

T.C
İSTANBUL AYDIN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



İĞNECİKLİ SİNİR AĞLARI İLE SES KAYNAĞI SINIFLANDIRMA

YÜKSEK LİSANS TEZİ

Hazırlayan

Peren Jerfi CANATALAY

Elektrik Elektronik Mühendisliği

Ana Bilim Dalı

Elektrik Elektronik Mühendisliği

Programı

ŞUBAT, 2016

T.C
İSTANBUL AYDIN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



İĞNECİKLİ SİNİR AĞLARI İLE SES KAYNAĞI SINIFLANDIRMA

YÜKSEK LİSANS TEZİ

Hazırlayan

Peren Jerfi CANATALAY
Y1313.10002

Elektrik Elektronik Mühendisliği

Ana Bilim Dalı

Elektrik Elektronik Mühendisliği

Programı

Tez Danışmanı: Prof. Dr. Osman Nuri UÇAN

ŞUBAT, 2016



T.C.
İSTANBUL AYDIN ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ MÜDÜRLÜĞÜ

Yüksek Lisans Tez Onay Belgesi

Enstitümüz Elektrik-Elektronik Mühendisliği Ana Bilim Dalı Elektrik-Elektronik Mühendisliği Tezli Yüksek Lisans Programı Y1313.100002 numaralı öğrencisi **Peren Jerfi CANATALAY**'ın "İĞNECİKLİ SİNİR AĞLARI İLE SES KAYNAĞI SINIFLANDIRMA" adlı tez çalışması Enstitümüz Yönetim Kurulunun 29.01.2016 tarih ve 2016/04 sayılı kararıyla oluşturulan jüri tarafından *aybulye* ile Tezli Yüksek Lisans tezi olarak *Kabul* edilmiştir.

Öğretim Üyesi Adı Soyadı

İmzası

Tez Savunma Tarihi :11/02/2016

1)Tez Danışmanı: Prof. Dr. Osman Nuri UÇAN

2) Jüri Üyesi : Prof. Dr. Mehmet Emin TACER

3) Jüri Üyesi : Prof. Dr. Sedef KENT

Not: Öğrencinin Tez savunmasında **Başarılı** olması halinde bu form **imzalanacaktır**. Aksi halde geçersizdir.

YEMİN METNİ

Yüksek Lisans tezi olarak sunduğum “İĞNECİKLİ SİNİR AĞLARI İLE SES KAYNAĞI SINIFLANDIRMA” adlı çalışmanın, tezin proje safhasından sonuçlanmasına kadarki bütün süreçlerde bilimsel ahlak ve geleneklere aykırı düşecek bir yardıma başvurulmaksızın yazıldığını ve yararlandığım eserlerin Bibliyografya’da gösterilenlerden oluştuğunu, bunlara atıf yapılarak yararlanılmış olduğunu belirtir ve onurumla beyan ederim. (9/02/2016)

Aday / İmza



ÖNSÖZ

Bu çalışma İstanbul Aydın Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Bölümünde hazırladığım, ses kaynağı konumlarının bulunması ve sınıflandırılması üzerine olan yüksek lisans tezimin sonucu olarak tamamlanmıştır.

Yüksek lisans öğrenimim sırasında ve tez çalışmalarım boyunca gösterdiği her türlü destek, öneri ve paylaştığı görüşlerinden dolayı çok değerli hocam Prof. Dr. Osman Nuri Uçan'a en içten dileklerle teşekkür eder ve maddi manevi her zaman yanımda olan aileme desteklerinden ve inanclarından dolayı sonsuz tesekkürü bir borç bilirim.

Eğitimim boyunca çalışmalarımı destekleyen İstanbul Aydın Üniversitesi ve Elektrik-Elektronik Mühendisliği Bölümüne teşekkürü borç bilirim. Çalışmamın tüm ilgililere yararlı olmasını dilerim.

Şubat 2016

Peren Jerfi CANATALAY

İÇİNDEKİLER

SAYFA

ÖNSÖZ.....	v
İÇİNDEKİLER	vii
ÇİZELGE LİSTESİ.....	ix
ŞEKİL LİSTESİ.....	xi
ÖZET.....	xiii
ABSTRACT	xv
1. GİRİŞ	1
1.1 Radar ve Sonar Sistemler	1
1.2 Tıp Uygulamaları	1
1.3 Kablosuz Haberleşme.....	2
1.4 Konum İyileştirme	4
2. LİTERATÜR ÖZETİ	5
3. YÖNTEM.....	12
3.1 Ses Yayılımı	13
3.2 Demet Oluşturma	156
3.3 Yer Bulma	188
4. İĞNECİKLİ SİNİR AĞLARI.....	23
5. SİNİR HÜCRESİ MODELLERİ	26
5.1 Hodgkin-Huxley Modeli	26
5.2 Tümlev-Al-Ateşle Model Ailesi	28
5.3 İğnecik Cevap Modeli	29
6. AĞ YAPILARI.....	31
6.1 İleri Beslemeli Ağlar	31
6.2 Geri Beslemeli Ağ Yapıları.....	32
6.3 Rezervuar Hesaplama.....	34
7. ÖRNEK İŞLEM	34
8. UYGULAMA VE BENZETİM ÇALIŞMALARI.....	40
9. SONUÇ.....	46
KAYNAKLAR	51
EKLER.....	53
ÖZGEÇMİŞ.....	65



ÇİZELGE LİSTESİ

	<u>SAYFA</u>
Çizelge 8.1 : Konuşmacı konumları	42
Çizelge 8.2 : Mikrofon konumları.	42
Çizelge 8.3 : Tespit edilen konuşmacı konumları.	45
Çizelge 8.4 : Konum hataları (mutlak)	46
Çizelge 8.5 : x y ve z eksenlerinde maksimum hatalar.....	46





ŞEKİL LİSTESİ

	Sayfa
Şekil 2.1 : Ses Dalgalarında Yayılma.....	5
Şekil 2.2 : Her yönlü.....	9
Şekil 2.3 : İki yönlü.....	10
Şekil 2.4 : Kardioid.....	11
Şekil 3.1 : Ses dalgasının mikrofona dizisine farklı zamanlarda varışı.....	13
Şekil 3.2 : Gürültü Kaynakları ve Mikrofon Dizileri 1.....	16
Şekil 3.3 : Geciktir ve Topla Tekniği.....	17
Şekil 4.1 : Biyolojik sinir hücresi çizimi.....	23
Şekil 4.2 : İlk yapay sinir modeli.....	24
Şekil 4.3 : İlk kuşak sinir modellerinde kullanılan etkinleştirme işlevleri.....	24
Şekil 4.4 : Örnek bir iğnecik raster grafiği.....	26
Şekil 5.1 : Hodgkin-Huxley modeli eylem potansiyeli.....	28
Şekil 5.2 : İğnecik Cevap Modeli.....	29
Şekil 5.3 : İğnecik Cevap Modeli Çıktısı.....	30
Şekil 6.1 : İleri Beslemeli Ağ Yapısı.....	31
Şekil 6.2 : Geri Beslemeli Ağ Yapısı.....	33
Şekil 6.3 : Rezervuar Hesaplama İçin Bir Ağ Yapısı.....	34
Şekil 7.1 : Ağ yapısı ve adaptasyon.....	37
Şekil 7.2 : Alıcı alanlar.....	38
Şekil 7.3 : Sınıflandırma Örneği.....	39
Şekil 8.1 : Konuşmacı konumları.....	41
Şekil 8.2 : Konuşma ses dalgaları.....	44
Şekil 8.3 : Sınıflandırma sonuç gösterimi.....	46



İĞNECİKLİ SINIR AĞLARI İLE SES KAYNAĞI SINIFLANDIRMA

ÖZET

Bu yüksek lisans tezinde, mikrofon dizilerinden alınan verilerin işlenmesi ile ses kaynağının yer tayini üzerine çalışılmıştır. Ses kaynağı konumu belirlemede başarıyı en yüksek yöntemlerden biri olan faz dönüşümlü yönlendirilmiş cevap gücü algoritması üzerine yoğunlaşmıştır. Mikrofon dizileri ile yapılan kayıtlarda kaynakların ayrı ayrı dinlenebilmesi için odaklama yapılması gerekmektedir ve bunun için de ses kaynağı konumunun kesin bir şekilde bulunabilmesi gerekmektedir. Bu amaçla konum bilgilerinin değerlendirilmesi ve yer tayini işlemi için iğnecikli sınır ağı ile sınıflandırma yapılmıştır. Elde edilen sonuçlar, oldukça yüksek bir başarıya işaret etmektedir. Belirlenen konumlar, bilinen gerçek konuşmacı konumları ile oldukça uyumludur ve hata oranları birkaç santimetreyi aşmamaktadır.

Anahtar Kelimeler: *İğnecikli sınır ağı, Ses kaynağı yer bulma*



AUDIO SOURCE CLASSIFICATION BY MEANS OF NEEDLE NEURAL NETWORK

ABSTRACT

In this master of science thesis, we have worked on determining sound source positions via processing data obtained from microphone arrays. We have focused on phase transformed steered response power algorithm which is one of the most powerful techniques on sound source localization. To listen sound sources separately, focusing is needed which requires the exact position knowledge of source localization when microphone arrays are used, for this purpose, to evaluate position information and localization, classification is incorporated using a spiking neural network. Results obtained indicate very a good performance. Determined positions are quite comparable with the known positions of five talkers, thus calculated positional errors are in a few centimeters range

Keywords: *Spiking neural network, Sound source localization*





1. GİRİŞ

Ses ve/veya konuşma kaydetmede ortam koşulları oldukça belirleyici olan olumsuz etkilere sahiptir. Bu etkilerin bertaraf edilmesi birden çok mikrofon kullanılması sayesinde sağlanabilmektedir. Birden çok mikrofonun kullanımı, mikrofonların yerleşim biçimlerine göre ortam içerisindeki tüm sesleri duyabilme olanağı vermektedir. Ancak, ortamdaki tüm seslerin duyulabilmesi, tüm seslere hakim olma anlamına gelmemektedir. Zira ortamda birden çok ses kaynağı bulunması ve/veya ortamın yankılama etkisinin yüksek olması gibi durumlarda sesler birbirine karışmakta, kimi sesler örtülmekte, geri planda kalmakta ve anlaşılammamaktadır. Bu durum genel olarak kokteyl partisi problemi olarak adlandırılmaktadır. Çözüm ise dizisel işaret işleme tekniklerinin çoklu mikrofon kayıtlarına uygulanmasıdır.

Bir diğer önemli problem de ses kaynağı ve/veya konuşmacının hareketli olması durumu veya bir başka ifadeyle kayıt esnasında yer değiştiriyor olması durumudur. Bu durumda da kayıt edilen ses ve/veya konuşmanın kalitesi sabit olamamakta ve zaman içerisinde çeşitli zayıflamalara maruz kalmaktadır. Nitekim her farklı konum, farklı gürültü kaynakları, bozucular ve/veya yansıtıcı nesnelere beraberinde getirmektedir. Bu tür bir problemin çözümü de yine dizisel işaret işleme yöntemlerine başvurarak çoklu mikrofon kayıtlarının işlenmesi olarak ortaya çıkmaktadır.

Dizisel işaret işleme, temel olarak birçok algılayıcıdan alınan verilerin, algılayıcıların uzaysal yerleşimlerine göre işlenmesi olarak tanımlanabilir. Sonuçta hedeflenenler ise genel olarak kaynakların sayı ve konumlarının belirlenmesi, işaret gürültü oranının iyileştirilmesi ve birden fazla hareketli kaynağın takibi olarak özetlenebilir. Radar, sonar, sismik araştırmalar, tomografi ve kablosuz iletişim gibi alanlarda kullanılan dizisel işaret işleme, hem alıcı hem de verici dizileri için uygulanabilmektedir. Bu durumda yayılmak istenen işaretler belirlenen yönelimdeki

belirlenen hedefe ulařtırılabilmektedir. Dizisel iřaret iřlemenin gerekleřtirilebilmesi iin bazı n kabuller yapılmaktadır. Bunlardan ilki, kaynaktan ıkan enerjinin tm ynlere dzenli bir Őekilde yayıldıđının ve ortamın enerjiyi dađıtmadıđını varsayılmasıdır. Bir diđer n kabul ise, iřaretle birlikte ortalaması sıfır olan bir beyaz grltnn varlıđıdır ki bu beyaz grlt iřaretle ilintili deđildir.

Kaynak sayısı, ynelim ve konum belirleme sorunlarının zmne ihtiya duyan pek ok alan mevcuttur. Bu kısımda, ne ıkan dizisel iřaret iřleme uygulama alanlarına kısaca deđinilecektir.

1.1 Radar ve Sonar Sistemler

Dizisel iřaret iřleme uygulamalarının bařlangı noktasını radar ve sonar sistemlerinin geliřtirilmesi oluřturmuřtur. Bu sistemlerde, kaynak veya kaynakların yerlerinin bulunabilmesi ve giriřimlerin nlenebilmesi iin anten dizileri kullanılmaktadır [1] ve [2].

Radar sistemleri, temel olarak radyo dalgalarını kullanmak suretiyle nesneleri tespit etmek iin kullanılır. Nesnelerin ynleri, ykseklikleri, hızları ve uzaklıkları bu Őekilde belirlenebilmektedir. Radar sistemleri bařta askeri donanımlarda yer almıř, sonradan sivil hayata girmiřtir. Radar uygulamalarında kullanılan aktif modlu anten dizisi, darbe Őeklinde dalgaları yayar ve sonra bu dalgaların geri dnřlerini dinler. Geri dnřlerin dinlenmesi sayesinde, hedefe ait hız, uzaklık ve geliř yn gibi parametrelerin belirlenebilmesi mmkn hale gelmektedir. Pasif modlu anten dizilerinde ise, hedefe ait olan parametrelerden yalnızca geliř yn parametresi belirlenebilmektedir [1] ve [2].

Sonar sistemleri, ses dalgalarının su altında yayılımından faydalanarak deniz altında veya yzeyinde bulunan nesneleri tespit amacı ile kullanılır. Radar sistemlerinde olduđu gibi sonar sistemlerde de aktif ve pasif modlu alıřma sz konusudur.

1.2 Tıp Uygulamaları

Dizisel iřaret iřleme yntemlerinin tıp alanında birok karřılıđı bulunmaktadır. Bařta gelen uygulama alanı tıbbi imge iřlemedir. Nitekim, manyetik rezonans, nkleer grntleme ve bilgisayarlı tomografi gibi grntleme tekniklerinde birok algılayıcının bulunduđu diziler ve/veya matrisler kullanılmakta, bu algılayıcılardan elde edilen veriler dizisel iřaret iřleme teknikleri ile

anlamlandırılabilen, dolayısıyla hastalık teşhisi gibi çıktılarına ulaşılabilir. En çok bilinen örnekler, ultrasonik görüntüleme, elektrokardiografi, elektroensefalografi, NMR (Nükleer Manyetik Rezonans) ve bilgisayarlı tomografi olarak sıralanabilir. Bu tür görüntüleme uygulamalarının yanı sıra tıpta hastalık tedavisi, kalp gibi iç organların durumları hakkında bilgi içeren dalga şekillerinin takip edilmesi ve beyin etkinliklerinin konumlandırılması ve çözümlenmesi gibi önemli uygulamalarda da dizisel işaret işleme teknikleri kullanılmaktadır. Bu tür uygulamalarda ultrasonik, manyetik, nükleer ve elektriksel dalga kaynaklarından yararlanılmaktadır [3].

1.3 Kablosuz Haberleşme

Geçtiğimiz 10-20 yıl içerisinde kablosuz haberleşme sistemleri çok büyük bir hızla gelişme göstermiştir. Bu gelişimde haberleşme teorisi yanı sıra düşük güçte çalışan cihazların tasarlanması ve üretilmesinin payı büyüktür. Temel olarak kablolu ve kablosuz haberleşmede sırasıyla elektriksel işaretler ve elektromanyetik dalgalar kullanılmaktadır. Anten dizileri ise, kablosuz haberleşmede klasik zaman ve frekans boyutlarına ek olarak uzay boyutunun da eklenmesini sağlamış, bu da kablosuz haberleşmede hatanın büyük ölçüde azaltılmasını sağlamıştır. Son 10 yılda ise, dizisel işaret işleme ve tahmin teknikleri, kablosuz haberleşmedeki sayısız problemin çözümü için çare olarak görülmeye başlanmıştır. Kablosuz haberleşme sistemlerinde, sistemin kalite ve performansını etkileyen problemler farklı kaynaklardan gelebilir.

Çok kullanıcı haberleşme durumu düşünüldüğünde, çok sayıda kullanıcı arasında kullanıcılar arası girişim olasılığı oldukça yüksektir ve bu da sistem performansını olumsuz etkilemektedir. Mobil haberleşme sistemlerinde, çoklu yol sorunu baz istasyonlarının çözmesi gereken en temel sorunlardan biridir. Baz istasyonları şiddetli çoklu yoldan kaynaklanan sönümlenme ile savaşmak için uzaysal çeşitliliği kullanmaktadır. Bunun için birkaç elemandan oluşan bir anten dizisi kullanan baz istasyonları daha yüksek seçicilik elde edebilmektedir. Alıcı anten, dizisel işaret işleme teknikleri kullanılarak bir süre için yalnızca tek bir kullanıcıya yönlendirilebilmekte, bu da başka kullanıcılardan kaynaklanabilecek girişimlerin önüne geçmektedir [3].

