ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DOKTORA TEZİ

DÜŞÜK VERİ HIZLARINDA ÇALIŞAN KONUŞMA KODLAYICILARINA GÜRBÜZ BİLGİ SAKLAMA VE DAMGALAMA

Ahmet Utku YARGIÇOĞLU

ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

ANKARA
2010

TEZ ONAYI

Ahmet Utku YARGIÇOĞLU tarafından hazırlanan **"Düşük Veri Hızlarında Çalışan Konuşma Kodlayıcılarına Gürbüz Bilgi Saklama ve Damgalama"** adlı tez çalışması 08/10/2010 tarihinde aşağıdaki jüri tarafından oy birliği ile Ankara Üniversitesi Fen Bilimleri Enstitüsü Elektronik Mühendisliği Anabilim Dalı'nda **DOKTORA TEZİ** olarak kabul edilmiştir.

Danışman : Doç. Dr. H. Gökhan İLK

Jüri Üyeleri:

Başkan :Doç. Dr. İsmail AVCIBAŞ
Başkent Üniversitesi
Elektrik-Elektronik Mühendisliği A.B.D

Üye :Doç. Dr. H. Gökhan İLK
Ankara Üniversitesi
Elektronik Mühendisliği A.B.D

Üye :Doç. Dr. Ziya TELATAR
Ankara Üniversitesi
Elektronik Mühendisliği A.B.D

Üye :Doç. Dr. A. Aydın ALATAN
Orta Doğu Teknik Üniversitesi
Elektrik-Elektronik Mühendisliği A.B.D

Üye :Yrd. Doç. Dr. Asım Egemen YILMAZ
Ankara Üniversitesi
Elektronik Mühendisliği A.B.D

Yukarıdaki sonucu onaylarım

Prof.Dr.Orhan ATAKOL
Enstitü Müdürü

ÖZET

Doktora Tezi

DÜŞÜK VERİ HIZLARINDA ÇALIŞAN KONUŞMA KODLAYICILARINA GÜRBÜZ BİLGİ SAKLAMA VE DAMGALAMA

Ahmet Utku YARGIÇOĞLU

Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Elektronik Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. H. Gökhan İLK

Bu tez kapsamında düşük hızlarda çalışan konuşma kodlayıcılarıyla günümüzün iki popüler konusu damgalama ve steganografi bir araya getirilmiştir. İlk aşamada MELP konuşma kodlayıcısının çok seviyeli vektör nicemleyicisine (MSVQ) alışılageldik nicemleme indeks modülasyonu (QIM) uyarlanmış, böylelikle özgün MELP-MSVQ-QIM veri gömme yöntemi geliştirilmiştir. QIM'in uygulandığı indeksler farklılaştırılarak MELP-MSVQ-QIM yöntemine ait çok sayıda varyasyon elde edilmiş, söz konusu varyasyonlar nesnel kalite ölçüm metrikleri ve steganaliz açılarından sınanarak içlerinden damgalama ve steganografiye en müsait olanları tespit edilmiştir. Bir sonraki aşamada, MELP-MSVQ-QIM varyasyonlarının sınanmasında faydalanılan kaotik öznitelikler steganalizi veri işlemenin bambaşka bir alanına aktarılmış, bu sayede özgün konuşma kodlayıcı tanıma uygulaması tasarlanmıştır. En basit anlatımla kodlayıcı tanıma; bilinmeyen bit dizilerinin analiz edilerek üreticileri olan kodlayıcıların bulunması olarak tarif edilebilir. Üçüncü aşamada kodlanmış konuşma sinyaline veri gömen yöntemleri yakalayabilecek Markov zincir modeli tabanlı bir steganaliz yöntemi önerilmiş, GSM 6.10 üzerinde gerçekleştirilen testler nihayetinde önerilen yöntemin en az kaotik öznitelikler steganalizi kadar verimli olduğu görülmüştür. Son olarak önerilen bu yeni steganaliz yöntemi karşısında görünmez olabilecek veri manipülasyon çözümleri araştırılmış, istatistiksel modellemeye dayalı yeni bir matris gömme yöntemi önerilmiştir.

Ekim 2010, 184 sayfa

Anahtar Kelimeler: MELP, QIM, MSVQ, kalite ölçüm metrikleri, steganaliz, kodlayıcı tanıma, GSM 6.10, Markov zincir modeli, matris gömme

ABSTRACT

Ph.D. Thesis

ROBUST DATA HIDING AND WATERMARKING IN LOW BIT-RATE SPEECH CODERS

Ahmet Utku YARGIÇOĞLU

Ankara University
Graduate School of Natural and Applied Sciences
Department of Electronic Engineering

Supervisor: Asc. Prof. Dr. H. Gokhan ILK

In this thesis, today's two popular subjects, watermarking and steganography are implemented on low rate speech coders. At the first stage, conventional quantization index modulation (QIM) is applied to the multistage vector quantizer (MSVQ) of MELP, so that a novel data embedding method, which is called as MELP-MSVQ-QIM, is developed. By altering the index bits on which QIM is operated, many variations of MELP-MSVQ-QIM are generated. In order to determine variations, which are the most appreciate for watermarking and steganographical usages, some performance tests were performed to find out objective quality metric and steganalyzer results of the variations. At the second stage, the chaotic feature steganalysis is adapted to a quite different signal processing application; the novel speech coder identifier is designed. In simplest, speech coder identifier is a new generation data analysis method recognizes the coders being used in unknown bitstreams. In the third stage, a new steganalysis method, Markov chain model based coded speech steganalysis method is designed against coded speech data hiding methods. After the tests executed on GSM 6.10 speech coder, the new steganalysis method is found to be as effective as the chaotic feature steganalysis. Finally, a counter data manipulating method, statistical model based matrix embedding is proposed against Markov chain model based coded speech steganalysis.

October 2010, 184 pages

Key Words: MELP, QIM, MSVQ, quality measurement metrics, steganalysis, coder identification, GSM 6.10, Markov chain model, matrix embedding

TEŞEKKÜR

Doktora eğitimimin boyunca engin bilgi ve deneyimini şahsımdan esirgemeyen, değerli önerileriyle bana yol gösteren danışman hocam Sayın Doç. Dr. H. Gökhan İLK'e (Ankara Üniversitesi) minnettar olduğumu belirtir ve teşekkür ederim. Tez çalışmalarımız boyunca bizi destekleyen, tezimizin şekillenmesinde aktif rol oynayan, hocalarım Sayın Doç. Dr. Ziya TELATAR (Ankara Üniversitesi) ve Sayın Doç. Dr. A. Aydın ALATAN'a (Orta Doğu Teknik Üniversitesi) teşekkürlerimi sunarım. Çalışmalarımıza ilham veren ve ufkumuzu açan, bu suretle tez çalışmalarımıza yardımcı olan Sayın Doç. Dr. İsmail AVCIBAŞ'a (Başkent Üniversitesi), Sayın Yrd. Doç. Dr. Asım Egemen YILMAZ'a (Ankara Üniversitesi) ve kıymetli meslektaşım Emrah YÜRÜKLÜ'ye teşekkürü borç bilirim.

Kurulduğu günden itibaren Türk Savunma Sanayi'nin bayraktarlığını yapan, Türkiye Cumhuriyeti'nin gelişmesi ve ilerlemesini amaç edinen, 1997 yılından beri bir parçası olmaktan gurur duyduğum ASELSAN AŞ'ye teşekkürlerimi sunarım. Sadece eğitim hayatım müddetince değil, iyi ve kötü günde her daim benim yanımda olan, eşsiz destek, fedakârlık ve hoşgörüden ötürü eşime, anneme, babama ve kardeşime teşekkür ederim.

Ahmet Utku YARGIÇOĞLU
Ankara, Ekim 2010

İÇİNDEKİLER

KISALTMALAR DİZİNİ

AbS	Sentez ile analiz (Analysis by synthesis)
ADPCM	Uyarlanabilir ayrımsal darbe kodu modülasyonu (Adaptive differential pulse code modulation)
AMR	Uyarlanabilir çok hızlı (Adaptive multi-rate)
BER	Bit hata oranı (Bit error rate)
CBR	Sabit bit hızı (Constant bit rate)
CELP	Kod uyarlamalı doğrusal öngörü (Code excited linear prediction)
DD	Doğrudan değiştirme
DoD	Savunma bölümü (Department of defence)
DPCM	Ayrımsal darbe kodu modülasyonu (Differential pulse code modulation)
EFR	Geliştirilmiş tam hızlı (Enhanced full rate)
EKOK	En Küçük Ortak Kat
EÖB	En Önemsiz Bit
ETSI	Avrupa Telekomünikasyon Standartları Enstitüsü (The European Telecommunications Standards Institute)
FEC	İleri hata kodlama (Forward error correction)
FNF	Hatalı komşu oranı (False neighbor fraction)

FNN	Hatalı en yakın komşular (False nearest neighbors)
FP	Hatalı pozitif (False positive)
FR	Tam hızlı (Full rate)
GSM	Mobil haberleşme için global sistem (Global system for mobile communications)
IS	Ters sinüs (Inverse sine)
ITU-T	Uluslararası Telekomünikasyon Birliği (International Telecommunication Union)
LAR	Logaritmik alan oranı (Log area ratio)
LMS	En küçük ortalama kare (Least mean square)
LPC	Doğrusal öngörü katsayıları (Linear prediction coefficients)
LSB	En önemsiz bit (Least significant bit)
LSE	En küçük kare hatası (Least square error)
LSF	Spektral hat frekansları (Line spectral frequencies)
LSP	Spektral hat çiftleri (Line spectral pairs)
LTP	Uzun süreli (vadeli) öngörü Long term prediction
MAC	Mesaj doğrulama kodu (Message authentication code)
MCLT	Kiplenmiş kompleks örtülmüş dönüşüm (Modulated complex lapped transform)

MELP	Karma uyarmalı doğrusal öngörü (Mixed excitation linear prediction)
MPE	Çoklu darbe uyarmalı (Multi-pulse excited)
MSVQ	Çok seviyeli vektör nicemleme (Multi stage vector quantization)
oyf	Olasılık yoğunluk fonksiyonu
PARCOR	Kısmi korelasyon (Partial correlation)
PCM	Dalga kodu modülasyonu (Pulse-code Modulation)
PESQ	Konuşma kalitesinin algısal değerlendirilmesi (Perceptual evaluation of speech quality)
ROC	Karar vericinin etkinliği (Receiver operating characteristics)
RPE	Düzgün darbe uyarmalı (Regular pulse excitation)
SBC	Alt-bant kodlaması (Sub-band coding)
SNR	Sinyal-gürültü oranı (Signal-to-noise ratio)
SVM	Destek vektör makinesi (Support vector machine)
QIM	Nicemleme indeks modülasyonu (Qunatization index modulation)
TP	Doğru pozitif (True positive)
VBR	Değişken bit hızı (Variable bit rate)
VQ	Vektör nicemleme (Vector quantization)

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

1. GİRİŞ

Günümüzde haberleşme sistemlerinin çok büyük bir kısmı, sağladığı avantajlar nedeniyle sayısallaşmıştır. Gelişen teknolojiyle beraber kanal kapasiteleri artmış, video, görüntü ve müzik gibi sinyallerin toplam iletişimdeki payları yükselmiştir. Lakin konuşma sinyalinin taşınmasıyla gerçekleştirilen haberleşme önemini yitirmemiştir. İnsan ses üretim modeli konusundaki bilimsel birikimin ve sayısal sinyal işlemedeki işlem gücünün çoğalmasıyla konuşma sinyalinin daha kaliteli ve yüksek oranlarda sıkıştırılabilmesi mümkün olmuştur. Halen çeşitli iletişim ortamlarındaki pratik darboğazlara bağlı olarak pek çok senaryoda düşük hızlarda çalışan konuşma kodlayıcılarına ihtiyaç sürmektedir.

Bilgi saklama ve gizli haberleşme yöntemleri çok eski çağlardan beri insanoğlunun ilgisini çekmektedir. Bilginin gizlenmesi ve yalnızca istenilen kişilerce erişilebilmesinin sağlanması steganografi bilimi olarak adlandırılır ve tarihte kafa derisine yapılan dövmelerden, mor ötesi ışıkta belirginleşen mürekkeplere kadar pek çok alternatif kullanımı belgelenmiştir. Diğer taraftan internet, sayısal haberleşme, bilgi paylaşımı ve çoklu ortam ürünlerinin kullanımının yaygınlaşması, steganografi faaliyetlerini sayısal ortama taşımıştır.

Çoğu zaman steganografiyle kavramıyla karıştırılan damgalama, bir belgenin içine, belge içeriğini örselemeden başka bir belge yerleştirmek olarak tanımlanmaktadır. Steganografiden farkı, yapılan yerleştirme işleminin çoğunlukla gizli tutulmaması, yerleştirilen belgenin yerleştirildiği belgeyle bir şekilde alakalı olmasıdır. Diğer yandan bazı kaynaklara göre steganografi, damgalamanın yalnızca bir alt kümesi olarak da nitelendirilmektedir. Damgalama, tıpkı steganografi gibi uzun yıllardır kullanılmaktadır; banknotlarda kullanılan filigranlardan kâğıtların üzerindeki logolara kadar değişik uygulamaları bulunmaktadır. Sayısallaşmanın yaygınlaşmasıyla beraber, damgalamanın sayısal ortam kullanımlarına gereksinim duyulmaya başlanmıştır.

Konuşma sinyali de sayısal haberleşmenin bir parçası olmasından dolayı zaman içerisinde damgalama – steganografi denemelerine maruz kalmış, konuşma sinyaline veri saklayan çeşitli yöntemler ortaya çıkmıştır. Ancak geliştirilen yöntemlerin büyük kısmı müzik sinyaline veri gömen yöntemlerin türevleridir; bahse konu yöntemlerin gizli veri içeren çıktıları düşük hızlı kodlayıcılara sokuldukları zaman maalesef gömülen veri işe yaramaz hale gelmektedir.

Eklenen gizli verinin düşük hızda çalışan bir kodlayıcı tarafından bozulmasını önlemenin en kestirme yolu, gömme işlemini bizzat kodlayıcının bir parçası haline getirmektir. Literatürde bu fikri MELP, CELP, GSM 6.10 kodlayıcıları üzerinde hayata geçiren türlü veri gizleme uygulamaları mevcuttur, nitekim tez çalışmaları da ilk olarak MELP üzerinde çalışan gizli veri gömme yönteminin incelenmesiyle başlamıştır.

Tez çalışmalarına MELP çok seviyeli vektör nicemleyicisinde (MSVQ) nicemleme indeks modülasyonu (QIM) uygulayarak gizli veri saklayan çalışan özgün MELP-MSVQ-QIM yöntemi geliştirilmesiyle devam edilmiş, geliştirilen yeni yöntem kalite metrikleri ve steganaliz açısından sınanmıştır. Bir sonraki aşamada, MELP-MSVQ-QIM yönteminin steganalizi konuşma işlemenin bambaşka bir alanına uyarlanmış, konuşma kodlayıcı tanıma türetilmiştir. Ardından, steganografi alanına odaklanılmış, kodlanmış konuşma sinyalleri için yeni bir steganaliz yöntemi, Markov durum makineleri tabanlı steganaliz yöntemi oluşturulmuştur. Hazırlanan steganaliz yöntemi, literatürde yer alan GSM 6.10 veri gömme yöntemi üzerinde denenmiştir. Son olarak, kodlanmış konuşma sinyalleri içinde veri gizleyen yöntemlere katkı sağlayabilecek, Markov steganaliz yöntemine karşı daha dayanıklı yeni özgün bir matris kodlaması türü, istatistiksel olarak iyileştirilmiş matris kodlama meydana getirilmiştir.

Bu tezi oluşturan bölümlerin bir özeti verilirse, Bölüm 2'de, tezin sonraki bölümlerinin anlaşılması ve takip edilebilmesi için gerekli olan kuramsal altyapı sunulmaktadır. Kuramsal altyapıda ilk önce konuşma sinyali tanımlanmakta, ardından konuşma sinyalinin üretim modeli verilmektedir. Daha sonra konuşma kodlayıcıları ve konuşma kodlama algoritmaları işlenmektedir. Bir sonraki aşamada veri gömme detaylı olarak incelenmekte, literatürdeki bu tezle en ilgili olan veri gömme yöntemleri tanıtılmaktadır.

Son olarak steganaliz konusu ele alınmakta, hemen öncesindeyse kaotik öznitelik steganalizinin anlaşılabilmesi için gerekli asgari kaos teorisi anlatımına yer verilmektedir.

Bölüm 3'te, MELP kodlayıcısı içerisinde gizli veri gömme uygulayan yöntemler konu alınmaktadır. Bölümün en başında Chang ve Yu'nun (2002) MSVQ'da kıpırtı benzeri veri gömme yöntemi yeniden değerlendirilmekte, bahse konu yöntemin aşırı derecede bozulmaya neden olduğu gösterilmektedir. Bu duruma birim zamanda gömülen gizli veri miktarının fazlalığın yol açtığı ileri sürülmekte, MELP MSVQ'sunda gömülen bit sayısının azaltılmasının steganografik performansı iyileştireceği öngörülmektedir. Bu bağlamda geliştirilen yeni özgün MELP-MSVQ-QIM yöntemi tanıtılmakta, çalışma esasları detaylı olarak anlatılmaktadır. Daha sonra MELP-MSVQ-QIM yönteminden pek çok varyasyon türetilmekte, türetilen varyasyonların çeşitli kalite metrikleri cinsinden performans sonuçları sunulmaktadır. Steganografik kullanıma uygun olan varyasyonlardan bazıları Kocal vd. (2008) tarafından önerilen kaotik öznitelikler steganalizine alınmakta, türetilen varyasyonların steganografik kullanım sınırları çıkartılmaktadır.

Bir sonraki bölümde kaotik öznitelikler steganalizinden türetilen ve literatüre ilk defa bu tez kapsamında kazandırılan özgün konuşma kodlayıcı tanıma anlatılmaktadır. Bahse konu konuşma kodlayıcı tanıma, konuşma sinyaline ait bit dizilerinin çeşitli analiz süreçlerine sokularak, üretimlerinde kullanılan kodlayıcıların tiplerinin tespit edilmesi olarak tarif edilebilmektedir. Öncelikle konuşma kodlayıcısının steganalizden nasıl türetildiği, daha sonra kuramsal açıdan nasıl çalıştığı açıklanmaktadır. Son olarak pratik uygulamalarına ait performans sonuçlarına yer verilmekte ve kaotik öznitelik steganaliziyle olan farklılıkları vurgulanmaktadır.

Bölüm 5'te, steganografi konusuna yeniden geri dönüş yapılmaktadır. Bu defa gizli veri gömme uygulaması MELP yerine daha yüksek kanal kapasiteli GSM 6.10 üzerinde gerçekleştirilmektedir. Uygulanan gizli veri gömme (GSM 6.10 RPE EÖB: *regular pulse excitation* en önemsiz bit) steganografik anlamda incelenmekte, konuşma kodlayıcılarının çıktılarının çerçeve yapısında olmalarına göre, Markov zincir

modelleme tabanlı yeni bir steganaliz yöntemi önerilmektedir. GSM 6.10 üzerinde yapılan veri gizleme yöntemi, yeni ve eski steganaliz yöntemlerince sınanmakta, yeni steganaliz yönteminin gömülen verileri yakalama konusunda başarılı olduğu ortaya konmaktadır. Daha sonra, steganografik izin azaltılması için istatistiksel modellemeye dayalı yeni bir matris gömme türü önerilmektedir. Yeni matris gömme yaklaşımı GSM 6.10 üzerinde uygulanmakta, eski ve yeni steganaliz yöntemlerine karşı ne kadar ilerleme sağladığı gösterilmektedir.

Sonuç bölümünde, bu tez kapsamında gerçekleştirilen özgün çalışmalara, MELP-MSVQ-QIM yöntemi, konuşma kodlayıcı tanıma, kodlanmış konuşma sinyalinde Markov zincir modelleme tabanlı yeni bir steganaliz ve istatistiksel modellemeye dayalı matris gömme çalışmalara vurgu yapılmakta, her birinin çalışma esasları son bir defa daha anlatılmaktadır.

Özgünlük iddiaları (*Claims of originality*)

- MELP MSVQ'sunda QIM gerçekleştirilerek kodlanmış ses sinyali içerisine gizli veri – damga bilgisi gömülmesi
- FNF özniteliklerinden istifade edilerek bit dizilerinin üretiminde kullanılan konuşma kodlayıcılarının tespit edilmesi
- Kodlanmış konuşma sinyali üzerinde Markov zincirlerine dayanan steganaliz yöntemi
- Markov zincir modelinden yararlanarak istatistiksel özellikleri korumayı amaçlayan sendrom kodlama yöntemi

2. KURAMSAL TEMELLER

Bu bölümde tezin anlaşılması ve yapılan çalışmaların takibi için gerekli olan kuramsal altyapı sunulmaktadır. Kuramsal altyapıda ilk önce, bu tezde önerilen özgün veri gömme yöntemlerinin çalışma sahası olan konuşma sinyali konu alınmaktadır; konuşma sinyalinin genel bir tanımı yapılmakta, insan vücudunca nasıl üretildiği anlatılmaktadır. Daha sonra, konuşma sinyalinin sayısal iletişim ortamlarından aktarımı için kullanılan konuşma kodlayıcıları ele alınmakta, çalışmalar sırasında yararlanılan kalite ölçüm metrikleri tanıtılmaktadır.

Konuşma sinyali ve konuşma kodlayıcı konularını takiben, veri gömme terminolojisi verilmektedir. Steganografi ve damgalama kavramları anlatılmakta, aralarındaki benzerlikler ve farklılıklar vurgulanmaktadır. Daha sonra bu tez kapsamında geliştirilmiş özgün yöntemlerle ilişkili olan literatür yöntemleri özetlenmektedir. Son olarak tezde önerilen özgün veri gizleme yöntemlerinin sınanmasında kullanılması amaçlanan steganaliz yöntemleri arz edilmektedir.

Kuramsal temellerde sunulan bilgilere ilaveten tezin sonunda yer alan eklerdeyse tezin anlaşılması kolaylaştırabilecek faydalı bilgilere yer verilmektedir. Ek 1'de doğrusal öngörü parametreleri konusu detayları göz önüne alınırken, Ek 2'de temel düzeyde kaos teorisi anlatılmaktadır. Daha sonraki eklerde (Ek 3-7) tez kapsamında kullanılan konuşma kodlayıcıları tanıtılmaktadır.

2.1 Konuşma Sinyali

Konuşma sinyali sinir sistemi koordinasyonunda gırtlak, akciğerler, kaslar, iskelet sistemleri tarafından yaratılan akustik bir sinyaldir. Akciğerlerdeki hava, diyafram ve karın kasları yardımıyla ses tellerinden geçerek ham sesi oluşturur. Ham ses daha sonra normal bir insanda toplam uzunluğu yaklaşık 17 cm olan ses üretim yolu vasıtasıyla (*vocal tract*) biçimlendirilir (Rotheberg 1981).

Ses üretim yolu, glotis olarak da isimlendirilen ses tellerinden ve aralarında yer alan boşluktan itibaren başlar, geniz, ağız boşluğu, dil, burun, sinüsler, dişleri takip ederek dudaklara kadar uzanır. Şekil 2.1'de gösterilen ses üretim yolu ham sesi formant frekanslarında rezonansa uğratarak ve 1 ms'nin katlarında korelasyonlara sahip olacak şekilde biçimlendirir. Söz konusu formantların frekansları sinir sistemince ses yolundaki organların pozisyonları değiştirilerek ayarlanır (Örneğin dilin pozisyonu). Matematiksel anlamda ses üretim yolu kısa süreli (*short-term*) özellikli bir süzgeçtir ve bahse konu bu süzgecin parametreleri (veya başka bir deyişle ses üretim yolunun şekli) zamanla, 20 ms'nin katlarında görece yavaş bir şekilde değişim gösterir (Quatieri 2001).

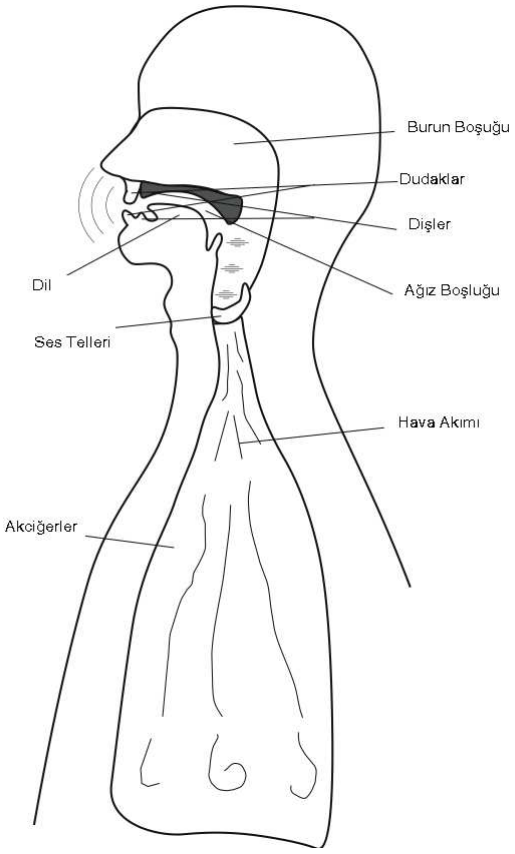
Konuşma sinyali maddeden oluşan bir ortamda moleküllerin titreşmeleriyle, sıkışıp genleşmeleri ile dalgalar halinde yayılır. Genellikle 500 – 2000 Hz frekansına sahip enine ve boyuna sinüzoidal dalgalardan meydana gelmektedir. Quatireri'ye (2001) göre konuşma sinyali anlık uyarım durumuna göre üç farklı sınıfa ayrıştırılabilir:

Ötümlü sesler (*voiced sounds*): Ötümlü sesler, ses tellerinin açılıp kapanmasıyla böylece ciğerlerden gelen havanın ara ara kesilip serbest bırakılmasıyla üretilirler. Ses tellerinin bu şekilde kullanılması neticesinde yarı periyodik (*quasi-periodic*) darbeler oluşur ve bu darbelerin oluşma sıklığı (*rate*) konuşma sinyale ait perde periyodunun (*pitch period*) alacağı değeri belirler. Perde periyotları 2 ila 20 ms'lik değerler arasında değişkenlik gösterir. Ötümlü seslerde hem kısa süreli hem de uzun süreli korelasyonlar bulunur.

Ötümsüz sesler (*unvoiced sounds*): Ötümsüz sesler glotis açıkken (ses telleri devrede değilken) ciğerlerden yola çıkan yüksek hızlı hava akımlarının ses üretim yolunda sürtünmeye uğratılarak akıtılmasıyla oluşturulur. Ses üretim yolunun engelleyici ve sürtünmeyi arttırıcı bir yapıya getirilmesi hızlı hava akımının türbülansa girmesine yol açar. Söz konusu türbülanstan geçen hava akımı gürültüyle benzer özellikler gösterir. Ötümsüz seslerde sadece kısa süreli korelasyonlar bulunur.

Doğada insan tarafından üretilen kimi sesler yukarıda tanıtılmış olan sınıflardan sadece birisine mensup olurken, kimi sesler birden fazla sınıfın özelliklerini bir arada taşır.

Örneğin, bünyelerinde sınıflar arası geçişler barındıran sesler ile sürtüşmeli ötümlüler birden fazla sınıfa ait özellikleri bir arada bulundurur.

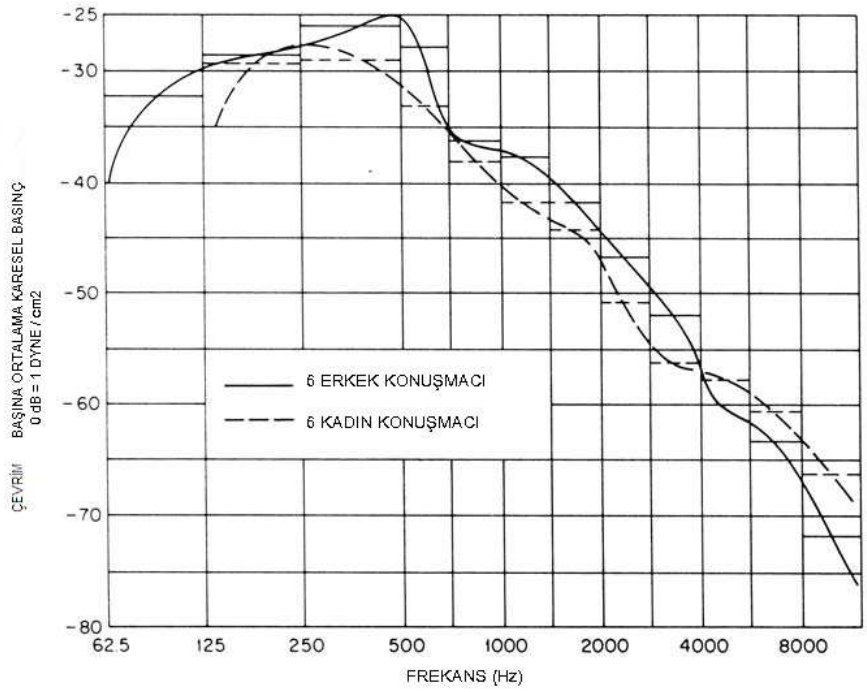


Şekil 2.1 Konuşma sinyalinin üretimi (Johnson 2010)

2.1.1 Konuşma sinyalinin modellenmesi

Bir önceki kısımda da belirtildiği üzere konuşma sinyali insan tarafından üretilen, kendine has bir takım özellikleri olan, hiç de rastgele olmayan özel bir tür akustik sinyaldir. Konuşma sinyalinin sadece dalga şekli baz alındığında, sinyal 4 ila 10 kHz bant genişliğine yayılabilmektedir (Şekil 2.2, Rabiner ve Schafer 1978), buna bağlı olarak 8 ila 20 kHz'de örneklenmelidir ve yüksek kaliteli sayısal aktarımı için her bir örnek 12 bitten fazla olacak (doğrusal nicemleme koşulları altında) şekilde sayısallaştırılmalıdır. Diğer bir deyişle konuşma sinyalinin dalga biçimi 96 ila 320 kbps'lik bilgi hızına sahiptir.

Öte yandan temel seviyede, çoğu dilde yalnızca 32 ila 64 adet fonem bulunmaktadır ve konuşma sinyalinin 1 saniyesinde 10-15 fonem yer almaktadır (Rabiner ve Schafer 1978). Bu duruma göre aslında konuşma sinyaliyle aktarılan dil kapsamındaki bilgi 1 saniyede yalnızca 60 ila 90 bit arasındadır. Elbette konuşma sinyali içerisinde dille taşınmayan, yani cümlelerle, kelimelerle, hecelerle ve fonemlerle taşınmayan bilgiler de mevcuttur. Ancak dil kapsamında olmayan bu bilgiler ve içerikleri ne olursa olsun, hakikatte düşük hızlı temel bilgi yüksek hızlı dalga biçimine dönüştürülür.

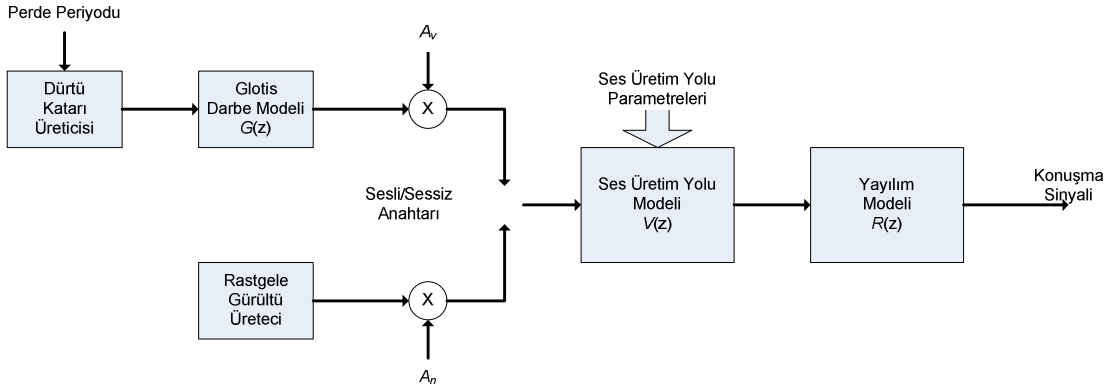


Şekil 2.2 Konuşma sinyalinin uzun süreli güç yoğunluğu spektrumu (Rabiner ve Schafer 1978)

Konuşma sinyali işleme mühendisliği en az işlem gücüne başvurarak ve en az bant genişliğini harcayarak en yüksek ses kalitesine ulaşmayı hedefler. Bu amacın yerine getirilebilmesi için konuşma sinyalinin üretim sürecinin doğru bir şekilde modellenmesi büyük önem arz eder. Şekil 2.3'te Rabiner ve Schafer (1978) tarafından hazırlanmış olan en genel-basit üretim modeli verilmiştir. Söz konusu modelde ötümlü sesler ile

ötümsüz sesler için ayrı şekillerde uyarımlar oluşturulmaktadır. Anahtarlanmış uyarım sinyali ses üretim yoluna sokulmakta, $V(z)$ tarafından biçimlendirilmektedir.

Girdi uyarım sinyaline kısa süreli korelasyonlar kazandırılarak konuşma sinyali elde edilmektedir. Kelly ve Lochbaum (1962) bahse konu ses üretim yolunu birbirlerine peş peşe eklenmiş kayıpsız tüpler olarak modellemişlerdir. Kayıpsız tüplerin genişlikleri ve uzunlukları sinir sistemi tarafından ayarlanmakta, bu sayede istenilen niteliklerde süzgeçler yaratılabilmektedir.



Şekil 2.3 Rabiner ve Schafer (1978)'e göre konuşma sinyalinin genel kesikli zaman üretim modeli

2.1.2 Doğrusal öngörü katsayıları

Konuşma sinyalinin modellenmesi bölümünde belirtildiği üzere, ses üretim yolunda girdi sinyal üzerinde kısa süreli korelasyonlar eklenir. Bahse konu korelasyonlar konuşma sinyali üretim modeline göre LPC (doğrusal öngörü katsayıları – *linear prediction coefficients*) sentez süzgeci tarafından oluşturulur. Bu noktada bir hatırlatma yapmak gerekirse, konuşma sinyali üretim modeline göre ses üretim yolunun yapısının yavaşça değiştiği varsayılır; örneğin 20 ms'lik bir çerçeveye ait tüm örnekler aşağı yukarı aynı değerlerdeki LPC sentez parametrelerince biçimlendirilir. Bu olguya dayanarak, kısa süreli LPC modellemesi yapılacak konuşma sinyali çerçevelere bölünerek her çerçeve için ayrı doğrusal öngörü katsayıları hesaplanır.

Tezin son kısmında yer almakta olan Ek 1 bölümünde doğrusal öngörü parametreleriyle ilgi faydalı olabilecek teknik detaylara yer verilmiştir. Bahse konu ek bölümde ilk olarak kayıpsız tüp modeliyle doğrusal öngörü parametreleri arasındaki ilişki anlatılmış, daha sonra LPC değerlerinin hesaplanması için geliştirilen matematiksel yöntemler tanıtılmıştır. Son olarak hesaplanan bu LPC değerlerinin en doğru şekilde nasıl nicemlenebileceği detaylı bir biçimde işlenmiştir.

2.1.3 Konuşma sinyalinde doğrusal olmayan özellikler

Konuşma sinyali, insan vücudu tarafından üretildiği sırada etki eden bazı mekanizmalar nedeniyle gerçek anlamda doğrusallığa sahip değildir, yapısında konuşma kodlayıcısı tasarımlarını zorlayan pek çok doğrusal olmayan unsur içermektedir.

Ses tellerinin doğrusal olmayan özellikleri: Bir doğrusal modelde çıktı ile girdi birbirleriyle orantılı olmalıdır, ancak ses tellerinde üretilen ses dalgaları farklı genlik değerlerinde farklı şekiller almaktadır. Titze (1988 ve 1992) tarafından gerçekleştirilen detaylı çalışmalarda, sadece değişen genlikle üretilen darbelerin spektral dağılımının değişmediği, bu olguya ek olarak temel frekansla beraber spektral zarfın da biçim değiştirdiği belirlenmiştir. Genlik etkisinin haricinde, ses tellerinin faz uzayında çeşitli kaotik çatallaşmalar da oluşturduğu anlaşılmıştır.

Üretim esnasında oluşan türbülans ve düzlem-dışı dalga yayılımı: Konuşma sinyalinin ötümsüz kısımları ses üretim yollarının daraltılmasıyla elde edilir; adı geçen daraltma işlemleri üretilen ses dalgasının türbülans oluşturmasına yol açar. Türbülans, üretilen konuşma sinyaline bariz kaotik etkiler katar. Her ne kadar konuşma sinyalinin geleneksel üretim modellerinde, vokal kanalların sesi düzlemdeymiş gibi yaydığı varsayılmış olsa da, gerçekte oluşturulan ses çeşitli girdaplar oluşturarak yayılım göstermektedir (Teager ve Teager 1990).

2.1.4 Konuşma sinyalinde kaotik belirtiler

İlk olarak Teager ve Teager (1990) yaptıkları çalışmalarda konuşma sinyalinin doğrusal olmadığı göstermişlerdir. Daha sonra Maragos (1991) bu konuda daha fazla ilerleme kaydederek konuşma sinyalinin kaotik özellikler içerdiği belirlemiştir; kapasite boyu yöntemiyle yaptığı hesaplamalarda sürtünmeli ünsüzler için en az 1.7, ünlüler için 1.2 civarında fraktal boyutu sonuçlarına ulaşmıştır.

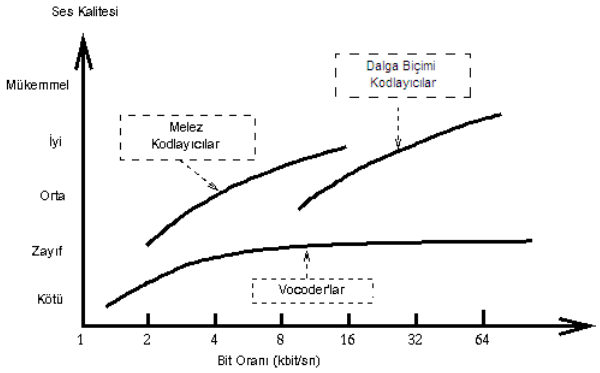
Boshoff (1991), McDonnel ve Datta (1994) da Maragos gibi fraktal boyu üzerinde çalışmışlar, fraktal boyutlarının 1'den 2'ye kadar değiştiğini belirlemişlerdir, ancak elde ettikleri sonuçların önemli oranda hata payı içerebileceğini ifade etmekten kaçınmamışlardır. Pickover ve Khorasani (1986) benzer fraktal boyut analizlerini cümleleri kapsayabilecek uzunluktaki ses örnekleri için tekrar etmişlerdir. Durağan olmayan uzun veri çerçeveleri üzerinde gerçekleştirdikleri çalışmalar sonucunda, konuşmacı özniteliklerini elde etmeyi başarmışlardır.

Marcato ve Mumolo (1993) LPC süzgecinden arta kalan tortu sinyali üzerinde çalışmışlar, söz konusu tortu sinyalinin belli ölçüde fraktal yapılardan oluştuğunu tespit etmişlerdir. Söz konusu fraktal yapıların Barnsley ve Sloan'ın (1989) görüntü kodlama çalışmalarındaki gibi kodlama verimliliğinin yükseltilmesinde fayda sağlayabileceğini belirtmişlerdir.

Konuşma sinyalinin fraktal boyutu üzerinde yapılan tartışmalar çeşitli çalışmalarda elde edilen çelişkili sonuçlar yüzünden bir türlü sonlandırılamamıştır. Tishby (1990) korelasyon yöntemiyle ünlülere ait fraktal boyutlarının 3'ten 5'e kadar olması gerektiğini öne sürmüş, Berhard (1991) ve Kubin (1991) Tishby'nin aksine yine eski değerleri önermişlerdir. Narayanan ve Alwan (1995) Lyapunov spektrum analizi neticelerine göre ünlülerin zaten kaotik yapıda olmadıklarını kaydetmişler, sürtünmeli ünsüzler içinse 4-7 arası boyutlar öngörmüşlerdir. Senvirathne vd. (1992) ise fraktal boyut hesabının sadece yararlı bir konuşma işleme aracı olarak ele alınması gerektiğini belirtmişler, rakamsal sonuçlara fazlaca rağbet edilmemesini tavsiye etmişlerdir.

2.2 Konuşma Kodlayıcıları

Günümüzde genel olarak kullanılmakta olan konuşma kodlayıcıları çalışma esaslarına göre üç farklı sınıfta (kategori altında) değerlendirilmektedirler (Chu 2003): Dalga biçimi kodlayıcıları, parametrik kodlayıcılar ve melez kodlayıcılar. Tipik dalga biçimi kodlayıcıları yüksek bit oranlarında (*high bit rate*) çalışırlar ve yüksek kaliteli ses kodlaması gerçekleştirirler. Vocoder'lar (*voice coder*) olarak da isimlendirilen parametrik kodlayıcılarsa konuşma sinyalini ses üretim yolunu modelleyerek çok düşük bit oranlarına kadar sıkıştırırlar, ancak kaliteden de ciddi ödünler verirler. Melez kodlayıcılar ise hem parametrik hem de dalga biçimi kodlamasını bir arada kullanırlar, düşük bit oranlarında çıktı üretirler. Şekil 2.4'te farklı kodlayıcı sınıflarının değişik bit oranlarında elde ettikleri ses kaliteleri sunulmuştur.



Şekil 2.4 Farklı kodlayıcı sınıflarının ses kalitesi vs bit oranı grafikleri

2.2.1 Parametrik kodlayıcılar

Parametrik kodlayıcılar, yani vocoder'lar konuşma sinyalini oluşturan dalga biçiminden ziyade üretim sürecini esas alarak kodlarlar. Genel bir parametrik kodlayıcı konuşma sinyalini azami oranda sıkıştırabilmek için önce konuşma sinyalinin üretim sürecini modellerinden yararlanır, süreci yerine getiren işlevsel blokların durumlarını belirleyen parametrelerin değerleri hesaplar. Önceki bölümlerde de anlatıldığı üzere ses üretim yolu zamana göre değişim gösteren bir süzgeç olarak tanımlanabilir. Söz konusu süzgeç

ise ya beyaz gürültü ile ya da darbeler dizisi ile uyarılır ve böylece konuşma sinyali üretilir. Parametrik kodlayıcılar ses üretim yolunu betimleyen süzgece ait parametreleri, söz konusu süzgecin uyarım şeklini varsa süzgecin girdisi olan darbe dizilerini kodlarlar.

Kodlayıcı tarafta model parametreleri frekans ve/veya zaman bölgesi tekniklerinden faydalanılarak çeşitlik şekillerde hesaplanabilir. Parametrik kodlayıcılar 2400 bps hızı civarında çalışırlar ve kod çözümü neticesinde sentetik ses anlaşılır olsa da doğallıktan uzaktır. Ancak çoğu iletişim uygulamasında konuşmanın anlaşılır olması doğal olmasından çok daha önemlidir. Parametrik kodlayıcılarca kullanılan bit oranının arttırılması kaliteyi ancak belli bir nebze arttırma potansiyeline sahiptir; vocoder'lar ses üretim yolunu basitleştirerek modelledikleri, gerçek ses üretim yolunu tam olarak modelleyemedikleri için, bit oranı ne kadar arttırılırsa arttırılsın kalite belli bir düzeyi geçemez. Parametrik kodlayıcılar yüksek işlem gücü duyarlar ve dalga biçimi kodlayıcılara kıyasla daha fazla gecikmeye sebebiyet verirler.

2.2.2 Dalga biçimi kodlayıcıları

Dalga biçimi kodlayıcılar, ses (konuşma) sinyalinin üretim şekline dair herhangi bir bilgiden yararlanmaksızın, yeniden yapılandırılan sinyale ait olan dalga biçimini orijinal dalga biçimine en az hatayla yaklaştırmaya çalışırlar. Bu yöntemler sinyallerden ve içeriklerinden bağımsızdır, bu yüzden konuşma sinyali dışındaki akustik sinyallerin kodlanmasında da başarıyla kullanılabilirler. Çoğunlukla düşük oranda karmaşıklık içerirler ve yüksek bit oranlarında yüksek kaliteli kodlama sunarlar. Bit oranı düşürülürse yeniden yapılandırılan sinyalin kalitesinde hızlı düşüşlere tanık olunur.

En basit dalga biçimi kodlaması PCM (darbe kodu modülasyonu – *pulse code modulation*) – darbe kodu modülasyonudur. Konuşma sinyali 4 kHz'lik bir frekans bandına sığar ve bu dar bantlı sinyalin 8 kHz'de örneklenmesi yeterlidir. 12 bitlik doğrusal nicemleme (96 kbps) ile iyi ses kalitesi elde edilebilir. Söz konusu 96 kbps'lık bit oranı doğrusal olmayan örneğin logaritmik nicemleme teknikleri ile biraz daha

azaltılabilir. PCM kodlamasının en önemli avantajları çok basit olması, yüksek kalite sağlaması ve çok düşük gecikmeye yol açmasıdır.

Ses kodlamada en çok kullanılan yöntemlerden birisi de bir sonraki değerin bir öncekinden öngörülmesi yaklaşımıdır. Konuşma sinyali üretiminin de bir sonucu olarak konuşma sinyali örnekleri arasında korelasyonlar bulunur. Bahse konu korelasyonlardan en basit yararlanma biçimi, DPCM (ayrımsal PCM – *differential PCM*) farksal darbe kodu modülasyonudur ve bu modülasyon tipinde örnekler arası farklılıklar nicemlenir. DPCM kodlaması mantığına öngörüye göre uyarlanabilir nicemleme yeteneklerinin eklenmesiyle ADPCM (uyarlanabilir ayrımsal PCM – *adaptive differential PCM*) kodlaması elde edilmiştir. 16, 24, 40 gibi seçenekleri de bulunan ADPCM kodlamasının 32kbps hızındaki versiyonu 64 kbps'lık PCM kodlamasına denk bir kalite sunar.

PCM, DPCM ve ADPCM gibi zaman bölgesinde çalışan basit dalga biçimi kodlayıcılarına ek olarak, frekans bölgesinde çalışan dalga biçimi kodlayıcıları da mevcuttur. Alt-bant kodlaması SBC (*sub-band coding*) girdi ses sinyalini alt bantlara ayırır, her bir alt bandı diğerlerinden bağımsız olarak örneğin ADPCM benzeri bir şekilde kodlar. Alıcı uçta birbirlerinden bağımsız olarak kodlanmış alt bantların kodları çözülür ve birleştirilir. Duyusal anlamda daha fazla hassasiyet gösteren frekans bantlarına kodlanmış çıktının daha çok biti ayrılarak hem gürültüye karşı dayanıklılık kazanılır ve hem de yeniden yapılandırılan sesin kalitesi arttırılabilir. 16-32 kbps hızlarında yüksek kaliteli konuşma sinyali taşınabilir. Süzgeçleme, zaman bölgesindeki sinyalin frekans alanına aktarılması gibi matematiksel işlemler kodlama karmaşıklığını arttırırlar ve fazladan gecikmeye neden olurlar. Yine de bu tip kodlayıcıların karmaşıklıkları ve neden oldukları gecikmeler parametrik ve melez kodlayıcılara kıyasla daha düşük olur.

2.2.3 Melez kodlayıcılar

Melez kodlayıcılar dalga biçimi kodlayıcılar ile parametrik kodlayıcılar arasındaki boşluğun doldurulması amacıyla geliştirilmişlerdir, her ikisinin de olumlu özelliklerini bir arada bulundururlar. Hem yüksek kaliteli hem de düşük hızlı ses kodlaması

gerçekleştirebilirler. Bir melez kodlayıcı, tıpkı bir parametrik kodlayıcıda olduğu gibi ses üretim yolunu modeller, ancak ses üretim yolunun uyarımında kullanılacak girdiyi daha farklı bir şekilde belirler. Sentetik sinyalin olabildiğince orijinal sinyale benzemesi için AbS (sentez ile analiz = *analysis by synthesis*) yöntemlerini kullanarak uyarım girdisini ayarlar.

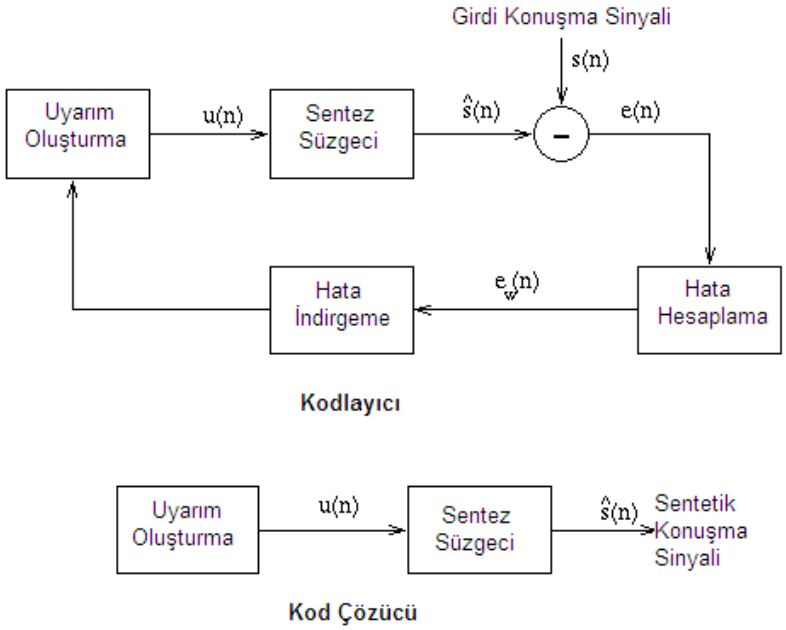
Bu tür kodlayıcıların ilk örneği olan MPE kodlayıcısı (çoklu darbe uyarmalı – *multi-pulse excited codec*) Atal ve Remde (1982) tarafından literatüre kazandırılmış olup, bu kodlayıcıyı düzgün darbe uyarmalı (*RPE: regular pulse excited*) ve kod uyarmalı doğrusal öngörü (*CELP: code-excited linear predictive*) (Schroeder ve Atal 1985) kodlayıcıları takip etmiştir. Şekil 2.5'te verilmiş olan tipik bir AbS kodlayıcısında, girdi konuşma sinyali önce yaklaşık 20 ms'lik çerçevelere ayrılır, sonra her bir çerçevenin sentez süzgeci parametreleri hesaplanır. Süzgeç parametrelerinin hesaplanmasından sonra sıra uyarım sinyalinin belirlenmesine gelir. Uyarım sinyali sentetik ve orijinal sinyaller arasındaki hata en düşük olacak şekilde belirlenir. Sentetik ve orijinal sinyaller arasındaki farkın en aza indirilmesi işlemi dalga biçimi kodlayıcılarının çalışmalarıyla benzerlikler gösterir.

Basit bir melez kodlayıcı kodlanacak konuşma sinyalini çerçeveler halinde işler. Her bir çerçeve için o çerçeveye ait süzgeç parametreleri ile hatayı en aza indirgeyen uyarım sinyali parametreleri hesaplanır. Hesaplanan parametreler bir araya getirilip paketlenir ve kodlanmış çerçeve olarak karşı uca iletilir.

Kod çözücüsünde tarafında da her kodlanmış çerçeve ayrı ayrı işlenir. Kodlanmış bir çerçeve için önce çerçeve kapsamında paketlenmiş içerik ayrıştırılır, daha sonra ayrıştırılan parametrelerden sentez süzgeci oluşturulur. Oluşturulan bu sentez süzgecine yine ayrıştırma sonucu elde edilmiş olan uyarım sinyali girdi olarak sokulur. Son olarak sentez süzgeci çalıştırılarak sentetik konuşma elde edilir.

Bir konuşma kodlayıcısına ait sentetik sesin kalitesi, konuşma sinyalinin ötümlü kısımlarında gözlemlenen uzun süreli periyodikliğin dikkate alınmasıyla daha da iyileştirilebilir. Eğer uzun süreli periyodiklik, yani başka bir ifadeyle perde periyodu,

uyarım sinyalinin üretiminde değerlendirilebilirse sentetik konuşma sinyali daha doğal bir nitelik kazanır. Çoklu darbe uyarmalı (MPE) ve RPE kodlayıcıları gibi perdeyi dikkate almadan çalışan melez kodlayıcılar tasarlanmış olsa da, CELP gibi daha gelişmiş kodlayıcılar için sentezlenen sesin kalitesi açısından perde periyodu ve doğru şekilde hesaplanması son derece önemlidir.



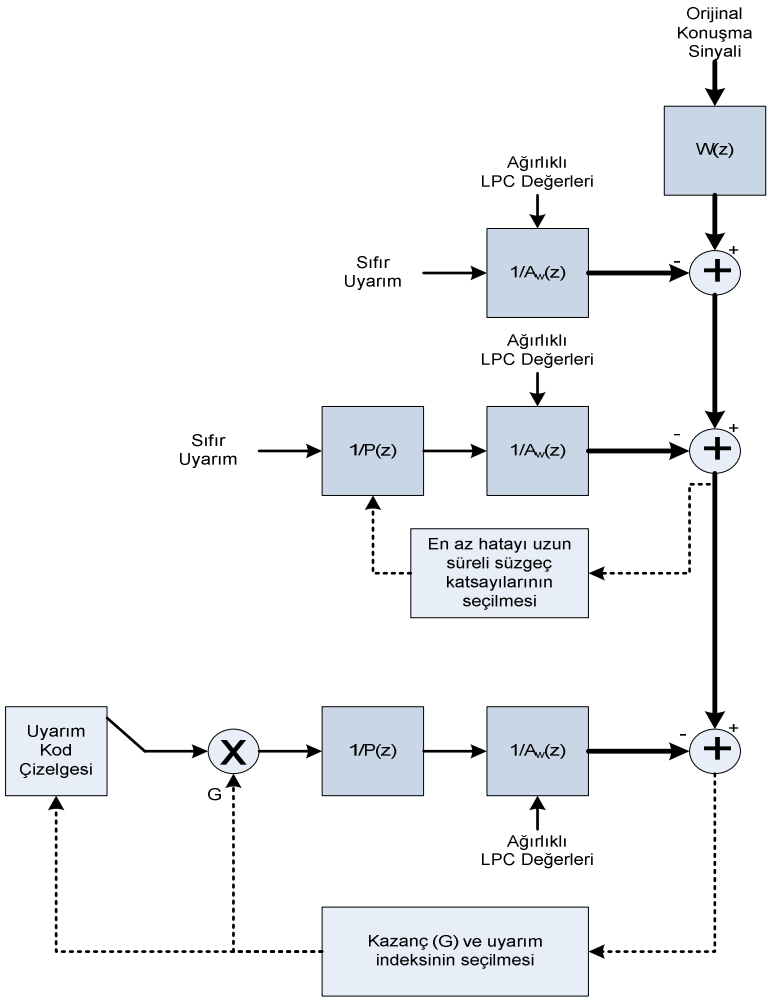
Şekil 2.5 Tipik bir AbS kodlayıcısı ve kod çözücüsü (Kondoz 2004)

Şekil 2.6'da blok şeması verilen genel CELP kodlayıcısında konuşma sinyali 20 ila 30 ms'lik örneklerden oluşturulmuş çerçeveler halinde işlenir. Konuşma sinyali ilk olarak ağırlık süzgecine sokulur. Bahse konu ağırlık süzgeci ile sentetik ve orijinal sinyaller arasındaki hata indirgemesinin (yani AbS'nin) insan duyu organının algısal özelliklerine göre gerçekleştirilmesi sağlanır.

İnsan duyu organına göre filtrelenmiş konuşma sinyali daha sonra kısa süreli analize sokularak LPC parametreleri hesaplanır. Süzgeci oluşturan parametrelerin sayısı, *p* genellikle 10 sayısı civarından seçilir. LPC'ler ufak hatalara karşı oldukça hassastır; bu yüzden bu parametreler daha kararlı olan LAR (logaritmik alan oranı – *log-area-ratio*), IS (ters sinüs – *inverse sine*) veya LSF (spektral hat frekansları – *line spectral*

frequencies) gibi değerlere dönüştürülür (Bknz: Ek 1). Hatırlanacağı üzere, Söz konusu LAR, IS ve LSF değerleri aslında LPC'lerin türlü matematiksel dönüşümleridir. Hesaplanan dönüştürülmüş LPC değerleri nicemlenerek kısa süreli analiz süzgeci oluşturulur. İnsan duyu organına göre filtrelenmiş sinyal oluşturulan kısa süreli analiz süzgecinden geçirilerek tortu sinyali elde edilir.

Elde edilen tortu sinyali bu defa uzun süreli analize alınır ve böylelikle gecikme ve skala değerleri hesaplanır. Tortu sinyalinin, gecikme ve skala parametrelerinden ibaret uzun süreli analiz süzgecinden geçirilmesiyle tortunun da tortusu bulunur. Kısa ve uzun süreli analiz süzgeçlerinin artığı olan bu en son kalan tortudan, sentetik sinyali orijinale en çok yakın kılan kod çizelgesi indeksleri ve kazanç değerleri seçilir.



Şekil 2.6 Genel CELP kodlayıcısı (Kondoz 2004)

2.2.4 Kodlanmış konuşma sinyali için kaotik özniteliklerin hesaplanması

Bu bölümde, kodlanmış konuşma sinyali için Ek 2'de tanımlanmış olan hatalı en yakın komşuluk oranı ve Lyapunov katsayılarının hesaplanması anlatılmaktadır. Ortalama ve varyans değerleri ait oldukları dizinin istatistiksel bazı özelliklerini nasıl ortaya koyuyorlarsa, bahse konu kaotik öznitelikler de hesaplandıkları dizinin kaotikliğini ortaya koyarlar. Çok kısa bir özet vermek gerekirse hatalı en yakın komşuluk oranı dizilerin gömme boyutlarının tespitinde faydalı olurken, Lyapunov katsayıları dizilerin hangi ölçüde öngörülebilir olduğunu belirtirler. Bahse konu öznitelikler Bölüm 2.6.6'daki (kaotik özniteliklere göre) steganalizde ve Bölüm 4'teki kodlayıcı tanımada görev almaktadırlar ve her iki tip uygulamada da farklı istatistikî – kaotik özelliklere sahip dizilerin birbirlerinden ayırt edilmesinde kullanılmaktadırlar. Gerçek hayatta gözlemler sonucunda kaydedilmiş dizilerin kaotik öznitelikleri sadece pratik araçlarca hesaplanabilirken, matematiksel olarak ifade edilebilen dizilerin kaotik öznitelikleri teorik olarak da hesaplanabilir.

Kodlanmamış konuşma sinyaline ait bazı özellikler (mesela istatistikî özellikler) belli oranlarda kodlanmış bit dizilerine nüfuz eder. Bahse konu nüfuz eden özelliklerin cinsleri ve katılım miktarları kodlayıcının çalışma esaslarına ve işleyiş detaylarına göre değişkenlik gösterir. Kodlanmış çıktıya nüfuz edecek özellikler ve bu özelliklerin nüfuz etme oranları-miktarları, kodlayıcının çalışma esaslarının sistematik olarak çözümlenmesiyle öngörülebilir. Hatta bu tip bir çözümleme-inceleme-öngörü zinciri sonucunda edinilecek bilgi birikimi, kodlayıcının iyileştirilmesinde faydalı olabilir. Ancak böyle bir öngörünün oluşturulabilmesi için gerekli olan çalışma başlı başına apayrı bir tez konusudur.

En yakın hatalı komşular ve Lyapunov katsayıları gibi kaotik öznitelikler ve bu özniteliklerin hesaplanmasını temin eden araçlar, aslında sadece matematiksel algoritmalardır ve içerikleri kaotik olsun olmasın tüm veri dizilerine uygulanabilirler. Kaotiklik kavramından bağımsız olarak çeşitli farklı sınıflara (Örneğin gizli veri içeren ve içermeyen) ait çıktıların hesaplanan özniteliklerinde ayrışmalar gözlemlenebiliyorsa, oluşan ayrışmalardan istifade eden sınıflandırıcılar (Örneğin steganalizör) tasarlanabilir.

Kaldı ki, konuşma sinyali zaten belli ölçüde kaotik davranışlara sahiptir ve literatürde bunu ispatlayan çok sayıda çalışma yapılmıştır. Özetle:

- Kodlanmış bit dizilerinin rastgele olmadıkları bilindiğine göre (Örneğin Rahikka vd. (1990) MELP çıktısının rastgele olmadığını ortaya koymuşlardır.)
- Gerçek hayata ait tüm diziler gibi kodlanmış bit dizilerinin de Çizelge 2.5'teki sınıfların bir birleşimi olduğu varsayılabileceğine göre
- Gizli verinin farklı bir sınıf (Sözde rastgelelik sınıfı) olarak bu birleşime eklenmesinin apaçık bir şekilde sınıfların katılım oranlarını değiştireceğine göre

gizli veri eklemenin kaotik özniteliklerde mesela fraktal boyutunda ve Lyapunov katsayılarında çeşitli değişikliklere yol açması beklenebilir. Burada önemli olan husus, söz konusu değişikliklerin ne kadar, hangi oranda ve hangi güvenilirlikte ölçülebilir olacaklarıdır. (*Bknz*: Bölüm 2.6.6 Kaotik özniteliklere göre steganaliz)

Bu tez kapsamında genel olarak iki tipteki kaotik özniteliklerden yararlanılmıştır: Hatalı en yakın komşular oranı ve Lyapunov katsayıları. Söz konusu kaotik öznitelikler bilgisayar ortamında TISEAN 3.0.1'de (Hegger vd. 2007) yer alan açık kaynak kodlu yazılımlar vasıtasıyla hesaplanmıştır:

lyap_spec: Sano ve Swada (1985) tarafından önerilen yöntem kullanılarak her bir boyut için *m* adet, *D* farklı boyut için $m \times D$ adet Lyapunov katsayısı hesaplanmaktadır.

false_nearest: Kennel ve Abarbanel (2002) tarafından önerilen yöntem kullanılarak *T* kadar geciktirilen *m* adet çoklu-değişkenin (*multi-variate*) *D* boyuttaki hatalı en yakın komşu oranları hesaplanmaktadır.

