1. GİRİŞ

Sayısal konuşma kodlama, sayısal bir kanalda (bir hücreli kanal, uydu kanalı ya da internet) sayısal ses sinyalinin aktarımı için aygıtlar ya da algoritmalar için kullanılır. Hücreli ve uydu kanalı ile haberleşme sistemlerinde son yirmi yıldaki hızlı gelişmeler, sayısal haberleşme sistemlerinde konuşma kodlamanın önemini arttırmıştır. Diğer bütün kodlama sistemleri gibi konuşma kodlamada da kodlayıcı ve çözücü olmak üzere iki temel bölüm vardır. *Düşük bit hızlı konuşma kodlama* (low bit rate speech coding) kaliteyi koruyarak ses sinyalini tanımlayan bitlerin sayısını azaltmayı amaçlar. Burada sayısal ses sinyali genellikle 8 kHz'de örneklenir ve 4 kHz'de bant sınırlanır.

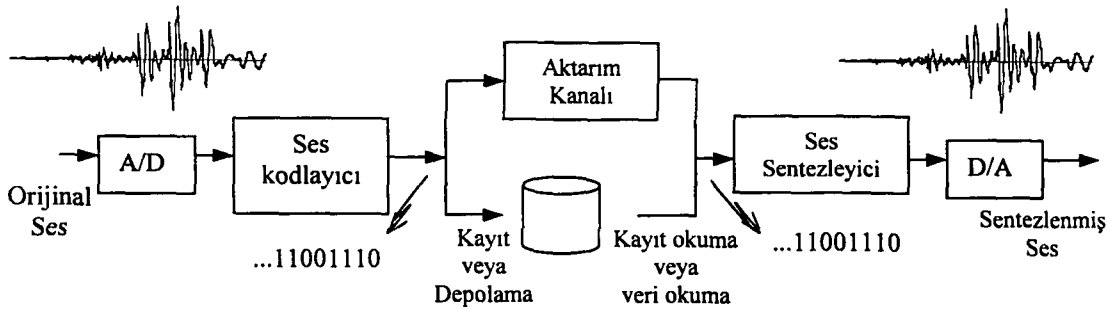
Konuşma kodlama, örnekleme ve genlik nicemlemeyi içerir. Nicemleme direkt ya da parametrik olabilir. Direkt nicemleme, ses örneklerinin ikili gösterimidir. En basit direkt nicemleme metodu, *vurum kod modülasyon* (Pulse Code Modulation - PCM) basitçe örneklenmiş genliklerin nicemlenmesidir. PCM ile nicemlenmiş sesi kodlamada 64 kbit/sn kullanılır ve buradaki ses "sıkıştırılmamış" olarak düşünülerek diğer kodlayıcılar ile karşılaştırmada kullanılır. Genellikle 16-8 kbit/sn arası konuşma kodlama için *orta-hızlı* (medium rate), 8-2.4 kbit/sn arası kodlama için *düşük-hızlı* (low rate), 2.4 kbit/sn altı kodlama için *çok düşük hızlı* (very low rate) terimi kullanılır.

Parametrik nicemleme, modellenmiş sesin ve/veya izgel parametrelerin ikili gösterimidir. Parametrik gösterimler ses-özellikli ya da ses-özellikli olmayan yapıda olabilirler. Ses özellikli olmayan yapıda kodlayıcılar ya da *dalga şekli* (waveform) kodlayıcılar, zaman düzlemindeki dalga şeklinin aslına uygun olarak tekrar yapılandırılması için çalışırlar ve genellikle orta hızlarda çalışırlar. Ses özellikli kodlayıcılar ya da *konuşma kodlayıcılar* (Voice

CODERs- vocoders), ses modellerine bağıdırlar ve dalga şeklini tekrar benzetmeye uğraşmadan yüzde olarak kabul edilebilir kalitede ses üretmeye çalışırlar. Konuşma kodlayıcılar, çok düşük bit hızlarında çalışabilirler fakat yapay ses üretmeye meyillidirler. Bu konuşma kodlamada kabul edilebilir olmasına rağmen, daha iyi performans almak için konuşma kodlayıcılar ve dalga şekli kodlayıcıların özelliklerini kullanan melez kodlayıcılar mevcuttur.

Orta-hızlar altında konuşma kodlama, analiz-sentez işlemiyle başarılıdır. Analizde ses, etkin olarak kodlanmış parametre kümesi ile gösterilir ve kodlanır. Sentez aşamasında, bu parametreler çözülür ve ses sinyalini tekrar oluşturmada kullanılırlar. Analiz açık döngü ya da kapalı döngü olabilir. Kapalı döngü analizde parametreler, orijinal ve tekrar oluşturulmuş ses arasındaki yüzde farkı minimum yapacak şekilde seçilirler. Bu yüzden kapalı döngü analiz, sentez kısmını da içerir ve analiz-sentez olarak adlandırılır.

Bir konuşma kodlama algoritması, bit hızı, tekrar oluşturulan sesin kalitesi, algoritmanın karmaşası, oluşan gecikme, ve algoritmanın kanal hataları ve akustik değişimlere dayanıklılığı ile ölçülür. Genellikle daha düşük bit hızları, daha büyük algoritma karmaşası ile başarılıdır. Şekil 1.1 genel bir konuşma kodlama blok şemasını göstermektedir.



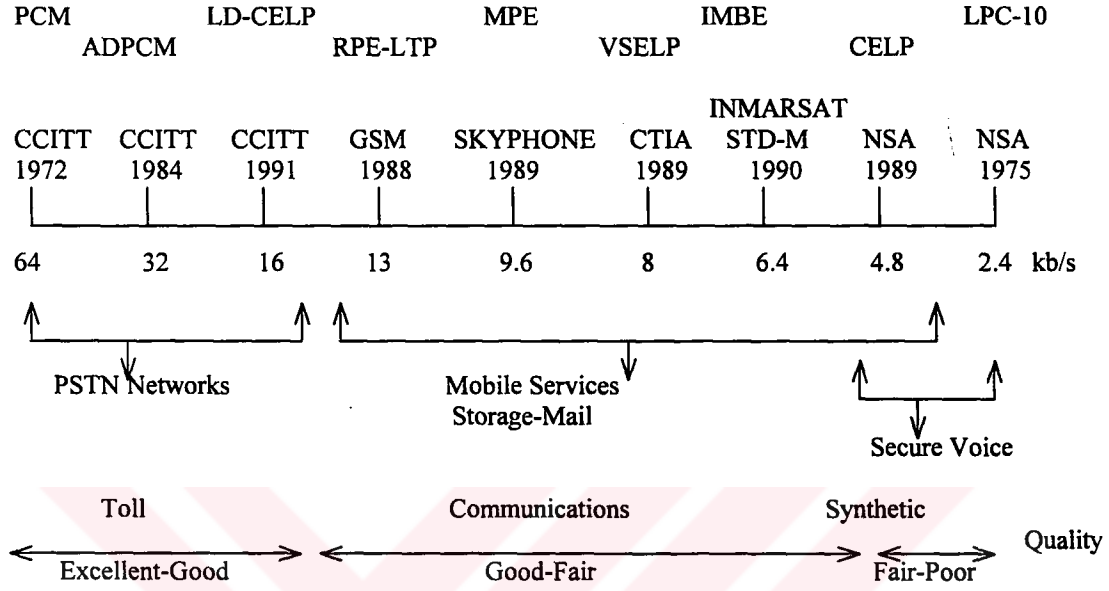
Şekil 1.1. Bir ses aktarım/depolama blok şeması

Konuşma Kodlama Tarihçesi

Konuşma kodlama araştırmaları, Bell telefon laboratuvarlarında başlamış ve 1939'da bir kanal kodlayıcı geliştirilmiştir [1]. PCM'in 1938'de [2] bulunuşu, sesin sayısal olarak işlenmesinde önemli bir adım olmuştur. Sayısal devreler ve yarı iletken teknolojidaki ilerlemeler, ses sinyalinin sayısal işlenmesini özellikle 1960'dan sonra hızla artırmıştır. Bir kaynak-sistem modeli, Fant tarafından 1960'da geliştirilmiştir [3]. Bu modelde, bir rasgele uyarma yada periyodik uyarma sinyali ile uyarılan ve *ses üretim yolunu* (vocal tract) temsil eden bir sistem mevcuttur. Flanagan 1966'da, ses sinyali için *kısa süreli Fourier dönüşümü* (Short Time Fourier Transform – STFT) ile analiz ve sentez metodunu önermiştir [4]. Bundan başka yetmişli yıllarda, *dönüşüm kodlama* (transform coding), *yan bant kodlama* (subband coding) ve *doğrusal öngörülü kodlama* (Linear Prediction Coding – LPC) metotları ile konuşma kodlamanın askeri haberleşmede uygulamaları başlamıştır. LPC algoritmasına dayalı bir ulusal standart FS-1015 LPC-10, 1982'de geliştirilmiştir [5,6].

Seksenli ve doksanlı yıllardaki araştırmalar, haberleşme ve mobil telefon uygulamaları için yüksek kalitede ses üreten güçlü düşük bit hızlı kodlayıcılar üzerinde odaklanmıştır. Bu çalışmaların pek çoğu, hücresel ve askeri uygulamalardaki dar bantlı ve güvenli haberleşme ihtiyacını karşılamak için yapılmıştır. Seksenli yıllarda geliştirilen bazı kodlayıcılar; Griffin ve Lim tarafından önerilen sesin sinüzoidal analiz ve sentezi [7], Buzo tarafından önerilen ve LPC kodlamada çok faydalı olan *vektör nicemleme* (Vector Quantization – VQ) metodu [8], Atal ve Schroeder tarafından önerilen *olasılıksal vektör uyarımlı doğrusal öngörü* (Codebook Excited Linear Prediction-CELP) algoritması konuşma kodlayıcılar için çok önemli adımlar olmuştur [9,10]. CELP metodundaki olasılıksal (stochastic) uyarım, yüzde ağırlıklı bir kapalı döngü (analiz/sentez) optimizasyon ile belirlenir. CELP

kodlayıcılar, konuşma kodlayıcılar ve dalga şekli kodlayıcıların ortak özelliklerinin birleşiminden oluştukları için, aynı zamanda *melez* (hybrid) kodlayıcılardır.



Şekil 1.2. Standart konuşma kodlayıcı örnekleri ve bit sayıları

U.S. standardı olan FS-1016 4.8 kbit/sn CELP kodlayıcı [13,15-24], güvenli telefon servis haberleşmesinde kullanılır. FS-1016'nın karmaşası orta ölçüdedir ve kalitesi orta düzeydedir. Avrupa'da düzenli vurum uyarıtımı (regular pulse excitation-RPE) algoritmasını kullanan GSM 13 kbit/sn RPE-LTP isimli bir standart geliştirilmiştir. [11]. Bir melez kodlayıcı olan 8 kbit/sn VSELP (Vector Sum Excited Linear Prediction) [12], 1990'da Kuzey Amerika sayısal hücresel standardı olarak seçilmiştir. IS-54'ün performansı RPE-LTP'den biraz daha iyidir. 6.7 kbit/sn VSELP [13], Japonya hücresel standardı olarak seçilmiştir. Inmarsat IMBE (Improved Multi Band Excitation) kodlayıcı [14]. INMARSAT denizcilik bölgesi haberleşmesi için geliştirilmiştir. IMBE kodlayıcının karmaşası VSELP kodlayıcının yarısı kadar, performansı ise radyo iletişimi için CELP kodlayıcılardan daha iyidir. Standart konuşma kodlayıcı örnekleri ve bit sayıları Şekil 1.2'de görülmektedir.

Tez yapısı

Bu tezin amacı, doğrusal öngörülü kodlama metoduna dayalı kodlayıcıların düşük bit hızlarında yüksek kaliteli kodlama algoritmaları için yeni yaklaşımlar sunmaktır. Tezin ilk bölümlerinde, ses sinyalinin özellikleri ve düşük bit hızlı kodlama için temel bazı konulara değinilmiştir. Düşük bit hızlı kodlamada önemli kodlama teknikleri ve bu algoritmalar için gerekli temel gereçler tanımlanmıştır. Bunlardan bazıları doğrusal öngörülü kodlayıcılar içinde kullanılmaktadır. İkinci bölümde, doğrusal öngörülü kodlama metodu ve bu metodun kuralları üzerinde durulmaktadır. Doğrusal öngörülü kodlama metoduna dayalı kodlayıcılar ile kodlama tekniğini çok düşük bit hızlarında gerçekleştirmek mümkündür. Doğrusal öngörülü kodlama tekniğine dayalı bir Amerikan federal standardı olan 2400 bit/sn MELP (Mixed Excitation Linear Prediction) kodlayıcı üzerinde durulmuştur. Vektör nicemleme metodundan hareketle, yeni bir matris nicemleme metodu geliştirilmiş ve hem vektör nicemleyiciler hem de matris nicemleme metotları ile kod çizelgeleri oluşturulmuştur. Perde peryodu ve sesli/sessiz sinyal ayırma metotları için kodlama teknikleri ve yeni algoritmalar üzerinde durulmuştur. 1200 bit/sn ve 1600 bit/sn hızlarında çalışabilen yeni bir doğrusal öngörülü kodlayıcı MQBC (Matrix Quantization Based Speech Coder) kodlayıcı oluşturulmuş ve performansı standart MELP kodlayıcı ve 2400 bit/sn LPC-10 kodlayıcı ile karşılaştırılmıştır. Türkçe için DRT tablosu oluşturulmuştur.

- Bölüm 1, konuşma kodlama hakkında bazı temel bilgiler ve konuşma kodlama tarihçesi hakkında incelemeleri içermektedir. Bölüm 2 ise, akustik sesbilim, ses sinyalinin sayısal olarak kodlanabilmesi için kullanılan metotlar ve kodlayıcı dizayn algoritmalarında göz önünde bulundurulması gereken performans kriterleri üzerinde durmaktadır. Bazı konuşma

kodlayıcılar hakkında bilgiler sunulmuş, diğer kodlayıcılara göre performansları incelenmiştir.

- Bölüm 3’de, düşük bit hızlı konuşma kodlama araçları anlatılmıştır. Düşük bit hızlı konuşma kodlama teknikleri için temel araçlar arasında olan perde belirleme, sesli/sessiz sinyal ayrımı ve doğrusal öngörü filtre katsayıları üzerinde durulmuştur. Doğrusal öngörü (LP) metoduna dayalı düşük bit hızlı konuşma kodlayıcılar için temel bir yaklaşım sunulmakta ve metotların birbirine üstünlükleri ve zayıf yönleri vurgulanmaya çalışılmaktadır. Doğrusal öngörü analizi için değişen bazı parametrelerin (pencere uzunluğu, filtre derecesi, analiz penceresi uzunluğu vb.) performansa etkisi incelenmiştir. Bazı objektif distorsiyon ölçüm metotları üzerinde durulmuştur.
- Bölüm 4, kodlayıcı tablosunda büyük yer tutan filtre katsayılarının etkin kodlanabilmesi üzerinde durulmuş vektör nicemleme teknikleri üzerinde durulmaktadır. LPC filtre katsayıları gösterimi için metotlar incelenmiştir. Vektör nicemleme metotları arasında olan çok aşamalı vektör nicemleme ve yarık vektör nicemleme metotları üzerinde durulmaktadır. Doğrusal öngörü katsayılarının alternatif gösterimleri olan LSF (Line Spectral Frequencies), LAR (Log Area Ratio) veya yansıma katsayıları hesaplama metotları incelenmiştir.
- Bölüm 5’de, vektör nicemleme için optimizasyon yöntemleri ve yeni geliştirilen matris nicemleme tekniği incelenmiştir. Bu metotlar ile kod çizelgeleri hazırlanmış, distorsiyon ve aykırı değer ölçümleri grafikler ve çizelgeler halinde sunulmuştur. (Bu çizelgelerden birisi olan ve artık ÇAMAN (Çok Aşamalı MATris Nicemleme) metodu ile hazırlanan 18 bit/sn kod çizelgesi tablosu MQBC kodlayıcılarda kullanılmıştır.) Bitişik

pencerelerin korelasyonu mantığından hareketle yeni bir artık sinyal hesaplama metodu sunulmuştur. Artık sinyal metodu vektör nicemleme ve matris nicemleme metotları üzerinde denenmiştir. Yeni artık ÇAMAN metodu ve bilinen ÇAVN vektör nicemleme metotları için kod çizelgeleri oluşturulmuş, izgel distorsiyon ve aykırı değer hesaplamaları yapılmıştır.

- Bölüm 6, standart MELP kodlayıcı ve yeni geliştirilen MQBC kodlayıcı prensipleri üzerinde durmaktadır. Standart MELP kodlayıcıyı diğer doğrusal öngörülü kodlayıcılardan ayıran beş temel özelliği içinde, Fourier genlikler dışında kalan bütün özellikler MQBC kodlayıcıda da tutulmaya çalışılmıştır. Karma uyartım, uyarlamalı izgel zenginleştirme filtresi, darbe yayılım gibi özellikler MQBC kodlayıcı içinde de korunmaktadır. Perde peryodu hesaplama metodu için MELP kodlayıcı için kullanılan metot kullanılmış ve yeni bir perde peryodu kod tablosu oluşturularak 6 bit ile nicemlenmiştir. Perde peryodu ve sesli/sessiz sinyal ayrımı MQBC kodlayıcıda birleşik olarak kodlanmıştır. Sentez bölümünde analizden aktarılan bu parametrelerle, sentezleyici için kullanılacak perde peryodu ve sesli/sessiz ayrımı belirlenmektedir.
- Bölüm 7, MELP [86], MQBC [104-108] ve LPC-10 [5] kodlayıcılar için performans ölçümleri üzerinde durmaktadır. Performans ölçümleri subjektif kalite ölçüm metotları ile yapılmıştır. Subjektif performans ölçümleri olarak MOS (Mean Opinion Score) testi ve DRT (Diagnositc Rhyme Test) testleri uygulanmıştır. MOS testi için İngilizce ve Türkçe ses kayıtları kullanılmıştır. DRT testi için ise Türkçe ses kayıtları hazırlanmış ve testler bu kayıtlar ile yapılmıştır. MELP ve MQBC kodlayıcılar için nicemleme hataları MSE ölçümü ile hesaplanmıştır. Kazanç, perde peryodu ve filtre katsayıları için distorsiyon hesaplamaları yapılmıştır.

2. SAYISAL İLETİŞİM ve KODLAMA

2.1. Giriş

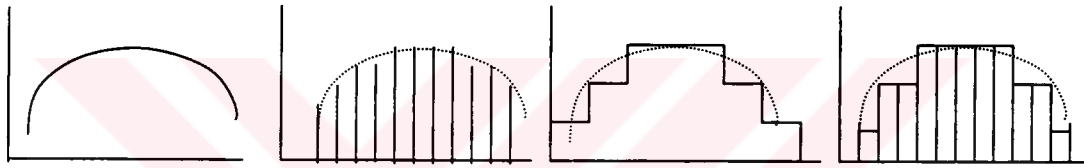
Ses sinyalinin sayısal kodlanması, son otuz yıldır geniş bir araştırma konusu olmuş ve sonuç olarak birçok stratejiler ve yaklaşımlar geliştirilmiştir. Bu yaklaşımlardan bazıları standart olarak kabul edilmiş ve kullanılmaktadır. Bu bölümde ses sinyalinin sayısal olarak kodlanabilmesi için kullanılan metotlar incelenmiştir. Ayrıca kodlayıcı dizayn algoritmalarında göz önünde bulundurulması gereken performans kriterlerine değinilmiştir.

2.2. Örnekleme ve Nicemleme

Bir ses sinyali zamanda sürekli ve sayısal devre kartlarında işlenmeden önce zamanda kesikli hale getirilmelidir. *Örnekleme* (sampling) işlemi, analog sinyalden periyodik aralıklarla örnekler alınarak zamanda sürekli olan sinyal kesikli hale çevrilir. Her saniyede alınan örneklerin sayısı (örnekleme hızı-sampling rate) artarsa, örneklenmiş sinyal sürekli sinyale çok benzer ve örnekleme hızı iyi ayarlanırsa, örneklenmiş sinyal sürekli sinyaldeki bütün bilgiyi içerir. Nyquist örnekleme teorisi farzeder ki, eğer bir sinyal içerdiği en yüksek frekans parçasının iki katına eşit veya daha büyük bir örnekleme hızı ile örneklenirse, en iyi şekilde tekrar oluşturulabilir [25].

Örneklenmiş bir sinyal hala zamanda sürekli genliklidir. *Nicemleme* (quantization), kesikli zamanda sürekli genlikli sinyali kesikli zamanda kesikli genlikli hale çevirir. Herbir sinyal örneğinin değeri, sınırlı bir küme içinden seçilir. Nicemlenmiş çıkış ve nicemlenmemiş giriş arasındaki fark, nicemleme hatası yada gürültü olarak adlandırılır ve bu hata değerinin küçük olmasına çalışılır. Bu nedenle, değişik nicemleme yöntemleri kullanılabilir.

Doğrusal nicemleme (linear quantization), logaritmik nicemleme, doğrusal olmayan nicemleme ve vektör nicemleme bazı nicemleme yöntemleridir. Ses sinyalleri 60 dB civarı bir dinamik değişime sahiptirler. Ayrıca insan kulağının, kazançtaki doğrusal artışlara logaritmik tepki vermesi nedeniyle doğrusal nicemleme pratik bir yöntem değildir. Logaritmik nicemlenmiş sinyallerde önemli derecede kalite artışı olmaktadır. Bütün nicemleme yöntemleri bölgesel olarak optimumdur ve en düşük bit hızı ile mümkün olan en yüksek kaliteyi sağlamaya çalışırlar. Fakat pratikte, nicemleyicinin kalitesi arttıkça karmaşa ve maliyet artmaktadır. Tipik bir örnekleme ve nicemleme işlemi Şekil 2.1’de görülmektedir. [17,25,26]



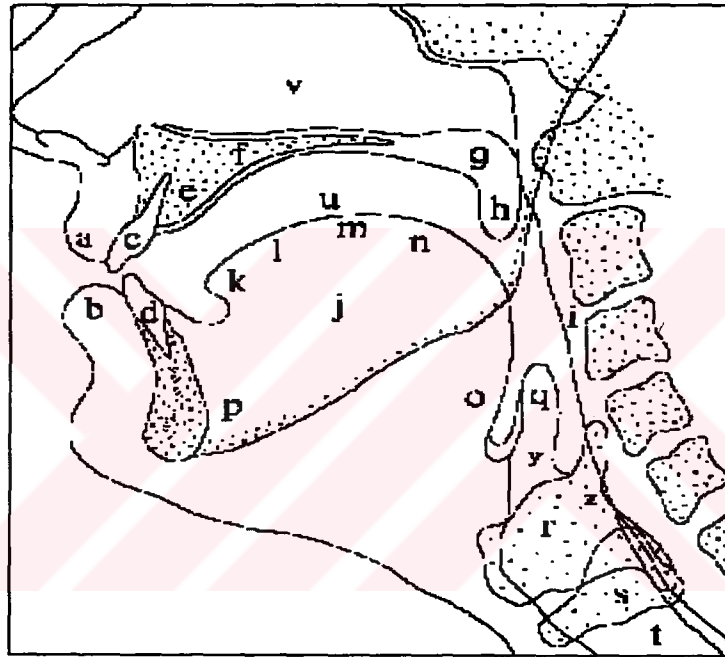
a) Sürekli Sinyal b) Örnekleme c) Nicemleme d) örnekleme & nicemleme
Şekil 2.1. Örnekleme ve nicemleme işlemi

2.3. Ses Üretimi ve Sınıflandırılması

Ses üretim işlemi, üç alt işlemi içerir. Bunlar, kaynak üretimi, boğumlanma (articulation) ve radyasyondur. İnsan ses üretim yolu (vocal tract); ciğerler, nefes borusu (glottis), gırtlak (larynx), yutak (pharynx), ve nazal ve oral oyuklar gibi pek çok parçayı içerir. Şekil 2.2, bu yapıyı göstermektedir¹. Yutak, dil kökü ve üst boğaz duvarları arasındaki oyuktur. Gırtlak, ses tellerini (vocal cords) tutan ve idare eden yapıdır. Gırtlak ve nazal oyuk arasındaki bölüm ses üretim yolu olarak alınır. Ses üretim yolu filtreleme ile bazı frekanslar geçirilir ve bazıları tutulur. Konuşma için gerekli havayı akciğerler sağlar. Akciğerlerden gelen hava, nefes borusundan geçerek gırtlakta bulunan

¹ Ref: <http://www.ims.uni-stuttgart.de/phonetik/EGG/pagem5.htm>

ses tellerine çarpar, ses telleri titrer, sesler tonlanır ve geniz, burun, ağız boşluğu, damak, dişler ve diş etlerinde biçimlenerek konuşmayı sağlayan sesler oluşur. *Yutak yardımcısı* (epiglottis), çiğneme sırasında yiyeceklerin ciğerler yerine mideye gitmesinde yutağa yardımcı olur. Çok az dil yutak yardımcısını ses üretmede kullanır. İngilizce ve Türkçe, bu dillerden biri değildir.



Şekil 2.2. Ses üretim yolu (vocal tract) organlarını içeren insan gırtlak yapısı

a. Üst dudak	i. Yutak (Pharynx)	q. Yutak yardımcısı (epiglottis)
b. Alt dudak	j. Dil yapısı	r. Troid kıkırdağı
c. Üst diş	k. Dil ucu	s. Siroid kıkırdağı
d. Alt diş	l. Dil üstü	t. Nefes borusu (Trachea)
e. Diş yurması	m. Dil ömü	u. Ağız boşluğu (Oral cavity)
f. Sert damak	n. Dil arkası	v. Nazal oyuk
g. Yumuşak damak	o. Dil kökü	y. Ses telleri (vocal cords)
h. Küçük dil	p. Alt çene	z. Gırtlak (larynx)

Nefes borusu, yada sağ ve sol ses telleri arasındaki delik, nefes alma sırasında genellikle açıkken konuşma sırasında daralır. Nefes borusu boyunca olan hava akışı, ses üretim yolu ve hava akışı arasındaki etkileşimde periyodik olarak açılıp kapanır. Ses üretim yolu güçlü bir şekilde zorlanırsa ve

ciğerlerden yükselen hava basıncı yüksek olursa, *ses üretim yolu titreşim periyodu* (vocal tract vibration period) kısadır ve sesin perdesi yükselir. Karşıt olarak, düşük hava basıncı düşük perde üretir. Bu ses üretim yolu titreşmesi, *perde periyodu* (pitch period) adını alır. Konuşma sinyalini işlemede sesler uyartım modlarına göre, *sesli* (voiced), *sessiz* (unvoiced- fricative) ve *plosive* diye üç ana sınıfta toplanabilir. Sesli sesler (örn. a,e, v.b.), ses üretim yolu titreşmesiyle oluşan *yarı periyodik* (quasi-periodic) uyartım sinyali ile üretilirler. Sessiz sesler (örn. b,f, v.b), ses üretim yolunda bir zorlanmayla gürültü türü uyartım ile üretilirler. Plosive sesler (örn. p,t, v.b), ses üretim yolunda kapanmayla bir basınç zorlanması ve basıncın ani bırakılışı ile oluşan akustik enerji patlamasıyla üretilirler.

2.4. Akustik Sesbilim

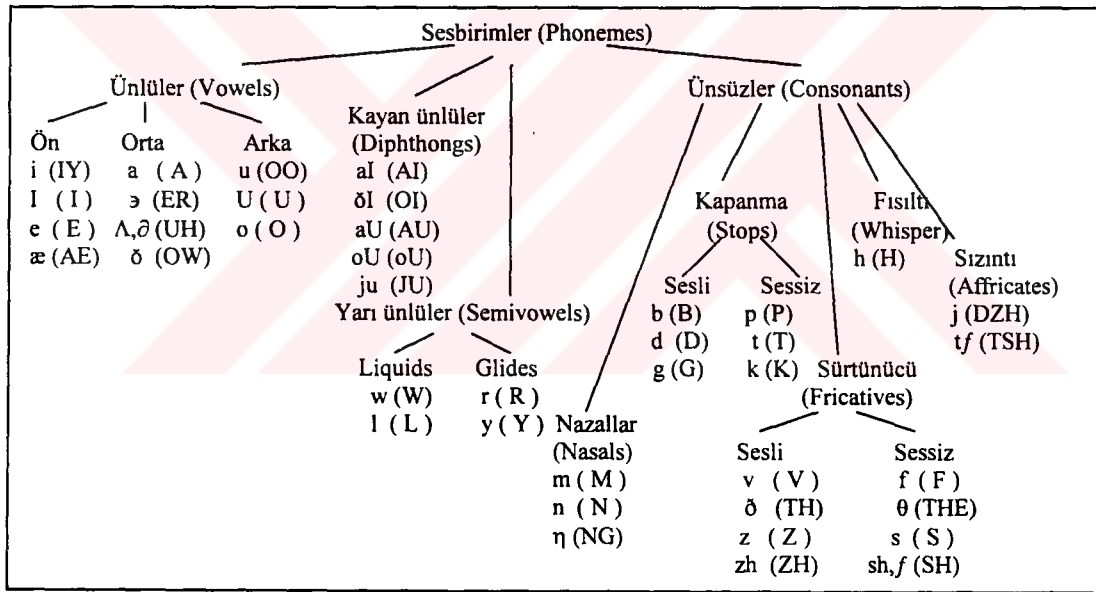
Dildeki seslerin özelliklerini inceleyen bilime *sesbilim* (phonetic) denir. Sesbilim seslerin oluşumunu, seslerin çıkış yerlerini ve seslerle ilgili olan tüm kuralları inceler. İnsan kulağı 20 Hz ile 20000 Hz arasındaki ses dalgalarını duyabilir. Akustik sesbilim doğadaki sesleri, ses dalgalarını, ton, ses şiddeti, tını gibi konuları inceler. Pek çok dilde farklı sesler *sesbirim* (phoneme) denilen kümede ifade edilirler. Bir sesbirim, sesin en küçük birimi olarak belirlenir.

2.4.1. İngilizce akustik sesbilim

Amerikan İngilizcesi 42 sesbirim içerir (Çizelge 2.1) ve *ünlü* (vowel), *kayan ünlü* (diphthong), *yarı ünlü* (semivowel) ve *ünsüz* (consonant) (sürtünücü-fricative, kapanma-stop, nazal-nasal, sızıntı-affricate ve sızıcı_sürtünücü-spirant) diye dört yaygın sınıfta toplanabilir. Ünlü sesler ses üretim yolunun yarı-periyodik darbelerle uyartımıyla üretilir. Kayan ünlü sesler ses üretim

yolunun bir ünlü için başlayıp başka bir ünlüye değişimini gösterir. “pay” kelimesindeki /eI/ ve “pow” daki /aU/ örnekleri arasındadır. Yarı ünlü sesler ünlü sesler gibidirler fakat ses üretim yolu biraz zorlanır. Genellikle bir ünlüden önce yada sonra gelen geçiş sesidirler. (/R/,/Y/,/W/,/L/). Sürtünücü sesler üretim yolunda bir zorlanma sonucu gürültü türü uyarımla üretilirler. Kapanma sesleri ses üretim yolunun tamamen kapanması sonucu basınçla oluşurlar. Bu basıncın ani bırakılışı gürültü benzeri uyarım sinyaline neden olur. Nazal seslerde ağız çukuru kapanarak nazal çukur kanalı açılır (/M/ , /N/). Sızıntı sesler kapanma-sürtünme birleşimi gibidirler. Fısıltı sesi /H/ ses teli titreşimi olmadan gırtlak türbelansı ile üretilir. [25,26]

Çizelge 2.1. Amerikan İngilizcesi’ndeki sesbirimlerin gösterimi



2.4.2. Türkçe akustik sesbilim

Türkçe’de sesler ünlüler ve ünsüzler olmak üzere iki temel bölümde incelenebilir. Ağız boşluğunda ve ses üretim yolunda hiçbir engelle çarpmadan çıkan seslere ünlüler denir. Türkçe’de /a,e,ı,i,o,ö,u,ü/ olmak üzere 8 ünlü vardır. Ağız boşluğunda ve ses üretim yolunda bazı engellere çarparak çıkan

seslere ünsüzler denir. Türkçe’de 21 ünsüz vardır. Bunlar /b,c,ç,d,f,g,ğ,h,j,k,l,m,n,p,r,s,ş,t,v,y,z/ olarak sıralanabilir.

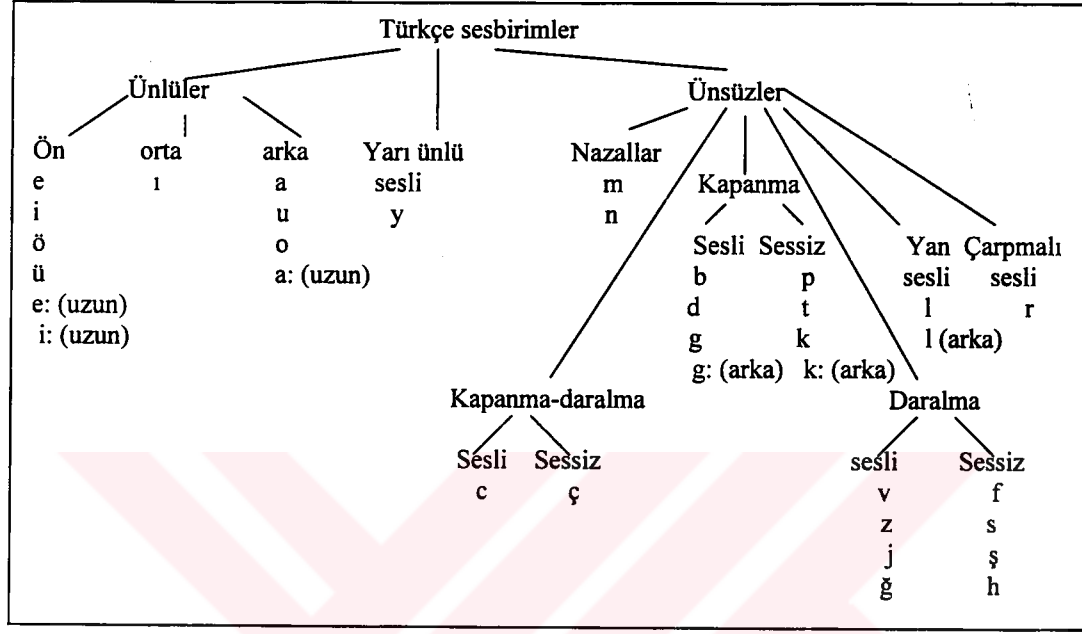
Ünlüler dilin önde, ortada yada arkada tümsekleşmesine göre üç sınıfa ayrılabilir. Çıkış yerlerine göre ünlülerde dilin geriye çekilmiş durumunda çıkarılan ünlülere kalın (arka) ünlüler denir. Ağız boşluğunda dilin ileri doğru sürülmüş durumunda çıkarılan ünlülere ince (ön) ünlüler denir. Ayrıca ünlüler dudakların aldığı şekle göre düz ve yuvarlak olarak gruplanabilir. Çenenin ve ağız boşluğunun durumuna göre çıkarılan ünlü grupları ise geniş ve dar ünlülerdir. Ayrıca uzun /a/ (örn. kânun), uzun /e/ (örn. mēzun), arka /g/ (örn. gaga), arka /k/ (örn. kadın), arka /l/ (örn. sal) gibi sesler vardır. Arka ünlüler daha kalın çıkmaktadır. Kayan ünlüler, bir ünlünün işlevini yerine getiren çift ünlüye denir. Türkçe’de kayan ünlü grubuna pek rastlanmamakla birlikte *ayran*, *avlan* gibi seslerde *ay,av* gibi birleşmelere kayan ünlü denilebileceği ileri sürülmektedir [101]. Yarı ünlüler ünlü ve ünsüz sınırındaki seslerdir. Türkçe’de yalnız /y/ sesi genellikle yarı ünlü olarak gösterilir.

Ünsüz sesler, ses üretim yolunun açılıp kapanma durumlarına göre sürekli ve süreksiz diye iki sınıfta incelenebilir. Süreksiz ünsüzlerde akciğerlerden gelen hava bir engelle karşılaşır ve bu engeli aşmak için zorlama olunca, ses patlama şeklinde çıkar. Sürekli ünsüzlerde ses herhangi bir kesintiye uğramadan ses üretim yolundan sızarak çıkar.

Sessiz sesler ses tellerinin titreşim durumuna göre sert (ötümsüz-sessiz) ve yumuşak (ötümlü-sesli) olarak iki grupta toplanabilir. Sert ünsüzler ses tellerini titreştirmeden çıkarlar. Yumuşak ünsüzler ise ses tellerini titreştirerek çıkarlar. Kapanma ünsüzleri ağız kanalında tam bir kapanmayla oluşurlar ve ötümlü yada ötümsüz olarak sınıflanabilirler. Daralma ünsüzleri ağız kanalının çeşitli yerlerinde bir daralma sonucunda oluşurlar ve sürtünücü (fricative)

olarak da adlandırılırlar. Kapanma-daralma ünsüzleri olan /c/ ve /ç/, ses üretim yolundaki bir kapanma sonucunda patlamalı sızmalı bir ses verirler.

Çizelge 2.2. Türkçe’de fonemlerin gösterimi.



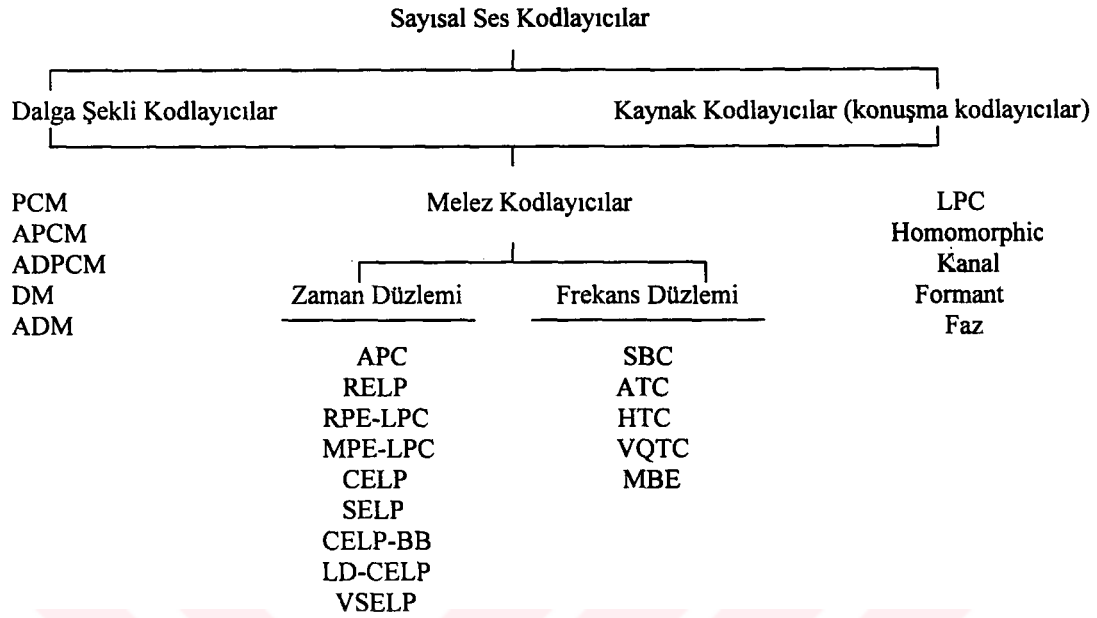
Nazal sesler, yutaktan gelen havanın hem ağza hem buruna geçirilmesi yoluyla olur. Yan ünsüzler soluğun dilin iki tarafından akmasıyla oluşur. Çarpmalı ünsüzler dilin dişetlerine çarpmasıyla oluşur. Türkçe’deki ses alfabeti Çizelge 2.2, deki gibi gösterilebilir. Türkçe ünlü ve ünsüz ses grupları ise Çizelge 2.3’de görülebilir.

Çizelge 2.3. Türkçe’de ünlü ve ünsüz ses grupları

Ünlüler	Düz		Yuvarlak	
	Geniş	Dar	Geniş	Dar
Kalın	A	ı	o	u
İnce	E	i	ö	ü

Ünsüzler	Süreksiz	Sürekli
Sert (ötümsüz)	p,ç,t,k	f,h,s,ş
Yumuşak (ötümlü)	b,c,d,g	ğ,j,l,m,n,r,v,y

2.5. Sayısal Kodlama Teknikleri



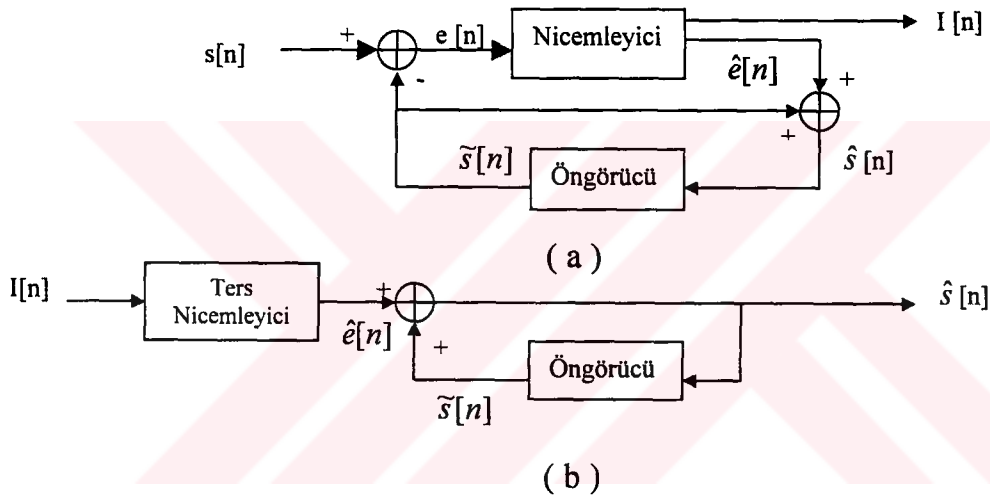
Çizelge 2.4. Sayısal kodlama tekniklerinin sınıflandırılması

Sayısal ses kodlama teknikleri, Çizelge 2.4’de görüldüğü gibi, dalga şekli kodlayıcılar, konuşma kodlayıcılar ve melez kodlayıcılar olarak üç ana sınıfa ayrılabilir. Melez kodlayıcılar, diğer iki kodlayıcı sınıfının ortak özelliklerini taşıyan kodlayıcılardır. Her üç kodlayıcı sınıfının genel amacı, sinyali analiz etmek, sinyaldeki istenmeyen şartları (gürültü, önemsiz kısımlar, vb.) kaldırmak, sinyali kendi sınıfının belli bir kalite yüzdesini koruyarak kodlamaktır. Yüksek bit hızlarında kodlayıcı kalitesi oldukça iyidir. Düşük bit hızlarında da, belli bir yüzde kaliteyi koruyarak kodlama yapılmaya çalışılır.

2.5.1. Dalga şekli kodlayıcılar

Dalga şekli kodlayıcılar, sinyal dalga şeklinden örnekler alıp en iyi şekilde kodlayıp tekrar üretmeye çalışırlar. Genellikle çok değişik sinyalleri kodlayabilmek için sinyal yapısından bağımsız dizayn edilirler. Gürültü, kanal hataları ve değişik konuşmacı seslerine karşı daha dirençlidirler.

Dezavantajları orta bit hızları altında iyi çalışmamalarıdır. Dalga şekli kodlayıcıların konuşma kodlamadaki bit hızı, 16 kbit/sn üzeri ile sınırlandırılmıştır. Dalga şekli kodlayıcılar, konuşma kodlayıcılara göre daha güçlüdürler çünkü çok değişik sinyallerde kullanılabilirler. Ayrıca, basit dizaynları ve maliyetlerinin düşüklüğü mobil radyo ve ticari telefon devrelerine kadar geniş bir alanda kullanılmalarını sağlar. Konuşma kodlayıcıların avantajları ise düşük bit hızlarında çalışmalarıdır. Konuşma kodlayıcılar sadece kodlanmış sinyaldeki sesin istatistiksel özelliklerini saklarlar.



Şekil 2.3. DPCM a) kodlayıcı b) çözücü blok şeması

Bazı dalga şekli kodlayıcı yöntemleri PCM, DPCM (Differential PCM), ADPCM (Adaptive Differential PCM), DM (Delta Modulation) ve ADM (Adaptive DM)'dir. Dalga şekli kodlayıcılar zaman ve frekans düzlemlerinde çalışabilirler.

İlk dünya çapındaki dalga şekli kodlayıcı standardı, CCITT'nin 64 kbit/sn logaritmik nicemlenmiş (μ -law ya da A-law tekniği) PCM [2][34]'dir. (μ -law, kuzey Amerika için ve A-law Avrupa için standart olarak kullanılmıştır.) PCM hala sayısal sistemlerde yaygın olarak kullanılmaktadır. 64 kbit/sn logaritmik

PCM ile kodlanmış ses, mükemmel kalite (toll quality) kabul edilir ve diğer kodlayıcılar için referans olarak alınır. 32 kbit/sn ADPCM, CCITT'nin PCM'den sonraki diğer bir standardıdır [29]. Bit hızını yarıya indirmek için, ses sinyalindeki önemsiz olduğu düşünülen kısımlar uyarlamalı nicemleme ve uyarlamalı tahmin metotlarıyla çıkarılmıştır [35][36].

PCM bitişik örnekler arasındaki varolan ilişki avantajını dikkate almazken, DPCM orijinal sinyal yerine hata sinyalini nicemleyerek bit hızını azaltır. Şekil 2.3, bir DPCM kodlayıcı ve çözücü blok şemasını göstermektedir. O andaki örnek, $s[n]$, ile onun tahmin değeri, $\tilde{s}[n]$, arasındaki fark, $e[n]$, hata değeri $\hat{e}[n]$, olarak nicemlenir. Hata kod kelimesi $\hat{e}[n]$ için, indeks değeri $I[n]$, kodlayıcıdan kod çözücüye aktarılır. Bir doğrusal öngörücü p önceden oluşturulan örnekler $\hat{s}[n-i] = \hat{e}[n-i] + \tilde{s}[n-i]$ ile o anki $s[n]$ örneğini tahmin eder. $\{a_i / 1 \leq i \leq p\}$ doğrusal öngörü katsayıları ise, tahmin değeri $\tilde{s}[n]$ aşağıdaki gibi açıklanabilir:

$$\tilde{s}[n] = \sum_{i=1}^p a_i \hat{s}[n-i] \quad (2.1)$$

Diğer bir dalga şekli kodlayıcı, sürekli değişken eğimli delta modülasyon (Continuous Variable Slope Delta Modulation-CVSDM), 32 kbit/sn'de özel uygulamalarda çok başarılıdır. Bu yüksek bit hızlı dalga şekli kodlayıcılar, günümüz şartlarında çok yeterli olmamalarına rağmen, hala büyük sistemlerde hizmet etmektedirler ve sistem yenileme maliyetinin yüksekliği nedeniyle hizmetlerini devam ettirme ihtimalleri yüksektir.

2.5.2. Konuşma kodlayıcılar

Dalga şekli kodlayıcılar, kodladıkları sinyalin yapısı hakkında bilgi içermezler. Konuşma kodlayıcılar (vocoders), kaynak kodlayıcılar (source

coders) olarak da bilinirler ve ses üretim yolunun bir modelini oluşturmaya çalışırlar.

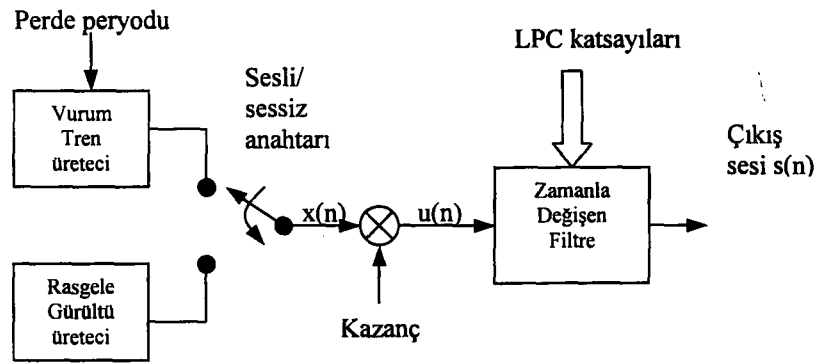
2.5.2.1. Doğrusal öngörülü kodlayıcı

Doğrusal öngörülü konuşma kodlama (LPC), son yirmi yıldır konuşma kodlama teknikleri için en yaygın kullanılan yöntemdir. Doğrusal öngörülü kodlayıcılar, ses üretimi için diğer konuşma kodlayıcılarla aynı modeli kullanırlar. Sadece ses üretim yolu belirlenmesi farklıdır. Bu kodlayıcı ses üretim yolunun bir tüm kutup sonsuz dürtü cevaplı filtre ($H(z)$) ile ifade edildiğini farzeder. p 'inci dereceden ileri doğrusal öngöründe, o andaki ses örneği, p adet geçmiş sinyalin birleşiminden tahmin edilir. Filtre katsayıları a_i ($i=1,2,...,p$), tahmin edilen ve gerçek değer arasındaki hatayı minimum yapacak şekilde seçilirler.

$$H(z) = \frac{G}{1 + a_1 z^{-1} + \dots + a_p z^{-p}} \quad (2.2)$$

Ses sinyalini sayısal formda göstermek için, Fant'ın doğrusal ses kaynak üretim sistem modeli pek çok uygulama için bir referans olmuştur [3]. Bu model, sesli ses sinyali için periyodik vurum treni uyarma sinyali ile veya sessiz ses sinyali için rasgele uyarım sinyali ile uyarılan ve ses üretim yolu denen zamanla yavaş değişen bir sistem içermektedir (Şekil 2.7). Ses üretim yolu, bir tüm kutup filtre fonksiyonu ile ifade edilen, zamanla yavaş değişen bir sistemdir. Kısa süreli (20ms civarı) analiz penceresi içindeki ses sinyali, p adet filtre katsayılarını belirlemek için analiz edilir. Orijinal ses, ters ses üretim yolu filtreden geçirilir ve artık sinyal bulunur. Bu işlem, bitişik örnekler arasındaki kısa süreli ilişkiyi kaldırır. Böylece, artık sinyal, ses sinyalindeki uzun süreli ilişkiyi daha net ortaya çıkardığından perde periyodunu belirlemede kullanılabilir. Ayrıca, sinyalin sesli/sessiz analizi yine

bu artık sinyalden yapılabilir. Doğrusal öngörülü kodlayıcılar, Şekil 2.4'deki ses üretim yolu modelleme çok iyi çalıştığından çok popülerdir ve 2.4 kbit/sn civarındaki bit hızlarında bile kodlama kaliteleri iyidir. [25,26,49]



Şekil 2.4. Ses üretimi için temel kaynak filtre (Fant) modeli

Doğrusal öngörülü kodlayıcılar, formant kodlayıcılarda ilk formant tahmini için kullanılabilirler çünkü, ses üretim yolu filtre kutupları sesteki formantlara çok yakın değerlerdir. Bazı doğrusal öngörü tabanlı kodlayıcılar, 2.4 kbit/sn LPC-10, Amerikan Federal Standardı 2.4 kbit/sn MELP, 9.6 kbps RELP (Residual Excited Linear Prediction) kodlayıcılarıdır.

2400 bit/sn LPC-10 konuşma kodlayıcı

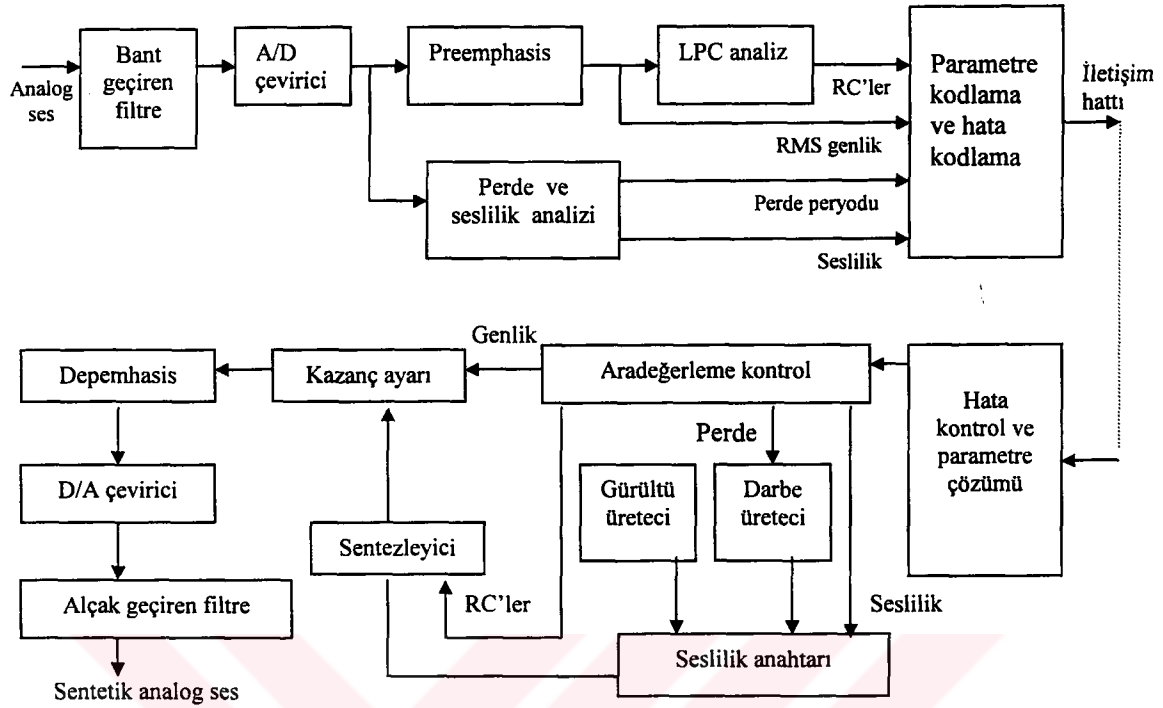
2400 bit/sn LPC-10 konuşma kodlayıcı için, analog ses sinyali önce şekil 2.5'de görüldüğü gibi, band geçiren filtreden geçirilir, A/D (analog/dijital) çeviriciye girer, önvurgulama (preemphasis) uygulanır ve RC yansıma katsayılarını bulmak için LPC analiz yapılır. RMS (Root Mean Square) genlik, seslilik durum analizi yapılır. Ses sinyali sesli ise, perde periyodu belirlenir (Perde periyodu olarak (20-156) arasında 60 perde değerinden biri (51-400 Hz) seçilir). Sessiz durum içeren ses sinyali için analiz penceresinin

sessiz ya da ses geiş durumunda olup olmadığı incelenir. RMS genlik, perde peryodu, yansıma katsayıları, sesli/sessiz durumu kodlanır. (Perde kodlama çizelgesinde tek bit sıfır olan değler sessiz, iki bit sıfır olan değler ses geiş durumu, diğler değler ise sesli perde peryodu için kullanılır.) Sessiz veya ses geiş durumları içeren pencerelerde aktarım hattı hata kontrol bitleri eklenir. Senkronizasyon biti ilave edilir ve Çizelge 2.5'deki 54 bit kodlama yapısı oluşur.

