

**YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**DÜŞÜNSEL YÖNELİM EEG'SİNİN
YAPAY SİNİR AĞLARI İLE KONTROLDA
KULLANIMINA YÖNELİK SİMÜLASYONU**

Elektrik Mühendisi Özber ÇİN

**FBE Elektrik Mühendisliği Anabilim Dalında
Hazırlanan**

YÜKSEK LİSANS TEZİ

Tez Danışmanı : Prof.Dr. Galip CANSEVER 21.07.05
Doc. Dr. Tülay YILDIZIM 21.07.05
Yrd. Doç. Dr. İbrahim B. KüçükDEMİREL 21.07.05

İSTANBUL, 2005

İÇİNDEKİLER

	Sayfa
SİMGE LİSTESİ	iii
KISALTMA LİSTESİ	iv
ŞEKİL LİSTESİ	v
ÇİZELGE LİSTESİ	vii
ÖNSÖZ.....	viii
ÖZET	ix
ABSTRACT	x
1. GİRİŞ.....	1
2. HÜCRE VE SİNİR SİSTEMİ	11
2.1 Sinir Sisteminin Ana Elemanları	14
3. ELEKTROENSEFALOGRAM VE İŞARETLERİNİN ÖLÇÜMÜ	20
3.1 EEG İşaretlerinin Ölçümü	22
3.2 EEG İşaretlerinin Analizi	28
3.2.1 Güç Spektrum Analizi	28
3.2.2 Spektral Yaklaşım Metotları.....	29
3.2.2.1 Yule-Walker AR Yöntemi.....	30
3.2.2.2 Burg Yöntemi	31
4. YAPAY SİNİR AĞLARI.....	33
4.1 YSA'nın Yapısı ve İşleyişi.....	36
4.2 YSA'nın Öğrenmesi ve Eğitilmesi.....	40
4.3 RBF (Radial Basis Function) Ağ Yapısı	45
4.3 Elman Ağ Yapısı.....	47
5. EEG SİNYALLERİNİN SİMÜLASYON ORTAMINDA TANINMASI.....	49
5.1 MATLAB Ortamında EEG Veri Setinin Tanınması	50
6. SONUÇLAR VE ÖNERİLER.....	70
KAYNAKLAR.....	72
ÖZGEÇMİŞ.....	75

SİMGE LİSTESİ

$S_{xx}(w)$	Güç spektrumu
$R_{xx}(m)$	Korelasyon sekansı
fs	Örnekleme frekansı (Hertz)
$P_{xx}(w)$	Gelişigüzel sinyal x_n 'in güç spektral yoğunluğu
$P_{AR}(f)$	AR metotları ile bulunan güç spektral yoğunluğu
$\hat{a}_p(k)$	AR parametreleri
W	Yapay sinir ağlarında ağırlık vektörü
Xk	O anki bayasları ve ağırlıkları ihtiva eden vektör
gk	O anki gradyen değeri
a_k	Öğrenme parametresi
β_k	Fletcher-Reeves güncelleme prosedürü
A_k	Hessian matrisini (performans indeksinin ikinci türevi)
J	Jacobiyen
mc	Momentum katsayısı

KISALTIMA LİSTESİ

EEG	Elektroensefalogram
E.R	Endoplazmik Retikulum
YSA	Yapay Sinir Ağları
DNA	Deoksiribonükleik asit
RNA	Ribonükleik asit
Na	Sodyum
K	Potasyum
V	Volt
mV	Mili volt (10^{-3} V)
μ V	Mikro volt (10^{-6} V)
Hz	Hertz
s	Saniye
dk	Dakika
Khz	Kilo Hertz (10^3 V)
MSS	Merkezi Sinir Sistemi
PSS	Perefral Sinir Sistemi
SSS	Somatik Sinir Sistemi
VGA	Video Graphics Array
SVGA	Super Video Graphics Array
AR	Auto-regressive
LMS	Least Mean Squares
FFT	Fast Fourier Transform
PSD	Power Spectral Density
MLP	Multi Layer Perceptron
RBF	Radial Basis Function
BCI	Brain Computer Interface
PCA	Principle Component Analysis
GPS	Global Pattern Similarity
HMM	Hidden Markov Model
ICA	Independant Component Analysis
API	Application Programming Interface
ERD	Event Related Desynchronization
VEP	Visually Evoked Potentials
MIT	Massachusetts Institue of Technology
GA	Genetik Algoritma

ŞEKİL LİSTESİ

	Sayfa
Şekil 1.1 Çalışma prensibi ve sonuçlar.....	4
Şekil 1.2 Sayma işleminin bip'ten önceki ve sonraki hali ile Fourier dönüşümü.....	5
Şekil 1.3 'Mind Balance' yazılımının ekran görüntüsü.....	8
Şekil 1.4 'Mind balance' sisteminin ara birimi	9
Şekil 2.1 Hücrenin yapı taşları	11
Şekil 2.2 Taşıyıcı proteinler.....	11
Şekil 2.3 İşaretsel proteinler	12
Şekil 2.4 Reseptör proteinler	12
Şekil 2.5 Taşıyıcı reseptörler.....	12
Şekil 2.6 Hücre membranı ve kanal proteinleri.....	13
Şekil 2.7 Simportlar.....	13
Şekil 2.8 Nöronlar ve sinir ağı.....	14
Şekil 2.9 Myelin kılıfı ve ranvier nodu	15
Şekil 2.10 Sinir kesiti ve aksiyon potansiyeli.....	15
Şekil 2.11 Sodyum potasyum pompası	16
Şekil 2.12 Aksiyon potansiyeli.....	17
Şekil 2.13 Aksiyon potansiyelinin myelinli aksandaki hareketi.....	18
Şekil 2.14 Sinaps ve nörotransmitterler.....	18
Şekil 2.15 Sinaptik yarık	19
Şekil 3.1 Elektroensfalogram kaydı	20
Şekil 3.2 Alfa dalgası	21
Şekil 3.3 Beta dalgası.....	21
Şekil 3.4 Teta dalgaları.....	21
Şekil 3.5 Delta dalgaları	22
Şekil 3.6 Darbe-dalga aktivitesi.....	22
Şekil 3.7 10/20 elektrot pozisyonları.....	23
Şekil 3.8 Diferansiyel amplifikatör	24
Şekil 3.9 Amplifikatör çalışma prensibi.....	24
Şekil 3.10 Ortak referans türetme.....	24
Şekil 3.11 Ortalama referans türetme	25
Şekil 3.12 Bipolar türetme.....	25
Şekil 3.13 Analog dalga formu.....	26
Şekil 3.14 240 Hz'de örnekleme.....	26
Şekil 3.15 50 Hz'de örnekleme.....	27
Şekil 3.16 Ortak referansla kaydedilmiş bilginin bir başka montaja dönüşümü.....	28
Şekil 3.17 Burg yöntemiyle bulunan spektral yoğunluk değerinin frekans oynamalarına karşı etkiye açıklığı.....	32
Şekil 4.1 Von-Neumann makinesi.....	34
Şekil 4.2 Basit nöron yapısı.....	36
Şekil 4.3 Transfer fonksiyonları	37
Şekil 4.4 Transfer fonksiyonları (devam).....	37
Şekil 4.5 Vektör girişli nöron	38
Şekil 4.6 Çok girişli nöron katmanı.....	38
Şekil 4.7 Çok katmanlı nöron ağı yapısı.....	39
Şekil 4.8 Resilient backpropagation algoritması	42
Şekil 4.9 Radial basis nöron yapısı.....	45
Şekil 4.10 Radial basis fonksiyonu	45

Şekil 4.11 RBF ağ yapısı	46
Şekil 4.12 Elman ağ yapısı.....	48
Şekil 5.1 Elektrot pozisyonları ve zamanlama sekansı.....	50
Şekil 5.2a "1"(sağ) durumunu inceleyen C1 kanalındaki dalga formu..	51
Şekil 5.2b "1"(sağ) durumunu inceleyen C3 kanalındaki dalga formu	52
Şekil 5.3a "2" (sol) durumunu inceleyen C1 kanalındaki dalga formu.....	52
Şekil 5.3b "2" (sol) durumunu inceleyen C3 kanalındaki dalga formu.....	52



ÇİZELGE LİSTESİ

	Sayfa
Çizelge 1.1 İkili ayırımdan elde edilen sonuçlar	6
Çizelge 1.2 Üçlü ayırımdan elde edilen sonuçlar	6
Çizelge 5.1 Yule-walker ön işleminden geçirilmiş verilerin YSA Algoritmalarıyla Eğitim Sonuçları	57
Çizelge 5.1a traingda	57
Çizelge 5.1b traingdx	57
Çizelge 5.1c trainrp	57
Çizelge 5.1d traingcf	58
Çizelge 5.1e traingcp	58
Çizelge 5.1f trainbfg	59
Çizelge 5.1g traingcb	59
Çizelge 5.1h trainoss	60
Çizelge 5.1j trainscg	60
Çizelge 5.1k trainlm	60
Çizelge 5.1l trainbr	61
Çizelge 5.1m newrb	61
Çizelge 5.1n newrbe, newgrnn, newgnn	62
Çizelge 5.1o newcf	62
Çizelge 5.1p newelm	63
Çizelge 5.2 Burg ön işleminden geçirilmiş verilerin YSA Algoritmalarıyla eğitim sonuçları	63
Çizelge 5.2a traingda	63
Çizelge 5.2b traingdx	63
Çizelge 5.2c trainrp	64
Çizelge 5.2d traingcf	64
Çizelge 5.2e traingcp	65
Çizelge 5.2f trainbfg	65
Çizelge 5.2g traingcb	66
Çizelge 5.2h trainoss	66
Çizelge 5.2j trainscg	66
Çizelge 5.2k trainlm	67
Çizelge 5.2l trainbr	67
Çizelge 5.2m newrb	67
Çizelge 5.2n newrbe, newgrnn, newpnn	68
Çizelge 5.2o newcf	68
Çizelge 5.2p newelm	69
Çizelge 5.3 Verimin örnekleme süresine göre değişimi	69

ÖNSÖZ

Bu tez çalışmasında yardımlarından dolayı başta değerli hocam Sayın Prof.Dr.Galip CANSEVER olmak üzere sevgili arkadaşım Tarık Veli MUMCU'ya; göstermiş oldukları sonsuz sabır ve anlayıştan dolayı da çok sevdiğim Özün GÖREL'e ve aileme teşekkürü bir borç bilirim.



ÖZET

Bu tez çalışmasında, elektroensefalogram (EEG) sinyallerini incelemek amacıyla sağ ve sol el hareketleri arasındaki farkı ayırt edip, fiziksel engellilere yardımcı dokunacak bir sınıflandırma sistemi sunulmuştur. Üç kanal üzerinden kayıt edilmiş olan data kümesi (Graz Data, Department of Medical Informatics, Institute for Biomedical Engineering, University of Technology Graz tarafından sağlanmıştır.) hayali sağa-sola yönelim ile kontrol edilen bir geribesleme seansı esnasında normal bir denekten elde edilmiştir. İleri beslemeli bir yapay sinir ağının farklı konfigürasyonları ile EEG datası belirlenmiş olan iki sınıftan birine çıkış verecek şekilde eğitilmiştir. Data boyutları, EEG'yi temsil etmesi için oto-regresif (AR) model kullanılarak aşağıya çekilmiştir. Ağ performansı test setinin verdiği sonuca göre oluşturulmuştur.

Anahtar Kelimeler: motor-görüntü, elektroensefalogram (EEG), yapay sinir ağları, oto-regresif (AR), fiziksel engelliler, düşünsel faaliyetler, BCI (Beyin Bilgisayar Arabirimi).

ABSTRACT

In this study, the discrimination of left and right hand movements by means of electroencephalogram (EEG) signals is formed into a classification system that has the purpose of helping physically handicapped individuals. Three-channel EEG data (Graz Data provided by Department of Medical Informatics, Institute for Biomedical Engineering, University of Technology Graz.) was recorded from a feedback session during which the task was to control a feedback bar by means of imagery left or right hand movements. A feed-forward neural network with different configurations is trained to classify EEG data into one of the two classes which corresponds to either right or left hand movements. Reduction of data's dimension is sought by means of using an autoregressive (AR) model to represent the EEG signals. The performance of network was based on the classification rate of the test set.

Keywords: motor imagery, electroencephalogram, neural networks, autoregressive (AR), physically handicapped, mental tasks, BCI (Brain Computer Interface).



1. GİRİŞ

Tarih boyunca, insan beyni, içinde sayısız sırlar barındıran bir kara kutu olarak algılanmıştır. Epilepsi ve diğer nöral rahatsızlıkların tedavisi için gerekli olan Elektroensefalogram (EEG) inceleme düzeneğinin ilk olarak 1949 yılında Houston’da kurulmasının [1] üzerinden bilimsel anlamda çok uzun bir zaman dilimi geçmesine rağmen, konu geçerliliğini hiç bir surette yitirmemiştir. Aksine, bilimin aydınlatıcı gerçekliğinin eşiğinde edinilmiş tecrübeler, bu mevzunun uzun vadede daha farklı alanlara da kayabileceğini göstermiştir. Teihard de Chardin (1959) “Düşünceler en temel anlamda özgürdür; hem çıkış itibariyle, hem de ortaya konuş biçimiyle.” demiştir. Belki de bu özgürlük bilim adamlarını ve akademisyenleri düşünceleri betimleyen EEG’yi incelemeye ve anlamaya yöneltmiştir.

Kuşkusuz, bir kontrol sisteminde tüm zamanların en aranılan kriteri mutlak kontrol olanağıdır. Bu yüzdendir ki insanoğlu içinde bulunduğu ham bilgilerden oluşan çevreyi işlemeye çalışmış, onu maniple etmenin farklı farklı yollarında gidip gelmiştir. Gerçekte üretilmiş olan hiçbir kontrol sistemi bizim düşünsel faaliyetlerimizi idare ettiğimiz kadar mükemmel çalışmamaktadır. İnsan beyni oto-kontrol mekanizması olarak halen en harika örnektir. Gerek ‘mimari’ olarak, gerek kendini geliştirebilme potansiyeli göz önüne alındığında, beynin serebral yetkinliği bilincin yansıması niteliğindedir. Bilincimizin belirlemiş olduğu kararlar, sinirsel uyarımlar olarak hedef organa ulaştığında, hareket, vücut sınırları içersinde en optimal şekilde gerçekleşir. Bu durumun işleyişinde kuşkusuz en takdire şayan özellik ardına yapılan hareketlerin bir zaman sonra daha kolay bir şekilde icra edilebilmesi, yani beynin bu bilgiye adapte olabilmesi veya diğer bir deyişle öğrenebilmesidir. Nöron denilen hücreler vücutta işlemsel öğrenmenin alt yapısını (bir bilgisayarın mikroişlemcilerini temsil etmekle eşdeğer bir şekilde) oluşturmaktadır. Geçmişte karşılaştığımız bilgiler beyinde yaygın olarak kodlanmış bir şekilde barınırken, bu veya buna benzer bir veri girdisi (bu bir düşünce veya hareket olabilir) önceki izlenimlerimizi ‘hatırlamamıza’ yol açacaktır. Yapacaklarımızın adı geçen bu izlenimlerin yorumlanması sonunda olacağı bir gerçektir. Benzer bir işleyişe sahip olan yapay sinir ağları nöronların davranışlarının pratik matematiksel modelleridir. Paralel işlem yürütebilmesi ve kendi ‘donanımını’ ihtiyaçlara göre değiştirebilmesi yeni öğrenen bir bebek kadar gerçeklerle baş edebilmesini sağlamaktadır. Bebek gittikçe gelişmekte bilgi ve tecrübelerinin birikimi sonucunda sayısız karar alabilmektedir. Yapay sinir ağlarının (YSA) bu esnekliği tanımlanamayan ya da çok karmaşık bir modele sahip pek çok konuda uygulama imkanı bulabilmesine yol açmıştır. Bilgisayar devriminin meyveleri olarak tabir edebileceğimiz yazılımsal ve donanımsal çözümler, YSA’nın işlem kabiliyetini

arttırmakla kalmayıp, bir takım bütünleşik algoritmalar sayesinde adaptif olarak öğrenebilen evrensel yakınsayıcıların (Bkz. Bölüm 4) oluşturulabilmesine imkan vermiştir.

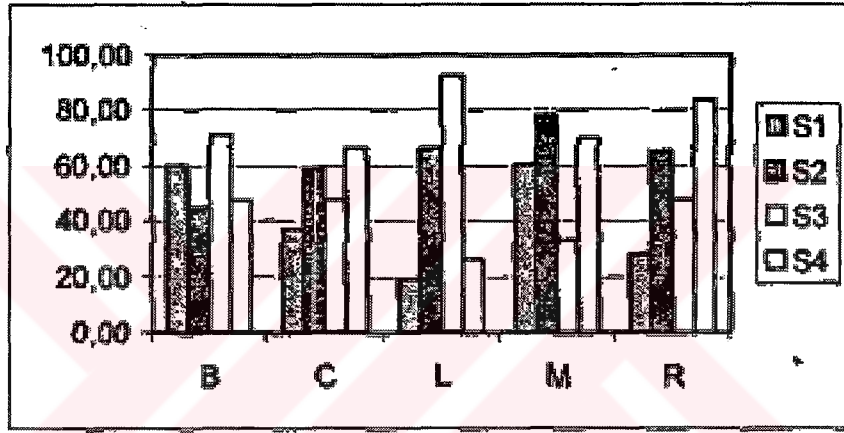
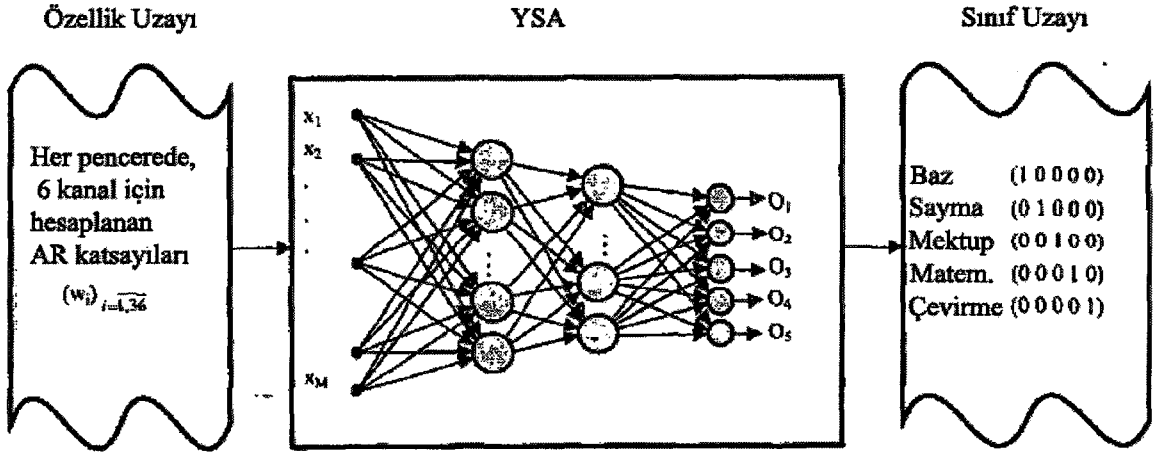
Kaderin bir oyunu olsa ki sınırların işleyişinden ortaya çıkmış bir teori (YSA), sonradan ‘gerçek sinir ağlarının’ nasıl çalıştığını öğrenebilmek için uygun bir araç olmuştur [14]. Düşüncelerin saklı dünyasında yolculuk etmeye alışkın olan bir bilgisayar arabirimi bir istenen olmaktan çıkmış, artık ihtiyaç duyulan bir uygulama haline gelmiştir. En ilkel bilgisayar arabirim örneklerinden biri olan sürgülü kartlardan sonra klavye ve fare çok rahat düzenekler gibi görülebilir; ama, dünya üzerinde bu araç gereçleri kullanamayan milyonlarca insan yaşamaktadır. Vücutsal işlevlerini yerine getiremeyen, yatağa veya tekerlekli sandalyeye bağımlı bireylerin isteklerini sadece düşünerek yerine getirebilmeleri belki de tahmin ettiğimizden çok daha yakındır (Guger, 2003).

Çoğu beyin bilgisayar arabirimi uygulaması temelinde EEG örüntülerinin bir şekilde tanınmasını ve bu tanınan parçacıkların sisteme dinamik olarak kodlanmasını barındırır. Daha sonra adı geçen kodlanmış veri bilgisayar aracılığıyla çözümlenerek, kontrol edilecek düzeneğe hareketi sağlayacak sinyaller gönderilir. Akla gelen soru, bunun direkt bir şekilde bilgisayar tarafından algılanıp algılanamayacağıdır. Düşündüklerimizin doğrudan okunabilmesine imkan verecek bir düzenek şu an için bile mümkün değildir. Bu konudaki asıl sorun işlemsel yetersizlikten çok insanların farklı düşünce normlarına sahip olması gerçeğidir. Tek bir düşünce kalıbı ile EEG’si arasındaki bağlantı sadece o şahıs için geçerli olacaktır. Ama, ufak bir benzerlik bile bazen yapay sinir ağlarının çıkartım yapmasını olanaklı kılmaktadır.

İnsanoğlunun düşünceleri ile çevresi arasında irtibat kurabilme çabası uzun süreden beridir varolan bir olgudur (Hinterberger, 2003). Genel olarak BCI (Brain Computer Interface, Beyin Bilgisayar Arabirimi) başlığı altında toplanmış bir çok çalışma (Cincotti, 2003), EEG datasını çeşitli sistematik yöntemler aracılığıyla inceleyip görsel veya yönelimsel bilgiyle ilişkilendirmeyi amaçlamıştır. EEG üzerinde yapılmış çalışmaların tarihine bakacak olursak görürüz ki, beyindeki elektriksel aktiviteler değişik bölgelerde meydana gelmekte ve bu elektriksel oluşumların senkronizasyonu sonucunda düşünsel faaliyetler ortaya çıkmaktadır. Dolayısıyla beyindeki EEG yayılımını (uzay domeni) ve EEG dalga formunu (zaman domenini) incelemek suretiyle tanıma gerçekleştirebilecek bir sistem tasarlayabilme imkanımız oldukça fazladır (Baillet, 2001). *Liu ve Gao* yayınladıkları makalelerinde (Liu ve Gao, 2003) hayali el hareketlerini tanıyabilen bir metoda işaret ederek olası stratejiyi açıklamışlardır. EEG kayıt datası 10/20 sistemine uygun olarak yerleştirilmiş 59 elektrot

aracılığıyla 100Hz'de örneklenmiştir. Üç bant-geçişli filtre aracılığıyla temel olarak θ - δ , α , β dalgalarına ayrıştırılmıştır. İlk başta bilginin gürültüden arındırılması amaçlanmıştır. Bu sağa ve sola yönelimi simgeleyen X_A ve X_B eğitim data setlerinin kovaryans matrisi toplamları alındığında oluşan matris PCA (Principle Component Analysis, ana komponent analizi) uygulanmak (matrisin rankını bulmak kaydıyla) suretiyle sağlanabilir. Ardından aranan komponentleri (sağa yönelim ve sola yönelim) içeren bilginin beyin loblarındaki yaklaşık yeri bulunmuştur. Burada Liu ve Gao, kendi geliştirdikleri bir algoritma olan 'Shrinking LORETA-FOCUSS'dan yardım almıştır. Gene kendi buluşları olan GPS (Global Pattern Similarity, Global Örüntü Benzerliği) kriterini kullanarak GPS'in maksimuma ulaştığı anlarda (T_R , Representative Time, Temsili Zaman), çözüm uzayındaki EEG özellikleri kaydedilerek bir sonraki aşamaya taşınmıştır. Bu sayede, temsili zaman T_R ve temsili zamanlara tekabül eden akım kaynaklarının (i) 3D tomograftaki yeri ile enerjisi (i^2) olmak üzere iki adet bilgiye ulaşılmış oluruz. Bu bilgiyle baş etmek için görevlendirilen HMM (Hidden Markov Model, Gizli Markov Modeli) son klasifikasyon işlemi yapar. HMM (Chiappa ve Bengio, 2004), (Obermaier vd., 2003), sinyalin bir kaç durum ve bu durumların olasılığıyla karakterize edildiği stokastik (olasılıklı) bir yöntemdir. X_A ve X_B eğitim datasına karşılık gelen iki HMM bulunmaktadır. Test datası bu iki modele uygulandığında en yüksek çıkışı verene (en yüksek olasılığa sahip olana) göre çıkarım yapılır (EEG datasında olasılık hesabı üzerine ek bilgi için (Felzer ve Freisleben, 2003) kaynağına göz atılabilir). Üç denekten alınan bilgidен elde edilen sonuçlar göstermiştir ki ortalama verim 81% olmaktadır.

Maiorescu ve Lazör yayınladıkları makalelerinde (Maiorescu vd., 2003) altı kanal üzerinden edinilmiş birbirinden farklı karakterdeki beş düşünsel aktiviteyi betimleyen EEG datasını öncelikle AR modelini çıkarmak suretiyle basite indirgemiş, ardından bunu ileri beslemeli bir yapay sinir ağına yedirerek sonuçlarını bilim camiası ile paylaşmışlardır. Edinilen verim bir test setine istinaden bulunmuştur (Şekil 1.1). Düşünsel faaliyetler (Millan vd., 2002), *sükunet* (deneğin gevşemesi), *mektup* (deneğin bir arkadaşına hayali bir mektup oluşturması), *sayma* (deneğin hayali bir kara tahtaya sıralı sayıların görünüp kaybolmasını düşünmesi), *matematik* (deneğin kafadan hesaplaması verilen sürede mümkün olmayan bir çarpım işlemi) ve son olarak da *çevirme* (deneğin 30 saniye boyunca izlediği üç boyutlu cisim bir eksen etrafında döndürdüğünü düşünmesi) olarak sıralanabilir. 9. dereceden alınan AR modeli Yule-Walker eşitliğiyle parametrelerine indirgenmiş, edinilen bilgi ileri beslemeli bir ağın öğrenimine sunulmuştur.