2.2.5 Tez kapsamında kullanılan konuşma kodlama algoritmaları

Bu tez düşük veri hızlarında çalışan konuşma kodlayıcılarına gürbüz bilgi saklama ve damgalamayı konu aldığından, pek çok konuşma kodlayıcısı çalışma kapsamına

alınmıştır. Giriş bölümünde de bahsedildiği üzere, bu tez dâhilinde ilk olarak US DoD tarafından standart haline getirilmiş karma uyarmalı doğrusal öngörü (*MELP: mixed excitation linear prediction*, Anonymous 1999) konuşma kodlayıcısı üzerinde çalışılmıştır. Söz konusu MELP kodlayıcısı üzerinde özgün bir veri gizleme yöntemi geliştirilmiş, daha sonra söz konusu özgün veri gizleme yönteminin steganografik dayanıklılığı ölçülmüştür. Ölçümler sırasında başvurulan kaotik özniteliklere göre steganaliz yönteminin çalışma esaslarından, steganografi dışında da yararlanılabileceği fark edilmiş ve böylece konuşma kodlayıcısı tanıma yöntemi geliştirilmiştir. Konuşma kodlayıcısı tanıma yöntemi MELP, PCM, G.726 (Anonymous 1990), G.729 (Anonymous 2006), GSM 6.10 (Anonymous 1999) ve GSM 6.60 (Anonymous 1999) kodlayıcıları üzerinde test edilmiştir. Tez çalışmaları kapsamında son olarak GSM 6.10 FR kodlayıcı üzerinde çalışan steganaliz ataklarına karşı dayanıklılığı arttırılmış bir veri gizleme yöntemi önerilmiştir.

2.3 Konuşma Sinyalinin Kalitesinin Ölçülmesi

Kodlama, veri saklama, vb. gibi sayısal sinyal işleme sistemlerinde çıktı sinyal ile girdi sinyal birbirlerinden farklılık arz ederler; nicemleme, süzgeçleme, vb. gibi sinyal işlemenin bünyesinde bulunan süreçler işlenen sinyalde çeşitli kayıplara veya bozulmalara neden olabilir. Bir sinyal işleme sisteminin performansı çıktısı olan sinyalin kalitesine doğrudan bağlıdır. Konuşma sinyalini işleyen sistemlerin performanslarının değerlendirilebilmesi için literatürde önerilmiş pek çok farklı tipte kalite ölçme kriterleri mevcut olup, bu tez kapsamında bunların üçünden faydalanılmıştır.

2.3.1 Spektral bozulma

Bir sinyal işleme sistemi için akla ilk gelebilecek objektif performans kriteri sinyal-gürültü oranı (SNR) olsa da, bahse konu SNR ölçümü sentetik konuşma sinyalinin kalitesinin değerlendirilmesinde (özellikle de parametrik kodlayıcılarının kullanıldığı senaryolarda) fazlaca işe yaramaz. Bu yüzden özellikle de konuşma kodlayıcılarının

performanslarının ölçümünde SNR'a benzeyen, ancak konuşma sinyalinin doğasına daha fazla uyum sağlamış performans kriterlerine başvurulur.

(2.1)'de formüle dökülmüş olan spektral bozulma, orijinal ve sentetik sinyallerin frekans bölgesindeki zarflarının karşılaştırılmasıyla hesaplanan objektif bir performans kriteridir. Bu tanımı biraz daha açmak gerekirse, Chu (2003) tarafından formüle döküldüğü biçimiyle, girdi sinyalin LP katsayılarının $(1/A_1(z))$ frekans bölgesindeki eşdeğerlerinin $(S_1(e^{jw}))$, çıktı sinyalin LP katsayılarının $(1/A_2(z))$ frekans bölgesindeki eşdeğerlerine $((S_2(e^{jw}))$, olan oranıdır.

$$SD^2 = \frac{1}{n_1 - n_0} \sum_{n=n_0}^{n_1-1} \left[10 \log\left(S_1\left(e^{\frac{j2\pi n}{N}} \right) \right) - 10 \log\left(S_2\left(e^{\frac{j2\pi n}{N}} \right) \right) \right]$$
$$S_1(e^{jw}) = \frac{1}{|A_1(e^{jw})|^2} \qquad S_2(e^{jw}) = \frac{1}{|A_2(e^{jw})|^2} \tag{2.1}$$

2.3.2 Kodlayıcı içinde indirgenmeye çalışılan hatalar

Bir konuşma kodlama algoritmasında, girdi sinyalin veya bit dizilerinin kodlanması sırasında çeşitli seviyelerde pek çok dönüşüm işlemi gerçekleştirilir. Örneğin gerçek sayılardan muhtelif vektörler olabilecek en az bit oranı işgal edecek şekilde tablo indekslerine dönüştürülür. Söz konusu dönüşümler, kodlanmamış sinyalde bulunan bazı bilgilerin kodlanmış sinyale aktarılmamasına yol açar. Oluşan bilgi kayıplarının minimize edilebilmesi için kodlayıcı, kodlanmış çıktı ile kodlanmamış girdi arasındaki hataları (kendi kriterlerine göre belirlediği ve hesapladığı hataları) en aza indirgemeye çalışır.

Hatırlanacağı üzere, genel (melez) bir konuşma kodlaması kısa süreli analiz, uzun süreli analiz ve uyarım belirleme aşamalarından oluşur. Birinci aşamada LPC değerleri belirlenir, ikinci aşamada perde değişkenleri hesaplanır ve son aşamada kalan tortuya göre uygun bir uyarım sinyali seçilir. Dolayısıyla birinci aşamada LPC'lere göre, ikinci aşamada perde değişkenlerine göre ve son aşamada uyarım sinyaline göre hatalar

oluşur. Genellikle bir önceki aşamadan kaynaklanan hatalar bir sonraki aşamada daha uygun kod kelimeleri seçilerek giderilemez.

Bu tez kapsamında MELP algoritmasında çalışan çok sayıda özgün gizli veri gömme yöntemi geliştirilmiştir. Söz konusu yöntemlerin veri saklama performanslarının nesnel bir şekilde değerlendirilebilmesi için gömülen gizli verilerin neden olduğu fazladan bozulma çeşitli kriterlere göre ölçülmüştür. MELP kodlaması sırasında LSF değerlerinin nicemlenmesi neticesinde ortaya çıkan ve indirgenmesi amaçlanan hata da (ağırlıklı Hamming uzaklıklarının) ölçüm kriterlerinden biri olarak seçilmiştir.

MELP algoritmasında LSF değerleri dört seviyeli bir vektör nicemleyici vasıtasıyla nicemlenir; nicemleyici sonuçları en az ağırlıklı Hamming uzaklıklarını (Anonymous 1999) sağlayan indeks değerlerinden oluşturulur. Standart MELP'te (gizli veri gömme uygulanmayan MELP'te) nicemlenmemiş LSF değerleri ile nicemlenmiş LSF değerleri arasındaki ağırlıklı Hamming uzaklığı farkı D_{sm} olarak tanımlanabilir:

$$D_{sm} = \sum_{k=1}^{10} w_k (l_k - l_k{}')^2 \tag{2.2}$$

(2.2)'de verilen D_{sm}, bir 22.5ms'lik MELP çerçevesi için LSF nicemleme işlemi sonucunda oluşan hata miktarıdır; eşitliği oluşturan (l_k') standart MELP ile nicemlenmiş LSF değerleri, (l_k) nicemlenmemiş LSF değerleri ve (w_k) ise algısal ağırlık katsayılarıdır. İlerleyen bölümlerde çok daha detaylı bir şekilde anlatılacağı üzere, bu tez kapsamında önerilmiş olan özgün gizli veri gömme yöntemleri LSF'lerin nicemlenme işlemlerini gizli veri bitlerine göre kontrol altında tutarlar – değiştirirler. Bu yüzden LSF indekslerine gizli veri gömülen MELP'te oluşan Hamming uzaklığı, D_{mm} standart MELP'e göre daha yüksek olabilir:

$$D_{mm} = \sum_{k=1}^{10} w_k (l_k - l_{mk}{}')^2 \tag{2.3}$$

(2.3)'te verilen (l'_{mk}) gizli veri gömen MELP tarafından nicemlenmiş LSF değerleridir. Gizli veri eklemenin ek maliyeti, modifiye edilmiş MELP'te oluşan ağırlıklı hatanın, standart MELP'te oluşan ağırlıklı hatadan farkıdır: ($D_{mm} - D_{sm}$).

D_{mm} ve D_{sm} değerleri örnekten örneğe değişkenlik gösteren nümerik değerler olduğundan, değerlendirilmeleri ve takip edilmeleri kolay değildir; bu yüzden bozulma maliyetinin göreceli olarak ifade edilmesi daha doğru bir yaklaşımdır. Fazladan oluşan bozulma ($D_{mm} - D_{sm}$), standart MELP'teki bozulmaya (D_{sm}) bölünerek, anlaşılması ve değerlendirilmesi daha kolay olan oransal – göreceli hata kriteri (D_{fm}) elde edilebilir:

$$D_{fm} = \frac{D_{mm} - D_{sm}}{D_{sm}} \tag{2.4}$$

2.3.3 Sinyal kalitesinin algısal özelliklerine göre değerlendirilmesi

Konuşma sinyali gibi, insan duyu organlarınca algılanan türdeki sinyallerin kalitelerinin değerlendirilmesinde insan faktörü çok önemli bir yer işgal eder. Bazı durumlarda nesnel kalite kriterlerine göre kaliteli olduğu izlenimi veren ses örnekleri insanlar tarafından yeteri kadar kaliteli bulunmayabilir, bazı durumlarda ise nesnel kalite ölçütlerine göre daha olumsuz değerlendirme sonuçlarına sahip ses örnekleri insanlar tarafından daha doğal olarak nitelendirilebilir. Bu yüzden algısal öneme sahip sinyal türlerinin kaliteleri bizzat insanlar tarafından ölçülmelidir.

Öte yandan, insanların şahsen katıldıkları testlerin gerçekleştirilmesi son derece maliyetli ve zordur; en basit testlerin bile güvenilir sonuçlar üretebilmesi için çok sayıda dinleyiciye gereksinim duyulur, her bir dinleyicinin duyma özellikleri eşsizdir (diğerlerinden farklıdır) ve dinleyicinin anlık psikolojik durumu kararlarında son derece etkindir. Uluslararası Telekomünikasyon Birliği, ITU-T, insan dinleyicilerle yapılan yüksek maliyetli testlere duyulan ihtiyacı azaltabilmek ve ürün geliştirme süreçlerini hızlandırabilmek için "konuşma kalitesinin algısal değerlendirilmesi" (*PESQ:*

perceptual evaluation of speech quality) adı altında, insan algısını modelleyerek konuşma sinyalinin kalite ölçümünü gerçekleştiren bir dizi standart yayımlamıştır.

Bu tez kapsamında geliştirilen ve içlerine yerleştikleri kodlayıcıların alt-parçalarıymış gibi işlev gören gizli veri gömme yöntemleri, konuşma sinyalinin standart kodlanmasını değişikliğe uğratmaktadırlar. Bu nedenle gizli veri gömen kodlayıcıların maliyetlerinin daha iyi anlaşılabilmesi ve ortaya konabilmesi için PESQ testlerinin uygulanması zaruri olmuştur. Testler kapsamında P.862 (Anonymous 2004) ve P.563 (Anonymous 2001) yazılımları kullanılmıştır ve bahse konu yazılımlarla MOS taklidi değerlendirme sonuçları elde edilmiştir. Yazılımlar hakkında bilgi vermek gerekirse, P.862 yazılımı tam referanslı (*FR: full reference*) bir PESQ yazılımıdır, MOS sonuçlarının üretimi için hem orijinal hem de kodlanmış sinyal örneklerine ihtiyaç duymaktadır. P.563 yazılımı ise P.862'nin aksine referanssız (*NR: no reference*) tipteki bir PESQ yazılımıdır ve P.862'ye kıyasla daha yeni ve karmaşıktır.

2.4 Veri Gömme

2.4.1 Steganografi

Steganografi, eski Yunancada "gizlenmiş yazı" anlamına gelmektedir ve bilgiyi gizleme bilimine verilen addır. Şifreleme bilimi anlamına gelen kriptografiden farklıdır ve şifrelemeye göre en büyük avantajı bilgiyi gören bir kimsenin gördüğü şeyin içinde önemli bir bilgi olduğunu fark edemiyor olmasıdır. Öte yandan şifreli bir mesaj, çözmesi zor olsa bile, gizemi dolayısıyla çoğu zaman tüm ilgiyi üzerine toplayabilmektedir.

Literatürde steganografi ile en fazla mahkûmlar problemi özdeşleştirilir. Simmons (1983) tarafından literatüre kazandırılan problemde, iki mahkûm Alice ve Bob kendilerini gözetleyen, aralarındaki her türlü mesajlaşmayı etkileyebilen gardiyan Wendy gözetiminde birbirleriyle gizlice haberleşmeye çalışırlar. Wendy, Alice ve Bob arasındaki tüm mesajları gözetleyebilir, inceleyebilir ve hatta dilerse değiştirebilir.

Elektronik ortamda gerçekleştirilen sayısal steganografi uygulamalarında gizli veri aktarımı, taşıyıcı sinyal üzerinde uygulanan steganografik kodlama ve kod çözme işlemleri ile gerçekleştirilir. Özellikle de video, görüntü ve ses gibi medya dosyaları büyük boyutları, yüksek oranda bilgi içermeleri sayesinde gizli verinin saklanabilmesi için uygun koşullar sunarlar.

Steganografide gizli verinin aktarıldığı kanala gizli veya örtülü kanal (*covert-channel*), bahse konu örtülü kanalın içerisine saklandığı herkese açık kanala taşıyıcı kanal veya örtücü kanal (*cover-channel*), gizli veri eklendikten sonra içeriği güncellenen taşıyıcı kanala ise stego kanal (*stego-channel*) denir. Kriptolojide kriptolama algoritmaları nasıl gizli kabul edilmeyip haberleşme güvenliğinin tüm sorumluluğu kripto anahtarlarına yükleniyorsa, steganografide de veri gizleme yönteminin gizli olduğu varsayılamaz. Bu yüzden veri gizleme yöntemlerinde de gizlilik bazı anahtarlarca, stego-anahtarlarla sağlanır. Bu tez genel olarak steganografi ve damgalama ile ilgili olduğundan, kullanılan anahtarlardan "stego-anahtar" olarak değil sadece "anahtar" olarak bahsedilecektir.

Kriptolu verilerin şifrelerinin kırılmasını konu alan bilime kriptoanaliz (*cryptanalysis*) olarak tanımlanmasına paralel olarak, steganografik metotlarca gizlenen verileri ortaya çıkartma bilimine de steganaliz denir. Steganaliz konusu ilerleyen bölümlerde detaylandırılacaktır.

2.4.2 Damgalama

Günümüzde konuşma, müzik, görüntü ve video gibi veriler kolay dağıtılabilme, üretilme ve değiştirilebilme imkânlarının sağladığı avantajlar nedeniyle yaygın olarak sayısal biçimdedirler. Sayısal biçimdeki bilgilerin üretiminin, dağıtımının ve değiştirebilmesinin kolaylığı kötü amaçlı kişilerce suiistimal edilebilmektedir.

Sayısal bilginin kriptolanması, sadece bilgiye olan erişimi çözülme işlemine kadar sınırlandırır. Bilginin açık halinin kötü amaçlı kullanımlardan uzak tutulabilmesi için

daha farklı çözümlere ihtiyaç duyulur. Damgalama, uygunsuz kullanımı engellenmek istenen bir bilginin, güvenlik sağlayan bir başka bilgiyle (damgayla) çeşitli teknikler kullanılarak birleştirilmesidir. Bahse konu damga, sahiplik haklarını temsil eden veya veri bütünlüğünü garanti edebilen bir tür sayısal bilgidir (Cox vd. 2008). Damgalama yapılırken yerine getirebilmesi gereken çeşitli kıstaslar bulunmaktadır:

a) Damga damgalanacak bilgiye gömülmelidir. Damga taşıyıcı bilgiye kodlanmış olmalıdır, bir başlıkta veya son bölümde yer almamalıdır.
b) Algılanabilir olmamalıdır. Eğer mümkünse yetkisiz kimseler tarafından istatistiksel yöntemlerle varlığı tespit edilememelidir.
c) Damgalama kriptolojik anahtarların kullanımı ile güvenli hale getirilmelidir.
d) Yetkisiz kimseler tarafından kaldırılamamalı, bazı saldırılar yapılarak bozulabildiğinde ise damgalanmış veri işe yaramayacak hale gelmelidir.
e) Damga kestirimi istatistiksel olarak güvenli olmalı, damgalanan veya damgalanmayan veriler içerisindeki damgaların varlığı veya yokluğu büyük bir doğrulukla tespit edilebilmelidir.

2.4.3 Steganografi – damgalama

Literatürde pek çok çalışmada, damgalama ve steganografi kavramları benzerlikleri nedeniyle birbirleriyle karıştırılarak kullanılmaktadır (Genellikle daha çok damgalama steganografi manasında kullanılmaktadır). Aslında söz konusu bu iki kavram tanımları ve gereksinimleri itibarıyla birbirlerinden oldukça farklıdır. İlk olarak birbirleriyle karıştırılmalarına yol açan benzerlikler ele alınırsa, her ikisi de veri gizlemenin alt-branşlarıdır. Hem damgalamada hem de steganografide kullanıcıya ait bir veri (gizli veri veya damga) taşıyıcı bir başka verinin içerisine gömülmektedir.

Farklıklar ele alındığında ise, en önemli farklılık gizlenen verinin kullanım amacındadır. Steganografide gömülen veri taşıyıcı veriden tamamen bağımsız, gizli kanal kullanıcılarının mesajlarından oluşur. Damgalamada ise gizlenen veri ya telif haklarına dair bir bilgidir (bir nevi kimlik bilgisi) ya da bütünlük kontrolü sağlayan bir bilgidir ki

her iki tipteki damgalama bilgisi de çoğu pratik uygulamada içine gömülecekleri taşıyıcı veri ile bir şekilde alaka içindedirler.

Steganografide en önemli kıstas gizlenen verinin algılanamazlığı ve tespit edilemezliğidir. Algılanamazlık kavramı daha çok insan duyu organlarını, fark edilmezlik ise matematiksel analizleri kastetmektedir. Damgalama da tıpkı steganografi gibi algılanamaz olmalıdır, ama matematiksel yöntemlere karşı fark edilmezlik olmazsa olmaz bir unsur değildir. Bir damgalama yöntemi için damganın çıkartılamaması yani dayanıklılığı (*robustness*) fark edilmezliğinden çok daha kritik bir öneme sahiptir. Diğer yandan steganografik bir sistem için de dayanıklılık daha az önemlidir.

Yukarıdaki paragraflarda aktarılan yorumlara ek olarak Çizelge 2.1'de steganografik ve damgalama yöntemlerinin benzerliklerinin ve farklılıklarının Wang ve Wang (2004) tarafından derlenmiş bir özeti sunulmuştur:

Çizelge 2.1 Damgalama vs steganografi

	İhtiyaçlar	**Damgalama**		**Steganografi**
		Özel	**Açık**	
Amaç	Entelektüel hakların korunumu	++++		–
	Şüpheye mahal vermeden gizli veri iletimi	–		++++
Şartlar	İnsan duyuları açısından görünmezlik	++++		+++++
	İstatistikî ve algoritmasal görünmezlik	+		+++++
	Yıkıma, hileye ve çıkarıma dayanıklılık	+++++		–
	Normal sinyal işlemeye karşı dayanıklılık	++++		+
	Sıkıştırmaya karşı dayanıklılık	++++		++
	Yüksek gömme kapasitesi	++		++++
Kestirim	Kör kestirim	–	++++	++++
	Orijinal taşıyıcı ile kestirim	++++	–	–
	Az işlem gücü ile kestirim	++		+++
Çok önemli: +++++ Gerekli: ++++ Önemli: +++ İstenir: ++ İşe yarar: + İlgisiz veya gereksiz: –				

Genel bir veri gömme sisteminin modelinin tanıtılmasından evvel, tez kapsamında oluşabilecek kavram kargaşasına engel olmak için ek bir terminolojinin tanımlaması gerekmiştir. Tez içerisinde sıklıkla geçmekte olan ses kelimesi duruma göre hem müziği hem de konuşmayı belirtebilmektedir. Damgalama terminolojisinde saldırı (atak), damgayı bozma amacı taşıyan veya taşımayan, ancak sonucu itibarıyla damgalanmış sinyali bir ölçüde değiştiren, dolayısıyla da damgayı etkileyen, herhangi bir eylem olarak tarif edilmektedir. Steganografi terminolojisinde ise atak kavramı, gizli verinin ifşasını hedefleyen her türlü yöntemi kastetmektedir.

2.4.4 Ses sinyali üzerinde çalışan genel veri gömme – kestirme sistem modeli

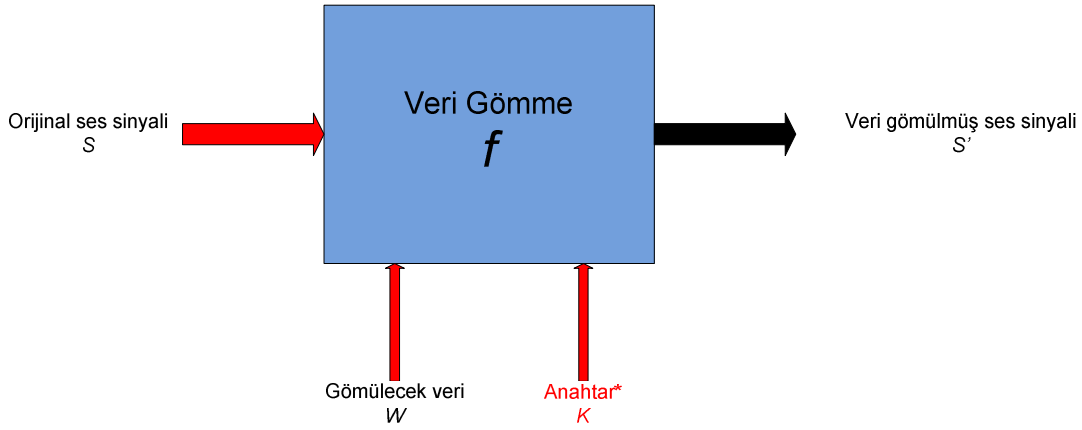
Birazdan görsel olarak anlatılacağı üzere, genel bir veri gömme sisteminde veri gömme işlemi, kullanıcıya ait verinin taşıyıcı veriye, üstelik de taşıyıcı verinin bazı kısımlarının değiştirilmesi pahasına katılmasıdır. Veri kestirim işlemi de katılan verinin en doğru şekilde taşıyıcı sinyalden yeniden kazanılmasıdır.

Şekil 2.7'de genel bir veri gömme sisteminin blok şeması verilmiştir (Cox vd. 2008). (2.5)'te matematiksel bir ifadesi sunulan veri gömme işleminde, damga veya gizli veri genellikle gizli bir anahtar kullanarak *f* gömme fonksiyonu vasıtasıyla taşıyıcı sinyale gömülür. Damgalamada duyulmazlığın, steganografide hem duyulmazlığın hem de istatistikî ve steganografik fark edilmezliğin sağlaması için gömülen sinyalin enerjisi orijinal sinyale göre ayarlanır.

S: Orijinal ses *S'* : Damgalanmış ses

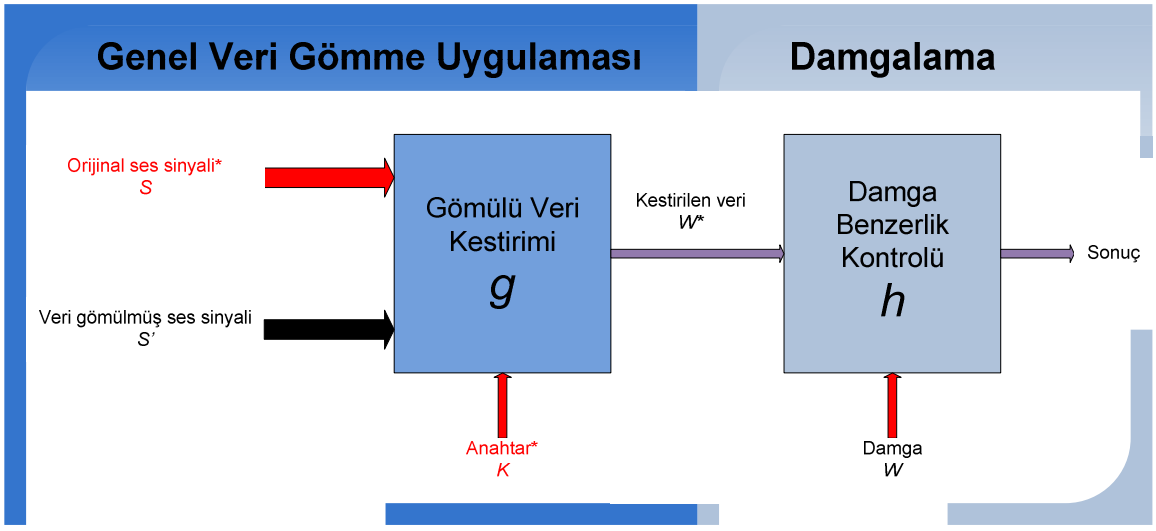
f: Damgalama fonksiyonu *W*: Damga *K*: Opsiyonel gizli anahtar

$$\boldsymbol{S' = S + f(S, W, K)} \tag{2.5}$$



Şekil 2.7 Genel veri gömme sistemi

Şekil 2.8'de blok şeması verilen genel veri kestirim sisteminde, kestirim işlemi *g* fonksiyonu vasıtasıyla gerçekleştirilir ve kestirim işlemi sonucunda tahmini veri, *W** elde edilir. Kimi veri gömme sisteminde gömme ve kestirme için herhangi bir anahtar kullanılmazken, çoğu sistemde gömme ve kestirim için aynı (ortak) anahtardan istifade edilir, bazı sistemlerde ise gömme ve kestirim amaçlı anahtarlar birbirlerinden farklı kılınabilir.



Şekil 2.8 Genel veri kestirim sistemi

Damgalama sistemlerine özgü olarak, elde edilen tahmini veri W^* h fonksiyonu vasıtasıyla, kontrol amaçlı bir damga W ile karşılaştırılır. Denklem (2.6)'da verilen formülle bulunan değer (2.7)'deki bir eşik değeri, E ile kıyaslanarak damga kestirimi tamamlanır. (2.6)'da verilen *. işlemi iki vektörün skalar çarpımını belirtmektedir.

Benzerlik Fonksiyonu:

$$Sim(W, W^*) = W^* *. W / \sqrt{W^* *. W^*} \tag{2.6}$$

Damganın olup olmadığına karar verilmesi:

$$Sim(W, W^*) > E \tag{2.7}$$

Yine sadece damgalama sistemlerine özgü olarak, kimi yöntemlerde orijinal sinyale gereksinim duyulurken kimisinde orijinal sinyale ihtiyaç duyulmamaktadır. Orijinal sinyale ihtiyaç duyulmayan damga kestirimine "kör damga kestirimi" adı verilmektedir. Performans açısından bir karşılaştırma yapıldığında orijinal sinyalden yararlanılarak yapılan damga kestirimlerinde kör kestirimlere kıyasla daha az hata oluşmakta, dolayısıyla daha az yanlış alarm verilmektedir. Normal şartlar altında steganografi uygulamalarında gömülü veri kestirimi için orijinal sinyal kullanılmamalıdır, yani başka bir deyişle steganografik metotlar gizli verileri olabildiğince kör kestirim ile elde etmelidir.

Literatürde bugüne kadar veri gömmek için sayısız yöntem geliştirilmiştir. Söz konusu yöntemlerin birbirleriyle karşılaştırılabilmelerini mümkün kılmak için bazı performans kriterleri tanımlanmıştır (Cox vd. 2008). Performans kriterlerinin en önemli olanları veri gömme kapasitesi, veri gömme verimliliği, taşıyıcı sinyaldeki bozulma ve gömülü verinin kestirimindeki başarı oranlarıdır.

Veri gömme kapasitesi gömülen verinin miktarını belirtir; bit/saniye, gizli veri miktarı/ taşıyıcı veri miktarı gibi değişik birimlerle ifade edilebilir. Veri gömme verimliliği ise taşıyıcı sinyalde birim değişiklik başına kaç birim gizli veri gömüldüğünü bildirir. Taşıyıcı sinyalde oluşan bozulma, gizli veri gömme işleminin sinyal kalitesini ne kadar

olumsuz etkilediğini işaret eder; pek çok farklı kalite ölçüm metriği cinsinden yazılabilir. Son olarak, gömülü verinin kestirimindeki başarı oranı adından da anlaşılacağı üzere gömülü bitlerin hangi oranda doğru elde edildiğini gösterir.

2.4.5 Veri gömme tekniklerinin sınıflandırılması

Veri gömme teknikleri sahip oldukları özelliklere göre sınıflandırılabilirler. Çizelge 2.2'de, Lee ve Jung (2001) tarafından veri gömme yöntemleri için hazırlanan bir sınıflandırma sistematiği sunulmuştur. Lee ve Jung veri gömme yöntemlerini sağlamlıklarına (*gürbüzlüklerine*), gömülen veri dizilerinin tiplerine, gömme sırasında kullandıkları işleme yöntemlerine ve gömülü verilerinin kestirimi için gerekli olan bilgilere göre çeşitli sınıflara ayırmışlardır.

Çizelge 2.2 Veri gömme sınıfları

Veri gömme özelliği	**Sınıflar**
Gömülen verinin sağlamlığı	*Sağlam, yarı kırılgan, kırılgan*
Gömülen veri	*Sözde rastgele dizi, Gauss dizileri, kaotik diziler, formatlanmış bilgi (görüntü, etiket)*
İşleme yöntemi	*Zaman bölgesi, frekans bölgesi, hem zaman hem de frekans bölgesi*
Verinin kestirimi için gerekli bilgiler	*Özel (private) damgalama, yarı özel (semi-private) damgalama, açık (public) veri gömme, açık anahtarlı damgalama (Public key watermarking)*

2.4.5.1 Gömülen verinin sağlamlığına göre sınıflandırma

Sağlam veri gömme (Sağlam damgalama):

Damgalamada en çok ihtiyaç duyulan veri gömme sınıfıdır. Taşıyıcı veriye gömülmüş damga verisi, bilgi üzerinde her türlü değişikliğe ve işleme rağmen (Bilgi kalitesi kabul edilebilir olduğu sürece) varlığını sürdürebilir. Çoğunlukla telif hakkı kontrolü için kullanılır. Gömme kapasiteleri kırılgan ve yarı kırılgan damgalama yöntemlerinin

sağladıklarına göre daha düşüktür. Gizli verinin varlığının saklanması çok zor olduğundan steganografide fazlaca uygulama alanı bulamaz.

Yarı kırılgan veri gömme (Yarı kırılgan damgalama):
Yarı kırılgan veri gömme yöntemleri taşıyıcı verinin sınırlı oranda derece değiştirilmesine tolerans gösterebilir. Genellikle kodlayıcı / sıkıştırıcı bulunan sistemlerde iletilen verinin bütünlüğünün kontrolü için kullanılır. Steganografik veri iletiminde de tercih edilebilir.

Kırılgan veri gömme (Kırılgan damgalama):
Kırılgan veri gömme kolaylıkla bozulup ortadan kaldırılabilir. Steganografik veri iletiminde veya taşıyıcı verinin bütünlüğünün kontrolünde kullanılır.

2.4.5.2 Gizli verinin bulunması için gereken bilgilere göre sınıflandırma

Özel damgalama:
Orijinal sese, gömülen damgaya ve eğer kullanılmışsa gizli anahtara ihtiyaç duyulur. Steganografide kullanılamaz.

Yarı özel damgalama:
Gömülen damgaya ve eğer kullanılmışsa gizli anahtara ihtiyaç duyulur. Steganografide kullanılamaz.

Açık veri gömme (Açık damgalama):
Eğer kullanılmışsa sadece gizli anahtara ihtiyaç duyulur. Steganografik veri iletimine uygundur.

Açık anahtarlı damgalama:
Sadece damgalama uygulamalarında kullanılabilir. Damga gömme işlemi gizli bir anahtar ile, kestirim işlemi herkese açık anahtar ile gerçekleştirilir. Bu sayede sadece

açık anahtarı bilen kimselerin damgayı taşıyıcı sinyalden sökebilmeleri engellenmiş olur (Furon vd. 2000). Orijinal sese ve gömülen damgaya ihtiyaç duyulmaz.

2.4.5.3 Taşıyıcı sinyalden parazit kapmaya göre sınıflandırma

Veri gömme yöntemleri, gömdükleri verilerin taşıyıcı sinyallerden parazit kapıp kapmamasına göre iki kategoriye ayrılırlar.

Taşıyıcı sinyalden parazit kapan veri gömme:
Gömülen verilerin taşıyıcı sinyalden parazit kaptığı yöntemler. Ör: Dağılmış spektrum veri gömme yöntemleri.

Taşıyıcı sinyalden parazit kapmayan veri gömme:
Gömülen verilerin taşıyıcı sinyalden parazit kapmadığı yöntemler. Ör: QIM veri gömme yöntemleri.

2.4.6 Veri bütünlüğünün sağlanması

Veri bütünlüğü mesaj doğrulama kodları vasıtasıyla sağlanır. Mesaj doğrulama kodu (*MAC: message authentication code*), aynı zamanda kriptolojik sağlama toplamı olarak da isimlendirilir. Mesaj doğrulama kodu, bütünlüğü garanti edilecek veri bloğuna ait özet (*hash*) bilgisinin simetrik veya asimetrik şifrelemeden geçirilmesiyle üretilir.

Simetrik şifreleme kullanım senaryosunda, mesaj doğrulama kodlarının üretimi ve kontrolü için gizli bir anahtar kullanılır. Asimetrik şifreleme kullanım senaryosunda ise mesaj doğrulama kodu gizli özel bir anahtarla üretilir, mesaj doğrulama kodunun kontrolü herkese açık anahtarla yapılır. Herkese açık anahtarla, gizli özel anahtar arasında şifreleme algoritmasından kaynaklanan matematiksel bir ilişki vardır; bir anahtardan diğerini elde edebilmek matematiksel olarak ispatlanmış bir işlem yükünü göze almayı gerektirir, yani pratikte çok güçtür.

Simetrik şifrelemede kullanılan gizli anahtarların anahtar boyu, genelde asimetrik şifrelemede kullanılanlara göre oldukça küçüktür. Simetrik şifreleme algoritmaları hızlı, asimetrik şifreleme algoritmaları ise yavaştır. Simetrik şifreleme kullanılarak iletişim gerçekleştirmek için hem alıcıda hem de vericide aynı gizli anahtarın yüklenmiş olması gerekir. Simetrik şifreleme algoritmaları veri şifrelemek için tercih edilirlerken, asimetrik şifreleme anahtarları anahtar değişimi (açık anahtarla kriptolama, gizli anahtarla çözme), sayısal imza (gizli anahtarla kriptolama/imzalama, açık anahtarla çözme/imza kontrol etme) gibi uygulamalarda tercih edilirler (Stallings 2003).

Veri bütünlüğünü sağlamaya yönelik yöntemler bütünlüğü sağlanacak verinin gizliliğinin sağlanması durumunda daha fazla çeşitlilik kazanmaktadır. Ancak veri gizliliğinin sağlanması koşuluyla uygulanabilecek olan yöntemler steganografik uygulamalara elverişli olmadıklarından kapsam dışında tutulmuşlardır. Çizelge 2.3'te steganografik – damgalama uygulamalarına müsait olan MAC alternatifleri listelenmiştir:

Çizelge 2.3 Veri bütünlüğünün sağlanması

İmza Tipleri	$M \parallel E_K[H(M)]$	Mesaj ve mesajın simetrik şifrelenmiş özeti
	$M \parallel E_{KRa}[H(M)]$	Mesaj ve mesajın imzalanmış özeti
	$M \parallel H(M \parallel S)$	Mesaj ve mesajın bir gizli veri ile birleştirildikten sonraki özeti
Tanımlar	M	Bütünlüğü sağlanacak veri bloku
	$E_K[]$	K anahtarı ile simetrik şifreleme
	$E_{KRa}[]$	KRa anahtarı ile asimetrik şifreleme, imzalama
	H	Herkes tarafından bilinen özetleme fonksiyonu, SHA 1-256-384-512, MD5, MD4
	S	Hem mesaj kontrol kodunu oluşturan, hem de mesaj kontrol kodunu kontrol edecek olanın simetrik şifreleme anahtarı gibi paylaştığı gizli bir veri

Not: Veri bütünlüğünü sağlayacak olan anahtarlarla, steganografi amacıyla örneğin gizli verinin yerleştirileceği bitlerin seçiminde kullanılan anahtarlar birbirlerinden tamamen bağımsızdır.

Bütünlüğü garanti edilen veri bloğunun içindeki en küçük bir değişiklik MAC kontrolü sırasında ortaya çıkar. Bit hatası içeren sistemlerde veri bütünlüğü kontrolü yapmak çoğu zaman mantıklı değildir, mutlaka yapılması gerekiyorsa bazı özel yöntemler geliştirilmelidir. Bit hatasının var olduğu sistemlerde, ses verisinin bütünlüğünün sağlanması için ses sinyalinin 1 ve 0'lardan oluşan dalga biçimini ifade eden sayı dizilerinin değil, bit hatalarına rağmen bozulması beklenmeyen, sesin içeriğini / karakterini yansıtan değerlerin bütünlükleri kontrol edilmelidir (Yuam ve Huss 2004).

2.5 Literatürden Seçilmiş Çeşitli Veri Gömme Yöntemleri

2.5.1 Veri gömmede konuşma – müzik sinyalleri açısından farklılıklar

Konuşma ve müzik sinyalleri üzerlerinde yapılan veri gömme yöntemleri birbirlerine benzemekle birlikte aralarında bazı temel farklar bulunmaktadır. En önemli fark, müzik sinyalinin bant genişliğinin konuşma sinyalinin bant genişliğinden fazla olmasıdır. Müzik üzerinde yapılan damgalamada gömülen veri daha geniş bir banda yayılabilmektedir. İkinci fark kabul edilebilir SNR oranlarındadır; kaliteli müzik için SNR değerinin mutlaka çok yüksek olması gerekirken, konuşma sinyali için SNR değerinin çok yüksek olması zaruri değildir.

Müzik sinyallerinin sayısal olarak iletilmesindeki temel amaç kaynaktan hedefe kadar ses sinyalinin mümkün olduğunca kayıpsız ve orijinale en yakın şekilde iletilmesidir. Konuşma sinyallerinde ise öncelik anlaşılırlığın korunmasıdır. Bundan dolayı müzik sinyallerini sıkıştırmayı hedefleyen algoritmalar ses kalitesini korumak zorunda olduklarından yüksek SNR'ye sahiptirler. Konuşma sinyallerini aktaran sistemler ise daha düşük SNR değerlerinde işlev görebilirler.

Müzik sinyallerinin olabildiğince yüksek kalitede tutulması ihtiyacının bir sonucu olarak, gömülen veri damgalama uygulamaları için duyulamaz, steganografi uygulamaları için hem duyulamaz hem de istatistikî fark edilemez olmalıdır, bu koşul da ancak damga enerjisinin düşük tutulması ile yerine getirilebilir. Müzik sinyallerini sıkıştıran algoritmalar gizli veri barındıran sesi mümkün olduğunca kayıpsız bir şekilde sıkıştırırken, farkında olmadan gömülü veriyi de kayıpsız halde taşımaya çalışırlar. Konuşma sinyalleri alanında ise ses kodlayıcıları daha düşük SNR'ye sahiptirler ve kodlama işlemini girdi sinyali büyük oranda değiştirerek yerine getiriler. Gizli verinin söz konusu yıkıcı sıkıştırmaya dayanıklı olabilmesi için ya kodlamaya özel olarak uyarlanması ya da yüksek enerjiyle taşıyıcı sinyale katılması lazımdır.

Üçüncü önemli fark ise, veri gömme açısından belki de en önemli fark, gömülü verinin hangi amaçla kullanıldığıdır. Müzik sinyalleri üzerinde yapılan veri gömme çalışmalarının büyük kısmı damgalama ve telif haklarını korumaya yönelik olurken, konuşma sinyallerinin telif hakkının korunması genel olarak saçma bir düşüncedir. Bu konuda bazı istisnalar olabilir: Son zamanlarda, *stand-up show* tarzı gösteriler CD'lere kaydedilip satışa sunulmaktadır, büyük ihtimalle müzik parçalarına telif hakkı amacıyla geliştirilen damgalama yöntemleri sadece insan konuşmasının bulunduğu bu CD'lerde işe yaramayacaktır, damgalama yapılsa da etkisiz hale getirilmesi daha kolay olabilecektir. Ama yine de belirtilen istisnai durum dışında, konuşma sinyallerinde kullanılacak olan veri gömme uygulamaları telif hakkı korumaktan çok içerik doğrulama ve steganografik iletişim ihtiyaçlarını karşılamaya yönelik olacaktır.

2.5.2 Önemsiz bit kodlama

Önemsiz bit kodlama, gizli verinin (damganın) taşıyıcı sinyalin en önemsiz bitlerine (EÖB) yerleştirildiği bir veri gömme yöntemidir – veri gömme yöntem ailesidir (Bender vd. 1996). Gizli veri gömme işlemi, doğrudan taşıyıcı sinyal üzerinde gerçekleştirilebileceği gibi, taşıyıcı sinyalin çeşitli dönüşümleri (örneğin frekans bölgesindeki dönüşümü) üzerinde de yerine getirilebilir. İdeal kanal kapasitesi örnekleme frekansı ile doğru orantılıdır, ancak steganografi uygulamalarında bu kapasitenin daha azından faydalanılır. Taşıyıcı kanalda oluşan gürültüden etkilenir,

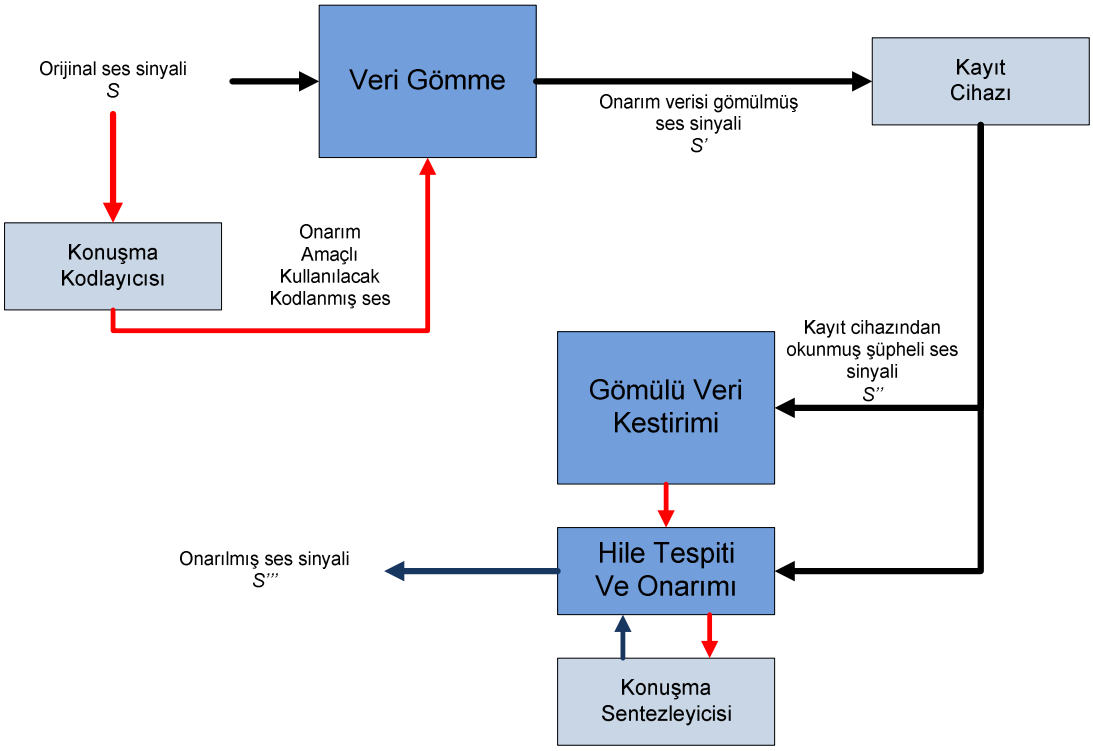
tekrardan örneklenmeye ..vb basit saldırılara karşı dayanıksızdır. Kolaylıkla bozulabildiği için telif hakkı uygulamalarında kullanılamaz. Steganografi ve veri bütünlüğü uygulamaları için oldukça uygundur.

Ses kodlayıcılarının bulunduğu sistemlerde de önemsiz bit kodlama kullanılabilir. Genellikle 128 Kbps olan, 16 bit 8 kHz'de örneklenmiş PCM ses sinyali kodlandıktan sonra hızı düşürülür. Kodlanmış sinyalin sesin kalitesi açısından en az önemli alanları gizli veri bitlerinin gömülmesi için kullanılabilir. Örneğin GSM 6.10 kodlayıcısında RPE düzenli darbe uyarım alanlarının en önemsiz bitleri içerisinde gizli veri bitleri iletilebilir (Yuan ve Huss 2004).

2.5.3 Ses içeriğini geri kazanabilen önemsiz bit kodlamalı damgalama

Genel blok şeması Şekil 2.9'da verilmiş ve Liu vd. (2004) önerisi olan yöntemde damgalama G.723.1 (Anonymous 1996) ses kodlayıcısı modifiye edilerek gerçekleştirilir; gizli veri bitleri G.723.1 kodlanmış çıktısında yer alan uyarım darbe pozisyon indekslerinin (*excitation pulse position indices*) en önemsiz bitlerine gömülür. Damga hem veri bütünlüğü ve hem de bozulan alanları bir ölçüde onarabilecek onarım verisi bilgilerinden oluşur. Damgaya bakılarak sinyal içerisinde olan bozulmanın maksatlı mı yoksa maksatsız mı gerçekleştirildiği anlaşılabilir. Kırılgan damga ile beraber gömülen onarım bilgileri sayesinde maksatlı bir şekilde değiştirilmiş olsa da belli bir dereceye kadar konuşma içeriğinin tekrar kazanılması sağlanır.

Veri bütünlüğünü alanları sayesinde gömülü damganın güvenilirliğini garanti edilebilir. Söz konusu alanlar için damgalamaya uygun MAC alternatiflerinden herhangi birisi tercih edilebilir. Mesela, kullanılan özetleme fonksiyonu gizli anahtarlı olabilir. Veyahut anahtarsız bir özetleme fonksiyonu şifrelemeyle (simetrik veya asimetrik) bir arada kullanılabilir. Bu teknikteki kırılgan damgalamanın başarılı bir şekilde çalışması için kodlanmış veride bit hatası bulunmaması veya bit hatasına karşı önlem alınması gereklidir.



Şekil 2.9 Ses içeriğini kazanabilen önemsiz bit kodlamalı damgalama sistemi

Gömülen damga, aslında isimsiz – standart dışı bir konuşma kodlayıcısı çıktısıdır; temelde LSF (L), perde (P) ve onarım (R) bilgilerinden oluşmaktadır. Damgalanacak ses çerçevelere bölündükten sonra, bölünen çerçeveler sonuncusu hariç F adet çerçeveden oluşan gruplara ayrılır. Her gruptaki her bir çerçeveye, hesaplanma yöntemleri (2.8), (2.9) ve (2.10) denklemlerinde verilmiş damga parçaları ve T çerçeve öncesine ait onarım verileri gömülür:

Grubun ilk çerçevesi: $$W_{g,1} = H_x\,(R_{gF+1-T},\, G_g,\, L_{g,1},\, P_{g,2}) \quad (2.8)$$

Grubun son çerçevesi: $$W_{g,\mathrm{eof}} = H_x\,(R_{gF+\mathrm{eof}-T},\, W_{g,\mathrm{eof}-1},\, L_{g,\mathrm{eof}},\, \mathrm{mod}(\mathrm{eof},2^b)) \quad (2.9)$$

Grubun diğer çerçeveleri: $$W_{g,f} = H_x\,(R_{gF+f-T},\, W_{g,f-1},\, L_{g,f}, P_{g,f+1}) \quad (2.10)$$

2.5.4 Parametrik modelleme ile damgalama

Gurijala vd. (2002) tarafından önerilen yöntemde konuşma sinyalinin damgalanması işlemi, taşıyıcı konuşma sinyalinin LPC değişkenlerinin belli ölçülerde değişikliğe

uğratılması ile gerçekleştirilir. Damganın varlığının tespit edilebilmesi için orijinal konuşma sinyali kullanılır.

Damganın gömülmesi için önce sayısal konuşma sinyalinin öz-ilinti yöntemi ile LPC değerleri bulunur. Sonra bulunan LPC değerleri ile sayısal analiz süzgeci oluşturulur, konuşma sinyali analiz süzgecinden geçirilerek tortu (hata) sinyali elde edilir. LPC değerleri bir şekilde değiştirilerek (Korelasyon tabanlı damga kestirimi yapılacak şekilde) damga gömülür. Değiştirilen LPC değerlerinden ve saklanan tortudan damgalanmış sinyal elde edilir.

Damganın tespit edilmesi için ilk olarak orijinal sinyalin tortusu damgalanmış sinyalden çıkartılır. Sonra değiştirilmiş LPC değerleri en küçük kare hatası (*LSE: Least square error*) kullanılarak kestirilir. Taşıyıcı sinyalin LPC'lerinin öz-ilintisi ile damgalanmış sinyalin kestirimi yapılmış LPC'lerinin öz-ilintisi karşılaştırılır, damga olup olmadığına karar verilir.

Damganın sağlamlığı gömülen damganın taşıyıcı sinyale olan oranına göre değişir. Yüksek enerjili damgalar sinyal kalitesinde anlaşılır şekilde kaliteyi düşürürlerken, düşük enerjili damgalar basit saldırılar karşısında bile bozulabilirler. Gurijala vd. (2002) göre taşıyıcı sinyalin yüzde biri kadar enerjisi olan damgalar hem fark edilmez hem de yeteri kadar sağlam olabilmektedir. Yapılan gürültü ekleme (SNR= 39.6-33.1dB), mp3 sıkıştırma, bazı parçaların atılması ile gerçekleştirilen saldırılar, seğirme (*jitter*), tekrar nicemleme ve bazı filtreleme işlemlerinde parametrik modelleme ile yapılan damgalama tekniğinin başarılı olduğu gözlemlenmiştir.

Parametrik modelleme ile damgalama yönteminin performansı, Gurijala ve Deller (2003) tarafından tanımlanmış bir sadakat kriterinin dikkate alınmasıyla arttırılabilir. Denklem (2.11)'de orijinal x ve damgalanmış sinyaller $\hat{x}$ arasındaki en büyük mutlak fark δ sadakat kriterini aşamaz. Küme üyelik filtreleme (*SMF : Set membership filtering*) algoritması yordamıyla tanımlı sadakat kriterine bağlı kalan damga vektörleri belirlenebilir; bunlar arasında en fazla dayanıklılık veya görünmezlik sağlayan

seçenekler taşıyıcı sinyale gömülebilir (Deller ve Huang 2002). Denklemde yer alan T sembolü, damganın gömülü olduğu zaman aralığını belirtmektedir.

$$\delta = max_{n \in T} |x[n] - \hat{x}[n]| \tag{2.11}$$

2.5.5 Zaman bölgesinde dağılmış spektrumlu ses damgalama

Wang ve Chai (2003) tarafından önerilen zaman bölgesinde dağılmış spektrumlu ses damgalama yönteminde damga, önce gizli bir anahtarla üretilmiş sözde rastgele diziyle çarpılır, ardından fark edilmemesi için frekans maskeleme ve geçici maskeleme işlemlerine tabi tutulur. Son olarak tüm frekans bandına yayılmış ve maskelenmiş olarak taşıyıcı sinyale eklenir.

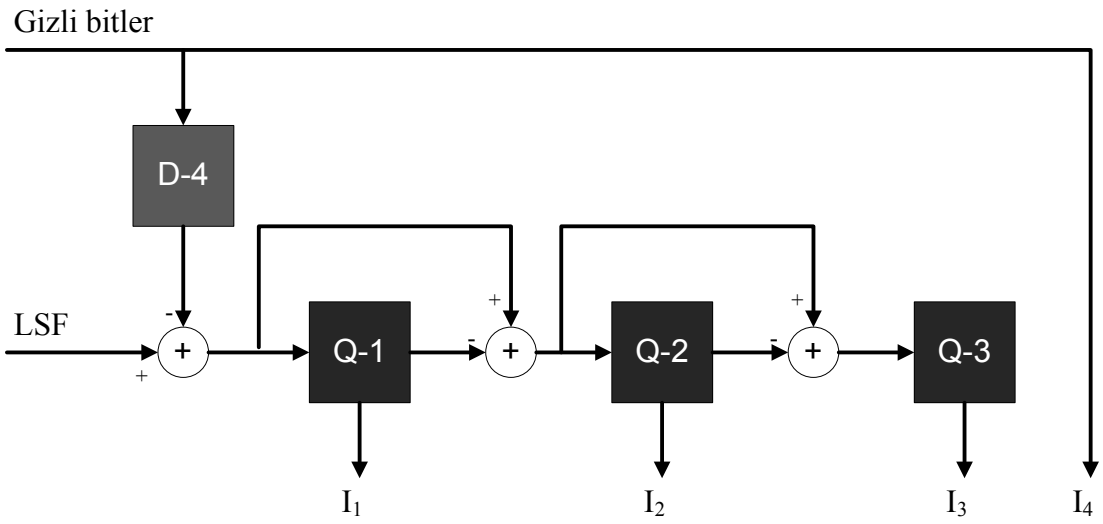
2.5.6 Frekans bölgesinde dağılmış spektrumlu ses damgalama

Frekans bölgesindeki dağılmış spektrum ses damgalama (Malvar 1999), Bölüm 2.5.5'te anlatılan damgalama yönteminin frekans bölgesinde işlev gören bir türevidir. Spektruma dağıtma işlemi yine gizli bir anahtarla üretilmiş sözde rastgele diziler yardımıyla yerine getirilir; bahse konu sözde rastgele diziler çip miktarında üretilir ve psiko-akustik maskeleme de yapılarak enerjileri ayarlanır (Swanson vd. 1998, Kirovski ve Malvar 2003, Wang ve Chai 2003). Daha sonra, damga, enerjisi ayarlanmış bulunan rastgele diziyle çarpılıp, taşıyıcı sinyalin frekans bölgesindeki dönüşümüne eklenir. Damganın tespitindeki başarı düzeyinin arttırılabilmesi için kepstrum filtreleme, *chess* damgalama, tekrarlayıcı blok kodlaması gibi ilave tekniklerden faydalanılabilir. Ayrıca algılanamazlığın arttırılabilmesi için, yüksek ve düşük enerjili kısımları birada barındıran çerçeveler damgalama kapsamı haricinde tutulabilir.

2.5.7 MSVQ'da kıpırtı benzeri veri gizleme

MSVQ'da kıpırtı benzeri veri gizleme yöntemi, Chang ve Yu (2002) tarafından bünyelerinde MSVQ bulunan G.729 (Anonymous 2006) ve/veya MELP (Anonymous

1999) tarzı konuşma kodlayıcıları için geliştirilmiştir. Örnek bir MELP-MSVQ uygulamasının gösterildiği Şekil 2.10'dan da takip edilebileceği üzere, önerilen yönteme göre çok seviyeli nicemleyicinin son seviyesi çalıştırılmamakta, söz konusu son seviyenin üretmesi gereken indeks bitleri yerine doğrudan gizli veri bitleri yerleştirilmektedir. İndeks bitlerinin yerlerine gizli veri bitleri konulması nedeniyle oluşan hatanın bir kısmı daha önceki seviyelerce telafi edilmeye çalışılmaktadır. Bu yüzden gizli veri bitlerinden oluşturulan indeksin gösterdiği vektör nicemlenecek girdi LSF vektöründen çıkartılmakta, elde edilen yeni LSF vektörü işleyişi aynı kalan ilk üç seviyeye sokularak nicemlenmektedir. Yöntemin üzerine dayandığı varsayımlar tartışmaya açıktır.

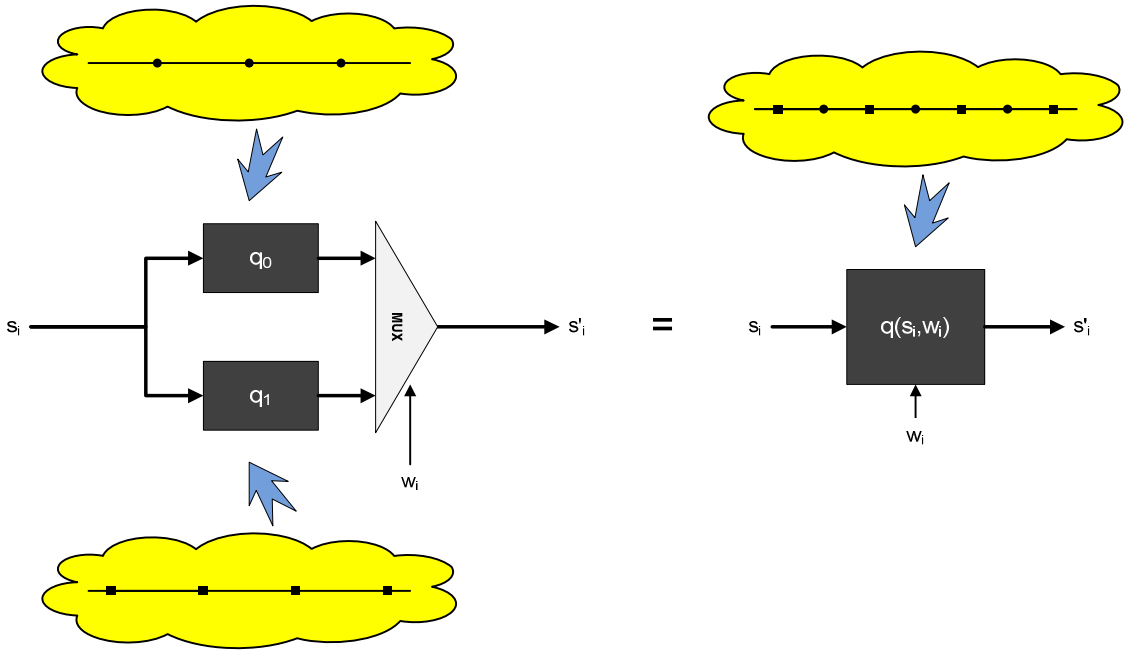


Şekil 2.10 Chang ve Yu (2002) tarafından önerilen MSVQ'da kırpıntı benzeri veri gizleme yöntemi

2.5.8 Nicemleme indeks modülasyonu ile veri gömme

Nicemleme indeks modülasyonu ile veri gömme ilk olarak Chen ve Wornell (2001) tarafından literatüre kazandırılmıştır ve çok sayıda türevi vardır. Damgalama ve steganografi uygulamalarında sıklıkla tercih edilir. En basit QIM yönteminde veri gömme skalar nicemleyici üzerinde yapılır, taşıyıcı sinyalin her örneğine bir adet gizli veri biti gömülür ve gömülen bitler taşıyıcı sinyalden parazit kapmaz. Şekil 2.11'se gösterildiği gibi çalışması son derece basittir; eğer gizli veri biti w_i 0'a eşitse üzerine

gizli veri biti gömülecek s_i örneği q_0 nicemleyicisi ile nicemlenir, eğer 1'e eşitse q_1 nicemleyicisi devreye sokulur. Gizli veri gömme sırasında oluşan hata, nicemleme hatası kadar olur; nicemleme basamaklarının arasının açılıp-kapanmasıyla, başka bir deyişle kullanılan nicemleme basamaklarının sayısının azaltılıp-arttırılmasıyla gizli veri bitinin taşıyıcı sinyal içindeki enerjisi arttırılır veya azaltılır. Gizli veri kestiriminde içinde gizli veri biti bulunan örneğin değerine bakılır; eğer örnek q_0 nicemleyicisine ait bir basamağa daha yakınsa 0, q_1 nicemleyicisine ait bir basamağa daha yakınsa 1 değeri kestirilir.