Çizelge 2.5. 2400 bit/sn LPC-10 kodlayıcı bit yerleşimi

2400 bit/sn LPC kodlayıcı	Sesli durum	Sesli olmayan durum
Perde & seslilik	7	7
RMS genlik	5	5
Yansıma katsayıları	41	20
Hata kontrol	-	20
Senkronizasyon	1	1
Kullanılmayan	-	1
Toplam	54	54

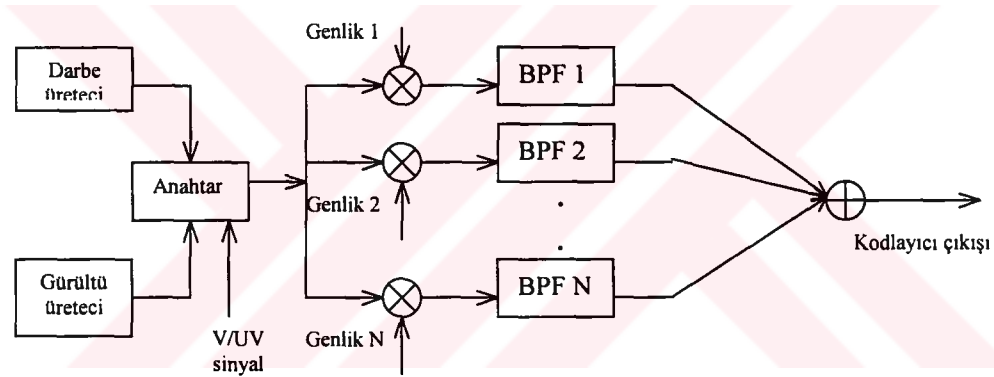
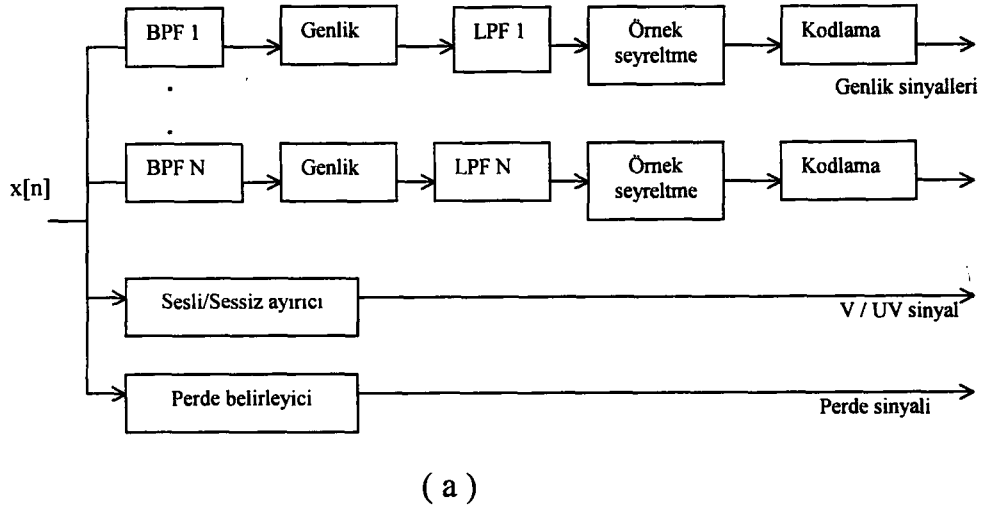
LPC sentezleyicide önce bir hata kontrol gerçekleştirilir. Aktarılan parametreler çözülür. Uygun sesli/sessiz, perde peryodu, genlik, yansıma katsayıları durumlarını sağlamak için aradeğerleme ve diğler işlemler yapılır. Sesli durum içeren ses sinyali için, perde uyumlu darbelerle uyarılmış 10'uncu dereceden bir tüm kutup filtre kullanılır. Sessiz durum içeren ses sinyali için ise rasgele değlerle uyarılmış bir 4'üncü dereceden tüm kutup filtre kullanılır. Hem sesli, hem sessiz durumları birlikte içeren pencereler için, sesli durum içeren pencere bölümü perde uyumlu darbelerle uyarılmış 10'uncu dereceden bir tüm kutup filtre, sessiz durum içeren pencere bölümü rasgele değlerle uyarılmış bir 4'üncü dereceden tüm kutup filtre ile filtrelenir. Yansıma katsayıları sentezleyici filtreye uygulanır. RMS genlik ile kazanç ayarlanır. Sonuç fonksiyon deemphasis filtreden geçirilerek D/A çevirici ile analog sinyale çevrilir ve alçak geçiren filtreden geçirilir. Deemphasis filtre transfer fonksiyonu, $1/(1-0.75z)$ şeklindedir.



Şekil 2.5. Bir LPC-10 kodlayıcı/sentezleyici blok şeması

2.5.2.2. Kanal kodlayıcı

Kanal kodlayıcıda, ses üretim yolu bilgisi, ses sinyali bant geçiren filtrelerden geçirilerek elde edilir. Perde peryodu, Sesli/Sessiz (Voiced/Unvoiced-V/U) bilgisi gibi bilgilerle birlikte kodlanıp alıcıya gönderilir. Kanalların band genişliği frekansla artan şekilde seçilir. Bu insan kulağının, logaritmik ölçeklendirilmiş frekanslara doğrusal cevap vermesi nedeniyledir. Kanal kodlayıcılar tipik olarak, 1200 bit/sn ve 9600 bit/sn arasında işletilirler. Kanal kodlayıcı ve ilk formant kodlayıcılar arasındaki temel fark, formant kodlayıcılardaki filtre bankasının rezonans karakteristiklerinin, formant izlerine kendini uyarlamasıdır. Dudley'in konuşma kodlama sistemi, kanal kodlayıcıların en eskisidir (Şekil 2.6). Bu kodlayıcı kulağın, kısa süreli faz bilgisine duyarsız oluşu avantajını alır. (20 msn bir ses sinyali için, sadece genlik bilgisi kullanılır).

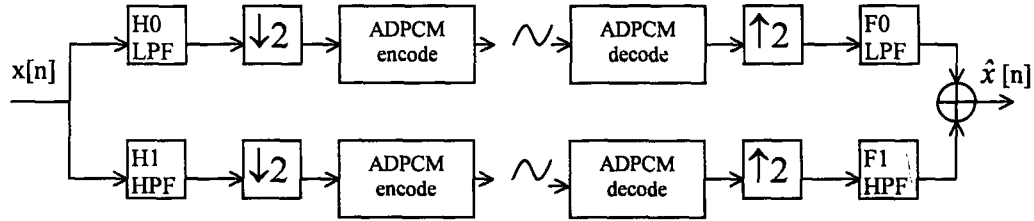


Şekil 2.6. Kanal kodlayıcı a) analiz b) sentez blok şeması

2.5.2.3 Alt bant kodlayıcı

Alt bant kodlama (subband coding –SBC), orta bit hızlarında (16 kbit/sn civarı) konuşma kodlama için iyi bilinen bir tekniktir. Ayrıca görüntü ve video sinyallerini kodlamada da yaygın olarak kullanılır. Temel bir alt bant kodlama işleminde, ses sinyali önce alçak geçiren filtre ve yüksek geçiren filtre ile frekans bantlarına ayrılır. Bantlara ayrılmış sinyal aşağı örneklenir (down sampling) ve aktarım için kodlanır. Alıcıda kanal sinyalleri çözülür,

aradeğerlenir ve yaklaşık sinyali oluşturmak için birleştirilirler [102,104]. Bir iki bantlı alt bant kodlayıcı sistem şeması Şekil 2.7’de görülmektedir.

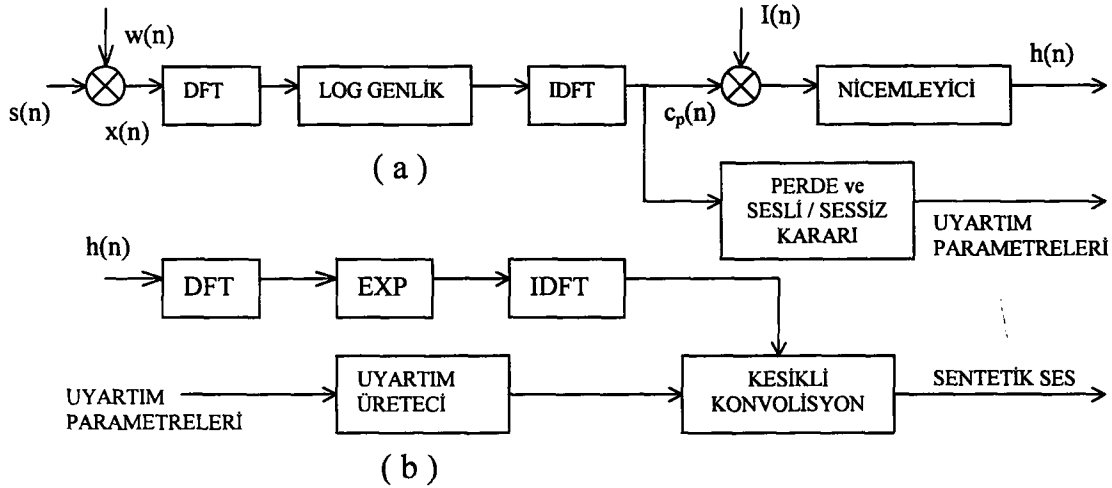


Şekil 2.7. İki bant analiz/sentez filtre bankası

Ses sinyali birden fazla alt banda ayrılabilir. Her alt bandın kodlanması için kullanılan bitlerin sayısı, ilgili bandın yüzde önemine göre değişebilir. Örneğin perde ve formant bilgisinin önemli kısmını içeren alçak frekans bantları daha çok bitle, gürültü türü seslerin olduğu yüksek frekans bantları daha az bitle kodlanabilir. Ayrıca, ayrı nicemleme basamakları kullanıldığında, bir bant için oluşan nicemleme gürültüsü, diğer bandı etkilemez ve böylece diğer frekans bantları korunur. Bu şekilde PCM kodlamaya göre önemli avantajlar kazanılır.

2.5.2.4 Homomorphic kodlayıcı

Ses analiz problemi, uyartım sinyali ve ses üretim yolu modelinin konvolisyon yapıda birleşik yapısını ayırıp ayrı parametreler olarak ifade etmektir. 4 kbit/sn hızlarında çalışabilen homomorphic (cepstral) kodlayıcılar, ses sinyalini zaman düzleminde, uyartım ve ses üretim yolu yapının logaritmik genlik toplamı olarak farzeder. Bu toplam, frekans düzlemde çarpım olarak alınır. Bir homomorphic kodlayıcı yapısını gösteren Şekil 2.8’de $s(n)$ ses sinyali, $l[n]$ cepstrum penceresi, $w[n]$ kısa süreli analiz penceresi, $c_p[n]$ cepstrum parametreleridir.

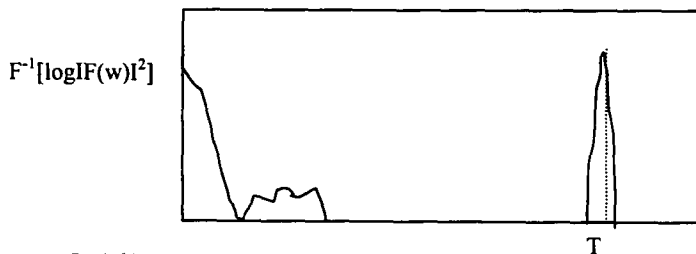


Şekil 2.8. Homomorphic kodlayıcı a) analiz b) sentez blok şeması

$S(e^{j\omega})$ ses spektrumu, $P(e^{j\omega})$ uyartım spektrumu, $V(e^{j\omega})$ ses üretim yolu spektrumu ise,

$$\log[S(e^{j\omega})] = \log[P(e^{j\omega})] + \log[V(e^{j\omega})] \quad (2.3)$$

ile ifade edilir. Cepstral katsayılar, logaritmik genlik spektrumu $|S(e^{j\omega})|$ 'nın ters fourier dönüşümü ile bulunur. Ses üretiminde, uyartım sinyali hızlı değişirken ses üretim yolu dürtü cevabı nispeten yavaş değişir. Bu zaman farkı nedeniyle, alçak geçiren filtre (Low pass filter-LPF) ile iki sinyal birbirinden ayrılabilir (Şekil 2.9). Bu nedenle perde bilgisi de doğru belirlenebilir.



Şekil 2.9. Gecikmeli perde bilgisi gösteren bir ses spektrumu

2.5.2.5 Formant kodlayıcı

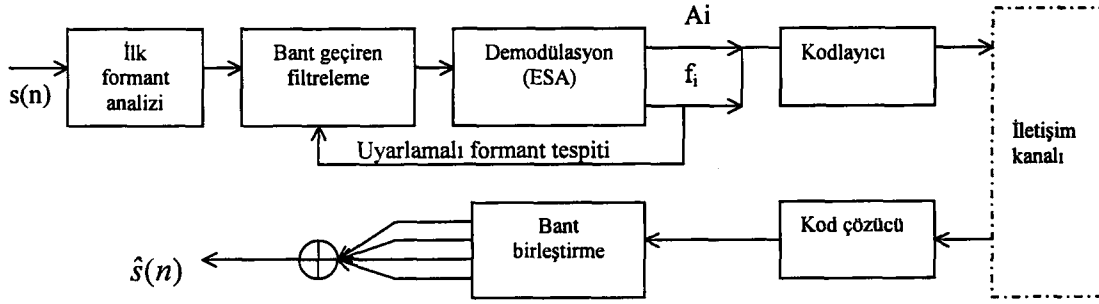
Ses sinyalindeki bilginin büyük çoğunluğu ses üretim yolu formant'ların pozisyonlarında ve bant genişliklerinde içerilir. Bu yüzden, eğer formantlar doğru belirlenebilirse, çok düşük bit hızlarında kodlama yapılabilir. Formant kodlayıcıda, filtre bankasının rezonans karakteristiği, filtre bilgisini izler. AM-FM modülasyon tekniği, formant kodlayıcıların iyi bir örneğidir. AM-FM modülasyon tekniğine dayalı kodlayıcılar, 4.8 kbit/sn ve 9.6 kbit/sn arasında iyi çalışırlar [37-43].

AM-FM Modülasyon

Kodlama teknikleri içinde, sesin ses üretim yolu yapısının zamanla değişen filtre parametreleri ile modellendiği doğrusal öngörü (Linear Prediction - LP) metodu yerine alternatif bir metot olan AM-FM (amplitude modulation-frequency modulation) konuşma kodlayıcı incelenmiştir [37-40,43]. Burada ses üretim modeli, “formant” denilen rezonans frekanslarını modelleyen doğrusal bir filtre boyunca geçen uyartım sinyalinin çıkışı olarak gösterilir. Uyartım genellikle sesli ses için bir periyodik vurum treni ve sessiz ses için gürültü türü sinyalle gösterilir. Verilen bir analiz penceresi içinde, uyartım sinyalinin sinüzoidal gösterimi zamanla değişen bir ses üretim yolu boyunca geçirildiğinde $s(n)$ ses dalga çıkışı,

$$s(n) = \sum_{l=1}^L A_l \cos(w_l n + \theta_l) \text{ 'dir.} \quad (2.4)$$

A_l , w_l ve θ_l sırasıyla sinüs dalga genlik, frekans ve fazı, L ise sinüs dalga sayısıdır. Bu sistemin doğruluğu için sinüs dalgaların ses üretim yolu sistem cevabına yani doğrusal filtreye göre yavaş değiştiği kabul edilir. Bir AM-FM kodlayıcı blok şeması Şekil 2.10'da görülmektedir.



Şekil 2.10. Bir AM/FM kodlayıcı/çözücü blok şeması

Burada genlik ve frekans sinyallerini tahmin etmek için, Maragos, Kaiser ve Quatieri [43] bir “Energy Separation Algorithm -ESA” geliştirmişlerdir. ESA, AM-FM sinyalin ani enerjisini izlemeye ve onu frekans ve genlik parçalarına ayırmada doğrusal olmayan bir enerji operatörü kullanmaktadır. Ses sinyali $x(t)$ için enerji operatörü Ψ_c ;

$$\Psi_c[x(t)] = x(t)^2 - \dot{x}(t) \cdot \ddot{x}(t) \quad (2.5)$$

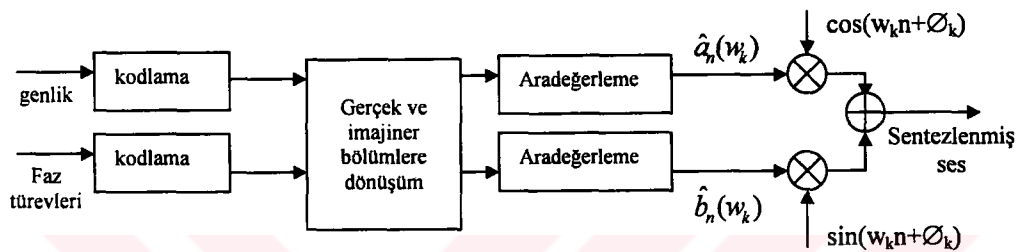
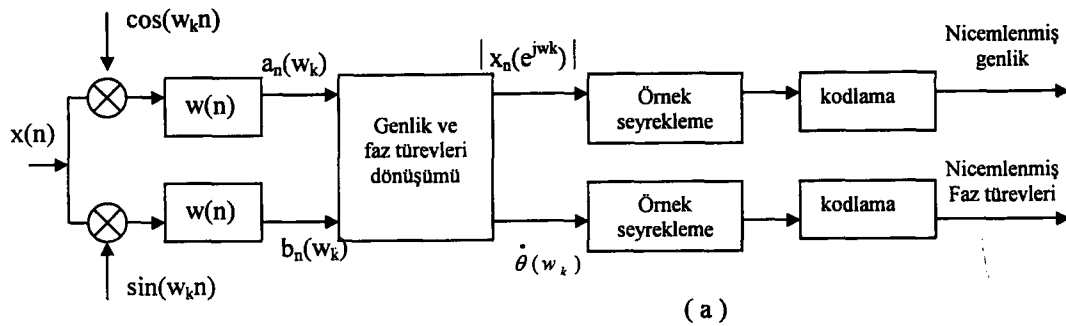
olarak tanımlanır. $\dot{x}(t)$ ve $\ddot{x}(t)$ sırasıyla $x(t)$ 'nin birinci ve ikinci dereceden türevlerine karşılık gelmektedir. ESA, tek bir ses rezonansına uygulandığı ve ses sinyali birden fazla rezonansın oluştuğu için, bu rezonansları bant-geçiren filtre ile filtrelemek gerekir. Bunun için kullanılan filtre yapısı Gabor filtresidir. Burada, filtrenin merkez frekansı ve filtre bant genişliği belirlenmesi çok önemlidir. Bant genişliği, istenen formant modülasyonunu içerecek kadar geniş fakat komşu formantları çıkaracak kadar dar olmalıdır. Formantları ve bant genişliklerini bulmada kullanılan iki yöntem MDA (Multiband Demodulation Algorithm) ve iterative ESA [43] dir.

AM-FM konuşma kodlayıcı modelde, modülasyon sinyallerinin alıcı tarafta çok iyi modellenmesi için yapılacak kodlamada çok fazla bit gerektirmesi nedeniyle, düşük bit hızlarında iyi performans vermediği sonucuna varılmıştır. AM-FM formant modellenmesi, doğrusal öngörü metoduna göre üstünlükler sağlayan bir yöntemdir. Fakat burada da formant yerlerinin çok iyi tespit

edilebilmesi gerekir. Özellikle gürültülü ve zayıf ses sinyalleri için yanlış formant tespitleri sesin kalitesinde bozulmalara yol açmaktadır. AM-FM modülasyon modelde rezonanslar yavaşlatılmış AM-FM sinyalleri olarak modellenir. AM-FM modülasyon model doğru formant izleri ve gerçek formant bant genişliği tahminleri yapar. Ayrıca, AM-FM modülasyon modelini frekans ve zaman düzleminde oluşturmak kolaydır ve doğrusal öngörü tabanlı formant izleyicilerdeki hatalardan kaçınır. Oysa, doğrusal öngörü metodu ile formant analiz nazal formantları tespit etmede, izgel sıfırları bulmada ve bant genişlik tahmininde problemlidir. Bu doğrusal öngörü metodunun parametrik olmasından, formantların bir analiz penceresi içinde sabit kabul edilmesinden ve izgel vadilerin modellenememesinden kaynaklanır. Doğrusal öngörü, ses akustiğini tam olarak modellemede yetersizdir [54].

2.5.2.6. Faz kodlayıcı

Faz kodlayıcı 1966'da önerilmiştir [4]. Bu sistemde ses sinyali, merkez frekansları ve analiz pencereleri istenen frekans bandını içerecek şekilde seçilmiş bant geçiren filtrelerle ifade edilir ve kompleks rezonanslar öyle seçilir ki, bütün kanallar toplamının bileşik cevabı doğrusal faz ve düz genlik çıkaracak şekilde seçilirler. Faz başlangıç şartları doğru belirlenemezse, tüm sistemde bir hata sonucunu verebilir. Bir faz kodlayıcı blok şeması Şekil 2.11'de görülebilir. Burada a_n ve b_n sinüzoidal giriş sinyalinin gerçek ve sanal bölümlerini göstermektedir. θ_k , sinüs sinyalinin zamanla değişen açısai değeri, ϕ_k ise faz kaymasıdır.

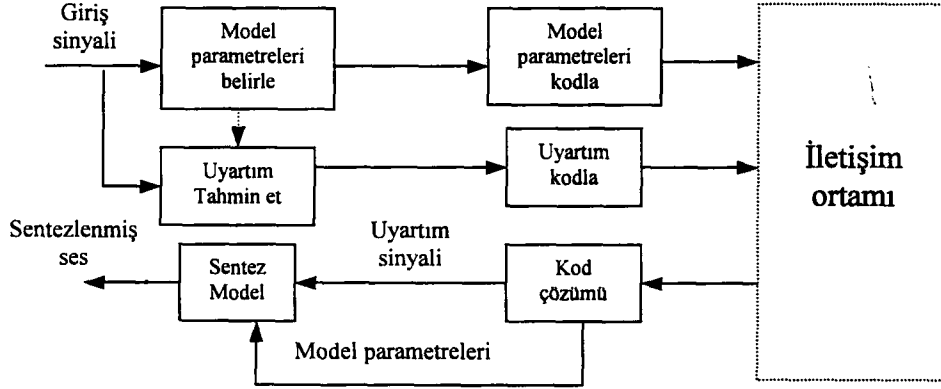


Şekil 2.11. Bir tek kanal faz kodlayıcı a) analiz b) sentez şeması

2.5.3. Melez kodlayıcılar

Dalga şekli kodlayıcılar, 32 kbit/sn civarı orta hızlarda çok iyi çalışırlar ve kodlanmış sinyali çok iyi bir şekilde tekrar üretebilirler. Konuşma kodlayıcılar ise düşük bit hızlarında (8 kbit/sn altında) çalışırlar. Sonuçta ürettikleri ses anlaşılabilir olmasına rağmen yapaydır. Dalga şekli kodlayıcılar ve konuşma kodlayıcıların bazı özelliklerini birleştirerek yüksek kaliteli ve düşük hızlı sesler üretebilmek mümkün olmuştur. Melez (hybrid) kodlayıcılar, ses dalga şeklinin yüzde olarak daha önemli kısımlarını korumaya çalışırlar. Melez kodlayıcılar, zaman düzlemi ve frekans düzlemi olmak üzere iki ana sınıfta incelenebilir. Frekans düzlemi melez kodlayıcıların ana yapısı, ses spektrumunu filtre bankalarıyla frekans bantlarına ya da blok dönüşüm metotlarıyla frekans parçalarına bölmektir. Alıcıda bu frekans bantları, filtre bankası toplamı veya ters dönüşüm metotlarıyla birleştirilerek ses sinyali

tekrar oluşturulmaya çalışılır. Frekans düzlemi kodlayıcıların esas temeli, ses sinyalinin kısa süreli spektrumunun zamanla yavaş değişmesidir. Bir melez kodlayıcı blok yapısı Şekil 2.12’de görülmektedir.



Şekil 2.12. Bir melez kodlayıcı/sentezleyici blok şeması

Zaman düzlemi melez kodlayıcılar, doğrusal öngörü şemaları ile dizayn edilirler. Ses sinyalinin istatistiksel yapısı, sessiz ses için rasgele (random) ya da sesli ses için periyodik darbelerle uyarılan bir kaynak filtre yapısıyla modellenenebilir.

Alt bant kodlayıcılar (SBC) ve uyarlamalı dönüşüm kodlayıcılar (ATC), frekans düzleminde sinyali filtre bankaları ile frekans parçalarına bölerler ve ses sinyalinin gösteriminde önemli bilgiler içeren parçaları daha çok bit sayısı ile, sinyalin tekrar oluşturulmasında büyük katkısı olmayan parçaları ise düşük bit sayılarıyla kodlarlar. Her iki kodlayıcı için bit oranını indirme, sesin kısa zaman güç spektrumunu çıkarma ve insan kulağının yüksek frekanslı seslere daha az duyarlı olma mantığına dayanır. Bazı melez kodlayıcı örnekleri, RELP, MPLP, CELP, LD_CELP, CELP_BB, VSELP, MELP, STC, MBE, TFI, RPE-LPC, MPE_LPC olarak sıralanabilir [17,49,85].

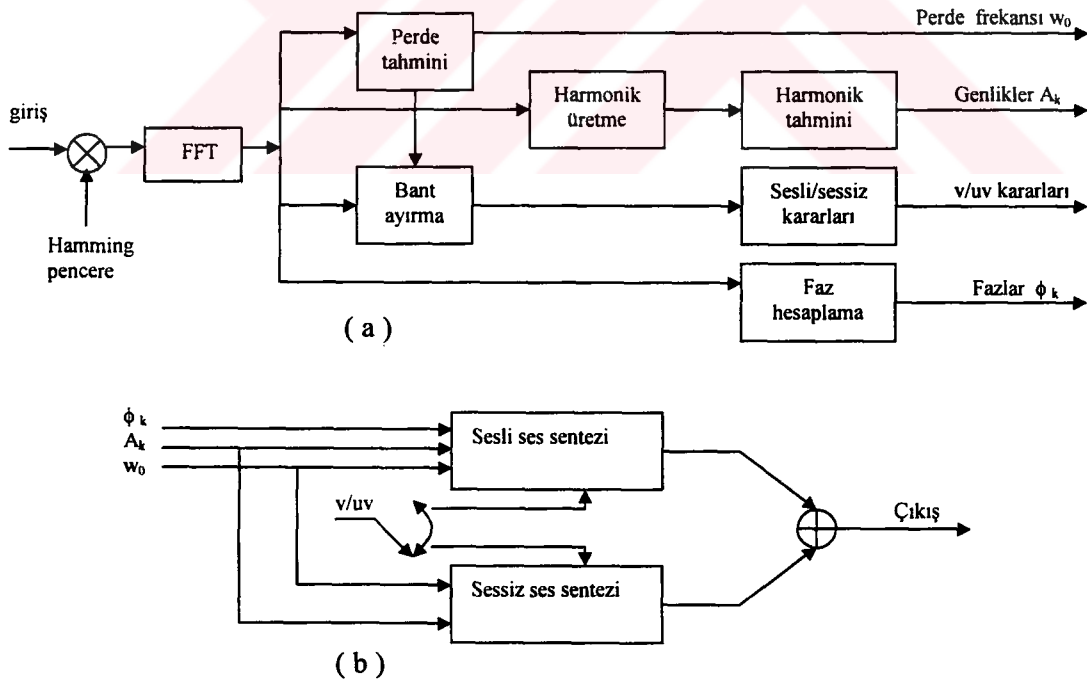
2.5.3.1. MBE (Multi Band Excitation)

Frekans düzlemi melez kodlayıcılara iyi bir örnek, 2.4 kbit/sn hızlarında dahi iyi performans veren sinüzoidal tabanlı MBE kodlayıcılardır. MBE modelde ses sinyali spektrumu, izgel zarf (spektral envelop) ve uyartım sinyalinin çarpımı olarak modellenir (Şekil 2.13). Ses analizinde model parametreleri, orijinal ve sentetik ses spektrumu arasında bir karşılaştırma ile elde edilir. Ses sentezinde, sesin sesli kısmı zaman düzleminde, sessiz kısmı frekans düzleminde sentezlenir. MBE kodlayıcının diğer kodlayıcı türlerine göre en büyük avantajı, her analiz penceresinde tek bir V/U kararı yerine, ses spektrumunu frekans bantlarına bölerek birden fazla V/U kararlarından oluşmasıdır. Böylece diğer kodlayıcılarda görülen “buzz - vızıltı” MBE kodlayıcılarda görülmez. [45-47] MBE kodlayıcılarda kodlanarak karşı tarafa aktarılan parametreler; uygulama frekansı (perde peryodu), harmonik genlikler, V/U kararları ve sesli sesteki harmonikler için faz bilgisidir. MBE de bit hızını indirmek için özellikle harmonik genliklerin kodlanması çok önemlidir. MBE de bit sayısını azaltmada kullanılan yöntemler; analiz penceresini küçültme ya da bitişik pencereler arası aradeğerleme metotlarıdır. Aradeğerleme metodunda, ses kalitesinin düşmemesi için özellikle perde peryodu her pencerede hesaplanıp alıcıya gönderilmelidir. [48]

MBE analiz : Perde peryodu ve izgel zarf MSE minimum olacak şekilde belirlenir. V/U bölgelerin belirlenmesi için, orijinal ve sentetik spektrum arasındaki hata kontrol edilir. Hata belli bir eşik üzerinde ise o bölge “sessiz” kabul edilir. Harmonik genlik hesabı için, genlikler V ve U bölgeler için ayrı hesaplanır. Sesli bölgelerde sentetik ve orijinal genlikler birbirine çok benzediğinden sesli harmonik genlikleri sentetik harmoniklerden elde edilir.

Sessiz bölgelerde ise orijinal ve sentetik harmonikler arasındaki fark büyük olduğundan, sessiz harmonik genlikleri RMS değerle hesaplanır.

MBE sentez: Sessiz sentez için, harmoniklerden elde edilen sessiz ses parçası, her pencere için rasgele gürültü sinyalinin Fourier dönüşümü alınır. Spektrumdaki sesli parçalar sıfır yapılır. Sessiz parçalar sessiz ses harmoniklerine normalize edilir. Uyarım sinyali yani gürültü eklenir. Ters fourier dönüşümü alınır. Sesli ses için uyarım sinyali osilatör bankası ile oluşturulur. Her osilatör, penceredeki sesli ses için gösterilir. Sessiz harmonik kısımlarındaki genlikler sıfır yapılır. Sesli harmonikler, bir önceki pencere analizi ve o anki değerler arasında aradeğerleme ile belirlenir. Uyarım sinyali eklenir. Faz bilgisi, sesli bölgeler için çok önemli olduğundan bütün fazlar sentez bölümünde, önceki pencere analizindeki perde peryodu ve fazı ile o anki penceredeki perde peryodu kullanılarak bulunur. [48]



Şekil 2.13. Bir MBE (a) kodlayıcı ve (b) çözücü blok şeması

2.6. Kodlayıcı Performansı

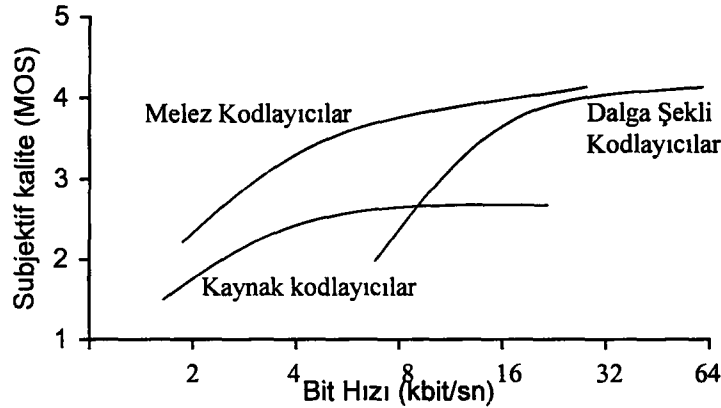
Herhangi bir dizayn işleminde sonuç ürün, belli gereksinimleri sağlamalıdır. Bu gereksinimler PSTN sistemler için CCITT gibi dünya çapında kuruluşlar tarafından belirlenebilir ya da özel veya mobil uygulamaları için ulusal ya da özel kuruluşlar tarafından belirlenebilir. Bu standartları belirlerken dikkate alınan pek çok dizayn kriteri, aşağıdaki ana başlıklar altında düşünülebilir. Bütün bu kriterlerden ortaya çıkan sonuç ile kodlayıcı performansı hakkında bilgi edinilir. [26,27]

2.6.1. Kalite ve bit hızı

Kalite, düşük bit hızlı sayısal sistemlerin dizaynında en önemli faktördür. Ses kalitesi ve bit hızı zıt durumlardadır. Yüksek kaliteli ses, düşük bit hızları ile elde edilmeye çalışılır. Düşük bit hızlarında kodlama çabası, konuşma kodlayıcıların önemini artırmaktadır. PSTN'e bağlı sistemler için kalite gereksinimi ve sınırlamalar CCITT tarafından belirlenmiştir. Böyle sistemlerin çok yüksek kalitede (toll quality) kodlama yapması istenir (dalga şekli kodlayıcılar). Yine de, ticari haberleşme ve askeri sistemler gibi kapalı sistemlerde kalite gereksinimi belli bir oranda düşürülebilir (melez kodlayıcılar). Mobil radyo sistemlerinde, iyi ve kötü aktarım hattı şartlarına göre ortalama bir kalite istenir

Kalite yargısına, uygulamalı durumlarda standartlaşmış kodlayıcı testleri ve uygun görülen sınırlar ile varılır. Kalite, radyo kalitesi (broadcast), mükemmel-iyi (toll, excellent-good), iyi-anlaşılır (communications, good-fair), anlaşılır-zayıf (synthetic, fair-poor), şeklinde sınıflandırılabilir (Bkz.

Şekil 1.2). Farklı bit hızları ve subjektif kalite arasındaki eğilim, Şekil 2.14’de görülebilir.



Şekil 2.14. Konuşma kodlayıcı türlerinin subjektif kalite karşılaştırılması

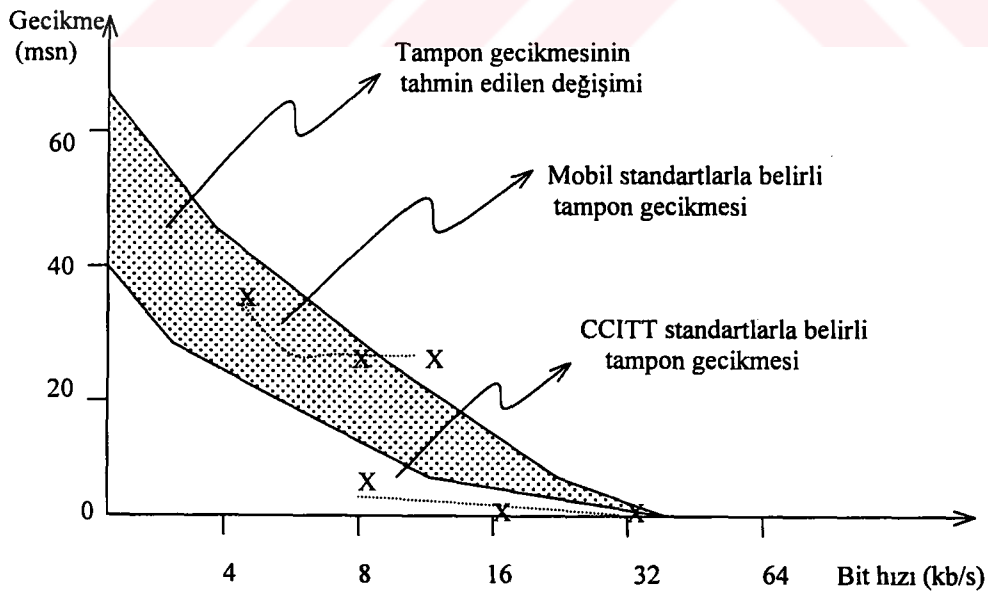
2.6.2. Kanal hatalarına dayanıklılık

Sayısal kodlayıcıların dayanıklılığı (robustness), farklı kanal şartları altında, kodlayıcı çıkış kalitesini gösterir. Dayanıklılık, sayısal kodlayıcıların hatalara karşı direncini göstermede kullanılır. Pek çok kodlayıcı algoritması FEC (Forward Error Correction) ve sinyal parametrelerini birlikte kodlarlar. Fiber optik gibi kodlama hatalarının daha az olduğu hatlarda hata bitleri azaltılabilir ve boşalan bitler sinyal kalitesini artırmada kullanılabilir. Bu özellik, askeri ve uydu sistemleri ile sabit hat sistemleri arasındaki en önemli farktır. Gürültünün varlığı tespit edildiğinde, aktarılan bitlerin tekrar kullanımı da yaygın bir metottur. Yine de bazen gürültü çok büyüktür ve alternatif metotlar gerekir. PSTN ağ durumunda, kabul edilebilir bit hata hızı (Bit Error Rate-BER) 10^{-6} ve 10^{-5} civarındadır. Böyle hata hızlarına güçlü algoritmaların dikkatli dizaynı ile izin verilir. Yine de, bazı çevre şartlarının artırıcı ya da azaltıcı etkileriyle değişebilirler. Kara haberleşmesi için ortalama BER %5, uydu haberleşmesinde ise %1 civarındadır.

2.6.3. İletişim gecikmesi

Sayısal kodlayıcı dizaynında önemli bir husus ta ses sinyalinin aktarımı ve sondan sona işlenmesi gecikmesidir. Aktarım yollarında uzun gecikmeler, gidiş-geliş hatlarındaki uyumsuzluklar nedeniyle alıcı ve verici tarafta rahatsız edici yankılara sebep olur. Uydu haberleşme sistemlerinde 250 ms civarında doğal bir yol gecikmesi vardır ve herhangi ilave bir gecikme büyük etkiler yapabilir. Yankı gidericiler, doğal aktarım gecikmelerinin etkilerini kaldırmada görevlendirilirler. Bu da sistem maliyetini artırır. İşlem gecikmeleri algoritma nedeniyledir ve teknoloji bağımlıdır. Bit hızı ve işlem gecikmesi arasındaki bağlantı Şekil 2.15'den görülebilir. Taralı bölgeler, mobil haberleşmesinde var olan standartlar nedeniyle olan gecikmelerdir. Görüldüğü gibi, daha düşük bit hızlarında gecikme kaçınılmaz olarak artmaktadır.

2.6.4. Birlikte çalışabilirlik ve uyumluluk



Şekil 2.15. Bit hızına karşı işleme gecikmesi

Pek çok standart ve onların farklı şehirlerde uyumsuz olarak işleme sokulmaları, uzak mesafe haberleşmelerinde değişik kodlama ve çözme algoritmaları sonucunu getirecektir. Bu ses sinyalinin çoklu kodlanması, ses sinyalinde tekrarlı kötüleşmelere neden olur. Böyle sorunlar, düşük bit hızlı ses sıkıştırma dizaynında, ses kalitesini artırmak için sinyalin son filtreden geçirilmesi nedeniyle oldukça ciddidir. Sayısal konuşma kodlayıcılar ilave bir maliyet ya da karmaşa gerektirmeden yeni sistemler ve makinelerle birleştirilebilmelidir. Bu yüzden, birlikte çalışabilirlik (interoperability) konuşma kodlayıcılar için önemlidir [33].

2.6.5. Ses bandı veri aktarımı

Başka bir önemli dizayn kriteri, algoritmanın ses dışında sinyalleri aktarabilme yeteneğidir. DTMF (Dual Tone Multi Frequency) sinyaller ya da modem tonları gibi ses spektrumu ile aynı bant üzerinde yer alan fakat farklı istatistiksel özelliklere sahip sinyaller gönderilebilmelidir. Düşük bit hızlı kodlayıcılarda olan son filtreleme ve yüzde filtreler ton sinyallerini bozmakta ve bu yüzden düşük bit hızlı kodlayıcıları atlayan ton tarayıcılar gittikçe yaygınlaşmaktadır.

2.6.6. Şifreleme

Şifreleme ya da kriptoloji, güvenli haberleşme için kullanılır. Genellikle askeri uygulamalar için kullanılır. Sayısal sinyallerin şifrlenmesi, analog sinyallere göre oldukça basittir. Bu yüzden daha güvenli bilgi aktarımı için daha karmaşık şifreleme algoritmaları kullanılabilir.

2.6.7. Karmaşa ve maliyet

Maliyet ve güç tüketimi, kodlayıcı karmaşasının başlıca yan etkileridir. Karmaşa genel olarak algoritma bağımlıdır. Yine de, VLSI (Very Large Scale Integration) çip teknolojisindeki ve mikroişlemci yapımındaki ilerlemeler, karmaşayı problem olmaktan çıkarmaktadır. Sistem maliyeti ise, algoritmanın karmaşıklığına bağlıdır. Karmaşa/güç tüketimi ve bu yüzden maliyet özellikle cep telefonları gibi taşınabilir donanımlar için hala bir sorundur. Güç tüketimini gidermenin bir yolu sayısal aradeğerleme ile kanal etkinliğini artırmaktır. Sayısal aradeğerleme ile, konuşmanın yarısında gerçek ses aktif olarak kullanılır; böylece aktif olmayan periyodlar sırasında kanal diğer amaçlar için kullanılabilir ve güç de korunmuş olur. Alternatif bir metot, ses aktif bölgeleri tespit edilerek sadece konuşma olan bölümlerde kodlama yapılması, konuşmanın olmadığı bölümlerin kodlanmamasıdır. Fakat eğer, konuşma yapılmayan bölümlerle sessiz ses sinyalleri karıştırılıp hatalı kodlama yapılırsa sonuç kodlanmış ses kalitesi kötüdür. Genellikle yüksek kaliteli sesi düşük bit hızlarında kodlamak için, yüksek karmaşa gerekir. Örneğin, düşük bit hızlı bir melez kodlayıcı algoritmasının bir sinyal işleme entegresi üzerinde çalışması saniyede 12 milyon civarında komut gerektirir.

2.7. Uluslararası İletişim Birliği

ITU (International Telegraph Union), değişik milletler arasındaki telgraf iletişimde uyum sağlamak amacıyla 1865’de 20 ülkenin katılımıyla Paris’te kurulmuştur. 1934’de “International Telecommunication Union” adını almıştır. 1997’de ITU, telekomünikasyon gelişimi için uluslararası bir kuruluş olarak, 188 üye devlet ve 402 sektör üyesi (bilimsel ve endüstriyel şirketler, kamu yada özel işletmeciler, ulusal/uluslararası kuruluşlar) içermektedir.

ITU, frekans spektrumu kullanımının uluslararası düzenlemelerdeki bütün karasal ve uydu sınırlamalarını belirler. Aynı zamanda farklı iletişim sistemlerinde uyum ve kolaylık için, kullanılan teknolojiye bağılı olmayan dünya çapında standartlar geliştirir. ITU birliği; teknik alan, geliştirme alanı ve politik alan olmak üzere üç ana başlıkta incelenebilir. *Teknik alan* ; İletişim kolaylıklarının etkin işletimini ve gelişimini sağlayarak, iletişim servislerinin etkinliğini ve yaygın kesimlerde kullanımını arttırmak amacıyla. *Geliştirme alanı*; iletişim alanında gelişmekte olan ülkelere sektördeki diğer ülkelerle birlikte teknik yardım sağlamak ve teknoloji transferi seçimi, yönetim, mali yatırım projeleri, kaynak kullanımı, araştırma-geliştirme gibi konularda yardımcı olmak, iletişim sektörü için gerekli insan ve mali yatırım kaynaklarını geliştirmek, yeni iletişim teknolojilerinin bütün insanların kullanımı için yaygınlaşmasını sağlamak amacıyla. *Politik alan*; Uluslararası düzeyde, toplumlarda iletişim sektörünün daha yaygın kullanımı için iletişim politikaları alanında teknik yardım sağlamak amacıyla.

ITU-T (ITU Telecommunication Standardization Sector) standart kümesi, Ek-1'de sunulmaktadır.

3. DÜŞÜK BİT HIZLI KONUŞMA KODLAMA ARAÇLARI

3.1. Giriş

Bir konuşma sinyalinin sıkıştırmak için, sinyalin özellikleri dikkate alınmalıdır. Düşük bit hızlı kodlama algoritmalarında, konuşma sinyali her birine ayrı bir sıkıştırma yöntemi uygulanabilecek farklı gruplara ayrılır. Bu yaklaşım bit hızını azaltmada çok etkilidir çünkü farklı gruplara kendi karakteristik özelliklerine göre davranılır.

Sesin sesli bölümleri ses yüzde kalitesine daha çok katkı sağlamaktadır. Bu yüzden sesli bölümlerin doğru kodlanması yüksek performanslı kodlayıcı elde etmede önemlidir. Sesli sesin en belirgin özelliği sesteki “perde (pitch)” olarak bilinen periyodik yapıdır. Yaklaşık bütün düşük bit hızlı konuşma kodlama algoritmaları sesli pencereleri kodlamak için bu özelliği kullanır. Bu yüzden perde belirleme işlemi konuşma kodlayıcılar için önemlidir ve perde periyodu belirlemek için birçok metot geliştirilmiştir.

Bu bölümde, konuşma kodlama teknikleri için iki temel araç olan perde belirleme ve V/U sınıflandırma incelenmiştir. Geçiş bölgeleri ve sesteki karmaşık (mixed) yapı, ikili bir V/U kararının doğruluğunu zorlaştırır. Ayrıca, çoklu bant gibi yaklaşımların performansı, düşük perdeli sinyaller için kaba (coarse) karar gibi faktörler nedeniyle sınırlanır.

İkinci bölümde, farklı perde belirleme algoritmaları incelenmiştir. Algoritmaların zaman ve frekans düzlemi gibi iki ana grubunu karşılaştıracak olursak, zaman düzlemi uygulamalar karmaşa azlığı nedeniyle düşük bit hızlı algoritmalarda geniş yer bulmaktadırlar ve genelde yüksek perde değerli (kadın) konuşmacılar için daha iyi sonuç verirler. Bu, böyle konuşmacılarda

daha çok perde çevrimi (cycle) olması nedeniyledir. Bir çevrimdeki örneklerin daha düşük sayısı nedeniyle, hesaplanan perde doğruluğu azalabilir.

3.2. Kısa Süreli Analiz

Bir ses sinyali zamanla değişir ve istatistiksel yapısı tam olarak bilinmemektedir. Ses sinyali kısa süreli analiz pencerelerine alınıp sesli ya da sessiz kısımlara ayrılarak istatistiksel yapısı ortaya çıkarılmaya çalışılır. Sesli bir ses sessiz seslere nispeten daha yüksek enerjilidir. Fakat sesli sesteki en önemli özellik, sinyaldeki periyodik yapıdır. Sessiz ses ise rasgele bir yapıda görünür. Geçiş bölgesi denen bazı ses bölgeleri hem sesli hem de sessiz parçaları birlikte içerirler ve iki durumdan birine geçişi gösterirler. Bazı kodlama algoritmalarında ses sinyalinin frekans düzlemi gösterimi kullanılır. Bu amaçla ses sinyalinin kısa süreli Fourier dönüşümü çok faydalıdır. Kısa süreli Fourier dönüşümü, ses dalga şeklinin kısa süreli değişen özelliklerini göstermede kullanılır. Pencerelemiş ses sinyalinin Fourier dönüşümü,

$$S_k(e^{j\omega}) = \sum_{n=-\infty}^{\infty} w(k-n) \cdot s(n) \cdot e^{-j\omega n} \quad (3.1)$$

olarak gösterilir. Burada $w(k)$, bir k zaman dizininde analiz edilecek ses sinyalini ayırmada kullanılan bir pencere sırası, $s(n)$ ses sinyalidir. Analiz sırasında, pencerenin şekli ve karakteristiği, ses sinyalinin frekans gösterimini etkiler.

3.2.1. Pencere etkisi

Ses zamanla değişen bir sinyaldir. Ses sinyalini etkili biçimde analiz etmek için konuşma kodlayıcı genellikle, ses sinyalini içerisinde sinyalin durağan kabul edilebileceği küçük analiz pencerelerine alır. Bu bloklar pencere olarak

isimlendirilir. Ayrıca daha yüksek zaman yoğunluğu gerektiren işlemler için *alt pencere* denen daha küçük bloklara ihtiyaç vardır.

Pencere boyu N ve öngörü derecesi p , LPC analizde hesaplamayı azaltmak için küçük tutulmalıdır. İzgel zarfın doğru tahmini için LPC analiz penceresi en azından iki perde periyodu içermelidir. Bu analiz penceresinin 20-30 msn arasında seçilmesini gerektirir. N 'nin düşük değerleri zayıf tahminlere yol açar, çünkü bazı analiz pencereleri uyartım etkileriyle bastırılır ve tüm-kutup model, bu uyartım etkileri ve ses üretim yolu arasında ayırım yapamaz. Büyük analiz penceresi boylarında, sinyalin durağan durumu artık geçerli değildir ve yine zayıf performansa yol açar. N için uygun bir değer, algoritmaya ve aynı zamanda uygun kapasiteye bağlıdır.

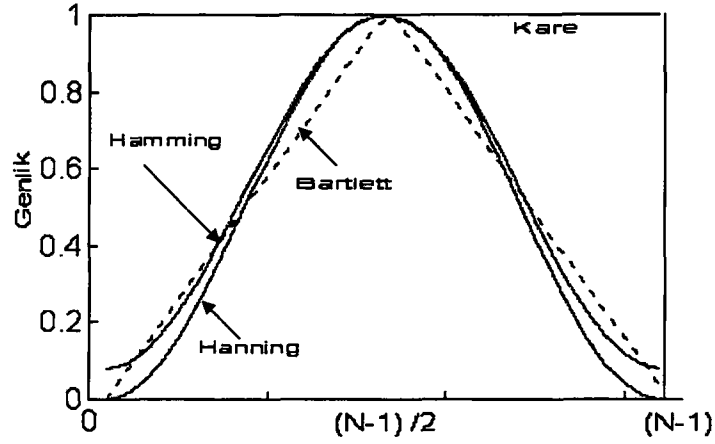
Özilişki metodunda uygulanan pencere bazı izgel distorsiyonlara sebep olur, çünkü zaman düzleminde pencere uygulanması, ses izge üretimine ve pencere frekans cevabına karşılık gelir. Pek çok pencere bir alçak geçiren frekans cevabına sahiptir ve bu yüzden pencerelenmiş ses, orijinal sesin düzleştirilmiş uyarlaması olur. Düzleştirme tipi ve miktarı pencerenin boyu ve yapısına bağlıdır. En bilinen pencereler, Kare, Bartlett, Hamming ve Hanning'tir. Şekil 3.1 ve Şekil 3.2 sırasıyla, bu pencerelerin zaman ve frekans düzleminde özelliklerini gösterir. Bu pencerelerin matematiksel özellikleri aşağıdaki gibidir:

$$W_{kare}(n) = \begin{cases} 1 & 0 \leq n < N \\ 0 & \text{diğer} \end{cases} \quad (3.2)$$

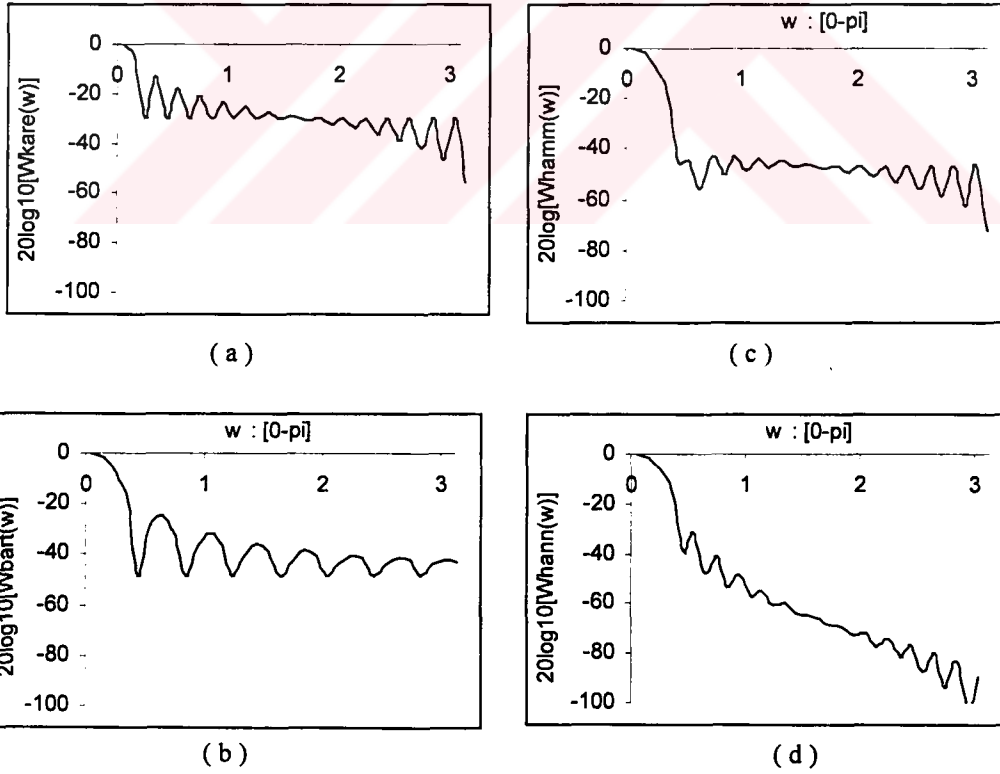
$$W_{hamm}(n) = \begin{cases} 0.54 - 0.46 \cos\left(\frac{2\pi n}{(N-1)}\right) & 0 \leq n < N \\ 0 & \text{diğer} \end{cases} \quad (3.3)$$

$$W_{hann}(n) = \begin{cases} 0.5 - 0.5 \cos\left(\frac{2\pi n}{(N-1)}\right) & 0 \leq n < N \\ 0 & \text{diğer} \end{cases} \quad (3.4)$$

$$W_{bart}(n) = \begin{cases} \frac{2n}{N-1} & ; 0 \leq n \leq \frac{N-1}{2} \\ 2 - \frac{2n}{N-1} & ; \frac{N-1}{2} \leq n \leq N-1 \\ 0 & ; \text{diğer} \end{cases} \quad (3.5)$$



Şekil 3.1. Değişik pencerelerin zaman düzlemi gösterimleri (a) Kare, b) Bartlett, c) Hamming, d) Hanning



Şekil 3.2. Değişik pencerelerin frekans düzlemi gösterimleri. (a) Kare b) Bartlett c) Hamming, d) Hanning)

Pencere işleminde ana sorun, frekans izgesinin yan bantları veya pikler sorunudur. Bunun etkisi zaman düzleminde pencere sınırları nedeniyle oluşan distorsiyonlardır. Yan bantlar sorunu çok önemlidir çünkü yüksek frekans bölgesindeki küçük parçalar pencere işlemiyle bozular. Kare pencere en dar ana *lob* nedeniyle en yüksek frekans yoğunluğuna sahiptir fakat Şekil 3.2'den de görüldüğü gibi en büyük frekans sızmasına sahiptir. Yan bantlardaki yüksek frekans sızması nedeniyle kare pencerelenmiş ses çok gürültülü görünür. Hamming penceresi kenarlarda hızlı geçişlere izin vermez ve kare pencereden daha küçük yan bantlara sahiptir. Hamming penceresi en popüler penceredir. Pencere uzunluğunun değişmesi bant genişliğini dolayısıyla ana lob'u azaltır. N için uygun bir değer 120-240 örnek arasındadır (15-30 ms).

