

# **TAŞINABİLİR CİHAZLAR İÇİN TÜRKÇE METİNDEN KONUŞMA SENTEZLEME SİSTEMİ**

**İLKER ÜNALDI**

Hacettepe Üniversitesi  
Lisansüstü Eğitim - Öğretim ve Sınav Yönetmeliğinin  
ELEKTRİK ve ELEKTRONİK Mühendisliği Anabilim Dalı İçin Öngördüğü  
YÜKSEK LİSANS TEZİ  
olarak hazırlanmıştır.

**2007**



# **TAŞINABİLİR CİHAZLAR İÇİN TÜRKÇE METİNDEN KONUŞMA SENTEZLEME SİSTEMİ**

## **TURKISH TEXT TO SPEECH SYNTHESIS SYSTEM FOR MOBILE DEVICES**

**İLKER ÜNALDI**

Hacettepe Üniversitesi

Lisansüstü Eğitim - Öğretim ve Sınav Yönetmeliğinin

ELEKTRİK ve ELEKTRONİK Mühendisliği Anabilim Dalı İçin Öngördüğü

YÜKSEK LİSANS TEZİ

olarak hazırlanmıştır.

2007

Fen Bilimleri Enstitüsü Müdürlüğü'ne,

Bu çalışma jürimiz tarafından ELEKTRİK VE ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI'nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Başkan : .....

Yrd. Doç. Dr. Harun Artuner

Üye : .....

Yrd. Doç. Dr. Ali Ziya Alkar

Üye : .....

Yrd. Doç. Dr. Mehmet Demirer

Üye : .....

Yrd. Doç. Dr. A. Semih Bingöl

Üye : .....

Dr. Umut Sezen (Tez Danışmanı)

ONAY

Bu tez .... / .... / .... tarihinde Enstitü Yönetim Kurulunca kabul edilmiştir.

.... / .... / ....

Prof. Dr. Erdem Yazgan

FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRÜ

*ailleme...*

# TAŞINABİLİR CİHAZLAR İÇİN TÜRKÇE METİNDEN KONUŞMA SENTEZLEME SİSTEMİ

**İlker Ünal**

## ÖZ

Taşınabilir cihazların hızla yaygınlaştığı bugünlerde, bu cihazlardaki donanımsal ve yazılımsal özelliklerin de gelişmesiyle bu cihazlar için talep edilen yazılımlarda da farklılıklar oluşmaktadır. İstenilen yazılımlardan bazıları da engelli insanların bu cihazları daha verimli olarak kullanabilmesine olanak tanıyan yazılımlardır. Son yıllarda cep telefonlarında en yaygın kullanılan hizmetlerden birisi olan kısa mesajlardan görme özürü insanlar faydalanamamaktadır. Bu çalışmada da öncelikle görme engelli insanların kısa mesaj servislerinden (SMS) faydalanmasını sağlamak amacıyla, taşınabilir cihazlarda çalışabilen bir Metinden Konuşma Sentezleme (MKS) yazılımı geliştirilmesi amaçlanmıştır.

Diğer taraftan metinden konuşma işleme sistemlerindeki Türkçe için olan çalışmalar az sayıdadır, taşınabilir cihazlar içinse bu tip bir çalışma bulunmamaktadır. Bu eksikliğin azaltılmasına yönelik olarak da sentezleme işlemi Türkçe olarak yapılmıştır.

Bu tezde mevcut MKS sistemleri ve bu sistemlerde kullanılan konuşma sentezleme yöntemleri ile taşınabilir cihazlar için var olan uygulama geliştirme platformları incelenmiştir. Bu inceleme sonucunda difonları birleştirerek sentezleyen Zaman Ekseninde Perde Senkronlu Örtüştürerek Ekleme (TD-PSOLA) yöntemi ile Java 2 Platformu, Mikro Yayımı (J2ME) platformunun kullanılmasında karar kılınmıştır. Bütün bu çalışmalar sonucunda J2ME platformunu destekleyen bir cep telefonu üzerinde çalışabilen ve cihaza gelen kısa mesajları konuşmaya çeviren bir Türkçe MKS yazılımı geliştirilmiştir.

**Anahtar Kelimeler:** Metinden Konuşma Sentezleme, MKS, PSOLA, TD-PSOLA, Taşınabilir Cihaz, J2ME, SMS.

**Danışman:** Dr. Umut Sezen, Hacettepe Üniversitesi, Elektrik ve Elektronik Mühendisliği Bölümü

# TURKISH TEXT TO SPEECH SYNTHESIS SYSTEM FOR MOBILE DEVICES

**İlker Ünal**

## **ABSTRACT**

In recent years, there has been significant improvements in both hardware and software of mobile devices and the demand for different kinds of software has emerged especially for handicapped people. Short message service (SMS) is one of the most popular features of the mobile phones, which can not be used by the blind. In this thesis, a text-to-speech (TTS) synthesis software is aimed to be developed especially to gain ability to let blind people use the SMS service.

On the other hand, there are only a few number of Turkish TTS systems developed for personal computers, and none for mobile devices. So as part of this thesis, a Turkish TTS software is developed for mobile devices.

In this thesis, existing TTS systems and mobile application development platforms are examined. As a result Time Domain Pitch Synchronous Overlap Add (TD-PSOLA) method and Java 2 Micro Edition (J2ME) platform are decided to be used. Finally, a software, which converts incoming SMS messages to speech on a J2ME supported mobile phone, is developed.

**Keywords:** Text to speech, TTS, PSOLA, TD-PSOLA, Mobile Device, J2ME, SMS

**Advisor:** Dr. Umut Sezen, Hacettepe University, Department of Electrical and Electronics Engineering

## TEŐEKKÜR

Bu alıőmada bana yaptıđı rehberlik ve katkılarından dolayı danıőmanım Dr. Umut Sezen'e, ve oluőturduđu difon veritabanını bu alıőmada kullanmama olanak sađlayan sevgili Őifa Serdar Őzen'e teőekkürlerimi sunarım.



## İÇİNDEKİLER DİZİNİ

|   |    |
|---|----|
| 1. GİRİŞ.....   | 1  |
| 2. KONUŞMANIN OLUŞUMU .....                                   | 5  |
| 3. MKS SİSTEMLERİNİN YAPISI.....                              | 10 |
| 3.1. Dil Çözümlemesi Birimi .....                             | 10 |
| 3.1.1. Metin Önışlemcisi .....                                | 11 |
| 3.1.1.1. Rakam ve Sayıların Ayrılması.....                    | 11 |
| 3.1.1.2. Kısaltmaların Ayrılması .....                        | 12 |
| 3.1.1.3. Özel İşaretlerin Ayrılması.....                      | 13 |
| 3.1.2. Fonem Seçici .....                                     | 13 |
| 3.1.3. Bürün Ayarlayıcı .....                                 | 15 |
| 3.2. Sayısal Sinyal İşleme Birimi.....                        | 16 |
| 3.2.1. Artikülatör Sentezleyiciler .....                      | 16 |
| 3.2.2. Kurallı Sentezleyiciler .....                          | 18 |
| 3.2.3. Birleştirek Sentezleme: .....                          | 21 |
| 3.2.3.1. Kullanılacak Parça Seçimi .....                      | 22 |
| 3.2.3.2. Birleştirme Sırasında Yapılan İşlemler:.....         | 23 |
| 3.2.3.2.1. PSOLA Yöntemi .....                                | 24 |
| 3.2.3.2.2. Sinüzoidal Yöntem .....                            | 27 |
| 4. TÜRKÇEDE BÜRÜN YAPISI .....                                | 28 |
| 4.1. Bürün Öğeleri.....                                       | 28 |
| 4.1.1. Süre.....  | 28 |
| 4.1.2. Durak.....   | 28 |
| 4.1.3. Ton .....  | 29 |
| 4.1.4. Ezgi .....   | 29 |
| 4.1.5. Vurgu.....   | 31 |
| 4.2. Yumuşak G .....  | 31 |
| 5. TAŞINABİLİR PLATFORMLAR .....                              | 32 |
| 5.1. J2ME.....  | 32 |
| 5.2. Symbian OS.....  | 32 |
| 5.3. J2ME Yapısı .....  | 33 |
| 5.3.1. J2ME'e genel bir bakış:.....                           | 33 |
| 5.4. J2ME Projelerinde Dikkat Edilmesi Gerekenler .....       | 34 |
| 6. GELİŞTİRİLEN UYGULAMA .....                                | 38 |
| 6.1. Birleştirme Sırasında Sinyal İşlemenin Gerekliliği ..... | 38 |
| 6.2. Dil Çözümlemesi Birimi .....                             | 41 |
| 6.3. Sayısal Sinyal İşleme Birimi.....                        | 46 |
| 6.3.1. Parça Seçimi .....                                     | 46 |
| 6.3.2. Perde Frakansı ve Uzunluk Değişimi .....               | 47 |
| 6.3.2.1. TD-PSOLA Pencereleme Bölgesi Seçimi .....            | 54 |
| 6.3.3. Enerji Seviyesinin Değişimi .....                      | 54 |
| 6.4. Kullanılan Algoritma .....                               | 55 |
| 6.5. Uygulama İçin Gerekli Java Ortamı .....                  | 60 |
| 6.6. Platform Sebebiyle Karşılaşılan Zorluklar .....          | 61 |
| 6.7. Geliştirilen Diğer Uygulamalar .....                     | 63 |

|  |    |
|--|----|
| 7. SONUÇ .....   | 64 |
| KAYNAKLAR.....   | 67 |
| SÖZLÜK.....  | 69 |
| EKLER .....  | 71 |
| EK A. Konuşma Sentezleme Sistemlerinin Kısa Tarihi ..... | 71 |
| EK A.1. Mekanik Konuşma Sentezleme Sistemleri .....      | 71 |
| EK A.2. Elektriksel Ses Sentezleyiciler .....            | 73 |
| EK B. MKS Sistemlerinin Kullanım Alanları .....          | 74 |
| EK C. Türkçede Bulunan Fonemler .....                    | 76 |
| EK D. J2ME Mimarisi.....                                 | 78 |
| EK D.1. J2ME'e genel bir bakış.....                      | 78 |
| EK D.2. Konfigürasyonlar .....                           | 79 |
| EK D.3. Sanal Makineler .....                            | 80 |
| EK.D.4. Profiller.....                                   | 81 |
| EK E. Hazırlanan Veritabanında Bulunan Difonlar.....     | 83 |
| EK F. Veritabanı Dosya Formatı .....                     | 86 |
| EK G. GSM 03.38 SMS Alfabetesi.....                      | 89 |
| ÖZGEÇMİŞ .....   | 90 |

## ŞEKİLLER DİZİNİ

|   |    |
|---|----|
| Şekil 2.1. İnsan Ses Organları [4].....   | 5  |
| Şekil 2.2. 'e' sesi için dalga şekli .....  | 7  |
| Şekil 2.3. 'l' sesi için dalga şekli.....   | 8  |
| Şekil 2.4. 'k' sesinin dalga şekli .....  | 8  |
| Şekil 3.1. MKS sisteminin işlevsel şeması [7].....  | 10 |
| Şekil 3.2. İnsan ses yolunun artikülatör modellenmesi .....   | 18 |
| Şekil 3.3. Kurallı sentezleyiciler için genel gösterim [7]. .....   | 19 |
| Şekil 3.4. Seri bağlı formant sentezleyicisinin temel yapısı .....  | 19 |
| Şekil 3.5. Paralel bağlı formant sentezleyicisinin temel yapısı .....   | 20 |
| Şekil 3.6. Laine tarafından tasarlanan formant sentezleyicisi [17].....   | 21 |
| Şekil 3.7. TD-PSOLA yöntemi ile perde frekansının değiştirilmesi [17] (a) ve (c) değişiklik yapılmak istenen sinyalleri ve pencereleri, (b) perde frekansı artırılmış sinyali, (d) ise perde frekansı azaltılmış sinyali göstermektedir.....                                | 26 |
| Şekil 5.1. J2ME uyumlu bir sistemde genel çalışma mimarisi .....  | 34 |
| Şekil 6.1. Aralarında faz farkı olan iki konuşma parçasının birbirine eklenmesi. (a) orijinal sinyali, (b) değiştirilmiş sinyali göstermektedir.....  | 39 |
| Şekil 6.2. 4500. örnekten sonra genliği %50 artırılmış konuşma sinyali. (a), orijinal sinyali, (b) değiştirilmiş sinyali göstermektedir. ....   | 39 |
| Şekil 6.3. 4500. örnekten sonra genliği %200 oranında artırılmış konuşma sinyali. (a) orijinal sinyali, (b) değiştirilmiş sinyali göstermektedir.....   | 40 |
| Şekil 6.4. Perde frekansı farklı iki konuşma parçasını birleştirilmesi .....  | 41 |
| Şekil 6.5. Veritabanında bulunan 'i2l1' difonu.....   | 48 |
| Şekil 6.6. İşlenmemiş (solda) ve TD-PSOLA yöntemiyle aynı perde frekansıyla yeniden sentezlenmiş (sağda) 'e' ünlüsünün zaman ekseninde gösterimi... 49  | 49 |
| Şekil 6.7. İşlenmemiş (solda) ve TD-PSOLA yöntemiyle aynı perde frekansıyla yeniden sentezlenmiş (sağda) 'e' ünlüsünün izgeleri.....  | 50 |
| Şekil 6.8. (a) İşlenmemiş 'e' fonemi dalga şekli ve izgesi ; (b) Perde frekansı %10 azaltılmış 'e' fonemi dalga şekli ve izgesi; (c) Perde frekansı %20 azaltılmış 'e' fonemi dalga şekli ve izgesi (d) Perde frekansı %30 azaltılmış 'e' fonemi dalga şekli ve izgesi..... | 52 |
| Şekil 6.9. (a) İşlenmemiş 'e' fonemi dalga şekli ve izgesi ; (b) Perde frekansı %10 artırılmış 'e' fonemi dalga şekli ve izgesi; (c) Perde frekansı %20 artırılmış 'e' fonemi dalga şekli ve izgesi (d) Perde frekansı %30 artırılmış 'e' fonemi dalga şekli ve izgesi..... | 53 |
| Şekil 6.10. "8 Aralık günü geliyorum." cümlesi için programın çalışma basamakları. ....   | 58 |

|   |    |
|---|----|
| Şekil 6.11. “8 Aralık günü geliyorum.” cümlesi için elde edilen dalga şekli. ....                               | 59 |
| Şekil 6.12. Uygulama için geliştirilen ve elle metin girişini sağlayan ara yüz. ....                            | 59 |
| Şekil A.1. Kratzenstein rezonatörleri .....   | 71 |
| Şekil A.2. Wheatstone tarafından yapılan, Kempelen’in konuşan makinesinin daha gelişmiş bir versiyonu [15]..... | 72 |
| Şekil D.1. J2ME uyumlu bir sistemde genel çalışma mimarisi.....   | 78 |
| Şekil D.2. Bir J2ME sisteminde tabakalar arası hiyerarşi [19] .....   | 82 |
| Şekil F.1. RIFF hücre yapısı [22]......   | 86 |
| Şekil F.2. Veritabanı dosya formatı [22]. .....   | 87 |
| Şekil F.3. ‘toc’ hücre yapısı [22]......  | 88 |
| Şekil F.4. Difon hücre yapısı [22]......  | 88 |

## ÇİZELGELER DİZİNİ

|   |    |
|---|----|
| Çizelge 2.1. Türkçedeki Ötümlü ve Ötümsüz sesler .....  | 6  |
| Çizelge 2.2. Türkçedeki Ünlü ve Ünsüz sesler .....  | 6  |
| Çizelge 2.3. Türkçedeki Süreksiz ve Sürekli Ünsüzler .....  | 7  |
| Çizelge 2.4. Türkçedeki seslerin oluşum biçimlerine göre sınıflandırılması [22].....  | 9  |
| Çizelge 3.1. FreeTTS programında kullanılan sözlükten bir alıntı. ....  | 14 |
| Çizelge 6.1. Türkçedeki harflerin milisaniye cinsinden ortalama süreleri.....   | 42 |
| Çizelge 6.2. Türkiye Türkçesindeki ünlülere ait, ortalama perde frekansı (F0),<br>birinci (F1), ikinci (F2) ve üçüncü (F3) formant değerleri. SS: Standart sapma<br>[16]..... | 43 |
| Çizelge 6.3. Sistemde kullanılan kısaltmalardan örnekler .....  | 44 |
| Çizelge 6.4. “kar” kelimesi için kullanılacak ‘a’ fonemini el ile seçtirme. ....  | 45 |
| Çizelge 6.5. Kelimelerin harf ikililerine ayrılması için örnekler .....   | 55 |
| Çizelge 6.6. Harf ikililerinde düzeltmeler .....  | 56 |
| Çizelge 6.7. Kullanılan cihazın teknik özellikleri .....  | 61 |
| Çizelge E.1. Veritabanında kullanılan difonlar [22] .....   | 85 |
| Çizelge G.1. GSM 03.38 standardına göre SMS alfabesi .....  | 89 |

## KISALTMALAR DİZİNİ

|                        |   |
|------------------------|---|
| <b>API</b>             | Uygulama Programlama Arayüzü<br>(Application Programming Interface)   |
| <b>CART</b>            | Sınıflandırma ve Regresyon Ağacı<br>(Classification And Regression Tree)  |
| <b>CDC</b>             | Bağlantılı Cihaz Konfigürasyonu<br>(Connected Device Configuration)   |
| <b>CLDC</b>            | Sınırlı Bağlantılı Cihaz Konfigürasyonu<br>(Connected Limited Device Configuration)                             |
| <b>DÇB</b>             | Dil Çözümlemesi Birimi  |
| <b>DOC<sup>1</sup></b> | Microsoft Word Dosya Biçimi   |
| <b>G/Ç</b>             | Giriş/Çıkış   |
| <b>GPS</b>             | Küresel Yer Belirleme Sistemi<br>(Global Positioning System)  |
| <b>J2ME</b>            | Java 2, Mikro Yayımı (Java 2, Micro Edition)  |
| <b>J2SE</b>            | Java 2, Standart Yayımı (Java 2, Standard Edition)  |
| <b>JSR</b>             | Java Özellik İstekleri (Java Specification Request)   |
| <b>KSY</b>             | Kişisel Sayısal Yardımcı  |
| <b>LP-PSOLA</b>        | Doğrusal Öngörümlü Perde Eşzamanlı Örtüştürerek<br>Ekleme<br>(Linear Prediction Pitch Synchronous Over Lap Add) |
| <b>LPC</b>             | Doğrusal Öngörümlü Kodlama<br>(Linear Prediction Coding)  |
| <b>MIDP</b>            | Taşınabilir Bilgilendirme Aygıtı Profili<br>(Mobile Information Device Profile)                                 |
| <b>MKS</b>             | Metinden Konuşma Sentezleme   |
| <b>MP3</b>             | MPEG-1 Audio Layer 3  |
| <b>MPEG</b>            | Moving Picture Experts Group  |
| <b>ms</b>              | Milisaniye  |
| <b>NTT</b>             | Nippon Telegraph and Telephone Corporation  |

---

<sup>1</sup> <http://www.microsoft.com/>

|                        |  |
|------------------------|--|
| <b>ODTÜ</b>            | Orta Doęu Teknik Üniversitesi  |
| <b>PDF<sup>2</sup></b> | Taşınabilir Belge Biçimi (Portable Document Format)  |
| <b>PSOLA</b>           | Perde Eşzamanlı Örtüştürerek Ekleme<br>(Pitch Synchronous Over Lap Add)  |
| <b>REL-PSOLA</b>       | Kalıntı Uyarımlı Doğrusal Öngörümlü Perde Eşzamanlı<br>Örtüştürerek Ekleme<br>(Residual Excited Linear Prediction Pitch Synchronous<br>Over Lap Add) |
| <b>RMI</b>             | Uzaktan Metod Çağırımı (Remote Method Invocation)  |
| <b>SİB</b>             | Sayısal Sinyal İşleme Birimi   |
| <b>SMS</b>             | Kısa Mesaj Servisi (Short Message Service)   |
| <b>TD-PSOLA</b>        | Zaman Ekseninde Perde Eşzamanlı Örtüştürerek<br>Ekleme<br>(Time Domain Linear Prediction Pitch Synchronous<br>Over Lap Add)                          |
| <b>TOBB</b>            | Türkiye Odalar ve Borsalar Birlięi   |
| <b>TRT</b>             | Türkiye Radyo ve Televizyon Kurumu   |
| <b>WAV</b>             | Dalgışekli Ses Dosyası Biçimi  |

---

<sup>2</sup> <http://www.adobe.com/>

## 1. GİRİŞ

Ses sentezleme işlemi her zaman bir ilgi ve araştırma alanı olmuştur, bu konudaki ilk çalışmalar yaklaşık iki yüz yıl önce mekanik cihazlar kullanarak başlamıştır [12] (Daha fazla bilgi için Ek A'ya bakınız). Teknolojik gelişmeler sonucunda bugün kişisel bilgisayarlarda işletim sistemlerinin bir parçası haline gelmiştir ve kullanımı artan cep telefonu veya Kişisel Sayısal Yardımcı (KSY) gibi taşınabilir cihazlar için de artık bir gereksinime dönüşmüştür.

Taşınabilir cihazlar gün geçtikçe yaygınlaşmışlardır ve hatta günümüzde hayatımızın bir parçası haline gelmişlerdir. En çok kullanılan taşınabilir cihazlar ise hiç şüphesiz cep telefonları olmuştur. Cep telefonlarını bu kadar cazip kılan etkenlerin başlıcaları insanların hayatında iletişimin önemli rol oynaması ile donanımların ucuzlaması, yaygınlaşması ve işlevselliklerinin artması olarak özetlenebilir.

İlk ticari hücreli ağ 1979 yılında Japon NTT şirketi tarafından kuruldu ve ilk ikinci nesil hücreli teknoloji 1991 yılında Finlandiya'da Radiolinja şirketi tarafından kullanıldı. Fakat 1990'ların başına kadar üretilen taşınabilir telefonların çoğu bir ceket cebinde taşınmayacak kadar büyüktü ve daha çok araç telefonları olarak kullanıldılar<sup>3</sup>. İlk üretilen cep telefonları şimdiki işlevselliklerinden uzaktı, sadece sesli telefon görüşmeleri için tasarlanmıştı ve çoğunda kısa mesaj alma veya gönderebilme özelliği bulunmamaktaydı. Zamanla gelişen cep telefonları şimdi çok daha küçük yapılı, ince ve hafif, sesli telefon konuşmaları yapmanın yanı sıra kısa mesaj gönderebilen, resimli ve sesli mesajlar gönderebilen, müzik dosyalarını okuyabilen, fotoğraf çekebilen ve video kaydedebilen, birçok türde dosya formatını tanıyabilen, dosyaları açıp düzenleyebilen, internete bağlanabilen, Küresel Yer Belirleme Sistemi (GPS) aracılığıyla dünya üzerindeki konumunuzu belirleyebilen, yakın çevrenizin haritasını bize ekranında gösterebilen cihazlar haline geldiler<sup>4</sup>. Bu değişimde telefonda kullanılan elektronik bileşenlerin boyutlarının küçülmesi etkili oldu.

Yukarıda sayılan değişikliklerin büyük ölçüde donanımsal ve yazılımsal gelişmelere de bağlı olduğu açıktır. İlk cep telefonlarında şimdikilerden çok daha

<sup>3</sup> [http://en.wikipedia.org/wiki/Mobile\\_phone](http://en.wikipedia.org/wiki/Mobile_phone)

<sup>4</sup> Nokia N95, bu tip bir telefona örnek olarak verilebilir



kısıtlı hafıza, daha yavaş ve az gelişmiş işlemciler, renksiz ve küçük ekranlar ve yeni uygulamaların cihaza yüklenmesine izin vermeyen ve sistem donanımını yeni yüklenen uygulamalara erişirmeyen yazılımlar bulunmaktaydı. Fakat artık yeni cihazlar daha gelişmiş işlemci ve bellek sistemlerine, daha gelişmiş hoparlörlere; ve ileri teknolojileri destekleyen, yeni uygulamaların cihaza yüklenmesine ve cihazın donanımına erişmesine olanak sağlayan yazılımlara (çoğu zaman işletim sistemlerine) sahip oldular<sup>5</sup>.

Yukarıdaki gelişmelerin ışığında artık taşınabilir cihazlar için de değişik yazılımlar talep edilmeye ve üretilmeye başlandı. Taşınabilir platformlar için yazılım geliştiren şirketler ve bir mobil yazılım piyasası oluştu. İnsanlar farklı ihtiyaçlar için farklı yazılımlar talep eder veya geliştirir duruma geldiler. Bu konudaki isteklerden bazıları da insan – makine ara yüzünü daha da iyi duruma getirmek üzere tasarlanan yazılımlar oldu. Günümüzde çoğu cep telefonunda arama yapmak için sadece aranacak kişinin adının söylenmesi yeterlidir<sup>6</sup> ve bazı cep telefonlarında ise entegre ses sentezleme sistemi sayesinde arayan kişinin adı da telaffuz edilmektedir<sup>7</sup>. Yalnız taşınabilir sistemlerde hâlihazırda Türkçe için bir Metinden Konuşma Sentezleme (MKS) desteği veya ayrı bir yazılım bulunmamaktadır.

Cep telefonlarında en yaygın kullanılan işlevselliklerden birisi de kısa mesaj servsidir (SMS). Kısa mesajların sesli konuşmaya oranla daha ucuz olması, aynı anda bir çok kişiye ulaşabilmesi, gönderilen kişinin anında ulaşılabilir olması zorunluluğu olmaması, mesajların saklanabilmesi gibi birçok nedenden dolayı kısa mesajlar popülerliklerini hep korudular. Ayrıca birçok haber, bilgi ve eğlence servisleri de cep telefonu operatörleri tarafından hizmete sunuldu.

Yukarıda sayılan etkenler doğrultusunda cep telefonlarında gelen kısa mesajların da okunabildiği Türkçe için bir MKS yazılımının da gerekliliği ortaya çıkmıştır. Böyle bir yazılım sayesinde öncelikle görme engelli insanlar kısa mesaj servisini kullanabilecek ve ayrıca diğer insanlardan veya operatörden gelen bilgi servisi gibi

---

<sup>5</sup> Cep telefonlarının teknik ayrıntı bilgileri için;

Nokia marka telefonlar için

SonyEricsson marka telefonlar için

Samsung marka telefonlar için

: <http://www.forum.nokia.com>

: <http://developer.sonyericsson.com>

: <http://developer.samsungmobile.com>

internet adreslerine bakılabilir.

<sup>6</sup> Örneğin, Nokia N73

<sup>7</sup> Örneğin, Nokia 5500 Sport

kimi kısa mesajları sesli olarak dinleyebileceklerdir. Ayrıca, bir şekilde taşınabilir cihazını kullanamayacak durumda olan bir kullanıcı (örneğin spor yapan veya elleri dolu olan birisi) da gelen kısa mesajları konuşma olarak dinlemek isteyebilecektir. Diğer taraftan, taşınabilir cihazlar günümüzde değişik dokümanları tanıyabilen cihazlar durumuna geldiler. Bu cihazlarda bulunan Microsoft Word Dosya Biçimi (DOC) ve Taşınabilir Belge Biçimi (PDF) dokümanlarının konuşmaya çevrilmesi de kullanıcı tarafından istenilebilir<sup>8</sup>.

Son yıllarda hız kazanan çalışmalara rağmen, Türkçe için yeterli miktarda ve nitelikte metinden konuşma sentezleme (MKS) sistemleri hala geliştirilebilmiş değildir. Taşınabilir platformlar için ise bu konuda gerçekleştirilmiş hiçbir çalışma bulunmamaktadır. Bu tez kapsamında yapılacak çalışma ile, böyle bir ihtiyacı karşılamak amacıyla Türkçe için taşınabilir platformlarda çalışabilecek bir MKS yazılımı geliştirilmesi amaçlanmaktadır.

Bu tez kapsamında geliştirilen MKS sisteminin, herhangi bir Türkçe metni sentezlemenin dışında, ilk amacı görme engelli insanların veya bazı durumlarda telefonlarına gelen mesajlara bakamayacak durumda olan kullanıcıların (fiziksel engelli olup cep telefonunu kullanmakta zorluk çeken bir kullanıcı veya spor yapan bir kullanıcı) cep telefonlarının kısa mesaj servisini kullanmasına olanak sağlamaktır. Böylece hem diğer insanlarla mesajlaşmaları, hem de kısa mesaj üzerinden verilen birtakım operatör servislerinden (haber, eğlence vs.) faydalanmaları sağlanmış olacaktır (MKS sistemlerinin diğer olası kullanım alanları için Ek B'ye bakılabilir).

Bu tezde, ilk önce Bölüm 2'de insanlarda konuşmanın nasıl oluştuğu ve temel olarak nasıl modellendiği üzerinde durulacaktır. Bölüm 3'te, MKS sisteminin yapısı ve ne gibi işlemler yapması gerektiği incelenecektir. Yine aynı bölümde, kullanılan MKS yöntemleri ve kullanım alanları incelenecek ve taşınabilir cihazlar için en uygun olan yöntem belirlenecektir. Sonrasında Bölüm 4'te Türkçedeki bürün yapısı anlatılacaktır. Bölüm 5'te ise taşınabilir cihaz platformları incelenecek ve MKS uygulaması geliştirmek için uygun bir platform ve donanım seçilecektir. Daha sonra Bölüm 6'da ise geliştirilen yazılımın temel olarak yapısı ve işlevselliği

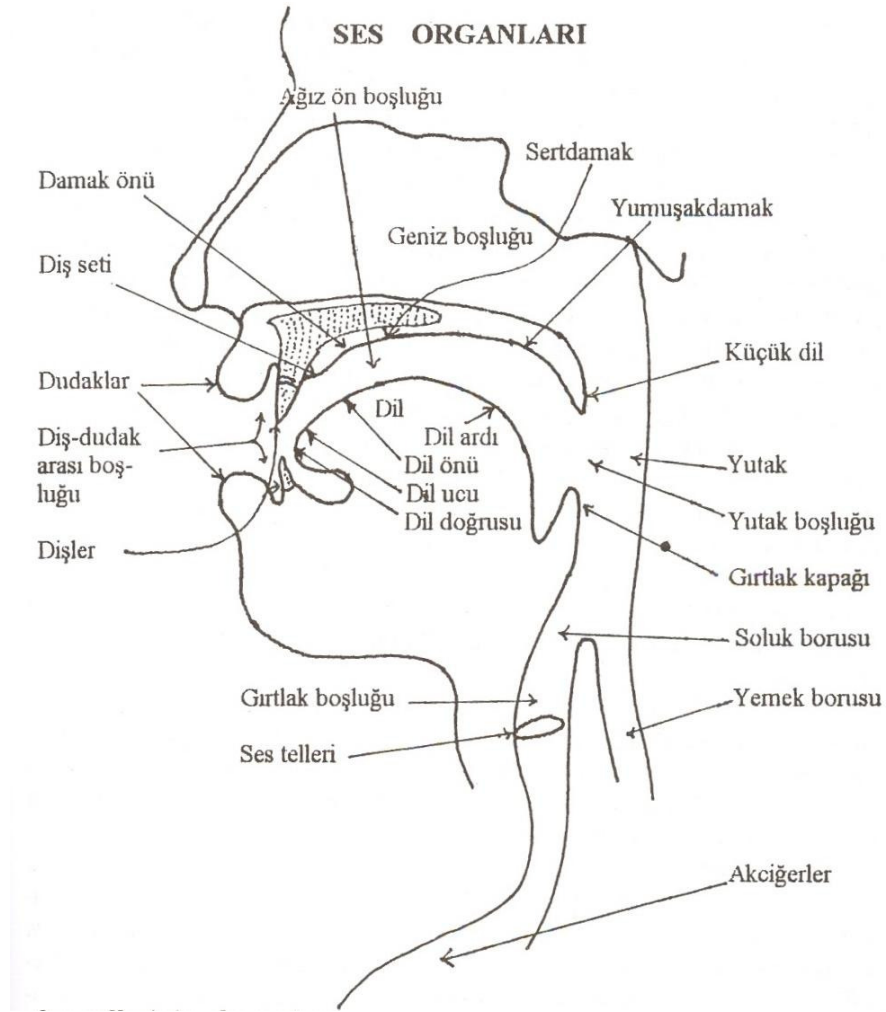
---

<sup>8</sup> Adobe Acrobat Reader programının 8. sürümünün Windows versiyonunda metinden konuşma sentezleme yetenekleri bulunmaktadır.

anlatılacaktır. Son olarak Bölüm 7'de de bu çalışmaların sonuçları değerlendirilecek ve ileriye dönük çalışmalar hakkında öneriler ortaya konulacaktır.

## 2. KONUŞMANIN OLUŞUMU

İnsanda konuşmayı oluşturan birtakım organlar vardır. Bu organların asıl görevleri, konuşma oluşturmaktan başka görevlerdir, örneğin akciğerlerin görevi solunum yoluyla kan temizlemek, dilin görevi ise tat almak ve yenenleri yutmaya yardım etmektir [4]. Sözel anlatımın seslendirilmesi, bu organlar için ikinci bir görevdir. Şekil 2.1'de insan ses organları görülmektedir.



Şekil 2.1. İnsan Ses Organları [4]

Konuşmanın üretilmesi için akciğerlerden havanın ağız bölgesine doğru itilmesi gerekmektedir. Havanın itilmesi, kaburgalar arası kaslar ve karın kaslarınınca

sağlanır. Bu kaslar, değişik sesler çıkarmak için (hece veya vurgulu sesler gibi) değişik şeklide kasılmalar yaparlar.