Şekil 2.11 Skalar nicemleyici üzerinde QIM

QIM yönteminde, gizli veri eklenmesi sırasında oluşan nicemleme hatası, detayı (2.12)'de sunulmuş basit bir iyileştirmeyle dizginlenebilir. Söz konusu iyileştirme sonucunda elde edilen yönteme, bozulma telafili QIM (*DC-QIM: distortion compensated QIM*, Chen ve Wornell (2001)) ismi verilmiştir. Öte yandan, bozulma dizginlemesinin bir bedeli olarak DC-QIM tarafından gömülen gizli veride, QIM yönteminde oluşmayan taşıyıcı sinyal kaynaklı parazitlenmeler görülür.

$$s_i'(n) = q(s_i, w_i) + (1-\alpha)[s_i - q(s_i, w_i)]$$

$$s_i'(n) = \begin{cases} q_0(s) + (1-\alpha)[s_i - q_0(s_i)] & \text{eğer} \quad w_i = 0 \\ \text{ya da} & \\ q_1(s) + (1-\alpha)[s_i - q_1(s_i)] & \text{eğer} \quad w_i = 1 \end{cases} \tag{2.12}$$

Daha önceden de belirtildiği üzere QIM yönteminin pek çok türevi mevcuttur: Bir örneğe birden fazla bit gömülebilir, birden fazla örneğe tek bir bit gömülebilir, vektörsel nicemlemede kullanılabilir, kullanılan nicemleyicilerdeki basamaklar üzerinde uyarlamalar yapılabilir, frekans ve dalgacık dönüşümlerinden sonra uygulanabilir, …vb.

QIM yöntemi konuşma kodlayıcılarında daha farklı (daha basit) bir şekilde uygulanabilir. Konuşma sinyalinin aşamalar halinde kodlanması sırasında nicemlenmesi gereken pek çok değer hesaplanmaktadır. Kodlayıcı içi QIM uygulamasında ise üzerine gizli veri gömülecek değerin nicemleme işlemi, gömülecek gizli veri bitlerine göre sınırlandırılır; bu sayede nicemleyici daha az optimum bir indeks seçmeye zorlanır ve nihayetinde seçilen indeks değerinin içinde gizli veri bitlerinin aynen yer alması veya diğer bir deyişle saklanması sağlanır.

2.5.9 Damgalama tekniklerini değerlendirme

Bölüm 2.5.2 – 2.5.8 arasında anlatılan yedi veri gömme yönteminin bu tez için anlam ve önemine kısaca değinmek gerekirse, adına başlık açılan ilk yöntem olan "önemsiz bit kodlama" çoğu steganografi ve veri bütünlüğü uygulamasının selefidir. Bu tezin ilerleyen bölümlerinde işlenen özgün GSM 6.10 veri gömme yöntemi önemsiz bit kodlamadan türetilmiştir. Ondan sonra anlatılan, ses içeriğini geri kazanabilen önemsiz bit kodlamalı damgalama yöntemine gelince, bu çalışma GSM 6.10'da uygulanan gizli veri gömme uygulaması ile birleştirilmiş, bu tez kapsamında önerilen pratik bir telekomünikasyon mühendisliği uygulamasına, konuşma-üzeri-konuşma (*Voice-over-Voice*) uygulamasına, ilham vermiştir.

Parametrik modelleme ile damgalama, MSVQ'da kıpırtı benzeri veri gizleme ve nicemleme indeks modülasyonu ile veri gömme yöntemleri bu tez kapsamında geliştirilmiş olan özgün MELP MSVQ QIM yöntemlerinin literatürdeki en yakın akrabalarıdır. "Parametrik modelleme ile damgalama" konuşma sinyalinin PCM biçiminde faaliyet gösterir; kodlama alanında veri gömmek için tasarlanmamıştır, yine de MELP MSVQ QIM yöntemleri ile benzer bir yaklaşıma sahiptir. Ana fikri LPC değerlerine gizli veri gömülebileceğidir. Kıpırtı benzeri veri gizleme yöntemi ise MELP MSVQ QIM yöntemlerine göre tam anlamıyla aykırı bir yönde ilerleme kaydetmiştir. Maalesef bu çalışma yanıltıcı sonuçlarla yayımlanmış olsa da, ana fikrinin oldukça budanmış bir kısmı faydalı bilgi içermektedir: Gizli veri LSF değerlerinin MSVQ indekslerinin en önemsiz bitlerine gömülebilir. QIM yöntemleri ise apaçık bir şekilde MELP MSVQ QIM yöntemlerinin atasıdır. Her ne kadar QIM yöntemleri dışındaki diğer iki yöntem bu teze doğrudan faydalı katkı sağlamamışlarsa da, adları bilimsel etiğin bir gereği olarak anılmıştır.

Son olarak DS ses damgalama yöntemlerine bir parantez açılırsa, bu yöntemler önemsiz bit kodlama haricinde literatürde en iyi bilinen ve en fazla türevi olan damgalama yöntemleridir. Bazı türevlerinde veri gömme işlemi dönüşüm bölgelerinde gerçekleştirilir. Genel olarak ses sinyalinin PCM biçiminde çalışırlar, steganografi uygulamalarından ziyade damgalama amaçlıdırlar. İşitsel fark edilmezlik ve ataklara karşı dayanıklılık öncelikli hedeflerindendir. Düşük hızda çalışan kodlayıcılara karşı dayanıklı olmaları veya olabilseler bile işe yarar bant genişliği sunmaları çok zordur. Tez çalışmalarında doğrudan kullanılmamışlardır, tüm bunlara rağmen literatürdeki yaygınlıkları dolayısıyla tezde kendilerine yer verilmesi gerektiği düşünülmüştür.

2.6 Steganaliz – Konuşma Sinyalinin Steganalizi

Steganaliz nesnelerin içinde gizli nesnelerin varlığını araştırma, eğer varsa ve mümkünse bu gizli nesneleri ortaya çıkartma sanatı ve bilimidir. Anlaşılmaz kriptolojik verilerin çözümlenmesi bilimi olan kriptoanalizle paralellik gösterir.

Klasik anlamda steganografi, hükümlü problemi ile özdeşleşmiştir. Alice arkadaşı Bob'a açık bir mesajın içerisine saklanmış gizli bir mesaj göndermek istemektedir. Gardiyan Wendy ise Alice ve Bob arasında iletişimi dinlemekte, gizli veri iletimine engel olmaya, gizli veri iletim kanallarını kapatmaya ve eğer varsa olası gizli verileri bozmaya çalışmaktadır. Klasik anlamdaki steganaliz probleminde ise Wendy gizli haberleşmeyi bozmak ve engellemek yerine sadece varlığını tespit etmekle yetinmekte, eğer mümkünse iletilen gizli verileri deşifre etmeye çalışmaktadır.

Steganografik yöntemler kullanılarak saklanan verileri açığa çıkartmak için ortaya konmuş genel bir steganaliz yöntemi bulunmamaktadır. Ancak çoğu steganaliz uygulamasının temeli matematiksel – istatistiksel analizlere dayanır: İçinde gizli veri içeren ve içermeyen taşıyıcı sinyallerin birbirlerinden ayrışan matematiksel – istatistiksel özellikleri tespit edilir, ayrışan istatistiki özellikleri en iyi bölen optimum eşik değerleri bulunur, bulunan eşik değerlerine göre sınıflandırma (steganaliz) işlemleri gerçekleştirilir.

Steganaliz yöntemleri amaçlarına göre iki gruba ayrılırlar:

- Pasif steganaliz: Gizli verinin sadece varlığını tespit eden yöntemler
- Aktif steganaliz: Gizli mesajın bir kısmını veya benzerini elde etmeyi sağlayan yöntemler

Steganaliz metotları, gizli veri içeren örtücü verinin toplandığı bölge üzerinde çalışırlar. Steganaliz metotlarının çalıştığı boyuta göre söz konusu metotlar üç kategoriye ayrıştırılabilir.

- Uzaysal dağılımlı veri üzerinde çalışan metotlar (Resim)
- Zamana dağılmış veri üzerinde çalışan metotlar (Ses)
- Hem uzaysal hem de zamana yayılmış veri üzerinde çalışan metotlar (Video)

Steganografi yöntemleri, steganaliz yöntemlerine karşı koyabilmek için iki iletişim kavramından fazlasıyla yararlanır.

- Baraj gürültüsü
- Kripto gizli veriler

Baraj gürültüsü: Gizli veriyi içeren taşıyıcı sinyalde, gizli veri eklenmesi haricinde oluşan, kayıt, iletim, kodlama ..vb nedenlerden kaynaklanan bozulma. Eğer gizli veri gömme işlemi sırasında oluşan bozulma sinyali baraj gürültüsünün altında kalıyorsa, dahası tüm istatistiksel özellikleri baraj gürültüsü ile aynı, çok yakın veya benzer tutulabiliyorsa gizli verinin varlığının tespit edilmesi (Pasif steganaliz) ve gizli verinin içeriğinin açığa çıkartılması (Aktif steganaliz) çok zorlaşmaktadır.

Kriptolu gizli veriler: Gizli veriler gömülmeden önce şifrelenirler. Gizli verinin şifrelenmesi aktif steganaliz yöntemlerine karşı kesin çözüm sağlar; daha açık bir anlatımla gizli veri gömülü olduğu sinyalden bir şekilde sökülebilse bile içeriğinin elde edilmesi için zorlu kriptoanaliz sürecinin başarıyla sonuçlandırılması gerekir. Bunların dışında, şifreleme işlemi kullanıcı verisinin istatistiksel dağılımını beyaz gürültüye benzetir. Bu sayede şifreli gizli veriler, beyaz gürültü dağılımlı baraj gürültüsü içeren sistemlerde pasif steganaliz yöntemlerine karşı dayanıklılık kazanırlar.

2.6.1 İdeal steganografi ve ideal kriptografi

İdeal Steganografi: Gömülecek gizli verilerin baraj gürültüsünü aşmayacak şekilde yerleştirildiği ve tüm istatistiksel özelliklerinin baraj gürültüsüne benzetildiği steganografi uygulaması. Başlıca zorluğu gizli verinin tüm istatistiksel özelliklerinin baraj gürültüsüne benzetilmesi kısmındadır. "Tüm istatistiksel özellikler" ifadesi belirsizlik içermektedir. Uygulamadaki en büyük dezavantajı bant genişliğinin düşük olmasıdır, ama yine de bazı kodlama teknikleri vasıtasıyla gizli kanal bant genişliği arttırılabilir. Örneğin 64 Kbps'lık bir taşıyıcı kanalda 10^{-3} 'lük bit hata oranlı baraj gürültüsü tanımlanabiliyorsa, baraj gürültüsü gömülecek gizli verileri etkilemiyorsa, söz konusu taşıyıcı kanalda kodlamasız 128 bps'lik, C(7,4) kodlamalı 220 bps'lik ve C(511, 501) kodlamalı 577 bps'lik stego-kanallar oluşturulabilir.

İdeal Kriptografi: Kullanıcı verilerinin gerçek rastgele değerlere göre anlaşılmaz hale getirilmesi. Günümüzde kullanılan kripto algoritmaları aslında çok sayıda durumu (*state*) olan, durumları kripto anahtarlarınca belirlenen devasa durum makineleridir (*state machine*). O kadar çok durumları vardır ki, tüm durumlarının hatta küçücük bir kısmının bile bir yerde saklanması, hesaplanması çok yüksek düzeyde işlem gücü ve depolama maliyeti gerektirir. Yine de bu özelliklerine rağmen asla gerçek rastgele veri kaynakları olarak nitelendirilemezler. İdeal kriptografi ise herhangi bir kök (anahtar) üzerinde algoritma çalıştırılarak rastgele sayı eldesine dayanmaz, kullanılan kriptolojik değerler tamamen doğa gözlemlerinden elde edilir ve gerçek rastgeledir. Uygulamadaki en büyük zorluk, şifrelenecek veri kadar rastgele dizinin haberleşecek uçlara dağıtımda yaşanır. Günümüzde halen kullanımına devam edilmektedir.

2.6.2 Ki - kare steganalizi

Westfeld ve Pfitzmann (1999) resimler içerisine EÖB yöntemlerince gömülen verilerin istatistiksel araçlara ihtiyaç duyulmadan, sadece görsel kontrolle tespit edilebileceğini göstermişlerdir. Görsel kontrolün mümkün olmadığı hallerde ise EÖB steganografisine karşı ki-kare (*Chi-square*) steganalizi olarak da isimlendirilen yöntemden faydalanılabilir: Steganalize alınan şüpheli verinin değer çiftlerinin dağılımı ile teorik dağılım arasında ki-kare testi uygulanır ve iki dağılım arasındaki benzerlik hesaplanır. Hesaplanan benzerlik miktarına göre şüpheli veride steganografik veri bulunma-bulunmama olasılıkları hesaplanır. Böhme ve Westfeld (2004) çalışmalarında, farklı mp3 kodlayıcılarının farklı istatistiksel özelliklerini (farklı ortalama dağılımlarını) tespit etmeye çalışmışlar ve her kodlayıcı için steganaliz işlemlerini özelleştirerek gizli veri varlığı kestirim doğruluğunu maksimize etmişlerdir.

2.6.3 Sinyaldeki bozulmanın dikkate alınması ile yapılan steganaliz

Avcıbaş vd. (2003) görüntü içerisine gömülmüş olan gizli veri mevcudiyetinin, görüntü kalite metriklerine bakılarak tespit edilebileceği fikrini ortaya atmışlardır. Önerilen yöntemde ilk olarak gizli veri içeren ve içermeyen eğitim kümeleri için Minkowsky, maksimum fark, sıralı maksimum fark, Czekanowski, açı ortalaması, çapraz-ilinti,

ağırlıklı spektral uzaklık ..vb görüntü kalite metrikleri ölçülmektedir. Daha sonra ANOVA testleri (Rencher 1995) kullanılarak gizli verinin (en fazla) etkilediği kalite metrikleri belirlenmektedir. Son olarak etkilenen kalite metrik değerleri regresyon analizine sokularak doğrusal sınıflandırıcı tasarlanmaktadır.

2.6.4 Dalgacık alanında steganaliz

Siwei ve Farid (2002) ile Shi vd. (2005) steganaliz için dalgacık alanını tercih etmişlerdir. Her iki çalışmada da dalgacık alanında yapılan istatistiksel analizlerle, resimler içerisinde gizli verilerin mevcudiyetleri araştırılmıştır. Gizli verilerin mevcudiyetleri, dalgacık alanında hesaplanan ortalama, değişinti ve çarpıklık değerleri ile tespit edilmeye çalışılmıştır. Zou vd. (2005) ise dalgacık karakteristik fonksiyonunun istatistiksel momentlerini kullanan steganaliz yöntemi önermişlerdir.

2.6.5 Entropi vasıtasıyla steganaliz

Bugüne kadar geliştirilmiş çoğu steganaliz yönteminin ortak dayanak noktası, taşıyıcı sinyaller ile stego sinyaller arasındaki olası istatistiksel ayrışmalardır. Eğer taşıyıcı sinyal ile gömülen sinyallerin enformasyon entropileri (Shannon entropisi) birbirlerinden farklılık arz ediyorsa, bu durumda gizli veri mevcudiyetinin ifşasında entropi ölçümlerinden yararlanılabilir. Malik vd. (2008) geleneksel QIM yöntemlerince gizlenen verileri entropi ölçümleri gerçekleştirerek tespit etmeye çalışmışlardır.

Steganaliz cephesinden bağımsız olarak MELP ve LPC cephelerinde de entropi ile ilgili türlü çalışmalar mevcuttur. Bu çalışmaların öncelikli hedefi ses kodlayıcılarının kodlama ve bant genişliği verimliliklerini arttırmaktır. Chou ve Lookabaugh (1990) LPC'ler üzerinde entropi kodlaması uygulayarak, kodlanmış çıktılar için ihtiyaç duyulan bant genişliklerini indirgemeyi başarmışlardır.

2.6.6 Kaotik özniteliklere göre steganaliz

Kocal vd. (2009) tarafından önerilen kaotik özniteliklere göre steganaliz yönteminde, gizli verinin varlığı konuşma sinyalinin FNF (hatalı komşuluk oranı = *false neighbor fraction*) ve Lyapunov katsayıları özniteliklerine göre belirlenmektedir.

Bir dizinin iki örneği, $s(n)$ ve $s(m)$, arasındaki ilişkinin hatalı komşuluk olarak nitelendirilebilmesi için D boyutlu faz uzayındaki d_D topolojik uzaklığının (2.13) $D+1$ boyutlu faz uzaydaki d_{D+1} topolojik uzaklığından (2.14) anlamlı ölçüde farklı (2.15) olması gerekmektedir.

$$d_D(s(n), s(m)) = \left(\sum_{k=0}^{D-1} (x(n+kT) - x(m+kT))^2 \right)^{1/2} \tag{2.13}$$

$$d_{D+1}(s(n), s(m)) = \left(d_D(s(n), s(m))^2 + \left(x(n+TD) - x(m+TD) \right)^2 \right)^{1/2} \tag{2.14}$$

$$\Delta = \left(x(n+TD) - x(m+TD) \right)^2 \tag{2.15}$$

Eğer (2.16)'te gösterildiği gibi X_c taşıyıcı dizisine 0 ortalamalı, σ^2değişintili bağımsız özdeşçe dağılmış E gizli veri dizisi eklenirse

$$x_s(n) = x_c(n) + \varepsilon(n) \tag{2.16}$$

Bu durumda topolojik uzaklık farklılığı

$$\Delta_s = \left(x_c(n+TD) + \varepsilon(n+TD) - x_c(m+TD) - \varepsilon(m+TD) \right)^2 \tag{2.17}$$

değerine eşit olur. Δ_s'nin beklenen değeri ise

$$E[\Delta_s] = [x_c(n+TD) - x_c(m+TD)]^2 - 2[x_c(n+TD) - x_c(m+TD)]E[\varepsilon(n+TD) - \varepsilon(m+TD)] + E[\varepsilon(n+TD)^2] + E[\varepsilon(m+TD)^2] - 2E[\varepsilon(n+TD)\varepsilon(m+TD)] \quad (2.18)$$

(2.18)'de yazıldığı gibidir. Öte yandan E gizli veri dizisinin ortalaması 0, değişintisi σ^2 olduğuna göre

$$E[\Delta_s] = \Delta_c + 2\sigma^2 \quad (2.19)$$

değerine eşit olur; bu sonucu yorumlarsak taşıyıcı sinyale bağımsız özdeşçe dağılmış gizli veri dizilerinin eklenmesi, FNF sonuçlarında her zaman artışa sebebiyet verir. Ancak eklenen gizli veri bağımsız özdeşçe dağılmış değilse

$$E[\Delta_s] = \Delta_c + 2\sigma^2 - 2r(n-m) \quad (2.20)$$

denklemine eşit olur; $r(n-m)$, E dizisinin $(n-m)$'nci korelasyonudur; bu durumda taşıyıcı sinyalin FNF değerleri ile gizli veri gömülmüş stego-sinyalin FNF değerleri birbirlerinden daha az ayrışma gösterirler.

Taşıyıcı sinyalin içerisine gömülen gizli veri dizileri, FNF özniteliklerini etkiledikleri gibi, Lyapunov katsayılarını da etkilerler. Her bir boyut için Lyapunov katsayısı (2.21)'de ifade edildiği gibidir:

$$\lambda = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \ln \frac{d(s(n+1), s(m+1))}{d(s(n), s(m))} \quad (2.21)$$

Lyapunov katsayılarının hesaplanmasında kullanılan topolojik uzaklık değerleri açıldığında

$$d(s(n+1), s(m+1))_c = \left(\sum_{k=0}^{D_E-1} (x(n+k+1) - x(m+k+1))^2 \right)^{1/2} \quad (2.22)$$

eşitliği elde edilir; eşitlikteki D_E gömme boyutudur ve söz konusu boyutun daha önceden herhangi bir şekilde bulunmuş olduğu varsayılmıştır. Denklem (2.24)'te stego-sinyalin beklenen Lyapunov katsayısı taşıyıcı sinyalin beklenen Lyapunov katsayısı cinsinden yazılabilir.

$$d(s(n+1), s(m+1))_s = \left(\sum_{k=0}^{D_E-1} \left(x(n+k+1) + \varepsilon(n+k+1) - x(m+k+1) - \varepsilon(m+k+1)\right)^2 \right)^{1/2} \quad (2.23)$$

$$E\left[d(s(n+1), s(m+1))_s{}^2\right] = d(s(n+1), s(m+1))_c{}^2 + 2\sigma^2 D_E \quad (2.24)$$

Yukarıda verilen (2.24) denklemine göre, gizli veri içeren dizilerin ortalama Lyapunov katsayılarının gizli veri içermeyen dizilerin Lyapunov katsayılarından daha fazla olacağı öngörülmektedir.

Kocal vd. (2009) tarafından önerilen kaotik özniteliklere göre steganaliz yönteminin kısa bir özeti sunulacak olursa, gizli veri içeren ve içermeyen dizilerin FNF ve Lyapunov katsayılarının beklenen değerleri birbirlerinden farklı olmalıdır (Söz konusu bu değerler kaotik araçlarca hesaplanmış nümerik özellikler olduğundan kaotik öznitelikler olarak da nitelendirilmektedir). Bu olguya dayanarak gizli veri içeren ve içermeyen diziler çeşitli sınıflandırıcılar vasıtasıyla belli oranlarda birbirlerinden ayrıştırılabilirler.

2.7 Kuramsal Temellerin Tezin Diğer Bölümleriyle Olan İlişkisi

Kuramsal temeller bölümünde, tezin sonraki bölümlerinin anlaşılabilmesi ve takip edilebilmesi için gerekli olan bilimsel altyapı sunulmuştur. Giriş bölümünde de belirtildiği üzere tez birbirleriyle alakalı dört özgün çalışmadan, Bölüm 3'teki MELP-MSVQ-QIM veri gömme, Bölüm 4'teki kodlayıcı tanıma, Bölüm 5'teki Markov zincir modellerine dayalı steganaliz ve istatistiksel olarak iyileştirilmiş matris gömme

yöntemlerinden meydana gelmektedir. Adı geçen dört özgün yöntemin hepsi birden konuşma sinyali ve konuşma kodlayıcılar için geliştirilmişlerdir; kuramsal altyapı da bu doğrultuda ilk olarak konuşma sinyalinin anlatımı ile başlamış, kodlayıcıların inceleme altına alınmasıyla devam etmiştir. Konuşma sinyalinin işlendiği bölümün bir alt başlığı olan 2.1.4'te konuşma sinyalinin kaotik özellikleri nakledilmiştir, söz konusu kaotik özellikler 2.8.6'daki kaotik öznitelik steganalizinin kuramsal dayanağını oluşturmaktadır.

Kodlayıcıların incelendiği Bölüm 2.2'de kodlayıcılar hakkında genel bilgiler verilmiş, Ek 3, Ek 4, Ek 5, Ek 6 ve Ek 7 bölümlerindeyse tez kapsamında yararlanılan kodlayıcıları tanıtılmıştır. Tanıtılan konuşma kodlayıcılarından olan MELP'ten Bölüm 3 ve 4'te, GSM 6.10 FR'den ise Bölüm 4 ve 5 yararlanılmaktadır. Diğer kodlayıcı türleri G.726, G.729 ve GSM 6.60 EFR'ye Bölüm 4'te ihtiyaç duyulmaktadır. Bölüm 2.3'te kodlayıcıların konuşma sinyali üzerindeki etkilerinin ölçümünü mümkün kılan kalite metrikleri ele alınmış, söz konusu kalite metrikleri vasıtasıyla Bölüm 3 ve 4'teki performans testleri yerine getirilmiştir.

Konuşma sinyali ve kodlayıcılarının konu alınmasından hemen sonra bu tezin ismine yakışır bir şekilde 2.4'te veri gömme, damgalama ve steganografi konuları işlenmiştir. 2.5'te konuşma sinyali ve kodlanmış konuşma sinyali üzerinde veri saklayan literatür yöntemleri konu alınmış, bu tezde gerçekleştirilen çalışmalarla olan ilişkileri açıklanmıştır. 2.5.7'deki MSVQ'da kıpırtı benzeri veri gizlemeyle 2.5.8'deki nicemleme indeks modülasyonu ile veri gizleme yöntemlerinden Bölüm 3'teki özgün MELP-MSVQ-QIM yönteminin tasarımında faydalanılmıştır.

Kuramsal temellerdeki Bölüm 2.6'da, steganografik veri gömme yöntemlerinin sınanması için geliştirilen steganaliz yöntemlerine değinilmiştir. Özellikle 2.6.6'da aktarılan kaotik öznitelikler steganalizi tez kapsamındaki tüm özgün yöntemler için kritik öneme sahiptir; Bölüm 3 ve 5'te verilen özgün veri gömme yöntemlerinin steganografik dayanıklılıkları kaotik öznitelikler steganaliziyle ölçülmüş, Bölüm 4'teki kodlayıcı tanıma kaotik öznitelikler steganalizinden ilham alınarak teşekkül ettirilmiştir.

Sonuç olarak Bölüm 2'nin bu en son alt başlığı altında kuramsal temellerin diğer bölümler ile olan ilişkileri vurgulanmıştır. Bir sonraki Bölüm 3'te tez kapsamında geliştirilmiş özgün MELP-MSVQ-QIM yöntemi anlatılacak, kalite metrik ve steganaliz test sonuçları sunulacaktır.

3. MELP-MSVQ-QIM VERİ GÖMME YÖNTEMİ

Sayısal veri gömme, çeşitli amaçlarla kullanılacak olan özel nitelikli sayısal bir bilginin video, görüntü veya ses türündeki taşıyıcı bir sayısal sinyale katılması işlemidir. Söz konusu katma işlemi kodlama tekniklerinin tatbik edilmesiyle yerine getirilir. Tekniğin özelliklerinin uygun olması kaydıyla gömülü bilgi telif hakları, veri bütünlüğü ve gizli veri haberleşmesi amaçlarıyla kullanılabilir. Tezin bu bölümü, özel bir ses sinyali türü olan konuşma sinyali içerisine nicemleme indeks modülasyonu uygulayarak veri saklayan özgün veri gömme yöntemini ve bahse konu veri gömme yönteminin çeşitli varyasyonlarını konu almaktadır.

Bölüm 2.5.1'de de detaylı olarak anlatıldığı üzere, konuşma sinyaliyle müzik (audio) sinyali arasında pek çok farklılık bulunmaktadır. Literatürde gerek konuşma gerekse müzik sinyallerine veri gömen pek çok yöntem önerilmiştir; bu tezin konusuyla en alakalı olanlar 2.5.2 – 2.5.8 bölümlerinde anlatılmıştır. Literatürde yer alan yöntemler genellikle konuşma sinyalinin dalga özelliklerini fazlaca bozmayan kodlama tekniklerinde çalışabilmektedir, ancak MELP gibi sinyalin sadece algısal özelliklerini koruyan düşük hızlı kodlayıcılar bahse konu yöntemlerce gömülen verileri işe yaramayacak ölçüde bozabilmektedir.

MELP kodlanmış konuşma sinyaline gizli veri gömme işlemi, en kolay bizzat MELP kodlayıcısının içerisinde gerçekleştirilebilir. Bu fikri uygulamaya geçiren ilk çalışma Chang ve Yu (2002) tarafından önerilen ve Bölüm 2.5.7'de bahsedilen MSVQ'da kıpırtı benzeri veri gömme yöntemidir. Söz konusu çalışmada, gizli veri MSVQ'nun son indeks bitlerinde taşınmaktadır. Yöntemin çalışma esaslarının dayandığı varsayımlarda tartışmaya açık pek çok nokta bulunmaktadır ve daha önemlisi veri gömme işlemi kodlanmış sinyal üzerinde yüksek oranda bozulmaya yol açmaktadır. MSVQ'da kıpırtı benzeri veri gömme yönteminin iyileştirilmesi gereksiniminin verdiği motivasyonla MELP MSVQ'sunda veri saklayan başka alternatif yöntemler araştırılmış; özgün MELP-MSVQ-QIM yöntemi geliştirilmiştir.

3.1 MSVQ'da Kıpırtı Benzeri Veri Gömme Yönteminin Tekrar Ele Alınması

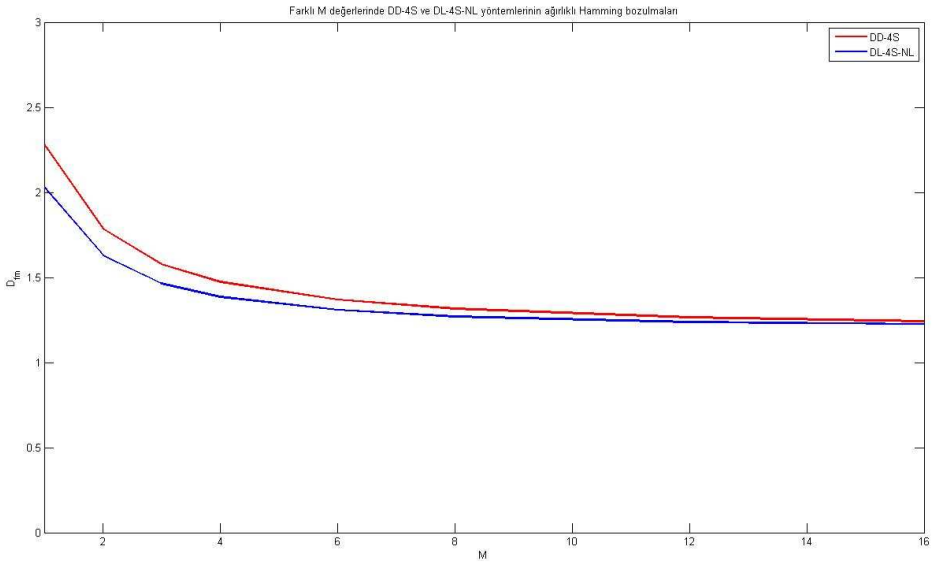
Bu tez içerisinde ilk olarak Bölüm 2.5.7'de tanıtılan MSVQ'da kıpırtı benzeri veri gömme yöntemi (Chang ve Yu 2002), G.729 (Anonymous 2006) ve/veya MELP (Anonymous 1999) gibi bünyelerinde MSVQ barındıran konuşma kodlayıcıları için geliştirilmiştir. Önerilen yöntemde (Bu tezde DL-4S-NL ismi verilmiştir.) çok seviyeli nicemleyicinin son seviyesi çalıştırılmamakta, son seviyenin üretmesi gereken indeks bitleri yerine doğrudan gizli veri bitleri yerleştirilmektedir. Son indeksin gizli veri bitleriyle doldurulması nedeniyle oluşan fazladan bozulma daha önceki seviyelerce telafi edilmeye çalışılmaktadır. Bu yüzden gizli veri bitlerinden oluşturulan indekse ait vektör nicemlenecek girdi vektöründen çıkartılmakta, elde edilen yeni vektör MSVQ'nun ilk üç seviyesine sokularak nicemlenmektedir.

Chang ve Yu önerdikleri yöntemi, çıkartma işleminin hiç uygulanmadığı hayali bir veri gizleme yöntemiyle (Bu tezdeki ismiyle DD-4S, DD: doğrudan değiştirme) karşılaştırmışlar, çıkartma işleminin ne kadar faydalı olduğunu ortaya koymaya çalışmışlardır. Bu tez kapsamında tekrarlanan testlerde DL-4S-NL ve DD-4S yöntemlerinin ikisinin de birden steganografi ve damgalama uygulamalarına uygun olmadıkları belirlenmiştir.

3.1.1 Çıkartma işleminin işlevi

Bir MSVQ'nun temel amacı, belirttikleri vektörler toplanınca girdi vektöre en yakın vektör elde edilen indeks değerlerini belirlemektedir. Bu hedefin yerine getirilmesi için, çok seviyeli vektör tabloları üzerinde çeşitli tarama işlemleri yapılır. Son indeks bitlerinin gizli veri aktarımı amacıyla değerlendirildiği durumlarda, şayet taramalar tüm ihtimallerin dikkate alınması suretiyle gerçekleştiriliyorsa, çıkartma işlemi yapılsın yapılmasın elde edilecek indeks değerleri aynı olur. O halde Chang ve Yu tarafından önerilen çıkartma işlemi, konsepte yönelik değil pratiğe yönelik bir iyileştirme önerisidir.

Gerçek hayattaki uygulamalarda ise hesaplama karmaşıklığından dolayı MSVQ taraması sırasında tüm olasılıklar ziyaret edilemez. İşte bu tip tüm olasılıkların taranamadığı MSVQ'larda, Chang ve Yu'nun önerdikleri çıkartma işlemi sayesinde bozulma miktarlarında kısmi iyileşmeler sağlanabilir; ancak yapılan iyileştirmelere rağmen DL-4S-NL yöntemi kabul edilebilir sınırların hala çok uzağındadır. Diğer yandan zaten standart MELP'te MSVQ taraması M-en iyi yakınlaştırıcısından (M=8) istifade etmektedir, bu yakınlaştırıcı sayesinde büyük M değerlerinde düşük hata içeren kombinasyonlar bulunabilir. Şekil 3.1'de sunulan grafikte DL-4S-NL ve DD-4S yöntemlerinin farklı M değerlerindeki oransal ağırlıklı Hamming bozulmaları (D_{fm} bkz. Bölüm 2.3.2) gösterilmiştir.



Şekil 3.1 DL-4S-NL ve DD-4S yöntemlerinin farklı M değerlerindeki D_{fm} sonuçları

Şekil 3.1'den de takip edilebileceği üzere DL-4S-NL yöntemi DD-4S yönteminden daha az bozulmaya sebebiyet vermektedir. Düşük M değerleri için bozulmalar arasındaki fark belirgindir, fakat M değeri büyütüldükçe yöntemler arasındaki fark giderek kapanmaktadır.

3.1.2 Yüksek oranda bozulma

MSVQ'da kıpırtı benzeri veri gömme yönteminin en büyük problemi gizli veri eklemenin neden olduğu yüksek orandaki bozulmadır. Her ne kadar çıkartma işlemi sayesinde bozulma biraz dizginlenebilmiş olsa da, gizli veri eklemeden kaynaklanan fazladan bozulma standart haberleşme ortamındaki bozulmanın oldukça üzerindedir. Aşağıda verilmiş olan Çizelge 3.1'de standart MELP'in, 10^{-3} bit hata oranlı standart MELP'in, DD-4S ve DL-4S-NL yöntemlerinin D_{fm}, gizli veri biti başına D_{fm}, spektral bozulma ve P.862 MOS sonuçları verilmiştir. Tablodaki #sb sembolü bir çerçeveye gömülen gizli veri biti sayısını belirtmektedir.

Çizelge 3.1 Standart MELP'in, DD-4S, DL-4S-NL ve 10^{-3} bit hata oranlı standart MELP'in kaydedilen D_{fm}, *SD* ve gizli bit başına D_{fm} sonuçları

Yöntemler	#sb	D_{fm}	D_{fm} / #sb	*SD (dB)*	P.862 MOS
ST	-	0	0	1.6406	2.76
DD-4S	6	1.3154	0.2192	3.7338	2.69
DL-4S-NL	6	1.2707	0.2118	3.6594	2.69
ST-0.001	-	0.1548	-	1.8655	2.74

3.1.3 Tek kipte çalışma

MSVQ'da kıpırtı benzeri veri gömme yöntemi MELP çerçevesi başına alternatifsiz 6 bit gömmektedir ve ne yazık ki 10^{-3}'lük bir haberleşme ortamının epey üstünde bozulmaya yol açmaktadır. Yöntem MELP MSVQ'sunun en anlamsız son seviyesinde bile yüksek oranda bozulmaya sebep olduğundan, daha anlamlı seviyelerde (1., 2. ve 3. seviyeler) çalışan benzerlerinin veri gömme amacıyla kullanılması düşünülemez. Yöntemin neden olduğu bozulma miktarı, gizli veri gömülen çerçevelerin azaltılmasıyla düşürülebilir, ancak gizli bit başına bozulma her zaman sabit kalır.

3.1.4 DL-4S-NL yönteminin genel bir değerlendirmesi

DL-4S-NL yani MSVQ'da kıpırtı benzeri veri gömme yönteminin genel bir değerlendirmesi yapılırsa, uygulanan çıkartma işlemiyle, gizli veri kaynaklı ek bozulma bir nebze de olsa azaltılabilmektedir. Öte yandan bahse konu yöntemde çerçeve başına çok fazla gizli bit gömüldüğünden, kaydedilen bozulma oranları kabul edilebilir değerlerin oldukça üzerindedir. Yöntemin neden olduğu bozulma en basitinden daha az çerçeveye daha az gizli veri biti gömülerek azaltılabilir; ancak bu tip bir yaklaşımla gömülen bit başına bozulma kriterinde iyileşme sağlanamaz. Sonuç olarak, DL-4S-NL yöntemi steganografi ve damgalama uygulamaları için elverişli değildir; şayet MELP'te gizli veri saklama uygulamalarının geliştirilmesi hedefleniyorsa gizli veri biti başına daha az bozulmaya neden olan yeni yöntemler tasarlanmalıdır.

3.2 MELP-MSVQ-QIM Yöntemi

Aşırı derecede ek bozulmaya yol açmasından dolayı DL-4S-NL yöntemi, MELP üzerinde gerçeklenebilecek steganografi ve damgalama uygulamaları için elverişli değildir. DL-4S-NL yönteminde zaten en az anlamlı indeks bitleri gizli veri gömme amacıyla değerlendirildiğinden, ancak çerçeve başına daha az gizli veri biti gömülerek meydana gelen ek bozulma hafifletilebilir. Diğer taraftan, steganografide yaygın bir kullanım alanı bulan indeks modülasyonda indeks başına gömülecek bit sayısı kolaylıkla ayarlanabilmektedir.

Bu tez kapsamında önerilen özgün MELP-MSVQ-QIM yöntemi konvansiyonel nicemleme indeks modülasyonunun (QIM) MELP MSVQ'suna uyarlanmış özel bir türevidir. Önerilen özgün yöntem iki açıdan geleneksel QIM yöntemlerinden farklılık gösterir. İlk farklılık gizli veri kestiriminin yapıldığı sinyalin türündedir. Çoğu geleneksel QIM uygulamasında gizli veri Malvar vd. (2003) olduğu gibi kodu çözülmüş sinyal üzerinden kestirilir. MELP-MSVQ-QIM yönteminde ise gizli veri sadece MELP kodlanmış bit dizisi üzerinden kestirilebilir ve gömülü haldeki gizli verinin mevcudiyeti kodlanmış konuşma sinyalinin mevcudiyetine bağlıdır. İkinci farklılık, QIM'in ne tür bir nicemleme üzerinde uygulandığıyla ilgilidir. QIM yöntemlerinde gizli veri taşıyıcı

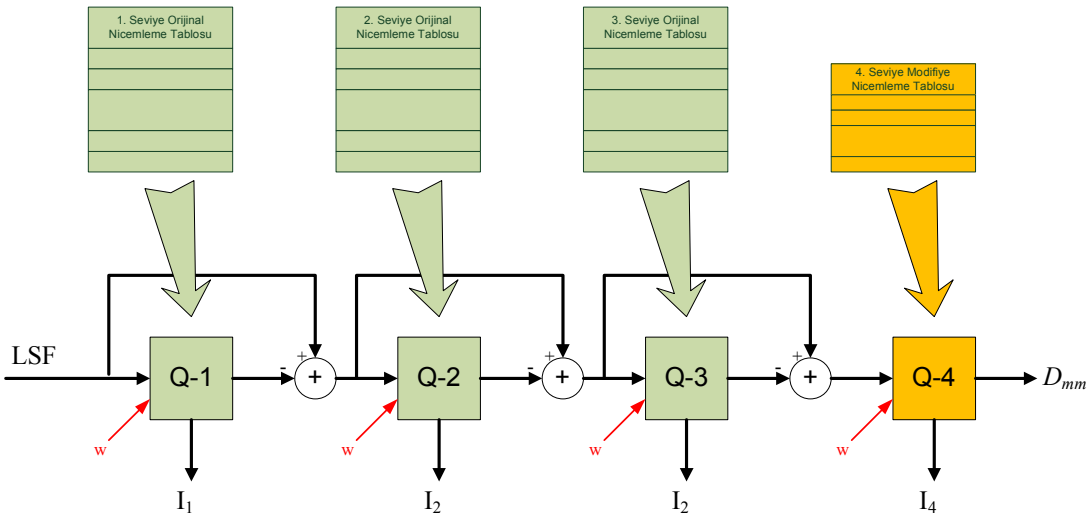
sinyale ekseriyetle skalar nicemleme aracılığıyla gömülür, MELP-MSVQ-QIM yöntemindeyse kodlayıcı bünyesinde mevcut olan MSVQ'dan istifade edilir. Son olarak, bölüm 2.4.5'te verilen sınıflandırma sistematiğine göre, MELP-MSVQ-QIM yöntemi yarı kırılgan, açık ve taşıyıcı sinyalden parazit kapmayan bir veri gömme yöntemi olarak nitelendirilebilir.

MELP-MSVQ-QIM yönteminin esası, bilindik nicemleme indeks modülasyonunun MELP MSVQ'sunun önceden seçilen indekslerinin önceden seçilen bitleri üzerindeki tatbikine dayanır. Ek 3'te de anlatıldığı üzere, MELP'teki MSVQ'dan LSF değerlerinin nicemlenmesinde yararlanılır ve bahse konu MSVQ'nun çalıştırılması sonucunda elde edilen 4 adet MSVQ indeksi 54 bitlik bir MELP çerçevesinin 25 bitini kaplar.

MELP-MSVQ-QIM yönteminde gömme işlemi, seçilmiş indekslerin seçilmiş bitlerinin, gizli veri bitleri tarafından modüle edilmesiyle yerine getirilir. Seçili bir indeksin seçili bir bitinin – bitlerinin, gizli bir veri bitiyle – bitleriyle modüle edilmesinden kasıt, nicemleme işlemi sonucunda seçili bitin – bitlerin değerinin gizli veri bitinin – bitlerinin değerine eşit kılınmasıdır. Bahse konu eşit kılma, nicemleme tablosunun daraltılmasıyla temin edilir. Daraltma işlemi gizli veri bitinin değerine – bitlerinin değerlerine ve indeksin seçilmiş olan bitine – bitlerine göre değişkenlik gösterir; daraltma sonucu elde edilen tablonun tüm indekslerinin seçili bitlerinde aynı gizli veri bit değerinin – değerlerinin bulunması sağlanır. Gizli verinin gömüleceği tablonun daraltılmasından sonra MSVQ yine eskisi gibi çalıştırılır, en az hatayı sağlayan MSVQ indeksleri tayin edilir (Şekil 3.2).

MELP-MSVQ-QIM yönteminde nicemleme tablolarının daraltılması maalesef bazı optimum indeks kombinasyonlarının yitirilmesine yol açar. Öte yandan, en az hatayı veren indeks değerleri yine de (daha az) çeşitlilik içeren bir tablo üzerinden, daha-az-optimum indeks kombinasyonları arasından aranır; böylece LSF'lerdeki bozulma, gömülecek gizli verinin varlığına rağmen olabildiğince sınırlandırılmaya çalışılır. Anlatılan, bu hata dizginleme yaklaşımı DL-4S-NL yönteminin hata dizginleme yaklaşımından farklı olarak tamamen kavramsal niteliktedir.

Aşağıda sunulan Şekil 3.2'de, MELP MSVQ'nun son seviyesi üzerinde gerçekleştirilen bir MELP-MSVQ-QIM uygulamasına yer verilmiştir. Şekildeki I_1, I_2, I_3 ve I_4 değerleri MSVQ tarafından bulunan, en az hatayı sağlayan indeks değerleridir ve gizli veri I_4 içinde saklanmaktadır. *w* algısal açıdan hangi LSF değerlerinin daha önemli olduğunu belirten bir ağırlık vektörüdür. MELP-MSVQ-QIM uygulamasının çalışmasına değinmek gerekirse, LSF değerleri önce Q-1 nicemleyicisi tarafından nicemlenmekte, en az ağırlıklı hatayı sağlayan indeks değeri, I_1 bulunmaktadır. Daha sonra Q-1 nicemleyicisinin nicemleyemediği kısım yani kalan hata Q-2 tarafından nicemlenmekte, onun kalan hatası da Q-3 nicemleyicisine sokulmaktadır. Q-3'ün kalan hatası da Q-4 nicemleyicisine alınmakta, ancak Q-4 nicemleyicisinde diğer üç nicemleyiciden farklı olarak daraltılmış tablo kullanılmaktadır.

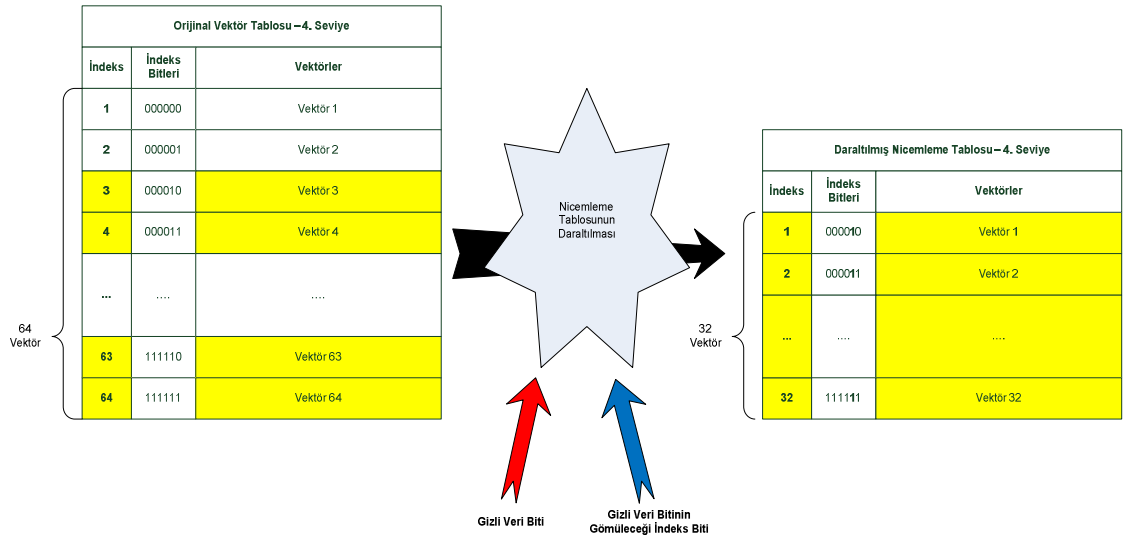


Şekil 3.2 MELP-MSVQ-QIM yönteminde MSVQ indekslerinin bulunması

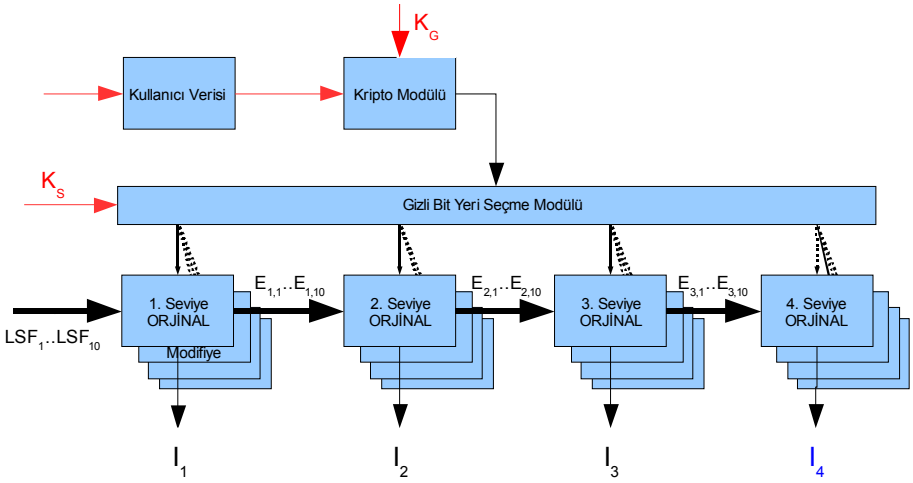
Şekil 3.2'yi takiben verilen Şekil 3.3'te, gizli veri biti MELP MSVQ'sunun 4. seviye indeksinin 5. en anlamlı bitine yerleştirilmektedir. Gizli veri bitinin değeri 1'e eşit olduğundan, daraltılmış tablonun tüm indekslerinin 5. anlamlı bitlerinin 1'e eşit olması gerekmektedir. Yapılan daraltma işlemi sonucunda son seviyenin nicemleme basamağı sayısı 64'ten 32'ye düşer. Eğer bir indeks değerine sadece bir bit gizli veri gömülüyorsa daraltma işlemi sonucunda elde edilen güncellenmiş tablonun boyu orijinalin yarısı kadardır. Diğer yandan eğer istenirse bir indekse birden fazla bit de

gömülebilir; örneğin bir indekse iki bit birden gömülürse bu durumda daraltılmış tablonun boyu orijinalinkinin dörtte birine iner.

Nicemleme tablosu daraltma işlemi, gizli veri gömülecek her indeks değeri için yeni baştan tekrar edilir. Arzu edilirse, gömülecek gizli veri bitleri farklı MELP çerçevelerinde ayrı indeks ve bit pozisyonlarına yerleştirilebilir ve hatta çerçeve başına gömülen bit sayısı bile değiştirilebilir. Gizli veri bitlerinin yerleştirilecekleri indeks ve bitler, gizli bir anahtarından üretilen sözde rastgele diziler doğrultusunda belirlenebilir. (Tıpkı frekans atlamalı haberleşmede, atlama yapılan frekansların gizli kripto anahtarlarınca belirlenmesi gibi) Şekil 3.4'te sunulan örnekte, gizli veri bitlerinin gömüldüğü indeks bitleri her MELP çerçevesi için değiştirilmektedir. Gömülecek gizli veri bitleri önce K_G anahtarı vasıtasıyla şifrelenmekte, daha sonra K_S anahtarının belirlediği indeks bitlerine gömülmektedir.

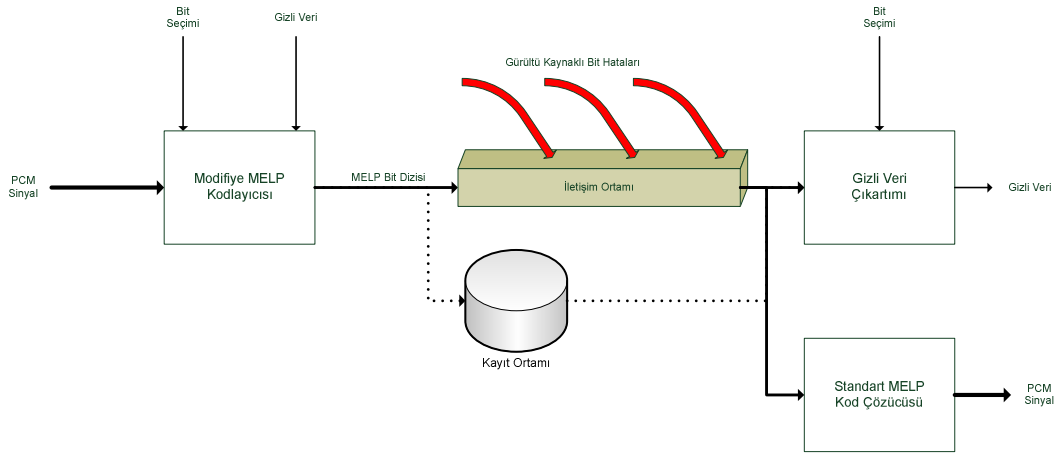


Şekil 3.3 MELP-MSVQ-QIM yönteminde nicemleme tablosunun daraltılması



Şekil 3.4 MELP çerçevelerinde gizli veri gömülecek bit yerlerinin tayini

Şekil 3.5'te MELP-MSVQ-QIM yönteminin steganografik kullanım senaryosu sunulmuştur. Kaynak uçta gizli veri MELP kodlayıcısına PCM biçimindeki konuşma sinyaline paralel olarak sokulur. Söz konusu gizli veri, gizli veri gömen MELP kodlayıcısı tarafından kodlanmış çıktının belli alanlarına iliştirilir. Gizli veri bitlerinin iliştirilecekleri yerler bit seçimi girdisine göre belirlenir. Daha sonra gizli veri içeren kodlanmış çıktı bir iletişim kanalı (muhtemelen bit hataları içeren bir iletişim kanalı) vasıtasıyla alıcı uca aktarılır. Alıcı uç bit seçimi girdisi sayesinde gizli veri bitlerinin iliştirildikleri indeks ve bitlere senkronize olur ve gömülü bitleri bulundukları yerlerden sökerek gizli verinin bütününü elde eder. Gizli verinin sökülmesi işlemi için MELP kod çözücüsüne ihtiyaç yoktur.



Şekil 3.5 MELP-MSVQ-QIM yönteminin steganografik kullanım senaryosu

3.3 MELP-MSVQ-QIM Varyasyonları

QIM uygulanan indekslerin değiştirilmesi suretiyle MELP-MSVQ-QIM yönteminin pek çok sayıda varyasyonu türetilebilir. Çizelge 3.2'de yer alan listede MELP-MSVQ-QIM yönteminden türetilmiş olan varyasyonlar takdim edilmiştir. QIM uyguladıkları indekslerin anlamlılıklarına ve gömdükleri veri miktarlarına bağlı olarak varyasyonlar değişik düzeylerde ek bozulmalara yol açarlar. Listenin en sol tarafında yer alan sütun MELP-MSVQ-QIM varyasyonlarının isimlerini, orta sütun varyasyonlar hakkındaki tanıtıcı açıklamaları ve en sağdaki sütun da varyasyonların birim zamanda gömdükleri veri miktarlarını içermektedir.

Çizelge 3.2 MELP-MSVQ-QIM Varyasyonları

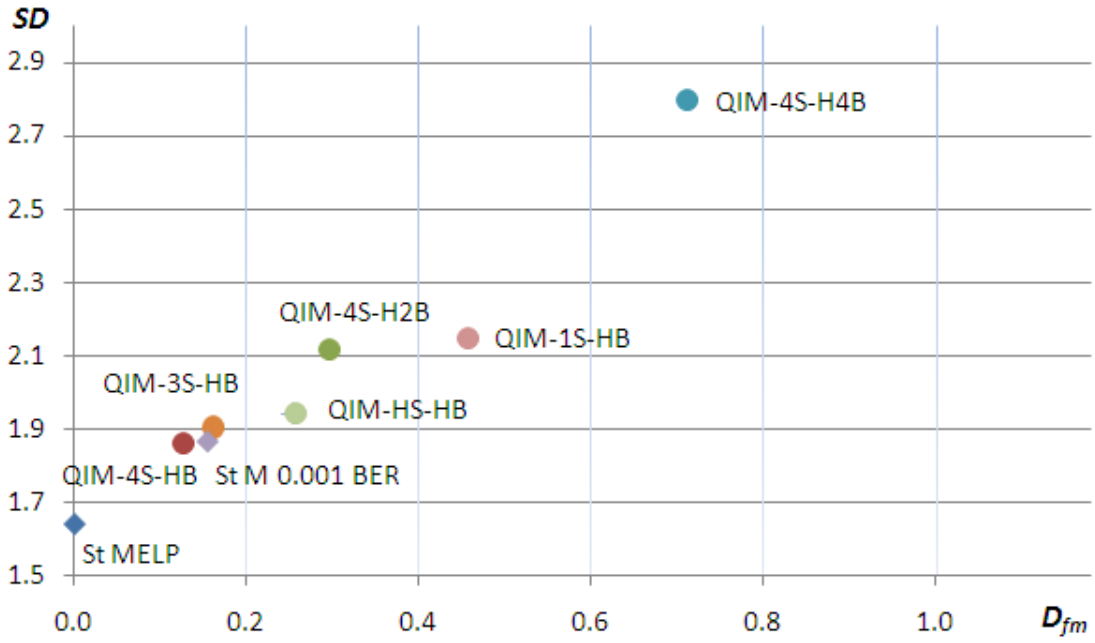
MELP-MSVQ-QIM Varyasyonu	Açıklama	Kapasite (bps)
QIM-1S-HB	1. Seviyenin Herhangi Biti Üzerinde QIM MELP MSVQ'sunun 1. indeksi üzerindeki herhangi bir bitte QIM uygulanarak gizli veri biti saklanması	44.44
QIM-2S-HB	2. Seviyenin Herhangi Biti Üzerinde QIM MELP MSVQ'sunun 2. indeksi üzerindeki herhangi bir bitte QIM uygulanarak gizli veri biti saklanması	44.44
QIM-3S-HB	3. Seviyenin Herhangi Biti Üzerinde QIM MELP MSVQ'sunun 3. indeksi üzerindeki herhangi bir bitte QIM uygulanarak gizli veri biti saklanması	44.44
QIM-4S-HB	4. Seviyenin Herhangi Biti Üzerinde QIM MELP MSVQ'sunun 4. indeksi üzerindeki herhangi bir bitte QIM uygulanarak gizli veri biti saklanması	44.44
QIM-4S-H2B	4. Seviyenin Herhangi 2 Biti Üzerinde QIM MELP MSVQ'sunun 4. indeksi üzerindeki herhangi iki bitte QIM uygulanarak gizli veri biti saklanması	88.88
QIM-4S-H4B	4. Seviyenin Herhangi 4 Biti Üzerinde QIM MELP MSVQ'sunun 4 indeksi üzerindeki herhangi dört bitte QIM uygulanarak gizli veri biti saklanması	177.76
QIM-HS-HB	Herhangi Seviyenin Herhangi Biti Üzerinde QIM MELP MSVQ'sunun herhangi bir indeksi üzerindeki herhangi bir bitte QIM uygulanarak gizli veri biti saklanması	44.44

3.4 MELP-MSVQ-QIM Varyasyonlarının Test Sonuçları

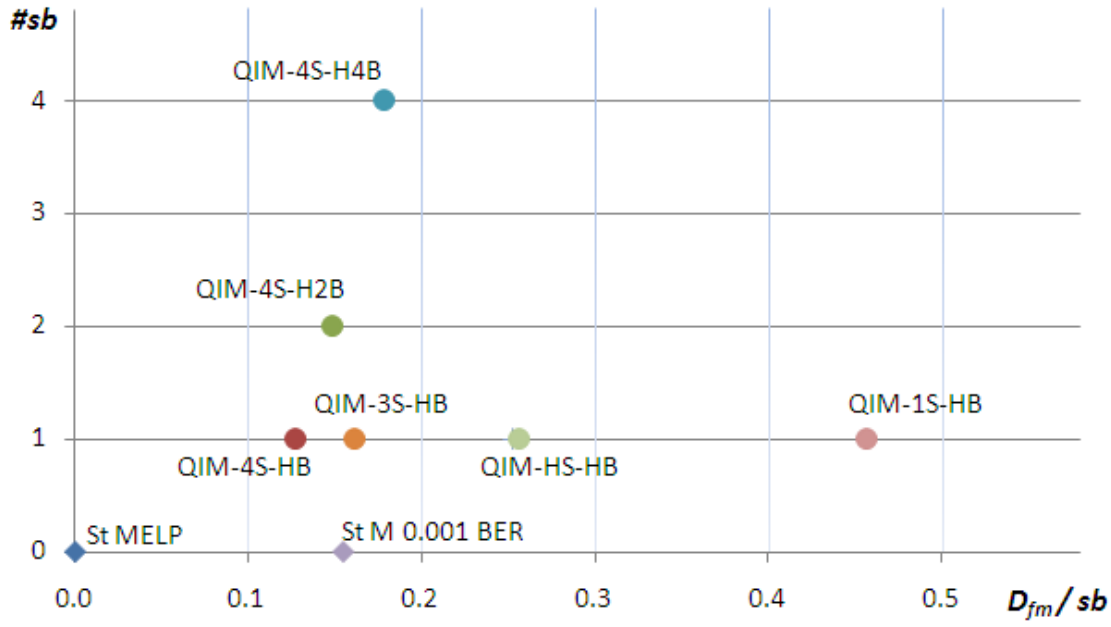
Bu tez kapsamında önerilen MELP-MSVQ-QIM yönteminin steganografik kullanıma uygunluğunun ölçülebilmesi için, türetilen MELP-MSVQ-QIM varyasyonları ayrı ayrı gerçeklenerek testlere sokulmuştur. Testler sırasında NTIMIT (Jankowski vd. 1990) veri tabanından faydalanılmış, veri tabanında bulunan tüm örnekler varyasyonlarca kodlanmıştır. Testler sonucunda varyasyonlara ait *SD*, D_{fm} , bit başına D_{fm} ve P.862 MOS sonuçları elde edilmiş, Çizelge 3.3'te tablosal, Şekil 3.6 ve 3.7'de grafiksel olarak sunulmuştur.

Çizelge 3.3 MELP-MSVQ-QIM varyasyonlarının için *SD*, D_{fm} , bit başına D_{fm} ve P.862 MOS metrikleri cinsinden ortalama bozulma miktarları

Yöntemler ve Varyasyonlar	**#sb**	D_{fm}	D_{fm} / **#sb**	***SD (dB)***	**P.862 MOS**
QIM-1S-HB	1	0.4564	0.4564	2.1479	2.72
QIM-2S-HB	1	0.2528	0.2528	1.9419	2.74
QIM-3S-HB	1	0.1613	0.1613	1.9040	2.74
QIM-4S-HB	1	0.1275	0.1275	1.8587	2.75
QIM-4S-H2B	2	0.2964	0.1482	2.1150	2.74
QIM-4S-H4B	4	0.7122	0.1780	2.7960	2.72
QIM-HS-HB	1	0.2563	0.2563	1.8931	2.74
ST	-	0	0	1.6406	2.76
DD-4S	6	1.3154	0.2192	3.7338	2.69
DL-4S-NL	6	1.2707	0.2118	3.6594	2.69
ST-0.001	-	0.1548	-	1.8655	2.74



Şekil 3.6 MELP-MSVQ-QIM varyasyonlarının SD vs D_{fm} sonuçları



Şekil 3.7 MELP-MSVQ-QIM varyasyonlarının #sb vs D_{fm}/#sb sonuçları

Çizelge 3.3, Şekil 3.6 ve Şekil 3.7'de verilen *SD*, D_{fm} , bit başına D_{fm} ve P.862 MOS sonuçları dikkatlice incelendiğinde, çizelge ve grafiklerde yer alan tüm kalite metriklerinin birbirleriyle uyum içerisinde oldukları görülmektedir. Fakat kalite metriklerinin hassasiyetleri birbirlerinden farklı düzeylerdedir; MOS sonuçlarındaki duyarlılık diğer kalite metriklerine göre daha azdır.