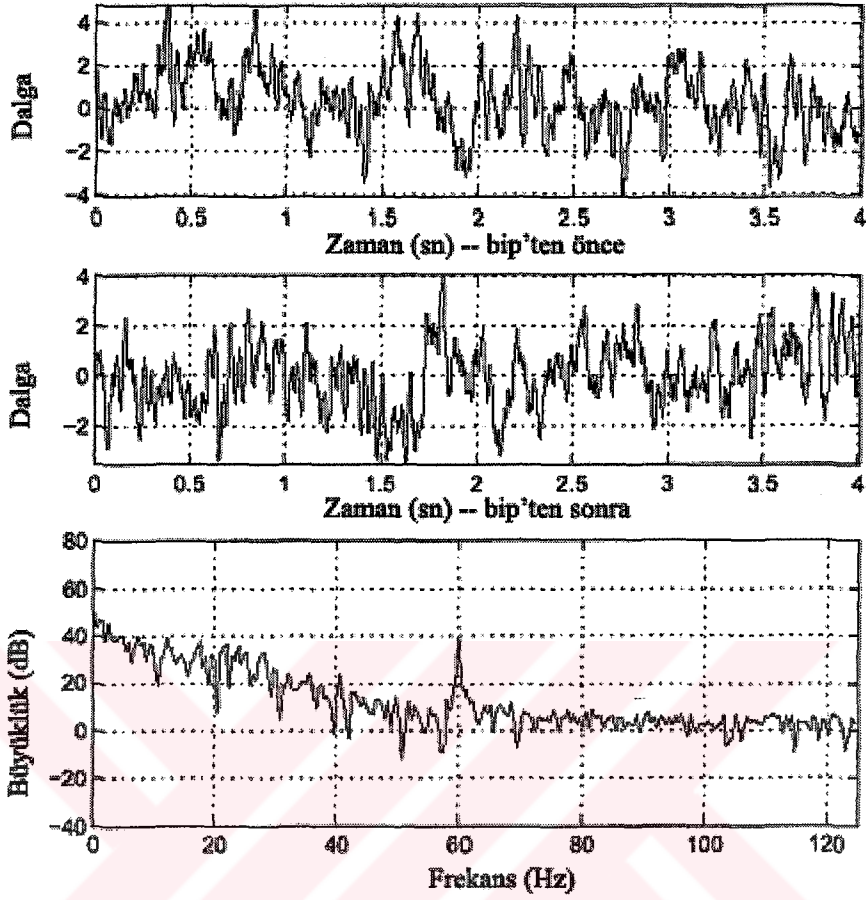


Şekil 1.1 Çalışma prensibi ve sonuçlar (Maioreescu vd., 2003)

Kullanılan sistem stratejisi ve elde edilen sonuçlar üstteki şekilde belirtilmiştir. Elde edilen sonuçlar göstermiştir ki data ilk başta bir ön işlemden geçirildiği takdir de ihtiyaç duyulan önemli bilginin bir kısmı heba olmaktadır. Bir diğer önemli unsur ise bir geri besleme olmaksızın denegin konsantrasyonunu sağlamanın korumasının zor olduğudur. Bu yüzden verim özellikle göz önünde canlandırması zor faaliyetler için düşmekte ve hatta diğer düşünsel işlerle aralarında YSA'nın fark edebileceği kadar değişiklik olamamaktadır.

Culpepper ve Keller (2003) makalelerinde 250Hz örnekleme hızıyla 12 kanal üzerinden kaydedilmiş EEG dadasını, 5 farklı beyinsel aktiviteyi temsil eden sınıflara ayırabilmenin mümkün olup olmadığını ortaya koymuşlardır. Burada bahsi geçen beyinsel aktiviteler (Maioreescu vd., 2003) kaynağındakilerle aynıdır. ICA (Independent Component Analysis, Bağımsız Komponent Analizi) (Delsanto vd., 2003) yardımı ile bazı bilinen ve istenmeyen dalga parçacıkları süzölmüş; edinilen verinin güç spektral yoğunluk vektörleri ise çok katmanlı YSA ile ilişkilendirilmiştir. ICA sayesinde göz ve kas hareketleri elimine edilirken

data 11 kanala indirilmiştir. Şekil 1.2 'sayma' işleminin bip sesinden önceki ve sonraki halini ve Fourier dönüşümünü göstermektedir.



Şekil 1.2 Sayma işleminin bip'ten önceki ve sonraki hali ile Fourier dönüşümü
(Culpepper ve Keller, 2003)

Bu makalenin en çarpıcı kısmı verinin tümünden ziyade ufak bir kısmından yola çıkarak çözüme ulaşılmasıdır. Zira 1.5 s gibi uzun bir süre boyunca (kontrol sistemleri için) örnekleme yapması gereken bir sistem hareket imkanını kısıtlayacaktır. Öte yandan saniyenin beşte biri veya belki de yirmi de birine indirgenmiş bir pencereleme durumu, ihtiyaçlarımızı karşılamaya yetecektir.

Çizelge 1.1'te temel beş beyinsel aktivitenin iki tanesi arasında ayırım yapabilen bir sistemin sonuçları belirtilmiştir. Çizelge 1.2'te ise arasında ayırım yapacağı aktiviteler üçe çıkartılırsa durumun ne olacağını belirtmek için verilmiştir. Edinilen sonuçlar ikiliden üçlüye geçişte bile fazlasıyla düştüğü için, bu aşamadan sonra beş aktivite arasında çıkarım yapabilecek bir sistem araştırılmamıştır.

Çizelge 1.1 İkili ayırmadan elde edilen sonuçlar (Culpepper ve Keller, 2003)

Pencere Uzunluğu (s, örnek)	En İyi Ağ (1.katman, 2. katman)	En İyi Verim	En Kötü Verim	Eğitim Süresi (dk)
(0.5, 125)	(40, 5)	94% geometri, çarpma	77% sayma, çarpma	7
(0.25, 62)	(40, 5)	90% baz, mektup	69% sayma, mektup	6
(0.125, 31)	(100, 10)	82% baz, sayma	69% sayma, mektup	3
(0.05, 13)	(1000, 100)	85% geometri, çarpma	67% sayma, mektup	10

Çizelge 1.2 Üçlü ayırmadan elde edilen sonuçlar (Culpepper ve Keller, 2003)

Pencere Uzunluğu (s, örnek)	En İyi Ağ (1. katman, 2. katman)	En İyi Verim	En Kötü Verim	Eğitim Süresi (dk)
(0.5, 125)	(100, 10)	86% baz, mektup, çarpım	71% sayma, mektup, çarpım	30
(0.25, 62)	(250, 50)	77% sayma, geo, mektup	63% sayma, geo, mektup	16
(0.125, 31)	(250, 50)	74% geo, mektup, çarpım	56% sayma, mektup, çarpım	8
(0.05, 13)	(250, 50)	66% geo, mektup, çarpım	51% sayma, mektup, çarpım	6

Fabiani, McFarland ve ekibi yayınladıkları makalelerinde (Fabiani vd., 2004), Wadsworth EEG bazlı bir BCI arabiriminde (Wolpaw vd., 2003) sensör-motor korteksindeki mu ve beta frekansındaki genlikleri kullanarak bilgisayarda cursor (imleç) hareketini oluşturmayı başarmıştır (Bkz. Mason vd., 2003). Ara birime alışık kişiler bir boyutlu veya iki boyutlu hareket oluşturabilme imkanına sahip olmuştur. Makalenin asıl amaçladığı kötürüm kişilerin ihtiyaçlarını karşılayabilecek yeni bir çeşit haberleşme imkanının sağlanabilmesidir. Bu durum bir veya iki EEG özelliğinden (farklı elektrot pozisyonlarından elde edilmiş spektral bantlar) çıkarılmış ampirik lineer fonksiyonlar sayesinde saniyede 10 kez pozisyon belirlenebilmesiyle yapılmaktadır. Makalede hangi metodun örüntü tanımada daha iyi sonuç verdiği, EEG özellik sayısının performansı nasıl etkilediği gibi sorulara cevap arandığı gözlenmiştir. Tüm metodların optimal özellik sayısı 10-20 olarak belirlenmişken, iki boyutlu lineer metodun tek boyutlu olanından bir miktar daha fazla performans verdiği anlaşılmıştır.

Lineer ve no-linear EEG BCI yöntemleri üzerinde ayrıntılı bilgi için (Müller vd., 2003) kaynağına göz atılabilir.

Schalk vd., (2004) yayınlarında ileri derecede engelli şahısların çevreleri ile etkileşmesine yardımcı olacak BCI sistemlerin çeşitli laboratuvarlar tarafından geliştirildiğini ama bu atılımların ortak bir platformda toplanamadığını belirtmiştir. Bunun sebebi, uygulamaların çoğunun birbiri ile uyumlu bir alt yapıya sahip olmamasında yatmaktadır. Böyle bir problemin çözümü olarak araştırma ve geliştirme platformu BCI2000 başlığı altında bir sistem yapılmıştır. Beyinsel sinyallerin incelenmesi için olabileceği gibi daha karmaşık kontrol kriterlerinin sağlanması için de uygulama sahası bulacağı belirtilmeye çalışılmıştır. Kullanılan sinyal analiz cihazları, operasyon ve çıkış protokolleri hali hazırda mevcut olan API'ler yardımı ile her türlü uygulamayı sağlayacak alt yapıya sahip olduğu izlenimini vermektedir.

Curan, Sykacek ve ekibi (Curran vd., 2003) makalesinde BCI uygulamalarında kullanılacak birbirinden farklı düşünsel faaliyetler arasından en uygun iki tanesini çıkarmayı amaçlamıştır. EEG aktivite farkının maksimum olduğu faaliyetlerin bilgi taşıma kapasitesinin de yüksek olacağı düşünülmüştür. Dört kanal üzerinden kayıt edilen EEG aktiviteleri sırasıyla, bilinen bir çevreyi düşünmek (deneğin kendi odasındaki detayları düşünmesi sağlanmıştır), bilinen bir şarkının hatırlanması, sağ ve sol elin açılıp kapanmasının düşünülmesi (açılıp kapandığında hissedilenler de düşüncelere katılmakta ama eller hareket ettirilmemektedir) olarak sayılabilir. AR modeli oluşturulduktan sonra mantıksal geriye dönüşüm (logistic regression) yöntemi ve no-linear jeneratif bir çıkarım işlemi yapılmıştır. Model çıkarımı esnasında dikkati çeken temel unsur, meydana gelen AR katsayılarının yansıma katsayılarının kullanılmasıdır. Zira AR katsayıları kendi aralarında fazlasıyla yakın ilişkilidir. Datanın alternatif bir sunumu olarak bu katsayıların lineer bir dönüşümü gösterilebilir. Yansıma katsayılarıyla sonuçlanacak bu dönüşüm işlemi (Pardey vd., 1996) kaynağında açıklanmıştır. En yüksek verim bilinen bir çevreyi düşünmek ile bilinen bir parçanın hatırlanması başlığı altında tanımlanmış olan düşünsel faaliyetlerde elde edilmiştir (sırasıyla 74%, 69%).

Lo ve Wu makalelerinde (Lo vd., 2004), EEG BCI çalışmaları için önemli olan bir diğer faktöre dikkat çekmiştir: İnsan faktörü. Şahısların, EEG spektral yayılımlarını kontrol etmek suretiyle BCI sistemlerini daha iyi maniple edeceği anlatılmaya çalışılmıştır. Zen meditasyonu yapan kişilerden elde edilen ERD (event related desynchronization, konu ilişkili desenkronizasyon) fenomenine yoğunlaşarak, şahsın yapması istenen düşünsel aktivite üzerindeki ilgi odağının kontrol üzerinde ciddi bir fark yarattığı görülmüştür. Bahsi geçen

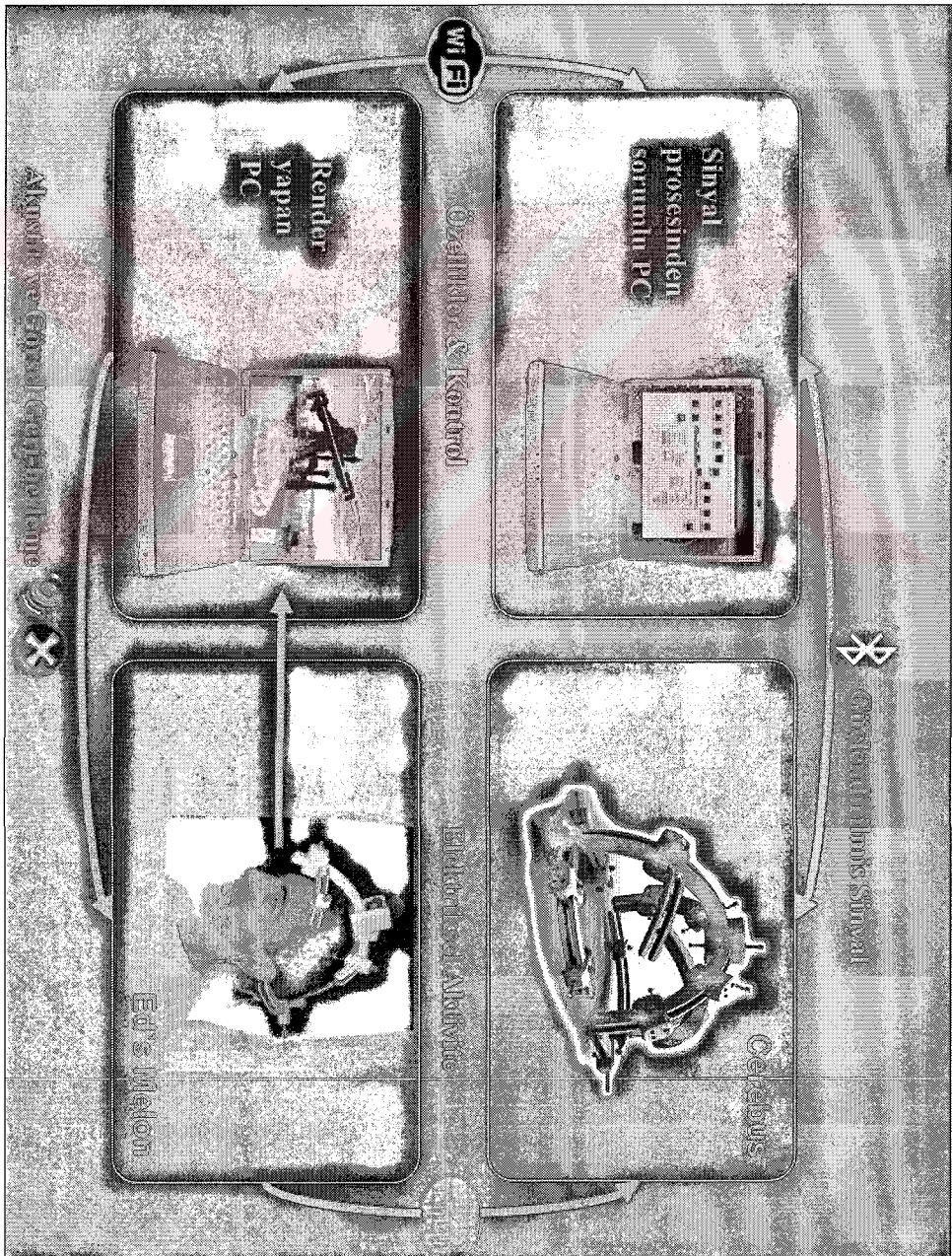
ERD, ilgilenilen frekans bandı içerisindeki yüzdesel güç miktarı artışını (referans değerine nazaran) göstermektedir. Sağ ve sol d şünsel el hareketi iin yapılan alıřmalarda bu baėıl artımın Zen meditasyonu yapan kiřilerde yapmayanlara kıyasla istenen deėerlere daha iyi oturduėu g zlenmiřtir. Edinilen sonuların veriminin geri besleme sistemi olduėunda daha y ksek olacaėı d ř n lmektedir.

Bahsi geecek son alıřma [13], EEG'nin bilgisayar uygulamalarında kullanılabileceėine dair kanıt niteliğindedir. Dublin  niversitesi b nyesinde ufak bir proje olarak bařlayan '*Mind Balance*' (akli denge) bař mimarı Robert Burke' n deyimiyle MIT Medya Avrupa Laboratuvarına tařmıřtır. Bir sonraki sayfada program ii g r nt s ne yer verilmiř olan sanal karakter 'Mawg', uzun gergin bir ipin  zerinde y r mek suretiyle ařaėıya d řmeden karřı tarafa gemeye alıřmaktadır. Bu oyunda kontrol ne bir joystickle, ne klavye ile, ne de fare ile saėlanmaktadır. Kontrol sisteminin dayandırıldıėı nokta insan EEG'sidir. Bařın osipital lobundan  l m yapılırken ekranın saė ve sol k řesinde yer alan kırpıřan topuklardan birine konsantre olunduėunda kırpıřma frekansıyla aynı frekansta olan beyinsel bir elektrik aktivite artımı aracılıėıyla saėa mı yoksa sola mı yoėunlařıldıėı tahmin edilebilmektedir. Bu fark edilebilen paracıklara VEP (Visually Evoked Potentials, G rsel Uyarım Potansiyelleri) adı verilmektedir. Basite ifade etmek gerekirse, Mawg saėa doėru kayarsa, kullanıcı onu sola eėmek iin ekranın sol yanında kırpıřan parlak topuėa bakacaktır.



řekil 1.3 '*Mind Balance*' yazılımının ekran g r nt s  [13]

EEG'deki bu anlık elektriksel deęişim bir VEP olarak algılanacak, akabinde tek boyutlu bir analog kontrol eksenine indirgenecek, bu sayede sanal yaratığın aőađıya dűőmesine mani olunacaktır. Teknik detaylara gűz atıldıđında C# lisansı ve grafik kűtűphaneleri aracılıđıyla alt yapı oluőturulmuőtur. Bahsi geen topukların saniyede en az 60 kare oynayacak őekilde gűsterilmesi gerekmektedir. Uygun olan 100 kare/s olarak seilmiőtir. Sistem iki ayrı laptopta alıőtırılmaktadır. Bir bilgisayar veri tanıma iőlemini gerekleőtirirken diđerisi grafik motoru alıőtırmaktadır. Oyunu oynamadan nce yaklaőık 45 saniye boyunca yazılım akustik geribesleme aracılıđıyla kullanıcının 'dűőnsel sűknet' (baseline) durumunu đrenmekte, bu esnada kullanıcı da iyi odaklanmış bir bakışın nasıl olduđunu idrak etmektedir. őekil 1.4'de bu sistem gűsterilmiőtir.



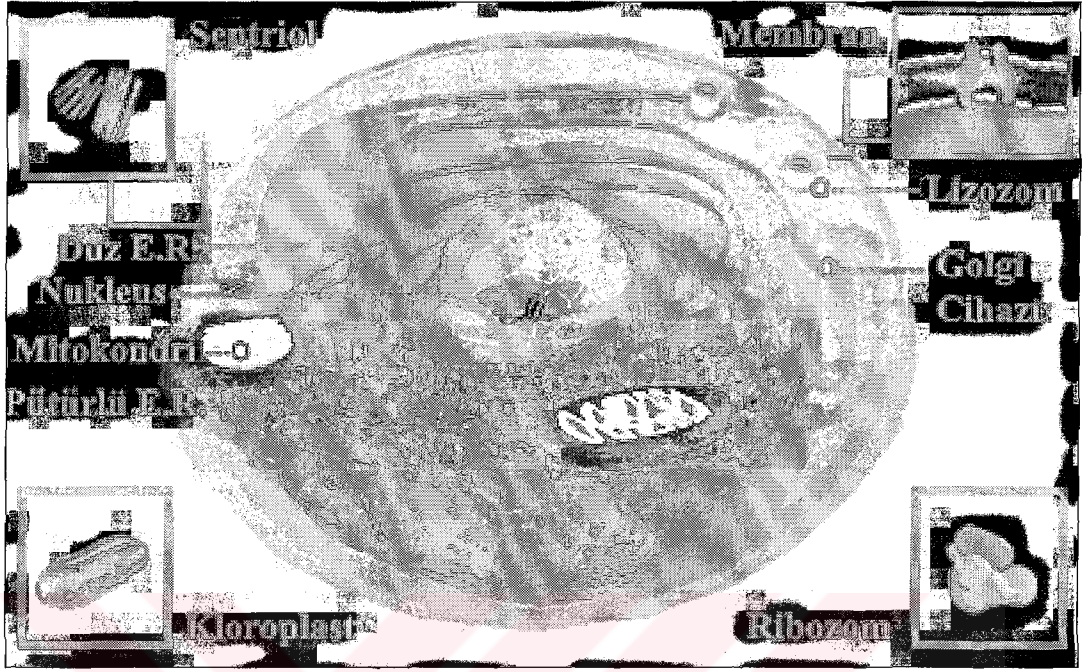
őekil 1.4 'Mind Balance' sisteminin arabirimi [13]

Biyolojimizin, insan neslinin doęuşundan beri başa çıkabildięi tanımlama becerisini eęer anlamak istiyorsak öncelikle yönelmemiz gereken yer gene bizim kendi doęamız, yani hücresel yapımız olacaktır.



2. HÜCRE VE SİNİR SİSTEMİ

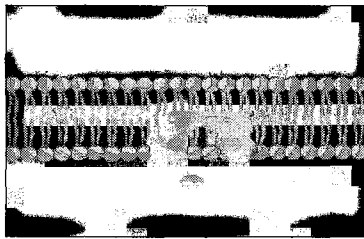
Tüm canlıların ortak noktası bünyelerinde yapıtaşı olarak hücreleri barındırmasıdır. Hücreler belli başlı üç bölümden oluşmaktadır: Hücre membranı, sitoplazma ve çekirdek.



Şekil 2.1 Hücrenin yapı taşları [2]

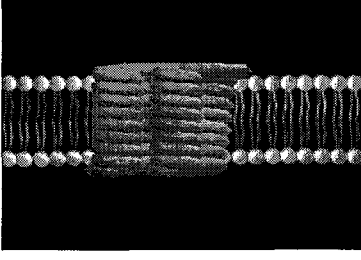
Hücre membranı, sitoplazmayı tamamen saran neredeyse görülemeyecek kadar ince bir yapıya sahiptir. Ayrıca endoplazmik retikulumu çekirdek membranına bağlamakla da görevlidir [2].

Hücre membranının içinde yüzen değişik tipte proteinler mevcuttur. Çoğunlukla globular protein olan bu yapılar önceden belli olan bir formasyonda değil de fosfolipit katmanında asılı kalacak şekilde dururlar. Yapısal olarak bu proteinleri genellikle üç kategoride sıralayabiliriz:



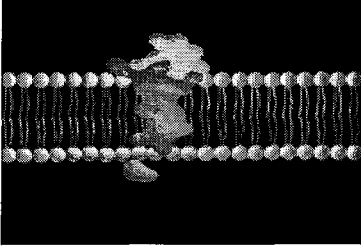
Taşıma ve difüzyondan sorumlu 'Taşıyıcı Proteinler'

Şekil 2.2 Taşıyıcı proteinler [2]



Hücreyi diğer hücelere tanımlayan ‘İşaretsel Proteinler’

Şekil 2.3 İşaretsel proteinler [2]

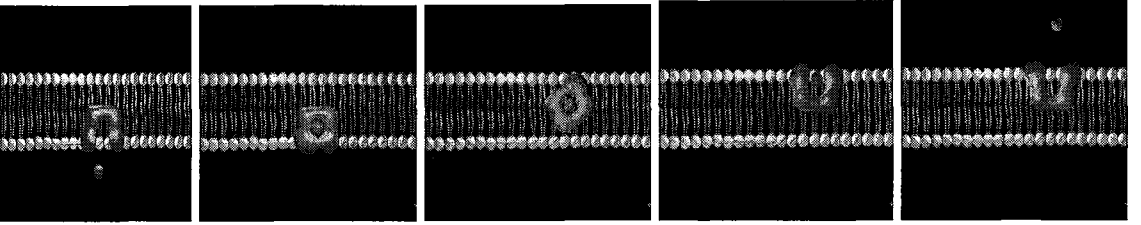


Gelen komutları almaya yarayan ‘Reseptör Proteinler’

Şekil 2.4 Reseptör proteinler [2]

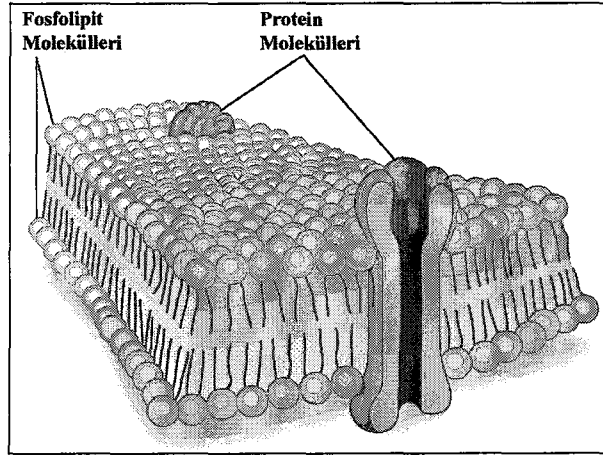
Hücre membranının üstlenmiş olduğu görev yukarıda bahsi geçen protein molekülleri etrafında döner. Bilinen genel işlevleri, reseptör proteinler yardımı ile iki hücre arasında haberleşmeyi sağlamak, taşıyıcı proteinler vasıtasıyla hücre içine giren çıkan maddelerin regulasyonu ve kontrolünü gerçekleştirmek, işaretsel proteinlerle ise hücrenin tipini belirtmek olarak sıralanabilir.