3.2.2. Doğrusal öngörü

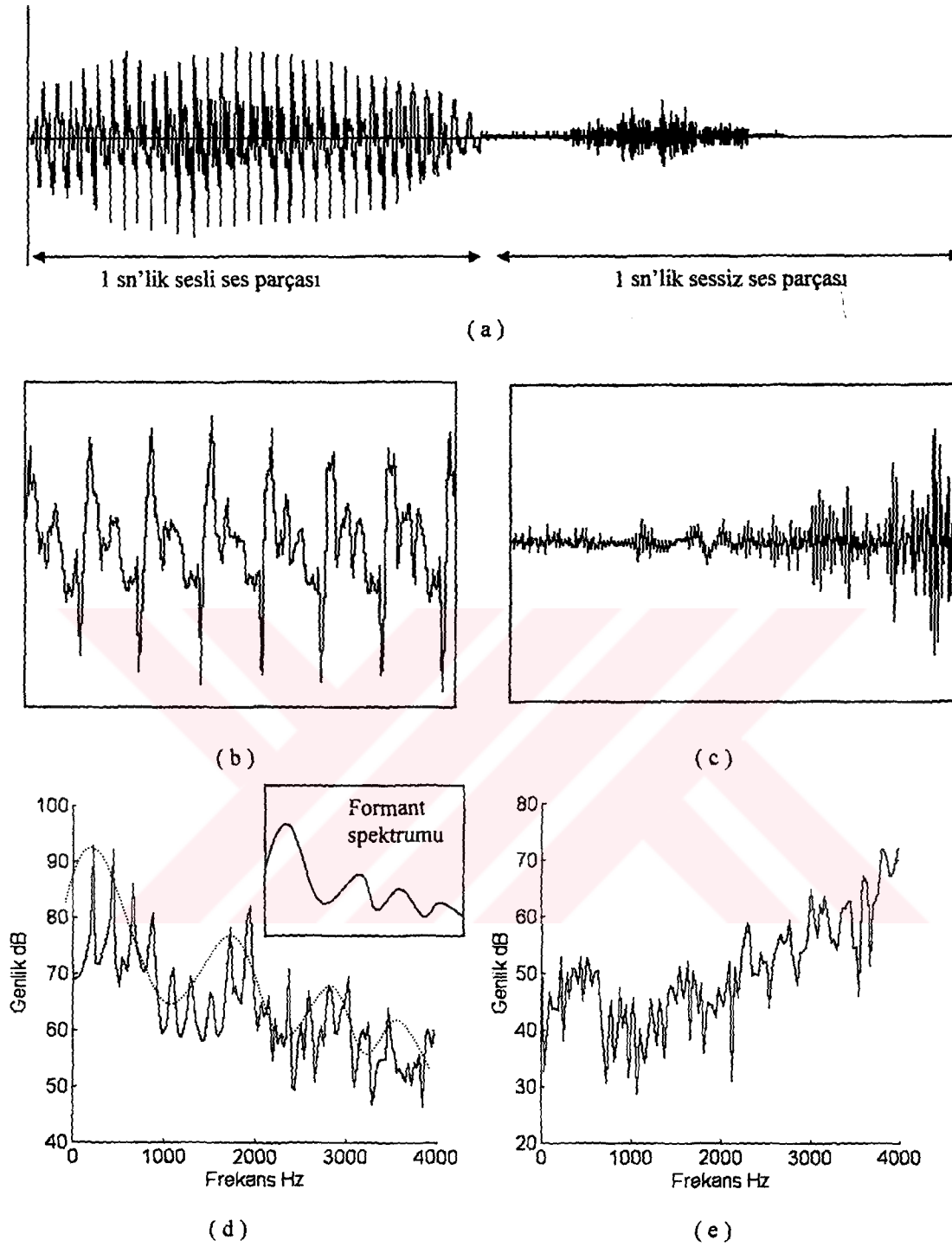
Güçlü bir ses analiz metodu olan LPC analizde, ses örnekleri arasındaki kısa süreli ilişki, ARMA (Auto Regressive Moving Average) filtre ile modellenenir. Burada ses sinyali $s(n)$, bir uyartım sinyali $u[n]$ ve p geçmiş ses örnekleri $\{s[n-1], \dots, s[n-p]\}$ ile tahmin edilir [5,49].

$$s(n) = \sum_{j=1}^p a_j s(n-j) + G \sum_{l=0}^q b_l u(n-l), \quad b_0 = 1 \quad (3.6)$$

G giriş sinyali için kazanç faktörü, a_k ve b_k filtre katsayılarıdır. Frekans düzleminde ifade edilirse:

$$H(z) = \frac{S(z)}{U(z)} = \frac{B(z)}{A(z)} = G \frac{1 + \sum_{j=1}^M b_j z^{-j}}{1 - \sum_{i=1}^N a_i z^{-i}} \quad (3.7)$$

eğer paydanın derecesi iyi ayarlanırsa sistem tüm kutup ile modellenenir. Tüm kutup sistemin eksikliği gırtlaksal (glottal) kaynak için izgel sıfırların gösterilmemesidir. Bu da nazal seslerin iyi ifade edilememesine yol açar. Genellikle izgel sıfırların izgel genliğe bir katkısı yoktur, sadece doğrusal faza



Şekil 3.3. Sesli ve sessiz ses bölümlerinin zaman ve frekans çizimleri (a) zaman düzleminde sesli ve sessiz bölümler içeren bir ses parçası (b) 150 msn'den itibaren 32 msn'lik sesli ses parçasının zaman düzlemi gösterimi (c) 1300 msn'den itibaren 32 msn'lik sessiz ses parçasının zaman düzlemi gösterimi (d) 150 msn'den itibaren 32 msn'lik sesli ses parçasının güç spektrumu (e) 1300 msn'den itibaren 32 msn'lik sessiz ses parçasının güç spektrumu

etki ederler, çünkü basit bir zaman gecikmesinden doğarlardı. Genellikle $p=10$ seçilerek izgel benzetme yapılır.

$$H(z) = \frac{G}{1 - \sum_{j=1}^p a_j z^{-j}} = \frac{G}{A(z)} \quad (3.8)$$

Bu şekilde izgel örüntü $1/A(z)$ ile modellenir. Ses üretimi zaman düzleminde ifade edilirse,

$$s(n) = Gu(n) + \sum_{j=1}^p a_j s(n-j) \quad (3.9)$$

Çıkış $s(n)$, ağırlıklı giriş $Gu(n)$ ve geçmiş örneklerin ağırlıklı toplamı ile bulunur. Burada a_j , ($j=1, \dots, p$) orijinal değerler ve α_j , a_j 'nin tahminlerini gösterirse, giriş ve çıkış sinyali arasındaki hata sinyali,

$$e(n) = s(n) - \sum_{j=1}^p \alpha_j s(n-j) \quad (3.10)$$

LPC analiz hatasız ise, α_j tahmin parametreleri a_j parametrelerine eşittir. O zaman, $e[n] = Gu[n]$ artık sinyal uyarım sinyalidir. $e[n]$ 'i bulmak için bir ters filtre $H^{-1}(z)$ oluşturulur.

$$H^{-1}(z) = 1 - \sum_{j=1}^p \alpha_j z^{-j} \quad (3.11)$$

Sesli ve sessiz bölümleri içeren bir orijinal giriş sinyali, ve bu sinyallerin kısa süreli analiz ile zaman düzlemi gösterimleri ve güç spektrumları şekil 3.3'de verilmiştir.

$s(z) = H(z).Gu(z)$ buradan $u(z) = (1/G).H^{-1}(z).s(z)$ dir. Bu artık sinyal perde periyodunu bulmada çok faydalıdır. Çünkü, ters filtre ile sinyaldeki kısa süreli ilişki kaldırılır.

3.2.2.1. Özilişki metodu

Kısa analiz pencerelerine (20 ms civarı) alınan ses sinyali örneklerinden tahmin katsayıları bulunmaya çalışılır. Bu nedenle tahmin katsayılarını bulmada pencereleme ilk basamaktır ve iyi bir analiz penceresi seçmek önemlidir. Özilişki (Autocorrelation) metodunda pencereleme ses sinyaline uygulandığı halde, ortak varyans metodu için pencereleme artık (residual) sinyale uygulanır. Özilişki metodunda ses sinyali $s[n]$ ilk olarak bir analiz penceresi $w[n]$ ile çarpılır ve $s_w[n]$ sinyali elde edilir. $s_w[n]$ sinyali için $0 \leq m \leq N-1$ analiz penceresi aralığı dışındaki değerler sıfır sayılır. Burada N , pencere içindeki örneklerin sayısıdır. Bu yüzden, $N \leq m \leq N+p$ aralığı için sıfır örnekler tahmin etmeye çalıştığımız için (gerçekte sıfır olmayan), bu örnekler için tahmin hatası sıfır olmayacaktır. Benzer şekilde, o andaki pencere başlangıcı önceki pencerede oluşan bu hatadan etkilenecektir.

$$s_w[n] = w[n].x[n] \quad (3.12)$$

Bir Hamming penceresi analiz için kullanılırsa, analiz penceresi dışındaki değerlerin sıfır sayılma etkisini, $s_w[n]$ 'in ilk ve son birkaç değerindeki kareli tahmin hatasını minimumlaştırarak azaltmaya çalışır.

Doğrusal öngörü katsayılarını çözmek için, hata sinyalinin enerjisi minimum yapılmaya çalışılır. E hata enerjisi olsun.

$$E = \sum_{n=-\infty}^{\infty} e^2[n] = \sum_{n=-\infty}^{\infty} \left[s_w(n) - \sum_{j=1}^p \alpha_j s_w(n-j) \right]^2 \quad (3.13)$$

$e[n]$, $s_w[n]$ pencerelenmiş giriş sinyali için hata sinyalidir. E yi minimumlaştıran a_k değerleri $\frac{\partial E}{\partial a_k} = 0$, $k = 1, 2, 3, \dots, p$ yaparak bulunur. Bu işlem

ile Yule-Walker eşitlikleri denen p bilinmeyenli p doğrusal eşitlik ortaya çıkar:

$$\sum_{k=1}^p a_k \sum_{n=-\infty}^{\infty} s_w[n-i] s_w[n-k] = \sum_{n=-\infty}^{\infty} s_w[n-i] s_w[n], \quad 1 \leq i \leq p \quad (3.14)$$

Pencerelemiş ses parçası $s_w[n]$ için özilişki aşağıdaki gibi verilir:

$$R(i) = \sum_{n=0}^{N-1} s_w(n) \cdot s_w(n-i), \quad 0 \leq i \leq p \quad (3.15)$$

burada özilişki fonksiyonu bir $R(i) = R(-i)$ şeklinde bir çift fonksiyondur. Lineer eşitlikler aşağıdaki gibi yazılabilir:

$$\sum_{k=1}^p R(|i-k|) \cdot a_k = R(i), \quad 1 \leq i \leq p \quad (3.16)$$

Matris şekilde, lineer eşitlikler kümesi $R \cdot a = v$ ile gösterilir ve aşağıdaki formda açık olarak ifade edilebilir:

$$\begin{bmatrix} R(0) & R(1) & \dots & R(p-1) \\ R(1) & R(2) & \dots & R(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ R(p-1) & \dots & \dots & R(0) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(p) \end{bmatrix} \quad (3.17)$$

yukarıdaki matris simetriktir ve alt ve üst üçgen matris elemanları eşittir, yani Toeplitz matristir. Toeplitz yapı $A(z)$ 'nin minimum faz olmasını (sıfırlar birim çember içinde) garantiler. Böylece karşılık gelen doğrusal öngörü sentez filtresi $H(z) = 1/A(z)$ kararlıdır. Bu özellik doğrusal öngörü analiz için özilişki metodunun seçilmesinin ana nedenidir. Bu özellikler kullanılarak matris çözülür ve α katsayıları bulunur.

3.2.2.2. Ortak varyans metodu

Ortak varyans metodunda pencerelemiş hatada minimumlaştırma gerçekleştirilir.

$$E = \sum_{n=-\infty}^{\infty} e_w^2[n] = \sum_{n=-\infty}^{\infty} e^2[n] w[n] \quad (3.18)$$

$\frac{\partial E}{\partial a_k} = 0$, $k = 1, 2, 3, \dots, p$ ile kısmi türevleri sıfıra eşitleyerek p adet doğrusal

eşitliğe sahip oluruz.

$$\sum_{k=1}^p \phi(i, k) a_k = \phi(i, 0), \quad 1 \leq i \leq p \quad (3.19)$$

burada ortak varyans fonksiyonu $\phi(i, k)$ aşağıdaki gibi belirlenir:

$$\phi(i, k) = \sum_{n=-\infty}^{\infty} w[n] s[n-i] s[n-k] \quad (3.20)$$

Açılmış ortak varyans matris sistemi $\phi a = \psi$ aşağıdaki gibidir:

$$\begin{bmatrix} \phi(1,1) & \phi(1,2) & \dots & \phi(1,p) \\ \phi(2,1) & \phi(2,2) & \dots & \phi(2,p) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(p,1) & \phi(p,2) & \dots & \phi(p,p) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} \psi(1) \\ \psi(2) \\ \vdots \\ \psi(p) \end{bmatrix} \quad (3.21)$$

burada $\psi(i) = \phi(i, 0)$, $i = 1, 2, \dots, p$ dir.

Ortak varyans matrisi simetri özelliğine sahiptir ve pozitif belirlidir. Fakat Toeplitz değildir bu yüzden özilişki metoduna göre daha az etkinliği vardır. Genellikle a_k 'ları çözmek için ilk basamak olarak Cholesky decomposition [49] kullanılır.

Ortak varyans metodu sentez filtresi $A(z)$ 'nin minimum faz olmasını garantilemediği için, bir değiştirilmiş ortak varyans metodu $A(z)$ kararlılığı için kullanılır [49].

Doğrusal öngörü analiz alçak geçiren filtreli sinyale uygulandığı zaman, yarı örnekleme frekansı yanında kaçırılan parçalar sonuç doğrusal öngörü katsayılarında önemli miktarda yanlışlık yaratabilir. Bu durumda ortak varyans matrisi ϕ gerçek değerden büyük doğrusal öngörü katsayıları verir. Bu yüzden, yüksek frekans uyumlandırma gereklidir.

3.2.2.3. Bant genişliği açma

Doğrusal öngörü analiz, yüksek perdeli seslerde izgel zarfı doğru olarak veremez. Keskin pik değerli izgel zarflar oluşur. Bu problemi aşmak için bant genişliği açma katsayısı kullanılır. Frekans cevabındaki formant pikleri, bant genişliklerini açma etkisine sahiptir. Bant genişlik açma katsayısı γ ile sonuçtaki tüm kutup sentez filtresi aşağıdaki gibidir:

$$H'(z) = \frac{1}{A'(z)} = \frac{1}{A(\gamma z)} \quad (3.22)$$

burada açılmış tahmin katsayıları:

$$a'_k(k) = a_k \gamma^k, \quad 1 \leq k \leq p \quad (3.23)$$

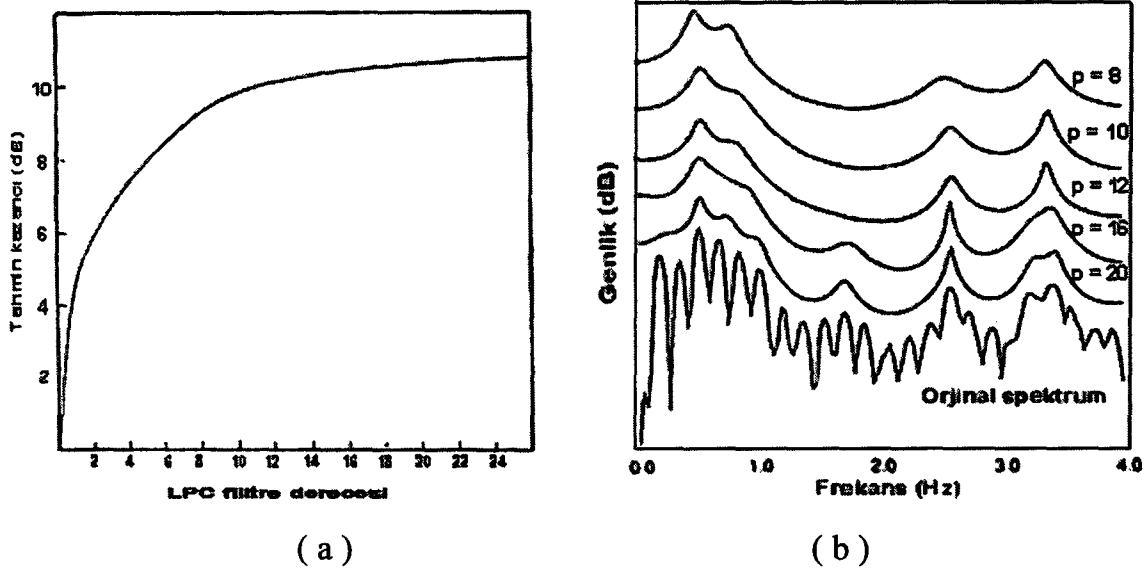
bant genişliği açılımı aşağıdaki gibi hesaplanabilir:

$$\Delta B = -\frac{1}{\pi T} \log(\gamma) \quad (3.24)$$

Örneğin $\gamma = 0.996$, 8 kHz'de örneklenmiş ses analizinin bant genişliği açılımı olarak 10 Hz civarı bir değer verir. Ses analizi için tipik olarak 10-25Hz civarı bant genişliği açılımı kullanılmaktadır.

3.2.2.4 Öngörü derecesi

LPC tabanlı sistemlerin dizaynında başka bir önemli parametre, öngörü derecesidir. Şekil 3.4.a'dan görülmektedir ki, daha iyi öngörü için daha yüksek dereceler gerekmektedir. Performans $p=8$ ve $p=16$ civarında doyuma ulaşır, ve bu yüzden pek çok algoritmada LPC derecesi $p=10$ civarında seçilir. Ses üretim yolu 3 ile 5 arası güçlü rezonans frekansı çıkarmakta ve bunlar LPC derecesi 10 seçilerek modellenenabilir. Fakat pek çok algoritmada bu değer bağımsızdır. LD-CELP 50'nci dereceden öngörü kullanırken APC (Adaptive Predictive coding) sistemler 2'inci derece doğrusal öngörü kullanır.



Şekil 3.4. a) LPC filtre derecesine karşı LPC tahmin kazancı b) LPC derecesine karşı LPC zarf çizimleri

3.3 Distorsiyon ölçümleri

Distorsiyon ölçümleri konuşma kodlamada önemli bir role sahiptir. Distorsiyon ölçümleri ile konuşma kodlama sistemlerinin performansı ölçülmeye çalışılır. Konuşma kodlayıcının kalite, performans, anlaşılabilirlik ve doğrallığının son aşaması insan konuşma sesidir. Objektif ölçümler kodlama algoritmalarının yüzde kalite tahminlerinde hızlı ve güvenilir sonuçlar verirler. Ayrıca objektif ölçümler doğrusal öngörülü kodlayıcılarda katsayıların vektörel nicemlenmesinde önemli bir rol oynarlar. Bu katsayılar kısa süreli ses analizinde izgel zarfı tanımladığından, vektör nicemleme çizelgesinden bir vektör için en iyi izgel zarfı seçmede kullanılırlar. Orijinal ve seçilen vektör arasındaki yüzde benzerlik objektif distorsiyon ölçümleriyle yapılır. Objektif ölçümler kazanç değişimleri ve gecikmelere duyarlıdır. Daha önemlisi, kulağın yüzde özelliklerini dikkate almazlar. [29]. Bu yüzden, fonetik olarak dengelenmiş ses kayıtları ile subjektif hesaplamalar kullanılır. Bunlardan bazıları, Diagnostic Rhyme Test (DRT) [30], Diagnostic Acceptability Measure (DAM) [39] ve Mean Opinion Score (MOS) [32] testleridir.

3.3.1. Zaman düzlemi objektif distorsiyon ölçümleri

SNR (Signal to Noise Ratio) veya kısmi SNR (segmental SNR), orijinal ve kodlanmış ses sinyali arasındaki yüzde farkı gösteren en yaygın metottur.

3.3.1.1. SNR

SNR, giriş sinyal gücü ve gürültü gücü arasındaki oran olarak tanımlanabilir ve dB cinsinden aşağıdaki gibi verilir:

$$SNR = 10 \log_{10} \left[\frac{\sum_{n=0}^M s^2(n)}{\sum_{n=0}^M (s(n) - \hat{s}(n))^2} \right] \quad (3.25)$$

burada $s(n)$ orijinal ses, $\hat{s}(n)$ kodlanmış sesi gösterir. M sinyaldeki örnek sayısıdır. SNR kalite ölçümü matematiksel olarak basittir. Yine de SNR, tekrar oluşan sesin doğruluğunu ölçmede uzun bir metottur ve özellikle düşük seviyeli sinyaller için geçici gürültüleri farketmek zordur. Performanstaki geçici etkiler, kısmi SNR ile daha kolay fark edilir. (Segmental SNR- SNR_{seg}).

3.3.1.2. Kısmi SNR

Eğer SNR ölçümleri kısa süreli analiz pencereleri içinde yapılır ve sonuçların ortalaması alınırsa, daha iyi bir kalite ölçümü yapılır. SNR_{seg} aşağıdaki gibi gösterilebilir. N , analiz penceresi içindeki örneklerin sayısıdır. L , toplam analiz penceresi sayısıdır.

$$SNR_{seg} = \frac{1}{L} \sum_{i=0}^{L-1} 10 \log_{10} \left[\frac{\sum_{n=0}^N s^2(iN + n)}{\sum_{n=0}^N (s(iN + n) - \hat{s}(iN + n))^2} \right] \quad (3.26)$$

Bazı durumlarda özellikle sessiz pencerelerde, SNR_{seg} ölçümleri sorun çıkarabilir. Çünkü çok fazla negatif SNR değerleri bütün SNR ölçümünde yanlılık (bias) yapabilir. Büyük ya da küçük SNR (0 dB altı ya da 35 dB üstü) içeren herhangi bir pencereyi çıkarmak için bir eşik değer kullanılabilir. SNR_{seg} 'ın bir benzeri frekans ağırlıklı SNR_{seg} ölçümüdür.

$$SNR_{fw} = \frac{1}{M} \sum_{j=0}^{M-1} \left[\frac{\sum_i W(j,i) 10 \log_{10} \left[\frac{s^2[n]}{(s[n] - \hat{s}[n])^2} \right]}{\sum_i W(j,i)} \right] \text{dB} \quad (3.27)$$

burada j segment indeksi, i indeks frekans bandı ve $W(j,i)$ frekans ağırlığıdır.

3.3.2. Frekans düzlemi objektif distorsiyon ölçümleri

Bir izgel zarf distorsiyon ölçümü f ve $\hat{f}$ olarak iki yoğunluk fonksiyonuna sahiptir, örneğin, bir negatif olmayan $d(f, \hat{f})$ f 'yi göstermek için kullanılan $\hat{f}$ 'daki distorsiyonu bulmak için kullanılır. Eğer $d(f, \hat{f})=0$ ise $f = \hat{f}$ dir. Bu şekildeki en yaygın ölçümler $f - \hat{f}$ farkının kullanıldığı fark distorsiyon ölçümleridir. Bunlar metrik mesafeler olduğundan $d(f, \hat{f})=d(\hat{f}, f)$ şeklinde simetri özelliği taşımaktadır. Genellikle izgel zarf distorsiyon ölçümünde, f bir x vektörü ve $\hat{x}$, x 'in bir gösterimi olarak belirlenir. O zaman tüm performans ölçümü aşağıdaki gibi ifade edilebilir:

$$D = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n d(x_i, \hat{x}_i) \quad (3.28)$$

Genellikle, izgel distorsiyon orijinal ve kodlanmış izgel zarf arasındaki seste fonetik olarak farklılıklara neden olan uyumsuzlukları ölçer. Bu uyumsuzluklar aşağıdaki gibi sıralanabilir:

- Orijinal ve kodlanmış izgel zarf arasındaki önemli derecede farklı formant ya da rezonans frekansları
- Kodlanmış izgel zarflar nedeniyle formant bant genişliklerinin değişimi

Logaritmik izgel distorsiyon ölçümü, Itakura-saito ölçümü, cepstral mesafe ve ağırlıklı öklid mesafe ölçümü önerilen bazı farklı mesafe ölçümleridir ve aşağıda bahsedilmiştir:

3.3.2.1. Logaritmik İzgel Distorsiyon Ölçümü

L_p normu logaritmik izgel mesafe ölçümü,

$$d_{SD}^p = \frac{2}{2\pi} \int_{-\pi}^{\pi} |10 \log_{10} S(w) - 10 \log_{10} \hat{S}(w)|^p dw \quad (3.29)$$

burada $S(w)$ frekans genlik spektrumudur,

$$S(w) = \frac{G}{|A(e^{jw})|^2} = \frac{G}{\left|1 - \sum_{n=1}^p a_n e^{jnw}\right|^2} \quad (3.30)$$

G doğrusal öngörü filtresi kazanç faktörü, ve $\{a_n\}$ doğrusal öngörü katsayılarıdır.

$p = 2$ ise, L_2 normu ya da rms (root mean square) logaritmik izgel distorsiyon ölçümü yapılır. Rms logaritmik izgel distorsiyon ölçümü,

$$d_{SD} = \sqrt{\frac{1}{w_u - w_l} \int_{w_l}^{w_u} \left[10 \log_{10} \frac{S(w)}{\hat{S}(w)}\right]^2 dw} \quad dB \quad (3.31)$$

burada w_l ve w_u integraldeki alt ve üst frekans limitlerini tanımlar. İdeal olarak w_l sıfıra eşittir ve w_u örnekleme frekansının yarısına karşılık gelir.

Pratikte *rms* logaritmik izgel mesafe sınırlı bir bant genişliğinde kesikli olarak hesaplanır. 8 kHz'de örneklenmiş ve 3 kHz bant genişlikli bir ses sinyali için, yaklaşık olarak her örnek 31.25 Hz, 125 Hz'den 3125 Hz'e 96 düzgün dağılmış nokta ile rms logaritmik izgel distorsiyon (SD) genellikle bir toplam olarak hesaplanır. Bu aşağıdaki şekilde ifade edilebilir:

$$SD = \sqrt{\frac{1}{n_1 - n_0} \sum_{n=n_0}^{n_1-1} \left[10 \log_{10} \frac{S(e^{j2\pi n/N})}{\hat{S}(e^{j2\pi n/N})} \right]^2} \quad dB \quad (3.32)$$

burada $N=256$, n_0 ve n_1 sırasıyla 1 ve 96 değerlerine karşılık gelir. Ortalama izgel distorsiyon yüzde kalite için tek başına ölçüt değildir. Aykırı değer sayısı da bilinmelidir. İyi bir kalite için 2 dB ve 4 dB arası izgel distorsiyon veren aykırı değer sayısı %2'den az olmalı ve SD 4 dB'den büyük distorsiyon veren aykırı değer olmamalıdır.

3.3.2.2. Itakura-Saito distorsiyon ölçümü

Itakura-saito distorsiyon ölçümü genellikle sesin yüzde kalitesinde daha iyi sonuç verir. Aynı zamanda likelihood oranı mesafe ölçümü olarak bilinir, nicemlenmiş LP filtre sonucu artık sinyal ile nicemlenmemiş LP filtre artık sinyal arasındaki enerji oranıdır. Aşağıdaki gibi belirlenir:

$$d_{IS} = \frac{1}{2\pi} \int_{-\pi}^{\pi} [e^{V(w)} - V(w) - 1] dw \quad (3.33)$$

burada iki spektrum arasındaki log izgel mesafe $V(w)$,

$$V(w) = \log S(w) - \log \hat{S}(w) \quad (3.34)$$

İntegralleri hesaplayarak, bu ölçüm polinom olarak açıklanabilir

$$d_{IS} = \left(\frac{G}{\hat{G}} \right)^2 \frac{\hat{a}^T R \hat{a}}{a^T R a} - 2 \log \left(\frac{G}{\hat{G}} \right) - 1 \quad (3.35)$$

burada $\hat{a} = [1, \hat{a}_1, \hat{a}_2, \dots, \hat{a}_p]^T$, $a = [1, a_1, a_2, \dots, a_p]^T$, ve R özilişki matrisidir. Kazançlar eşit sayılırsa, Itakura-Saito ölçümü basitce,

$$d_{IS} = \frac{\hat{a}^T R \hat{a}}{a^T R a} - 1 \quad (3.36)$$

Fakat, Itakura-Saito matrisi simetrik değildir. Simetri için değiştirilmiş bir Itakura-Saito ölçümü kullanılır:

$$d_{IS} = \frac{1}{2} \left[\frac{\hat{a}^T R \hat{a}}{a^T R a} - \frac{a^T R \hat{a}}{\hat{a}^T R \hat{a}} - 2 \right] \quad (3.37)$$

İnsan kulağının yüzde ayırım özelliği avantajını kullanmak için Itakura-Saito ölçümüne bir ağırlık terimi katılabilir:

$$d_{WIS} = \frac{1}{2\pi} \int_{-\pi}^{\pi} W(e^{jw}) [e^{V(w)} - V(w) - 1] dw \quad (3.38)$$

3.3.2.3. Ağırlıklı Euclidean LSF mesafe Ölçümü

İki vektör arasında Euclidean mesafe ölçümü basitçe aşağıda verilmektedir:

$$d(x, \hat{x}) = (x - \hat{x})^T (x - \hat{x}) = \|x - \hat{x}\|^2 \quad (3.39)$$

Genellikle nicemlenmiş ve nicemlenmemiş vektör arasındaki ortalama kareli hatayı minimum yapmak, en iyi kod vektörü seçmektir. LSF'ler izgel zarf yapısı ile direkt bağlantılı olduğundan LSF ölçümü bu ölçümle özdeşleştirilir. Euclidean mesafe ölçümü LSF vektörünün bağımsız parçaları için eşit ağırlıklar işaretler. İzgel duyarlılıklar bir ağırlıklı Euclidean mesafesi ile belirlenir. Genellikle Euclidean LSF mesafesi :

$$d(x, \hat{x}) = (x - \hat{x})^T W (x - \hat{x}) \quad (3.40)$$

burada W ($m \times m$) simetrik ve x 'e bağılı olabilen pozitif belirli bir ağırlık matrisidir. Eğer W bir $w_{ii} > 0$ olacak şekilde köşegen matris ise, mesafe aşağıdaki gibidir:

$$d(x, \hat{x}) = \sum_{i=1}^m w_{ii} (x_i - \hat{x}_i)^2 \quad (3.41)$$

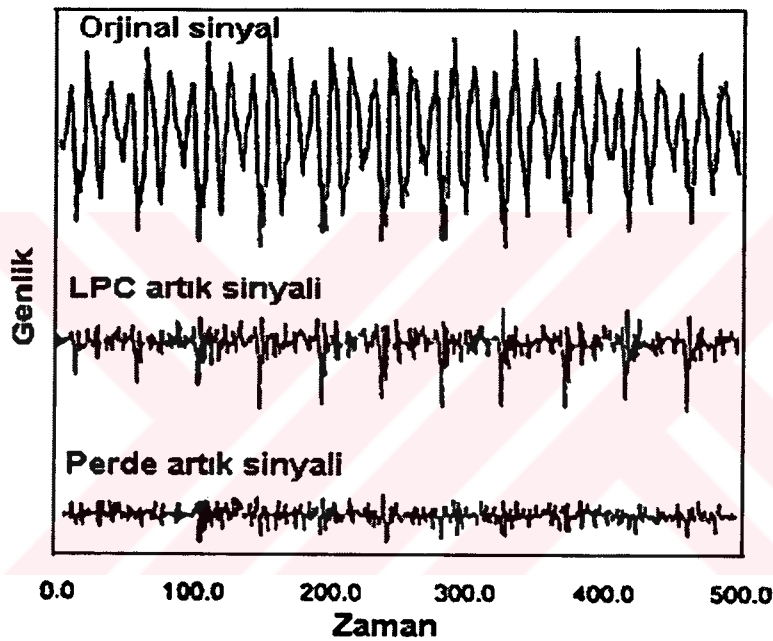
3.4. Perde Peryodu Belirleme

Peryodik yapı sesin sesli bölümlerinde en bilinen özelliktir. Bu özellik, perde peryodu (pitch period) olarak bilinir. Hemen bütün düşük bit hızlı konuşma kodlama teknikleri sesli sesi etkin bir biçimde kodlayabilmek için bu parametreyi kullanırlar. Sonuç olarak, perde belirleme konusunda düşük bit hızlı kodlayıcılarda birçok araştırmalar mevcuttur. Bu çalışmalarda, sesli sesin kısa bölümü yani pencere uzunluğu içinde perde peryodu sabittir ve bu değer bulunmaya çalışılır.

Perde peryodu belirlemede birçok algoritma olmasına rağmen, bu metotların hiçbirisi mükemmel çalışmaz ve bütün ses malzemeleri için kullanılabilecek bir perde peryodu belirleme algoritması henüz çözülememiştir. Bu esas olarak kısa süreli analiz penceresi içinde sabit kabul edilen ses sinyalinin dinamik yapısından kaynaklanır. Ayrıca, bu metotlar tam olarak peryodik olmayan (quasiperiodic) ses sinyali için perde peryodu bulmaya çalışırlar. Sonuçta hesaplanan perde peryodunda zaman zaman hatalar oluşur. Ama yine de, perde peryodu belirleme, düşük bit hızlı konuşma kodlayıcılarda önemli bir parametre olduğundan, bu metotlar ses sıkıştırma algoritmalarında geniş yer bulurlar. Bu algoritmalarda, giriş ses sinyali perde peryodu belirleme işleminden önce alçak geçiren filtreden geçirilir, Bu ön-işleme tekniği insan sesinin perde frekansının sınırlı (50 ve 400 Hz arasında) kabul edilmesi

nedeniyledir. Bu nedenle alçak geçiren filtre için 800 Hz civarı bir kesim frekansı uygundur.

Giriş ses sinyali, LPC filtreden geçirildiğinde, ses sinyalinin uzun süreli ilişkisi daha net biçimde görülebilir (Şekil 3.5). Perde belirleme metotları, zaman düzlemi ve frekans düzlemi olarak iki grupta toplanabilir. [14,50,51]



Şekil 3.5. Orijinal sinyal, LPC ve artık perde zaman düzlemi çizimleri

3.4.1. Zaman düzlemi algoritmalar

Zaman düzleminde perde belirleme tekniklerinde, sesli sesin bir parçası kaymış versiyonları ile karşılaştırılır. Bu metotlarda, 2 sinyal arasında en yüksek benzerlikle sonuçlanan kayma, sesin o bölümü için perde periyodu olarak açıklanır. Ters durum olarak, hata fonksiyonu iki sinyalin benzemeyişi konusunda bir gösterge olabilir. Bu durumda, minimum hatayı veren kayma,

perde peryodu olarak düşünülür. Özilişki ve ortalama genlik fark fonksiyonu (Average Magnitude difference Function-AMDF), perde peryodunu zaman düzleminde belirlemek için en bilinen yöntemlerdir.

3.4.1.1. Özilişki metodu

Kaymayı bulmak için en yüksek benzerliği bulmayı amaçlayan en popüler metot, iki sinyal arasındaki direkt mesafenin minimumlaştırılmasıdır. Bu mesafe aşağıdaki gibi açıklanabilir:

$$E(\tau) = \frac{1}{N} \sum_{n=0}^{N-1} [s(n) - s(n + \tau)]^2 \quad (3.42)$$

s ses sinyali, τ kayma (gecikme olarak da adlandırılır) ve N pencere uzunluğudur. Ses sinyalinin kısa zaman peryodu içinde sabit kabul ederek, yukarıdaki eşitlik aşağıdaki şekilde yazılarak özilişki (Autocorrelation-AC) fonksiyonunu ifade eder. $R(0)$ değeri sabit olduğundan, hata fonksiyonunun minimumlaştırılması AC fonksiyonunun maksimumlaştırılmasına eşittir. Böylece en yüksek korelasyonla sonuçlanan kayma parametresi τ perde peryodu olarak seçilir. Şekil 3.6, sesli sesin bir parçasını ve ilgili AC değerini gösterir.

$$E(\tau) = 2[R(0) - R(\tau)] \quad (3.43)$$

$$R(\tau) = \frac{1}{N} \sum_{n=0}^{N-1} s(n)s(n + \tau) \quad (3.44)$$

Yukarıda tarif edilen AC metodu, önemli bir problem olan perde çarpımlarına karşı hassastır. Bu durum özellikle sesli sesteki ani volume değişmesinden kaynaklanır. Bitişik çevrimlerdeki (cycle) değişim şekilden görülmektedir. AC fonksiyonu görür ki, fonksiyondaki maksimum gerçek perdenin üç katında oluşur. Bu yüzden yanlış perde peryodunu seçer. Bu problemi aşmak için,

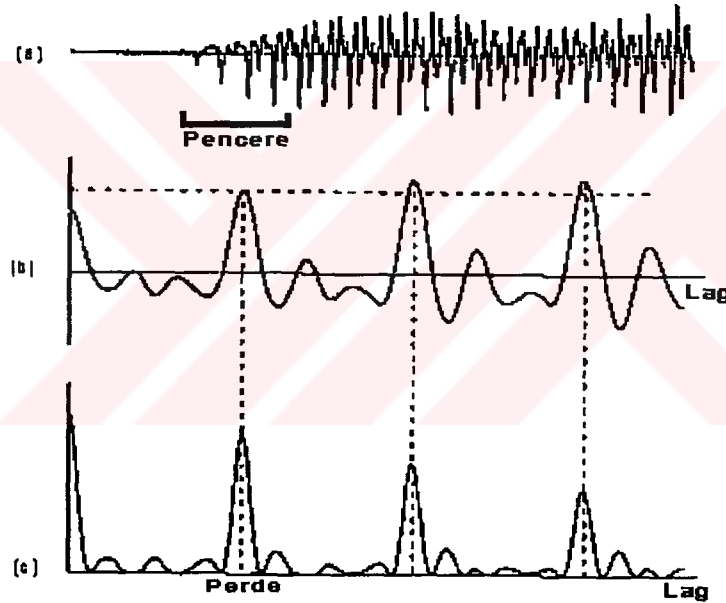
enerji değişimi yukarıdaki formüllerde dikkate alınmalıdır. Bu hata fonksiyonunda bir ağırlık faktörünü dikkate alarak gerçekleştirilir.

$$E(\tau) = \frac{1}{N} \sum_{n=0}^{N-1} [s(n) - \beta s(n + \tau)]^2 \quad (3.45)$$

β bir kazanç faktörüdür. $\partial E(\tau, \beta) / \partial \beta = 0$, aşağıdaki β değerini verir.

$$\beta = \frac{\sum_{n=0}^{N-1} s(n)s(n + \tau)}{\sum_{n=0}^{N-1} s^2(n + \tau)} \quad (3.46)$$

Bu hata değeri yukarıda yerine konarak yeni bir formül oluşturulur.



Şekil 3.6. Sesin özilişkisi. a) orijinal sinyal b) direkt c) normalize edilmiş özilişki fonksiyonları

$$E(\tau, \beta^*) = \sum_{n=0}^{N-1} s^2(n) - \frac{[\sum_{n=0}^{N-1} s(n)s(n + \tau)]^2}{\sum_{n=0}^{N-1} s^2(n + \tau)} \quad (3.47)$$

yukarıdaki eşitlikteki ilk terim, pencere enerjisini gösterir ve bu yüzden τ ile değişmez. Sonuç olarak hatayı minimize etmek için, normalize edilmiş AC fonksiyonu olarak da bilinen ikinci terimi maksimum yapmak yeterlidir:

$$R_n^2(\tau) = \frac{[\sum_{n=0}^{N-1} s(n)s(n+\tau)]^2}{\sum_{n=0}^{N-1} s^2(n+\tau)} \Rightarrow R_n(\tau) = \frac{[\sum_{n=0}^{N-1} s(n)s(n+\tau)]}{\sqrt{\sum_{n=0}^{N-1} s^2(n+\tau)}} \quad (3.48)$$

$R_n^2(\tau)$ terimi pay ve paydadaki kareler nedeniyle fonksiyon negatif ilişki verir. Sonuç olarak perde peryodu maksimum negatif ilişki olarak seçilebilir. Bu genellikle perde yarılama (halving) hatası verir. Bunu önlemek için, yukarıdaki eşitliğin karekökünü maksimum yapmak tercih edilebilir. Bu metot, doğru perde belirlemede kullanılabilir.

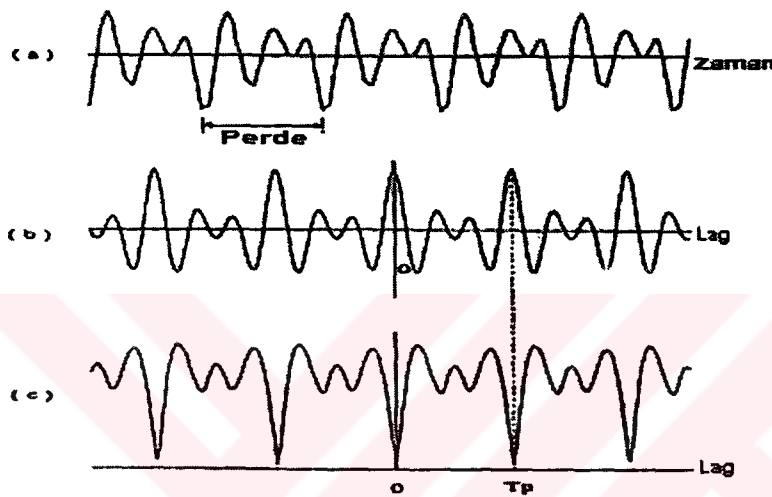
3.4.1.2. AMDF metodu

AC metodu, perde peryodunu doğru belirlemede etkilidir. Yine de, fazla hesaplama gerektirir. Az hesaplama istenen kodlayıcılarda AMDF (Average Magnitude Difference Method) metodu tercih edilebilir. Karmaşayı azaltmak için bu metod başka bir hata fonksiyonu sunar:

$$E(\tau) = \frac{1}{N} \sum_{n=0}^{N-1} [s(n) - s(n+\tau)] \quad (3.49)$$

Bu metodun başka bir avantajı, hata fonksiyonundaki daha düşük dinamik değişimdir. Bu özellikle fixed point DSP çiplerinde gerçek uygulamalarda önemlidir. Şekil 3.7, Şekil 3.6'ya AMDF tekniğinin uygulanmasını gösterir. Seçilen perde Şekil 3.7'de gösterilir.

AMDF tekniđi basit yapısı ile iyi sonuçlar sağlar. Amerikan hükümeti standardı LPC10 konuşma kodlayıcıda kullanılmıştır. Yine de, 1980'li yılların ortalarında çıkmaya başlayan gelişmiş DSP çipleri sayesinde, AMDF basitliđi artık büyük bir avantaj değildir. Ayrıca AC tekniđi gibi AMDF tekniđi de, sesli penceredeki perde ve/veya genliđin ani deđişimlerinde doğru olmayan sonuçlar verebilir.



Şekil 3.7. a) Orijinal sesli ses sinyali b) özilişki fonksiyonu c) AMDF

3.4.2. Frekans düzlemi algoritmalar

Frekans düzleminde perde frekansı (perde peryodunun tersi) sesli sesin analizi ile bulunur. Frekans düzleminde sesli sesin ana özelliđi, harmonik yapısı yani eşit frekans aralıklarında olan genlik spektrumudur. Bu bütün frekans düzlemi perde belirleme algoritmaları için temel özelliktir. Bazı teknikler aşağıda görölmektedir.

3.4.2.1. Harmonik benzerlik metodu

Frekans spektrumundan perde belirlemenin en doğru yolu, ilk harmoniđi tespit etmektir. Bu en düşük frekans pikini bularak başarılır. Fakat, ilk harmonik piki

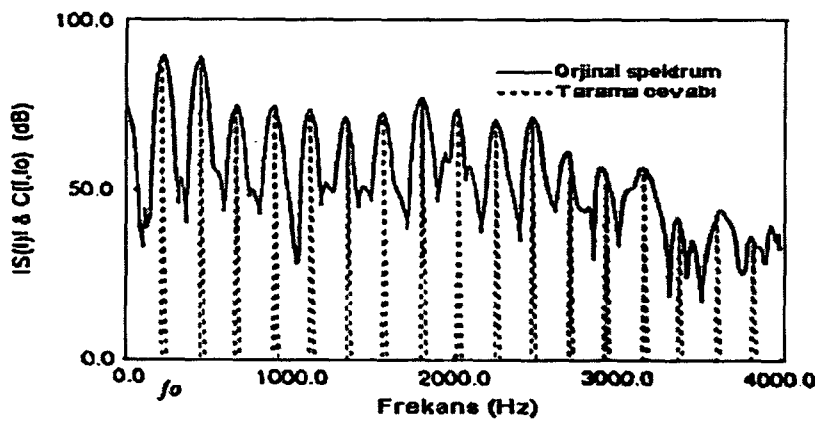
genelde zayıftır ve spektrumda görülmeyebilir. Daha güvenli bir metot, bütün pik frekanslarını tespit etmek ve bu frekanslar arasında mesafeden perde frekansını (pitch frequency – fundamental frequency) bulmaktır. Harmonikler arasındaki mesafe eşit deneme frekanslarındaki genlik spektrum örneklerini toplayarak ve en yüksek toplamı veren frekansı perde frekansı olarak seçerek bulunur. Bu metodu bir $C(w, w_0)$ tarama fonksiyonuna uygulayarak spektrum örneklemede kullanılabilir.

$$C(w, w_0) = \sum_{k=1}^{\Omega_m/w_0} \delta(w - kw_0) \quad k = 1, 2, \dots, \frac{\Omega_m}{w_0} \quad (3.50)$$

Ω_m , ses spektrumundaki maksimum frekanstır. Eşit w_0 aralıklarında örneklenmiş ses spektrumu toplamı aşağıdaki gibi elde edilebilir.

$$A(w_0) = \frac{w_0}{\Omega_m} \sum_{k=1}^{\Omega_m/w_0} S(kw_0)C(kw_0) \quad (3.51)$$

$S(w)$ frekans spektrumunu gösterir. Perde frekansı (w_0) eşitlik 3.51'deki A fonksiyonunu maksimum yapan frekanstır. Şekil 3.8, bu metotla pik'lenmiş uygulama frekansını gösterir.



Şekil 3.8. Harmonik pik benzerlik metodu

3.4.2.2. Spektrum benzerlik metodu

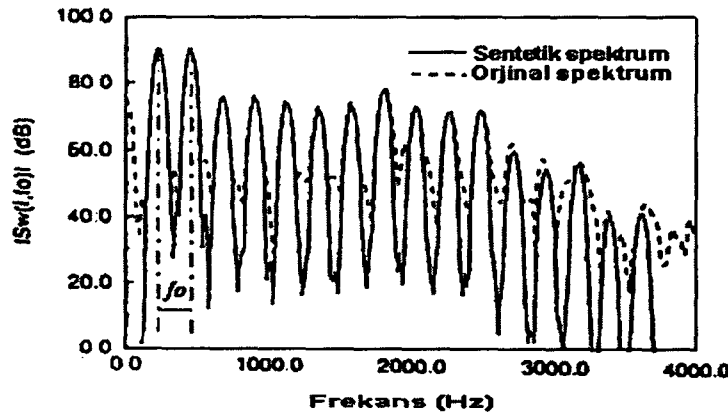
Bu metotta, ses penceresinin orijinal spektrumu deneme perde frekanslarının serisiyle oluşturulan sentetik spektrumla karşılaştırılır. Orijinal spektrumla en iyi benzerliği sağlayan perde adayı perde frekansı olarak seçilir (Şekil 3.9). Deneme harmoniklerindeki sentetik spektrum genliği ses spektrumundan hesaplanır ve benzerlik kriteri, izgel hatanın minimumlaştırılmasıdır. Bu kriter aşağıdaki gibi formüle edilebilir:

$$E(w_x) = \frac{1}{2\pi} \int_{-\pi}^{\pi} |S_w(w) - \hat{S}_w(w, w_x)|^2 dw \quad (3.52)$$

S_w orijinal ses penceresi spektrumu, $\hat{S}_w(w, w_x)$ w_x frekansında tekrar oluşturulan spektrum ve $E(w_x)$ sonuç hata fonksiyonudur. MBE kodlama, perde belirlemede bu tekniği kullanır.

3.4.3. Cepstral perde peryodu belirleme

Bu teknik ses üretim sisteminde iyi bilinen bir modele dayalıdır. Bu modelde, sesli ses perde frekansı ile ayrılan darbe sırasının bir filtre tarafından uyarıldığını farzeder. Bu filtre ve darbe sırası, sırasıyla ses üretim yolu ve ses



Şekil 3.9. Spektrum benzerlik PDA metodunda kullanılan orijinal ve sentetik ses spektrumu

kordonu titreşimini gösterir. Bu demektir ki eğer $e(t)$ ve $h(t)$ uyarım ve filtre impuls cevabını ifade ediyorsa, üretilen ses $s(t)$ aşağıdaki gibi ifade edilebilir:

$$s(t) = e(t) * h(t) \quad (3.53)$$

Frekans düzleminde, bu model aşağıdaki gibi ifade edilebilir:

$$S(f) = E(f) \cdot H(f) \quad (3.54)$$

Güç izgel yoğunluğu (PDF) aşağıdaki gibi hesaplanabilir:

$$P(f) = S(f) \cdot S^*(f) \quad (3.55)$$

$S(f)$ yerine konarak ve logaritmik olarak ifade edilirse:

$$\log[P(f)] = 2\log|E(f)| + 2\log|H(f)| \quad (3.56)$$

Yukarıdaki formül, PDF'i logaritmik yapıda ses üretim yolu ve uyarım'la birleştirilmiş iki frekans parçasının toplamı olarak ifade eder. Bu fonksiyonda gözlenen harmonik yapı, ses üretim yolunun uyarıma göre yavaş değişmesi nedeniyledir. Gerçek cepstrum, logaritmik PDF'in IFFT'si olarak ifade edilir:

$$C(\tau) = F^{-1}[\log[P(f)]] \quad (3.57)$$

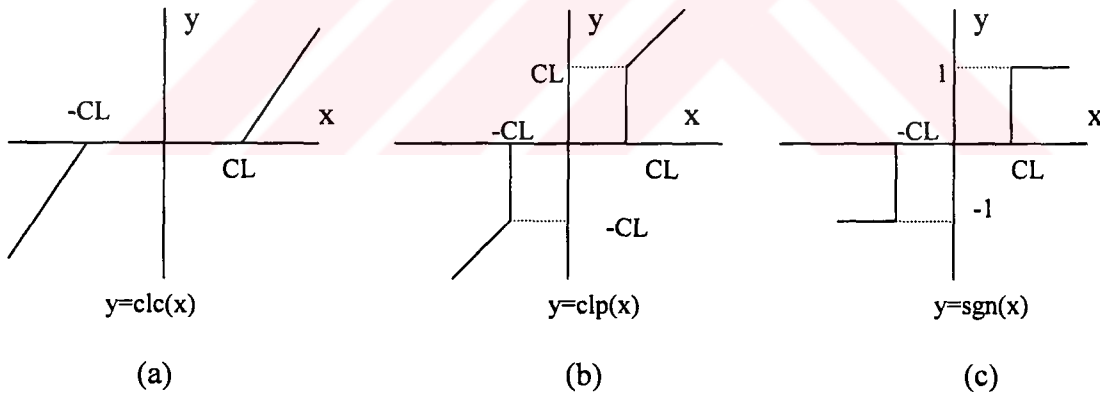
veya eşdeğer olarak

$$C(\tau) = 2F^{-1}[\log|E(f)|] + 2F^{-1}[\log|H(f)|] \quad (3.58)$$

$C(\tau)$ farklı izgel özelliklerin katkısının ölçüldüğü bir zaman düzlemi fonksiyonudur. Bu fonksiyonda perde yapısı kolayca toplanabilen yüksek pikler olarak görünür. Cepstral değerin ilk piki perde peryodunu direkt olarak belirler. Cepstrum perde belirleme algoritması yüksek doğruluk sağlamaktadır. Yine de, pratik uygulaması işlem fazlalılığı nedeniyle DSP çipleri çıkmadan önce olamamıştır. Bu metodun dezavantajı, ses üretim modeline dayalı olması nedeniyle daima doğru olmamasıdır. Ayrıca, geri plan gürültüsü varlığında kötüleşir.

3.4.4. Spektrum düzleştirme

Perde peryodunun doğru tahminindeki sorunlardan birisi, sesin peryodikliğine bağlı ölçümlerdeki formant yapısının etkisidir. Bu sorun kısmen perde sesteki ana formanttan etkilendiği zaman olur. Formantların bu etkisini kaldırarak, daha doğru bir perde peryodu hesaplanabilir. Bu işlem izgel düzleştirme (spectral flattening) olarak bilinir. Soundhi [52] bu amaçla merkez kırpma metodunun daha popüler olduğu iki metot önermektedir. Bu metotta, bir eşik değeri altındaki sinyal değerleri sıfıra set edilir ve üzerindeki değerler kırpma seviyesine ofset edilir. Bu metodun giriş-çıkış karakteristiği şekil 3.10.a ve b'de verilmiştir. Şekil 3.9.c'de ise, kırpma seviyesi üzerindeki sinyal sabit tutularak enerji değişimleri azaltılır. Soundhi tarafından önerilen kırpma seviyesi CL , pencere içindeki sesin maksimum mutlak değerinin yüzde 30'udur. Perde peryodu için, kırpılmış sinyale özilişki metodu uygulanır. Bir pik tarayıcı ile özilişkinin maksimum olduğu aralık (perde peryodu) belirlenir.



Şekil 3.10. Merkez kırpıcıların giriş-çıkış karakteristiği

3.4.5. Farklı metodların karşılaştırılması

Farklı perde belirleme algoritmaları arasında AC metoduna dayalı zaman düzlemi tekniği en yaygın olarak kullanılan metottur. Bu metodun popülerliği kısmen pek çok DSP çiplerinde tek bir komut olarak bulunabilen toplama ve

çarpma içermesidir. Ayrıca, frekans düzlemi algoritmalar hesapsal bir yük getiren sesin Fourier dönüşümünü içermektedir. Hızlı FFT algoritmaları geliştirilmesine rağmen daha etkili dönüşüm metotları kullanılabilir. Frekans düzlemi algoritmalar hala zaman düzlemi algoritmalarından daha karışıktır. Yine de, frekans düzlemi metotlar, diğer fonksiyonlar içinde FFT içeren kodlama şemaları ile iyi uyum sağlarlar.

Zaman düzlemi metotların doğruluğu örnekleme hızına bağlıdır. Bir 8 kHz örnekleme frekansı için, bu metotların doğruluğu 50-400 Hz arası perde frekansları için sırasıyla yüzde 0.3'den 2.5'e değişir. Daha yüksek doğruluk için, ses sinyali yukarı örneklenmelidir. Frekans düzlemi algoritmaların doğruluğu aynı zamanda FFT yoğunluğuna bağlıdır. Çoklu bant kodlama algoritmasında, AC metodu başlangıç perde değerini belirlemede, spektrum benzerlik metodu ise perde tahmin doğruluğunu artırmada kullanılmaktadır.