Yöntemler ve varyasyonlar arasında en başarılı olanı tartışmasız QIM-4S-HB varyasyonudur. Söz konusu varyasyonda gizli eklemenin neden olduğu bozulma, 10^{-3} bit hata oranlı bir haberleşme ortamında ölçülen bozulmanın dahi altındadır. QIM-3S-HB varyasyonu ise başarıda ikinci sıradadır, onun sonuçları da QIM-4S-HB sonuçlarına oldukça yakın çıkmıştır.

Bu arada, MELP'in ötümsüz çerçevelerinde uzun vadeli (*LT: long term*) ilişkiler tanımlı değildir, bu sayede ötümlü MELP çerçevelerinde LT değerleri için ayrılan alanlar ötümsüz çerçevelerde ileri hata kodlamasıyla değerlendirilir. Buna bağlı olarak, bit hatası içeren haberleşme ortamında çalışan standart MELP uygulamasında, ötümsüz çerçevelerden hesaplanan D_{fm}, ötümlü çerçevelerden hesaplanan D_{fm}'den düşük olur. Ötümlü ve ötümsüz çerçevelerin kümülatif D_{fm}'si ise, hesaplanan ötümlü ve ötümsüz D_{fm}'lerin aritmetik sonucudur. Ancak, konuşma sinyalinin anlaşılabilirliği açısından ötümlü çerçeveler ötümsüz çerçevelere göre daha baskındır, çünkü enerjileri daha yüksektir. Bu yüzden gizli veri yöntemlerinin performansının ölçülmesinde ötümlü çerçevelere ait olan D_{fm} dikkate alınmalıdır.

Sonuç olarak, QIM-2S-HB, QIM-4S-H2B ve QIM-HS-HB varyasyonlarının D_{fm}'leri 10^{-3} bit hata oranlı haberleşme ortamının D_{fm}'sinden kötü olsalar da, bahse konu ortamın ötümlü çerçeve D_{fm}'sininden kötü değillerdir. Buna bağlı olarak da söz konusu yöntemler kalite metrikleri açısından kabul sınırlarının içerisinde değerlendirilebilirler. Hatta QIM-HS-HB varyasyonu gizli veriyi daha geniş bir alan üzerinde sakladığından, diğer tüm varyasyonlardan daha güvenli olduğu kabul edilebilir. (Daha geniş frekans aralığında yapılan frekans atlamasının daha güvenli kabul edilmesi gibi) QIM-1S-HB ve QIM-4S-H4B varyasyonlarına gelince, bunlarda görülen bozulma maalesef kabul sınırlarının ötesindedir.

3.5 MELP-MSVQ-QIM Varyasyonlarının Entropi Açısından Değerlendirilmesi

Entropi kavramı enformasyon teorisinde, rastgele değişken ile ilgili belirsizliğin bir ölçüsü olarak tanımlanır. Shannon'a göre bir rastgele değişkenin değeri bilinmediği zaman, o değişkenin içerdiği bilginin miktarını ölçebilmek için entropi (Shannon entropisi) kullanılır. Shannon entropisinden steganografik bir gösterge olarak faydalanılabilir; örneğin Malik vd. (2008) geleneksel QIM yöntemlerinin steganalizinde entropi ölçümlerinden yararlanmışlardır.

Klasik steganografi uygulamalarında saklanacak gizli veri, taşıyıcı veriden (sinyalden) genel olarak bağımsızdır, bundan dolayı taşıyıcı veriyle gizlenecek ham verinin Shannon entropileri birbirlerinden farklı olabilir. Birbirlerinden farklı iki verinin (gizli veri ve taşıyıcı veri) çeşitli mekanizmalarla (veri gömme yöntemi) bir araya getirilmeleriyle ortaya çıkan bileşik verinin entropisi de girdi entropilerinden farklı olabilir. Entropiler arasındaki farklılığın artması, (bazı) istatistikî özelliklerdeki ayrışmanın da arttığı anlamına gelir, bu durumdan istifade edebilen steganalizörlerin işleri kolaylaşır. Bu bölümde kullanılmakta olan steganografik iz kavramı, taşıyıcı ve gömülü veriler arasındaki istatistikî ayrışmayı belirtmektedir.

Entropi kavramı açısından, MELP-MSVQ-QIM yöntemini en fazla ilgilendiren çalışmayı Rahikka, Fuja ve Fazel (1999) gerçekleştirmiştir. Onlar, konuşma sinyalinin MELP tarafından kodlanması sonucu elde edilen bit dizisinin hafızasız, eşit ihtimalli ve düzgün dağılımlı olmadığının altını çizmişler, analiz ettikleri MELP çıktılarının değişik alanları için değişen miktarlarda artıklık (*redundancy*) hesaplamışlardır. Gerçekte, makalelerinde tanımladıkları artıklık, Markov durum makinelerine göre hesaplanan bir nevi Shannon entropisidir.

MELP-MSVQ-QIM varyasyonları ile gömülen gizli veri, kullanıcı verisi şifrelendiği ve beyaz gürültüye benzetildiği için artıklık içermez, sıkıştırılamaz. Bu yüzden gizli veri gömülmüş MELP parametrelerinin entropileri gizli veri gömülmemiş türdeşlerinin entropilerden daha fazla olmalıdır. Bu durumun pratikte de ortaya konabilmesi için, MELP-MSVQ-QIM varyasyonları ile DD-4S ve DL-4S-NL yöntemlerine ait çıktıların

entropileri hesaplanmıştır. Entropi hesabı, Rahikka, Fuja ve Fazel'in yaptığı gibi Markov modellerine dayanmaktadır ve her MSVQ indeksi için ayrı entropi değeri bulunmuştur. Çizelge 3.4'te gizli veri gömmenin indeks entropileri üzerindeki etkileri sunulmaktadır. Bir indeksinin entropisi, o indeksi oluşturan bitlerin ortalama olarak hangi verimle değerlendirildiğini belirtmektedir. Veri gömme yöntemlerinin entropiyi etkilemediği gözler boş bırakılmıştır.

Çizelge 3.4 Gizli veri gömme yöntemleri ve standart MELP'in entropi sonuçları (B/Ç : Bit / Çerçeve, çerçeve başına bit sayısı)

Yöntem	LSF-1 (B/Ç)	LSF-2 (B/Ç)	LSF-3 (B/Ç)	LSF-4 (B/Ç)
St.MELP (0 BER)	4.10	4.53	5.24	5.57
QIM-4S-HB				5.56
QIM-4S-H2B				5.73
QIM_4S-H4B				5.84
QIM-3S-HB			5.38	5.58
QIM-2S-HB		4.82	5.25	5.57
QIM-1S-HB	4.83	4.56	5.25	5.57
QIM-HS-HB	4.38	4.65	5.28	5.59
DD-4S				5.96
DL-4S-NL	4.41	4.70	5.37	5.96

Çizelge 3.4'te verilen sonuçlara göre, gizli veri gömme işleminin uygulandığı indeks ve ondan sonraki tüm indekslere ait entropilerin arttığı gözlemlenmektedir. 1. seviyeyi değiştiren yöntemler tüm seviyelerin, 2. seviyeyi değiştiren yöntemler 2. 3. ve 4. seviyelerin, 3. seviyeyi değiştiren yöntemler 3. ve 4. seviyelerin entropisini yükseltmekteyken, 4. seviyeyi değiştiren yöntemler sadece 4. seviyenin entropisini yükseltmektedir. Çerçeve başına gömülen gizli bit sayısı ile beraber entropi artışları şiddetlenmiştir.

Çizelge 3.4'teki sonuçlar steganografik açıdan yorumlanırsa, en düşük steganografik ize QIM-4S-HB varyasyonunda, en büyük steganografik izlereyse QIM-1S-HB varyasyonu ile DD-4S ve DL-4S-NL yöntemlerinde tanık olunmaktadır. Bir indeksin entropisi yükseldikçe o indeksin gizli veri kabul potansiyeli artmaktadır ki bu duruma göre QIM-4S-HB'nin steganalize karşı en dayanıklı varyasyon olabileceği öngörülebilir.

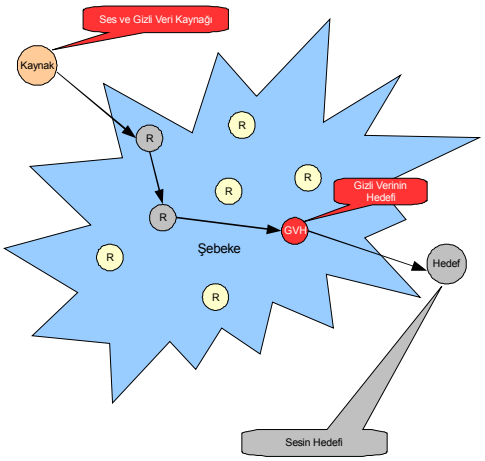
Entropi sonuçlarının en ilginç çıkarımlardan biri QIM-HS-HB varyasyonuyla ilgili olanıdır, QIM-HS-HB varyasyonu gizli veriyi en geniş alan üzerinde saklamasından dolayı daha güvenlidir, ancak entropi ve steganografik iz açısından QIM-4S-HB varyasyonuna göre daha güvensizdir. Özetle, gizli verinin daha geniş alana yayılmasının sağladığı güvenlikle steganografik güvenlik aynı kavramlar değildir. Diğer taraftan Çizelge 3.4'teki entropi sonuçlarıyla Çizelge 3.3'teki kalite metrik sonuçları birbirleriyle uyum içerisindedir, ancak bu durum kısmen rastlantısaldır. Normal şartlar altında sıradan bir MSVQ'nun ilk indekslerinin sonrakilerden daha düşük entropilere sahip olması beklenebilir, ama bu durum tam anlamıyla bir kaide değildir. Eğer MSVQ hatalı bir şekilde eğitilseydi, anlamlı indekslerin entropileri bir şekilde anlamsızlarınkinden daha yüksek tutulsaydı, hesaplanan entropi sonuçları Çizelge 3.4'teki gibi olmayacak, anlamsız indekslerin steganografik izleri daha büyük, anlamlı indekslerin steganografik izleri daha küçük olabilecekti.

3.6 MELP-MSVQ-QIM Varyasyonlarının Pratik Uygulamaları

Video, görüntü ve müzik sinyalleri üzerinde çalışan gizli veri gömme uygulamaları konuşma sinyali üzerinde çalışan gizli veri uygulamalarına göre çok daha yaygın, çeşitli ve gereklidir, ancak bu duruma rağmen sınırlı olsa da konuşma sinyaline gizli veri gömme uygulamaları tasarlanabilir. Bu bölümde, MELP-MSVQ-QIM varyasyonlarından istifade edilerek konuşma sinyalleri üzerinde gerçekleştirilebilecek çeşitli pratik uygulama önerileri yer almaktadır.

3.6.1 Steganografik kullanım

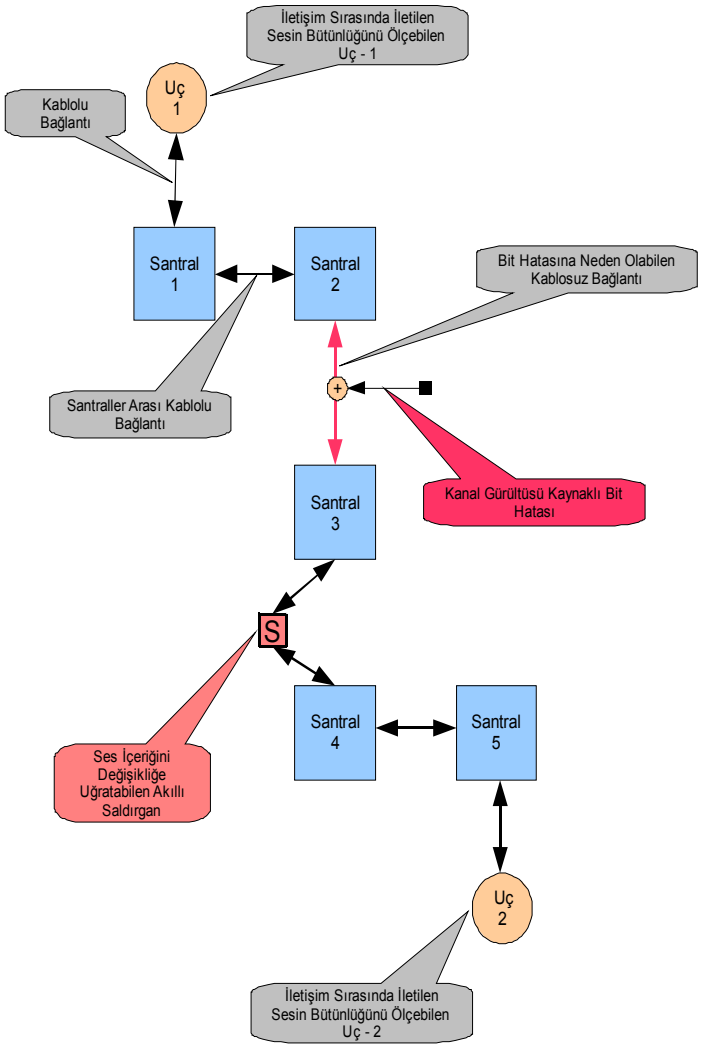
İki uç arasındaki MELP kodlanmış konuşma haberleşmesi sırasında, kaynak iletim rotası üzerinde bulunan herhangi bir noktaya gizli veri aktarımı yapabilir. Uygulamanın başarılı olabilmesi için konuşma sinyalindeki bozulma olabildiğince düşük tutulmalıdır, kaynaktan kodlanmış bir halde çıkan verinin ara noktalarda değişikliğe uğratılmaması garanti edilmelidir (Şekil 3.8).



Şekil 3.8 Haberleşme rotası içindeki bir noktaya gizli veri transferi

3.6.2 Konuşma sinyalindeki bozulmanın ölçümü

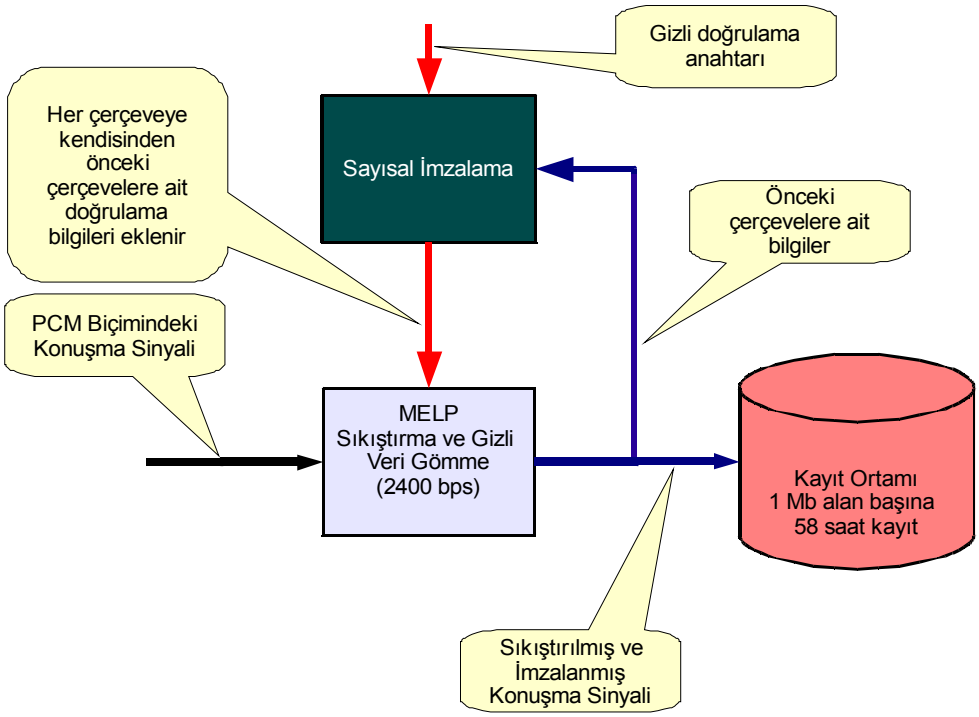
Ana düğüm noktaları birbirlerine kablosuz linkler vasıtasıyla bağlanmış mobil sistemlerde, herhangi iki düğüme bağlı bulunan iki uç arasında aktarılan MELP çerçevelerinin hangi ölçüde bozulduğu veya değiştirildiği (tanımlı bir güvenilirlikte) veri gizleme metotları yardımıyla ölçülebilir (Şekil 3.9).



Şekil 3.9 İletişim sırasında aktarılan konuşma sinyalinin bütünlüğünün ölçümü

3.6.3 Kaydedilmiş sesin veri bütünlüğünün kontrolü

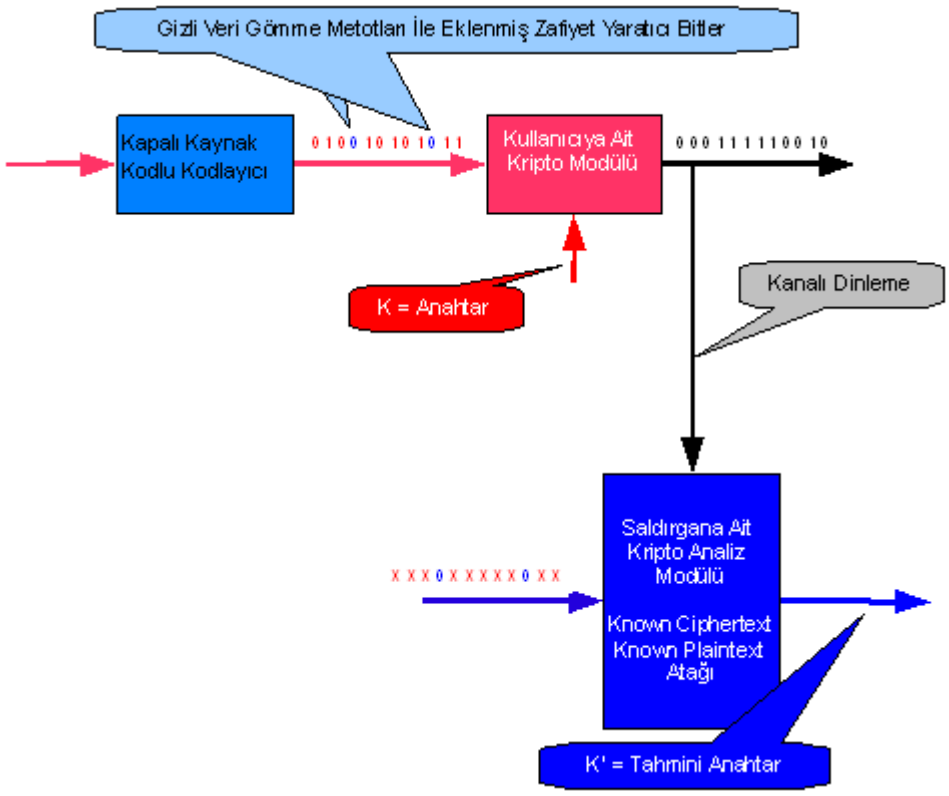
Konuşma kodlama algoritmaları konuşma sinyalinin sayısal kayıt ortamlarında saklanması sırasında sıkıştırma amaçlı olarak kullanılabilir. Veri gizleme metotları, saklanan konuşma örneklerinin veri bütünlüklerinin kontrolünde ve / veya kötü niyetle değiştirilmiş alanların tespitinde kullanılabilir (Şekil 3.10).



Şekil 3.10 Kayıt cihazlarında veri bütünlüğü uygulaması

3.6.4 Satılan ürünlere güvenlik zafiyeti katılması

Kapalı kaynak kodları ile ürün geliştiricilere veya kullanıcıya satılan kodlayıcılar içerisine gizli veri gömme metotları eklenerek, kodlayıcı çıktılarında önceden belirlenmiş dizilimlerin oluşması sağlanabilir. Eğer söz konusu kodlayıcı çıktıları oldukları gibi şifrelenirse bilinen açık metin – bilinen şifreli metin atağına imkân tanınmış olur (Şekil 3.11).



Şekil 3.11 Gizli veri gömme yöntemleri ile güvenlik zafiyeti yaratma

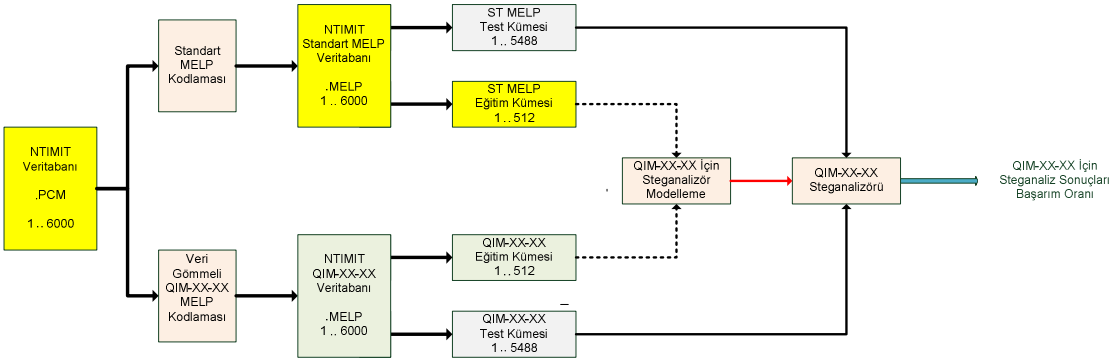
3.7 MELP-MSVQ-QIM Steganalizi

Steganaliz testleri MELP-MSVQ-QIM yönteminin sadece üç varyasyonu üzerinde gerçekleştirilmiş, daha evvelden ek bozulma açısından incelenmiş pek çok MELP-MSVQ-QIM varyasyonu ile DL-4S-NL ve DD_4S yöntemleri bu bölümde sunulan steganaliz testlerine dâhil edilmemişlerdir. Bu durumun iki temel nedeni bulunmaktadır. Birinci neden, ek bozulmalar açısından sadece QIM-4S-HB, QIM-4S-H4B ve QIM-HS-HB bir sonraki aşamada, steganaliz aşamasında incelemeye değerdir. QIM-4S-HB en az bozulmaya yol açan, QIM-HS-HB gizli veriyi en geniş alan üzerinde saklayan ve QIM-4S-H4B çerçeve başına en fazla gizli veri gömen MELP-MSVQ-QIM varyasyonlarıdır. İkinci neden ise steganalize alınan bu üç varyasyonun sonuçları sayesinde steganalizi yapılmayanların sonuçlarının iyi kötü öngörülebilmesidir. Örneğin DL-4S-NL yönteminin QIM-4S-H4B varyasyonuna göre daha düşük steganografik fark edilmezliğe sahip olacağı son derece açıktır.

MELP-MSVQ-QIM varyasyonlarının steganaliz testleri iki aşamadan oluşmaktadır. İlk aşamada her varyasyon için ayrı bir steganalizör eğitilmiş, ikinci aşamada eğitilen steganalizörler teste tabi tutulmuşlardır. Hem eğitim hem de testlerde NTIMIT (Jankowski vd. 1990) konuşma veri tabanından yararlanılmıştır. Bahse konu veritabanından 1 ila 3 saniye arasındaki uzunluklara (Ortalama = 1,8 saniye) sahip 6000 adet örnek seçilmiştir. Seçilen örnekler üzerinde standart MELP, QIM-HS-HB, QIM-4S-HB ve QIM-4S-H4B uygulanmış, böylece her biri 6000 örnekten oluşan 4 adet ana-küme üretilmiştir. Daha sonra her bir ana-küme, biri 512 diğeri 5488 örnekten oluşan alt-kümelere bölünmüştür. 512'lik alt-kümeler steganalizör eğitiminde, 5488'lik alt-kümeler ise steganalizör testlerinde kullanılmışlardır.

Şekil 3.12'de de gösterildiği üzere her bir gizli veri gömme yöntemi için ayrı bir steganalizör modeli eğitilmektedir. Bir steganalizör modelinin bir veri gömme yöntemini üzerinde uzmanlaşabilmesi için birisi gizli veri gömülmemiş, diğeri gizli veri gömülmüş örnekleri barındıran iki ayrı kümeyi işlemesi gerekmektedir. Girdi olarak alınan kümelerin, aynı örneklerin gizli veri içeren ve içermeyen sürümlerinden oluşturulması eğitimin kalitesi açısından oldukça mühimdir. Steganalizör eğitimi basittir (Kocal vd. 2008); önce her iki kümede yer alan tüm örnekler için kaotik öznitelikler (FNF ve Lyapunov katsayıları) hesaplanır. Daha sonra, veri gömmenin hangi kaotik özniteliklerde etki gösterdiği araştırılır, hangi özniteliklerin kümeler arasında ayrıştığı tespit edilir, ayrışmayan veya çok az ayrışan öznitelikler elimine edilir. Eğitimin en son kısmında ayrışan özniteliklerden doğrusal bağlanım sınıflandırıcısı tasarlanır.

Eğitilmiş bir steganalizör (içerisinde gizli veri olup olmadığı araştırılan) girdi bit dizisini "gizli veri var" veya "gizli veri yok" diye sınıflandırır. Sınıflandırmanın yapılabilmesi için, ilk olarak girdi bit dizisinin kaotik öznitelikleri hesaplanır. Ardından, bulunan kaotik öznitelikler eğitilmiş sınıflandırıcıya sokulur, sınıflandırıcının kararı beklenir.



Şekil 3.12 QIM-XX-YY varyasyonu için steganalizör modelinin eğitilmesi ve testi

İkinci aşamada, eğitilen dört steganalizör modeli de sırayla test edilmiştir. Daha önceden de ifade edildiği üzere, testlerde steganalizör başına iki adet 5488'lik alt-küme değerlendirmeye alınmış, gizli veri gömülmüş ve gömülmemiş örneklerin sayısı birbirlerine eşit tutulmuştur. Örnekler sırayla steganalizörlere sokulmuş, her bir steganalizör için ayrı ayrı başarım oranı hesaplanmıştır. Denklem (3.1)'de başarım oranının nasıl hesaplandığı verilmiştir:

$$\text{Başarım Oranı} = \frac{TP + TN}{TP + TN + M + FA} \tag{3.1}$$

TP	(True Positive)	: Gizli veri varlığının başarıyla ifşa edilmesi
TN	(True Negative)	: Gizli veri yokken olmadığının belirlenmesi
M	(Miss)	: Var olan gizli veri varlığının tespit edilememesi
FA	(False Alarm)	: Gizli veri olmadığı halde olduğunun zannedilmesi

Çizelge 3.5'in 1. sütununda MELP-MSVQ-QIM varyasyonlarının steganaliz sonuçları sunulmuştur. Başarım oranı 0.5 değerine yaklaştıkça veri gizlemedeki steganografik fark edilmezlik artar. Sonuçlar 0.5'ten uzaklaştıkça gizli veri varlığının tespiti kolaylaşır. Verilen çizelge incelendiğinde, steganaliz sonuçlarının kaydedilen bozulma değerlerine paralel seyrettiği görülmektedir. En az bozulmaya yol açan QIM-4S-HB aynı zamanda en fazla steganografik fark edilmezliğe sahip MELP-MSVQ-QIM varyasyonudur. Çerçeve başına gömülen bit sayısı arttırıldıkça, steganalizörlerin başarım oranları yükselmektedir.

Çizelge 3.5 QIM Yöntemlerinin Steganaliz Başarım Oranları

MELP-MSVQ-QIM Varyasyonu	Başarım Oranı (0 BER)	Başarım Oranı (10-3 BER)	Başarım Oranı (10-2 BER)
QIM-4S-HB	0.5227	0.5171	0.5007
QIM_HS_HB	0.5828	0.5767	0.5481
QIM_4S_H4B	0.6517	0.6478	0.5989

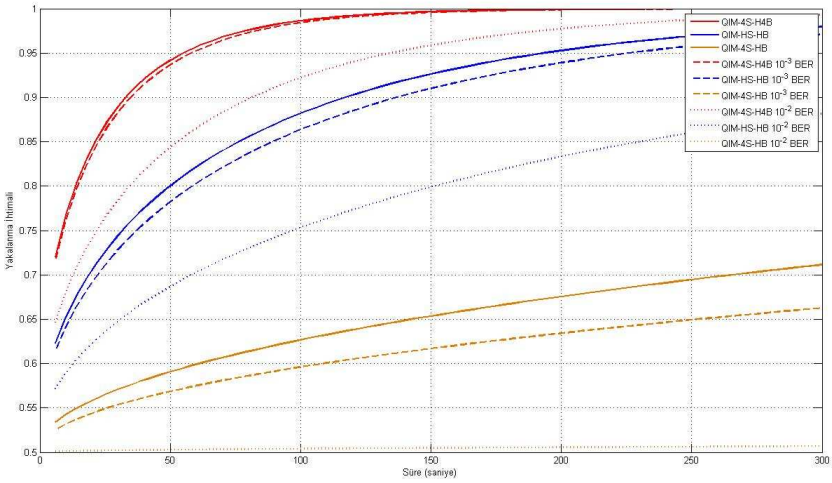
MELP-MSVQ-QIM yöntemi varyasyonlarında gizli veri, taşıyıcı bit dizisinin değişikliğe uğratılmasıyla gömülür. Gömülen veri genellikle şifreli olduğundan beyaz gürültüyü andırır. Haberleşme ortam gürültüsü de taşıyıcı bit dizisi üzerinde benzer etkilere sahiptir. Bu yüzden gürültü kaynaklı değişikliklerin veri gömme kaynaklı değişikliklerden ayırt edilmesi mümkün değildir ve haberleşme ortam gürültüsünün olası varlığı steganaliz işlemlerinin başarım oranlarını düşürür. Haberleşme ortam gürültüsünün steganaliz üzerindeki etkilerinin gözlemlenebilmesi amacıyla, test kapsamındaki steganalizörler 10^{-3} ve 10^{-2} bit hatası koşullarında denemeye tabi tutulmuşlardır. Çizelge 3.5'te de sunulan test sonuçlarına göre haberleşme ortam gürültüsündeki artış steganaliz başarım oranlarını düşürmektedir.

Bir MELP-MSVQ-QIM varyasyonu denk olduğu gürültü düzeyinin, çok üzerindeki bir gürültü düzeyi kadar bit değerlerini değiştirir. Burada sayısal olarak ifade etmek gerekirse, QIM-4S-HB varyasyonunu yaklaşık olarak 10^{-3} BER'lik haberleşme ortamı kadar bozulmaya neden olur, ancak 4.5×10^{-2} BER'lik haberleşme ortamı kadar bit değiştirir ($4.5 \times 10^{-2} = 1 / 22.22$). Steganalizde esas olan bozulma değil de sadece değişikliğe uğratılan bit oranı olduğundan, MELP-MSVQ-QIM varyasyonları bozulma açısından eşdeğer oldukları bit hata oranlarının çok üzerinde steganografik iz bırakırlar.

Çizelge 3.5'teki steganaliz sonuçlarına göre MELP-MSVQ-QIM varyasyonlarının tamamı 0.5 değeri dışında başarım oranlarına sahiptirler. Bu sonuçlara göre, eğer bir steganalizci yeteri kadar uzunlukta bir dizi toplayabilirse ve topladığı dizinin tamamı gizli veri içeriyorsa veya içermiyorsa, dizi üzerinde MELP-MSVQ-QIM varyasyonlarının uygulanıp uygulanmadığı tanımlı bir olasılıkta hesaplanabilir. Şekil

3.13'te, MELP-MSVQ-QIM varyasyonlarının çeşitli haberleşme ortam koşullarındaki tespit edilebilme olasılığı vs süre grafikleri verilmiştir.

Şekil 3.13'te grafiksel olarak sunulan steganaliz sonuçlarına göre MELP-MSVQ-QIM varyasyonlarının tespit edilemez olmadıkları ortaya çıkmıştır. Diğer yandan, Kocal vd. (2008)'de steganaliz sonuçlarına göre en fazla gizlilik sağlayan veri gömme yöntemi MP3STEGO (Anonymous 2010) olmuştur. Bahse konu referansta süreleri 3 ila 4 saniye arasında değişen konuşma sinyali örnekleri MP3STEGO ile kapasitelerinin %10'u kadar gizli veriyle doldurulmuşlar, daha sonra kaotik öznitelik steganalizine alınmışlardır. Test sonuçlarına göre MP3STEGO yönteminin uygulandığı örnekler %80'in üzerindeki bir oranda tespit edilebilmiştir. QIM-4S-H4B ise, MELP-MSVQ-QIM varyasyonları arasında en kötü steganaliz ve bozulma sonuçlarını sergileyen varyasyondur. Söz konusu varyasyonda %7.4 oranında gizli veri gömülmektedir ve 3.6 saniyelik bir örnek üzerindeki uygulaması ortalama %69 ihtimalle yakalanabilmektedir. Tüm bunlara karşın, elde edilen steganaliz sonuçlarından yola çıkılarak MELP-MSVQ-QIM – MP3STEGO kıyaslaması yapılması mantıklı değildir. Veri gömme kapasiteleri, taşıyıcı kanal hızları ve veri gömme verimliliği tüm yöntem ve varyasyonlarda değişkenlik gösterebilmektedir. Kıyaslamayı engelleyen bu faktörlerin haricinde MP3 kodlayıcısı (dolayısıyla MP3STEGO) konuşma sinyalinin kodlaması için düşünülebilecek en son tercihtir, asıl vazifesi müzik sinyalinin kodlanmasıdır.



Şekil 3.13 Farklı BER koşulları altında çalışan, farklı veri gömme yöntemleri için süreye karşı steganaliz başarım oranları

3.8 MELP-MSVQ-QIM Yönteminin Genel Bir Değerlendirmesi

Tezin bu bölümünde MELP MSVQ'su üzerinde çalışan gizli veri gömme yöntemleri ele alınmıştır. İlk olarak MSVQ'da kıpırtı benzeri veri gömme yöntemi (Chang ve Yu 2002) gerçeklenmiş, ancak bahse konu yöntemin aşırı derecede bozulmaya yol açtığı, bu yüzden de steganografik kullanımının mümkün olamayacağı belirlenmiştir. Aşırı derecedeki bozulmaya çerçeve başına çok fazla gizli veri biti gömülmesinin neden olduğu anlaşılmıştır. Bunun üzerine kıpırtı benzeri veri gömme yöntemine alternatifler aranmaya başlanmış, bilindik QIM yöntemleri MELP MSVQ'suna uyarlanarak özgün MELP-MSVQ-QIM yöntemi geliştirilmiştir.

MELP-MSVQ-QIM çerçeve başına farklı miktarlarda (Kıpırtı benzeri veri gömme yönteminden daha az) veri gömülmesine izin veren bir yöntemdir. Çalışma esası basittir: Önce gizli verinin saklanacağı indeks bitleri seçilmektedir. Daha sonra, nicemlemede kullanılan tablolar illa indeks bitlerinin gizli veriye eşit olacağı şekilde daraltılmaktadır. LSF vektörlerinin nicemlenmesi bu daraltılmış tablolar vasıtasıyla gerçekleştirilmekte, böylelikle nicemleme sonucunda elde edilen indekslerde gizli verinin yer alması sağlanmaktadır. MELP-MSVQ-QIM yönteminden her biri farklı indekslere veri gömen pek çok varyasyon türetilmiştir.

NTIMIT (Jankowski vd. 1990) veri tabanı kullanılarak, MELP-MSVQ-QIM varyasyonları test edilmiş, her bir varyasyon için ortalama D_{fm}, gizli veri biti başına D_{fm}, spektral bozulma ve P.862 MOS skorları hesaplanmıştır. Söz konusu kalite metriklerine göre QIM-2S-HB, QIM-3S-HB, QIM-4S-HB ve QIM-HS-HB varyasyonları kabul edilebilir sınırlar içerisinde kalmışlardır. Buna karşılık diğer varyasyonlar, QIM-1S-HB ve QIM-4S-H4B varyasyonları yüksek oranda bozulma sergilemişlerdir. MSVQ indekslerine gömülen bit sayısı arttırıldıkça hem D_{fm} ve hem de gizli veri biti başına D_{fm} artmıştır; bu olgu kıpırtı benzeri veri gömme yönteminin kabul edilemez orandaki bozulmasını da açıklamaktadır. Kalite metriklerine dayanan test sonuçlarını özetlemek gerekirse, QIM-4S-HB en az bozulmaya yol açan, QIM-HS-HB gizli veriyi en geniş alan üzerinde saklayan ve QIM-4S-H4B çerçeve başına en fazla gizli veri gömen MELP-MSVQ-QIM varyasyonları olmuşlardır.

Kalite ölçüm testlerinin ardından, sadece üç MELP-MSVQ-QIM varyasyonu (QIM-4S-HB, QIM-HS-HB ve QIM-4S-H4B) kaotik öznitelikler steganalizine (Kocal vd. 2008) sokulmuşlardır. Kaotik özniteliklere göre steganaliz iki aşamadan, eğitim ve test aşamalarından oluşmaktadır, her iki aşamada da NTIMIT veritabanından yararlanılmıştır. Varyasyonların steganalizleri ayrı ayrı gerçekleştirilmiş, elde edilen steganaliz sonuçlarına göre üç varyasyonun da tespit edilemez olmadığı ortaya çıkmıştır. Ancak bu durum Kocal vd. (2008)'de incelenen tüm gizli veri gömme yöntemleri için de geçerlidir. Öte yandan, MELP-MSVQ-QIM varyasyonlarının steganografik izleri Westfeld (2005)'in matris gömmesi, Markov Modellemeli Sendrom Kodlama (Bölüm 5), vb. teknikler kullanılarak kolaylıkla azaltılabilir.

4. KONUŞMA KODLAYICI TANIMA

Konuşma kodlayıcı tanıma, literatüre ilk defa bu tez kapsamında tanıtılmıştır. Konuşma kodlayıcı tanıma internetten, havadan, …vb iletişim ortamlarından aktarılırken yakalanmış veya kayıt ortamlarında saklanmış konuşma sinyaline ait bit dizilerinin çeşitli analiz süreçlerine sokularak, üretimlerinde kullanılan kodlayıcıların tiplerinin tespit edilmesidir.

Sadece bu tezin adından yola çıkılırsa, bu bölümde anlatılan kodlayıcı tanımanın tez içeriğiyle alakası olmadığı izlenimi edinilebilir. Lakin diğer yandan kodlayıcı tanıma bu tezde sıklıkla göreve çağrılan kaotik öznitelikler steganalizinden steganalizör işlevsellik testleri sırasında türetilmiştir. Bu noktada bir parantez açmak gerekirse steganalizör işlevsellik testleri, makalelerde yayımlanmış (ve çoğu yazılım modülleri bizzat makale yazarlarından edinilmiş) kaotik öznitelikler steganalizinin doğru gerçeklenip gerçeklenmediğinin kontrol edildiği testlerdir. Test edilmesi amaçlanan veri gizleme yöntemlerinin steganalizörlere yakalanıp yakalanmayacakları, yakalanırlarsa bile hangi ölçüde yakalanacakları en başta bilinmediğinden, gerçeklenen steganalizör ilk olarak farklı kodlayıcı çıktıları ile test edilmiştir; steganalizörün farklı kodlayıcı çıktılarını yüksek başarı oranıyla ayırt edebilmesi gerektiği varsayılmıştır. Sonra bu fikir daha da genişletilmiş, daha fazla konuşma kodlayıcısının sınıflandırmaya dâhil edilmesiyle kodlayıcı tanıma geliştirilmiştir. Tüm bunların dışında, kodlayıcı tanıma ile kaotik öznitelikler steganalizi birbirlerine oldukça benzeyen çalışma esaslarına sahiptirler, bahse konu bu benzerlik olgusu bile kodlayıcı tanımanın tez kapsamına alınması için yeterli gerekçe sunar.

Genellikle video, audio veya konuşma haberleşmelerinde kullanılacak kodlayıcıların tipine, haberleşme kanallarının kurulması esnasında karar verilmektedir. Karşılıklı uçlar arası haberleşmeyi gözlemleyen ve bağlantı kontrol protokolünü çözümleme yetisine sahip herhangi birisi, aktarım esnasında kullanılan – kullanılacak kodlayıcının tipini kolaylıkla açığa çıkartabilir. Kayıt ortamlarında ise kullanılan kodlayıcılara ait bilgiler, bit dizilerinin saklandığı dosyaların başlık bölümlerinde yer alabilir. Bu olgulara ilaveten, gerçek dünyadaki haberleşme altyapılarında dokümana dökülmüş –

dökülmemiş pek çok bağlantı kontrol protokolü türevi kullanılmaktadır, tüm bu türevlerin teker teker incelenmesi ve her biri için ayrı bir çözümleme yeteneğinin kazanılması son derece külfetlidir. Kayıt ortamlarında da kodlanmış bit dizilerinin başlıksız veya sahte başlıklarla saklandığı durumlar hayli yaygındır. Tezin bu bölümünde sadece kodlanmış konuşma sinyaline ait bit dizisini analiz ederek kullanılan kodlayıcının tipini belirleyebilen özgün bir yöntem tanıtılmıştır.

Değişik kodlayıcılara ait olan çıktıların istatistiksel özelliklerinin birbirlerinden farklı olacağı öngörüsünde bulunulursa, bilinmeyen bit dizilerinin hangi kodlayıcılarca üretildiği konusu küçük istatistikî farklılıkları bile tespit edebilen steganaliz yöntemlerince çözüme kavuşturulabilir. Standart bir steganalizörün eğitim fazında, ilk önce gizli veri içeren ve içermeyen veri kümeleri için ayrışan istatistiksel özellikler belirlenir. Daha sonra ayrışan istatistiksel özellikleri farklı kümeleri maliyet fonksiyonuna göre en iyi ayrıştıran eşik değerleri hesaplanır. Sınıflandırıcı tasarımının doğası gereği, steganalizörce verilen kararlar ya doğru pozitif, ya doğru negatif, ya yanlış pozitif ya da yanlış negatiftir. Bu tezde önerilen konuşma kodlayıcısı tanımlayıcısı Kocal vd. (2009) tarafından önerilen konuşma steganalizinden türetilmiştir; bahse konu yöntem kaotik özniteliklerin ayrışması üzerine kuruludur ve gizli veri ekleme işlemlerinin en yakın hatalı komşuluk oranlarını ve Lyapunov katsayılarını arttıracağı saptamasına dayanmaktadır.

Kodlayıcı tanımlayıcısının temel görevi, herhangi bir şüpheli bit dizisinde kullanılagelen kodlayıcı tipini belirlemektedir. Tanıma prosedürü, ayırt edici pek çok bilginin işlenmesiyle yerine getirilebilir. Hâlbuki çoğu senaryoda her türlü bilginin toplanması, özünün çıkartılması ve dönüşümlere uğratılması mümkün ve uygun olmayabilir. Bu yüzden olabildiğince basit, pratik açıdan gerçekleştirilebilir ve olabildiğince genel bir kodlayıcı tanıma modelinin tasarımına ihtiyaç vardır. Bahse konu niteliklere sahip genel bir modelin tasarlanabilmesi için aşağıda listelenmiş olan çalışma esasları belirlenmiştir:

İstatistikî farklılıkların tespiti: Tüm kodlayıcıların birbirlerinden farklı istatistikî özelliklere sahip çıktılar ürettiği-üreteceği varsayılmıştır. Ufak uygulama farklılıklarına

karşın aynı istatistikî özellikle sahip kodlayıcılar aynı sınıf içerisinde değerlendirilecektir.

Bit dizisi üzerinden analiz: Tüm kodlayıcı çıktılarının *"unsigned byte"*'lardan oluşan bit dizileri ürettikleri varsayılmıştır. Analiz edilecek bit dizileri sekizerlik gruplar halinde, baytlar halinde işlenecektir. Bit dizilerinin bayt dizilerine dönüştürülmesinin ertesinde, baytlar sırayla gerçek sayılara çevrilecek, istatistiksel analizler bu gerçek sayılar üzerinden yerine getirilecektir. Elbette çoğu kodlayıcı tarafından üretilen bit dizilerinin çeşitli kısımlarında anlamlı tam sayılar ve/veya gerçek sayılar yer almaktadır, ancak hangi bitlerin hangi kısımlarının hangi anlamlı değişkenleri oluşturduğunun söylenebilmesi için zaten hâlihazırda kullanılan kodlayıcının türünün biliniyor olmasını şart koşmaktadır.

Veri hızından bağımsızlık: Değişik kodlayıcılar, birim zamanda değişik miktarlarda bit dizisi üretmektedirler. Birim zamanda üretilen bit miktarı bile, kullanılan kodlayıcının ne olduğunun belirlenmesini sağlayabilir veya olası kodlayıcıların sayısının azaltılmasını mümkün kılabilir. Kodlayıcı sınıflandırıcısı, birim zamanda üretilen bit miktarını dikkate almadan, bilinmediğini varsayarak çalışır, girdi olarak tanımlı çerçeve boyu miktarındaki diziyi işler, bir çerçeve boyu bilginin ne kadar zamanda toplandığı ile ilgilenmez. Esasen, kayıt ortamlarında saklanan başlıksız bit dizilerinde zaman bilgisi yer almaz, başka bir yönden internet gibi paket anahtarlamalı bazı şebekelerde paketler farklı rotalardan iletilebildiklerinden, veri hızının tespit edilmesi kolay olmayabilir.

Bit yerleşimine dair bilgilerden yoksunluk: Bazı kodlayıcıların çıktıları, mantıksal olarak birbirleri ile çeşitli şekillerde ilişkilendirilmiş kısımlar içerebilir. İleri hata kodlamaları, her çerçevede artan senkronizasyon sayıları bit dizileri içerisinde yer alabilir. Ayıklanabilen ilişkilere göre kodlayıcı sınıflandırması yapılabilir. Elbette sınıflandırmanın bu şekilde yapılabilmesi için, sınıflandırılacak her bir kodlayıcının ayrı ayrı incelenmesi, ilişkileri tespit edilebilen özelleştirilmiş yazılım modüllerinin hazırlanması lüzumunu doğurur. Bu çalışmada geliştirilen konuşma kodlayıcısı sınıflandırıcısı ise, mantıksal ilişkilerden yararlanmaya çalışmaz; aslında diziler

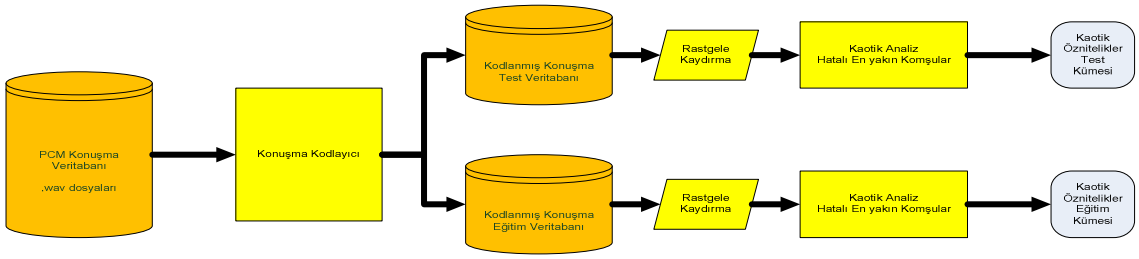
içerisinde yer alan mantıksal ilişkiler belli oranlarda istatistiksel özelliklere nüfuz eder, bu sayede kodlayıcı sınıflandırıcısı dolaylı olarak mantıksal ilişkileri kullanmış olur.

Başlangıç ve son bilgilerinden yoksunluk: Analiz işlemi, bit dizisinin herhangi bir noktasından başlayarak gerçekleştirilebilir. Kodlayıcının çerçevelerine senkronize olmak gibi bir gereksinim yoktur. Örneğin iki ucun haberleşmesi sırasında herhangi bir andan itibaren kaydedilen bit dizileri üzerinde derhal kodlayıcı tanımları analizleri uygulanabilir.

4.1 Genel Sınıflandırma Modeli

Konuşma kodlayıcı tanıma uygulamasının ardında yatan ana fikir oldukça basittir: "Farklı kodlayıcılar farklı istatistikî özelliklere sahip çıktı üretirler. Çıktılardaki istatistiksel farkları ayırt etme kabiliyetine sahip herhangi bir araç, kodlayıcı tipinin tespit edilmesinde kullanılabilir. Steganalizörler istatistiksel farklara oldukça duyarlıdırlar ve bu yüzden kodlayıcı tanımada görev alabilirler."

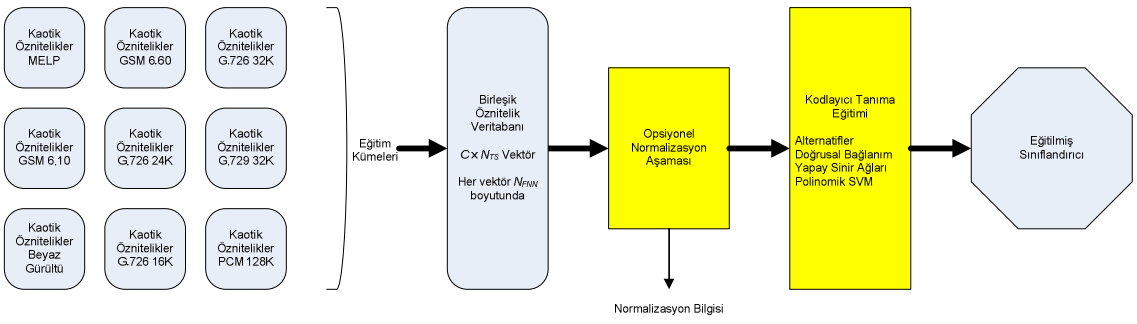
Bu tez kapsamında önerilmiş olan, konuşma kodlayıcı tanıma yöntemi yukarıda yazıya dökülmüş yaklaşım doğrultusunda Kocal vd. (2009) tarafından önerilen konuşma steganalizi yönteminden türetilmiştir. Aşağıda yer alan Şekil 4.1'de, her bir kodlayıcı için test ve eğitim kümelerinin nasıl elde edildiği gösterilmiştir. Test ve eğitim kümelerinin hazırlanması prosedürü her bir kodlayıcı için tekrarlanmakla, sonuç olarak tüm kodlayıcılar için biri eğitim diğeri test amaçlı değerlendirilecek olan iki adet kaotik öznitelikler kümesi elde edilmektedir.



Şekil 4.1 Herhangi bir kodlayıcı için eğitim ve test kümelerinin hazırlanması

Şekil 4.1'de verilmiş blok şemadan detaylı olarak bahsetmek gerekirse, ilk olarak genel bir konuşma sinyali veri tabanı test ve eğitim amaçlı olarak ikiye bölünerek kodlanır, bu sayede iki alt-veritabanı elde edilir: XXX kodlayıcısı için kodlanmış konuşma sinyali test veritabanı ve XXX kodlayıcısı için kodlanmış konuşma eğitim veritabanı. Alt-veritabanlarının oluşturulması takiben iki alt-veritabanında yer alan bit dizilerinin kaotik öznitelikleri hesap edilir. Yalnız, analize sokulmadan önce kaotik öznitelikleri hesaplanacak her bit dizisinin başlangıç kısmı rastgele büyüklükte atılır, kaotik özniteliklerin hesaplanması bu kısaltma işlemi sonrasında vuku bulur. Böylece kodlayıcı tanımlayıcısı tasarım kriterlerinden başlangıç ve son bilgilerinden yoksunluk şartı yerine getirilir.

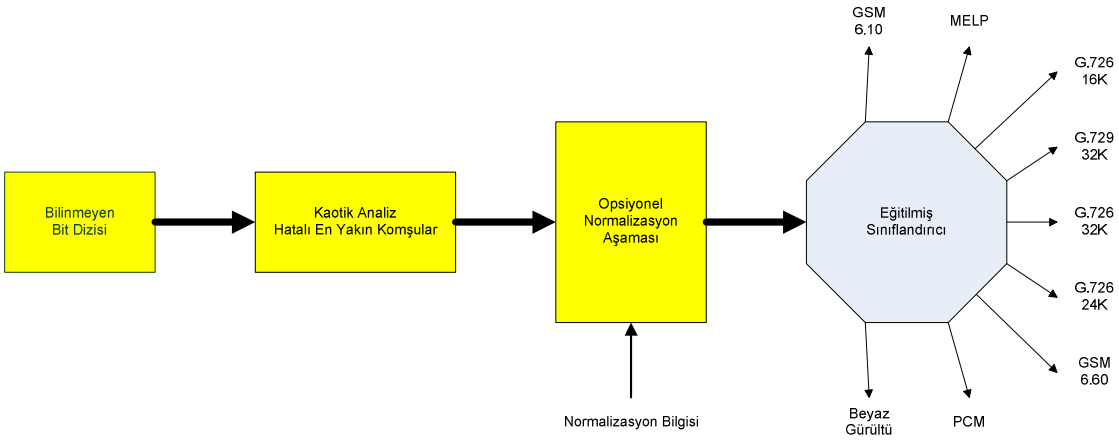
Tüm kodlayıcılara ait kaotik özniteliklerden kurulu eğitim ve test kümelerinin hazırlanmasının ardından, kodlayıcı tanımlayıcısının eğitilmesine başlanır. Aslında kodlayıcı tanımlayıcısı (opsiyonel) normalize edilmiş girdilerle çalışan standart bir sınıflandırıcıdan ibarettir. Sınıflandırıcının eğitimi sınıflandırıcı türüne göre değişkenlik gösterir; bir doğrusal bağlanım sınıflandırıcısında eğitim süreci yalnızca bazı matris hesaplamalarından oluşmaktayken, yapay sinir ağı tabanlı bir sınıflandırıcının eğitim süreci geri yayılım algoritmasının çalıştırılması üzerine kuruludur. Şekil 4.2'de sınıflandırıcı eğitiminin şematik bir özeti sunulmuştur.



Şekil 4.2 Genel sınıflandırma modelinin eğitimi

Bir kere eğitilmiş sınıflandırıcıya sahip olunduktan sonra, bir dizinin kodlayıcı tipinin öğrenilmesi için yapılacak ilk işlem, bahse konu dizinin kaotik öznitelikleriniin

hesaplanmasıdır. Daha sonra hesaplanan öznitelikler (gerekirse) normalize edilerek eğitilen sınıflandırıcıya sokulur. Sınıflandırıcı da girdi özniteliklere göre sınıflandırma işlemini yerine getirir (Şekil 4.3).



Şekil 4.3 Eğitilmiş genel sınıflandırma modeli ile sınıflandırma

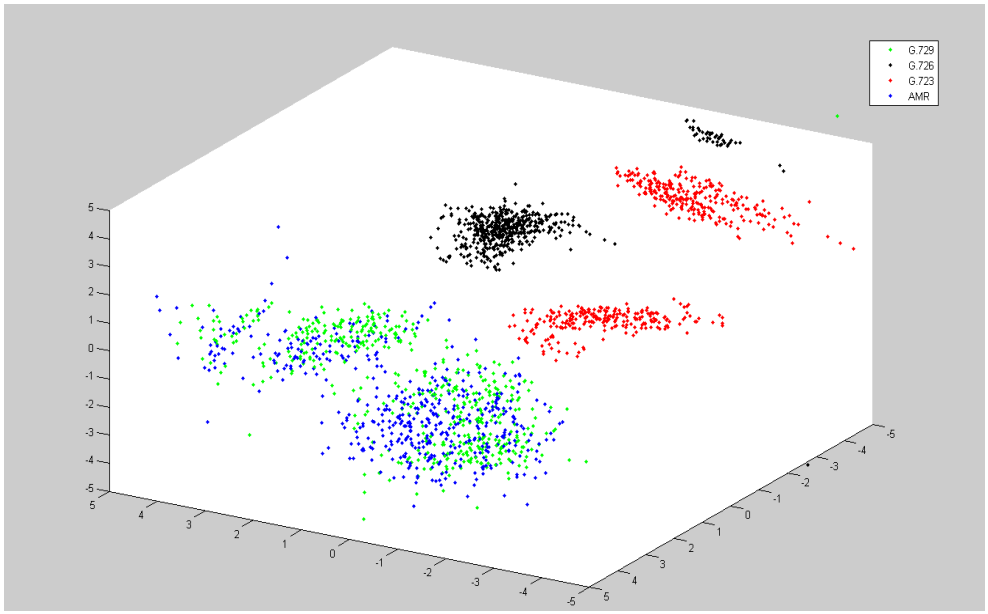
4.2 Kodlayıcı Tanımada Kullanılan Özniteliklerin Seçimi

Kocal vd. (2009) yaptıkları çalışmada, gizli veri içeren ve içermeyen bit dizilerinin, hatalı en yakın komşular oranı (FNF) ve Lyapunov katsayıları gibi kaotik özniteliklerinin birbirlerinden farklılaşacağını ortaya koymuşlardır. Bu tespitlerden yola çıkılarak, aynı kaotik özniteliklerin farklı kodlayıcı çıktıları için de ayrışacağı, hatta istatistiksel farklılıkların daha fazla olmasından dolayı daha fazla ayrışacağı düşünülebilir.

Bu tezin kronolojik gelişimi sırasında, ilk olarak QIM yöntemlerinin steganografik performansları ölçülmüş, bahse konu ölçümlerin yapılabilmesi için Kocal vd. (2009)'den steganaliz yöntemine ait kaynak kodları temin edilmiştir. Steganalizörlerin amaçları dışında da değerlendirilebilecekleri, farklı kodlayıcılara ait çıktıların steganalizörlerce sınıflandırılabileceği hipotezi ilk kurulduğunda, kodlayıcı tanımanın hayata geçirilebilmesi için zaten her türlü araç, gereç ve yazılım toplanmış olduğu

anlaşılmıştır; bu yüzden teorik bir altyapı oluşturulması beklenmeksizin derhal testlere başlanmıştır.

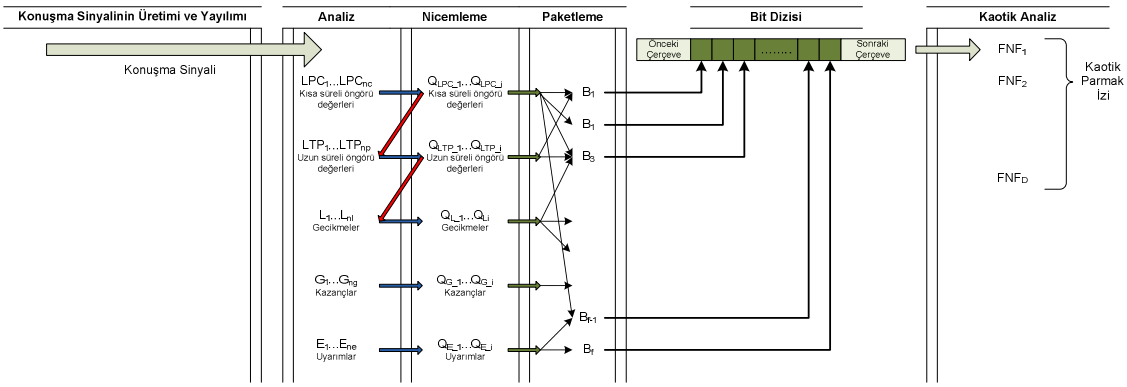
Kodlayıcı sınıflandırıcısının ilk testlerinde, mevcut steganalizör tasarımı aynen 2'li kodlayıcı sınıflandırıcısı olarak kullanılmıştır. Gerçekleştirilen testlerin sonucunda son derece tatmin edici sonuçlara ulaşılmış, çok yüksek başarım oranlarının getirdiği güvenle kodlayıcı sınıflandırıcısı üzerinde sadeleştirme işlemleri bile uygulanmıştır. Sadece FNF sonuçları ile de yüksek başarım oranlarına ulaşılabildiği anlaşılmış, Lyapunov katsayılarının değerlendirme dışı bırakılmasıyla bir yandan sınıflandırıcının karmaşıklığı azaltılırken diğer yandan çalışma hızı arttırılmıştır. Bir sonraki aşamada sınıflandırmanın nasıl olup da bu kadar iyi sonuçlandığı sorgulanmıştır. Farklı kodlayıcılara ait FNF kaotik öznitelikleri üzerinde ana bileşen analizleri tatbik edilmiş, Şekil 4.4'te de gösterildiği gibi, test kapsamındaki kodlayıcıların G.726 24K (G.723), G.726 16K, G.729 ve GSM 6.60 (AMR) özniteliklerinin birbirlerinden ayrıştığı gözlemlenmiştir.