Taşıyıcı proteinler iki forma sahiptir. Bunlardan ilki Carrier (Taşıyıcı) Proteinler olarak adlandırılan belirli moleküllerle arabirimi oluşturan yapıdır. Spesifik moleküllerin her seferde bir tanesiyle bileşik oluşturup membrandan geçirilerek diğer tarafa aktarılmasını sağlar. Aşağıda buna ilişkin şekiller görülmektedir.



Şekil 2.5 Taşıyıcı proteinler [2]

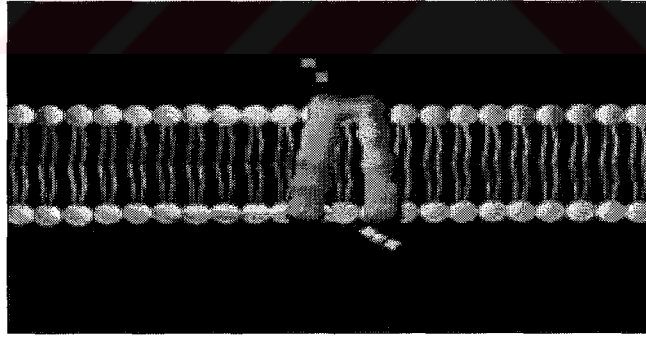
Bir diğer taşıyıcı protein ise kanal proteinleridir. Membranda bir çeşit por oluşturarak moleküllerin geçmesine izin verirler. Temsili resim Şekil 2.6’da mevcuttur.



Şekil 2.6 Hücre membranı ve kanal proteinleri [15]

Kanal proteinleri, taşıyıcı proteinlerinin aksine enerjiye ihtiyaç duymadan moleküllerin hücrenin içine veya dışına difüzyon ile ulaşmasına izin verirler. Difüzyon bilindiği gibi, moleküllerin yarı geçirgen bir zardan nüfuz etmek suretiyle yüksek konsantrasyondan alçak konsantrasyona taşınması anlamına gelir.

Simportlar(symports) da difüzyon yapmasına rağmen biraz farklı şekilde işlevini yerine getirir. Simportlarda bir molekül içeri girerken bir başka molekülü de beraberinde sürükler. (Sodyum'a takılan glikoz molekülleri de içeri girerler.) Aşağıda olayı betimleyen şekil mevcuttur.



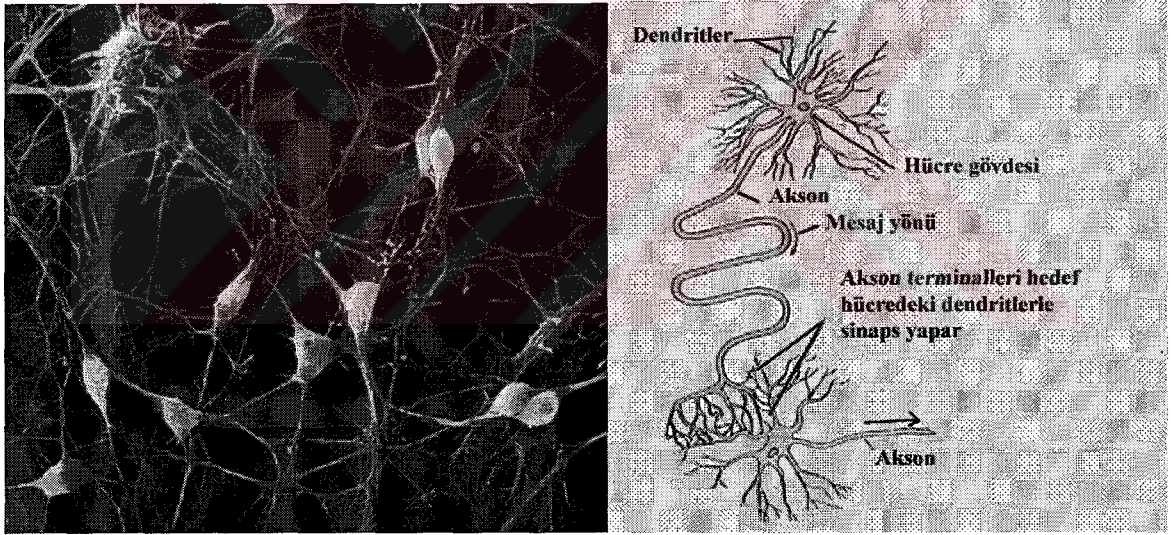
Şekil 2.7 Simportlar [2]

Bazı proteinler hücre içindeki ATP'den enerji kullanarak alçak konsantrasyondan yüksek konsantrasyona (difüzyona tam ters mahiyette) molekül sürükleyebilir. En bilindik örneği Na(Sodyum)/K(Potasyum) pompasıdır. Burada fosfatın (üstteki şekilde kırmızıyla gösterilen) enerjisi kullanılarak sodyum atomları ile potasyum atomlarının yer değiştirmesi sağlanır.

2.1. Sinir Sisteminin Ana Elemanları

Sinir sistemini oluşturan temel birim sinir dokusudur [3]. Sinir dokusu iki tür hücreden oluşmuştur: Nöronlar ve glial hücreler. Nöronlar sinirsel sinyalleri iletirken, glial hücreler de nöronlara doğrudan temas ederek onları sarar, desteklenmesini ve çeşitli fizyolojik etkilerden korunmasını sağlar. Bilinmesi gereken diğer bir konu ise glial hücrelerin iletken olmadığıdır. Böylece, nöronlar tarafından iletilmesi gereken sinyale etki etmezler.

Nöron, sinir sisteminin ana ünitesidir. Sadece insan beyninde ortalama yüz milyon nöron olduğu hesaplanmıştır. Boyut ve şekil itibariyle varyasyon göstermesine rağmen, nöronlar üç parçadan oluşmaktadır: Dendritler, hücre gövdesi ve aksonlar. Dendritler kısa ama, oldukça dallanmış bir yapıya sahip sinirsel sinyalleri hücre gövdesine taşımakla görevli uzantılardır. Hücre gövdesi çekirdek, mitokondri ve tipik bir ökaryot hücresinde görülmesi beklenen organellerin bulunduğu kısımdır. Aksonlar ise, nörondan alınan bilgiyi hücre gövdesinden alıp götüren uzantılar olarak ifade edilebilir. Şekil 2.8’de gerçek bir sinir ağı ve temsili resimleri görülmektedir.



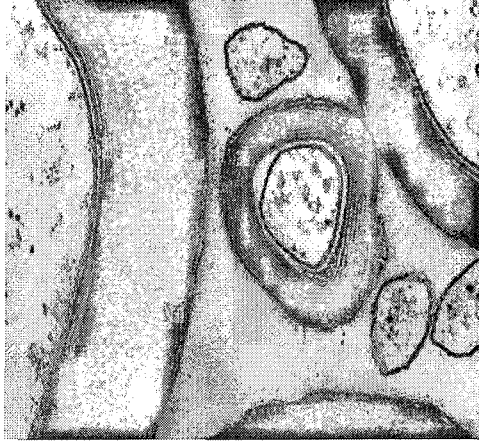
Şekil 2.8 Nöronlar ve sinir ağı [3]

Üç tip nörona rastlamak mümkündür: Sensör nöronlar, motor nöronlar, inter-nöronlar.

Sensör nöronlar adından da anlaşılacağı gibi, algılayıcı reseptörlerden gelen mesajları merkezi sinir sistemine ulaştıran, uzun dendritli, kısa aksonlu hücrelerdir. Böylece, çevre bilgisinin, beynin prosesleri yürüten kısmına ve omuriliğe varması sağlanmış olur. Efferant nöronlar olarak da anılırlar.

Motor nöronlar ise merkezi sinir sisteminden gelen mesajları kaslara veya salgı organlarına iletmekle görevli uzun aksonlu, kısa dendritli sinir hücreleridir. Bilgiyi inter-nöronlardan alarak cevap üreten efektör hücelere verirler. Diğer bir adı da efektör nöronlardır.

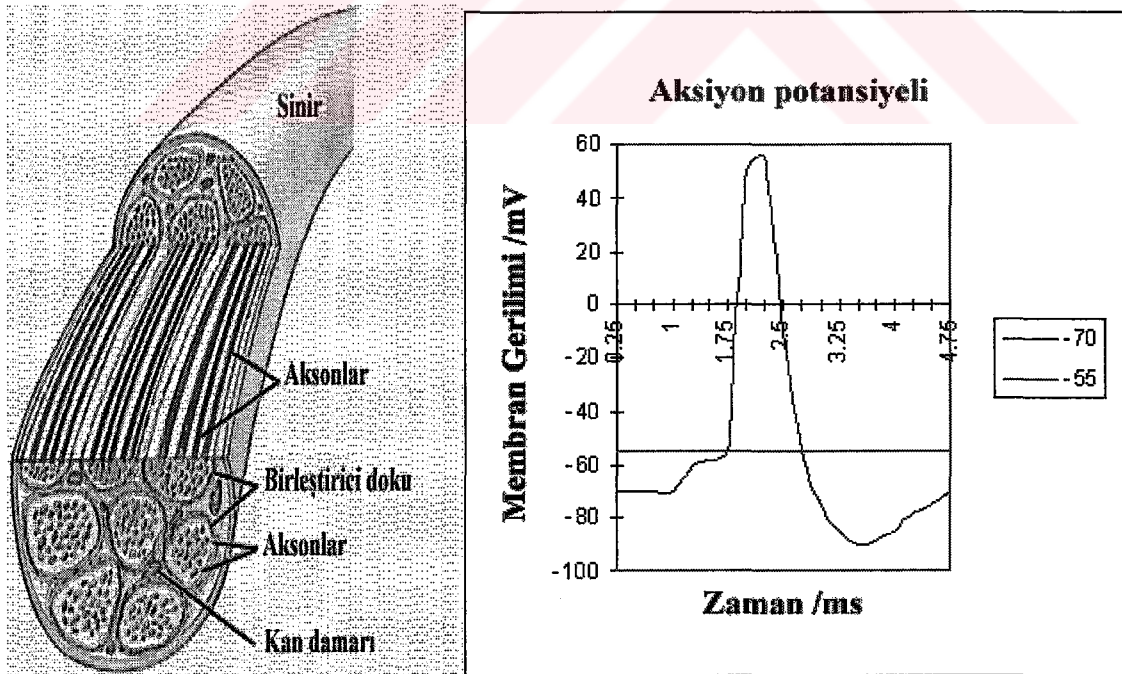
İnter-nöronlar ise sadece merkezi sinir sisteminde bulunur. Bir veya daha çok sensör nörondan gelen bilgiyi motor nöronlara iletir. Konnektör nöronlar olarak da anılırlar.



Bazı aksonlar myelin denilen bir kılıfla kaplıdır. Myelin, özel bir glial hücre olan Schwann hücresinin plasma membranından oluşmuştur. Schwann hücreleri, nöronlar için destek, besin ve servis faaliyetlerini sağlamakla yükümlüdür. Schwann hücreleri arasında kalan boşluğa Ranvier Nodu adı verilir.

Şekil 2.9 Myelin kılıfı ve ranvier nodu [3]

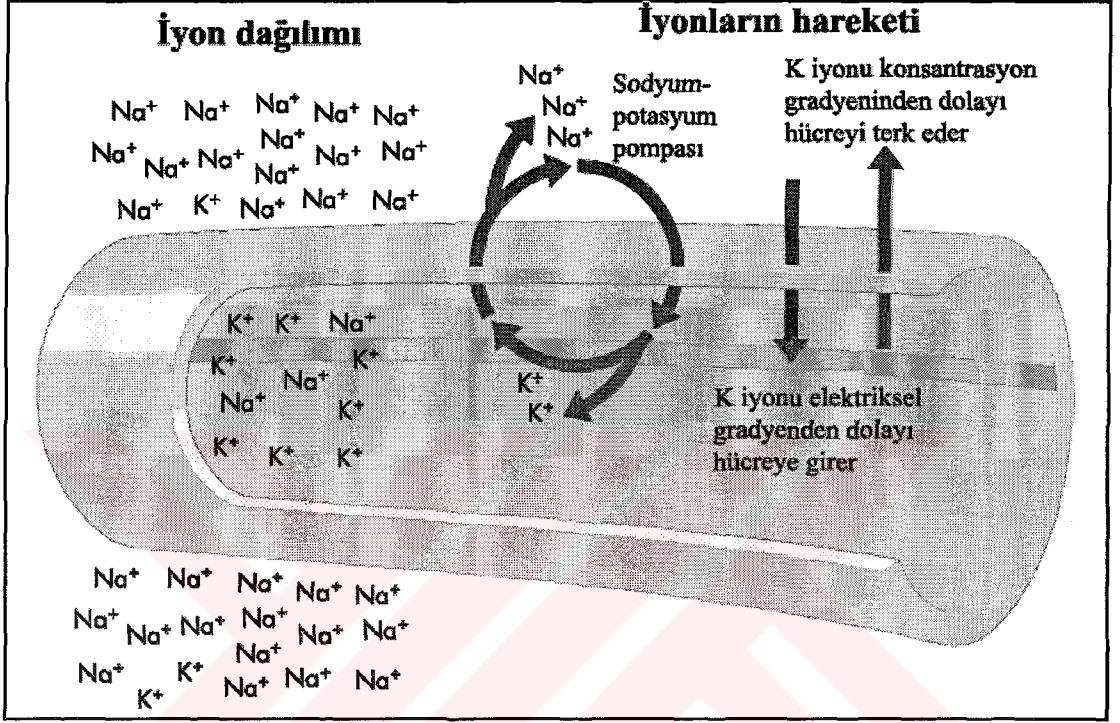
Nöronun sinyal üretme hattında mesajı noddan noda atlatarak, akson yüzeyinde gideceğinin yaklaşık olarak yüz katı hızında sürebilmesine imkan verir. Böylece, beynimizin başparmağımızla haberleşmesi saniyenin binde biri gibi bir sürede olup biter.



Şekil 2.10 Sinir kesiti ve aksiyon potansiyeli [3]

Göz atılması gereken diğer bir konu ise sinirsel mesajın nasıl üretildiğidir. Diğer hücelerde olduğu gibi nöronlarda da plasma membranının her iki tarafı farklı iyon konsantrasyonuna

sahiptir. Membranın dışı pozitif yüklü iken, iç kısım negatif karakterdedir. Bu yük farkı 'dinlenme potansiyeli' olarak bilinir. Birimi milivolt'tur. Hücre membranından iyonların geçiş yapması elektriksel yükün hücrede ilerlemesine neden olur. Hücrenin dinlenme halinde bu bir potansiyel farkı olarak gözlenecektir. Adı geçen dinlenme potansiyeli -65mV (milivolt) olarak ölçülmüştür. Dinlenme potansiyeli sodyum(Na), potasyum(K) gibi pozitif yüklü iyonlar ile sitoplazmadaki negatif yüklü iyonlar arasındaki yük farkından doğar [4].



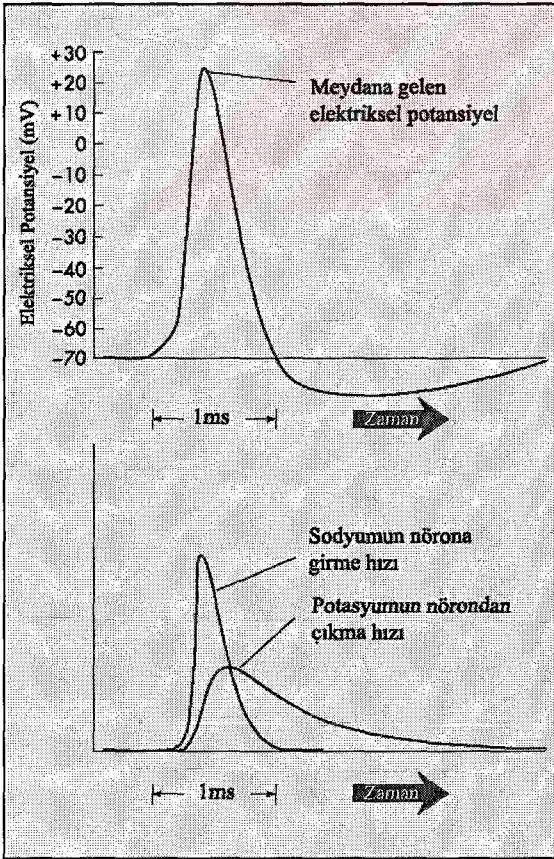
Şekil 2.11 Sodyum potasyum pompası [15]

Sodyum iyonları membran dışında daha bol bulunurken, potasyum iyonları ise membran içinde fazlalığa sahiptir. Bu dengesizlik iyonların sodyum potasyum pompası ile aktif taşınmasıyla sağlanır. Sodyum potasyum pompasının bu işlemi iyonların konsantrasyon gradyenlerine ters bir mahiyette yapıyor olması gözden kaçmamalıdır.

Bahsi geçen durumu sözle ifade etmek gerekirse şöyle diyebiliriz: Önce hücre içerisindeki üç sodyum iyonu taşıyıcı proteine (Symporta) tutunur. Sonra ATP'den bir fosfat grubu taşıyıcı proteine yapışarak (ATP hidrolizi ile) proteinin şekil değiştirmesini sağlar. Böylelikle sodyum iyonları hücre dışına çıkar. Hücre dışındaki iki potasyum iyonu ise taşıyıcı proteine tutunur. Fosfatın proteinden kopması (defosforizasyon) ile protein orijinal formuna dönerek potasyum iyonlarının hücre içine gönderilmesini sağlar. Bu çalışma stili aktif taşınmaya bir örnek niteliğindedir [5].

Membranın deęişmiş polaritesi (aksiyon potansiyeli) sinirsel darbenin membran boyunca hareket etmesine sebep olur. Aksiyon potansiyeli, membranın elektriksel potansiyelinin bir kaç milisaniyeliğine terslenmesi anlamına gelir. Sodyum ve potasyum iyon kanalları açılarak membrandan iyonların geçmesine izin verir. Sodyum ve potasyum iyonları membran iyon akışını ayarlamakla görevli olan açılıp kapanabilen protein kanal kapılarından geçerler. Önce sodyum geçiş yapar. Membran negatif potansiyelinin maksimum noktasına ulaşıldığında, potasyum kanalları açılarak potasyum iyonlarının membran dışına çıkmasına izin verilir. İkinci olarak potasyum geçecektir. Böylelikle iyon yayılımında bir deęişiklik olur. Bu deęişim sodyum-potasyum pompasının sürekli çalışması sağlanarak daima reset edilir. Sonunda yeterli miktarda potasyum iyonu dışarıya çıkararak dinlenme potansiyeline geri dönülür [6].

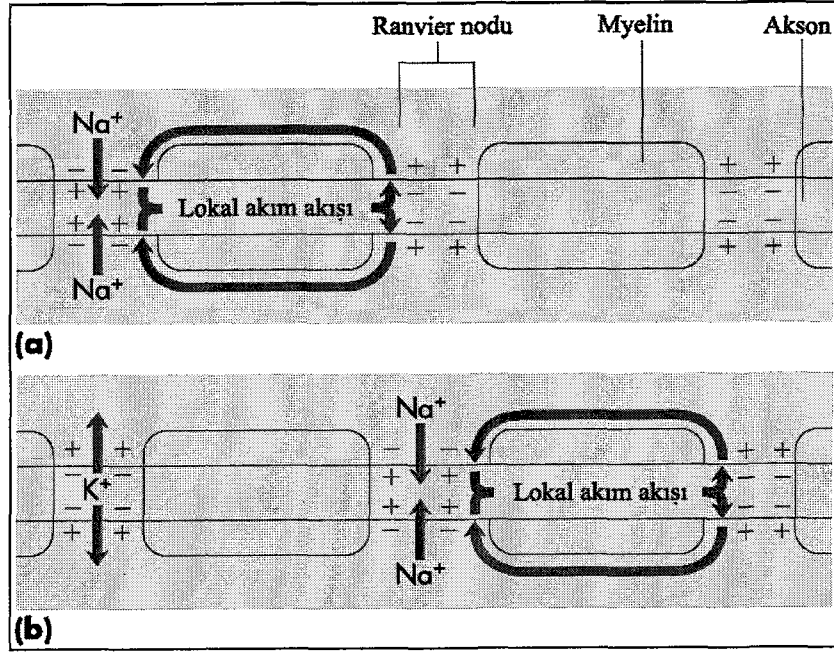
Aksiyon potansiyeli, membranın bir noktasında başlayıp komşu bölgelere yayılarak, mesajın, hücre membranı boyunca iletilmesini mümkün kılar. Aksiyon potansiyelinin geçmesinden sonra bir çeşit kendini toparlama periyodu (refractory period) gelir ki bu süre zarfında hücre stimule edilemez. Bu sayede mesajın geriye iletilmesi önlenmiş olur. Mesaj daimi olarak ileri götürülür [3].



Aksiyon potansiyelinin evrelerini sıralamak gerekirse:

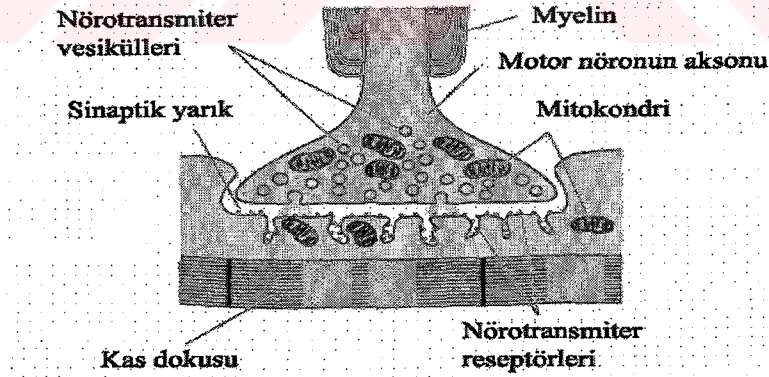
1. Dinlenme anında membran dışı içerden daha pozitifdir.
2. Sodyum, hücre içine girerek aksiyon potansiyelini oluşturur. Membran içi dışarıdan daha pozitif olur.
3. Potasyum iyonları hücreden dışarı akarak dinlenme potansiyelinin net yük durumuna dönülür.
4. Sodyum iyonları hücreden dışarı, potasyum iyonları ise içeri pompalanır. Böylece iyonların orijinal yayılımına dönülmüş olur.

Şekil 2.12 Aksiyon potansiyeli [15]

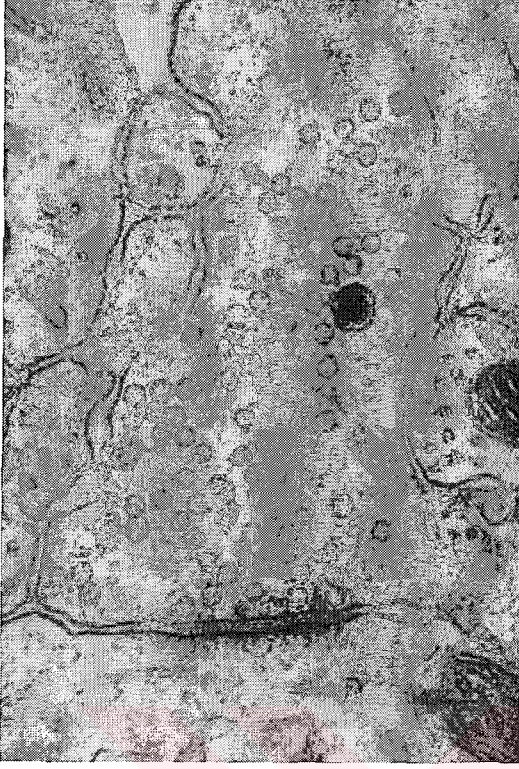


Şekil 2.13 Aksiyon potansiyelinin myelinli aksondaki hareketi [15]

İki nöral hücrenin birleştiği yere sinaps denilir. Sinirsel mesajlar bilindiği gibi nöronlarda elektriksel aksiyon potansiyelleri olarak yol alırlar. İki sinir hücresi arasındaki boşluk sinaptik yarık olarak bilinir. Bu yarığı geçmek için nörotransmitter denilen maddelerin aktif hale gelmesi gerekir. Nörotransmitterler sinaptik bölmelerde saklı bir şekilde aksonun tam ucunda bulunur.