3.5. Sesli/Sessiz Sinyal Sınıflandırma

Ses sınıflandırmanın amacı ses penceresinin sesli ya da sessiz olduğuna karar vermektir. En gerçekçi metot sesli sesteki temel özellikleri bulup ortaya çıkarmaktır. Yine de burada iki problem vardır. Birincisi geçiş pencerelerindeki karar verme zorluğu (hem sesli hem sessiz durumları içeren pencereler), ve, hem periyodik hem de rasgele parçalar içeren (V,Z.. gibi fricative'ler) sesler. Böyle bir durumda, bir ikili sesli/sessiz (Voiced/Unvoiced-V/U) karar ses penceresinin durumunu göstermede yetersiz kalır. Bunun için basit bir çözüm, ses penceresini alt pencerelere bölerek, bu alt pencerelerde V/U analizi yapmaktır. MBE de kullanılan başka bir yaklaşım, her bir pencereyi sesli ve sessiz durumların bir birleşimi olarak düşündürmektir. MBE, V/U analizi için frekans düzleminde bir analiz yapar ve hangi frekansın sesli ya da sessiz olduğuna karar verir. Bunu başarmak için,

ses penceresinin orijinal spektrumu sentetik spektrum ile karşılaştırılır. (Sentetik spektrum, bütün izgel parçaların sesli olduğunu farzederek oluşturulur). İki spektrumdaki frekans bantları arasında benzerliğin belli bir eşik üzerinde olduğu bantlar sesli, diğerleri sessiz olarak kabul edilir. Böyle bir yaklaşım çok etkili görünmesine rağmen, sesli ve sessiz durumların içerildiği pencerelerde hatalar yapmaktadır. V/U kararlarda aşırı bit sayısını önlemek için, düşük bit hızlı MBE uygulamalarda her bir alt analiz bandı üç harmonik içermektedir ve bantların sayısı sınırlanmaktadır. Sonuç olarak, MBE doğruluğu perde peryodunun doğruluğuna bağlıdır. Bu yüzden , düşük perde peryoduna sahip (erkek) konuşmacılar için doğru karar vermesi zordur ve çıkıştaki ses kalitesi derecelidir.

V/U sınıflandırmada hesapsal karmaşa diğer önemli bir faktördür. Aslında, sınıflandırıcının kalitesi ile yapısı birlikte ele alınmalıdır. Sonuç olarak, V/U sınıflandırıcı için bütün kodlayıcılara uygulanabilecek tek bir yöntem yoktur. Her bir kodlama algoritması uygun bir sınıflandırıcı seçebilir. [48,53,54]

4. VEKTÖR NİCEMLEME TEKNİKLERİ

4.1. Giriş

İzgel parametrelerin vektör nicemlenmesi, LPC parametrelerinin çok az bit sayısı ile çok iyi kodlanması gereken düşük bit hızlı konuşma kodlayıcılar (4 kbit/sn altında) için çok önemlidir. Bir vektör nicemleyici dijital bir kanalda vektörlerin kodlanması için, distorsiyon artışına neden olmadan bit sayısını azaltan bir sistemdir. Vektör nicemleme yöntemi ile saklanan değerler, filtre katsayıları, yansıma katsayıları ya da LSF katsayılarıdır. Vektör nicemlemede örneklenmiş dalga şeklinin belli bir periyodu çıkarılır ve bu periyoddaki örüntü, tek bir kodla gösterilir. Bu işlem, tipik dalga şekli örüntülerini depolayarak başlanır ve her örüntü için bir kod verir. Örüntüler ve kodlar arası karşılığı gösteren çizelge bir *kod çizelgesi* (codebook) diye adlandırılır. Kod çizelgesi, önceden belirlenmiş bütün aralıklarda her örüntü ile karşılaştırılır ve en yüksek benzerliğe sahip örüntüye ait kod seçilerek yollanır. Kod çizelgesi dizaynında önemli bir nokta, sınırlı sayıda örüntü ile değişik tiplerde dalga şekilleri karşılaştırılacağı için tüm distorsiyonu minimum yapacak uygun örüntü kümeleri belirlemek gerekir. [8]

Bu bölümde vektör nicemleme için gerekli olabilecek bazı tanımlamalar ve hesaplamalar, vektör nicemleme tanımı ve bazı vektör nicemleme yöntemleri üzerinde durulmuştur.

4.2. LPC Alternatif Gösterimleri

LPC katsayılarının farklı şekillerde gösterimi, kanal hata şartları ve nicemleme sırasında iyi performans için çok önemlidir. LPC katsayılarının değişik

gösterimleri mevcuttur. Seçim, onların sağlamlık ve nicemleme yeterliliklerine bağlıdır. LPC filtre, aşağıdaki şekilde ifade edilir:

$$H(z) = \frac{1}{1 - \sum_{j=1}^p a_j z^{-j}}$$

p , LPC filtre derecesidir. a_j katsayıları, LPC katsayılarıdır. $H(z)$ filtresi eğer minimum faz ise karardır, yani eşitliğin bütün kökleri birim çember içindedir. Eğer a_j direkt nicemlenirse, $H(z)$ kararlılığı garanti edilemez, çünkü eşitliğin kökleri kararlılık kontrolü için hesaplanamaz. [17,25,26,49]

4.2.1. Yansıma katsayıları

Levinson-Durbin algoritması [49] ile elde edilen yansıma katsayıları, ses üretim yolunun bir akustik tüp modelini oluştururlar. Eğer doğrusal öngörü derecesi akustik tüp modelindeki bölüm sayısına eşitse, yansıma katsayıları ses dalga şeklinin doğrusal öngörü analizinden kolayca hesaplanabilir ve her bir katsayı ses üretim yolunun akustik tüp modeli bölümlerinin alanını belirlerler. Ayrıca yansıma katsayıları, sentez filtresi kararlılığı için gerekli ve yeterli şartları sağlarlar. $a_j^p = \alpha_j$, $1 \leq j \leq p$ olarak iterasyon başlar. $i = p, p-1, \dots, 1$ için aşağıdaki eşitlikler uygulanır.

$$a_j^{i-1} = \frac{(a_j^i + a_i^i \cdot a_{i-j}^i)}{(1 - k_i^2)}, \quad 1 \leq j \leq i-1 \quad (4.1)$$

$$k_{i-1} = a_{i-1}^{i-1} \quad (4.2)$$

Eğer, $|k_i| \leq 1.0$ ise LPC filtre karardır. k_i kararlılık için kolayca kontrol edilebilmesine rağmen, nicemleme için uygun değildir. Çünkü, düz olmayan bir izgel duyarlılığı vardır, yani bire yakın k_i değerleri daha çok nicemleme doğruluğu gerektirirler.

LPC parametreleri için *yansıma katsayıları* (k_j), $1 \leq j \leq p$ çok bilinen bir gösterim şeklidir. Yansıma katsayıları genellikle latis şekilde sayısal filtrelerle oluşturulur ve $-1 \leq k_j \leq 1$ arasında yapısal bir kararlılık gösterirler.

4.2.2. LAR ve IS katsayıları

Yansıma katsayıları yukarıda belirtildiği gibi doğrusal olmayan bir izgel duyarlılığa sahiptirler. İlk birkaç yansıma katsayısı, pek çok sesli seslerde çarpık bir dağılıma sahiptirler. Daha yüksek dereceli katsayılar Gaussian dağılıma sahiptirler. İlk yansıma katsayısı, k_1 , 1'e, ikinci yansıma katsayısı, k_2 , ise -1'e çok yakındır. Eğer ($f \leq 10\text{kHz}$) şeklinde düşük bir örnekleme frekansı ses için kullanılırsa, diğer yansıma katsayıları 0.7'den daha küçüktürler. Yansıma katsayılarında $[-1,1]$ arası doğrusal nicemleme işe yaramaz. Doğrusal olmayan duyarlılık yüzünden, doğrusal olmayan nicemleme uygundur. Bunun için bir yöntem LPC katsayılarının başka bir gösterim şekli olan LAR (Log Area Ratio) katsayılarıdır. LAR katsayıları l_j , $1 \leq j \leq p$ dir ve yansıma katsayılarından türetilirler.

$$l_j = \log \frac{(1 + k_j)}{(1 - k_j)} = 2 \tanh^{-1} k_j \quad (4.3)$$

l_j katsayıları, özellikle dar bant genişlikli kutuplarda, 1'e çok yakın $|k_j|$ değerleri için nicemleme gürültüsüne çok duyarlı olan yansıma katsayılarından daha az çizgisel duyarlıdır.

LPC katsayılarının başka bir gösterim şekli IS (Inverse Sinus) (s_j) çevrimidir.

$$s_j = \arcsin k_j \quad (4.4)$$

l_j ve s_j gibi düşük gürültü duyarlılıklı LPC parametre gösterimleri ile, her pencerede 8/10 daha az bit kullanarak geleneksel LPC gösterimlerle aynı ortalama nicemleme hatası sağlanabilir.

4.2.3. Çizgi spektrum frekans (LSF) katsayıları

LAR veya IS katsayıları ile dönüşüm metodunda, minimum distorsiyonu başarmak için $p=10$ 'uncu derece bir filtre için 35-40 kbit/pencere kodlama gereklidir. Ayrıca LPC parametrelerinin pencereden pencereye gerçekte var olan ilişkisi, LAR ve IS dönüşümlerde net olarak görülmez. Son olarak çok popüler bir LPC parametre gösterimi, *Line Spectral Pair –LSP* veya *Line Spectral Frequency – LSF* dir. LSP parametreleri iyi bir dinamik değişim davranışı ve kararlılığa sahiptir ve LPC parametreleri kodlanmasında yaygın olarak kullanılır. LSF dönüşüm, yukarıdaki her üç metodun dezavantajlarını bir ölçüde ortadan kaldırır. Frekans düzleminde LSF dönüşüm, ses sinyalindeki izgel özelliklerini iyi korur, ayrıca LSF parametreleri pencereler arası interpolasyon yapmaya elverişlidirler.

LSP gösterimi yapaydır ve verilen bir m -aşama model için, ses üretim modelinin (vocal-tract) her $(m+1)$ 'inci aşamada açık ya da kapalı olduğunu farzeder. Bu ona, bir ters filtrenin sıfırlarının birim çember üzerinde yer alması gibi ilginç bir özellik verir ve LSP katsayıları bu sıfırların frekanslarına karşılık gelir. LSF'lerin yakın değerleri rezonans frekanslarına karşılık gelir, yaygın gösterimler ise izgenin eğimine sebep olurlar. LPC parametrelerinin LSF gösterimi, diğer gösterimlere göre nicemleme bit oranında %30 tasarruf sağlar. [67,74,77]

LSF frekansları doğrusal öngörü parametrelerinin başka bir gösterimidir. Doğrusal öngörü parametreleri çok büyük bir değişim aralığına sahiptir ve nicemleme için uygun değildir. LSF'ler ise iyi bir dinamik değişime sahiptirler. Eğer LSF düzleminde aradeğerleme yapılırsa, sonuç sentez filtresinin kararlılığını garantilemek daha kolaydır. Eğer doğrusal öngörü parametreleri LSF'ler olarak kodlanırsa, LSF'lerin herbiri için aynı sayıda bit

kullanılmasına gerek kalmaz. Çünkü daha yüksek sayılı LSF'ler daha yüksek LSF frekanslarına karşılık gelirler ve yüksek frekans parçası ses üzerinde daha az etkiye sahiptir. Bu yüzden yüksek frekans parçası içeren LSF'ler daha az bit sayısı ile kodlanırlar. Böylece ses kalitesi aynı kalırken bit sayısı önemli miktarda azalır. LSF'ler bir frekans düzlemi gösterime sahiptirler. Genellikle LSF'ler formantlar etrafında odaklanmışlardır. Ayrıca herbir LSF'in izgel yoğunluğu bölgeseldir. LSF'deki bir değişim, güç spektrumunda ilgili LSF frekansı komşuluğunda bir değişime neden olur. LSF'in başka bir özelliği, p 'inci dereceden LSF'in $(p-1)$ 'inci dereceden LSF'le bağlantılı oluşudur. Bu karşılıklı bağlantılı durum, formant frekans bölge sınırında bir bağ sağlar.

4.2.3.1. LSF katsayıları hesaplanması

Doğrusal öngörü analiz filtresi $A(z)$, doğrusal öngörü katsayıları a_k 'lar cinsinden aşağıdaki şekilde yazılabilir. [67]

$$A(z) = 1 - \sum_{k=1}^p a_k \cdot z^{-k} \quad (4.5)$$

$A(z)$, p 'inci dereceden bir filtredir. $(p+1)$ 'inci derece simetrik ve anti-simetrik polinomlar $P(z)$ ve $Q(z)$, $A(z)$ 'den elde edilebilir:

$$\begin{aligned} P(z) &= A(z) + z^{-(p+1)} A(z^{-1}) \\ Q(z) &= A(z) - z^{-(p+1)} A(z^{-1}) \end{aligned} \quad (4.6)$$

burada

$$A(z) = \frac{1}{2}[P(z) + Q(z)] \quad (4.7)$$

$P(z)$ ve $Q(z)$ 'nin üç önemli özelliği vardır:

- $P(z)$ ve $Q(z)$ polinomlarının bütün kökleri birim çember içindedir
- $P(z)$ ve $Q(z)$ kökleri bağlantılıdır

- Eğer nicemleme ve aradeğerleme işleminden sonra yukarıdaki iki özellik korunuyorsa, $A(z)$ 'nin minimum faz özelliği korunur.

İlk özellikten görülür ki, $P(z)$ ve $Q(z)$ kökleri $w_i (e^{j\omega})$ cinsinden ifade edilebilir. Bu w_i 'ler LSF'ler olarak adlandırılır. $P(z)$ ve $Q(z)$, $z=1$ ve $z=-1$ olarak iki köke sahiptir. $P(z)$ ve $Q(z)$ ile aynı köklere sahip fakat $z=1$ ve $z=-1$ de kökleri olmayan $N_1(z)$ ve $N_2(z)$ olarak iki yeni polinom alınsın.

$$N_1(z) = \begin{cases} \frac{P(z)}{1+z^{-1}} & \text{her çift } p \text{ için} \\ P(z) & \text{her tek } p \text{ için} \end{cases} \quad (4.8)$$

$$N_2(z) = \begin{cases} \frac{P(z)}{1-z^{-1}} & \text{her çift } p \text{ için} \\ \frac{P(z)}{1-z^{-2}} & \text{her tek } p \text{ için} \end{cases} \quad (4.9)$$

Bu iki eşitlikten görülmektedir ki, $N_1(z)$ ve $N_2(z)$ aynı çift dereceye sahip ve simetriktirler. Kökleri karmaşık eşlenik (complex conjugate) olarak oluşur. Ayrıca sadece üst yarım dairedeki kökler hesaplanır. $N_1(z)$ ve $N_2(z)$ 'nin kökleri sırasıyla $2m$ ve $2n$ olarak alınsın. O zaman;

$$m = \begin{cases} \frac{p}{2}, & p \text{ çift için} \\ \frac{p+1}{2}, & p \text{ tek için} \end{cases} \quad (4.10)$$

$$n = \begin{cases} \frac{p}{2}, & p \text{ çift için} \\ \frac{p-1}{2}, & p \text{ tek için} \end{cases} \quad (4.11)$$

$N_1(z)$ ve $N_2(z)$ açık olarak yazılırsa,

$$N_1(z) = 1 + N_1(1)z^{-1} + N_1(2)z^{-2} + \dots + N_1(m)z^{-m} + \dots + N_1(1)z^{-(2m-1)} + z^{-2m} \quad (4.12)$$

$$N_2(z) = 1 + N_2(1)z^{-1} + N_2(2)z^{-2} + \dots + N_2(n)z^{-n} + \dots + N_2(1)z^{-(2n-1)} + z^{-2n} \quad (4.13)$$

Başka bir ifadeyle,

$$N_1(e^{jw}) = e^{-jwm} N_1'(w) \quad (4.14)$$

$$N_2(e^{jw}) = e^{-jwn} N_2'(w) \quad (4.15)$$

burada,

$$N_1'(w) = 2 \cos mw + 2N_1(1) \cos(m-1)w + \dots + N_1(m) \quad (4.16)$$

$$N_2'(w) = 2 \cos nw + 2N_2(1) \cos(n-1)w + \dots + N_2(n) \quad (4.17)$$

$N_1'(w)$ ve $N_2'(w)$ fonksiyonlarının kökleri LSF frekanslarıdır. Kabal ve Ramachandran [67], x 'de m 'inci dereceden bir $T_m(x) = \cos(mw)$ Chebyshev polinomu kullanmıştır. Burada $T_m(x) = 2xT_{m-1}(x) + T_{m-2}(x)$ dir. Şimdi, $N_1'(w)$ ve $N_2'(w)$:

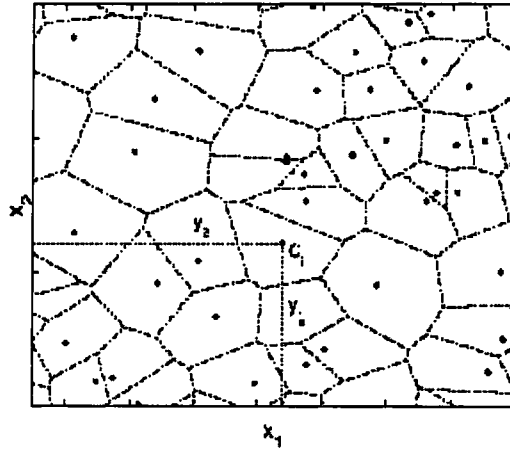
$$N_1'(x) = 2T_m(x) + 2N_1(1)T_{m-1}(x) + \dots + N_1(m) \quad (4.18)$$

$$N_2'(x) = 2T_n(x) + 2N_2(1)T_{n-1}(x) + \dots + N_2(n) \quad (4.19)$$

Açılmış polinomların kökleri $[-1,1]$ aralığında işaret değişimine bakarak belirlenir ve sonra $w = \cos^{-1}(x)$ ile LSF frekansları bulunur.

4.3. Vektör Nicemleme

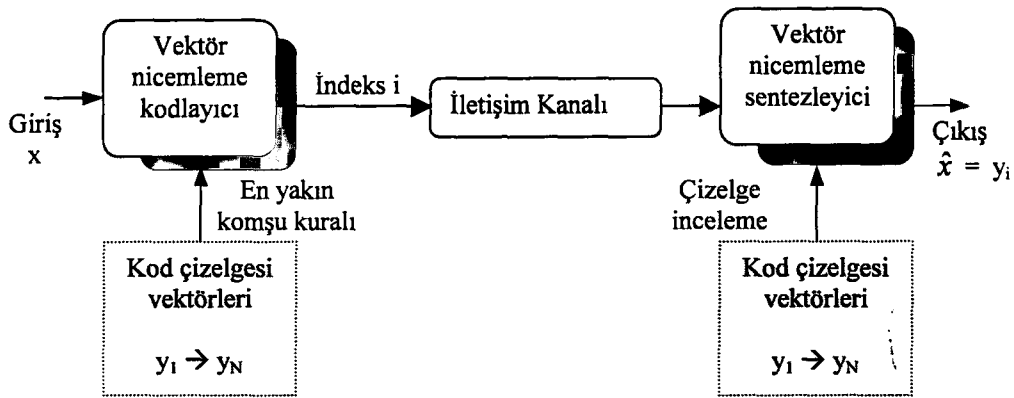
Vektör nicemleme (vector quantization), skalar nicemlemeye göre yüksek etkinliğe sahip bir kodlama metodudur. Bunun yanında hesaplama ve depolama maliyeti yüksektir. Hesaplama ve depolama maliyetine göre ve nicemleyici performansına göre sınıflanan birçok vektör nicemleme tipleri mevcuttur. Bazı kod çizelgeleri önceden hesaplanır ve değişmez, bazıları ise nicemleme sırasında yeniden uyarlanır. [17]



Şekil 4.1. İki boyutlu uzay örneği ($N = 2$). C_i hücresindeki bütün giriş vektörleri. $y=[y_1,y_2]$ kod vektörü olarak nicemlenir

$x=[x_1,x_2,\dots,x_N]^T$, bir N boyutlu ve gerçekteğerli ve sürekli genlikli vektör olsun. Vektör nicemleme işleminde x vektörü, başka bir N boyutlu, gerçekteğerli ve kesikli genlikli y vektörü ile eşleştirilmeye çalışılır. Genellikle y vektörü, $Y = y_i$, $1 \leq i \leq L$, $y_i = [y_{i1}, y_{i2}, \dots, y_{iN}]^T$ sınırlı kümesinden seçilmektedir. Y , kod çizelgesi kümesidir. L kod çizelgesinin büyüklüğü, y_i ise kod çizelgesindeki vektörlerin kümesidir. Böyle bir kod çizelgesi dizayn etmek için, N boyutlu uzay L hücreye (cell) bölünür. Bu hücreler C_i , $1 \leq i \leq L$ olarak adlandırılır ve her bir y_i vektörü bir C_i hücresi ile eşleştirilmeye çalışılır. Eğer x bir C_i hücresi içinde yer alıyorsa, nicemleyici x 'i y_i vektörüne atar. ($x \in C_i$ ise $q(x) = y_i$ dir). Şekil 4.1, iki boyutlu bir uzayda ($N = 2$) vektör nicemleme için hücre bölme yapısını göstermektedir. [8,27,44,45,55-78]

Şekil 4.2 genel bir vektör nicemleme yapısını göstermektedir. Normal olarak, vektör nicemleyici eğitim giriş dizisi içindeki seslerin şartlarını taşıyan seslerle kullanılırsa, sonuç çok iyidir. Fakat, farklı giriş şartları ve ortamlardaki sesler için performans düşer. Gerçek performans için, vektör nicemleyici kendi eğitim giriş dizisi dışındaki seslerle denenerek performans ölçümü yapılır.



Şekil 4.2. Vektör nicemleme modeli

Bir tam aramalı (full-search) vektör nicemleyicide, 1dB'den az SD için çok büyük kod çizelgesi gereklidir. Oysa, kod çizelgesi yapısını uygun şekilde dizayn ederek hem hafıza gereksinimi hem de yapısal karmaşa önemli miktarda azaltılabilir. İki önemli metot, çok aşamalı vektör nicemleme [44,45] ve yarık vektör nicemleme [49,84] metotlarıdır.

Vektör nicemleme işleminde iki kritik faktör çok önemlidir. Bunlar, eğitim giriş vektör sayısı ve GLA iterasyon sayısıdır. Küçük bir eğitim giriş dizisi vektör sırası istatistiksel karakteristiğini iyi tanımlayamaz ve iyi bir nicemleyici kod çizelgesi sonucunu veremez. Çok büyük eğitim giriş dizileri de zayıf performansa yol açar. Eğitim giriş dizisinin kod çizelgesine oranlaması 50'den az 200'den fazla olmamalıdır.

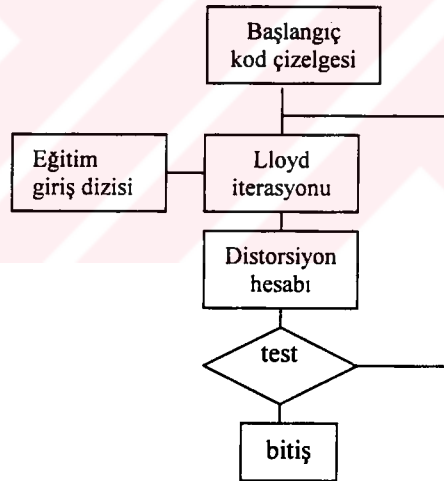
4.3.1. Genelleştirilmiş Lloyd algoritması

Başlangıç kod çizelgesinden sonra, bir uyarlamalı algoritma ile vektör nicemleme işlemi başlatılır. GLA (Generalized Lloyd Algorithm), vektör kod çizelgesi oluşturmada kullanılan en yaygın uyarlamalı bölütleme algoritmasıdır (Çizelge 4.1).

Her GLA iterasyonu (Çizelge 4.2), vektör nicemleyicinin ortalama distorsiyonunu yavaş yavaş azaltır ve belli bir iterasyondan sonra değişim durma eğilimine girer.

Çizelge 4.1. GLA kod çizelgesi hazırlama algoritması

ADIM 1. Bir başlangıç kod çizelgesi C_1 ile başla. $m = 1$, al.
ADIM 2 Verilen kod çizelgesi $C_m = \{y_i\}$ için yeni kod çizelgesi C_{m+1} üretmek için Lloyd iterasyonu uygula: Eğitim giriş dizisini en yakın komşu kuralına göre, R_i kümelerine ayr. T , eğitim giriş kod çizelgesi uzunluğu ise, $R_i = \{x \in T / d(x, y_i) \leq d(x, y_j), \forall j \neq i\}$. Her R_i kümesinin ağırlık merkezini ($ağ_mrz$) hesapla. Yeni kod çizelgesi $C_{m+1} = \{ağ_mrz(R_i), i = 1, \dots, N\}$ olsun.
ADIM 3 C_{m+1} için ortalama distorsiyonu hesapla. Eğer, bir önceki iterasyona göre



Çizelge 4.2. Lloyd (GLA) algoritması akış çizelgesi

4.3.2. Yarı vektör nicemleme

Yarı vektör nicemleme-YVN (Split Vector Quantization) metodunda, çok boyutlu bir vektör iki ya da daha alt sayıda alt vektörlere bölünür ve herbirisi

ayrı olarak vektör nicemlenebilir. Bir k boyutlu özellik vektörü x , k_i boyutlu m adet alt vektör f_i 'nin birleşimidir.

$$x = [f_1^T, \dots, f_m^T]^T, \quad (\sum_{i=1}^m k_i = k) \quad (4.20)$$

YVN, bölütlenmiş vektör nicemleme olarak da bilinir. YVN, çok aşamalı vektör nicemlemenin özel bir sınıfıdır da denilebilir. Belli özellik vektör parçaları sıfıra zorlanır. Örneğin, x 'in i 'inci alt vektörü f_i , ilk $\sum_{j=1}^{i-1} k_j$ ve son $\sum_{j=i+1}^m k_j$ vektör parçalarının sıfıra zorlandığı i 'inci aşama k -boyutlu vektör x_i 'ye eşittir

4.3.3. Çok aşamalı vektör nicemleme

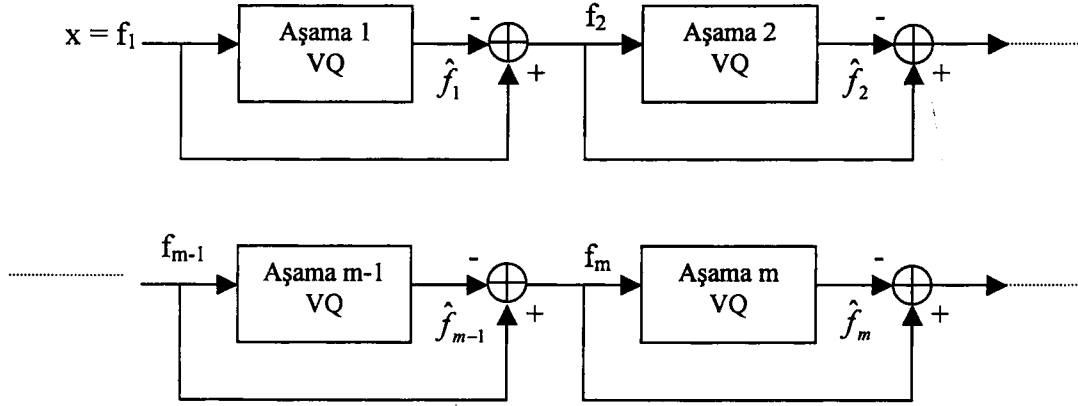
Çok aşamalı vektör nicemleme-ÇAVN (Multi Stage Vector Quantization-MSVQ) [27,44,45], vektör nicemleme aşamalarının bir kaskatını içerir. Herbir aşama f_i vektörünün bir özelliğini nicemler. Şekil 4.3'de görüldüğü gibi, m -aşamalı bir nicemleyicide, önceki aşamalardan nicemlenmiş vektörlerin toplamı giriş vektöründen çıkarılarak f_i vektörünün i 'inci aşama özelliği elde edilir.

$$f_i = x - \sum_{j=1}^{i-1} \hat{f}_j \quad (4.21)$$

İlk özellik vektörü $f_1 = x$ 'dir. ÇAVN için, giriş vektörü x , ilk aşamada kabaca nicemlenir. İkinci aşama için bir sonuç nicemleme hata vektörü giriş vektörü olarak kullanılır. Herbir alt aşama önceki aşamadaki artık vektörü nicemleyerek giriş vektörü için daha iyi bir nicemleme sonucu verir. M aşamalı ÇAVN için, tekrar oluşturulan vektör m nicemlenmiş artık vektörleri toplayarak elde edilir:

$$\hat{x} = g(\hat{f}_1, \dots, \hat{f}_m) = \sum_{i=1}^m \hat{f}_i \quad (4.22)$$

ÇAVN aşama sayısı arttıkça depolama ve karmaşa yönünden azalmasına rağmen performansı derecelenir. M aşamalı ÇAVN için depolama maliyeti $\sum_{j=1}^m 2^{b_i}$ dir. b_i i'inci aşamada kullanılan bitlerin sayısıdır.



Şekil 4.3. Çok aşamalı vektör nicemleyici (m-ÇAVN)

Bir ÇAVN sistemde, p LSP (Line Spectral Parameters) katsayısı içeren x parametre vektörü, yaklaşık olarak $\hat{x}$ vektörüne eşittir.

$$\begin{aligned} \hat{x} &= y_0^{(l_0)} + y_1^{(l_1)} + \dots + y_{K-1}^{(l_{K-1})}, & y_j^{(k)} &: (pxl) \\ &= B_0^{(l_0)} c_0 + B_1^{(l_1)} c_1 + \dots + B_{K-1}^{(l_{K-1})} c_{K-1}, & B_j^{(k)} &: (pxL_jp), \quad c_j : (L_jpxl) \\ &= Bc, & B &: (pxLp), \quad c : (Lpxl) \end{aligned} \quad (4.23)$$

$\hat{x}$, x 'in nicemlenmiş şeklidir. K , aşamaların sayısıdır $y_j^{(k)}$ ve, j 'inci aşamadaki k 'inci kod vektördür (Burada bütün vektörler kolon vektör kabul edilir). j 'inci aşamada, L_j kod vektör saklanmaktadır ve c_j vektörü, j 'inci aşama için olan kod vektörleri biraraya toplayarak oluşturulmaktadır.

$$c_j = [y_j^{(0)T} \quad y_j^{(1)T} \quad \dots \quad y_j^{(L_j-1)T}] \quad (4.24)$$

c_j vektörü, j 'inci aşamadaki kod çizelgesidir. c kolon vektörü tüm kod çizelgesi yapısına karşılık gelir :

$$c = [c_0^T \quad c_1^T \quad \dots \quad c_{K-1}^T] \quad (4.25)$$

j 'inci aşama için seçme matrisi $B_j^{(k)}$, Toeplitz matristir ve $B_j^{(k)} c_j = y_j^{(k)}$ dir. $B_j^{(k)}$ matrisi, j 'inci kod çizelgesinden k 'inci kod vektörü seçer. Sonuç olarak, çok aşamalı seçme matrisi,

$$B = [B_0^{(l_0)} B_1^{(l_1)} \dots B_{k-1}^{(l_{k-1})}] \quad (4.26)$$

olur. Giriş vektörü x 'e en yakın $y_0^{(l_0)}$ vektörünü seçmek için bir sıralı araştırma işlemi kullanılırsa,

$$d(x, y_0^{(l_0)}) \leq d(x, y_0^{(k)}) \quad \forall k \neq l_0 \quad (4.27)$$

daha sonra, $(x - y_0^{(l_0)})$ 'e en yakın $y_1^{(l_1)}$ vektörü seçilir.

$$d(x - y_0^{(l_0)}, y_1^{(l_1)}) \leq d(x - y_0^{(k)}, y_1^{(k)}) \quad \forall k \neq l_1 \quad (4.28)$$

Bu şekilde K aşama için işlem tekrar eder. Fakat sıralı araştırmada, bir sonraki aşamanın değerleri sıfır kabul edilerek kod çizelgesi oluşturulduğu için, aşama sayısı arttığında performans azalmaktadır. Bütün aşamalardaki bütün kod çizelgesi yapısını dikkate alabilmek için daha etkili dizayn ve araştırma metotları kullanmak gerekir. Bunun için, özyineli sıralı dizayn ve birleşik dizayn metotları kullanılmaktadır [27,44,45,61,79].

4.3.3.1. Özyineli sıralı kod çizelgesi metodu

Özyineli sıralı (iterative sequential) dizayn için distorsiyon formülü:

$$d_r = \sum_n (x_n - \hat{x}_n^{(r)})^T W_n (x_n - \hat{x}_n^{(r)}) \quad (4.29)$$

giriş matrisi ve yaklaşık kod matris arasındaki hata:

$$\begin{aligned} x_n - \hat{x}_n &= x_n - B_{n,j}^{(l_j)} c_j - \sum_{k=0, k \neq j}^{K-1} B_{n,k}^{(l_k)} c_k \\ &= e_{n,j} - B_{n,j}^{(l_j)} c_j \end{aligned} \quad (4.30)$$

burada $x_n = [x_1, x_2, \dots, x_m]^T$. Özyineli sıralı optimizasyon için, bütün diğer aşamalar sabit ve biliniyor kabul edilerek, j 'inci aşama optimize edilir. İlk başta, kod çizelgelerinin ilk değeri verildiğinde, her aşama, o ana kadarki diğer aşamalarla optimize edilir. Böylece, sonuç WMSE (Weighted Mean Square Error) d_r minimumlaştırılır. (4.30) formülü (4.29)'da yerine konularak d_r yeniden oluşturulur.

$$\begin{aligned} d_r &= \sigma_j^2 - 2\psi_j^T c_j + c_j^T \Psi_j c_j \\ \sigma_j^2 &= \sum_n e_{n,j}^T W_n e_{n,j} \end{aligned} \quad (4.31)$$

$$\begin{aligned} \psi_j &= \sum_n B_{n,j}^{(l_j)T} W_n e_{n,j} \\ \Psi_j &= \sum_n B_{n,j}^{(l_j)T} W_n B_{n,j}^{(l_j)} \end{aligned} \quad (4.32)$$

Özyineli sıralı dizaynda (Çizelge 4.3), her iterasyonda bir kod çizelgesi hesaplandığı için yakınsaklık hızı yavaştır. Yakınsaklık hızı ve performans artışı birleşik kod çizelgesi dizaynı ile elde edilir. [44,45,79]

Çizelge 4.3. Özyineli sıralı kod çizelgesi hazırlama algoritması

1. **Başlangıç:** $r = 1, j = 0$. K aşamalı başlangıç kod çizelgesi oluşturun.
2. **Sınıflandırma:** Ψ_j ve ψ_j elemanlarını sıfırlayın ve eğitim giriş dizisindeki her vektör için, WMSE'yi min yapacak kod çizelgesi indislerini belirleyin. Ψ_j ve ψ_j ve d_r 'i hesaplayın.
3. **Kod çizelgesi hesaplama.** Yeni kod çizelgesi c_j 'yi hesaplayın.
4. **Yakınsaklık testi.** $[(d_{r-1} - d_r) / d_r] > \varepsilon$ ise $r = r+1, j = (j+1) \bmod K$. Adım 2'ye gidin.
5. **Bitiş.**

4.3.3.2. Birleşik kod çizelgesi metodu

Birleşik (joint) dizaynda amaç, her iterasyonda, bütün aşamalardaki kod vektörleri aynı anda optimize etmektir. Bu yüzden, birleşik dizayn, özellikle

aşama sayısı arttığında (K büyüdüğünde) sonuçtaki düşük distorsiyona daha çabuk yakınsar ve bazen de daha düşük distorsiyon verir. Eğitim ve test aşamaları için M - L arama yöntemi kullanılarak, tam aramalı metoda çok yakın sonuçlar elde edilebilmektedir (Çizelge 4.4).

Birleşik dizayn için, giriş vektörleri minimum distorsiyon araştırmasına göre kümelendirilir. Bu kümeler içinde, kod çizelgeleri tekrar optimize edilir. Yeni kod çizelgeleri ile tekrar optimizasyon yapılır. İşlem tekrar ettirilerek, ortalama distorsiyon değerinin düşürülmesine çalışılır. Giriş vektörlerinin her kümeleneşmesi aşamasında birleşik olarak bütün kod vektörlerin optimize edilmesi için, ortalama distorsiyon d_r minimum yapılmaya çalışılır (Burada $\hat{x}_n = B_n c$).

$$\begin{aligned} d_r &= \sum_n (x_n - \hat{x}_n)^T W_n (x_n - \hat{x}_n), \\ d_r &= \left(\sum_n x_n^T W_n x_n \right) - 2c^T \left(\sum_n B_n^T W_n x_n \right) + c^T \left(\sum_n B_n^T W_n B_n \right) c \\ d_r &= d_0 - 2c^T Y + c^T Q c, \quad Y: (Lp \times 1) \quad Q: (Lp \times Lp) \end{aligned} \quad (4.33)$$

Verilen c kod çizelgesinde, bütün giriş vektörleri için genel seçme matrisi B_n belirlenir, ve daha sonra verilen B_n için c kod çizelgesi hesaplanır. Minimumlaştırma çözümü için, $Q.c = Y$ 'yi sağlamak gereklidir. Buradan kod çizelgesi $c = Q^{-1}Y$ dir. Fakat, genelde Q matrisi “tam-rank” olmadığı için, matrisin tersi (Q^{-1}) mevcut değildir. c vektörünü bulmak için kullanılabilecek pek çok metotlardan birisi “izdüşüm (projection) metodu” dur. Burada j 'nci aşamadaki c_j , diğer bütün c_k , $k \neq j$ 'i sabit tutarak $j = 0, 1, 2, \dots, K-1$ için hesaplanır ve daha sonra, tüm kod çizelgesi c , her aşamadaki kod çizelgelerinin bir fonksiyonu olarak yazılır.

$$\begin{aligned} c &= [c_0^T \ c_1^T \ \dots \ c_{j-1}^T \ c_j^T \ c_{j+1}^T \ \dots \ c_{K-1}^T] \\ c &= \bar{c}_j + S_j c_j \end{aligned} \quad (4.34)$$

$$\bar{c}_j = [c_0^T \ c_1^T \ \dots \ c_{j-1}^T \ 0_{p \times L_j}^T \ c_{j+1}^T \ \dots \ c_{K-1}^T] \quad (4.35)$$

$0_{p.L_j}$, $(L_j \times p)$ boyutlu sıfır vektörü, S_j kaydırma matrisidir. (4.34), (4.33)'de konarak yeniden düzenlenirse,

$$\begin{aligned} d_r &= d_{rj} - 2 \cdot c_j^T Y_j + c_j^T Q_{jj} c_j \\ d_{rj} &= d_0 - 2 \cdot \bar{c}_j^T Y + \bar{c}_j^T \cdot Q \cdot \bar{c}_j \\ Q_{jj} &= S_j^T \cdot Q \cdot S_j \\ Y_j &= S_j^T \cdot (Y - Q \cdot \bar{c}_j) \end{aligned} \quad (4.36)$$

j 'inci aşamadaki kod çizelgesi, $c_j = Q_{jj}^{-1} \cdot Y_j$ dir. Q_{jj} , bir köşegen matris olduğundan c_j 'yi bulmak çok kolaydır. 3 aşamalı kod çizelgesi için, örn. $j = 1$ alındığında $c_1 = [0]$, $S_1 = [[I] [0] [0]]$ ile $Y_1 = Y - Q_{12} \cdot c_2 - Q_{13} \cdot c_3$ ve $c_1 = Q_{11}^{-1} \cdot Y_1$ ile c_1 bulunur. c_2 ve c_3 kod çizelgeleri de benzer şekilde oluşturulmaktadır. (Q_{jj} 'ler köşegen matrisler olduğundan $j=1,2,3$ için $c_j = Y_j / Q_{jj}$ olarak hesaplanabilir.) [44,45,79]

Çizelge 4.4. Birleşik dizayn ÇAVN metodu algoritması

1. **Başlangıç:** $r = 1$. K aşamalı başlangıç kod çizelgesi oluşturun.
2. **Sınıflandırma:** Y ve Q matrislerini sıfırlayın. Sonra, eğitim giriş dizisindeki bütün vektörler için WMSE'yi minimum yapan indisleri belirleyin. $B_n^T w_n B_n$ ve $B_n^T w_n x_n$ 'i hesaplayın ve devam eden toplam Q , Y ve d_r 'i yeniden hesaplayın.
3. **Yeni Kod Çizelgesi Hesaplama:** $j: [0,1,...,K-1]$ için $c_j = Q_{jj}^{-1} \cdot Y_j$ 'yi hesaplayın.
4. **c 'nin yakınsaklığı:** adım 3'ü yakınsaklık oluşuncaya kadar tekrarlayın.
5. **Yeniden sıralama:** Kod çizelgesini her aşama için, enerjilerine göre yeniden sıralayın.
 $c_j^T c_j \leq c_k^T c_k$, $\forall j > k$, ve $\sum_{k=0}^{L_j-1} y_j^{(k)} = 0_p$, bütün kod çizelgesi toplamı sıfır olmalıdır. İlk kod çizelgesi, ortalama değer çıkarıldıktan sonra hesaplanır
6. **Yakınsaklık testi.** $[(d_{r-1} - d_r) / d_r] > \varepsilon$ ise $r = r+1$ ve adım 2'ye gidin.
7. **Bitiş.**

5. VEKTÖR VE MATRİS NİCEMLEMEDE GETİRİLEN YENİLİKLER

5.1. Giriş

Bu bölümde çok aşamalı vektör nicemleme metodu üzerinde yapılan bazı yenilikler ve oluşturulan kod tabloları üzerinde durulmuştur. Vektör nicemlemede kullanılan vektör için, bitişik pencereler arasındaki ilişkiden faydalanarak hesaplanan “artık vektör” hesaplamasında yeni bir metot geliştirilmiş ve bunun izgel distorsiyon ve aykırı değer sayısında düşme sağladığı gözlemlenmiştir. Matris nicemleme metodu, artık vektörün kullanıldığı çok aşamalı vektör nicemleme metodu üzerinde denenmiş ve artık çok aşamalı matris nicemleme metodu oluşturulmuştur. Böylece distorsiyon ve aykırı değerde önemli derecede azalma sağlanmıştır. Her vektör nicemleme metodu için oluşturulan kod çizelgelerinin distorsiyon ve aykırı değerleri çizelgeler ve grafikler halinde sunulmuştur.

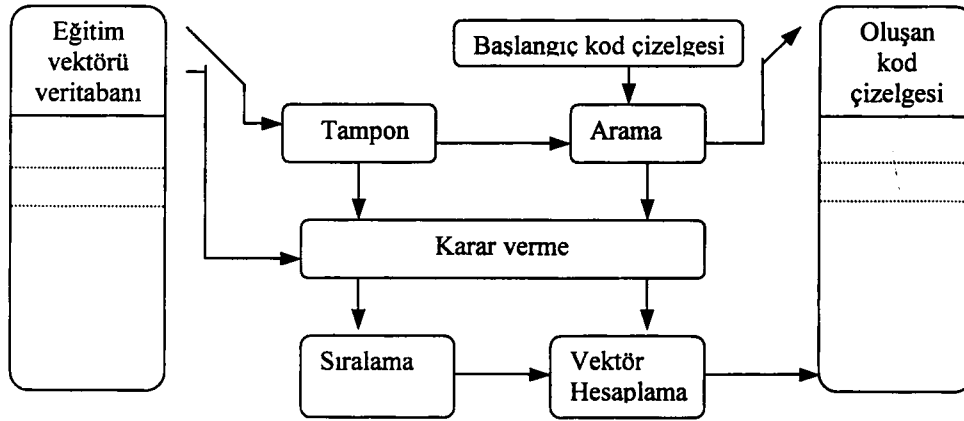
5.2. Kod Çizelgesi Oluşturma ve Eğitim

Kod çizelgesi dizayn iterasyonu, bir başlangıç kod çizelgesi içeren vektör nicemleyici ile başlar ve eğitim giriş dizisi kullanılarak en yakın komşu kuralına göre yeni bir kod çizelgesi oluşturulur. Bu işlem, ortalama kod çizelgesi distorsiyonu belli bir limitin altına ininceye kadar sürer. Şekil 5.1’de, kod çizelgesi dizayn işlemi blok gösterimi görülmektedir.

5.2.1. Eğitim giriş dizisi dizaynı

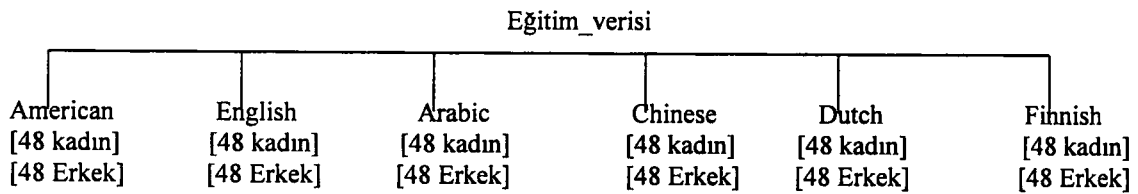
Farklı vektör nicemleyicilerin performansını karşılaştırmak için, bir ses veritabanı kullanılmıştır. LSF vektörleri izgel kodlamanın performans

sonuçları bir eğitim ve test dizisine dayalıdır. Eğitim vektör verileri için 16 kHz örnekleme frekansı ile örneklenmiş, filtrelenmiş, 8 kHz'e aşağı



Şekil 5.1. Kod çizelgesi dizayn işlemi blok gösterimi

örneklenmiş ve 100 Hz - 3800 Hz bant sınırlandırılmış ses dosyaları kullanılmıştır. Bu ses dosyaları, özel stüdyolarda hazırlanmış ve herbirinin 8 sn dinleme süresi olan, değişik milletlerden kadın ve erkek sesleri içeren ve söylenişi ve analizi zor özel cümlelerinin konuşulduğu, ses işleme uygulamaları için özel hazırlanmış, fonetik olarak dengelenmiş dosyalardır. Yaklaşık olarak toplam 60 dakika süreli ve erkek ve kadın konuşmacılardan oluşan 600 farklı ses kaydı içeren ses veri tabanının yarısı eğitim için, diğer yarısı test için ayrılmıştır. Bu ses dosyalarının bulunduğu veritabanı dizini Şekil 5.2'deki gibidir.



Şekil 5.2. Eğitim giriş dizisi için kullanılan ses dosyaları yerleşimi

Özilişki metoduna dayalı bir 10'uncu derece LP analiz, her 22.5 ms'n'de Hamming pencerelemiş ses parçaları üzerinde yapılmıştır. LPC spektrumunda keskin piklerden kaçınmak için bir 10 Hz bant genişlik açma ($\gamma = 0.996$) uygulanmıştır. Bu ses verileri ile oluşturulan eğitim vektör dizisi 65686 LSF vektörü içermektedir. Bu eğitim giriş dizisi ile çeşitli bit seviyelerinde kod çizelgeleri oluşturulmuştur. Her bit seviyesindeki kod çizelgesi distorsiyon ölçümleri yapılmış ve 9 bit seviyesinde dahi çok anlaşılır kodlama sonuçları elde edilmiştir. Test verisi için 20 dakika konuşma süreli ve eğitim vektör dizisindeki konuşmaların içerilmediği bir İngilizce diyalog kaydından çıkarılmış 28400 LSF vektör kullanılmıştır. (Ek 2).

5.2.2. Başlangıç kod çizelgesi dizaynı

Kod çizelgesi oluşturmak için, büyük miktarda giriş veri tabanı gerekmektedir. Etkili bir kod çizelgesi dizaynı için uygun bir başlangıç kod çizelgesi elde etmek önemli bir basamaktır. Başlangıç kod çizelgesi oluşturmak için birçok teknikler vardır.

Rasgele kodlama, kod çizelgesi elde etmede en uygun basamaktır. Eğitim giriş dizisi içinden rasgele seçilen başlangıç vektör değerleriyle kod çizelgesi dizayn işlemi başlatılır. Minimum distorsiyon hesabına göre LSF değerleri hesaplanarak ilk kod çizelgesi oluşturulur. Bu işlem, oluşturulan kod çizelgesi için hata değeri belli bir limit değeri altına ininceye kadar devam ettirilir.

Bu tezde oluşturulan kod çizelgeleri için, rasgele metoda benzer fakat biraz daha yapısal bir uygulama ile başlangıç LSF kod çizelgeleri oluşturulmuştur (Çizelge 5.1).

Çizelge 5.1 Oluşturulan kod çizelgeleri için kullanılan başlangıç kod çizelgesi hazırlama algoritması

- ADIM 1** Eğitim giriş dizisi vektörleri içinde ilk elemanlarının en küçük ve en büyük olduğu vektörleri referans olarak al
- ADIM 2** En büyük ve en küçük vektörlerin birinci elemanlarının farkını, başlangıç kod çizelgesi büyüklüğü N'ye oranla. Sonucu 'step' olarak belirle. Step2 = 0 olsun.
- ADIM 3** Step2 = Step2+Step/2
- ADIM 4** Her eğitim giriş vektörünün ilk elemanının Step2'den Step/2 kadar mesafe içinde olup olmadığını kontrol et. Bu mesafeler içinde kalan kod vektörü başlangıç kod vektörü olarak seç.
- ADIM 5** Adım 3'ü, N kod çizelgesi buluncaya kadar tekrarla.

5.3. Artık LSF Vektör Hesaplanması

ÇAVN yöntemi ile bilinen LSF kodlama tekniği yerine, iki analiz penceresi arasındaki LSF vektörlerinin farkı ile kod çizelgesi oluşturan yeni bir yöntem olan LSF artık (residual) sinyal metodu [49,80] değiştirilip düzenlenerek denenmiştir¹ [105,106]. Artık sinyal, bir önceki analiz penceresinde kodlanan LSF değerlerinin belli bir yüzdesinden o andaki ses penceresinde kodlanacak LSF değerlerinin farkı alınıp kodlanarak oluşturulmakta ve alıcı tarafta, vericide çıkarılan fark değeri tekrar eklenerek normal LSF değerine ulaşılmaktadır. Böylece LSF'ler üzerindeki ilişki en aza indirildiğinden daha az bitle kodlama mümkün olabilmektedir.

Artık sinyal için, eğitim giriş dizisindeki LSF vektörlerinin ortalamasıyla elde edilen LSF ortalama vektör I_{DC} , i 'inci pencere için I_i LSF vektörlerden çıkarılır ve LSF fark vektörü hesaplanır (Şekil 5.3).

$$d_i = I_i - I_{DC} \quad (5.1)$$

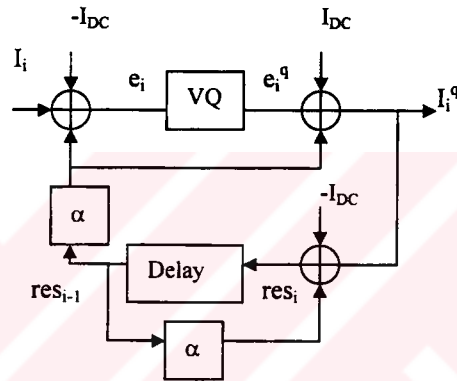
¹ Yazar, konuşma kodlama, vektör nicemleme, kod çizelgeleri hazırlama ve artık sinyal oluşturma tekniği için değerli görüşlerinden faydalandığı Sn. Prof.Dr. Ahmet Kondoza çok teşekkür eder.

Bitişik pencerelerdeki LSF vektörler arasındaki ilişki fazladır. Bu ilişki, “geri yönde öngörü – backward prediction” ile kaldırılabilir. Nandkumar metodu’nda [80], i ’inci pencere için LSF artık vektörü,

$$e_i = d_i - \alpha d_{i-1} \quad (5.2)$$

α , deneysel olarak 0.375 bulunmuştur. Alıcı kısımda, i ’inci pencere için nicemlenmiş I_i^q LSF vektörü,

$$I_i^q = e_i^q + I_{DC} + \alpha.(I_{i-1}^q - I_{DC}) \quad (5.3)$$



Şekil 5.3. Artık LSF kodlama şeması blok şeması

Yeni artık sinyal metodu için fark vektörü, geçmiş analiz pencerelerindeki bütün LSF’leri kapsayacak şekilde yeniden düzenlenmiştir:

$$e_i = d_i - \alpha.res_{i-1}, \quad res_0 = 0 \quad (5.4)$$

i ’inci pencere için nicemlenmiş I_i^q LSF vektörü,

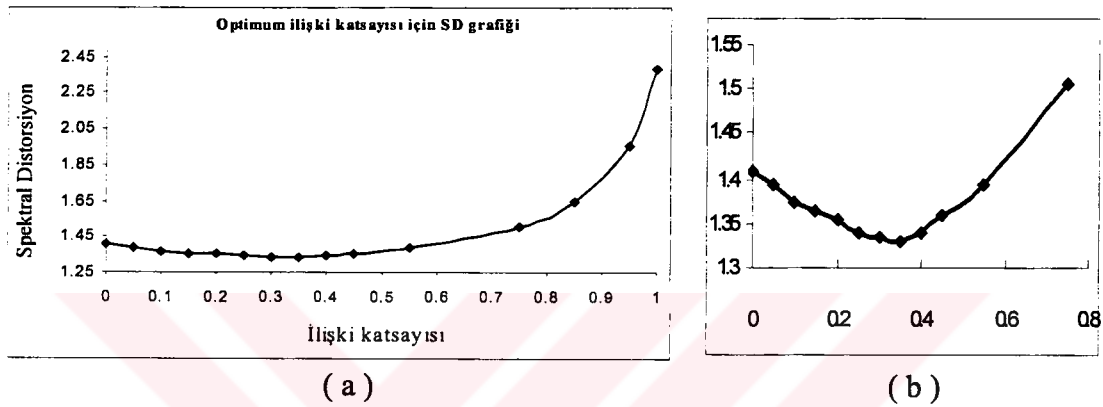
$$I_i^q = e_i^q + I_{DC} + \alpha.res_{i-1} \quad (5.5)$$

$$res_i = I_i^q - I_{DC} + \alpha.res_{i-1} \quad (5.6)$$

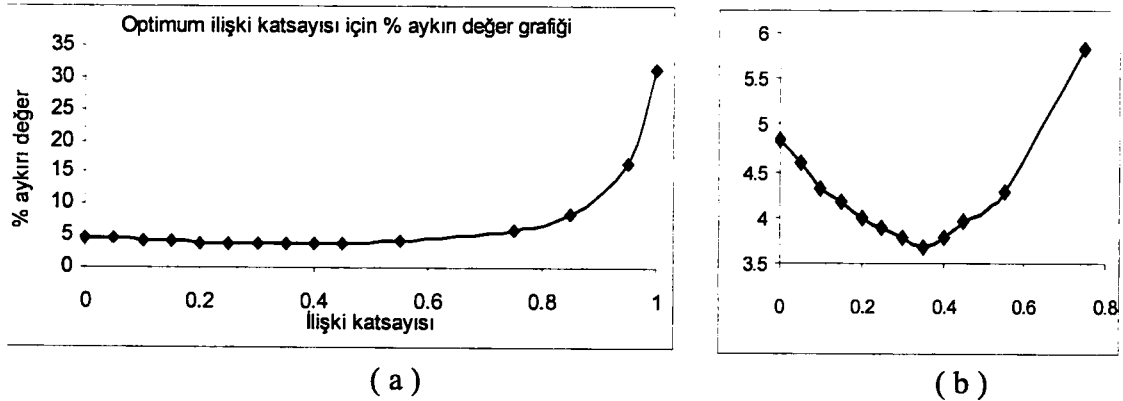
LSF artık vektörü e ve nicemlenmiş LSF artık vektörü $\hat{e}$ için distorsiyon ölçümü,

$$d(e, \hat{e}) = \sum_{j=1}^p w_j (e_j - \hat{e}_j)^2, \quad (5.7)$$

28400 LSF vektörü içeren eğitim giriş dizisi ile, α ilişki katsayısı (0-1) arasında 0.05 basamakla değiştirilip, [6,6,6]-3 aşamalı kod çizelgeleri oluşturulmuş ve SD ve aykırı değer hesaplanmıştır (Şekil 5.4 ve Şekil 5.5). Optimum SD için α : [0.30-0.35] arası bir değerde bulunmuştur. Burada yapılan inceleme ile $\alpha = 0.325$ olarak bulunmuştur.