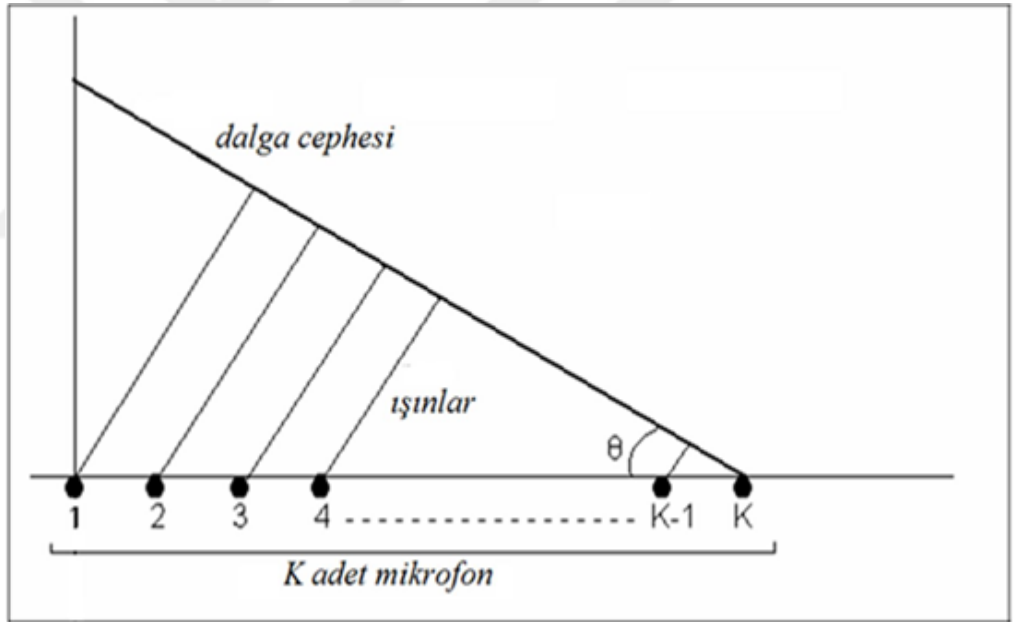
1.4 Konum İyileştirme

Konuşma iyileştirme ve işleme de dizisel işaret işlemenin yer aldığı alanlardan birisidir. Tümüyle otomatik sistemler haline gelen akustik ara yüz sistemlerinin işletim ortamları birçok başka akustik kaynağı da içermektedir. Bunlar, harici gürültüler olabileceği gibi arzu edilen konuşma işaretini zayıflatan ve bastıran hoparlör işaretlerinin akustik kuplajı da olabilmektedir. Bu harici kaynaklara ek olarak konuşmacı ve mikrofon arasındaki bağıl uzaklık da istenilen işaretin gücünün düşmesine neden olmaktadır. Dizisel işaret işleme teknikleri, konuşma işaretinin olumsuz etkilenmeden ve kalitesinden ödün vermeden gürültü ve yankının bastırılmasına olanak tanımaktadır. Genel olarak dizisel işaret işleme yöntemleri konuşma iyileştirme problemi üzerinde hesapsal gücü azaltmada ve sistem performansının kalitesini arttırmada kullanılmaktadır. İşareti alt bantların bir toplamı olarak ifade etmek ve alt bant işaretleri için kaldırma süzgeçleri uyarlamak talep edilen hesapsal gücü azaltmakta ve daha yüksek performanslı bir sisteme olanak tanımaktadır. Çoklu giriş kanalının bulunması, daha yüksek kalitede sistem tasarlanmasının önünü açmaktadır. Bu da tek bir kanal ile başaramayacak olan; yer bulma, takip ve ayırma gibi problemlerin çözümünü kolaylaştırmaktadır

2. LİTERATÜR ÖZETİ

Bu bölümde, mikrofon dizileri kullanılarak ve dizisel işaret işleme yöntemleri “uygulanarak ses kaynağının yerini belirleme üzerine yapılan çalışmalar gözden geçirilmiş ve özetlenmiştir. İlk aşamada faz farklı dizilerden ve karşılaşılan güçlüklerde söz edilecek ve demet oluşturma etkisine yer verilecektir.

Faz aralıklı dizilen mikrofonlar için [4] numaralı kaynakta da belirtildiği gibi şöyle bir temel model öne sürülmektedir. Şekil 2.1’de görüldüğü gibi K adet mikrofondan oluşan bir alıcı dizisi θ açısı altında bir düzlemsel ses dalgasına maruz kalıyor olduğu durumu ele alalım.



Şekil 2.1 : Ses Dalgalarında Yayılma

Burada dalga cephesi mikrofon dizisine belirli bir θ açısı ile yaklaştığından dolayı ses dalgası dizideki mikrofonların her birine aynı zamanda ulaşmayacaktır. Alıcılar arasında d mesafesinin bulunduğunu ve bu mesafenin eşit olduğunu kabul edersek

ses dalgası K mikrofonuna ulaştığında, K-1 mikrofonuna ulaşmak için fazladan bir $d \cdot \sin(\theta)$ kadar yol kat etmesi, K-2 numaralı mikrofonu ulaşmak için ise $2d \cdot \sin(\theta)$ 'lık bir mesafeyi aşması gerekmektedir.

Ses dalgalarının zamanda ve uzayda periyodik olduğu varsayılırsa, mikrofon dizisine ulaşan ses dalgaları farklı alıcılarda farklı fazlarla algılanacaktır. Her θ açısı için değişik faz farkı kombinasyonları elde edilecektir. Dolayısı ile mikrofon dizisindeki her bir elemandan alınan kayıta faz kaydırma uygulanabilir. Bu durumda dizideki tüm elemanlardan alınan ve fazları kaydırılmış olan kayıtlar toplandığında, belirli bir θ açısı için olumlu bir birleşim meydana gelecektir.

Alıcılar arasında bırakılan boşluklar mikrofon dizisinin performansında belirleyici bir etken teşkil eder. [4] numaralı kaynakta da belirtildiği gibi doğrusal bir dizide mikrofonlar arasındaki mesafe arttıkça ana ışın demeti küçülmektedir. Bununla birlikte alıcılar arasındaki boşlukların artırılması gelişigüzel gerçekleştirilemez. Zira, kritik uzaklık aşıldığında ekstra ana demetler ortaya çıkacaktır.

Yukarıda ses dalgasının düzlemsel olduğunu varsaymıştık. Bu varsayım mikrofonların, ses kaynağının uzak alan bölgesine yerleştirilmesi durumunda geçerlilik kazanır. Bu bölge için koşul ise şöyle ifade edilir [5].

$$d = \frac{2L^2}{\lambda} \quad (2.1)$$

Burada, kritik d mesafesinin elde edilmesi, kaynağın açıklığı 'L' ve dalga boyu ' λ ' parametrelerine bağlıdır.

Belirli bir θ yönünden gelen tüm ses kaynaklarını algılayabilmek için mikrofonların yönlendirilmesi işlemine demet oluşturma adı verilir [5]. Demet oluşturma işleminde parametrelerin belirlenmesi, uyarlanabilir veya sabit olarak ikiyolla gerçekleştirilebilir. Sabit yöntemlerde parametreler başta bir kez tanımlanırken uyarlanabilir yöntemlerde alınan işaretlere bağlı olarak parametreler uyarlanır ve güncellenir. Bununla birlikte, demet oluşturma tekniklerinde ortamdaki gürültü ile ilgili yapılan kabuller de belirleyici rol oynar. Gürültü için belirli bir istatistiksel dağılım fonksiyonu çoğu zaman işe yarar ise de her zaman verimli sonuçlar doğurmayabilmektedir.

Yönelimli işitsel algılama insan ve diğer hayvanlarda doğuştan gelen bir yetenektir.

Bu algılama yeteneği, algılayıcı organların işleyişine bağlı olduğu gibi ses dalgalarının uzayda yayılma biçimine de bağlıdır [6]. Ses algılayışımızda sağdan ve soldan gelen sesleri ayırabildiğimiz gibi seslerin yukarıdan veya aşağıdan gelip gelmediğini de ayırabilmekteyiz. Bu, ses kaynağının yerini bulma anlamında 3 boyutlu bir fonksiyon ortaya koymaktadır [7].

Ses kaynağının yerini bulmak için iki temel içerikten yararlanılır. Bunlar mikrofonlar arasındaki genlik ve faz farklarıdır [6, 8 ve 9]. Bu veriler elde edildiğinde ses kaynağının konumu elde edilebilir. Ancak bu aşamada bir kritik nokta daha vardır. Bu nokta veri toplama sistemidir ki, mikrofon dizisinin performansını doğrudan etkiler. Veri toplama sistemi kaydedilen analog işaretleri dijital veriye çevirerek bilgisayar ortamına aktarır. Burada da örnekleme kriteri – Nyquist koşulunun sağlanması gerekmektedir [6 ve 9]. Kayıtların bilgisayar ortamına aktarılması ile işaret işleme tekniklerinin kullanılması aşamasına geçilebilir. Örneğin hızlı Fourier dönüşümü kullanarak elde edilen işaretin frekans içeriği belirlenebilir, kısa dönem Fourier dönüşümü ile de frekans içeriğindeki değişim takip edilebilir [6 ve 8]. Bunun dışında, sabit mikrofonlar arası mesafede dizideki eleman sayısını arttırmak dizinin akustik bant genişliğini azaltmaktadır [10]. Bir başka ifadeyle, dizideki mikrofon sayısının artması, sistemin frekans seçiciliğini arttırmakta ve belirlenen yönelim için daha düşük bir frekans aralığının kayıt edilebilmesi anlamına gelmektedir. Sonuç olarak optimum mikrofon sayısının belirlenmesi önemli bir adım olarak belirmektedir. Dizideki mikrofon sayısını belirlerken sayının, dar bir yönelim sağlayacak kadar çok, ancak insan sesini kapsayan bant genişliğine sahip olacak kadar az olması gerekmektedir.

Birçok uygulamada hareketli ses kaynağının yerinin dinamik bir şekilde tespit ve takip edilmesi için robotlar kullanılmıştır [6]. Bu tür sistemlerde daha hassas yer belirleme için imge işleme yöntemlerine de yer verilmektedir. Genel olarak gerçekleştirilen uygulama çalışmalarında, ses kaynağının yerini belirlemede kuramsal açıdan net bir başarı elde edilmesine karşın deneysel uygulamaya geçildiğinde gürültü, yankı ve üst üste binme gibi bozucu etkilerden dolayı başarımların yüzdeleri düşmektedir [7].

Kayıt ortamının, birçok parametreyi şekillendiren bir faktör olması nedeniyle oda akustiği ile ilgili birtakım temel husulara değinmek gerekmektedir [11]. Genel olarak bir odanın akustiğini belirlemede, oda frekans ve bant genişliği özelliklerine göre 4

ayrı bölgeye ayrılmaktadır. Bu bölgeler, X, A, B ve C bölgeleri olarak adlandırılmaktadır. X bölgesi 0 ile f_X frekans aralığını kapsar ve f_X frekansı şöyle ifade edilir,

$$f_X = \frac{c}{2L} \quad (2.2)$$

Burada L parametresi, odanın en büyük boyutudur. Bu frekans aralığı mikrofon ile algılanan aralığın dışında kalır, bu yüzden de bu bölge bilinmeyen anlamında X ile tanımlanmıştır. Sıradaki bölge olan A bölgesi, f_X ile f_A frekans bölgeleri arasında kalan banttır. Bu frekans aralığı oda boyutlarına bağlı olan rezonans frekansları ile karakterize edilir ve f_A frekansı şöyle ifade edilir,

$$f_A = 11250 \sqrt{\frac{T}{V}} \quad (2.3)$$

Burada, V odanın hacmi, T de yankılanma süresidir. B frekans bölgesi için ise,

$$f_B = 4 f_A \quad (2.4)$$

ifadesi kullanılmakta ve bu bölge baskın difüzyon ve kırınım ile karakterize edilmektedir. Son olarak, C frekans bölgesi ise f_B frekansından daha yüksek frekanslar için tanımlanır. Bu bölge yankılanma ile karakterize edilir.

Bilgi ileten işaretlerde gürültü girişimi bozulmaya neden olmakta ve uygulamada doğru olmayan sonuçların elde edilmesine yol açmaktadır [12]. Özellikle ses ve konuşma kayıtları bağlamında gürültü oluşturan başlıca kaynaklar: dahili ve/veya harici mekanik cihazların sesleri ve kayıt odasının iyi yalıtılmaması halinde diğer odalar ve/veya dışarıdan gelen seslerdir. Bununla birlikte, mikrofon dizilerinin üç temel gürültü tipine maruz kaldıkları belirtilmektedir [5]. Bu gürültü tipleri farklı konumlarda bulunan gürültü kaynaklarının birbirleri ile olan ilintileri bağlamında açıklanmaktadır.

Eş evreli gürültü: Eş evreli gürültü alanları, gürültü kaynağından mikrofon dizisine doğrudan yol alan gürültü işaretleridir. Çevresel koşullara bağlı olmaksızın

yansımaya veya dağılmaya uğramazlar. Eş evreli alanlarda, mikrofon dizisine uygulanan gürültü girdileri kuvvetli bir şekilde ilintilidir.

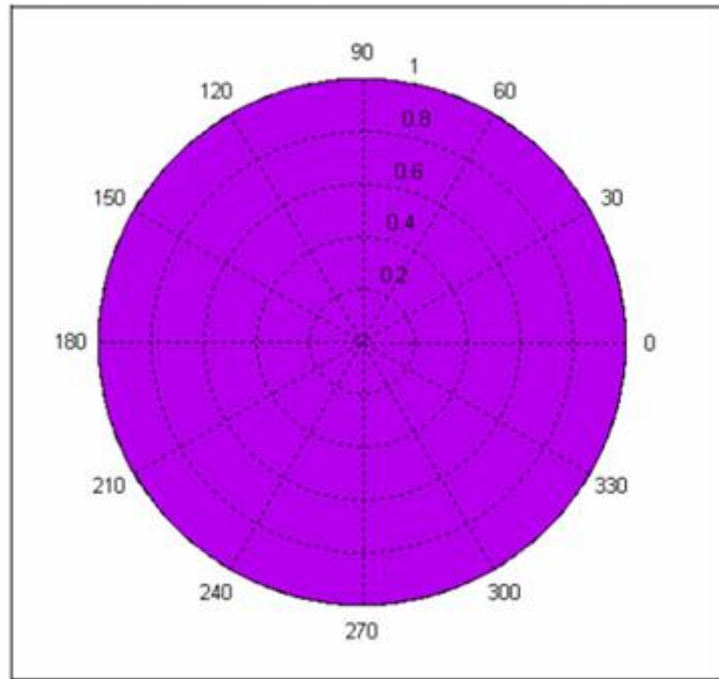
Eş evreli olmayan gürültü: Eş evreli olmayan gürültüye örnek olarak mikrofon dizisi ve birlikte kullanılan elektronik donanımın kendi gürültüsü gösterilebilir. Aynı zamanda uzaysal beyaz gürültü olarak da adlandırılan bu tip gürültüler rasgele kabul edilebilir, zira ilintileri sıfıra çok yakındır.

Dağılan gürültü: Dağılan gürültü alanları, ortamda her yerde yaklaşık aynı enerji seviyesinde, düşük oranda ilintili gürültü alırlar. Gelen gürültü çoğunlukla bu türdür ve dağılan gürültü olarak karakterize edilir.

Bunlar dışında gürültü tanımlama için bazı renk adlandırmaları yapılmıştır [10]. Bunlar da şöyledir:

Beyaz gürültü: İzgesel güç yoğunluğu sabit olan ve tüm spektrum için frekanstan bağımsız olan gürültü tipidir.

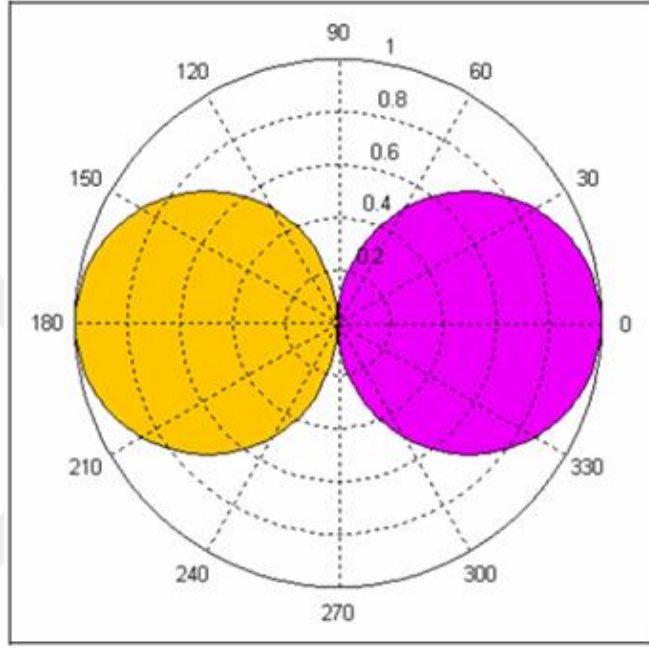
Pembe gürültü: $1/f$ gürültüsü olarak da adlandırılır. Bu adlandırmadan da anlaşılacağı üzere frekansa bağlı olan ve bu yüzden beyaz olmayan gürültü türüdür.



Şekil 2.2 : Her yönlü

Mikrofon dizilerinde en sık kullanılan mikrofon tipleri kapasitif mikrofonlar ve

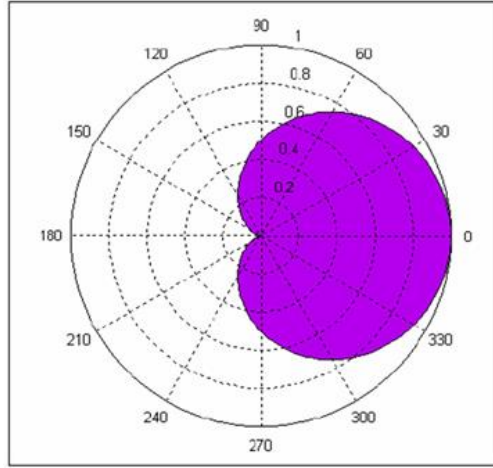
dinamik mikrofonlar olarak özetlenebilir. Kapasitif mikrofonun çalışma prensibi fiziksel olarak ses dalgalarının bir kondansatörün plakalarını titreştirmesine dayanmaktadır. Bu titreşimler sonucunda meydana gelen gerilim değişimleri sayesinde ses algılanır. Dinamik tip mikrofonlarda ise manyetik alan içerisine bir bobin yerleştirilmiştir. Bu bobin ses dalgaları ile karşılaştığında yine titreşme meydana gelir, ancak bu kez elektrik akımı üretir ve bu sayede ses algılanır.