Ses telleri, gırtlığın içinde bulunur ve bir çift kas ya da esnek uzantıdan oluşur. Konuşurken ses telleri kapanır, ciğerlerdeki hava basıncı %1 oranında artar, ses telleri açılır, hava çıkışı olur, kısa bir süre sonra ses tellerinin iki tarafındaki hava basıncı eşitlenir ve ses telleri yine kapanır. Bu döngü devam eder ve böylece ses telleri titreşimi oluşur. Bu titreşime perde frekansı veya titreşim frekansı denir. Bu frekans ses tellerinin kalınlığına, gerginliğine ve uzunluğuna ve de ciğerlerden gelen havanın basıncına bağlıdır. Konuşurken insan, ses tellerinin uzunluğunu, gerginliğini ve ciğerlerden gelen hava basıncını ayarlayarak titreşim frekansını değiştirebilir. Değişken olmakla birlikte, bu frekansın varsayılan değeri erkekler için 110 Hz, kadınlar için 200 Hz, çocuklar için 300 Hz civarındadır [15].

Titreşen ses telleri, sedalı (ötümlü) seslerin üretilmesine olanak sağlar. Ses telleri açık tutulup titreştirilmediğinde ise, ses yolunun ilerisindeki organlar ile sedasız (ötümsüz) sesler oluşturulabilir. Ötümlü ve ötümsüz sesler Çizelge 2.1'de gösterilmiştir.

|                       |  |
|-----------------------|--|
| <b>Ötümlü Sesler</b>  | 'a', 'e', 'ı', 'i', 'o', 'ö', 'u', 'ü', 'b', 'c', 'd', 'g', 'j', 'l', 'm', 'n', 'r', 'v', 'y', 'z' |
| <b>Ötümsüz Sesler</b> | 'p', 'ç', 't', 'k', 'f', 'h', 's', 'ş'   |

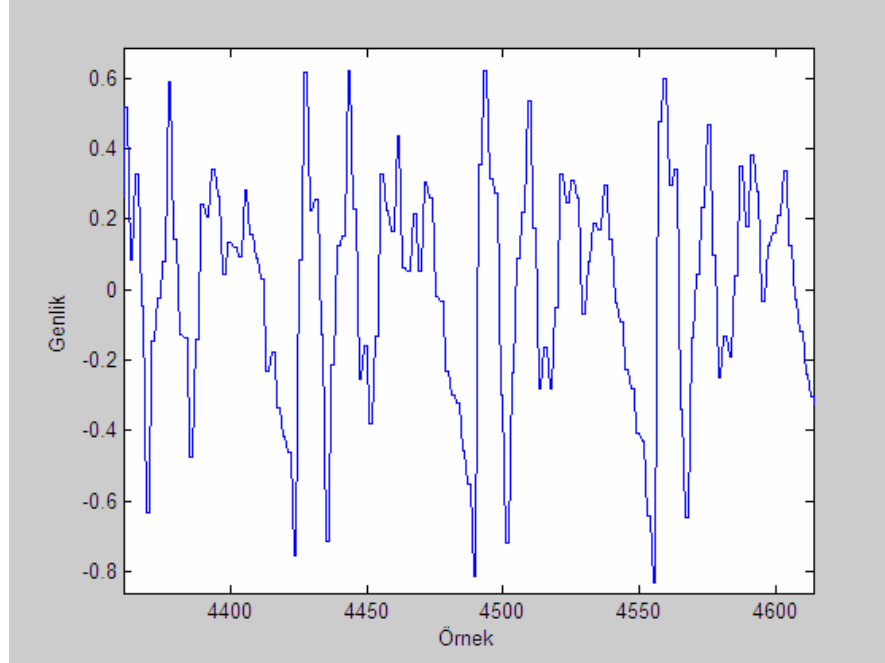
Çizelge 2.1. Türkçedeki Ötümlü ve Ötümsüz sesler

Ses organlarının, hava akımının takıntıya uğramamasını sağlayacak şekilde hareket etmesiyle ünlü seslerin oluşması sağlanır. Eğer ses organları hava akımını tamamen veya kısmen engeller, hava akımının takıntıya uğramasına sebep olursa ünsüz sesler oluşur. Türkçede bulunan ünlü ve ünsüz harfler Çizelge 2.2'de gösterilmiştir.

|                     |  |
|---------------------|--|
| <b>Ünlü Sesler</b>  | 'a', 'e', 'ı', 'i', 'o', 'ö', 'u', 'ü'   |
| <b>Ünsüz Sesler</b> | 'b', 'c', 'd', 'g', 'j', 'l', 'm', 'n', 'r', 'v', 'y', 'z', 'p', 'ç', 't', 'k', 'f', 'h', 's', 'ş' |

Çizelge 2.2. Türkçedeki Ünlü ve Ünsüz sesler

Ünlü harfler oluşurken ses tellerinde oluşan hava akımı ses yolunda engellemeye uğramadığı için, ünlülerin dalga şekli periyodikliğe oldukça yakındır. 'e' sesinin dalga şekli Şekil 2.2'de gösterilmiştir.



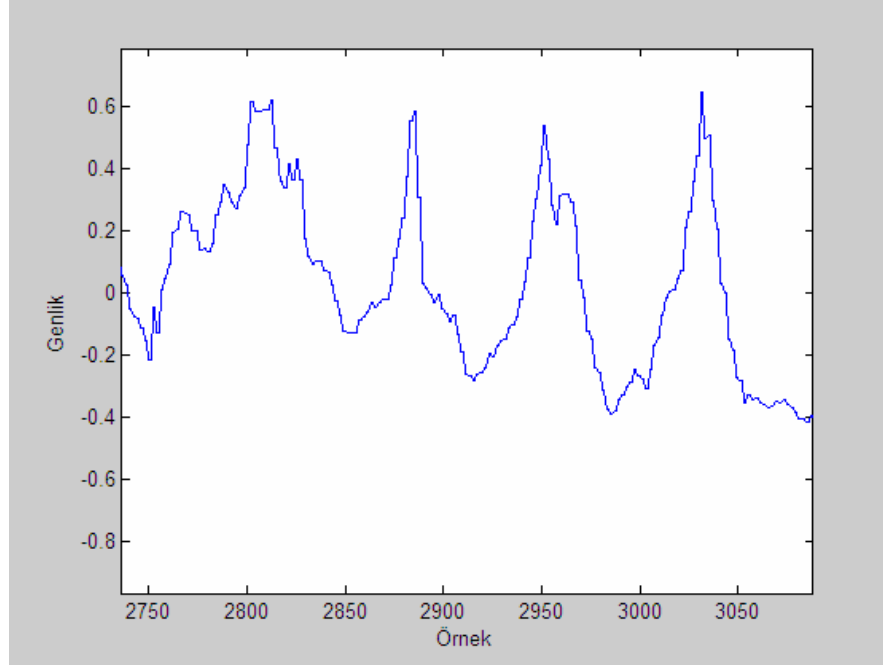
Şekil 2.2. 'e' sesi için dalga şekli

Eğer ünsüz sesler oluşurken yapılan engelleme tam olur, hava akımının ilerlemesi bir süreliğine tamamen engellenmiş olursa patlamalı (süreksiz) ünsüzler oluşur. Eğer bu engelleme tam olmaz, hava akımı için bir miktar açıklık kalırsa sürekli ünsüzler oluşur. Sürekli ve süreksiz ünsüzler Çizelge 2.3'te verilmiştir.

|                          |  |
|--------------------------|--|
| <b>Süreksiz Ünsüzler</b> | 'b', 'c', 'd', 'g', 'p', 'ç', 't', 'k'                     |
| <b>Sürekli Ünsüzler</b>  | 'f', 'h', 'j', 'l', 'm', 'n', 'r', 's', 'ş', 'v', 'y', 'z' |

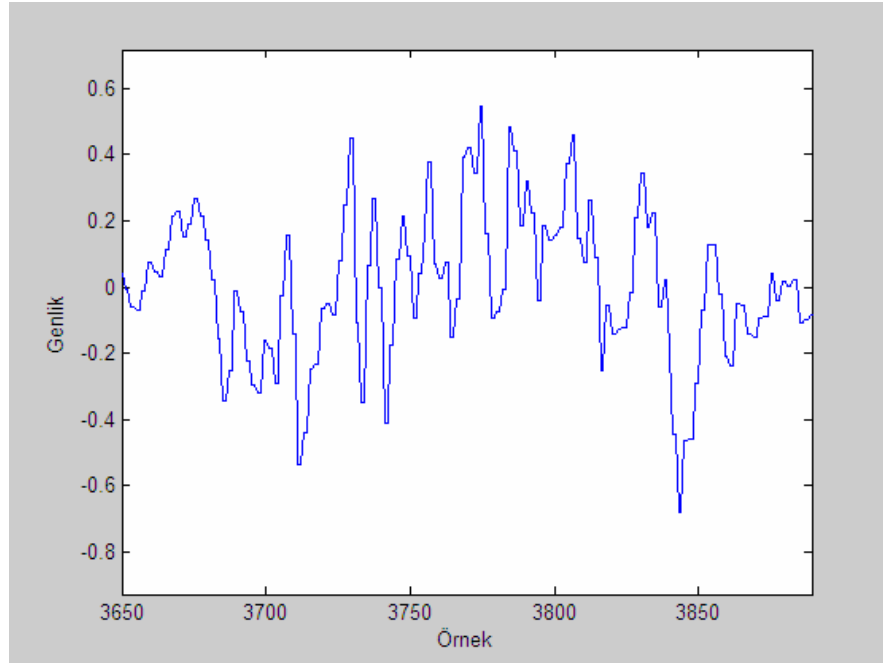
Çizelge 2.3. Türkçedeki Süreksiz ve Sürekli Ünsüzler

Ötümlü ünsüzler, ünlüler kadar olmasa da yine de az çok periyodik bir dalga şekline sahip olurlar. Şekil 2.3'de 'l' sesi için dalga şekli gösterilmiştir.



Şekil 2.3. 'l' sesi için dalga şekli

Ötümsüz ünsüzler oluşurlarken hava akımı, aldığı yol sırasında bir yerlere çarpıp, sürtünüp engellendiği için düzenli bir dalga şekline sahip değildir. Şekil 2.4'de 'k' sesi için dalga şekli gösterilmiştir.



Şekil 2.4. 'k' sesinin dalga şekli

Yukarıda anlatılan sınıflandırmalar Çizelge 2.4 ile özetlenebilir [22]. Yukarıdaki çizelgelerde 'ğ' harfinin bulunmaması dikkat çekebilir. 'ğ' bir ses ifade etmediği için bu çizelgelere eklenmemiştir. Bu konu Bölüm 4'te daha ayrıntılı olarak incelenmiştir.

| Ötümlü Sesler             |                           |                   | Ötümsüz Sesler   |                   |
|---------------------------|---------------------------|-------------------|------------------|-------------------|
| Ünlüler                   | Sürekli Ünsüzler          | Süreksiz Ünsüzler | Sürekli Ünsüzler | Süreksiz Ünsüzler |
| a, e, ı, i,<br>o, ö, u, ü | j, l, m, n,<br>r, v, y, z | b, c, d, g        | f, h, s, ş       | p, ç, t, k        |

Çizelge 2.4. Türkçedeki seslerin oluşum biçimlerine göre sınıflandırılması [22]

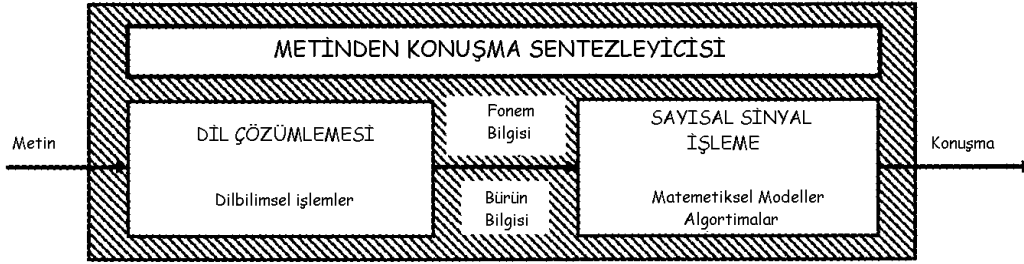


### 3. MKS SİSTEMLERİNİN YAPISI

Metinden konuşma sentezleme (MKS), yazılı bir metnin akustik ortama taşınması, sesli hale getirilmesidir [22]. Metnin sisteme girişi, operatör aracılığıyla veya doğrudan yazının sisteme gösterilmesiyle veya optik karakter tanıma ile olabilir [7]. Konuşma doğal ve anlaşılır olmalıdır. MKS sistemlerinin diğer konuşabilen sistemlerden farkı, karşılaşılan yeni kelimeler için de bir konuşma sentezleyebilmeleridir. Buna karşılık, örneğin bir bankanın telefonla bankacılık hizmetindeki konuşan sistemler sadece önceden kayıtlı konuşmaları dinleyiciye aktarabilmekte fakat sözlüklerinde bulunmayan kelimeler için yeni bir konuşma sentezleyememektedirler [5].

MKS sistemlerinde sentezlenen konuşmanın doğallığı, üç temel unsura bağlıdır. Bu unsurlar konuşmada kullanılan sedalı seslerin perde frekansı, kullanılan fonemlerin enerjileri ve kullanılan fonemlerin süreleri olarak sıralanır [20].

Bu üç unsurun hangi değerlerle sağlanacağı ve konuşmanın fiziksel olarak nerede sentezleneceği iki ayrı alt birim kullanılarak gerçekleştirilir, bunlar Dil Çözümlemesi Birimi ile Sayısal Sinyal İşleme Birimidir. Bu birimlerin birbiriyle olan işlevsel ilişkisi de Şekil 3.1'de gösterilmiştir.



Şekil 3.1. MKS sisteminin işlevsel şeması [7]

#### 3.1. Dil Çözümlemesi Birimi

Dil Çözümlemesi Birimi (DÇB), sisteme girdi olarak verilen metni, gerekli işlemleri yaparak, Sayısal Sinyal İşleme Biriminin (SİB) kullanabileceği veriyi üreten parçadır. Günümüzde MKS sistemlerinde metin işlenirken, sayıların ve

kısaltmaların işlenmesi ve doğru bürün değerlerinin seçilmesi ana problemlerdendir [17].

Dil Çözümlemesi Birimi, yaptığı işler bazında üç ana parçaya ayrılıp, bu parçalar Metin Önişlemcisi, Fonem Seçici ve Bürün Ayarlayıcı olarak adlandırılabilir. Sisteme girilen metin ilk önce Metin Önişlemcisine verilir. Bu parça, metnin içindeki rakamlar, sayılar, özel karakterler gibi unsurları doğrudan okunabilecek, sadece harflerle yazılmış bir metne çevirir. Fonem Seçici, Metin Önişlemcisinden gelen okunamayan karakterlerden ayıklanmış yazıyı işler ve hangi fonemlerin kullanılması gerektiğine karar verir. Bürün Ayarlayıcı da fonemlerin birleştirilerek konuşmaya dönüştürülmesi sırasında Sayısal Sinyal İşleme Biriminin kullanacağı bürün değerlerini (fonem uzunlukları, perde frekansı ve enerji eğrileri) tespit eder.

### **3.1.1. Metin Önişlemcisi**

MKS sistemlerinin ideal olarak herhangi bir yazıyı okuması beklenir. Bu durumda sisteme girilen yazının içinde sadece harflerle oluşturulmuş, doğrudan seslendirilecek kelimeler değil, başka formatta yazılar da olabilir. Bu durumda bir metin önişlemcisine ihtiyaç duyulur. Bu gibi dönüşüm gerektiren metin parçalarını rakamlar ve sayılar, kısaltmalar ve özel işaretler olmak üzere üç ana bölümde inceleyebiliriz.

#### **3.1.1.1. Rakam ve Sayıların Ayrılması**

Rakamlar ve sayılar düz yazı haline getirilmelidirler. Örneğin “1983” sayısı “bin dokuz yüz seksen üç” kelimelerine dönüştürülür ve sistemdeki sonraki birimler bu kelimelerden konuşma sentezleyebilirler. Fakat rakamların ve sayıların okunması birçok durumda önceki örnektekinden daha karmaşık olur. Örneğin, bir telefon markasından ve tipinden bahsederken “Nokia 6630” her zaman için “nokya altmış altı otuz” şeklinde okunur, “nokya altı bin altı yüz otuz” şeklinde okunmaz. Bir DÇB'nin bu gibi ayrımları algılaması ve uygun bir şekilde rakam ve sayıların kelime karşılıklarını bulması beklenir.

Bazı sayılar, sıradan bir sayıyı ifade etmek yerine özel bir anlam taşıyabilirler. Örneğin telefon numaralarının özel bir okunuş tarzı vardır. Bir metinde geçen “2976865” rakamlar topluluğu, hem bir sayıyı, hem de bir telefon numarasını ifade ediyor olabilir. Eğer “2976865 piriç tanesinin ağırlığı tam bir kilogram

gelmektedir.” şeklinde bir cümle varsa, bu rakamlar topluluğu “iki milyon dokuz yüz yetmiş altı bin sekiz yüz altmış beş” şeklinde kelimelere dönüştürülmelidir. “Hacettepe Üniversitesi Fen Bilimleri enstitüsünün telefon numarası 2976865’dir.” şeklinde bir cümlede ise bu rakamlar topluluğu “iki yüz doksan yedi altmış sekiz altmış beş” şeklinde kelimelerle ifade edilmelidir.

Romen rakamlarıyla yazılmış olan sayılarda harfler kullanılır, fakat bunların okunuşu harflerin okunmasıyla olmaz, “VIII” rakamı “viii” şeklinde okunmamalıdır, “sekiz” kelimesine dönüştürülmelidir. Ayrıca bazı Romen rakamlarının, birtakım kısaltmalarla karıştırılmaması gerekmektedir. Örneğin MCM, hem Romen rakamlarıyla “1900” anlamına gelmektedir, hem de bir televizyon kanalının adıdır.

Sıra belirten rakamların okunmasında noktaya dikkat edilmelidir. “2.” yazısında geçen nokta “ikinci” anlamına geliyor olabilir, veya devrik bir cümlenin son karakteri olan “2”den sonra gelen nokta cümle bitimini işaret ediyor olabilir. Yine DÇB bu durumları ayırt etmekle yükümlüdür.

### **3.1.1.2. Kısaltmaların Ayrılması**

Kısaltmaların okunması da dikkat isteyen bir iştir. Kısaltmalar, açılımlarına dönüştürülebilir, harf harf okunabilir, kısaltmanın kendisi bir kelimeymiş gibi okunabilir veya kısaltma bambaşka bir şekilde okunabilir. Türkçede, “H.Ü.” ve “B.Ü.” kısaltmaları her zaman için açılımlarına dönüştürülür, “Hacettepe Üniversitesi” ve “Boğaziçi Üniversitesi” şeklinde okunur, fakat “ODTÜ” çoğu zaman “odtü” şeklinde okunur. “N.A.T.O.” ise hiçbir zaman açılımıyla okunmaz, kendisi bir kelimeymiş gibi davranılır ve “nato” şeklinde telaffuz edilir. “T.R.T.” ise harf harf, yani “te re te” şeklinde okunur. “MPEG”, “MP3” ve “T.O.B.B.” gibi bazı kelimeler ise “empeg”, “em pe üç” ve “tob” şeklinde düzensiz bir şekilde okunur. Kısaltmanın nasıl okunması gerektiğine DÇB karar verir.

Bazı kısaltmalar birden fazla anlama gelebilir. “H.Ü.” kısaltması, bazı metinlerde “Hacettepe Üniversitesi”, bazı metinlerde de “Harran Üniversitesi” veya “Haliç Üniversitesi” anlamına gelebilir. Benzer şekilde “M.S.” kısaltması, “Milattan Sonra” anlamında kullanılabileceği gibi bir tıp terimi (bir hastalık ismi) olan “Multiple Skleroz” anlamına da gelebilir. Bu gibi durumlarda DÇB’nin hangi kısaltmayı kullanması gerektiğine karar vermesi gerekmektedir.

### 3.1.1.3. Özel İşaretlerin Ayrılması

'%', '/', '-', '+' gibi karakterler de okunurken yazıda farklı anlamlar yüklenip farklı şeklide okunabilirler. Örneğin "1/4" ifadesi matematik ile ilgili bir metinde "bir bölü dört" şeklinde okunması gerekirken, bir adres ifadesinde kullanılırsa "bir taksim dört" şeklinde okunmalıdır. Fakat bir tarih yazımı söz konusuysen "/" karakteri okunmaz. "08/02/1983" tarihi "sekiz şubat bin dokuz yüz seksen üç" şeklinde okunur. "/" burada karakterinin görevi ay, gün ve yıl rakamlarını yazıda birbirinden ayırmaktır, "/" karakterinin kendisi konuşmaya dönüşmez. Benzer şekilde "3-1" eğer matematik ile ilgili bir metinde geçiyorsa "üç eksi bir" şeklinde okunuyorken, bu bir spor karşılaşmasının sonucunu belirtiyorsa "üç bir" şeklinde okunmalıdır.

### 3.1.2. Fonem Seçici

Sisteme girilen karmaşık metnin sadece karakterlerden oluşan bir metin haline çevrilmesi işleminden sonra kullanılacak olan fonemlerin seçilmesi gerekmektedir. [17] Birçok yabancı dilde daha karmaşık olan bu işlem, Türkçenin büyük ölçüde okunduğu gibi yazılan bir dil olması dolayısıyla Türkçede göreceli olarak daha kolaydır.

Fonem Seçici, girilen kelimelerin uygun telaffuz karşılığını bulmakla yükümlüdür [17]. Türkçe büyük ölçüde konuştuğu gibi yazılan bir dil olduğu için genellikle harflere karşılık gelen fonemler doğrudan kullanılabilir. Fakat Türkçedeki fonem sayısı, harf sayısından daha fazladır, yani bazı harflere birden fazla fonem düşmektedir [11]. Bu durumda, birçok harf farklı durumlarda farklı fonemlerle seslendirilir. Örnek olarak aşağıdaki iki kelimenin fonetik yazılışlarını verebiliriz:

|        |           |
|--------|-----------|
| kasaba | : kɑ'saba |
| keyif  | : ce'jlf  |

Burada görülen kelimelerin ilk harfi olan "k" harfi iki kelimedede farklı bir şekilde, yani iki farklı fonemle okunmaktadır. Hangi "k" harfine karşılık hangi fonemin kullanılacağına DÇB karar verir. Türkçede var olan fonemler Ek C'de sunulmuştur.

Bu noktada karar vermek için istatistiksel verilerden veya bir sözlükten faydalanılabilir. Örneğin, Türkçede kelime başındaki "k" harfinden sonra gelen sesli harf, ince bir sesli harf ise "c", kalın bir sesli harf ise "k" foneminin kullanıldığı

Ergenç'in çalışmasında gözlemlenmektedir [11]. Bu gibi istatistiksel veriler kural haline getirilip DÇB'ne eklenebilir. İngilizce gibi yazıldığı gibi okunmayan dillerde ise harflerin kelime içindeki yerleri ve dizilişlerine göre bazı algoritmalar geliştirilmiş ve bu algoritmalar aracılığıyla hangi fonemlerin kullanılacağına karar verilmektedir. Bazı sistemlerde ise bu iş için bir sözlük kullanılmaktadır. Örneğin, FreeTTS<sup>9</sup> programında kelimelerin fonem karşılıklarının bulunması için bir sözlük kullanılmaktadır. Bu sözlükten bir alıntı Çizelge 3.1'de verilmiştir.

|              |                     |
|--------------|---------------------|
| ...          |                     |
| shot0        | sh aa1 t            |
| shotgun0     | sh aa1 t g ah1 n    |
| shotguns0    | sh aa1 t g ah1 n z  |
| shots0       | sh aa1 t s          |
| shott0       | sh aa1 t            |
| shotts0      | sh aa1 t s          |
| shotwell0    | sh aa1 t w eh1 l    |
| shougang0    | sh aw1 g aa1 ng     |
| shough0      | sh aw1              |
| should0      | sh uh1 d            |
| shoulder0    | sh ow1 l d er       |
| shouldered0  | sh ow1 l d er d     |
| shouldering0 | sh ow1 l d er ih ng |
| shoulders0   | sh ow1 l d er z     |
| shouldntv    | sh uh1 d n t        |
| shoults0     | sh ow1 l t s        |
| shoultz0     | sh ow1 l t s        |
| shoumaker0   | sh uw1 m ey1 k er   |
| shoun0       | sh aw1 n            |
| shoup0       | sh uw1 p            |
| ...          |                     |

Çizelge 3.1. FreeTTS programında kullanılan sözlükten bir alıntı.

FreeTTS programında kullanılan sözlükte yaklaşık 105.000 kadar kelime bulunmaktadır. Program, bir sözcükle karşılaştığı zaman, bu sözlük içinde, o sözcüğün karşılığı olacak telaffuzu (fonemleri) aramakta ve bulursa kullanmaktadır. Eğer bu sözlükte bulamazsa, o zaman program bir algoritma ile sözcüğün telaffuzunu tahmin etmeye çalışmaktadır.

<sup>9</sup> <http://sourceforge.net/projects/freetts>

Bu sözlüğün boyutu ve içereceği kelime sayısı da karar verilmesi gereken bir özelliktir. Elbette olası bütün kelimeleri böyle bir sözlüğe koymak, veritabanı boyutunu çok artıracığı için neredeyse imkânsızdır, onun için bazı kelimeler sözlük dışı kalabilir. Bu sınırlamada fikir vermek açısından belirtmek gerekirse, FreeTTS programındaki sözlükte “İstanbul” ve “Ankara” kelimelerinin telaffuzu bulunmakta, fakat Türkiye’den başka bir büyük şehrin telaffuzu bulunmamaktadır.

Türkçede de yazıldığı gibi okunmayan bazı kelimeler bulunmaktadır. Türkçede bazı sesler, kendilerinden önceki veya sonraki seslerin söylenişinde farklılığa yol açmaktadır. Örneğin “saklayalım” kelimesi “saklıyalım” şeklinde okunur, “gelmezse” kelimesi “gelmesse” şeklinde okunur [11]. Bu tip farklılıklar da bir sözlük kullanımı veya belirli kurallar yardımı ile bulunup düzeltilerek daha doğal bir konuşma sentezlenebilir.

Bunların dışında, yabancı özel isimlerin okunuşu da problem yaratabilir. Birçok Türkçe metinde Türkçe olmayan ve Türkçe düzeninde okunmayan kelimeler bulunmaktadır. “New York” kelimesi bu şekilde yazılmakta “niv york” şeklinde okunmaktadır.

### **3.1.3. Bürün Ayarlayıcı**

Ses sinyalindeki süre, durak, ton, ezgi ve vurgu ile ilgili parçalarüstü birimler bürün olarak tanımlanmaktadır. Konuşmada geçen fonem süreleri, ezgi, vurgu ve durakların doğru değerler ile ifade edilmesi konuşmanın doğallığı için önemlidir (bu konu daha ayrıntılı bir biçimde Bölüm 4’te incelenecektir). Bürün Ayarlayıcı bu değerleri bulur ve sayısal sinyal işleyicisinin anlayabileceği sembollere çevirir. Bürün değerleri, konuşmanın doğal ve anlaşılabilir olmasını sağlayacak şekilde seçilmelidir. Bürün değerleri aynı zamanda konuşmacının içinde bulunduğu ruh hali ve duygularının konuşmaya yansımalarını da sağlamaktadır. Yazılı metinden konuşmacının bu özellikleri saptanamasa da, MKS sistemine bu konuda bazı ön değerler verilerek konuşmayı istenen şekilde sentezletmesi sağlanabilir. Bugünkü bazı konuşma sentezleme sistemlerinde konuşma sentezlenirken normal, heyecanlı, üzgün, bağırma gibi bazı özel şekillerde konuşma sentezletmek mümkündür<sup>10</sup>.

---

<sup>10</sup> <http://emofilt.sourceforge.net/>

Bunun dışında, bazı kelimeler için kullanılan bürün değerleri kelimenin anlamını bile değiştirebilir. Örneğin, “görüşme” kelimesi, söylenme şekline göre “mülakat” anlamına gelebileceği gibi, görüşülmemesini isteyen (emreden) bir anlama da gelebilir. DÇB'nin yazıda bu kelimenin yazıda hangi şekilde kullanıldığını algılayıp uygun bürün değerlerini oluşturup bunu Sayısal Sinyal İşleme Birimine iletmelidir.

### **3.2. Sayısal Sinyal İşleme Birimi**

Sayısal Sinyal İşleme Birimi, DÇB'nin çıktıları (fonem bilgisi, bürün bilgisi gibi) kullanarak konuşma sinyalini sentezleyen kısımdır. İleride açıklanacağı üzere konuşma sinyalinin sentezlenmesi işlemi, önceden kaydedilmiş parçaların birleştirilmesiyle, konuşma sinyalinin modellerinden faydalanılarak veya insan ses yolunun etkileri hesaplanarak yapılabilir.

İnsan kulağı fonemlerin birleşme kısımlarındaki hatalara karşı hassas olduğu için bu kısımların sentezlenmesi daha fazla önem kazanmaktadır. Kurallı sentezleyiciler bu kısımların sentezlenmesini matematiksel kurallara bağlarken birleştirerek sentezleme yapan birçok sistemde bu kısımlar önceden kaydedilir ve bu kısımlar mümkün olduğu kadar değiştirilmeden konuşmada kullanılır [22].

Konuşmanın matematiksel olarak sentezlendiği sistemlerde oluşturulan konuşmanın doğallığı az ama anlaşılabilirliği fazladır. Önceden kaydedilmiş ses örneklerini kullanan sistemlerde konuşma daha doğaldır, ancak konuşma parçaları eklenirken problemler yaşanır ve bu noktalarda yapılan hatalar ve eksikler konuşmanın anlaşılabilirliğini önemli ölçüde azaltabilir.

Sentezleme yapan sistemlere örnek olarak, artikülatör sentezleyiciler, kurallı sentezleyiciler, sinüzoidal sentezleyiciler ve birleştiren sentezleyiciler örnek olarak verilebilir.

#### **3.2.1. Artikülatör Sentezleyiciler**

Bu yöntemde konuşma oluşumunun prensipleri ve konuşma üretiminin bütün öğeleri simüle edilebilir. Ne var ki, konuşma oluşumu fazlasıyla karışık bir işlemdir ve henüz tamamen anlaşılabilmiş bir olgu değildir. Bu da araştırmacıları bu konuda araştırma yapmaya teşvik etmektedir.

Ses dalgası, “ses yolu filtre sisteminin bir veya daha fazla ses kaynağına verdiği tepkidir” şeklinde tanımlanabilmektedir ve bu fikir kurallı ve artikülatör sentezleyicilerin temel dayanağı olmuştur<sup>11</sup>. Artikülatör sentezleyicilerin kurallı sentezleyicilerden farkı; kurallı sentezleyicilerin doğrudan formant frekansı ve bant genişliklerini parametre olarak kabul etmesi fakat artikülatör sentezleyicilerin ise ses yolu geometrisinin tanımını ve olası ses kaynaklarının yerini ses yolu süzgecinin karakteristiklerini belirlemek için kullanmasıdır.

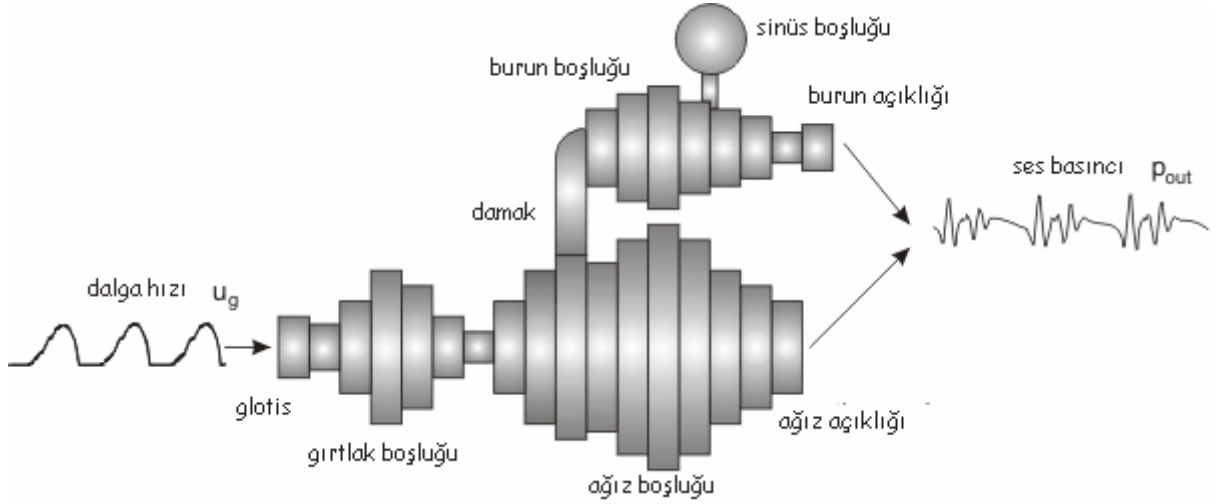
İnsan ses yolu kesit alanında farklılıklar olan bir akustik boru olarak yorumlanabilir. Akciğerlerden gelen hava titreşen glotisten enerji kazanır ve bu enerji ses dalgalarının yayılmasını sağlar. Alan farklılıklarından ve sonlanma etkilerinden dolayı oluşan yansımalar, boru şeklinin rezonans karakteristiklerine sonuç verir. Bu yayılım karakteristiklerini modellemek artikülatör sentezlemenin problemlerinden birisidir. Geniz boşluğu, ağız boşluğuna damak eşleniği ile bağlanmıştır, ağız boşluğu ile arasında enerji transferi gerçekleşebilmektedir ve ağız boşluğuna benzer bir şekilde davranır. Belirli bir konuşmacı için, ağız boşluğunun tersine, geniz boşluğunun şekli sabittir.