Şekil 2.14 Sinaps ve nörotransmitterler [3]



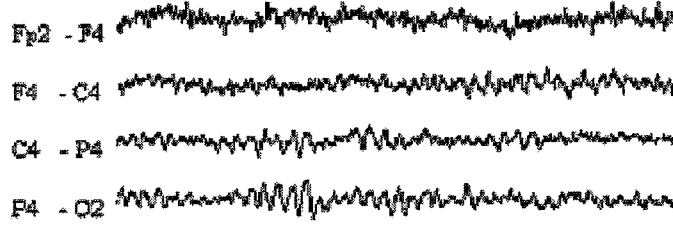
Şekil 2.15 Sinaptik yarık [3]

Aksiyon potansiyelinin ulaşması ile bölmeciklerin bir kısmı aksonun ucuna hareket eder ve içindekileri sinaptik yarığa boşaltır. Çıkan nörotransmitterler yarıktan geçerek diğer hücrenin membranındaki reseptörlere tutunurlar. Böylelikle o hücredeki iyon kanalları açılır. Bazı nörotransmitterler aksiyon potansiyeline sebep olurken, diğer bir kısmı inhibe edici (yatıştırıcı) etkide bulunur. Nörotransmitterler küçük moleküller ya da hormon olarak karşımıza çıkar. Nörotransmitterin aksiyonu için gereken süre 0.5–1 milisaniye arasındadır. Bu maddeler ya spesifik enzimler ile sinaptik yarıқта yok edilir, ya yarıktan dışarı difüzyona uğrar ya da hücre tarafından geri emilir.

Otuzdan fazla organik molekül nörotransmitter olarak davranır. Yarıktan geçen nörotransmitterler diğer hücrenin reseptör moleküllerine yapışarak mesajın hücre membranından geçmesi için baskıda bulunur. Asetilkolin ve norepinefrin değişik türde iki nörotransmitterdir. Kısa süreliğine aktif olma eğiliminde olan nörotransmitterler görevlerini tamamlayınca sinaptik yarıktaki enzimler tarafından kullanım dışı bırakılarak, aksonun içine geri çekilip yeniden dönüşüm döngüsüne sürüklenirler.

3. ELEKTROENSEFALOGRAM VE İŞARETLERİN ÖLÇÜMÜ

Elektroensefalogram (EEG), beyinde ortaya çıkan sinirsel faaliyetleri tasvir etmekle yükümlü biyoelektriksel işaretlerdir. Yapı olarak karmaşık bir değişim gösteren EEG potansiyelleri, serebral korteksten ve alttaki bir çok noktadan gelen işaretlerin toplamı niteliğindedir [7].



Şekil 3.1 Elektroensefalogram kaydı [7]

İlk EEG kaydı 1929 yılında Hans Berger tarafından gerçekleştirilmiş olmasına rağmen, bu konuyla ilgili olarak kobay hayvanlar üzerinde yapılan çalışmalar 1870'li yıllara dayanmaktadır.

Elde edilmiş olan dalga şekillerinin beyin yüzeyindeki (korteks) aktiviteyi yansıttığı düşünülmektedir. Bu aktivitenin aslında beynin korteks altındaki katmanlarındaki elektriksel hareketlerinden kaynaklandığı bulunmuştur.

Beynimizdeki sinir hücreleri aksiyon potansiyeli adı verilen sinyaller meydana getirir. Bu sinyaller bir sinir hücresinden diğerine aradaki sinaps denilen boşluğu aşarak nöro-transmitterler aracılığıyla taşınır. İki tip nöro-transmitterden biri, aksiyon potansiyelinin komşu hücreye geçmesine yardım ederken, diğeri bu hareketi engelleyici bir mahiyette etki eder.

EEG aktivitesi oldukça küçük bir genliğe sahiptir (μV). İlgilenilen temel frekanslar yaklaşık olarak 30 Hertz (Hz) civarına kadar değişim gösterir.

EEG dalgalarını beş farklı frekans aralığında sıralamak mümkündür [8]:

- Gama aktivitesi (22 Hz - 30 Hz)
- Beta aktivitesi (12 Hz - 22 Hz)
- Alfa aktivitesi (8 Hz - 12 Hz)
- Teta aktivitesi (4 Hz - 7 Hz)
- Delta aktivitesi (0.5 Hz – 3.5 Hz)

Bu dalga şekillerinden alfa aktivitesi, uyanık ve normal kimselerde başın arka kısmındaki kanallardan elde edilen dalgalardır. Nispeten simetrik bir yapıda olan alfa dalgaları

40 μ V–100 μ V aralığında genliğe sahiptir. Sadece gözler kapalıyken görünürler. Gözler açıldığında ya genlikleri oldukça azalır ya da hepten yok olur. Uyanık kişi dikkatini özel bir işe yöneltecek olursa, alfa dalgaları yerine daha yüksek frekanslı ama daha düşük genlikli beta dalgaları gözlenir. Şekil 3.2’de tipik bir alfa dalgası görülmektedir.



Şekil 3.2 Alfa dalgası [8]

Beta aktivitesi, gözler açık veya kapalıyken görülen normal bir aktivitedir. Genellikle kafanın merkezinden veya frontal bölgesinden kaydedilen beta dalgaları, 25 Hz frekansında salınımına sahiptir. Nadir de olsa 50 Hz’e kadar uzandıkları da görülür. Beta dalgalarını Beta 1 ve Beta 2 diye ikiye ayırmak mümkündür. Beta 2 dalgalarının frekansı Beta 1 dalgalarının iki katıdır. Zihinsel aktivitenin artmasıyla alfa dalgalarında olduğu gibi yok olurlar ve yerlerini düşük genlikli asenkron sinyaller alırlar. Beta 2 dalgaları, merkezi sinir sisteminin kuvvetli bir şekilde uyarıldığı durumlarda ya da ruhsal gerginlik hallerinde sıklıkla görülmektedir. Şekil 3.3’te tipik bir beta aktivitesi görülmektedir.



Şekil 3.3 Beta dalgası [8]

Teta aktivitesi kişinin yaşına ve durumuna göre normal veya anormal olarak sınıflandırılabilir. Yetişkinlerde, eğer kişi uykulu ise, bu durum normaldir. Şayet yetişkin tümüyle uyanık ve tetikte ise, teta aktivitesi, kişideki beyin fonksiyonlarındaki bir bozukluğa delalet edebilir. Bazı hallerde duygusal gerginliğin ve düş kırıklığının habercisi niteliğinde de olabilir. Gençlerdeyse, beynin arka ve orta kısmından elde edilen temel dalganın teta aktivitesine sahip olması muhtemeldir. Şekil 3.4’te genlikleri 100 μ V’tan küçük bu dalgaların karakteristik yapısı görülmektedir.



Şekil 3.4 Teta dalgaları [8]

Delta aktivitesi yetişkin derin uykudayken görülmektedir. Eğer bu durum dışında gözlenirse beyinsel fonksiyon bozukluğuna işaret eder. Süt çocuklarında ise bu dalga formu hakimdir. Şekil 3.5'te bu dalga formu görülmektedir.



Şekil 3.5 Delta dalgaları [8]

Gama dalgaları olarak nitelendirilen beyin aktivitesi, bazı araştırmacılar tarafından Beta 2 dalgaları sınıfına sokulur. Uykunun tüm karakteristik belirtilerini taşırlar.

Bu temel dalgaların dışında, bazı hastalıklar veya anomalilerin sebep olduğu dalga formlarına rastlamak da mümkündür. Örneğin; darbe-dalga aktivitesi nöbet alameti olarak gözlenebilir. Epilepsi nöbeti geçirmiyor olmasına rağmen bir takım dengesizliklerin olması bu dalgaların meydana gelmesini sağlar. Şekil 3.6'da darbe-dalga aktivitesi görülmektedir.



Şekil 3.6 Darbe-dalga aktivitesi [8]

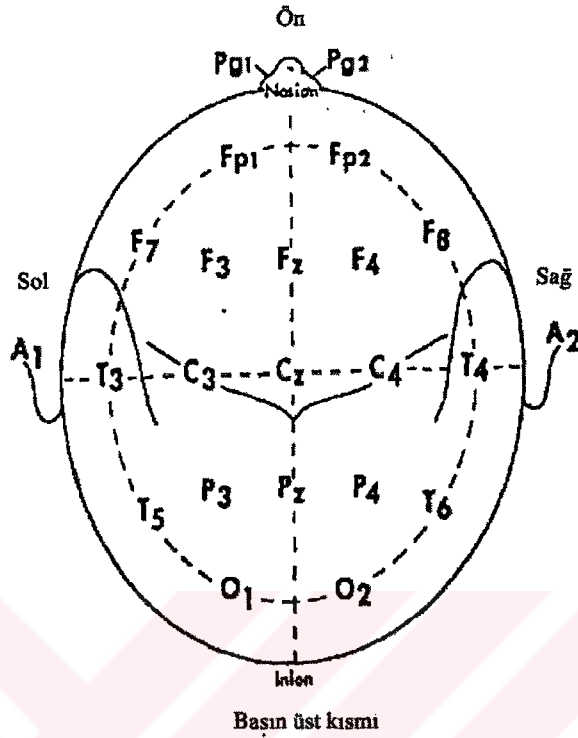
Yukarıda görülen temel dalga şekillerinin çeşitli kombinasyonlarının mevcut olması, EEG kayıtlarının incelenmesini ve yorumlanmasını zor kılar. Anormal olarak nitelendirdiğimiz özellikler durumlara özgü bir mahiyette görülmeyebilirler. Böyle anlar da daha farklı incelemelere gitmek gerekmektedir.

3.1. EEG İşaretlerinin Ölçümü

Kara kutu olarak nitelendirilen bir sistemi, kendinden ufak belli başlı parçalara ayıramayacağımız için, girişine bir sinyal uygulayıp elde edilecek çıkışı analiz ederiz. Bahsi geçen sistem beyinde de mevcuttur. Tam işleyişleri hakkında yeterli bilgiye sahip olamadığımız beynimizin parçalarının özelliklerini giriş bilgisi olarak nitelendireceğimiz duyular yardımı ile bulabiliriz. Bu duyular görsel, işitsel, duyumsal (deri), kokusal ve tadımsal olarak sıralanabilir. Beyin kara kutusunun çıkışı olarak da EEG gözlemlenecektir (Vinther, 2002).

EEG kaydı, kafanın belirli yerlerine yerleştirilmiş elektrotlar yardımı ile yapılır. Şekil 3.7'deki gibi montajı yapılan elektrotlardan kulaklara iliştirilmiş olanları toprak olarak seçilir. Her

elektrot bölgesi bir harf-numara ikilisiyle temsil edilir. Harfler başın hangi bölgesinde olduğunu (F:Frontal lob, C:Central Lob, T:Temporal Lob, P:Parietal Lob, O:Osipital Lob), çift numaralar başın sağ yanını, tek numaralar ise sol yanını temsil eder [7].



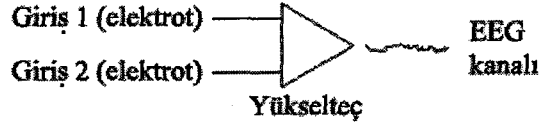
Şekil 3.7 10/20 elektrot pozisyonları [7]

Elektrot pozisyonları uluslararası 10/20 Sistemi kullanılarak yerleştirilmiştir. Bu yöntemde göre baş dört temel noktayla işaretlenmiştir: Nasion (burun), inion (başın arka kısmı), sol ve sağ preauricular (kulak). İki kulak arasındaki mesafe %10-20-20-20-10 olacak şekilde bölünerek elektrotların bir kısmı yerleştirilir. Kalanlar ise o eksene simetrik olarak bir daire yapacak şekilde konulur (Şekil 3.7) (Yazgan ve Korürek, 1996).

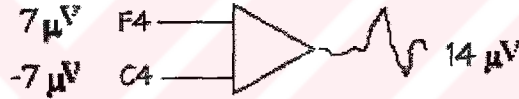
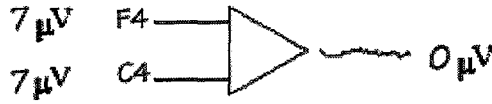
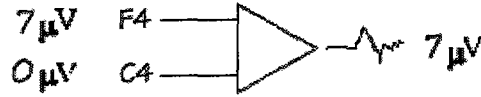
Birçok elektrot çeşidi olmasına rağmen, yaygın olarak düzgün bir temas yüzeyine sahip scalp disk tipi kullanılır. Elektrot temas direnci 10 kOhm'un altında olacak şekilde temas yüzeyi temizlendikten sonra, bir nevi iletken kremle elektrotun deriye temas etmesi sağlanır. Elektrotlar, paslanmaz çelik, teneke, altın ve gümüş klorür kaplamalı olabileceği gibi alternatif bir yöntem olarak bir tür başlığın içine önceden yerleştirilmiş bir şekilde de bulunabilir [7].

EEG makineleri bir diferansiyel kuvvetlendirici yardımı ile her kanalı oluşturur ve bir bakıma beyinsel aktiviteleri izler. Her kuvvetlendirici ucuna elektrotların bağlanacağı iki adet girişe sahiptir. Bilindiği gibi, diferansiyel kuvvetlendiriciler girişlerdeki iki sinyal arasındaki gerilim

farkını ölçerler. Elde edilen fark yükseltilerek EEG aktivitesini tanımlayan bir kanal olarak gözlemlenir [9].



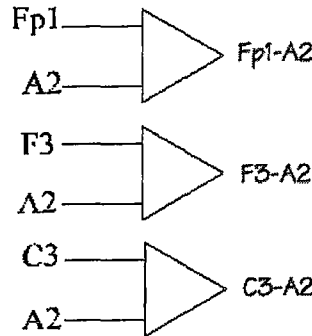
Şekil 3.8 Diferansiyel amplifikatör [9]



Şekil 3.9 Amplifikatör çalışma prensibi [9]

Elektrot ikililerinin EEG makinesinin işlemsel kuvvetlendiricilerine bağlanma şekilleri montaj tipini belirler. Üç tip bağlama mevcuttur: Ortak referans, ortalama referans ve bipolar.

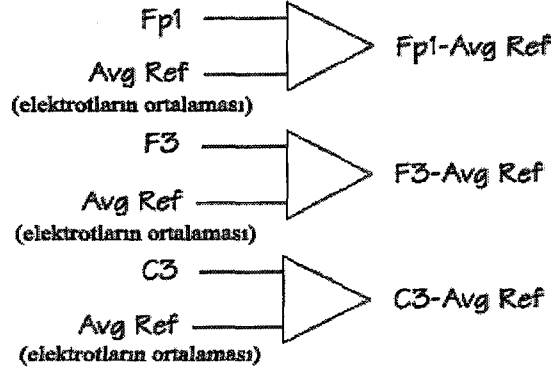
Ortak referans türetiminde (Şekil 3.10) her kuvvetlendirici başa bağlanmış olan bir elektrot ile referans elektrotu arasındaki farkı ölçer. Her kanal için aynı referans elektrotu kullanılır. Genellikle A1, A2 veya A1 ve A2 kısa devre edilerek referans olarak seçilir.



Şekil 3.10 Ortak referans türetim [9]

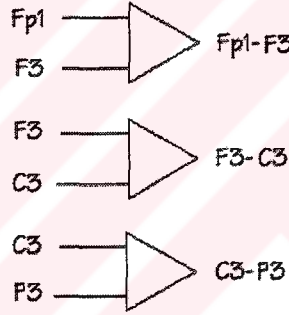
Ortalama referans türetiminde ise (Şekil 3.11) tüm elektrotlardaki aktivite ölçülür, toplanır ve ortalamaları alınarak büyük değerli bir dirençten geçirilip referans elektrotu olarak

kuvvetlendiricilerin ikinci girişine uygulanır. EEG sistemlerinin tümünde, kullanıcının hangi elektrotları bu işleme dahil edeceğini seçmesine izin verilmektedir.



Şekil 3.11 Ortalama referans türetme [9]

Bipolar türetme ise tüm elektrotları (başın önünden arkasına doğru), birinin ikinci girişi diğerinin birinci girişi olacak şekilde bağlamak suretiyle elde edilir. Boyuna olduğu gibi enine de bağlantı yönü olarak seçilebilir. Şekil 3.12’de bu bağlantı tipi gösterilmektedir.



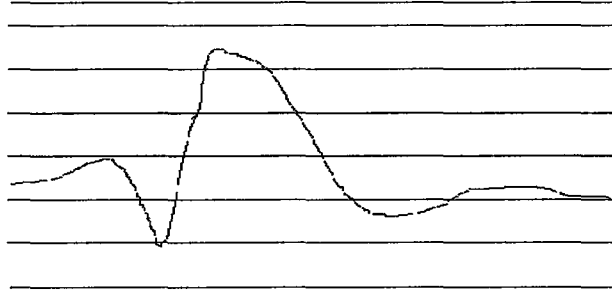
Şekil 3.12 Bipolar türetme [9]

Sinyallerin okunmasını gerçekleştirecek makine önceden bahsedilmiş olan kuvvetlendiricilerden, bir galvanometreden ve bir yazıcıdan oluşur. Galvanometre, manyetik alan içersinde sarım halinde bulunan bir iletkenidir. Yükseltilmiş olan sinyal bu sarımdan geçerek endüksiyon kuvveti meydana getirip salınım yapmaya başlar. Ucuna yerleştirilmiş olan bir yazıcı mekanizma yardımı ile bu yukarı ve aşağı salınımlar hareket eden kağıdın üzerine çizilir [10].

Analog bir EEG cihazında yükselteç çıkışı, yüksek veya alçak frekans filtre devreleri kullanılarak istenen EEG değer aralığında (bant genişliği) kalması sağlanır. Hassasiyet ise ekranda gösterilmekte olan aktivitenin boyutunu kontrol etmekle yükümlüdür. Örneğin; 10 μ V/mm hassaslıktan kastedilen 100 μ V genliğe sahip bir sinyalin, 1cm düşey sapmaya yol

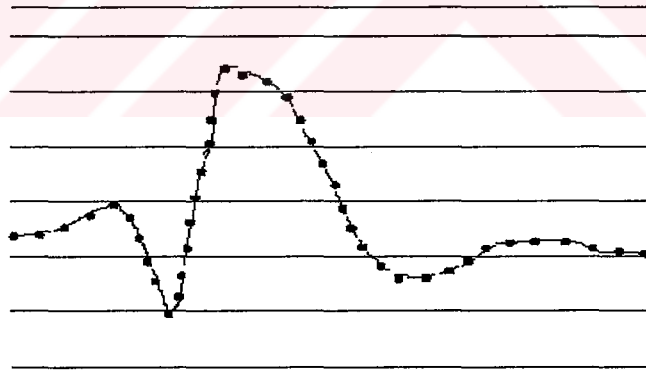
açacağıdır. Yazıcı çıkışında dalga formalarının görünüşünü etkileyen bir diğer faktör ise kağıdın kayma hızıdır.

Bir dijital EEG sisteminde ise dalga formu nümerik değerler dizisine çevrilerek işlenmektedir. Bu işlem “Analogdan-Dijitale-Dönüştürme” olarak bilinir (Şekil 3.13). Okunan sinyallerden elde edilen sayısal değerler, bilgisayarın hafıza birimine kaydedilmek suretiyle saklanıp, bir takım işlemlerden geçirilmesinin ardından ekranda bir dalga formunun gösterilmesinde kullanılabilir. Bu analog dijital dönüştürme sürecinde en önemli nokta, örnekleme hızının yeterli seviyede olmasıdır.



Şekil 3.13 Analog dalga formu [10]

Hertz ile ölçülen örnekleme hızı, saniyedeki örnekleme hızını temsil etmektedir. Minimum örnekleme hızı, ilgilenilen en yüksek frekans değerinin 2.5 katı olarak kabul edilir ki bu sayı, günümüzdeki bir çok EEG makinesinde 240 Hz olarak belirlenmiştir (Şekil 3.14).

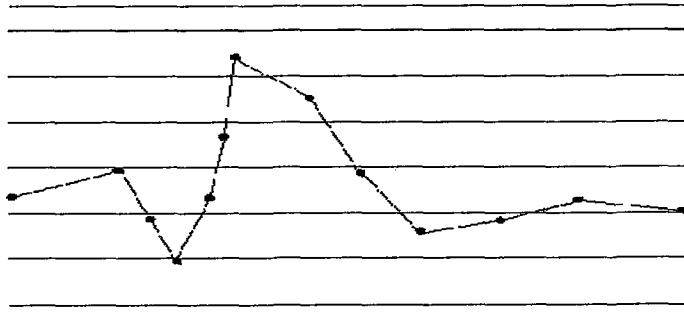


Şekil 3.14 240 Hz’de örnekleme [10]

Bazı hallerde 240 Hz örnekleme hızı bile yetersiz kalmaktadır. Böyle bir durum beyin yüzeyinden doğrudan ölçüm yapıldığında karşılaşılmaktadır. Bu yüzden bazı dijital EEG sistemleri 480 Hz örnekleme değerine de destek vermektedir.

Eğer örnekleme hızı idealin altında kalacak olursa bu, analog sinyal değerlerinin yeterince iyi incelenemediğine ve dolayısıyla bilgisayar ekranında gösterilecek olan dalganın orijinaline benzemeyeceğine işaret eder. Şekil 3.15’te 50 Hz örnekleme sonucunda elde edilen bir

dalga formu gösterilmektedir. Bu şekilde ilk haliyle çok az ilişkisi olduğu açıkça görülmektedir.



Şekil 3.15 50 Hz'de örnekleme [10]

Örnekleme performansını etkileyen faktörlerden biri de örnekleme kaymasıdır. Örnekleme kayması kanalların aynı anda okunamamasından kaynaklanır. Ardı ardına okunan kanallar arasındaki gecikmeler elde edilen bilginin eksik veya bozuk olmasına sebep olur. Gecikmeyi giderici bir teknik olan burst modu örnekleme işlemi, ilgilenilen kanalların örnekleme hızını yükseltmeyi amaçlar.

Verimliliğe etki eden bir diğer faktör ise dalgaların gösterileceği ekrandır. Ekranın veriyi uygun şekilde gösterebilmesi için yeter sayıda piksele (nokta) ihtiyacı vardır. Mevcut nokta sayısı ekranın çözünürlüğünü gösterir ki bu, 'Yatay nokta sayısı X Düşey nokta sayısı' notasyonu ile ifade edilmektedir (Örneğin; 1024x768 çözünürlüğe sahip bir ekran).

VGA (Video Graphics Array) bir ekran 640 x 480 çözünürlüğe sahipken, bu değer SVGA (Super Video Graphics Array) bir ekranda 1024 x 768 olarak bilinir. Tipik bir EEG sayfası 10 saniyelik bir veri içerir. 240 Hz örnekleme yapan bir dijital EEG sistemi, ekranda yatay olarak 2400 (240x10) örnek (nokta) göstermek durumundadır. Günümüzde bu değere karşılık gelen bir ekran mevcut olmadığı için dijital bir sistem ekranda piksel başına iki veri noktası çizmek durumundadır. Bu sayede tüm noktalar örnekleme hızı düşürülmeden ekranda gösterilebilmektedir.

Dijitize edilmiş EEG sinyalleri montaj şeklini değiştirmek suretiyle örnekleme anında (on-line) ya da daha sonra (off-line) ayarlanabilir. Bahsi geçen bu ayarlama işlemi, EEG kanallarını montaj şekli gözetmeksizin ortak elektrot kullanarak kaydetmekle mümkün olur. Ortak referans elektroduna göre bilgisayara kaydedilmiş olan örnek verileri sonradan değişik montaj hallerine uygun olarak gösterebilmek mümkündür. Aslında montaj işlemi nümerik notasyonda ortak referansın götürüldüğü bir çıkartma işleminden başka bir şey değildir.

Şekil 3.16'da ortak referansa göre elde edilen bilginin başka bir montaj haline nasıl dönüştürülebildiği gösterilmiştir.

$$\begin{array}{ccc}
 \text{Giriş 1} & & \text{Giriş 2} \\
 (F3 - A1) & - & (F4 - A1) \\
 & \downarrow & \\
 F3 - A1 & - & F4 + A1 \\
 & \downarrow & \\
 & F3 - F4 &
 \end{array}$$

Şekil 3.16 Ortak referansla kaydedilmiş bilginin bir başka montaja dönüşümü [10]

Ortak referansı A1 olan iki girdiyi tek bir kanalda göstermek için birbirinden çıkartmak yeterlidir keza bu işlem A1 referansını matematiksel olarak götürmek suretiyle iki giriş arasındaki farkı dile getirecektir.

3.2. EEG İşaretlerinin Analizi

EEG raporları, deneğin nefes alışverişi ve ışıkla uyarımı esnasında oluşan aktivite değişimlerini gösterebildiği gibi hastanın kayıt esnasındaki durumu hakkında da yorum yapılabilmesini sağlar. İlgili doktor, sonradan bu verileri incelenen hastalık normları çerçevesinde değerlendirerek, izlenmesi gereken tedavi yöntemini seçer. Ama artan veri boyutları bir takım algoritmalar yardımıyla ilgilenilen pikleri ve nöbet anında ortaya çıkan karakterleri aramayı gerekli kılmaktadır (Pang vd., 2003). Klinisyenin tüm kayıtları incelemesi mecburi olsa da en azından ilgilenilen kesimlerin bilgiler ışığında vurgulanması sağlanmış olmaktadır [11].

Bir EEG incelemesi çok sayıda detay içermektedir. Veriler 16 kanal üzerinden saatler boyunca kayıt edilebilmektedir. Zira, hastanın problemini betimleyen teşhis bilgisi, kayıt edilmiş verilerin ufak bir kesrinde ortaya çıkabilmektedir. Uzun bir zaman dilimini hedef alan EEG taraması, kompleks yapısından dolayı, analizcileri, bir takım otomatik veri azaltma yöntemlerine muhtaç bırakmaktadır (Yazgan ve Korürek, 1996). Geliştirilmiş olan yöntemlerden biri Güç Spektrum Analizidir.