Şekil 2.3 : İki yönlü

Kapasitif mikrofonlar dinamik mikrofonlar ile karşılaştırıldığında, kapasitif mikrofonlar küçük ölçekli basınç değişimlerine karşı çok daha hassas tepki verirler, ancak bunun yanında çok daha narin ve pahalıdırlar. İnsan sesi, dolayısı ile konuşma kaydediliyorsa, kapasitif mikrofonlara ve de yüksek hassasiyete ihtiyaç yoktur, dinamik mikrofonlar kullanılabilir.

Mikrofonların fiziksel özellikleri yanında yönelim özellikleri de mevcuttur. Yönelimlerine göre her yönlü, iki yönlü ve kardiooid olarak sınıflandırılabilirler. Bu üç yönelime ilişkin yönelim diyagramları sırası ile Şekil 2.2, Şekil 2.3 ve Şekil 2.4'te gösterilmiştir.



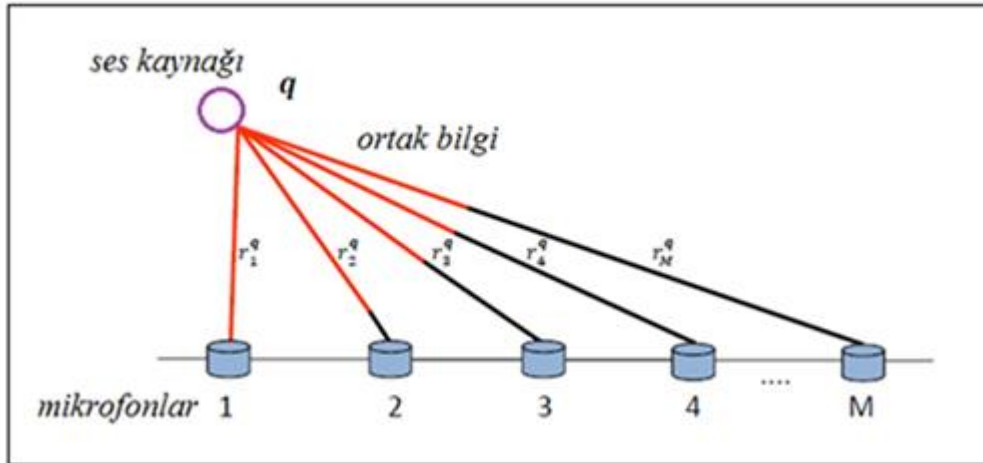
Şekil 2.4 : Kardioid



3. YÖNTEM

Bu bölümde çalışmamızda sunduğumuz algoritma ve bu algoritmanın anlaşılmasını kolaylaştıracak temel teknikler izah edilecektir.

Önceden de tanımlandığı üzere bir mikrofon dizisi, birden çok mikrofondan oluşan ve kayıt edilen akustik işaretlerden uzaysal bilginin elde edilmesini sağlayacak şekilde konumlandırılan bir çoklu algılayıcı sistemidir [13]. Akustik kayıt alma sistemi olarak kullanılan bir mikrofon dizisi ile gelen ses dalgalarının uzaysal özellikleri verimli bir şekilde ortaya çıkarılabilir ve bu işlem tek bir mikrofon ile mümkün olmayan bir işlemdir. Bu işlemin gerçekleşebilmesi mikrofon dizisinin her bir elemanına gelen ses dalgaları arasındaki zaman farkına dayanmaktadır. Varyasyonlu zaman farkı olarak adlandırılabilen bu olgu, mikrofonlara ulaşan bilgi işaretleri arasında meydana gelen farklı zaman gecikmeleri olarak açıklanmaktadır. Bu olgu, mikrofonlar arasındaki fiziksel uzaklıkların bir sonucudur ve dizisel işaret işleme algoritmalarının üzerine bina edildiği, algılayıcı dizilerinin pek çoğu için geçerli olan en temel özelliktir.



Şekil 3.1 : Ses dalgasının mikrofon dizisine farklı zamanlarda varışı

Şekil 3.1’de verilen görselleştirmede; “M” mikrofon sayısını, “q” ses kaynağının konumunu, “ r^q ” ise m. mikrofondan ses kaynağına olan uzaklığı ifade etmektedir. Bu durumda “ r^q ” ile q noktasından yayılan ses dalgasının m. mikrofonta ulaşması için gereken süre gösterilebilir. Bu süre şöyle ifade edilir:

$$\tau_m^q = \frac{r_m^q}{c} \quad (3.1)$$

Burada, c ses hızıdır ve 343 m/s olduğu bilinmektedir. R_m^q mesafeleri, q noktasının konumu bilindiği takdirde Pisagor teoremi uygulanarak basit bir geometri ile elde edilebilir. Yine bu bağıntıyı kullanarak m ve l mikrofonları arasındaki zaman gecikmesi aşağıdaki gibi de belirlenebilir.

$$\tau_{ml}^q = \tau_m^q - \tau_l^q = \frac{r_m^q - r_l^q}{c} \quad (3.2)$$

Buraya kadar olan açıklamalar ile mikrofon dizisi için temel bilgilere giriş yapılmıştır. Gelecek adımlarda da ses fiziği ile birlikte kullanılan ve/veya temel teşkil eden yöntem ve algoritmalara değinilecektir.

3.1 Ses Yayılımı

Sesin yayılımını incelemek ve ses kaynağı yer tespiti gibi uygulamalarda ses dalgasının modellenmesi gerekmektedir. Pratik uygulamalar için makul olabilecek ve geçerliliği gösterilmiş bir model ile yola çıkmak temel teşkil edecektir. Bu bağlamda ses dalgasını modellerken göz önünde bulundurulması gereken nitelikler aşağıdaki gibi sıralanabilir [13], [14] ve [15].

- Ses kaynağının noktasal kabul edilmesi, modelin basit ancak kullanılabilir olmasını sağlar. Nitekim sesin insan kafasından çıkması ve karmaşık yayılım örüntüleri göstermesi gibi gerçekçi durumlar göz önüne alındığında model amacından uzaklaşmakta, pratikliğini yitirmektedir.
- Sesin yayılım vasatın homojen kabul edilmesi ses hızının ortamın tamamında sabit olarak kullanılabilmesinin önün açmaktadır. Bununla birlikte ortamdaki nem ve/veya gaz dağılımı homojen

olmadığında, ses dalgası kırılma ve sapma gibi etkilere maruz kalacaktır.

- Ses dalgasının yayıldığı vasatın kayıpsız olduğu kabul edilir ve bu sayede ses dalgasının kaynaktan çıktıktan sonra ve/veya yayılım sırasında enerji kaybına uğramadığı kabul edilmiş olur.
- Son olarak, Doppler etkisinin ihmal edilebilir olduğu kabul edilerek de ses kaynağının yerinin değişmesi halinde yayılan sesin frekans içeriğinin değişmediği var sayılır ve bu da işaret işleme açısından bakıldığında, işlem yükü açısından oldukça önemli bir kazanç olarak görülmektedir.

Mikrofonlardan elde edilen ve dijitale çevrilerek bilgisayar ortamına aktarılan işaretlerin işlenebilmesi için matematiksel olarak ifade edilmesi gerekmektedir. Buradan yola çıkarak t anında m mikrofonundan alınan işaret,

$$x_m(t) = s(t) + n_m(t) \quad (3.3)$$

bağıntısı ile gösterilir. Burada, s(t) kaynaktan salınan ses dalgası, nm(t) ise m. mikrofon için gürültü bileşenidir. Gürültü terimi kaynaktan yayılan akustik dalga ile değil, ancak mikrofon ile ilişkilendirilmiştir. Dolayısı ile gürültü, her bir mikrofon için bağımsız stokastik süreçler olarak modellenmektedir [13].

(3.3) denkleminde ortam koşullarından kaynaklanan yankılanmalar mevcut değildir. [14] numaralı kaynağa göre, yankılanmalar gürültüden farklı olarak s(t) işaretime bağlıdır ve genellikle ihmal edilir. Bu halde yankılanmadan kaynaklanan ek işaretlerin de s(t) işaretinin içerisinde olduğu varsayılmakta ve basitlik de göz önünde bulundurularak modele dahil edilmemektedir. (3.2) denklemindeki notasyon kullanılır ise,

$$x_m(t, \mathbf{q}) = s(t - \tau_{am}^{\mathbf{q}}) + n_m(t) \quad (3.4)$$

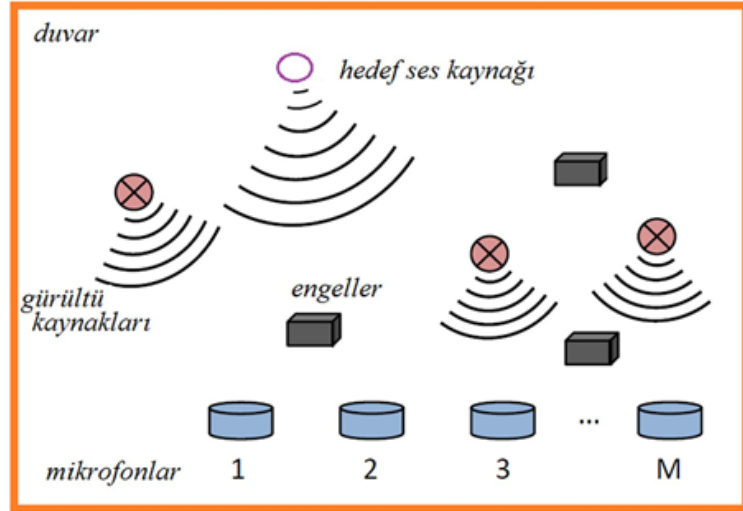
denklemi elde edilir. Burada denklem zaman ile birlikte q mesafesine de bağlı olan bir fonksiyon olarak verilmiştir. Farklı olarak, mikrofon dizisindeki mikrofonlardan biri referans kabul edilmiş ve a ile temsil edilmiştir. Çoğunlukla referans mikrofon,

a, ses kaynağı q'dan en uzaktaki mikrofon olarak seçilir. Nitekim, $\tau_a^q - \tau_m^q$ hiçbir zaman negatif olamayacaktır [16].

3.2 Demet Oluşturma

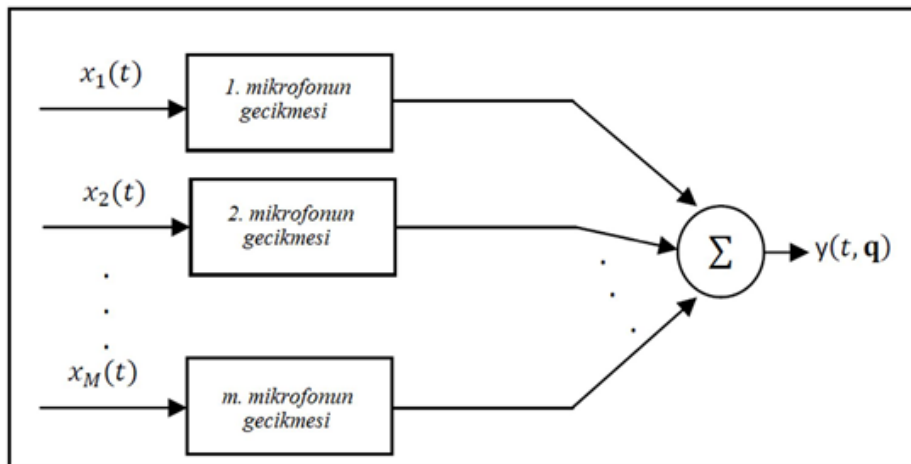
Ses kaynağının yerini bulma ve/veya konuşma iyileştirme gibi uygulamalar söz konusu ise, çoğunlukla kullanılacak yöntem demet oluşturma ve/veya türevleri olan algoritmalarıdır. Demet oluşturma teknikleri temelde mikrofon dizisinin bir hedef noktasına odaklanması ve dolayısı ile hedeflenen bu kaynak noktasından gelen işaretleri almasıdır. Hedef dışındaki tüm uzaydan gelen işaretler zayıflatmaya ve/veya sıfırlanmaya tabi tutularak girişim ve bozulmalar engellenir. Bu yöntem uzaysal filtreleme olarak da adlandırılır ve sonuçta elde edilen işaret demet oluşturma işleminin yönlendirilmiş cevabı olarak nitelendirilir [13].

Şekil 3.2'de örnek bir kayıt ortamında hedef ses kaynağı, engeller, gürültü kaynakları ve mikrofon dizisi görülmektedir. Bu ortam oldukça gerçekçi bir şekilde betimlenmiştir. Ses kaynağından yayılan akustik dalgalar duvarlardan ve yayılıma engel nesnelere yankılanacak, çeşitli gürültü kaynaklarından çıkan seslerden dolayı girişime maruz kalacak ve mikrofon dizisine oldukça bozulmuş ve/veya karışmış bir halde ulaşacaktır. Tüm bu olumsuz koşulların varlığına karşın, demet oluşturma yöntemlerinin kullanımı ile hedef ses kaynağının yeri belirlenebilmekte ve/veya konuşma üzerinde işlem yapılabilir [15]. Elbette ki burada hangi demet oluşturma tekniğinin kullanılacağı da önem kazanmaktadır, nitekim bu çalışmada bu tekniklerden başarı oranı en yüksek olan yöntemlerden biri olan faz dönüşümlü yönlendirilmiş cevap gücü yöntemi üzerinde durulacaktır.



Şekil 3.2 : Gürültü Kaynakları ve Mikrofon Dizileri

Demet oluşturma, kısaca özetlenirse uzaydaki bir nokta için yönlendirilmiş bir cevap bulma işlemidir [13]. Bu içerikte pek çok algoritma ortaya konmuştur. Bu algoritmalar arasında en basit ancak başarılı olan yöntem geciktir ve topla tekniğidir. Adından da anlaşıldığı üzere, bu teknikte her bir mikrofondan alınan işarete mesafe farklılıklarını telafi edecek gerekli gecikmeler uygulanır ve böylece tüm işaretler zamansal olarak üst üste getirilmiş olur. Üst üste gelen ses sinyalleri toplandığında ve/veya ortalaması alındığında ise her bir orijinal ses işaretinden daha zengin ve daha temiz bir işaret elde edilmiş olur. Geciktir ve topla algoritması temsili olarak Şekil 3.3'te gösterilmiştir. Burada $y(t, q)$ ile gösterilen yönlendirilmiş cevap işaretidir ve hem mesafeye hem de zamana bağlı olan bir fonksiyon olarak ifade edilmiştir.



Şekil 3.3 : Geciktir ve Topla Tekniği

Şekil 3.3'te şematik olarak gösterilen işlemi uygulamada kullanılabilir hale getirmek için matematiksel olarak aşağıda verilen (3.5) denkleminde olduğu gibi ifade etmek gerekmektedir.

$$y(t, \mathbf{q}) = \sum_{m=1}^M x_m(t - \tau_{am}^{\mathbf{q}}) \quad (3.5)$$

Bu ifadenin Fourier dönüşümünü alarak frekans içerikleri cinsinden de işlem yapmak mümkündür. (3.5) denkleminin frekans domeninde yazılışı aşağıda verilen (3.6) denklemi ile gösterilmektedir.

$$Y(\omega, \mathbf{q}) = \sum_{m=1}^M X_m(\omega) e^{-j\omega\tau_{am}^{\mathbf{q}}} \quad (3.6)$$

burada “ ω ” terimi frekans parametresini göstermektedir. Zaman gecikmesine ilişkin terim ise bu ifadede karmaşık üstel olarak yer almaktadır.

(3.5) denkleminde $y(t, \mathbf{q})$ ifadesi, (3.6) denkleminde de $Y(\omega, \mathbf{q})$ ifadesi konumu işaret eden \mathbf{q} terimine hassas bir şekilde bağlıdır. Eğer her bir mikrofondaki gürültüler, çoğunlukla kabul edildiği gibi birbirlerinden bağımsız ise ve \mathbf{q} konumu ses kaynağının yerini doğru bir şekilde belirtiyor ise geciktir ve topla algoritması sonucunda gürültüler filtrelenmiş ve konuşma işareti güçlenmiş ve net bir şekilde elde edilmiş olur. Ancak, \mathbf{q} konumu ses kaynağının yerini doğru göstermiyorsa tersi bir etki meydana gelecek, konuşma işareti zayıflamış ve gürültülü bir şekilde elde edilmiş olacaktır. Sonuç olarak demet oluşturma yöntemi ile oluşturulan yönlendirilmiş cevap sinyali yalnızca konum doğru olarak bulunmuş ve/veya biliniyor ise işlevini yerine getirecektir.

Geciktir ve topla algoritmasının filtreleme işlevi de eklenebilmektedir. Zaman domeninde konvolüsyon çarpımı olarak ifade edilen filtreleme işlemi, frekans domeninde basit çarpma olarak gerçekleştirilebildiğinden dolayı genellikle filtreleme frekans domeninde gerçekleştirilir ve (3.7) denklemi ile gösterilir.

$$Y(\omega, \mathbf{q}) = \sum_{m=1}^M W_m(\omega) X_m(\omega) e^{-j\omega\tau_{am}^{\mathbf{q}}} \quad (3.7)$$

burada $W_m(\omega)$ terimi filtreyi temsil etmektedir. Uygun ve yeterli bir filtre seçimi ortam koşulları ve ses kaynağının tipi gibi etmenlere göre belirlenir.