Şekil 5.4. a) Artık sinyal metodunda, optimum ilişki katsayısı (α) için, α : [0.0-1.0] aralığında SD grafiği b) SD grafiğinin α : [0.0-0.75] aralığı için incelenmesi



Şekil 5.5. a) Artık sinyal metodunda, optimum ilişki katsayısı (α) için, α : [0.0-1.0] aralığında % aykırı değer grafiği b) % aykırı değer grafiğinin α : [0.0-0.75] aralığı için incelenmesi

5.4. İndirgenmiş Birleşik Kod Çizelgesi Metodu

Kod çizelgelerinin birleşik olarak eniyileştirilmesi sırasında, özellikle yüksek bit sayılı kod çizelgeleri için, ortaya büyük matrisler çıkmaktadır. Bu matrislerin önemli bir kısmı, fazla miktarda sıfır elemanı içermektedir. (Seyrek (sparse) matrisler). Hesaplamalar sırasında, bu gereksiz sıfır çarpımlarından kurtulmak ve hem hafızadan hem de zamandan tasarruf etmek için c ve Q matrisleri küçük bloklara ayrılıp, $c^T \cdot Q \cdot c$ matrisi üzerinde sürekli sıfır olan bölgeler yok edilmiş ve hesaplama kolaylığı için bütün matrisler vektör olarak düşünülmüştür [108]. Yeniden düzenlenmiş Y , Q ve c matrisleri, (4.33)'de d_r ifadesinde yerine konarak yeni bir distorsiyon formülü oluşturulmuştur. Sonuç olarak, $c^T \cdot Q \cdot c$, vektör çarpımlarının toplamına dönüşmüştür. Ayrıca yeni düzenlemede B_n indis seçme matrisi kullanılmayarak, bunun yerine ilgili indis değerleri için Y ve Q matrislerinin ilgili indis sırasına değerler direkt olarak ilave edilmektedir. Q_{ij} matrisleri ($L_i p \times L_j p$) köşegen matrisler, Q_{ij} matrisleri ise, ($p \times p$) lik köşegen matrisler içeren ($L_i p \times L_j p$) matrislerdir. Yeni düzenlemede Q_{ij} matrisleri, ($L_i p \times 1$)'lik köşegen elemanları içeren vektör olarak alınmıştır. Q_{ij} ($L_i p$ by $L_j p$) matrisleri ($L_i p \times L_j$) boyutunda ($(p \times p)$ 'lik köşegen matrisleri ($p \times 1$) vektörlere dönüştürülmüş) yeni bir matris olarak düzenlenmiş ve hesaplama kolaylığı açısından ($L_i p \cdot L_j \times 1$) boyutunda vektör olarak düşünülmüştür.

$$c^T \cdot Q \cdot c = \sum_{i=1}^K c_i^T \cdot Q_{ii} \cdot c_i + \sum_{i=1}^{K-1} \sum_{j=i+1}^K c_i^T Q_{ij} \cdot c_j + \sum_{i=1}^{K-1} \sum_{j=i+1}^K c_j^T Q_{ji} \cdot c_i \quad (5.8)$$

burada c_i ($L_i p \times 1$) i 'inci aşama için bir kod çizelgesidir. Y vektörü Y_i ($L_i p \times 1$) alt vektörlerine dönüştürülmüştür. Eğitim işlemi sırasında, bir K aşamalı ÇAVN kod çizelgesi dizaynı için, eğer s_i , i 'inci aşama için seçilmiş bir kod çizelgesi indeksi ve eğer lsf ve wcb vektörleri sırasıyla o anki p boyutlu giriş LSF eğitim vektörü ve ağırlık katsayıları ise, d_r için eğitim sırasında uyarlanan parametreler,

$$\sum_{i=1}^K \sum_{j=1}^p Y_i[s_i \cdot p + j] = \sum_{j=1}^p wcb[j] lsf[j] \quad (5.9)$$

$$\sum_{i=1}^K \sum_{j=1}^p Q_{ii}[s_i \cdot p + j] = \sum_{j=1}^p wcb[j] \quad (5.10)$$

$$\sum_{i=1}^{K-1} \sum_{j=i+1}^K \sum_{n=1}^p Q_{ij}[s_i \cdot p \cdot L_i + L_j + n] = \sum_{n=1}^p wcb[n] \quad (5.11)$$

$$d_0 = \sum_{i=1}^p lsf[i] \cdot wcb[i] \cdot lsf[i] \quad (5.12)$$

Yukarıdaki vektör toplamları bütün giriş eğitim vektörleri için yeniden değişir ve d_r için d_0 , Q ve Y vektörleri elde edilir. Eğer Q_{ii} vektörü içinde herhangi bir sıfır parametresi olursa, $c_i = Y/Q_{ii}$ için çözüm mümkün değildir. Bu yüzden, eğitim giriş dizisi içindeki ilgili giriş vektörü yeniden düzenlenmelidir. (0-1) arasında bir sabit sayı bu vektörle çarpılır. Bizim eğitim işlemi sırasında $\beta = 0.9$ alınmıştır ve yukarıdaki işlem Q_{ii} vektöründe bir sıfır olmayan durum oluşuncaya kadar tekrar ettirilir.

$$\sum_{i=1}^K \sum_{j=1}^{Lip} (if (Q_{ii}[j] == 0) \quad c_i[j] = c_i[j] \cdot \beta) \quad (5.13)$$

$C_{ij} = Q_{ij} \cdot c_j$ ve $C_{ji} = Q_{ji} \cdot c_i$ alarak, C_{ij} ve C_{ji} vektörleri aşağıdaki gibi hesaplanır:

$$\sum_{a=1}^{Li \cdot p} C_{ij}(a) = \sum_{a=1}^{Li} \sum_{b=1}^{Lj} \sum_{k=1}^p [C_{ij}(a \cdot p + k) + Q_{ij}((a \cdot p + k) \cdot L_j + b) \cdot c_j \cdot (b \cdot p + k)] \quad (5.14)$$

$$\sum_{a=1}^{Lj \cdot p} C_{ji}(a) = \sum_{a=1}^{Li} \sum_{b=1}^{Lj} \sum_{k=1}^p [C_{ji}(a \cdot p + k) + Q_{ji}((b \cdot p + k) \cdot L_i + a) \cdot c_i \cdot (b \cdot p + k)] \quad (5.15)$$

Yeni $c^T Q \cdot c$, d_r 'a uygulanır ve eğitim işleminde d_r hesaplanmasında büyük bir indirme elde edilir. d_r hesaplama karmaşası bir 18 bit/pencere VQ kod çizelgesi için yaklaşık olarak 15 kat azalır (5.16). Birleşik kod çizelgesi hesaplamadaki bütün diğer işlemler [44,45]'deki gibi uygulanır.

$$\sum_{i=1}^K L_i p \cdot L_i p \Rightarrow \sum_{i=1}^K L_i p + \sum_{i=1}^{K-1} \sum_{j=i+1}^K L_i p \cdot L_j \quad (5.16)$$

En büyük indirgeme Q matrisi üzerinde olmuştur. Böylece örneğin 3 aşamalı, (8,8,8) 24 bit, ve herbir vektör eleman değeri $p = 10$ olan bir kod çizelgesi için, indirgenmeden önceki Q matrisi eleman sayısı ($3 \times 256 \times 10 \times 3 \times 256 \times 10$) iken, indirgenme işleminden sonra ($3 \times 256 \times 10 + 6 \times 256 \times 256 \times 10$)'a düşmüştür. Burada yaklaşık 15 kat bir azalma söz konusudur. Böylece hem hafızadan, hem işlem zamanından büyük oranda tasarruf sağlanmıştır. Aynı metot kullanılarak değişik aşamalarındaki kod çizelgeleri için indirgenme yöntemi kolaylıkla uygulanabilmektedir.

5.4.1. LSF ve artık sinyal metodları ile kod çizelgesi hazırlanması

Kod çizelgeleri hazırlamada kullanılan eğitim giriş dizisindeki LSF vektör sayısı 65685'dir. Minimum değere yakınsaklık değeri $\varepsilon < 0.0001$ alınmıştır. Her bit seviyesinde, ağırlıklı distorsiyon d_r , izgel distorsiyon (spectral distortion-SD), % aykırı değer hesaplamaları ve kod çizelgesi oluşturulması için süren iterasyon sayıları (loop) incelenmiştir. SD hesaplanmasında (5.17) formülünden yararlanılmıştır ve [2-4dB] arası ve [>4 dB] için izgel distorsiyon değerleri incelenmiştir. (5.17) formülünde, a_k ve $a_{q,k}$, LPC filtre katsayılarına karşılık gelmektedir. T , eğitim giriş vektör sayısıdır. n_0 ve n_1 , 100Hz ve 3.8kHz frekanslarına karşılık gelir. $n_0=3$, $n_1=122$, $A(e^{j2\pi n/N})$ ve $A_q(e^{j2\pi n/N})$ 'i hesaplamak için $N=256$ alınmıştır. Kod çizelgeleri için M - L araştırma yöntemi kullanılarak, $M = 8$ kabul edilmiştir. Çizelgelerde dr_max ve dr_min değerleri, kod çizelgesi iterasyonuna başlamadan önceki ve iterasyon bittiğinde hesaplanan ağırlıklı distorsiyon değerlerini göstermektedir ve d_r hesaplanmasında (4.29) formülünden yararlanılmıştır.

$$SD = \sqrt{\frac{1}{T} \sum_{i=1}^T d_{SD}^2(e^{j2\pi i/N})} \quad dB^2 \quad (5.17)$$

Burada,

$$A(z) = 1 + \sum_{k=1}^p a_k z^{-k}, \quad A_q(z) = 1 + \sum_{k=1}^p a_{q,k} z^{-k},$$

$$d_{SD}(A_q(z), A(z)) \approx \sqrt{\frac{1}{n_1 - n_0} \sum_{n=n_0}^{n_1-1} \left[10 \log_{10} \left(\frac{|A_q(e^{j2\pi n/N})|^2}{|A(e^{j2\pi n/N})|^2} \right) \right]^2}$$

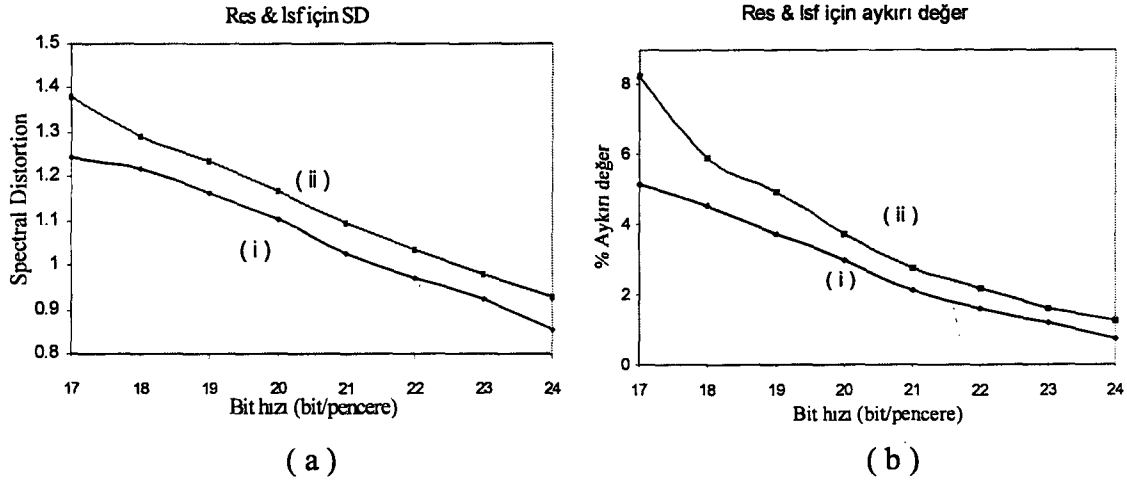
dir. Birleşik dizayn ÇAVN metodu ve birleşik dizayn yeni artık LSF ÇAVN metodu (res) kullanılarak değişik bit seviyelerinde oluşturulan 3-aşamalı kod çizelgeleri sırasıyla Çizelge 5.2 ve Çizelge 5.3'de görülmektedir. Bu çizelgelerin grafikleri ise Şekil 5.6'da görülebilir.

Çizelge 5.2. Birleşik dizayn LSF kodlama tekniği ile değişik bit seviyelerinde hazırlanmış vektör kod çizelgeleri

BİRLEŞİK DİZAYN LSF vektör İzgel Distorsiyon (SD) ve % Aykırı Değer						
Bit sayısı	SD	%[2-4dB]	%[>4dB]	dr_max	dr_min	loop
[665] - 17	1.3793	8.1767	0.0518	0.005054	0.000804	127
[765] - 18	1.2928	5.8597	0.0274	0.004392	0.000693	102
[766] - 19	1.2341	4.9295	0.0228	0.004412	0.000623	118
[776] - 20	1.1665	3.6934	0.0289	0.004308	0.000544	142
[876] - 21	1.0930	2.7540	0.0137	0.003253	0.000467	102
[877] - 22	1.0361	2.1907	0.0137	0.003252	0.000413	95
[887] - 23	0.9787	1.6214	0.0076	0.003181	0.000363	93
[888] - 24	0.9289	1.2834	0.0061	0.003180	0.000324	81
[888] test_63	1.0033	2.1033	0.0193	*	*	*
[7666] melp	1.4009	7.2605	0.0000	*	*	*

Çizelge 5.3. Birleşik dizayn artık LSF kodlama tekniği ile değişik bit seviyelerinde hazırlanmış vektör kod çizelgeleri

BİRLEŞİK DİZAYN LSF artık (RES) vektör İzgel Distorsiyon ve % Aykırı Değer						
Bit sayısı	SD	%[2-4dB]	%[>4dB]	dr_max	dr_min	loop
[665] - 17	1.2458	5.1335	0.0487	0.003212	0.000638	44
[765] - 18	1.2198	4.4743	0.0350	0.003666	0.000608	56
[766] - 19	1.1647	3.6949	0.0213	0.004340	0.000545	66
[776] - 20	1.101	2.9550	0.0198	0.003547	0.000474	49
[876] - 21	1.0261	2.1009	0.0122	0.002579	0.000403	48
[877] - 22	0.9715	1.6229	0.0091	0.002578	0.000356	44
[887] - 23	0.9255	1.2011	0.0046	0.003047	0.000321	40
[888] - 24	0.8568	0.7673	0.0030	0.003047	0.000273	77



Şekil 5.6. Birleşik dizayn LSF kodlama ve artık LSF kodlama teknikleri ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri için (a) SD ve (b) [2-4 dB] arası %aykırı değer grafikleri (i) : birleşik kod çizelgesi dizayn metodu ile artık ÇAVN (ii): birleşik kod çizelgesi dizayn metodu ile ÇAVN)

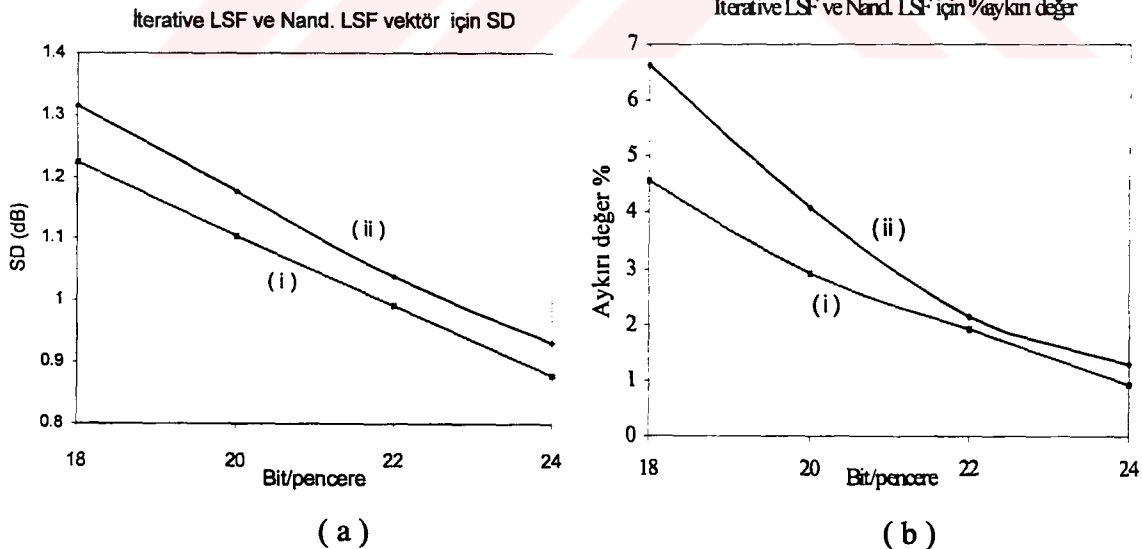
Özyineli sıralı dizayn ÇAVN metodu ve birleşik dizayn Nandkumar LSF ÇAVN metodu (Nand) kullanılarak değişik bit seviyelerinde oluşturulan 3- aşamalı kod çizelgeleri sırasıyla Çizelge 5.4 ve Çizelge 5.5'de görülmektedir. Bu çizelgelerin grafikleri ise Şekil 5.7'de görülebilir. Nandkumar metodu ile Birleşik dizayn LSF metodunda (8,8,8) 24 bit kod çizelgesi için, 63bin LSF vektör içeren test giriş dizisi ve MELP konuşma kodlayıcının kod çizelgesi ile SD hesaplamaları yapılmıştır. Yeni artık sinyal metodu ile oluşturulan kod çizelgeleri için, daha az bit seviyelerinde daha belirgin olmak üzere her bit seviyesindeki hesaplamalarda belirgin bir azalma görülmüştür. Ayrıca, minimum değere yakınsaklık hızı, normal LSF kodlama yöntemine göre iki kat daha fazla olmuştur. Yeni ve eski kodlama çizelgeleri, bir konuşma kodlama programı üzerinde denenerek, sonuç sesler dinlenmiştir. Yeni metotla kodlanan seslerde, anlaşılabilirlik daha fazla olmuştur.

Çizelge 5.4. Özyineli sıralı LSF kodlama tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri

Özyineli sıralı LSF vektör İzgel Distorsiyon ve % Aykırı Değer						
bits_total	SD	% [2-4dB]	% [>4dB]	dr_max	dr_min	loop
[765]-18	1.3151	6.5874	0.0579	0.004418	0.000723	148
[776]-20	1.1781	4.0526	0.0274	0.004345	0.000558	152
[877]-22	1.0386	2.1177	0.0152	0.003265	0.000415	155
[888]-24	0.9326	1.2879	0.0046	0.003745	0.000326	122
[7666]-24	1.0119	1.9685	0.0046	0.005053	0.000391	174

Çizelge 5.5. Birleşik dizayn Nandkumar artık LSF vektör dizayn tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri

Birleşik dizayn NANDKUMAR artık LSF vektör SD ve % Aykırı Değer						
bits_total	SD	% [2-4dB]	% [>4dB]	dr_max	dr_min	Loop
[765]-18	1.2231	4.5322	0.0274	0.003459	0.000611	57
[766]-19	1.1781	3.7877	0.0213	0.004217	0.000559	44
[776]-20	1.1037	2.8881	0.0228	0.003599	0.000478	44
[876]-21	1.0466	2.3461	0.0137	0.029408	0.000424	66
[877]-22	0.9931	1.8938	0.0183	0.029406	0.000376	61
[887]-23	0.9306	1.3215	0.0076	0.028950	0.000324	73
[888]-24	0.8789	0.9317	0.0046	0.028946	0.000288	55



Şekil 5.7. Özyineli sıralı LSF kodlama ve Nandkumar artık LSF kodlama teknikleri ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri için (a) SD ve (b) %aykırı değer grafikleri ((i) : özyineli sıralı kod çizelgesi dizayn metodu ile Nandkumar ÇAVN (ii): özyineli sıralı kod çizelgesi dizayn metodu ile ÇAVN)

Eğitim verisi içinde, *aykırı değer* (outlier) denilen ve eğitim verisi içinde çok az temsil edilen ve ortalamadan çok yüksek spektral distorsiyonla nicemlenen vektör değerlerinin etkisini azaltmak için distorsiyon hesabı sırasında kullanılan $d_n = (x_n - x_n)^T \cdot W_n \cdot (x_n - x_n)$ yerine $d_n = SD^p \cdot (x_n - x_n)^T \cdot W_n \cdot (x_n - x_n)$ ($0 \leq p \leq 5$) formülü ile izgel distorsiyona göre hücresel vektör sınıflandırması yapılarak, aykırı değerlerin etkisi azaltılmaya çalışılmıştır. p burada, [0-5] arası değiştirilerek her bit seviyesinde etkisine bakılmıştır. p büyüdükçe aykırı değer sayısı azalmakta, fakat tüm kod çizelgesine ait ortalama SD değeri büyümektedir. Burada örn. 18 bit için, 122695 eğitim giriş verisi için $p=0$ iken, 2dB den büyük aykırı değer sayısı 5243 ve $SD = 1.167642$ iken $p = 4$ için 2dB den büyük aykırı değer sayısı 5075 e düşmüş fakat $SD = 1.314912$ olmuştur. Bu distorsiyon değeri, normal LSF kodlamadaki 24 bit kodlama seviyesindeki SD ye çok yakın değerlerdir. Değişik kod çizelgesi dizayn metotları için ağırlıklı distorsiyon sonuçları ve ağırlıklı SD hesaplamaları Ek-2’de bulunabilir.

5.5. Matris Nicemleme

Matris nicemleme (matrix quantization), ses formant parametrelerinin orijinale çok benzer şekilde ve düşük bit sayılarında kodlanabilmesi için kullanılan nicemleme metodudur ve vektör nicemlemeye göre önemli bit azalması sağlamaktadır. Burada, bitişik pencerelerdeki ses sinyalleri arasındaki ilişkinin yüksek olması özelliğinden hareketle, birkaç pencere bir araya getirilerek kodlama yöntemi kullanılmaktadır. Kod çizelgesindeki uzama bir dezavantaj gibi görülmesine rağmen, ortalama olarak her penceredeki bit sayısını azaltıp, distorsiyonda çok fazla artışa neden olmaması, düşük bit hızlarına ulaşabilmek için bir avantaj sağlamaktadır. Matris nicemleme metodu, daha düşük bit hızlarında fazla distorsiyon artışına neden olmadan kod çizelgesi dizayn edilmesini sağlamaktadır. Örneğin 350 bit/sn de işletilen bir tam aramalı (full-

search) vektör nicemleyici ile 150 bit/sn'de işletilen LPC matris nicemleyici eşdeğer distorsiyona sahiptir. Böylece daha düşük bit hızlarında veya daha yüksek kaliteli kod çizelgesi dizaynları matris nicemleyici ile gerçekleştirilebilir. [82]

5.5.1 Artık çok aşamalı matris nicemleme metodu

Bir matris nicemleme sisteminde, M ses penceresi için LPC vektörlerden oluşmuş M-matrisi gruplanır ve GLA ((generalized Lloyd algorithm) bu matris sırasına uygulanır [81,83]. Tekrar üretilen matrislerin kod çizelgesi bir matris kod çizelgesi olarak adlandırılır. Bir i 'inci ses penceresi için, $a_i = (aa_0^i, aa_1^i, \dots, aa_{p-1}^i)^T$ LPC katsayılarının vektörü olsun. a_i , artık LSF vektör $x_i = (xx_0^i, xx_1^i, \dots, xx_{p-1}^i)^T$ e çevrilir. Sonra eğer M ≥ 1 bir tamsayı ve LPC filtre derecesi p ise, pxM boyutlu matris X,

$$X = \begin{bmatrix} xx_0^i & xx_0^{i+1} & \dots & xx_0^{i+M-1} \\ xx_1^i & xx_1^{i+1} & \dots & xx_1^{i+M-1} \\ \vdots & \vdots & \dots & \vdots \\ xx_{p-1}^i & xx_{p-1}^{i+1} & \dots & xx_{p-1}^{i+M-1} \end{bmatrix}_{pxM} = \begin{bmatrix} x_i \\ x_{i+1} \\ \vdots \\ x_{i+M} \end{bmatrix}^T \quad (5.18)$$

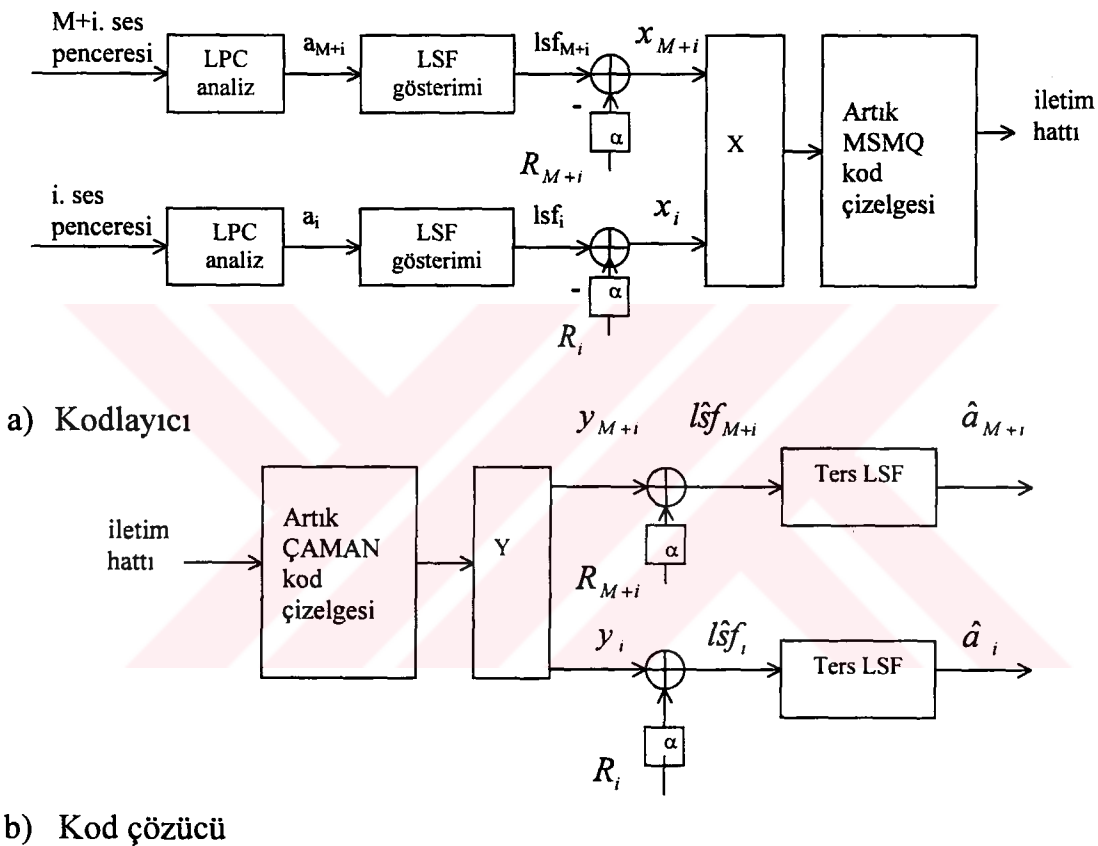
ve nicemlenmiş vektörlerden oluşan matris Y,

$$Y = \begin{bmatrix} yy_0^i & yy_0^{i+1} & \dots & yy_0^{i+M-1} \\ yy_1^i & yy_1^{i+1} & \dots & yy_1^{i+M-1} \\ \vdots & \vdots & \dots & \vdots \\ yy_{p-1}^i & yy_{p-1}^{i+1} & \dots & yy_{p-1}^{i+M-1} \end{bmatrix}_{pxM} = \begin{bmatrix} y_i \\ y_{i+1} \\ \vdots \\ y_{i+M} \end{bmatrix}^T \quad (5.19)$$

$$R_i = I_{DC} + \alpha \cdot res_{i-1} \quad (5.20)$$

$$res_i = l\hat{s}f_i - I_{DC} + \alpha \cdot res_{i-1} \quad (5.21)$$

Şimdi, her aşama M analiz penceresi üzerinde belirlenen X matrisindeki izgel katsayılar kullanılarak oluşturulabilir. Şekil 5.8, artık çok aşamalı matris nicemleyici-artık ÇAMAN (residual multi stage matrix quantization – residual MSMQ) blok yapısını göstermektedir. Her aşama için, aşamaların sayısı ve kod çizelgesi uzunlukları karmaşa/depolama durumuna göre keyfi olarak seçilebilir.



Şekil 5.8. Artık ÇAMAN kodlayıcı ve kod çözücü şeması

Bir WMSE (weighted mean square error) distorsiyon kriteri [79] kod çizelgesi eğitimi ve kod vektörlerin seçimi sırasında kullanılmaktadır. Orijinal ve nicemlenmiş parametreler arasındaki WMSE,

$$D(X, Y) = \frac{1}{M} \sum_{i=1}^M d(x_i, y_i) \quad (5.22)$$

$$d(x_i, y_i) = (x_i - y_i)^T w_i (x_i - y_i) \quad (5.23)$$

Eğer eğitim giriş dizisi için GLA, $d(x_i, y_i)$ için bölgesel bir minimuma yakınsıyorsa, $d(x_i, y_i)$ yakınsaktır (convex). Aynı zamanda, $d(x_i, y_i)$ 'lerin toplamı $D(X, Y)$ yakınsaktır. Böylece $D(X, Y)$ için GLA eğitim giriş dizisi için bölgesel bir minimuma yakınsar.

$$\min_Y E(D(X, Y)) = \frac{1}{M} \sum_{j=1}^M \min_{y_i} E(d(x_i, y_i)) \quad (5.24)$$

Yani, X matrislerinin hücrelerinin ağırlık merkezi (centroid), X içindeki bağımsız x_i 'lerin ağırlık merkezlerinden oluşur. Pratikte, *estimation* E , hücrenin aritmetik ortalaması olarak ifade edilir. Bir x_i vektörünün ağırlık merkezi, örnek özilişkiler $r_x(j)$, $j=1, \dots, p$ 'nin aritmetik ortalaması olarak ifade edilir. B tekrar üretim matrisi (kod çizelgesi) olsun. $Y = \{Y_j, j=1, \dots, B\}$ kod kelimesi ve $X = \{X_j, j=1, \dots, K\}$ bir LPC matrislerden oluşan eğitim giriş dizisi olsun. Eğitim sırası içindeki her bir X matrisi kod kelimesi Y 'ye minimum $D(X, Y)$ hesabı ile eşlenir. Eğitim sırası X , B hücreye bölünür. Her bir hücre için ağırlık merkezi $\bar{x}$, örnek özilişki katsayılarının ortalaması alınarak bulunur.

$$\bar{x} = [\bar{x}_1, \dots, \bar{x}_M] \quad (5.25)$$

$$\bar{x}_j = \frac{1}{L} \sum_{i=1}^L r_{xji}(k), \quad k=1, \dots, p, \quad j=1, \dots, M \quad (5.26)$$

L , hücredeki matrislerin sayısıdır. Centroid $\bar{x}$ 'lerden oluşan kod çizelgesi Y elde edilir ve bu işlem, ortalama distorsiyondaki yüzde değişim, bir eşikten daha küçük olana kadar sürer.

$$D_{avg} = \frac{1}{K} \sum_{j=1}^K \min_{Y \in \{Y\}} D(X_j, Y) \quad (5.27)$$

Böylece bölgesel bir minimuma yaklaşılr.

X matrislerinin eğitim sırası ilk matrisi ses pencerelerinin eğitim sırası boyunca kaydırarak elde edilir, öyle ki sıradaki j 'inci matris $X_j = [x_j, x_{j+1}, \dots, x_{j+M-1}]$ dir. (bu kaydırma blok tekniği ile eğitim giriş dizisindeki matrislerin en büyük kombinasyonu elde edilir ve eğitim giriş dizisi için kullanılan ses datasındaki bütün fonem geçişleri içerilir). M LPC penceresinin test sırası $[t_1, t_2, \dots, t_j, \dots]$, $T_j = (t_{M(j-1)+1}, t_{M(j-1)+2}, \dots, t_{Mj})$ matris sırası içine gruplanır, burada $t_j = (tt_0^j, tt_1^j, \dots, tt_{p-1}^j)^T$. Sonra, tekrar üretim matrisi Y minimum distorsiyon hesaplanmasına göre elde edilir ve T_j test sırasındaki j 'inci matris için aktarılan kod indeksi m :

$$m : D(T_j, Y_m) \leq D(T_j, Y_k), \quad \{Y_k \in [Y: Y_1, Y_2, \dots, Y_B]\} \quad (5.28)$$

burada B , Y kod çizelgesinin uzunluğudur.

5.5.2. Artık çok aşamalı matris nicemleme metodu ile kod çizelgesi hazırlanması

Artık ÇAMAN kod çizelgesi metodunda, ÇAVN algoritmasından faydalanıldı. Matris nicemleme için birden fazla pencere parametreleri birleştirildi. Artık ÇAMAN'a birleşik kod çizelgesi ve özyineli sıralı kod çizelgesi dizayn metotları uygulandı. Nicemlenen parametreler, ses pencereleri içindeki artık LSF parametreleridir.

8 kHz örneklenmiş ses sinyali için 100-3800Hz frekans bandı üzerinde SD hesaplanmıştır. Eğitim giriş dizisi (65685 vektör) filtrelenmiş ve 8 kHz örneklenmiş İngilizce cümlelerdir. Artık ÇAMAN kod çizelgesi için, iki yanyana ses penceresi bir süper pencereye gruplanmış ve birleşik olarak nicemlenmişlerdir. Birleşik dizayn ÇAVN ve birleşik dizayn Artık ÇAMAN kod çizelgeleri SD sonucu ve aykırı değer hesaplanması çizelgelerde görülebilir.

Çizelge 5.6. Birleşik dizayn LSF vektör kodlama tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri

Birleşik dizayn LSF vektör kodlama ile İzgel Distorsiyon (SD) ve % Aykırı Değer						
Bit sayısı	SD	% [2-4dB]	% [>4dB]	dr_max	dr_min	loop
[333]-9	2.2526	47.6722	2.0233	0.0277060	0.0025728	82
[433]-10	2.1130	40.5703	1.3123	0.0094530	0.0022140	89
[443]-11	1.9989	35.0867	0.8251	0.0091760	0.0019160	90
[444]-12	1.8843	29.4781	0.4628	0.0091750	0.0016720	105
[544]-13	1.7605	22.9775	0.3182	0.0068990	0.0014260	87
[554]-14	1.6536	18.2764	0.1796	0.0067740	0.0012370	123
[555]-15	1.5622	14.4795	0.1248	0.0067720	0.0010830	118
[655]-16	1.4615	10.8334	0.0685	0.0051670	0.0009260	95
[665]-17	1.3793	8.1767	0.0518	0.0050540	0.0008040	127
[765]-18	1.2928	5.8597	0.0274	0.0439200	0.0006930	102

Çizelge 5.7. Birleşik dizayn LSF MATRİS kodlama tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri

Birleşik dizayn LSF MATRİS kodlama ile İzgel Distorsiyon (SD) ve % Aykırı Değer						
Bit sayısı	SD	out[2-4dB]	out[>4dB]	dr_max	dr_min	loop
[765]-18	1.9449	34.7858	0.1720	0.008581	0.001817	100
[776]-20	1.8226	27.6988	0.0594	0.008488	0.001562	102
[877]-22	1.7110	21.3641	0.0320	0.007158	0.001348	72
[888]-24	1.6052	15.8377	0.0122	0.007089	0.001160	96

Çizelge 5.8. Özyineli sıralı dizayn LSF Nandkumar MATRİS kodlama tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri

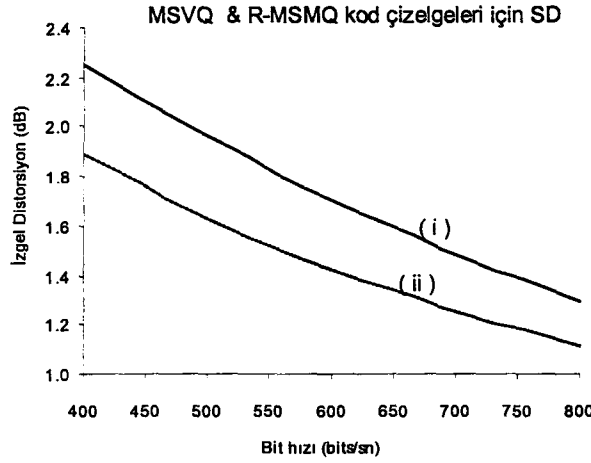
Özyineli sıralı dizayn LSF Nandkumar MATRİS kodlama ile İzgel Distorsiyon (SD) ve % Aykırı Değer						
Bit sayısı	SD	out[2-4dB]	out[>4dB]	dr_max	dr_min	loop
[995544]-36	1.1418	3.2567	0.0000	0.004723	0.000598	63
[10-998]-36	1,1192	1,5673	0,0000	0.004492	0.000503	56

Çizelge 5.9. Özyineli sıralı dizayn LSF MATRİS kodlama tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri

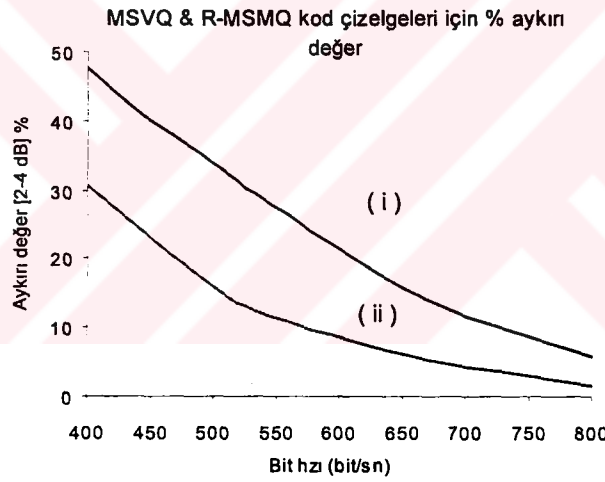
Özyineli sıralı dizayn MATRİS LSF kodlama ile İzgel Distorsiyon (SD) ve % Aykırı Değer						
Bit sayısı	SD	out[2-4dB]	out[>4dB]	dr_max	dr_min	loop
[765]-18	1.9627	35.9443	0.2055	0.007431	0.001860	115
[776]-20	1.8316	28.1525	0.0609	0.007181	0.001583	122
[877]-22	1.7176	21.6168	0.0350	0.006164	0.001360	137
[888]-24	1.6153	16.3447	0.0122	0.006104	0.001178	86
[6666]-24	1.6801	18.9176	0.0365	0.008449	0.001294	186
[8877]-30	1.3888	7.0702	0.0030	0.006104	0.000829	118
[10-998]-36	1.1329	1.9944	0.0000	0.004295	0.000516	102

Çizelge 5.10. Özyineli sıralı dizayn artık LSF MATRİS kodlama tekniği ile değişik bit seviyelerinde hazırlanmış kod çizelgeleri

Özyineli sıralı dizayn artık LSF MATRİS kodlama ile İzgel Distorsiyon (SD) ve % Aykırı Değer							
Bit sayısı	Bit hızı (bit/sn)	SD	% [2-4dB]	% [>4dB]	dr_max	dr_min	loop
[765]-18	400	1.8882	30.5032	0.1355	0.007041	0.001703	47
[776]-20	444	1.7821	24.1196	0.0594	0.006968	0.001485	30
[877]-22	489	1.6549	17.2627	0.0289	0.006157	0.001249	49
[888]-24	533	1.5495	12.3392	0.0122	0.006073	0.001071	46
[8877]-30	666	1.3161	5.2385	0.0025	0.005732	0.000625	54
[10-9-9-8]-36	800	1.1109	1.4189	0.0000	0.004365	0.000492	72
[995544]-36 $\epsilon < 0.00005$	800	1.1247	1.8524	0.0000	0.000598	0.000572	21
[10-998]-36 $\epsilon < 0.00005$	800	1,1023	1,3246	0,0000	0.000503	0.000485	32



Şekil 5.9. Kod çizelgeleri bit hızları ve izgel distorsiyon (SD) arasındaki bağlantı grafiği i) Birleşik dizayn tekniği ile ÇAVN ii) Özyineli sıralı kod çizelgesi dizayn tekniği ile Artık ÇAMAN



Şekil 5.10. Kod çizelgeleri bit hızları ve [2-4 dB] arası % aykırı değer arasındaki bağlantı grafiği i) Birleşik dizayn tekniği ile ÇAVN ii) Özyineli sıralı kod çizelgesi dizayn tekniği ile Artık ÇAMAN

Çizelge 5.10, özyineli sıralı kod çizelgesi dizayn tekniği ile 22.5 msn'lik analiz pencereleri kullanılarak dizayn edilen 400 bit/sn ve 800 bit/sn Artık ÇAMAN kod çizelgeleri için aykırı değer ve SD sonuçlarını göstermektedir. Şekil 5.9 ve Şekil 5.10'deki simulasyon sonuçları göstermektedir ki, Artık ÇAMAN şeması, spektral distorsiyon ve aykırı değer bakımından ÇAVN

metodundan çok daha iyi sonuçlar vermektedir. Dizayn edilen Artık ÇAMAN kod çizelgeleri, matris nicemleme tabanlı kodlayıcı (MQBC) üzerinde denenmiştir. Test ses datası için, dinleme testleri, SD sonuçları ve aykırı değer sonuçlarına göre, 18 bit/pencere Artık ÇAMAN kod çizelgesi 22 bit/pencere ÇAVN kod çizelgesi ile aynı kaliteyi vermiştir.

5.6. Kod Çizelgesi Arama Karmaşası

Kod çizelgesi arama karmaşası, nicemlenmiş parametreleri elde etmek için gerekli aritmetik işlemlerin sayısı olarak belirlenir ve genellikle logaritmik olarak ifade edilir. Bir ağacın farklı düğümlerindeki distorsiyon hesaplanması pekçok ortak elemana bağlıdır ve eğer M her aşamadaki en düşük distorsiyonu veren kod vektör sayısı ise, bulunan eniyi yol sayısı M 'den daha az olabilir. j ($j \neq 0$) aşamadaki yol sayısı ise;

$$\hat{M}_j = \min \left(M, \prod_{k=0}^{j-1} L_k \right) \quad (5.29)$$

Hesapsal karmaşayı hesaplamak için, m 'inci aday yaklaşık vektör $\hat{x}^{(m,j)}$ olarak alınsın. x ve $\hat{x}^{(m,j)}$ arasındaki ağırlıklı hata:

$$u^{(m,j)} = x - \hat{x}^{(m,j)} \quad (5.30)$$

$$d(x, \hat{x}) = u^{(m,j)T} W u^{(m,j)} - 2u^{(m,j)T} W y_j^{(l_j)} + y_j^{(l_j)T} W y_j^{(l_j)} \quad (5.31)$$

(4.63)'de ilk terim ana düğümden minimum distorsiyondur ve bilinir; ikinci terim, $(\min(L_j, \hat{M}_j) \cdot p + \hat{M}_j p \cdot L_j)$ çarpma-toplama ve son terim $2 \cdot p \cdot L_j$ çarpma-toplama gerektirir. Toplam karmaşa:

$$C_{MSVQ} = \log_2 \left(3pL_0 + p \sum_{j=1}^{K-1} (\min(L_j, \hat{M}_j) + (\hat{M}_j + 2)L_j) \right) \quad (5.32)$$

ÇAMAN için p yerine p_x (analiz pencere sayısı) konularak hesaplama yapılır. ÇAVN için hafıza gereksinimi:

$$M_{MSVQ} = p \sum_{j=0}^{K-1} L_j \quad (5.33)$$

$p=10$ ve $M = 8$ alınarak, [7666] şeklinde 4 aşamalı MELP kod çizelgesi için arama karmaşası:

$$C = \log_2(3 \cdot 10 \cdot 128 + 10 \cdot \sum_{j=1}^3 (8 + (8 + 2) \cdot L_j)) = \log_2(23280) = 14,506$$

Hafıza gereksinimi ise (4.65) formülünden $M = 3200$ çıkar.

Aynı şekilde [995544] şeklinde 6 aşamalı Artık ÇAMAN kod çizelgesi için arama karmaşası:

$$C = \log_2(3 \cdot 10 \cdot 2 \cdot 512 + 10 \cdot 2 \cdot \sum_{j=1}^5 (8 + (8 + 2) \cdot L_j)) = \log_2(76560) = 16,224$$

Hafıza gereksinimi $M=10 \cdot 2 \cdot (2 \cdot 512 + 2 \cdot 32 + 2 \cdot 16)=22400$ çıkar.

[10-998] şeklinde 4 aşamalı Artık ÇAMAN kod çizelgesi için arama karmaşası ise:

$$C = \log_2(3 \cdot 10 \cdot 2 \cdot 1024 + 10 \cdot 2 \cdot \sum_{j=1}^3 (8 + (8 + 2) \cdot L_j)) = \log_2(158960) = 17,278$$

Hafıza gereksinimi $M=10 \cdot 2 \cdot (1024 + 2 \cdot 512 + 256)=46080$ çıkar.

6. MATRİS NİCEMLEME TABANLI KONUŞMA KODLAYICI

6.1. Giriş

Bu bölüm MELP konuşma kodlayıcı ve matris nicemleme tabanlı konuşma kodlayıcı (MQBC) sistemleri üzerinde durmakta ve her iki kodlayıcıyı karşılaştırmaktadır. MQBC kodlayıcı, 2400 bit/sn MELP kodlayıcının yarı hızında çalışabilmekte ve ses kalitesi bakımından MELP kodlayıcıya yaklaşmaktadır. MQBC kodlayıcıda getirilen bazı yenilikler, artık çok aşamalı matris nicemleme metodu ile hazırlanmış kod çizelgesi kullanımı, 1 bitlik karma uyartım sinyali, perde periyodu ve sesli/sessiz kararlarının birleşik olarak kodlanması, MELP kodlayıcıda bulunan *aperiyodik bayrak* (aperiodic flag) bilgisinin kodlanmasına gerek olmadan sentezleyici bölümde, analiz parametrelerinden çıkarılması gibi metotlar uygulanmıştır.

6.2. MELP Kodlayıcı Sistemi

Doğrusal öngörülü kodlama metoduna dayalı LPC kodlayıcı, konuşma kodlama için tamamen parametrik model kullanır. Sentezleyici bölüm (decoder), sesi tekrar oluşturmak için sadece aktarılan parametreleri kullanır. Bu yüzden çoğunlukla ses çıkışı doğal insan sesine benzemez. Sentezlenmiş sese vızıltı (buzz), gümbürtü (thump), tonal gürültü gibi rahatsız edici sonuçlar vardır. 1995’de McCree ve Barnwell [93] tarafından karma uyartımlı yeni bir kodlayıcı olan MELP konuşma kodlayıcı önerilmiştir. MELP kodlayıcı geleneksel LPC kodlayıcıya benzemesine rağmen, yukarıdaki olumsuz etkileri azaltıcı beş ilave özellik içerir. Bu özellikler, MELP kodlayıcının insan doğal sesine yakın çıkış sinyali vermesine neden olurlar:

1. Karma darbe (mixed pulse) ve gürültü uyarımı
2. Peryodik ya da periyodik olmayan darbeler
3. Uyarlamalı izgel zenginleştirme (adaptive spectral enhancement-ASE) filtresi
4. Darbe yayılım (pulse dispersion) filtresi
5. Fourier genlikler

Karma uyarım, frekans bantlara ayrılan sinyalin sesli ya da sessiz durum analizine göre karma uyarım sinyali uygulanmasıdır. Karma uyarım LPC kodlayıcılarda özellikle akustik gürültü ortamlarında görülen vızıltıyı (buzz) azaltır. MELP kodlayıcı periyodik ya da periyodik olmayan darbeler sesli giriş sinyallerini sentezlemede kullanılırlar [89-94].

Peryodik olmayan darbeler, sesli ve sessiz ses sinyal parçaları arasındaki geçiş bölgelerinde kullanılırlar. Bu özellik kodlayıcının tonlu sesler yerine kararsız glottal darbeler üretmesine neden olur.

ASE filtre, doğrusal öngörülü sentez filtresinin kutuplarına bağlıdır. Sentetik sesin formant yapısını zenginleştirmede ve sentetik ve doğal bant geçiş dalga şekilleri arasındaki benzerliği arttırmada kullanılır. Sentetik sese daha doğal bir kalite verir.

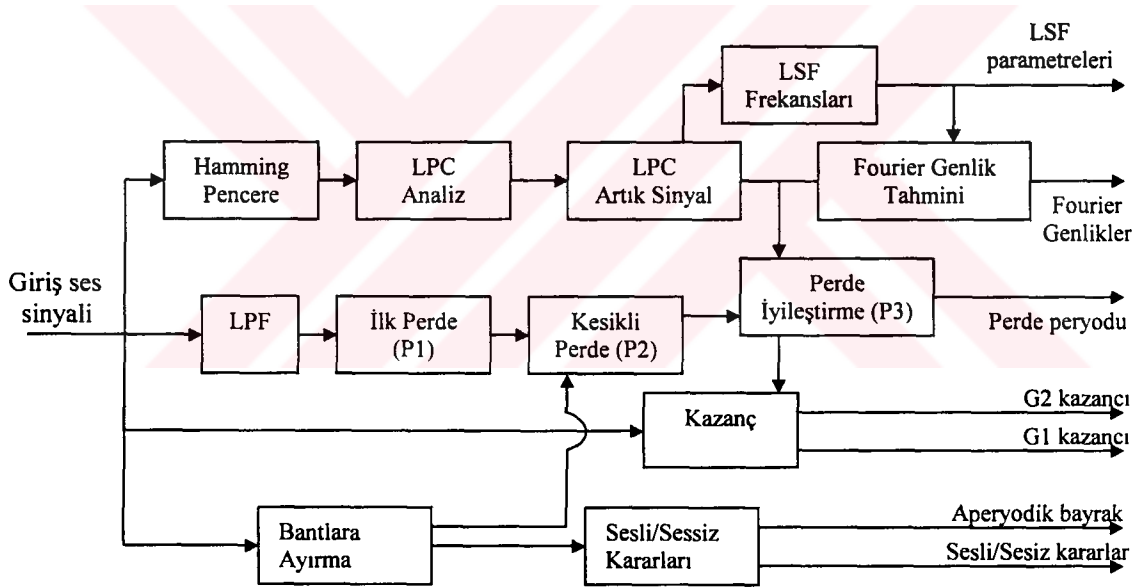
Darbe yayılım filtresi, izgel olarak düzleştirilmiş bir sabit filtreye bağlıdır. Bu filtre uyarım enerjisini bir perde periyodu içinde yayarak sentetik sesin rahatsız edici (harsh) yapısını azaltır.

İlk on Fourier genlikler, doğrusal öngörü artık sinyalin Fourier dönüşümünden belirlenir. Bu katsayılardaki bilgi yüzde olarak daha önemli düşük

frekanslarda ses üretim modelinin doğruluğunu artırır. Bu özellikle gürültülü zeminlerde erkek konuşmacılarda sentetik sesin kalitesini artırır.

6.2.1. 2400 bps MELP kodlayıcı analiz yapısı

Tüm MELP kodlayıcı işlemi parametre analizi, seçilen parametrelerin nicemlenmesi, interpolasyon ve kalite artırma teknikleri olmak üzere dört sınıfa ayrılabilir. Şekil 6.1, MELP kodlayıcı blok şemasını göstermektedir. Giriş ses sinyali önce 60 Hz kesim frekanslı yüksek geçiren filtreden geçirilerek düşük frekans enerjisi kaldırılır. Bütün analiz işlemleri bu filtrelenmiş ses sinyali üzerinde yapılır.



Şekil 6.1. MELP kodlayıcı analiz şeması

6.2.1.1. Perde tahmini

Perde peryodu hesaplamak için giriş ses sinyali 1 kHz 6'ıncı dereceden Butterworth alçak geçiren filtreden geçirilir. Normalize edilmiş özilişki

fonksiyonu $r(\tau)$ 'nin maksimum olduğu ve $\tau = 40, 41, \dots, 160$ arasında olan τ değeri tamsayı perde peryodu $P1$ olarak belirlenir:

$$r(\tau) = \frac{c_\tau(0, \tau)}{\sqrt{c_\tau(0, 0) \cdot c_\tau(\tau, \tau)}} \quad (6.1)$$

$$c_\tau(m, n) = \sum_{k=-[\tau/2]-80}^{-[\tau/2]+79} s_{k+m} \cdot s_{k+n} \quad (6.2)$$

$[\tau/2]$ bir tamsayıya yuvarlamayı göstermektedir. [0-500Hz] bant geçiren filtreden geçirilmiş giriş sinyali üzerinde kesikli perde peryodu $P2$ belirlenir. Önceki pencerede bulunmuş $P1$ ve yeni hesaplanan $P1$ değerleri alınarak, bu iki değer $+5$ ve -5 aralığındaki değerler için eşitlik (6.1) kullanılarak 2 farklı maksimum değer bulunur. En büyük özilişki değerini veren değer $P2$ olarak alınır. $P2$ etrafında *kesikli perde iyileştirme metodu* (fractional pitch refinement) [90] ile daha ince hesaplama yapılır. Sonuç perde peryodu $P3$, 1kHz kesim frekanslı Butterworth filtreden geçirilmiş doğrusal öngörü artık sinyal üzerinde hesaplanır. $P3$ perde peryodu *perde katlanma kontrol metodu* (pitch doubling check) [90] ile kontrol edilerek gerçek perde peryodunun katlarında olup olmadığına bakılır. Sonuç perde peryodu, aktarılacak perde peryodu olarak seçilir. $P3$, 7-bit index tablosu ile nicemlenir (Çizelge 6.1). Bir ve iki ağırlıklı Hamming hata koruma kodları için 28 bit saklanır. Bir bit set ise, analiz penceresi sessiz kabul edilir ve tüm sıfır kullanılır. İki bit set ise, “erasure” yani pencere tekrarlama algoritması kullanılır. Diğer durumlarda, nicemlenmiş perde değerine karşılık gelen indeks değeri kodlanıp gönderilir.

6.2.1.2. Sesli/sessiz sinyal analizi

Sesli/sessiz sinyal ayrımı için, giriş ses sinyali 6'ncı dereceden Butterworth filtre ile (0-500, 500-1000, 1000-2000, 2000-3000, 3000-4000 Hz) olmak üzere beş frekans bandına ayrılır. Bu frekans bantlarındaki ses kuvvetleri Vbp_i , $i=1,2,\dots,5$ hesaplanır. $Vbp_1 < 0.5$ ise aperyodik bayrak 0, değilse 1 alınır.

Aperiyodik bayrak 1 ise, sentezleyici uyartım sinyalinin periyodik değil aperiyodik yapıda olabileceğini hesaplar. Vbp_i ses kuvvetleri aynı zamanda perde iyileştirme ve karşılık gelen normalize edilmiş özilişki değerleri için kullanılır.