3.2.1. Güç Spektrum Analizi

Spektral yaklaşımın amacı bir sinyal içersinde saklı olan güç dağılımını frekansa bağlı olarak sınırlı sayıda bilgidan elde edebilmektir. Bu tarz bir hareket, fazlasıyla gürültülü sinyallere ulaşabilmemiz açısından özellikle önem arz etmektedir (Matlab Help, 2002).

Güç spektrumu olarak adlandırdığımız büyüklük x_n olarak kabul edeceğimiz rasgele prosesin korelasyon sekansıyla yakından ilişkilidir. Bu yaklaşım aşağıdaki formülle (3.1) ifade edilir:

$$S_{xx}(\omega) = \sum_{m=-\infty}^{\infty} R_{xx}(m)e^{-j\omega m} \quad (3.1)$$

$\omega : 2\pi f/f_s$,

f_s : örnekleme frekansını temsil etmektedir.

(3.1) fiziksel frekansın bir fonksiyonu şeklinde yazılırsa,

$$S_{xx}(f) = \sum_{m=-\infty}^{\infty} R_{xx}(m)e^{-2\pi jfm/f_s} \quad (3.2)$$

ifadesini elde etmiş oluruz. Korelasyon sekansını ise ters ayırık Fourier dönüşümü uygulayarak,

$$R_{xx}(m) = \int_{-\pi}^{\pi} \frac{S_{xx}(\omega)e^{j\omega m}}{2\pi} d\omega = \int_{-f_s/2}^{f_s/2} \frac{S_{xx}(f)e^{2\pi jfm/f_s}}{f_s} df \quad (3.3)$$

şeklinde buluruz. x_n sekansının ortalama gücü Nyquist aralığı boyunca,

$$R_{xx}(0) = \int_{-\pi}^{\pi} \frac{S_{xx}(\omega)}{2\pi} d\omega = \int_{-f_s/2}^{f_s/2} \frac{S_{xx}(f)}{f_s} df \quad (3.4)$$

eşitliğiyle hesaplanır.

$$P_{xx}(\omega) = \frac{S_{xx}(\omega)}{2\pi} \quad P_{xx}(f) = \frac{S_{xx}(f)}{f_s} \quad (3.5)$$

Eşitlik 3.5'de geçen büyüklükler gelişigüzel sinyal x_n 'in güç spektral yoğunluğu (PSD) olarak bilinir. Bu sinyalin ortalama gücü verilmiş olan frekans bandında $[\omega_1, \omega_2]$, $0 \leq \omega_1 < \omega_2 \leq \pi$ olmak üzere,

$$\bar{P}_{[\omega_1, \omega_2]} = \int_{\omega_1}^{\omega_2} P_{xx}(\omega) d\omega + \int_{-\omega_2}^{-\omega_1} P_{xx}(\omega) d\omega \quad (3.6)$$

şeklinindedir. PSD'nin birimi güç/frekans olarak verilir.

3.2.2. Spektral Yaklaşım Metotları

Tüm yöntemleri üç kategoriye ayırmamız mümkündür:

- Parametrik olmayan metotlar

- Parametrik metotlar
- Uzay-altı (sub-space) metotlar

Parametrik olmayan metotlarda PSD doğrudan sinyalin kendisinden elde edilmektedir. Periodogram yöntemi bu metotlardan biriyken, adı geçen bu yöntemin daha gelişmiş bir versiyonu olarak Welch metodunu örnek gösterebiliriz. Daha modern bir teknik ise Multitaper metodudur (Matlab Help, 2002).

Parametrik metotlarda PSD beyaz gürültü tarafından sürülmekte olan lineer bir sistemden elde edilen bir sinyalden hesaplanmaktadır. Örnekler arasında Yule-Walker's Autoregressive (AR) metodu ve Burg metodu sayılabilir. Bu yöntemler uydurulacak olan lineer sistemin katsayılarını önce hesaplar, daha sonra da oluşturulan sistemden sinyali simule ederler. Parametrik olmayan yöntemlere göre data uzunluğu kısa olan uygulamalarda daha doğru sonuç verebilmektedir.

Uzay-altı metotlar aynı zamanda yüksek çözünürlük (ya da süper çözünürlük) yöntemi olarak da bilinir. Yöntem yapısı gereği bir sinyal için özvektör analizinden yola çıkarak frekans komponentleri oluşturur.

3.2.2.1 Yule-Walker AR Yöntemi

En çok kullanılan lineer sistem modeli 'all-pole' modelidir ki, bir filtre olarak tüm sıfırları z -düzleminin orijininde yer alır. Bu mahiyette olan bir filtrenin beyaz gürültü girişine vereceği çıkış oto-regresif (autoregressive, AR) bir prosestir (Matlab Help, 2002).

Tüm AR metotları PSD yaklaşık sonucunu,

$$\hat{P}_{AR}(f) = \frac{1}{f_s} \frac{\epsilon_p}{\left| 1 + \sum_{k=1}^p \hat{a}_p(k) e^{-2\pi j k f / f_s} \right|^2} \quad (3.7)$$

formülü ile hesaplar. Yule Walker AR metodu AR parametrelerini sinyalin oto-korelasyon fonksiyonunun eğilimli (biased) yakınsamasını oluşturmak suretiyle hesaplar ve ileri tahmin hatasının (forward-prediction error) karesel minimizasyonunu (least squares minimization) gerçekleştirerek çözer. Bu Yule-Walker eşitliklerini (3.8) verir.

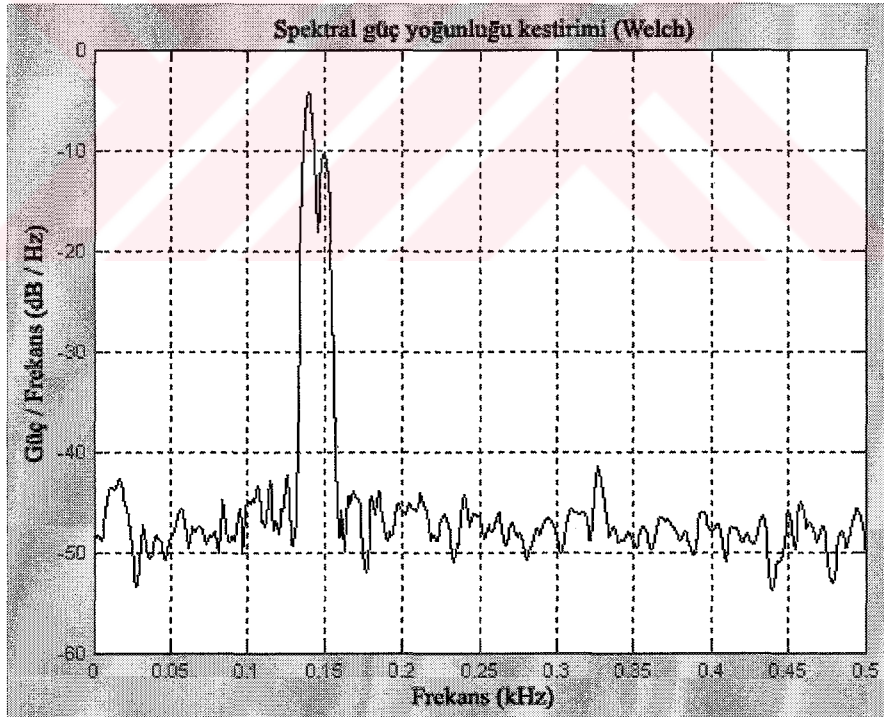
$$\begin{bmatrix} r(1) & r(2)^* & \dots & r(p)^* \\ r(2) & r(1) & \dots & r(p-1)^* \\ \vdots & \ddots & \ddots & \vdots \\ r(p) & \dots & r(2) & r(1) \end{bmatrix} \begin{bmatrix} a(2) \\ a(3) \\ \vdots \\ a(p+1) \end{bmatrix} = \begin{bmatrix} -r(2) \\ -r(3) \\ \vdots \\ -r(p+1) \end{bmatrix} \quad (3.8)$$

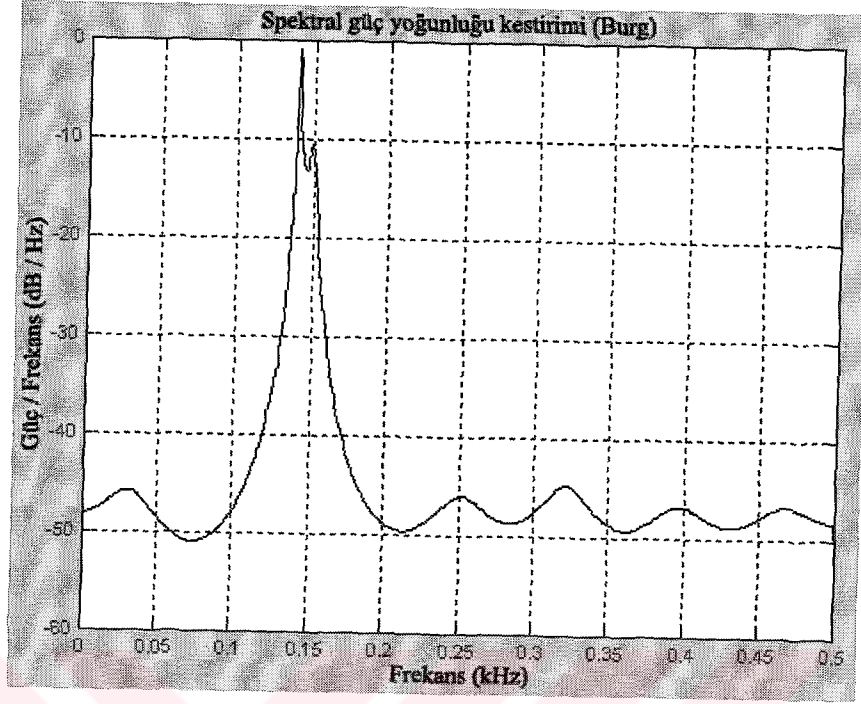
3.2.2.2 Burg Yöntemi

Burg yöntemi ileri ve geri tahmin hatasını minimize ederken Levinson-Durbin yinelemesini (Levinson-Durbin recursion) sağlamak üzerine kuruludur. Diğer AR yaklaşım metotlarının aksine, Burg yöntemi oto-korelasyon fonksiyonunu hesaplamaz, yansıma katsayılarını direkt olarak bulur (Matlab Help, 2002).

Yöntemin avantajlarından biri sinyal içerisinde birbirine yakın yerleşimli düşük gürültü seviyelerindeki sinüzoidleri çözümleyebilmesi ve kısa data kayıtlarından sonuç çıkartabilmesidir. Her iki durumda da AR güç spektral yoğunluk yakınsamaları gerçek değerlere yakındır. Burg metodu dengeli bir AR modeli oluştururken, etkin bir hesaplama kabiliyetine sahiptir.

Öte yandan Burg yöntemi verimi, yüksek seviyeli modellerde, uzun data kayıtlarında ve yüksek sinyal/gürültü durumlarında oldukça düşer. Ayrıca, Burg yöntemiyle bulunmuş olan spektral yoğunluk yaklaşık değeri gürültülü sinyalin başlangıç fazından kaynaklanan frekans oynamalarına karşı etkiye açıktır. Bu etki, küçük data sekanslarında oldukça fazladır. Adı geçen etki Şekil 3.17’de açıkça görülmektedir.





Şekil 3.17 Burg yöntemiyle bulunan spektral yoğunluk değerinin frekans oynamalarına karşı etkiye açıklığı (Matlab Help, 2002)

4. YAPAY SİNİR AĞLARI

Yapay sinir ağlarına olan ilgi McCulloch&Pitts'in 1943 yılında basitleştirilmiş nöronları bilim camiasına kazandırması ile başladı. Tanıtılan bu ilk çalışma hayvansal nöron hücrelerinin bir modeli olarak ortaya çıkmış, takip eden dönemde sayısal hesaplamaların gerçekleştirilebildiği devrelerin bir parçası olarak kullanılmıştır (Krose ve Patrick, 1996). Minsk ve Papert'in 1969 tarihli 'Perceptrons' adlı kitaplarını çıkartmaları ile nöral ağların zayıf yanları ve eksiklikleri gözler önüne serildi. Bu durum nöral ağların araştırılması için ayrılmış fonun başka bir bölüme kaydırılmasına sebep olurken, araştırmacıların çoğu bu alanı terk etti. Geriye kalan Kohonen, Grossberg, Anderson ve Fukushima gibi şahıslar ise kendi bireysel çabalarıyla araştırmalarını sürdürdü. Seksenlerin sonunda 'hatanın geriye yayılımı' algoritmasının keşfiyle, donanımda olan gelişmelerden istifade edilmek suretiyle, tekrar nöral ağlara olan ilgi artış gösterdi. Günümüzde çoğu üniversite, bünyesinde nöral ağ araştırmalarını takip ve öncülük ettiği psikoloji, fizik, elektrik ve elektronik, bilgisayar bilimleri ve biyoloji departmanlarını barındırmaktadır.

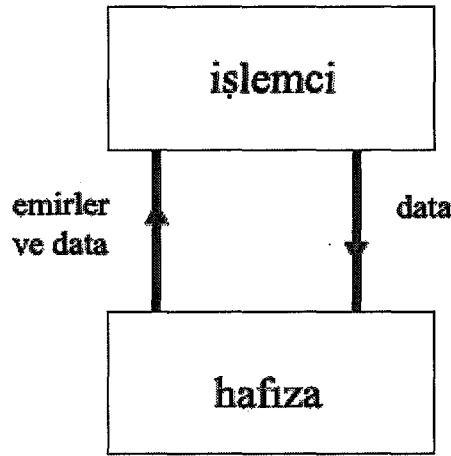
Yapay sinir ağları (YSA) yapı itibariyle kendi aralarında bağlı bir çok işlemci biriminden (nöronlar) oluşur. YSA'ları tıpkı insan ve diğer canlılar gibi örnekler ile öğrenir. Örüntü tanıma ve bilgi sınıflandırma uygulamaları için spesifik olarak konfigüre edilen yapay sinir ağlarının öğrenme işlemi, biyolojik sistemlerdekine benzer bir biçimde nöronları birbirine bağlayan ağırlıkların değiştirilmesi ile gerçekleştirilir. Bu durumun biyolojik karşılığı nöronlar arasındaki sinaptik bağlantının kimyasal anlamda adaptasyonu olarak düşünülebilir [12].

YSA'nın kompleks ve/veya eksik bilgi ile baş edebilmesi, örüntülerin matematiksel olarak anlaşılması güç durumlar içinden tanınmasına imkan verir. YSA'nın analizi istenen bilgi üzerine eksperlik kazanmasının ardından, sorulan sorulara doğruya en yakın cevapları vermesi beklenir. Diğer bir takım avantajları aşağıda sıralanmıştır:

1. Adaptif Öğrenme: Verilen bilgiye göre (ilk tecrübe) çıkış verebilme yeteneği.
2. Kendini Düzenleme: Kendi düzenini veya bilginin temsil edilmesini öğrenme sürecinde gerçekleştirebilmesi.
3. Gerçek Zamanlı Operasyon: YSA hesaplamaları birbirine paralel olarak yürütebilir. Dolayısıyla, bu özelliği kullanan özel ekipmanlar aracılığıyla eş zamanlı yürütüm gerçekleştirilebilir.

4. Hata ve/veya Eksik Bilgiye Karşı Tolerans: Ağ yapısının bir kısmının yok olması bilginin uçup gitmesi anlamına gelmez. YSA yayılmış bir bilgi saklama tekniği kullandığı için yapısal olarak cüzi bir zarar görmesi sadece tanımlama veriminin düşmesine neden olur. Hatta bazı durumlarda verim kaybı bile gözlenmeyebilir.

YSA problemlerin çözümünde bilgisayarlardan daha farklı bir yaklaşım kullanır. Geleneksel bilgisayarlar bilinen durumlar için önceden belirlenmiş bir algoritma yürütmekle yükümlüdür. Eğer yürütmesi gerekli olan adımlar bilinmiyorsa, bilgisayarın çözüm üretmesi beklenemez. Diğer yandan, yapay sinir ağlarını tam olarak bu yüzden kullanma ihtiyacı duyarız: Çözüm yöntemini bilmediğimiz bir problem türü için. YSA, insan beyninin bilgiyle başa çıkmasına benzer biçimde işler. Ağ birbiriyle bağlantılı bir çok birimin paralel işlem kabiliyetinden yararlanarak spesifik bir problemin çözümünü verir. Benzerlikleri ve örnekleri baz alarak işlediği için klasik anlamda "IF-THEN" önermeleri şeklinde programlanması mümkün değildir. Örnek bilgi çiftleri (giriş-çıkış) doğru şekilde seçildiği takdirde çözüm mevcuttur. Aksi halde, yapılandırma işlemi doğru olarak gerçekleştirilemediği için ağ doğru şekilde çalışmıyor olabilir. En büyük dezavantajı, ağ yapısı problemi çözme yolunu kendisi ürettiği için yaptığı operasyon önceden tahmin edilemeyebilir. Diğer yandan, klasik bilgisayarlar problem çözümüne daha kavramsal bir şekilde yaklaşır. Problemi çözmenin mümkün olduğu yöntem biliniyor ise basit bir mahiyete indirgenerek yüksek seviyeli bir dilde yazılan program aracılığıyla donanımın ne yapması gerektiği ifade edilir. Akabinde yürütülen program ile işlemcinin istenen komutlar vasıtasıyla çözüm üretmesi kabil olur. Şekil 4.1'de tüm bilgisayarların operasyon mantığını tanımlayan bir model olan 'von Neumann' makinesi görülmektedir (Gurney ve Kevin, 2000).



von-Neumann makinesi

Şekil 4.1 von-Neumann makinesi (Gurney ve Kevin, 2000)

Bilgisayar devamlı olarak aşağıda belirtilmiş olan çevrimsel olayları gerçekleştirir:

1. Hafızadan komutu oku
2. Hafızadan komut için gerekli olan her bilgiyi(data) oku
3. Komutu yürütmek suretiyle bilgiyi işle
4. Sonuçları hafızaya kaydet
5. Birinci basamağa geri dön.

Birçok problemi algoritmalar aracılığıyla bilgisayara tanımlayabiliriz. Böylelikle makine, önceden belirlenmiş bir prosedür eşliğinde cevabı garantilemiş olur. Bir denklemin hesaplanması veya veritabanındaki bir verinin aranması gibi işler daha basit önermeler kümesiyle tanımlanmış bir algorithmadan hareketle ufak parçalara ayrılıp, CPU'nun anlayacağı ve yürüteceği komutlara indirgenir. Çözüm üretebilmesi için, bilgisayarın aşağıdaki kriterlere uyması gerekmektedir:

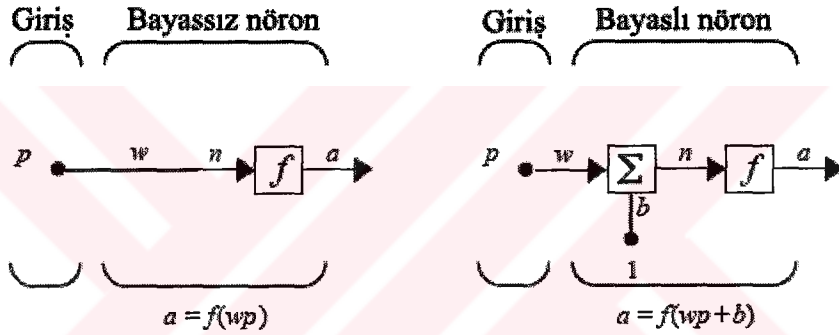
1. Problemin çözülmesini sağlayacak algoritma (eğer biliniyorsa) makineye detaylı bir biçimde hiçbir mantıksal hataya düşmeden anlatılmak zorundadır. Bu birbiri ardına gelen bir komut silsilesiyle bilgisayar yazılımını ortaya çıkarmaktadır.
2. Makinenin ilgilenmesi gerektiği bilgi kesin bir formata sahip olmalıdır. Zira gürültü içeren bilgi makineyi yanıltır.
3. Donanım kolaylıkla işlemez hale gelebilir. Eğer önemli derecedeki hafıza noktalarından biri zarar görürse, bu durum makinenin işlememesine veya diğer bir deyişle 'crash' etmesine (çakılmasına) neden olur.
4. İlgilenilen veri (sayı, kelime, veritabanı girdileri) ile donanım arasında orantısal bir ilişki mevcuttur. Her obje hafıza bloğunda bir yere atanır.

Sembolik çözümün başarısının kalbinde yatan gerçek problemi doğru şekilde tanımlayıp çözebilmeyi sağlayacak algoritmanın bilinmesi gerekliliğidir. Aksi halde çözüme ulaşamaz. Öte yandan günlük hayatta bizim rahatlıkla başa çıkabildiğimiz görsel veya işitsel çıkarım, el yazısının tanınması, yazılı bir metinden veya söylenen sözlerden yola çıkarak görsel veya işitsel bir takım anıların hatırlanması, yüzünü daha önce o açıdan görmediğimiz bir kişiyi tanımamız, bilgisayar programlama mantığına algoritmalar aracılığıyla nasıl indirgenebilir? Bu tür zorlukları bertaraf etmek için beynimizin fiziksel yapısı ve işleme mantığını kullanabiliriz.

YSA ve klasik algoritmalar kullanan bilgisayarlar birbiriyle rekabet etmemekte, aksine tamamlamaktadır. Günümüzde algoritmik çözüme uygun problemler (aritmetik işlemler) olduğu gibi, YSA'nın rahatlıkla başa çıkabileceği durumlar da mevcuttur. Hatta rahatlıkla söylenebilir ki çağımızın bilişim teknolojileri bu iki yaklaşımı aynı potada eritebilmiş bir takım yeni yöntemleri aranan bir kriter haline getirmiştir. Zira klasik bir bilgisayarın yönettiği YSA maksimum performansın elde edilmesine imkan vermektedir.

4.1 YSA'nın Yapısı ve İşleyişi

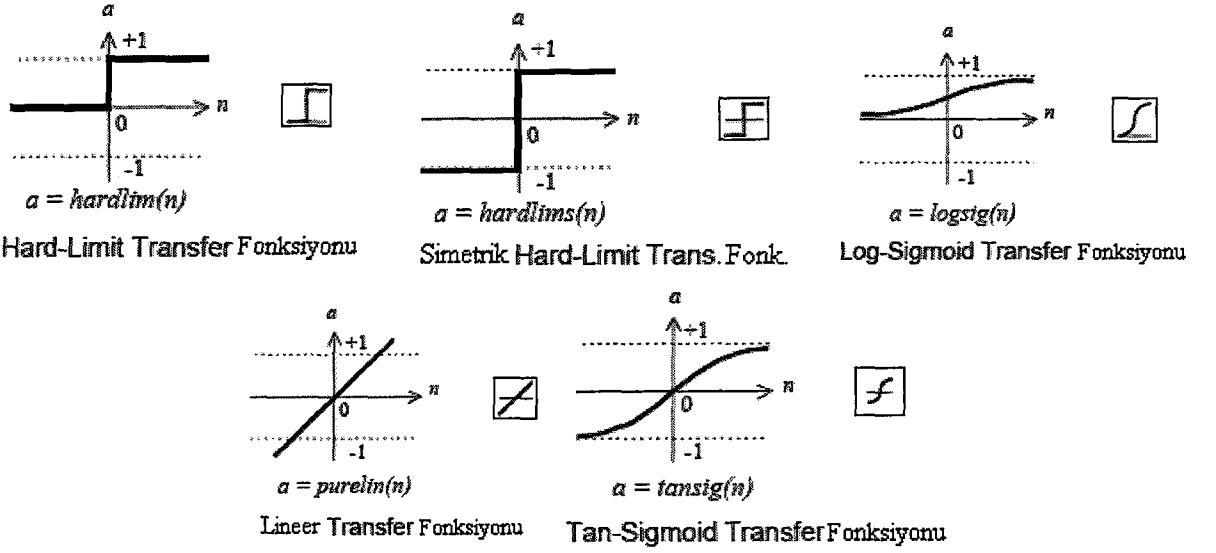
Basit bir nöron ünitesi kendisine gelmekte olan girişi (p), skaler bir ağırlık olan w ile çarparak gene skaler bir büyüklük olan wp 'yi oluşturur. Burada ağırlıklı giriş, f fonksiyonunun yegane argümanı niteliğindedir. Dolayısıyla, çıkış gene skaler bir değer olan a olacaktır (Demuth, 2002).