3.3 Yer Bulma

Ses kaynağı yer bulma algoritmalarında, bulunan ya da belirlenen konumun doğru konum olup olmadığı, algoritmanın performansını büyük ölçüde etkiler. Zira, elde edilen işaretin enerjisi yalnızca konum doğru olarak bilindiğinde maksimum olacaktır. Demet oluşturulan konum için çıkış işaretinin enerji karakteristiğinden yararlanma da bu kapsamda geliştirilen bir yöntemdir. Çıkıştaki işaretin enerji karakteristiğinden yararlanmak için de belirlenen ya da bulunan konum için yönlendirilmiş cevap gücü hesaplanır. Belirli bir \mathbf{q} konumu için yönlendirilmiş cevap gücü (3.8) denklemi ile elde edilir.

$$P(\mathbf{q}) = \int_{-\infty}^{\infty} |y(t, \mathbf{q})|^2 dt \quad (3.8)$$

Yönlendirilmiş cevap gücü, frekans domeninde de tanımlıdır. Zira, Parseval teoremine göre tüm t zamanı boyunca toplanan dalganın içerdiği toplam enerji, tüm ω frekans bileşenleri üzerinden toplanan Fourier dönüşümü dalgasının sahip olduğu toplam enerjiye eşittir. Bu durumda frekans domeninde tanımlanan yönlendirilmiş cevap gücü (3.9) denklemi ile hesaplanır.

$$P(\mathbf{q}) = \int_{-\infty}^{\infty} |Y(\omega, \mathbf{q})|^2 d\omega \quad (3.9)$$

Eğer $Y(\omega, \mathbf{q})$ ifadesinin yerine (3.7) denklemdeki açılımı konursa, filtre eklenmiş geciktir ve topla algoritmasının yönlendirilmiş cevap gücünü elde etmiş olunur, ve bu da (3.10) denkleminde gösterilmiştir.

$$P(\mathbf{q}) = \int_{-\infty}^{\infty} \left| \sum_{m=0}^{M-1} W_m(\omega) X_m(\omega) e^{-j\omega\tau_{am}^{\mathbf{q}}} \right|^2 d\omega \quad (3.10)$$

burada, $W_m(\omega)$ terimi, (3.7) denkleminde olduğu gibi filtreyi temsil etmektedir.

Bu çıkarımın ardından, uzayda bir q konumu için akustik enerjinin ölçülmesi mümkün hale gelmektedir. Bu ölçüm de ses kaynağının yerini bulmada bir algoritmanın oluşturulmasına imkan vermektedir. Nitekim, ses kaynağı ve/veya konuşmacının bulunduğu konum diğer tüm konumlara göre daha yüksek akustik enerjiye sahip olacaktır. Dolayısı ile, ses kaynağının bulunabileceği konumlar arasından en yüksek yönlendirilmiş cevap gücüne sahip olan konum potansiyel olarak aranılan konuma işaret edecektir. Matematiksel olarak ses kaynağı konumunun tahmin ifadesi, \hat{q} , aşağıdaki (3.11) denkleminde gösterildiği gibi kullanılacaktır.

$$\hat{q} = \underset{q \in Q}{\operatorname{argmax}} P(q) \quad (3.11)$$

burada Q ile gösterilen ise, kullanıcı tarafından belirlenecek olan, algoritmadan taramasının istendiği koordinatlardan oluşan kümeyi temsil etmektedir. Bu küme çoğunlukla, elemanları geometrik bir şekil oluşturacak şekilde seçilmektedir.

Eğer (3.10) denklemindeki $W_m(\omega)$ filtresi, faz dönüşümü işlevini yapan bir filtre olarak seçilir ise, elde edilen sonuç faz dönüşümlü yönlendirilmiş cevap gücü olacaktır. Bu yaklaşım [16] numaralı referansta öne sürülmüş ve gürültülü ve/veya yankılı ortamlarda ses kaynağının yerini bulmak için halen bilinen en iyi yöntemlerden biri olma özelliğini elinde bulundurmaktadır. Faz dönüşüm filtresi (3.12) denklemi ile hesaplanabilir.

$$W_m(\omega) = \frac{1}{|X_m(\omega)|} \quad (3.12)$$

Bu dönüşüm (3.10) denkleminde yerine konursa, frekans domeninde faz dönüşümlü yönlendirilmiş cevap gücü ifadesi elde edilmiş olur ve bu ifade (3.13) denklemi ile aşağıda verilmiştir.

$$P(q) = \int_{-\infty}^{\infty} \left| \sum_{m=1}^M \frac{X_m(\omega)}{|X_m(\omega)|} e^{-j\omega\tau_{am}^q} \right|^2 d\omega. \quad (3.13)$$

Faz dönüşümü filtresi aslen ses işaretindeki genlik bilgisini ortadan kaldırmakta ve geride yalnızca faz bilgisini bırakmaktadır. Bir başka ifadeyle, faz dönüşüm filtresi, ses işaretine ait frekans bileşenlerinin tümünü eşit olarak ağırlıklandırmaktadır. Bu sayede ise, yönlendirilmiş cevap gücü hesaplanırken yankılanma ve gürültülerin sinyale katkıları elenmiş olmaktadır. Bununla birlikte, ses işaretinin mikrofon dizisindeki elemanlara gelişinde her birinde farklı zaman gecikmesinin bulunması, Fourier dönüşümü düşünüldüğünde bu zaman gecikmesine ait bilginin yalnızca faz bileşeninde saklanıyor olduğu göz önüne alınmaktadır. Nitekim, genlik bileşeninin gürültü bilgisini de taşıdığı ve bu nedenle kullanılmasının yanıltıcı sonuçlara yol açabileceği de ihmal edilmemelidir.

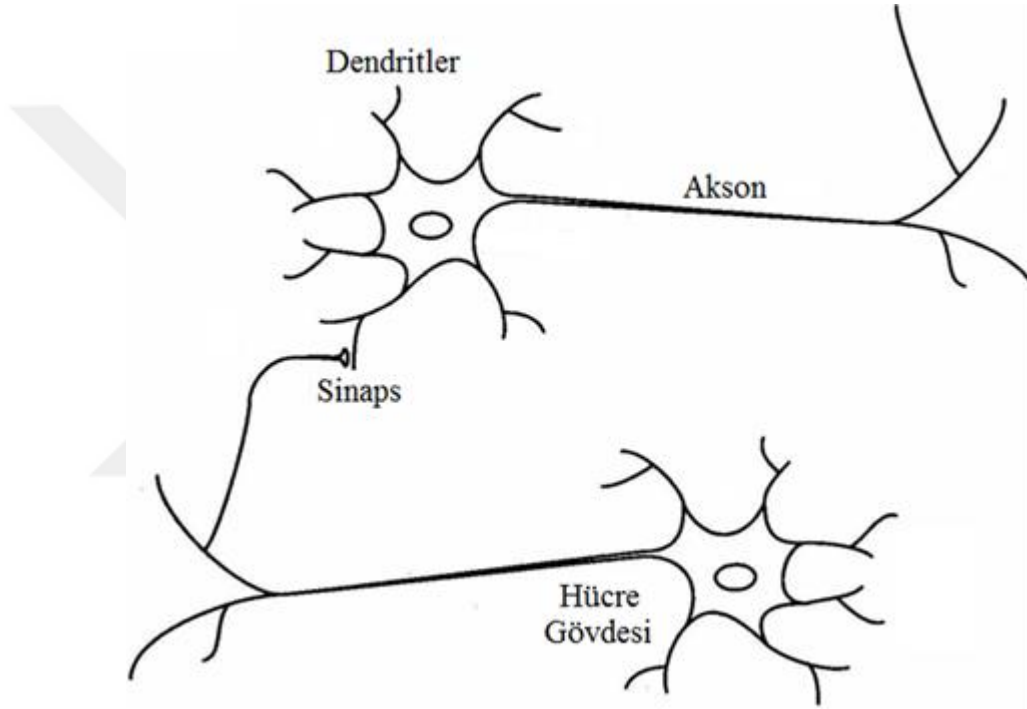
Faz dönüşümlü yönlendirilmiş cevap gücü yönteminin ses kaynağının yerini bulma konusunda yüksek başarımlı yeteneğinin yanında, hesapsal olarak yoğunluk içermesi gibi bir dezavantajı mevcuttur. Çoğu uygulamada, bu hesapsal yoğunluk ile konum tespitinde hassasiyet arasında bir pazarlık yapılmakta ve orta yol çözümlerine gidilmektedir.

Hesapsal yoğunluğa getirilen ve faz dönüşümlü yönlendirilmiş cevap gücü algoritmasının aslında özünde barındırdığı bir çözüm olarak genelleştirilmiş çapraz bağlantı yaklaşımına yer verilmesidir. Bu yaklaşım ile faz dönüşümlü yönlendirilmiş cevap gücü, faz dönüşümlü genelleştirilmiş çapraz bağlantıların toplamı olarak ifade edilebilmekte ve bu sayede işlemsel hesap yükü büyük ölçüde hafifletilmekte ve böylece bu dezavantaj elimine edilmektedir.



4. İĞNECİKLİ SINIR AĞLARI

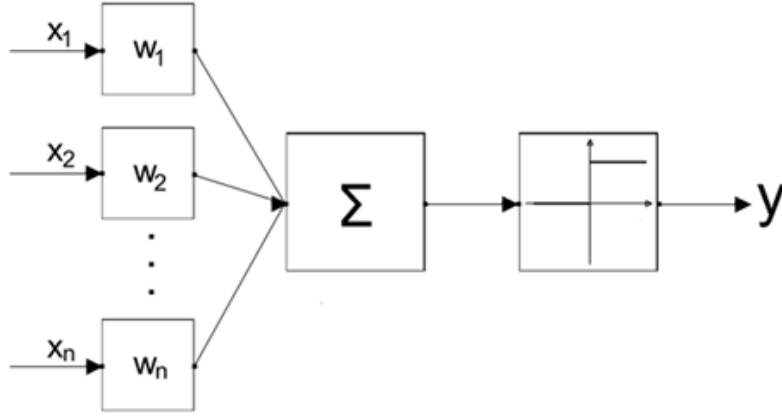
Yapay sinir ağı, canlılarda merkezi sinir sistemini oluşturan, birbirine bağlı nöronlardan oluşan oldukça yoğun sinir ağlarında yer alan hesaplamaları taklit ederek yapay bir biçimde gerçekleştirmeye çalışmaktadır. Biyolojik sinir hücresinin temsili bir çizimi Şekil 4.1’de görülmektedir.



Şekil 4.1 : Biyolojik sinir hücresi çizimi [19]

McCulloch ve Pitts, 1943’te yaptıkları çalışmada basitleştirilmiş ve ikili düzende işleyen sinir hücresi modeline dayalı bir sinir ağı önermişlerdir [17]. Burada bir sinir hücresi, “aktif” veya “aktif değil” şeklinde durumları olan basit bir eşik işlevine göre hareket etmektedir. Her bir sinirsel hesaplama adımında, ilgili hücreye bağlı diğer hücrelerin durumlarının ağırlıklı toplamına bağlı olarak sinir hücresinin

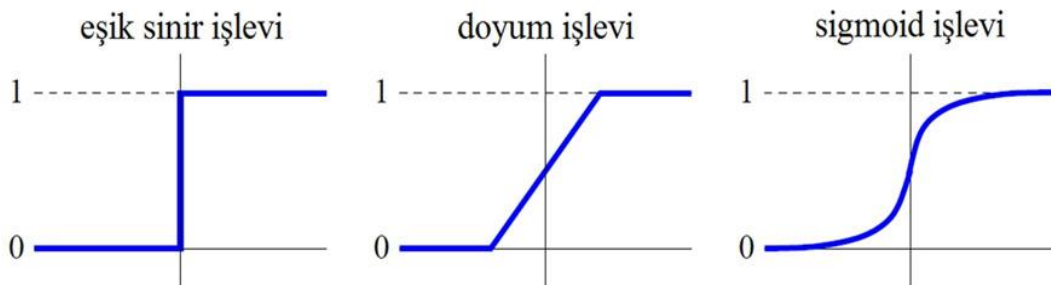
durumu belirlenmekte ve güncellenmektedir. 1943 yılında McCulloch ve Pitts tarafından önerilen [17], tek bir sinir hücresine ait matematiksel model Şekil 4.2’de görülmektedir.



Şekil 4.2 : İlk yapay sinir modeli [17]

Takip eden çalışmalarda, girdiler ve çıktılar reel değerli olarak kullanılmış ve doğrusal olmayan eşik işlevi yerini doğrusal giriş-çıkış haritalama ve/veya yine doğrusal olmayan ancak türevi alınabilen bir sigmoid işlevine bırakmıştır. Şekil 4.3’te bu dönem modellerinde kullanılmaya başlanan etkinleştirme işlevleri görülmektedir.

Hem belirli bir görevi yerine getirilmesi için (denetimli öğrenme), hem de farklı özneliklerin ağ tarafından öğrenilmesi için (denetimsiz öğrenme) çok sayıda algoritma geliştirilmiştir. Denetimli öğrenme algoritmaları sinir ağı davranışını belirli bir hedef işleve uydurmayı amaçlar ve bu kategoride en tanınmış yöntem hata geri yayılımı algoritmasıdır [18].



Şekil 4.3 : İlk kuşak sinir modellerinde kullanılan etkinleştirme işlevleri [30]

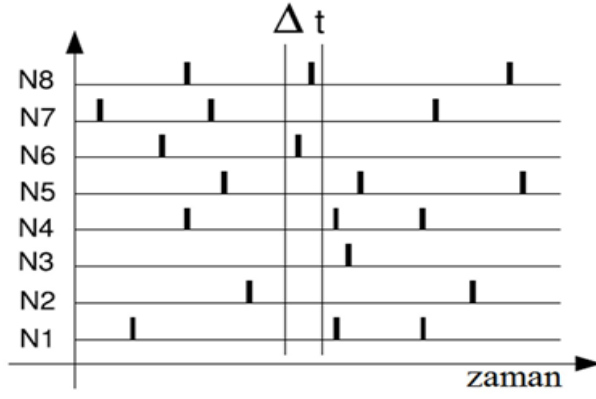
Denetimsiz öğrenme konusunda ise çalışmalar Donald Hebb'in doğal öğrenimin kuralları üzerine olan 1949 çalışmasına, [19], kadar geçmişe uzanmaktadır. Doğal öğrenimden esinlenen bu tür denetimsiz öğrenme kuralları, Hopfield ağında [20] da olduğu gibi, çoğunlukla Hebbçi kurallar olarak adlandırılır.

Genel olarak yapay sinir ağları bir mühendislik aracı olarak örüntü tanıma, kontrol, biyoinformatik ve robotik gibi çeşitli alanlarda oldukça başarılı bir şekilde halen kullanılmaktadır [30]. Ancak gelenekselleşmiş yapay sinir ağı türlerinin, temel olarak büyük ölçekte veri işleyememe ve/veya değişen ortam koşullarına çabuk uyum sağlayamama gibi kendilerine has kısıtları bulunmaktadır. Bu kısıtlar, biyolojik sinir ağlarındaki işlem kapasitesi ile kıyaslandığında oldukça bağlayıcı bir durum ortaya koymaktadır [31].

Aslına uygun - yapay olarak modellenen biyolojik sinir hücreleri temel olarak iğnecik şeklinde ani darbeler oluştururlar ve çoğu hücre birbirleriyle bu iğnecikler üzerinden haberleşir. Aksiyon potansiyeli adı verilen bu ani darbeler hücre gövdesinde oluşur ve akson boyunca iletilerek diğer sinir hücrelerinin dendritlerine ulaşır. İki sinir hücresi arasındaki bağlantı sinaps denilen bölgelerde gerçekleşir ve sinaps öncesi ve sinaps sonrası potansiyeller kimyasal olarak hücreden hücreye aktarılır [31].

Biyolojik sinir sistemlerinde bilginin iğnecik zamanlaması ile kodlandığı düşünülmektedir [21]. Kod çözüm için de bu doğrultuda çeşitli öneriler ve teknikler mevcuttur. Bunlardan ilki kodun bir nöron topluluğu tarafından ateşlenen iğneciklerin toplamı olarak ele alınmasıdır. Bir diğer teknik ise ikili kodlamaya benzeyen ve çıkışta görülen iğnecikleri bir arada ikili düzende bir sayı olarak ele alır [31].

Zamanlamaya bağlı kodlama ise, iğnecik zamanlamalarını hassas bir biçimde belirlenebilmesine bağlı olarak daha yüksek miktarda bilgi taşıyabilir. Son olarak, öncelik sırasına bağlı kodlama - kod çözme yöntemine göre, iğneciklerin salım sırasına bağlı olarak bilgi kodlanmaktadır [31].



Şekil 4.4 : Örnek bir iğnecik raster grafiği [25]

İğnecikli sinir ağlarında da bilgi iğnecik oluşması ve zamanlaması ile kodlanmaktadır. Zamansal kodlamanın görsel hale getirilebilmesi, dolayısıyla iğnecikli sinir ağı içerisinde gerçekleşen işlemlerin anlaşılabilmesi için iğnecik raster grafikleri kullanılmaktadır. Örnek bir raster grafiği Şekil 4.4'te gösterilmiştir.

Buradaki grafikte, “N” 1’den 8’e kadar olan sinir hücresi sayısını göstermekte ve bu hücrelerin hangi anda iğnecik oluşturdukları ortaya konmaktadır. Genellikle hesaplama sırasında küçük zaman aralıkları (Δt) içerisinde ateşlenmiş iğnecikler ele alınmaktadır.

İğnecikli sinir ağlarının yapay sinir ağlarına göre oldukça farklı bir işleyiş mekanizmasına sahip olmalarına karşın çoğu çalışma geleneksel öğrenme yöntemleri ve/veya bunların yeniden adapte edilmesi üzerinde durmaktadır.