Çizelge 6.1. MELP kodlayıcı için 7-bit perde indeks çizelgesi

Kod	İndeks	Kod	İndeks	Kod	İndeks	Kod	İndeks	Kod	İndeks
0x0	Sessiz	0x1A	10	0x34	30	0x4E	51	0x68	75
0x1	Sessiz	0x1B	11	0x35	31	0x4F	52	0x69	76
0x2	Sessiz	0x1C	12	0x36	32	0x50	Erase	0x6A	77
0x3	Erase	0x1D	13	0x37	33	0x51	53	0x6B	78
0x4	Sessiz	0x1E	14	0x38	34	0x52	54	0x6C	79
0x5	Erase	0x1F	15	0x39	35	0x53	55	0x6D	80
0x6	Erase	0x20	Sessiz	0x3A	36	0x54	56	0x6E	81
0x7	0	0x21	Erase	0x3B	37	0x55	57	0x6F	82
0x8	Sessiz	0x22	Erase	0x3C	38	0x56	58	0x70	83
0x9	Erase	0x23	16	0x3D	39	0x57	59	0x71	84
0xA	Erase	0x24	Erase	0x3E	40	0x58	60	0x72	85
0xB	1	0x25	17	0x3F	41	0x59	61	0x73	86
0xC	Erase	0x26	18	0x40	Sessiz	0x5A	62	0x74	87
0xD	2	0x27	19	0x41	Erase	0x5B	63	0x75	88
0xE	3	0x28	Erase	0x42	Erase	0x5C	64	0x76	89
0xF	4	0x29	20	0x43	42	0x5D	65	0x77	90
0x10	Sessiz	0x2A	21	0x44	Erase	0x5E	66	0x78	91
0x11	Erase	0x2B	22	0x45	43	0x5F	67	0x79	92
0x12	Erase	0x2C	23	0x46	44	0x60	Erase	0x7A	93
0x13	5	0x2D	24	0x47	45	0x61	68	0x7B	94
0x14	Erase	0x2E	25	0x48	Erase	0x62	69	0x7C	95
0x15	6	0x2F	26	0x49	46	0x63	70	0x7D	96
0x16	7	0x30	Erase	0x4A	47	0x64	71	0x7E	97
0x17	8	0x31	27	0x4B	48	0x65	72	0x7F	98
0x18	Erase	0x32	28	0x4C	49	0x66	73		
0x19	9	0x33	29	0x4D	50	0x67	74		

6.2.1.3. LPC analiz

200 örnek (25 msn) Hamming pencere içindeki giriş ses sinyali üzerinde, 10'uncu dereceden doğrusal öngörü analiz gerçekleştirilir. Levinson-Durbin algoritması ile bilinen özilişki analiz işlemi hesaplanır. Doğrusal öngörü artık sinyal, giriş ses sinyalini doğrusal öngörü filtresi ile filtreleyerek hesaplanır. a_i

, ($i = 1, \dots, 10$) tahmin katsayılarına bir bant genişlik açma katsayısı olarak 0.994 (15 Hz) uygulanır. Burada herbir katsayı 0.994^i ile çarpılır.

6.2.1.4. LSF parametrelerinin hesaplanması

İlk olarak a_i , $i=1,2,\dots,10$ tahmin katsayıları LSF frekanslarına çevrilir. Sonra LSF frekansları için 50 Hz *minimum ayırma algoritması* (minimum separation algorithm) [90] uygulanır. Sonuç LSF vektörler, ÇAVN vektör f nicemleyici ile nicemlenir. ÇAVN kod çizelgesi 128,64,64 ve 64 olmak üzere dört aşama içerir. Nicemlenmiş vektör $\hat{f}$, arama işlemiyle her aşamadan seçilen vektörlerin toplamıdır. ÇAVN arama, nicemlenmiş ve nicemlenmemiş LSF vektörleri arasındaki ağırlıklı öklid mesafesinin karesi d^2 'yi minimum yapan kod çizelgesi vektörünü bulur.

$$d^2(f, \hat{f}) = \sum_{i=1}^{10} w_i (f_i - \hat{f}_i)^2 \quad (6.3)$$

$$w_i = \begin{cases} P(f_i)^{0.3}, & 1 \leq i \leq 8 \\ 0.64P(f_i)^{0.3}, & i = 9 \\ 0.16P(f_i)^{0.3}, & i = 10 \end{cases} \quad (6.4)$$

f_i , nicemlenmemiş LSF vektörünün i 'inci parçasıdır ve $P(f_i)$, f_i frekansında ters tahmin filtresi güç spektrumudur. Arama işlemi *M-eniyi* (M-best) yaklaşımı ile yapılır ve her aşamadan $M = 8$ en iyi vektör gelecek aşama için korunarak sonuç nicemlenmiş vektör hesaplanır. Nicemlenmiş vektör Fourier genlik hesaplamada kullanılır.

6.2.1.5. Fourier genliklerin hesabı

Fourier genlik analiz, nicemlenmiş tahmin katsayılarından üretilmiş artık sinyalin ilk 10 harmoniğinin Fourier genliklerini hesaplar. İlk olarak, nicemlenmiş öngörü katsayıları nicemlenmiş LSF vektörden hesaplanır.

Nicemlenmiş öngörü katsayıları ile artık sinyal hesaplanır. Hamming pencere uygulanır ve sinyalin kompleks FFT'si alınır. Kompleks FFT çıkışı genliklere çevrilir ve harmonikler *izgel pik toplama algoritması* (spectral peak picking) [90] ile belirlenir. 10 fourier genlik, 8-bit vektör nicemleyici ile nicemlenir. Kod çizelgesi düşük frekansları vurgulayan yüzde ağırlıklı Öklid mesafesi ile aranır. Ağırlıklar aşağıdaki gibi belirlenir.

$$w_i = \left[\frac{117}{25 + 75 \left(1 + 1.4 \left(\frac{f_i}{1000} \right)^2 \right)^{0.69}} \right]^2, \quad i = 1, 2, \dots, 10 \quad (6.5)$$

burada $f_i = 8000i/60$, perde peryodu 60 için, i 'inci harmoniğe karşılık gelen Hz cinsinden frekanstır.

6.2.1.6. Kazanç hesabı

Giriş ses sinyali kazancı perde uyarlamalı bir pencere içinde iki kez ölçülür. Eğer $Vbp1 > 0.6$ ise, pencere uzunluğu $P2$ nin 120 örnekten daha büyük en kısa çarpanıdır. Bu değer 320 örneği aşarsa, ikiye bölünür. $Vbp1 \leq 0.6$ ise, pencere uzunluğu 120 örnek olarak alınır. İlk pencerenin kazanç hesaplanması $G1$ olarak alınır. İkinci pencere, ilk pencereden giriş ses sinyal örneklerinin yarısı kadar geride yerleşir ve hesaplanan kazanç değeri $G2$ olarak alınır. Kazanç s_n giriş ses örnekleri için dB cinsinden ölçülen RMS bir değerdir.

$$G_i = 10 \log_{10} \left[0.01 + \frac{1}{L} \sum_{n=1}^L s_n^2 \right] \quad (6.6)$$

L pencere uzunluğudur. 0.01 terimi logaritmik fonksiyonun sıfıra çok yakın olmasını önler. Kazanç hesabı, giriş ses sinyalinin 32768 ve -32767 arasında olduğunu farzeder.

6.2.2. Parametrelerin nicemlenmesi

Perde periyodu $P3$, logaritmik değere çevrildikten sonra $\log 20$ ve $\log 160$ arasında 99-seviye doğrusal nicemleyici ile nicemlenir. Bu perde değerleri ile 7 bitlik bir kod tablosunda uygun bit değerleri seçilir. $Vbp_1 \leq 0.6$ ise sessiz durumu gösteren tüm sıfır kod kelimesi yollanır. Hamming ağırlığı 1 ya da 2 olan 28 kod kelimesi hata koruma için kullanılır. $G2$ kazanç değeri, 10 ve 77 dB arasında 5-bit nicemleyici ile nicemlenir. $G1$, $G2$ referans alınarak uyarlamalı bir algoritma ile belirlenir ve 3-bit ile nicemlenir.

Band geçiren ses kuvvetleri Vbp_i , $i=1,2,...,5$ ilgili bandın sesli ya da sessiz olarak kabul edilmesi için kullanılırlar. $Vbp_1 \leq 0.6$ ise kalan ses kuvvetleri Vbp_i , $i=2,3,4,5$ sıfır olarak nicemlenirler. $Vbp_1 > 0.6$ ise, kalan ses kuvvetleri, eğer 0.6'dan büyüklerse bir, değilse sıfır olarak nicemlenirler. Özel bir durum olarak, Vbp_i , $i=2,3,4,5$ kuvvetleri 0001 ise, Vbp_5 sıfır olarak nicemlenir. LSF parametreleri ve Fourier genlikler için seçilen indis parametreleri kodlanarak yollanır.

Kanal hatalarına karşı performans arttırmak için, sessiz durumdaki kullanılmayan kodlayıcı parametreleri, ileri hata doğrulama bitleri için kullanılır. Hamming (7,4) kodu ile tek bit hata doğrulama ve Hamming (8,4) kodu ile ilave olarak iki bit hata tespit yapılır. Sessiz durum için, ÇAVN indeksinin ilk yedi biti, $G2$ kazancının ilk dört biti ve $G1$ kazancının ilk üç biti Hamming bitleri ile korunurlar. Çizelge 6.2, 2400 bit/sn MELP kodlayıcı için parametrelerin yerleşimini göstermektedir.

Çizelge 6.2. 2400 bit/sn MELP kodlayıcıda 22.5 msn analiz penceresi için parametre yerleşimi

Parametreler	Sesli	Bit hızı (kbit/sn)	Sessiz	Bit Hızı (kbit/sn)
LSF parametreleri	25	1.11	25	1.11
Fourier Genlikler	8	0.36	-	-
Kazanç (Her pencerede iki)	8	0.36	8	0.36
Perde, tüm sesli durumu	7	0.31	7	0.31
Band geçiren kuvvetleri	4	0.18	-	-
Aperyodik bayrak	1	0.04	-	-
Hata doğrulama	-	-	13	0.58
Senkronizasyon biti	1	0.04	1	0.04
Toplam bit/22.5 msn pencere	54	2400 bit/sn	54	2400 bit/sn

6.2.2.1. Doğrusal nicemleme

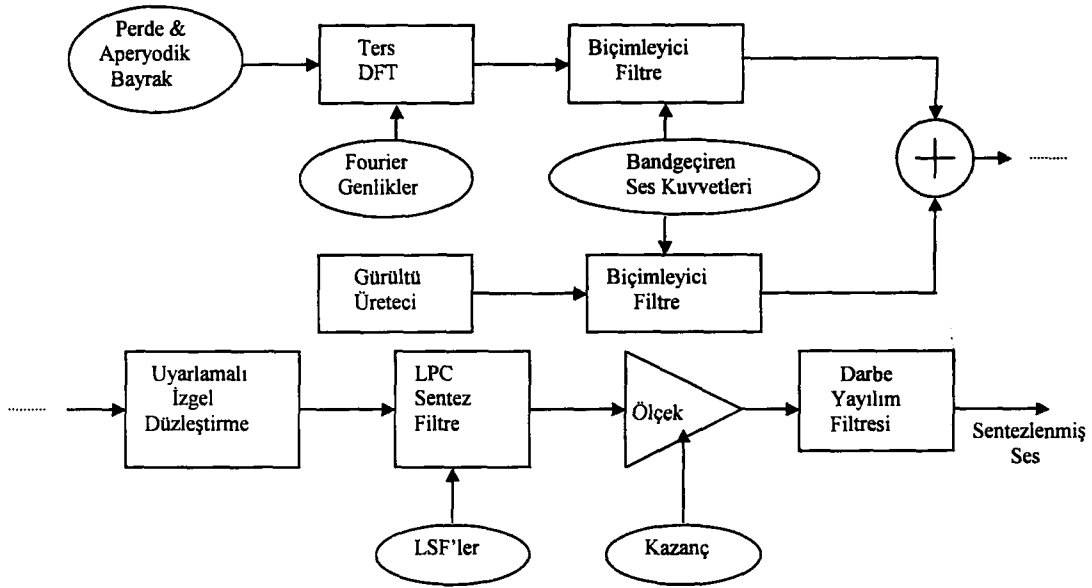
Perde ve kazanç nicemleme işlemleri için aşağıdaki doğrusal nicemleme metodu kullanılır ve bulunan indeks değeri sentez bölümüne yollanır. n seviye bir nicemleyici x_1 ve x_2 aralığında değişiyorsa, nicemleme aralığı ,

$$s = \frac{x_2 - x_1}{n - 1}$$

dir. ($i=0,1,...,n-2$) için i ve i+1 arası eşik değerleri $x_1 + (0.5+i).s$ ve olabilecek çıkış değerleri, $x_1 + i.s$, ($i=0,1,...,n-1$) dir. Örneğin, $x_1=1$, $x_2=7$ ve $n=7$ olsun. $s=1$ çıkar ve seviyeler $i=1,2,...,7$ dir. Eşik değerler ise 1.5, 2.5, ..., 6.5'dir. $x \leq 1.5$ ise indeks = 0 alınır. $1.5 \leq x \leq 2.5$ ise indeks = 1 alınır ve böylece devam eder.

6.2.3. Kodlayıcı sentezleme ve aradeğerleme

İletişim kanalından alınan bitler çözülür ve parametrelere atanır. Parametre sentezi sesli ve sessiz durumlar için farklıdır. Durum bilgisi içerdiği için ilk olarak perde çözülür. Perde kodu tüm sıfır ise ya da sadece bir bit set ise, sessiz durum kullanılır. İki bit set ise, bir önceki pencere bilgileri tekrar kullanılır (frame erasure). Diğer durumlarda, perde değeri çözülür ve sesli durum kullanılır. Şekil 6.2, MELP kodlayıcı sentez bölümünü göstermektedir.



Şekil 6.2. MELP sentezleyici blok şeması

6.2.3.1. Sesli ses sentezi

Sessiz durum için, (8,4) Hamming kodu ile tek bit hatalar doğrulanmaya ve çift bit hatalar tespit edilmeye çalışılır. Doğrulanamayan hata tespit edilirse, bir önceki pencere bilgileri tekrarlanır. Pencere tekrarı olursa, bir önceki analiz penceresinde elde edilen bütün parametreler o anki analiz penceresi parametreleri için kullanılır. Pencere tekrarı yoksa, kanaldan gelen diğer parametre bilgileri çözülür. LSF bilgileri kontrol edilir. Sessiz durum için, perde değeri 50, jitter %25, bütün bant geçirme kuvvetleri sıfır ve Fourier genlikler 1 alınır.

6.2.3.2. Sessiz ses sentezi

Sesli durum için, $V_{bp1}=1$, aperyodik bayrak 1 ise jitter %25, 0 ise jitter %0 alınır. Kanaldan alınan bant geçirme kuvvet bitleri ile V_{bp_i} ($i=2,3,4,5$) bir ya da sıfır olarak alınır.

6.2.3.3. Aradeğerleme

Çok küçük giriş sinyalleri için, $G1$ ve $G2$ kazanç parametrelerine bir miktar kazanç zayıflatma uygulanır.

$$G_{att} = -10 \log_{10} (1 - 10^{0.1[G_n + 3 - G_1]}) \quad (5.7)$$

G_n , dB cinsinden arka plan gürültü tahmini ve $G1$ ilk kazanç terimidir. Aynı işlem $G2$ için tekrarlanır.

Bütün MELP sentez parametrelerine herbir sentezlenmiş perde peryodu için perde uyumlu aradeğerleme (interpolation) yapılır. Aradeğerlenmiş parametreler, kazanç, LSF parametreleri, perde peryodu, jitter, Fourier genlikler, karma uyartım için darbe ve gürültü katsayıları ve ASE filtre için izgel eğim katsayısıdır. Kazanç dışındaki parametreler için bir önceki analiz penceresi ve o anki pencere parametreleri arasında doğrusal aradeğerleme metodu uygulanır. Yeni perde peryodu başlama noktası $t_0 = 0, 1, \dots, 179$ aradeğerleme faktörü olarak uygulanır. Aradeğerleme faktörü $int = t_0/180$ dir. eğer t_0 90'dan küçük ise, kazanç önceki pencere ikinci kazanç değeri $G2p$ ve $G1$ arasında aradeğerlenir; diğer durumda, kazanç $G1$ ve $G2$ arasında aradeğerlenir.

Karma uyartım filtreli darbe ve gürültü uyartımlarının toplamı olarak üretilir. Darbe uyartımı $e_p(n)$, $n=0, 1, \dots, T-1$, bir perde peryodu uzunluğunda ter DFT ile hesaplanır.

$$e_p(n) = \frac{1}{T} \sum_{k=0}^{T-1} M(k) \cdot e^{j2\pi nk/T} \quad (5.8)$$

Perde peryodu tamsayı cinsinden $T = aradeğerlenmiş_perde + jitter * perde$ ile hesaplanır. $M(k)$ reel olduğundan darbe uyartımının bütün fazları sıfıra eşitlenir. $e_p(n)$ reel olduğundan, genlikler:

$$M(T-k) = M(k), k=1, 2, \dots, L \quad (5.9)$$

Burada T çift sayı ise $L=T/2$, T tek sayı ise $L=(T-1)/2$ dir. DC terimi $M(0)=0$ dır. Genlik terimleri $M(k)$, $k=1,2,...,10$ Fourier genliklerin aradeğerlenmiş değerleridir. Gürültü üretici $[-1732, 1732]$ arasında bir rasgele sayı üretici ile elde edilir. Darbe ve gürültü uyartım sinyalleri filtrelendir ve karma uyartım için toplanırlar. Darbe filtresi, sesli frekans bantları için bütün bant geçiren katsayılarının toplamı, gürültü filtresi ise, sessiz bantlar için bütün gürültü katsayılarının toplamıdır. Bu filtre katsayıları perde uyumlu aradeğerlenirler.

ASE filtre, karma uyartımlı sinyale uygulanır. Bu filtre birinci dereceden bir eğim katsayılı kutup-sıfır filtresidir. Filtre katsayıları aradeğerlenmiş LSF katsayıları ile oluşan $A(z)$ doğrusal tahmin filtresi ile oluşturulur. ASE filtresi $H_{ase}(z)$:

$$H_{ase}(z) = \frac{A(\alpha z^{-1})}{A(\beta z^{-1})}(1 + \mu z^{-1}) \quad (5.10)$$

$$\alpha = 0.5p, \beta = 0.8p$$

eğim katsayısı μ , $\max(0.5k_l, 0)$ ile hesaplanır. Sinyal olasılığı p , arka plan gürültüsü G_n ve aradeğerlenmiş kazanç G_{int} 'i karşılaştırarak tahmin edilir.

$$p = \frac{G_{int} - G_n - 12}{18} \quad (5.11)$$

Uyartım keyfi bir seviyede üretildiği için, ses kazancı sentezlenmiş sese uygulanmalıdır. Ölçek faktörü S_{gain} sentetik ses sinyali $\hat{s}_n$ 'nin RMS değeri ve aradeğerlenmiş kazanç değerinden hesaplanır:

$$S_{gain} = \frac{10^{G_{int}/20}}{\sqrt{\frac{1}{T} \sum_{n=1}^T \hat{s}_n^2}} \quad (5.12)$$

T , sentezlenmiş perde peryodudur. Ses sinyalindeki süreksizlikleri önlemek için bu ölçek değeri önceki ve o anki değerler arasında aradeğerlenir.

Darbe yayılım filtresi 65th dereceden sabit katsayılı FIR filtredir ve izgel olarak düzleştirilmiş üçgen bir darbeden üretilmiştir.

Herbir perde peryodu işlendikten sonra, sentezleyici t_0 'ı T sentezlenmiş perde peryoduna ekleyerek güncelleştirir. Eğer $t_0 < \text{pencere uzunluğu}$ ise, o anki pencere için parametre interpolasyonu işlemleri tekrarlanır. Diğer türlü sentezleyici gelecek pencere analizi için hazırlanır.

6.3. Matris Nicemleyici Tabanlı Konuşma Kodlayıcı (MQBC)

MQBC konuşma kodlama sistemi, matris kod çizelgesi ile bitişik pencerelerin birleştirilmesiyle oluşturulan bir yapıya dayanmaktadır. 1200 bps hızında çalışabilen bir MQBC kodlayıcı için iki pencere birleştirilmiş ve 36 bit matris kod çizelgesi oluşturulmuştur [106,107,108]. MQBC analiz, yanyana iki pencere parametreleri için yapılır. Her iki pencere analizi sonucu çıkarılan parametreler birlikte kodlanarak sentezleyiciye gönderilir.

MQBC kodlayıcıda ilk olarak 1200 bps konuşma kodlayıcı yapısı üzerinde durulmuştur. Kodlayıcı kısmında kullanılan LSF parametrelerinin kodlanması, sesli/sessiz sinyal ayrımı, perde peryodu ve kazanç parametrelerinin en optimum şekilde kodlanabilmesi için metotlar geliştirilmiştir. LSF parametrelerinin kodlanabilmesi için matris nicemleyici metodu ile elde edilen kodlayıcı tablosu kullanılmış ve konuşma kodlayıcı yapısı, matris nicemleyici için düzenlenmiştir. LSF katsayıları hesaplanmasında; her bir pencere için LSF katsayıları hesaplandıktan sonra bir artık sinyal vektörüne dönüştürülerek matris yapısında kodlanır. Yaklaşık LSF vektörü için, sentez bölümünde, analizde çıkarılan kısım ilave edilir. Sesli/sessiz sinyal ayrımı için iki farklı metot uygulanmıştır. Birinci metot için 25 ms'nlik pencereler içindeki ses sinyali, 2 frekans bandına bölünerek ses sinyali için sesli, sessiz ya da sesli ve sessiz durumların içerildiği karma (mixed) durum elde edilmiştir. İkinci metot ise ses sinyalini doğrusal sınıflandırıcı metodu ile

analiz ederek analiz sonucu iki ayrı sesli sessiz kararı çıkarmaktadır [48]. Sesli ve sessiz durumlar, sentez kısmında perde peryoduna bakarak çıkarılabilmektedir ve bu yüzden kodlanmasına gerek olmamıştır. Karma durumu ise bir bitle kodlanmıştır. Kazanç parametresi 5 bitle logaritmik olarak ölçeklendirilip kodlanmış ve sentez kısmında uygulanan yeni bir interpolasyon metodu ile pencereler arasındaki kazanç değişimlerinden doğan süreksizlikler en aza indirilmiştir. Ayrıca sentez kısmında, analiz kısmından gelen bütün parametrelere interpolasyon metodu uygulanarak pencereler arası değişen parametrelerin devamlılığı sağlanmıştır. Kodlayıcı hatasını en aza indirebilmek için kodlanarak sentez kısmına gönderilen artık (residual) sinyal, gönderilmeden sentez kısmında o anki LPC parametreleri ve geçmiş ses sinyalinden çıkarılmış ve sonuç sesin iyileşmesinde olumlu bir etkisi gözlenmiştir. Perde peryodu için 6 bitlik kod tablosu kullanılmış ve 20 ve 160 arasında logaritmik olarak ölçeklendirilip kodlanmıştır. Toplam olarak her 25 msn'de 30 bitle parametreler kodlanmış ve 1200 bit/sn'lik kodlayıcı yapısına ulaşılmıştır. Sonuç kodlayıcı ile kodlanan ses sinyalleri dinlenmiş çok iyi sonuçlar alınmıştır.

MQBC kodlayıcı MELP kodlayıcıda bulunan karma darbe ve gürültü uyartımı, peryodik ya da peryodik olmayan darbeler, uyarlamalı izgel zenginleştirme filtresi, darbe yayılım filtresi özelliklerini içermektedir. MELP kodlayıcıda bulunan Fourier genlikler yerine sabit değerler kullanılmış ve bunun kodlanan sesler üzerinde olumsuz etkisi dinleme testlerinde farkedilememiştir.

MQBC kodlayıcı işlemi parametre analizi, seçilen parametrelerin nicemlenmesi, interpolasyon ve kalite arttırma teknikleri olmak üzere dört sınıfa ayrılabilir. Şekil 6.3, MQBC kodlayıcı blok şemasını göstermektedir. Giriş ses sinyali önce 60 Hz kesim frekanslı yüksek geçiren filtreden

periyodu 7 bit yerine 6 bitle kodlanmıştır. MELP kodlayıcısındaki 7 bitlik indeks tablosu içinde bulunan “UNVOICED” ve “ERASURE” durumları MQBC kodlayıcıda içerilmemektedir. Bunun yerine sadece indeks 0 durumu, “UNVOICED” ses durum bilgisi olarak kullanılmıştır. Diğer 63 indeks değeri nicemlenmiş perde periyod değerlerine karşılık gelmektedir. MQBC kodlayıcı için perde kodlama çizelgesi, çizelge 6.3’de görülmektedir.

Çizelge 6.3. MQBC kodlayıcı için 6-bit perde indeks çizelgesi

Kod	İndeks	Kod	İndeks	Kod	İndeks	Kod	İndeks	Kod	İndeks
0x0	Sessiz	0xD	12	0x1A	25	0x27	38	0x34	51
0x1	0	0xE	13	0x1B	26	0x28	39	0x35	52
0x2	1	0xF	14	0x1C	27	0x29	40	0x36	53
0x3	2	0x10	15	0x1D	28	0x2A	41	0x37	54
0x4	3	0x11	16	0x1E	29	0x2B	42	0x38	55
0x5	4	0x12	17	0x1F	30	0x2C	43	0x39	56
0x6	5	0x13	18	0x20	31	0x2D	44	0x3A	57
0x7	6	0x14	19	0x21	32	0x2E	45	0x3B	58
0x8	7	0x15	20	0x22	33	0x2F	46	0x3C	59
0x9	8	0x16	21	0x23	34	0x30	47	0x3D	60
0xA	9	0x17	22	0x24	35	0x31	48	0x3E	61
0xB	10	0x18	23	0x25	36	0x32	49	0x3F	62
0xC	11	0x19	24	0x26	37	0x33	50		

6.3.2. Sesli/sessiz sinyal analizi

Sesli/sessiz sinyal ayrımı için iki farklı metot denenmiştir. Birinci metot ses sinyalini iki ayrı frekans bandına bölüp, bu frekans bantlarından sesli/sessiz kararları çıkarmaktadır. İkinci metot ise ses sinyalini doğrusal sınıflandırıcı metodu ile analiz ederek analiz sonucu iki ayrı sesli sessiz kararı çıkarmaktadır. Her iki metot için, sesli/sessiz sinyal analizi için iki ayrı ses kuvveti (V_0 , V_1) hesaplanmaktadır. $V_1 \neq V_0$ ise $karma = 1$ değilse $karma = 0$ alınır. 6 bitlik perde periyodu için analiz ve sentez kısmında $V_1 = 0$, $V_0 = 0$ ve $karma = 0$ ise, perde periyodu indeksi = 0 gönderilir. Sentez kısmında sessiz durum için perde periyodu sabit = 50 olarak kabul edilir. $V_1 = 1$, $V_0 = 1$ ve $karma = 0$ ise perde periyodu indeksi olarak, perde periyoduna karşılık gelen ve

sıfır olmayan bir indis değeri gönderilir. Karma = 1 ve perde indeksi $\neq 0$ ise $V_0=1$, $V_1=0$ durumu gönderilmiş olur ve sentez kısmında böyle değerlendirilir. Diğer durumlara göre nispeten çok az gerçekleşen karma = 1, $V_0 = 0$, $V_1=1$ durumunda ise perde indeksi = 0 yollanır ve yine sentezde perde periyodu değeri = 50 alınır. Sessiz kabul edilen ses pencereleri için set olan ve aperyodik darbelerin uygulandığı aperyodik flag analiz kısmında kodlanarak gönderilmeden sentez bölümünde sesli/sessiz/karma parametrelerinden çıkarılmaktadır. Sentez kısmında sesli uyarma (excitation) sinyali aşağıdaki şekilde uygulanmaktadır.

6.3.2.1. Frekans bantlarına ayırma ile sesli/sessiz sinyal analizi

Bantlara ayırma ile sesli/sessiz sinyal sınıflandırıcı dizaynında, ses sinyali bant geçiren filtreler ile [0-2000 Hz] ve [2000-4000 Hz] frekans bantlarına ayrılarak iki ayrı ses kuvveti (V_0 , V_1) hesaplanmaktadır. Eğer ses kuvvetleri 0.5 üzerinde ise ilgili frekans bandı sesli (1), değilse sessiz (0) olarak alınır.

V_0	V_1	0-2000 frekans bandı için	2000-4000 frekans bandı için
0	0	rastgele sinyal	rastgele sinyal
0	1	rastgele sinyal	peryodik darbe sinyali
1	0	peryodik darbe sinyali	rastgele sinyal
1	1	peryodik darbe sinyali	peryodik darbe sinyali

Ses sinyal spektrumunu beş banda ayırarak ve iki banda ayırarak kodlanan bir ses sinyali için frekans bantlarının görünümü Şekil 6.4'deki gibidir. 5 frekans bandı yerine 2 frekans bandına bölünüp sesli/sessiz kararları alınmış ses sinyali için, dinleme testlerinde sesin doğallığında azalma hissedilmekle birlikte anlaşılabilirlik etkilenmemiştir.

6.3.2.2. Doğrusal ayırıcı ile sesli/sessiz sinyal analizi

Sesli ve sessiz pencereleri ayırmada temel özellik olan sesin periyodik yapısını incelemek yanında, diğer bazı özelliklerde dikkate alınmalıdır. Her pencere için, pencerenin iki yarısı için analiz yapılır ve V/U kararları alınır. Bunun için izlenen işlem sırası şöyledir:

- Giriş sesi filtrelenir.
- Sesslilik parametre ölçümü yapılır
- V/U sınıflandırma yapılır
- V/U sınıflandırma için smoothing yapılır.

I. Filtreleme

Giriş sinyali alçak geçiren filtreden geçirilir ve düşük bant sesli uyarım sinyali çıkarılır. Alçak geçiren filtre karakteristikleri maksimum sınıflandırmayı gerçekleştirecek şekilde seçilir. Burada 19 point FIR equiripple linear phase filter kullanılmıştır (800 Hz passband, 1 dB ripple, 1336 Hz stopband, 30 dB zayıflatma).

II. Sesslilik parametreleri ölçümü

Sesslilik penceresi ses dalga şeklinin durağan (hızlı değişimlerin olmadığı) yerde yerleştirilir. Her yarı pencere için aşağıdaki parametreler hesaplanır:

- *Peryodik yapı* : Bu sesli ses için en temel özelliktir ve değişik şekillerde incelenebilir. Kısa süreli ve uzun süreli tahmin kazancı, periyodik sinyaller için daha yüksektir. Peryodik sinyaller genellikle güçlü bir kısa süreli ilişki gösterirler. Daha yüksek genlikler ve LP katsayıları, özellikle ilk katsayı, sesli sesi ayırmada kullanılabilir. Sesli ses spektrumu bir harmonik yapı gösterir. Bu yüzden, orijinal ve hesaplanan perde kullanılarak tahmin

edilen spektrumun benzerliği peryodik yapıyı ortaya çıkarmada kullanılabilir. Perde peryoduna karşılık gelen cepstral piklerin yüksek genliği de ses peryodikliğini bulmada kullanılabilir.

- *Düşük bant ses enerjisi:* Bu özellik sesli/sessiz ayrımında kullanılır. Enerji, V/U kararında başka bir önemli özelliktir. Genellikle, sesli ses sessiz sestten yaklaşık 30 dB daha yüksek enerjiye sahiptir. Ses yapısı alçak geçiren filtre gibidir. Sonuç olarak sesli seste, enerjinin büyük çoğunluğu alçak frekanslarda içerilir. Bu özellikle sessiz seste daha azdır. Bu yüzden, alçak frekans bandındaki enerjinin yüksek frekans bandına oranı V/U sınıflandırmada kullanılabilir. Enerjideki bir problem, konuşmacı bağımlı olmasıdır ve bazı konuşmacılar için sonuç çok değişebilir. Bunun için bir çözüm V/U kararda enerji eşik değeri kullanmaktır.
- *Süreklilik :* Sesli veya sessiz bölümlerin süresi genellikle analiz penceresinden daha uzundur. Bu genellikle sesli bölümler için doğrudur. Sonuç olarak, ileri ve geri bakma karar teknikleri V/U sınıflandırıcının performansını arttırmada kullanılabilir. Sesli sesteki perde peryodu değişimi sınırlıdır ve süreklilik ölçümünde kullanılabilir.
- *Sıfır geçiş oranı:* Yüksek sıfır geçiş, ses sinyalinde yüksek frekanslar olduğunun ve ilgili kısmın sessiz olabileceğinin göstergesidir. Sıfır geçiş, giriş sinyalindeki işaret değişimlerini sayar. Uygulama frekansının doğal sınırlaması ve sesli sesteki alçak frekans harmoniklerin enerjisinin nispeten daha yüksek olması nedeniyle, bu sinyaller sessiz sesteki rasgele sinyallere kıyasla daha düşük sıfır geçişe sahiptirler
- *Yansıma katsayıları:* İlk yansıma katsayısı, tüm ses geçiş bandındaki eğimi ölçer. Sesli seste örnekler arası ilişki fazla olduğundan artan frekansla

azalan genlik izgel eğimin büyük olmasına yol açar. Sessiz seste bu eğim azdır. İkinci yansıma katsayısı, genellikle sesli seste yüksek olan düşük bant izgel pik derecesini ölçer. (4:1) azaltılmış ve 800 Hz LPF ile filtrelenmiş ses üzerinde hesaplanır.

- *Önvurgulanmış enerji oranı*: giriş sesi fark sinyalinin giriş sesine oranıdır. Ağırlıklı yüksek bant enerjisini ölçer. Bant geçişlerinde eğime daha az duyarlıdır. Böylece “plosive” sesleri tespit edebilir.
- *Perde tahmin kazançları* : 1. dereceden ileri ve geri perde tahmin kazançları, ileri ve geri perde periyodikliğini izlemeye kullanılırlar. Böylece sesli/sessiz geçişler farkedilebilir.

III. Sesslilik sınıflandırma

a_{ij} ve c_j ağırlıklarının doğru seçebilmek için, bulunan parametrelerin akustik gürültü seviyesine uyarlanabilmesi gerekir. $SNR = (\text{average fullband voicing energy} / \text{average fullband unvoiced energy})$ ile akustik gürültü belirlenir. Yüksek enerji sesslilik (voicing) sınıflandırmaları güç olduğundan sessiz enerji filtresine giriş önceki girişten 10 dB daha az ile sınırlanır. Sesli/sessiz kararı için N parametre ile, M -seviye sınıflandırıcıda j parametresi, eşitlik (6.1)’i kullanarak SNR ’a göre belirlenir.

$$\sum_{i=1}^N a_{ij} \cdot p_i + c_j > 0 \quad j \in (0, \dots, M-1) \quad (6.1)$$

p_j ölçülen sesslilik parametreleridir. a_{ij} ve c_j parametrelerini bulmada bir teknik Fisher’in doğrusal sınıflandırıcı metodudur. [48]

IV. Smoothing

Bir yarı pencerede (U,V,U) ya da (V,U,V) şeklinde geçiş oldu ise aradaki geçiş kaldırılır.

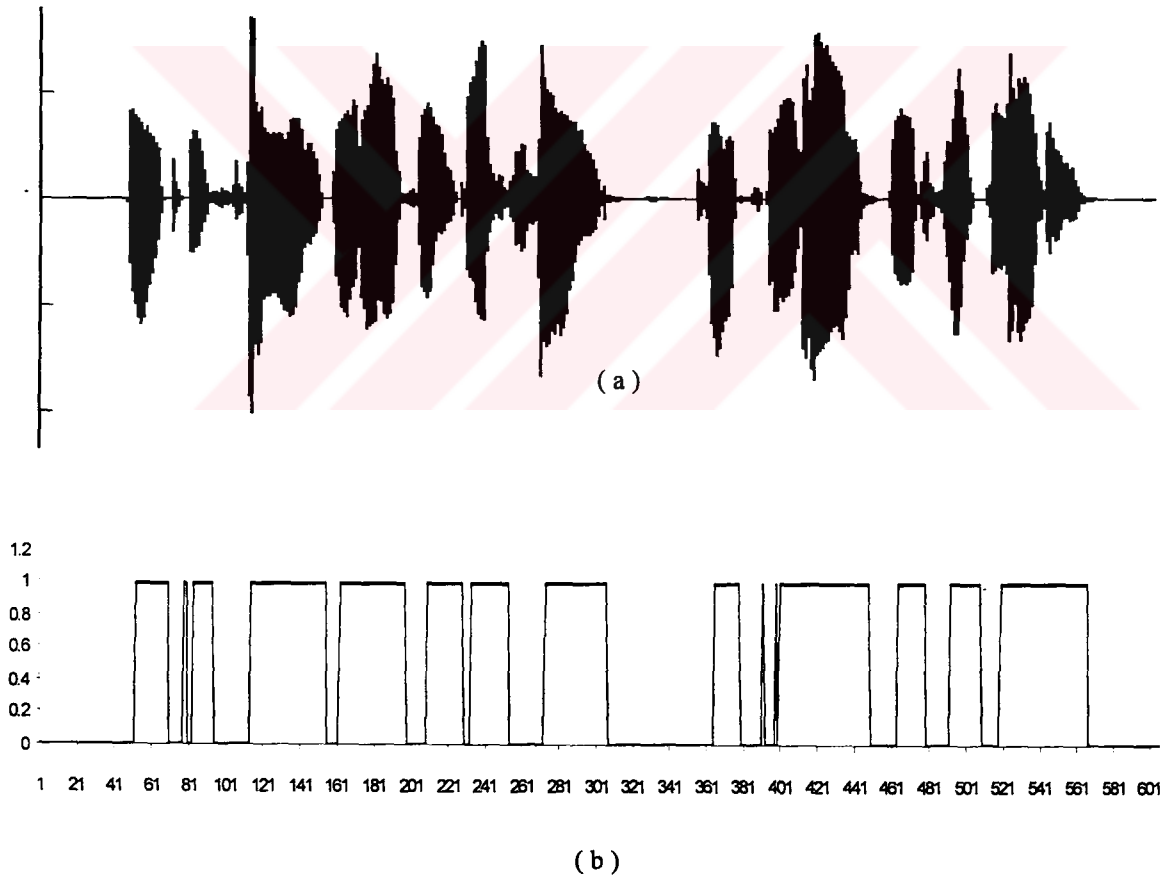
V/U kararda bütün bu özellikler kullanılmasına rağmen, herbirinin etkinliği ses sinyalinin yapısına bağlıdır. Yani, bir özellik bir analiz penceresi içinde iyi çalışırken, başka birkaç analiz penceresi için diğerleri daha etkin olabilir. Örneğin, ortalama enerji genellikle çok etkin bir ölçümdür. Fakat, plosive'ler gibi enerjinin yüksek olduğu sessiz sinyallerde başarısızdır. Sıfır geçiş oranı ve alçak frekans bant enerjisinin yüksek frekans bant enerjisine oranı, sessiz sinyaller düşük genlikli sesli sinyallerle birleşik yapıda ise iyi sonuç vermez. Herbir özelliği seçmek ve herbiri için uygun bir başlangıç değeri bulmak için bazı istatistiksel analiz yapmak gerekir.

Onset ve offset pencere: Ses sınıflandırmada, onset (sessiz durumdan sesli duruma geçiş) ve offset (sesli durumdan sessiz duruma geçiş) özel bir dikkat gerektirir. Onset pencereler kısmen sesli kabul edilebilir. Yine de, bir ikili V/U kararda, ya sesli ya da sessiz olarak karar verilmelidir. Her iki durum için, doğru olmayan karar nedeniyle sentetik seste distorsiyon beklenir. Onset ve ofset pencerelerin sesli seçilmesi, sonuç seste vızıltıya (buziness) sebep olur, diğer durumda çıkış rahatsız edici (harshness) dir. Onset ve ofset pencereler, sesli ve sessiz pencereler arasında interpolasyonla sentezlenmelidir. Onset pencereleri belirlemek için bir metot, sessiz bir durumdan sonra sesli bir durumun gelebileceği hakkında önceki pencereyi doğru analiz etmektir. Başka bir yaklaşım, pencereyi analiz parçalarına ayırmak herbir bölümü ayrı analiz etmektir. Sesli bölüme geçiş buradan yakalanabilirse onset durumu belirlenebilir ve pencere sesli olarak açıklanabilir.

Doğrusal sınıflandırıcı ile sesli/sessiz ayırım metodu kullanılarak kodlanan bir ses sinyali için bant ayrımları Şekil 6.4'de görülmektedir. Doğrusal

sınıflandırıcı ile sesli/sessiz sinyal ayrımı yapılmış ses sinyali için, sentezleme algoritması MELP sentezleyici algoritmasına çok benzediği için, frekans bantlarına ayırma metoduna göre ses kalitesinde çok az bir bozulma vardır. LPC-10 konuşma kodlayıcının sentez metodu kullanıldığında ise LPC'nin zayıf performansı yüzünden ses çıkışı LPC kodlayıcı çıkışına benzemektedir. İyi bir zaman düzlemi sentezleme metoduna dayalı kodlayıcı algoritması ile kalitenin çok artacağı düşünülmektedir.

“ Oak is strong and also gives shade cats and dogs each hate the other”



Şekil 6.4. a) Orijinal ses sinyali b) Doğrusal sınıflandırıcı metodu ile sesli/sessiz analiz sonuçları

6.3.3. LPC analiz

LP analiz gerçekleştirmek için bazı parametrelerin seçilmesi gereklidir. Bu parametrelerdeki değişim, performansı etkiler. LP analiz yapılmadan önce, Çizelge 6.4'deki parametre kümesi sırasıyla MELP ve MQBC kodlayıcılar için kullanılmıştır. 220 örnek (27.5 msn) Hamming pencere içindeki giriş ses sinyali üzerinde, 10'uncu dereceden doğrusal öngörü analiz gerçekleştirilir. Levinson-Durbin algoritması ile bilinen özilişki analiz işlemi hesaplanır. Doğrusal öngörü artık sinyal, giriş ses sinyalini doğrusal öngörü filtresi ile filtreleyerek hesaplanır. a_i , ($i = 1, \dots, 10$) tahmin katsayılarına bir bant genişlik açma katsayısı olarak 0.994 (15 Hz) uygulanır. Burada herbir katsayı 0.994ⁱ ile çarpılır. Pencere içindeki sinyal 25 msn içinde durağan kabul edilebileceğinden, analiz penceresi uzunluğunu 25 msn'ye çıkartmak sonuçtaki ses sinyalinde farkedilir bir bozulma yaratmamıştır.

Çizelge 6.4. MELP kodlayıcı ve MQBC kodlayıcı için bazı analiz parametreleri

	MELP kodlayıcı	MQBC kodlayıcı
Örnekleme hızı	8 kHz	8 kHz
LP filtre derecesi	10	10
Pencere uzunluğu	22.5 msn (180 örnek)	25 msn (200 örnek)
Hamming analiz penceresi	25 msn (200 örnek)	27.5 msn (220 örnek)
LP katsayılar tahmin metodu	Özilişki (Autocorrelation)	Özilişki

6.3.4. LSF parametrelerinin hesaplanması

LPC parametreleri hesaplanmasında kullanılan Hamming penceresi 25 msn'lik pencere analizi için kullanılırken, pencere kenarındaki süreksizlikleri

gidermek için daha büyük seçilmesi gereklidir. 27,5 ms'n'lik sayısal Hamming penceresi (N=220 parametrelili) yeniden hesaplanmıştır. Hamming penceresi,

$$w(n) = \begin{cases} 0.54 - 0.46 \cos 2\pi \frac{n}{N-1} ; & 0 \leq n \leq N-1 \\ 0 & ; \text{diğer} \end{cases}$$

Konuşma kodlayıcının analiz kısmında, 25 ms'n'lik pencereler içindeki ses sinyalinden LPC parametreleri bulunarak LSF parametrelerine çevrilir ve LSF artık sinyali bulunur. Ayrıca kazanç, perde periyodu ve sesli/sessiz durumu tespit edilir. Bir sonraki pencere içindeki ses sinyalinden yeni LPC parametreleri bulunarak LSF parametreleri ve LSF artık sinyali elde edilir. Birinci ve ikinci pencere analizinden elde edilen LSF artık sinyali, 36 bitlik matris nicemleyici kod çizelgesinde kodlanarak sentez kısmına gönderilecek indis değerleri bulunur. Birinci ve ikinci pencere için bulunmuş kazanç, perde periyodu, sesli/sessiz/karma durumları ve her iki pencere analizinden çıkarılmış LSF parametreleri için indis değerleri kodlanarak sentez kısmına gönderilir.

Her bir pencere için LSF katsayıları hesaplandıktan sonra aşağıdaki gibi artık sinyale dönüştürülerek kodlanmaktadır. Birinci pencere için artık sinyal vektörü:

$$res1[i] = lsf1[i] - mean[i] - FAC * prevres1[i], \quad i = 0, \dots, 9 \quad (6.2)$$

$$\begin{bmatrix} res1_0 \\ res1_1 \\ \dots \\ res1_9 \end{bmatrix} = \begin{bmatrix} lsf1_0 \\ lsf1_1 \\ \dots \\ lsf1_9 \end{bmatrix} - \begin{bmatrix} mean_0 \\ mean_1 \\ \dots \\ mean_9 \end{bmatrix} - FAC * \begin{bmatrix} prevres1_0 \\ prevres1_1 \\ \dots \\ prevres1_9 \end{bmatrix}$$

Benzer şekilde ikinci pencere için artık sinyal vektörü:

$$res2[i] = lsf2[i] - mean[i] - FAC * prevres2[i], \quad i = 0, \dots, 9 \quad (6.3)$$

Her iki pencere için artık sinyal vektörü tek bir *res* (20×1) kod vektörü birleştirilip *reshat* (20×1) yaklaşık kod vektörü elde edilir.

$$res = \begin{bmatrix} res1 \\ res2 \end{bmatrix} = [res1_0, res1_1, \dots, res1_9, res2_0, res2_1, \dots, res2_9]^T \quad (6.4)$$

Her iki pencere için kodlanmış LSF vektörleri:

$$\begin{aligned} lsflat1[i] &= reshat[i] + mean[i] + FAC * prevres1[i], \quad i = 0, \dots, 9 \\ lsflat2[i] &= reshat[i+10] + mean[i] + FAC * prevres2[i], \quad i = 0, \dots, 9 \end{aligned} \quad (6.5)$$

prevres1 ve *prevres2* vektörleri ise aşağıdaki gibi hesaplanır:

$$\begin{aligned} prevres1[i] &= lsflat1[i] - mean[i] + FAC * prevres1[i], \quad i = 0, \dots, 9 \\ prevres2[i] &= lsflat2[i] - mean[i] + FAC * prevres2[i], \quad i = 0, \dots, 9 \end{aligned} \quad (6.6)$$

mean vektörü, kod çizelgesi oluşturulurken hesaplanan ve eğitim giriş dizisindeki bütün kod vektörlerin ortalaması ile hesaplanan sabit bir değerdir. FAC değeri ise, $FAC=0.325$ olarak alınmıştır. (Bkz. Şekil 5.4)

6.3.5. Kazanç hesabı

Giriş ses sinyali kazancı perde uyarlamalı bir pencere içinde bir kez ölçülür. Eğer $Vbp1 > 0.6$ ise, pencere uzunluğu $P2$ 'nin 120 örnekten daha büyük en kısa çarpanıdır. Bu değer 320 örneği aşarsa, ikiye bölünür. $Vbp1 \leq 0.6$ ise, pencere uzunluğu 120 örnek olarak alınır. Kazanç s_n giriş ses örnekleri için dB cinsinden ölçülen RMS bir değerdir.

$$Gi = 10 \log_{10} \left[0.01 + \frac{1}{L} \sum_{n=1}^L s_n^2 \right] \quad (6.6)$$

L pencere uzunluğudur. 0.01 terimi logaritmik fonksiyonun sıfıra çok yakın olmasını önler. Kazanç hesabı, giriş ses sinyalinin 32768 ve -32767 arasında olduğunu farzeder.

6.3.6. Parametrelerin nicemlenmesi

Perde peryodu P3, 20 ve 60 örnek arasında değişen 63-seviye nicemleyici ile logaritmik ölçekte nicemlenir. Bu perde değerleri ile 6 bitlik bir kod tablosunda uygun bit değerleri seçilir. MELP kodlayıcıda olduğu gibi $V_{bp1} \leq 0.6$ ise sessiz durumu gösteren tüm sıfır kod kelimesi yollanır. İki ayrı kazanç değeri yerine MQBC kodlayıcı için G kazanç değeri, 10 ve 77 dB arasında 5-bit nicemleyici ile nicemlenir. Karma durumu, Çizelge 6.5'deki gibi alınır. Sesli/sessiz durumu ve perde peryodunun birleşik nicemlenmesi Çizelge 6.6.'daki gibidir. Peşpeşe iki analiz penceresi içinde hesaplanan LSF parametreleri için seçilen indis parametreleri kodlanarak yollanır. Kanal hatalarına karşı koruma için kullanılan Hamming kodu, MQBC kodlayıcıda yoktur. Yine Fourier genliklerin hesaplanması MQBC kodlayıcıda yoktur.

Çizelge 6.5. 25 msn 1200 bit/sn MQBC pencereleri için parametre yerleşimi

Parametreler	1. pencere	2. pencere	Bit hızı (kbit/sn)
LSF parametreleri	36		0.72
Kazanç	5	5	0.20
Perde, tüm sesli durumu	6	6	0.24
Karma uyarım	1	1	0.04
Toplam bit	60		1.20

Çizelge 6.6. Sesli/sessiz durumu ve perde peryodunun birleşik nicemlenmesi

Sesli/sessiz kararlar	Karma uyarım ve Perde peryodu için nicemleme indeksi	
	Perde indeksi	Karma uyarım
UU	0	0
UV	0	1
VU	1->63	1
VV	1->63	0

6.3.7. Kodlayıcı sentez ve aradeğerleme

MELP kodlayıcıya benzer şekilde, iletişim kanalından alınan bitler çözülür ve parametrelere atanır. Parametre sentezi sesli ve sessiz durumlar için farklıdır. Durum bilgisi içerdiği için ilk olarak perde çözülür. Perde kodu tüm sıfır ise ya da sadece bir bit set ise, sessiz durum kullanılır. İki bit set ise, bir önceki pencere bilgileri tekrar kullanılır (frame erasure). Diğer durumlarda, perde değeri çözülür ve sesli durum kullanılır. Sesli ses veya sessiz ses sentezi için MELP kodlayıcıdaki koşullar dikkate alınarak hareket edilmektedir. Fakat MQBC kodlayıcıda Hamming kodlama olmadığı için hata doğrulama yoktur.

MQBC sentez işlemi MELP kodlayıcı sentez bölümüne benzemektedir. Fourier genlikler yerine sabit katsayılar alınmıştır. Bütün MQBC sentez parametrelerine herbir sentezlenmiş perde peryodu için perde uyumlu aradeğerleme yapılır. Aradeğerlenmiş parametreler, kazanç, LSF parametreleri, perde peryodu, jitter, karma uyartım için darbe ve gürültü katsayıları ve ASE filtre için izgel eğim katsayısıdır. Kazanç dışındaki parametreler için bir önceki analiz penceresi ve o anki pencere parametreleri arasında doğrusal aradeğerleme metodu uygulanır. Yeni perde peryodu başlama noktası $t_0 = 0,1,...,199$ aradeğerleme faktörü olarak uygulanır. Aradeğerleme faktörü $int = t_0/200$ dir. Kazanç sentezlenirken, bitişik pencereler arasındaki büyük kazanç değişimlerinden doğan süreksizliği azaltmak için büyük bir ses dosyası üzerinde inceleme yapılmıştır ve buradan çıkarılan kazanç eğitim datasında yapılan inceleme ile belirlenen yeni bir interpolasyon formülü uygulanmıştır.

Bu formül farzeder ki, bir önceki pencere kazancı (G_p) ve o andaki pencere kazancı (G) arasındaki kazanç farkı 10 dB'den büyükse, kazançlar arasında doğrusal olmayan bir değişim vardır ve bu değişim basitlik açısından sabit bir

değer olarak alınmıştır. Kazançlar arası fark 10 dB'den küçükse değişim doğrusal kabul edilebilir. Eğer yeni perde peryodu başlama noktası $f_0 = 0, 1, \dots, frame$ pencere/2'den az ise Kazanç G_p ve ortalama değer G_x arasında doğrusal interpolate edilir, değilse G ve G_x arasında interpolate edilir (Şekil 6.5). İnterpolasyon faktörü $fact$ yeni perde peryodunun başlama noktasına bağlıdır.

$$fact = f_0 / frame$$

```

if (G < 10dB)  Gx = 10dB
else if (G > Gp) then
    if (G-Gp) < 10dB,  Gx = Gp+0.5*(G-Gp)
    else if (G-Gp) ≥ 10dB,  Gx = G - (G-Gp)*0.3
else if (Gp > G) then
    if (G-Gp) < 10dB,  Gx = G+0.5*(Gp-G)
    else if (G-Gp) ≥ 10dB  Gx = Gp - (Gp-G)*0.3

```

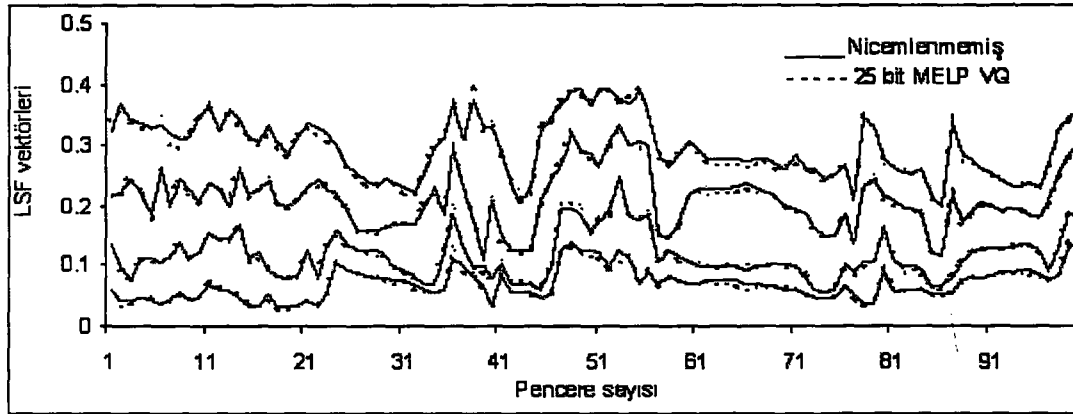
Şekil 6.5. MQBC kodlayıcıda kullanılan kazanç interpolasyon algoritması

Karma uyarım hesaplanması ve ASE filtre ile darbe yayılım filtresinin karma uyarımlı sinyale uygulanması MELP kodlayıcıda olduğu gibidir. Sentezlenmiş perde peryodu T , ses sinyalindeki süreksizlikleri önlemek için bu ölçek değeri önceki ve o anki değerler arasında aradeğerlenir. Herbir perde peryodu işlendikten sonra, sentezleyici t_0 'ı T sentezlenmiş perde peryoduna ekleyerek güncelleştirir. Eğer $t_0 < \text{pencere uzunluğu}$ ise, o anki pencere için parametre interpolasyonu işlemleri tekrarlanır. Diğer türlü sentezleyici gelecek pencere analizi için hazırlanır.