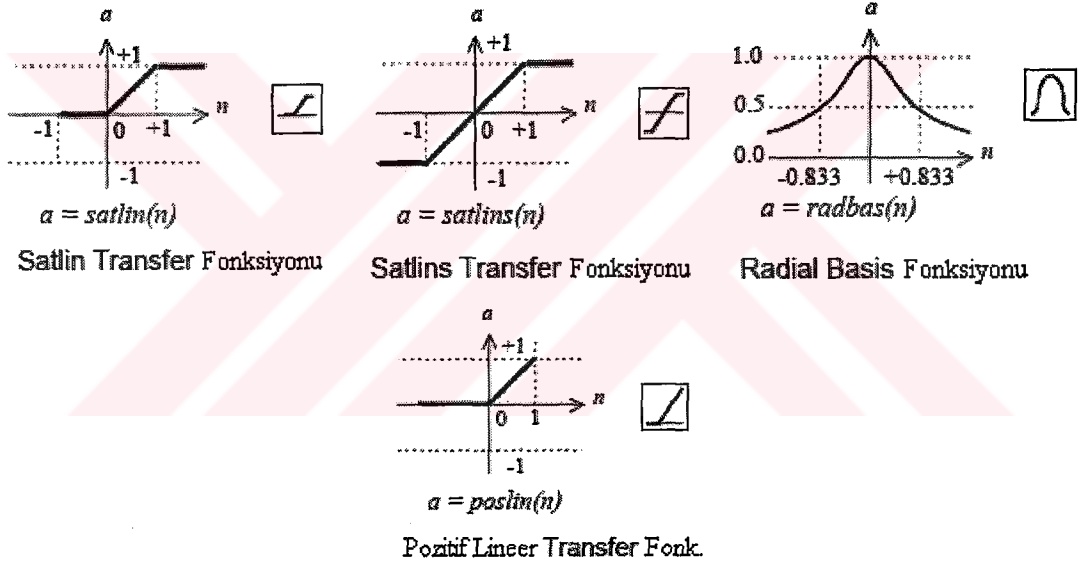
Şekil 4.2 Basit nöron yapısı (Demuth, 2002)

Üst sağdaki nöron ise b biasına sahiptir. Ağırlıklı toplam işlemine dahil edilen bir giriş gibi etkisi olan b , sabit bir değerden başka bir şey değildir. YSA'nın asıl mantığı parametrelerini manipüle etmek suretiyle arzu edilen davranış biçimini sergilemesini sağlayabilmektir. Bu durumda w ve b nöronun ayarlanabilir skaler parametreleridir. Böylece ağırlık ve biasın ayar edilmesiyle, yapay sinir ağını eğitmeyi becerebiliriz. Hatta daha da ileri giderek uygun bir algoritma yardımıyla ağın kendi kendini eğitmesi mümkündür diyebiliriz.

Söz edilmesi gereken diğer bir konu ise transfer fonksiyonu olarak tanımladığımız, yapay nöronun davranış biçimini belirleyen f fonksiyonudur. Yıllardır sıklıkla kullanılan gelmiş bir takım transfer fonksiyonları aşağıda görülmektedir.



Şekil 4.3 Transfer fonksiyonları (Demuth, 2002)

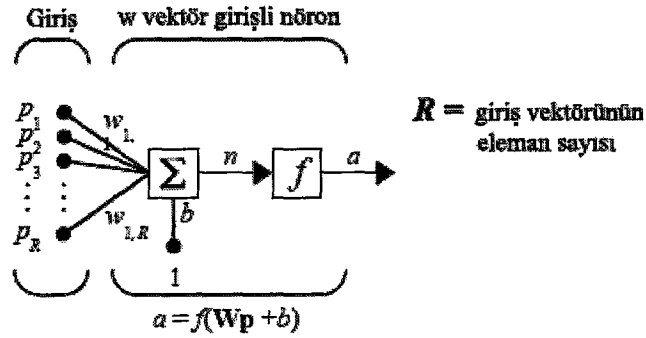


Şekil 4.4 Transfer fonksiyonları (devam) (Demuth, 2002)

'Hard-limit' transfer fonksiyonu nöron çıkışının '0' ile '1' arasında kalmasını sağlar. Eğer net giriş n sıfırdan küçükse '0', sıfıra eşit veya büyükse '1' değerini alır. Bu fonksiyon sınıflandırma kararları verebilen nöronlar oluşturmak için kullanılmaktadır.

Öte yandan 'lineer' transfer fonksiyonu lineer yakınsayıcılarda, 'log-sigmoid' fonksiyonu ise eksi sonsuz artı sonsuz aralığında tüm girişleri kabul edip, çıkışı '0' ile '1' arasında tutmak suretiyle, geriye yayımlı ağlarda sıklıkla karşımıza çıkmaktadır.

Gerçeğe yakın bir işleyiş tarzı isteniyorsa tek girişe sahip bir nöronun günümüz mühendislik uygulamaları için kafi gelmeyeceği aşikardır. Bu yüzden R-elemanına sahip tek nöronlu bir ağ yapısı, ardından da çok katmanlı ağ tipi incelenecektir (Demuth, 2002).

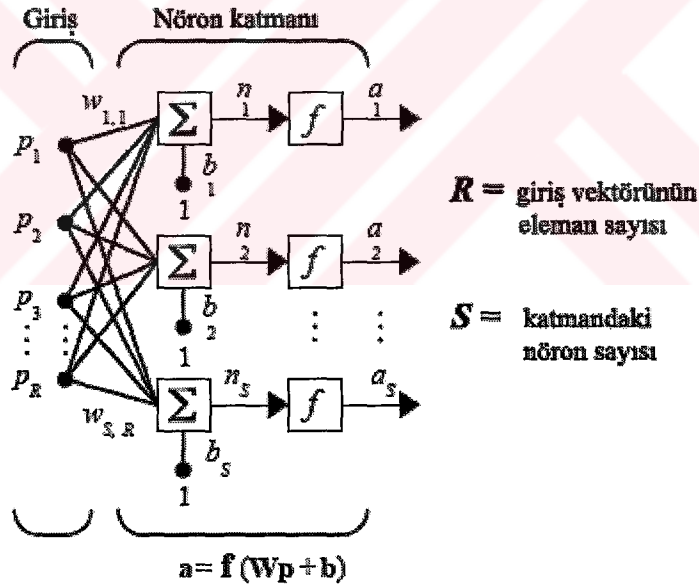


Şekil 4.5 Vektör girişli nöron (Demuth, 2002)

Çok girişli ağ yapısında ağırlıklı toplam işlemi,

$$n = w_{1,1} p_1 + w_{1,2} p_2 + \dots + w_{1,R} p_R + b \quad (4.1)$$

formülü ile belirtilmiş olup f fonksiyonun sonucu ise çıkışı ifade etmektedir.



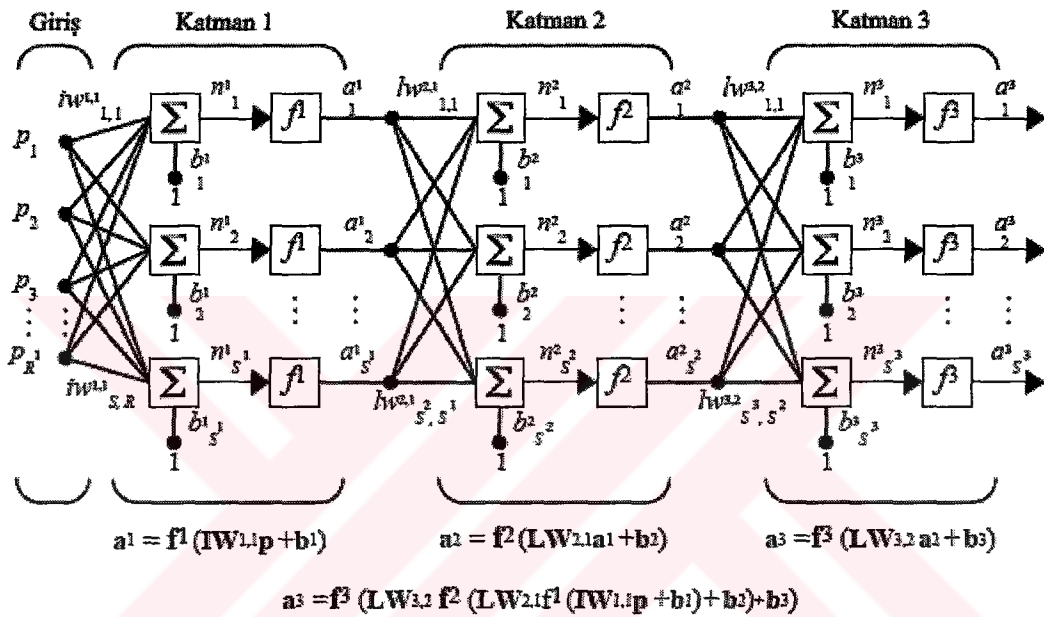
Şekil 4.6 Çok girişli nöron katmanı (Demuth, 2002)

Üstte görülmekte olan tek katmanlı ağ yapısında p giriş vektörünün her elemanı nöronlara W ağırlık matrisi ile bağlıdır. Dolayısıyla belirtilmiş olan ağ yapısı p giriş vektörünü, matris ile

$$W = \begin{bmatrix} w_{1,1} & w_{1,2} & \dots & w_{1,R} \\ w_{2,1} & w_{2,2} & \dots & w_{2,R} \\ \vdots & \vdots & \ddots & \vdots \\ w_{S,1} & w_{S,2} & \dots & w_{S,R} \end{bmatrix} \quad (4.2)$$

çarpar, sonucu b bayas değeri ile toplayarak f fonksiyonundan geçirir (Demuth, 2002).

Ama bu bile mühendislik uygulamaları için kafi gelmeyebilir. Zira, evrensel yakınsayıcılar (günlük uygulamaların çoğunun yorumlanması için şarttır) en az iki katmana ihtiyaç duyar.



Şekil 4.7 Çok katmanlı nöron ağı yapısı (Demuth, 2002)

Çok katmanlı ağ (Şekil 4.7), yapısında tek katmanlıdan bir çok adet barındırır. Her katman bir ağırlık matrisi W 'ya, bir bayas vektörü b 'ye ve bir çıkış vektörü a 'ya sahiptir. Ağırlık matrisinin yazım notasyonu w geldiği giriş, gittiği nöron numarası şeklinde ifade edilmiştir. Şekil 4.7'de R^1 giriş sayısını, S^1 , S^2 ve S^3 ise sırasıyla birinci, ikinci ve üçüncü katmandaki nöron sayısını ifade etmektedir (Demuth, 2002).

Ağın çıkışını sağlayan katmana çıkış katmanı, kendisi dışındaki tüm katmanlara ise gizli katmanlar adı verilir. Çok katmanlı ağ yapıları, oldukça doygun matematiksel güce sahiptir. İki katmanlı bir ağ olan evrensel yakınsayıcı ilk katmanında sigmoid fonksiyonunu, ikinci katmanında ise lineer transfer fonksiyonunu barındırır. Bu sayede bilinen tüm fonksiyonları yakınsamak için geriye yayılım algoritması ile birlikte sıklıkla kullanılır.

4.2 YSA'nın Öğrenmesi ve Eğitilmesi

Tüm eğitim metotları iki temel kategoride toplanabilir: Öğreticili öğrenme ve Öğreticisiz öğrenme [12].

Öğreticili öğrenmede dışsal bir eğitici elemanı, giriş sinyaline bakarak çıkış ünitesine ne yapması gerektiğini tanımlar. Bu öğrenme tipinde global bilgiye ihtiyaç duyulur. Hata düzeltme öğrenimi, pekiştirmeli öğrenme (reinforcement learning), olasılıklı öğrenme (stochastic learning) öğreticili öğrenme tipine örnek olarak gösterilebilir. Bu öğrenme türünde ilgilenilmesi gereken en önemli konu hatanın en aza indirgenebilmesidir. Hatanın azaltılması ağırlık değerlerinin o problem için optimal değerlerine ulaşmasıyla mümkün olur. Böylece hata geriye çekilmişken, ağırlık giriş desenini ezberlememesi sağlanır. Kullanıla gelmiş en bilinen metot LMS (least mean square) yakınsamasıdır.

Öğreticisiz öğrenme tipinde ise dışsal bir eğitici yoktur. Dolayısıyla ağ sadece lokal bilgiye bakmak suretiyle işlemektedir. Kendini düzenleme ile öğrenme olarak da bilinen bu yöntem verilen dataya bakarak benzerlikleri bulur, bu benzer yapıları öğrenerek işleyişini sürdürür. Hebbian öğrenimi ve rekabetçi öğrenme öğreticisiz öğrenme tipine örnek olarak verilebilir.

Dikkate değer bir diğer konu ise çevrimiçi ve çevrimdışı kavramlarının kendisidir. Ağ yapısı eğer öğrenme işlemi ve operasyonu (çıkış verme işlemi) ayrı vakitlerde gerçekleştiriyorsa çevrimdışı, eğer aynı zamanda yapıyorsa çevrimiçi eğitim gerçekleştirmektedir. Çoğunlukla öğreticili öğrenme çevrimdışı, öğreticisiz öğrenme ise çevrimiçi karakter taşır.

YSA için önemli olan bir diğer konu ise eğitim stilleridir. Bahsi geçecek olan eğitim tiplerinden biri artışı (incremental) eğitimidir. Bu tip eğitimde YSA'nın ağırlık ve bayasları ağa giriş verildikçe değişime uğrar. Öte yandan grup (batch) halinde eğitimde ise ağırlık ve bayas güncelleştirme işlemi tüm girişler ve hedefler verildikten sonra yapılır (Demuth, 2002).

Hedef ile asıl çıkış arasındaki fark ağın hatasını ifade eder. İstenilen durum örnek ne olursa olsun istenilen sonucun gerçeği ile uyuşmasıdır. Bu sebepten dolayı bir takım eğitme algoritmaları geliştirilmiştir. En tanınanı 'Geriye Yayılım' algoritmasıdır. Şu an hız olarak adı geçen bu algoritmadan daha hızlı yöntemler olmasına karşın basitliği ve anlaşılabilirliği neticesinde halen etkin bir bilgi kaynağıdır.

En basit anlamda algoritma ağırlık ve bayasları, performans fonksiyonunun en hızlı azaldığı yönde (hata gradyeninin tersi yönde) günceller. Tek bir iterasyon şu şekilde yazılabilir:

$$X_{k+1} = X_k - \alpha_k g_k \quad (4.3)$$

Burada X_k o anki ağırlık ve bayasları ihtiva eden vektör, g_k o anki gradyeni ($\nabla E[w]=[\partial E/\partial w_0, \dots \partial E/\partial w_n]$), α_k ise öğrenme parametresini ifade etmektedir.

Algoritmanın işleyişi şu şekilde ifade edilebilir:

- Başlangıç ağırlıklarını rasgele seç
- Hata istenilen değerden yüksekken ($E > E\text{-istenen}$)
 - Her eğitilecek örüntü için
 - * Girişleri YSA'ya ver
 - * Her nöronun çıkışını hesapla (girişten çıkışa kadar)
 - * Çıkışlardaki hatayı hesapla
 - * Çıkış hatasını kullanarak gizli katmanlar için hata sinyallerini hesapla
 - * Hata sinyallerini kullanarak ağırlıklardaki değişimleri hesapla
 - * Ağırlıkları güncelle

Öğrenme parametresi ne kadar büyükse atılacak adım da o kadar büyük olur. Eğer bu parametrenin değeri çok yüksek olursa algoritma kararsız bir yapıya sürüklenir (önüne geçilemeyen osilasyonlar gözlenir). Eğer çok küçük bir değer verilirse bu sefer de yakınsama işlemi çok uzun zaman alır. Daha hızlı yakınsama gerçekleştirebilmek için algoritmanın işleme fonksiyonuna bir momentum terimi iliştirebiliriz (Demuth, 2002).

Yerleştirdiğimiz bu terim ağırlık, sadece hatanın lokal gradyenine değil aynı zamanda hata yüzeyindeki değişimlere de cevap verebilmesini sağlar. Alçak-geçirgen bir filtre gibi davranarak, hata yüzeyindeki ufak değişimlerden ağırlık etkilenmemesi sağlanmış olur. Momentum terimi olmasa yayvan bir lokal minima noktasında takılıp kalınabilir. Bunu bertaraf etmek için iterasyona son değişimin bir kesri eklenmelidir. Yeni iterasyon,

$$X_{k+1} = mc X_k - \alpha_k g_k \quad (4.4)$$

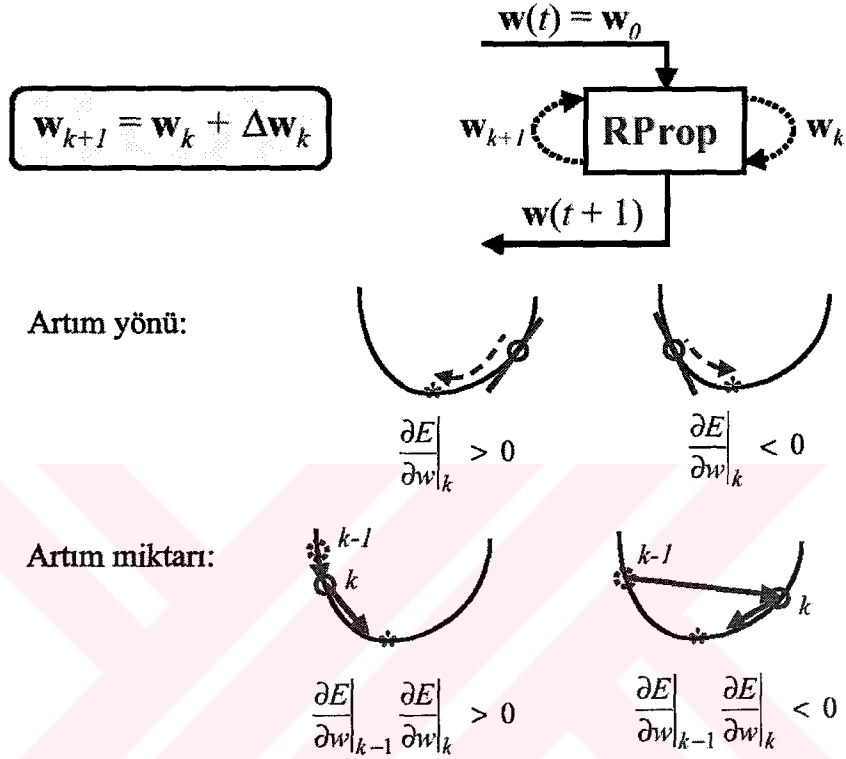
şeklinde gösterilebilir. Momentum katsayısı 'mc' 0 ile 1 arasında bir sayıdır. Eğer bu değer bir olursa değişim sadece gradyene göre oluşacaktır. Şayet bu sayı 1 olarak seçilirse bu sefer de iterasyon son ağırlık değişimine dayandırılacaktır. Bu değer çoğunlukla 0.9 olarak seçilir.

Diğer yöntemler kısıtlı olan hafızayı daha uygun şekilde kullanmak suretiyle hız elde etmeye yöneliktir. Bahsi geçen bu yöntemler sırasıyla aşağıda verilmiştir (Demuth, 2002).

Traingda Algoritması: Öğrenme parametresinin seçimi klasik geriye yayılım algoritmasında büyük önem arz etmektedir. Zira eğer büyük seçilirse, osilasyonlar ve stabil olmayan koşullar

gözlemlenirken, çok küçük seçimi ise algoritmanın çok uzun sürede yakınsamasına sebep olabilir. Bu yüzden adaptive öğrenme algoritması ile öğrenme adımlarının büyüklüğünün olabildiği kadar yüksek tutulması amaçlanır. Böylece öğrenme oranı hata yüzeyinin karmaşıklığına göre şekillenir.

Trainngdx Algoritması: Üstteki yönteme momentum terimi eklenmesi ile elde edilir.



Şekil 4.8 Resilient backpropagation algoritması [16]

Resilient Backpropagation Algoritması (trainrp): Steepest descent ile sigmoid fonksiyonlu ağları eğitirken gradyen değerleri çok küçülebilir, dolayısıyla bu da optimal değerlerine ulaşılmadığı halde bayas ve ağırlıkların çok az değişmesine sebebiyet verir. Bunun önüne geçmek için Rprop tepe değerlerinden ziyade türevlerin işaretlerini baz alarak ağırlıkları günceller. Ağırlık güncellemesi ayrı bir fonksiyonla ifade edilebilir. Şöyle ki eğer performans fonksiyonunun ağırlığa göre türevi iki iterasyon sonunda aynı işarete sahip oluyorsa (yani aynı yönde ilerleniyorsa) ağırlık güncelleme faktörü, (delt_inc kadar) artırılır. Keza bayas için de bu geçerlidir. Eğer işaret değişimi gözleniyorsa, ağırlık ve/veya bayas güncelleme miktarı (delt_dec kadar) azaltılır. Eğer işaret değişimi yoksa, güncelleme miktarı aynı kalır. Şayet ağırlıklarda osilasyonlar gözleniyorsa, ağırlık değişimi azaltılır. Yok eğer ağırlıklar bir çok iterasyon sonunda aynı yönde değişmeye devam ediyorsa, bu vakitte ağırlık değişiminin değeri arttırılacaktır.

Conjugate Gradient Algoritması: Klasik BP algoritmasında ağırlıklar hatanın gradyenin ters yönünde değişim göstermekle yükümlüdür. Bunun sebebi hatanın o yönde en hızlı şekilde azaldığı içindir. Ama fonksiyonun o yönde en hızlı şekilde azalması, en hızlı şekilde yakınsayacağı anlamına gelmez. Her iki yönde de bir arama işlemi gerçekleştirilir. Performans fonksiyonunu (hatayı) minimize edecek şekilde, o doğru üzerinde konjuge gradyen yönünde bir arama yapılır.

Fletcher-Reeves Güncellemesi (traincgf): Önce bir yönde arama yapılır. Ve başlangıçta (her zamanki gibi) gradyenin ters yönü seçilir.

$$P_0 = -g_0 \quad (4.5)$$

Bir çizgi boyunca arama yöntemi ile başlanan yönde ilerlenecek optimal uzaklık keşfedilir.

$$X_{k+1} = X_k + \alpha_k p_k \quad (4.6)$$

Daha sonra yeni bir arama yönü seçilir (konjuges). Burada kullanılan, yeni yön bulma prosedürü ile bir evvelki yönü (steepest descent yönü) birleştirmek amacını güder diye ifade edilebilir.

$$p_k = -g_k + \beta_k p_{k-1} \quad (4.7)$$

Konjuge gradyenler de Fletcher-Reeves güncelleme prosedürü olarak bahsi geçen β_k ile bulunur.

$$\beta_k = \frac{g_k^T g_k}{g_{k-1}^T g_{k-1}} \quad (4.8)$$

Polak-Ribière Güncellemesi (traincgp): Arama yönü seçimi Fletcher-Reeves algoritmasının aynıdır. Tek farkla:

$$\beta_k = \frac{\Delta g_{k-1}^T g_k}{g_{k-1}^T g_{k-1}} \quad (4.9)$$

Powell-Beale Restarts (traincgb): Tüm konjuge gradyen algoritmalarında arama yönü periyodik olarak gradyenin tersine(başlangıçta seçilen yöne) reset olur. Bu reset durumu doğal olarak iterasyon sayısının ağırlık ve bayas sayısına eşit olduğu durumda ortaya çıkar. Ama bu her zaman verimli olacak diye bir kural yoktur. Başka bir reset metodu da Powell-Beale yöntemidir.

$$|g_{k-1}^T g_k| \geq 0.2 \|g_k\|^2 \quad (4.10)$$

(4.10) ifadesi sağlanırsa, arama yönü gradyenin tersine reset olur.

Scaled Conjugate Gradient (traincg): Evvelki tüm konjuge gradyen algoritmaları çizgi üzerinde arama yöntemlerini kullanıyordu. Bu yöntemin hesap açısından getirdiği yük bahsi geçen bu yöntemle giderilmeye çalışılmıştır. İzlenen yol ise Levenberg-Marquardt algoritmasıyla konjuge gradyen yöntemini bağdaştırmaya dayalıdır.

Quasi-Newton Algoritması: Bu yöntem konjuge gradyen yöntemlerine alternatiftir.

BFGS Algoritması (trainbfg): Bu algoritmayı temsil eden (4.11) eşitliği, içinde A_k 'yi

$$X_{k+1} = X_k - A_k^{-1} g_k \quad (4.11)$$

Yani, Hessian matrisini (performans indeksinin ikinci türevi) barındırmaktadır. Eğer A_k kolayca hesaplanabilecek bir yöntemle sahipse, o zaman Newton yöntemi konjuge gradyen yöntemlerinden daha hızlı yakınsamayı becerir. İkinci türevleri doğrudan hesaplamadan bu işlemler gerçekleştirilebilmektedir. Bu yöntem Quasi-Newton metodu adı verilmektedir. Yöntem yaklaşık bir Hessian matrisini her iterasyonda günceller. Bu update işlemi ise gradyenin bir fonksiyonu şeklindedir.

One Step Secant Algoritması (trainoss): Bu yöntem konjuge gradyen algoritması ve Quasi-Newton(secant) algoritması arasındaki boşluğu kapamak için vardır. Hessian matrisinin tamamı saklanmaz. Her iterasyonda, evvelki Hessian'ın birim matris olduğunu varsayar. Böylece yeni arama yönü bir matrisin tersini almadan bulunur.