Artikülatör sentezleyicilerde, sentezleyiciye göre ses yolu geometrisi bir, iki veya üç boyutlu olarak tanımlanabilir. Bir boyutlu modellerde ses dalgası akustik boru içerisinde, yüksek hızlı darbeli iletim hatlarındaki yayılıma benzer bir şekilde modellenir. Akustik boru, bir dizi daha küçük bölümlere ayrılabilir (yaklaşık 40 tane) ve bu bölümlerin her biri, fiziksel gerçek empedansı oluşturan alt parçalar olarak kullanılır. İlk zamanlarda bu parçalar sığalar ve indüktörlerden oluşmaktaydı ve bu sistemler gürbüz olmamaya meyilliydi. Yakın zamanlarda bilgisayarlar sayesinde simülasyonlar yaygınlaştı. Fakat bu modelleme modeli kontrol ederken, gerçek ses yolu ile ilgili veri eksikliğinden, kontrol edilmesi gereken parametre çokluğundan ve kullanılan parça sayısından sorunlar çıkabilmektedir [12]. Şekil 3.2’de bu model gösterilmektedir.

---

<sup>11</sup> [http://www.wicg.informatik.uni-rostock.de/~piet/speak\\_main.html](http://www.wicg.informatik.uni-rostock.de/~piet/speak_main.html)





Şekil 3.2. İnsan ses yolunun artikülâtör modellenmesi<sup>12</sup>

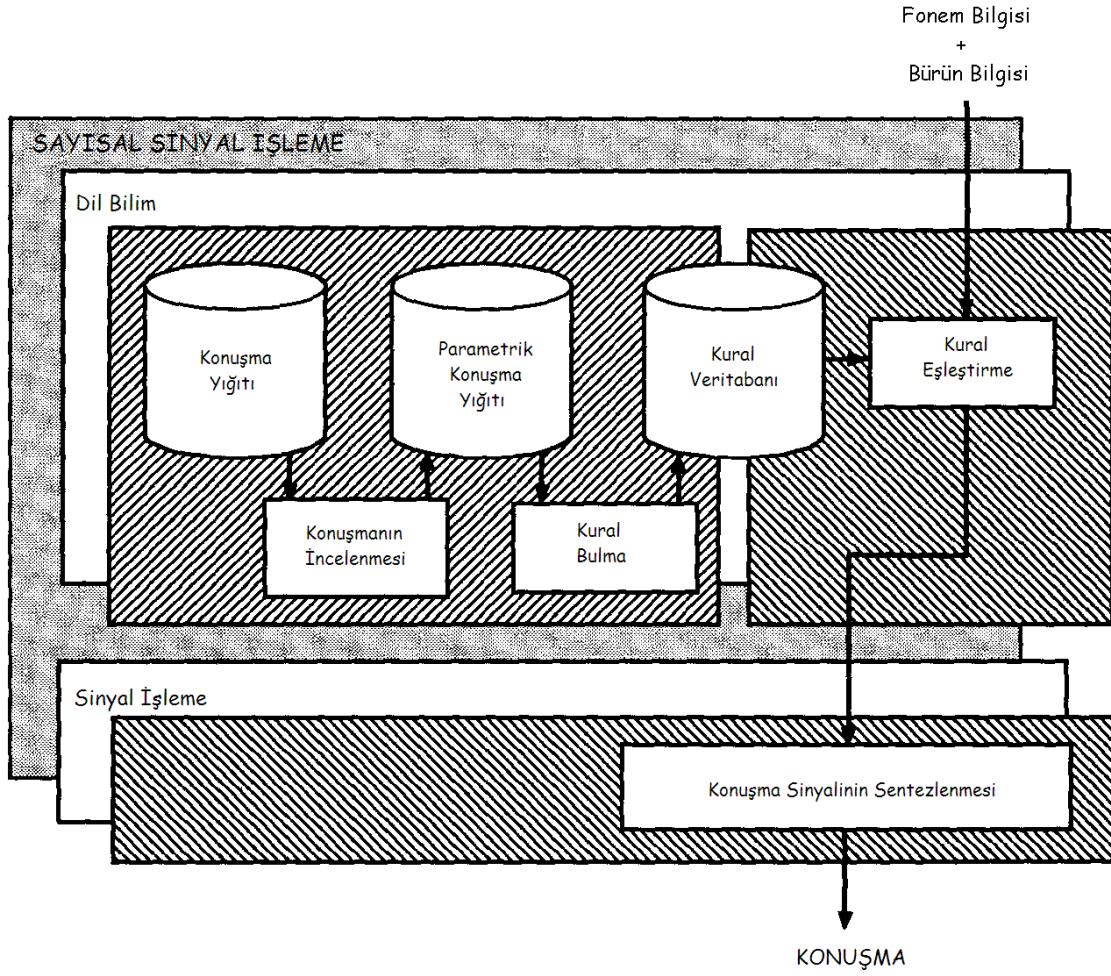
### 3.2.2. Kurallı Sentezleyiciler

Kurallı sentezleyiciler, tarihte en çok kullanılmış olan ve hala da kullanımda kalmayı başarmış sentezleyicileridir. Kurallı sentezleyiciler, konuşmayı daha önce belirlenmiş birtakım kurallara göre sentezleyen konuşma sentezleyicileridir. Pratik nedenlerden dolayı formant sentezleyicisi olarak yapılandırılırlar [7]. Genel şeması Şekil 3.3'teki gibidir [7].

Kurallı sentezleyiciler daha önceden belirlenmiş birtakım kurallara ihtiyaç duyar. Bu kuralları oluşturmak için, olası tüm geçişleri kapsamaması istenen ve genellikle ünsüz-ünlü-ünsüz harf sırasıyla yapılanmış konuşma parçaları önceden kaydedilir. Daha sonra bu konuşma parçaları incelenerek, fonemler arası geçişleri modelleyecek kurallar ve modelde kullanılacak beyaz gürültü standart sapmaları gibi parametreler bulunur. Formant sentezleyiciler, sentezleme işlemi sırasında 60'a yakın parametre kullanırlar.

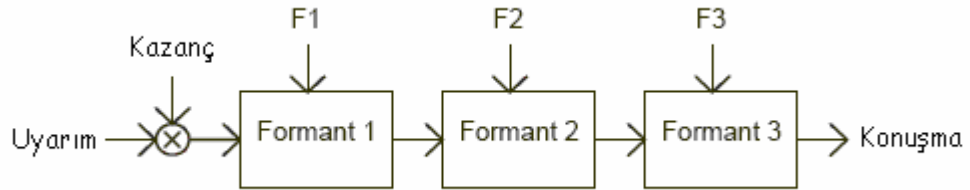
Anlaşılabilir bir konuşma oluşturmak için genellikle en az üç formant gereklidir. Daha kaliteli bir konuşma sentezlemek için ise beş formant kullanılabilir. Genellikle her formant iki kutuplu bir rezonatör ile modellenir. Rezonatörler formant frekansını ve bant genişliğini tanımlarlar [17].

<sup>12</sup> [http://www.icg.informatik.uni-rostock.de/~piet/pictures/tract\\_overview.gif](http://www.icg.informatik.uni-rostock.de/~piet/pictures/tract_overview.gif)



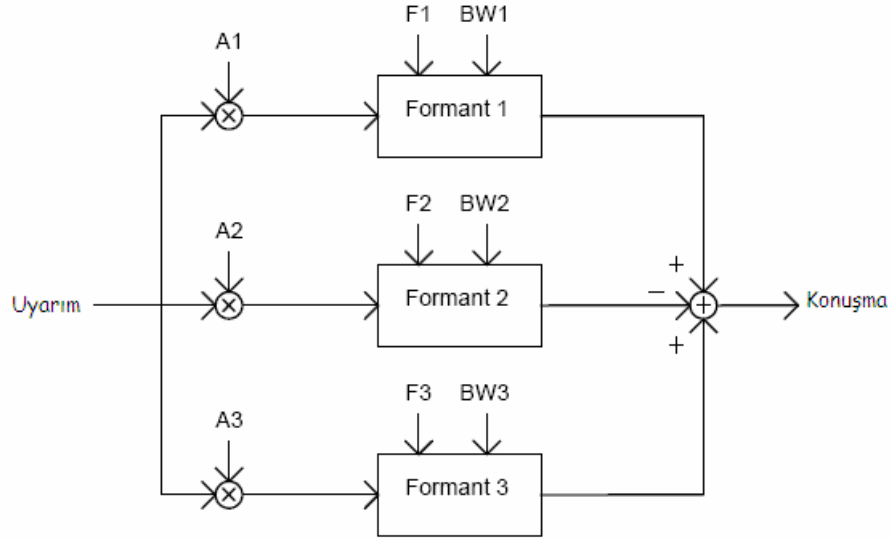
Şekil 3.3. Kurallı sentezleyiciler için genel gösterim [7].

Kullanılan rezonatörler farklı biçimde bağlanabilirler. Rezonatörler Şekil 3.4'te gösterildiği gibi seri şekilde bağlandıkları zaman, her rezonatörün çıkışı, sonraki rezonatörün girişine verilir. Tetikleyici sinyal sadece ilk rezonatöre uygulanır. Bu tip sentezleyiciler, ağız boşluğunda üretilen sedalı seslerin ('a', 'e', 'ı', 'i', 'o', 'ö', 'u', 'ü', 'b', 'c', 'd', 'g', 'j', 'l', 'r', 'v', 'y', 'z') sentezinde daha başarılıdır.



Şekil 3.4. Seri bağlı formant sentezleyicisinin temel yapısı

Rezonatörler Şekil 3.5'teki gibi paralel bağlandıklarında ise tetikleyici sinyal bütün rezonatörlere uygulanabilir ve rezonatör çıkışları toplanır. Ayrıca her rezonatörün katkısı belirli bir çarpan sayesinde ayrı ayrı ayarlanabilir. Paralel bağlanmış rezonatörler, geniz seslerinin ('m', 'n') ve ağız boşluğunda üretilen sedasız seslerin ('f', 'h', 's', 'ş', 'p', 'ç', 't', 'k') sentezinde daha iyi sonuç vermektedirler.

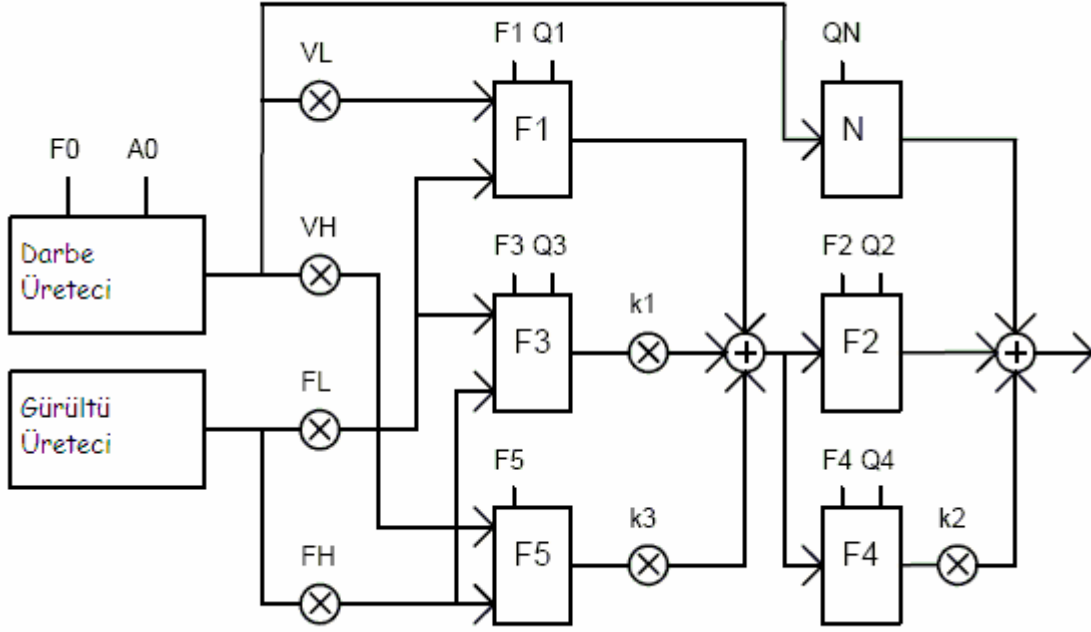


Şekil 3.5. Paralel bağlı formant sentezleyicisinin temel yapısı

Bu iki yöntemin dışında daha karmaşık bağlantılar kullanan sentezleyiciler de tasarlanmıştır. Bu sentezleyiciler, her iki modelin de güçlü yönlerini kullanmaya çalışırlar. Fin dili için Laine tarafından tasarlanan model Şekil 3.6'da gösterilmiştir [17]. Bu modelde, ses yolu iki kısmi transfer fonksiyonu ile modellenmiş, ve  $k_1$ ,  $k_2$  ve  $k_3$  katsayıları sayesinde paralel yolların kazançları eşitlenmiştir.

Kurallı sentezleyicilerde sentezlenen konuşmanın kalitesi aşağıdaki unsurlara bağlıdır [7]:

- İçsel kural verimliliği: konuşma parçaları mümkün olduğu kadar az duyulabilir hata ile parametreler tarafından tanımlanabilmelidir.
- Kaydedilen konuşmaların kalitesi.
- Konuşmayı modellemede kullanılan modelin doğruluğu.



Şekil 3.6. Laine tarafından tasarlanan formant sentezleyicisi [17]

Kurallı sentezleyiciler, özellikle birleştirerek sentezleme yöntemine göre daha az konuşma veritabanı boyutu istemesine karşın ciddi bir işlem gücü istemektedir. Bu sebeple işlem gücünün fazla olmadığı ortamlarda tercih edilmemektedirler.

### 3.2.3. Birleştirerek Sentezleme:

Adından da anlaşılacağı gibi, birleştirerek sentezleme yapan MKS sistemleri, önceden kaydedilmiş konuşma parçalarını, belirli kuralları veya algoritmaları uygulayarak birleştirerek konuşma sentezlerler. Bu sentezleyicilerde işlem yükü artikülatör ve kurallı sentezleyicilere göre daha azdır ve konuşmanın doğallığı daha fazladır [22]. Konuşma için önemli unsurlar, yani perde frekansı, enerji seviyesi ve fonem uzunluğu, konuşma veritabanındaki parçalar farklı metinlerden elde edildiği için farklılık gösterecektir. Birleştirerek sentezleme yapan sistemlerin bu farklı özellikleri istenilen hale getirmesi gerekmektedir. Kullanılacak parçaların çeşidi ve kullanılacak birleştirme algoritmasının ne olacağına MKS sisteminin tasarımcısının karar vermesi gerekmektedir. Sıradaki bölümlerde bu konu daha detaylı şekilde incelenecektir.

### 3.2.3.1. Kullanılacak Parça Seçimi

Veritabanına kaydedilecek parçalar için birkaç seçenek mevcuttur. Bu seçenekler içinde genellikle veritabanı boyutu ve konuşma kalitesi arasında doğru bir orantı bulunmaktadır.

Konuşmada en küçük yapıtaşı fonemdir. Dolayısıyla oluşabilecek en küçük konuşma veritabanı fonemlerle oluşur. Türkçede 43 fonem bulunmaktadır. Fakat fonemlerin birleştirilerek konuşma oluşturulması MKS sistemlerinde neredeyse hiç kullanılmamaktadır. Bunun sebebi, fonemlerin birleşme bölgelerinin net olmamasından dolayı tek tek kaydedilmelerinin sorunlu oluşu ve fonemlerin birleşme bölgelerinin doğru bir şekilde sentezlenmesindeki zorluktur. Ayrıca, fonemler neredeyse hiçbir zaman tek başlarına kullanılmazlar [28] ve kendilerinden önce ve sonra gelen fonemlerden etkilenirler ve bu yüzden tek başlarına kaydedilmek yerine bitişik fonemler halinde çoklu olarak kaydedilip kullanılırlar [2].

Bu problemlerden dolayı difon kullanımı daha yaygındır. Difonlar, iki tane yarım fonemden oluşmaktadır. Böylece fonemlerin daha hassas olan birleşme bölgeleri veritabanında kayıtlı bulunmakta, fonemlerin görece daha durağan ve dolayısıyla işlenmesi daha kolay olan merkez kısımları, difonların sınırlarını oluşturmaktadır.

Türkçede 43 fonem bulunur, bu fonemler Ek C'de gösterilmiştir. Bu fonemlerin yanında kelime baş ve sonlarında kullanılmak üzere boşluk da bir difon oluşumunda bir eleman olarak kullanılırsa Türkçede 44\*44 adet difon oluşabilir. Fakat gerçek rakam bunun altındadır. Çünkü Türkçedeki kelimelerde bütün fonemlerin yan yana gelmez. Örneğin Türkçede genellikle sesli ve sessiz harfler yan yana bulunur. Sessiz harfler çok sık yan yana gelmez. Ayrıca 'p', 't' ve 'k' seslerinden önce boşluk bulunmaktadır, dolayısıyla 'xp', 'xt', 'xk' gibi difon parçaları veritabanına konulmazlar. Burada 'x' herhangi bir harfi ifade etmektedir.

Daha küçük parça boyutu, daha fazla işlem ve birleşme noktalarında daha çok bozulma demektir. Daha az işlem ve daha az bozulma için daha büyük konuşma parçaları kullanılabilir. Daha büyük konuşma parçaları olan trifonlar, sesliden sesliye parçalar ve sessizden sessize parçalar kullanılacak diğer seçenekler

arasındadır. Daha önceki çalışmalarda Ayhan [2] sesliden sesliye, Salor [23] sessizden sessize, Eker [8] ve Özen [22] ise difon parçalarını kullanmışlardır. Bunların dışında uygulama geliştirici kendi tanımladığı konuşma parçalarını da kullanabilmektedir [10].

### 3.2.3.2. Birleştirme Sırasında Yapılan İşlemler:

Daha önce belirtildiği gibi difon parçaları birleştirilirken öncelikle konuşma parçalarının enerji seviyeleri eşitlenmelidir. Bu iş için birleştirilecek parçaların birleşme noktasına yakın bir bölgesinde yerel olarak enerji seviyeleri hesaplanır ve daha sonra bu seviyeleri eşitleyecek şekilde konuşma parçalarından birisinin (veya her ikisinin de) genlikleri değiştirilir. Birinci sıradaki parçanın enerji seviyesi  $E_1$  ise, ve ikinci sıradaki parçanın enerjisi de  $E_2$  ise, birinci parçanın genliği  $\sqrt{E_2/E_1}$  ile çarpılabilir, veya ikinci parçanın genliği  $\sqrt{E_1/E_2}$  ile çarpılır. Bunların dışında her iki parçanın da genliği değiştirilerek enerji seviyeleri eşitlenebilirse de, bu durum fazladan işlem yükü gerektirecektir.

Sentezlenen konuşmadaki enerji seviyesinin, istenilen enerji eğrisine ulaştırılması içinse sentezlenen konuşma üzerinde genlik değişikliği yapılır. Bunun için istenilen konuşma eğrisi üzerindeki yerel enerji seviyesi, sentezlenen konuşmadaki yerel enerji seviyeleriyle karşılaştırılır ve aradaki oran genlik değiştirmede kullanılır. Yerel enerjinin hesaplanması için, yerel olarak kabul edilen bölgede belirli bir uzunlukta pencereleme yapılır ve bu pencerenin altında kalan alanın enerjisi hesaplanır [23]. Yerel enerji seviyesinde uygulanan “yerel”lik sınırı farklılık gösterebilir. İdeal olan her sentezlenen konuşmadaki her nokta için pencereleme ve enerji hesaplanması yapılmasıdır, ama bu işlem yükünü çok artıracığı için belirli aralıklarla enerji hesaplanması ve genlikte değişiklik yapılması uygulanabilir [22].

İstenilen fonem uzunluğu birçok zaman veritabanındaki örneklerin uzunluğundan farklı olacaktır. Eğer istenilen uzunluk veritabanındaki örnekten kısaysa, konuşma parçasından periyot çıkartılarak bu işlem gerçekleştirilebilir. Aksi bir durumda da periyot eklenerek konuşmanın uzunluğu artırılır. Özellikle sedalı seslerde periyodiklik belirgin olduğu için bu işlem kolaylıkla yapılır. Periyodikliği daha az olan sedasız seslerde ise belirli aralıklarla perde işaretleri konur ve işlemin geri

kalanı sinyal periyodikmiş gibi yapılır. Parça çıkarılacağı zaman, insan kulağı fonemlerin birleşme noktalarına daha hassas olduğu için fonemlerin birleşme noktalarından uzakta parçaların çıkarılması tercih edilir [22]. Aynı şekilde parça ekleneceği zaman, fonemin en durağan olduğu, fonemin orta noktalarından parça eklenir. Aynı zamanda bu noktalarda periyodiklik yüksek olduğu için de konuşmadaki bozulma daha az olur.

Perde frekansının değiştirilmesi ise biraz daha karmaşık bir iştir ve bu iş için farklı yöntemler uygulanmaktadır. Bunlardan en yaygınları Perde Eşzamanlı Örtüştürerek Ekleme (PSOLA) yöntemleri ve sinüzoidal yöntemdir.

### 3.2.3.2.1. PSOLA Yöntemi

PSOLA metodunun temel mantığı, özellikle sedalı konuşma sinyalindeki tepe noktaları çevresindeki yerel bölgedeki kısa dönem sinyallerin elde edildikten sonra, perde frekansı değiştirilerek, elde edilmiş olan örneklerin üst üste eklenmesi ve yeni perde frekanslı konuşma sinyalinin elde edilmesine dayanmaktadır.

PSOLA metodunun birkaç versiyonu bulunmaktadır. Zaman Ekseninde Perde Eşzamanlı Örtüştürerek Ekleme (TD-PSOLA) yönteminde konuşma sinyalindeki yerel tepe noktaları etrafında pencereleme işlemi yapılır ve elde edilen kısa dönemli parçalar daha sonra istenilen perde periyodu kadar aralıklarla art arda getirilip toplanırlar. Konuşma sinyali  $x(n)$ ,  $m$ 'inci kısa dönem sinyali  $x_m(n)$ , kullanılan pencere  $h(n)$ , yerel tepe noktaları arasındaki uzaklık  $T_0$  ile gösterilirse, kısa dönem sinyaller

$$x_m(n) = h(n - mT_0)x(n)$$

şeklinde gösterilebilir. Burada kullanılan pencere genellikle Hamming veya Hanning tipi penceredir [3]. Pencere uzunluğu perde periyodunun iki katı olarak seçilir. Pencerenin orta noktası, yerel tepe noktasına ayarlanır.

Konuşma sinyali,  $T$  perde periyodu ile yeniden oluşturulmak istenirse

$$\tilde{x}(n) = \sum_{m=-\infty}^{\infty} x_m(n - m(T - T_0))$$

şeklinde yeniden oluşturulabilir. Burada  $\tilde{x}(n)$  yeniden oluşturulan konuşma sinyalini ifade etmektedir.

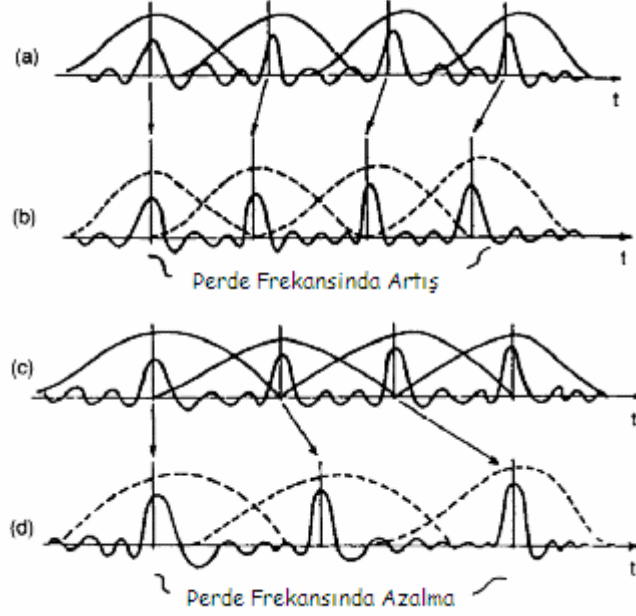
Bu formüller, sinyalin  $-\infty$  ve  $+\infty$  arasında periyodik olmasını gerektirir. Gerçekte bu şekilde bir konuşma sinyali olmamasına rağmen, kısa dönemli de olsa periyodik olan konuşma sinyallerine bu yöntem uygulandığında sonuç alınabilmektedir.

Burada önemli olan bir nokta da yerel tepe noktalarının işaretlenmesidir. Görüldüğü gibi bu yöntemde yerel tepe noktalarının doğru işaretlenmesi çok kritiktir. Yerel tepe noktaları bir algoritma yöntemiyle bulunabileceği gibi el ile de işaretlenebilir.

Şekil 3.7'de perde frekansının artması ve azalması için bir örnek gösterilmiştir. Bu şekilde (a) ve (c) değişiklik yapılmak istenen sinyalleri ve pencereleri, (b) perde frekansı artırılmış sinyali, (d) ise perde frekansı azaltılmış sinyali göstermektedir. Şekil 3.7.(a)'da bulunan sinyal işaretlenmiş olan tepe noktaları etrafında iki periyot genişliğindeki Hanning veya Hamming pencereleriyle pencerelenir. Daha sonra elde edilen pencerelenmiş parçalar istenilen perde periyodu kadar uzaklıklarla aynı sırayla sıralanır ve bu örneklerin üst üste kalan kısımları toplanarak perde frekansı artırılmış sinyal Şekil 3.7.(b)'deki gibi oluşturulur. Benzer işlemler Şekil 3.7.(c)'deki sinyale uygulanarak, Şekil 3.7.(d)'deki gibi perde frekansı azaltılmış sinyal oluşturulur.

TD-PSOLA yönteminin bazı bilinen zayıflıkları bulunmaktadır. Eğer yerel tepe noktası işaretleri ile pencere tepe noktaları örtüşmezse bir faz uyumsuzluğu oluşur. Eğer birleştirilecek parçalar aynı perde frekansıyla kaydedilmedilerse, her iki parçadan gelecek olan kısa dönem örneklerin boyutu farklı olacaktır, (halbuki teorik temelde bu parçaların eşit boyda olması esas alınmıştır) bu da sentezlenen konuşmada bozulmaya yol açacaktır [6].





Şekil 3.7. TD-PSOLA yöntemi ile perde frekansının değiştirilmesi [17] (a) ve (c) değişiklik yapılmak istenen sinyalleri ve pencereleri, (b) perde frekansı artırılmış sinyali, (d) ise perde frekansı azaltılmış sinyali göstermektedir

Başka PSOLA yöntemleri de geliştirilmiştir. Bunlardan biri de Doğrusal Öngörümlü Perde Eşzamanlı Örtüştürerek Ekleme (LP-PSOLA) yöntemidir. TD-PSOLA yönteminde perde periyodu değişikliği tamamen zaman ekseninde yapılmaktayken LP-PSOLA yönteminde, konuşma üretimi için kaynak filtre modeli temel alınır. Bu modelde, sedalı sesler, bir Doğrusal Öngörümlü Kodlama (LPC) süzgecinin dürtü dizisi girdisine verdiği çıkış olarak tanımlanır. Sedasız sesler de, bir LPC süzgecinin beyaz gürültü girdisine verdiği çıkış olarak tanımlanır. Bu durumda sedalı seslerin perde frekansını değiştirmek için, önce istenen konuşma parçası için, önceden kaydedilmiş konuşma parçaları kullanılarak LPC katsayıları bulunur ve bu katsayılardan bir LPC süzgeci oluşturulur. Daha sonra istenen perde frekansıyla bir dürtü dizisi bu süzgece girdi olarak verilir ve böylece süzgecin çıkışında bu istenen perde periyotlu konuşma elde edilmiş olur [22].

LP-PSOLA yönteminde kullanılan yöntemdeki bulunan doğrusal kestirim katsayıları genellikle yapay olarak nitelenebilecek, doğal gözükmeyen bir konuşma sentezlenmesine yol açar. Bu soruna karşılık, LPC filtresine girdi olarak dürtü dizisi vermek yerine, önceden kaydedilmiş konuşma sinyalinin LPC filtresinin tersi ile

filtrelenmiş hali (kalıntısı) verilir. Bu yöntem Kalıntı Uyarımlı Doğrusal Öngörümlü Perde Eşzamanlı Örtüştürme Ekleme (RELP-PSOLA) olarak adlandırılır.

### **3.2.3.2.2. Sinüzoidal Yöntem**

Bu yöntem sedalı seslerin, sinüzoidallerin toplamı şeklinde ifade edilebileceği varsayımına dayanmaktadır. Bu yöntemde perde frekansı değiştirilmek istendiği zaman, bulunan sinüzoidallerin frekansı değiştirilerek konuşma yeniden sentezlenir [21]. Bu yöntem sedalı seslerin sentezlenmesinde başarılı olsa da sedasız seslerin sentezlenmesinde sorunlar yaşamaktadır [17]. Bunun için sedasız seslerin sentezlenmesi için olasılıksal bileşenler kullanılabilir [22].

## 4. TÜRKÇEDE BÜRÜN YAPISI

Bölüm 3'te belirtildiği gibi, MKS sistemlerinde bir dil işleme birimi bulunmaktadır. Dil işleme birimi her dil için farklıdır ve o dilin bürün özelliklerine göre girilen metinden fonetik sembol verilerini sonraki birime, yani sinyal işleme birimine aktarır. Sinyal işleme birimi de gelen veriye göre konuşma sinyalini oluşturur. Bu bölümde Türkçe için temel bürün özellikleri incelenecektir.

### 4.1. Bürün Öğeleri

Konuşma sentezleme sırasında kullanılan parametrelerin doğru bir şekilde belirlenmesi MKS sistemlerindeki en büyük zorluklardan birisidir. Bürün terimi ses sinyalindeki yeğlilik, yükseklik ve uzunlukla ilgili parçalarüstü birimleri tanımlamaktadır. Bürün olguları her dilde dilbilimsel açıdan anlam ayırt edici bir işlev yerine getirmez [11], ancak konuşmanın doğallığı, konuşmacının ruh hali ve duyguları gibi etkiler için gereklidir. Bürün öğeleri Süre, Durak, Ton, Ezgi ve Vurgu olarak sıralanabilir.

#### 4.1.1. Süre

Sedalı fonemlerin çıkarılış süreleri dilde anlam ayırt edici güçtedir. Ayrıca fonem sürelerinin doğruluğu konuşmanın anlaşılabilirliği ve doğallığı açısından da önemlidir. Uzun ünlüler ya yabancı kökenli sözcüklerde, ya da ses yitimiyle oluşan özel durumlarda karşımıza çıkmaktadır [11]. Aşağıdaki örneklerde görüldüğü gibi, ilk ünlü sesin süresi uzatılınca kelimelerin anlamları farklılaşmaktadır.

|       |     |
|-------|-----|
| düğün | dağ |
| dün   | da  |

#### 4.1.2. Durak

Durak, bir cümle ya da kelime içinde bulunan birden çok bilgi öbeği arasında verilen kısa aralardır. Konuşmacının iletmek istediği duygu ve düşüncelerin, metnin amacının en etkili ve anlaşılır biçimde alıcı tarafından algılanabilmesine yardımcı olan duraklar, gereği gibi kullanılmadıklarında yanlış anlamalara neden olacak ölçüde önemli ve anlam ayırıcı özellik taşımaktadırlar [11]. Aşağıdaki iki örnekte durak bulunan yerler “/” karakteri ile işaretlenmiştir:

Dilenci/ adama doğru elini uzattı.

Dilenci adama/ doğru elini uzattı.

Duraklar, cümle içerisinde doğru yerlerde virgül kullanılarak okuyucuya veya MKS sistemlerine aktarılabilir; örneğin “Dilenci, adama doğru elini uzattı” cümlesinde olduğu gibi.

#### 4.1.3. Ton

Bir seslemdeki sıklık yüksekliği ya da düşüklüğü, yani bir seslemin tiz ya da pes sesletilmesi olarak tanımlanan ton, Çince, Japonca, İsveççe, Norveççe gibi kimi dillerde sözcüklere yeni anlamlar yüklerken, Türkçede ezgi birimine bağlı olarak genellikle tek sözcükten oluşan bildirimlerde anlam ayırıcı bir özellik taşır. Aşağıdaki örneklerde ton değişikliğini belirtmek için tonun değiştiği heceden sonra değişim yönünde bir ok işareti kullanılmıştır.

efendim↑ (anlamadım)

e↑fendim (çağrıya yanıt)

ha↑ (anlamadım)

ha↓ (anladım)

Tonun kullanımı kişiye, onun ruhsal durumuna ve genel yapısına bağlı olarak çeşitlilik gösterebildiği gibi ulusların karakterlerine uygun olarak değişik biçimlerde ortaya çıkabilir. Örneğin İtalyanca ve İspanyolca, kullanılan ton yüksekliği açısından önde gelirken, Türkçede zayıf, İngilizcede ise çok zayıf ton yüksekliklerinden söz edilebilir [11].

Bu tür ton değişiklikleri, ‘?’ ve ‘!’ gibi doğru noktalama işaretleri kullanılarak okuyucuya ve MKS sistemlerine doğru şekilde aktarılabilir; örneğin: “efendim?” ve “efendim!” şeklinde.