5. SINIR HÜCRESİ MODELLERİ

Bir iğnecikli sinir ağı oluşturulurken kullanılacak sinir hücresi modelinin belirlenmesi ağ dinamiğini ve hesapsal işlemleri doğrudan etkilediği için oldukça önemlidir. Çok sayıda sinir hücresi modeli mevcut olmakla birlikte burada en bilinen ve en sık karşılaşılan modeller üzerinde durulmuştur.

5.1 Hodgkin-Huxley Modeli

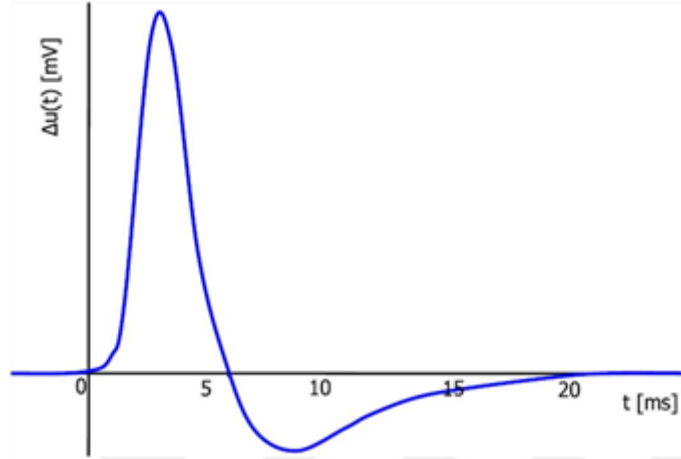
İğnecikli sinir hücresi modeli için fizyolojik temel oluşturan ve oluşturdukları model ile Nobel ödülü alan bu model Hodgkin – Huxley'nin çalışması ve isimleri ile anılmaktadır [22].

$$\begin{aligned} C \frac{du}{dt} &= -g_{Na} m^3 h (u - E_{Na}) - g_K n^4 (u - E_K) - g_L (u - E_L) + I(t) \\ \tau_n \frac{dn}{dt} &= -[n - n_0(u)] \\ \tau_m \frac{dm}{dt} &= -[m - m_0(u)] \\ \tau_h \frac{dh}{dt} &= -[h - h_0(u)] \end{aligned} \tag{5.1}$$

(5.1) numaralı denklemde matematiksel olarak ifade edilen bu modelde elektrokimyasal bilgi aktarımı, modellemede kolaylık sağlamak için kondansatör ve direnç içeren elektrik devrelerine benzetilmiştir. C kondansatör, R direnç ve g ile gösterilen terimler de hücre zarında bulunan ve elektriksel etkinlikten sorumlu olan sodyum (Na), potasyum (K) gibi farklı iyon kanallarının iletkenlikleridir. E

terimlerin bu kanallara karşılık gelen Nernst potansiyelleri ve m, h, n parametreleri de gerilime bağlı iyon kanallarının açılma-kapanma olasılık işlevleridir.

Hodgkin - Huxley modeli biyolojik deneyler sonucu elde edilen verilerden yola çıkılarak oluşturulmuştur. Biyolojik olarak ölçülebilen eylem potansiyellerinin içermesi gereken özellikleri içermekte ve Şekil 5.1'de gösterilen dalga şeklini üretmektedir.



Şekil 5.1 : Hodgkin-Huxley modeli eylem potansiyeli [22]

İğnecikli sinir ağları açısından bakıldığında Hodgkin - Huxley modeli karmaşıktır ve yoğun hesaplamalar gerektirmektedir [26]. Bu nedenle büyük ölçekli ağlar için kullanıma elverişli değildir.

5.2 Tümlev-Al-Ateşle Model Ailesi

Tümlev-al-ateşle türü sinir hücresi modelleri fikir olarak yirminci yüzyılın başlarına kadar gitmektedir [23]. Bu modeller ya da model ailesi içerisinde en geniş kullanım gören model sızdıran tümlev-al-ateşle modelidir [24].

Hesaplama açısından Hodgkin - Huxley modeline göre çok daha basit ve kullanışlıdır. Basit olmasının sebebi eylem potansiyeli dalga şeklini ihmal etmesinden ve eşik değer geçildiğinde iğnecik oluştuğunu varsaymasından kaynaklanmaktadır. Sızdıran tümlev-al-ateşle modeli basit ve kullanışlı olmasına karşın pek çok biyolojik özellikten yoksundur ve bu da biyolojik uygunluk açısından bir dezavantajdır.

$$C \frac{du}{dt} = -\frac{1}{R}(u(t) - u_{rest}) + I(t)$$

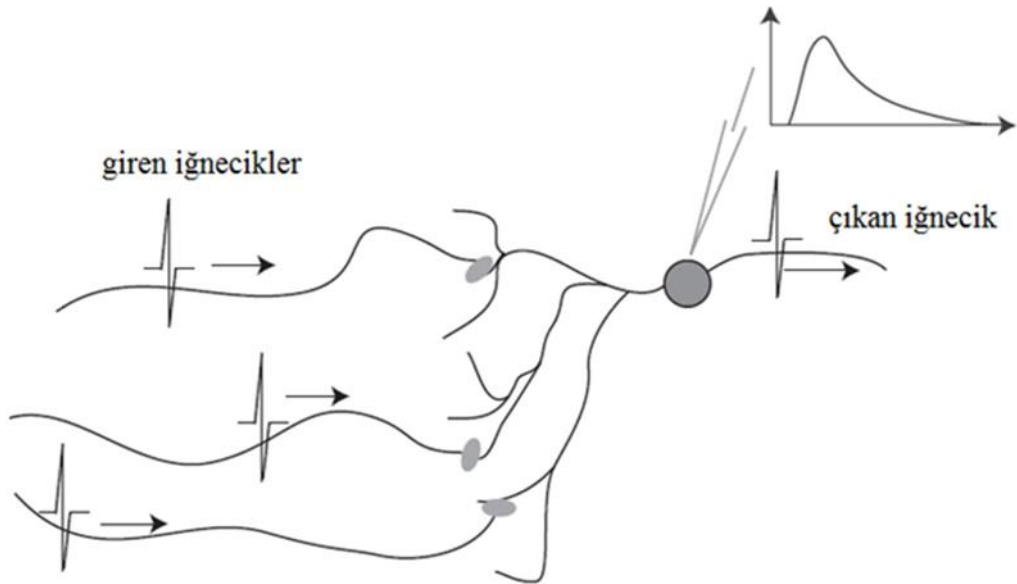
$$u(t^{(f)}) = \vartheta$$

$$u'(t^{(f)}) > 0$$
(5.2)

Hodgkin - Huxley modelinde olduğu gibi yine elektrik devre analogisi ile ifade edildiğinde, (5.2) numaralı denklemde ifade edilen 1. dereceden doğrusal diferansiyel denklem elde edilmektedir. Burada ve sonraki denklemlerde θ , eşik değerini temsil etmektedir. Eşik değerine gelindiğinde, $t(f)$ anında, ateşleme gerçekleşir ve potansiyel resetlenir.

5.3 İğnecik Cevap Modeli

Hodgkin - Huxley ve tümlev-al-ateşle türü sinir hücresi modelleri ve de bu modeller ile ilişkili benzer modellerde genellikle kuplajlı ve doğrusal olmayan diferansiyel denklem çözümleri yer almaktadır. İğnecik cevap modeli ise bunların aksine daha kolay anlaşılır ve daha kolay kullanılabilir bir modeldir [25 ve 26].



Şekil 5.2 : İğnecik Cevap Modeli [25]

İğnecik cevap modeli “ t ” zamanında “ u ” zar potansiyelini geçmiş üzerinden bir

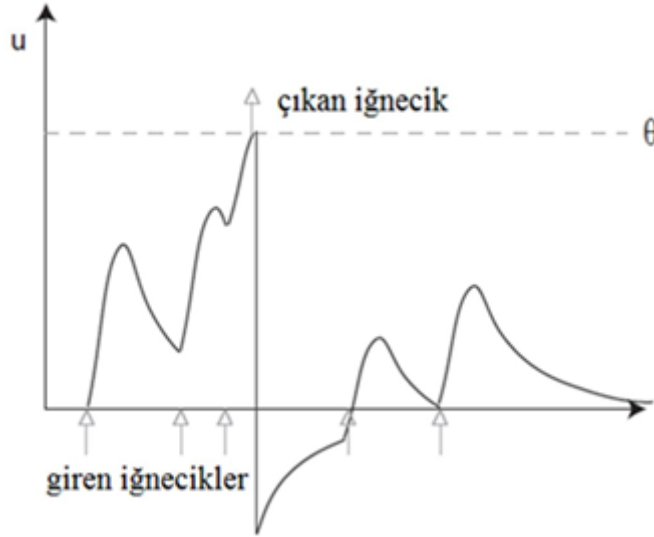
tümlev olarak ifade eder. Fenomenolojik bir sinir hücresi modelidir ve iğnecik salımı olup olmamasına bağlıdır ve temsili işleyişi Şekil 5.2’de görülmektedir. Matematiksel olarak ise, j. sinir hücresinin t anındaki durumu (5.3) numaralı denklemlerle ifade edilir.

$$u_j(t) = \sum_{t_j^{(f)} \in \mathcal{F}_j} \eta_j(t - t_j^{(f)}) + \sum_{i \in \Gamma_j} \sum_{t_i^{(f)} \in \mathcal{F}_i} w_{ij} \varepsilon_{ij}(t - t_i^{(f)}) + \underbrace{\int_0^\infty \kappa_j(r) I(t - r) dr}_{\text{harici giriş akımı}}$$

$$\eta_j(s) = -\vartheta \exp\left(-\frac{s}{\tau}\right) \mathcal{H}(s)$$

(5.3)

Burada η_j koçan işlevi iğnecik salımından sonra potansiyelin resetlenmesi, ε_{ij} koçan işlevi zar potansiyelinin sinaps öncesi iğneciklere cevabını ve son olarak κ_j koçan işlevi de zar potansiyelinin harici giriş akımına karşı cevabını belirtmektedir. Şekil 5.3’te ise belirtilen matematiksel ifadelerden elde edilen örnek bir çıktı görülmektedir.



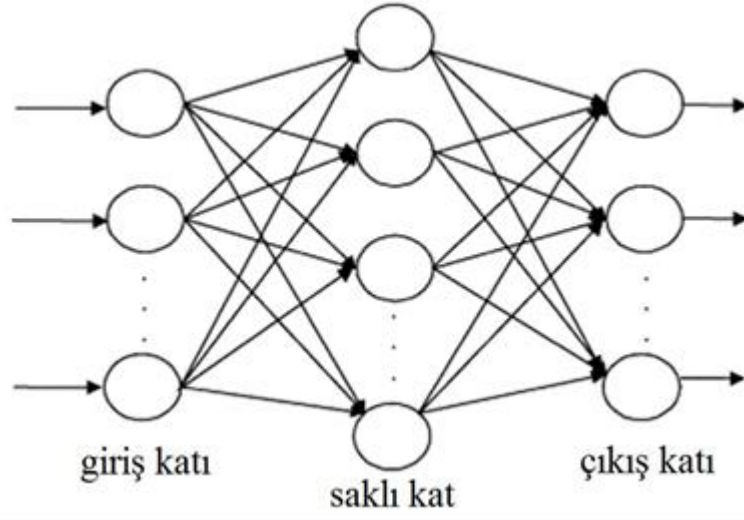
Şekil 5.3 : İğnecik Cevap Modeli Çıktısı [25]

6. AĞ YAPILARI

Geleneksel yapay sinir ağları için, çoğu ileri beslemeli olmak üzere pek çok ağ yapısı kullanılmıştır. İğnecikli sinir ağları için de başlangıçta geleneksel yapılara yer verilmiş ancak son on yılda bu ağa özgü özel yapılar üzerinde durulmaya başlanmıştır. Bu bölümde iğnecikli sinir ağlarında kullanılan ağ mimarileri kullanılacaktır.

6.1 İleri Beslemeli Ağlar

İleri beslemeli bir sinir ağında işlem birimleri olan sinir hücreleri genellikle katmanlar halinde bir yapıya sahiptirler. İşaretler, giriş katından çıkış katına doğru tek yönlü bağlantılarla iletilir. Sinir hücreleri bir kattan diğer bir kata bağlantı kurarlarken, aynı kat içerisinde birbirleri ile bağlantıları bulunmaz. Şekil 6.1’de ileri beslemeli ağ yapısı blok diyagram halinde gösterilmiştir.



Şekil 6.1 : İleri Beslemeli Ağ Yapısı [30]

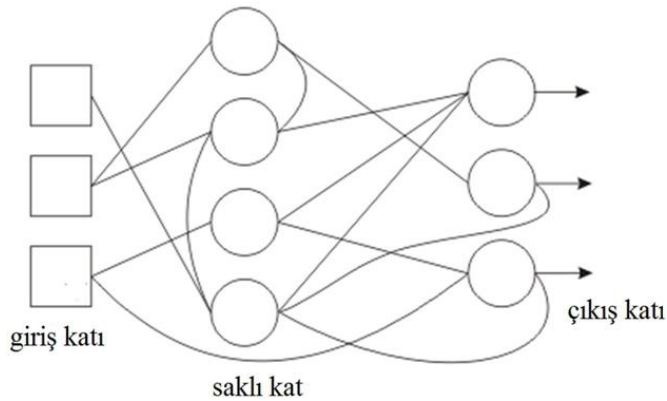
Tek katlı ileri beslemeli yapay sinir ağı en basit ağ yapısıdır. Bir giriş katı ve bir çıkış katı vardır. Bu tip bir ağda bilgi girişten çıkışa doğru ilerler dolayısı ile ağ ileri beslemelidir. Tek katlı olarak isimlendirilmesinin sebebi, giriş katının veri üzerinde hiçbir işlem yapmadan veriyi doğrudan çıkış katına iletmesidir.

Çok katlı ileri beslemeli yapay sinir ağı yapısı bir veya daha fazla saklı kat içermektedir. Saklı katların amacı giriş ve çıkış katları arasında gerekli bir takım işlemler yapmaktır. Giriş katı geniş olduğu zaman saklı katlar sayesinde yüksek dereceli istatistiksel veri elde edilebilir.

Çok katlı yapılarda (n)'inci katın çıkış sinyalleri (n+1)'inci katın giriş sinyalleri olarak kullanılır. m adet giriş düğümü, ilk saklı katta h1 adet sinir hücresi, ikinci saklı katta h2 adet sinir hücresi ve çıkış katında q adet sinir hücresi bulunan bir çok katlı ileri beslemeli ağ m-h1-h2-q ağı olarak adlandırılır. Eğer her katta bulunan sinir hücreleri bir sonraki katın tüm sinir hücreleri ile bağlantı yapıyor ise bu tip ağa tam bağlantılı ağ denir. Eğer bu sinaptik bağlantılardan bazıları eksikse ağ, kısmi bağlantılı ağ adını alır.

6.2 Geri Beslemeli Ağ Yapıları

Bir geri beslemeli sinir ağı, çıkış ve ara katlardaki çıkışların, giriş birimlerine veya önceki ara katlara geri beslendiği bir ağ yapısıdır. Böylece, girişler hem ileri yönde hem de geri yönde aktarılmış olur. Şekil 6.2'de bir örnek bir geri beslemeli ağ yapısı görülmektedir. Bu çeşit yapay sinir ağlarının dinamik hafızaları vardır ve bir andaki çıkış hem o andaki hem de önceki girişleri yansıtır. Bundan dolayı, özellikle önceden tahmin uygulamaları için uygundur. Bu ağlar çeşitli tipteki zaman-serilerinin tahmininde oldukça başarı sağlamışlardır.



Şekil 6.2 : Geri Beslemeli Ağ Yapısı [30]

Geri beslemeli sinir ađı yapısına benzer olarak bir de dngl ađ yapısı mevcuttur. Dngl yapay sinir ađlarının geri beslemeli ađlardan farkı, yapı ierisinde en az bir adet geri besleme evriminin olmasıdır. Yukarıda verilen sınıflandırmada, sinir hcreleri arasındaki bađlantıların simetrik veya asimetrik olması durumuna gre alt sınıflar ortaya ıkar. i nronundan j nronuna ynelik bir bađlantı varsa j 'den i 'ye ynelik bir bađlantı da vardır. Bu iki bađlantının ađırlıkları $w_{ij}=w_{ji}$ eđitse bađlantı simetriktir denir. Eđitsizlik durumunda, bađlantı asimetrik olur [30].

Farklı katlara ait sinir hcrelerinin bađlantısına, katlar arası (interlayer) bađlantı denir. Aynı kattaki sinir hcrelerinin bađlantısına, kat ii (intralayer) bađlantı, komđu olmayan katlardaki sinir hcrelerinin bađlantısında katlar st (supralayer) bađlantı denir. Bunlardan bařka, bir sinir hcrei kendisine de bađlantılı olabilir. Sıka kullanılan bir terim olan tam-bađlantılı olma, bir kata ait tm sinir hcrelerinin komđu kattaki tm sinir hcreleriyle bađlantılı olduđu durumu tanımlamaktadır [30].

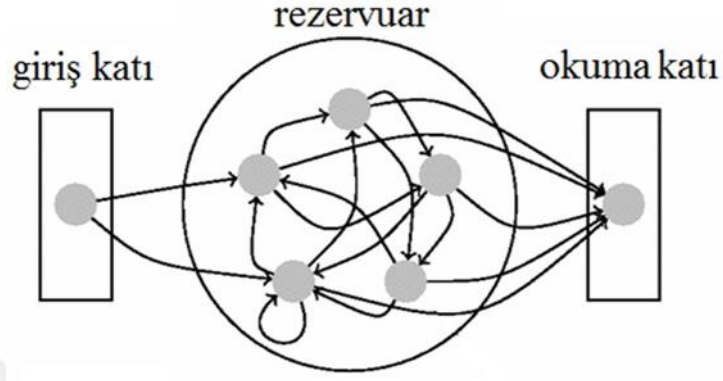
6.3 Rezervuar Hesaplama

İđnecikli sinir hcrelerinden kurulan ađlar, geleneksel yapay sinir ađlarından farklı mekanizmalara sahip olduklarından ve farklı davranıř dinamikleri gsterdiklerinden dolayı yukarıda anılan geleneksel ve/veya katı ađ biimleri ile alıřma zorunluluđu yoktur.