Levenberg-Marquardt Algoritması (trainlm): Quasi-Newton yöntemi gibi bu algoritma da Hessian matrisini hesaplama külfetini ortadan kaldırmak için tasarlanmıştır. Eğer performans fonksiyonu karesel toplamdan ihtiva olmuşsa ki ileri beslemeli ağlarda bu tipiktir, Hessian matrisi,

$$H = J^T J \quad (4.12)$$

ve gradyen de

$$g = J^T e \quad (4.13)$$

şeklinde ifade edilebilir. J burada Jacobiyeni temsil eder ki bu da ağırlık ve bayaslara göre birinci türevi demektir. Güncelleme ise Quasi-Newton'a benzer bir şekilde sağlanır:

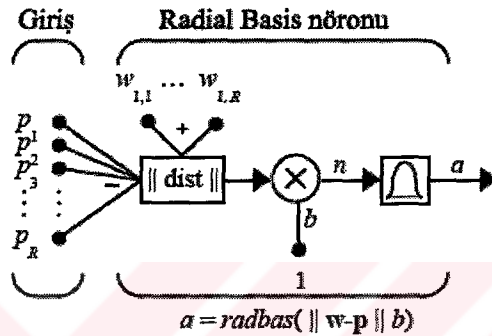
$$X_{k+1} = X_k - [J^T J + \mu I]^{-1} J^T e \quad (4.14)$$

μ sıfırken üstteki eşitlik Newton yöntemi, μ büyükken ise küçük adım büyüklüğüne sahip bir 'gradient descent' olur.

Bayesian Regülasyonu (trainbr): Burada ağ sadece karesel hatanın minimize edilmesini üstlenmez aynı zamanda, ağırlık ve bayasların karesel ortalamasını da aşağıya çekmeye çalışır. Bu sayede ağı daha küçük ağırlıklara ve bayaslara kavuşması sağlanır. Sonuç olarak ağı cevabı daha iyi olacak ve çıkışın eğitim datasına tam oturması sağlanacaktır.

4.3 RBF (Radial Basis Function) Ağ Yapısı

'Radial Basis' ağı standart ileri sürümlü ağlardan sayıca daha fazla nöron içermesine rağmen, eğer yeterli miktarda eğitme vektörüne sahipse ileri sürümlü ağların yakınsamasının çok ufak kesri olan bir sürede sonuç verebilir (Demuth, 2002).

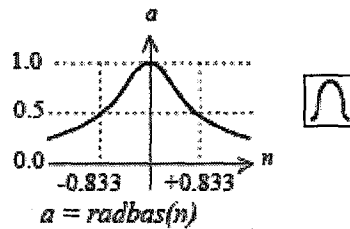


Şekil 4.9 Radial basis nöron yapısı (Demuth, 2002)

Üstte R adet girişe sahip bir ağ yapısı görülmektedir. Şekil 4.9 dikkatle incelenirse 'radbas' nöronuna gelen net giriş diğer nöron yapılarından biraz olsun farklıdır. Çünkü burada, net giriş olarak tabir ettiğimiz büyüklük, ağırlık vektörü w ile giriş vektörü p arasındaki vektörel uzaklığın bayas b katından başka bir şey değildir. Radial Basis nöronunun transfer fonksiyonu,

$$\text{radbas}(n) = \exp(-n^2) \quad (4.15)$$

şeklinde ifade edilmiştir. Şekil 4.10'da Radial basis fonksiyonu verilmektedir.

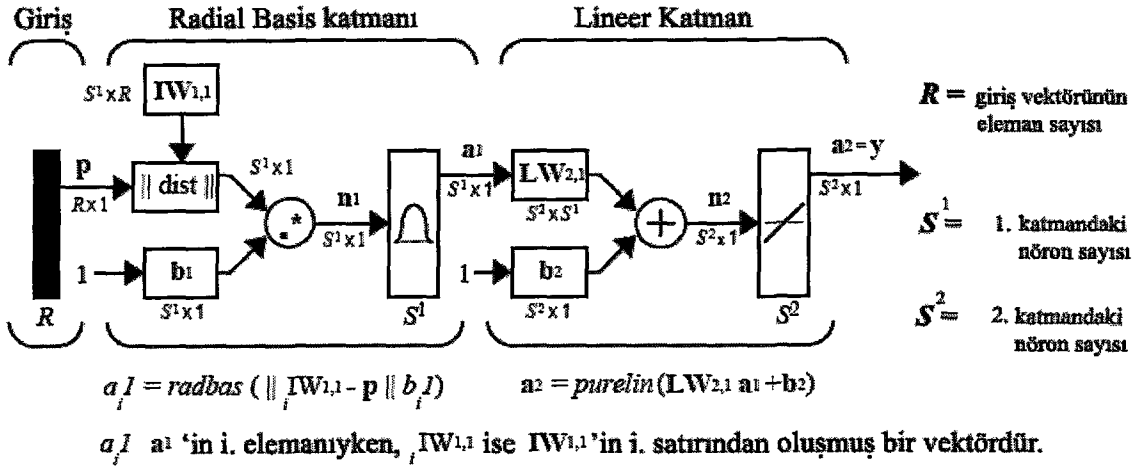


Radial Basis Fonksiyonu

Şekil 4.10 Radial basis fonksiyonu (Demuth, 2002)

Radial Basis (radbas) fonksiyonu maksimum değeri olan 1'e girişi 0 olduğu zaman ulaşır. Diğer bir deyişle w ve p arasındaki uzaklık azaldıkça, çıkışı artış gösterir. Bu sebepten ötürü radial basis fonksiyonu giriş vektörünün, ağırlık vektörüyle benzeşip benzeşmediğini

algılayan (w vektörü, p ile uyuyorsa çıkışı 1'e çekmek suretiyle) bir detektör gibi davranır. Bayas b , radbas nöronunun ayarlanabilen hassaslık derecesini göstermektedir.



Şekil 4.11 RBF ağ yapısı (Demuth, 2002)

Şekil 4.11'de yapısı görülmekte olan radbas ağı, radial basis katmanı olan S^1 ile çıkış katmanı olan lineer S^2 nöronlarından oluşmaktadır.

Bu ağ yapısına benzeyen ama bir çok açıdan daha esaslı işlem kabiliyeti gerektiren varyasyonu olan Gerçek Dizayn (Exact Design, newrbe) ağını şu dakikada incelemek doğru bir davranış olacaktır. Giriş sayısı kadar nöron oluşturmak suretiyle giriş ile çıkış arasındaki ilişkiyi tayin eder. Ama bazı durumlarda çok sayıda giriş yer alacağı için giriş çıkış deseni ilişkisini tanımlayabilmek mümkün olamayabilir.

Diğer yandan 'newrb' fonksiyonu iteratif bir şekilde ağa nöron ekleyerek toplam karesel hatanın kabul edilebilir bir değere kadar düştüğü an veya maksimum nöron sayısına erişildiği zaman duracak şekilde ayarlanabilir. Dikkat edilmesi gereken husus yayılmanın (spread) yeterince büyük olmasıdır. Böylelikle ağ, giriş uzayının üst üste geçmiş patern bölgelerine cevap verebilecektir. Fakat çok da büyük olmaması gerekmektedir zira bu durumda da nöronların neredeyse aynı şekilde çıkış vermesi sağlanmış olur. Bir çok örnek göstermiştir ki radial basis ağı, sigmoid/çizgisel yapıya sahip ileri sürümlü bir ağdan daha az nörona ihtiyaç duymaktadır.

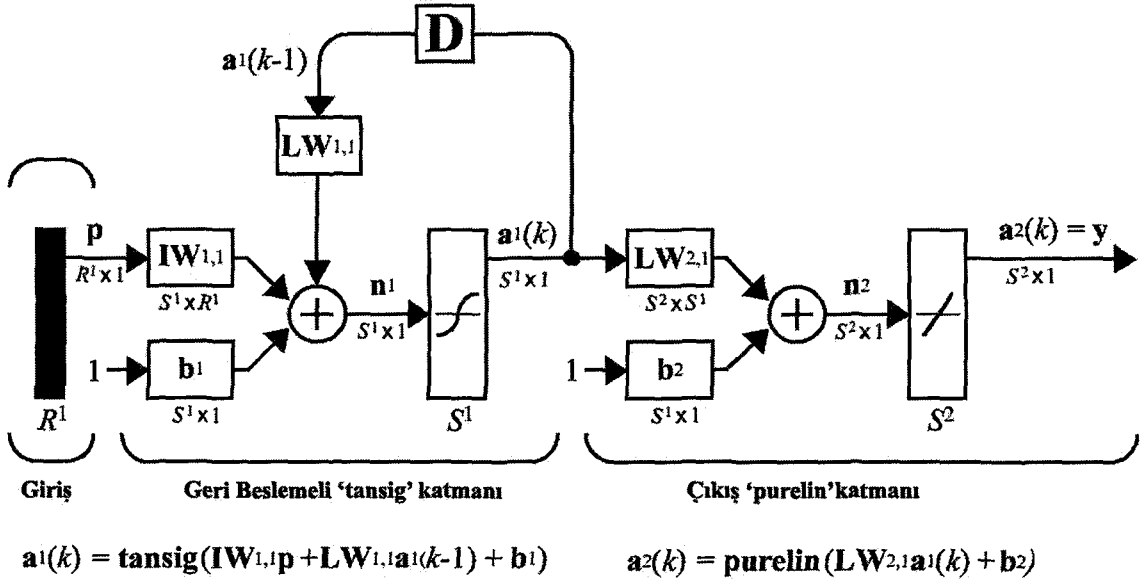
Adı geçen bu iki türün dışında 'newrbe' ile büyük benzerlik içeren bir ağ yapısı olan GRNN (Generalized Regression Networks, Genelleştirilmiş Regresyon Ağları) ağı mevcuttur. Radial Basis katmanı ve özel bir lineer katmandan ihtiva olmuş bu ağ yapısı fonksiyon yakınsamasında sıklıkla kullanılmaktadır. İkinci katmanda 'nprod' fonksiyonu kullanılarak a^1 giriş vektörünün lineer katman ağırlıkları olan $LW^{2,1}$ ile çarpımının a^1 elemanlarının toplamı ile normalize edilmesi sağlanmış olur. Bu sayede S^2 eleman sayısı n^2 eleman içeren bir

vektöre indirgenir. İlk katman tıpkı 'newrbe' ağı gibi P'nin içindeki giriş/hedef vektörü sayısı kadar nörona sahiptir. Bu ağın tasarlanmasına imkan veren komut 'newgrnn' olarak verilmiştir.

Diğer bir RBF ağ tipi ise PNN (Probabilistic Neural Network, Olasılıklı Nöral Ağlar) olarak karşımıza çıkar. PNN sınıflandırma problemlerinde kullanılan bir ağ yapısıdır. Ağa bir giriş uygulandığında, birinci katman giriş vektörü ile eğitilen giriş vektörü arasındaki uzaklığı hesaplayarak, girişin eğitim desenine ne kadar yakın olduğunu ifade eden bir vektör çıkışı verir. İkinci katman ise her sınıfa tekabül eden bu vektörleri toplayarak net bir çıkış olarak bir olasılık vektörü oluşturur. Son olarak da 'compete' (rekabet) transfer fonksiyonu bu olasılıkların maksimumunu seçerek o sınıf için '1' , diğer sınıflar için ise '0' çıkışı verir. Olasılıksal bir hesaplama kriteri içermesi bu ağ yapısının eğitim şekline bağımlı kalmasını engeller. Öte yandan her iki ağ yapısı da bir sebepten dolayı dezavantaja sahiptir ki bu hem cinslerinden daha fazla hesaplama barındırmaları yüzünden kaçınılmaz olarak daha yavaş olmalarıdır.

4.4 Elman Ağ Yapısı

Şekil 4.12'de yapısı görülmekte olan Elman ağı ilk katmanın çıkışı ile girişi arasında bir geri besleme barındırır. Bu geri beslemeli bağlantı, Elman ağının zamana bağlı değişen örüntüleri tanımasına imkan verir. Saklı katmanındaki 'tansig' nöronları ile ikinci katmandaki 'purelin' nöronları sınırlı sayıda süreksizlik içeren fonksiyonların tanınması için gerekli 'evrensel yakınsayıcı' yapısını oluşturur. Sözü geçen tanımlamanın gerçekleştirilebilmesi gizli katmandaki nöron sayısına bağlıdır. Yakınsanacak fonksiyonun kompleksliğinin yüksek olması gizli nöronların da sayısının nispi ölçüde çok olması yükümlülüğünü doğuracaktır. Diğer ağ tiplerinden farklı olarak ilk katmanda kullanılmış olan geri besleme önceki zaman basamağındaki bilgiyi tutup diğer basamakta kullanmaya yaramaktadır. Bu yüzden, iki Elman ağının, bayasları ve ağırlıkları aynı seçilmiş bile olsa, her hangi bir girişe vereceği cevapları, değişik geri besleme anlarında elde ettikleri bilgiye göre değişim gösterecektir. Diğer bir deyişle aynı geri-besleme durumlarına sahip olmadıkları için farklı sonuç vereceklerdir.



Şekil 4.12 Elman ağ yapısı (Demuth, 2002)

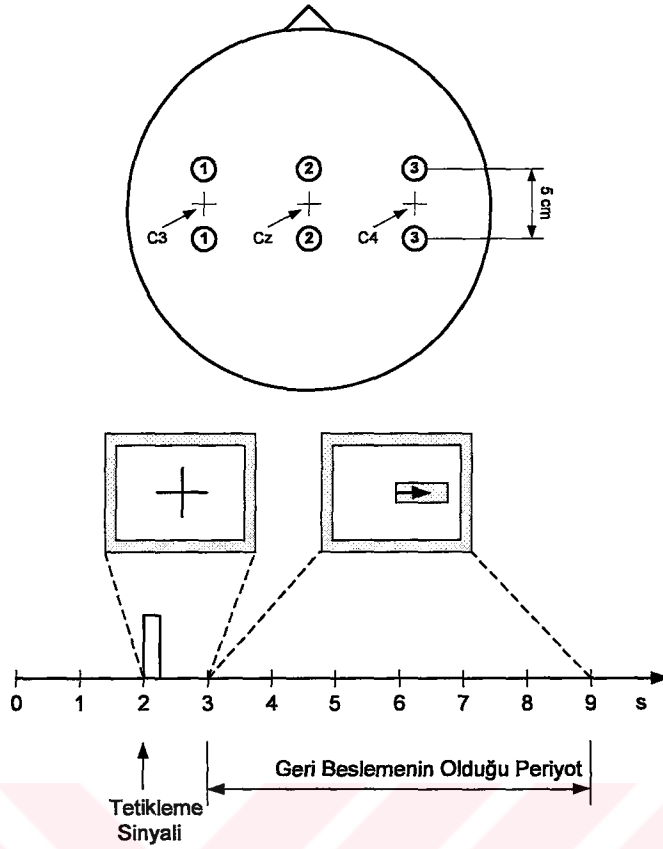
Elman ağları diğer geri beslemeli ağlar kadar güvenilir olamayabilmektedir. Bunun nedeni hem eğitimde hem de adaptasyon esnasında hata gradyeninin yaklaşık değerinin kullanılmasındandır. Eğer öğrenme işleminin doğru şekilde gerçekleşmesi isteniyorsa gizli katmanda diğer yöntemlerde kullanıldığından daha çok nöron bulunmasına dikkat edilmelidir (Diğer ileri beslemeli ağlarda 100 nöron optimal olarak kabul edilmişken, çalışmamızdaki Elman ağında 150 nöron kullanmamızın sebebi budur.).

5. EEG SİNYALLERİNİN SİMÜLASYON ORTAMINDA TANINMASI

EEG sinyallerini tanımayı amaç edinmiş bir simülasyonu yürütmek için gerek ve yeter şartlar, hali hazırda mevcut bir veri kümesinin (dataset) varlığı ile simülasyon ortamını sağlayacak bir dinamik sistemin oluşturulması olarak sayılabilir. Öncelikle incelenecek olan veri kümesinin ayrıntılarıdır.

'University of Technology Graz' (Avusturya) aracılığıyla düzenlenmiş olan 'BCI Data Classification 2003' yarışmasında 'Graz Data' adlı bir veri kümesi kullanılmıştır. Graz Data ile ilgili çalışmalar (Blanchard ve Blankertz, 2004), (Blankertz ve Müller, 2004) ve (Lem vd., 2004) kaynaklarında mevcuttur. Yarışma yürütücüsü Alois Schlögl'in izniyle edinmiş olduğumuz bu veri kümesinde yirmi beş yaşlarında bir bayanın EEG örüntüleri mevcuttur. Adı geçen veri kümesi, sestem ve her türlü dış etkiden arındırılmış özel bir hücrede, deneğin ekrandaki imlêci saęa veya sola yönlendirmesine yarayan bir geri-besleme mekanizmasını kontrol ettięi esnada kaydedilmiştir. Gözümüzün önünde canlandırabilmemiz açısından basitçe ifade etmek gerekirse, ekranda sol (ya da saę) yönü simgeleyen oku gördüğünde deneğin yapması gereken tek şey konsantrasyonunu bozmadan –ki burada bahsedilen sadece o anı düşünmek- geri besleme barını sola çektiğini düşündürmektedir. Saęa veya sola okların çıkma sırası gelişigüzel olarak belirlenmiştir (Schlögl, 2003); (Schlögl ve Keinrath, 2003).

Deney 7 seferden ibarettir. Her seferde 40 adet deneme gerçekleştirilmiştir. Deneğin her seferinde bir kaç dakika kadar dinlenmesine izin verilmek kaydıyla, deney aynı gün içersinde tamamlanmıştır. Böylelikle her biri 9 saniyelik toplam 280 deneme gerçekleştirilmiştir. EEG 128Hz ile örneklenmiş, 0.5 ve 30 Hz aralığı filtre edilmiştir. MATLAB dosya formatına uygun olarak doldurulmuş olan bu veri kümesi 3x140x1152'lik üç boyutlu bir matris şeklindedir. Burada 3 örneklenmiş olan kanalları, 140 her bir denemeyi, 1152 ise veri noktalarını (128x9) temsil etmektedir.



Şekil 5.1 Elektrot pozisyonları ve zamanlama sekansı (Schlögl, 2003)

İlk 2 saniye sükunet halinde geçerken, $t=2s$ 'de akustik bir stimulus denemenin başlangıcını belirtmektedir. Akabinde ekranda “+” işareti görünür ve 1 saniye boyunca kalır. Daha sonra $t=3s$ 'de bir ok (sağa veya sola) görünür. Denekten $t=3s$ ile $t=9s$ aralığında barı ekranda belirtilen yöne kaydırması istenir. Bar geri besleme sistemi kanal 1 (C3) ve kanal 3 (C4)'ün AAR parametrelerinin diskriminant analizi yardımı ile indirgenmiş olan tek bir çıkış parametresi ile yakından ilişkilidir. Kayıt işlemi G.tec yükseltici ve Ag/AgCl elektrotlar yardımı ile gerçekleştirilmiştir. Üç bipolar EEG kanalından C3, Cz ve C4 elektrotları üzerinden ölçüm işlemi uygulanmıştır.

5.1 MATLAB Ortamında EEG Veri Setinin Tanınması

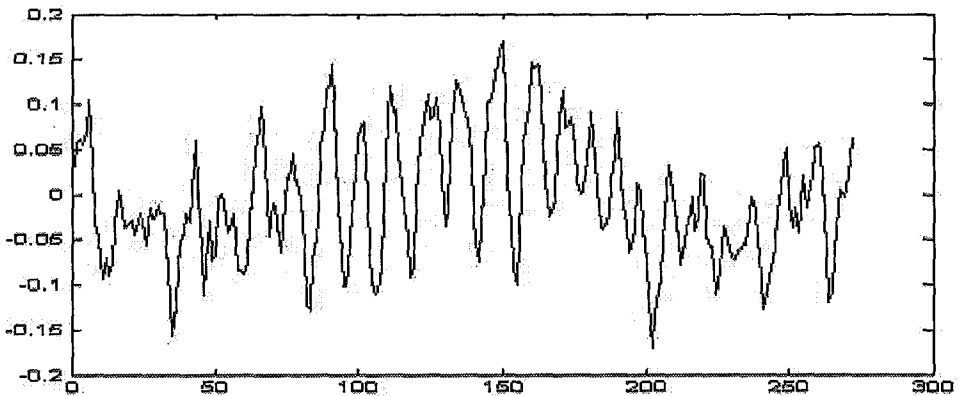
Matlab 7.0 akademik çevrelerce kabul görmüş olan sayılı simülasyon ortamlarından biridir. İçermiş olduğu alt yapı sayesinde hali hazırda mevcut olan bir takım algoritmaları ve yöntemleri kullanıcı tarafından C++ programlama lisanına yakın bir üslupta yazılabilen scriptler ile koşturmak mümkündür. Çoğu algoritmanın barındırdığı karmaşık yapı işleme mantığı kavrandığında kendisini meydana getiren parçacıkları deşifre etmeyi bile lüzumsuz kılmaktadır. Program alt yapısında kullanılmış olan blok mantığı bir çok konuda işlevselliği ön plana çıkarabilmektedir. Matlab yazılımının en can alıcı noktası da belki burasıdır. Yanında

verilmiş olan ve kendi tasarlayabildiğimiz araçlarla uygulamaya özel çözümler üretebilmek mümkündür. Açıklamalar ve örnekler sistemi anlama sürecini ciddi şekilde aşağıya çekmektedir.

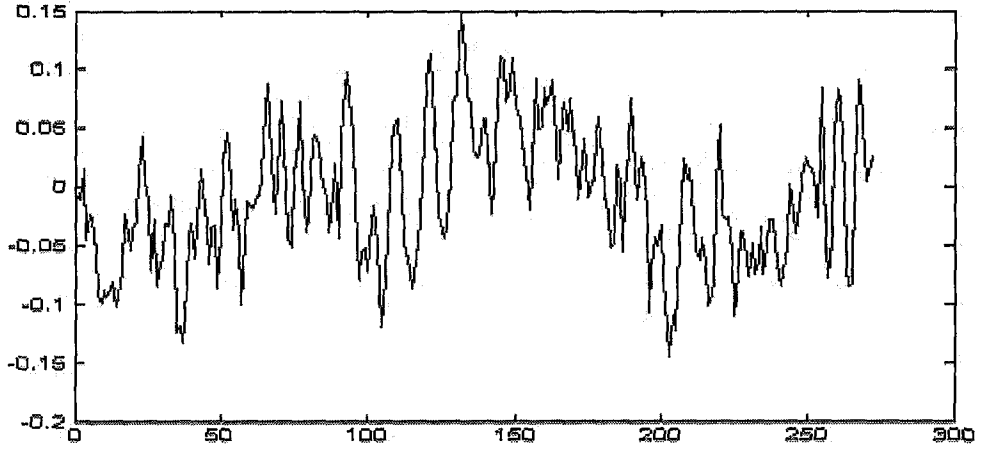
Veri setini, YSA gibi dinamik sistemlerin tanınmasını amaçlıyorsak, ilk yapmamız gereken hareket kuşkusuz bu veri setini öncelikle “kendimiz” anlamaya çalışmaktır. Zira insan zekası ve sezgisi çoğu zaman en içinden çıkılmaz gibi gözükten durumlardan sıyrılmamıza olanak vermektedir. Bu durumun temelinde anlamayla ilişkili olduğu açıktır. Eğer programcı problemi anlamıyorsa, çözüm üretmek için strateji geliştiremiyorsa, onun oluşturacağı programın sonuca ulaşacağını düşünmek boş bir umuttan ileriye gidemez. Bilgisayarlarla kıyaslandığında kusursuz olmayan inceleme kabiliyetimiz bazen bizim tek dostumuz olabilmektedir. Bu yüzden yapılması gereken ilk şey verileri bir işleme tabi tutmadan önce onların bir düzlem üzerindeki dağılımlarını grafiksel olarak incelemektir. Belki kaotik veri kalabalığında ilgilendiğimiz konuya özel bir kriter yakalayabiliriz. Burada bahsi geçen, sağa veya sola yönelim arasındaki temel EEG farklılığı olduğuna göre bu iki duruma ait fonksiyonların birlikte incelenmesi doğası hakkında çıkarım yapmamıza imkan verebilir.

Alta sağa ve sola yönelimi simgeleyen iki durumun C1 ve C3 kanallarına ait EEG dalgaları görülmektedir. Burada basitlik açısından dalganın sadece ilgilendirilmiş olan kısmı yansıtılmıştır. Dalgalar arasındaki farkı gözlemek, iki durumu birbirinden ayıran bir sistem oluşturmak için yararlıdır.

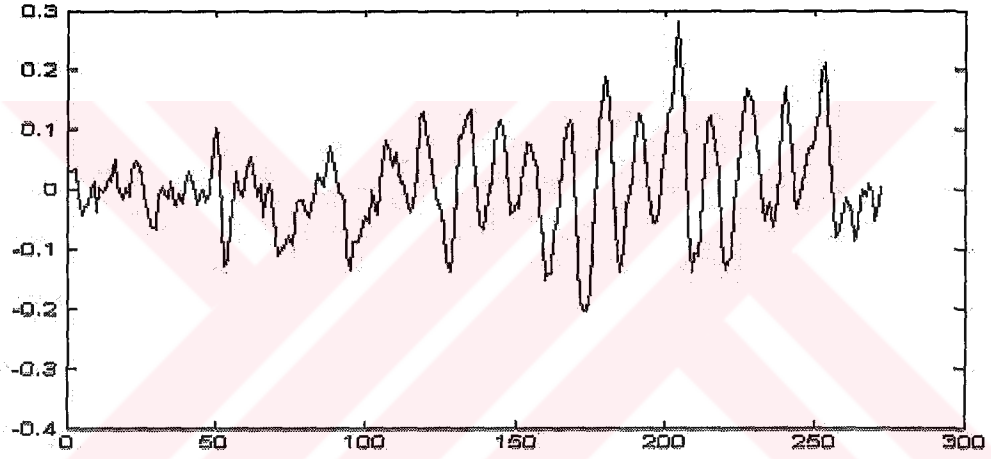
Şekil 5.2 “1” (sağ) durumunu inceleyen C1 ve C3 kanallarındaki dalga formları sırasıyla a ve b şekillerinde ifade edilmiştir.