Şekil 4.4 Sammon analizi sonunda elde edilmiş 3 boyutlu vektör dağılımı

4.3 Genel Konuşma Kodlayıcısı Modeli

Kodlayıcı tanımanın gerçek dünyada gösterdiği başarının tasdikinden sonra, önerilen yöntemin teorik izahatına odaklanılmıştır. Farklı konuşma kodlayıcı çıktılarının neden farklı özniteliklere malik olduklarının gösterilebilmesi için önce söz konusu çıktıların üreten kodlayıcılar dikkate alınmalıdır. Sabit bit hızında (*CBR: constant bit rate*) çıktı üreten konuşma kodlayıcıları Şekil 4.5'te sunulan genel bir modelle tarif edilebilirler.



Şekil 4.5 Genel konuşma kodlayıcı modeli

Modeli meydana getiren alanları sırayla açıklamak gerekirse;

Konuşma sinyali: Rastgele değildir, kısa ve uzun süreli korelasyonlar içerir, frekans bölgesinde formant yapısındadır, kaotik özellikler de taşır.

LPC (doğrusal öngörü katsayıları): Konuşma sinyalinin biriktirildiği çerçeveler üzerinden hesaplanır. Sinyaldeki kısa süreli korelasyonları betimler. Ardı sıra gelen çerçevelerin LPC'leri arasında korelasyon bulunur.

LTP (Uzun süreli öngörü) katsayıları: Konuşma sinyalinin biriktirildiği çerçevelerin alt-çerçeveleri üzerinden hesaplanır. Sinyaldeki uzun süreli korelasyonları betimler. Ardı sıra gelen çerçevelerin LTP'leri arasında korelasyon bulunur.

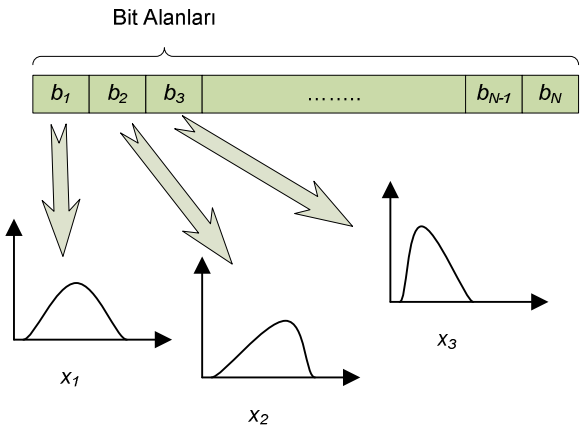
Uyarım, kazanç ve gecikmeler: Konuşma sentezleme modeline göre LTP ve LPC süzgeçlerine girilen girdileri, çarpanları, vb. belirtirler.

Analiz sırasında elde edilen tüm bu değerler, rastgele değillerdir kendilerine her birinin kendisine özgü – kullanıcıya ve konuşma sinyalinin içeriğine göre değişen istatistiksel dağılımı vardır.

Nicemlenmiş analiz değerleri: Analiz sırasında elde edilen değerlerin, çeşitli nicemleyicilerce (vektör nicemleyiciler, çok seviyeli vektör nicemleyiciler, uyarlanabilir nicemleyiciler… vb) indeks değerlerine kodlanmış halidir. Değerlerin istatistiksel dağılımın özelliklerini azaltılmış - belli oranda yansıtırlar; kodlama verimliliği arttıkça beyaz gürültüyü andırırlar. CBR kodlayıcılar için kesinlikle çerçeveler arası korelasyonlar tanımlanabilir.

Bit dizisi: Nicemlenmiş analiz değerlerinin parçalanıp - bir araya getirilmesiyle oluşturulur. Her bit alanının nicemlenmiş değerlerden gelen karma - kendine özgü istatistiksel özellikleri vardır. CBR kodlayıcılar için çerçeveler arası korelasyonlar tanımlanabilir.

Şekil 4.5'de bit dizisi olarak gösterilen, Şekil 4.6'te de büyütülmüş olarak sunulan kodlanmış konuşma sinyali kendini yineleyen N bayt uzunluğundaki çerçevelerden oluşmaktadır. Her bir bit alanının, b_i, (veya her bir baytın) kendine özgü olarak bir olasılık yoğunluk fonksiyonu (oyf), x_i bulunmaktadır. Gerçekte N değerinin kodlayıcı tanımlayıcısı açısından bir önemi yoktur; N sadece kodlayıcı tanıma yönteminin matematiksel izahatında faydalanılan varsayımsal bir değerdir. Öte yandan, bilinmeyen bir bit dizisinin analizi sırasında çeşitli öz-ilinti yöntemleri kullanılarak olası N değerleri hesaplanabilir ve daha sonra hesaplanan bu değerler bit dizisini üreten kodlayıcının tanımlanmasında yararlı olabilir.



Şekil 4.6 Kodlanmış çıktının farklı dağılımlı bit alanlarından oluşması

Bahse konu çerçeve boyu her zaman kodlayıcı çıktısının tanımlı çerçeve boyuna eşit değildir; çünkü kodlayıcı çıktısındaki *Ç* bitlik çerçeve boyu 8'in katlarına hizalanmamış olabilir; bu durumda *N* *EKOK*(*Ç*, 8)/8 değerine eşit olur. (*EKOK*(y_1,y_2, … y_n): y_1,y_2, … y_n sayılarının en küçük ortak katı) Bazı kodlayıcıların çıktıları ise zaten çerçeveler halinde değildir, bu tip kodlayıcılar için *N* değeri 1'dir. Bazı VBR (*VBR: variable bit rate*, değişken bit hızı) özellikli konuşma kodlayıcılarında ise çerçeve boyları değişken olabilir; bu tip kodlayıcılarda *N* değerinin 1'e veya kullanılan çerçeve boylarının en büyük ortak bölenine eşit olduğu varsayılabilir.

4.4 FNF Değerlerinin Hesaplanması

Bu bölümde sırasıyla ilk olarak bit dizilerinin FNF hesaplamasına nasıl hazır hale getirileceği anlatılacak, daha sonra FNF değerlerini belirleyen matematiksel ilişkiler ortaya konacaktır. Matematiksel ilişkilerin ortaya konmasını takiben, bazı varsayımların gerçekleşmesi durumunda, söz konusu matematiksel ilişkilerin nasıl sadeleştirilebildiği gösterilecek, çok basit örnekler için teorik FNF değerleri hesaplanacaktır. Bu bölümde son olarak, farklı konuşma kodlayıcılarının FNF değerlerinin neden ve nasıl farklılık arz ettiği açıklanacaktır.

4.4.1 Bit dizisinin FNF hesabına uygun hale getirilmesi

Herhangi bir haberleşme ortamından yakalanarak bitler halinde kaydedilmiş, FNF değeri hesaplanacak bit dizisi ilk olarak komşu bitleri sekizerlik gruplar halinde bir araya getirilerek bayt dizisine dönüştürülür. Daha sonra dâhili değerleri 0 ila 255 arasında değişen bayt dizisi normalizasyon işlemine alınarak değerleri -1 ila 1 bit arasında değişen gerçek sayı $S(n)$ dizisine çevrilir. FNF değerleri elde edilen bu gerçek sayı dizisi üzerinden hesaplanır.

4.4.2 FNF'si hesaplanacak bit dizisinin çözümlenmesi

FNF'si hesaplanacak gerçek sayı dizisi bölüm 4.2.1'de verilen genel konuşma kodlayıcısı modeline göre yine gerçek sayılardan oluşan alt-dizilere çözümlenebilir. Genel konuşma kodlayıcısı modeline bakılırsa, kodlayıcı çıktısı olan bir bit dizisi N baytlık yinelemeler gösterir. Bahse konu bayt değerleri esas alınarak hesaplanmış olan $S(n)$ gerçek sayı dizisi denklem (4.1)'deki gibi ayrıştırılabilir:

$$\begin{aligned} S(n) &= \{S(0), S(1), .., S(N), S(N+1), S(N+2), .., S(2N), \dots, S(L)\} \\ X_1(n) &= \{S(0), S(N), S(2N), \dots..\} \\ X_2(n) &= \{S(1), S(N+1), S(2N+1), \dots..\} \\ X_N(n) &= \{S(N-1), S(2N-1), S(3N-1), \dots, S(L)\} \end{aligned} \tag{4.1}$$

Bu işlem sonucunda $S(n)$ gerçek sayı dizisi N adet $X_{\mathrm{i}}(n)$ alt-dizisine bölünmüş olur. $X_{\mathrm{i}}(n)$ alt-dizisinin oyf'si ise x_{i} olarak tanımlanmıştır.

Şüphesiz N yineleme sayısının değeri kodlayıcının türüne bağlıdır, kodlayıcı tanıma uygulaması sırasında girdi dizinin hangi kodlayıcı tarafından üretildiği bilinmediğinden, N değerinin de bilinmesi mümkün değildir. Hakikatte, kodlayıcı tanıma işleminin icrası sırasında kati suretle (4.1)'deki çözümlemeye ihtiyaç duyulmaz; bahse konu çözümleme yalnızca kodlayıcı tanımanın teorik izahatı açısından fayda sağlayacaktır.

4.4.3 Hatalı komşuluk testi

Bir dizinin *d*. boyuttaki iki noktası $S(n)$ ve $S^{(r)}(n)$ arasında komşuluk ilişkisi olup olmadığı bahse konu noktaların birbirlerinden faz alanındaki uzaklıklarına göre belirlenir. *d*. boyutta $S(n)$ ve $S^{(r)}(n)$ noktalarının faz alanındaki yerleri (4.2)'deki gibi tanımlanmıştır (Kennel vd. 1992):

$$\begin{aligned} S(n) &= \big(S(n), S(n+T), .., S(n+dT-T)\big) \\ S^{(r)}(n) &= \big(S(n+r), S(n+T+r), .., S(n+dT+r-T)\big) \end{aligned} \tag{4.2}$$

(4.2)'de sunulan *T* değeri gecikme değeri olup *r* sembolü ise $S^{(r)}(n)$ noktasının $S(n)$ noktasına göre görece pozisyonunu gösterir. Faz alanında yerleri (4.2)'deki gibi tanımlanmış *d* boyutlu iki nokta arasındaki Kartezyen uzaklık, denklem (4.3) ile hesaplanabilir (Kennel vd. 1992):

$$R_d^2(n,r) = \sum_{k=0}^{d-1} \big[S(n+kT) - S^{(r)}(n+kT)\big]^2 \tag{4.3}$$

(4.3)'te verilmiş olan $R_d(n,r)$ noktalar arasındaki Kartezyen uzaklığıdır. $R_d^2(n,r)$ise noktalar arasındaki Kartezyen uzaklığı karesidir. İki nokta arasında *d*. boyutta komşuluk ilişkisi olduğuna karar verilebilmesi için iki koşulun yerine getirilmesi gereklidir.

Birinci koşul (Kennel vd. 1992), (4.4)'te $DF1_d(n,r)$ fonksiyonu ile ifade edilmiştir; esasen $DF1_d$ fonksiyonu $S^{(r)}(n)$ ve $S(n)$ noktaları için *d*. boyuttaki uzaklığın *d*+1. boyutta ne kadar arttığını kontrol etmektedir.

$$DF1_d(n,r) = \begin{cases} 0 \quad eğer & \left(\dfrac{\big(S(n+Td) - S^{(r)}(n+Td)\big)^2}{R_d^2(n,r)} \geq R_{tol}^2 \right) \\ 1 \quad eğer & \left(\dfrac{\big(S(n+Td) - S^{(r)}(n+Td)\big)^2}{R_d^2(n,r)} < R_{tol}^2 \right) \end{cases} \tag{4.4}$$

Eğer faz alanındaki uzaklık sonraki boyutta belli bir limitin üzerinde artış sergilemişse, komşuluk ilişkisinin bozulduğu hükmüne ulaşılabilir. Denklemde kullanılan R_{tol} değeri sabittir ve küçük değerleri komşuluk ilişkisi oluşmasını zorlaştırırken, büyük değerleri kolaylaştırır. Pratik uygulamalarda genellikle R_{tol} 10'dan büyük değerler arasından seçilir.

İkinci koşul (Kennel vd. 1992), denklem (4.5)'te $DF2_d(n,r)$ fonksiyonu ile ifade edilmiştir; $DF2_d(n,r)$ fonksiyonu kısaca $S^{(r)}(n)$ ve $S(n)$ noktalarının *d*. boyuttaki komşuluklarının geçerli olup olmadığını sınamak için *d*+1. boyuttaki uzaklığı dizi varyansıyla karşılaştırır. Eğer faz alanındaki uzaklık kodlanmış dizinin varyansını A_{tol} misli aşarsa komşuluk ilişkisinin artık mevcut olmadığını farz edilir.

$$DF2_d(n,r) = \begin{cases} 0 & eğer \quad \left(\frac{R_{d+1}(n)}{R_A} \geq A_{tol}\right) \\ 1 & eğer \quad \left(\frac{R_{d+1}(n)}{R_A} < A_{tol}\right) \end{cases} \tag{4.5}$$

A_{tol} de tıpkı R_{tol} sabit bir sayıdır ve genelde 2'den büyük eşit değerler arasından seçilmektedir. Küçük değerleri komşuluk ilişkisi oluşmasını zorlaştırırken, büyük değerleri kolaylaştırır. (4.6)'da dizi varyansının nasıl hesaplandığı gösterilmiştir. *L* değeri varyansı hesaplanacak dizinin toplam uzunluğudur.

$$R_A^2 = \frac{1}{L}\sum_{n=1}^{L}[S(n) - S]^2 \qquad \bar{S} = \frac{1}{L}\sum_{n=1}^{L} S(n) \tag{4.6}$$

Dizi içerisinde yer alan iki noktanın *d*. boyutta birbirleriyle komşu kabul edilmesi için her iki koşul da mutlak yerine getirilmelidir (Kennel vd. 1992). (4.7)'deki $FN_d(n,r)$fonksiyonu *d*. boyut için $S(n)$ ile $S^{(r)}(n)$ noktalarının komşuluk ilişkilerini belirtmektedir.

$$FN_d(n,r) = \begin{cases} 1 & eğer \quad DF1_d(n,r) = 1 \; ve \; DF2_d(n,r) = 1 \\ 0 & Diğer \; tüm \; durumlar \end{cases} \tag{4.7}$$

$S(n)$ ile $S^{(r)}(n)$ noktalarının birbirleriyle d. boyutta hatalı komşu olabilmeleri için $FN_{d-1}(n,r)$ 1'e eşit olurken $FN_d(n,r)$ 0'a eşit olmalıdır; yani bahse konu iki nokta $d-1$. boyutta birbirleriyle komşu olurlarken, d. boyutta ise bu ilişkiyi kaybetmiş olmalıdırlar.

Bir dizinin d. boyuttaki hatalı komşuluk oranına FNF_d denir (Kennel vd. 1992). FNF_d ifadesinin matematiksel karşılığı denklem (4.8)'de gösterilmiştir. $\#FN_d$ dizi içerisinde yer alan tüm $S(n)$ ve $S^{(r)}(n)$ noktaları için d. boyuttaki toplam hatalı olmayan komşu sayısıdır.

$$FNF_d = \frac{\#FN_{d-1} - \#FN_d}{\#FN_{d-1}} \qquad \forall S(n), S^{(r)}(n) \in b_1, b_2 \dots b_N \tag{4.8}$$

4.4.4 Genel konuşma kodlayıcısı modeline göre FNF hesaplama

Kennel vd. (1992)'de sunulan FNF_d tanımı ile genel konuşma kodlayıcısı modeli bir araya getirilirse; denklem (4.9)'da gösterildiği üzere, çerçeveler halinde olduğu varsayılan kodlanmış konuşma dizisinin d. boyuttaki hatalı komşuluk oranı, diziyi oluşturan tüm bit alanlarının d. boyuttaki hatalı komşuluk oranlarının bir fonksiyonudur. Denklemde verilen FNF_{d,b_i} i. bit alanının d. boyuttaki hatalı komşuluk oranıdır.

$$FNF_d = f\left(FNF_{d,b_1}, FNF_{d,b_2}, \dots, FNF_{d,b_N}\right) \tag{4.9}$$

i. bit alanının d. boyuttaki hatalı komşuluk oranını belirten FNF_{d,b_i} değeri, dizinin sadece i. bit alanındaki noktaların dizinin tüm bit alanlarındaki noktalara olan hatalı komşuluk oranıdır. Denklem (4.10), denklem (4.8)'in FNF_{d,b_i} için yeniden yazılmış bir halidir; yeni denklemin (4.8)'den en önemli farkı $S(n)$ noktalarının sadece i. bit alanından seçilirken $S^{(r)}(n)$ noktalarının tüm bit alanlarından seçilebilmeleridir. Yine

aynı denklemde kendine yer bulmuş olan $\#FN_{d,b_i}$ değeri, dizi içerisinde sadece *i*. bit alanında yer alan tüm $S(n)$ noktalarının dizi içerisindeki tüm bit alanlarında yer alabilen tüm $S^{(r)}(n)$ noktaları ile *d*. boyutta kurduğu toplam hatalı olmayan komşuluk sayısıdır.

$$FNF_{d,b_i} = \frac{\#FN_{d-1,b_i} - \#FN_{d,b_i}}{\#FN_{d-1,b_i}} \tag{4.10}$$

$$\forall S(n) \in b_i \text{ ve } \forall S^{(r)}(n) \in b_1, b_2 \dots b_N$$

(4.10)'da kullanılan $\#FN_{d,b_i}$ değeri şartları sağlayan $S(n)$ ve $S^{(r)}(n)$ noktalarının $FN_d(n,r)$ sonuçlarından hesaplanır. $S(n)$ noktaları yalnızca *i*. bit alanıyla sınırlandırılmış olduklarından bunların komşuluk ilişkilerini sınayan $FN_d(n,r)$ fonksiyonu $FN_{d,b_i}(n,r)$ olarak isimlendirilmiştir. $FN_{d,b_i}(n,r)$ fonksiyonu (4.4)'teki $DF1_d(n,r)$ ile (4.5)'teki $DF2_d(n,r)$ fonksiyonlarının sonucudur, her iki fonksiyon ise $R^2_{d+1}(n,r)$ ile $R^2_d(n,r)$ uzaklıklarına bağlıdır. Bu durumda denklem (4.11)'de matematiksel olarak gösterildiği üzere $FN_{d,b_i}(n,r)$ $R^2_{d+1}(n,r)$ ile $R^2_d(n,r)$ uzaklıklarının bir fonksiyonu olmaktadır. $R^2_{d+1}(n,r)$ ile $R^2_d(n,r)$ fonksiyonları sadece $S(n)$'nin *i*. bit alanı için hesaplandıklarından, söz konusu iki fonksiyon $R^2_{d+1,b_i}(n,r)$ ve $R^2_{d,b_i}(n,r)$ olarak yeni baştan isimlendirilebilir.

$$FN_{d,b_i}(n,r) = g\left(R^2_{d+1,b_i}(n,r), R^2_{d,b_i}(n,r)\right) \tag{4.11}$$

Denklem (4.12)'de yer alan r_{d,b_i} faz alanındaki $R^2_{d,b_i}(n,r)$ uzaklıklarının oyf'sidir. Bahse konu r_{d,b_i} oyf'si bit alanlarının x_i oyf'lerinin bilinmeyen bir *h* fonksiyonuna göre ortak dağılımıdır. Diğer bir yaklaşıma göre ise r_{d,b_i} oyf'si, *i*. bit alanının diğer bit alanlarıyla arasındaki faz uzaklık oyf'lerinin bilinmeyen bir *p* fonksiyonuna göre ortak dağılımıdır. Denklemde verilen $r_{d,i,j}$ *i*. bit alanının *j*. bit alanıyla arasındaki uzaklığın oyf'sidir.

$$r_{d,b_i} = h(x_1, x_2, .. x_i, .., x_N) = p(r_{d,i,1}, r_{d,i,2}, \dots, r_{d,i,i}, \dots, r_{d,i,N}) \quad (4.12)$$

Eğer *T*=1 kabul edilirse bu sefer $r_{d,i,j}$ oyf'si $x_i, x_{i+1}, \dots, x_{i+d-1}, x_j, x_{j+1}, \dots, x_{j+d-1}$ bit alanı oyf'lerinin bilinmeyen bir q fonksiyonuna göre ortak dağılımı olur. (4.13)'ün dikkate değer tek özelliği $r_{d,i,j}$ oyf'sinin tüm x_i oyf'lerine değil, sadece bazı x_i oyf'lerine bağımlı olmasıdır. *T*'nin 1 değerinden farklı olduğu durumlarda bu kabul geçerlidir, sadece bağımlı olunan x_i oyf'lerinin *i* indeksleri farklıdır.

$$r_{d,i,j} = q\left(x_i, x_{i+1}, \dots, x_{i+d-1}, x_j, x_{j+1}, \dots, x_{j+d-1}\right) \quad (4.13)$$

Yukarıda sunulmuş olan denklemlerde, yalnızca genel konuşma kodlayıcısı modeline göre FNF değerlerinin teorik hesaplaması için gerekli olan çeşitli matematiksel ilişkiler ortaya konulmuştur; söz konusu denklemler bit alanlarına ait oyf'ler ve ortak dağılım fonksiyonları bilinmediği sürece çözülemezler, sadeleştirilemezler. Öte yandan, bir kodlayıcı çıktısını oluşturan bit alanlarının dağılımları ve birbirleri ile olan ilişkileri matematiksel olarak ifade edilebilse dahi FNF değerlerinin teorik olarak hesaplanması oldukça zordur. Çerçeve içerisinde bazı bit alanlarının birbirlerinden bağımsız olduklarının varsayılması ortak dağılım fonksiyonlarının sadeleştirilmesini temin edebilir. Takip eden bölümde, bahse konu bit alanlarının birbirlerinden bağımsız olduğu durum ele alınacak, ortak dağılım fonksiyonlarının nasıl sadeleştirilebileceği anlatılacaktır.

4.4.5 Tüm bit alanlarının birbirlerinden bağımsız olması durumu

Eğer konuşma sinyalinin bit alanlarının dağılımlarının hepsi birbirlerinden bağımsızsa, bir önceki bölümde verilen matematiksel ilişkilerde sadeleştirmelerin yapılması ve bu sayede FNF değerlerinin teorik olarak hesaplanması mümkündür. Denklem (4.14)'te *T*=1 durumunda *d*. boyut için *i*. bit alanında bulunan $S(n)$ noktaları ile *j*. bit alanında bulunan $S^{(r)}(n)$ noktalarının birbirlerinden faz alanındaki uzaklıkları $R^2_{d,i,j}(n,r)$ sunulmuştur.

$$R_{d,i,j}^2(n,r) = \left(S(n) - S^{(r)}(n)\right)^2 + \left(S(n+1) - S^{(r)}(n+1)\right)^2 \ldots + \left(S(n+d-1) - S(n+d-1)\right)^2 \quad (4.14)$$

(4.14)'teki denklem dikkatli bir şekilde incelenirse ilk teriminin 1. boyuta ait faz alanı uzaklığına, $R_{1,i,j}^2$'ye eşit olduğu görülebilir.

$$R_{1,i,j}^2(n,r) = \left(S(n) - S^{(r)}(n)\right)^2 \quad (4.15)$$

(4.14) denkleminde tüm terimler (4.15)'teki gibi 1. boyuttaki faz alanı uzaklıklarına göre sadeleştirilirse denklem (4.16) elde edilir.

$$R_{d,i,j}^2(n,r) = R_{1,i,j}^2(n,r) + R_{1,i+1,j+1}^2(n,r) \ldots + R_{1,i+d-1,j+d-1}^2(n,r) \quad (4.16)$$

(4.16) denklemine ek olarak, (4.15)'deki $R_{1,i,j}^2(n,r)$ değerlerinin dağılımına karekökü $R_{1,i,j}(n,r)$ değerlerinin dağılımından ulaşılabilir. (4.17) denkleminde $r_{1,i,j}^2$ oyf'si ile $r_{1,i,j}$ oyf'si arasındaki matematiksel bağlantı verilebilir.

$$r_{1,i,j}^2(t) = r_{1,i,j}\left(\sqrt{t}\right)\frac{1}{2\sqrt{t}} \quad (4.17)$$

(4.15)'e göre $r_{1,i,j}$ dağılımı *i*. bit alanındaki $S(n)$ ve *j*. bit alanındaki $S^{(r)}(n)$ değerlerinin oyf'lerince belirlenmektedir. *i*. bit alanındaki değerlerin oyf'si x_i, *j*. bit alanındaki x_j olduğuna göre ve x_i ve x_j oyf'leri birbirlerinden bağımsız olduklarından problem çözümü basitleşmektedir. Olasılık teorisinde birbirlerinden iki bağımsız rastgele değişkenin toplamının oyf'si, bağımsız değişkenlerin oyf'lerinin birbirleriyle katlanmasıdır. Buna dayanarak iki rastgele değişkenin farklarının oyf'si olan $r_{1,i,j}$ denklem (4.18)'deki gibi yazılabilir; denklemde yer alan * işlemcisi katlama anlamına gelmektedir.

$$r_{1,i,j}(t) = x_i(t) * x_j(-t) \tag{4.18}$$

(4.16) denkleminde *d* boyutu *N* yineleme sayısından küçük olduğu sürece, tüm alt terimler birbirlerinden bağımsızdır. Bu durumda $R^2_{d,i,j}(n,r)$ değerlerinin dağılımı olan $r^2_{d,i,j}$, alt terimlerin oyf'lerinin birbirleriyle katlanmasından hesaplanabilir. (4.19) denklemindeki * işlemcisi katlama işlemini belirtmektedir.

$$r^2_{d,i,j} = r^2_{1,i,j} * r^2_{1,i+1,j+1} * \dots * r^2_{1,i+d-1,j+d-1} \tag{4.19}$$

4.4.6 Basit dağılımlı bit dizilerinde FNF değerlerinin analitik olarak hesaplanması

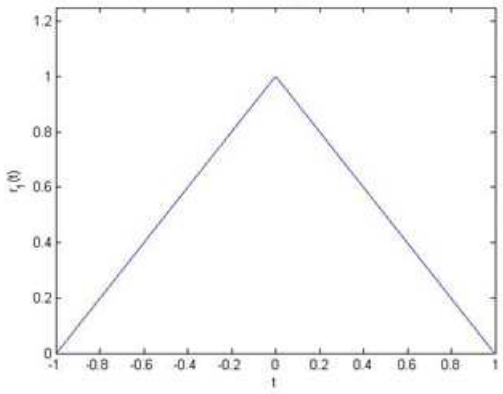
Bu bölümde tüm bit alanları biri üçgen diğeri düzgün dağılımlı iki farklı bit dizisinin FNF sonuçlarının birbirlerinden ne kadar ayrıştığı gösterilmiştir. Her iki bit dizisinde de tüm bit alanları birbirlerine özdeş olduğuna göre, tüm *i*. ve *j*. indeksler için faz alanındaki uzaklıkların karelerinin dağılımları da birbirleriyle özdeştir; bu bilgi bir önceki bölümde verilen (4.19) denklemiyle birleştirilirse

$$r^2_d = r^2_1 * r^2_{d-1} \tag{4.20}$$

sonucuna ulaşılır. Tüm alanların özdeş olmasından dolayı oyf'ler *i* ve *j* indekslerine göre değişiklik göstermez; bu yüzden de (4.20)'deki oyf'lerde söz konusu indeks değerlerine yer verilmesi gereksinimi kalmamıştır. (4.20)'de verilen tekrarlamalı denkleme göre r^2_d olarak nitelendirilmiş faz alanındaki uzaklıkların karelerinin oyf'si temelde r^2_1 oyf'sine, r^2_1 oyf'si ise r_1 oyf'sine bağlıdır.

r_1 oyf'sini açıklamak gerekirse, bahse konu oyf birbirlerinden bağımsız iki düzgün veya üçgen dağılımlı rastgele sayının farklarının oyf'sidir. Düzgün dağılımlı rastgele sayıların farklarının oyf'si (4.21)'de verilmiş, Şekil 4.7'de ise bahse konu oyf'nin grafiksel gösterimine yer verilmiştir:

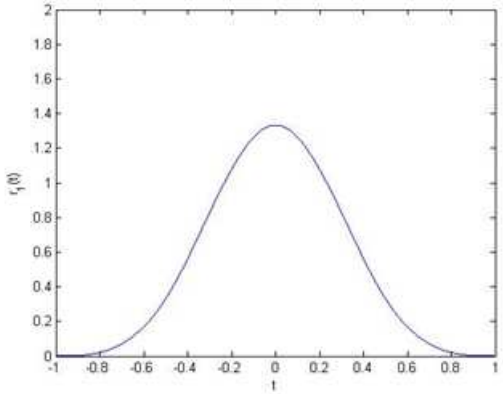
$$r_1(t) = (t+1)u(t+1) - 2t.u(t) + (t-1).u(t-1) \tag{4.21}$$



Şekil 4.7 Düzgün dağılımlı iki bit alanının farklarının dağılımı

Düzgün dağılım için yerine getirilen işlemler üçgen dağılım için yinelenirse, üçgen dağılımın r_1 oyf'si (4.22)'deki gibi, grafiksel gösterimi ise Şekil 4.8'deki gibi olur:

$$r_1(t) = \frac{8}{3}(t+1)^3u(t+1) - \frac{32}{3}\left(t+\frac{1}{2}\right)^3 u\left(t+\frac{1}{2}\right) + 16t^3u(t) - \frac{32}{3}\left(t-\frac{1}{2}\right)^3 u\left(t-\frac{1}{2}\right) + \frac{8}{3}(t-1)^3u(t-1) \tag{4.22}$$



Şekil 4.8 Üçgen dağılımlı iki bit alanının farklarının dağılımı

r_1 oyf'lerinin hesaplanmasından sonra her iki bit dizisine ait r_1^2 oyf'leri bulunur. r_1 oyf'leri *y* eksenine göre simetrik olduğundan r_1^2 oyf'lerinin hesabı sadece r_1 oyf'lerinin 0'dan büyük kısımları üzerinden yapılabilir. (4.23) ve (4.24)'teki ${r_1}^+$ oyf'leri, r_1 oyf'lerinin 0'dan büyük kısımlarını belirtmektedir; ${r_1}^+$ oyf'lerinin 0'dan önceki kısımlardaki olasılık yoğunlukları 0'a eşittir. Her oyf'nin eksi sonsuz ile artı sonsuz arasındaki integrali 1'e eşit olması gerektiğinden, ${r_1}^+$ oyf'lerinin 0'dan büyük kısımlarındaki olasılık yoğunlukları, r_1 oyf'lerinin aynı kısımlarındaki olasılık yoğunluklarının iki misline eşit olur.

Düzgün dağılım için:

$$ {r_1}^+(t) = 2[(1-t)u(t) + (t-1)u(t-1)] \tag{4.23}$$

Üçgen dağılım için:

$$ {r_1}^+(t) = \left(16t^3 - 16t^2 + \frac{8}{3}\right)u(t) - \frac{64}{3}\left(t-\frac{1}{2}\right)^3 u\left(t-\frac{1}{2}\right) + \frac{16}{3}(t-1)^3 u(t-1) \tag{4.24}$$

${r_1}^+$ oyf'leri bulunduktan sonra r_1^2 oyf'leri hesaplanır. Oyf'si bilinen bir rastgele değişkenin karesinin oyf'sini bulmak için (4.17)'de tanımlanmış olan yerine koyma işlemi uygulanır.

Düzgün dağılım için:

$$ r_1^2(t) = 2\left[\left(1-\sqrt{t}\right).u\left(\sqrt{t}\right) + \left(\sqrt{t}-1\right)u(\sqrt{t}-1)\right]\frac{1}{2\sqrt{t}} \tag{4.25}$$

Üçgen dağılım için:

$$ r_1^2(t) = \left[\left(16t^{3/2} - 16t + \frac{8}{3}\right)u(t) - \frac{64}{3}\left(\sqrt{t}-\frac{1}{2}\right)^3 u\left(\sqrt{t}-\frac{1}{2}\right) + \frac{16}{3}\left(\sqrt{t}-1\right)^3 u(\sqrt{t}-1)\right]\ \frac{1}{2\sqrt{t}} \tag{4.26}$$

(4.25) ve (4.26)'daki denklemler sadeleştirilirse ve basamak fonksiyonlarındaki köklü ifadeler giderilirse (4.27) ve (4.28) denklemleri elde edilir.

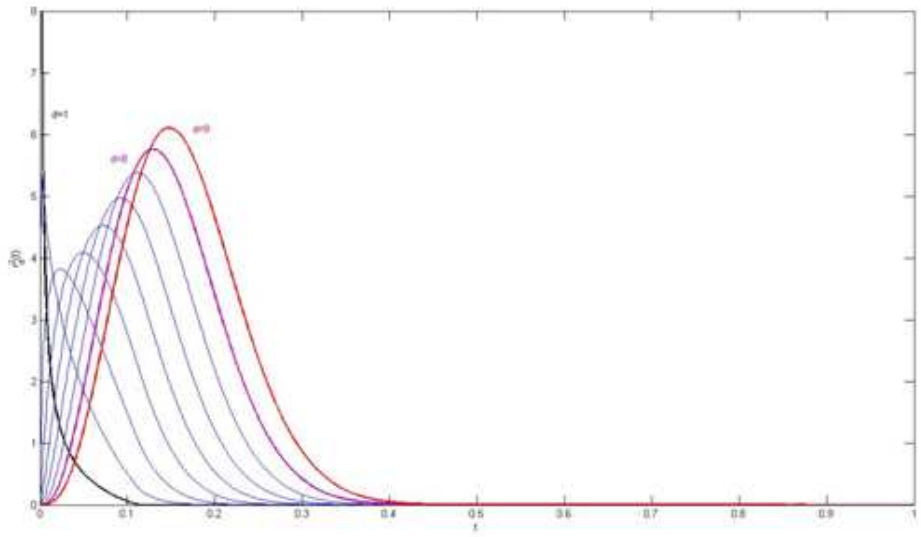
Düzgün dağılım için:

$$r_1^2(t) = \left(\frac{1}{\sqrt{t}} - 1\right) u(t) + \left(1 - \frac{1}{\sqrt{t}}\right) u(t-1) \tag{4.27}$$

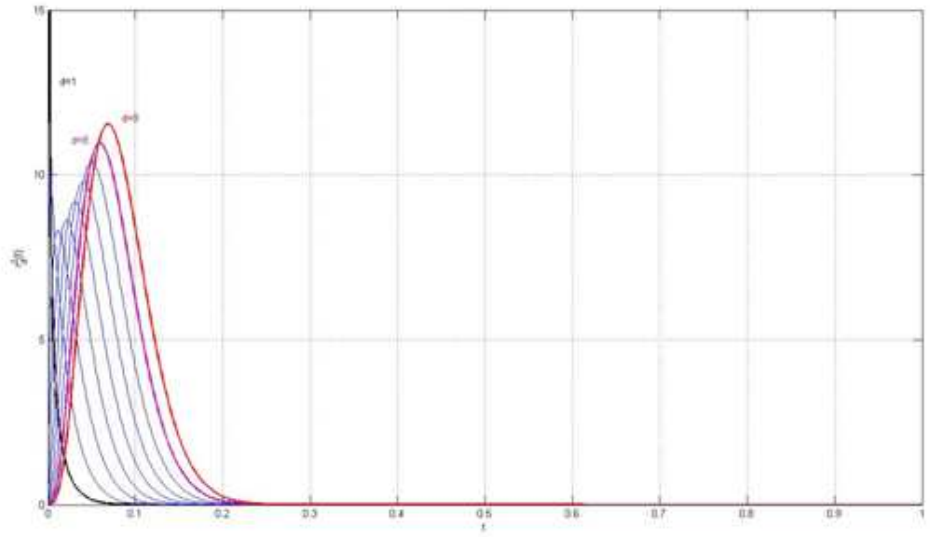
Üçgen dağılım için:

$$r_1^2(t) = \frac{\left(8t^{3/2} - 8t + \frac{4}{3}\right)}{\sqrt{t}} u(t) - \frac{32}{3\sqrt{t}}\left(\sqrt{t} - \frac{1}{2}\right)^3 u\left(t - \frac{1}{4}\right) + \frac{8}{3\sqrt{t}}\left(\sqrt{t} - 1\right)^3 u(t-1) \tag{4.28}$$

Her iki dizinin de $d = 1$ haricindeki r_d^2 oyf'lerinin sembolik ifadeleri çok karmaşıktır; bundan dolayı r_d^2 oyf'leri ancak nümerik yöntemler vasıtasıyla hesaplanabilmiştir. Düzgün ve üçgen dağılımların numerik hesaplanan r_d^2 oyf'leri Şekil 4.9 ve 4.10'da görsel olarak sunulmuştur.



Şekil 4.9 Bit alanları düzgün dağılımlı olan bit dizisinin r_d^2 oyf'si



Şekil 4.10 Bit alanları üçgen dağılımlı olan bit dizisinin r_d^2 oyf'si

Yukarıda üçgen dağılımlar için hesaplanmış olan r_d^2 oyf'leri vasıtasıyla toplam hatalı olmayan komşu sayısı $\#FN_d$ hesaplanabilir. Denklem (4.29)'da $\#FN_d$ K ile FNR_d çarpımı olarak tanımlanmıştır; denklemdeki K gerçekleştirilen komşuluk testi sayısını, FNR_d diziye ait iki noktanın komşuluk ilişkisi içerisinde olma olasılığını belirtmektedir.

$$\#FN_d = K.FNR_d \tag{4.29}$$

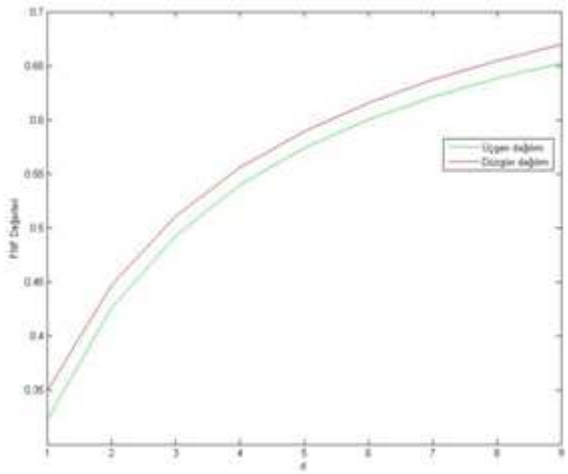
Denklem (4.30)'da, komşuluk ilişkisinin sadece 2. koşula bağlandığı örnek bir senaryo için FNR_d'nin doğrudan r_d^2 oyf'sinden nasıl hesaplanabileceği gösterilmiştir. Denklemde yer alan R_A değeri (4.6)'da tanımlanmış olan dizi varyansının nümerik karşılığı, A_{tol} değeri ise sabit bir sayıdır.

$$FNR_d = \int_0^{(R_A A_{tol})^2} r_d^2(t).dt \tag{4.30}$$

Denklem (4.8)'deki $\#FN_d$ ve $\#FN_{d-1}$ ifadelerinin yerlerine (4.29)'daki açılım konursa denklem (4.31) elde edilir.

$$FNF_d = \frac{\#FN_{d-1} - \#FN_d}{\#FN_{d-1}} = \frac{K.FNR_{d-1} - K.FNR_d}{K.FNR_{d-1}} = \frac{FNR_{d-1} - FNR_d}{FNR_{d-1}} \quad (4.31)$$

Son olarak her bir d boyutu için, bit dizilerinin numerik biçimdeki r_d^2 oyf'leri kullanılarak (4.30)'daki integral numerik olarak hesaplanır ve bulunan sonuçlar (4.31)'deki formülle sokularak FNF_d hatalı komşuluk oranları elde edilir. Şekil 4.11'de düzgün ve üçgen dağılımlı bit dizilerinin numerik olarak hesaplanan hatalı komşuluk oranları sunulmuştur.



Şekil 4.11 Bit alanları düzgün ve üçgen dağılımlı olan bit dizilerinin değişen boyutlardaki FNF değerleri

4.4.7 Bit dizisi özelliklerinin FNF değerler üzerindeki etkilerinin incelenmesi

Bölüm 4.3'te ayrıntılı bir şekilde anlatılan genel konuşma kodlayıcısı modeline göre kodlanmış bit dizileri yinelenen çerçevelerden oluşmaktadır. Her konuşma kodlayıcısı kendisine özgü bir çerçeve yapısına sahip olmakla beraber, bahse konu çerçevelerin içerisinde tanımlı bulunan bit alanlarının sayısı, dağılımları ve aralarındaki ilişkiler kodlayıcıdan kodlayıcıya değişiklik göstermektedir. Hatırlanacağı üzere bölüm 4.4.4'te, FNF değerlerinin bit alanları, dağılımları ve birbirleriyle olan ilişkilerince belirlendiği ortaya konmuştur. Bu durumda her bir kodlayıcının kendisine özgü r_d^2 oyf'lerine ve FNF değerlerine sahip olacağı öngörülebilir.

Aynı kodlayıcıya ait değişik çıktıların bit alanlarına ait dağılımlar genelde benzer olsa da özdeş değildir; bu yüzden aynı kodlayıcının değişik çıktılarının FNF sonuçları birbirlerinden farklı olabilir. Öte yandan bir kodlayıcının çerçevelerinde tanımlı bulunan bit alanları arasındaki mevcut ilişkiler tüm çıktılar için aynıdır; çıktıdan çıktıya değişiklik göstermeyen bu ilişkiler farklı çıktılara ait FNF değerlerinin bir arada toplanmasına yol açarlar. Söz konusu bit alanları arasındaki ilişkiler dört genel tip altında tanımlanabilir:

Tip I. Çerçeveler arası ilişkiye sahip bit alanları: Bir çerçevenin *i*. bit alanındaki değerin diğer çerçevelerin *i*. bit alanındaki değerlerle ilişkili olması

Tip II. Çerçeve içinde birbirleriyle ilişkili bit alanları: Aynı çerçeveyi paylaşmakta olan *i*. ve *j*. bit alanlarındaki değerlerin birbirleriyle ilişkili olması

Tip III. Hem çerçeveler arası hem de çerçeve içinde birbirleriyle ilişkili bit alanları: Bir çerçevenin *i*. ve *j*. bit alanlarındaki değerlerin hem birbirleriyle hem de diğer çerçevelerin *i*. ve *j*. bit alanlarındaki değerlerle ilişkili olması

Tip IV. İlişkisiz bit alanları: Bir çerçevenin *i*. bit alanındaki değerin hem çerçevenin diğer bit alanlarındaki değerlerle, hem de diğer çerçevelerin *i*. bit alanlarındaki değerlerle ilişkili olmaması

Bu durumda yukarıda sunulan bilgilere göre, kodlayıcı tanımanın başarılı olabilmesi için FNF sonuçlarında

- Ya bit alanları arasındaki ilişkilerin bit alanları dağılımındaki farklılıklara göre daha baskın olması
- Ya kodlayıcıların bit alanlarındaki dağılım farklılıklarının, dağılım benzerliklerine göre önemsiz kalması
- Ya da bit alanları arasındaki ilişkilerin dağılım farklılıklarından bağımsız olarak gözlemlenebilmesi

gereklidir. Birinci koşul olabilecek tüm kodlayıcı ve bit alanları için garanti edilemez. İkinci koşul ise değerlendirmeye alınan örneklerin uzunluklarının yeterli düzeylere çekilmesiyle yerine getirilebilir. Son koşul ise bazı konuşma kodlayıcıları için doğru olabilir.

4.4.7.1 Bazı örnek bit dizileri için FNF değerlerinin yorumlanması

Bu kısımda farklı çerçeve yapıları ve bit alanı dağılımlara sahip bit dizilerine ait FNF değerlerinin ne kadar ayrıştığı, ayrışmaların hangi boyutlarda hangi oranlarda oluştuğu, boyuttan boyuta ne kadar değişiklik gösterdiği konusu ele alınmıştır. Pratik testler gerçekleştirilerek FNF değerlerinin hangi etkenlere bağlı olduğu araştırılmış; araştırma sırasında Çizelge 4.1'de sunulmuş olan farklı bit dizisi türlerinden yararlanılmıştır.

Çizelge 4.1'de verilen farklı bit dizisi türlerinden bahsetmek gerekirse, bir çerçeve içerisinde Tip I türü ilişkiye sahip bit alanlarının sayısı "$\dot{I}$" simgesinin altsimgesinde, Tip III türü ilişkiye sahip bit alanlarının sayısı "$A\dot{I}$" simgesinin alt simgesinde gösterilmektedir. N simgesinin altsimgesi çerçevenin kaç bit alanından oluştuğunu belirtmektedir. Tip I ve Tip III türlerindeki ilişkiye sahip olanlar haricindeki bit alanlarının ilişkisiz olduğu kabul edilmiştir. Hiçbir bit dizisi türünde ise Tip II tipi ilişkiye yer verilmemiştir. Son olarak bütün bit dizisi türlerinde de bit alanlarının sekizer bitten oluştuğu kabul edilmiştir.

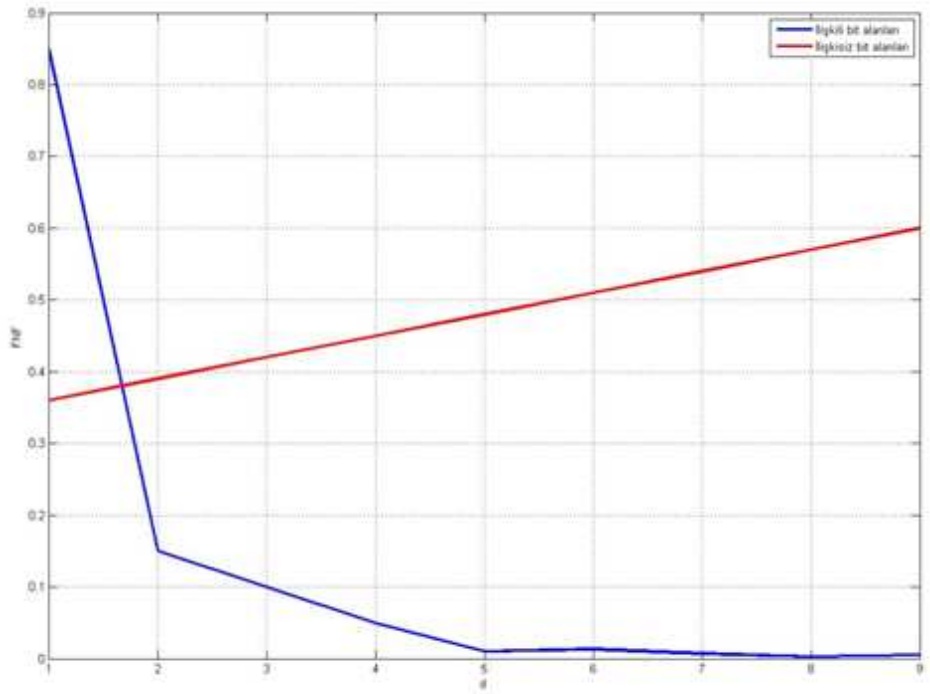
Çizelge 4.1'de yer alan ve pratik testlerde kullanılması amacıyla çeşitlendirilen bit dizisi türleri aslında alt bayt dizilerinden oluşmaktadır. Kabaca iki cins alt bayt dizisi bulunmaktadır: İlişkili ve ilişkisiz alt bayt dizileri. İlişkili alt bayt dizilerinde, diziyi oluşturan değerler önceki sonraki değerler ile ilişki içerisindedir. Diğer yandan ilişkisiz alt bayt dizilerinde ise bu durum tam tersinedir, değerler birbirlerinden bağımsızdır.

Çizelge 4.1 Testlerde kullanılan kodlayıcı tanımlayıcı alternatifleri

Bit Dizisi Tipi	**Çerçeve Boyu**	**Açıklama**
$\dot{I}_1N_{10}$	10	Çerçeve içindeki 1 bit alanı diğer çerçevelerdeki eşlenikleriyle ilişkili. Çerçevede yer alan diğer 9 bit alanı ilişkisiz
$\dot{I}_9N_{10}$	10	Çerçeve içindeki 9 bit alanı diğer çerçevelerdeki eşlenikleriyle ilişkili. Çerçevede yer alan diğer 1 bit alanı ilişkisiz
$\dot{I}_{10}N_{10}$	10	Çerçeve içindeki 10 bit alanı da diğer çerçevelerdeki eşlenikleriyle ilişkili.
$\dot{I}_{10}A\dot{I}_3N_{10}$	10	Çerçeve içindeki 10 bit alanı da diğer çerçevelerdeki eşlenikleriyle ilişkili. Bunun haricinde aynı çerçeve dâhilindeki peş peşe 3 bit alanı birbirleriyle de ilişkili
$\dot{I}_1N_{14}$	14	Çerçeve içindeki 1 bit alanı diğer çerçevelerdeki eşlenikleriyle ilişkili. Çerçevede yer alan diğer 13 bit alanı ilişkisiz
$\dot{I}_{13}N_{14}$	14	Çerçeve içindeki 13 bit alanı diğer çerçevelerdeki eşlenikleriyle ilişkili. Çerçevede yer alan diğer 1 bit alanı ilişkisiz
$\dot{I}_{14}N_{14}$	14	Çerçeve içindeki 14 bit alanı da diğer çerçevelerdeki eşlenikleriyle ilişkili.
$\dot{I}_{14}A\dot{I}_3N_{14}$	14	Çerçeve içindeki 14 bit alanı da diğer çerçevelerdeki eşlenikleriyle ilişkili. Bunun haricinde aynı çerçeve dâhilindeki peş peşe 3 bit alanı birbirleriyle de ilişkili

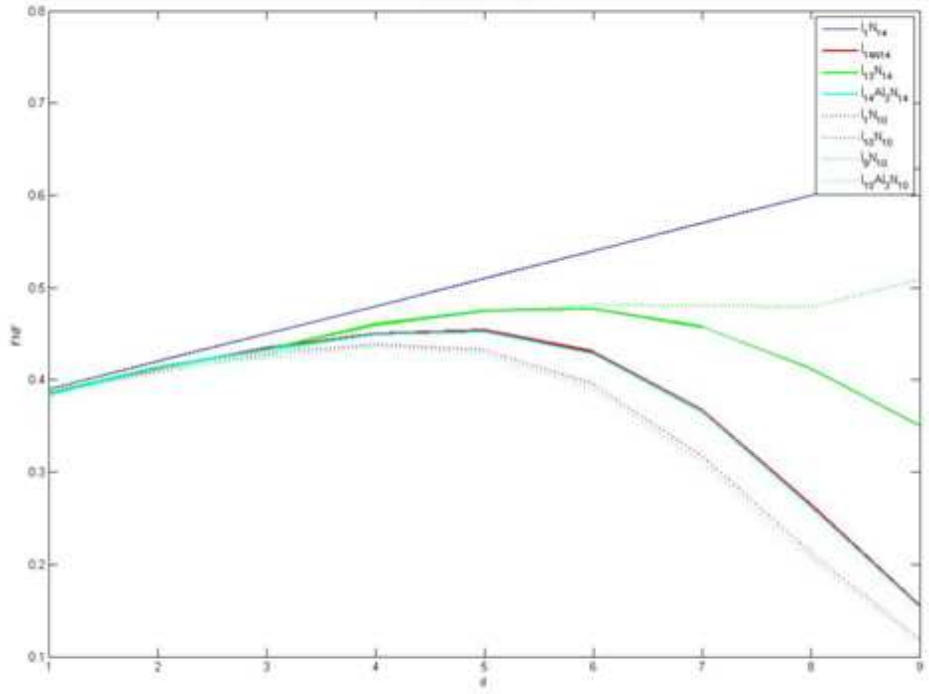
Gerçekleştirilen ilk aşamadaki testlerde, FNF sonuçları üzerinde, sadece bit alanı yerleşimlerinin etkisini tespit edilebilmesine çalışılmıştır; tüm bit dizisi türleri özdeş

dağılımlara sahip ilişkili ve ilişkisiz alt bayt dizilerinden üretilmiştir. Tüm Tip I ve Tip III bit alanlarının kendi aralarında dağılımları aynıdır, diğer tarafta tüm Tip IV bit alanlarının kendi aralarında dağılımları aynıdır. Şekil 4.12'de, bit dizisi türlerini oluşturan özdeş alt bayt dizlerinin FNF sonuçları sunulmuştur:



Şekil 4.12 Bit dizisi türlerinin üretiminde kullanılan alt bayt dizilerinin hesaplanan FNF değerleri

Daha sonra, özdeş alt bayt dizilerinden oluşturulan bit dizisi türlerinin FNF sonuçları bulunmuştur; aşağıdaki Şekil 4.13'te bahse konu FNF sonuçları yer almaktadır. Sunulan şekil dikkatlice incelendiğinde $\dot{I}_9N_{10}$ ve $\dot{I}_{13}N_{14}$ bit dizisi türlerinin diğerlerinden ayrıştığı, $\dot{I}_{10}N_{10}$'un $\dot{I}_{10}A\dot{I}_3N_{10}$ ile, $\dot{I}_{14}N_{14}$'ün $\dot{I}_{14}A\dot{I}_3N_{14}$ ile oldukça yakın seyrettiği ve maalesef $\dot{I}_1N_{10}$ ile $\dot{I}_1N_{14}$'ün birbirlerinden hiç ayrışmadığı görülmektedir.

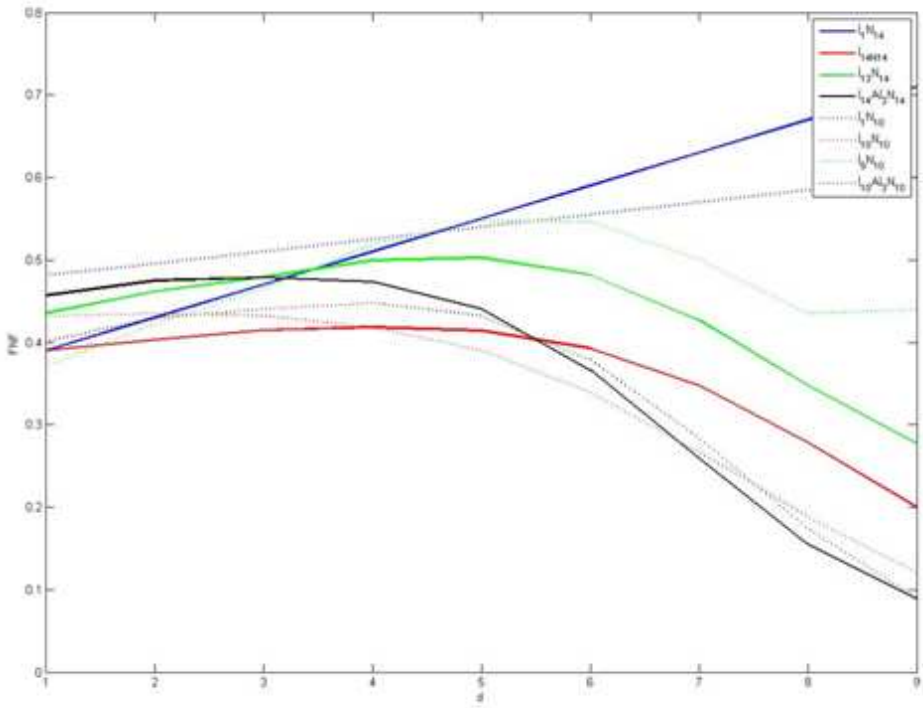


Şekil 4.13 Özdeş alt bayt dizilerinden üretilmiş bit dizisi türlerinin FNF sonuçları

Öte yandan sınıflandırılacak dizilerin farklı *T* değerleri için FNF sonuçları hesaplanıp değerlendirmeye alınırsa, ayrıştırmadaki başarı arttırılabilir. *T* gecikme değeri komşuluk ilişkisi açısından çok önemli olan ilişkili değerlerin peş peşe gelme varyasyonlarını etkilemektedir. Örneğin *T*=1 durumunda ayırt edilemeyen $\dot{I}_1N_{10}$ ile $\dot{I}_1N_{14}$ dizileri, *T*=10 veya *T*=14 durumlarında birbirlerinden ayrıştırılabilir; fakat olası tüm *T* değerlerine göre FNF hesabı yapmak pratik bir yaklaşım değildir.

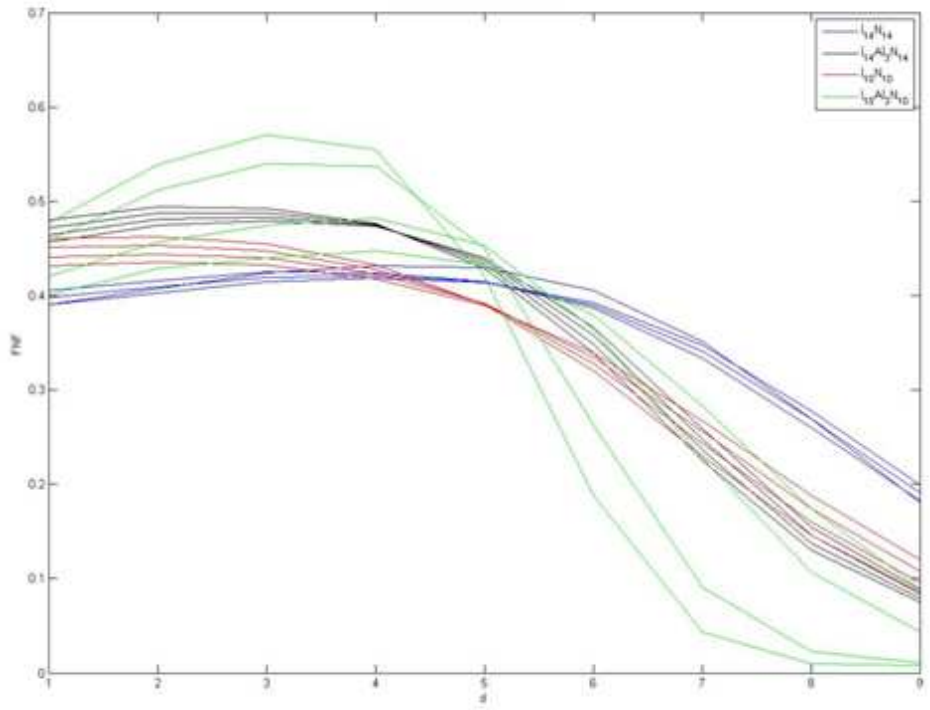
Tüm bunların haricinde, hatırlanacağı üzere Şekil 4.13'te FNF sonuçları verilen bit dizisi türlerinin hepsi aynı özdeş alt bayt dizilerinden oluşturulmuştur. Gerçek hayatta ise, bit alanları arasındaki ilişki farklılıklarına ek olarak, bizzat bit alan dağılımları da başkalık gösterebilir. Bir sonraki aşamada hem bit alanları ilişkilerinin hem de bit alanı dağılımlarının farklılaştığı durumlar için FNF sonuçlarının nasıl etkilendiği konusu ele alınmıştır. Çizelge 4.1'de yer alan bit dizileri (türleri) bu sefer farklı dağılımlara sahip alt bayt dizilerinin bir araya getirilmesiyle oluşturulmuş, daha sonra oluşturulan bu bit dizilerinin FNF sonuçları hesaplanmıştır. Bulunan sonuçlar Şekil 4.14'te yer almaktadır;

dizi türlerinin FNF sonuçlarının Şekil 4.13'e kıyasla Şekil 4.14'te daha fazla ayrışma sergilediği görülmektedir.



Şekil 4.14 Farklı dağılımlı alt bayt dizilerinden üretilmiş bit dizisi türlerinin FNF sonuçları

Son aşamada, aynı kodlayıcının farklı çıktılarında görülen FNF sapmaları konusu irdelenmiştir. Bu aşamadaki gerçekleştirilen testlerde, Şekil 4.14'te birbirlerine en yakın FNF sonuçlarına sahip olan, dört bit dizisi türü kullanılmıştır. Teste dâhil edilen her dizi türü için alt bayt dizilerinin dağılımı düşük oranda değiştirilerek dört adet varyasyon üretilmiştir. Bit dizisi türü başına dört tane, toplam on altı bit dizi varyasyonunun FNF değerleri hesaplanmış, bulunan sonuçlar Şekil 4.15'de sergilenmiştir.



Şekil 4.15 Bit alanlarının dağılımlarının düşük oranda değiştirildiği hallerde FNF sonuçları

Sonuç olarak bu bölümde, bit alanları arası ilişkilerin ve bit alanı dağılımlarının FNF değerleri üzerindeki etkileri araştırılmış, oluşturulan bazı hayali bit dizisi türleri için FNF değerleri hesaplanmıştır. Elde edilen test sonuçlarına göre etkili tanıma mekanizmasının, hem bit alanları arası ilişkilerdeki, hem de bit alanları dağılımlarındaki farklılıklara dayandığı anlaşılmıştır. Öte yandan aynı kodlayıcının farklı çıktılarına ait bit alan dağılımlarının örnekten örneğe (Burada örnek ifadesi değerlendirmeye alınan girdi bit dizisini kastediyor.) değişmesinden dolayı, FNF sonuçlarında sapmalar oluşabilmektedir. Tanıma işleminin başarılı olabilmesi için bit alanı dağılımlarındaki değişkenliğin azaltılması son derece önemlidir; örneklerin değerlendirmeye alınan boylarının arttırılması bit alanı değişkenliğini azaltabilmektedir. Tüm bu yorumlara ek olarak, girdi dizilerin FNF sonuçları arasındaki ayrışmalar *T* gecikme değerine göre azalıp artabilmektedir; bundan dolayı olabildiğince çok sayıda *T* gecikmesi için FNF hesaplanması tanıma performansını iyileştirebilir. *T* gecikmesinin *N* yineleme sayısına eşit olması durumunda, hesaplanan FNF değerlerindeki ayrışmalar daha gözlemlenebilir olacaktır.