#### 4.1.4. Ezgi

Bir konuşma zincirindeki seslem, biçimbirim ve sözcükleri kapsayan ton değişimlerinin tümü, cümlenin ezgisini oluşturmaktadır. Konuşmacının eğitimi, sosyal düzeyi, konuşmanın geçtiği ortam ve kişinin o andaki ruhsal durumuna göre değişiklik gösterdiği için ezginin belirli kalıplara, kesin kurallara bağlanması oldukça zordur.

Tüm bu güçlülere karşın dilbilimciler, çeşitli dillerdeki ezgi sistemini araştırmakta, işleyiş kurallarını belirlemeye çalışmaktadırlar. Ancak, ezginin kullanımı yukarıdaki koşullara bağlı olduğundan hemen hemen her dil için belirlenmiş kimi temel ezgi biçimlerinin dışında, evrensel nitelikli kurallar koymak olanaksızdır denilebilir. Belirlenebilen kurallar ışığında temelde üç tür ezgiden söz edilebilir:

**a) Biten Ezgi:** Cümlelerin bittiğini, iletilmek istenen bildirinin sona erdiğini dinleyiciye iletmek işleviyle yüküldür. Ses tonunun cümlelerin sonunda düşmesiyle ortaya çıkar.

Bunu beğenmedim↓.

Çocuklar sinemaya gittiler↓.

**b) Süren Ezgi:** Genellikle yan cümlelerden ya da sıralı cümlelerden oluşmuş bildirilerde dinleyiciye bildirinin süreceğini iletmek üzere ses tonunun ezgi doruğuyla aynı düzeyde kalmasıyla ya da bir iki perde yükselmesiyle oluşur.

Geldim, ↑→ gördüm, ↑yendim. ↓

**c) Soru Ezgisi:** Dinleyiciden herhangi bir konuda bilgi ya da yanıt istendiğinde, ses tonunun cümle sonunda yükselmesiyle ortaya çıkar.

Geliyor↑ mu↓ ?

Yazacak↑ mısın↓ ?

Türkçede soru ekleri (–mi, –mu vb.) üzerine vurgu almadığı için ezgi doruğu bir önceki seslemin üzerinde kalır.

Cümlelerin ezgisi, konuşmacının o andaki ruhsal durumuna, konuşma ortamına ya da önceden belirlenmiş kimi kurallara bağlı olarak değişebilir. Bir söyleşi sırasında kullanılan bir cümlelerin ezgisiyle aynı cümlelerin eldeki metinden okunuşunda ortaya çıkan ezgi, doğal olarak farklıdır. Karşılıklı konuşmada kurulan cümleler, sözdizimi kurallarına uymasalar da, konuşucu sesinin tonuyla dinleyicinin dikkatini odaklamak istediği bilgi öbeğine çekebilir. Metne bağlı konuşmalarda ise ezgi, ton, durak ve kavşaklar, metnin hazırlanış biçimiyle ve sözdizimiyle yakından ilişkilidir [11].

#### 4.1.5. Vurgu

Sözcük vurgusu, bir konuşmanın akışı içerisinde anlamına göre sözcüğün bir sesleminin (öne çıkarmak için) diğerlerine oranla daha baskılı, daha soluklu söylenmesidir [11].

Soluk baskısı “kuvvetli” ve “zayıf” olmak üzere iki derecelidir. İşlevi, en az iki heceli ve daha uzun sözcük ve sözlerde ortaya çıkar [4]:

bebek (arabası)      yalnyz (adam)      (iki) açma (çörek)  
bebek (koyu)      yalnyz (sen gel)      açma (kapıyı)

Türkçe sözcüklerde vurgu, karşıt bir durum söz konusu değilse genel olarak son söylemedir. Fakat bu kurala uymayan birçok örnek de bulunmaktadır. Ayrıntılı bilgi için Ergenç'in [11] çalışmasına bakılabilir.

#### 4.2. Yumuşak G

Türkçenin alfabesinde harf olarak yer alan <ğ>'nin ses olarak gerçekleşmediği ancak dilde pek çok işlev üstlendiği bilinmektedir [25], [11].

a) Sözcüğün içerisinde aynı nitelikli ünlüler arasında, sözcüğün sonesinde bir ünlüden sonra ve önünde bir ünlü ardında bir ünsüz varken söyleyişte yitirilen <ğ> ünlülerin uzamasına neden olur [25].

uğur ~ /u:r/,      dağı ~ /da:/,      yağımur ~ /ya:mur/

b) Sözcüğün içerisinde bulunan ve söyleyişte yitirilen <ğ>'nin önünde ve ardında ayrı nitelikli ünlüler bulunuyorsa konuşma dilinde ünlü kayması oluşur.

ağıt ~ /a:ıt/,      doğua ~ /do:a/,      öğe ~ /ö:e/

c) Eğer birlikte bulunduğu ünlüler düz öndil ünlüleriye /y/ sesine dönüşebilir. Yarı ünlü sayılan /y/ sesinin çıkış yeri /i/ ünlüsüne çok yakın olduğu için sözde ünlü kayması ortaya çıkar.

eğitim ~ /ejitim/ ~ /e·itim/  
eğilence ~ /ejlence/ ~ /e·ilence/

## 5. TAŞINABİLİR PLATFORMLAR

Bu çalışmanın odaklandığı ana konulardan birisi de bir MKS sistemini taşınabilir bir platformda uygulamaktır. Bu amaçla özellikle günümüzde kullanımı oldukça yaygın olan cep telefonlarında çalışabilecek bir uygulama geliştirilecektir.

Cep telefonu için program geliştirmek isteyen birisi için çok farklı teknoloji kullanan iki ana seçenek vardır, bunlar Java 2 Platformu, Mikro Yayımı<sup>13</sup> (J2ME)<sup>14</sup> ve SymbianOS<sup>15</sup> platformlarıdır [13]. Bu platformlar dışında mobil cihazlar için Python, .net Compact Framework, PocketPC, PalmOS, FlashLite gibi platformlarda da uygulama geliştirilebilir, fakat bu platformlar yaygınlık, dökümantasyon, uygulama geliştirilmesi için araç ve geliştirme ortamı desteği gibi konularda J2ME ve SymbianOS platformlarının gerisinde kalmışlardır<sup>16</sup>. Bu yüzden yazılım geliştirilecek platform için J2ME ve SymbianOS arasında bir tercih yapılması daha uygun olacaktır.

### 5.1. J2ME

Java programlama dili kullanılarak cep telefonlarında J2ME uygulaması geliştirilebilir. Java uyumlu taşınabilir cihazlar gün geçtikçe yaygınlaşmakta ve neredeyse bütün cep telefonu üreticileri tarafından kullanılmaktadır. Her ne kadar başka yorumlanan programlama dilleri kullanılma imkanı varsa da, J2ME açık bir standarttır, Java tecrübesi olan geliştiriciler tarafından kolaylıkla öğrenilebilir ve diğerlerine göre çok daha fazla donanım üreticisi tarafından desteklenmektedir [13].

### 5.2. Symbian OS

SymbianOS, kendi Uygulama Programlama Arayüzü (API'si) kullanılarak genelde C++ dilinde yazılan ve uygun makine koduna göre derlenen uygulamaları çalıştırabilen bir işletim sistemidir. Başka işletim sistemleri bulunsa da, telefon üreticilerinin %70'i Symbian'ı kullanmaktadırlar ve bu Symbian işletim sistemini diğer işletim sistemlerinden daha yaygın duruma getirmektedir. Ayrıca, SymbianOS açık bir standarttır ve herhangi bir donanım üreticisi için lisanslanabilir ve özel olarak kablosuz iletişim cihazları için geliştirilmiştir [13].

<sup>13</sup> <http://java.sun.com/javame/index.jsp>

<sup>14</sup> Kısaltma olarak "J2ME" yerine "Java ME" de kullanılmaktadır.

<sup>15</sup> <http://www.symbian.com>

<sup>16</sup> [http://en.wikipedia.org/wiki/Mobile\\_development](http://en.wikipedia.org/wiki/Mobile_development)

Java destekleyen telefonlar, piyasada Symbian OS destekleyen telefonlardan sayıca daha fazladır. J2ME, Java 2, Standart Yayımının (J2SE) bir alt kümesi olduğu için, Java tecrübesi olan geliştiriciler J2ME ortamına daha kolay geçebilmektedirler. Diğer taraftan Java platformu Symbian'dan daha yaygındır ve Java için geliştirilen bir uygulama, Symbian için geliştirilen bir uygulamadan daha çok sayıda telefonda çalıştırılabilir.

### **5.3. J2ME Yapısı**

J2ME, Sun Microsystem'in cep telefonları, KSY'lar ve diğer küçük ve taşınabilir gömülü cihazlar piyasası için çıkarttığı bir Java sürümüdür. Ortaya çıktığından beri Palm, Nokia, Motorola gibi büyük şirketler dâhil 600'den fazla şirket geliştirme çalışmalarına katılmıştır. J2ME, küçük cihazlar için gelişmiş uygulamalar yaratmak için eksiksiz bir dizi çözüm sunmaktadır.

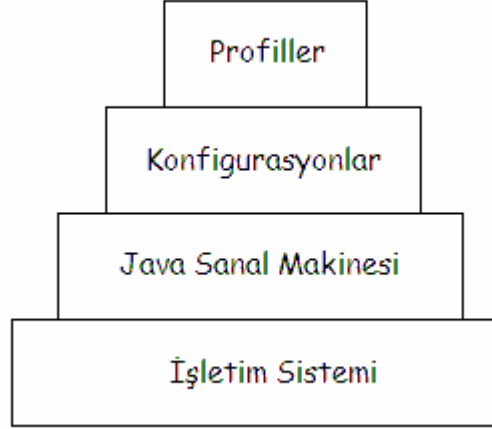
#### **5.3.1. J2ME'e genel bir bakış:**

J2ME, genel olarak aşağıdaki bileşenlerle tanımlanır:

- Her biri farklı gereksinimleri karşılayan ve her biri farklı tipte küçük cihaz üzerinde kullanılmak için olan bir dizi Java sanal makinesi.
- “Konfigürasyon” ve “Profil” olarak bilinen ve her sanal makine üzerinde çalışabilen bir grup kütüphane ve API'leri.
- Dağıtım ve cihaz konfigürasyonu için çeşitli araçlar.

İlk iki öge J2ME çalışma ortamını oluşturur. Şekil 5.1 çalışma ortamının ilişkilerini yansıtmaktadır. Cihazın işletim sistemi üzerinde Java sanal makinesi çalışmaktadır. Bunun üzerinde cihaza özel, cihazın kaynak gereksinimleri için temel fonksiyonellik sağlayan kütüphaneleri içeren J2ME konfigürasyonu bulunmaktadır. Konfigürasyonun üzerinde de, benzer cihazlar üzerinde benzer işlevsellikler için kullanılan ilave kütüphaneleri içeren profillerden bir veya birden fazlası bulunmaktadır [19]. Konfigürasyonlar daha alt seviyedeki kütüphaneleri tanımlarlar (örneğin kullanılan sayı tipler ve temel sınıf yapıları gibi), profiller ise daha üst seviyedeki ve cihaza daha bağımlı kütüphaneleri tanımlar (cihazın üzerindeki ekranda çizilen yüksek seviyeli nesnelere gibi).





Şekil 5.1. J2ME uyumlu bir sistemde genel çalışma mimarisi

Konfigurasyonlar, işlem gücü ve bellek miktarını temel alarak ürünler için bir gruplama tanımlar. Şu anda J2ME dünyasında iki standart konfigurasyon bulunmaktadır: Bağlantılı Cihaz Konfigurasyonu (CDC) ve Sınırlı Bağlantılı Cihaz Konfigurasyonu (CLDC). Aynı zamanda bu konfigurasyonlar, Java API'leri olarak standartlaştırılmışlardır. CDC 1.0, JSR-36 olarak, CDC 1.1, JSR 218 olarak, CLDC 1.0, JSR-30 olarak, CLDC 1.1, JSR 139 olarak tanımlanmıştır.

Profiller, uygulamaların cihaza has bazı özelliklere ulaşmasını sağlayan, konfigurasyonların üzerinde çalışan bir grup API'dir. Profiller de konfigurasyonlar gibi standartlaştırılmıştır. Bazı örnekleri Taşınabilir Bilgilendirme Aygıtı Profili (MIDP), Foundation Profili, Personal Profili ve Uzaktan Metot Çağırımı (RMI) Profili olarak sıralanabilir.

J2ME platformu ile daha ayrıntılı bilgi Ek D'de bulunabilir.

#### **5.4. J2ME Projelerinde Dikkat Edilmesi Gerekenler**

J2ME platformu gömülü yazılım geliştirilen bir ortamdır ve diğer platformlardan farklıdır. Bu farklılıkların içinde işlem gücü, bellek miktarı gibi donanımsal kısıtlamaların yanı sıra, cihaz üzerinde çalışan ev sahibi ortam (işletim sistemi, Java sanal makineleri gibi) da birtakım kısıtlamalar getirebilirler. Bu kısımda J2ME platformunda yazılım geliştirilirken dikkat edilmesi gereken bazı hususlara değinilecektir.

**a) Geliştirilmesi gereken yazılımdan beklenenler:** Bir yazılım geliştirilmek istendiği zaman, yazılımdan beklenen özelliklerin gerçekleştirilme olasılığı bulunup bulunmadığı araştırılmalıdır. J2ME platformunda da J2SE'e göre bir çok kısıtlama olduğu için, istenilen yazılımın bu kısıtlamalarla sorun yaşayıp yaşamayacağı tespit edilmelidir [27].

Örneğin bu çalışmada cep telefonuna gelen kısa mesajların okunması için "JSR-120 Telsiz Mesajlaşma Programcı Arayüzü"<sup>17</sup> ve sentezlenen konuşmanın ses dalgaları halinde dışarıya verilebilmesi için "JSR-135 Taşınabilir Ortam Programcı Arabirimi"<sup>18</sup> kütüphaneleri kullanılmıştır. Bu kütüphaneler (veya bu kütüphanelere eş başka kütüphaneler) var olmasaydı bu çalışma gerçekleşemeyecekti. Sözgelimi, JSR-120 ve JSR-135 2002 yılında ortaya çıkarılmıştır. Dolayısıyla 2002 yılından önce, hiçbir cep telefonunda kısa mesajların Java yazılımları tarafından girdi olarak alınması ve bu mesajın konuşmaya dönüştürülmesi de mümkün değildi.

**b) Hedef cihazların seçimi:** Yazılımın koşacağı cihazlar seçilirken uygun donanımsal ve yazılımsal desteği sağlayabilecek bir cihaz seçilmesine dikkat edilmelidir.

Bu çalışmada geliştirilen uygulama bir Nokia 6630 markalı cep telefonu üzerinde koşturulmuştur. Bu cihaz bir cep telefonu olarak kısa mesajları kabul edebilmekte, J2ME platformunu ve JSR-120, JSR-135, CLDC 1.1 ve MIDP 2.0 API'lerini desteklemekte, böylece JSR-120 sayesinde cihaza gelen kısa mesajlar uygulamaya girdi olarak alınabilmekte, JSR-135 sayesinde sentezlenen konuşma dışarıya analog sinyaller olarak verilebilme, CLDC 1.1 ile gelen "float" veri tipi sayesinde hassas matematiksel işlemler hızlı ve doğru bir şekilde kodlanarak yapılabilme ve MIDP 2.0 sayesinde telefon ekranında grafiksel bir ara yüz kullanılabilir [9], [18]. Diğer taraftan bu cihaz üzerindeki işlemci gücü ve bellek miktarı uygulamayı koşturmak için yeterli miktardadır.

**c) Uygulama geliştiricinin platform bilgisi:** Uygulama geliştirici platform hakkında yeteri kadar bilgi sahibi olmalıdır. Gerek platformun üzerinde çalıştığı donanımlar, gerek yazılımlar hakkında bilgi sahibi olmalı, ayrıca yazılım

<sup>17</sup> <http://jcp.org/en/jsr/detail?id=120>

<sup>18</sup> <http://jcp.org/en/jsr/detail?id=135>

geliştirmek için gerekli olan araçları ve yardımcı uygulamaları tanımalı ve kullanmalıdır. J2ME'in daha kısıtlı sistemler için geliştirildiğini göz önüne alıp, yazılım tasarımında da bellek ve işlemci gücünü mümkün olduğunca verimli kullanmaya çalışmalıdır. Diğer taraftan platform ile ilgili birtakım olası hatalar, güvenlik açıkları ve üreticiye bağlı kısıtlama, farklılık veya faydalardan haberdar olmalıdır [14], [27].

Bir cep telefonunun teknik veri dokümanlarında desteklendiği söylenen bir API ile ilgili bazı kısıtlamalar olabilir ve API'nin tüm işlevselliği kullanılamayabilir, örneğin, kamerası olmayan bir telefon, kamera kullanımını destekleyen bir API'yi destekliyorsa da bu API tam işlevselliğiyle kullanılamayacaktır. Cihaza özel farklılıklar olabilir, örneğin dosya sisteminde dosya ayırıcı karakterler farklı olabilir ('/' veya '\', veya daha başka bir karakter). Cihazda desteklenmeyen bir Java API'si yerine başka bir API kullanılarak uygulama geliştiricisine fayda sağlanmış olabilir. Örneğin, bazı Nokia marka telefonlarda kısa mesaj servisini kullanmayı sağlayan JSR-120 Telsiz Mesajlaşma Programcı Arayüzü yerine "Nokia SMS API"si kullanılmıştır. Bu durumda JSR-120'nin desteklenmemesi, cihazda kısa mesajlara erişilemeyeceği anlamına gelmez. Ayrıca üreticiler, cihazlarıyla ilgili bilinen sorunları internet sayfalarında uygulama geliştiricilerine bildirmekte ve kimi zaman sorunlara alternatif çözümler önermektedirler. Uygulama geliştiricinin bu tip olaylardan haberdar olması gerekmektedir.

**d) Cihaz üzerinde yeterli test yapılması:** Uygulama geliştiricisi, uygulamasının gerçek cihaz üzerinde de doğru bir şekilde çalıştığına emin olmalıdır. Mobil platformlar için geliştirilen yazılımlar genellikle öncelikle birtakım simülatörlerde denenir. Simülatörler sayesinde uygulama, geliştirilme sırasında cihaz yüklenmeden, hızlı bir şekilde kişisel bilgisayar üzerinde çalıştırılabilir, sonuçlar gözlenebilir ve hata ayıklama yapılabilir. Fakat kullanılan simülatörler her zaman gerçek cihazın özelliklerini tamamen aynı şekilde yansıtmayabilir. Özellikle hız ve kullanılan bellek miktarı konusunda farklılıklar yaşanabilir [27].

Nitekim bu uygulama geliştirilirken de "Nokia Prototype SDK 4.0" ve "Nokia S60 2nd FP2 MIDP SDK" simülatörleri kullanılmıştır. Simülatörler ile gerçek cihaz arasında önemli bir farka rastlanmamış olsa da hız konusunda gerçek cihaz ile simülatör arasında farklılıklar gözlemlenmiştir. Örneğin, "Sahilde duran motor

güzeldi.” cümlesinin sentezlenmesi Nokia 6630 cep telefonunda 14 saniye sürmekteyken, “Nokia Prototype SDK 4.0” simülatöründe 320 saniye sürmektedir.

## 6. GELİŞTİRİLEN UYGULAMA

Bu çalışmada, taşınabilir bir cihazda koşabilecek, J2ME platformunda çalışan bir MKS uygulaması geliştirilmiştir. Bu MKS uygulaması cep telefonuna gelen kısa mesajları sesli olarak okuyabilme yeteneğine sahiptir, aynı zamanda isterse kullanıcı sisteme el ile metin girişi yapabilir. Sistemde eksiksiz bir işlevselliğe sahip olmasa da bir dil çözümlemesi birimi gerçekleştirilmiş, birleştirerek sentezleme yöntemi kullanılmış, birleştirme sırasında difon parçaları veritabanı kullanılmıştır. Perde frekansı ve fonem uzunluğu ayarlamaları için TD-PSOLA metodu kullanılmıştır.

Sıradaki bölümde birleştirme sırasında sinyal işleme yöntemlerinin uygulanmadığı durumda ortaya çıkabilecek sorunlar incelenecektir. Daha sonraki bölümlerde de bu sorunları ortadan kaldırmak üzere geliştirilen uygulamadan bahsedilecektir.

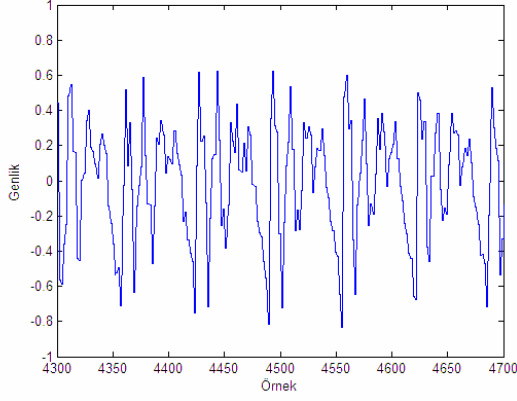
### 6.1. Birleştirme Sırasında Sinyal İşlemenin Gerekliliği

Bu noktada, ses parçalarını birleştirme için niçin özel metotlar veya algoritmalar geliştirildiği merak edilebilir. Her ne boyutta olursa olsun, önceden kaydedilmiş konuşma parçalarını, hiçbir işleme tabi tutmadan art arda yerleştirip konuşma sentezlenmeye çalışırsa ortaya sorunlu bir konuşma çıkacaktır. Bunun başlıca nedenleri şunlardır:

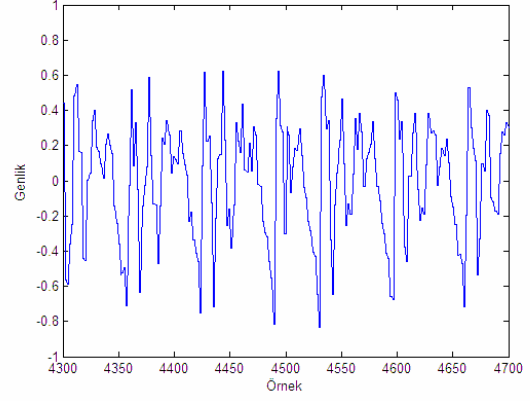
**a) Konuşma parçaları arasındaki faz uyumsuzluğu:** Konuşma parçaları birleştirilirken özellikle sedalı seslerde periyodiklik olduğu vurgulanmıştı. Fakat konuşma veritabanında bulunan parçalar arasında bulunacak faz uyumsuzluğu, bu periyodikliğin ortadan kalkmasına ve seste önemli bir bozulmaya yol açacaktır.

Şekil 6.1'de bu durumun sonucu gözükmemektedir. Şekil 6.1'de (a) orijinal sinyal, (b) faz farkı yaratılmış sinyaldir. Şekilde görülen kısım "ilker" kelimesinin "e" harfi dalga şeklinden bir parçadır. Birleştirme sırasında, parçalar arasındaki faz farkını incelemek için 4550. örnekten sonraki ilk negatif tepe noktası 25 birim öncesine kaydırılmış ve 3.125 milisaniyelik bir faz farkı yaratılmıştır. Bu ses dinlendiği zaman, faz farkının yaratıldığı noktada bir anlık bir hışırtı, anlık bir gürültü duyulmaktadır. Bu da ses kalitesinde kayıp anlamına gelmektedir.

Bu sorunu gidermek için konuşma parçalarındaki tepe noktaları senkronize edilir.



(a)

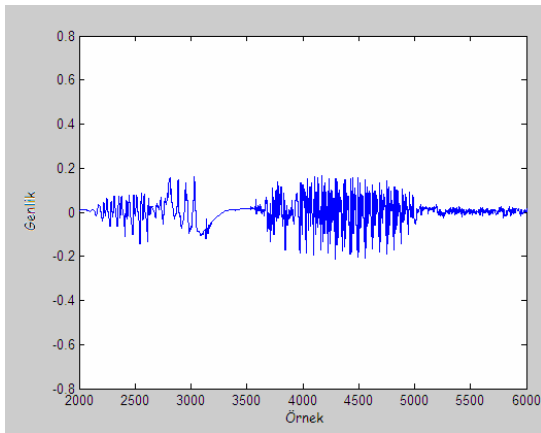


(b)

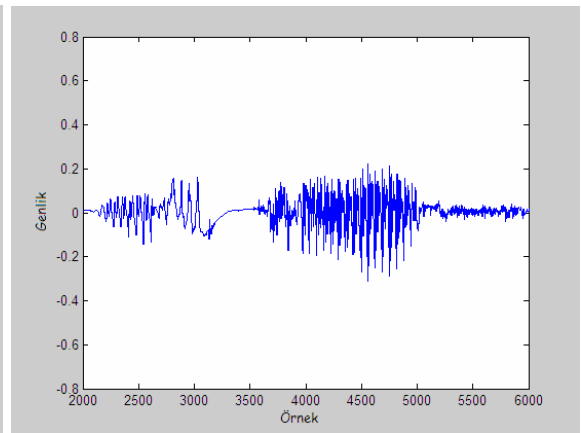
Şekil 6.1. Aralarında faz farkı olan iki konuşma parçasının birbirine eklenmesi. (a) orijinal sinyali, (b) değiştirilmiş sinyali göstermektedir.

**b) Konuşma parçaları arasındaki enerji seviyesi farklılığı:** Konuşma parçaları kaydedilirken bazı parçalar daha yüksek enerji seviyesiyle (daha yüksek genlikte), bazı parçalar da daha düşük enerji seviyesiyle (daha düşük genlikte) kaydedilmiş olabilir. Bunun sonucu olarak sentezlenen konuşmanın doğallığı azalır.

Şekil 6.2'deki grafikte, yine (a) orijinal sinyal, (b) değiştirilmiş sinyaldir. 4500. örnekten sonraki kısımların genliği %50 artırılmıştır. Böylece yine “e” harfinin bir kısmı daha düşük, bir kısmı daha yüksek enerjili olmaktadır. Bu durumda ses sinyali dinlendiği zaman anlaşılabilirlikte bir azalma yaşanmamıştır, fakat “e” harfi üzerinde vurgu duyulmuştur. Konuşma sentezlemesi sırasında daha da fazla parça birleştirildiği zaman, %50 seviyelerindeki enerji farklılıkları, cümlede istenmeyen vurgulanmalara yol açacaktır.



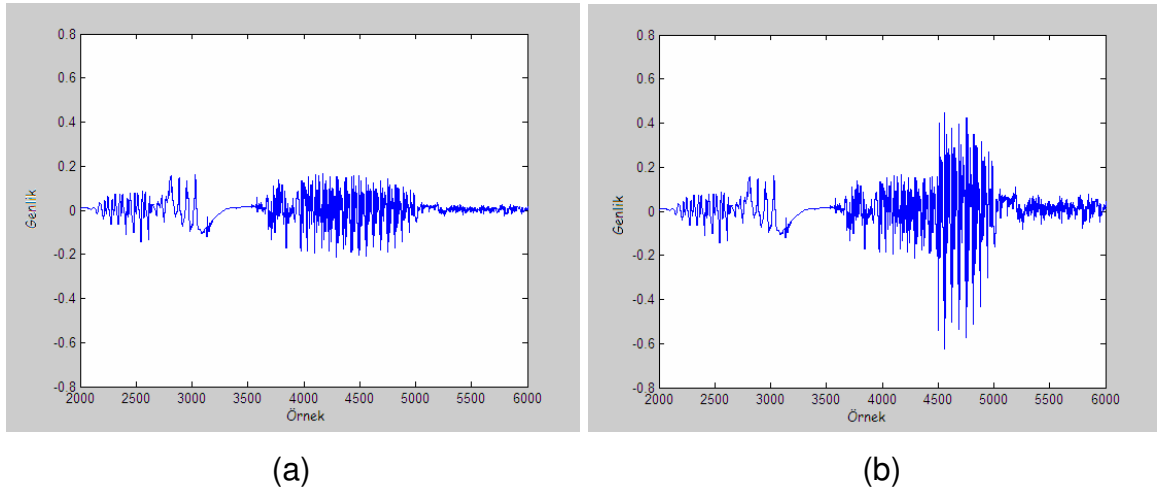
(a)



(b)

Şekil 6.2. 4500. örnekten sonra genliği %50 artırılmış konuşma sinyali. (a), orijinal sinyali, (b) değiştirilmiş sinyali göstermektedir.

Enerji farklılığın daha fazla olduğu durum incelenmek için bu sefer 4500. örnekten sonraki örneklerin genliği %200 oranında artırılmıştır. Bu durum Şekil 6.3'te gösterilmiştir. Şekil 6.3'te, yine (a) orijinal sinyal, (b) orijinal sinyalin değiştirilmiş halidir. 4500. örnekten sonraki parçalar için, genliği artırılmış sinyalin enerjisi, orijinal sinyalin enerjisinin 9 katıdır. Enerji farklılığın 9 kat ve daha fazla olduğu durumlarda kelime dinlenince enerji farklılığının rahatsız edici boyutlarda olduğu görülmüştür. Artık bu farklılık vurgu olarak nitelenemeyecek kadar kaba bir artış olmakta ve konuşma kesinlikle doğal hissedilmemektedir.



Şekil 6.3. 4500. örnekten sonra genliği %200 oranında artırılmış konuşma sinyali. (a) orijinal sinyali, (b) değiştirilmiş sinyali göstermektedir.

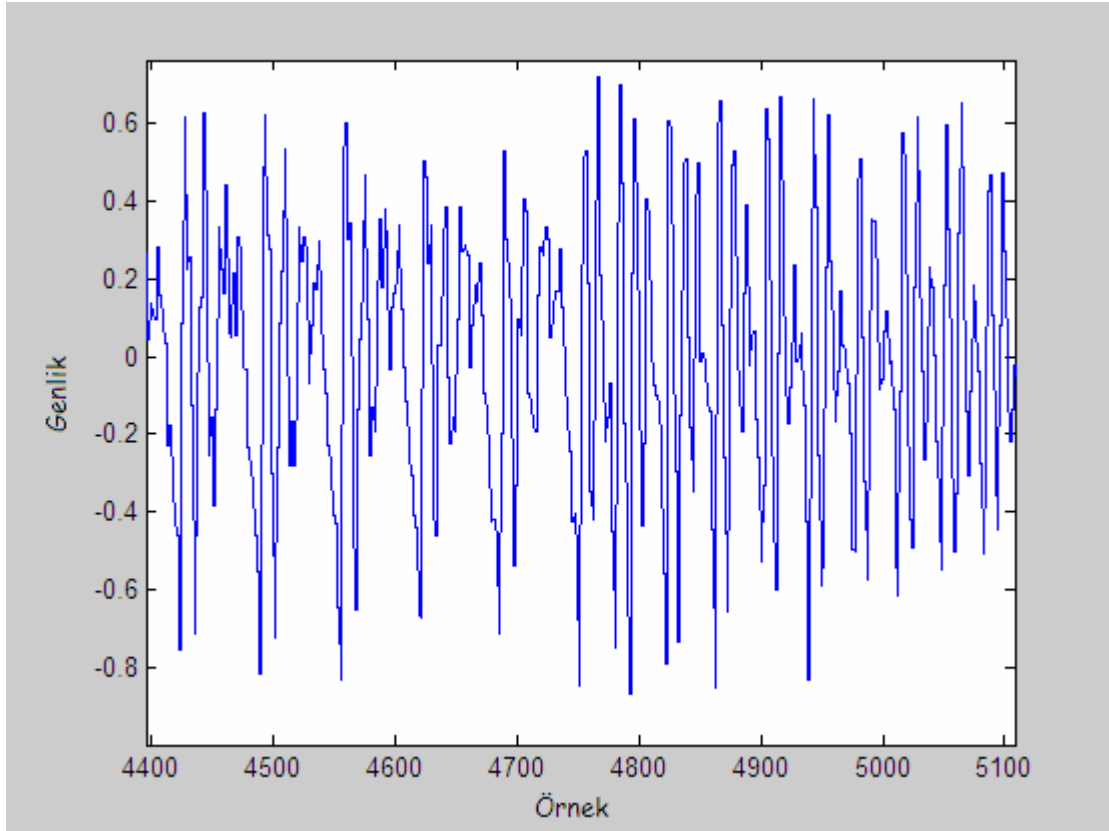
Bu sorunu engellemek için ise ya konuşma parçaları veritabanına eşit enerji seviyesiyle kaydedilmeli, ya da kullanılacak olan konuşma parçaları, kullanılmadan önce enerji normalizasyonuna tabi tutulmalıdır.

**c) Konuşma parçaları arasındaki perde frekansı uyumsuzluğu:** Önceden kaydedilen konuşma parçalarındaki perde frekansları her kayıt için farklı olabilecektir. Bu durumda birleştirilen parçalarda bazı kısımlarda farklı perde frekansı, diğer kısımda da farklı bir perde frekansı oluşur ve bu da konuşmanın anlaşılabilirliğini bozar.

Bu sorunun incelenmesi için "ilker" kelimesi iki farklı perde frekansında kaydedilmiş ve perde frekansı farklı "e" harfi bölgeleri perde frekansları değiştirilmeksizin art arda eklenmiştir. Elde edilen ses sinyalinin birleşim noktası Şekil 6.4'te gösterilmektedir. 4750. örnek birleşim noktasıdır. Sinyalin tepe noktaları arasındaki uzaklıkların 4750. örneğin öncesinde ve sonrasında farklı

olduğu görülebilir. 4750. örneğin öncesinde perde frekansı 123 Hz iken, 4750. örneğin sonrasında 224 Hz'tir.

Bu sinyal dinlendiği zaman, birleşim noktasından sonraki kısım insan kulağına, konuşma olarak nitelenemeyecek kadar farklı gelmektedir. Dinlenen ses bir "patlama" veya "çınılama" sesi olarak nitelendirilebilir ve sonradan eklenen perde frekansı yüksek olan kısmın, başka bir konuşmadan alınan parça olduğunun anlaşılması bile zordur.