Biyolojik sistemlerde bulunan sinir ađları uzayda bořluklu ve dzensiz olarak bađlantılar oluřturmaktadır. İđnecik akıř eřitliliđi ve ortalama etkinliđin dřklđ de zamansal olarak dzensiz iletiřim kurduklarını gstermektedir. Bunun bir sonucu olarak da ađ topolojisinin sinirsel dinamiđe katkısının az olduđu ıkarılabilir, zira yalnızca etkin sinir hcreleri bilgi iřlemeye katkıda bulunmaktadır. Herhangi bir t anında, etkin sinir hcrelerinin belirlediđi alt topoloji, mevcut ađ mimarisinden olduka farklı ve bořluklu bir yapıda olabilir. Kısacası bir iđnecikli sinir ađ mimarisi dzenli olmak zorunda deđildir.

Daha da tesinde, iđnecikli sinirsel birimlerden oluřan bir ađ rasgele tanımlanabilir [27 ve 28]. Veya, sinir hcreleri ve/veya sinirsel iřlem birimlerinin verilen olasılık oranlarınca birbirlerine projeksiyon yaptıđı daha serbest belirlenmiř ađ mimarileri de mmkndr [29].

Buradan yola çıkarak, iğnecikli sinir hücrelerinin zamansal giriş-çıkış örüntülerini işeyiş mekanizmasına tam olarak uyan yeni bir ağ yapısı ve/veya yapı ailesi geliştirilmiştir. Bu yeni yapıya, öncülü olan, yankı durum ağları ve sıvı durum makinaları olarak adlandırılmış olan iki ayrı yapıyı bir araya getiren rezervuar hesaplama adı verilmiştir.



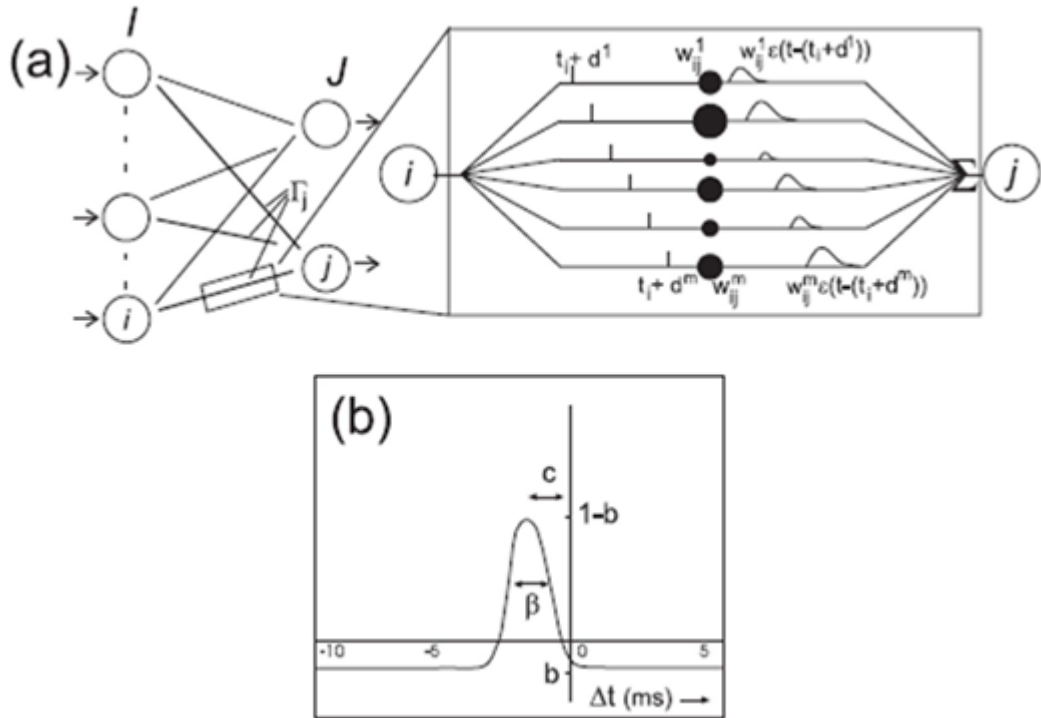
Şekil 6.3 : Rezervuar Hesaplama İçin Bir Ağ Yapısı [27]

Rezervuar hesaplamalı ağlarda temel olarak Şekil 6.3'te gösterilmiş olan ağ yapısına benzer ve şu özellikler ile belirlenir.

- Rezervuara doğru bağlantıları bulunan sinir hücrelerinden oluşan bir giriş katı olmalıdır.
- Rasgele ve boşluklu bağlantılar ile kurulmuş, sinir hücrelerinden oluşan döngülü bir ağ olmalıdır ki bu ağ rezervuarı (ya da sinir hücresi havuzunu) teşkil etmektedir.
- Son olarak da bir okuma katı bulunmalıdır. Bu katta okuma sinir hücreleri yer almakta olacak ve eğitilmiş bağlantılar ile rezervuara bağlı bulunacaktır.

7. ÖRNEK İŞLEM

Bohte'nin çalışmalarında verilen iğnecikli sinir ağı yapısı ve ilgili yöntemler [22], [23], [24] kullanılmıştır. Buna göre ağ yapısında geleneksel ileri beslemeli yapay sinir ağlarında olduğu gibi, katmanlar arası tam bağlantı kullanılmıştır. İlk katmanda sinir hücreleri girişe göre alıcı (reseptif) alan sinir hücrelerinden oluşmakta ve ikinci katmanda ise çıkış sinir hücreleri merkezden yayılan (radyal) taban işlevli sinir hücrelerinden oluşmaktadır.



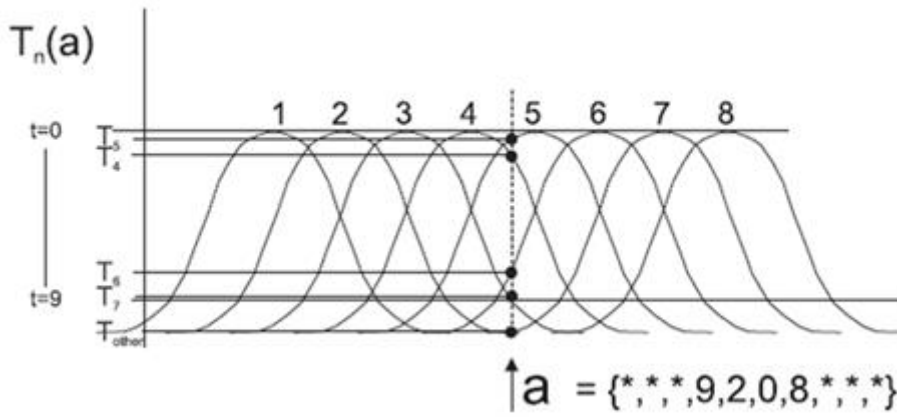
Şekil 7.1 : Ağ yapısı ve adaptasyon [23]

Sinirler arası çoklu bağlantı (sinaps) kullanılan bu yapı Şekil.7.1 (a)'da kesikli olarak ayrıntıda gösterilmiştir ve buradaki her alt bağlantıdaki gecikme farklıdır ve bu da toplam cevabı etkilemektedir. Sonuç olarak, bağlantı (sinaps)

sonrası potansiyel, alt bağlantı potansiyellerinin zamanda (temporal) toplanması ile elde edilmektedir [25].

Gözetimsiz sınıflandırma aşamasında temel olarak Bohte'nin çalışmalarında verilen [22], [23] yöntem ve yapı kullanılmıştır. Buna göre ağ yapısında geleneksel ileri beslemeli yapay sinir ağlarında olduğu gibi, katmanlar arası tam bağlantı kullanılmıştır. İlk katmanda sinir hücreleri girişe göre alıcı (reseptif) alan sinir hücrelerinden oluşmakta ve ikinci katmanda ise çıkış sinir hücreleri merkezden yayılan (radyal) taban işlevli sinir hücrelerinden oluşmaktadır.

Sinirler arası çoklu bağlantı (sinaps) kullanılan bu yapı Şekil.7.1 (a)'da kesikli olarak gösterilmiştir ve buradaki her alt bağlantıdaki gecikme farklıdır ve bu da toplam cevabı etkilemektedir. Sonuç olarak, bağlantı (sinaps) sonrası potansiyel, alt bağlantı potansiyellerinin zamanda (temporal) toplanması ile elde edilmektedir.



Şekil 7.2 : Alıcı alanlar [24]

Giriş katmanındaki sinir hücreleri, çakışan Gaussçu alıcı alanlar ile zamanda kodlama yapmaktadır. 8 sinir hücresi için (örneğimizde bu bir boyutun karşılığıdır) bir kodlama örneği Şekil.7.2'de gösterilmiştir. Burada "a" ile gösterilen giriş değişkeni örnek bir veri noktasına karşılık gelmektedir ve her bir sinir hücresi için ateşleme zaman gecikmesine çevrilmiştir. Buna göre en büyük değeri alan 5. sinir 0'a en yakın zamanda ateşlenirken en küçük değeri alan 7. sinir en son ateşlenmiş olacaktır.

Öğrenme, gözetimsiz olarak, Kazanan-Hepsini-Alır kuralına bağlı pekiştirmeli öğrenme yöntemi kullanılarak gerçekleştirilmiştir. Ağırlıkların güncellenmesi aşağıdaki bağıntı ile sağlanmaktadır.

$$\sum w_{ij} \sum L(\sum t_{ij}) \quad (7.1)$$

Buradaki öğrenme işlevi ise Şekil.7.1 (b)'deki Gauss eğrisi ile tanımlanmış ve (2) ile aşağıdaki gibi ifade edilmiştir. Burada, β öğrenme işlevinin genişliğini, c işlev tepesinin konumunu ve b de eğrinin sıfırın altında kalan kısmını belirler.

$$x_j(t) = \sum w_{ij} \sum(t - t_i) \quad (7.2)$$

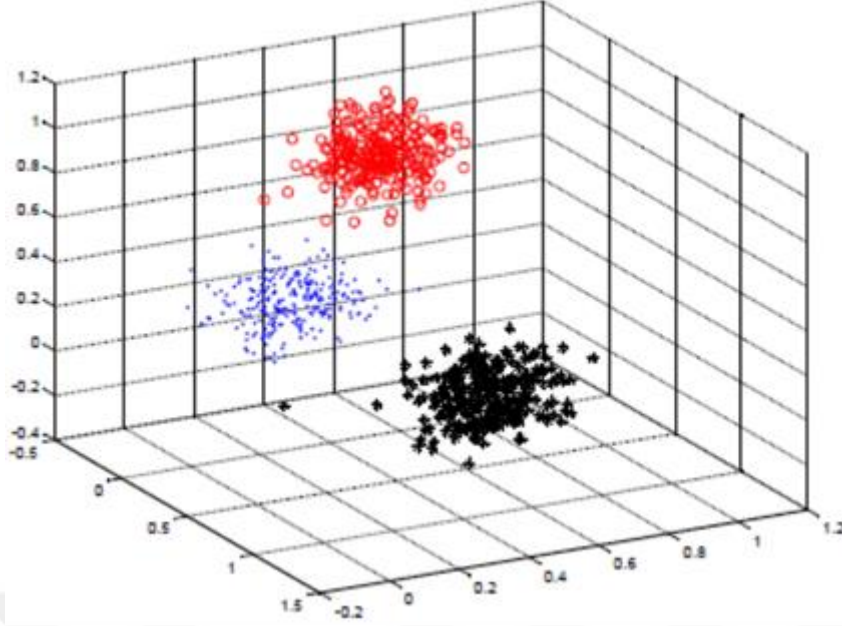
Alt bağlantılara ait dinamik, $x_j(t)$ dinamiği, ise yukarıda belirlenen ağırlıklar ile ağırlıklandırılmış iğnecik cevap işlevlerinin toplamıdır.

$$x_j(t) = \sum w_{ij} \sum(t - t_i) \quad (7.3)$$

İğnecik cevap işlevi bağıntısı ise aşağıdaki gibidir.

$$x_j(t) = \sum w_{ij} \sum(t - t_i) \quad (7.4)$$

Benzetim çalışması Matlab yazılımı kullanılarak gerçekleştirilmiştir. İki boyutlu veri kümesinin her bir boyutu için ağın ilk katmanında her birinin arasında 10 ms zaman aralığı bulunan sekizer adet sinir yerleştirildi. Bu katmandaki sinirlerin her birinde ise altışar adet alt bağlantı (alt-sinaps) bulunmaktadır. Çıkış katmanına ise üç adet merkezden yayılan (radyal) taban işlevli sinir yerleştirilmiştir.



Şekil 7.3 : Sınıflandırma Örneği

Bu parametrelerin farklı büyüklükteki kombinasyonları kullanılarak deneysel bir veri oluşturulmuştur. Veri kümesi, farklı varyanslara sahip gürültü seviyeleri eklenerek üç zorluk derecesinde yapay veri kümeleri oluşturulmuştur. Bu veri kümeleri için gürültü varyansları sırası ile 1, 1.3 ve 1.6 olarak kullanılmıştır. Bu veri kümelerinin ayrıştırılmış ve sınıflandırılmış halde çizimleri Şekil 4.3'te görülebilmektedir.

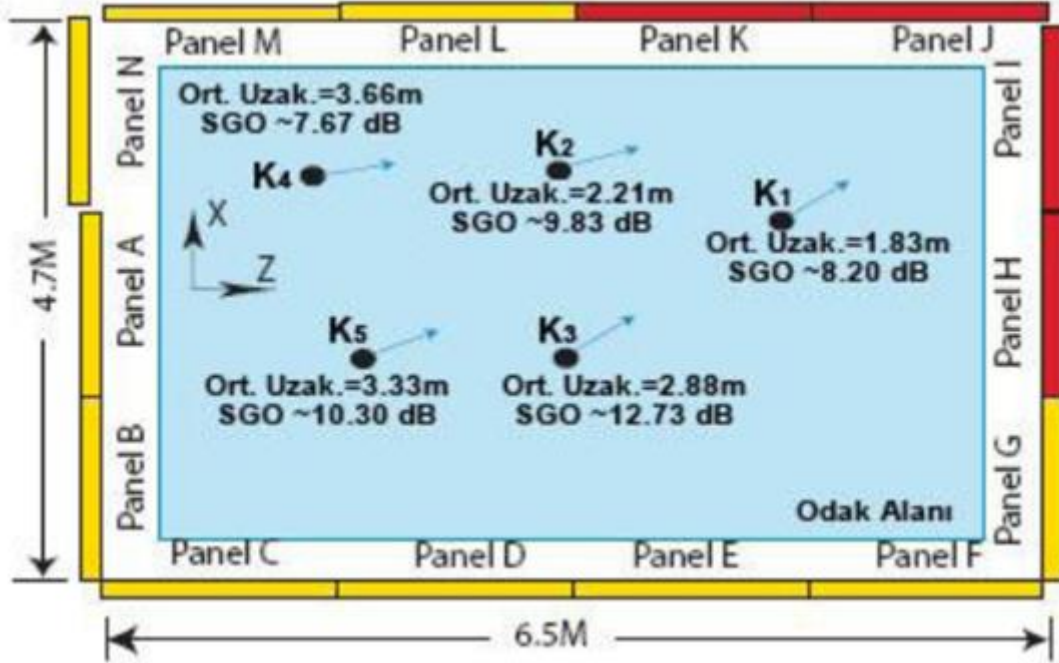
Veri kümesinin yaklaşık %50'lık bir kısmı eğitim kümesi, kalan %50'ı da test kümesi olarak ayrılmıştır. Öğrenme hızı 0.25 ve epok sayısı 10 olarak seçilmiştir. Her bir öğrenme adımında veri noktaları ağa verilerek (1) denklemine göre ağırlıklar güncellenmiştir.

Benzetim çalışmasında elde edilen sonuçlara göre veri kümesinde ayrılabilirlik azaldıkça modelin performansı da buna bağlı olarak düşmüştür. Üç ayrı veri kümesi için başarımlar sırasıyla; % 98,67 / % 95,73 / % 90,13 olarak hesaplanmıştır.

8. UYGULAMA VE BENZETİM ÇALIŞMALARI

Bu bölümde, telekonferans tipi bir uygulamadan elde edilen veriler kullanılarak, önerilen faz dönüşümlü yönlendirilmiş cevap gücü yöntemi ile yer bulma ve iğnecikli sinir ağı yöntemi ile de ses kaynaklarının sınıflandırılması test edilmiştir.

Kullanılan veri Brown Üniversitesi, LEMS Laboratuvarı (Laboratory for Engineering Man/Machine Systems), tarafından sağlanmıştır [16]. Bu veride 5 ayrı konuşmacı bir oda içerisinde konuşmalar yapmakta ve çevresel olarak yerleştirilmiş 24 adet mikrofondan oluşan mikروفon dizisi ile eş zamanlı olarak kayıt yapılmaktadır. Bu kayıtlarda konuşmalar doğal olarak birbirlerine karışmakta ve anlaşılabilir değildir. Telekonferans uygulamaları için örnek olabilecek bu kayıt ortamı Şekil 8.1’de görülmektedir. Burada konuşmacıların ortalama uzaklıkları ve desibel cinsinden çıkan seslerin güçleri de yaklaşık olarak verilmektedir.