4.5 Sınıflandırıcılar

Tezin bu bölümünde önerilen özgün konuşma kodlayıcı tanımlayıcısı en yalın biçimde, kodlanmış konuşma sinyaline ilişkin bit dizilerini FNN (hatalı en yakın komşular) özniteliklerine göre kategorilere ayıran standart bir sınıflandırıcı olarak tarif edilebilir. Sınıflandırıcı performansının arttırılabilmesi ve/veya eğitim maliyetinin düşürülebilmesi için girdi öznitelikler üzerinde normalizasyon gibi türlü ön işlemler uygulanabilir.

Kodlayıcı tanımlayıcısında kullanılabilecek en basit sınıflandırıcı, Kocal vd.'nin (2009) de gizli veri varlığını ifşa etmekte yeğlemiş olduğu doğrusal bağlanım sınıflandırıcısıdır. Doğrusal bağlanım sınıflandırıcısı eğitimi, özniteliklerin bit dizisinin sınıfı üzerindeki ağırlıklarının hesaplanmasından ibarettir. Aşağıda verilmiş olan (4.32)'de özniteliklerin ağırlıklarının nasıl hesaplanacağı gösterilmektedir (Rojas 1996). *w* vektörü sınıflandırmada kullanılacak vektör, *X n* adet örneğe ait *m* adetlik öznitelik değerlerinden oluşturulmuş eğitim matrisidir. *a* ise *n* adet örneğin dahil oldukları sınıfları gösteren vektördür.

$$\mathrm{w} = (X^T X)^{-1} X^T a \tag{4.32}$$

Bu tezde önerilen kodlayıcı tanımlayıcısının ilk uygulamalarında, sınıflandırıcı olarak bağlanım sınıflandırıcılarından yararlanılmış, ancak daha sonradan doğrusal olmayan girdilerin sınıflandırılmasında kullanılabilen polinomik SVM (Eads 2003) ve yapay sinir ağları devreye alınmıştır (SVM: destek vektör makinesi – *support vector machine*).

4.6 Test Sonuçları

4.6.1 Test ve eğitimde kullanılacak kaotik öznitelik kümelerinin hazırlanması

Kodlayıcı tanıma testleri PCM 128K, GSM 6.60 EFR, GSM 6.10 FR, G.726 (16K, 24K, 32K), G.729, MELP kodlanmış bit dizileriyle ve rastgele bit dizileri üzerinde gerçekleştirilmiştir. NTIMIT veritabanından süreleri değişkenlik gösteren 2560 konuşma sinyali örneği gelişigüzel seçilerek geçici bir PCM veri tabanı oluşturulmuştur.

Bahse konu geçici PCM veri tabanı biri 512, diğeri 2048 örnek içerecek şekilde iki bölünmüş, (kalıcı olacak) PCM veri tabanları meydana getirilmiştir.

Bölünme sonucunda elde edilen PCM veri tabanlarındaki örneklerin tamamı test kapsamındaki bütün kodlayıcılarla kodlanmıştır. Bu sayede her bir kodlayıcı için biri test amaçlı, diğeri amaçlı eğitim kullanılacak iki adet kodlanmış konuşma veri tabanı elde edilmiştir. Diğer yandan kodlanmış konuşma veri tabanlarındaki kadar örnek içeren, eğitim ve test maksatlı iki adet rastgele gürültü veri tabanı hazırlanmıştır. Oluşturulmuş olan tüm bu eğitim ve test veritabanları kaotik analize tabi tutulmuş ve böylece PCM 128K, GSM 6.60 EFR, GSM 6.10 FR, G.726 (16K, 24K, 32K), G.729, MELP ve rastgele sayı türlerinin her biri için biri eğitim diğeri test amacıyla kullanılacak iki adet kaotik öznitelik kümesi üretilmiştir. Adı geçen kaotik analizler, TISEAN (2007) kütüphanesindeki hatalı en yakın komşular yazılımı vasıtasıyla gerçekleştirilmiş, örnek girdi başına beş farklı boyutta (d_e = 1, 2, 3, 4 and 5), boyut başına üç, toplam 15 kaotik öznitelik hesaplanmıştır.

4.6.2 Test ve eğitimde kullanılacak kaotik öznitelik kümelerinin hazırlanması

Testler, Çizelge 4.2'de sunulan her biri değişik sınıflandırma yöntemi ve girdi normalizasyon kombinasyonunu bir araya getiren beş çeşit kodlayıcı tanımlayıcısının sınanmasıyla yerine getirilmiştir. Geliştirilen kodlayıcı tanımlayıcıları girdi bit dizilerini dokuz ayrı sınıfta kategorize edebilmektedirler. Bu sınıflar sırasıyla PCM 128K, GSM 6.60 EFR, GSM 6.10 FR, G.726 (16K, 24K, 32K), G.729, MELP ve rastgele sayı sınıflarıdır.

Çizelge 4.2 Testlerde kullanılan kodlayıcı tanımlayıcı alternatifleri

Sınıflandırıcı Tipi	Girdi Normalizasyonu
Yapay sinir ağları	Normalizasyon yok
Yapay sinir ağları	Varyansa göre normalizasyon
Doğrusal bağlanım	Normalizasyon yok
Polinomik SVM	Normalizasyon yok
Polinomik SVM	Varyansa göre normalizasyon

Eğitilen kodlayıcı tanımlayıcılarının performansları (4.33)'te tanımlanmış olan denklem cinsinden hesap edilmiştir. Denklemde yer alan #TP ifadesi doğru sınıflandırılan bit dizilerinin toplam sayısını, #FP ifadesi ise yanlış sınıflandırılan bit dizilerinin toplam sayısını belirtmektedir.

$$\text{Başarım Oranı} = \frac{\#TP}{\#TP + \#FP} \tag{4.33}$$

Çizelge 4.3'te tüm kodlayıcı tanımlayıcısı alternatiflerinin performans sonuçları yer almaktadır.

Çizelge 4.3 Test kapsamındaki kodlayıcı tanımlayıcılarının performans sonuçları

Sınıflandırıcı Tipi	Başvurulan Normalizasyon	Farklı pencere boyları için elde edilen başarım oranları		
		512	1024	1536
Yapay sinir ağları	Normalizasyon yok	53.3	90.5	91.8
Yapay sinir ağları	Varyansa göre normalizasyon	56.2	91.6	95.6
Doğrusal bağlanım	Normalizasyon yok	18.5	24.5	34.0
Polinomik SVM	Normalizasyon yok	75.0	88.5	90.3
Polinomik SVM	Varyansa göre normalizasyon	84.4	97.5	98.7

Yapılan testlerde, doğrusal olmayan sınıflandırıcıların doğrusal sınıflandırıcılara kıyasla daha başarılı olduğu gözlemlenmiştir. Ancak beklenildiği üzere, kullanılan öznitelik vektörlerinin ön işleme koşullarına göre sınıflandırıcı performansında değişiklikler tespit edilmiştir. Polinomik SVM ve yapay sinir ağları sınıflandırıcıları varyans normalizasyonlarında en iyi performansları yakalarlarken, doğrusal bağlanım sınıflandırıcısı en yüksek performansına normalizasyonun hiç uygulanmadığı koşullarda ulaşmıştır. Sınıflandırma başarısının gecikme değerlerinden etkilenmediği ($d < 10$, düşük gecikme değerleri için) gözlemlenmiş, değişik T değerleri için tekrarlanan

testlerde, benzer yüksek başarımlı sonuçlara ulaşılmıştır. Boyut sayısı arttırıldıkça sınıflandırma başarısının önce yükseldiği ancak sonra doyuma ulaştığı görülmüş; bu durum yüksek *d* değerlerinin güvenilirliklerinin düşük olmasıyla açıklanmıştır. Özetle kodlayıcı tanımanın performansının üç etkene bağlı olduğu belirlenmiştir:

Sınıflandırıcı tipi: Değişik kodlayıcı tiplerine ait olan kaotik öznitelikler birbirlerinden sadece düzlemler kullanılarak izole edilemediklerinden doğrusal bağlaşım sınıflandırıcıları yüksek başarım oranlarına ulaşamamaktadırlar. Diğer taraftan, doğrusal olmayan SVM ve yapay sinir ağları sınıflandırıcıları kaotik öznitelikleri daha doğru gruplandırabilmekte ve böylelikle kabul edilebilir başarım oranlarına erişebilmektedirler.

Pencere boyu: Hatalı en yakın komşuluk oranı hesabı yapılan pencere boyu ne kadar arttırılırsa, hesaplanan FNF sonuçları da o kadar güvenilir olmaktadır. Ne yazık ki pencere boyunun, örnek toplama süresi, maksimum paket boyu, hesaplama karmaşıklığı gibi nedenler yüzünden sınırlandırılması icap etmektedir; dolayısıyla testlerde sadece 512'nin katları olan 512, 1024 ve 1536'lık pencere boyları tercih edilmiştir. 512'lik pencere boyu SVM sınıflandırıcılarının %50 başarım oranını aşmaya başladıkları bir pencere boyudur; 1536 ise hem maksimum Ethernet çerçeve boyuna yakın bir değerdir, hem de çoğu sınıflandırıcının başarım oranı %90'nın üzerinde erişebilmektedir. Gerçek şu ki, FNF değerleri aslında hesaplandıkları bit dizisinin bir nevi özet bilgisini sunarlar, doğaları gereği her uzunluktaki bit dizisinden hesaplanabilirler, bu yüzden eğitim ve testlerin illa sabit uzunluklu pencere boyları üzerinde gerçekleştirilecek diye bir kaide bulunmamaktadır.

Normalizasyon: Kodlayıcı çıktılarının istatistiksel özellikleri tüm kaotik özniteliklere tesir eder. Bahse konu özniteliklerin sınıflandırma öncesi normalize edilmeleri, eğitim fazındaki yakınsanma hızını arttırır, yuvarlama kaynaklı nümerik hataları indirger.

Aşağıda verilmiş olan Çizelge 4.4'te en başarılı konuşma kodlayıcısı tanımlayıcısının (1536'lık pencere boyunda çalışan varyans normalizasyonlu polinomik SVM) karışıklık matrisi verilmiştir. Bahse konu matrise göre, GSM 6.60 EFR en iyi

tanımlanabilen (sınıflandırılan) kodlayıcı olurken, MELP en kötü tanıma performansına sahip olmuştur. En fazla karışıklık, %2 oranı ile MELP 2400 bps ile G.726 24 Kbps arasında yaşanmıştır.

Çizelge 4.4 1536'lık pencere boyunda çalışan varyans normalizasyonlu polinomik SVM'ye ait karışıklık matrisi

	Tanımlanan Kodlayıcı									
Gerçek Kodlayıcı		**GSM 6.60 4.5K**	**G.726 24K**	**G.726 32K**	**G.729 32K**	**G.726 16K**	**PCM 128K**	**GSM 6.10 13.2K**	**MELP 2400**	**Rastgele Bit Dizisi**
	AMR 4.5K	1.0000	0	0	0	0	0	0	0	0
	G.726 24K	0	0.9707	0	0	0	0	0.0049	0.0244	0
	G.726 32K	0	0	0.9956	0	0	0.0029	0.0010	0	0.0005
	G.729 32K	0.0005	0	0	0.9990	0	0	0	0.0005	0
	G.726 16K	0.0005	0.0005	0	0.0005	0.9961	0	0.0005	0.0010	0.0010
	PCM 128K	0	0	0.0024	0	0	0.9976	0	0	0
	GSM 6.10 13.2K	0.0010	0.0151	0.0005	0	0.0020	0.0015	0.9653	0.0142	0.0005
	MELP 2400	0	0.0205	0.0005	0	0.0059	0	0.0093	0.9639	0
	Rastgele Bit Dizisi	0	0	0	0.0020	0.0024	0.0005	0.0005	0	0.9946

5. İSTATİSTİKSEL OLARAK İYİLEŞTİRİLMİŞ MATRİS KODLAMA

Literatürde resim, video ve müzik üzerindeki benzerleri kadar yaygınlık göstermese bile konuşma sinyalin üzerinde de çok sayıda gizli veri gömme çalışması (Cheng ve Sorensen 2001, Gurijala vd. 2002, Wu ve Kuo 2002) bulunmaktadır. Çalışmaların ağırlıklı bir kısmında gömme işlemi için sinyalin PCM biçimi tercih edilmişken, azınlıkta kalmış bir kısmı gizli veriyi kodlama sırasında gömmeyi (Tian vd. 2009, Chang ve Yu 2002) önermektedir. Bu çalışma kodlanma sırasında gizli veri gömmen yöntemlerle aynı kategoride yer almakta olup, taşıyıcı sinyallerin Markov zincir modellerinden yararlanarak steganaliz yöntemlerine karşı daha dayanıklı veri gömme mekanizmalarının tasarımını konu almaktadır.

Bir gizli veri gömme uygulamasının steganaliz yöntemlerine karşı sağlayabileceği dayanıklılık ve oluşturulan gizli kanal kapasitesi gibi başlıca özellikleri, doğrudan gizli verinin gömüldüğü taşıyıcı sinyalin yapısal özelliklerine bağlıdır. Konuşma sinyalinin yapısal özelliklerine atıfta bulunursak, çoğu konuşma kodlayıcısı pratikte sabit bit oranında (*CBR*) trafik üretmektedir. Öte yandan konuşma sinyali halinde taşınan bilgi gerek anlamsal olarak, gerekse kendisini oluşturan dalga şekli açısından birim zamanda eş miktarda bilgi içermez. Başka bir değişle konuşma sinyali bariz bir şekilde değişen bit oranındaki (*VBR*) trafik özelliklerine sahiptir. Buna karşın VBR özellikli konuşma bilgisinin, her çerçevenin mutlaka bir şeylerle doldurulduğu CBR kanalda iletilmeye zorlanması, VBR bilginin pek çok çerçeveye yayılmasına ve/veya tekrarlanmasına yol açar. Aynı bilginin ardışık değişik biçimlerde pek çok çerçevede yer alması, çerçeveler içinde tanımlanmış bit alanlarının rastgele olmamasıyla ve de ardışık bit alanları arasında ilinti yaratılmasıyla sonuçlanır. Bahse konu rastgele olmama durumu, bit alanlarına ait 1. derece Markov zincirleri modellerinde gözlemlenebilir.

Diğer taraftan, evrensel steganografinin bakış açısından bir değerlendirme yapıldığında, konuşma sinyaline eklenen gizli veri, konuk olduğu konuşma sinyali içerisinde bir takım ek bozulmalara yol açar. Konuşma sinyali senaryoları için, gizli veri gömme yönteminin başarısı, oluşan bozulmaların insan işitme sistemi ve steganaliz yöntemleri için algılanabilir olmamasına bağlıdır. Kodlanmış sinyale ait bit dizisinde tanımlanmış

bit alanlarının en önemsiz bitleri (EÖB) aynı zamanda algısal açıdan da en az öneme sahip yerler olduklarından, gizli veri gömme yöntemlerinin öncelikle yararlanacağı kısımlar olurlar. Hatta steganaliz kıstası açısından incelendiklerinde, genellikle kendilerinden daha anlamlı anlam taşıyan bitlere göre daha yüksek entropiye sahiptirler, bu sayede entropisi 1'e eşit kriptolu gizli veri bitlerinin saklanabilmesi için daha uygun koşullar sunarlar. Ne yazık ki, pratikte EÖB'lerin entropisi yine de şifreli verinin entropisi kadar yüksek olamadığından, entropi kriterinin hiç dikkate alınmadığı EÖB gizli veri uygulamaları insan işitme sistemi açısından algılanamazlık kriterini başarıyla yerine getirebilseler de, olası steganaliz yöntemlerinin sömürebileceği zafiyetleri bünyelerinde ihtiva edebilirler.

Bir steganaliz yöntemi gizli veri varlığını ifşa edebilmek için kodlayıcı çıktısın içindeki tüm bitleri işler; gizli veri gömülmüş ve gömülmemiş veri örneklerinin istatistiksel özelliklerine odaklanır, olası ölçülebilir sapmaları belirlemeye kalkışır. Matris gömme (Westfeld 2001) veya diğer adıyla sendrom kodlama, taşıyıcı kanal gürültüsüne karşı daha fazla hassasiyet pahasına, konak sinyalde gizli veri biti başına en az değişikliği yapmayı amaçlar. Bu sayede istatistiksel farklılaşma *gizlenen bit sayısı* / *değiştirilen bit sayısı* oranında azaltılır, steganaliz yöntemlerine karşı dayanıklılıkta önemli bir iyileşme sağlanır. Ancak gizli veri bitleri, hala taşıyıcı sinyalin istatistiksel dağılımından bağımsız bir şekilde, sadece en az değişiklik yapılması amaçlanarak gömülmüştür; bu yüzden EÖB'nin steganalize karşı olan olası zafiyetleri azalan oranlarla olsa da matris gömmede de bulunur.

Kodlanmış konuşma sinyali içine saklanmış gizli veri, çıktı bit dizisinin Markov modelindeki sapmaları inceleyen bir steganaliz yöntemince tespit edilebilir. Entropiyi kaile almaksızın işleyen bir veri gömme yöntemi için, gizli veri içeren ve içermeyen bit dizilerin Markov zincir modellerinin birbirlerinden farklı olması beklenebilir. Matris gömme gibi veri gömme verimliliğini arttıran mekanizmalar, Markov zincir modelleri arasındaki farklılığı, arttırılan veri gömme verimliliği oranında indirgeyebilirler. Ancak herhangi bir şekilde gizli veri gömmenin istatistiksel modellere uyumlandırılmasıyla yerine getirilmesi, veri gömme verimliliğini arttırmaktan daha etkin bir savunma sağlayabilir.

Bu çalışmada tanıtılan özgün "entropisi düşük bit dizilerinde Markov modele sahip sendrom kodlaması", matris gömmenin Markov olarak modellenebilen tek boyutlu bit dizileri üzerinde çalışan iyileştirilmiş bir türevidir. Birim uzunluk başına aynı miktarda gizli veri gömebilmesine karşın, matris gömmeye kıyasla taşıyıcı sinyal üzerinde daha az istatistiksel farklılaşmaya yol açar. Yöntemin temel felsefesi, en az sayıda değişikliğin istatistiksel farklılaşma açısından her zaman en iyi tercih olmadığı öngörüsüne dayanır; bazen en az sayıda değil de daha fazla bit değiştirmek konak sinyalin (bit dizisinin) istatistiksel özelliklerinin korunması için daha doğru olabilir.

Çalışmada öncelikle, tek boyutlu bit dizilerinin Markov modellemesi konusu işlenecek, daha sonra elde edilen Markov zincir modellerinin steganalizde kullanımları üzerinde durulacaktır. Markov zincir modellerinin mevcut EÖB ve matris gömmede etkinliği tartışılacak, Markov zincir modellemeli steganalize karşı aynı esaslara bağlı kalarak işini yerine getiren yeni özgün bir gizli veri gömme yöntemi, "Markov modellemeli sendrom kodlaması yöntemi" tanıtılacaktır. Son olarak GSM 6.10 (*Global System for Mobile Communications* 6.10, Anonymous 1999) kodlayıcısı üzerinde pratik olarak gerçeklenen EÖB, matris gömme (Westfeld 2001) ve Markov modellemeli sendrom kodlamalı gizli veri gömme yöntemleri sırasıyla Markov zincir modellemeli steganaliz, kaotik özniteliklere göre steganaliz ve PESQ P.563 (Anonymous 2004) yazılımları tarafından sınanacaktır.

5.1 Tek Boyutlu Kodlayıcı Çıktısı Dizilerin Markov Zinciri Olarak Modellenmesi

Kodlanmış konuşma, müzik gibi çerçeveler ve/veya alt-çerçeveler halinde düzenlenmiş sinyallerin bazı kısımları ergodik Markov zincirleri (Meyn ve Tweedie 1993) ile modellenebilir. Konuşma ve müzik sinyalleri kodlama esnasında çerçeveler halinde işlenerek kodlandığından, kodlanmış bit dizisindeki çeşitli alanlar birbirlerinden bağımsız durum makineleri olarak kabul edilebilir. Söz konusu bu bağımsız durum makinelerine ait durumlar çerçeveler ilerledikçe birbirleri arasında geçişler akdeder. Herhangi bir bit alanını modelleyen bir durum makinesi L tane farklı durum içeren $S = \{s_1, s_2, ... s_L\}$ durum kümesi ve P geçiş olasılık matrisi ile ifade edilebilir (5.1):

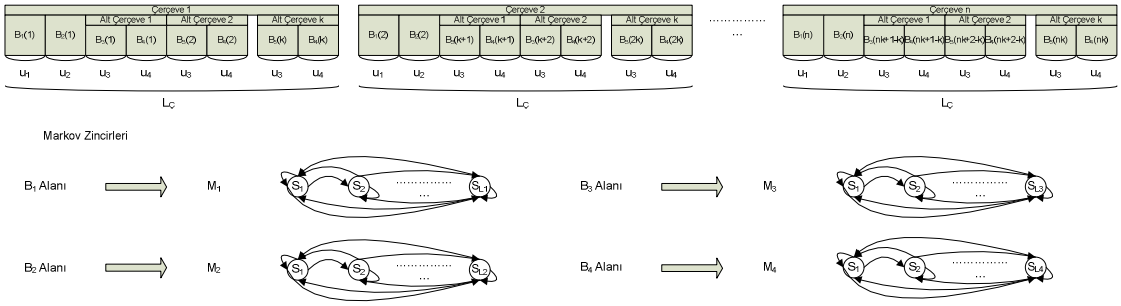
$$P = \left[\begin{pmatrix} p_{11} & \cdots & p_{1L} \\ \vdots & \ddots & \vdots \\ p_{L1} & \cdots & p_{LL} \end{pmatrix} \right] \tag{5.1}$$

Markov zinciri modellemesinde kodlayıcıya ait yeterli uzunlukta çıktı kullanılarak, her tanımlanmış bit alanının durumları arası geçişlerini modelleyen bir *P* matrisi hesaplanır. π_i, her durumun gerçekleşme olasılığı olmak üzere, *P* matrisini oluşturan durum geçiş olasılıkları (5.2)'ye sokularak, bit alanının kullanım verimliliği *E* bit alanı entropisi (Rahikka vd. 1999) cinsinden rakamsal olarak ifade edilebilir.

$$E = -\sum_{i=1}^{L} \pi_i \sum_{j=1}^{L} p_{ij} \log p_{ij} \tag{5.2}$$

Durum makinesi olarak tanımlanan bir bit dizisi alanında, istatistiksel olarak gizlilik sağlayarak saklanabilecek en yüksek veri miktarı, alanın entropisi kadardır. Ancak gizliliğin olabilecek en üst düzeyde derece sağlanması koşuluyla, bir alanın entropisine eş miktarda gizli veri gömülmesi konak diziye kendi bilgisini gönderecek hiç alan bırakılmaması sonucunu doğurur. Bu yüzden normal koşullar altında hiçbir alana entropisi kadar gizli veri gömülmez. Alanların EÖB alanlarına gizli veri eklenmesi, toplam entropisinin sınırlandırılmış bir kısmının steganografik amaçlara tahsis edilmesi manasına gelmektedir.

Şekil 5.1'de hayali bir kodlayıcıya ait çıktının ergodik Markov zinciri olarak modellenmesi gösterilmiştir. Çıktı çerçevelerden çerçeveler ise alt-çerçevelerden oluşmaktadır. B_1 ve B_2 alanları çerçeve yapılarının, B_3 ve B_4 ise alt-çerçeve yapılarının içinde yer almaktadır. Her bir alan diğerlerinden bağımsız bir Markov zinciri ile modellenmektedir. B_1, B_2, B_3 ve B_4 alanlarını modelleyen Markov zincirleri sırasıyla MZ_1, MZ_2, MZ_3 ve MZ_4 olarak tanımlandığında her bir zincir sırasıyla, $L_1 = 2^{u_1}, L_2 = 2^{u_2}, L_3 = 2^{u_3}, L_4 = 2^{u_4}$ sayıda farklı duruma sahiptir. Alanların çerçeve veya alt-çerçevelerde bulunması, alanları tanımlayacak durum makinelerinin birim zamanda gerçekleştirecekleri basamak sayısını belirlemektedir.



Şekil 5.1 Hayali bir kodlayıcının çıktısının çerçeveleri içerisinde yer alan bit alanlarının Markov zincirleri olarak modellenmesi

Bir kodlayıcı çıktısına ait çerçeve yapısında Markov zincirleri ile modellenen bit alanlarının birbirlerinden bağımsız olduğu varsayımı haddizatında pratik uygulamalar için pek doğru olmayabilir. Öte yandan, örnek olarak her biri u bite sahip K adet alan olarak modellenen bir çerçeve parçasının $u \times K$ bit uzunluğunda tek bir alan olarak modellenmesi de fiziksel tatbik hudutları yüzünden mümkün değildir. Bit alanının büyüklüğü arttırıldıkça, P matrisinin saklaması için gerekli olan hafıza miktarı muazzam miktarda artar, bundan da önemlisi bahse konu P matrisinin yakınsanabilmesi için çok daha uzun kayıt çıktılarına ihtiyaç duyulur.

5.2 Markov Zincir Modelleri Kullanarak Steganaliz

Eklenen gizli veri bitleri, taşıyıcı sinyale ait Markov zincir modelindeki durumlar arası geçiş olasılıklarında sapmalara neden olabilir. Söz konusu geçiş olasılıklarındaki sapmalar, gizli veri varlığını ifşasını arzulayan bir steganaliz yönteminin tasarımında değerlendirilebilir. Markov zincir modellemeli steganaliz yöntemi literatürdeki (Yuan 1999) MM(1) tipi dizi sınıflandırıcısından faydalanılarak türetilmiş olup, söz konusu yöntemin K adedinin (K: Bir çerçevede tanımlanmış bit alanı sayısı) paralel olarak çalıştırıldığı bir türevidir. Yuan (1999) makalesinde MM(1) sınıflandırıcısı, hücre içi proteinlerin genetik bilgilerini taşıyan DNA kaynağının tespitinde kullanılmaktadır.

Kodlanmış konuşma sinyaline saklı bir vaziyette bulunan gizli veri bitlerinin varlığının açığa çıkartılabilmesi için ilk olarak bu iş için kullanılacak bir steganalizörün eğitilmesi gerekmektedir. Bir önceki paragrafta da belirtildiği üzere, steganalizör aslında iki adet

sınıf arasından seçim yapan K adet MM(1) dizi sınıflandırıcısından, yani $2 \times K$ adet basit MM(1) modelinden oluşmaktadır. Steganalizör eğitimi için *TUS* olarak isimlendirilmiş PCM biçimindeki konuşma dizisi kullanılır. Denklem (5.3)'te de gösterildiği üzere, *TUS* dizisi eğitim sonuçlarının yakınsanabilmesi için (yeterli sayıda) N_S adet eleman içeren bir dizidir ve "*Train Uncoded Stream*" anlamına gelmektedir.

$$TUS = \{TUS^1, TUS^2, \dots \dots TUS^{N_S}\} \tag{5.3}$$

İlk olarak *TUS* dizisi, hem gizli veri ekleyen hem de orijinal kodlayıcılarla kodlanır ve bu sayede kodlanmış *TECS* ve *TCCS* dizileri elde edilir. (5.4) ve (5.5)'te detaylandırılan dizilerden *TECS* "*Train Embedded Coded Stream*", *TCCS* "*Train Clean Coded Stream*" anlamına gelmektedir, her iki dizi de N_W adet çerçeveden oluşmaktadır, kodlayıcının ürettiği çerçevelerin uzunluğu L_W kadar olduğundan, toplam uzunlukları L_WxN_W değerine eşittir.

$$TCCS = \{TCCS^1, TCCS^2, \dots \dots TCCS^{N_W x L_W}\} \tag{5.4}$$

$$TCCS = \{TECS^1, TECS^2, \dots \dots TECS^{N_W x L_W}\} \tag{5.5}$$

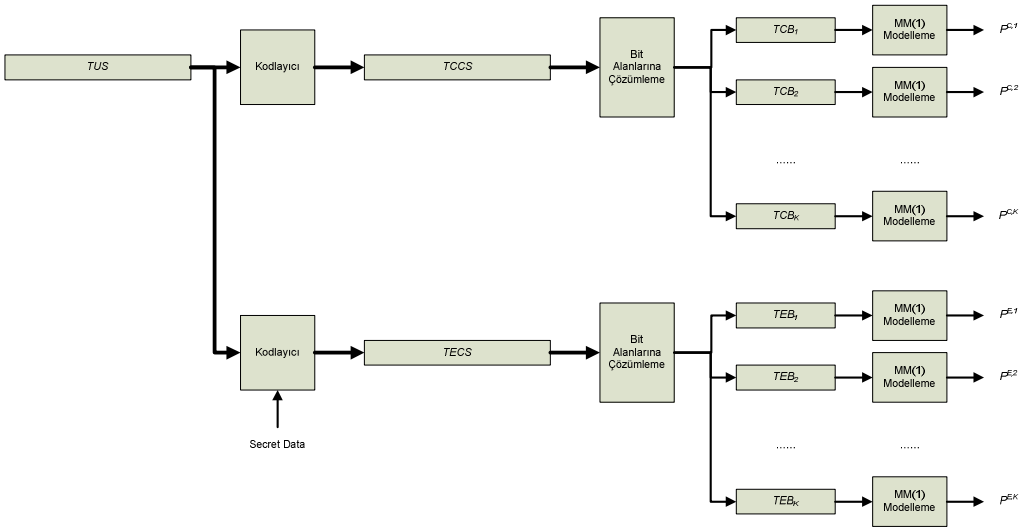
Kodlanmış bit dizilerinin elde edilmesinden sonra, Şekil 5.2'de grafiksel olarak gösterildiği gibi, biri gizli veri içeren diğeri içermeyen iki dizi bit alanlarına çözümlenir, her bir örnek için K adet, ikisi için toplam $2 \times K$ adet dizi üretilir. (TEB_i : "i^{th} *Train Embedded Bit Field*", TCB_i : "i^{th} *Train Clean Bit Field*") elde edilir. Her bir dizi eğitim dizisi *TUS*'un çerçeve sayısı, N_W kadar elemana sahiptir ((5.6) ve (5.7)).

$$TEB_i = \{TEB_i{}^1, TEB_i{}^2, \dots . TEB_i{}^{N_W}\} \tag{5.6}$$

$$TCB_i = \{TCB_i{}^1, TCB_i{}^2, \dots . TCB_i{}^{N_W}\} \tag{5.7}$$

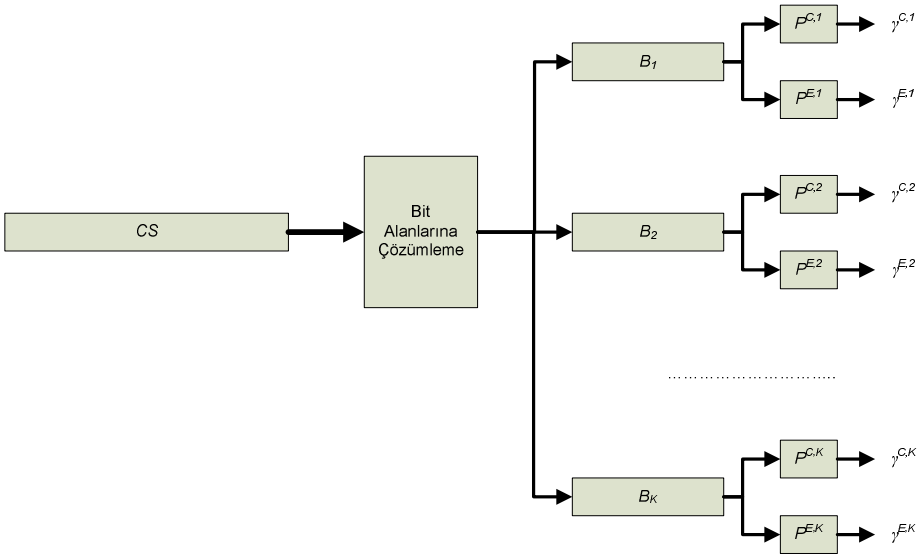
Çözümleme işlemini takiben her bir sınıfın her bir dizisinin Markov zincir modeli, yani P matrisi hesaplanır. P matrisleri ait oldukları bit alanına ve hesaplandıkları diziye göre

isimlendirilmişlerdir: $P^{E,i}$: "*P matrix of ith bit field's sequence with embedded data*", $P^{C,i}$: "*P matrix of ith bit field's sequence without embedded data*". Bu sayede steganalizörü oluşturacak toplam $2 \times K$ adet P matrisi üretimi tamamlanmış olur.



Şekil 5.2 Markov zincir modellemeli steganalizörün eğitilmesi

İçinde gizli veri olup olmadığı bilinmeyen girdi kodlanmış konuşma dizisi (*CS*) de aynen eğitim sırasında tatbik edildiği üzere bit alanlarına ait alt-dizilere ayrıştırılır (Şekil 5.3).



Şekil 5.3 Kodlanmış konuşma sinyaline ait bir bit dizisin steganalizi

Steganaliz işlemi, alt-dizilerin aidiyet olasılılıklarının hesaplanmasıyla gerçekleştirilir. B_i alt-dizisinin gizli veri içerme ($\gamma^{E,i}$) ve içermeme olasılıkları ($\gamma^{C,i}$) (5.8) ve (5.9)'da sunulmuştur. Gizli veri içermeme olasılığının hesaplanma formülündeki α sabiti ($0 < \tau < \infty$) steganalizörün optimize edilmesinde, steganalizöre ait ROC (karar vericinin etkinliği – *receiver operating characteristics*) çizilmesinde işe yaramaktadır, varsayılan değeri 1'dir.

$$\gamma^{C,i} = \prod_{j=2}^{N_B} p_{j\,j+1}^{C,i} \tag{5.8}$$

$$\gamma^{E,i} = \prod_{j=2}^{N_B} \tau . p_{j\,j+1}^{E,i} \tag{5.9}$$

Her bir alt-dizinin sınıflar için aidiyet olasılıkları bulunduktan sonra *CS* dizisinin bütünü için gizli veri içerme (γ^{E}) ve içermeme (γ^{C}) olasılıkları hesaplanır (Denklem (5.10) ve (5.11)) .

$$\gamma^{C} = \prod_{i=1}^{K} \gamma^{C,i} \tag{5.10}$$

$$\gamma^{E} = \prod_{i=1}^{K} \gamma^{E,i} \tag{5.11}$$

Son olarak, hesaplanan iki değer birbirleri ile karşılaştırılır, eğer γ^{E} γ^{C}'den büyük ise *CS* dizisinin içinde gizli veri içerdiği, aksi takdirde gizli veri içermediği kanısına ulaşılır.

5.3 PESQ P.563 Steganalizi

Taşıyıcı sinyal içerisine saklanmış gizli veri, konuk olduğu sinyal üzerinde çeşitli bozulmalara yol açar. Bahse konu bozulmalar, sinyalin tüm özelliklerine yayılabilir: İstatistiksel özellikler, algısal kalite ..vb. Konuşma sinyali üzerinde geliştirilmiş pek çok steganografi uygulamasında (Tian vd. 2009, Shahbazi vd. 2010, Mazurczyk ve Lubacz

2009, Liu ve Chen 2004) çeşitli PESQ yazılımları sıklıkla başvurulan bir kalite değerlendirme kriterleri olmuştur. Bu durumun haricinde, Ozer vd. (2003) audio sinyalleri içinde steganografik uygulamaların tespiti için, steganaliz amaçlı olarak doğrudan audio kalite metriklerinin kullanılabileceğini belirtmiştir. Ozer vd. (2003) önerdiği yaklaşıma dayanarak, konuşma sinyaline has P.563 (Anonymous 2004) yazılımı da steganaliz yöntemi olarak kullanılabileceği düşünülmüştür.

PESQ P.563 steganalizi, gizli veri içeren ve içermeyen sinyallerin P.563 sonuçlarının birbirlerinden farklı olmasına dayanmaktadır. Eklenen gizli veri, taşıyıcı sinyal içerisinde ek bozulmalara neden olacağından, muhtemelen PESQ skorlarında düşüşlere yol açacak, bu sayede gizli veri içeren ve içermeyen PESQ sonuç dağılımları birbirlerinden ayrışma gösterecektir. PESQ sonuç dağılımlarının birbirlerinden en iyi ayrıştıran eşik değerinin tespiti için eğitim aşamasına ihtiyaç olacaktır. Steganaliz işlemi için analiz edilecek sinyalin P.563 skoru elde edilir. Daha sonra elde edilen bu skor değeri eşik değeri ile karşılaştırılarak gizli verinin var olup olmadığı yargısına ulaşılır. Gerekirse eşik değeri ile oynanarak ROC'lar çizilebilir.

5.4 Bit Alanlarının En Önemsiz Bitleri Üzerinde Sendrom Kodlaması

Sendrom kodlaması (Bierbrauer 2004), *C(N,K)* doğrusal blok kodlayıcılar kullanılarak bir çerçevenin veya alt-çerçevenin içinde tanımlanmış *N* tane bit alanı üzerinde uygulanır. *f.* çerçevede yer alan *N* tane bit alanının oluşturduğu $B^f = \{b_1^f, b_2^f, \dots\ b_N^f\}$ vektörünün en önemsiz bitlerinden *N* bitlik bir vektör $D^f = \{d_1^f, d_2^f, \dots\ d_N^f\}$ oluşturulur. Denklem (5.12)'de en önemsiz bit alma işleminin matematiksel anlatımı verilmiştir:

$$d_i^f = b_i^f \mod 2 \qquad 1 \le i \le N \tag{5.12}$$

Oluşturulan vektör öyle bir hata vektörü $Y(c) = \{y_1^f(\mathrm{c}), y_2^f(\mathrm{c}), \dots\ y_N^f(c)\}$ ile toplanır ki (5.13), elde edilen *N* bitlik vektör $X(c) = \{x_1(c), x_2(c), \dots\ x_N(c)\}$, *H* parite kontrolü matrisi ile çarpıldığında çarpım sonucu gizli veri bitlerine eşit olur (5.14). Bu şartı sağlayabilen 2^K adet *Y* vektörü olduğundan, 2^K adet de hata vektörü vardır. *c* sayısı şartı

sağlayan 2^K adet olasılıktan hangisi olduğunu belirten bir indekstir. Bu sayede N alanda N-K adet gizli veri biti ($W^f = \{w_1^f, w_2^f, \ldots\ w_{N-K}^f\}$ vektörü) taşınmış olur.

$$X^f(c) = D^f + Y^f(c) \tag{5.13}$$

$$W^f = HX^f(c) \tag{5.14}$$

W^f gizli verisinin sendrom olarak gömülmesini sağlayacak $X^f(c)$ vektörü seçildikten sonra, bit alanları bu sendromu taşıyacak şekilde güncellenir. Denklem (5.15)'te $\widehat{b_i^f}(c)$'nin en önemsiz bitleri atılıp yerlerine sendromu içeren $x_i^f(c)$ bitleri konulmaktadır:

$$\widehat{b_i^f}(c) = \left(b_i^f \ mod\ 2\right) + x_i^f(c) \quad 1 \le i \le N \tag{5.15}$$

Güncellenmiş $\widehat{B^f}(c)$ vektörü, gizli veri içermeyen konak B^f vektörünün en önemsiz bitlerine, W^f sendromunu sağlayan $X^f(c)$ vektörünün gömülmüş halidir.

5.5 Matris Gömme

Matris gömme, bu çalışmada kendisine verilen kısaltılmış ismiyle LSB_M, F5 gizli veri gömme yönteminde (Westfeld 2001) kullanılan özel bir tür sendrom kodlamasıdır. Kodlama sırasında D vektörü içerisinde en az sayıda 1 barındıran Y hata vektörü ile toplanır; C(N, K) doğrusal blok kodlamada $N = 2^g - 1,\ K = 2^g - (g + 1)$ (g: Kod kelimesi başına gömülen gizli bit sayısı) olarak tanımlanırsa, N bit içinden en fazla birinin değiştirilmesi tüm olası sendrom durumları yaratılabilir. Gizli veri gömme işleminin en az sayıda bit değiştirilerek yapılması suretiyle gömme verimliliği (Cox vd. 2008) arttırılır, konak sinyalin istatistiksel özelliklerinde daha az sapmalara neden olunur. İstatistiksel özelliklere daha fazla sadakat gösterilmesine karşın hala gizli veri konak sinyalin istatistiksel özellikleri, evvelki çerçevelerin içerikleri gibi bilgiler dikkate alınmaksızın taşıyıcı sinyale katılmaktadır.

5.6 Markov Zincirleri ile İstatistiksel Modelleme Yapılarak Sendrom Kodlama

Bu çalışmada önerilen, Markov zincirleri ile istatistiksel modelleme yapılarak sendrom kodlamasında (LSB_MM) da gizli veriler denklem 14 ve 15'in uygulanmasıyla gömülür. Ancak bu sefer, *D* vektörü matris gömmedeki gibi her zaman en az sayıda "1" içeren hata vektörüyle toplanmaz.

Eğer gömme işlemi uygulanacak bit alanının en önemsiz bitleri zaten gerekli sendrom değerini sağlıyorsa, doğal olarak üzerinde herhangi bir değişikliğe gidilmesi gerekmez. Ancak söz konusu koşul sağlanmıyorsa, bu durumda en önemsiz bitlerin sendromu sağlayacakları hale getirilmeleri, yani değiştirilmeleri gerekir. Değişikliği sağlayan 2^K tane alternatif vardır, buradaki temel mesele alternatifler içerisinden hangisinin seçileceğidir. Matris gömmedeki gibi en az değişiklik gerektiren alternatif seçilebileceği gibi, Markov zincir modeline göre görülmesi en olası deseni oluşturan alternatif de tercih edilebilir. Bu ifadeyi tersten tekrardan yazıya dökersek, Matris gömmenin seçtiği, en az değişikliğe yol açacak ideal izlenimi veren tercih belki de Markov zincir modeline göre en az olası olan – hatta olmaması gereken – bir seçenek olabilir.

Özetle, Markov zincir modeline göre değişikliği sağlayan 2^K tane alternatifin her birinin kendine özgü – diğerlerden farklı – bir oluşma olasılığı vardır ve gizli veri gömme işlemi – alternatif seçme – her bir alternatifin oluşma olasılığını dikkate alarak yapılmalıdır. Her zaman en olası alternatifin seçilmesi de Markov zincir modeline uygunluk sağlamaz, az olası alternatifin nadir, çok olası alternatifin sık seçileceği, daha doğru bir ifadeyle her bir alternatifin kendi oluşma olasılığına doğru orantılı oranda seçileceği bir mekanizma tasarlanmalıdır. İşte bu çalışma kapsamında geliştirilen özgün mekanizma, (Markov zincirleri ile istatistiksel modelleme yapılarak sendrom kodlama yönteminin omurgası) alışılageldik sözde kod halinde sunulmamış, daha anlaşılır olabilmesi için bir çizelge biçiminde takdim edilmiştir (Çizelge 5.1).

Çizelgenin 1. sütununa gizli veriyi sendromlarında taşıyan kod kelimeleri, $X^f(c)$ vektörleri yazılır. Çizelgenin 2. sütununda ise en önemsiz bitlerinde 1. sütundaki kod

kelimesinin gömülü olduğu $\widehat{B^f}(c)$ vektörleri yer alır. Yeterli uzunlukta çıktının içinde yer alan bit alanlarının analiz edilmesiyle oluşturulan *P* matrisi ise $\widehat{B^f}(c)$ vektör alternatiflerinin gerçekleşme olasılıklarının $\widehat{B^{f-1}}(c)$ vektörüne göre hesaplanabilmesini mümkün kılar. Denklem (5.16)'da *c*. $\widehat{B^f}(c)$ vektör alternatifinin gerçekleşme olasılığı α(*c*) verilmiştir.

$$\propto(c)=\prod_{i=1}^{N} p_{b_i^{f-1}\widehat{b_i^f}(c)} \tag{5.16}$$

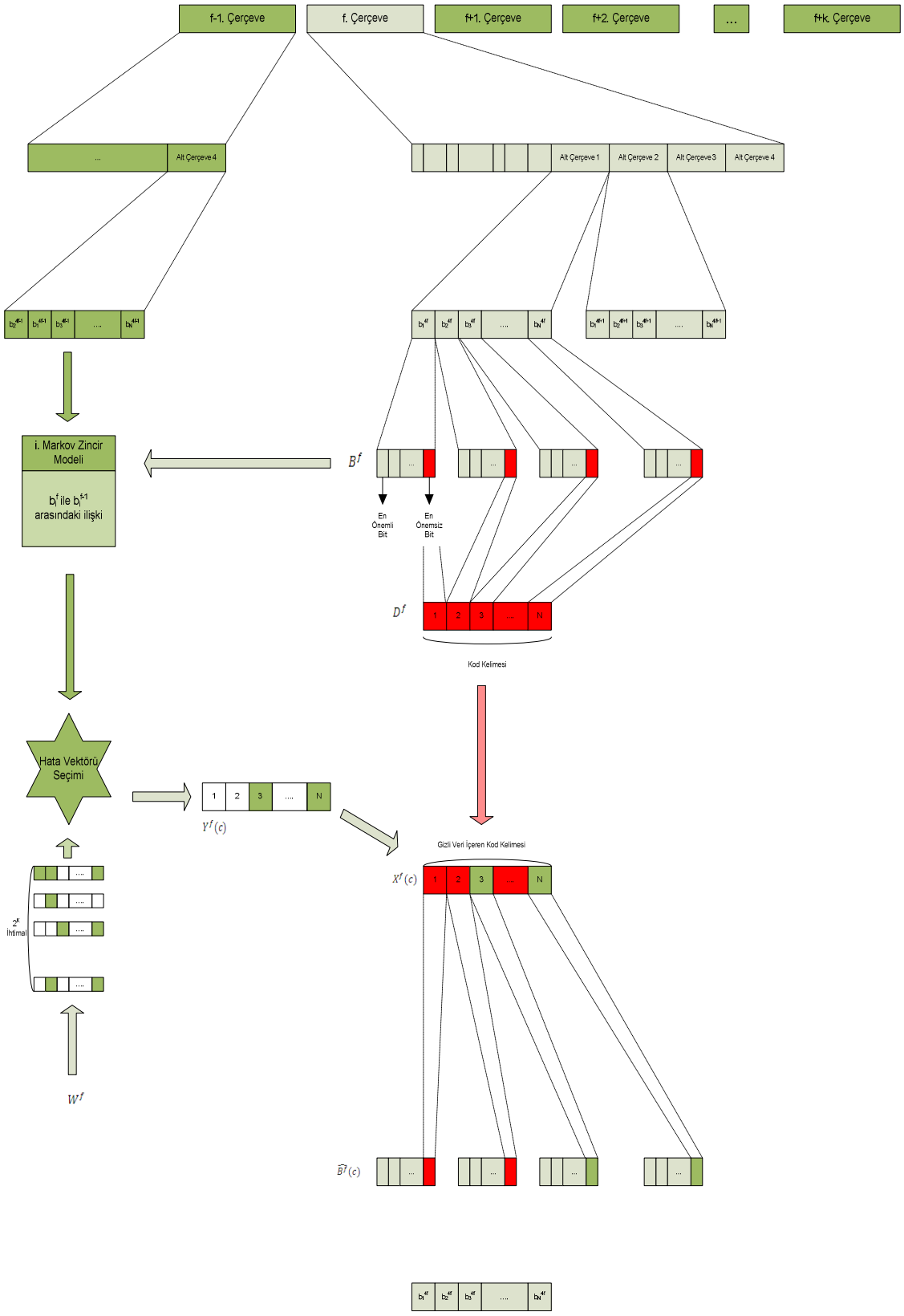
Çizelge 5.1 Markov zincirleri ile bit alanlarına gizli veri gömme

Sendromu Sağlayan Kod Kelimeleri	Gizli Veri Gömülü Olası Bit Alanları	α	Seçim Koşulu
$x_1^f(1) \dots x_N^f(1)$	$\widehat{b_1^f}(1) \dots \widehat{b_N^f}(1)$	$\alpha(1)$	$0 \le \varphi < \alpha(1)$
$x_1^f(2) \dots x_N^f(2)$	$\widehat{b_1^f}(2) \dots \widehat{b_N^f}(2)$	$\alpha(2)$	$\alpha(1) \le \varphi < \alpha(1) + \alpha(2)$
…….	…….	…….	…….
$x_1^f(2^K) .. x_N^f(2^K)$	$\widehat{b_1^f}(2^K) \dots \widehat{b_N^f}(2^K)$	$\alpha(2^K)$	$\sum_{n=1}^{2^K-1} \alpha(n) \le \varphi \le \sum_{n=1}^{2^K} \alpha(n)$

Çizelgenin 4. sütunu gizli veriyi taşıyacak $\widehat{B^f}(c)$ vektörünün seçiminde kullanılmaktadır. Seçim için önce değeri 0 ila 1 arasında değişen bir rastgele sayı, γ üretilir. Denklem (5.17)'de de gösterildiği gibi üretilen sayı tüm $\alpha(c)$ değerlerinin toplamıyla çarpılır ve seçim koşulunun testinde kullanılacak φ değeri elde edilir.

$$\varphi = \gamma \sum_{n=1}^{2^K} \alpha(n) \tag{5.17}$$

φ değeri elde edilmesinden sonra, Çizelge 5.1 taranarak koşulu sağlayan $\widehat{B^f}(c)$ konak sinyaldeki B^f alanının üzerine yazılır. Markov zincirleri ile istatistiksel modelleme yapılarak sendrom kodlamalı veri gömme yönteminin yazılı anlatımına ek olarak Şekil 5.4'te blok şeması da sunulmuştur. Şemada PCM konuşma sinyalinin analiz edilmesi sırasında oluşturulmuş *f.* çerçevenin *1.* alt-çerçevesinin içine gizli veri bitlerinin nasıl katıldığı gösterilmektedir. Markov zincir modeli kullanımının bir niteliği olan, gömülecek gizli veri bitlerinin bir önceki alt-çerçeveye bağlı kalınarak seçilmesi bilhassa vurgulanmaktadır. Şekilde pembe renk ile ifade edilen alanlar gizli veri bitlerinin taşınacağı B_i^f bit alanlarının en önemsiz bitleridir, kırmızı renk ise gizli veri gömme uğruna değiştirilecek kısımları temsil etmektedir. Şemada yer alan hata vektörünün seçimi bloğu, φ değerini üretmekten, Çizelge 5.1'in taranmasından ve böylece müsait bir $Y^f(c)$ vektörünün seçiminden sorumludur.

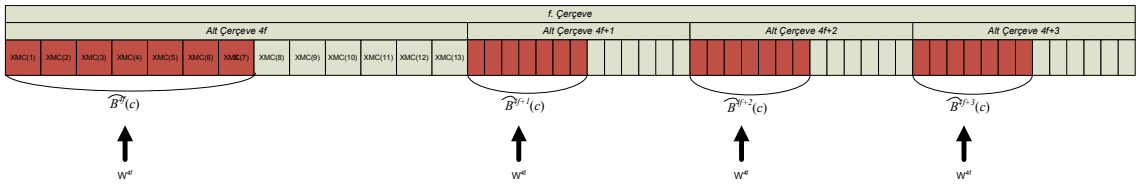


Şekil 5.4 Markov zincirleri ile istatistiksel modelleme yapılarak sendrom kodlamasının blok şeması

5.7 GSM 6.10 Kodlayıcısı Üzerinde MMSK Uygulaması

Bu makalede önerilen özgün veri gizleme yöntemi (Markov zincirleri ile istatistiksel modelleme yapılarak sendrom kodlama), dünyada en yaygın olarak kullanılan kodlayıcılardan biri olan GSM 6.10 üzerinde uygulanmıştır. GSM 6.10 kodlayıcısı 20 ms içinde 8 KHz'de örneklenmiş 160 örneği 33 bayt olarak kodlar. Gizli veri, LTP (*long term prediction*) analizi sırasında elde edilen RPE (*regular pulse excitation*) darbe indekslerinin en önemsiz bitlerine saklanır. RPE darbe indeksleri GSM 6.10 standardında ve Linux için hazırlanmış açık kaynak kodlu yazılım kütüphanesinde XMC olarak isimlendirilmiştir.

Şekil 5.5'te gösterildiği üzere darbe indeksleri GSM 6.10 çerçevesinin içerisinde bulunan alt-çerçevelerde taşınmaktadır. Bir çerçeve dört adet alt-çerçeveden oluşmaktadır, her bir alt çerçevede üçer bitlik 13 adet darbe indeksi bulunur. 3 adet gizli veri biti, 13 indeksten ilk 7'sine C(7,4) sendrom kodlama vasıtasıyla gömülür. Bir çerçeveye toplam 12 gizli veri biti yerleştirilmiş olur; yaratılan gizli kanalın kapasitesi *600* bps'tir.



Şekil 5.5 GSM 6.10 çerçevelerine Markov modelli sendrom kodlama ile gizli veri gömme

5.8 Gizli Veri Gömme Yöntemlerinin GSM 6.10 Üzerindeki Performansları

Bu kısımda GSM 6.10 üzerinde gerçeklenen Markov modelli sendrom kodlaması (LSB_MM), EÖB (LSB) ve matris gömme yöntemlerinin (LSB_M) performans ve steganaliz sonuçları sunulmuştur. Gizli veri gömme yöntemlerinin adil bir şekilde yargılanabilmesi için, entropi ve gizli veri gömme verimliliği olmak üzere iki

performans kriterinden, PESQ P.563, kaotik özniteliklere göre ve Markov zincir modellemeli olmak üzere üç steganaliz yönteminden faydalanılmıştır.

İlk olarak gizli veri gömen yöntemlerin ve standart GSM'in entropi değerleri (Rahikka vd. 1999) ve gizli veri gömme verimlilikleri (Cox vd. 2008) hesaplanmış, bulunan sonuçlar birbirleri ile mukayese edilmiştir. Çizelge 5.2'de, *#gb* çerçeve başına gömülen gizli bit sayısını, *#db* ise gizli veri gömme amacıyla çerçeve başına değiştirilen bit sayısını ifade etmekteyken *#gb* / *#db* oranı bit gizli veri gömme yönteminin verimliliğini tanımlamaktadır. *E* entropi değeri ise bit cinsinde verilmiştir ve $13 \times 4 \times 3 = 156$ bitlik alanın ne kadar verimlilikle kullanıldığını belirtmektedir.

Çizelge 5.2 Gizli veri gömme yöntemlerinin performans ölçütlerine göre karşılaştırılması

Yöntemin Adı	E	#gb	#gb/#db
Standart GSM 6.10	132.48	0	0
Matris gömmeli GSM 6.10	133.63	12	3.429
EÖB'li GSM 6.10	134.26	12	2.000
Markov modelli sendrom kodlamalı GSM 6.10	132.34	12	1.1206

Performans ölçütlerinde iki nokta oldukça dikkate şayandır: Birinci nokta, Markov modellemeli sendrom kodlama ile gizli veri gömme uygulanınca gizli veri içeren çıktının entropisi gizli veri içermeyen bir sinyalin entropisine çok benzemektedir. Başka bir ifadeyle gizli veri gömülü sinyalin istatistiksel özelliklerini gizli veri içermeyen sinyale yakınlaştırma entropi kriterine göre başarılmıştır. İkinci nokta ise, entropiyi çok az etkileyerek gizli veri gömme bedelini ortaya çıkarmış, gizli veri gömme verimliliğinde belirgin düşüş gözlemlenmiştir. Markov modellemeli sendrom kodlama,

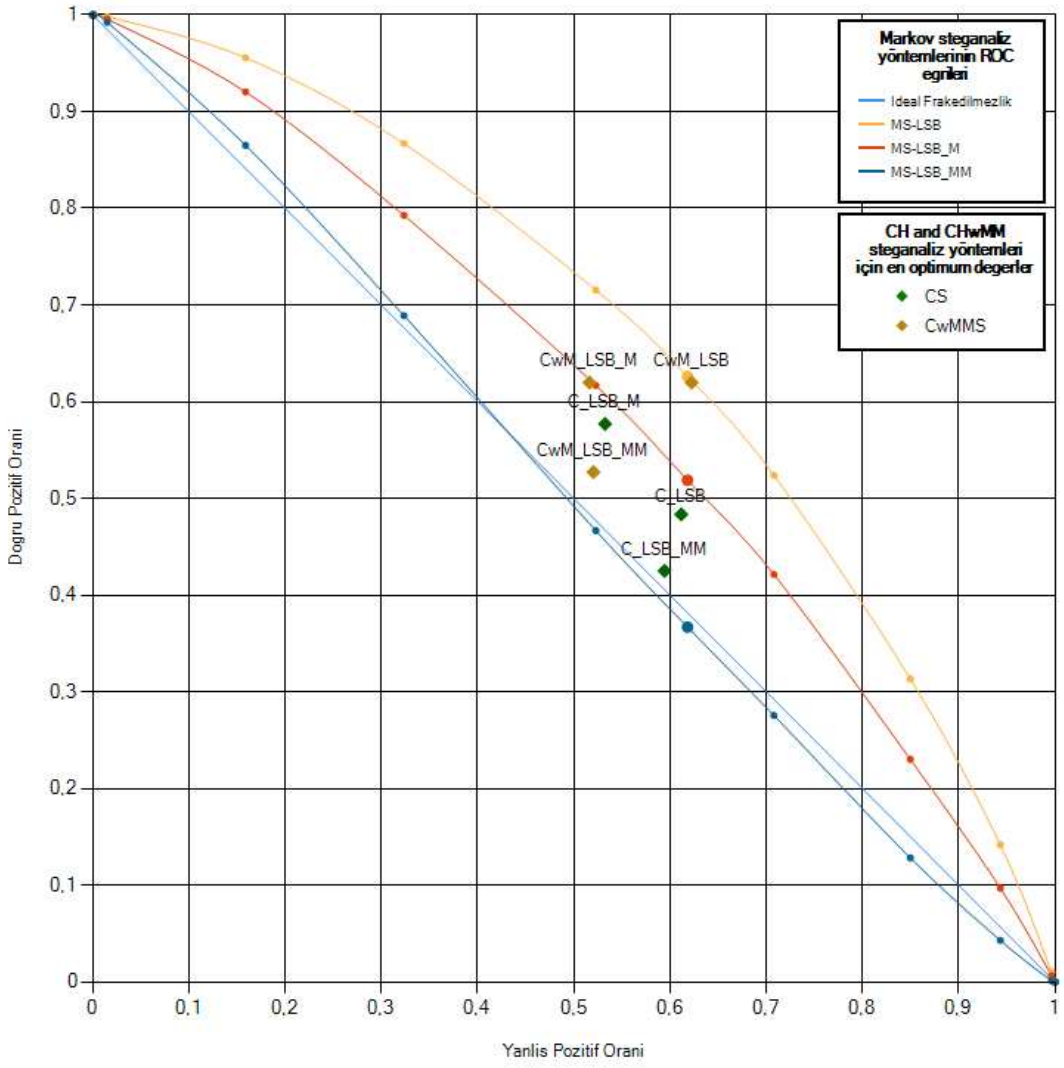
dağılıma uyabilmek adına çok sayıda biti değiştirmekten kaçınmamıştır, matris gömmeyle aynı sendrom kodlama mantığını paylaşsa da, gizli veri gömme verimliliği konusundaki etkisi matris gömmenin tam tersi yönünde olmuştur.

Bir nevi ön bilgi sağlayan performans kriterlerinin yorumlanmasının ertesinde, çalışma kapsamındaki yöntemler steganografi açısından çok daha fazla ehemmiyet taşıyan steganaliz yöntemleri ile test edilmişlerdir. Daha önceden de belirtildiği gibi, birbirlerinden farklı üç ana steganaliz yöntemine başvurulmuştur. Yöntemlerden PESQ P.563 esasında bir steganaliz yöntemi değildir, steganalizle uzaktan yakından bir alakası yoktur, zaten yüksek başarımlı ayrıştırma yapabilmesi beklenmemektedir. Testlere gizli verinin yaratmış olduğu bozulmanın insan duyuları tarafından ne ölçüde tespit edilebildiğinin irdelenmesi için dâhil edilmiştir. Kodlanmış konuşma sinyalinde Markov zincir modellemeli steganaliz ise bu çalışmaya özgüdür, diğer taraftan kaotik özniteliklere göre steganaliz (Kocal vd. 2008) her türlü gizli veri ifşasında görev alacak mahiyette genellik arz etmektedir. Üç ana yöntem haricinde, kaotik öznitelikli steganaliz ile Markov zincir modellemeli steganalizin birleştirilmesiyle melez bir steganaliz yöntemi de türetilmiştir. Melez steganaliz yönteminde kaotik öznitelikler ile MM aidiyet olasılıklarını bir araya getirip, kompozit bir öznitelik vektörü oluşturmaktadır. Elde edilen kompozit öznitelik vektörü tıpkı kaotik özniteliklerin sınıflandırılmasında olduğu gibi sınıflandırıcıya sokulmakta, böylelikle steganaliz sonuçlarının üretilmesinde kullanılmaktadır.