Şekil 6.4. Perde frekansı farklı iki konuşma parçasını birleştirilmesi

Bu sorunun üstesinde gelmek için ise, Bölüm 3.2.3.2'de anlatılan metotlar kullanılarak konuşma sinyali perde frekansları ayarlanacak şekilde yeniden sentezlenmelidir.

## 6.2. Dil Çözümlemesi Birimi

Bölüm 3'te anlatıldığı gibi MKS sistemlerinde bir dil çözümlemesi birimi bulunur ve bu birim fonetik sembol dizisini sayısal sinyal işleme birimine gönderir. Sayısal sinyal işleme birimi de bu verileri kullanarak konuşmayı sentezler. Bu çalışma için yapılan dil çözümlemesi birimi aşağıdaki işlemleri yapmaktadır:



| <b>Harf</b> | <b>Cümle İçinde Ortalama Süre</b> |
|-------------|-----------------------------------|
| a           | 112                               |
| b           | 55                                |
| c           | 67                                |
| ç           | 105                               |
| d           | 47                                |
| e           | 105                               |
| f           | 71                                |
| g           | 48                                |
| h           | 52                                |
| ı           | 81                                |
| i           | 82                                |
| j           | 73                                |
| k           | 83                                |
| l           | 56                                |
| m           | 72                                |
| n           | 72                                |
| o           | 109                               |
| ö           | 110                               |
| p           | 76                                |
| r           | 60                                |
| s           | 112                               |
| ş           | 123                               |
| t           | 79                                |
| u           | 81                                |
| ü           | 84                                |
| v           | 52                                |
| y           | 45                                |
| z           | 80                                |

Çizelge 6.1. Türkçedeki harflerin milisaniye cinsinden ortalama süreleri

**a) Difon uzunlukları için uygun değerlerin bulunması:** DÇB'nin yapması gereken bürün ayarlamalarından birisi uygun fonem uzunluklarının seçilmesidir. Türkçedeki harf uzunlukları ile ilgili Şayli tarafından yapılmış bir çalışmada, harflerin cümle içindeki ortalama uzunluk değerleri bulunmuştur [26]. Şayli'nin çalışmasında bulunan, harflerin bir cümle içerisinde kullanıldıklarında aldıkları ortalama süre değerleri Çizelge 6.1'deki gibi sıralanmıştır. Bu tez sırasından geliştirilen uygulamada da cümle içerisindeki kelimelerin fonem uzunluğunun ayarlanması için Çizelge 6.1'deki değerler kullanılmıştır.

**b) Difonlar için perde periyotlarının seçilmesi:** Ergenç'in çalışmasına göre [11] perde frekansı değerleri Türkçede pek değişmemektedir. Bu yüzden konuşmanın anlaşılabilirliği üzerinde perde frekansı değişikliklerinin çok büyük etkisi olmamaktadır. Fakat çalışmamızda perde frekansı etkisini gözlemlemek için, yine Ergenç'in ve Abdullahbeşe'nin çalışmaları doğrultusunda, soru cümlelerinde, soru ekinden önceki kelimenin perde frekansında artış uygulanmıştır [1], [11]. Ayrıca Türkçedeki ünlülerin perde frekansları ve ilk üç formant frekansları üzerine yapılmış olan bir çalışma bulunmaktadır [16]. Bu çalışmadan elde edilen sonuçlar Çizelge 6.2'de gösterilmiştir. Geliştirilen uygulamada ünlüler için bu değerler kullanılmıştır.

| Ünlüler  | F0 (Hz) |       | F1 (Hz) |       | F2 (Hz) |        | F3 (Hz) |        |
|----------|---------|-------|---------|-------|---------|--------|---------|--------|
|          | Ort.    | SS    | Ort.    | SS    | Ort.    | SS     | Ort.    | SS     |
| <i>a</i> | 132     | 8,65  | 664     | 34,77 | 1081    | 19,63  | 2577    | 240,85 |
| <i>e</i> | 131     | 7,85  | 526     | 40,78 | 1772    | 69,26  | 2525    | 106,51 |
| <i>ı</i> | 145     | 8,53  | 355     | 40,64 | 1482    | 136,03 | 2405    | 159,62 |
| <i>i</i> | 151     | 9,37  | 278     | 13,94 | 2275    | 55,93  | 2570    | 106,08 |
| <i>o</i> | 132     | 10,06 | 432     | 28,40 | 811     | 60,00  | 2587    | 201,86 |
| <i>ö</i> | 141     | 12,95 | 427     | 24,08 | 1548    | 83,24  | 2346    | 78,70  |
| <i>u</i> | 149     | 9,28  | 295     | 25,70 | 786     | 97,36  | 2226    | 214,92 |
| <i>ü</i> | 155     | 9,81  | 279     | 56,05 | 1715    | 66,85  | 2288    | 69,03  |

Çizelge 6.2. Türkiye Türkçesindeki ünlülere ait, ortalama perde frekansı (F0), birinci (F1), ikinci (F2) ve üçüncü (F3) formant değerleri. SS: Standart sapma [16].

**c) Difonlar için enerji seviyesinin ayarlanması:** Ergenç [11] çalışmalarında Türkçede vurgunun genellikle son seslemede olduğunu belirtmiştir. Bu bilgi kullanılarak kelimelerde son hecede enerji seviyesi artırılmıştır.

**d) Bazı kısaltmalar için destek:** DÇB'nin kısaltmaları uygun olarak açması beklenir. Bütün kısaltmaların okunabilmesi bu çalışmanın kapsamını aşmaktadır. Fakat yine de böyle bir olayın gösterimi için seçilen bazı kısaltmaların doğru şekilde konuşmaya çevrilebildiği gösterilmiştir. Bunun için seçilen bazı kısaltmalar bir dosya içine kaydedilmiştir ve program çalışma esnasında girilen kelimeler içerisinde bu kısaltmaların olup olmadığını kontrol etmektedir, eğer sözlükte bulunan bir kısaltma bulunursa, kısaltmanın karşılığı okunmaktadır. Bu kısaltmalardan bazıları Çizelge 6.3'te gösterilmiştir. Program kısa mesajlar ile ilgilendiği için kısa mesaj yazımı sırasında kullanılan bazı kısaltmalara da yer verilmiştir. Bu kısaltmaların bulunduğu sözlük dosyası istenildiğinde kolaylıkla genişletilebilir.

| Kısaltma | Açılımı         |
|----------|-----------------|
| slm      | selam           |
| nbr      | ne haber        |
| kib      | kendine iyi bak |
| dr       | doktor          |

Çizelge 6.3. Sistemde kullanılan kısaltmalardan örnekler

**e) Karmaşık metinler:** Sisteme girilen ve harf ve rakamların birleşiminden oluşan kelime parçaları, harf harf ve rakam rakam okunmaktadır. Örneğin "r4e5" metni "re dört e beş" şeklinde okunmaktadır.

**f) Noktalama işareti duraklamaları:** Cümlede geçen her nokta için 500 ms., her noktalı virgöl için 400 ms., ve her virgöl ve iki nokta işareti için 200 ms. duraklama eklenmiştir.

**g) Kelimeler arası duraklamalar:** Cümledeki kelimeler arasında 200 ms. aralık bırakılmıştır. Kelime aralarında nokta veya virgöl varsa, bu işaretlerden gelen bekleme süreleri de eklenir.

**h) Yumuşak G:** ‘ğ’ karakteri için Bölüm 4’te anlatıldığı üzere herhangi bir ses kullanılmamaktadır. Bu çalışmada ise ‘ğ’ karakterinden önceki fonemin uzunluğu artırılarak ‘ğ’ işleme konulmuştur. Veritabanında olası bir ünlü kaymasında kullanılacak difon bulunmadığı için de ‘ğ’ geçen kelime aralıksız bir şekilde iki kelimeye ayrılmıştır. Bu işlem için bir örnek “Kağrı” kelimesi için Çizelge 6.6’da gösterilmiştir.

**i) Girilen metin için hangi difonların kullanılacağına karar verilmesi:** Bilindiği gibi Türkçede bazı harflere karşılık gelen bir yada iki fonem olabilir (bkz. Ek C). Yazıda geçen ve iki farklı fonem karşılığı olan kelimelerin kullanıldığı durumlarda kullanıcı istediği bir fonemi kullanarak konuşma sentezletmek isteyebilir. Bu gibi durumlar için uygulamada kullanıcıya el ile fonem seçimi desteği geliştirilmiştir. İstenilen fonemin seçilmesi için, harf girildikten sonra “#” veya “&” karakterlerinden birisi yazılır. Burada ‘#’, veritabanında bulunan ve genellikle “ince” olarak nitelenebilecek, “predorsal” ve “ön” olarak tanımlanan fonemleri, ‘&’ ise veritabanında bulunan ve genellikle “kalın” olarak nitelenebilecek, “postdorsal” ve “arka” olarak tanımlanan fonemleri seçmeye yararmaktadır. İstisnai olarak, üç difona karşılık genel tek harf “r” harfidir, ama “r” harfinin hangi fonem ile seslendirileceği tamamen kelime içerisindeki durumuna bağlıdır [11]. Kelimenin başında, ortasında ve sonunda bulunan “r” harfleri için belirli fonemler kullanıldığı için “r” harfi için herhangi bir işleme gerek olmadan doğru fonem seçimi zaten yapılmış olmaktadır. Çizelge 6.4’te “kar” kelimesindeki ‘a’ harfi fonem seçimi için bu kullanım şekli örneklendirilmiştir. Eğer bu özel karakterler kullanılmazsa program difon veritabanı listesinde bulduğu ilk difonu seçecektir.

| Yazım Şekli | Seçilen ‘a’ fonemi    | Kelimenin Anlamı  |
|-------------|-----------------------|---|
| kar         | <listedeki ilk fonem> | <kestirilemez>  |
| ka#r        | a                     | Alışveriş işlerinin sağladığı para kazancı.                       |
| ka&r        | α                     | Havada beyaz ve hafif billurlar biçiminde donarak yağan su buharı |

Çizelge 6.4. “kar” kelimesi için kullanılacak ‘a’ fonemini el ile seçirme.

Seçim yapılırken bir önceki difonun ikinci bileşeni ile, sonraki difonun birinci bileşeninin aynı olmasına dikkat edilir. Örneğin "bunalım" kelimesinde 'na' parçasının ikinci yarısı için "α" fonemi kullanılmışsa bunun ardından "al" parçasının ilk yarısı için yine "α" fonemine sahip bir difon seçilmeye çalışılır. Eğer "α" fonemine sahip bir difon bulunamazsa sentezlenen konuşmadaki kalite kaybı kabul edilerek "a" fonemine sahip bir difon kullanılır.

Normalde bir Dil Çözümleme Biriminde Bölüm 3'te anlatıldığı şekilde daha fazla işlevsellik beklenebilir. Fakat DÇB geliştirilmesi işi kendi başına bir çalışma alanıdır, sadece DÇB üzerine olan çalışmalar bulunmaktadır [1], [20], [26] ve bu tezin kapsamını aşmaktadır. Bu yüzden işlevsellik yönünden eksiksiz bir DÇB tasarlanamamıştır.

### **6.3. Sayısal Sinyal İşleme Birimi**

Bölüm 3'te anlatıldığı gibi, MKS sistemlerinde, DÇB'nin ürettiği fonetik sembolleri konuşma sinyaline çeviren bir Sayısal Sinyal İşleme Birimi bulunur. Bu tez kapsamında geliştirilen uygulamada birleştirerek sentezleme yapan, birleştirmede difon parçalarını kullanan, fonem uzunluğu ve perde frekansı değişikliği için TD-PSOLA yöntemini kullanan bir Sayısal Sinyal İşleme Birimi geliştirilmiştir.

TD-PSOLA yöntemi, Bölüm 3.2'de anlatılan diğer yöntemlere göre oldukça düşük işlem yükü isteyen bir yöntemdir. LP-PSOLA yöntemine göre yaklaşık 10 kat daha az işlem gerekmektedir [7] ve bu yüzden işlem gücü kısıtlı olan taşınabilir sistemler için uygundur. Kurallı sentezleyiciler ve artikülator sentezleyiciler ise daha da yüksek işlem gücü ihtiyaçları sebebiyle tercih edilmemişlerdir.

#### **6.3.1. Parça Seçimi**

Bu çalışmada geliştirilen MKS sistemi birleştirerek sentezleme yapmaktadır. Birleştirilen parçalar da difonlardan oluşmaktadır. Difon seçiminin sebebi, veritabanında az yer kaplamalarıdır. Difon veritabanının az yer kaplaması, uygulamanın koşacağı platformun kısıtlamaları göz önüne alındığında son derece önemlidir. Daha az yer kaplayabilecek fonem veritabanları mümkünse de, daha önce açıklanan sebeplerden dolayı neredeyse kullanılmamaktadır. Veritabanı boyutunun büyüklüğü, taşınabilir bir cihaza programı yüklerken ve programın çalışması sırasında kullanılan bellek miktarı açısından sorun yaratabilmektedir.

Hemen her taşınabilir cihazın kabul edebileceği midlet için belirli fiziksel sınırlar vardır (Örn. Nokia 6220 cep telefonu için 63 KB). Diğer taraftan, program çalışması sırasında büyük veritabanı dosyalarının belleğe alınması için daha fazla zaman gerekecektir, işlem gücü sınırlı bir cihazda bu gecikmeler büyük miktarda olup sıkıntı yaratabilir. Ayrıca veritabanı program tarafından okunduktan sonra bellekte belirli bir miktar yer kaplamaktadır, birçok cep telefonunda yığın boyutu birkaç yüz kilobayt seviyesindedir (Örn. Nokia 6220 için 229 KB). Bu sebeplerden ötürü difon veritabanı boyutu bu uygulamada hayati önem taşımaktadır.

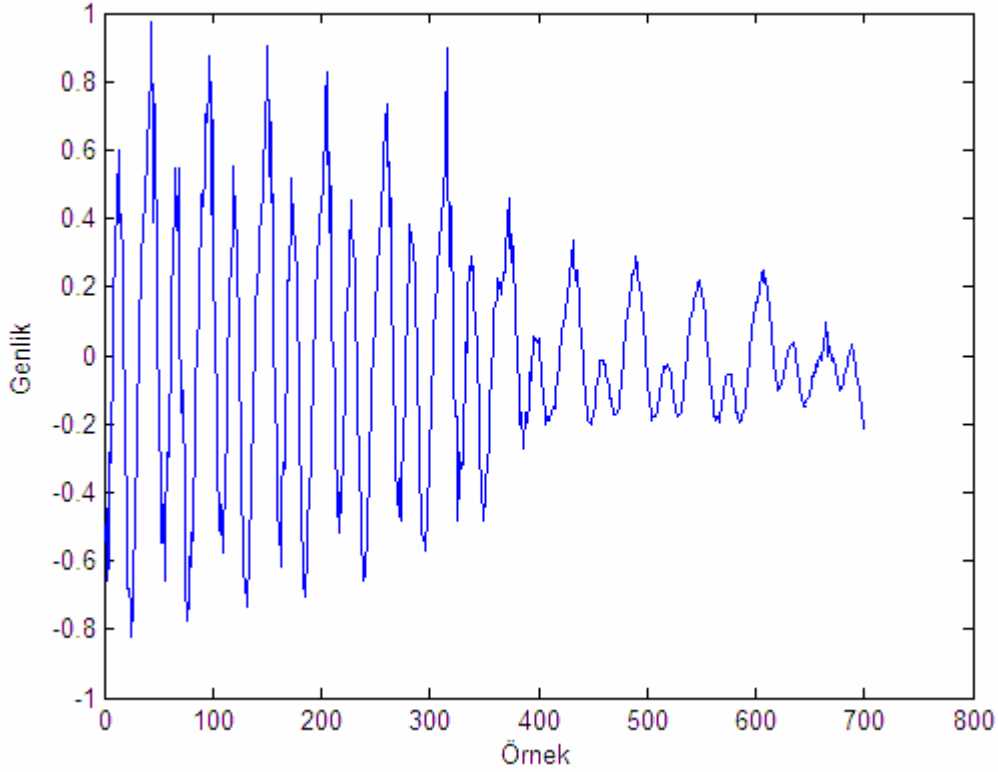
Bu çalışmada kullanılan veritabanı 265 difon içermektedir ve yaklaşık 950 KB kadar yer kaplamaktadır, bu da kullanılan donanım için uygundur. Veritabanında kullanılan difonlar Ek E'de listelenmiştir. Daha düşük perde frekansına sahip olan erkek sesinin işlenmesi daha kolay olduğundan veritabanında erkek sesi tercih edilmiştir ve örnekleme frekansı 8 KHz'dir. Kaydedilen seslerdeki tepe noktası işaretlemeleri elle yapılmıştır. Belirgin bir periyodikliğe sahip olmayan sedasız sesler için 50 örnek aralıklarla tepe noktası işaretlemesi yapılmıştır. Bu çalışmada kullanılan veritabanı, Özen'in çalışmasında [22] kullandığı veritabanının 30 adet difon eklenerek geliştirilmiş bir sürümüdür. Veritabanında bulunan 'i2l1' difonuna ait dalgaşekli Şekil 6.5'te gösterilmiştir. Veritabanı oluşturulması sırasında kullanılan dosya formatı RIFF dosya formatıdır. Dosya formatı ile ilgili diğer ayrıntılar Ek F'de gösterilmiştir.

### **6.3.2. Perde Frakansı ve Uzunluk Değişimi**

TD-PSOLA yöntemi, hem ezgi değişikliğini, hem de uzunluk değişikliğinin yapılmasını sağlayan bir yöntemdir. Bölüm 3.2.3.2.1'de açıklandığı gibi, tepe noktaları olan perde işaretleri etrafında iki periyotluk Hanning pencereleri yerleştirilmiş ve elde edilen pencerelenmiş sinyal parçaları daha sonra birleştirilerek konuşma sentezlenmiştir. Birleştirme sırasında pencerelenmiş parçalar arasındaki uzaklık istenen perde periyodu kadar ayarlanmış ve böylece perde frekansı değişikliği elde edilmiştir.

Difon uzunluğunun artması gereken durumlarda difonun en ucundaki pencerelenmiş parçalar yeniden eklenerek fonemin uzunluğu artırılmıştır. Difonların en uç noktaları, fonemlerin orta noktalarına denk geldiği için bu bölge fonemin en durağan olduğu bölgedir. Yine aynı sebepten, difonun uzunluğunun

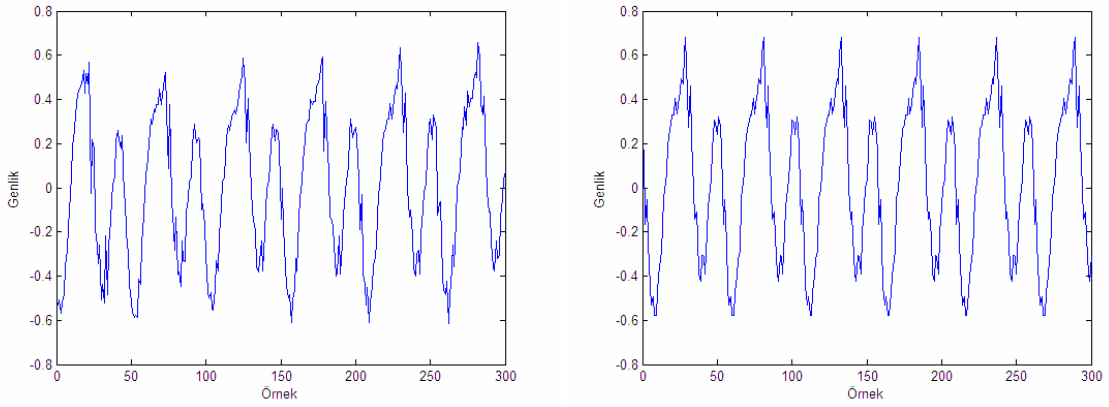
azaltılması gerektiği zaman da, difonun uç bölgesindeki parçalar atılarak difon süresi kısaltılmıştır. İnsan kulağı difonların birleşim bölgelerine daha duyarlı olduğu için difonların birleşim bölgelerinden atılacak parçalar dinleyiciyi daha fazla rahatsız edecektir.



Şekil 6.5. Veritabanında bulunan 'i2l1' difonu

Şekil 6.5'teki difonda, 1-6 arasındaki tepe noktaları 'i2' difonuna, 7-12 arasındaki tepe noktaları 'l1' difonuna aittir. Burada konuşma sentezlenirken, "i" difonunun yapılandırılmasında 6. tepe noktası etrafından başlanır. Daha sonra sırasıyla 5., 4., 3., ve 2. tepe noktası etrafındaki pencerelenmiş örnekler kullanılır. Eğer hala istenilen difon uzunluğuna erişilmemişse 2. tepe noktası etrafındaki örnekler kullanılmaya devam edilerek fonem uzatılır. Burada 1. tepe noktası etrafındaki örnekler kullanılmamaktadır, bunun sebebi yerleştirilen pencerenin boyutlarının, şekilde gösterilen çerçevenin dışına taşmasıdır. Zaten bu veritabanının hazırlanması sırasında fonemin durağan bölgesinden en az üç periyotluk örnek alınmasına dikkat edildiği için [22] en uçtaki tepe noktalarının kullanılmaması sorun yaratmamaktadır. 'l1' difonunun işlenmesi de benzer şekilde, 7. tepe noktasından başlayarak, 11. tepe noktasına doğru yapılmaktadır.

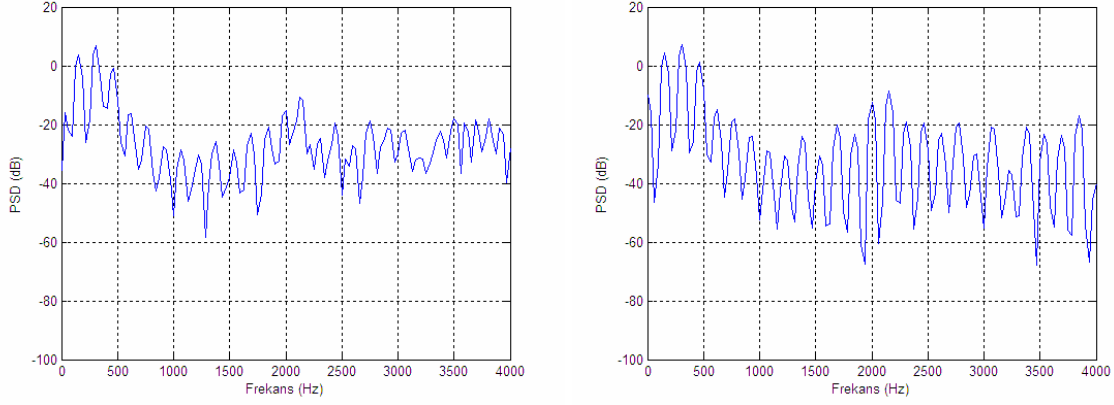
Bölüm 3.2.3.2.1’de anlatıldığı üzere, TD-PSOLA yöntemi ile periyodik bir sinyal önce pencerelenip, daha sonra pencerelenmiş örnekler aynı perde periyodu aralıklarla üst üste örtüştürülüp toplanıp birleştirilerek yeniden elde edilebilir. Bu olay Şekil 6.6’da incelenebilir. Şekil 6.6’da soldaki grafikte konuşma veritabanından alınan ‘e’ sesi işlenmemiş olarak zaman ekseninde gösterilmiştir. Soldaki grafikte de aynı sesin, TD-PSOLA yöntemiyle yine aynı perde frekansı ile sentezlenmesi sonucunda ortaya çıkan sinyal bulunmaktadır. Grafiklerden anlaşılacağı üzere iki sinyal de zaman ekseninde oldukça benzerdirler.



Şekil 6.6. İşlenmemiş (solda) ve TD-PSOLA yöntemiyle aynı perde frekansı ile yeniden sentezlenmiş (sağda) ‘e’ ünlüsünün zaman ekseninde gösterimi.

Benzer bir durum bu konuşma parçalarının izgeleri için de geçerlidir. Şekil 6.7’de soldaki grafikte konuşma veritabanından alınan ‘e’ sesinin işlenmemiş izgesi gösterilmiştir. Sağdaki grafikte de aynı sesin, TD-PSOLA yöntemiyle yine aynı perde frekansı ile sentezlenmesi sonucunda ortaya çıkan izgesi bulunmaktadır. Yine grafiklerden anlaşılacağı üzere iki sinyal izge bakımından da oldukça benzerdirler. Özellikle konuşmanın anlaşılabilirliği açısından önemli olan ilk üç formant bileşeninin benzerlikleri dikkat çekicidir.





Şekil 6.7. İşlenmemiş (solda) ve TD-PSOLA yöntemiyle aynı perde frekansıyla yeniden sentezlenmiş (sağda) 'e' ünlüsünün izgeleri.

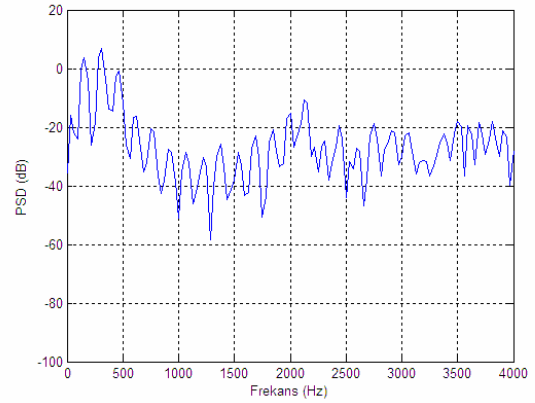
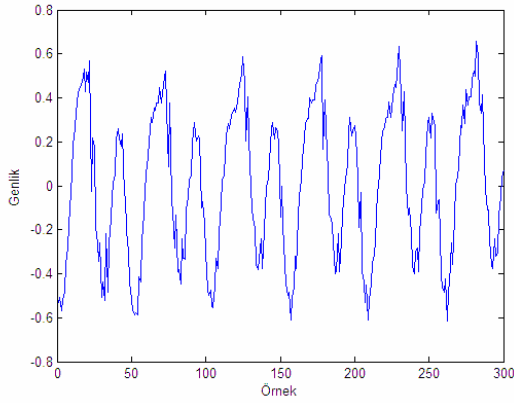
TD-PSOLA yönteminin asıl amacı perde frekansını değiştirmektir. Yapılan denemelerde elde edilen sonuçlar Şekil 6.8'de ve Şekil 6.9'da gösterilmiştir. Şekil 6.8'de TD-PSOLA yöntemi kullanılarak 'e' foneminin perde frekansının %10, %20 ve %30 oranında azaltılması sonucu elde edilen sinyaller zaman ekseninde ve izge olarak gösterilmiştir. Şekil 6.9'da ise yine TD-PSOLA yöntemi kullanılarak 'e' foneminin perde frekansının %10, %20 ve %30 oranında artırılması sonucu elde edilen sinyaller yine zaman ekseninde ve izge olarak gösterilmiştir.

Hem Şekil 6.8'de, hem de Şekil 6.9'da sinyalin zarfında ve izgesinde önemli bir bozulma olmadan sinyalin yeni perde frekanslarıyla sentezlendiği görülmektedir. Ancak perde frekansının değiştirilme oranının büyümesiyle hem zaman ekseninde, hem de izge açısından bozulmaların daha belirgin hale geldiği de açıktır. Şekil 6.8 (a)'da gösterilen orijinal (işlenmemiş) "e" foneminin dalga şekli ve izgesi ile Şekil 6.8 (b)'de gösterilen perde frekansı %10 azaltılmış "e" foneminin dalga şekli ve izgesi birbirine oldukça benzemektedir. Fakat Şekil 6.8 (a)'da gösterilen orijinal (işlenmemiş) "e" foneminin dalga şekli ve izgesi ile Şekil 6.8 (c)'de gösterilen perde frekansı %20 azaltılmış "e" foneminin dalga şekli arasında farklılıklar gözlemlenmeye başlamıştır. Şekil 6.8 (c)'de de %30'luk perde frekansı azaltımı sonucunda artık hem sinyalin dalga şekli, hem de izge orijinal sinyalden oldukça farklı gözükmemektedir. Şekil 6.9'da da benzer şekilde perde frekansındaki düşük oranlı değişimlerde az, yüksek oranlı değişimlerde de fazlaca bozulma olduğu gözlemlenebilir. Bu grafiklere bakılarak, TD-PSOLA yönteminin %10 gibi düşük oranlı perde frekansı değişimlerinde başarılı olduğu, ancak %20 gibi daha yüksek

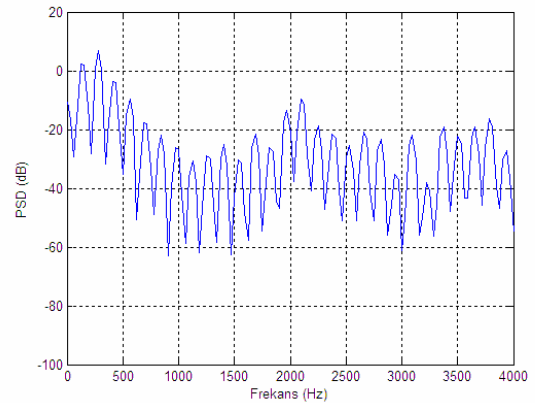
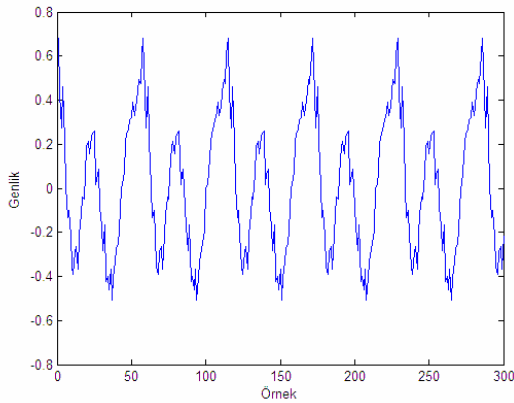
oranlardaki perde frekansı deęişimlerinde sinyalin dalga řekli ve izgesinde bozulmalar yařanmaya bařladığı ve %30'luk bir perde frekansı deęiřimi sonucunda ise hem sinyalin dalga řeklinde hem de izgesinde önemli bozulmalara yol ađtığı sonucu çıkarılabilir. Bu yorum, perde frekansının hem azaltılmasında, hem de artırılmasında geçerlidir.

Sentezlenen bu konuşmalar dinlendiğinde ise %10'luk perde frekansı artırım ve azaltımında konuşmanın doęallığını ve anlaşılabilirliğini koruduęu görülmüřtür. Fakat %30'luk perde frekansı artırım ve azaltımı sonucunda oluřan sinyalde ise çok belirgin bir sentetiklik duygusu bulunmaktadır, fakat yine de konuşmanın zor da olsa anlaşılabilir olduęu söylenebilir.

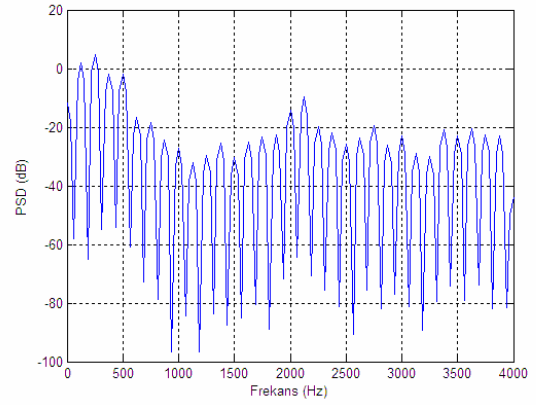
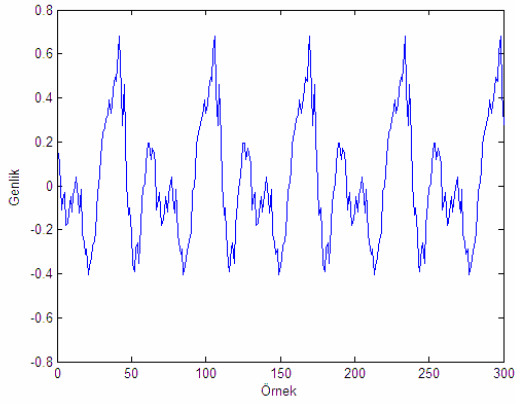
Perde frekansı deęiřtirilmesi iřleminin orijinal sinyalin izgesini mümkün olduęu kadar bozmaması istenir. Bu problem için konuşma veritabanında farklı perde frekansı ile kaydedilmiş konuşma parçaları bulundurulması önerilmektedir [22]. Böylece perde frekansları deęiřtirilirken perde frekansları daha düşük bir oranla deęiřtirilebilecek ve sentezlenen konuşmalardaki bozulma azalacaktır.