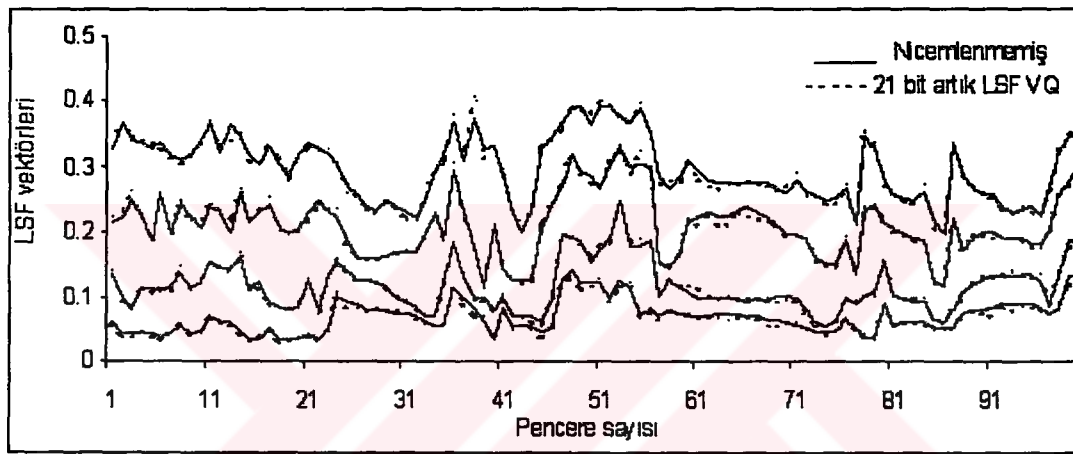
6.3.8. MQBC kodlayıcı performans kriterleri

1600 bit/sn ve 1200 bit/sn MQBC kodlayıcılar için performans kriterleri aşağıdaki şekilde sağlanmıştır:

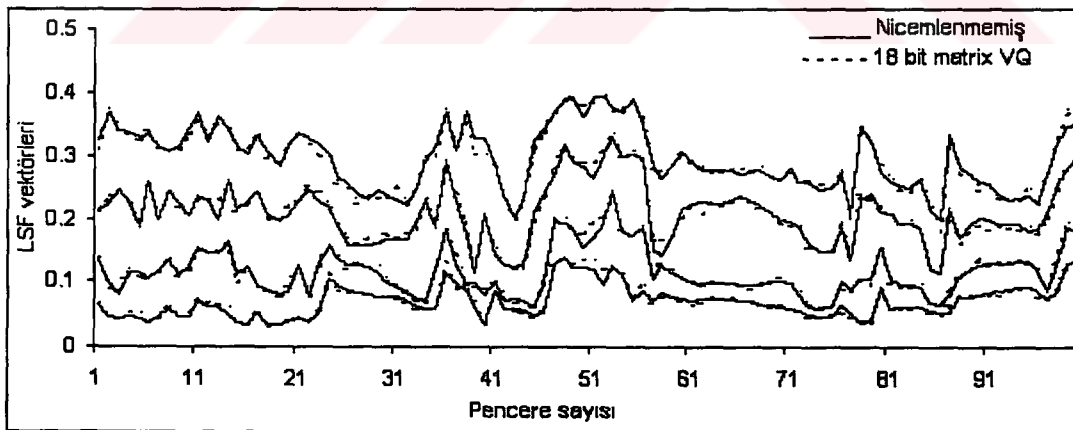
- *Kalite ve bit hızı*: Düşük bit hızlarında kalite kaçınılmaz olarak düşmektedir. MQBC kodlayıcılarda düşük bit hızlarına inildiği halde, kalitedeki düşüm objektif ve subjektif testlerde çok düşük olmuştur.
 - *Kanal hatalarına dayanıklılık* : MQBC kodlayıcılarda hata kontrol bitleri kullanılmamıştır. MQBC kodlayıcı bir iletişim kanalı üzerinde kullanılacaksa, kanal hatalarına karşı koruma bitleri kullanılmalıdır.
 - *İletişim Gecikmesi* : MQBC kodlayıcıda iki pencere birleştirilip kodlandığı için 2 pencere süresi (50 msn) bir gecikme vardır. Bu gecikme yankı gidericilerle telafi edilebilir.
 - *Birlikte çalışabilirlik*: MQBC kodlayıcının, MELP kodlayıcı ile birlikte çalıştırılabilirliği sağlanmıştır.
 - *Ses bandı veri aktarımı* : MQBC kodlayıcıda son filtreleme (postfiltering) kullanılmadığı için, ses dışındaki sinyalleri aktarabilir.
 - *Şifreleme* : Şifreleme, bir konuşma kodlayıcı yapısı üzerine sonradan ilave edilebilir özellikte olduğundan MQBC kodlayıcılar üzerinde istenirse şifreleme kullanılabilir.
 - *Karmaşa ve maliyet* : MQBC kodlayıcıda karmaşayı artıran özellik, kod çizelgesi arama karmaşasının yüksekliğidir. Fakat, karmaşa yüksek hızlı sinyal işleme devreleri ile (örn. TMS320VC33, TMS32067x v.b.) sorun olmaktan çıkmıştır.
-



(a)

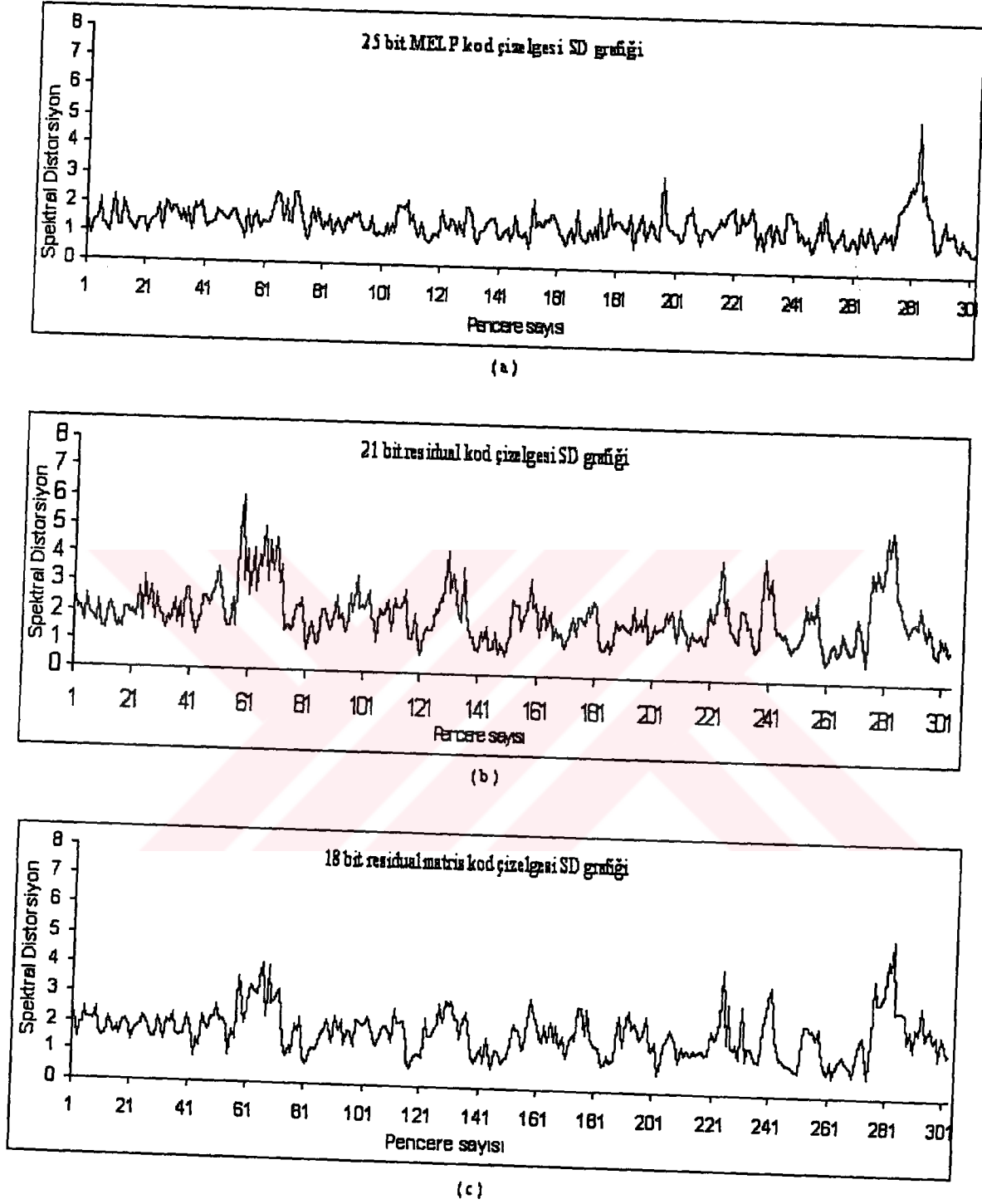


(b)

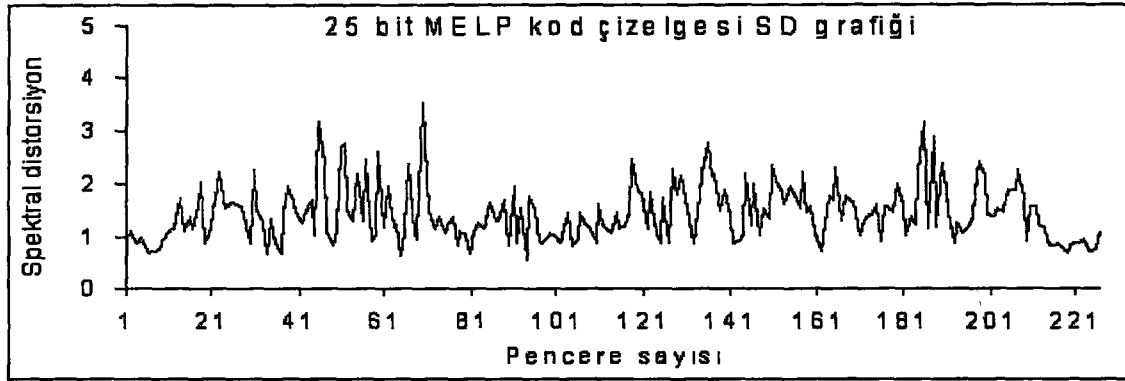


(c)

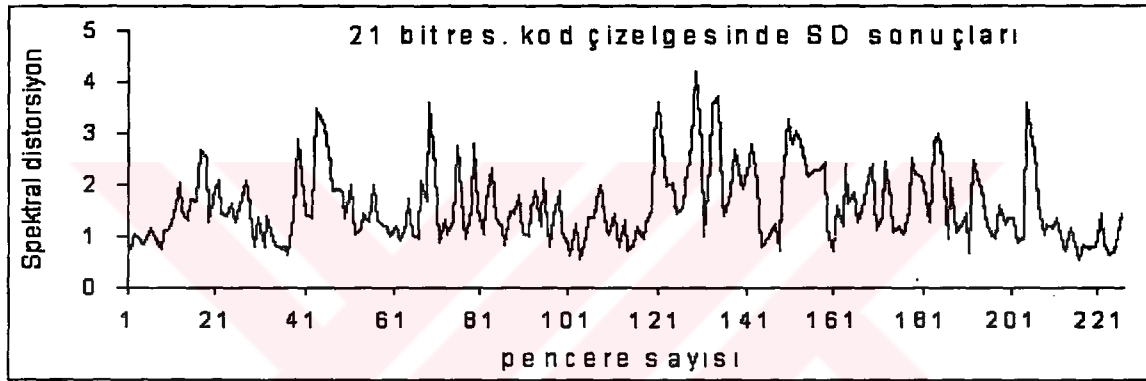
Şekil 6.6. Nicemlenmiş ve nicemlenmemiş ilk 4 LSF vektörlerinin a)25 bit MELP kod çizelgesi için b) 21-bit artık LSF kod çizelgesi için c)18 bit matris kod çizelgesi için grafikleri. Analiz penceresi uzunluğu 22.5 msn dir. LSF analizde kullanılan ses örnekleri “Oak is strong and also ...”



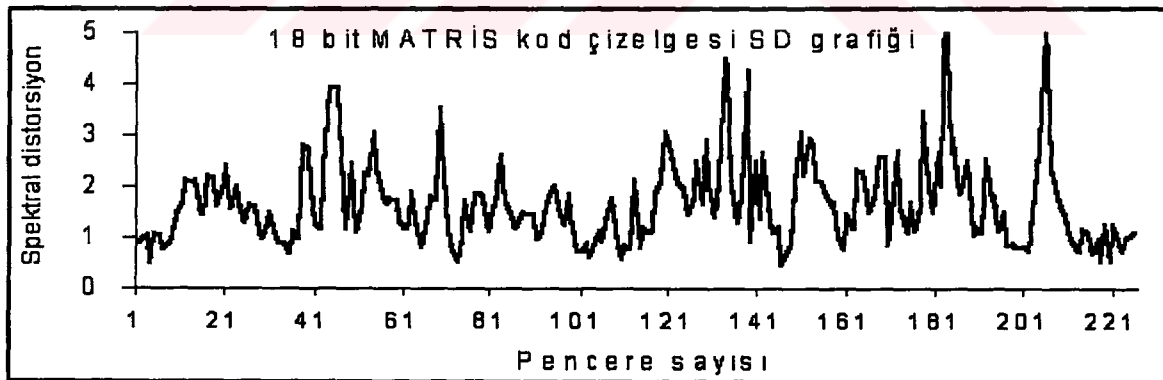
Şekil 6.7. Nicemlenmiş LSF vektörlerinin; a)25 bit MELP kod çizelgesi için b) 21-bit artık LSF kod çizelgesi için c)18 bit matris kod çizelgesi için spektral distorsiyon grafikleri. LSF analizde kullanılan ses örnekleri. "Oak is strong and also ..."



(a)

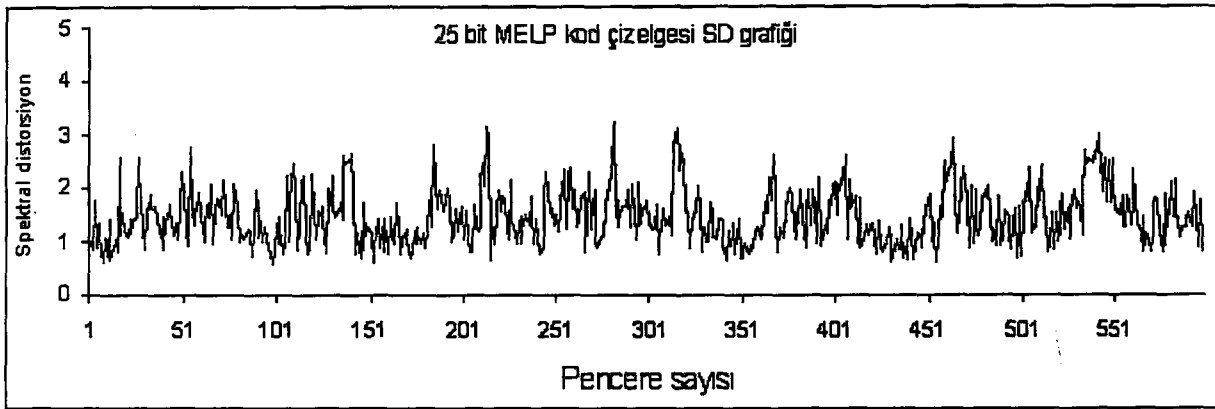


(b)

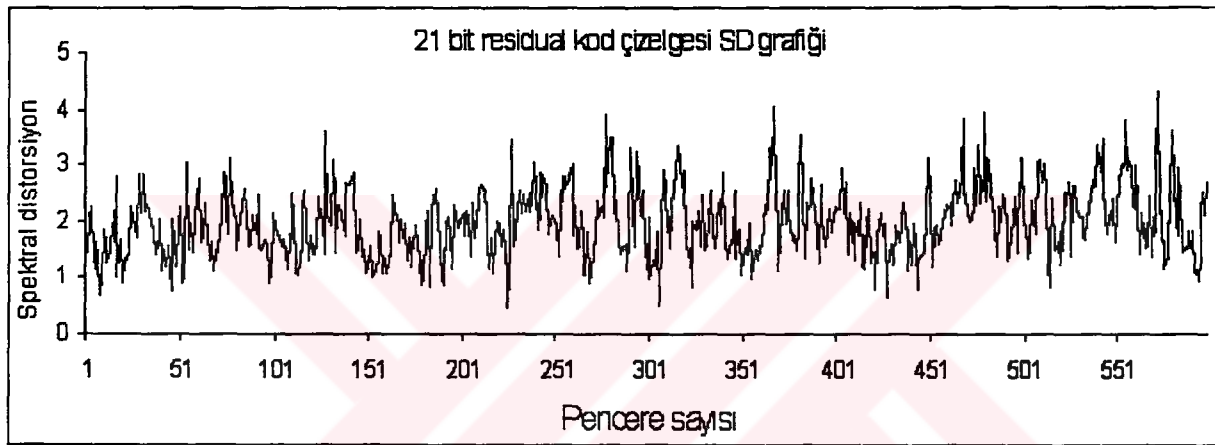


(c)

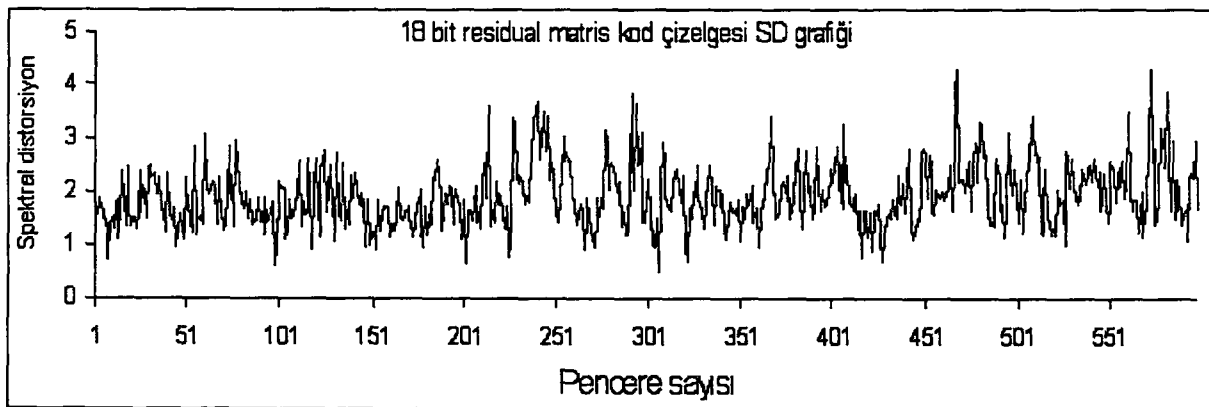
Şekil 6.8. Eğitim giriş dizisi içinde yer alan bir ses verisi için a) 25 bit MELP kod çizelgesi b) 21-bit residual kod çizelgesi c) 18 bit residual matris kod çizelgesi uygulandığında spektral distorsiyon (SD) grafikleri



(a)



(b)



(c)

Şekil 6.9. Test giriş dizisi içinde bir ses verisi için a) 25 bit MELP kod çizelgesi b) 21 bit residual kod çizelgesi c) 18 bit residual Matris kod çizelgesi için SD grafiği

7. FARKLI KODLAYICILAR ve PERFORMANS ÖLÇÜMLERİ

7.1. 2000 bit/sn MELP Konuşma Kodlayıcı

Bu tezde geliştirilen artık ÇAVN metodu ile oluşturulmuş bir 21 bit kod çizelgesi, MELP konuşma kodlayıcı üzerinde denenerek 2000 bit/sn MELP kodlayıcı yapısı oluşturulmuştur. Analiz pencere sayısı 22.5 msn'den 25 msn'ye çıkarılmıştır. Böylece sadece yeni geliştirilen kod çizelgesi ve analiz penceresi değerinin değiştirilmesiyle MELP kodlayıcı kalitesinin ne kadar değişebileceği gözlenmeye çalışılmıştır [109,110]. Yeni kodlayıcının bit yerleşimi ve pencere sayısı Çizelge 7.1'de görülmektedir. 2000 bit/sn MELP kodlayıcı ile yapılan subjektif test sonuçları Çizelge 7.7'de sunulmaktadır.

Çizelge 7.1. 2000 bit/sn MELP kodlayıcıda 25 msn analiz penceresi için parametre yerleşimi

Parametreler	Sesli	Bit hızı (kbit/sn)	Sessiz	Bit Hızı (kbit/sn)
LSF parametreleri	21	0.84	21	0.84
Fourier Genlikler	8	0.32	-	-
Kazanç (Her pencerede iki)	8	0.32	8	0.32
Perde, tüm sesli durumu	7	0.28	7	0.28
Bant geçiren kuvvetleri	4	0.16	-	-
Aperiyodik bayrak	1	0.04	-	-
Hata doğrulama	-	-	13	0.52
Senkronizasyon biti	1	0.04	1	0.04
Toplam bit/22.5 msn pencere	50	2000 bit/sn	50	2000 bit/sn

7.2. 1600 bit/sn Matris Nicemleme Tabanlı Konuşma Kodlayıcı

1200 bit/sn matris nicemleme tabanlı konuşma kodlayıcı (MQBC) için sesli/sessiz sinyal analizi üzerinde iyileştirme yapılarak, matris nicemleme tabanlı konuşma kodlayıcının kalitesinin ne kadar değişebileceği ölçülmeye çalışılmıştır. Bu amaçla, MELP konuşma kodlayıcıda kullanılan Fourier genlikler ve frekans bantlarına ayırma yöntemi 1200 bit/sn MQBC konuşma

kodlayıcıya ilave edilmiştir. Frekans bant sayısı MELP kodlayıcıda beş iken 1600 bit/sn kodlayıcı için dört'tür. 1600 bit/sn kodlayıcı bit yerleşimi Çizelge 7.2'deki gibidir. Kullanılan analiz penceresi uzunluğu 25 msn'dir. Frekans bantları (0-500, 500-1000, 1000-2000, 2000-4000) olarak ayrılmıştır. (0-500) frekans bandı için 1 yada 0 olma durumu, perde bilgisi ile birlikte kodlanmış, kalan üç bit ise kodlanarak sentez bölümüne gönderilmiştir. Aperyodik bayrak, sentez bölümünde V/UV sinyal incelenerek bulunmaktadır. Bu şekilde oluşturulan 1600 bit/sn kodlayıcı için dinleme testleri sonuçları 2400 bit/sn MELP kodlayıcıya çok yakın kalitede çıkmıştır.

Çizelge 7.2. 25 msn 1600 bit/sn MQBC pencereleri için parametre yerleşimi

Parametreler	1. pencere	2. pencere	Bit hızı (kbit/sn)
LSF parametreleri	36		0.72
Kazanç	5	5	0.20
Perde, tüm sesli durumu	6	6	0.24
Fourier Genlikler	8	8	0.32
Karma uyartım	3	3	0.12
Toplam bit	80		1.60

7.3. Kodlayıcı Subjektif Kalite Ölçümleri

Bir sistemin ses kalitesi hakkında objektif hesaplamalar bazı bilgiler içerir. Ama yine de, gerçek bir kalite ölçümü için subjektif testler gereklidir. Subjektif testler anlaşılabilirlik testi ve kalite testi olmak üzere iki temel sınıfa ayrılabilir. İki sınıf birbirinden ayrılmaz ölçüdedir. İyi kalite aynı zamanda iyi anlaşılabilirliği gerektirir. Körler için okuma makinesi gibi bazı uygulamalarda yüksek oranda anlaşılabilirlik, doğallıktan daha önemlidir. Diğer yandan, multimedya gibi uygulamalarda ise doğallık önemlidir. Test ölçümleri değişik seviyelerde yapılabilir. Örneğin, bilginin ihtiyaç duyulma şekline göre, sesbirimler (phoneme), kelime yada cümle kontrolleri ile ölçüm metotları

mevcuttur. Performans ölçümü için pek çok test olmasına rağmen, sistemin bütün performansı hakkında bilgi verebilecek tek bir test yoktur. [28,29,30]

7.3.1. Ses kalitesi ölçümü

Ses kalitesi belirlenmesinde en basit yaklaşımlardan birisi, çift karşılaştırma testidir (A/B testi). Bu test için her bir cümle iki ayrı kodlayıcıda işlenir ve dinleyiciye rasgele sunularak hangisinin daha iyi olduğu sorulur. Dinleyici A ya da B cümlesinin daha iyi olduğuna karar verir.

Subjektif test ölçümü için diğer bir test, DAM (Diagnostic Acceptibility Measure) testidir. Bir DAM testi uygulamak için, dinleyicilere ihtiyaç vardır. Dinleyiciler iyi bir performans ölçümü için çok iyi eğitilmiş olmalıdır. DAM test skorları 16 ayrı ölçeklendirme ile yapılır. Bazıları “fluttering”, “crackling”, “muffling”, “buzzing” ve “hissing” dir. DAM popülerliği, iyi parçalanmış parametrik ölçeklendirme nedeniyle. Yine de DAM test uzunluğu uygulanabilirlik için bir dezavantajdır.

Ses haberleşme sistemleri için başka bir yaygın olarak kullanılan subjektif kalite hesaplama metodu MOS (Mean Opinion Score) metodudur ve tüm ses kalitesini hesaplamada kullanılır. Bu metotta dinleyiciye kodlanmış sese Çizelge 7.3’de görüldüğü gibi [1-5] arası bir tamsayı değer vermesi istenir. Aynı tabloda DMOS (Degradation MOS), MOS ölçüm değerlerinin başka bir versiyonudur ve ses sinyalinde ne kadar rahatsız edici sinyal algılandığının bir ölçüsü bulunmaya çalışılır. DMOS için kesirli değerler verilebilmektedir bu yüzden MOS teste göre daha hassas ölçüm yapılabilir. Bu tezde DMOS ölçümü, sadece ofis ve araba gürültülü sesler için yapılmış ve arka plan gürültünün sese olan etkisi ölçülmeye çalışılmıştır.

Çizelge 7.3. MOS ve DMOS test ölçek değişimleri

Ölçek değerleri	MOS kalite ölçeklendirme	DMOS kalite ölçeklendirme
5	Mükemmel (excellent)	Bozulma farkedilmez (inaudible)
4	İyi (good)	Rahatsız etmeyen farkedilme (audible but not annoying)
3	Anlaşılır (fair)	Biraz rahatsız edici (slightly annoying)
2	Zayıf (poor)	Rahatsız edici (annoying)
1	Kötü (bad)	Çok rahatsız edici (very annoying)

7.3.2. Ses anlaşılabilirliği ölçümü

Ses anlaşılabilirlik testleri, dinleyicilerin birbirine yakın sesbirimlerini ayırabilmelerini sağlar. En yaygın kullanılan anlaşılabilirlik testi, DRT (Diagnostic Rhyme Test)'dir [30]. DRT bir ANSI (ANSI S3.2-1989) standardıdır ve bu testte dinleyicilere birbirine yakın söylenişli 96 kelime çifti sunulur ve hangi kelime sırasının söylendiği sorulur. Kelime çiftleri birbirinden sadece ilk harfle ayrılmaktadır. Bu kelime çiftleri altı özel sınıfa ayrılır ve tüm sistem anlaşılabilirliği için birlikte değerlendirilirler. DRT ölçümü Q ,

$$Q = \left[\frac{R - W}{T} \right] \cdot 100 \quad (7.1)$$

ile belirlenir. R , doğru cevap sayısı, W yanlış cevap sayısı ve T , toplam dinleyici sayısıdır. Tipik DRT değişimleri 75 ile 95 arasındadır. İyi bir DRT skoru 90 üzeri olmalıdır. DRT testleri temiz ses için sınırlandırılmıştır. Herbir DRT skoru bir performans kategorisine sahiptir. Bu kategoriler eğer (95-100) arası ise “excellent”, (87-95) arası ise “good”, (79-87) arası ise “anlaşılır”, (70-79) arası ise “zayıf” kaliteyi gösterir. 70 aşağısı kabul edilemez ve kötü kaliteyi gösterir. DRT testlerinde dinleyici etkisi büyüktür.

DRT, yaygın olarak kullanılan bir metottur ve kelimenin ilk harfindeki ünsüzlerin ne kadar anlaşılır olduğunu göstermede faydalı bir metottur. Fakat herhangi bir ünlü harf testi yapmadığı için tüm kalite hakkında karar verilebilecek bir metot değildir. Diğer bir yetersizliği, test malzemesi oldukça sınırlıdır ve test parçaları bütün olasılıkları içermez.

7.3.2.1. İngilizce DRT test tablosu

96 kelime çifti içeren İngilizce DRT test tablosu Çizelge 7.4’de görülmektedir. Kelime çiftleri sistem eksikliğini daha iyi tespit için altı ayrı sınıfta incelenmektedir. Bütün skor sonuçları birlikte toplanıp değerlendirilir.

Çizelge 7.4. 192 DRT test İngilizce kelime çiftleri

Voicing		Nasality		Sustentation		Sibilation		Graveness		Compactness	
Veal	feel	meat	beat	vee	bee	Zee	thee	weed	reed	yield	wield
bean	peen	need	deed	sheet	cheat	cheep	keep	peak	teak	key	tea
gin	chin	mitt	bit	vill	bill	jilt	gilt	bid	did	hit	fit
dint	tint	nip	dip	thick	tick	sing	thing	fin	thin	gill	dill
zoo	sue	moot	boot	foo	pooh	juice	goose	moon	noon	coop	poop
dune	tune	news	dues	shoes	choose	chew	coo	pool	tool	you	rue
vole	foal	moan	bone	those	doze	joe	go	bowl	dole	ghost	boast
goat	coat	note	dote	though	dough	sole	thole	fore	thor	show	so
zed	said	mend	bend	then	den	jest	guest	met	net	keg	peg
dense	tense	neck	deck	fence	pence	chair	care	pent	tent	yen	wren
vast	fast	mad	bad	than	dan	jab	gab	bank	dank	gat	bat
gaff	calf	nab	dab	shad	chad	sank	thank	fad	thad	shag	sag
vault	fault	moss	boss	thong	tong	jaws	gauze	fought	thought	yawl	wall
daunt	taunt	gnaw	daw	shaw	chaw	saw	thaw	bong	dong	caught	thought
jock	chock	mom	bomb	von	bon	jot	got	wad	rod	hop	fop
bond	pond	knock	dock	vox	box	chop	cop	pot	tot	got	dot

7.3.2.2. Türkçe DRT test tablosu

96 kelime çifti içeren Türkçe DRT test tablosu Çizelge 7.5’de görülmektedir¹. Kelime çiftleri sistem eksikliğini daha iyi tespit için altı ayrı sınıfta incelenmektedir. Bütün skor sonuçları birlikte toplanıp değerlendirilir.

Çizelge 7.5. 192 DRT test Türkçe kelime çiftleri

Voicing (seslilik)	Nasality (burunsal)	Sustentation (duraklamalı)	Sibilation (ıslıklamalı)	Graveness (sertlik)	Compactness (bitişiklik)
van fan	mal bal	veri beri	zam tam	pas tas	yer ver
ver fer	mil bil	var bar	çar kar	bul dul	yan van
bir pir	nil dil	şark çark	çil kil	bar dar	kat tat
bak pak	nem dem	şile çile	çok kök	beri deri	kel tel
bul pul	göl döl	fas pas	çek kek	fiş diş	fas has
cam çam	gam dam	fes pes	jel gel	meşe neşe	hakir fakir
cin çin	geri deri	fil pil	sur dur	fide side	çat pat
zar sar	nar dar	tek dek	sahil dahil	say fay	çile pile
zil sil	gem dem	tam dam	saç taç	mil nil	cadı dadı
zor sor	melek belek	tiz diz	semiz temiz	but dut	şer ser
beş peş	nur dur	dümen tümen	ser ter	mal nal	şal sal
zevk sevk	nem dem	fare pare	sim tim	mey ney	şap sap
gaz kaz	muz buz	şal çal	söz göz	fal sal	gel bel
gül kül	mey bey	şut but	cam gam	pek dek	göl döl
göz köz	gün dün	den ten	salı halı	ser fer	çat pat
gaf kaf	giz diz	dur tur	ser her	sil fil	çıpa tıpa

7.4. MELP Kodlayıcılar ve MQBC Kodlayıcılar için Subjektif Testler

Bu tezde subjektif kalite ölçümü için MOS, DMOS ve DRT testleri uygulanmış, dinleyicilere İngilizce ve Türkçe ses kayıtları sunulmuştur. İngilizce konuşma kayıtları, pek çok kodlayıcı testi için yaygın olarak kullanılan cümlelerdir ve ses kayıtları hazır olarak kullanılmıştır. Türkçe ses kayıtları ise, ilk defa bu tezde seçilmiş ve bilgisayar ortamında kaydedilerek hazırlanmıştır. Testlerde uygulanan İngilizce ve Türkçe ses kayıtları Ek-3’de sunulmaktadır.

¹ Yazar, bu kelime çiftlerinin hazırlanmasında ve Türkçe akustik sesbilim konularında değerli görüşlerinden faydalandığı Sn. Prof.Dr. Mehmet Demirezen'e çok teşekkür eder.

7.4.1. İngilizce cümleler ile MOS test sonuçları

2.4 kbps MELP ve 1.2 kbps MQBC kodlayıcı ses kalite ölçümleri yapılmıştır. Bunlar, bir giriş ses datası ölçeği 5 alınarak ölçeklendirilmiştir. Giriş ses sinyali 3 farklı kadın ve 3 farklı erkek konuşmacı için 12 farklı ses cümlesi içermektedir. Konuşmacı giriş şartları üç farklı ortam için denenmiştir:

- Temiz ortamda kaydedilmiş ses sinyali
- Ofis ortamında kaydedilmiş ses sinyali
- Araba gürültülü ortamda kaydedilmiş ses sinyali

Ofis ortamı ve gürültülü ses sinyali için DMOS derecelendirme, temiz ses için MOS derecelendirme kullanılmıştır. On farklı dinleyici herbir ses cümlesini ayrı ayrı derecelendirmiş ve sonuç skorlar, bu değerlerin hepsinin ortalaması alınarak bulunmuştur. Sonuçlar Çizelge 7.6'da görülmektedir.

Çizelge 7.6. MOS test sonuçları

	Mean Opinion Score (MOS)		
	Sessiz	Ofis Gürültüsü	Araba Gürültüsü
2.4 kbps LPC-10	2.3	2.1	1.3
2.4 kbps MELP	3.5	3.0	2.7
1.2 kbps MQBC	3.1	2.7	2.2

Derecelendirme sonucunda, bütün kodlayıcılarda erkek seslerinin daha iyi derecelendirildiği gözlenmiştir. MQBC kodlayıcı kalitesi MELP kodlayıcıdan biraz kötü, LPC kodlayıcıya göre çok iyi çıkmıştır. Gürültülü ortamda MQBC kodlayıcı kalitesi biraz düşmektedir.

2400 bit/sn MELP konuşma kodlayıcı, 2000 bit/sn MELP konuşma kodlayıcı, 1600 bit/sn MQBC konuşma kodlayıcı ve 1200 bit/sn MQBC konuşma

kodlayıcı için dinleme testi sonuçları Çizelge 7.7’de görülmektedir. Dinleme testleri için üç kadın ve üç erkek olmak üzere altı değişik konuşmacı ve 12 farklı cümlelerin konuşulduğu temiz İngilizce ses kayıtları kullanılmıştır. Bu ses kayıtları için, bir önceki MOS testinde bulunmamış on farklı dinleyici testlere katılmıştır. Test sonuçları kadın ve erkek konuşmacıların ortalama puanları ve toplam ortalama puanlar olmak üzere çizelgede gösterilmiştir. Bütün kodlayıcılarda belirgin olarak erkek konuşmacıların puanları daha yüksek çıkmıştır.

Çizelge 7.7. Değişik kodlayıcılar için İngilizce MOS test sonuçları

Kodlayıcı Türü	Erkek-1	Kadın-1	Erkek-2	Kadın-2	Erkek-3	Kadın-3	TOPLAM
2400 bit/sn MELP	3.9	3.6	3.7	3.6	3.3	3.1	3.53
2000 bit/sn MELP	3.6	3.5	3.4	3.5	3.5	3.3	3.46
1600 bit/sn MQBC	3.3	3.1	3.2	3.5	3.2	3.2	3.25
1200 bit/sn MQBC	3.0	2.5	2.8	2.7	3.1	2.7	2.80
2400 bit/sn LPC-10	2.1	1.8	2.0	2.3	2.2	2.1	2.08

7.4.2. Türkçe cümleler ile MOS ve DRT test sonuçları

2400 bit/sn MELP kodlayıcı, 2400 bit/sn LPC-10 kodlayıcı ve 1200 bit/sn MQBC kodlayıcılar üzerinde, Çizelge 7.4’deki Türkçe DRT testi ve Türkçe MOS testi için bilgisayar ortamında kaydedilmiş ses dataları ile testler yapılmıştır. Ses dataları 44100 Hz ile örneklenerek bilgisayarda kaydedilmiş ve test dataları için bu ses dataları 8000 Hz’e indirgenmiştir. Ses dataları üzerinde kayıt ortamından kaynaklanan ve tam olarak yok edilememiş gürültüler mevcuttur. Testlere 10 dinleyici katılmıştır. MOS testi için 4 kadın konuşmacı ve 4 erkek konuşmacı ikişer Türkçe cümle konuşmaktadır. Dinleyicilerden herbir kadın konuma cümlesi ve erkek konuşma cümlesi için MOS puanları vermeleri istenmiştir. Sonuçlar Çizelge 7.8’de sunulmaktadır. Türkçe konuşma cümleleri için de MQBC kodlayıcı, LPC-10 kodlayıcı ve 2400 MELP kodlayıcı arası kalitede çıkmıştır.

Çizelge 7.8. Değişik kodlayıcılar için Türkçe MOS test sonuçları

Kodlayıcı Türü	Erk-1	Kdn-1	Erk-2	Kdn-2	Erk-3	Kdn-3	Erk-4	Kdn-4	TOPLAM
2400 bit/sn MELP	4	2.8	3.2	3.2	4	2.6	3.9	3.2	3.3375
1200 bit/sn MQBC	3.2	2.5	3.2	2.3	2.7	2.6	2.8	2.9	2.8375
2400 bit/sn LPC-10	2.5	1.8	2.9	2.3	2.5	2.2	2.3	2.2	2.0833

Türkçe DRT testi için dinleyiciler, Çizelge 7.4'deki sırada fakat rasgele sıralamada hazırlanan ses datalarının hangisinin daha önce söylendiğini tespit etmeye çalışmışlardır. Orijinal ses dataları dinleyicilere dinletilmemiştir. Kendilerinden ses kayıtlarını sadece bir kez dinlemeleri ve kesin olarak ayıramadıkları söylenişleri işaretlememeleri istenmiştir. Bu şekilde yapılan DRT test sonuçları için hatalı skorlar Çizelge 7.9'da sunulmaktadır. DRT test sonuçları ise Çizelge 7.10'da sunulmaktadır. Test sonuçları sunulurken, Çizelge 7.4'deki sınıflandırma yapısı içinde hangi gruplarda daha çok hata yapıldığı da tespit edilmeye çalışılmıştır. Buna göre, en çok hata yapılan grubun "graveness" grubu içinde olduğu ve 'm-n', 'b-d', 's-f' seslerinin ayırımında güçlük çekildiği tespit edilmiştir. Orijinal söylenişlerde de çok düzgün bir ayırım yapılamamakla birlikte, yukarıdaki ses benzeşmeleri doğrusal öngörülü kodlayıcıların tekrar oluşturmakta güçlük çektiği söylenişlerdir. Çok nadir olarak MELP kodlayıcıda 'g-d', 'z-s', 'v-b', 'f-h' ses benzeşmeleri, MQBC kodlayıcıda ise 'g-d', 'ş-ç', 'f-h' ses benzeşmeleri ayırt edilememiştir. Türkçe DRT testi sonuçlarında 1200 bps MQBC kodlayıcı, anlaşılabilirlik yönünden en yüksek çıkmıştır. LPC-10 kodlayıcıda ise 'z-s', 'm-b', 'n-d', 'g-d', 'ş-ç', 'f-p', 't-d', 's-t', 'b-d', 'f-d', 'm-n', 'f-s', 'p-d' sesleri hatalı kodlanmıştır.

Çizelge 7.9. Değişik kodlayıcılar ile Türkçe DRT test sonuçları için on dinleyicinin işaretlediği toplam hata sayısı

Kodlayıcı Türü	Voicing	Nasal	Sustene	Sibile	Grave	Compact	TOPLAM hata	TOPLAM hata %
2400 bit/sn MELP	2	3	3	2	15	-	25	2.56
1200 bit/sn MQBC	-	2	1	-	16	1	20	2.08
2400 bit/sn LPC-10	2	9	7	2	28	11	59	6.15

DRT skor hesabı $Q = (R-W)/T = (96 \times 10 - \text{toplam yanlış cevap}) / (96 \times 10)$ ile hesaplanmıştır.

Çizelge 7.10. Değişik kodlayıcılar ile on dinleyici için Türkçe DRT % test sonuçları

Kodlayıcı Türü	2400 bit/sn MELP	1200 bit/sn MQBC	2400 bit/sn LPC-10
DRT %	97.4	97.9	93.8

7.5. Performans

MQBC ve MELP kodlayıcılarda parametre nicemlenmesi sırasında yapılan nicemleme hatası, bir test datası için MSE ölçümü yapılarak hesaplanmıştır. Eğer işleme sokulan parametre x ve nicemlenmiş parametre $\hat{x}$ ise ve T test giriş datasının uzunluğu ise, MSE (6.11) formülündeki gibi hesaplanabilir.

$$MSE = (\sum (x - \hat{x})^2) / T \quad (7.2)$$

Test giriş datasının pencere uzunluğu, herbir analiz penceresi uzunluğu 22.5 msn ise $T = 28400$ 'dür. MELP ve MQBC kodlayıcıda G_1 kazancı kodlanma metodu ve kullanılan bit sayısı aynıdır. Buna göre, MELP kodlayıcıda

hesaplanan G_2 ara pencere kazancı ve $\hat{G}_2$ arasındaki nicemleme hatası, ortalama $MSE = 0.978$ çıkmıştır. MQBC kodlayıcıda sentezde hesaplanan G_x kazancı ve G_1 arasındaki hata $MSE = 4.265$ çıkmıştır. G_2 değeri sentezde $G_{avg} = (G_2p + G_2)/2$ ile hesaplanırsa, G_2 ve G_{avg} arasındaki $MSE = 4.684$ çıkmıştır.

Perde periyodu nicemleme hatası, 7 bit nicemleme aralığı kullanan MELP kodlayıcı için ortalama $MSE = 0.228$, 6 bit nicemleme aralığı kullanan MQBC kodlayıcı için $MSE = 0.477$ çıkmıştır.

MELP ve MQBC kodlayıcılarda kod çizelgeleri performansını ölçmek için eğitim ve test giriş dizileri için SD grafikleri çıkarılmıştır. 18 bit/pencere MQBC kodlayıcı kod çizelgesi, eğitim giriş dizisi içindeki ses kaydı için MELP kodlayıcı kod çizelgesi ile yaklaşık aynı performansı verirken, test giriş dizisi için MQBC kod çizelgesi performansı kötüleşmiştir (Bkz. Şekil 6.6-6.9).

7.6. Birlikte Çalışabilirlik

Birlikte çalışabilirlik, bir konuşma kodlayıcının taşınması gereken özelliklerden biridir. Bu bölümde, 2400 bit/sn MELP kodlayıcı ve 1200 bit/sn MQBC kodlayıcının birlikte çalışabilirliği incelenmiştir. MELP kodlayıcıda alıcıda kodlanarak gönderilen parametreler MQBC sentezleyicide kullanılacak ise önce kodlanmış parametrelerin çözülmesi gereklidir. 2400 bit/sn MELP kodlayıcı parametreleri ve 1200 bit/sn MQBC kodlayıcı parametreleri Çizelge 7.11'de sunulmaktadır. MELP kodlayıcıda kodlanarak MQBC'de sentezlenecek kodlayıcı parametreleri içinde bazı bilgilerin yok farzedilmesi yada sabit değer olarak alınması gereklidir. Örneğin Fourier genlikler hesaba katılmayıp sabit değer 1 olarak alınırlar. Yine senkronizasyon biti ve hata kodlama bitleri MQBC sentez bölümü için kullanılmayan bilgilerdir. Aperyodik bayrak, perde bilgisi ve V/UV analizden çıkarıldığı için bu veri biti

de dikkate alınmamaktadır. MELP kodlayıcıdan gelen kazanç bilgisi G_1, G_2 kazançlarından 5 bitlik G_1 kazancı kullanılmakta, G_2 kazancı dikkate alınmamaktadır. LSF parametreleri MELP kodlayıcı tablosunda çözülüp yaklaşık LSF değerleri bulunarak alınır. Perde bilgisi, perde kod çözücü tablosuna göre çözülerek kullanılır. MELP kodlayıcıdan gelen ve 5 bitlik sesli/sessiz algoritması ise, MQBC kodlayıcıda iki bitlik V/UV bilgisi kullanıldığından Çizelge 7.12'deki durum bilgisine göre iki bite indirgenir ve kullanılır. MELP ve MQBC kodlayıcıların sentez bölümleri çok benzer özellikler taşıdığından MELP kodlayıcıda kodlanan parametreler yukarıdaki şekilde alınarak MQBC kodlayıcıda sentezlenebilmektedir.

Çizelge 7.11. 2400 bit/sn MELP kodlayıcı ve 1200 bit/sn MQBC parametre yerleşimi

Parametreler	2400 MELP		1200 MQBC	
	Sesli	Sessiz	1.pencere	2.pencere
LSF parametreleri	25	25	36	
Fourier Genlikler	8	-	-	-
Kazanç (Her pencerede iki)	8	8	5	5
Perde, tüm sesli durumu	7	7	6	6
Band geçiren kuvvetleri	4	-	1	1
Aperiyodik bayrak	1	-	-	-
Hata doğrulama	-	13	-	-
Senkronizasyon biti	1	1	-	-
Toplam bit/22.5 msn pencere	54	54	60	

Çizelge 7.12. MELP kodlayıcı sesli/sessiz sinyallerin MQBC kodlayıcı sesli/sessiz sinyallerine dönüştürülmesi

MELP sesli/sessiz	MQBC sesli/sessiz
$V_0 V_1 V_2 V_3 V_4$	$V_0 V_1$
0 x x x x	0 0
1 x x 0 x	1 0
1 x x 1 x	1 1

MQBC kodlayıcıda alıcıda kodlanarak gönderilen parametreler MELP sentezleyicide kullanılacak ise önce kodlanmış parametrelerin çözülmesi gereklidir. MQBC kodlayıcıda kodlanarak MELP’de sentezlenecek kodlayıcı parametreleri içinde, MELP kodlayıcıda beklenen bazı bilgiler yoktur. Örneğin Fourier genlikler yoktur ve sabit değer 1 olarak alınırlar. Yine senkronizasyon biti ve hata kodlama bitleri MQBC kodlayıcıdan gönderilmeyen bilgilerdir. Aperyodik bayrak, MELP sentezleyicide perde bilgisi ve V/UV analizden çıkarılmalıdır. MELP kodlayıcıdan gelen kazanç bilgisi $G1$ yanında, MELP sentezleyicinin beklediği $G2$ kazancı, $G2 = (G1 + G1_{previous})/2$ ile hesaplanır. LSF parametreleri MQBC kodlayıcı tablosunda çözülüp yaklaşık LSF değerleri bulunarak alınırlar. Perde bilgisi, perde kod çözücü tablosuna göre çözülerek kullanılır. MQBC kodlayıcıdan gelen ve 2 bitlik sesli/sessiz algoritması ise, MELP kodlayıcıda beş bitlik V/UV bilgisi kullanıldığından Çizelge 7.13’deki durum bilgisine göre beş bite çıkarılır ve kullanılır.

Çizelge 7.13. MELP kodlayıcı sesli/sessiz sinyallerin MQBC kodlayıcı sesli/sessiz sinyallerine dönüştürülmesi

MQBC sesli/sessiz	MELP sesli/sessiz
V_0V_1	$V_0V_1V_2V_3V_4$
0 0	0 0 0 0 0
1 0	1 1 1 0 0
1 1	1 1 1 1 1

7.7. 1200 bit/sn MF_MELP Konuşma Kodlayıcı

1200 bit/sn MF_MELP kodlayıcı [94] algoritmasında analiz penceresinde, her 22.5 msn’de 2400 bit/sn standart MELP kodlayıcı yapısı ile elde edilen parametreler, bu üç pencerenin birleşimi olan 67.5 msn bir süper pencerede

gruplanır ve birleşik olarak nicemlenir. Büyük perde hatalarından kaçınmak için kullanılan bir perde düzleştirici nedeniyle 129 sample gecikme oluşur. Bu yüzden 1200 bit/sn MF_MELP kodlayıcı için toplam gecikme 103.75 msn'dir. Süper pencere yapısını etkin şekilde kodlayabilmek için vektör nicemleme ve aradeğerleme metodları kullanılır. Sesli ve sessiz pencerelerin istatistiksel özellikleri dikkate alınır. Doğru olmayan bir tespit sentezlenmiş seste büyük hatalara neden olacağından kanal hatalarına karşı çok iyi koruma gerektirmektedir.

Perde nicemleme : Perde bilgisi sadece sesli pencereler için aktarılır. Farklı sesli/sessiz durumları için farklı nicemleme şemaları kullanılır. İki yada üç sesli pencere içeren süper pencerelerde perde parametreleri önce logaritmik değere çevrilir. Logaritmik perde değerleri, distorsiyon hesabına göre hazırlanmış perde vektör nicemleyici kod çizelgesi ile vektör nicemlenir. Kod çizelgesi içindeki en uygun perde değerleri araştırması yine bir distorsiyon hesabına göre yapılır. Sadece bir sesli pencere içeren süper pencerede, sesli pencere perde değeri için standart MELP kodlayıcıdaki skalar nicemleme kullanılır.

Perde ve sesli/sessiz kararların birlikte nicemlenmesi : Süper penceredeki sesli/sessiz kararları ve perde değerleri 12 bit kullanarak birlikte nicemlenir (Çizelge 7.14). 12 bit birleşimi, sekiz farklı sesli/sessiz durumunu içeren 3 mod biti ve 9 perde biti içerir. 6 ayrı perde kod çizelgesi vardır. 5 tanesi 9-bit'e sahiptir, bir tanesi ise 2400 bit/sn MELP kodlayıcıda bulunan skalar nicemleyicidir. Önce sesli/sessiz durumu 3 bit ile kodlanır. Bu üç bit ile 6 farklı perde kod çizelgesinden biri seçilir. Sadece bir pencerenin sesli olduğu veya bütün pencerelerin sessiz olduğu ilk dört durum için, farklı modlar arasındaki ayırım perde bilgisi ile belirlenir.

Çizelge 7.14. Perde ve sesli/sessiz kararların birlikte nicemlenmesi

Sesli/sessiz kararlar	3-bit CB	9-bit CB
UUU	000	Perde değeri 2400 bit/sn MELP kodlayıcısındaki 99-seviye perde nicemleme tablosu ile nicemlenir. Perde değeri ve sessi/sessiz durumu 9-bit kod çizelgesine haritalanır.
UUV		
UVU		
VUU		
VUU	001	Bu sesli/sessiz durumları, 512 kod vektör içeren perde kod çizelgesini kullanır.
VUV	010	
UVV	100	
VVV	011	512-seviye kod çizelgesi A
	101	512-seviye kod çizelgesi B
	110	512-seviye kod çizelgesi C
	111	512-seviye kod çizelgesi D

LSF nicemlenmesi : Üç analiz penceresinden çıkarılan LSF vektörler l_1 , l_2 ve l_3 olsun. UUU, UUV, UVU ve VUU durumları için, sessiz pencerelerin LSF vektörleri 9-bit kod çizelgesi sesli pencerelerin LSF vektörleri 25-bit standart MELP kod çizelgesi ile nicemlenir (Çizelge 7.15). Diğer sesli/sessiz pencerelerin LSF vektörleri bir ileri-geri aradeğerleme kullanılarak nicemlenir [94].

Çizelge 7.15. LSF nicemleme için bit yerleşimi

V/U Örüntü	LSF l_1	LSF l_2	LSF l_3	aradeğerleme katsayısı	l_1, l_2 aradeğerleme	Toplam
UUU	9	9	9	0	0	27
VUU	7 6 6 6	9	9	0	0	43
UVU	9	7 6 6 6	9	0	0	43
UUV	9	9	7 6 6 6	0	0	43
UVV	0	0	7 6 6 6	4	8 6	43
VUV						
VVV						
VVU	0	0	9	4	8 6 6 6	39

Bit yerleşimi: 1.2 kbit/sn MF_MELP kodlayıcı bit yerleşimi Çizelge 7.16'deki gibidir. Kodlayıcıda, her analiz penceresinde iki kazanç parametresi hesaplanır ve logaritmik değere çevrilir. Üç pencere için çıkarılan toplam 6 kazanç parametresi, bir süper pencerede 10-bit kazanç nicemleyici kod çizelgesinde nicemlenirler.

Her pencerede 5 bantta elde edilen V/U ses kararları için sadece sesli bandlar dikkate alınır ve ilk band ve kalan 4 band, 2 bit ile nicemlenirler. Fourier genlikler, sadece sesli pencereler için hesaplanır ve süper penceredeki son sesli pencere için bulunan Fourier genlikler 8 bit ile nicemlenirler; diğer pencereler için aradeğerleme ile hesaplanırlar. Aperyodik bayrak, sadece sesli pencerelerden elde edilir.

Çizelge 7.16. 1.2 kbit/sn MF_MELP kodlayıcı için 67.5 msn süper penceredeki bit yerleşimi

Parametreler	Süper pencere U/V örüntüleri				
	VVV	UVV VUV	VVU	UUV UVU VUU	UUU
Perde & U/V kararları	12	12	12	12	12
LSF'ler	43	43	39	43	27
Kazançlar	10	10	10	10	10
Bant geçirme güçleri	6	4	4	2	0
Fourier Genlikler	8	8	8	8	0
Aperyodik bayrak	1	1	1	1	0
Senkronizasyon	1	1	1	1	1
Hata koruma	0	2	6	4	31
Toplam	81	81	81	81	81

Test Sonuçları : MF_MELP kodlayıcı için iki farklı performans ölçümü yapılmıştır. Bunlardan birisi, anlaşılabilirlik için DRT testi ve kalite ölçümü için DAM testidir. Testler temiz ses, HMMWV (High Mobility Multi purpose Wheeled Vehicle) çevre gürültüsü ve 0.5% rastgele kanal hataları için yapılmıştır. DRT ve DAM test sonuçları sırasıyla Çizelge 7.17 ve Çizelge 7.18'de görülebilir.

Çizelge 7.17. DRT test sonuçları

Kodlayıcı	DRT Skorları		
	Quiet	HMMWV	0.5% BER
2.4 kbit/sn MELP	93.8	72.5	93.4
1.2 kbit/sn MF_MELP	92.1	70.2	86.3

Çizelge 7.18. DAM test sonuçları

Kodlayıcı	DAM Skorları	
	Quiet	HMMWV
2.4 kbit/sn MELP	68.2	52.2
1.2 kbit/sn MF_MELP	61.8	48.9

Test sonuçlarından görülebileceği gibi, anlaşılabilirlik iyi çıkmış fakat ses kalitesi azalmıştır. Kanal hatalarında MF_MELP anlaşılabilirliği azalmaktadır. aktarım hatalarına karşı kalite artırmanın etkin yollarından biri hata koruma bitlerini arttırmaktır.

7.8. 1200 bit/sn MQBC ve 1200 bit/sn MF_MELP Kodlayıcılar Arasındaki farklar

1200 bit/sn MF_MELP kodlayıcı, standart 2400 bit/sn MELP kodlayıcı algoritmasını değiştirmeden, sonuçta kodlanacak parametreler üzerinde yapılan işlemlerle bit sayısını azaltmayı amaçlar. Üç pencerenin birleşimiyle oluşan süper pencere ile kodlayıcı gecikmesi 103.75 msn'dir. Bu değer 1200 bit/sn MQBC kodlayıcı için 50 msn'dir. Ayrıca üç pencerenin birleşimi içerisinde kodlanan perde parametreleri için hatalı oluşabilecek sesli/sessiz kararları, perde değerini de direkt olarak etkileyeceğinden sonuç sentezlenmiş seste bozulmalar olacaktır. MQBC kodlayıcıda perde parametreleri, her pencerede kodlanarak karşı tarafa gönderilmektedir. 1200 bit/sn MF_MELP kodlayıcı için sesli ve sessiz pencerelerde ayrı olarak nicemlenen LSF parametrelerinin, 2400 bit/sn MELP kodlayıcıdaki LSF parametrelerine göre performans karşılaştırması yapılmamış olduğundan, kalitesi hakkında yorum yapılamamaktadır. Ayrıca MF_MELP kodlayıcı için, sentezleyici bölüm hakkında da bir bilgi mevcut değildir. Sentezleyici bölümün MELP kodlayıcı ile aynı olduğu düşünülmeyle beraber, her üç pencereden aktarılan bilgilerin sentezleyicide nasıl sınıflandırıldığı net olarak bilinmemektedir. Yine

MF_MELP kodlayıcıda kullanılan pencerelerin sesli/sessiz olarak ayrımını yapan algoritma hakkında da net bir bilgi yoktur. MQBC kodlayıcı sesli/sessiz ayrımı için Chambel ve Tramain'in önerdiği algoritmayı kullanmaktadır ve bu metodun doğruluğu %2 civarındadır [48].