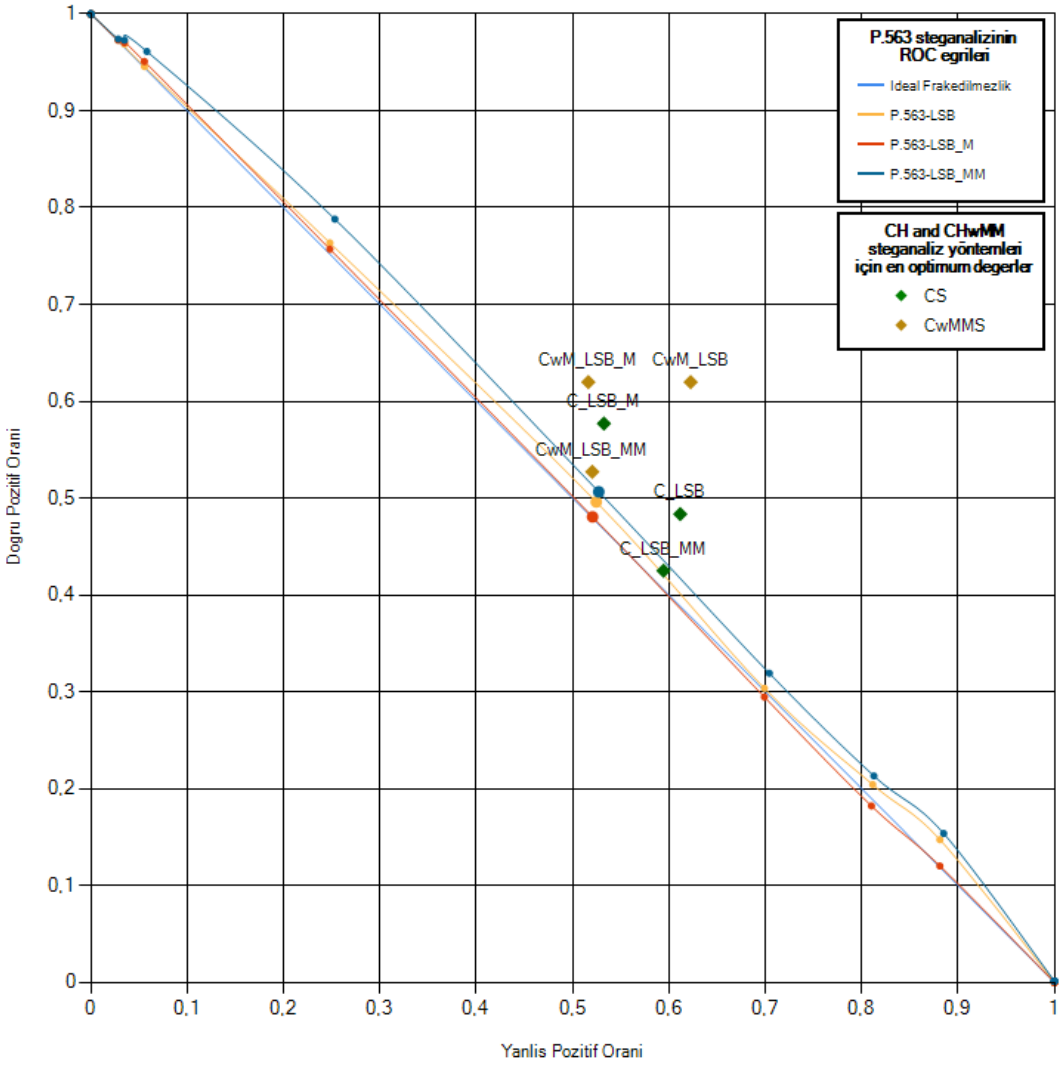
Steganaliz yöntemlerinin veri gizleme yöntemleri karşısındaki performansları (veya zıt bir anlatımla gizli veri gömme yöntemlerinin steganaliz yöntemleri karşısındaki performansları) ROC eğrileri ve/veya optimum etkinlik noktaları cinsinden sunulmuştur. PESQ (P.563) ve Markov zincir modellemeli steganaliz yöntemlerinde, bazı sabitler değiştirilerek kolaylıkla ROC'lar çizilebilirken kaotik özniteliklere göre steganalizde ve onun türevi olan melez steganalizde sadece optimum etkinlik noktaları hesaplanabilmektedir. Üç farklı gizli veri gömme yönteminin, dört farklı steganalize karşı performanslarının hepsinin bir arada sunulmaya çalışılması, sonuçların yer alacağı grafiğin karmaşıklığını arttıracağından, bulunan sonuçlar iki ayrı grafik arasında (Şekil 5.6 ve 5.7) paylaştırılmış, kullanılan kısaltmalar Çizelge 5.3'te tanımlanmıştır:

Çizelge 5.3 Performans grafiklerindeki kısaltmaların tanımları

Kısaltmalar	**Kullanılan Steganaliz Yöntemi**	**Sınanan Gizli Veri Gömme Yöntemi**
MS-LSB	Markov Zincir Modelleri ile Steganaliz	EÖB'li GSM 6.10
MS-LSB_M	Markov Zincir Modelleri ile Steganaliz	Matris gömmeli GSM 6.10
MS-LSB_MM	Markov Zincir Modelleri ile Steganaliz	Markov modelli sendrom kodlamalı GSM 6.10
CS-LSB	Kaotik Öznitelikler ile Steganaliz	EÖB'li GSM 6.10
CS-LSB_M	Kaotik Öznitelikler ile Steganaliz	Matris gömmeli GSM 6.10
CS-LSB_MM	Kaotik Öznitelikler ile Steganaliz	Markov modelli sendrom kodlamalı GSM 6.10
CwMMS-LSB	Hem Markov zincir modelleri ile hem de kaotik öznitelikler ile steganaliz	EÖB'li GSM 6.10
CwMMS-LSB_M	Hem Markov zincir modelleri ile hem de kaotik öznitelikler ile steganaliz	Matris gömmeli GSM 6.10
CwMMS-LSB_MM	Hem Markov zincir modelleri ile hem de kaotik öznitelikler ile steganaliz	Markov modelli sendrom kodlamalı GSM 6.10
P.563-LSB	P.563 ile steganaliz	EÖB'li GSM 6.10
P.563-LSB_M	P.563 ile steganaliz	Matris gömmeli GSM 6.10
P.563-LSB_MM	P.563 ile steganaliz	Markov modelli sendrom kodlamalı GSM 6.10



Şekil 5.6 Steganaliz yöntemlerinin vs gizli veri gömme yöntemleri



Şekil 5.7 Steganaliz yöntemleri vs gizli veri gömme yöntemleri

Şekil 5.6 ve 5.7'de verilen performans sonuçları incelendiğinde aşağıdaki vargılara ulaşılmıştır:

- Markov modellemeli steganaliz yöntemi, EÖB ve matris gömme yöntemlerine karşı son derece etkili olmuş, bariz bir şekilde kaotik özniteliklere dayalı steganalizden ve P.563'lü steganalizden daha yüksek başarım oranlarına erişmiştir.

- Markov modelinin ve kaotik özniteliklerin beraber kullanıldığı melez steganaliz yöntemi, EÖB ve matris gömme yöntemlerine karşı MM(1) steganalizi kadar etkin olmuştur. Başka bir ifadeyle kaotik özniteliklerin kullanımı, MM(1) steganalizinin başarım oranına katkıda bulunmamıştır.

- Markov steganalizine karşı özel olarak geliştirilen, Markov modellemeli sendrom kodlama gizli veri gömme yöntemi amacına ulaşmış, söz konusu steganalize karşı ideale yakın fark edilmezlik sağlamıştır.

- Kaotik özniteliklerle yapılan steganaliz yöntemlerine göre de en fark edilmez veri gizleme yöntemi Markov modellemeli sendrom kodlaması olmuştur.

- Markov modellemeli sendrom kodlaması sadece P.563'te diğer yöntemlere kıyasla başarısız olmuştur. Gizli bit başına en çok değişiklik bu yöntemde uygulandığından çıkan sonuç şaşırtıcı değildir. P.563 karşısında en fazla tespit edilebilirlik, duyusal anlamda en fazla iz bırakmak anlamına gelmektedir.

- P.563'ün en etkin olduğu noktadaki başarım oranı melez steganalizin (CwMMS'in) en etkin olduğu noktadaki başarım oranının altındadır. Diğer bir deyişle bu sonuç, Markov modellemeli sendrom kodlamasının P.563 karşısındaki zayıflığını değil, P.563'ün genel bir steganaliz yöntemi olarak yetersizliğini ortaya koymaktadır.

- Steganaliz açısından Markov modellemeli sendrom kodlaması duyusal anlamda ek bozulmaya neden olsa da kesinlikle matris gömmeden daha fazla gizlilik temin etmektedir.

5.9 MM(1) Steganalizinin ve MMSK'nın Genel Bir Değerlendirmesi

Bu çalışmada ilk olarak, tek boyutlu bit dizisi olan nitelendirilebilecek kodlanmış konuşma sinyalinin ergodik Markov zincirleri ile modellenebileceği belirtilmiş, bu

yaklaşıma dayanan bir steganaliz yöntemi geliştirilmiştir. Daha sonra bu steganalize panzehir olabilecek, gizli veriyi bit dizilerinin MM(1) modelinde en az sapmalara yol açacak şekilde yerleştirebilen özgün bir gizli veri gömme yöntemi önerilmiştir. Önerilen Markov zincir modellemeli sendrom kodlamalı gizli veri gömme yönteminin temel felsefesi en az sayıda değişikliğin istatistiksel farklılaşma açısından her zaman en iyi tercih olmadığı öngörüsüne dayanmaktadır. Söz konusu yöntem steganografi dünyasında sıkça başvurulan matris gömmenin bir türevidir. Matris gömmede olduğu gibi gizli veri sendrom kodlaması uygulanarak konak sinyale (bit dizisine) gömülmekte, gizli verinin içine saklanacağı sendromu taşıyacak hata vektörü, MM(1) modelinden elde edilen olasılıklara göre rastgele seçilmektedir.

Yöntemin kuramsal tasarımının ardından uygulama safhasına geçilmiş, önerilen yöntem ile bilindik EÖB ve matris gömme gizli veri gömme yöntemleri, iletişimde yaygın olarak kullanılan GSM 6.10 kodlayıcısı üzerinde gerçeklenmiştir. Testler kapsamında gizli veri gömme yöntemlerinin taşıyıcı bit dizilerinin entropisi üzerindeki etkileri araştırılmış, veri gömme verimlilikleri ölçülmüştür. Performans ölçütlerine ek olarak yöntemler steganografik açıdan güvenilirlikleri incelenmiş, MM(1), KÖ ve P.563 steganaliz yöntemlerince sınanmışlardır.

İlk olarak bu çalışmada geliştirilen MM(1) steganalizinin yetenekleri büyüteç altına alınmış, test sonuçlarına göre EÖB ve matris gömmeye karşı MM(1) steganalizinin KÖ ve P.563 steganalizlerinden daha etkin olduğu deneysel olarak kanıtlanmıştır. Söz konusu steganalize karşı koyması amacıyla geliştirilen Markov zincir modelleme ile sendrom kodlamalı gizli veri gömme yöntemi beklentileri boşa çıkarmamış, MM(1) steganalizine karşı ideale yakın fark edilmezliğe ulaşmıştır. KÖ steganalizi sonuçlarında da rakiplerini alt etmeyi başarmış, yalnızca P.563 steganalizinde EÖB ve matris gömmenin gerisinde kalmıştır. Lakin P.563 steganalizinin KÖ ve MM(1) steganalizlerine nazaran nispeten tesirsiz bir steganaliz yöntemi olmasından dolayı, bu olumsuzluk sadece duyusal açıdan fark edilebilirlik sınırlarının altında daha fazla bozulma anlamına gelmektedir.

Özetle bu çalışmada, taşıyıcı sinyal istatistikî özelliklerini MM(1)'e göre koruyan gizli veri gömme yaklaşımının, sadece en az biti değiştirmeyi hedefleyen gizli veri gömme yaklaşımından üstün olduğu ortaya konulmuştur. Elbette daha uzun kod kelimeleri kullanılması durumunda, matris gömmenin fark edilmezliği gömme verimliliğinin artışına bağlı olarak artmaktadır. Markov zincir modelleme ile sendrom kodlamalı gizli veri gömme yöntemi şu çalışmada sunulduğu saf haliyle, uzun kod kelimelerinde matris gömmeyle rekabet edemeyebilir, ancak şayet gizli veri gömerken gerçekleştirdiği rastgele seçimler az değişiklik gerektiren olasılıklardan olacak şekilde sınırlandırılabilirse, matris gömmenin avantajlarıyla, istatistiksel modellemenin getirdiği avantajlar bir araya getirilebilir. Diğer taraftan, kodlanmış konuşma sinyali uygulamalarında uzun kod kelimelerinin kullanımının neden olacağı çeşitli pratik problemler de kulak ardı edilmemelidir.

6. SONUÇLAR

Bugüne kadar konuşma sinyali için geliştirilen veri gömme yöntemlerinin düşük hızlı kodlayıcılar için başarısız olmalarından dolayı, veri gömme işlemi kodlayıcının bünyesine alınmıştır. Konuşma sinyalinin telif haklarının korunması görece lüzumsuz olduğundan, veri gömme yönteminin gürbüzlükle ilgili gereksinimleri tasarımı kolaylaştıracak şekilde esnetilmiştir. Saklı verilerin biçim değişikliklerine karşı muvaffak olması gereksiniminden vazgeçilerek gizli veri gömmenin kodlayıcının parçası haline getirildiği özgün MELP-MSVQ-QIM yöntemi oluşturulmuştur. Geleneksel QIM yönteminin MSVQ'nun önceden seçilmiş indekslerinin önceden seçilmiş bitleri üzerinde uygulanmasına dayanan MELP-MSVQ-QIM yönteminin, farklı indekslere veri gömen pek çok varyasyonu türetilmiştir.

NTIMIT (Jankowski vd. 1990) veri tabanı kullanılarak, MELP-MSVQ-QIM varyasyonları kalite metrikleri, entropi ve steganaliz açılarından test edilmişlerdir. Kalite metrik sonuçlarına göre QIM-2S-HB, QIM-3S-HB, QIM-4S-HB ve QIM-HS-HB varyasyonlarının 10^{-3}'lük haberleşme ortam gürültüsü kadar bozulmaya neden oldukları gözlemlenmiştir. Entropi analizleri ve steganaliz sonuçlarına göreyse tüm çeşitli düzeylerde varyasyonların tespit edilebilir olduğu anlaşılmış, QIM-4S-HB'in (dördüncü indekse bir bit saklayan varyasyon) en az, QIM-4S-H4B'nin (dördüncü indekse dört bit saklayan varyasyon) en fazla fark edilebilir varyasyonlar olduğu görülmüştür. Keza benzer şekilde Kocal vd. (2008)'de incelenen tüm gizli veri gömme yöntemleri de fark edilebilirlik çıkmışlardır. Sonuç olarak, QIM-4S-HB'nin steganografik, QIM-HS-HB'nin ise veri bütünlüğü uygulamalarına en uygun varyasyonlar olduğu belirlenmiştir. Steganografik uygulamalarda kullanılacak MELP-MSVQ-QIM varyasyonlarının matris kodlama gibi ilave yöntemlerle güçlendirilmelerinin faydalı olacağı anlaşılmıştır.

Bu tez kapsamında özgün olarak kaotik öznitelikler steganaliz yöntemindeki hatalı en yakın komşuluk oranına dayanan sınıflandırma yaklaşımı, bambaşka bir veri işleme uygulamasına uyarlanmış ve böylece konuşma kodlayıcı tanıma türetilmiştir. Literatüre ilk kez bu tez kapsamında kazandırılan konuşma kodlayıcı tanıma internetten, havadan, …vb iletişim ortamlarından aktarılırken yakalanmış veya kayıt ortamlarında saklanmış

konuşma sinyaline ait bit dizilerinin çeşitli analiz süreçlerine sokularak, üretimlerinde kullanılan kodlayıcıların tiplerinin otomatik olarak tespit edilmesi olarak tanımlanmaktadır. Çalışma esası her kodlayıcının kendisine özgü çerçeve yapısı olduğu öngörüsüne dayanmaktadır; başka bir ifadeyle kodlanmış konuşma sinyalini oluşturan bit alanları, bit alanları sayısı, bit alanları arası ilişkiler ve bit alanlarının dağılımları her kodlayıcı türü için farklıdır ve özeldir. Bir kodlayıcı çıktısının hatalı en yakın komşuluk oranları (FNF) ise çerçeve yapı özelliklerince belirlenmektedir, dolayısıyla bir bit dizisi için hesaplanan FNF değerlerinde kendisini kodlayan kodlayıcı türünün izleri bulunur. Aynı kodlayıcının farklı çıktılarında görülen bit alan dağılımlarındaki farklılıklara rağmen, her kodlayıcının FNF sonuçları diğer kodlayıcılardan ayrışma göstermektedir.

Yinelenen çerçevelerden oluşan kodlanmış konuşma sinyali ergodik Markov zincirleri ile modellenmiştir. Gizli veri eklemenin ergodik Markov zincirlerindeki geçiş olasılıklarını değiştireceği varsayılarak, Markov modelindeki geçiş olasılıklarının gerçekleşme ihtimaline göre işleyen bir steganaliz yöntemi, MM(1) steganaliz yöntemi hazırlanmıştır. Hazırlanan yöntem, kaotik öznitelik steganaliziyle beraber GSM 6.10 EÖB gizli veri gömme yöntemi üzerinde denenmiştir. Test sonuçları uyarınca MM(1) steganalizi kaotik öznitelik steganalizinden daha yüksek tespit oranına ulaşmıştır. Bir sonraki aşamada GSM 6.10 EÖB yöntemi, steganalizde kullanılan MM(1) modeli ve sendrom kodlaması bir araya getirilerek özgün istatistiksel olarak iyileştirilmiş matris kodlama (Diğer adıyla Markov zincirleri ile istatistiksel modelleme yapılarak sendrom kodlama) yöntemi geliştirilmiştir. İyileştirilmiş matris kodlama aslında tam anlamıyla bir veri gömme yöntemi değildir, daha çok taşıyıcı sinyalin gömülecek verileri bozmasını indirgeyen bir veri manipülasyon tekniğidir; MM(1) modellemesinin yapılabildiği herhangi bir kodlanmış dizi üzerinde başka bir veri gömme yöntemiyle beraber kullanılabilir.

Gerçekleştirilen testler sonucunda, istatistiksel olarak iyileştirilmiş matris kodlamanın hem MM(1) steganalizine hem de kaotik öznitelikler steganalizine karşı başarılı olduğu görülmüştür. Diğer taraftan uygulanan bu yeni tür matris gömmenin algısal anlamda daha fazla bozulmaya sebebiyet verdiği anlaşılmış, bunun üzerine oluşan fazladan bozulmanın bir steganaliz yöntemi tarafından istifade edilip edilemeyeceği konusu

araştırılmıştır. P.563'den algısal bozulmaya dayanan bir steganaliz yöntemi türetilmiş, ancak bu yeni P.563 steganalizinin kaotik öznitelik ve MM(1) steganalizlerine nazaran nispeten tesirsiz kaldığı ortaya konulmuştur.

Sonuç olarak bu tez çalışmasıyla birlikte steganografi ve damgalama açısından literatüre üç yeni özgün yöntem katılmıştır. İlk olarak MELP kodlamasında veri saklamak için özgün MELP-MSVQ-QIM yöntemi ve varyasyonları geliştirilmiş, ardından MELP-MSVQ-QIM varyasyonlarının steganalizinde kullanılan kaotik öznitelikler steganalizinden konuşma kodlayıcı tanıma tasarlanmıştır. Son olarak, Markov zinciri olarak modellenebilen diziler üzerinde çalışan veri gömme yöntemlerinin performansını arttırmaya yönelik olarak özgün istatistiksel olarak iyileştirilmiş matris kodlama yöntemi önerilmiştir.

KAYNAKLAR

Anonymous. 1990. G.726: 40, 32, 24, 16 kbit/s adaptive differential pulse code modulation (ADPCM). ITU-T.

Anonymous. 1996. G.723.1: Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s. ITU-T.

Anonymous. 1999. EN 300 726: Digital cellular telecommunications system (Phase 2+) (GSM) enhanced full rate (EFR) speech transcoding GSM 06.60 version 8.0.1. ETSI.

Anonymous. 1999. EN 300 961: Digital cellular telecommunications system (Phase 2+) (GSM) full rate speech transcoding GSM 06.10 version 8.1.1. ETSI.

Anonymous. 1999. MIL-STD-3005: MELP, Analog-to-Digital Conversion of Voice by 2400 bit/second Mixed-Excitation Linear Prediction. US DoD.

Anonymous. 2001. Recommendation P.862: Perceptual evalution of speech quality. An objective method for end-to-end speech quality assessment of narrow band telephone networks and speech codecs. ITU-T.

Anonymous. 2004. Recommendation P.563: Single-ended method for objective speech quality assessment in narrow-band telephony applications. ITU-T.

Anonymous. 2006. G.729.1: G.729 based embedded variable bit-rate coder: An 8-32 kbit/s scalable wideband coder bitstream interoperable with G.729. ITU-T.

Anonymous. 2010. Mp3stego, available: http://www.petitcolas.net/fabien/ steganography/mp3stego

Avcıbas, I., Menon, N. and Sankur, B. 2003. Steganalysis using image quality metrics. IEEE Transactions on Image Processing, Vol 12, Issue 2, pp. 221-229.

Banbrook, M. 1990. Nonlinear analysis of speech from a synthesis perspective. PhD Thesis. The University of Edinburgh.

Barnsley, M.F. and Sloan, A.D. 1989, A better way to compress images. BYTE, January 1989, pp. 215-223.

Bender, W., Gruhl, D., Morimoto, N. and Lu, A. 1996. Techniques for data hiding. IBM Systems Journal, Vol 35, Issue 3.4, pp. 313-336.

Bernhard, H.P. and Kubin, G. 1991, Speech production and chaos. Proceedings XIIth Int. Congress Phonetic Sciences.

Bierbrauer, J. 2004. Introduction to coding theory. Chapman & Hall/CRC, 390, USA.

Boshoff, H.F.V. and Grotepass, M. 1991. The fractal dimension of fricative speech sounds. IEEE Proceedings Communications and Signal Processing COMSIG 1991, pp. 12-16.

Böhme, R. and Westfeld, A. 2004. Statistical characterisation of MP3 encoders for steganalysis. Proceedings of the Multimedia and Security Workshop 2004, pp. 25-34.

Chang, B.P. and Yu, H. 2002. Dither-like data hiding in multistage vector quantization of MELP and G.729 speech coding. Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, Vol 2, pp. 1199-1203.

Cheetham, B. 1987. Adaptive LSP filter, IEE Electronics Letters, Vol 23, Issue 2, pp. 89-90.

Chen, B. and Wornell, G.W. 2001. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. IEEE Transactions on Information Theory, Vol 47, Issue 4, pp. 1423-1443.

Cheng, Q. and Sorensen, J. 2001. Spread spectrum signaling for speech watermarking. IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol 3, pp. 1337-1340.

Chou, P.A. and Lookabaugh, T. 1990. Conditional entropy-constrained vector quantization of linear predictive coefficients. IEEE International Conference on Acoustics, Speech and Signal Processing, Vol 1, pp. 197-200.

Chu, W.C. 2003. Speech coding algorithms. John Wiley and Sons, 585, New Jersey.

Cook, P.R. 1990. Identification of control parameters in an articulatory vocal tract model with applications to the synthesis of singing. PhD Thesis. Department of Music, Stanford University.

Cox, I.J., Miller, M.L., Bloom, J.A., Fridrich, J. and Kalker, T. 2008. Digital watermarking and steganography, Elsevier - Morgan Kaufmann, 593, USA.

Deller, J.R. and Huang, Y.F. 2002. Set-membership identification and filtering for signal processing applications. Circuits, Systems and Signal Processing, Vol 21, Number 1, pp. 69-82.

Falaschi, A., Giustiniani, M. and Pierucci, P. 1990. A finite states Markov quantizer for speech coding. IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol 1, pp. 205-208.

Furon, T., Moreau, N. and Duhamel, P. 2000. Audio public key watermarking technique. IEEE International Conference on Acoustics, Speech, and Signal Processing,Vol 6, pp. 1959-1962.

Gray, A.H. 1980. Passive cascaded lattice digital filters. IEEE Transactions on Circuits and Systems, Vol 27, Issue 5, pp. 339-344.

Gurijala, A.R., Deller, J.R., Seasle, M.S. and Hansen, J.H.L. 2002. Speech watermarking through parametric modeling. 7th International Conference on Spoken Language Processing, pp. 621-624.

Gurijala, A.R. and Deler, J.R. 2003. A qualified fidelity criterion for parameter-embedded watermarking of audio archive. International Conference on Digital Libraries, pp. 237-239.

Hegger, R., Kantz, H. and Schreiber, T. 2007. TISEAN: Nonlinear time series analysis, http://www.mpipks-dresden.mpg.de/~tisean/

Itakura, F. 1975. Line spectrum representation of linear predictor coefficients of speech signals. The Journal of The Acoustical Society of America, Volume 57, pp. 35.

Jankowski, C., Kalyanswamy, A., Basson, S. and Spitz, J. 1990. NTIMIT: A phonetically balanced, continuous speech, telephone bandwidth speech database. IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol 1, pp. 109-112.

Jayant, N.S. and Noll, P. 1984. Digital coding of waveforms: Principles and applications to speech and video. Prentice-Hall, 688, New Jersey.

Johnson, D. 2010. Speech signal analysis. http://cnx.org/content/col10476/1.3/pdf

Kabal, P. and Ramachandran, R.P. 1986. The computation of line spectral frequencies using Chebyshev polynomials. IEEE Transactions on Acoustics, Speech and Signal Processing, Vol 34, Issue 6, pp. 1419-1425.

Kaplan, J. and Yorke, J. 1979. Chaotic behavior of multidimensional difference equations. Functional Differential Equations and Approximation of Fixed Points, Lecture Notes in Mathematics, Vol 730, pp. 204-227.

Kennel, M.B. and Abarbanel, H.D.I. 2002. False neighbors and false strands: a reliable minimum embedding dimension algorithm. Physical Review E, Vol 66, Issue 2, 026209.

Kirovski, D. and Malvar, H.S. 2003. Spread-spectrum watermarking of audio signals. IEEE Transactions on Signal Processing, Vol 51, Issue 4, pp. 1020-1033.

Kocal, O.H., Yuruklu, E. and Avcıbas, I. 2008. Chaotic-type features for speech steganalysis. IEEE Transactions on Information Forensics and Security, Vol 3, Isuue 4, pp. 651-661.

Kondoz, A.M. 2004. Digital speech: Coding for Low Bit Rate Communication Systems. John Wiley & Sons Ltd, 459, England.

Lee, S. and Jung, S. 2001. A survey of watermarking techniques applied to multimedia. IEEE International Symposium On Industrial Electronics, Vol 1, pp. 272-277.

Li, T.Y. and Yorke, J. 1975. Period three implies chaos. American Mathematical Monthly, Vol 82, pp. 985-992.

Liu, C.H. and Chen, O.T.C. 2004. A fragile speech watermarking scheme with recovering speech contents. IEEE International Midwest Symposium on Circuits and Systems, Vol 2, pp. 165-168.

Malik, H., Subbalakshmi, K.P. and Chandramouli, R. 2008. Nonparametric steganalysis of QIM data hiding using approximate entropy. IS&T SPIE: Security, Forensics, Steganography, and Watermarking of Multimedia Contents. Vol 6819, pp. 681914-12.

Malvar, H.S. 1999. A modulated complex lapped transform and its applications to audio processing. IEEE International Conference on Acoustics, Speech and Signal Processing, Vol 3, 1421-1424.

Malvar, H.S. and Florencio, D.A.F. 2003. Improved spread spectrum: a new modulation technique for robust watermarking. IEEE Transactions on Signal Processing, Vol 51, Issue 4, pp. 898-905.

Mandelbrot, B.B. 1977. Fractals: form, chance and dimension. W.H. Freeman and Company, 365, San Francisco.

Maragos, P. 1991. Fractal aspects of speech signals: dimension and interpolation. International Conference on Acoustics, Speech and Signal Processing, Vol 1, pp. 417-420.

Marcato, L. and Mumolo, E. 1993. Coding of speech signal by fractal techniques. EUROSPEECH '93, pp. 745-748.

Mazurczyk, W. and Lubacz, J. 2009. LACK — a VoIP steganographic method. Telecommunication Systems, Vol 43, Numbers 2-3, pp. 153-163.

Meyn, S.P. and Tweedie, R.L. 1993. Markov chains and stochastic stability. Springer-Verlag, 548, London.

McDowell, P.S. and Datta, S. 1994. The fractal characterisation of isolated human speech. Proceedings of the Institute of Acoustics, Vol 16, No 5, pp. 247-253.

Narayanan, S.S. and Alwan, A.A. 1995. A nonlinear dynamical system analysis of fricative consonants. The Journal of the Acoustical Society of America, Vol 97, pp. 2511–2524.

Özer, H., Sankur, B., Memon, N. and Avcıbaş, İ. 2003. Steganalysis of audio based on audio quality metrics. Security and Watermarking of Multimedia Contents Proceedings of the SPIE, Vol 5020, pp. 55-66.

Pickover, C.A. and Khorasani, A. 1986. Fractal characterization of speech waveform graphs. Comput and Graphics, Vol. 10, No 1, pp. 51-61.

Quatieri, T.F. 2001. Discrete-time speech signal processing: principles and practice. Prentice Hall, 816, USA.

Rabiner, L. and Schafer, R. 1978. Digital processing of speech signals. Prentice Hall, 512, USA.

Rahikka, D.J., Fuja, T.E. and Fazel, T. 1999. Optimized error correction of MELP speech parameters via maximum a posteriori (MAP) techniques. IEEE Workshop on Speech Coding Proceedings, pp. 78-80.

Rencher, A.C. 1995. Methods of multivariate analysis. Wiley-Interscience, 627, New York.

Rothenberg, M. 1981. Acoustic interaction between the glottal source and the vocal tract. Vocal Fold Physiology, K, N. Stevens and M. Hirano, Eds., University of Tokyo Press, pp. 305-328.

Ruelle, D. 1980. Strange attractors. The Mathematical Intelligencer, Vol 2, pp. 126-137.

Sano, M. and Sawada, Y. 1985. Measurement of the Lyapunov spectrum from a chaotic time series. Phys. Rev. Lett. Vol 55, pp. 1082-1085.

Senvirathne, T.R., Bohez, E.L.J. and VanWinden, J.A. 1992. Amplitude scale method: new and efficient approach to measure fractal dimension of speech waveforms. IEEE Electronics Letters, Vol 28, Issue 4, pp. 420-422.

Shahbazi, A., Soltanmohammadi, E., Rezaei, A.H., Sayadiyan, A. and Mosayyebpour, S. 2010. Content dependent data hiding on GSM full rate encoded speech. International Conference on Signal Acquisition and Processing, pp. 68-72.

Shi, Y.Q., Xuan, G., Yang, C., Gao, J., Zhang, Z., Chai, P., Zou, D., Chen, C. and Chen, W. 2005. Effective steganalysis based on statistical moments of wavelet characteristic function. International Conference on Information Technology: Coding and Computing. Vol 1, pp. 768-773.

Siwei, L. and Farid, H. 2002. Detecting hidden messages using higher-order statistics and support vector machines. Information Hiding, Lecture Notes in Computer Science, Vol 2578, pp. 340-354.

Sprott, J.C. 2003. Chaos and time-series analysis. Oxford University Press, 528, UK.

Stallings, W. 2003. Network security essentials: application and standards. Prentice Hall, 410, New Jersey.

Swanson, M.D., Tewfik, A.H. and Boney, L. 1998. Robust audio watermarking using perceptural masking. Signal Processing, Vol 66, Issue 3, pp. 337-355.

Takens, F. 1981. Dynamical systems and turbulence. Lecture Notes in Mathematics, Springer, Berlin, Vol 898, pp. 366-381.

Teager, H.M. and Teager, S.M. 1990. Evidence for nonlinear sound production mechanisms in the vocal tract. Proc NATO ASI on Speech Production and Speech Modelling, pp. 241-261.

Tishby, N. 1990. A dynamical systems approach to speech processing. IEEE International Conference on Acoustics, Speech and Signal Processing, Vol 1, pp. 365-368.

Titze, I.R. 1988. The physics of small-amplitude oscillation of the vocal folds. The Journal of The Acoustical Society of America, Vol 83, pp. 1536-1552.

Titze, I.R. 1992. Phonation threshold pressure: A missing link in glottal aerodynamics. The Journal of The Acoustical Society of America, Vol 91, pp. 2926-2935.

Tian, H., Zhou, K., Jiangi H. and Feng, D. 2009. Digital logic based encoding strategies for steganography on voice-over-IP. Proceedings of The Seventeen ACM International Conference on Multimedia Table of Contents, Beijing, China , pp. 777-780.

Wang, R. and Chai, P. 2003. A new adaptive audio watermarking algorithm for copyrigth protection. International Conference on Natural Language Processing and Knowledge Engineering, pp. 281-286.

Westfeld, A. 2001. F5 – a steganographic algorithm: high capacity despite better steganalysis. Proceedings of the 4th International Workshop on Information Hiding, Springer-Verlag, pp. 289-302.

Westfeld, A. and Pfitzmann, A. 2000. Attack on steganographic systems. Lectures Notes in Computer Science. Vol 1768, pp. 61-75.

Wu, C.P. and Kuo, C.C.J. 2002. Fragile speech watermarking based on exponential scale quantization for tamper detection. IEEE International Conference on Acoustics, Speech and Signal Processing, Vol 4, pp. 3305-3308.

Yuan, S. and Huss, S.H. 2004. Audio watermarking algorithm for real-time speech integrity and authentication. Proceedings of The 2004 Multimedia and Security Workshop on Multimedia and Security, pp. 220-226.

Yuan, Z. 1999. Prediction of protein subcellular locations using Markov chain models. FEBS Letters, Vol 451, pp. 23-26.

EKLER

EK 1 Doğrusal Öngörü Katsayıları

EK 2 Kaos Teorisi

EK 3 MELP Konuşma Kodlayıcısı

EK 4 G.726 Konuşma Kodlayıcısı

EK 5 GSM 6.10 FR Konuşma kodlayıcısı

EK 6 G.729 Konuşma Kodlayıcısı

EK 7 GSM 6.60 EFR Konuşma Kodlayıcısı

EK 1 Doğrusal Öngörü Katsayıları

Tezin bu kısmı doğrusal öngörü katsayısı hakkında faydalı ek bilgiler vermek için hazırlanmış olup, kuramsal temeller bölümünde sunulması gerekli görülmeyen teknik detayları ve matematik formülleri içermektedir. Ekte ilk olarak Kelly ve Lochbaum'un (1962) kayıpsız tüpler modeli ile doğrusal öngörü katsayıları arasındaki ilişki ortaya konmakta, doğrusal öngörü katsayılarının aslında kayıpsız tüp modelinin basit bir sonucu olduğu belirtilmektedir. Daha sonra sayısallaştırılmış sinyale ait LPC değerlerinin nasıl hesaplanabileceği gösterilmekte, bu iş için geliştirilmiş teknikler özetlenmektedir. Son olarak LPC değerlerinin nicemlenmesi konusu ele alınmakta, nicemleme sonucu oluşabilecek hataların azaltılabilmesi için kullanılan alternatif LPC dönüşümleri tanıtılmaktadır.

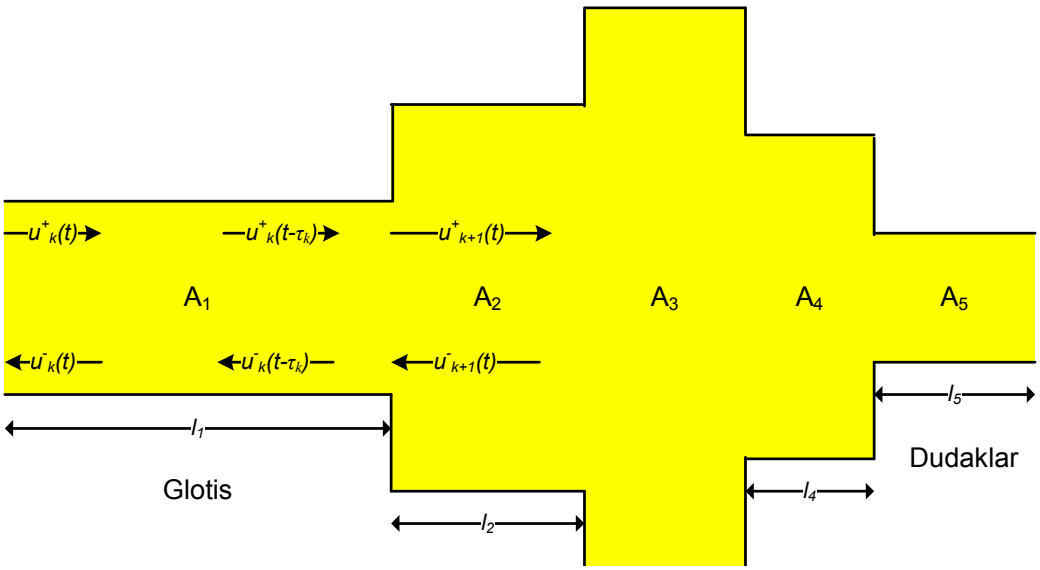
Kayıpsız Tüp Modeli

Kelly ve Lochbaum (1962) ses üretim yolunu birbirlerine peş peşe eklenmiş kayıpsız tüpler modelinden olarak modellemişlerdir (Şekil 1). Kayıpsız tüplerin genişlikleri ve uzunlukları sinir sistemi tarafından ayarlanmakta, bu sayede istenilen niteliklerde süzgeçler yaratılabilmektedir.

Her bir tüp içinde biri dışarı diğeri yansıma sonucu içeri doğru hareket eden iki akış tanımlanmıştır. (1.1)'de gösterildiği üzere *t* anında *k*. tüp içerisindeki *x* pozisyonundaki net akış içeri ve dışarı giden akışların vektörsel toplamları kadardır. Aynı noktalardaki basınç ise her iki akışın skalar toplamlarıyla orantılıdır.

$$u_k(x,t) = u_k^+\left(t - \frac{x}{c}\right) - u_k^-\left(t + \frac{x}{c}\right)$$

$$p_k(x,t) = \frac{\rho c}{A_k}\left[u_k^+\left(t - \frac{x}{c}\right) + u_k^-\left(t + \frac{x}{c}\right)\right] \qquad 0 \le x \le l_k \qquad (1.1)$$



Şekil 1 Ses üretim yolunun kayıpsız tüpler olarak modellenmesi

Tüplerin birleşim noktalarındaki basınç ve akış değerleri iki tüp için ayrık değerler alamayacağına göre (1.2) yazılabilir:

$$\begin{aligned} p_k(l_k,t) &= p_{k+1}(0,t) \\ u_k(l_k,t) &= u_{k+1}(0,t) \end{aligned} \tag{1.2}$$

Birleşim noktalarındaki ileri ve geri akış miktarları (1.1) ve (1.2)'nin bir araya getirilmesiyle hesaplanabilir (1.3):

$$\begin{aligned} &\frac{A_{k+1}}{A_k}\left[u_k^+(t-\tau_k)+u_k^-(t+\tau_k)\right] = u_{k+1}^+(t)+u_{k+1}^-(t) \\ &u_k^+(t-\tau_k)-u_k^-(t+\tau_k) = u_{k+1}^+(t)-u_{k+1}^-(t) \end{aligned} \qquad \tau_k = l_k/c \tag{1.3}$$

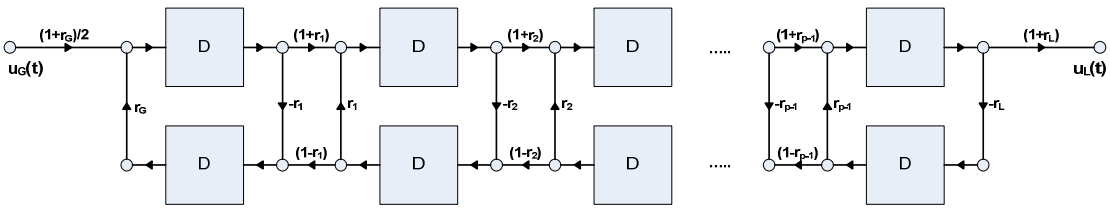
(1.4) bir önceki denklemin sadeleştirilmiş bir halidir:

$$\begin{aligned} u_{k+1}^+(t) &= \left[\frac{2A_{k+1}}{A_{k+1}+A_k}\right]u_k^+(t-\tau_k)+\left[\frac{A_{k+1}-A_k}{A_{k+1}+A_k}\right]u_{k+1}^-(t) \\ u_k^-(t+\tau_k) &= -\left[\frac{A_{k+1}-A_k}{A_{k+1}+A_k}\right]u_k^+(t-\tau_k)+\left[\frac{2A_{k+1}}{A_{k+1}+A_k}\right]u_{k+1}^-(t) \end{aligned} \tag{1.4}$$

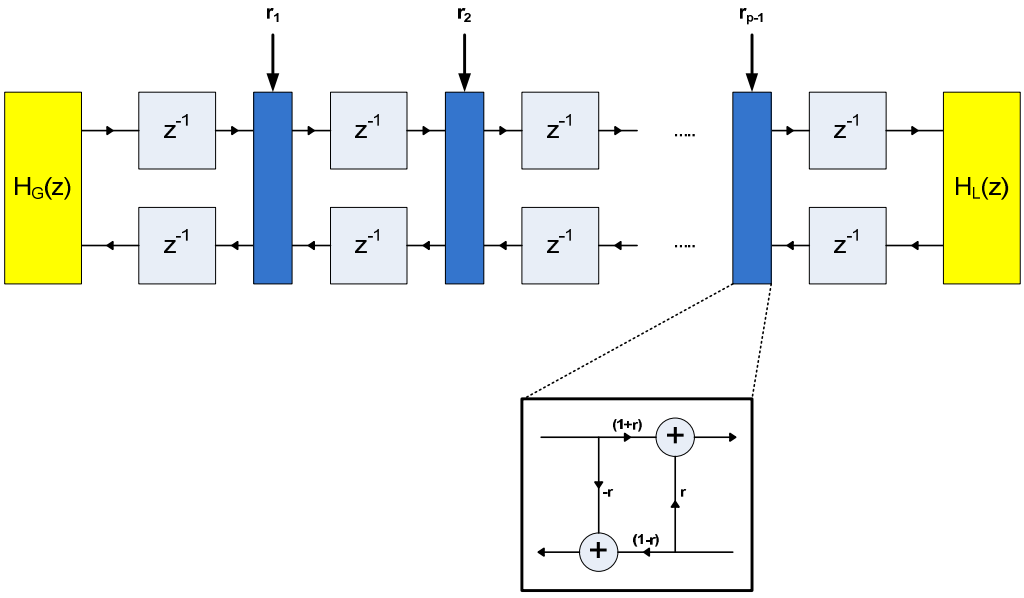
(1.5)'te ise yansıma katsayısı tanımlanmasıyla matematiksel ifadeler daha da basitleştirilmiştir:

$$u_{k+1}^{+}(t) = (1 + r_k)u_k^{+}(t - \tau_k) + r_k u_{k+1}^{-}(t)$$
$$u_k^{-}(t + \tau_k) = -r_k u_k^{+}(t - \tau_k) + (1 - r_k)u_{k+1}^{-}(t) \qquad r_k = \left[\frac{A_{k+1} - A_k}{A_{k+1} + A_k}\right] \qquad (1.5)$$

Denklemlerde verilen r_k değerleri yansıma katsayısı olarak isimlendirilmektedir. Şekil 2'de yansıma katsayıları kullanılarak p adet tüpün yan yana eklenmesiyle oluşan bir yapıya ait bir sinyal akış çizgesi sunulmuş, Şekil 3'te bu akış çizgesinden ses üretim yolunu betimleyen bir kesikli zaman sistemi elde edilmiştir:



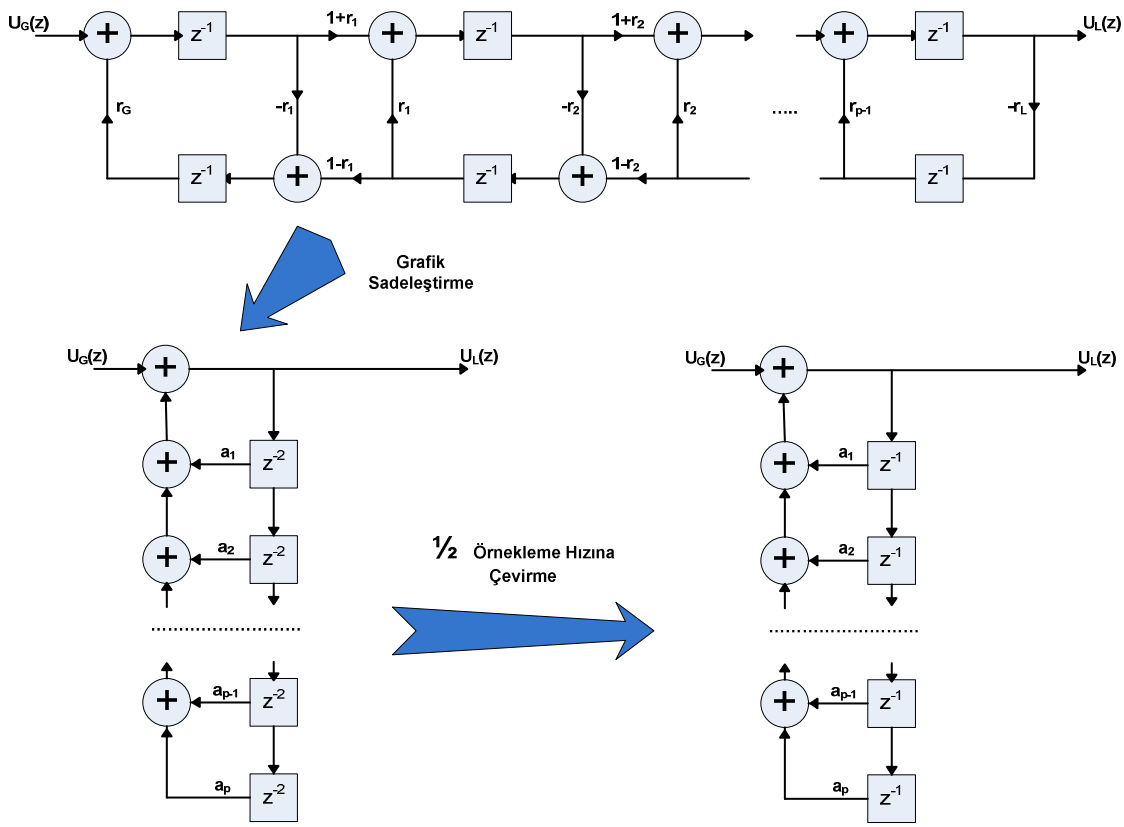
Şekil 2 Ses üretim yoluna ait sinyal akış çizgesi



Şekil 3 Ses üretim yolunun eşdeğeri olan kesikli zaman sistemi

Gray (1980), Şekil 2'deki kesikli zaman sisteminin Şekil 3'teki sayısal bir süzgece eşdeğer olduğunu göstermiştir; transfer fonksiyonu (1.6)'ya eşit olan bu sayısal süzgeç, LPC analiz süzgeci olarak da bilinmektedir.

$$\frac{U_L(z)}{U_G(z)} = \frac{1}{A(z)} = \frac{1}{1 - \sum_{i=1}^{p} a_i z^{-i}} \tag{1.6}$$

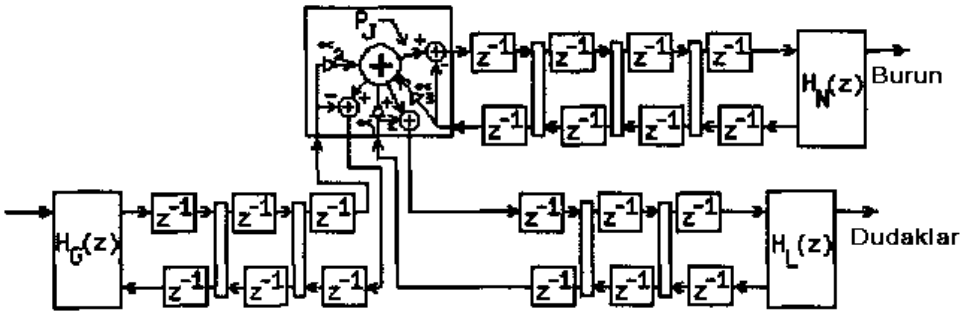


Şekil 4 Ses üretim yolunun bir süzgeç olarak modellenmesi

Kayıpsız tüp sayısı 3'e eşit (aynı zamanda p = 3) olan örnek bir sistem için $A(z)$ polinomunun katsayıları (1.7)'de sunulmuştur:

$$\begin{aligned} a_1 &= r_1 r_2 + r_2 r_L - r_1 r_G \\ a_2 &= r_1 r_L - r_G b r_1 r_2 - r_2 r_G \\ a_3 &= r_G r_L {r_1}^2 {r_2}^2 \end{aligned} \tag{1.7}$$

Her ne kadar konuşma sinyalinin sadece dudaklardan yayıldığına dair bir modelleme yapılmışsa da, gerçekte insan yapımı ses dudaklar haricinde burundan da yayılım göstermektedir. Şekil 5'te Cook'dan (1990) alınmış burundan yayılımlı ses üretim modeli özellikle müzik sinyalinin modellenmesinde faydalı olabilmektedir.



Şekil 5 Burun yayılımlı ses üretim yolu modeli

Kondoz'a (2004) göre LPC parametrelerinin hesaplanması için genel olarak üç farklı yaklaşım önerilebilir: Öz-ilinti, ortak değişinti (*covariance*) ve kafes (*lattice*) LPC hesaplama yöntemleri.

LPC değerlerinin öz-ilinti ile hesaplanması

LPC hesaplama işlemi *N* adet örneklik çerçeveler üzerinde gerçekleştirilir.

$$\emptyset_n(i,j) = \sum_{m=0}^{N-1-(i-j)} s_n(m)s_n(m+i-j) \qquad 1 \le i \le p,\ \ 0 \le j \le p \qquad (1.8)$$

LPC parametreleri kayıpsız tüp sisteminin girdisi ve çıktısı arasındaki bağlantıyı verir, aynı parametreler çıktının öz-ilinti değerleri arasında da oransal ilişkilerin oluşmasına yol açar.

$$\emptyset_n(i,j) = R_n(|i-j|)$$
$$R_n(j) = \sum_{m=0}^{N-1-j} s_n(m)s_n(m+j) \tag{1.9}$$

$$\sum_{j=1}^{p} \alpha_j \, R_n(|i-j|) = R_n(i) \tag{1.10}$$

(1.10)'a göre öz-ilintiler arasındaki ilişkilerden *p* değişkenli *p* adet eşitlik tanımlanabilir. (1.11)'de matris haline getirilmiş bu *p* adet eşitliğin çözülmesiyle LPC parametreleri hesaplanabilir.

$$\begin{bmatrix} R_n(0) & R_n(1) & \cdots & R_n(p-1) \\ R_n(1) & R_n(0) & \cdots & R_n(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ R_n(p-1) & \cdot & \cdots & R_n(0) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_n(1) \\ R_n(2) \\ \vdots \\ R_n(p) \end{bmatrix} \tag{1.11}$$

(1.11)'de eşitliğin sol tarafındaki matris köşegeni boyunca simetrik olan bir Toeplitz matrisidir. α katsayıları basit bir mantıkla Toeplitz matrisinin tersinin alınmasıyla çözülebilir, ancak matrisin tersinin bulunması sırasında oluşan hesaplama hataları sonuçların doğruluğunu azaltır. Diğer yandan sırf Toeplitz matrislerinin çözümü için Levinson-Durbin gibi yinelemeli algoritmalar ve türevleri geliştirilmiştir ve bu algoritmalardan elde edilen sonuçlar matrisin tersini hesaplamaya göre daha doğru sonuçlar verir.

LPC değerlerinin ortak değişinti ile hesaplanması

Ortak değişinti ile LPC değerlerinin hesaplanması yöntemi öz-ilinti ile hesaplanması yöntemine zıt bir yaklaşıma sahiptir.

$$\emptyset_n(i,j) = \sum_{m=-i}^{N-1-(i-j)} s_n(m)s_n(m+i-j) \qquad 1 \le i \le p,\ \ 0 \le j \le p \qquad (1.12)$$

LPC katsayıları öz-ilintilerde olduğu gibi ortak değişintiler arasında da bazı ilişkilerin oluşmasına neden olur.

$$\sum_{j=1}^{p} \alpha_j \emptyset_n(i-j) = \emptyset_n(i) \qquad (1.13)$$

Ortak değişinti yönteminde de tıpkı öz ilintide olduğu gibi çözülmesi gereken p değişkenli p adet eşitlik yazılabilir (1.14).

$$\begin{bmatrix} \emptyset_n(1,1) & \emptyset_n(1,2) & \cdots & \emptyset_n(1,p) \\ \emptyset_n(2,1) & \emptyset_n(2,2) & \cdots & \emptyset_n(2,p) \\ \vdots & \vdots & \ddots & \vdots \\ \emptyset_n(p,1) & \cdot & \cdots & \emptyset_n(p,p) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} \emptyset_n(1,0) \\ \emptyset_n(2,0) \\ \vdots \\ \emptyset_n(p,0) \end{bmatrix} \qquad (1.14)$$

(1.14)'ün hesaplanması sol tarafında yer alan matrisin Toeplitz olmamasından dolayı (1.11) kadar basitlik göstermez, çünkü bu sefer denklemin çözülmesi için bir şekilde (ör: Cholesky çözümlemesi) matrisin tersinin hesaplanması gerekir.

LPC Değerlerinin Kafesleme ile Hesaplanması

LPC değerlerinin kafesleme ile hesaplanması yöntemi esasında öz-ilinti ilişkilerine göre çalışan bir yöntemdir. (daha doğrusu yöntemler grubu) Kafesleme yönteminde diğer yöntemlerde ayrı aşamalar halinde gerçekleştirilen matris korelasyon değerlerinin hesaplanması ve doğrusal denklemlerin çözülmesi işleri tek bir aşamada birleştirilmiştir. En büyük avantajı, kafesleme yöntemince hesaplanan doğrusal öngörü katsayılarının her zaman kararlı süzgeçler vereceğinin garanti edilmiş olmasıdır. Diğer taraftan öz-ilinti ve ortak değişinti yöntemlerinden üretilen süzgeçler için bu tip bir garanti verilememektedir.

Doğrusal öngörü katsayılarının nicemlenmesi

Doğrusal öngörü analizi çok güçlü bir analiz tekniğidir ve birçok konuşma kodlama algoritmasının temelini oluşturur. Konuşma sinyalinin kodlanması sırasında hesaplanan LPC değerleri hem en az bant genişliği tüketecek hem de en az spektral bozulmaya neden olunacak şekilde nicemlenmelidir. LPC değerlerinin skalar veya vektör nicemlenmesi konusunda pek çok çalışma yapılmıştır; genellikle düşük bit oranlarında çalışan kodlayıcılarda doğrudan LPC değerlerinin nicemlenmesi uygulanabilir değildir. $H(z) = 1/A(z)$ süzgecinin kararlı olabilmesi için tüm kutupları birim çember içinde yer almalıdır. Eğer α_i değerleri doğrudan nicemlenirse $H(z)$ süzgecinin kararlılığı kolaylıkla garanti edilemez. Bu yüzden PARCOR (*partial correlation*) katsayıları kullanılır. Kondoz'da (2004) yer alan sözde programlar vasıtasıyla PARCOR ve LPC değerleri birbirlerine kolaylıkla dönüştürülebilir:

LPC'den PARCOR'a

$$a_j^p = \alpha_j \quad 1 \le j \le p$$

for i=p, p-1, …, 1

$$a_j^{i-1} = \left(a_j^i + a_i^i a_{i-j}^i\right)/1 - k_i^2 \quad 1 \le j \le i-1$$
$$k_{i-1} = a_{i-1}^{i-1}$$

PARCOR'dan LPC'ye

for i=1, 2, …, p

$$a_i^i = k_i$$
$$a_j^i = a_j^{i-1} - k_i a_{i-j}^{i-1} \qquad 1 \le j \le i-1$$
$$\alpha_j = a_j^p \qquad 1 \le j \le p$$

Her ne kadar LPC süzgecinin kararlılığı $|k_i| \le 1.0$ olmak kaydıyla yerine getirilmiş olsa da, hala yassı olmayan spektral hassasiyet nedeniyle nicemleme sorunsuz değildir; 1 değerine yakın k_i değerlerinin yüksek doğrulukla nicemlenmesi gerekirken, 1

değerinden uzak olanların yüksek doğruluğa gereksinimi yoktur. Bu yüzden doğrusal olmayan k_i dönüşümlerine ihtiyaç duyulur. Bunlar LAR (log-area-ratio) ve IS *(inverse sine*) dönüşümleridir (Kondoz 2004).

PARCOR'dan LAR'a

$$g_i = \log\left(\frac{1-k_i}{1+k_i}\right) \qquad 1 \le i \le p \tag{1.15}$$

LAR'dan PARCOR''a

$$k_i = \frac{1-10^{g_i}}{1+10^{g_i}} \qquad 1 \le i \le p \tag{1.16}$$

PARCOR'dan IS'ye

$$s_i = \sin^{-1}(k_i) \qquad 1 \le i \le p \tag{1.17}$$

IS'den PARCOR'a

$$k_i = \sin(s_i) \qquad 1 \le i \le p \tag{1.18}$$

Kondoz'a (2004) göre LAR ve IS değerleri kullanılarak skalar nicemleyicilerde tatmin edici sonuçlara ulaşılmışsa da, söz konusu değerlerin temelde iki dezavantajı bulunmaktadır. Birinci dezavantaj LAR – IS değerleri için ayrılması gereken yer konusundadır; kabul edilebilir bir spektral bozulma için değişken başına en az 4 bitlik nicemleme yapılmalıdır; bu da 10. dereceden bir süzgeç için çerçeve başına 35-40 bit nicemlenme çıktısı demektir. Bu durumda, 10 ms'lik çerçeve boyunda çalışan bir kodlayıcı için bir saniyede aktarılan bitlerden 3500 – 4000 kadarı sırf LAR – IS değerleri için ayrılmalıdır. Söz konusu LAR – IS değerleri için harcanan bant genişliği, 8000 bps'lik taşıyıcı kanalın kapasitesinin yarısına, 4800 bps'lik taşıyıcı kanalın kapasitesinin yüzde seksenine tekabül etmektedir. İkinci dezavantaj ise LPC'lerin birbirleriyle olan muhtemel korelasyonunun görmezden gelinmesi konusunda ortaya

çıkmaktadır; LAR ve IS değerlerinde LPC'ler arası olası korelasyonlar gözetilmemekte, bu yüzden kodlama verimliliği arttırılamamaktadır.

LSP (spektral hat çiftleri – *line spectrum pairs*) veya LSF (spektral hat frekansları – *line spectral frequencies*) konsepti ilk olarak sinyal işleme dünyasına Itakura (1975) tarafından kazandırılmıştır. LSF'ler konuşma sinyalinin spektral bilgisini frekans bölgesinde kodlarlar, skalar nicemlemede LAR ve IS'e karşı bariz bir iyileşme sağlamazlar, öte yandan kodlayıcıların formantlara ve duyusal algı modellerine göre çalışmalarına imkân tanırlar. LSF $H(z)$ değerlerinin PARCOR değerlerine dönüştürülmesi işlemi aşağıdaki denklemlerde gösterilmiştir (Kondoz 2004):

$$H(z) = \frac{1}{A_p}\ (z)$$
$$A_p(z) = 1 + \sum_{k=1}^{p} \alpha_k z^{-k} \tag{1.19}$$

PARCOR değerlerinin LPC'ler ile ilişkisi göz önüne alındığında

$$\begin{aligned} A_{p-1}(z) &= A_p(z) - k_p B_{p-1}(z) \\ B_p(z) &= z^{-1}\left[B_{p-1}(z) - k_p A_{p-1}(z)\right] \\ B_0(z) &= z^{-1} \quad A_0(z) = 1 \\ B_p(z) &= z^{-(p+1)} A_p(z^{-1}) \end{aligned} \tag{1.20}$$

eşitlikleri yazılabilir. $B_p(z)$ fonksiyonu sayesinde $A_p(z)$ transfer fonksiyonu tek ve çift simetriye sahip *P* ve *Q* transfer fonksiyonlarına ayrıştırılabilir.

$$
\begin{aligned}
P_{p+1}(z) &= A_p(z) - B_p(z) \quad \mathrm{k_{p+1}} = 1 \\
Q_{p+1}(z) &= A_p(z) + B_p(z) \quad \mathrm{k_{p+1}} = -1 \\
A_p(z) &= \frac{1}{2}\left(P_{p+1}(z) + Q_{p+1}(z)\right)
\end{aligned}
\tag{1.21}
$$

P ve Q transfer fonksiyonları sadece $A_p(z)$ cinsinden yazılırsa geriye köklerinin bulunması gereken iki denklem kalır:

$$
\begin{aligned}
P_{p+1}(z) &= A_p(z) - z^{-(p+1)}A_p(z^{-1}) \\
P_{p+1}(z) &= 1 + \left(\alpha_1 - \alpha_p\right)\mathrm{z}^{-1} + \cdots + \left(\alpha_p - \alpha_1\right)\mathrm{z}^{-\mathrm{p}} - \mathrm{z}^{-(\mathrm{p}+1)} \\
P_{p+1}(z) &= \mathrm{z}^{-(\mathrm{p}+1)} \prod_{i=0}^{p+1}(z + a_i) \\
Q_{p+1}(z) &= \mathrm{z}^{-(\mathrm{p}+1)} \prod_{i=0}^{p+1}(z + b_i)
\end{aligned}
\tag{1.22}
$$

p+1. derecedeki P ve Q polinomlarının birer köklerinin 1 veya -1 olduğu bilindiğinden her iki polinom da birer alt derece olan p. dereceye indirgenebilir.

$$
\begin{aligned}
P'(z) &= \frac{P_{p+1}(z)}{1 - z} = A_0 z^p + A_1 z^{p-1} + A_2 z^{p-2} + \cdots + A_p \\
Q'(z) &= \frac{Q_{p+1}(z)}{1 + z} = B_0 z^p + B_1 z^{p-1} + B_2 z^{p-2} + \cdots + B_p \\
A_0 &= 1 \quad B_0 = 1 \\
A_k &= \left(\alpha_k - \alpha_{p+1-k}\right) + A_{k-1} \\
B_k &= \left(\alpha_k + \alpha_{p+1-k}\right) - B_{k-1}
\end{aligned}
\tag{1.23}
$$

LSF katsayıları p. derecedeki $P'(z)$ ve $Q'(z)$ polinomlarının değerleri 0 ve π arasında değişen kökleridir. Tüm kökler birim çember üzerinde yer alırlar ve kesinlikle sıralıdırlar. Köklerin hesaplanması için karmaşık kök, gerçek kök, oran süzgeci (Jayant ve Noll 1984), Chebysev dizileri (Kabal ve Ramachandran 1986) ve uyarlanabilir ardışık LMS (Cheetham 1987) gibi yöntemler önerilmiştir.

$$0 \leq \omega_{q,0} \leq \omega_{p,0} \leq \omega_{q,1} \leq \omega_{p,1} \leq \pi \tag{1.24}$$

LSF değerlerinden LPC' ye dönüşümü işlemi görece LPC-LSF dönüşümüne göre çok daha basittir. LPC değerlerinin elde edilmesi için sırasıyla P' ve Q' polinomları hesaplanır, bu polinomlardan P ve Q elde edilir ve bu polinomların toplamlarının yarısından $A_p(z)$'ye ulaşılır.