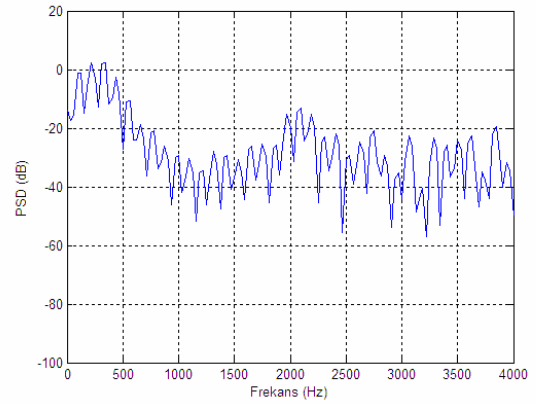
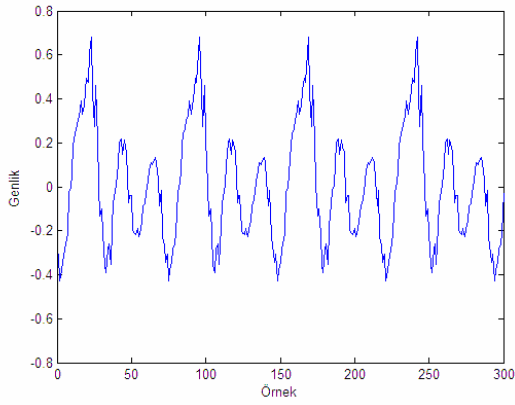
(a)



(b)

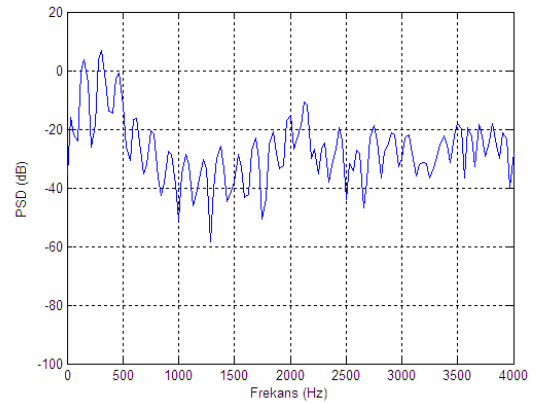
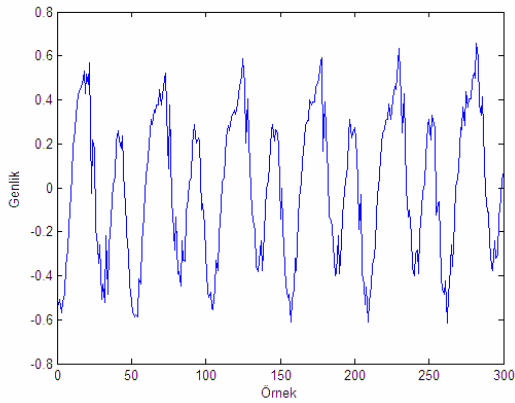


(c)

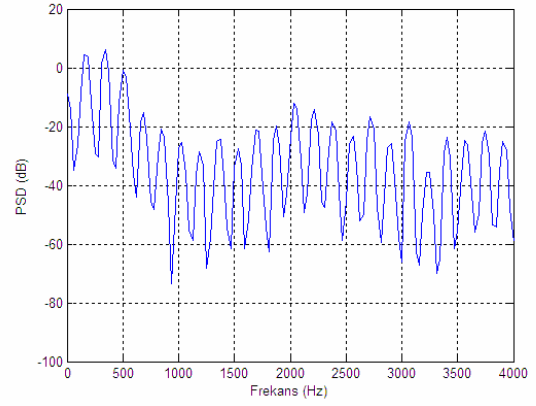
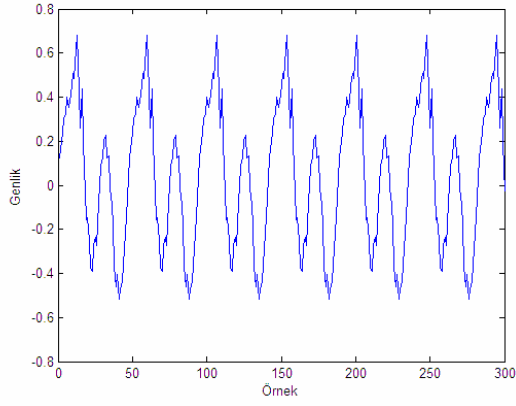


(d)

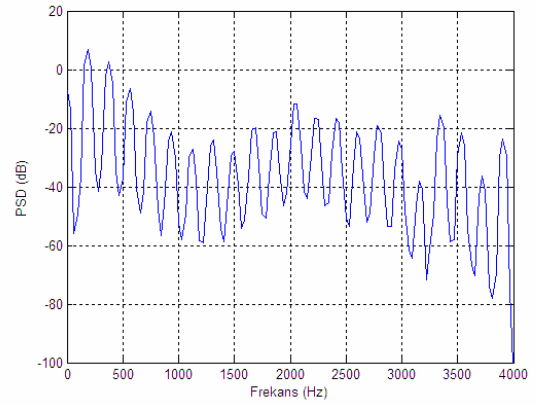
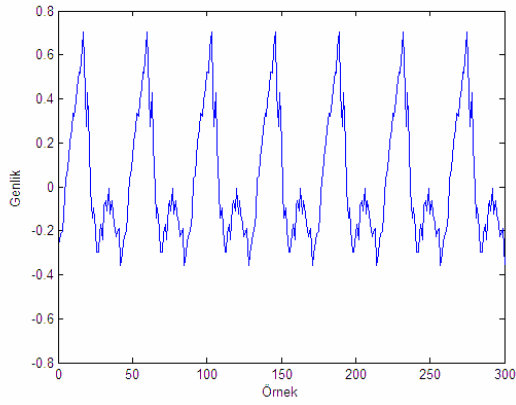
Şekil 6.8. (a) İşlenmemiş 'e' fonemi dalga şekli ve izgesi ; (b) Perde frekansı %10 azaltılmış 'e' fonemi dalga şekli ve izgesi; (c) Perde frekansı %20 azaltılmış 'e' fonemi dalga şekli ve izgesi (d) Perde frekansı %30 azaltılmış 'e' fonemi dalga şekli ve izgesi



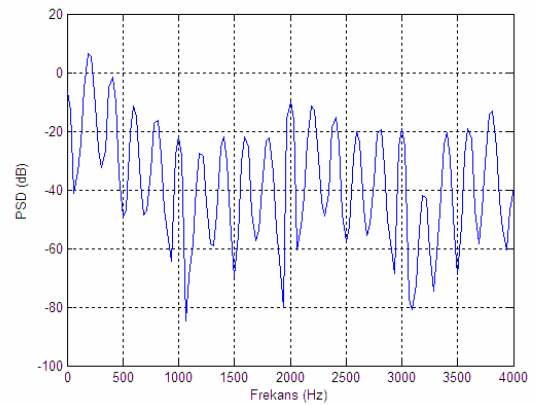
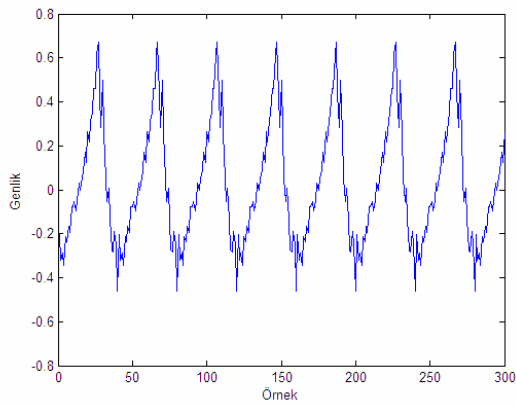
(a)



(b)



(c)



(d)

Şekil 6.9. (a) İşlenmemiş 'e' fonemi dalga şekli ve izgesi ; (b) Perde frekansı %10 artırılmış 'e' fonemi dalga şekli ve izgesi; (c) Perde frekansı %20 artırılmış 'e' fonemi dalga şekli ve izgesi (d) Perde frekansı %30 artırılmış 'e' fonemi dalga şekli ve izgesi

### **6.3.2.1. TD-PSOLA Pencereleme Bölgesi Seçimi**

Bölüm 3.2.3.2.1’de anlatılan kısımda, TD-PSOLA yönteminin uygulamasının gösterildiği sinyal sabit bir perde frekansına sahipti, bir diğer deyişle tepe noktaları arasındaki mesafe aynıydı. Fakat gerçek sinyallerde durum böyle değildir. Örneğin Şekil 6.5’te verilen difonun ‘i2’ parçasının tepe noktaları arasındaki uzaklık “54 – 56 – 57 ...” şeklinde gitmektedir. Bu durumda pencerelerin uygun şekilde yerleştirilmesinde bir ikilem ortaya çıkmaktadır. Eğer pencere tepe noktaları, sinyaldeki tepe noktaları ile örtüştürülürse, pencere, birbirine yakın olan tepe noktalarında sorun yaratacak, yakın olan komşu tepe noktalarını açacaktır. Bu durumun sentezlenen konuşmada bozulmaya yol açtığını Salor [23] çalışmalarında göstermiştir.

Bu problemin çözülmesi için TD-PSOLA’nın tanımdan biraz sapılarak, tepe noktalarının pencerenin tam ortasına denk getirilmesi yerine, pencere uzunluğu, ilgili tepe noktasının komşu tepe noktaları arasındaki uzunluk kadar seçilir ve pencere tam olarak komşu tepe noktaları arasına sığdırılır. Bu durumda pencerenin tepe noktası, ortadaki sinyal tepe noktasıyla örtüşmez ama böylece pencerelenen konuşma parçalarının en uç bölgelerinde oluşan bozulmalar engellenmiş olur.

### **6.3.3. Enerji Seviyesinin Değişimi**

Bu çalışmada enerji seviyesi değişimi iki sefer olmaktadır. Bunlardan birincisi Bölüm 3’te anlatıldığı üzere veritabanından alınan fonemlerin enerjilerinin eşitlenmesidir. Bu işlem Bölüm 3.2.3.2’de anlatıldığı gibi, kullanılacak olan parçaların enerjilerinin eşitlenmesi için genliklerinin değiştirilmesi şeklinde yapılmaktadır.

Enerji seviyesiyle ilgili ikinci işlem ise vurgu için enerji seviyesinin artırımıdır. Bölüm 3.2.3.2’de incelenen yöntemdeki enerji hesaplama sırasında kullanılacak her örnek için pencereleme işlemi ve pencerelenmiş alanın enerjisinin hesaplanması işlemleri sistemdeki işlemci gücünü zorlayacaktır. Diğer taraftan bu çalışmada herhangi bir enerji eğrisi kullanılmayıp sadece son hecede vurgu ekleneceği için, Bölüm 3.2.3.2’de anlatılan yöntem kullanılmayıp, sadece

sentezlenmiş olan konuşma sinyalinde vurgu eklenmek istenen bölgenin genliği %30 oranında artırılmıştır.

#### 6.4. Kullanılan Algoritma

Geliştirilen program yukarıda belirtilen işlemleri aşağıdaki şekilde belirtilen sırayla yapmaktadır:

- a) Önce sisteme girdi olarak verilen metin çözümlenir, yani karakter öbeklerine (jetonlarına) ayrılır. Bu uygulamada boşluk karakteri karakter öbeği ayırıcı olarak kullanılmaktadır. Çözümleme işleminden sonra oluşan karakter öbekleri, oluşturulan nesnelere saklanır. Burada nesne kullanılmasının sebebi, bu sayede yalnızca çözümlenmiş metinlerin değil, bunların yanı sıra ileri aşamalarda kullanılmak üzere, sonraki karakterin de (nokta, virgül veya soru işareti gibi) bu nesnelere dahilinde saklanabilmesidir.
- b) Karakter öbekleri ayrıldıktan sonra kelimelere dönüştürülür. Kelimelere dönüşme sırasında rakamlar, sözlük dosyasından metin karşılıkları bulunarak metin olarak ifade edilirler, kısaltmaların da yine sözlük dosyasında karşılıkları bulunarak açık halleri olan kelimeler kullanılır. Harf ve rakam kombinasyonlarından oluşan kelimeler de bu aşamada Bölüm 6.2'de anlatıldığı gibi ayrıştırılır.
- c) Kelimeler harf ikililerine ayrıştırılır. Daha önce bahsedildiği gibi difonlar, iki yarı fonemden oluşmaktadırlar. Bu aşamada oluşturulan harf ikililerinin daha sonra difon karşılıkları bulunacaktır. Bu aşamada genel olarak kelime içerisindeki her ardışık harf çifti belirlenerek sırayla kaydedilir. '#', ve '&' karakterlerine sıradan harf gibi davranılmaz. Bu karakterler, diğer harflere yapışık olarak kaydedilir. Bu aşamada yapılan işlem için birkaç örnek Çizelge 6.5'te verilmiştir.

| Kelime  | Harf İkili Dizisi Karşılığı     |
|---------|---------------------------------|
| örnek   | _ö + ör + rn + ne + ek + k_     |
| örn#ek  | _ö + ör + rn# + n#e + ek + k_   |
| örn#e&k | _ö + ör + rn# + n#e& + e&k + k_ |

Çizelge 6.5. Kelimelerin harf ikililerine ayrılması için örnekler

- d) Fonemler üzerinde düzeltmeler yapılır. 'ğ', Bölüm 6.1'de anlatıldığı gibi kaldırılır, yerine kısa süreli boşluk konur ve önceki fonemin süresi uzatılır. Ayrıca yine bu aşamada 'xp', 'xk', 'xt' gibi difonların önüne boşluk konur. Bu bölümde yapılan işlemlerle ilgili örnekler Çizelge 6.6'da verilmiştir.

| Harf İkili Dizisi           | Düzeltilmiş Harf İkili Dizisi    |
|-----------------------------|----------------------------------|
| _k + ka + ağ + ğn + nı + ı_ | _k + ka + a_ + _n + nı + ı_      |
| _ö + ör + rn + ne + ek + k_ | _ö + ör + rn + ne + e_ + _k + k_ |

Çizelge 6.6. Harf ikililerinde düzeltmeler

- e) Bu bölümde harf ikilileri için uygun difonlar seçilir. Bölüm 6.1'de açıklanan kurallar doğrultusunda veritabanından hangi difonların kullanılacağı belirlenir. Veritabanında bulunan difonların konuşma örnekleri veritabanından okunur ve konuşma sinyallerinin saklanacağı ilgili nesnelere oluşturulur. Yine Bölüm 6.2'de sözü edilen, noktalama işaretleri ve kelimeler arası için eklenen boşluk zamanları da tıpkı difon nesnelere gibi tanımlanır.
- f) Bu bölümde difonlar için bürünsel değerler belirlenir. Bürünsel değerler belirlenirken Bölüm 6.2'de anlatılan kurallar uygulanır. Bir önceki aşamada oluşturulan difon nesnelere, bürün değerlerinin saklanması için bazı üyeler tanımlanmıştır. Bu üyeler için uygun değerler atanır. Böylece DÇB görevini tamamlamış olur.
- g) Bir sonraki aşamada sayısal sinyal işleme modülü bulunmaktadır. Sayısal sinyal işleme modülü öncelikle difon nesnelere içinde bulunan bilgileri okuyup, TD-PSOLA yönteminde kullanılacak, tepe noktaları etrafında Hanning penceresiyle pencerelenmiş örnekleri oluşturur. Daha sonra difonda belirtilen fonem uzunluğunu ve perde frekansını dikkate alarak bu parçalardan kaç adet gerektiğini hesaplar. Bölüm 6.3.2'de anlatılan yöntem ile pencerelenmiş örnekleri sırasıyla sentezlenecek konuşma sinyaline eklemeye başlar. Bütün parçaların eklenmesiyle de konuşma sentezlenmiş olur. En son olarak da, enerji değişimi için Bölüm 6.2'de anlatılan şekilde oluşturulmuş konuşma sinyalinde uygun bölgelerin genliği artırılır.

h) Bölüm 6.5'te açıklanacağı üzere, cep telefonunda sentezlenen bir dalga şeklinin cihazda sesli olarak çalınabilmesi için bu dalga şeklinin Dalgaşekli Ses Biçimi (WAV) formatına dönüştürülmesi gerekmektedir. Bu işlem için geliştirilen bir metod sentezlenen konuşmayı WAV formatına dönüştürür. 8-bit örnekleme yapılmış olan WAV dosyalarında örnekler 0...255 aralığında ve tamsayı olarak kaydedilir. Dönüştürülme sırasında, sentezlenen konuşma sinyali, konuşma sinyalinin çözünürlüğünü yüksek tutmak amacıyla 3...253 aralığında normalize edilir ve tamsayıya çevrilir. Daha sonra WAV dosyasının başlık kısmı veri uzunluğu, örneklerde kullanılan bit sayısı (8-bit kullanılmıştır), kullanılan kanal sayısı (tek kanal kullanılmıştır) gibi parametreler de göz önüne alınarak oluşturulur. Dalga şeklinin de başlık kısmının sonuna dosya formatına uygun şekilde eklenmesiyle dosya oluşturulmuş olur. Daha sonra J2ME platformu içindeki uygun ara yüz ve metodlar kullanılarak cihazın oluşturulan WAV dosyasını çalması sağlanır.

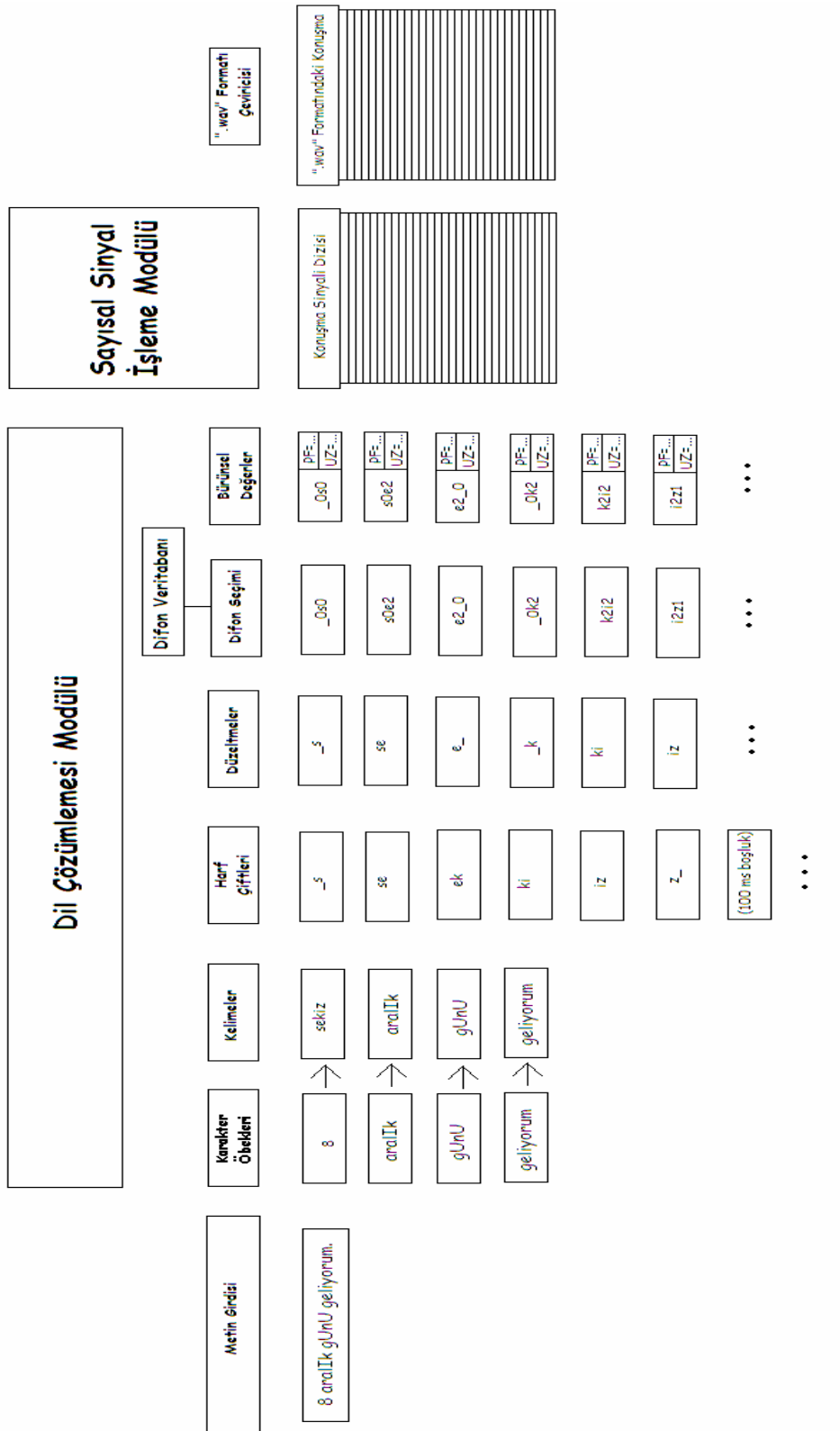
Yukarıda anlatılan işlemleri göstermek için “8 Aralık günü geliyorum.” cümlesinin nasıl işlendiği Şekil 6.10'da gösterilmiştir. Bu cümlenin işlenmesi sırasında önce cümlenin karakter öbeklerine ayrılması, sonra “8” rakamının kelimeye çevrilmesi, daha sonra harf ikililerine ayrılması, “sekiz” kelimesi içindeki “k” harfi için düzeltme yapılması, harf ikilileri için difonların bulunması ve bürünsel değerlerin difonlara eklenmesi işlemlerinin yukarıda anlatılan şekilde ve sırayla yapıldığı şekilde görülebilir.

Geliştirilen uygulamada Türkçe karakter girişlerinde sorun yaşamamak için ‘ç’ yerine ‘C’, ‘ı’ yerine ‘I’, ‘ğ’ yerine ‘G’, ‘ö’ yerine ‘O’, ‘ş’ yerine ‘S’, ‘ü’ yerine ‘U’ kullanılmaktadır. Bunlar dışında kalan bütün harfler küçük harf olarak sisteme girilmektedir. Bu kısıtlamanın iki sebebi vardır. Birincisi, sisteme el ile giriş yapılacağı zaman telefon ara yüzünden Java programına bazı Türkçe karakterlerin geçirilmesi mümkün olmamaktadır (‘ğ’, ‘ı’, ‘ş’). İkinci olarak da, standart SMS alfabesinde Türkçe karakterler yer almamaktadırlar (GSM 03.38 için SMS alfabesi Ek G'de verilmiştir).

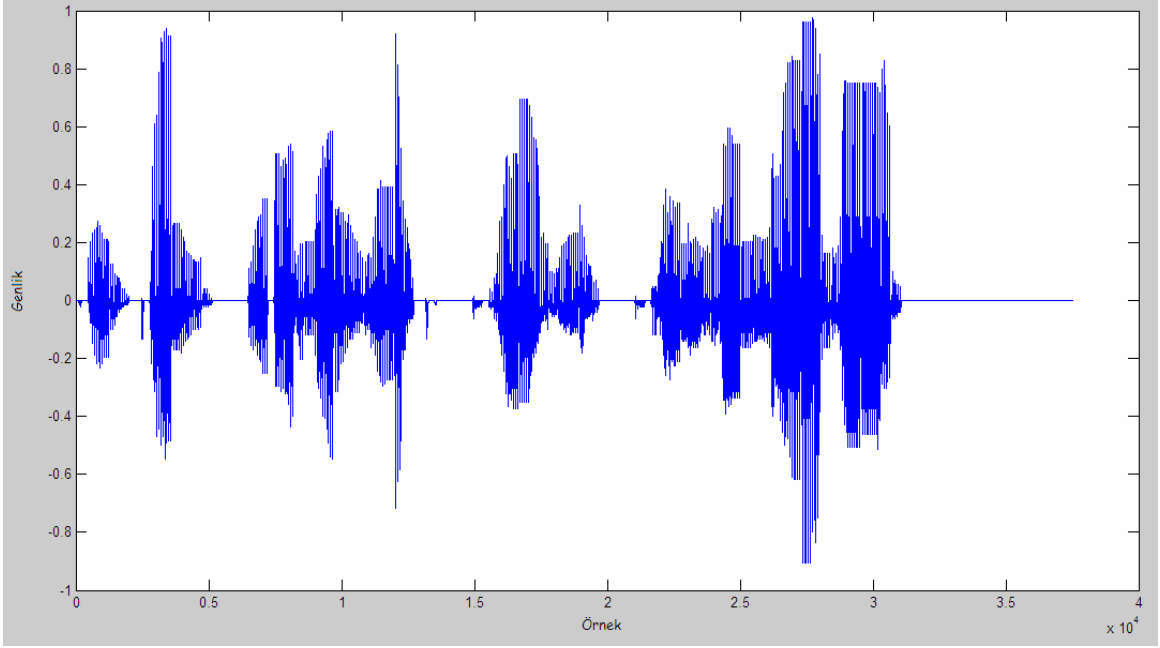
Geliştirilen program bir Nokia 6630 cep telefonunda başarıyla çalıştırılmıştır. 220 MHz ARM-9 işlemciye ve 10Mb hafızaya sahip olan bu cep telefonunda “8 Aralık günü geliyorum.” cümlesinin konuşmasının sentezlenmesi ve ses çalınması 13



saniye sürmektedir. Geliştirilen uygulama ile elde edilen “8 Aralık günü geliyorum.” cümlesine ait dalga şekli Şekil 6.11’de gösterilmiştir.



Şekil 6.10. “8 Aralık günü geliyorum.” cümlesi için programın çalışma basamakları.



Şekil 6.11. “8 Aralık günü geliyorum.” cümlesi için elde edilen dalga şekli.

Geliştirilen programın simülör üzerindeki görüntüsü de Şekil 6.12’de gösterilmiştir.



Şekil 6.12. Uygulama için geliştirilen ve elle metin girişini sağlayan ara yüz.

## 6.5. Uygulama İçin Gerekli Java Ortamı

Daha önce belirtildiği gibi geliştirilen uygulama J2ME platformunda çalışmaktadır. Yine daha önce vurgulandığı gibi, J2ME platformunu destekleyen bir taşınabilir cihazda, Java sanal makinesi, kullanılan konfigürasyon, profiller ve diğer API'ler cihazdaki Java ortamını oluşturmaktadır. Bu uygulamada geliştirilen programın da çalışabilmesi için bazı API'lerin cihaz tarafından desteklenmesi gerekmektedir. Gereken API'ler şunlardır:

a) Bağlantılı Cihaz Konfigürasyonu (CLDC) 1.1 (JSR-139): Bu API sistemde kullanılan en temel veri tiplerini ve sınıf yapılarını tanımlar. Bu uygulama dahilinde oldukça fazla miktarda ve hassasiyet isteyen matematiksel hesaplamalar yapıldığı için, sayıların ondalık değerlerini ifade etmeye yarayan "float" veri tipine ihtiyaç duyulmuştur. CLDC 1.0'da desteklenmeyen bu veri tipi CLDC 1.1'de desteklenmiştir. Her ne kadar "float" veri tipi kullanılmadan ondalık sayılarla işlem yapma yöntemleri var ise de, bu yöntemler fazladan işlem yükü ve programcılık çabası gerektireceği için uygun görülmemiştir. Geliştirilen uygulamanın çalışması için cihazın CLDC 1.1'i desteklemesi zorunludur.

b) Taşınabilir Bilgilendirme Aygıtı Profili (MIDP) 2.0 (JSR-118): Bu API cihazda kullanıcı ara yüzü için gerekli sınıfları tanımlar. Geliştirilen uygulamada kullanıcının el ile sentezlenecek metni girmesi için bir ara yüze gereksinim bulunmaktadır. Ara yüzü oluşturmak için kullanılacak sınıf yapıları JSR-118 kapsamında tanımlanmıştır. Bunun için uygulamanın çalıştırılmak istendiği cihazda MIDP 2.0 desteklenmelidir.

c) Telsiz Mesajlaşma Programcı Arayüzü (JSR-120): Daha önce belirtildiği gibi, uygulamanın çalıştığı cep telefonuna gelen kısa mesajlar konuşmaya çevrilmektedir. Bu durumda cihazın içinde çalışan Java uygulamasının, cihaza gönderilen kısa mesajlara erişebilmesi gerekmektedir. Bu erişimi düzenleyen API, JSR-120 numaralı API'dir. Bu API içinde tanımlanan sınıf yapıları, metodlar ve arayüzler sayesinde geliştirilen uygulama cihaza gelen kısa mesajlara erişebilmektedir.

d) Taşınabilir Ortam Programcı Arabirimi (JSR-135): Girilen metne veya kısa mesaja göre konuşma sentezlendikten sonra bu konuşmanın dış dünyaya

iletilebilmesi, yani fiziksel olarak sese dönüştürülmesi gerekmektedir. Bunu sağlayan JSR-135 numaralı API olmaktadır. JSR-135 sayesinde cihazlardaki Java uygulamaları, WAV dosyalarını cihazın hoparlörünü veya kulaklıklarını kullanarak fiziksel olarak sese dönüştürebilme imkanına sahiptir. Bu tez için geliştirilen uygulamada da konuşma dalga şekli sentezlendikten sonra WAV formatına dönüştürülmekte ve JSR-135 içerisinde tanımlanmış metod, sınıf ve ara yüzler sayesinde cihazın hoparlörlerinde çalınabilmektedir.

Uygulama, yukarıda belirtilen ortamı sağlayabilen Nokia 6630 cep telefonunda çalıştırılmıştır. Kullanılan cihazın özellikleri Çizelge 6.7’de özetlenmiştir.

| <b>Kullanılan Cihazın Teknik Özellikleri</b> |  |
|--|--|
| <b>İşlemci</b>                               | ARM-9                                  |
| <b>İşlemci Frekansı</b>                      | 220 MHz                                |
| <b>Toplam Bellek Miktarı</b>                 | 10 MB (Dahili) + 64 MB (Hafıza Kartı)  |
| <b>Kullanılabilir Bellek Miktarı</b>         | 3.5 MB (Dahili) + 25 MB (Hafıza Kartı) |
| <b>İşletim Sistemi</b>                       | Symbian 8.0a                           |
| <b>J2ME Konfigurasyonu</b>                   | CLDC 1.1 (JSR-139)                     |
| <b>J2ME Profili</b>                          | MIDP 2.0 (JSR-118)                     |
| <b>Diğer Java API’leri</b>                   | JSR-120, JSR-135                       |

Çizelge 6.7. Kullanılan cihazın teknik özellikleri

## **6.6. Platform Sebebiyle Karşılaşılan Zorluklar**

Bölüm 5’te incelendiği üzere taşınabilir cihazlarda birtakım kısıtlamalar bulunmaktadır ve bu kısıtlamalar uygulama geliştiricinin dikkat etmesi gereken unsurlardır. Bu çalışma için uygulama geliştirilirken de birtakım problemlerle karşılaşmıştır.

Sistemdeki bellek kısıtlaması nedeniyle kullanılan yapılar mümkün olduğunca sabit bellek tahsisi ile kullanılmaya çalışılmıştır. Özellikle kullanılan difon veritabanı ve oluşturulan konuşma bellekte önemli miktarda yer kaplamaktadır. Difon veritabanının boyutu yaklaşık 950 kilobayt kadar tutmaktadır. Bu bellek miktarı bir cep telefonu için çok büyük sayılabilecek bir miktardır. Bu yüzden uygulamada veritabanının sadece bir defa yüklenmesine dikkat edilmiş ve yeniden kullanılacağı

zaman önceden kullanılan bellek alanından çağırılmıştır. 8 kHz ve 8 bit örnekleme ile oluşturulan konuşma sinyallerinin kapladığı yer ise saniyesi yaklaşık 8 kilobayt üzerinden hesaplanabilir. Bu miktar düşük gibi gözükse de, art arda yapılan konuşma sentezlemeleri toplanarak artan bir bellek kullanımını beraberinde getirecektir ve bu da bellekte çok büyük bir yerin kullanılmasına sebep olacağı için sentezlenen konuşma sinyalleri de bellekte sabit bir bölgede tutulup, bir konuşma sentezlendikten sonra yeni bir sentezleme yapıldığında yine aynı bellek bölgesi kullanılmıştır.

Diğer Java platformlarında olduğu gibi J2ME’de de bir artık bellek toplayıcısı (Garbage collection) bulunmaktadır ve bu toplayıcı artık kullanılmayan bellek bölgelerini yeniden Java sanal makinesinin kullanımına verir [24]. Fakat bu toplayıcının da çalıştığı belirli zaman aralıkları vardır ve bu toplayıcının çalışması da bir miktar işlem gücü istemektedir. Bu yüzden bellek yönetiminde dikkatli olunmalı, sadece artık bellek toplayıcısına güvenmek yerine önceki paragrafta anlatılan önlemler alınmalıdır. Aksi takdirde bellek yetersizliği hataları ile karşılaşılabilir. Örneğin, veritabanı yüklemesi her konuşma sentezlenmesi sırasında yinelenirse yaklaşık 5 konuşma sentezlenmesinden sonra cihaz bellek yetersizliği yönünde uyarı vermektedir. Oysaki artık bellek toplayıcısının veritabanının önceki yüklemelerinde kullandığı bellek bölgesini yeniden kullanıma açması beklenirdi. Fakat artık bellek toplayıcısının kendi kısıtlamalarından dolayı bellek toplanması işlemi yeteri kadar hızlı olamamaktadır ve bu da J2ME programcısını daha dikkatli olmaya teşvik etmektedir.

İşlem gücünün düşüklüğü dolayısıyla da bazı hususlara dikkat edilmiştir. Java’da yeni bir nesne oluşturmak işlem gücü isteyen bir işlemdir. Bu yüzden mümkün olduğu kadar yeni nesne oluşturmadan basit veri tipleriyle işlemler gerçekleştirilmeye çalışılmıştır. Aynı zamanda Java’da zaman harcayan fonksiyon çağırma işlemleri de mümkün olduğu kadar az yapılmaya çalışılmış, örneğin, mümkünse döngü içerisinde fonksiyon çağırarak yerine, döngü başında fonksiyon çağırılıp, değer başka bir değişkene atanmaktadır.

Bölüm 5’te belirtildiği üzere J2ME platformunda kullanılan kütüphanelerde J2SE’e göre birtakım farklılıklar ve eksiklikler bulunmaktadır. Bir Java programcısının J2SE kütüphanelerinden alışık olduğu “ArrayList” ve “HashMap” gibi bazı oldukça

kullanışlı sınıflar varsayılan J2ME kütüphanesinde bulunmamaktadır. Yine de bu gibi eksik sınıf yapıları uygulamaya ayrıca dâhil edilebilirler. Bu uygulamada da, internet üzerinde bulunan “J2ME Polish” kütüphanesinden alınan<sup>19</sup> “ArrayList” ve “HashMap” gibi sınıflar ayrıca uygulamaya dâhil edilmiş ve kullanılmıştır.