Şekil 8.1 : Konuşmacı konumları

Çizelge 8.1 : Konuşmacı konumları

	X (metre)	Y (metre)	Z (metre)
1. Konuşmacı	0.7296	-0.4800	2.5052
2. Konuşmacı	-0.7944	-0.2900	4.0292
3. Konuşmacı	-0.2048	-0.4000	2.5052
4. Konuşmacı	-0.7944	-0.4400	3.1148
5. Konuşmacı	1.0344	-0.5500	3.7244

3 boyutlu oda ortamında, 5 konuşmacının ve 24 adet mikrofonun kartezyen koordinatlarda konumları, bir referans / odak noktasına göre sırası ile Tablo 8.1 ve Tablo 8.2'de gösterilmiştir.

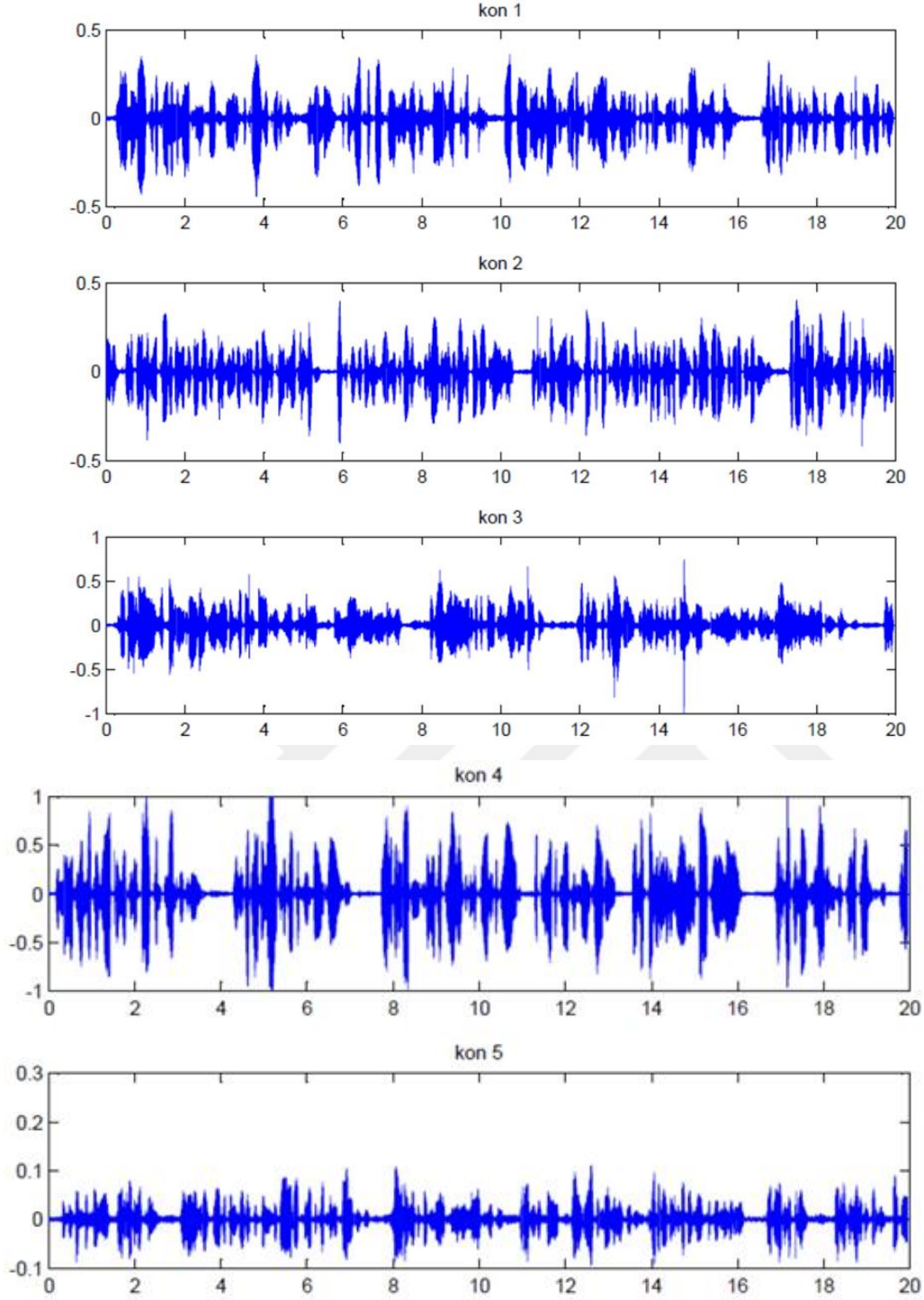
Çizelge 8.2 : Mikrofon konumları

Mikrofon	X (metre)	Y (metre)	Z (metre)
1	-0.3538	0.5322	6.0139
2	-0.3227	0.4428	6.0393
3	0.0272	0.5847	6.0094
4	0.0703	0.6763	5.9877
5	0.1321	0.8122	5.9566
6	0.4541	0.7401	5.9639
7	0.9688	0.4314	6.0451
8	1.0329	0.5533	6.0123
9	1.3674	0.5226	6.0263
10	1.5724	0.5445	6.0183
11	1.6151	0.6712	5.9869
12	1.9399	0.6913	5.9796

Çizelge 8.2 : (Devam)

13	2.0231	0.7032	5.5899
14	2.0278	0.6220	5.5896
15	2.0332	0.6213	5.1069
16	2.0221	0.7456	4.9109
17	2.0351	0.5077	4.9116
18	2.0335	0.6592	4.6104
19	2.0216	0.6856	4.2289
20	2.0346	0.4166	4.1408
21	2.0248	0.6269	3.8395
22	2.0268	0.5679	3.6933
23	2.0275	0.4774	3.6918
24	2.0166	0.5960	3.2143

Alınan ses kayıtları 20 kHz ile örneklenmiş ve kayıt yaklaşık 20 saniye kadar sürdürülmüştür. Her bir konuşmacıya ait ses kayıtları ayrılmış olarak Şekil 8.2’de verilmiştir. Buradaki ayırma işlemi önerilen yöntem ile gerçekleştirilmiştir.



Şekil 8.2 : Konuşma ses dalgaları

3. bölüm olan yöntem kısmında, burada kullandığımız yer bulma algoritması olan faz dönüşümlü yönlendirilmiş cevap gücü tanımlanmış ve formülize edilmişti. Bu kısımda yukarıda açıklanan veri kümesine ilk aşamada faz dönüşümlü yönlendirilmiş cevap gücü - yer bulma algoritması kullanılarak sınıflandırmada kullanılacak ikincil veri kümesi elde edilmektedir.

Veri kümesinden her 1024 örnekleme bir olmak üzere 512 örneklemlerle adımlarla ilerleyerek kısa dönemli ses verisi yer bulma algoritmasına uygulanmıştır. Bu durumda 20 kHz ile örneklenen veri kümesinden toplamda 780 adet kısa dönem ses kayıt verisi elde edilmiştir.

Ortamda 5 adet konuşmacı bulunmakta ve bu konuşmacılar, konuşmalarında farklı içerikleri farklı zamanlama ve farklı vurgulamalar ile gerçekleştirmektedirler. Bundan dolayı her bir kısa dönem için çalıştırılan yer bulma algoritması, her bir çıktısında öne çıkan, dolayısı ile farklı, konumları tespit etmektedir. Bu konumlar o an için sesi, vurgusu veya zamanlaması baskın olan konuşmacıya ait olan konumlar olarak değerlendirilmelidir.

Sonuç olarak mevcut örnekleme frekansı ve kayıt süresi ile seçilen kısa dönem periyotları, 780 adet kısa dönemli ses örnekleme cümlesine dönüştürülmüş, tüm bu kısa dönemli örneklemlerin her biri için faz dönüşümlü yönlendirilmiş cevap gücü - yer bulma algoritmasının çalıştırılması ile belirlenen konumlardan da 780 adet 3 eksenli kartezyen koordinat verisi elde edilmiştir.

Bu ikincil veride konuşmacıların kartezyen koordinatlarının gürültü, anlık yönelimler, karşılıklı etkileşim ve girişimler gibi nedenlerden ötürü dağınık ve düzensiz bir haldedir. Oysa ki demet oluşturarak en iyi sesi alabilmek için konum bilgisinin gerçeğe en yakın konumu göstermesi gerekmektedir. Bu nedenle bu ikincil veri üzerinde bu aşamada bir sınıflandırma yapılması ve konuşmacıların kesin yerlerinin belirlenmesi gerekmektedir.

Çizelge 8.3 : Tespit edilen konuşmacı konumları

	X (metre)	Y (metre)	Z (metre)
1. Konuşmacı	0.7664	-0.4901	2.5098
2. Konuşmacı	-0.8002	-0.2676	4.0084
3. Konuşmacı	-0.1826	-0.3809	2.4489
4. Konuşmacı	-0.71470	-0.4276	3.0903
5. Konuşmacı	1.0307	-0.5064	3.7854

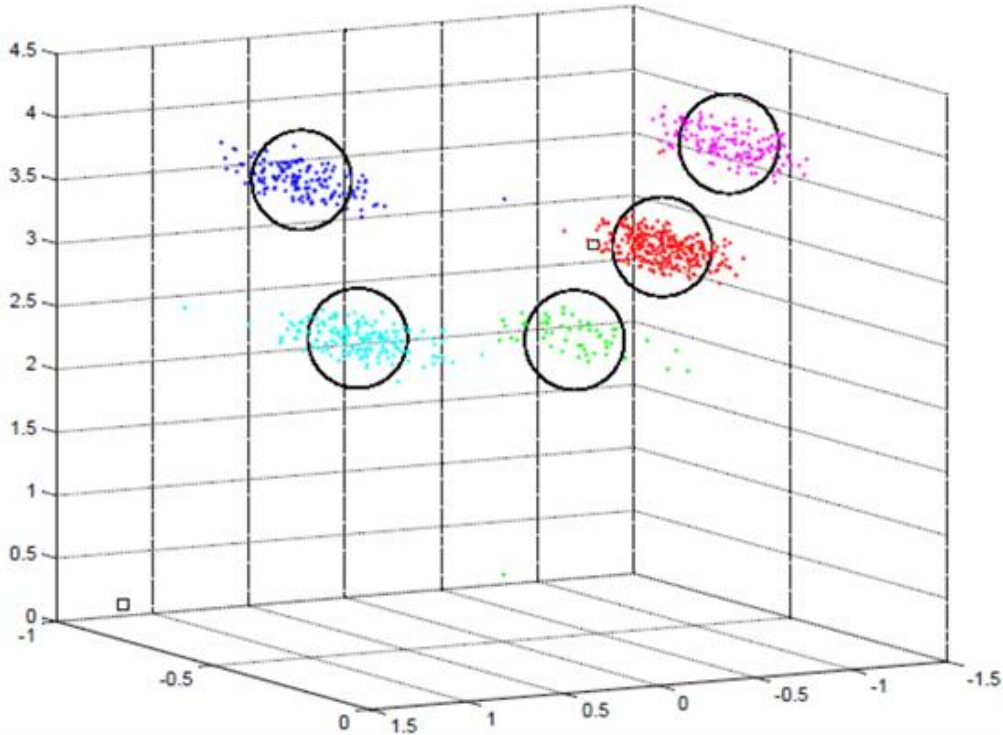
Çizelge 8.4 : Konum hataları (mutlak)

	X (metre)	Y (metre)	Z (metre)
1. Konuşmacı	0.0368	0.0101	0.0046
2. Konuşmacı	0.0058	0.0224	0.0208
3. Konuşmacı	0.0222	0.0191	0.0563
4. Konuşmacı	0.0797	0.0124	0.0245
5. Konuşmacı	0.0037	0.0436	0.0610

Çizelge 8.5 : x y ve z eksenlerinde maksimum hatalar

	X	Y	Z
Maksimum hata (cm)	7.97	4.36	6.1

Sınıflandırma aşamasında dört, beş ve altıncı bölümlerde anlatılan iğnecikli sinir ağı kullanılmıştır. Yedinci bölümde gösterilen örnek uygulamadaki yapı ve algoritma korunmuş, farklı olarak çıkış katmanında 5 konuşmacıyı temsilen 5 adet radyal tabanlı hücreye yer verilmiştir.



Şekil 8.3 : Sınıflandırma sonuç gösterimi

Veri kümesi, iğnecikli sinir ağı girişine uygulandıktan sonra çıkışta beş merkeze kümelenen konum bilgileri Şekil 8.3'te görüldüğü gibi elde edilmiştir. Bu kümelenen konum bilgilerinin ortalamaları alınarak nihai konum bilgilerine ulaşılmıştır ve bu bilgiler Tablo 8.3'te verilmiştir. Tespit edilen konum bilgileri gerçek konum değerlerine oldukça yakındır, zira Tablo 8.4'te de hatalar gösterilmiştir. Sonuç olarak x, y ve z eksenleri için en yüksek hatalar da Tablo 8.5'te görülmektedir.





9. SONUÇ

Bu çalışmada, birçok ayrı alanda uygulanabilirliği gösterilmiş olan, dizisel işaret işleme alanının akustik veri işleme üzerine çalışılan kolu olan mikrofon dizileri ile ilgilidir. Mikrofon dizileri ile, teorik olarak mikrofon sayısı kaynak sayısından büyük eşit olmak üzere, ses kaynaklarının kartezyen koordinatları oldukça yüksek bir hassasiyet ile bulunabilmektedir.

Ses kaynağı konumu belirlemede başarıyı en yüksek yöntemlerden biri olan faz dönüşümlü yönlendirilmiş cevap gücü algoritması üzerine çalışılmış ve uygulama ve benzetim bölümünde de bu algoritmaya yer verilmiştir. Gerçek zamanlı uygulamalar göz önünde bulundurularak algoritmaya 512 örneklem adımları ile 1024 örnekli bölümler halinde girdiler uygulanmıştır. 20 kHz'lik örnekleme frekansı düşünüldüğünde 1024 örnek yaklaşık olarak 0.05 saniyeye karşılık gelmektedir. Bu da, algoritmanın 25 milisaniyelik adımlar ile çalıştırıldığı anlamına gelmektedir. Bir telekonferans örneği düşünüldüğünde bu süre gerçek zamanlı kontrol için oldukça uygundur.

Ses kaynağı yer bulma algoritması çıktı olarak 3 boyutlu uzayda kartezyen koordinat bilgisi vermektedir. 25 milisaniyelik adımlarla alınan 50 milisaniyelik veri parçaları her periyotta farklı konum bilgilerini çıktı olarak vermektedir çünkü konuşmacıların konuşma zamanları, tonlamaları, ses yükseklikleri birbirinden farklıdır. Bunun yanı sıra ortamda gürültü ve yankı gibi bozucu etkenler de bulunmaktadır.

Mikrofon dizileri ile yapılan kayıtlarda kaynakların ayrı ayrı dinlenebilmesi için odaklama yapılması – dolayısı ile demet oluşturulması gerekmektedir ve bunun için de ses kaynağı konumunun kesin bir şekilde bulunabilmesi gerekmektedir. Bu amaçla faz dönüşümlü yönlendirilmiş cevap gücü algoritmasından yukarıda

belirtilen sıklıklarda alınan konum bilgilerinin tekrar değerlendirilerek ses kaynağı konumlarının tayin edilmesi gerekmektedir. Bu yeniden değerlendirme işlemi için iğnecikli sinir ağı ile sınıflandırma yapılması uygun görülmüştür. Bu şekilde gerçek zamanlı uygulamalara uygunluk gözetilerek sıralı öğrenme tekniğine yer verilen iğnecikli sinir ağı da aynı zamanda paralel olarak işleyerek yer bulma algoritmasının çıktıları olan 3 boyutlu konum bilgilerini girdi olarak almaktadır. İğnecikli sinir ağının çıktısı olan 5 öbek ve harici veri noktaları da belli periyotlarla değerlendirilerek bir sınıflandırma gerçekleştirilmektedir. Sınıflandırma sonucunda belirlenen öbeklerin ortalaması 3 boyutlu uzaysak koordinatlarda konuşmacının konumu olarak tayin edilmektedir.

Elde edilen sonuçlar, oldukça yüksek bir başarıma işaret etmektedir. Belirlenen konumlar, bilinen gerçek konuşmacı konumları ile oldukça uyumludur ve hata oranları birkaç santimetreyi aşmamaktadır. Bu çalışmanın, kurulması amaçlanan bir deney düzeneğinden elde edilecek gerçek veriler ile doğrulanması ve geliştirilmesi planlanmaktadır.