7.9. MQBC Kodlayıcıda Kazanç ve Perde Peryodu için Matris Nicemleme Metodu

Kazanç ve perde peryodu için matris nicemleme metodu denenmiştir. 28400 analiz penceresi içeren eğitim datası kullanılarak eğitim kazanç giriş datası ve eğitim perde peryodu giriş datası hazırlanmıştır. İki pencerenin birleştirilmesi ile hazırlanması düşünülen kazanç giriş vektör tablosu için, eğitim giriş datası kullanılarak 7 bitlik bir kazanç kod çizelgesi oluşturulmuştur. Sonuç distorsiyon değeri “dr=1.34” civarında çıkmıştır. Kazanç kod çizelgesi MQBC programında uygulanmış ve analiz bölümünde hesaplanan kazanç değerleri ve kod çizelgesinde seçilen değerler karşılaştırılmıştır. Büyük kazanç değişimlerinin olduğu kısımlarda hesaplanan ve kodlanan kazanç değerleri arasında büyük hatalar olduğu gözlenmiştir. Perde peryodu için iki pencerenin birleştirilmesiyle oluşturulan 7 bitlik perde kod çizelgesi için distorsiyon değeri “dr=7.68” gibi büyük bir değer çıktığı için matris yapıda oluşturulan perde kod çizelgesinin verimli olamayacağı sonucuna varılmıştır.

7.10. MSMQ Kod çizelgesi Arama ve Kodlayıcı Karmaşası

Bir K aşamalı ÇAMAN kod çizelgesi için, herbir kod çizelgesi $m.p.L_j$ eleman içerir ($j=1,2,...,K$). Toplam kod çizelgesi eleman sayısı, $\sum_{j=1}^K m.p.L_j$ 'dir. Bir ÇAMAN kod çizelgesi için, karmaşa değeri depolama ihtiyacı ile direkt orantılıdır. bilinen ağaç aramalı ÇAVN kod çizelgesi için, arama performansı

tam aramaya yakındır. MQBC kodlayıcıda kullanılan 18 bit/pencere ÇAMAN kod çizelgesi için SD, 1 dB civarındadır ve 4 dB'den büyük aykırı değer yoktur. 2 dB'den büyük aykırı değer sayısı %2 'den küçüktür.

ÇAVN metodu için kullanılan arama karmaşası formülü türetilerek [45], m pencerenin birleşimiyle oluşturulan bir ÇAMAN kod çizelgesi için arama karmaşası hesaplanmıştır.

$$\hat{M}_j = \min \left(M, \prod_{k=0}^{j-1} L_j \right) \quad (7.3)$$

$$C_{MSMQ} = \log_2 \left(3.p.m.L_0 + m.p \sum_{j=1}^{K-1} (\min(L_j, \hat{M}_j) + (\hat{M}_j + 2)L_j) \right) \quad (7.4)$$

ÇAMAN kod çizelgeleri, ÇAVN kod çizelgelerine göre önemli miktarda bit kazanımı sağlamaktadır. Fakat, arama ve depolama karmaşaları yüksektir. Örneğin, 4 aşamalı 18 bit/pencere [10,9,8,8] ÇAMAN kod çizelgesi 46080 eleman içermektedir ve arama karmaşası (7.4) formülüne göre 17,2'dir. Buna rağmen, 25 bit/pencere 4 aşamalı [7,6,6,6] ÇAVN kod çizelgesi 3200 eleman içermektedir ve arama karmaşası 14,5'dir.

MELP kodlayıcılar ve MQBC kodlayıcılar için kodlayıcı C kodlarının çalışma süreleri, 150 MHz pentium bilgisayarda Borland C ile çalıştırılıp karşılaştırılmıştır. Kodlayıcılar, 356 pencere içeren kadın sesi ve 356 pencere içeren erkek sesi için 5'er kez çalıştırılıp bulunan sürelerin ortalamaları alınmıştır. Çizelge 7.19'da görüleceği gibi, 1200 bit/sn MQBC kodlayıcı, 2400 bit/sn MELP kodlayıcıya göre her pencerede ortalama iki kat süre harcamaktadır.

Çizelge 7.19. Farklı kodlayıcıların her pencerede harcadığı ortalama süre

Kodlayıcı	1200bpv	1200MQBC	1600MQBC	2000MELP	2400MELP
Çalışma süresi/pencere(ms)	41	51	43	27	26

8. SONUÇ ve ÖNERİLER

Sonuç

Doğrusal öngörülü kodlama metoduna dayalı, 1200 bit/sn ve 1600 bit/sn bit hızlarında çalışabilen yeni bir konuşma kodlayıcı (MQBC) dizayn edilmiştir. Kodlama tablosu içinde geniş yer tutan filtre parametreleri kodlanması için vektör nicemleme, matris nicemleme yaklaşımları ve bu metotlarla oluşturulan kod çizelgeleri sunulmuştur. Yeni geliştirilen artık ÇAMAN metodu ile oluşturulan 18 bit/sn kod çizelgesi tablosu, 1200 bit/sn MQBC kodlayıcıda kullanılmıştır. Bitişik pencerelerin korelasyonu mantığından hareketle yeni bir artık sinyal hesaplama metodu sunulmuştur. Artık sinyal metodu, vektör nicemleme ve matris nicemleme metotları üzerinde denenmiştir. Kodlamadaki etkinliği artırmak için, 1600 bit/sn MELP kodlayıcı [18] uygulamasında vektör nicemleme tablosunu etkinleştirmek ve sesli/sessiz sinyal ayrımında frekans bant sayısını artırmak gibi teknikler kullanılmaktadır. 1200 bit/sn MELP kodlayıcı [94] uygulamasında ise bitişik pencerelerin aradeğerlemesi metodu kullanılmıştır. Burada uygulanan 1200 bit/sn MQBC kodlayıcı için metodların başka kodlayıcılar içinde kullanılabilir olması ve yeni gelişimlere açık olmasına dikkat edilmiştir. Ses sinyalinin doğru olarak sentezlenebilmesi için, analiz parametreleri içinde sesli/sessiz sinyal ayrımının önemi büyüktür. Sesli sesteki periyodik yapı ve sessiz ses sinyallerinin rasgele yapısı nedeniyle sesli ve sessiz bölümlerin iyi belirlenip ona göre uyartım sinyali uygulamak gerekir. Sesli/sessiz sinyal ayrımı için frekans bölgesinde ses sinyalini frekans bölgelerine ayırıp sınıflandırma iyi bir metottur. Fakat, iyi bir sınıflandırma için frekans bölümlerinin de arttırılması gerekir. Bu da, kodlama için gerekli bit ihtiyacını arttırır. Zaman düzlemi sınıflandırma içinde, doğrusal ayırıcı metodu ile sınıflandırma çok iyi sonuç vermektedir. Ayrıca zaman düzleminde sesli/sessiz geçiş bölgelerinin daha yavaş oluşu, çok az bit sayısı ile iyi bir

sınıflandırma yapılabilmesi sonucunu getirir. Bu tezde, sesli/sessiz sınıflandırma için, hem frekans düzleminde frekans bantlarının bölünmesi hem de zaman düzleminde doğrusal sınıflandırma metodu uygulanmıştır. MELP kodlayıcıda 5 olan frekans bantlarının MQBC kodlayıcıda 2'ye indirilmesi sonuç sesin biraz derecelenmesine neden olmuştur. Doğrusal sınıflandırıcı metodu ise, sesli/sessiz ayrımını çok iyi yapabilmesine rağmen uygun bir zaman düzlemi sentezleme metodu uygulanamadığı için gerçek performansı ölçülememiştir. Fakat her iki metodunda, MQBC kodlayıcı üzerinde uygulanmalarında sonuç sesin anlaşılabilirliği değişmemiş ve kalite kaybı çok az olmuştur. Objektif ve subjektif hesaplamalar sonucunda MQBC kodlayıcının MELP kodlayıcının yarısı kadar bit hızında çalışıyor olmasına rağmen, performansının yarıya düşmediği ve LPC-10 kodlayıcı'dan çok iyi ve MELP kodlayıcıya oldukça yakın bir ses kalitesine ulaştığı gözlenmiştir.

Gelecek Çalışma İçin Öneriler

Doğrusal öngörülü düşük bit hızlı kodlayıcılar için gelecek çalışma, kalite arttırımı ve bit hızı ile birlikte karmaşa azaltmak için yöntemler geliştirilmesi olabilir. Uygulanabilecek bazı metotlar aşağıdaki gibi gruplanabilir:

- **Daha düşük bit hızı için filtre katsayılarının yeniden tanımlanması**

Filtre katsayıları bir konuşma kodlayıcıda en iyi ifade edilmesi gereken ve çok fazla bit gerektiren parametrelerden biridir. Filtre katsayılarının düşük bit hızlarında ve az karmaşa ile ifade edilebilmesi için, ilgili kod çizelgesinin düşük bit hızları ile ifade edilebilmesi yanında arama ve depolama karmaşasının az olması gereklidir. Artık ÇAMAN ile dizayn edilen kod çizelgeleri düşük bit hızlarında iyi performans vermelerine rağmen arama ve depolama karmaşaları yüksektir.

- **Değişik parametreler için matris nicemleme metodu**

Bu çalışmada matris nicemleme metodu, filtre katsayıları için uygulanmıştır. Perde peryodu ve kazanç için de matris nicemleme metodu denenmiş, kazanç için düşük distorsiyon veren kod çizelgelerinin ilave bit kazandırmayacağı düşünülerek vazgeçilmiştir. Perde peryodu için, 7 bitlik kod çizelgesi distorsiyonunun büyük çıkması nedeniyle, doğru kodlanması çok önemli olan perde bilgisi için matris kod çizelgesi kullanılmamıştır. Uygun metotlar geliştirilerek filtre katsayıları dışındaki diğer parametrelere matris nicemleme metodunun uygulanması ile daha düşük bit hızlarına ulaşılabileceği düşünülmektedir.

- **Daha etkili bir perde peryodu hesaplama metodu**

Konuşma kodlama algoritmalarında önemli parametrelerden birisi de perde bilgisidir. Perde bilgisinin doğru belirlenmesi, sesli ses bilgisinin daha doğru olarak sentezlenmesini sağlamaktadır. Bu tezde, MELP kodlayıcı için kullanılan perde peryodu belirleme metodu kullanılmıştır ve perde belirleme metotları ayrıntılı olarak incelenmemiştir. Düşük bit hızlarında daha etkili bir metot ile kodlayıcı performansı artırılabilir.

- **Ses sınıflandırıcısının iyileştirilmesi**

Bu tezdeki bütün kodlayıcıların performansı sesli/sessiz ayırımının doğru yapılmasına bağlıdır. MQBC algoritması içinde kullanılan doğrusal sınıflandırıcı ile sesli/sessiz sinyal ayırım metodu, iyi performans vermesine rağmen geçiş parametrelerinin iyi kodlanabilmesi için bit sayısını arttırmak gerekir.

- **Etkili Sentezleme metotları**

Bir konuşma kodlayıcının iyi performans vermesi için analizden aktarılan parametrelerin doğruluğu kadar sentezleyici bölümde bu parametrelerin iyi sentezlenmesi de çok önemlidir. Bu yüzden, etkili sentezleme metotları ile sonuç kodlanmış ses çok daha iyi olabilir. Ayrıca, doğrusal sınıflandırıcı metodu ile sesli/sessiz ayrımı için zaman düzlemi sentezleme metodu için uygun uyartım sinyali kullanılırsa, sonuç sessin performansının iyi olacağı hesaplanmaktadır.

- **Değişken bit uygulamasının kullanılması**

MELP ve MQBC kodlayıcılar sabit bit hızlarında çalışmaktadırlar. Değişken bit sayılı pencere analizi ile, sessiz sesler için kodlanmasına gerek olmayan perde bilgisi için kullanılan bitler başka amaçlarla kullanılabilir.

- **Konuşma olmayan bölgelerin tespiti**

Ses sinyalinde konuşma olmayan bölgelerin doğru tespit edilmesi halinde, konuşmanın yapılmadığı bölümler için sentez bölümüne parametre aktarımına gerek kalmaz ve bit sayısından büyük oranda tasarruf sağlanabilir. Fakat konuşma olmayan bölgeler ve sessiz ses sinyallerinin birbirine karıştırılması durumunda ise sonuç ses sinyalinde bilgi kayıpları ve performansta kötüleşme olacaktır. İyi bir konuşma bölgesi tespit algoritması ile bit sayısından önemli oranda tasarruf edilebilir.

- **Aradeğerleme metodu**

Konuşma kodlama algoritmalarında bit sayısını azaltmanın bir yolu da aradeğerleme metodudur. Ses sinyalindeki bazı parametrelerin (LSF parametreleri, kazanç gibi) pencereden pencereye değişimlerinin yavaş olması, aradeğerleme metodu için büyük avantaj sağlamaktadır. Ses sinyalinin enerjisi kullanılarak yapılacak bir aradeğerleme metodu diğer aradeğerleme metotlarına göre avantaj sağlamaktadır. [105]



KAYNAKLAR

1. H. Dudley, The Vocoder, 1939, **Bell Labs. Record.**, V17, p. 122.
 2. B.M. Oliver, J.Pierce, and C.E. Shannon, 1948, The Philosophy of PCM, **Proc. IRE**, s. 1324-1331.
 3. G. Fant, 1960, Acoustic Theory of Speech Production, **Mouton and Co.**, Gravenhage, The Netherlands.
 4. J. Flanagan and Golden, 1966, Phase Vocoder, **Bell Tech. J.**, 45, s. 1493
 5. T.E. Tremain, April 1982, The Government Standard Linear Predictive Coding Algorithm. LPC-10, **Speech Technology**, s. 40-49.
 6. National Communications System, 1984, Analog to Digital Conversion of Voice by 2400 bit/second Linear Predictive Coding, **Federal Standard FS-1015 LPC**.
 7. Griffin, D.W, Lim, J.S., August 1988, Multiband Excitation Vocoder, **IEEE Transactions on Acoustics, Speech, and Signal Processing**, Vol.36, No.8, s.1223-1235.
 8. Andres Buzo, Augustine Gray, Robert M. Gray, October 1980, Speech Coding Based Upon Vector Quantization, **IEEE Trans. on Acoustics, Speech and Signal Processing**, Vol ASSP-28, No.5, s.562-574.
 9. B. Atal and M.R. Schroeder, May 1984, Stochastic Coding of Speech Signals At Very Low Bit Rates, **Proc. Int. Conf. Comm.**, s. 1610-1613.
 10. M.R. Schroeder and B. Atal, April 1985, Code-Excited Linear Prediction (CELP). High Quality Speech at Very Low Bit Rates, **Proc. ICASSP_85**, s. 937.
 11. P. Vary et al, April 1988, Speech Codec for the European Mobile Radio System , **Proc. ICASSP_88**, s.227-230.
 12. Gerson and M. Jasiuk, April 1990, Vector Sum Excited Linear Prediction (VSELP) Speech Coding at 8 kbits/s , **Proc. ICASSP_90**, s. 461-464, New Mexico.
 13. Gerson, M. Jasiuk, M. McLaughlin, and E. Winter, 1990, Combined Speech and Channel Coding at 11.2 kbits/s , **Signal Processing V. Theories and Applications**, Elsevier, s. 1339-1342.
 14. Y.Medan, E. Yair, and D. Chazan, January 1991, Super Resolution Pitch Determination of Speech Signals, **IEEE Transactions on Signal Processing**, Vol.39, No.1, s. 40-48.
-

15. J. Hardwick and J. Lim, May 1991, The Application of the IMBE Speech Coder to Mobile Communications, **Proc. ICASSP_91**, s. 249-252.
 16. Allen Gersho, Robert M. Gray, 1991, Vector Quantization and Signal Compression, **Kluwer Academic Publishers**.
 17. Furui, S., Sondhi, M.M, 1992, Advances in Speech Signal Processing, **Marcel Dekker Inc**.
 18. A.McCree & J.C. De Martin, September 1997, A 1.6 kbps MELP coder for wireless communications , **IEEE Speech Coding Workshop**, Pennsylvania.
 19. M. Oshikiri, M. Akamine, 1998, A 2.4 kbps Variable Bit Rate ADP-CELP Speech Coder, **ICASSP_98**
 20. J. Campbell, T.E. Tremain, and V. Welch, April 1990, The Proposed Federal Standard 1016 4800 bps Voice Coder. CELP, **Speech Technology**, s. 58-64.
 21. National Communications System, February 1991, Telecommunications. Analog to Digital Conversion of Radio Voice by 4800 bps Code Excited Linear Prediction (CELP), **Federal Standard 1016**.
 22. National Communications System, January 1992, Details to assist in implementation of Federal Standard 1016 CELP, **Technical Information Bulletin**.
 23. **CCITT Draft Recommendation G.728**, Coding of Speech at 16 kbit/s Using Low-Delay Code Excited Linear Prediction (LD-CELP), 1992.
 24. Stan McClellan, Jerry D. Gibson, March 1997, Variable-Rate CELP based on Subband Flatness, **IEEE transactions on Speech and Audio Processing**, Vol.5, No.2 , s.120-130.
 25. Lawrence R. Rabiner, Ronald W. Schafer, 1978, **Digital Processing of Speech Signals**, Prentice Hall
 26. Sadaoki Furui, 1989, **Digital Speech Processing, Synthesis & Recognition**, Marcel Dekker, Inc.
 27. Wai-Yip Chan, Smita Gupta, Allen Gersho, November, 1992, Enhanced Multi Stage Vector Quantization by Joint Codebook Design, **IEEE Transactions on Communications**, Vol.40, No.11, s.1693-1697.
 28. S.R. Quackenbush, T.P. Barnwell, and M.A. Clements, 1988, **Objective Measures for Speech Quality**, Prentice Hall, Englewood Cliffs.
-

29. Moore, 1989, **An Introduction to the Physiology of Hearing**, Academic Press.
 30. G. Fairbanks, 1958, Test of Phonemic Differentiation. The Rhyme Test, **J. Acoust. Soc. Am.**, Vol. 30, s. 596-600.
 31. W.D. Voiers, May 1977, Diagnostic Acceptability Measure for Speech Communications Systems, **Proc. ICASSP_77**, s. 204.
 32. R. Kubichek, April 1991, Standards and Technology Issues in Objective Voice Quality Assessment , **Digital Signal Processing. A Review Journal**, s. 38-44.
 33. S.H. Chen, 1990, High Quality 16 kb/s Speech Coding with ne way delay less than 2ms, **Proc. of ICASSP**, s. 453-456.
 34. N.S. Jayant, May 1974, Digital Coding of Speech Waveforms. PCM, DPCM, and DM Quantizers, **Proc. IEEE**, vol. 62, s.611-632.
 35. **CCITT Red Book**, Recommendation, G.721, volume 3
 36. **CCITT Yellow Book**, Recommendation, G.711, volume 3
 37. Hanson, H.M, Maragos, P., Potamianos, A., 1993, Finding Speech Formants and Modulations via Eenergy Seperations with Application to a Vocoder, **Proc. IEEE ICASSP93**, s. 716-719.
 38. Potamianos, A., Maragos, P., 1995, Speech Formant frequency and bandwidth tracking using multiband energy demodulation, **Proc. IEEE ICASSP_95**, s.784-787, Detroit.
 39. Preeti Rao, 1996, A Robust Method for the Estimation of Formant Frequency Modulation in Speech Signals, **ICASSP_96**, s.813-816.
 40. L. Welling, H. Ney, 1996, A Model for Efficient Formant Estimation, **ICASSP_96**, s.797-800.
 41. J. Schoentgen, S. Ciocea, 1997, Kinematic Formant to Area Mapping, **Speech Communication**, V.21, s.227-244.
 42. Chang-Sheng Yang, Hideki Kasuya, 1998, Automatic estimation of formant and voice source parameters using a subspace based algorithm, **ICASSP_98**
 43. Maragos, P., Quatieri, T. F., Kaiser, J. F, May, 1991, Speech nonlinearities, modulations and energy operators , **Proc. IEEE ICASSP_91**, s.421-424, Toronto, Canada.
-

44. Bhaskar Bhattacharya, Wilf LeBlanc, S. Mahmoud, Vladimir Cuperman, 1992, Tree Searched Multi-Stage Vector Quantization of LPC parameters for 4 kbps Speech Coding, **ICASSP_92**, s.105-108.
 45. W.P.LeBlanc, S.A. Mahmoud, V. Cuperman, 1993, Joint design of MSVQ Codebooks for LSP Quantization with Applications to 4 kbps Speech Coding, **Speech and Audio processing for Wireless and Network Applications, Kluwer Academics Publishers.**
 46. Tian Wang, Kun Tang, Chongxi Feng, 1996, A High Quality MBE-LPC-FE Speech Coder at 2.4 kbps and 1.2 kbps, **ICASSP_96**, s.208-211.
 47. Michele Jomrozik, John Gowdy, 1997, Modified Multiband Excitation Model at 2400 bps, **ICASSP_97**, s.1603-1606.
 48. Joseph P. Campbell, Thomas E. Tremain, Voiced/Unvoiced Classification of Speech with Applications to the U.S. Government LPC-10E Algorithm, **ICASSP_86**, s.473-476.
 49. Kondozi, A.M, 1994, **Digital Speech, Coding for Low Bit Rate Communication Systems**, John Wiley & Sons.
 50. J. Coi, J.Q.Liu, 1997, Robust Pitch Detection of Speech Signals using Steerable Filters, **IEEE Int. Conf. Acoust. Speech, Signal Processing**, s. 1427-1430.
 51. Jean Rouat, Yong Chun Liu, Daniel Morissette, 1997, A pitch Determination and Voiced/Unvoiced Decision Algorithm for Noisy Speech, **Speech Communication**, v.21, s.191-207.
 52. Man Mohan Soundhi, June 1968, New Method of Pitch Extraction, **IEEE Transactions Audio and Electroacoustics**, s. 262-266.
 53. Leah J. Siegel, alan C. Bessey, June 1982, Voiced/Unvoiced/Mixed Excitation Classification of Speech, **IEEE Trans. on Acoustics, Speech, and Signal Processing**, Vol. ASSP-30, No.3, s.451-460.
 54. S. Ghaemmaghami, M. Deriche, 1998, A New Approach to Modelling Excitation in Very Low-Rate Speech Coding, **ICASSP_98**
-

55. Yoseph Linde, Andres Buzo, Robert M. Gray, January 1980, An Algorithm for Vector Quantizer Design, **IEEE Trans. on Communications**, Vol. Com-28, No.1, s.84-95.
 56. Robert M. Gray, Andres Buzo, Augustine H. Gray, August 1980, Distortion Measures for Speech Processing, **IEEE Trans. on Acoustics, Speech, and Signal Processing**, Vol. ASSP-28, no.4, s.367-376.
 57. Biing-Hwang Juang, David Y. Wong, Augustine H. Gray, April 1982, Distortion Performance of Vector quantization for LPC Voice Coding, **IEEE Trans. on Acoustics, Speech, and Signal Processing**, Vol. ASSP-30, No.2, s.294-303.
 58. Hüseyin Abut, Robert M. Gray, Guillermo Rebolledo, June 1982, Vector Quantization of Speech and Speech-Like Waveforms, **IEEE Trans. on Acoustics, Speech and Signal Processing**, Vol. ASSP-30, No.3, s.423-435.
 59. Stuart P. Lloyd, March 1982, Least Squares Quantization in PCM, **IEEE Transactions on Information Theory**, Vol. IT-28, No.2, s.129-137.
 60. David Y. Wong, Biing-Hwang Juang, Augustine H. Gray, October 1982, An 800 bps Vector Quantization LPC Vocoder, **IEEE Trans. on Acoustics, Speech and Signal Processing**, Vol. ASSP-30, No.5, s.770-780.
 61. Biing-Hwang Juang, A.H. Gray, 1982, Multiple Stage Vector quantization for speech Coding, **ICASSP_82**, s.597-600.
 62. R.M. Gray, April 1984, Vector Quantization, **IEEE ASSP Magazine**, s. 4-29.
 63. Michael J. Sabin, Robert M. Gray, June 1984, Product Code Vector Quantizers for Waveform and Voice Coding, **IEEE Trans. on Acoustics, Speech and Signal Processing**, Vol. ASSP-32, No.3, s.474-488.
 64. Robert M. Gray, April, 1984, Vector Quantization, **IEEE ASSP Magazine**, s.4-29
 65. John Foster, Robert M. Gray, May 1985, Finite-State Vector Quantization for Waveform Coding, **IEEE Trans. on Inform Theory**, Vol. IT-31, No.3, s.348-359.
 66. John Makhoul, Salim Roucos, Herbert Gish, November 1985, Vector Quantization in Speech Coding, **Proceedings of the IEEE**, Vol. 73, No.11, s.1551-1588
-

67. P. Kabal, R.P. Ramachandran, December 1986, The Computation of Line Spectral Frequencies Using Chebyshev Polynomials, **IEEE Transactions on Acoustics, Speech, and Signal Processing**, Vol. ASSP-34, No.6, s.1419-1426.
68. A.Enis Çetin, Vijitha Weerackody, December 1988, Design Vector Quantizers Using Simulated Annealing, **IEEE Trans. on Circuits and Systems**, Vol.35, No.12, s.1550.
69. K.K. Paliwal, B.S. Atal, 1991, Efficient Vector Quantization of LPC Parameters at 24 bits/frame, **IEEE Int. Conf. on ASSP**, s.661-664.
70. Vladimir Cuperman, January 1993, Joint Bit Allocation and Dimensions Optimization for Vector Transform Quantization, **IEEE Trans. On Information Theory**, Vol.39, No.1, s.302-305.
71. William R. Gardner, Bhaskar D. Rao, September 1995, Theoretical Analysis of the High-Rate Vector quantization of LPC parameters, **IEEE Trans. On speech and Audio Processing**, Vol.3, No.5, s.367-381.
72. Amitava Dad, Ajit V.Rao, Allen Gersho, July 1996, Variable Dimension Vector Quantization, **IEEE Signal Processing Letters**, Vol.3, No.7.
73. W. Bastiaan Kleijn, Roar Hagen, August, 1996, On memoryless Quantization in Speech Coding, **IEEE Signal Processing Letters**, Vol.3, No.8, s.228-230.
74. S.Grassi , A. Dufaux, M. Ansorge, F. Pellandini, 1997, Efficient Algorithm to Compute LSP Parameters from 10th-order LPC Parameters , **IEEE Int. Conf. Acoust. Speech, Signal Processing**, s. 1707-1710.
75. John H.L. Hansen, Bryan L. Pellom, 1997, Text-Directed Speech Enhancement Employing Phone Class Parsing and Feature Map Constrained Vector Quantization, **Speech Communication**, V.21, s.169-189.
76. Kuldeep K. Paliwal, 1998, Spectral Subband Centroid Features for Speech Recognition, **ICASSP_98**
77. Jianping Pan, Thomas R. Fischer, March 1998, Vector Quantization of Speech Line Spectrum Pair Parameters and Reflection Coefficients, **IEEE Trans. On Speech and Audio Processing**, Vol.6, No.2, s.106-115.

78. L.M. da Silva, A. Alcaim, March 2000, Differential Coding of Speech LSF Parameters Using Hybrid Vector Quantization and Bidirectional Prediction, **IEEE Transactions on Speech and Audio Processing**, Vol.8, No.2, s.208-211.
 79. W.P. Leblanc, B. Bhattacharya, S.A. Mahmoud, October 1993, Efficient Search Design Procedures for Robust Multi-Stage VQ of LPC Parameters for 4 kbps Speech Coding , V1. No.4, **IEEE Transactions on Speech and Audio Processing**, s. 373-385.
 80. Nandkumar, S., Swaminathan, K., Bhaskar, 1998, Robust Speech Model based LSF Vector Quantization for Low Bit Rate Speech Coders, **IEEE Int. Conf. on ASSP_98**
 81. Tsao,C., Gray,R.M, June 1985, Matrix quantizer design for LPC speech using the generalized Lloyd Algorithm, **IEEE Transactions on ASSP**, Vol.Assp-33, No.3 , s.537-545.
 82. Yoshinao Shiraki, Masaki Honda, September 1988, LPC speech coding based on Variable length Segment Quantization, **IEEE trans. on ASSP**, Vol.36, No.9, s.1437-1444.
 83. Stefan Bruhn, 1995, Matrix Product Quantization for Very-Low-Rate Speech Coding, **ICASSP_95**
 84. Costas S. Xydeas, Charalampos Papanastasiou, March 1999, Split Matrix Quantization of LPC Parameters, **IEEE Trans. On Speech and Audio Processing**, Vol.7, No.2, s.113-125.
 85. C.H.Chen, 1998, **Signal Processing Handbook**, TK5102.2, SI6.
 86. A.V. McCree and T.P. Barnwell III, July 1995, A Mixed Excitation LPC Vocoder Model for Low Bit Rate Speech Coding, **IEEE Transactions on Speech and Audio Processing**, Vol.3, No.4, s. 242-250.
 87. L. Arslan, A. McCree, V. Viswanathan, 1995, New Methods for Adaptive Noise Suppression, **Proc. IEEE Int. Conf. Acust. Speech Signal Processing**, Vol.1. s. 812-815.
 88. A.Hasib, K. Hacıoglu, June 1996, Source-Combined Linear Predictive Analysis in Pulse based Speech codecs, **IEE Proc. Vis. Image Signal Process**, Vol.143, No.3, s.143-148.
-

89. A. McGree, K. Truong, E.B. George, T.P. Barnwell, V.Viswanathan, 1996, A 2.4 kbits/s MELP Coder Candidate for the New U.S. Federal Standard, **Proceedings of IEEE ICASSP**, s.200-203.
 90. FIPS Publication on MELP, June 1997, Analog to Digital Processing of Voice by 2400 bit/second Mixed Excitation Linear Prediction (MELP), **National Institute of Standards and Technology**.
 91. L.M. Supplee, R.P. Cohn, J.S. Collura, 1997, MELP. The New standard at 2400 bps, **IEEE Int. Conf. Acoust. Speech, Signal Processing**, s. 1591-1594.
 92. M.A. Kohler , 1997, US Department of Defense, A comparison of the New 2400 bps MELP federal standard with other standard coders, **IEEE Int. Conf. Acoust. Speech, Signal Processing**.
 93. Alan McCree, Juan Carlos De Martin, 1998, A 1.7 kbps MELP Coder with Improved Analysis and Quantization, **ICASSP_98**
 94. Tian Wang, Kazuhito Koishida, Vladimir Cuperman, Allen Gersho, 2000, A 1200 bps Speech Coder Based on MELP, **ICASSP_2000**
 95. Y. Shoham, 1997, Very Low Complexity Interpolative Speech Coding at 1.2 to 2.4 kbps, **IEEE Int. Conf. Acoust. Speech, Signal Processing**, s. 1599-1602.
 96. Khashayar Yaghmie, 1997, **Prototype Waveform Interpolation Based Low Bit Rate Speech Coding**, University of Surrey.
 97. Y. Hiwasaki, K. Mano, 1997, A new 2 kbps Speech Coder Based on Normalized Pitch Waveform , **IEEE Int. Conf. Acoust. Speech, Signal Processing**, s. 1583-1586.
 98. A.Papanastasiou, C.S. Xydeas, 1997, Efficient Mixed Excitation Models in LPC based prototype interpolation speech coders, **IEEE Int. Conf. Acoust. Speech, Signal Processing**, s. 1555-1558.
 99. Jan. S. Erkelens, Piet M.T. Broersen, March 1997, Bias Propagation in the Autocorrelation Method of Linear Prediction, **IEEE Transactions on Speech and Audio Processing**, Vol.5, No.2, s.116-119.
-

100. Ning Bi, Jingyong Qi, March 1997, Application of Speech Conversion to Alaryngeal Speech Enhancement, **IEEE Trans. On Speech and Audio Processing**, Vol.5, No.2.
 101. Doğan Aksan, 1990, **Her Yönüyle Dil**, Türk dil kurumu yayını.
 102. Ali N. Akansu, Richard A. Haddad, 1992, **Multiresolution signal decomposition, transforms-subbands-wavelets**, Academic Press Inc., TK5102.5, A414, C2.
 103. T. Islam, April 2000, **Interpolation of linear prediction coefficients for speech coding**, Msc Thesis, McGill University, Montreal, Canada.
 104. S. Özaydın, G. Bozdağı, December 1999, Tree-structured subband coding of speech with two different filter bank, **ELECO'99, International Conference on Electrical and Electronics Engineering**, s.380-384, Bursa.
 105. S. Özaydın, B. Baykal, Haziran 2000, LSF parametrelerinin çok aşamalı vektör nicemlenmesi için distorsiyon ve bellek azaltımı, **SIU2000, 8. Sinyal İşleme ve Uygulamaları Kurultayı**, Antalya.
 106. S. Ozaydin, B. Baykal, March 2001, Multi stage matrix quantization for very low bit rate speech coding, **SPAWC2001, Third IEEE Signal Processing Workshop on Signal Processing Advances in Wireless Communications**, Taoyuan, Taiwan.
 107. S. Ozaydin, B. Baykal, May 2001, Matrix quantization based speech coder at 1200 bps, **ISCAS2001, IEEE International Symposium on Circuits and Systems**, Sydney, Australia.
 108. S. Ozaydin, B. Baykal, May 2001, A 1200 bps speech coder with LSF matrix quantization, **ICASSP2001, IEEE International Conference on Acoustics Speech and Signal Processing**, Utah, USA.
 109. S. Özaydın, B. Baykal, Nisan 2001, 1200 bit/sn matris nicemleme tabanlı konuşma kodlayıcı, **SIU2001, 9. Sinyal İşleme ve Uygulamaları Kurultayı**, Kıbrıs
 110. S. Özaydın, B. Baykal, Nisan 2001, Düşük bit hızlı konuşma kodlama için çok aşamalı matris nicemleyici metodu, **SIU2001, 9. Sinyal İşleme ve Uygulamaları Kurultayı**, Kıbrıs
-

EK-1

ITU-T G- serisi ve H- serisi standard seri kümesi

In force ITU-T Recommendations

- A-series Recommendations - Organization of the work of ITU-T**
 - B-series Recommendations - Means of expression: definitions, symbols, classification**
 - C-series Recommendations - General telecommunication statistics**
 - D-series Recommendations - General tariff principles**
 - E-series Recommendations - Overall network operation, telephone service, service operation and human factors**
 - F-series Recommendations - Non-telephone telecommunication services**
 - G-series Recommendations - Transmission systems and media, digital systems and networks**
 - H-series Recommendations - Audiovisual and multimedia systems**
 - I-series Recommendations - Integrated services digital network (ISDN)**
 - J-series Recommendations - Transmission of television, sound programme and other multimedia signals**
 - K-series Recommendations - Protection against interference**
 - L-series Recommendations - Construction, installation and protection of cables and other elements of outside plant**
 - M-series Recommendations - TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile, and leased circuits**
 - N-series Recommendations - Maintenance: international sound programme and television transmission circuits**
 - O-series Recommendations - Specifications of measuring equipment**
 - P-series Recommendations - Telephone transmission quality, telephone installations, local line networks**
 - Q-series Recommendations - Switching and signalling**
 - R-series Recommendations - Telegraph transmission**
 - S-series Recommendations - Telegraph services terminal equipment**
 - T-series Recommendations - Terminals for telematic services**
 - U-series Recommendations - Telegraph switching**
 - V-series Recommendations - Data communication over the telephone network**
 - X-series Recommendations - Data networks and open system communications**
 - Y-series Recommendations - Global information infrastructure and Internet protocol aspects**
 - Z-series Recommendations - Languages and general software aspects for telecommunication systems**
 - Superseded ITU-T Recommendations**
 - List of withdrawn ITU-T Recommendations**
-

EK-2

Farklı vektör nicemleyiciler ve farklı bit sayıları ile oluşan kod çizelgeleri için 40 iterasyonda ağırlıklı distorsiyon hesapları sonuçları

loop	res-9bit dr (T=15767)	res-9bit dr (T=28400)	res-12bit dr (T=15767)	res-12bit dr (T = 65685)
1	0.011451	0.016902	0.009656	0.008547
2	0.003584	0.003827	0.002363	0.002377
3	0.003081	0.003092	0.002116	0.002024
4	0.002942	0.002864	0.001983	0.001875
5	0.002882	0.002773	0.001921	0.001829
40	0.002587	0.002454	0.001711	0.001617

loop	res-9bit (fullsearch)	%out(>4dB)	dr	T
40	SD0 = 2.043087	0.015494	0.002454	28400
98	SD1 = 2.082407	0.012747	0.002478	28400
27	SD2 = 2.149932	0.012677	0.002602	28400
162	SD3 = 2.180230	0.011338	0.002621	28400

loop	res - 15bit dr (T=15767)	res-15bit dr (T=28400)	res-15bit dr (T = 65685)	res-15bit dr (T=122695)
1	0.006139	0.005998	0.014068	0.006139
2	0.001644	0.001638	0.001657	0.001654
3	0.001384	0.001398	0.001414	0.001414
4	0.001306	0.001312	0.001332	0.001331
5	0.001267	0.001272	0.001285	0.001284
40	0.001126	0.001129	0.001132	0.001132

loop	res-15bit SD results SD	% out (> 2dB)	dr	T
25	SD0 = 1.426137	0.11951	0.001129	28400
31	SD1 = 1.458321	0.11722	0.001143	28400
45	SD3 = 1.533700	0.13394	0.001214	28400
60	SD0 = 1.414187	0.107184	0.001132	122695
52	SD3 = 1.540545	0.125914	0.001256	122695

loop	res-18bit dr (T=15767)	res-18bit dr (T=28400)	res-18bit dr (T = 65685)	res-18bit dr (T=122695)	res-24bit dr (T=15767)	res-24 bit dr (T = 65685)
1	0.005255	0.004984	0.005058	0.005072	0.003347	0.003061
2	0.001263	0.001198	0.001259	0.001266	0.000635	0.000651
3	0.001026	0.001000	0.001033	0.001033	0.000527	0.000529
4	0.000962	0.000934	0.000953	0.000955	0.000482	0.000484
5	0.000925	0.000896	0.000908	0.000912	0.000455	0.000546
40	0.000771	0.000754	0.000763	0.000772	0.000331	0.000353

loop	res-18bit SD results			
	SD	%out (>2dB)	dr	T
53	SD0 = 1.178302	0.047572	0.000754	28400
45	SD1 = 1.208246	0.043840	0.000769	28400
61	SD3 = 1.264538	0.042959	0.000817	28400
60	SD0 = 1.167642	0.042732	0.000764	122695
24	SD3 = 1.292821	0.043971	0.000877	122695
106	SD4 = 1.314955	0.041363	0.000895	122695

loop	full vector 9 bit	full vector 10 bit	full vector 11 bit	full vector 12 bit	full matrix (m=2) 12 bit	joint matrix (m=2) (4,4,4-12)
1	0.003908	0.003249	0.002449	0.001800	0.003095	0.005034
2	0.002693	0.002239	0.001726	0.001252	0.002203	0.004081
3	0.002467	0.002042	0.001575	0.001147	0.002025	0.003857
4	0.002368	0.001949	0.001510	0.001103	0.002004	0.003745
5	0.002307	0.001898	0.001473	0.001080	0.001992	0.003701
40	0.002012	0.001632	0.001254	0.000892	0.001753	0.003415

loop dr	(3,3,3)-9bit dr (T=15767)	(4,4,4)-12bit dr (T=15767)	(5,4,4)-13bit dr (T=15767)	(5,5,4)-14bit dr (T=15767)	(5,5,5) - 15bit dr (T=15767)
1	0.002134	0.011197	0.008426	0.008241	0.008225
2	0.004454	0.003048	0.002550	0.002371	0.002205
3	0.003699	0.002472	0.002083	0.001891	0.001743
4	0.003257	0.002309	0.001924	0.001734	0.001599
5	0.002877	0.002225	0.001840	0.001652	0.001524
40	0.002848	0.001885	0.001623	0.001432	0.001226

loop dr	(6,5,5)-16bit dr (T=15767)	(6,6,5)-17bit dr (T=15767)	(6,6,6)-18bit dr (T=15767)	(8,8,8)-24bit dr (T=15767)
1	0.005117	0.006221	0.006076	0.003474
2	0.001584	0.001683	0.001522	0.000766
3	0.001260	0.001274	0.001164	0.000553
4	0.001164	0.001154	0.001049	0.000484
5	0.001116	0.001096	0.000995	0.000442
40	0.000903	0.000825	0.000814	0.000325

EK - 3**MOS testi için kullanılan İngilizce Cümleler**

The birch canoe slide on the smooth planks
Glue the sheet to the dark blue background

Oak is strong and also gives shade
Cats and dogs each hate the other

Two blue fish warm in the tank
Her purse was full of useless trash

Sunday is the best part of the week
The doctor cured him with these pills

On the island the sea breeze is soft in mild
The play began as soon as we set down

A six comes up more often than a ten
Lush fern grow on the lusty rock

A list of names is carved around the box
The sheep were led home by a dog

The theft of pearl pen was kept the secret
Shake hands with this friendly chant

A king ruled the state in the early days
The ship was thorn apart on the sharp reef

MOS Testi için Kullanılan Türkçe Cümleler

1. Barış kararı toplantıdaki herkesi sevindirdi.
Petrol zammı beklenen fiyatın üzerinde idi.
2. Melon şapkalı turist çocukların ilgi odağı oldu.
Tüm balıklar dar fânus içine hapsedildi.
3. Çay cam demlikte daha güzel olmuş
Pas lekesi blûzun her yerinde görünüyor
4. Sarı beyaz priz daha yüksek voltaj veriyor.
Düşen vazo parça bölük hale geldi
5. Pazar haftanın en sevilen günüdür.
Sinema saat beşte başlayıp sekizde bitti.
6. Sahil kıyısı şehir merkezinin çok uzağında idi
Park'ta bütün köşelerde su şişesi var
7. El üstünde tutulurken êl sözüne kandı
Tanık olarak dinlenirken sanık oluverdi
8. Aşağı kadar ancak merdivenle inebildik.
Top patlar patlamaz çocuklar bağırdı
9. Serginin en göz alıcı parçası çinişi halı idi
Söylenenleri hiçe sayıp püfür püfür sigara içti.
10. Maçın tek gôlünü penaltıda attılar
Lüfer semt hâline hiç gelmiyormuş
11. Gri gömlekli öğrenci hala sınavdan çıkmadı
Toplantıda veliler müdüre şikayetlerini bildirdiler.
12. Dik kayalar ve derin çukurlar dağa tırmanışı güçleştirdi
Pembe kız çocuklarının favori rengidir.

EK - 4

2400, 2000, 1600, 1200, 2400 LPC10e kodlayıcılar için quiet.wav dosyasının kodlanmış sonuçları testi:

	E1	K1	E2	K2	E3	K3	TOP	Puanlama
1.wav								5. EXCELLENT
2.wav								4. GOOD
3.wav								3. FAIR
4.wav								2. POOR
5.wav								1. BAD
								Not : puanlar tamsayı olmalı

2400, 1200 ve LPC10e kodlayıcılar için quiet.wav, office.wav ve car.wav dosyalarının kodlanmış sonuçları testi:

	E1	K1	E2	K2	E3	K3	TOP	Office ve quiet
Quiet								Zeminli sesler (Tam puan)
Office								5. EXCELLENT
Car								4. GOOD
Q1								3. FAIR
Q2								2. POOR
Q3								1. BAD
Q4								Araba gürültüsü
Q5								Sesler : (kesirli puanlama)
Q6								5-4. INAUDIBLE
Q7								4-3. AUDIBLE BUT NOT ANNOYING
Q8								3-2. SLIGHTLY ANNOYING
Q9								2-1. ANNOYING
								1-0. VERY ANNOYING

2400, 1200, 2400 LPC10e kodlayıcılar için türkce.wav dosyasının kodlanmış sonuçları testi:

	E1	K1	E2	K2	E3	K3	E4	K4	TOP
toplam.mlp									
5. EXCELLENT 4. GOOD 3. FAIR 2. POOR 1. BAD									

2400, 1200, 2400 LPC10e kodlayıcılar için Türkçe drt dosyasının kodlanmış sonuçları testi:

Voice.mlp		Nazal.mlp		Sustane.mlp		Sibile.mlp		Grave.mlp	Compact.mlp			
☐ van	☐ fan	☐ mal	☐ bal	☐ veri	☐ beri	☐ zam	☐ tam	☐ pas	☐ tas	☐ yer	☐ ver	
☐ ver	☐ fer	☐ mil	☐ bil	☐ var	☐ bar	☐ çar	☐ kar	☐ bul	☐ dul	☐ yan	☐ van	
☐ bir	☐ pir	☐ nil	☐ dil	☐ şark	☐ çark	☐ çil	☐ kil	☐ bar	☐ dar	☐ kat	☐ tat	
☐ bak	☐ pak	☐ nem	☐ dem	☐ şile	☐ çile	☐ çök	☐ kök	☐ beri	☐ deri	☐ kel	☐ tel	
☐ bul	☐ pul	☐ göl	☐ döl	☐ fas	☐ pas	☐ çek	☐ kek	☐ fiş	☐ diş	☐ fas	☐ has	
☐ cam	☐ çam	☐ gam	☐ dam	☐ fes	☐ pes	☐ jel	☐ gel	☐ meşe	☐ neşe	☐ hakim	☐ fakir	
☐ cin	☐ çin	☐ geri	☐ deri	☐ fil	☐ pil	☐ sur	☐ dur	☐ fide	☐ side	☐ çat	☐ pat	
☐ zar	☐ sar	☐ nar	☐ dar	☐ tek	☐ dek	☐ sahil	☐ dahil	☐ say	☐ fay	☐ çile	☐ pile	
☐ vil	☐ sil	☐ gem	☐ dem	☐ tam	☐ dam	☐ saç	☐ taç	☐ mil	☐ nil	☐ cadı	☐ dadı	
☐ zor	☐ sor	☐ melek	☐ belek	☐ tiz	☐ diz	☐ semiz	☐ temiz	☐ but	☐ dut	☐ şer	☐ ser	
☐ beş	☐ peş	☐ nur	☐ dur	☐ dümen	☐ tümen	☐ ser	☐ ter	☐ mal	☐ nal	☐ şal	☐ sal	
☐ zevk	☐ sevk	☐ nem	☐ dem	☐ fare	☐ pare	☐ sim	☐ tim	☐ mey	☐ ney	☐ şap	☐ sap	
☐ gaz	☐ kaz	☐ muz	☐ buz	☐ şal	☐ çal	☐ söz	☐ göz	☐ fer	☐ ser	☐ gol	☐ bol	
☐ gül	☐ kül	☐ mey	☐ bey	☐ şut	☐ but	☐ cam	☐ gam	☐ pek	☐ dek	☐ göl	☐ böl	
☐ göz	☐ köz	☐ gün	☐ dün	☐ den	☐ ten	☐ salı	☐ halı	☐ sal	☐ fal	☐ çat	☐ pat	
☐ gaf	☐ kaf	☐ gol	☐ dol	☐ dur	☐ tur	☐ ser	☐ her	☐ sil	☐ fil	☐ çipa	☐ tıpa	

EK-5

MQBC KODLAYICI/ÇÖZÜCÜ YAZILIMI GENEL YAPISI

MQBC kodlayıcı ve çözücü yazılım yapısının genel bir tanımı yapılmış ve kullanılan global değişkenler/sabitler ve ana programdan çağrılan alt programların tanımları verilmiştir.

VERİ TANIMLARI

Global Sabitler : Aşağıdaki değişkenler ve sabitler MQBC yazılımını başlatmak için kullanılırlar.

msmq_cb : Artık Çok aşamalı matris (artık ÇAMAN) kod çizelgesi
sabitlerini içeren dizin
msmq_mq_par: Artık ÇAMAN kod çizelgesi aşamalarını tutan pointer
mbc_param : Perde periyodu, sesli/sessiz ayrımı v.b. ses
parametreleri için pointer
voiced_enc : Sesli/sessiz ayrımı için kullanılan pointer
in_name : Giriş ses veri dosyası
out_name : Çıkış ses veri dosyası

PROGRAM YAPISI

Bu bölümde, program yapısı alt program isimleri ile verilmiştir.

MQBC CODER {Main}

```

    Read_input_file;    // Giriş veri dosyasını okur
    Open_output_file;   // Çıkış verisi için dosya açar

if (analysis)
    mbc_ana_init;
    msmq_ana_init;
    v_uv_init;

if (synthesis)
    mbc_syn_init;
    msmq_syn_init;

repeat
    if (analysis)
    {
        read_input_speech_framel;
        mbc_analys1;
        speech_dc_rmv;
        speech_lpf;
        find_pitch_P12;
        lpc10_encode;
        lpcini
        preemp
        lpfilt
        hpfl00
    }

```

```

        invfilt
        onset
        placev
        voicin
        vparms
        tbdm
        residual_lpc_calc;
        find_pitch_P3;
        find_gain;
        quantize_pitch1;
        quantize_gain1;
        quantize_voice1;
        residual_lsf_calculation1;

        read_input_speech_frame2;
        mbc_analys2;
        speech_dc_rmv;
        speech_lpf;
        find_pitch_P12;
        lpc10_encode;
        lpcini
        preemp
        lpfilt
        hpf100
        invfilt
        onset
        placev
        voicin
        vparms
        tbdm
        residual_lpc_calc;
        find_pitch_P3;
        find_gain1;
        quantize_pitch2;
        quantize_gain2;
        quantize_voice2;
        residual_lsf_calculation2;

        msmq_encoder;

        mbc_chn_write;
        pitch_voicing;
        coded_parameters;
    }

    if (synthesis)
    {
        mbc_chn_read;
        decoded_parameters;
        dec_pitch_voicing;
        lsf_hat_calc;
        gain_decode;

        mbc_syn1;
        pulse_coeff;
    }

```

```

        noise_coeff;
        add_pulse_noise;
        interp_syn_parameters;
        app_ase_pilter;
        lpc_synthesis;
        app_pulse_disper;
    write_frame1;
    mbc_syn2;
        pulse_coeff;
        noise_coeff;
        add_pulse_noise;
        interp_syn_parameters;
        app_ase_pilter;
        lpc_synthesis;
        app_pulse_disper;
    write_frame2;
}
until end of file

```

ALT PROGRAMLAR

mbc_ana_init : MQBC kodlayıcı bölümünde, giriş ses datası, perde ve lpc bilgisi için başlangıç değişkenlerini açar.

msmq_ana_init : Çok aşamalı matris kod çizelgesi için, kod çizelgesi analiz bölümünde başlangıç değerlerini belirler.

v_uv_init : Sesli/sessiz ayrımı alt programı için pointer başlatımı

mbc_syn_init : MQBC sentez programı için başlangıç değerleri belirler.

msmq_syn_init : Çok aşamalı matris kod çizelgesi için, kod çizelgesi sentez bölümünde başlangıç değerlerini belirler.

read_input_speech_frame1 : 25 msn ses penceresi içindeki (200 örnek) giriş ses datalarını okur.

mbc_analys : MQBC kodlayıcı analiz bölümündeki işlemlerle ilgili alt programların toplandığı program yapısı

speech_dc_rmv : Ses sinyalinin DC sinyali kaldırır.

speech_lpf : Ses sinyali LFP'den geçirilir.

find_pitch_P12 : İlk perde peryodu P1 kullanılarak P2 kesirli perde peryodu bulunur.

lpc10_encode : Sesli/sessiz sinyal ayrımı için doğrusal sınıflandırıcı algoritmasını modelleyen alt programların toplandığı alt program. V1,V2

sesli/sessiz bitlerini verir.

residual_lpc_calc : Kesirli son perde peryodu P3 için artık sinyali hesaplar.

find_pitch_P3 : P3 perde peryodunu hesaplar.

find_gain : Kazanç değerini bulan alt program.

residual_lsf_calculation : Artık LSF vektörünü hesaplar.

quantize_pitch : Logaritmik perde değerini nicemler.

quantize_gain : Logaritmik kazanç değerini nicemler.

quantize_voice : karma (mixed) durumu kodlanır.

msmq_encoder : Artık çok aşamalı matris kod çizelgesi ile iki pencerenin artık LSF vektörleri kodlanır.

mbc_chn_write : Analiz bölümünde iki pencereden çıkarılan nicemlenmiş parametreler kodlanır.

pitch_voicing : Perde ve sesli/sessiz kararları birlikte kodlanır.

coded_parameters : Nicemlenmiş parametreler ikilik düzende bitlere dönüştürülür.

mbc_chn_read : İkilik düzendeki parametreler çözülür ve parametre indeks değerleri bulunur.

decoded_parameters : Her iki pencere parametreleri için, indeks değerleri kullanılarak, parametreler nicemlenmiş değerlere çevrilirler.

dec_pitch_voicing : Her iki pencere için Perde ve sesli/sessiz kararları ayrılıp çözülür.

lsf_hat_calc : Her iki pencere için yaklaşık LSF vektörler çözülür.

gain_decode : Her iki pencere için kazanç değerleri çözülür.

mbc_syn değerleri : Her bir pencere için, nicemlenmiş parametre kullanılarak sentezleme yapılır.

pulse_coeff : Vurum katsayıları üretir.

noise_coeff : Gürültü katsayıları üretir.

add_pulse_noise : Sesli/sessiz durumuna göre vurum ve gürültü katsayılarını toplayarak uyartım sinyali oluşturur.

interp_syn_parameters : Sentez parametrelerini bir önceki analiz penceresi ve o anki değerler arasında aradeğerleme ile yeniden bulur.

app_ase_pilter : formant parametrelerini güçlendirmek için ASE filtre uygular.

lpc_synthesis : Nicemlenmiş LSF katsayılarından bulunan LPC filtre parametreleri ile LPC sentez filtresini oluşturur.

app_pulse_disper : Yaklaşık ses sinyaline pulse dispersion filtre uygular.

write_frame : Sentez bölümünde oluşturulan yaklaşık ses sinyalini çıkış dosyasına kaydeder.

PERDE ALGORİTMASI

find_pitch_P12 : Tamsayı perde peryodu P1 bulur.

bpvc_ana : Ses sinyalini frekans bantlarına bölerek ses kuvvetlerini bulur ve P2 hesaplamasında kullanır

residual_lpc_calc : Ses sinyalinden çıkarılan LPC katsayılarını kullanarak, artık sinyali bulur ve P3 hesaplanmasında kullanır.

find_pitch_P3 : Artık sinyal kullanıp P3 perde değerini hesaplar.

quantize_pitch : son perde değerini logaritmik değere çevirip, [40-160] arasında bir değere nicemler.

SESLİ/SESSİZ ALGORTIMASI

lpc10_encode : Sesli/sessiz ses sinyali için alt program.

lpcini : Sesli/sessiz ses sinyali için giriş değişkenlerini ve sabitlerini başlatır.

preemp : giriş sinyalini, 0.9375 katsayılı tek sıfır bir filtre ile filtreler.

lpfilt : 31 Point Equiripple FIR Low-Pass Filter

hpfl00 : 100 Hz HPF

invfilt : 2. derece inverse filtre ile ses sinyal örnekleri 4:1 azaltılır.

onset : Ses sinyali penceresi için onset kontrolü

placev : Onset gözönüne alınarak voicing pencere yerleşimi

voicin : Pencerelemiş sesteki seslilik parametrelerinin iki yarı pencerede ölçümü

vparms : Sesli/sessiz ayrımı için parametreleri hesaplar

tbdm : yüksek yoğunluklu AMDF fonksiyonunu hesaplar.