Bölüm 6.4’te anlatıldığı üzere sisteme Türkçe karakter girişi yapılacağı zaman ‘ç’ yerine ‘C’, ‘ı’ yerine ‘I’, ‘ğ’ yerine ‘G’, ‘ö’ yerine ‘O’, ‘ş’ yerine ‘S’, ‘ü’ yerine ‘U’ kullanılmaktadır. Bu kısıtlamanın iki sebebi vardır. Birincisi sisteme el ile giriş yapılacağı zaman telefon ara yüzünden Java programına bazı Türkçe karakterlerin geçirilmesi mümkün olmamaktadır (‘ğ’, ‘ı’, ‘ş’). İkinci olarak da standart SMS alfabesinde Türkçe karakterler yer almamaktadırlar (GSM 03.38 için SMS alfabesi EK G’de verilmiştir).

### **6.7. Geliştirilen Diğer Uygulamalar**

Cep telefonunda çalışan uygulamanın kısa mesaj alması için, bu programa kısa mesaj gönderen bir başka J2ME programı daha yazılmıştır. Bunun sebebi, JSR-120 ile ilgili kısıtlamalardan dolayı midletlerin J2ME platformunda, cep telefonundaki kısa mesaj gelen kutusuna doğrudan erişememesidir. Fakat telefonun portlarından seçilen birisi dinlenerek bu porta gönderilen kısa mesajlar midlet tarafından yakalanabilir. Bu yüzden, midletin kısa mesajları yakalayabilmesi için önceden seçilmiş bir porta (örneğin bu çalışmada 5000 numaralı port kullanılmıştır) kısa mesaj gönderilmelidir ve bu portu dinleyen midlet, kısa mesajı konuşmaya çevirebilmektedir. Yazılan bu ikinci uygulama bir başka cep telefonunda çalışmakta, MKS midletinin çalıştığı telefona kısa mesaj göndermekte ve MKS midleti konuşmayı sentezlemektedir. Fakat yeni geliştirilen J2ME kütüphaneleri sayesinde gelecekte midletler doğrudan kısa mesaj gelen kutusuna erişebileceklerdir.

Hata ayıklamak ve hızlıca denemeler yapmak amacıyla, bu programın bir de J2SE versiyonu geliştirilmiştir. Tamamen aynı algoritmayı kullanan J2SE versiyonu ile oluşturulan konuşma sinyali WAV formatında kaydedilebilmekte ve veritabanı dosyasındaki bazı hatalar (Perde işareti uzunluğunun 0 olması, veya perde işaretlerinin hatalı sırada yazılması gibi) saptanabilmektedir.

---

<sup>19</sup> <http://www.j2mepolish.org>

## 7. SONUÇ

Bu yüksek lisans tezi kapsamında, mevcut metinden konuşma sistemleri incelenmiş, taşınabilir bir cihaz için uygun olan sistemler ve birleştirme algoritmaları tespit edilmiştir. Edinilen bilgiler doğrultusunda da bir cep telefonu üzerinde çalışan ve anlaşılabilir düzeyde sentezleme yapabilen bir Türkçe metinden konuşma sentezleme sistemi tasarlanmıştır. Ayrıca daha ayrıntılı sentezleme ve analiz yapmak için aynı algoritmayı kullanan ve J2ME platformunda çalışan bir program da geliştirilmiştir.

Günümüzde, artık birçok taşınabilir cihazın, bu çalışmada olduğu gibi fazlaca matematiksel işlem isteyen uygulamaları yükleyip çalıştırabileceği gösterilmiştir. Bu önemli bir göstergedir ve gelecekte taşınabilir cihazların daha fazla yaygınlaşmasıyla da birlikte bu cihazların işlevselliklerinde artışlar gözlenebilecektir.

Fakat taşınabilir cihazlardaki anlatılan gelişmelere rağmen, taşınabilir cihazların doğası gereği her zaman için daha hafif, küçük boyutlu ve az enerji harcayan donanımlardan üretilmeleri gerekmektedir. Bu durumun, taşınabilir cihazlar için geliştirilen yazılımlarda daima dikkate alınması gerekecektir.

Bu çalışmada da yapılan araştırmalar sonucunda diğer yöntemlere göre daha az işlem yükü isteyen birleştirerek sentezleme yöntemi kullanılmış ve ezgi değişiklikleri için TD-PSOLA yöntemi seçilmiştir. Birleştirme için de diğer seçeneklere göre daha az yer kaplayan difon parçaları kullanılmıştır. Geliştirilen uygulamanın cihazın markası, içinde bulunan diğer yazılımlar gibi etkenlerden etkilenmemesi ve mümkün olduğu kadar fazla sayıda taşınabilir cihazda kullanılabilmesi için J2ME platformunda geliştirilmiştir. Bu seçimler günün koşullarındaki taşınabilir cihazlar için en uygun seçimlerdir. Bununla beraber, taşınabilir cihazlardaki donanımsal ve yazılımsal çalışmalara bağlı olarak gelecekte diğer yöntemlerin veya ortamların seçilmesi mümkün olabilir. Nitekim, 2002 yılından önce böyle bir çalışmanın yapılması dahi mümkün değildi.

Bu çalışma, taşınabilir bir cihaz için geliştirilen ilk Türkçe MKS sistemi çalışması olmuştur. Türkçedeki birtakım temel bürün işlemlerini de yapan bir Dil Çözümlemesi Birimi ve anlatılan yöntemleri kullanan bir Sayısal Sinyal İşleme

Birimi geliştirilmiştir. Geliştirilen uygulama bir cep telefonu üzerinde çalıştırılabilmekte, cihaza gelen kısa mesajlardan veya el ile girilen Türkçe bir metinden, anlaşılabilir bir şekilde konuşma sentezlenebilmektedir.

Taşınabilir cihazlar ve bu cihazlar için geliştirilen platformların henüz gelişmesini tamamlayıp standartlaşmaması sonucu olarak bazı problemlerle karşılaşmıştır. Sisteme girdi olarak Türkçe karakterler vermekte yaşanan problemler, hem büyük, hem de küçük harfler kullanılarak aşılmıştır. J2ME uygulamaları, kısıtlamalardan dolayı cep telefonlarında sadece belirli bir portu dinleyebildiği için, uygulamanın çalıştırıldığı cihaza belirli bir porttan kısa mesaj gönderen ikinci bir uygulama da geliştirilmiştir. İşlem gücü ve bellek kısıtlamalarına karşı ise ilgili bölümde bahsedilen önlemler alınarak hızlı ve bellekte az yer kaplayan bir kodlama yapılmaya çalışılmıştır.

Kullanılan difon veritabanı dosyası başka bir çalışmadan alınmasına rağmen 30 adet difon eklenerek geliştirilmiştir (eklenen difonlar için Ek E'ye bakılabilir). Ayrıca veritabanındaki diğer veriler incelenmiş, bulunan bazı uyumsuz veriler ('0' veya negatif bir sayı olarak belirtilen difon birleşme noktaları veya küçükten büyüğe sıralanmayan perde işaretleri gibi) düzeltilmiştir. Diğer uygulamalarda pek görülmeyen bir şekilde, kullanıcıya istediği fonemi el ile seçtirme imkânı sunulmuştur. Veritabanında bulunmayan olası difonlar için mümkünse benzer bir difon seçtirilip konuşmanın sentezlenmesi sağlanmıştır.

TD-PSOLA yönteminin ezgi değişikliklerini uygun bir şekilde yaptığı, sistemin çıktıları incelenip yorumlanarak gösterilmiştir.

Türkçe için geliştirilecek tam işlevsellikli bir Dil Çözümlemesi Birimi için gerekli altyapıyı sağlayacak olan çalışmalar henüz yetersizdir. Yine de bir Dil Çözümlemesi Biriminin yapacağı temel işlemleri yapan bir Dil Çözümlemesi Birimi geliştirilmiş ve bu birimin doğru bir şekilde çalıştığı gösterilmiştir.

Türkçe için Dil Çözümlemesi Birimi üzerine son yıllarda yapılan çalışmalar ümit vericidir. Fakat daha önceki yıllardaki eksikliklerden kaynaklanan boşluk henüz dolmamıştır. Özellikle Türkçedeki bürün yapısının incelenmesi için gerekli olan algoritmalarda MKS sistemlerinde kullanılabilecek özelliklere yönelik gelişme sağlanabilir. Bu konuda Sınıflandırma ve Regresyon Ağaçları (CART)



oluşturulabileceği gibi [29] (İngilizce için örnekler: FreeTTS ve Flite) daha gelişmiş doğal dil işleme algoritmaları da kullanılabilir. Çünkü yapılacak olan çalışmalarda, bürün özelliklerinin bulunması için cümlelerin öğelerine ayrılması ve daha sonra yorumlanması gerekecektir. Türkçe MKS sistemleri için şu andaki asıl sorun birleştirme algoritmasının kalitesinden ziyade sentezleme sırasında kullanılacak bürünsel parametrelerin doğru seçimidir. Özen'in çalışmasında [22] geliştirilen Kalıntı Uyarımlı Doğrusal Öngörümlü Zaman Ekseninde Perde Eşzamanlı Örtüştürme Ekleme (RELP-PSOLA) yöntemini kullanan MKS sistemi uygun bürünsel parametreler seçildiği takdirde gayet anlaşılabilir konuşma üretmekle beraber bu sistem bürünsel parametreleri üretmekten yoksundur. Bununla beraber Türkçe bir metin için bürünsel parametreleri tamamen ve yeterli seviyede, yani sentezlenen konuşmayı doğal olarak yorumlanabilecek şekilde üreten bir sistem bulunmamaktadır. Buna karşın Bölüm 4'te anlatılan temel bürünsel özelliklerin bir kısmı bu çalışmada geliştirilen yazılımda, Bölüm 6.2'de açıklandığı şekilde uygulanmıştır.

Türkçede dil işleme sorununu cümle analizi yönünden karşılayacak "Zemberek" isimli açık kaynak bir proje bulunmaktadır. Bundan sonra, Türkçe için açık kaynak bir MKS sistemi projesi hayata geçirilebilir. İngilizce için FreeTTS gibi çok başarılı örnekleri bulunmaktadır.

Taşınabilir cihazlar her geçen gün daha da gelişmektedir. Piyasaya sürülen her cihaz, önceki cihazlardan daha güçlü bir işlemciye sahip olmakta, daha fazla hafıza alanı içermekte ve daha gelişmiş işletim sistemlerine sahip olmaktadır. Ayrıca gün geçtikçe cep telefonları için yeni Java kütüphaneleri geliştirilmekte ve bu kütüphaneler sayesinde cep telefonları üzerinde Java programının yetkileri ve kabiliyetleri artmaktadır.

Dolayısıyla yeni taşınabilir cihazlar ile bu tez kapsamında karşılaşılan bazı problemler aşılabilecektir. İşlem kapasitesi daha yüksek işlemciler sayesinde yazılan program daha hızlı çalışabilecek ve daha fazla matematiksel işlem gücü gerektiren algoritmalar (örn. RELP-PSOLA) kullanılabilir ve daha fazla sistem belleği kullanımı sayesinde daha büyük konuşma veritabanları yüklenebilecektir. Bu teknolojik gelişmeler sayesinde ses kalitesinde ve sentezleme zamanında iyileşmeler gözlenecektir.

## KAYNAKLAR

- [1] Abdullahbeşe, E., 2001, "Fundamental Frequency Contour Synthesis for Turkish Text-to-Speech", Boğaziçi Üniversitesi Yüksek Lisans tezi
- [2] Ayhan, K.,1998, "Text to Speech Synthesizer in Turkish Using Non Parametric Techniques", ODTÜ, Yüksek Lisans tezi
- [3] Bozkurt, B., 2000, "Reading Aid For Visually Impaired", Boğaziçi Üniversitesi Yüksek Lisans tezi
- [4] Demircan, Ö., 2001, "Türkçenin Ses Dizimi", Der Yayınları.
- [5] Dutoit T., 1993, "High Quality Text-To-Speech Synthesis of the French Language", Faculté Polytechnique de Mons Doktora tezi
- [6] Dutoit T., Leich H., 1993, "MBR-PSOLA : Text-to-Speech Synthesis Based on an MBE Re-Synthesis of the Segments Database", <http://tcts.fpms.ac.be>
- [7] Dutoit, T., 1997, "An Introduction to Text-to-Speech Synthesis", Kluwer Academic Publishers.
- [8] Eker, B., 2002, "Turkish Text to Speech System", Bilkent Üniversitesi Yüksek Lisans tezi
- [9] Embree, P. M., 1999, "C++ Algorithms for Digital Signal Processing", Prentice Hall.
- [10] Erer, M. S., 1994, "Karma Söz Üretme Yöntemi ile Türkçe Yazılı Metinden Söze Geçirme", İTÜ Yüksek Lisans tezi
- [11] Ergenç, İ., 2002, "Konuşma Dili ve Türkçenin Söyleyiş Sözlüğü", Multilingual Yabancı Dil Yayınları.
- [12] Hill, D. R., Manzara, L., Schock, C., 1995, "Real-time Articulatory Speech-Synthesis-by-Rules", Proceedings of AVIOS '95, the 14th Annual International Voice Technologies Applications Conference of the American Voice I/O Society, San Jose
- [13] -, 2003, "J2ME & Symbian OS: A Platform Comparison v.1.0", Forum Nokia, (<http://www.forum.nokia.com/>), [http://sw.nokia.com/id/f6869109-b85c-4dab-a293-1f1860f85963/J2ME\\_Symbian\\_OS\\_1\\_01.pdf](http://sw.nokia.com/id/f6869109-b85c-4dab-a293-1f1860f85963/J2ME_Symbian_OS_1_01.pdf)
- [14] -, 2004, "Efficient MIDP Programming v.1.1", Forum Nokia, (<http://www.forum.nokia.com/>), [http://sw.nokia.com/id/d307878f-bbd6-415a-af25-bf7fb3efc9d3/Efficient\\_MIDPProgramming\\_v1\\_1\\_en.pdf](http://sw.nokia.com/id/d307878f-bbd6-415a-af25-bf7fb3efc9d3/Efficient_MIDPProgramming_v1_1_en.pdf)
- [15] Karlı, A., 2005, "Örnek Bir Dizi Cümle İçin Türkçe Metinden Konuşma Sentezleyici", Ankara Üniversitesi Yüksek Lisans tezi

- [16] Kılıç, M. A., 2000, "Türkiye Türkçesindeki Ünlülerin Fonetik Özellikleri", X. Uluslararası Türk Dilbilimi Kurultayı.
- [17] Lemmetty, S., 199, "Review of Speech Synthesis Technology", Helsinki University of Technology Yüksek Lisans tezi
- [18] Lyon, D. A., Rao, H. V., 1997, "Java Digital Signal Processing", M & T Books
- [19] Mahmoud, Q., 2001, "Learning Wireless Java", O'Reilly
- [20] Oskay, B., 2002, "Automatic Modelling of Turkish Prosody", ODTÜ Yüksek Lisans tezi
- [21] Önür, Ç., 2001, "Concatenative Speech Synthesis Based on a Sinusoidal Speech Model", ODTÜ Yüksek Lisans tezi
- [22] Özen, Ş. S., 2002, "Türkçe Metinden Konuşma Sentezleme", Hacettepe Üniversitesi Yüksek Lisans Tezi.
- [23] Salor, Ö., 1999, "Signal Processing Aspects of Text to Speech Synthesizer in Turkish", ODTÜ Yüksek Lisans tezi
- [24] Schildt, H., 2003, "Java 2", Alfa Yayınları
- [25] Selen, N., 1979, "Söyleyiş Sesbilimi, Akustik Sesbilim ve Türkiye Türkçesi", TDK
- [26] Şayli, Ö., 2002, "Duration Analysis and Modelling for Turkish Text-to-Speech Synthesis", Boğaziçi Üniversitesi Yüksek Lisans tezi
- [27] Taylor, M. 2003, "Why J2ME Projects Fail?", Developnet Consulting Limited, <http://www.developnet.co.uk>
- [28] Üçok, N., 1951, "Genel Fonetik", Ankara Üniversitesi Dil, Tarih ve Coğrafya Fakültesi Yayınları
- [29] Vural, E., 2003, "A Prosodic Turkish Text-to-Speech Synthesizer", Sabancı Üniversitesi Yüksek Lisans tezi

## SÖZLÜK

|                                     |   |
|-------------------------------------|---|
| Alan fonksiyonu                     | İng. Area function  |
| Artık Bellek Toplayıcısı            | İng. Garbage Collection   |
| Biçimbirim                          | Kelimelere dil bilgisi bakımından biçim veren çoğu ek hâlinde olan kelime parçaları         |
| Bürün                               | İng. Prosody  |
| Yüksek hızlı darbeli iletim hattı   | İng. High-speed pulse transmission-line   |
| Doğrusal Öngörümlü Kodlama          | İng. Linear Prediction Coding   |
| Durak                               | Konuşmada anlamın gerektirdiği biçimde kelimeler arasındaki ses kesintisi.                  |
| Ezgi                                | Bir konuşma zincirindeki seslem, biçimbirim ve sözcükleri kapsayan ton değişimlerinin tümü. |
| Ev sahibi ortam                     | İng. Host environment   |
| Fonem                               | Dil biliminde anlamda farklılaşmaya yol açabilen en küçük konuşma parçası, ses birimi.      |
| Formant                             | Akustik bir sistemin rezonant frekanslarından kaynaklanan, frekans izgesindeki tepeler.     |
| İzge                                | İng. Spectrum   |
| Karakter öbeği                      | İng. Token  |
| Kısa mesaj servisi                  | İng. Short Message Service  |
| Kişisel Sayısal Yardımcı            | İng. Personal Digital Assistant (PDA)   |
| Midlet                              | MIDP profilinde çalıştırılmak üzere geliştirilen uygulama.                                  |
| Perde eşzamanlı örtüştürerek ekleme | İng. Pitch Synchronous Over Lap Add   |
| Seslem                              | Hece  |

|              |  |
|--------------|--|
| Ton          | Bir seslemdeki sıklık yüksekliđi ya da dūřüklüđü, bir seslemin tiz ya da pes sesletilmesi.                   |
| Vurgu        | Konuřma, okuma sırasında bir hece veya kelime üzerine diđerlerinden daha farklı olarak yapılan baskı, aksan. |
| Yeđinlik     | řiddet   |
| Yıđın Boyutu | İng. Heap size   |
| Yıđıt        | İng. Corpus  |

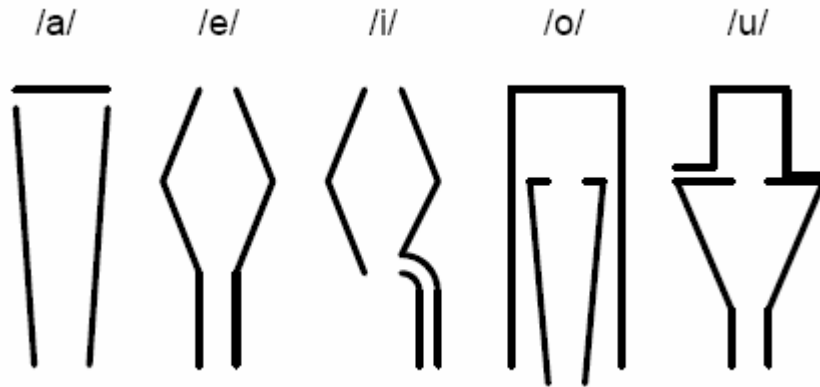
## EKLER

### EK A. Konuşma Sentezleme Sistemlerinin Kısa Tarihi

Metinden konuşma sentezleme (MKS) sistemlerinin konuşma sentezlemesi bir zorunluluktur. Bu bölümde konuşma sentezleme sistemlerinin geçmişi kısaca incelenecektir.

#### EK A.1. Mekanik Konuşma Sentezleme Sistemleri

En eski ses sentezleme çalışmaları iki yüz yıl öncesine kadar gitmektedir [12]. 1779 yılında Sen Petersburg şehrinde Rus profesör Christian Kratzenstein beş sedalı ses (/a/, /e/, /i/, /o/ ve /u/) için fizyolojik farklar üzerine çalıştı ve bu sesleri yapay olarak üretmek için bir cihaz üretti. Bu iş için insan ses yoluna benzer rezonatörler tasarlandı ve tıpkı bir müzik çalgı aletinde olduğu gibi, rezonatörleri uyarmak için titreşen kamışlar kullanıldı. Yalnızca /i/ sesi alt taraftaki boruya hava üflenmesi ile oluşturuluyordu. Rezonatörlerin yapısı Şekil A.1'de gösterilmektedir [17].

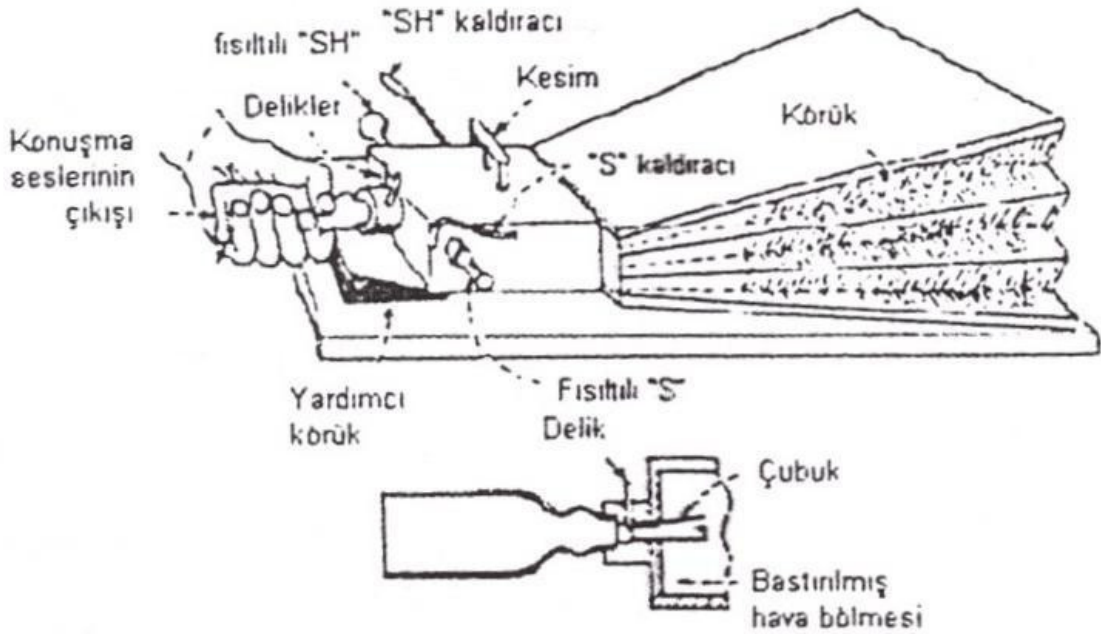


Şekil A.1. Kratzenstein rezonatörleri

1791 yılında Wolfgang von Kempelen bazı sesleri tek başına, bazı sesleri de oluşturulan seslerin birleşimleri olarak oluşturabilen "Akustik-Mekanik Konuşma Makinesi"ni tanıttı. Bu makine temel olarak bir körük, ses telleri gibi davranan bir düdük ve geniz boşluğu gibi davranan bir deri borudan oluşuyordu. Deri borunun şekli değiştirilerek değişik sedalı sesler üretilebiliyordu. Sedasız sesler ise dört

farklı dar geçit aracılığıyla oluşturuluyordu ve parmaklarla denetleniyordu. Patlamalı sesler için ise bağlı bir dil ve hareketli dudakları olan bir geniz boşluğu modeli kullanılıyordu. Kempelen'in çalışmaları ses yolunun akustik telaffuzun kaynağı olduğunu gösterdi. Kempelen'den önce gırtlığın konuşma üretiminin merkezi olduğu sanılıyordu.

1800'lerin ortalarında Charles Wheatstone, Kempelen'in konuşan makinesinin daha gelişmiş bir sürümünü geliştirdi. Bu makine tüm sedalı sesleri ve sedasız seslerin büyük bir kısmını üretebiliyordu ve hatta bazı kelimeleri tüm olarak sentezlemek bile mümkün olabiliyordu. Bu aletin bir çizimi Şekil A.2'de görülebilir [15].



Şekil A.2. Wheatstone tarafından yapılan, Kempelen'in konuşan makinesinin daha gelişmiş bir versiyonu [15]

Sedalı bir ses ile ses yolu arasındaki ilişki Willis tarafından 1838 yılında bulundu. Willis, değişik sedalı sesleri boru rezonatörler ile oluşturmayı başardı ve ayrıca sedalı ses kalitesinin borunun uzunluğuna bağlı olduğunu ve borunun çapına bağlı olmadığını buldu.

Mekanik ve yarı elektriksel sistemler üzerine yapılan çalışmalar 1960'lı yıllara kadar sürdü fakat bu alanda önemli bir başarı elde edilemedi [17].

## **EK A.2. Elektriksel Ses Sentezleyiciler**

Bilinen ilk elektriksel ses sentezleyici 1922 yılında Stewart tarafından tanıtılmıştır. Bu sentezleyici bir elektrik zili ve ses yolu için iki rezonans devresi kullanıyordu. Bu makine tek sedalı sesleri en düşük iki formant ile üretebiliyor fakat sedasız ses veya bağlı sesler üretemiyordu. Daha sonra Wagner tarafından dört rezonatörlü benzer bir cihaz geliştirildi. 1932 yılında Japon araştırmacılar Obata ve Teshima sedalı seslerdeki üçüncü formantı keşfettiler. Üç formant, yapay olarak sentezlenmiş sesler için genellikle yeterli kabul edilmektedir [17].

Konuşma sentezleyicisi olarak kabul edilebilecek ilk cihaz 1939 yılında Homer Dudley tarafından New York Dünya Fuarı'nda tanıtılan VODER (Voice Operating Demonstrator) oldu. VODER, Bell laboratuvarlarında 1930'ların ortasında geliştirilen VOCODER'den ilham alınarak tasarlanmıştır.

VODER'in ortaya çıkmasından sonra bilim dünyasının ses sentezleme üzerine ilgisi daha da artmıştır. VODER'la birlikte yapay olarak anlaşılabilir bir sesin sentezlenebileceği gösterilmiştir.

İlk formant sentezleyicisi, 1953 yılında Walter Lawrence tarafından tanıtılan Parametrik Yapay Konuşucu (Parametric Artificial Talker) oldu. PAT, paralel bağlanmış rezonatorler kullanıyordu. Yaklaşık aynı zamanlarda Gunner Fant, OVE I adlı formant sentezleyicisi tanıttı. Bu cihazda ise rezonatörler seri bağlıydı. OVE projesi OVE II, OVE III ve GLOVE olarak devam etti.

İlk artikülatör sentezleyici 1958 yılında George Rosen tarafından Massachusetts Teknoloji Enstitüsü'nde geliştirilen DAVO (Dynamic Analog of the VOcal tract) oldu. 1960'larda ilk doğrusal kestirim kodlanması denemeleri yapıldı.

İngilizce için ilk eksiksiz MKS sistemi Japonya'da, Electrotehcnical Laboratory'de 1968 yılında Noriko Umeda ve yardımcıları tarafından geliştirildi. Sistem, artikülatör model kullanılarak geliştirilmişti. Sentezlenen ses anlaşılabilir fakat şimdiki sistemlerin kalitesinden uzaktı.



## **EK B. MKS Sistemlerinin Kullanım Alanları**

Yapay konuşma, bir çok deęişik sistemde kullanılmaktadır. MKS sistemlerinin kalitesi arttıkça kullanım alanları da genişlemektedir. Konuşma sentezleme sistemlerinin fiyatları da daha uygun olmaya başladıkça bu sistemler günlük hayat için daha uygun hale gelmektedir [17]. Aşağıda MKS sistemlerinin kullanıldığı bazı yerlere kısaca değinilmiştir.

**a) Görme engelliler için uygulamalar:** MKS sistemlerinin en yaygın olarak kullanıldığı ve yararlı olduğu uygulamalar görme engelliler için iletişim yardımı sağlayan uygulamalardır. Görme özürü insanların karşılaştıkları bir metni okumasına yardımcı olmak için optik karakter tanıma destekli MKS sistemleri, basılı bir metnin okunmasını mümkün kılmaktadır. Diğer taraftan kişisel bilgisayarlar için geliştirilmiş bazı uygulamalar da entegre edilen MKS sistemleriyle görme özürü insanlar tarafından kullanılabilir. Bilkent Üniversitesi'nde geliştirilen "OKU" isimli program bu tip programlara örnektir. Bu programda menü seçenekleri bile kullanıcıya sesli olarak okunmakta, böylece görme özürü birisi bir kişisel bilgisayar programını tam işlevselliğiyle kullanabilmektedir.

**b) Sağır ve dilsizleri için uygulamalar:** Sentezlenmiş konuşma, işaret dilini bilmeyen kişilerle sağır dilsizlerin iletişim kurmasını sağlayabilir. Ayrıca MKS sistemleri sayesinde dilsiz insanlar telefonda veya başka bir şekilde sessel bir yoldan iletişim kurabilirler [15].

**c) Haberleşme hizmetleri:** MKS sistemleri, konuşmanın sadece içeriğinin önemli olduğu, kim tarafından, hangi vurgulamalarla söylendiğinin ve diğer bürünsel özelliklerin önemli olmadığı bazı durumlarda, konuşmanın kendisi yerine sadece konuşmanın metne dökülmüş halinin gönderilmesinde kullanılabilir. Bu durumda, konuşmacının söylediği sözler bir konuşmadan metin sentezleme istemi ile metin karakterlerine (harfle) veya fonetik sembollere dönüştürülür ve iletişim hattından sadece bu karakterler veya semboller aktarılır. Bu şekilde iletişim hattındaki bant genişliği daha verimli kullanılabilir [7].

**d) Dil eğitimi:** Yüksek kaliteli MKS sistemleri, yeni bir dil öğrenen öğrenciler için yazının okunması konusunda yardımcı olabilir. Ayrıca, başka bir ülkeye giden bir yabancı, o ülkedeki gördüğü yazıların okunmasında veya elindeki rehber

kitapta o ülkenin diliyle yazılmış bazı cümle kalıplarının okunmasında MKS sistemlerini kullanabilir [7].

**e) Makine - insan arayüzünde iyileşmeler:** MKS sistemleri sayesinde makineler insanlarla konuşma yoluyla iletişim kurabilme kapasitesine sahip olacaklardır. Bu, iletişimin insan açısından daha doğal yollardan gerçekleşmesi demektir.

Diğer taraftan MKS sistemleri yazılı bilgiye ulaşmanın kolay olmadığı başka durumlarda da kullanılabilir. Örneğin bir çarpışan bir savaş pilotu veya tank operatörü, MKS sistemleri sayesinde işlerini daha verimli yapabileceklerdir.

## EK C. Türkçede Bulunan Fonemler

Türkçede bulunan fonemler [11], [22] ve bu çalışmada kullanılan kodları aşağıdaki gibidir.

| kod | fonem | tanım                                | örnek        |
|-----|-------|--------------------------------------|--------------|
| a1  | ɑ     | geniş, düz, arkadil (postdorsal)     | <b>an</b>    |
| a2  | a     | geniş, düz arkadil (predorsal)       | <b>laf</b>   |
| e1  | e     | geniş, düz, öndil (kapalı)           | <b>elma</b>  |
| e2  | ɛ     | geniş, düz, öndil (açık)             | <b>dere</b>  |
| i1  | ɪ     | dar, düz, öndil (açık)               | <b>iğde</b>  |
| i2  | ɪ     | dar, düz, öndil (kapalı)             | <b>simit</b> |
| i5  | ĩ     | dar, düz, arkadil (ortadil)          | <b>ısı</b>   |
| o1  | õ     | geniş, yuvarlak, arkadil (açık)      | <b>soru</b>  |
| o2  | o     | geniş, yuvarlak, arkadil (kapalı)    | <b>oğlak</b> |
| o6  | œ     | geniş, yuvarlak, öndil (açık)        | <b>örtü</b>  |
| o7  | Ø     | geniş, yuvarlak, öndil (kapalı)      | <b>öğren</b> |
| u1  | U     | dar, yuvarlak, arkadil (açık)        | <b>kulak</b> |
| u2  | u     | dar, yuvarlak, arkadil (kapalı)      | <b>uğur</b>  |
| u6  | Y     | dar, yuvarlak, öndil (açık)          | <b>ümit</b>  |
| u7  | Y     | dar, yuvarlak, öndil (kapalı)        | <b>düğme</b> |
| b0  | b     | ötümlü, patlamalı, çift dudak        | <b>balık</b> |
| d0  | d     | ötümlü, parlamalı, dilucu-dişardı    | <b>dede</b>  |
| f0  | f     | ötümsüz, sızmalı, dudak-diş          | <b>efe</b>   |
| g1  | g     | ötümlü, patlamalı, dil-artdamak      | <b>karga</b> |
| g2  | J     | ötümlü, patlamalı, dil-artdamak (ön) | <b>genç</b>  |
| h0  | h     | ötümsüz, sızmalı, gırtlak            | <b>hasta</b> |
| j0  | ç     | ötümlü, sızmalı, dil-öndamak         | <b>müjde</b> |
| k1  | k     | ötümsüz, patlamalı, artdamak         | <b>akıl</b>  |
| k2  | c     | ötümsüz, patlamalı, artdamak (ön)    | <b>kedi</b>  |
| l1  | l     | ötümlü, yan dar, dilucu-öndamak      | <b>lale</b>  |
| l2  | ł     | ötümlü, yan dar, dilucu-öndamak      | <b>kul</b>   |
| m0  | m     | ötümlü, genizsi, çift dudak          | <b>dam</b>   |
| n1  | n     | ötümlü, genizsi, dilucu-dişeti       | <b>anı</b>   |
| n2  | ŋ     | ötümlü, genizsi, dil-artdamak        | <b>süngü</b> |

|    |    |                                    |              |
|----|----|------------------------------------|--------------|
| p0 | p  | ötümsüz, patlamalı, çift dudak     | <b>ip</b>    |
| r1 | r  | ötümlü, çok vuruşlu, dilucu-dişeti | <b>raf</b>   |
| r2 | r  | ötümlü, tek vuruşlu, dilucu-dişeti | <b>ırmak</b> |
| r3 | ᵝ  | ötümsüz, sızıcı                    | <b>bir</b>   |
| s0 | s  | ötümsüz, sızıcı, dilucu-dişeti     | <b>ses</b>   |
| s5 | /  | ötümsüz, sızıcı, dil-öndamak       | <b>aşı</b>   |
| t0 | t  | ötümsüz, patlamalı, dilucu-dişardı | <b>ütü</b>   |
| v1 | v  | ötümlü, sızıcı, dudak-diş          | <b>var</b>   |
| v2 | v  | ötümlü, sızıcı, yarı ünlü          | <b>tavuk</b> |
| y1 | j  | ötümlü, dil-öndamak, yarı ünlü     | <b>yat</b>   |
| z1 | z  | sızıcı, dilucu-dişeti              | <b>azık</b>  |
| z2 | ž  | ötümsüz, sızıcı, dilucu-dişeti     | <b>yoz</b>   |
| c0 | dʒ | ötümlü, sızıcı, dil-öndamak        | <b>cam</b>   |
| c5 | t/ | ötümlü, sızıcı, dil-öndamak        | <b>seçim</b> |

## EK D. J2ME Mimarisi

J2ME, Sun Microsystem'in cep telefonları, KSY'lar ve diğer küçük ve taşınabilir gömülü cihazlar piyasası için çıkarttığı bir Java sürümüdür. Ortaya çıktığından beri Palm, Nokia, Motorola gibi büyük şirketler dâhil 600'den fazla şirket geliştirme çalışmalarına katılmıştır. J2ME, küçük cihazlar için gelişmiş uygulamalar yaratmak için eksiksiz bir dizi çözüm sunmaktadır.