KAYNAKLAR

- [1] **P Stoica, R Moses.** *Spectral Analysis of Signals*, NJ: Prentice Hall, 2005
- [2] **Jha, RakeshMohan,** *Trends in Adaptive Array Processing*, 2012
- [3] **Viberg, Mats,** *Sensor Array Signal Processing: Two Decades Later*, 1995
- [4] **Visser H.,** *Array and Phased Array Antenna Basics*. New Jersey: Wiley, 2005
- [5] **McGowan I.,** *Microphone Arrays: A tutorial*, April 2001.
- [6] **Kushleyev A., Sound Localization.** University of Maryland, Institute for Systems Research, Summer 2004.
- [7] **Gregory E., Cole J.,** *Audio Localization*. Connexions Project, 2004.
- [8] **Kazuhiro N., Hiroshi O., Hiroaki K.,** *Auditory Fovea Based Speech Enhancement and its Application to Human-Robot Dialog System*. Kyoto University, Graduate School of Informatics.
- [9] **Basu S., Schwartz S., Pentland A.,** *Wearable Phased Arrays for Sound Localization and Enhancement*. MIT Media Laboratory, IEEE 2000.
- [10] **EECE 503,** *Audio Engineering. Course Notes.*
- [11] **Zeren A.,** *Müzik Fiziği*, Pan Yayıncılık, 2003
- [12] **Beranek L.,** *Noise Reduction Precared for a Special Summer Program at the Massachusetts Institute of Technology*. New York: McGraw-Hill, 1960
- [13] **Benesty J., Chen J., Huang Y.,** *Microphone Array Signal Processing*, Springer, 2008
- [14] **Johnson D. H., Dudgeon D. E.,** *Array Signal Processing: Concepts and Techniques*, Prentice Hall, 1993
- [15] **Brandstein W., Ward D.,** *Microphone Arrays: Sibnal Processing Techniques and Applications*, Springer, 2001
- [16] **DiBiase J. H.,** *A High-Accuracy, Low-Latency Technique for Talker Localization in Reverberant Environments Using Microphone Arrays*, Brown

Üniversitesi, 2000

- [17] **W.S. McCulloch and W. Pitts.** A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943
- [18] **D. E. Rumelhart, G. E. Hinton, and R. J. Williams.** Learning internal representations by back-propagating errors. *Nature*, 323:533–536, 1986.
- [19] **D.O. Hebb.** *The Organization of Behaviour.* Wiley, New York, 1949.
- [20] **J.J. Hopfield.** Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci.*, 79(8):2554–2558, 1982.
- [21] **R. Van Rullen and S. Thorpe.** Rate coding versus temporal order coding: what the retinal ganglion cells tell the visual cortex. *Neural Computation*, 13:1255–1283, 2001.
- [22] **A.L. Hodgkin and A.F. Huxley.** A quantitative description of ion currents and its applications to conduction and excitation in nerve membranes. *J. of Physiology*, 117:500–544, 1952
- [23] **L. Lapicque.** Recherches quantitatives sur l’excitation électrique des nerfs traitée comme une polarisation. *J. Physiol. Pathol. Gen.*, 9:620–635, 1907. cited by Abbott, L.F., in *Brain Res. Bull.* 50(5/6):303–304.
- [24] **D.I. Standage and T.P. Trappenberg.** Differences in the subthreshold dynamics of leaky integrate-and-fire and Hodgkin-Huxley neuron models. In *IJCNN’2005, Int. Joint Conf. On Neural Networks*, pages 396–399. IEEE-INNS, 2005.
- [25] **W.. Gerstner.** Time structure of the activity in neural network models. *Physical Review E*, 51:738–758, 1995.
- [26] **W.M. Kistler, W. Gerstner, and J.L. van Hemmen.** Reduction of Hodgkin-Huxley equations to a single-variable threshold model. *Neural Computation*, 9:1015–1045, 1997.
- [27] **W. Maass, T. Natschläger, and H. Markram.** Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, 14(11):2531–2560, 2002.
- [28] **H. Jaeger.** Tutorial on training recurrent neural networks, covering BPTT, RTRL, EKF and the “echo state network” approach. Technical Report TR-GMD-159, German National Research Center for Information Technology, 2002.
- [29] **W. Maass.** Networks of spiking neurons: The third generation of neural network models. *Neural Networks*, 10:1659–1671, 1997
- [30] **Haykin S.,** (1994) *Neural Networks*, Macmillan, New York.
- [31] **W. Gerstner and W. Kistler** ‘Spiking Neuron Models – Single Neurons, Populations, Plasticity’ Cambridge Univ. Press (2002)

EKLER

EK A: Kodlar





Kod A: Veri hazırlama

```
clear all; close all; clc
```

```
load mic_loc.mat
```

```
Y1 = wavread('5stalkers_loc3');  
Y2 = wavread('5stalkers_fs20kHz_24mics');
```

```
%s = Y2(4097:4096+2048,:);
```

```
s = Y2;
```

```
% % % % s=s(1:500,1:5);  
% % % % kkrec=[];  
% % % % for i=2:6  
% % % %  
% % % % j = i/2;  
% % % %  
% % % % kk = s( (j-1)*20+1 : j*20 ,:);  
% % % %  
% % % % kkrec=[kkrec; kk];  
% % % % end
```

```
%[finalpos,finalsrp,finalfe] = srplems(s, mic_loc, 20000, [-2 -1 0], [2 0  
6]);
```

```
posrec=[  
]; for  
j=2:780,
```

```
    i=j/2;
```

```
    pos = srplems(s(((i-1)*1024+1):i*1024,:), mic_loc, 20000, [-2 -1 0], [2  
0 6]);  
    posrec = [posrec;pos];
```

```
end
```

```
figur
```

```
e(1)
```

```
plot3(posrec(:,1),posrec(:,2),posrec(:,3),'x'),grid dlmwrite('posdat.txt',posrec);
```

Kod B: Faz dönüşümlü yönlendirilmiş cevap gücü algoritması

```
function [finalpos,finalsrp,finalfe]=srplems(s, mic_loc, fs, lsb, usb)
```

```
warning off all
```

```
if nargin < 5, usb=[2 0 6]; end
```

```
if nargin < 4, lsb=[-2 -1 0]; end
```

```
if nargin < 3, fs=20000; end
```

```
L = size(s,1); %%% determine frame-length
```

```
M = size(mic_loc,1); %%% number of microphones (also number of channels  
of the input X).
```

```
np=M*(M-1)/2; %%% number of independent pairs
```

```
%steplength = L/4; %%% non-overlapped length of a frame (75%  
overlap) dftsize = L; %%% dft size equals to frame size
```

```
temperatureC=24.0;
```

```
speedofsound=331.4*sqrt(1.0+(temperatureC/273));
```

```
magiconst=10*fs/speedofsound;
```

```
%% Determine the maximum end-fire length (in samples) of a microphone  
pairs:
```

```
mdist=pdist(mic_loc);
```

```
efmax=max(mdist);
```

```
efs=2*(fix(efmax*fs/speedofsound)); %%% End-fire is symmetric about the  
0th sample so multiplying by 2.
```

```
%efs=801;
```

```
hefs=round(efs/2); %%% half of the end-fire region
```

```
sf=fft(s,dftsize);
```

```
%% FFT of the original
```

```
signals csf=conj(sf);
```

```
%% Initialize yv to store the SRP-
```

```
yv=zeros(np,efs);  
PHAT samples
```

```
p=1;
```

```
for i=1:M-1
```

```
    su1mic=sf(:,i)*ones(1,M);
```

```
%% Create M copies of each signal's
```

```
    FFT prodall=su1mic.*csf;
```

```
%% Calculate the cross-power
```

```
    spectrum:
```

```
    = fft(x1).*conj(fft(x2))
```

```
    ss=prodall(:,i+1:M);
```

```
%% ss will be the cross-power
```

```
    spectra of microphone pairs (i,i+1), (i,i+2)...(i,M)
```

```
    ss=ss./(abs(ss)+eps);
```

```
%% PHAT weighting
```

```
ssiff=real(iff(ss,dftsize)); %%% Get the GCC-PHAT ssiff, which is the  
IFFT of the cross-power spectra
```



```

        %newssifft=[ssifft(end-hefs:end,:); ssifft(1:efs-hefs-1,:)]; %%% Only
select 'efs' samples (the beginning+end portions)
        newssifft=[ssifft(end-hefs+1:end,:); ssifft(1:efs-hefs,:)];
        newssiftr = newssifft';          %%% Transpose it
        yv(p:p+M-i-1,:)=newssiftr;      %%% Store in yv
        p=p+M-i;                          %%% Update the current index
        of yv clear su1mic prodall ss ssifft newssifft newssiftr
end

```

%%% Doing cubic-splines interpolation (factor of 10) on the GCC-PHAT:

```

xx=[1:1:efs];
x=[1:efs];
yintp=spline(x,yv,xx);
yintpt=yintp';

```

%%% Initialize to do

SRC:

```

efsintp=length(xx)/2;

```

```

row1=(0:np-

```

```

1]*2*efsintp)';

```

```

randpts=3000; %%% J0 in SRC. Depending on the size of the search volume,
choose an appropriate value here (Here, 3000 is for a  $V_{\text{search}} = 4\text{m} \times 1\text{m} \times 6\text{m}$ )

```

```

npoints=100; %%% Best N points. Again, choose an appropriate
number according to your problem.

```

%% Doing SRC:

```

[bestpos,bestsrp,nofes]=src(magiconst,lsb,usb,randpts,npoints,mic_loc,yintpt
,row1,efsintp);

```

% Outputs:

```

finalpos=bestpos %%% Final source location estimate
finalsrp=bestsrp/np %%% Final source's SRP-PHAT value (normalized by
number of pairs)
finalfe=nofes+randpts %%% Number of fe's used.

```

%%% SRC algorithm:

```

function [bestpos,bestsrp,nofes] =
src(magiconst,bstart,bend,randpts,npoints,mic_loc,yintpt,row1,efsintp)
    %%% This function does SRC algorithm.
    %%% It calls subroutine "fe".

```



```

        points newbestvec=bestvec(gidx,:); %%% keep all the points
        having srp
values above the mean
        la5=numel(a5);
        nonewpts(kk)=npoints-la5; %%% the number of new points
        needed
to evaluate

        newbstart=[min(newbestvec(:,1)) min(newbestvec(:,2))
min(newbestvec(:,3))]; %%% Contract search-volume to the volume
containing only the high points
        newbend=[max(newbestvec(:,1))
max(newbestvec(:,2)) max(newbestvec(:,3))];

        if nonewpts(kk)>=1
            for i=1:
                nonewpts(kk)
                a4=0;
                while a4 < meana4 && countfe <=maxFEs %%%kee
p evaluating new random points until we fill up 'nonewpts' points that have
SRP values above the mean
                    [a4,goodpos] =
fe(newbstart,newbend,magiconst,mic_loc,yintpt,row1,efsintp);
                    countfe=countfe+1;

                end

                if a4~=0
                    a5=[a5 a4];
                    newbestvec=[newbestvec;
goodpos];
                end

            end
        end
        srp_vec=a
        5; clear
        bestvec
        bestvec=newbestvec;
        clear a5 newbestvec a4 goodpos newbstart newbend;

        searchVol=(max(bestvec(:,1))-
min(bestvec(:,1)))*(max(bestvec(:,2))-min(bestvec(:,2)))*(max(bestvec(:,3))-
min(bestvec(:,3))); %%% The new search-volume containing only the best N
points

        TotalFEs=randpts+countfe;
        kk=kk+1;
    end

```

```

    [v,i]=max(srp_vec);
end
bestsrp=v;
bestpos=bestvec(i,:);
nofes=TotalFEs;
clear searchVol srp_vec

```

%%% Functional Evaluation sub-routine (calculate SRP-PHAT value for a point in the search space):

```

function [yval1,position1] =
fe(bstart,bend,magiconst,mic_loc,yintpt,row1,efsintp)
    %%% This function evaluates an 'fe' in the search space, gives back the
'fe' position and its SRP-PHAT value.

```

```

M=size(mic_loc,1); %%% number of mics

```

```

    %%% generate a random point in the search space:

```

```

    y=rand(1,3);
    x=(bstart.*(1-y)+bend.*(y)); % This vector x defines
the coordinates (x,y,z) of the rand point x

```

```

    %%% Find the distances from M-microphones to the
point: a1=ones(M,1);

```

```

    xx1=a1*x;
    xdiff1=xx1-
mic_loc;
    dists=sqrt(sum(xdiff1.*xdiff1,2));

```

```

    %%% Differences in distances

```

```

ddiffs_ones=ones(M,1)*dist
s'; ddm=ddiffs_ones-
ddiffs_ones';

    %%% Calculate the TDOA index:
% v=ddm(gidM);
v=nonzeros(tril(ddm,0
));

%ddiffsi32=int32(round(magiconst*v+efsintp));
%ddiffsi32=floor(magiconst*v+efsintp)+1;
ddiffsi32=round(magiconst*v+efsintp+1.5);
row=row1+ddiffsi32;

    %%% Pull out the GCC-PHAT value corresponding to that TDOA:
v1=yintpt(row);
    %%% SRP-PHAT value of the point:
yval1=sum(v1);
position1=x;
end
end

```

Kod C: İğnecikli sinir ağı

```
clear; clear all; close all; clc
```

```
aux1=dlmread('posdat.txt'); % veri girisi
```

```
szi = size(aux1);  
shft = randperm(szi(1));  
aux1 = aux1(shft,:) + .3*(rand(szi)-.5);  
%-----
```

```
% baslangic parametreleri
```

```
out_neu=5; % çıkış sinir hucre sayisi (sinif sayisi)
```

```
rho=0.1; % zamanda cozunurluk
```

```
conj=400; % egitim kumesi
```

```
tau=3; % cte de tempo do
```

```
EPSP
```

```
rf=[8 8 8]; % girdi basina sinir hucre sayisi
```

```
sigma=[1/(1.5*(rf(1)-2)) 1/(1.5*(rf(2)-2)) 1/(1.5*(rf(3)-2))]; % alici  
alanlarin genislikleri
```

```
f_cut=0.9;
```

```
maxd=10; % kodlama araligi
```

```
%-----
```

```
% Ogrenme parametreleri
```

```
d=[0 1 2 3 4 5 6 7 8 9]; % alt sinapslarin gecikmeleri w_max=1;
```

```
w_min=0; % agirliklarin maksimum ve minimum degerleri
```

```
max_epoch=3; % maksimum epok sayisi
```

```
t_max=30; % maksimum egitim suresi
```

```
eta=0.35; % ogrenme hizi
```

```
beta=0.2; % cte para janela de
```

```
aprendizado nu=5.0; % komsuluk
```

```
dx=2.3; % ogrenme penceresinin kaymasi
```

```
% p/ esquerda (+) ou direita (-)
```

```
%-----
```

```
% parametreleri hesaplar
```

```
kapa=1-(nu^2)/(2*log(beta/(beta+1))); % cte para janela de aprendizado
```

```
in_neu=sum(rf); % qtde de neuronios na entrada
```

```
ssin=size(d,2); % qtde de sub-sinapses
```

```
%teta=1.5*(in_neu/out_neu)*(w_max-
```

```
w_min)*ssin/2; teta=12; % esik degeri
```

```
%-----
```

```
% baslangic agirliklari
```

```
w=randn(in_neu,out_neu,ssin);
```

```
for i=1:in_neu
```

```
for j=1:out_neu
    w(i,j,:)=w_min+rand(1,ssin).*(w_max-
    w_min);
```



```

end
%-----

% egitime
başla tic;
aus = kodlama(aux1,rf,maxd,sigma,f_cut,rho,0,0); % girdileri kodlar
ctrl_1 = 1; % kontrol dongusu
delta_t=zeros(1,1,ssin);
h=waitbar(ctrl_1/max_epoch,sprintf('%i no"lu iterasyonun %i no"su',
max_epoch,ctrl_1));

f_times=zeros(2,1);
while
ctrl_1<=max_epoch
    for z=1:conj
        tr=aus(z,:);
        [neu, inst] = atesleme(tr,t_max,w,d,tau,rho,teta,0);
        f_times=[f_times [neu;inst]];
        if neu~=0 &
            size(neu,2)==1 for
                i=1:in_neu
                    if tr(i)==-1
                        w(i,neu,:)=w(i,neu:)-eta;
                    else
                        delta_t(1,1,:)=tr(i)+d-inst; % delta_t hesapla
                        w(i,neu,:)=w(i,neu:)+eta.*((1+beta).*exp(-
(((delta_t+2.3).^2)/(2*kapa-2)))-beta);
                    end
                    ndx=find(w(i,neu,:)<w_min); % limitleri garantiler
                    for u=1:size(ndx,1)
                        w(i,neu,ndx(u))=w_min;
                    end
                    ndx=find(w(i,neu:)>w_max); % limitleri garantiler
                    for u=1:size(ndx,1)
                        w(i,neu,ndx(u))=w_max;
                    end
                end
            end
        end
    end
end
    en
end d
end d
    ctrl_1=ctrl_1+1; % epok sayisini 1 arttir
    waitbar(ctrl_1/max_epoch,h,sprintf('%i no"lu iterasyonun %i no"su',
max_epoch,ctrl_1));
    teta=teta+(0.3*teta)/max_epoch;

end
close
(h);
toc;
%-----

% agi test eder

```

```
[conj,in_neu]=size(aus);  
class=zeros(1,conj); % sinif etiketleri olusturur for  
i=1:conj  
    tr=aus(i,:); % test kumesi  
    [neu,inst] = atesleme(tr,t_max,w,d,tau,rho,teta,0); % atesleyen hucre ve  
    zamani  
    if size(neu,2)>1 % se deu  
        empate... neu=0; % sinif = 0
```



```

class(i)=neu; % guarda a classe
end
%-----

a1=find(class==1);      m1      =
mean(aux1(a1,:)); a2=find(class==2); m2
= mean(aux1(a2,:)); a3=find(class==3);
m3      =      mean(aux1(a3,:));
a4=find(class==4);      m4      =
mean(aux1(a4,:)); a5=find(class==5); m5
= mean(aux1(a5,:));

m=[m1;m2;m3;m4;m5];
%-----

% test sonuclarini ciz
ptos={'.' 'r' 'm' 'g' 'c' 'sk' 'sk'... 'sk' 'sk' 'ob'
'+k' 'xr' 'sb' 'dk'};

figure(1)
for i=1:out_neu
    a=find(class==
    i); for
    n=1:size(a,2)
        plot3(aux1(a(n),1),aux1(a(n),2),aux1(a(n),3),char(ptos(i))) hold
        on
        plot3(m(i,1),m(i,2),m(i,3),'ok','MarkerSize',50) hold
        on
    end
end
a=find(class==
0); for
n=1:size(a,2)
    plot3(aux1(a(n),1),aux1(a(n),2),aux1(a(n),3),char(ptos(out_neu+1))); hold
    on
end grid hold off

```


VESİKALIK
FOTO

ÖZGEÇMİŞ

Ad-Soyad : Peren Jerfi CANATALAY

Doğum Tarihi ve Yeri : 1989/ANKARA

E-posta : perenjerfi@hotmail.com

ÖĞRENİM DURUMU:

- **Lisans** : 2013, İstanbul Aydın Üniversitesi, Mühendislik – Mimarlık Fakültesi, Elektrik Elektronik Mühendisliği (İngilizce)
- **Yükseklisans** : 2016, İstanbul Aydın Üniversitesi, Elektrik Elektronik Mühendisliği Anabilim Dalı, Elektrik Elektronik Mühendisliği Programı