EK 2 Kaos Teorisi

Bu ek bölüm, kaotik öznitelikler steganaliziyle kodlayıcı tanımanın anlaşılabilmesi için gerekli olan asgari kaos teorisi altyapısını barındırmaktadır. Hatırlanacağı üzere, kaotik özniteliklere dayalı steganaliz, hatalı en yakın komşu oranı ve Lyapunov katsayılarına dayanmaktadır. Kodlayıcı tanıma ise hatalı en yakın komşu oranına göre çalışmaktadır. Söz konusu kaotik öznitelikler gerçekte dinamik sistemlerin kaotiklik derecesinin betimlenmesinde kullanılan basit matematiksel araçlardır. Okuyucunun bu matematiksel araçların nasıl işlev gördüklerini idrak edebilmesi için, temel düzeyde kaos teorisine hakim olması zaruridir. Tezin kuramsal temeller kısmının daha çok literatürdeki güncel çalışmaları içermesi arzu edildiğinden, kaos teorisinin okuyucuya ek bir bölümde aktarılmasının daha uygun olacağı düşünülmüştür.

Kaos teorisinin tarihçesi

Mühendislik bilimlerinde çözümlerin çoğunlukla denklemlerde arandığı, veri işlemenin son derece zor ve külfetli olduğu yıllar boyunca doğrusal sistemler doğrusal olmayan sistemlere göre daha fazla ilgi çekmiştir. Ancak bu durum teknoloji ilerledikçe ve teknolojinin ürünleri olan bilgisayarların insan hayatına derinlemesine nüfuz etmesiyle tersine dönmüştür. Bilgisayarların en önemli getirilerinden olan yüksek işlem gücü sayesinde, doğrusal olmayan sistemlerin incelenebilmesi kolaylaşmış ve böylece doğrusal olmayan devimbilim ve kaos teorileri bilim – mühendislik için vazgeçilemez öneme haiz olmuşlardır.

Kaos teorisi ilginç bir şekilde, farklı disiplinlerde birbirlerinden alakasız teoriler ve buluşlarla ortaya çıkmıştır. Bilim dünyası kaos teorisindeki ilk heyecan verici gelişmeyi Rus matematikçi Lyapunov'a borçludur. Kendisi, doğrusal olmayan devimbilim davranışları üzerinde çalışırken bahse konu davranışlardan doğrusal olmayan sistemlerin ilk durumlarının neden olduğu hassasiyeti ölçmeyi başarmıştır. Onun ardından Poincare topolojik olarak, karmaşık gibi gözüken sistemlerin düzenli davranışlar sergileyebileceğini göstermiş, Julia ve Fatou ise basit denklemlerin dahi

sonsuza giden geometrik kümeler üretebileceğini tespit etmişlerdir. Daha sonra, Mandelbrot söz konusu sonsuza giden geometrik kümelere fraktal ismini koymuştur. Meteoroloji gibi farklı bir disiplinde çalışan Lorenz hava tahminleri modelini geliştirmeye çalışırken belki de kaosun en bilindik sembolü olan Lorenz çekicisini (*attractor*) bulmuştur.

Kaos'un görsel olarak ifade edilmesi sayesinde teorinin bilimsel kitleye yayılması kolaylaşırken Feigenbaum, May ve Yorke teoriyi oluşturan fikirleri bir araya getirerek teoriyi oluşturan temel prensipleri belirlemişlerdir. Prensiplerin geliştirilmesi için sayısız bilim adamı ter dökmüşse de, bulunan matematiksel bulguların kaos terimi ile ilişkilendirilmesi Li ve York'a (1975), kaos teorisinin özerk bir alan kabulü de Ruelle (1980) ve Takens'a (1981) nasip olmuştur. Günümüzde kaos teorisi pek çok uygulamalarda yaygın bir şekilde kullanılabilmektedir; bunlardan başlıca olanları fraktal kodlama, sinyal ayrıştırma, radar imzaları, sualtı haberleşmesi, kaotik öngörüm ve saklı Markov modelleridir.

Kaotik sistemler

Kaotik sistemlerin ortak özelliklerinin bir tanımını yapmak gerekirse, bu sistemler genel olarak aperiyodiktirler, gerekircilerdir (*deterministic*), ilk koşullara oldukça hassastırlar, kendi kendine benzeyen yapıtaşlarından oluşurlar ve uzun vadede sınırlandırılmış (*bounded*) davranış gösterirler (Sprott 2003). Herhangi bir sistemin sırf çıktısına bakarak gerekirci olmayan rastgele mi olduğu yoksa gerekirci kaotik mi olduğunu belirlemek oldukça güçtür. Örmeğin bazı gerekirci kaotik çıktılar çeşitli rastgelelik testlerinden geçebilirler, diğer taraftan tamamen kaotik özellik gösteren çıktılar da bilgisayar ortamında sayısallaştırılırken kırpma (*truncation*) gürültüsüne maruz kalırlar, bu sayede bir miktar rastgelelik kazanırlar.

Faz uzayı ve çekici kavramları

Genel bir doğrusal olmayan dinamik sistem, gözlemlenebilir çıktılarının (durum makinelerinin) aldığı çeşitli durumlar ile betimlenebilir. Örneğin salınmakta olan bir sarkaç sistemi, sarkacın zaman içindeki açısal pozisyonları ve açısal hızları ile tasvir edilebilir. *m* çeşit gözlemlenebilir çıktıya sahip olan bir dinamik sistem, faz uzayında *m* boyutundaki $S = (s_1, s_2, ..s_m)$ vektörü ile anlatılır. Faz uzayındaki bu vektör, zaman içerisinde önceki durumlarına göre evrimleşir. Genel bir dinamik sistemin n+1. zaman indeksindeki durumunun n. zaman indeksindeki durumuna olan bağlılığı (2.1)'de matematiksel olarak gösterilmiştir:

$$S(n+1) = f(S(n)) \tag{2.1}$$

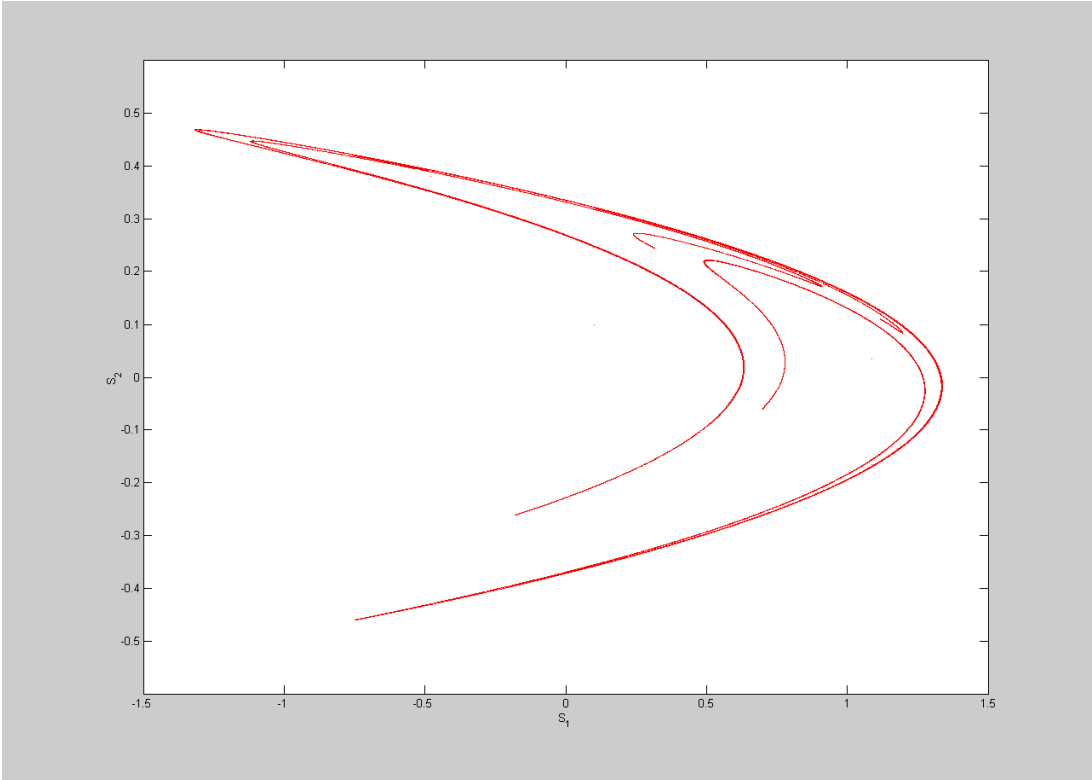
Denklem (2.2)'de verilmiş olan Henon kesintili eşleme (*discrete mapping*) fonksiyonu, (2.1)'de tanımlanan genel dinamik sistem'in pratik bir uygulamasıdır: (s_1 ve s_2 dinamik sistemin gözlemlenebilir çıktılarıdır.)

$$\begin{aligned} s_1(n+1) &= 1 + s_2(n) - 1.28s_1^2 \\ s_2(n+1) &= 0.35s_1 \end{aligned} \tag{2.2}$$

Eşleme fonksiyonu çok basit gibi gözükse de, çıktıları son derece karmaşıktır, çıktılarda asla tekrarlar ve örtüşmeler bulunmaz. Öte yandan tüm bu karmaşıklığa rağmen, faz uzayı sayesinde karmaşık çıktılara sahip olan dinamik bir sistemin basit ve görsel olarak anlatımı yapılabilir.

Şekil 1'de Henon kesintili eşleme fonksiyonunun çekicisi yer almaktadır. Kaos terminolojisinde çekici (*attractor*), bir dinamik sisteme ait değişkenlerin (çıktıların) uzun bir zaman zarfında (evrilirken) bir araya getirilmesi veya hesaplanmasıyla oluşturulmuş, dinamik sistemin bütün davranışlarını sergileyebilme kabiliyetine sahip, çok boyutlu çıktı kümesi anlamına gelmektedir (Sprott 2003). Söz konusu çıktı kümesi sistemi oluşturan değişkenlerin sayısına bağlı olarak değişik boyutlarda ifade edilebilir.

Henon kesintili eşleme fonksiyonunun iki gözlemlenebilir çıktısı olmasından dolayı, çekicisi iki boyutlu bir grafikle ifade edilebilmektedir. Çekici, s_1 çıktısının s_2 çıktısına göre değişimini görsel olarak sunmaktadır.



Şekil 1 Henon çekicisi

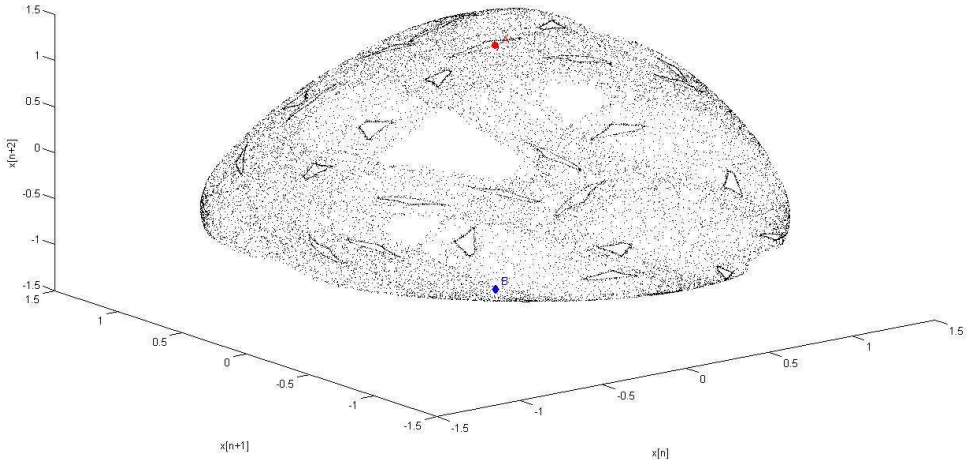
Hatalı en yakın komşular ve gömme boyutu

Dinamik sistemlerin bulundukları durumların görsel aktarımında kullanılan çekici çizimleri (faz uzayı çizimleri) daha çok kaos teorisi uygulamaları ile özleştirilse de, tüm dinamik sistemlerin davranışlarının anlatımında faydalı olabilir. Aşağıda yer alan denklem (2.3)'te Lorenz'in üç boyutlu kaotik eşlenimine yer verilmiş, bahse konu eşlenime ait çekicisi faz alanında Şekil 2'de üç boyutlu, Şekil 3'teyse iki boyutlu olarak gösterilmiştir.

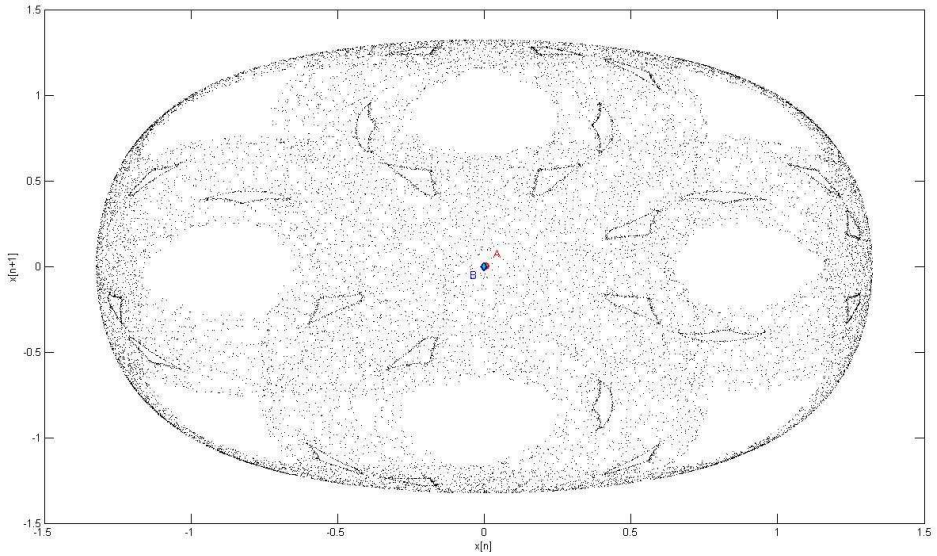
$$X[n+1] = X[n] \times Y[n] - Z[n]$$
$$Y[n+1] = X[n] \quad (2.3)$$
$$Z[n+1] = Y[n]$$

İki ve üç boyutlu çekiciler dikkatle incelendiğinde, iki boyutlu çekicide komşuluk ilişkisine sahip bazı noktaların, üç boyutlu çekicide birbirlerinden uzak kaldıkları görülmektedir. Şekil 2 ve 3'te gösterildiği gibi eğer düşük boyuttaki bir komşuluk ilişkisi yüksek boyutlarda ortadan kalkıyorsa bu durum *hatalı en yakın komşuluk* ilişkisi olarak tanımlanmaktadır.

Yukarıdaki çekicileri oluşturulan denklemler zaten bütünüyle bilindiğinden dolayı gömme boyutunun tespit edilmesi için hatalı komşuluk ilişkilerinin incelenmesi gerekli değildir. Öte yandan nasıl çalıştığı tam olarak bilinmeyen, sadece çıktıları değerlendirmeye alınarak hakkında yorum yapılması hedeflenen bir dinamik sisteme ait çıktıların incelenerek, içindeki gömülü bilginin kaç boyutlu yapıya sahip olduğunun incelenmesi kaotik analiz için son derece önemlidir. Matematikte gömme kavramı matematiksel bir yapı örneğinin, başka bir örneğin içinde yer alması anlamına gelmektedir. Gömme boyutu ise bir dinamik sistemin davranışlarının faz uzayında noksansız bir şekilde betimlenebilmesi için ihtiyaç duyulacak eksen sayısıdır. Daha basit bir ifadeyle dinamik bir sistemin eksiksiz modellenebilmesi için gerekli olan asgari değişken sayısıdır.



Şekil 2 Lorenz'in üç boyutlu kaotik eşleniminin üç boyutlu çekicisi



Şekil 3 Lorenz'in üç boyutlu kaotik eşleniminin iki boyutlu çekicisi

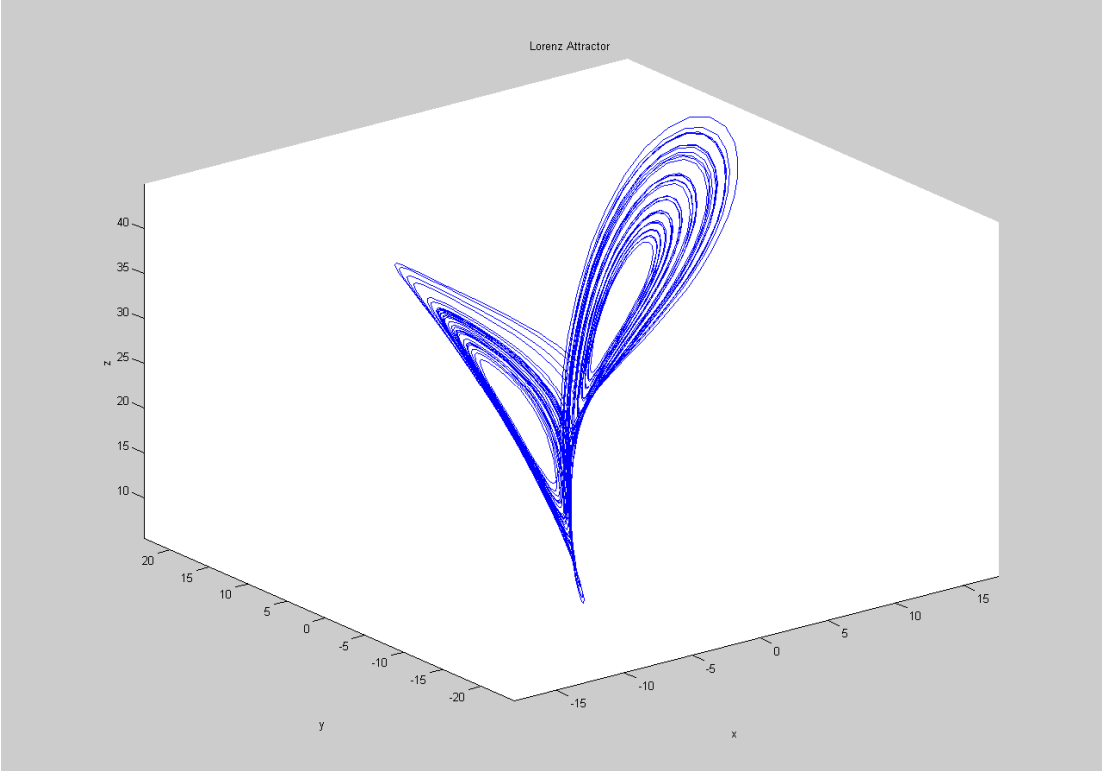
Lorenz tuhaf çekicisi, fraktal kavramı ve kelebek etkisi

Gösterildiği topolojik boyuta sığmayan, fraktal (*fractal*) isimli yapıların sonsuza kadar yinelendiği çekiciler, "*tuhaf*" olarak nitelendirilir; bu tanıma doğrultusunda tuhaf özelliklere sahip çekiciler de "*tuhaf çekiciler*" olarak isimlendirilir (Sprott 2003). Fraktal kavramının basit bir tanımı yoktur; ancak literatürde üzerinde en fazla uzlaşılan tanımlamaya göre, "fraktal" demek kendini yineleyen, olağan topolojik boyutlarla ifade edilemeyen, kendi kendine benzeyen (*self-similar*) matematiksel yapı demektir (Sprott 2003).

Kaos teorisiyle birlikte en çok anılan ve en bilindik olan tuhaf çekici, Lorenz'in atmosfer modellemesi esnasında geliştirdiği Lorenz tuhaf çekicisidir. Lorenz atmosferi hepsi (2.4)'te yer alan 3 temel denklem ile modellemiştir. Denklemde yer alan X konveksiyon akışını, Y yatay ısı dağılımını ve Z de dikey ısı dağılımını belirtmektedir. (σ=160, τ=40, b=4)

$$\begin{aligned} \dot{X} &= \sigma(Y - X) \\ \dot{Y} &= \tau X - Y - XZ \\ \dot{Z} &= -bZ + XY \end{aligned} \tag{2.4}$$

Şekil 4'te Lorenz denklemlerine göre çizilmiş olan tuhaf bir çekici yer almaktadır. Lorenz modelinin en önemli özelliği ilk koşullara olan hassasiyetidir. Model o derecede ilk koşullara hassastır ki, analog bilginin sayısallaştırılmasından kaynaklanan nümerik hatalar, ilk değerlerdeki çok küçük sapmalar bile sistemin gelecekte bambaşka davranışlar sergilemesine neden olur. Lorenz'in üzerinde çalıştığı konu dünya atmosferinin modellenmesi olduğundan, bahse konu modele göre de Meksika'daki bir kelebeğin kanat çırpışlarının Kaliforniya'daki bir kasırgaya nedeni olabileceğini ortaya koyduğundan, söz konusu ilk değerlere karşı hassasiyet "*kelebek etkisi"* olarak adlandırılmaktadır.



Şekil 4 Lorenz çekicisi görsel gösterimi

Fraktal boyutu

Kaotik çekicilerin en genel özellikleri, yer aldıkları faz uzayını tam dolduramayışlarıdır. Şekil 1'de yer alan Henon çekicisi 2 boyutlu uzayı dolduramasa da, tek boyuta da sığmamaktadır. Bu durumda Henon çekicisinin gerçekte kaç boyutludur? Bu sorunun açık cevabı boyutun 1 ila 2 arasında olduğudur. Mandelbrot (1977) bu soruyu ele almış, çözümünü de fraktal boyutu kavramıyla ilişkilendirmiştir. Sprott (2003), bugüne kadar önerilmiş fraktal boyutu hesaplama yöntemlerini, yaklaşımlarına göre dört ana sınıf altında özetlemektedir: Benzerlik boyutu, kapasite boyutu, korelasyon boyutu ve Lyapunov boyutu.

Öngörülebilirlik kavramı ve Lyapunov katsayıları

Dinamik bir sistemin gelecekteki durumlarının ve çıktılarının öngörülebilmesi ihtiyacı, kaos teorisinin geliştirilmesindeki en önemli anaçlardan biri olmuştur. Beklendiği üzere

kaotik bir sistem, kısa vadedeki öngörülebilirliği ve uzun vadedeki öngörülemezliği ile betimlenebilir.

Daha önceden de belirtildiği üzere, kaotik bir sistem faz uzayında hem yakınsallık hem de ıraksallık özellikleri gösterir. Bu ikili kombinasyon, bir sistemin iki benzer noktadan zaman içinde bambaşka pozisyonlara evirilmesine yol açabilir. İlk değerlere karşı hassasiyet, bir dinamik sistemin öngörülebilmesine temel kısıtlar koyar. Öte yandan kaotik bir sistem başlangıcından itibaren ne kadar evrilirse evirilsin asla çekicisini terk edemez. (Çekicinin kendisi aslında bir öngörüdür.)

Lyapunov katsayıları dinamik bir sistemin ne kadar öngörülebilir olduğunu belirten matematiksel değerlerdir. Matematikteki tanımlarına göre Lyapunov katsayıları, sonsuz yakınlıktaki yörüngelerin ilk değerlerine göre birbirlerinden ayrışma oranlarıdır. Faz uzayında ilk değerleri arasında δZ_0 kadar fark bulunan iki yörünge zaman ilerledikçe birbirlerinden (2.5)'teki

$$|\delta Z(t)| \approx e^{\lambda t}|\delta Z_0| \tag{2.5}$$

kadar ayrışırlar, bu denklemde tanımlanmış olan λ değeri Lyapunov katsayısıdır. Ayrılma oranı yörüngenin değişik yönleri için farklılık gösterir; bu sebepten çekimcinin yer aldığı her bir boyut için ayrı katsayı hesaplanır. *d* tane boyut için hesaplanmış katsayılardan en önemlisi en büyük olanıdır (MLK: Maksimum Lyapunov katsayısı); çünkü dinamik bir sistemin öngörülebilirliğini en büyük Lyapunov katsayısınca belirlenir. MLK'nin nasıl hesaplanacağı (2.6)'da sunulmuştur:

$$\lambda = \lim_{t\to\infty} \frac{1}{t} \ln \frac{|\delta Z(t)|}{|\delta Z_0|} \tag{2.6}$$

Lyapunov katsayısının 0'dan büyük olması durumunda, dinamik sistem kaotiklik ve kararsızlık gösterir, eğer katsayı 0'dan küçükse bu durumda sistem kayıplıdır, tek bir noktaya veya periyodik yörüngeye yakınsanmaktadır. Lyapunov katsayısının 0'a eşit olduğu durumlarda sistem kayıpsızdır ve korumacıdır (Sprott 2003).

d boyutlu bir çekimcisi olan dinamik bir sistem için, Lyapunov katsayısı spektrumları $\{ \lambda_1, \lambda_2, \dots , \lambda_d \}$ halindedir. Genellikle hesaplanacak bu değerler başlangıç noktası X_0'a bağlıdırlar. Lyapunov katsayıları faz uzayının tanjant alanındaki vektörlerin davranışını betimlerler ve Jacobian matrisi ile tanımlanırlar:

$$J^t(x_0) = \left. \frac{df^t(x)}{dx} \right|_{x_D} \tag{2.7}$$

J^t matrisi, başlangıç noktası x_0'daki küçük bir farkın son noktaya, $f^t(x_0)$'a kadar ne kadar yayılacağını gösterir:

$$L(x_0) = \lim_{t\to\infty} (J^t \cdot Transpose(J^t))^{1/2t} \tag{2.8}$$

Lyapunov katsayıları $L(x_0)$ matrisinin eigen değerleri olan $\Lambda_i(x_0)$'ın logaritmasından hesaplanır.

$$\lambda_i(x_0) = \log \Lambda_i(x_0) \tag{2.9}$$

Ergodik sistemler için hesaplanacak Lyapunov katsayıları başlangıç noktalarının seçiminden etkilenmezler; değişik başlangıç noktalarından yola çıkılarak hesaplanan Lyapunov katsayıları aşağı yukarı aynı sonuçları verir.

Bir dinamik sistemin kaotik olup olmadığını belirlemek için MLE değerinin hesaplanması yeterlidir. Diğer taraftan Lyapunov katsayısı spektrumu dinamik bir sistem hakkında daha fazla bilgi içerir; Denklem (2.10)'da da gösterildiği üzere bir cins enformasyon boyutu olan Kaplan-Yorke boyutu D_{KY}, (Kaplan ve Yorke 1979) Lyapunov spektrumu sayesinde bulunabilir:

$$D_{KY} = k + \sum_{i=1}^{k} \frac{\lambda_i}{|\lambda_{k+1}|} \tag{2.10}$$

k: Toplamları hala 0'dan büyük olan en büyük k Lyapunov katsayısı

Çatallaşma

Çatallaşma, İngilizce karşılığıyla "*bifurcation*", matematikte bir parametrenin değerindeki ufak bir değişikliğin topolojinin yapısında aniden büyük çapta değişiklik yaratması olarak tanımlanmaktadır (Sprott 2003). Bir dinamik sistem çekicisinin, bir dinamik sistem parametresinin değerindeki ufak bir değişime bağlı olarak bambaşka bir hale bürünmesi durumu kaotik çatallaşma olarak nitelendirilmektedir.

Dinamik davranış hiyerarşisi

Sprott (2003) her biri farklı davranışlara sahip olan birbirlerinden bağımsız 10 adet temel dinamik sistem sınıfı tanımlamıştır. Çizelge 1'de Sprott tarafından tanımlanan temel dinamik sistem sınıflarının listesi yer almaktadır. Listede aşağıya doğru inildikçe, sınıfa ait dinamik sistemlerde öngörülemezlik artmakta ve tanık olunan davranışlar karmaşıklaşmaktadır.

Konuşma sinyali de dâhil olmak üzere gerçek hayatta bilimsel incelemeye tabii tutulan pek çok sinyal türü veya veri dizisi, söz konusu temel dinamik sistem sınıflarının çeşitli oranlarda birbirlerine katılmış kombinasyonlarını içeren sistemlerce üretilmektedirler.

Çizelge 1 Dinamik davranış hiyerarşisi

Sınıf	Örnek
Düzenli öngörülebilir	Gezegen yörüngeleri, saatler, gelgitler
Düzenli öngörülemez	Yazı-tura atımı
Geçici kaos	Tilt oyunu
Aralıklı kaos	Lojistik eşleme A=3.8284
Dar bantlı kaos	Rössler çekimcisi
Geniş bantlı, düşük boyutlu kaos	Lorenz çekimcisi
Geniş bantlı, yüksek boyutlu kaos	Yapay sinir ağları
Kolerasyonlu (renkli) gürültü	Rastgele gezinim (*random-walk*)
Sözde rastgelelik	Bilgisayar tarafından üretilen rastgele diziler, kripto dizileri
Rastgele gürültü	Radyoaktivite

EK 3 MELP Konuşma Kodlayıcısı

MELP (Anonymous 1999) Aralık 1999 tarihinde DoD (Amerika Birleşik Devletleri Savunma Bölümü) tarafından MIL-STD-3005 olarak standartlaştırılan, analog insan konuşmalarının karma uyarlamalı doğrusal öngörü (*mixed excitation linear prediction*) yöntemleri kullanılarak 2400 bps hızındaki sayısal kodlanmış bilgiye dönüştürülmesini gerçekleştiren bir konuşma kodlama algoritmasıdır.

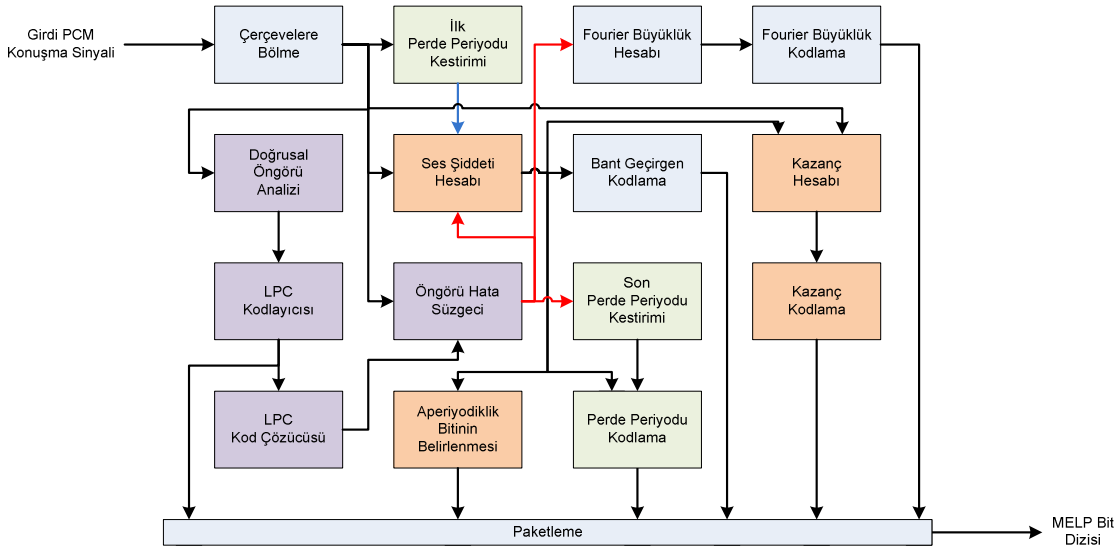
MELP özellikle zorlu arka plan gürültüsü içeren ortamlarda işlev görebilecek bir şekilde tasarlanmıştır. Kodlama ve kod çözme için gereken işlem gücü olabildiğince düşük tutulmuştur; bu sayede taşınabilir cihazlardaki güç tüketiminin sınırlanması hedeflenmiştir. MELP konuşma sinyalinin modellenmesi için kapsamlı başvuru çizelgeleri (*lookup tables*) kullanır, bahse konu başvuru çizelgeleri daha çok erkek İngilizce ve Almanca dilleri için optimize edildiğinden, diğer diller için ses kalitesi düşüş gösterir.

Şekil 1'de sunulmuş olan MELP algoritması geleneksel doğrusal öngörü kodlamasına (LPC) dayanmakta olup, fazladan beş özellik daha içermektedir:

1. Karma uyarma (*mixed excitation*)
2. Aperiyodik darbeler (*Aperiodic pulses*)
3. Uyarlanır spektral iyileştirme (*Adaptive spectral enhancement*)
4. Darbe yayılımı (*Pulse dispersion*)
5. Fourier büyüklük modellemesi (*Fourier magnitude modeling*)

MELP algoritmasında karma uyarma çok bantlı karıştırma modeli kullanılması ile yerine getirilir. Bu model esas olarak, frekans tabanlı seslendirme etkisini sabit bir süzgeç kümesinden gerçeklenen uyarlanabilir süzgeç yapısını kullanarak taklit eder. Karma uyarmanın birincil getirisi geleneksel LPC ses kodlayıcılarındaki geniş bantlı akustik gürültü kaynaklı çınlama etkisinin ortadan kaldırılmasıdır.

Kodlanacak konuşma ötümlü olduğunda MELP kodlayıcısı periyodik veya aperiyodik darbeler sentezleyebilir. Aperiyodik darbeler çoğunlukla ötümlü-ötümsüz çerçeveler arasındaki geçiş çerçevelerinde yaratılırlar. Bu özellik sayesinde konuşma sentezleyicinin düzensiz gırtlak darbelerini tonlu sesler olmaksızın yeniden üretilebilmesi sağlanır.



Şekil 1 MELP blok şeması (Chu 2003)

Uyarlanır spektral iyileştirme süzgeci doğrusal öngörü sentez süzgeçlerinin kutuplarını temel alır. Kullanımı sonucunda üretilecek sentetik konuşmanın formant yapısı doğal sesin formant yapısını daha fazla andıracak şekilde geliştirilir, buna bağlı olarak da sentetik konuşma daha doğal bir görünüm kazanır.

Darbe yayılımı, frekans bölgesinde düzleştirilmiş üçgen darbenin temel alındığı, sabit bir süzgeç kullanımı ile yapılır. Bu süzgeç uyarım enerjisini tüm perde periyoduna yayar ve bu yapılan yayılım işlemi sonucunda sentetik konuşmayı doğallıktan uzaklaştıran sert sesler azaltılır.

Öngörü tortu sinyali Fourier dönüşümüne uğratılarak ilk on Fourier büyüklüğü elde edilir. Bu Fourier büyüklükleri, yüksek frekans bölgesine kıyasla, özellikle algılanabilirliği daha fazla olan düşük frekans bölgesindeki ses bilgilerini taşır.

Kullanımı sayesinde, arka planlarında yüksek gürültü içeren ortamlardaki erkek konuşmacıların sentetik konuşmalarının kalitesi arttırılır.

Bu raporda, ilerleyen bölümlerde önerilen gizli veri gömme tekniğinin MELP kodlayıcısı içerisinde uygulanmasından dolayı, MELP kodlayıcısının çalışmasının bir özeti verilmiştir. Öte yandan, gömülü gizli verinin kestirimi işlemi MELP kod çözücüsünden ve çalışmasından tamamen bağımsızdır, bu yüzden MELP kod çözücüsü çalışması hakkında herhangi bir özete yer verilmeyecektir.

MELP algoritması analiz etme işlemini gerçekleştirirken girdi olarak 1 çerçeve uzunluğunda 8 kHz'de örneklenmiş 16 bitlik 180 adet ses örneğini almaktadır. Yapılan konuşma sinyali analizi sırasında bu örnekler sadece 54 bit uzunluğundaki bir veri olarak kodlanır. 8 kHz'deki 180 adet örnek ve 54 bitlik kodlanmış bilgi 22,5 ms'lik konuşmayı içermektedir.

a. Düşük Frekans Bileşenlerinin Atılması

MELP kodlayıcısı ilk önce kesim frekansı 60 Hz, söndürme kuşağı bastırması 30 dB (*stop-band rejection*) olan 4. dereceden bir Chebychev II yüksek geçiren süzgeci kullanarak çok düşük frekans bileşenlerini atar. MELP kodlayıcısı her zaman ihtiyacı olan miktardaki en güncel konuşma örneklerini tampon hafızasında tutar. Kodlayıcıya giren en son örnek referans noktası olarak kabul edilmektedir.

b. Tam Değer Perde Periyotunun Hesaplanması

Çok düşük frekans bileşenlerinin atılmasından hemen sonra perde periyodunun tam değer olarak hesaplanmasına başlanılır. Önce 6. dereceden 1 kHz'den aşağı frekans bileşenlerini geçiren bir Butterworth süzgeci kullanılarak 1 kHz'den yüksek frekans bileşenleri atılır. Sonra kalan sinyalin 40 ila 160. değerleri arasında normalleşmiş öz-ilintisi en büyük değeri veren indeks bulunur. Bulunan indeks P_1 değişkeni olarak saklanır.

c. Bant Geçirgen Ses Analizi

Konuşma sinyali, her biri 0-500, 500-1000, 1000-2000, 2000-3000 ve 3000-4000 Hz aralıklarını geçiren beş adet altıncı dereceden Butterworth süzgeci kullanılarak, beş adet farklı frekans bantlarındaki seslerin oluşturduğu alt sinyallere bölünür. 0-500 Hz alt sinyali kullanılarak yeniden perde periyotu hesaplanır. Perde periyodu analizi bir önceki aşamada da yapıldığı gibi en yüksek normalleşmiş öz-ilinti değerini veren indeks değerinin seçilmesi ile gerçekleştirilir. Fakat bu aşamada bir önceki stepte olduğu gibi 40'dan 160'a kadar olan tüm indekslerin normalleşmiş öz-ilinti değerleri hesaplanmaz. Hesaplanan değerler P_1'in kendisi, 5 örnek önceki ve 5 örnek sonraki indeks değerleri ile MELP analizi tamamlanmış bir önceki 22.5 ms'lik çerçevenin P_1'i ile bu değerin 5 örnek önceki ve 5 örnek sonraki indeksleri ile sınırlı tutulur. En yüksek normalleşmiş öz-ilinti değeri V_{bp1} değişkeni, bu değeri sağlayan indeks ise kesirli P_2 değişkeni olarak kaydedilir. Bulunan tam sayı perde periyodu ve bu tam sayı indeksteki öz-ilinti değeri üzerinde kesir perde periyodu düzeltmesi gerçekleştirilir. Diğer alt-bantlarda da aynı işlem uygulanarak V_{bp2}, V_{bp3}, V_{bp4}, V_{bp5} değişkenleri elde edilir.

d. Kesir Perde Periyotu Düzeltilmesi

Girdi olan perde periyodunun ve normalleşmiş öz-ilinti değerlerinin daha fazla kesinleştirilmesi için kullanılır.

e. Aperiyodik Bitinin Belirlenmesi

Eğer elde edilen V_{bp1} değişkeni 0,5'ten küçükse bu bit 1'e eşitlenir. Eğer V_{bp1} değişkeni 0,5'ten büyük eşitse 0'a eşitlenir.

f. Doğrusal Öngörü Analizi

Referans değerinden itibaren tampon hafızada tutulan 25 ms'lik Hamming pencere içine alınmış örnekler üzerinde 10. dereceden doğrusal öngörü analizi gerçekleştirilir. Levinson-Durbin öz-ilinti analiz prosedürleri vasıtasıyla doğrusal öngörü katsayıları (LPC) bulunur. Bulunan tüm katsayılar 0.994 ile çarpılır.

g. Doğrusal Öngörü Tortu Hesaplanması

Kodlanacak PCM formatındaki ses sinyalinin, LPC katsayılarından elde edilen süzgeçten geçirilmesi ile tortu sinyali elde edilir.

h. Tortu Sinyalinin Aşırılık Hesabı (Peakness Calculation)

Tortu sinyalinin son 160 örnekte ne kadar doruk oluşturduğuna bakılır. Eğer elde edilen aşırılık değeri tanımlanmış eşik miktarlarından yüksekse V_{bpi} değişkenlerinde düzeltmeler yapılır.

i. Son Olarak Perde Periyodunun Hesaplanması

Son yapılacak perde periyodu hesabı için tortu sinyali, 1 KHz'lik kesim frekansı olan 6. dereceden bir Butterworth süzgecinden geçirilir. Daha önceden elde edilmiş P_2 indeks değeri ile bu değerden 5 önce ve 5 sonrası aralığında en yüksek öz-ilinti veren indeks değeri bulunur. Bulunan indeks ve normalleşmiş öz-ilinti değeri kesir perde periyodu düzeltmesine tabi tutulur. Perde periyodunun çarpanlarının varlıkları bir prosedürle kontrol edilir, sonucun hatalı çıkmasına neden olabilecek çarpanların etkisi aynı prosedürle giderilir ve son düzeltmeler yapılarak en son perde periyotu hesabı (P_3) tamamlanır.

j. Kazanç Hesabı

Her bir PCM konuşma sinyali penceresi üzerinde perde periyoduna göre uzunluğu uyarlanabilir değişen pencereler kullanılarak iki adet kazanç katsayısı hesaplanır. İki kazanç katsayısının pencere referans noktaları arasında 90 örneklik fark bulunur.

k. Öngörü Parametrelerinin Nicemlenmesi

Önce elde edilmiş olan 10 tane LPC katsayısı spektral hat frekanslarına (LSF) dönüştürülür. Bundan sonraki aşamada LSF değerleri giderek artacak ve aralarında en az 50 Hz'lik uzaklık olacak şekilde dönüşüme uğratır. Elde edilen LSF vektörü 4 seviyeli bir vektör nicemleyiciye sokularak nicemlenir. Seviyeler sırayla 128, 64, 64, 64 farklı vektör içerir. Nicemleme Öklid mesafeleri ve ağırlık değerlerine göre hesaplanan hata değerinin en aza indirgenmeye çalışılması ile yapılır. Kod çizelgelerinde yer alan tüm değerlerin birlikte oluşturdukları ihtimallerin her birini kontrol edip, en az hataya

neden olanların seçilmesi yerine M-en iyi (*M-best approximation*) yakınlaştırması kullanılır. MELP kodlayıcısı M=8 en iyi yakınlaştırıcısı ile vektör nicemleme işlemini gerçekleştirir. Nicemlenmiş LSF'lerin artan sırayla olması ve aralarında en az 50 Hz olmasını sağlayacak işlemler yinelenir. Sonuç olarak LSF değerleri 7+6+6+6=25 bite nicemlenmiş olur.

l. Perde Periyodunun Nicemlenmesi

Ötümlü çerçeveler için en son elde edilmiş perde periyodu P_3, düzgün logaritmik ölçekli 99 seviyeli nicemleyici tarafından nicemlenir. 99 farklı seviye 7 bit ile ifade edilir; 7 bitlik alan 128 farklı ihtimal sunduğundan, kalan 29 değer perde periyodu tanımının anlamsız olduğu ötümsüz ve silme çerçevelerini ifade eder. Kısaca perde periyodu bitleri, ötümlü-ötümsüz ayrımı bilgisi ile eğer çerçeve ötümlü ise nicemlenmiş perde periyodunun indeks değerini aynı alanda taşır.

m. Kazanç Nicemlenmesi

Elde edilen kazançlar birisi 3 bit diğeri 5 bit olacak şekilde nicemlenir. 5 bitlik yapılan nicemleme, üst ve alt değerleri 10 ila 77 dB arasında düzgün, 3 bitlik nicemleme uyarlanabilir olarak yapılır.

n. Bantların Ötümlülük Nicemlemesi

Bant geçirgen ses analizi sırasında bulunmuş olan beş adet V_{bpi} değişkeni 0 veya 1 şeklinde nicemlenirler. Eğer V_{bp1} değişkeni 0,6'dan küçükse bu MELP kodlanmış çerçevenin ötümsüz olduğu anlamına gelmektedir, diğer değişkenler de 0 olarak nicemlenirler. Eğer V_{bp1} değişkeni 0,6'dan büyükse diğer V_{bpi} değişkenleri kendi değerlerinin 0,6'dan fazla olup olmamasına göre 1 veya 0 olarak nicemlenirler. Yapılan işlemden de kolaylıkla anlaşılabileceği gibi V_{bpi} değişkenleri sadece ötümlü çerçeveler için anlamlıdır.

o. Fourier Büyüklüklerinin Hesaplanması ve Nicemlenmesi

PCM konuşma sinyali nicemlenmiş LSF değerlerinin elde edilmesiyle yaratılan süzgeçten geçirilerek tortu sinyali bulunur. Tortu sinyali üzerinden bir Hamming penceresi geçirilir, ardından pencere geçirilmiş sinyalin üzerinde FFT dönüşümü

uygulanır. FFT dönüşümün tamamlanmasının ertesinde FFT'nin karmaşık çıktısından büyüklükler elde edilir, spektral doruk noktası bulma algoritmasıyla harmonikler tespit edilir. 10 adet FFT büyüklüğü 256 farklı vektör içeren bir kod çizelgesi ile 8 bit olarak nicemlenir. Nicemleme sırasında ağırlıklı Öklid uzaklığından kaynaklanan hatanın en aza indirgenmesine çalışılır.

p. Hata Kodlama ve Bitlerin Paketlemesi

MELP kodlanmış olan çerçevelerin ötümlü veya ötümsüz tipte olmalarına göre, kod çözücüye iletilmesine ihtiyaç duyulan parametrelerinin sayısı değişmektedir. Ötümlü MELP çerçevelerinde kodlayıcı tarafından bulunmuş tüm değişkenler gerekli olmaktayken, ötümsüz MELP çerçevelerinde V_{bpi} bitlerine, Fourier büyüklüklerine, aperiyodik bitine ve perde periyodu tanımına ihtiyaç duyulmamaktadır. Perde periyodunun ifade edildiği bitlerle aynı zamanda ötümlü-ötümsüz ayrımı da yapıldığından, bu bitler başka amaçlarla kullanılamaz. Diğer taraftan V_{bpi} bitleri, Fourier büyüklüklerini ifade eden 8 bit, aperiyodik biti ötümsüz çerçevelerin bit hatalarına karşı dayanıklılığı arttırmak için kullanılır. Ötümsüz çerçevelerde taşınan anlamlı değişkenlerin önemli olanlarının ileri hata düzeltimi (FEC) bilgileri, bahse konu hem anlamsız hem de kullanılabilir alanlarda taşınır. Ötümsüz çerçevelerde ileri hata düzeltimi kullanılması, iletişim kanalından kaynaklanabilecek olan hatalara karşı MELP kod çözücüsünün performansını arttırmaktadır.

Her 22,5 ms'lik PCM ses sinyalinin kodlanması sonucunda 54 bit elde edilmektedir. Bu 54 bitin 25 biti 4 adet LSF indeksline, 7 biti ötümlü-ötümsüz ayrımı ve nicemlenmiş perde periyotu indeksine, 8 biti nicemlenmiş Fourier büyüklüklerine, 5 biti nicemlenmiş birinci kazanç bilgisine, 3 biti nicemlenmiş ikinci kazanç bilgisine, 4 biti farklı frekans bantlarına ait ötümlülük bilgisine, 1 biti aperiyodikliği ifade etmeye ve 1 biti de senkronizasyon sağlanmaya ayrılmıştır.

EK 4 G.726 Konuşma Kodlayıcısı

G.726 (Anonymous 1990) ITU-T tarafından standartlaştırılan, konuşma sinyalini ADPCM teknikleri ile kodlayan ve çözen basit bir damga biçimi kodlayıcısıdır. G.726 standardı yürürlüğe girdikten sonra selefleri olan G.721 ve G.723 standartlarının yerini almıştır.

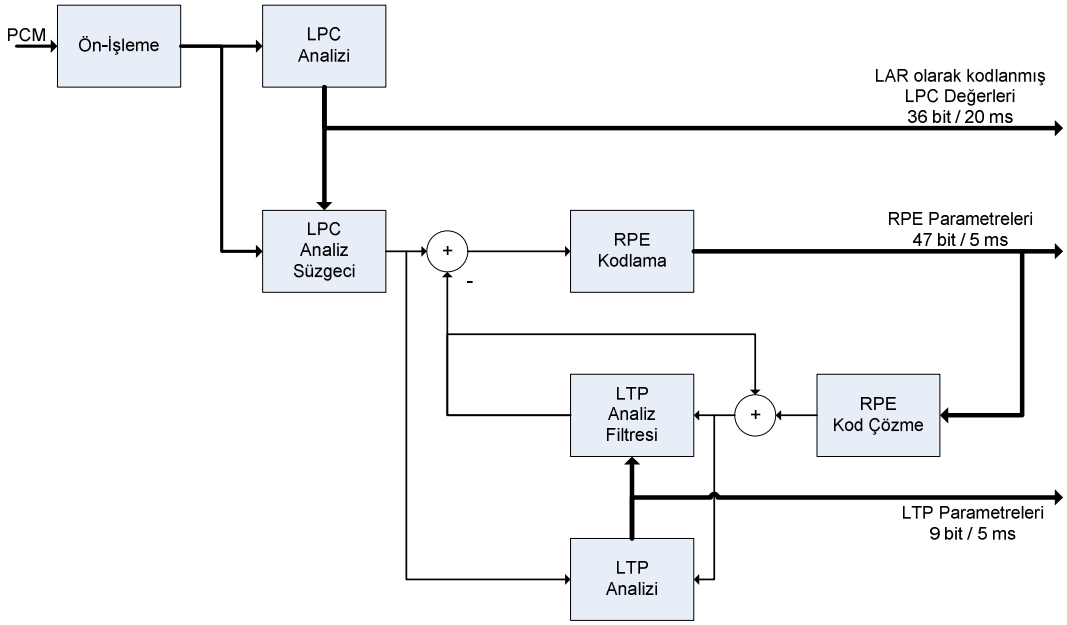
Kodlama sırasında önce PCM sinyali işlenerek fark sinyali (DPCM) elde edilir. Sonra 31, 15, 7 veya 3 seviyeli uyarlanabilir bir nicemleyiciye sokularak fark sinyali kodlanır. Uyarlanabilir nicemleyiciye ait nicemleme basamakları (*quantization steps*) kodlama sırasında gerçekleştirilen öngörüye göre sürekli güncellenir. Nicemleyicide kullanılan basamak sayısına bağlı olarak 4 farklı hızda çıktı üretilir: 40, 32, 24 ve 16 kbps.

EK 5 GSM 6.10 FR Konuşma kodlayıcısı

ETSI tarafından hazırlanmış olan GSM 6.10 (Anonymous 1999) standardı, GSM tarafından kullanılan ilk devreye alınan RPE-LTP (*Residual pulse excitation*, *long-term prediction*) tipi konuşma kodlayıcı standardıdır. "*Full Rate*" ve GSM-FR olarak da isimlendirilir ve 13 kbps hızında çıktı üretir. Ses üretim yolunu farklı genişliklerdeki silindirler olarak modeller. Modern kodlayıcılarla kıyaslandığında kalite hususunda hayli geride kalmıştır, buna rağmen halen pek çok GSM şebekesi GSM 6.10 ile ses haberleşmesi gerçekleştirmektedir. Gelecekte yürürlükten kalkacağı, yerine de EFR (geliştirilmiş tam hızlı) ve AMR (uyarlanabilir çok hızlı) standartlarına göre çalışan GSM-EFR kodlayıcısına devredeceği öngörülmektedir.

Aşağıdaki Şekil 1'de blok şeması verilen GSM 6.10 kodlayıcısı, girdi olarak 8kHz'de örneklenmiş 13 bitlik doğrusal PCM kabul etmektedir. Kodlama işlemi 20 ms'lik zaman dilimine karşılık gelen 160 örnekten oluşan çerçeveler halinde yerine getirilir. Kodlayıcının girdi sinyali karşılayan ilk işlevsel bloğu ön-işlemede ofset dengelemesi ve ön-vurgu filtrelemesi yapılır. Ofset dengelemesi, kodlama sırasındaki işlem karmaşıklığını azaltırken, ön-vurgu süzgeci girdi sinyalin yüksek frekans bileşenlerini iyileştirir.

Ön-işlemesi tamamlanan sinyal daha sonra kısa süreli LPC analizine sokulur. LPC algoritmasına göre her örnek daha önceki örneklerin doğrusal bir birleşimidir ve söz konusu analizde doğrusal birleşimi yaratan katsayılarının ne oldukları bulunmaya çalışılır. LPC katsayıları bulunduktan sonra nicemleme ve iletim hatalarına karşı çok hassas oldukları için LAR değerlerine dönüştürülür. LAR değerleri nicemlendikten sonra LPC analiz süzgecinde kullanılması amacıyla yeniden LPC değerlerine dönüştürülür. Nicemlenmiş LAR'lardan elde edilen LPC'lerle LAR'a dönüştürülen LPC değerleri birbirlerine eşit değillerdir. LPC'lerle işlemler tamamlandıktan sonra ön-işlenmiş girdi sinyali LPC analiz süzgecine sokularak tortu sinyali elde edilir.



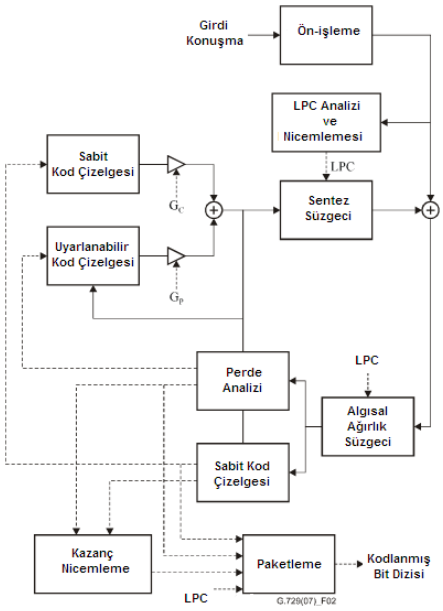
Şekil 1 GSM 6.10 analizi (Anonymous 1999)

Kısa süreli analiz sonucu çıkan tortu sinyali bu defa uzun süreli analize alınır. Uzun süreli analizde 160 örneklik çerçeve 40'ar örneklik 4 adet alt-çerçeve olarak işlenir. Uzun süreli her bir alt-çerçeve için analiz gecikme (*lag*) ve kazanç (*gain*) değerleri hesaplanır. Bir sonraki aşamada tortu sinyalinin içinden uzun süreli ilişkiler ayıklanır. Bu iş için hesaplanan gecikme ve kazanç değerlerinden oluşan LTP analiz süzgeci kullanılır. LTP analiz süzgeci çıktısı olan tortunun tortusu sinyali, tortusal darbe uyarım (RPE) sinyali olarak adlandırılır ve gürültüye benzer. 40 örnekten oluşan RPE alt-örnekleme ile üçte birine 13 örneğe indirilir. Alt-örnekleme işlemi rastgele yapılmaz; en fazla enerji içeren dizi seçilir. Bir sonraki aşamada elde edilmiş bu 13 örneğin tamamına ait bir ölçekleme faktörü (6 bitlik) bulunur. Son olarak bulunan ölçekleme faktörü yardımıyla örnekler, örnek başına 3 bit olacak şekilde teker teker nicemlenir.

EK 6 G.729 Konuşma Kodlayıcısı

ITU-T tarafından standart haline getirilmiş olan G.729 (Anonymous 2006), CS-ACELP (*conjugate structured algebraic CELP*) tipi bir konuşma kodlayıcısıdır. Söz konusu kodlayıcıda konuşma sinyali 10 ms boyundaki çerçevelere ayrıldıktan sonra işlenir. Bir çerçevede sadece iki tane 5 ms'lik alt-çerçeve tanımlanmıştır. Şekil 1'de sunulduğu üzere girdi sinyal ilk olarak ön-işlemeye alınır. Ön işleme sırasında girdi sinyal 0,5 ile çarpılarak ölçeklenir ve yüksek geçirgen süzgece sokulur. Bu işlemleri takiben ön-işlenmiş sinyalin kısa süreli analizi gerçekleştirilir ve LPC değerleri hesaplanır. Sonra LPC'ler LSF değerlerine dönüştürülür ve dönüştürülen bu değerler 2 seviyeli vektör nicemleyicisinde nicemlenir.

Bir sonraki aşamada nicemlenmemiş LPC değerleri LAR değerlerine çevrilerek algısal ağırlık süzgeci hazırlanır. Ön-işlenmiş sinyal algısal ağırlık süzgecinden geçirilerek uzun süreli analiz için kullanılacak ağırlıklı konuşma sinyali elde edilir. Söz konusu sinyalden AbS ile en az hatayı sağlayan gecikmeler, kazanç değerleri ve uyarım indeksleri bulunur.



Şekil 1 G.729 konuşma kodlayıcısı (Anonymous 2006)

EK 7 GSM 6.60 EFR Konuşma Kodlayıcısı

ETSI tarafından standart haline getirilmiş olan GSM 6.60 (Anonymous 1999), ACELP (*algebraic code excited linear prediction*) tipi bir konuşma kodlayıcısıdır. Kodlama işlemi 20 ms'lik çerçeveler üzerinde gerçekleştirilir ve 12,2 kbps hızında kodlanmış çıktı üretilir. Çerçeveleri oluşturan örnekler 13 bit olacak biçimde nicemlenir. Bir çerçevede sadece dört tane 5 ms'lik alt-çerçeve tanımlanmıştır, LSF değerlerinin hesaplandığı kısa süreli analizi takiben tüm CELP yöntemlerinde olduğu gibi AbS gerçekleştirilerek algısal ağırlıklı süzgeci çıktısından gecikme, kazanç ve uyarım indeksleri bulunur. Çalışma esasları G.729 kodlamasıyla büyük benzerlikler gösterir.

ÖZGEÇMİŞ

Adı Soyadı: Ahmet Utku YARGIÇOĞLU
Doğum Yeri: Gölcük
Doğum Tarihi: 30 Kasım 1975
Medeni Hali: Evli
Yabancı Dili: İngilizce

Eğitim Durumu (Kurum ve Yıl)

Lise: TED Ankara Koleji Vakfı Özel Lisesi, 1986-1993

Lisans: Orta Doğu Teknik Üniversitesi Elektrik-Elektronik Mühendisliği – Bilgisayar Alanı, 1993-1997

Yüksek Lisans: Orta Doğu Teknik Üniversitesi Elektrik-Elektronik Mühendisliği – Bilgisayar Alanı, 1997-2000

Çalıştığı Kurum/Kurumlar ve Yıl

Aselsan A.Ş., Haberleşme ve Bilgi Teknolojileri Grup Başkanlığı, 1997-2010

Yayınları (SCI ve diğer)

Yargıçoğlu, A.U. and İlk, H.G. 2010. Hidden data transmission in mixed excitation linear prediction coded speech using quantisation index modulation. IET Information Security, Vol 4, Issue 3, pp. 158-166. (SCI)

Yargicoglu, A.U and Ilk, H.G. 2007. MELP kodlanmış konuşma sinyalinde nicemleme indeks modülasyonu ile saklı veri iletimi. IEEE 15th Signal Processing and Communications Applications, SİU 2007, pp. 1-4.

Yargicoglu, A.U and Ilk, H.G. 2009. MELP algoritmasında nicemleme indeks modülasyonu uygulayarak gizli veri saklayan yöntemlerin kaotik tipteki özniteliklere göre steganalizi. IEEE 17th Signal Processing and Communications Applications, SİU 2009, pp. 69-72.

Yargıçoğlu, A.U, İlk, H.G. ve Kalaycıoğlu, A. 2010. Markov zincirleri ile

istatistiksel modelleme yapılarak sendrom kodlamalı gizli veri gömme, SIU 2010, IEEE 18th Signal Processing and Communications Applications, SİU 2010.