### EK D.1. J2ME'e genel bir bakış

J2ME, genel olarak aşağıdaki bileşenlerle tanımlanır:

- Her biri farklı gereksinimleri karşılayan ve her biri farklı tipte küçük cihaz üzerinde kullanılmak için olan bir dizi Java sanal makinesi.
- “Konfigürasyon” ve “Profil” olarak bilinen ve her sanal makine üzerinde çalışabilen bir grup kütüphane ve API'leri
- Dağıtım ve cihaz konfigürasyonu için çeşitli araçlar.

İlk iki öge J2ME çalışma ortamını oluşturur. Şekil D.1 çalışma ortamının ilişkilerini yansıtmaktadır. Cihazın işletim sistemi üzerinde Java sanal makinesi çalışmaktadır. Bunun üzerinde cihaza özel, cihazın kaynak gereksinimleri için temel fonksiyonellik sağlayan kütüphaneleri içeren J2ME konfigürasyonu bulunmaktadır. Konfigürasyonun üzerinde de, benzer cihazlar üzerinde benzer işlevsellikler için kullanılan ilave kütüphaneleri içeren profillerden bir veya birden fazlası bulunmaktadır [19]. Konfigürasyonlar daha alt seviyedeki kütüphaneleri tanımlarlar (örneğin kullanılan sayı tipler ve temel sınıf yapıları gibi), profiller ise daha üst seviyedeki ve cihaza daha bağımlı kütüphaneleri tanımlar (cihazın üzerindeki ekranda çizilen yüksek seviyeli nesnelere gibi).



Şekil D.1. J2ME uyumlu bir sistemde genel çalışma mimarisi

## **EK D.2. Konfigürasyonlar**

Cep telefonları, KSY'lar ve diğer küçük cihazlar şekil, özellik ve işlevsellik açısından farklılaşmışlardır. Yine de sıklıkla benzer işlemcileri kullanmaktadırlar ve birbirlerine yakın miktarda belleğe sahiptirler. Bu sebepten J2ME tasarımcıları “konfigürasyon”ları yaratmışlardır. Konfigürasyonlar, işlem gücü ve bellek miktarını temel alarak ürünler için bir gruplama tanımlar. Bu doğrultuda konfigürasyonlar aşağıdakilerin ana hatlarını çizer:

- Desteklenen Java programlama dili özellikleri
- Desteklenen Java sanal makinesi özellikleri
- Desteklenen temel Java Kütüphaneleri ve API'leri

Şu anda J2ME dünyasında iki standart konfigürasyon bulunmaktadır: Bağlantılı Cihaz Konfigürasyonu (CDC) ve Sınırlı Bağlantılı Cihaz Konfigürasyonu (CLDC). Aynı zamanda bu konfigürasyonlar, Java API'leri olarak standartlaştırılmışlardır. CDC 1.0, JSR-36 olarak, CDC 1.1, JSR 218 olarak, CLDC 1.0, JSR-30 olarak, CLDC 1.1, JSR 139 olarak tanımlanmıştır<sup>20</sup>.

### **a) CDC**

CDC, aralıklarla bir ağa bağlanabilen, internet televizyonları, otomobil navigasyon sistemleri, benzerlerine göre daha gelişmiş olan KSY ve cep telefonları gibi güçlü cihazları hedef almaktadır. CDC, bugünkü J2SE Java sanal makinesine yakın bir Java sanal makinesine sahiptir. Bu ikisi arasındaki farklılıklar cihazların hafıza ve görüntü birimlerinin özelliklerinden kaynaklanmaktadır.

CDC cihazları için bazı sistem gereksinimleri şöyledir:

- Cihazda 32 bitlik bir işlemci bulunmalıdır
- Cihazda Java için 2 MB veya daha fazla bellek olmalıdır.
- Cihaz Java 2 “Blue book” sanal makinesinin tüm işlevselliğine sahip olmalıdır.

---

<sup>20</sup> <http://www.jcp.org/en/jsr/all>

- Cihazın bir tür ağa, sınırlı bir bant genişliğiyle, sıklıkla kablosuz bir şekilde bağlanabilmelidir.
- Cihaz, kesinlikle zorunlu olmamakla birlikte, belirli bir dereceye kadar bir kullanıcı ara yüzüne sahip olabilir.

## **b) CLDC**

J2ME dünyasındaki ikinci bir tip ve daha yaygın olan konfigürasyon CLDC'dir. Bu konfigürasyon, CDC'de kullanılanlara göre daha zayıf cihazlar (cep telefonları, KSY'lar, kişisel organizatörler gibi) için kullanılmaktadır. CLDC, ilk olarak 1999 yılında, kütüphaneler, güvenlik, Giriş/Çıkış (G/Ç) ve ağ açısından ortak ve en düşük seviyede bir Java platformu oluşturma fikriyle ortaya çıkmıştır.

CLDC cihazları için bazı sistem gereksinimleri şöyledir:

- Cihazda Java platformu için en az 160KB bellek bulunmalıdır.
- Cihazda, pille çalıştırılma gibi, sınırlı miktarda güç bulunabilir.
- Cihazın bir tür ağa, sınırlı bir bant genişliğiyle, sıklıkla kablosuz bir şekilde bağlanabilmelidir.
- Cihaz, kesinlikle zorunlu olmamakla birlikte, belirli bir dereceye kadar bir kullanıcı ara yüzüne sahip olabilir.

## **EK D.3. Sanal Makineler**

Daha önce belirtildiği gibi, CLDC ve CDC konfigürasyonları, Java sanal makinesinden destekledikleri kendi özelliklerini tanımlarlar. Bu yüzden, her biri, kendi sanal makinesine ihtiyaç duyar. CLDC, CDC'den daha az özellik desteklediği için sanal makinesi de daha küçük bir yapıdadır. CLDC sanal makinesi Kilo Sanal Makinesi (KVM) olarak adlandırılırken, CDC sanal makinesine Kompakt Sanal Makinesi (CVM) denmektedir.

### **a) KVM**

KVM küçük cihazlar için bir Java çalışma ortamı sunar. Küçük cihazların uygun biçimde çalışması için gerekli birkaç farklılık dışında, “Java Virtual Machine Specifications”a uygun bir Java sanal makinesidir. Özellikle kısıtlı kaynaklı ve birkaç yüz KB’lık belleğe sahip cihazlar için tasarlanmıştır. KVM için ilk çıkış noktası “Palm Connected Organizer” cihazları için bir Java sanal makinesi geliştirilmesiydi.

### **b) CVM**

CVM daha büyük son kullanıcı ve gömülü sistemler için tasarlanmıştır. Bütün Java 2, versiyon 1.3 sanal makinesi özelliklerini bünyesinde barındırır.

### **EK.D.4. Profiller**

Profiller, uygulamaların cihaza has bazı özelliklere ulaşmasını sağlayan, konfigürasyonların üzerinde çalışan bir grup API’dir. Profiller de konfigürasyonlar gibi standartlaştırılmıştır. Bazı örnekleri şunlardır:

#### **a) MIDP**

Taşınabilir Bilgilendirme Aygıtı Profili (MIDP), CLDC ile kullanılmak için tasarlanmıştır ve cep telefonları ve çağrı cihazları gibi taşınabilir cihazlar için bir kısım API sağlamaktadır. MIDP, kullanıcı arayüzü, kalıcı depolama ve ağ için sınıflar sağlar. Ayrıca yeni uygulamaların indirilebilmesi için standart bir çalışma ortamını içerir. MIDP altında çalışan uygulamalara “MIDLet” denilmektedir.

#### **b) Foundation Profili**

Foundation profili, CDC üzerinde çalışmak için tasarlanmıştır ve CDC API’lerini genişletir. Foundation profili kullanıcı arayüzü için API sağlamaz. Foundation Türkçe’de “temel, kaide” anlamına gelmektedir ve adından da anlaşılacağı gibi diğer profiller (Örneğin Personal profili veya RMI profili) için bir temel oluşturur.

#### **c) Personal Profili**

Personal profili, Java ağ uygulamalarını (applet’ler) çalıştırmaya hazır bir grafiksel kullanıcı arayüzü sağlayacak şekilde Foundation profilini uzatır.



#### d) RMI Profili

RMI profili, Foundation profilini cihazlar için Uzaktan Metot Çağırımı (RMI) desteği sağlayacak şekilde genişletir. RMI profili CDC ile kullanılmak üzere tasarlanmıştır ve CLDC ile kullanılamaz.

Şekil D.2'de yukarıda anlatılan birimlerin ilişkileri gösterilmiştir.



Şekil D.2. Bir J2ME sisteminde tabakalar arası hiyerarşi [19]

## EK E. Hazırlanan Veritabanında Bulunan Difonlar

| Difon | Difon Kodu | Difonun Alındığı Kelime |
|-------|------------|-------------------------|
| _a    | _0a1       | azık                    |
| _b    | _0b0       | bet                     |
| _c    | _0c0       | cam                     |
| _ç    | _0c5       | çalı                    |
| _d    | _0d0       | dere                    |
| _e    | _0e1       | ezel                    |
| _f    | _0f0       | fay                     |
| _g    | _0g2       | gezi                    |
| _h    | _0h0       | his                     |
| _i    | _0i2       | insan                   |
| _ı    | _0i5       | ısı                     |
| _k    | _0k2       | kil                     |
| _l    | _0l1       | lir                     |
| _m    | _0m0       | meze                    |
| _n    | _0n1       | ney                     |
| _o    | _0o1       | oto                     |
| _ö    | _0o6       | öcü                     |
| _p    | _0p0       | pul                     |
| _r    | _0r2       | rom                     |
| _s    | _0s0       | siz                     |
| _ş    | _0s5       | şen                     |
| _t    | _0t0       | vahit                   |
| _u    | _0u1       | ulu                     |
| _v    | _0v1       | vahit                   |
| _y    | _0y1       | yaş                     |
| _z    | _0z1       | zen                     |
| a_    | a1_0       | dava                    |
| aa    | a1a1       | ağaç                    |
| ab    | a1b0       | abla                    |
| ac    | a1c0       | acem                    |
| ad    | a1d0       | vadi                    |
| af    | a1f0       | afet                    |
| ah    | a1h0       | vahit                   |
| aj    | a1j0       | ajite                   |
| al    | a1l2       | sal                     |
| am    | a1m0       | cam                     |
| an    | a1n1       | sanı                    |
| ar    | a1r2       | arı                     |
| ar    | a1r3       | kar                     |
| as    | a1s0       | yastık                  |
| aş    | a1s5       | yaş                     |
| av    | a1v1       | dava                    |
| ay    | a1y1       | fay                     |
| az    | a1z1       | azık                    |
| al    | a2l1       | normal                  |
| ba    | b0a1       | bal                     |
| be    | b0e2       | bet                     |
| bi    | b0i2       | bit                     |
| bl    | b0l2       | abla                    |
| bo    | b0o1       | bol                     |

|      |      |            |
|------|------|------------|
| bö   | b0o6 | bön        |
| ca   | c0a1 | cam        |
| ce   | c0e2 | ece        |
| ci   | c0i2 | tecim      |
| co   | c0o1 | cop        |
| cö   | c0o6 | cömert     |
| cu   | c0u1 | cuma       |
| cü   | c0u7 | öcü        |
| ç_   | c5_0 | ağaç       |
| ça   | c5a1 | çalı       |
| * çe | c5e2 | seçenek    |
| da   | d0a1 | dava       |
| de   | d0e1 | dere       |
| de   | d0e2 | dede       |
| di   | d0i2 | vadi       |
| dı   | d0i5 | kaldı      |
| * do | d0o1 | bidon      |
| du   | d0u1 | durum      |
| dü   | d0u6 | düş        |
| dü   | d0u7 | düğün      |
| eb   | e1b0 | ebe        |
| ec   | e1c0 | tecim      |
| * eç | e1c5 | seçenek    |
| ed   | e1d0 | kedi       |
| ef   | e1f0 | eife       |
| ei   | e1i2 | eğik       |
| el   | e1l1 | elek       |
| em   | e1m0 | yem        |
| en   | e1n1 | zen        |
| er   | e1r2 | dere       |
| eş   | e1s5 | eş         |
| ev   | e1v1 | ev         |
| ez   | e1z1 | meze       |
| e_   | e2_0 | ile        |
| ec   | e2c0 | imece      |
| el   | e2l1 | sel        |
| em   | e2m0 | acem       |
| en   | e2n1 | sen        |
| * es | e2s0 | cumartesi  |
| ey   | e2y1 | ney        |
| * f_ | _0f0 | fotoğraf   |
| fa   | f0a1 | fay        |
| fe   | f0e2 | afet       |
| * fo | f0o1 | fotoğraf   |
| ge   | g2e1 | gece       |
| ge   | g2e2 | ege        |
| * gi | g2i2 | bilgisayar |
| gö   | g2o6 | göz        |
| gü   | g2u6 | gül        |
| * ha | h0a1 | haziran    |
| he   | h0e1 | hece       |
| hi   | h0i2 | vahit      |
| hm   | h0m0 | ahmak      |

|      |      |            |
|------|------|------------|
| * hs | h0s0 | talihsiz   |
| i_   | i2_0 | vadi       |
| * id | i2d0 | bidon      |
| * ih | i2h0 | talihsiz   |
| il   | i2l1 | kil        |
| im   | i2m0 | tecim      |
| in   | i2n1 | sizin      |
| ir   | i2r2 | sirk       |
| ir   | i2r3 | lir        |
| is   | i2s0 | his        |
| * iş | i2s5 | işletme    |
| iy   | i2y1 | piyaz      |
| iz   | i2z1 | sizin      |
| ı_   | i5_0 | kaldı      |
| il   | i5l2 | akıl       |
| * im | i5m0 | kasım      |
| in   | i5n1 | kın        |
| ıs   | i5s0 | ısı        |
| iş   | i5s5 | ışık       |
| ji   | j0i2 | ajite      |
| k_   | k1_0 | azık       |
| ka   | k1a1 | kaldı      |
| kı   | k1i5 | kın        |
| ko   | k1o1 | koza       |
| ku   | k1u1 | kura       |
| k_   | k2_0 | elek       |
| ke   | k2e1 | kedi       |
| ki   | k2i2 | kil        |
| kö   | k2o6 | kör        |
| kü   | k2u6 | öykü       |
| l_   | l1_0 | gül        |
| la   | l1a2 | lapa       |
| ld   | l1d0 | belde      |
| le   | l1e2 | ile        |
| * lg | l1g2 | bilgisayar |
| li   | l1i1 | veli       |
| lm   | l1m0 | gelme      |
| lü   | l1u6 | bölüm      |
| la   | l2a1 | abla       |
| ld   | l2d0 | kaldı      |
| lı   | l2i5 | çalı       |
| lu   | l2u1 | ulu        |
| m_   | m0_0 | durum      |
| ma   | m0a1 | mal        |
| ma   | m0a2 | normal     |
| * mb | m0b0 | çarşamba   |
| * md | m0d0 | şimdi      |
| me   | m0e2 | dümen      |
| mi   | m0i2 | mit        |
| mo   | m0o1 | moda       |
| mu   | m0u1 | muz        |
| mü   | m0u6 | ömür       |
| n_   | n1_0 | on         |
| na   | n1a1 | nal        |
| nc   | n1c0 | kanca      |
| ne   | n1e2 | ney        |
| ni   | n1i2 | yeni       |

|      |      |          |
|------|------|----------|
| nı   | n1i5 | sanı     |
| no   | n1o1 | nova     |
| nö   | n1o6 | nöron    |
| ns   | n1s0 | insan    |
| nu   | n1u1 | nur      |
| nü   | n1u6 | nüve     |
| o_   | o1_0 | oto      |
| ob   | o1b0 | soba     |
| oc   | o1c0 | ocak     |
| od   | o1d0 | moda     |
| ol   | o1l2 | bol      |
| om   | o1m0 | omuz     |
| on   | o1n1 | on       |
| or   | o1r2 | orak     |
| or   | o1r3 | bor      |
| os   | o1s0 | yosma    |
| oş   | o1s5 | boş      |
| ot   | o1t0 | oto      |
| ov   | o1v1 | ova      |
| oz   | o1z1 | koza     |
| oy   | o2y1 | koy      |
| öc   | o6c0 | öcü      |
| öl   | o6l1 | bölüm    |
| öm   | o6m0 | ömür     |
| ön   | o6n1 | bön      |
| ör   | o6r2 | nöron    |
| ör   | o6r3 | kör      |
| öz   | o6z1 | söze     |
| öz   | o6z2 | göz      |
| öe   | o7e2 | öge      |
| öü   | o7u6 | öğün     |
| öy   | o7y1 | öykü     |
| pa   | p0a1 | lapa     |
| * pe | p0e1 | perşembe |
| pu   | p0u1 | pul      |
| ra   | r2a1 | kara     |
| * rb | r2b0 | şerbet   |
| re   | r2e2 | dere     |
| * rg | r2g2 | gergedan |
| rı   | r2i5 | sırık    |
| ro   | r2o1 | nöron    |
| * rş | r2s5 | çarşamba |
| ru   | r2u1 | durum    |
| rü   | r2u6 | sürü     |
| r_   | r3_0 | kar      |
| * s_ | s0_0 | mayıs    |
| sa   | s0a1 | sal      |
| se   | s0e2 | sel      |
| si   | s0i2 | siz      |
| sı   | s0i5 | ısı      |
| sm   | s0m0 | yosma    |
| so   | s0o1 | soba     |
| sö   | s0o6 | söze     |
| sü   | s0u6 | süzme    |
| ş_   | s5_0 | yaş      |
| se   | s0e1 | sele     |
| * şa | s5a1 | şapka    |

|      |      |         |
|------|------|---------|
| şe   | s5e2 | şen     |
| * şı | s5i2 | şimdi   |
| şı   | s5i5 | ışık    |
| * şı | s5i1 | işletme |
| * şu | s5u1 | şubat   |
| t_   | t0_0 | vahit   |
| ta   | t0a1 | tay     |
| te   | t0e2 | ajite   |
| tı   | t0i5 | yastık  |
| * tm | t0m0 | işletme |
| to   | t0o1 | oto     |
| tu   | t0u1 | tuz     |
| u_   | u1_0 | ulu     |
| * ub | u1b0 | şubat   |
| ul   | u1i2 | ulu     |
| um   | u1m0 | durum   |
| ur   | u1r2 | kura    |
| ur   | u1r3 | nur     |
| * us | u1s0 | ağustos |
| uz   | u1z2 | muz     |
| ü_   | u6_0 | sürü    |
| ül   | u6i1 | gül     |
| üm   | u6m0 | bölüm   |
| ün   | u6n1 | gün     |
| ür   | u6r2 | sürü    |

|      |      |        |
|------|------|--------|
| ür   | u6r3 | ömür   |
| üş   | u6s5 | düş    |
| üv   | u6v1 | nüve   |
| üz   | u6z1 | süzme  |
| ün   | u7n1 | düğün  |
| üü   | u7u7 | düğün  |
| va   | v1a1 | vahit  |
| ve   | v1e1 | veli   |
| ve   | v1e2 | nüve   |
| * vi | v1i2 | vida   |
| y_   | y1_0 | ney    |
| ya   | y1a1 | yastık |
| ye   | y1e1 | yem    |
| * yı | y1i5 | mayıs  |
| yn   | y1n1 | ayna   |
| yo   | y1o1 | yoza   |
| yr   | y1r2 | ayran  |
| z_   | z2_0 | kez    |
| za   | z1a1 | koza   |
| ze   | z1e2 | söze   |
| zi   | z1i2 | sizin  |
| zı   | z1i5 | azık   |
| zm   | z1m0 | süzme  |

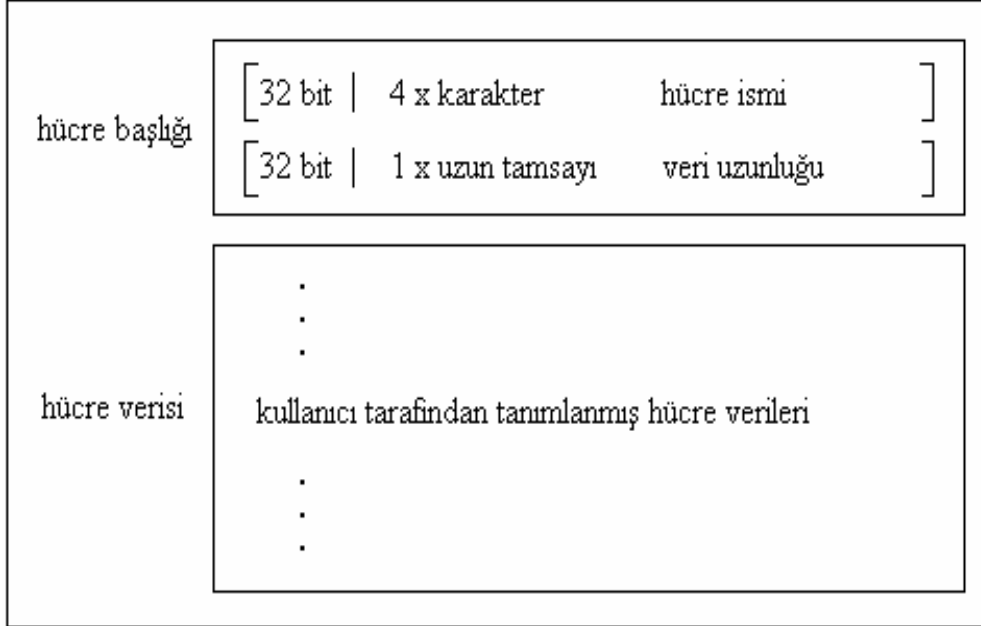
Çizelge E.1. Veritabanında kullanılan difonlar [22]

Özen'in çalışmasında [22] olmayıp bu çalışmada eklenen 30 adet difon '\*' karakteri ile işaretlenmiştir.

## EK F. Veritabanı Dosya Formatı

Veritabanı dosya formatı standart RIFF (Resource Interchange File Format) dosya yapısı Şekil F.1 dikkate alınarak hazırlanmıştır.

riff hücre yapısı

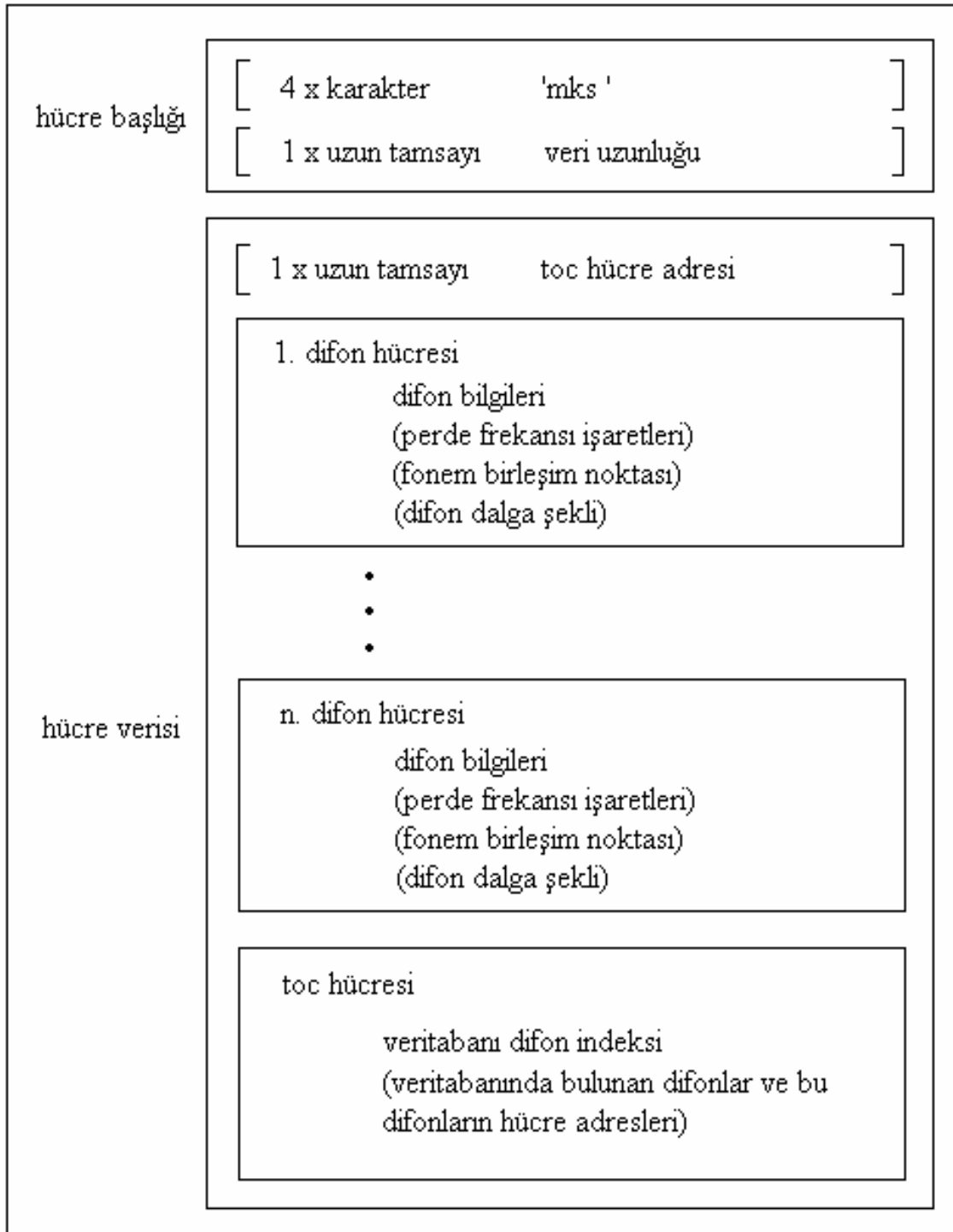


Şekil F.1. RIFF hücre yapısı [22].

Her RIFF hücresi, başlık bloğu ile bu başlığın ilişkili olduğu veri bloğundan oluşmaktadır. Hücre başlığı 4 karakterlik (32 bit) hücre ismi ve hücre veri bloğunun uzunluğunu (byte cinsinden) içeren uzun tamsayıdan (32 bit) oluşur. Hücre veri bloğunun içeriği kullanıcı tarafından tanımlanır.

Bu çalışmada kullanılan veritabanı dosya formatı Şekil F.2'de görülebilir. Veri bloğunun ilk elemanı 'toc' hücresinin adresidir. 'toc' hücresi veritabanında bulunan difonları ve bu difonlara ait bilgilerin bulunduğu difon hücrelerinin adreslerini içerir (Şekil F.3). Veritabanından istenilen bir difona ait bilgilerin alınması için öncelikle 'toc' hücrelerine bakılarak o difonun veritabanında bulunup bulunmadığı saptanır. Eğer difon veritabanında bulunuyorsa, o difona ait hücrenin adresi okunur. Difon hücrelerinin formatı Şekil F.4'de gösterilmiştir. Difon hücreleri, difona ait perde frekansı işaretlerini, difonu oluşturan fonemlerin birleştiği perde frekansı işaretini ve difonun dalga şeklini içerirler. Difon dalga şekli 32 bit kayan noktalı (float) olarak kaydedilmiştir [22].

veritabanı yapısı



Şekil F.2. Veritabanı dosya formatı [22].

toc hücre yapısı

|               |   |
|---------------|---|
| hücre başlığı | [ 4 x karakter 'toc ' ]                                 |
|               | [ 1 x uzun tamsayı veri uzunluğu ]                      |
| hücre verisi  | [ 1 x uzun tamsayı veritabanında bulunan difon sayısı ] |
|               | [ 4 x karakter 1. difon adı ]                           |
|               | [ 1 x uzun tamsayı 1. difon hücre adresi ]              |
|               | ⋮   |
|               | [ 4 x karakter n. difon adı ]                           |
|               | [ 1 x uzun tamsayı n. difon hücre adresi ]              |

Şekil F.3. 'toc' hücre yapısı [22].

difon hücre yapısı

|               |  |        |                         |
|---------------|--|--------|-------------------------|
| hücre başlığı | [ 4 x karakter difon ismi ]  |        |                         |
|               | [ 1 x uzun tamsayı veri uzunluğu ]   |        |                         |
| hücre verisi  | perde frekansı hücresi<br>( difon titreşim frekansı işaretleri )<br>( $k = \text{veri uzunluğu} / 2$ ) | başlık | 'pm '<br>veri uzunluğu  |
|               |  | veri   | k x kısa tamsayı        |
|               | fonem birleşimi hücresi<br>( fonem birleşim yeri bilgisi )   | başlık | 'sep '<br>veri uzunluğu |
|               |  | veri   | 1 x kısa tamsayı        |
|               | difon dalga şekli hücresi<br>( difon dalga şekli verisi )<br>( $L = \text{veri uzunluğu} / 4$ )        | başlık | 'data'<br>veri uzunluğu |
|               |  | veri   | L x float               |

Şekil F.4. Difon hücre yapısı [22].

## EK G. GSM 03.38 SMS Alfabeti

| GSM 03.38 |           |    |    |    |    |    |    |    |    |    |           |            |    |           |    |    |
|-----------|-----------|----|----|----|----|----|----|----|----|----|-----------|------------|----|-----------|----|----|
|           | x0        | x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 | x9 | xA        | xB         | xC | xD        | xE | xF |
| 0x        | @         | £  | \$ | ¥  | è  | é  | ù  | ì  | ò  | Ç  | <u>LF</u> | Ø          | ø  | <u>CR</u> | À  | à  |
| 1x        | Δ         | _  | Φ  | Γ  | Λ  | Ω  | Π  | Ψ  | Σ  | Θ  | Ξ         | <u>ESC</u> | Æ  | æ         | ß  | É  |
| 2x        | <u>SP</u> | !  | "  | #  | ¤  | %  | &  | '  | (  | )  | *         | 0          | ,  | -         | .  | /  |
| 3x        | 0         | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | :         | ;          | <  | =         | >  | ?  |
| 4x        | j         | A  | B  | C  | D  | E  | F  | G  | H  | I  | J         | K          | L  | M         | N  | O  |
| 5x        | P         | Q  | R  | S  | T  | U  | V  | W  | X  | Y  | Z         | Ä          | Ö  | Ñ         | Ü  | §  |
| 6x        | ç         | a  | b  | c  | d  | e  | f  | g  | h  | i  | j         | k          | l  | m         | n  | o  |
| 7x        | p         | q  | r  | s  | t  | u  | v  | w  | x  | y  | z         | ä          | ö  | ñ         | ü  | à  |

Çizelge G.1. GSM 03.38 standardına göre SMS alfabeti<sup>21</sup>

<sup>21</sup> [http://en.wikipedia.org/wiki/Short\\_message\\_service](http://en.wikipedia.org/wiki/Short_message_service)



## ÖZGEÇMİŞ

Adı Soyadı : İlker Ünaldı

Doğum Yeri : Adana

Doğum Yılı : 1983

### Eğitim ve Akademik Durumu:

Lise : 1997-2000 Gaziantep Fen Lisesi

Lisans : 2000-2004 Hacettepe Üniversitesi Elektrik ve Elektronik  
Mühendisliği Bölümü, 2004

Yabancı Dil: İngilizce

### İş Tecrübesi:

2006 - ... Yazılım Test Mühendisi, AYESAS, ANKARA