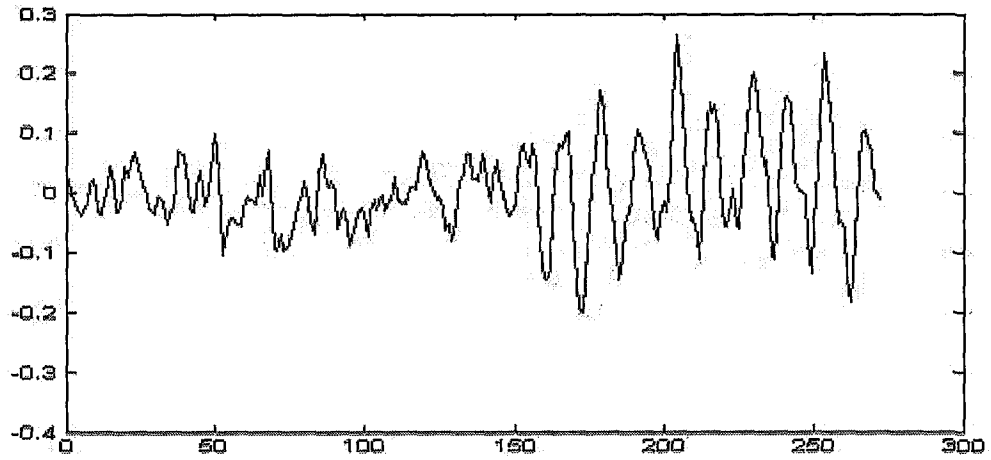
Şekil 5.2a “1” (sağ) durumunu inceleyen C1 kanalındaki dalga formu



Şekil 5.2b "1" (sağ) durumunu inceleyen C3 kanalındaki dalga formu



Şekil 5.3a "2" (sol) durumunu inceleyen C1 kanalındaki dalga formu



Şekil 5.3b "2" (sol) durumunu inceleyen C3 kanalındaki dalga formu

Şekil 5.3 “2” (sol) durumunu inceleyen C1 ve C3 kanallarındaki dalga formları sırasıyla a) ve b) şekillerinde ifade edilmiştir.

YSA’yı sınırlandıran asıl faktörler ilgilendiği verinin ne kadar basit olarak ifade edildiği ve eğitime işlemine son verilme zamanı olarak belirtilebilir. Zira temel özelliklerini tutmak kaydıyla veri ne kadar basite indirgenirse cevap süresi o kadar kısalmaktadır. Eğitime işlemine son verme anı da ağır veriyi ezberlemesine engel olmak için önem arz etmektedir. Verinin ezberlenmesi bizim amaçladıklarımıza ters düşmektedir. Çünkü yazmış olduğumuz scriptin her durumda “sağı” “sol”dan ayırabilmesini istemekteyiz. Eğer YSA’yı eğitime örüntüsüne fazlasıyla yakınsatırsak, onun test ettiğimiz örüntüyü tanıma verimi oldukça düşük olacaktır. Yani, aldığı veriyi iyice çiğnemiş yutmuş, kendi özümsemiği bilgiyle karşılaştığında 100% verime sahip olan ama diğer hiçbir durumda tanımlama gerçekleştiremeyen bir sistem elde etmiş oluruz. Bu kabul edilemez bir sonuçtur.

Beyin dalgalarını yansıtan EEG örüntüleri yapısal olarak kaotik parçacıklar barındırır (Freeman, 2003). Bu durum, klasik yöntemlerle tanımayı zorlaştırmanın ötesinde tekrarlanabilirliğe ciddi anlamda büyük bir darbe vurur. Nöron yapısının kullanılmasıyla ağır bizim çıkartamadığımız kuralları deneme yanılma yöntemiyle bulmasına imkan verilmiş olur (Haselsteiner ve Pfurtscheller, 2000). Nöronlar tekrarlanan örüntü kalıplarına rastladığında bu durumları öğrenip ileriki iterasyonlarda, ağırlıklarını yakınsama kriteri sağlanacak şekilde günceller. Ağ yapısını ve eğitime algoritmasını seçerken öncelikle arzuladığımız hızlı ve doğru bir şekilde sonuç verebilmesi olmalıdır. Denemiş olduğumuz eğitime algoritmalarında parametrelerine bağlı değişimleri incelenmiş ve bu durum iki ön işlem (Burg ve Yule-Walker yöntemleri) içinde tekrarlanmıştır. Burada amaçlanan EEG datasını yakınsayacak optimal ön işlem ve ağ tipi/eğitime algoritmasını bulabilmektir.

140 deneyin 70 adeti ağır eğitimi için kullanılırken, kalanı test etmek için saklanmıştır. Bu durumda 70 adet eğitim datası gelişigüzel olarak seçilmiş, kalanı ile test işlemi gerçekleştirilmiştir. Akabinde ağ test işleminde kullanılan data ile eğitilip, eğitim için saklanan veri ile test edilmiştir. Bu işlem iki kez tekrarlanmış, dört sonucun ortalaması alınarak gerçekçi sonuçlar elde edilmeye çalışılmıştır. Eğitim veri kümesinin 502-772. noktaları arasındaki bilgi öncelikle Yule-Walker’s Method kullanılarak PSD (Güç Spektral Yoğunluğu) formuna dönüştürülmüştür. 6. dereceden bir yakınsama kriteriyle 64 FFT uzunluğuna sahip bir çerçeveleme seçilerek dönüşüm işlemi tamamlanmıştır. Böylelikle 1152x70 tane veri elimine edilerek 66x70 adete indirgenmiş olur. Bu yayılması uygun bir değer seçilmiş bir RBF ağına, bir MLP’ye veya Elman Ağına verilmiştir. Bahsi geçen durum

Burg Yöntemi için 6. dereceden bir AR modeli kullanılarak tekrarlanmıştır. FFT uzunluğu yapılan denemeler sonucunda optimal olarak 256 bulunmuştur. Böylelikle 1152x70 tane veri 14x70 adete sıkıştırılabilmektedir. Eğitim tamamlandıktan sonra test deseniyle sınama gerçekleştirilmiştir. Sonuçları yorumlayan kriter sağı '1', solu '2' olarak göstermektedir. Bu durum ağıın çıkış katmanına binary modunda kodlanmıştır. 'tansig' 'purelin' şeklinde iki adet katmandan ihtiva olmuş bir evrensel yakınsayıcı kullanılmış olduğu için bu binary kod biraz modifiye edilerek lojik sıfır olarak tabir ettiğimiz değer '-1' ile yer değiştirilmiştir. Dolayısıyla 1 (-1 1) ile 2 ise (1 -1) ifade edilmiştir.

YSA ile eğitim ve test sistemimizi ifade eden algoritma,

- * 140 deneyin rasgele 70'ini 'Eğitim' için ayır, kalanını 'Test' için tut
- * Güç Spektral Yoğunluğunu meydana getirecek AR Model derecesini seç (6. dereceden)
- * 'Eğitim' ve 'Test'in PSD'sini hesapla
- * PSD değeri ile YSA'yı eğit
- * 'Test' ile YSA'yı test et, sonucu Sonuc1'e kaydet
- * 'Eğitim'i 'Test' ile yer değiştir
- * YSA'yı yeni 'Eğitim' değerleri ile eğit
- * YSA'yı yeni 'Test' değerleri ile test et, sonucu Sonuc2'e kaydet
- * Sonuc1 ve Sonuc2'nin ortalamasını al bunu Sonuc3'e kaydet
- * Üsttekileri bir kez tekrarla, ortalamasını al ve Sonuc4'e kaydet
- * Sonuc3 ve Sonuc4'ün ortalamasını al, Sonuc'a kaydet
- * Dur.

şeklinde ifade edilebilir.

MLP ve RBF ağ yapısını meydana getiren evrensel sınıflandırıcının gizli katmanında 100 adet nöron bulundurulmuşken Elman Ağında 150 adet nöron kullanılmıştır. Bunun sebebi Elman Ağlarının karmaşık fonksiyonların yakınsanmasında genellikle MLP akranlarından daha fazla nörona ihtiyaç duymasıdır. Bunun dışında Elman Ağında 'TRAINLM' veya 'TRAINRP' gibi büyük basamaklarla işleyen algoritmaların kullanılması sakıncalı bulunduğu için bu bahsi geçen yöntemler kullanılmamıştır. Adı geçen algoritmalar tarafından yürütülen gradyen performansının, Elman Ağında varolan gecikmeler yüzünden, sadece yaklaşık değeri kullanılabilmekte; bu da büyük basamaklı algoritmalar için öğrenmeyi zor kılmaktadır.

‘newcf’ ve Elman Ağlarında kullanılan parametreler Çizelge 5.1’de en yüksek verim elde edilmiş olanlar ile aynıdır (Örneğin, ‘newelm’ ağında ‘trainoss’ eğitim algoritmasının parametreleri, Çizelge 5.1’deki ‘trainoss’ eğitim algoritmasına ayrılmış kısımdaki en yüksek verime sahip satırdaki parametrelerle aynıdır).Çizelgelerde verilmiş olanların haricindeki tüm parametreler ‘default’ değerleriyle kullanılmıştır.

Çizelgelerde geçen parametreler, ‘lr’ (öğrenme parametresi), ‘lr_inc’ (öğrenme parametresinin artış oranı), ‘lr_dec’ (öğrenme parametresinin azaltılma oranı), ‘mc’ (momentum katsayısı), ‘delt_inc’ (ağırlık değişimi artış miktarı), ‘delt_dec’ (ağırlık değişimi azalış miktarı), ‘scal_tol’ (deltaya bölüldüğünde çizgisel arama toleransı bulunur), ‘alpha’ (performansta kabul edilebilir bir düşüşe imkan veren katsayı), ‘beta’ (arama adımlarının yeteri büyüklükte olmasını sağlayan katsayı), ‘delta’ (aranan aralıktaki adım büyüklüğü), ‘gama’ (performanstaki ufak değişimlerin etkisini engellemek için var olan parametre), ‘sigma’ (ikinci türev yaklaşık değeri için ağırlıktaki değişimi belirleyen büyüklük), ‘lambda’ (Hessian’ın sonsuzluk değerini regüle eden parametre), ‘mu’ (marquardt ayarlama parametresi), ‘mu_dec’ (mu azaltma parametresi), ‘mu_inc’ (mu artırma parametresi), ‘mem_reduc’ (hafıza/hız arasındaki dengeyi sağlayan faktör, default 1), ‘spread’ (Radial Basis fonksiyonlarının yayılım değerleri), ‘MN’ (kullanılacak maksimum nöron sayısı), ‘DF’ (iki gösterim arasında eklenecek nöron sayısı), ‘BTF’ (Geri besleme ağının eğitim algoritması), ‘BLF’ (Geri besleme ağının ağırlık/bayas öğrenme fonksiyonu), ‘learngd’ (gradient descent ağırlık/bayas öğrenme fonksiyonu), ‘learnghm’ (momentumlu gradient descent ağırlık/bayas öğrenme fonksiyonu), ‘PF’ (performans fonksiyonu), ‘MSE’ (ortalama karesel hata performans fonksiyonu), ‘MSEREG’ (regülasyonlu ortalama karesel hata performans fonksiyonu) olarak sıralanmıştır.

Çizelge 5.1 incelendiğinde en yüksek verimin ‘traincgb’ algoritması ile elde edilmiş olduğu görülmektedir. Gerek hızı gerek rasgele örüntüleri tanıma performansındaki başarısı açısından kabul edilebilir bir değere sahiptir. RBF ağlarından ‘newpnn’ uygun bir yayılım değeri seçildiğinde hızlı ve iyi bir yaklaşım yapabildiğini kanıtlamıştır. ‘newcf’ ağı ‘traingdx’ eğitim algoritması, ‘learngd’ öğrenme yöntemi ve ‘MSE’ performans fonksiyonu ile oldukça nitelikli bir sonuç elde edilmesine imkan vermiştir.

Çizelge 5.2’de Burg Yöntemi ön işlemi kullanılarak Çizelge 5.1’de yapılmış olanlar tekrarlanmıştır. İki durum karşılaştırıldığında dikkate değer unsur, Yule-Walker ön işleminin, Burg Yöntemine kıyasla çok daha başarılı sonuçlar çıkarabilmemize imkan vermiş olmasıdır. Neredeyse tüm eğitim algoritmalarında Yule-Walker AR Yöntemi, Burg Yönteminden daha

iyi sonuçlar vermiştir. Bunun sebebinin Yule-Walker Yönteminin oto-korelasyon fonksiyonunu hesaplarken, Burg Yönteminin bu fonksiyonu hesaplamayıp, yansıma katsayılarını direkt olarak bulmasından kaynaklandığı düşünülmektedir. Yule-Walker Yöntemi ile irdelenen bilginin YSA ile daha doğru bir uyuşum içerisinde olduğu kabul edilmiştir.

Eğer dikkat edilirse verinin yaklaşık olarak 1.7 saniyelik kısmının incelenmesi sonucunda cevap verilmektedir. Gerçek bir kontrol sisteminde bu uzunlukta bir inceleme süresi pek olanaklı gözükmemektedir. Örneğin; bir elektronik protezi hareket ettirecek EEG sinyali algılanma hızının saniyede bir sembol olması kolu çevirmeyi büyük ölçüde zorlaştıracaktır. Diğer yandan, eğer bu hız saniyede 16 sembolün üzerindeyse istenilen durumu sağlamak mümkün olabilir. O yüzden Çizelge 5.1’de verilmiş olan düzeneklerden en iyi performansa sahip bir kaçı alınarak tekrar deneyler yapılmış ve sonuçları Çizelge 5.3’de irdelenmiştir. Kullanılan nöron sayısı gene MLP için 100, Elman ağı için ise 150 olarak alınmıştır. Bu kısımda amaçlanan örnekleme süresinin verime etkisini irdelemektir.

Gerçek zamanlı kontrol uygulamaları göz önüne alındığında 0.05-0.5s aralığındaki örnekleme süresine sahip formasyonlar nispeten iyi performansa sahiptir. Dikkate değer bir husus ise örnekleme süresinin azalmasının performansı olumsuz yönde etkilediğidir. YSA’nın az miktarda örnekten yola çıkarak bir yakınsama yapmaya mecbur bırakılması yüzünden bahsedilen durumun gerçekleştiği düşünülmektedir. Çizelge 5.3’te Yule-Walker Güç Spektral Yoğunluğu kullanılmış ve tüm parametreler, Çizelge 5.1’de yanlarına ‘*’ konulmuş olanlar (en yüksek verim elde edilmiş değerler) ile aynı seçilmiştir. Çizelge 5.3 incelendiğinde ‘traincgb’ algoritması ile eğitilen bir YSA’nın düşük örnekleme süresinde de ortalama bir sonuç verdiği görülmüştür. Fakat bu bile gerçekçi bir kontrol sistemini maniple edecek yeterliliğe sahip olarak düşünülemez. Örnekleme süresi azaldıkça yararlı (gürültü) / (yararlı bilgi) oranı yükselişe geçmekte, keskin bir kontrol stratejisi üretilmesini zorlaştırmaktadır. Söz konusu durum bilgisayar veya yazılımın yeterliliğinden çok, insan düşünce sisteminin ‘bulanık’ yapısından kaynaklanmaktadır. Sağa veya sola yönelimi hayal eden bireyin bu durumu hangi zaman diliminde ne derece doğru yaptığı bir o kadar önemlidir. İncelenen kesimde düşüncelerin doğru şekilde odaklanıp odaklanmadığı, şahısın aklındaki imgeye ne kadar bağlı kaldığı (gürültü) / (yararlı bilgi) oranını etkileyen ana unsurlardır. Gene de örnekleme hızının yüksek olması, bir takım tolere edici çözümler üretilbilmesine de imkan verebilir. Eğer kıstasların dışında yer alan bir durumla karşılaşırsa (örneğin alfa beta veya gama aktivitelerinde aşırı bir artış gibi), yazılım, yanlış bir cevap üretimini engellemek için örnekleme hızını düşürerek, (gürültü) / (yararlı bilgi) oranını yükseltmeyi amaçlayabilir. Fakat

sözü geçen yazılımsal çözümün, ana kontrol planına, hesaplama süresine yansiyacak ek bir külfet getireceği gözden kaçmamalıdır.

Çizelge 5.1 Yule-Walker ön işleminden geçirilmiş verilerin YSA algoritmalarıyla eğitim sonuçları

Çizelge 5.1a) traingda

lr	lr_inc	Lr_dec	verim (%)
0.1	1.05	0.7	83.2
0.01	1.05	0.7	83.9
0.001	1.05	0.7	82.7
0.1*	1.2*	0.5*	86.4*
0.01	1.2	0.5	84.2
0.001	1.2	0.5	85.5
0.1	1.5	0.6	80.0
0.01	1.5	0.6	83.0
0.001	1.5	0.6	80.3

Çizelge 5.1b) traingdx

lr	Mc	verim (%)
0.1	0.9	86.8
0.01	0.9	83.0
0.001	0.9	84.3
0.1	0.55	82.8
0.01	0.55	81.8
0.001	0.55	85.0
0.1*	0.25*	87.1*
0.01	0.25	81.3
0.001	0.25	85.3

Çizelge 5.1c) trainrp

lr	delt_inc	Delt_dec	verim (%)
0.1*	1.2*	0.5*	87.0*
0.01	1.2	0.5	83.6
0.001	1.2	0.5	86.1
0.1	1.5	0.6	84.6
0.01	1.5	0.6	82.8
0.001	1.5	0.6	86.1
0.1	1.7	0.8	85.0
0.01	1.7	0.8	86.4
0.001	1.7	0.8	81.8

Çizelge 5.1d. traincgf

scal_tol	alpha	beta	delta	gama	verim (%)
20	0.001	0.1	0.01	0.1	85.0
10	0.001	0.1	0.01	0.1	83.2
40	0.001	0.1	0.01	0.1	84.6
20	0.01	0.1	0.01	0.1	83.0
10	0.01	0.1	0.01	0.1	84.6
40	0.01	0.1	0.01	0.1	86.4
20	0.001	0.5	0.01	0.1	85.0
10*	0.001*	0.5*	0.01*	0.1*	88.9*
40	0.001	0.5	0.01	0.1	85.3
20	0.001	0.1	0.1	0.1	87.9
10	0.001	0.1	0.1	0.1	81.4
40	0.001	0.1	0.1	0.1	83.2
20	0.001	0.1	0.01	0.5	85.0
10	0.001	0.1	0.01	0.5	84.3
40	0.001	0.1	0.01	0.5	85.0

Çizelge 5.1e traincgp

scal_tol	Alpha	beta	delta	gama	Verim (%)
20	0.001	0.1	0.01	0.1	87.9
10	0.001	0.1	0.01	0.1	84.3
40	0.001	0.1	0.01	0.1	81.8
20	0.01	0.1	0.01	0.1	85.7
10	0.01	0.1	0.01	0.1	85.0
40	0.01	0.1	0.01	0.1	86.1
20	0.001	0.5	0.01	0.1	84.3
10	0.001	0.5	0.01	0.1	81.8
40*	0.001*	0.5*	0.01*	0.1*	88.9*
20	0.001	0.1	0.1	0.1	87.5
10	0.001	0.1	0.1	0.1	84.6
40	0.001	0.1	0.1	0.1	87.9
20	0.001	0.1	0.01	0.5	83.9
10	0.001	0.1	0.01	0.5	86.1
40	0.001	0.1	0.01	0.5	83.6

Çizelge 5.1f trainbfg

scal_tol	alpha	beta	delta	gama	verim (%)
20	0.001	0.1	0.01	0.1	79.6
10	0.001	0.1	0.01	0.1	83.6
40	0.001	0.1	0.01	0.1	87.5
20	0.01	0.1	0.01	0.1	83.2
10	0.01	0.1	0.01	0.1	83.5
40	0.01	0.1	0.01	0.1	82.9
20	0.001	0.5	0.01	0.1	84.3
10	0.001	0.5	0.01	0.1	86.4
40	0.001	0.5	0.01	0.1	86.8
20*	0.001*	0.1*	0.1*	0.1*	87.9*
10	0.001	0.1	0.1	0.1	87.1
40	0.001	0.1	0.1	0.1	83.6
20	0.001	0.1	0.01	0.5	85.0
10	0.001	0.1	0.01	0.5	85.4
40	0.001	0.1	0.01	0.5	83.9

Çizelge 5.1g traincgb

scal_tol	Alpha	beta	delta	gama	verim (%)
20	0.001	0.1	0.01	0.1	83.6
10	0.001	0.1	0.01	0.1	86.8
40	0.001	0.1	0.01	0.1	86.4
20	0.01	0.1	0.01	0.1	85.4
10	0.01	0.1	0.01	0.1	84.6
40*	0.01*	0.1*	0.01*	0.1*	89.6*
20	0.001	0.5	0.01	0.1	83.6
10	0.001	0.5	0.01	0.1	83.9
40	0.001	0.5	0.01	0.1	87.1
20	0.001	0.1	0.1	0.1	81.1
10	0.001	0.1	0.1	0.1	81.4
40	0.001	0.1	0.1	0.1	88.9
20	0.001	0.1	0.01	0.5	80.3
10	0.001	0.1	0.01	0.5	83.1
40	0.001	0.1	0.01	0.5	85.0

Çizelge 5.1 h) trainoss

scal_tol	alpha	beta	delta	gama	verim (%)
20	0.001	0.1	0.01	0.1	79.9
10	0.001	0.1	0.01	0.1	83.9
40	0.001	0.1	0.01	0.1	81.4
20	0.01	0.1	0.01	0.1	83.6
10	0.01	0.1	0.01	0.1	86.8
40	0.01	0.1	0.01	0.1	85.4
20*	0.001*	0.5*	0.01*	0.1*	89.3*
10	0.001	0.5	0.01	0.1	83.9
40	0.001	0.5	0.01	0.1	83.9
20	0.001	0.1	0.1	0.1	80.7
10	0.001	0.1	0.1	0.1	87.1
40	0.001	0.1	0.1	0.1	85.7
20	0.001	0.1	0.01	0.5	86.1
10	0.001	0.1	0.01	0.5	86.8
40	0.001	0.1	0.01	0.5	81.8

Çizelge 5.1j trainseg

sigma	lambda	verim (%)
5.0e-5	5.0e-7	83.6
5.0e-6*	5.0e-7*	87.9*
5.0e-7	5.0e-7	84.3
5.0e-4	5.0e-7	83.6
5.0e-3	5.0e-7	82.1
5.0e-5	5.0e-6	83.6
5.0e-5	5.0e-5	78.6
5.0e-5	5.0e-8	83.2
5.0e-5	5.0e-9	84.3

Çizelge 5.1k trainlm

mu	mu_dec	mu_inc	mem_reduc	verim (%)
0.001	0.1	10	1	85.7
0.01	0.1	10	1	82.9
0.0001	0.1	10	1	86.4
0.005	0.1	10	1	86.8
0.001	0.01	10	1	83.9
0.001	0.001	10	1	82.5
0.001	0.5	10	1	84.6
0.001	0.1	5	1	86.4
0.001	0.1	20	1	83.2
0.001*	0.1*	15*	1*	87.5*

Çizelge 5.11 trainbr

mu	mu_dec	mu_inc	mem_reduc	verim (%)
0.005	0.1	10	1	82.5
0.05	0.1	10	1	82.1
0.0005	0.1	10	1	83.3
0.001	0.1	10	1	85.4
0.005*	0.01*	10*	1*	86.8*
0.005	0.001	10	1	84.8
0.005	0.5	10	1	84.0
0.005	0.1	5	1	81.1
0.005	0.1	20	1	85.1
0.005	0.1	15	1	86.3

Çizelge 5.1m newrb

spread	MN	DF	verim (%)
1	70	5	67.5
1	100	1	67.5
0.5	100	1	62.9
0.25	100	1	66.8
0.125	100	1	60.0
0.0125	100	1	81.1
0.0060	100	1	65.7
0.00125	100	1	51.0
0.0124	100	1	80.0
1.5	100	1	61.7
0.02	100	1	70.4
2	100	1	60.4
3	100	1	70.5
0.0129*	100*	1*	81.4*

Çizelge 5.1n newrbe, newgrnn, newpnn

newrbe		newgrnn		newpnn	
spread	verim (%)	spread	verim (%)	spread	verim (%)
1	63.0	1	57.9	1	45.7
0.5	65.5	0.5	45.4	0.5	52.5
0.25	64.6	0.25	60.7	0.25	62.1
0.125	63.2	0.125	57.5	0.125	62.9
0.0125	60.7	0.0125	84.3	0.0125	82.5
0.0060	56.1	0.0060	85.0	0.0060	81.4
0.00125	45.6	0.00125	84.6	0.00125	85.0
0.0124	62.5	0.0124*	86.8*	0.0124*	87.5*
1.5	58.7	0.0123	84.6	0.0123	84.3
0.02	64.6	1.5	53.6	0.0122	81.8
2	66.1	0.02	85.0	1.5	60.8
3	59.3	2	45.4	0.02	77.9
0.0129	61.8	3	45.5	2	46.1
0.0135	63.3	0.0129	82.1	3	47.3
2500*	82.8*	0.0135	79.3	0.0129	81.4
2480	80.1	2500	47.1	0.0135	85.7
2510	80.3	1	57.9	2500	45.0

Çizelge 5.1o newcf

BTF	BLF	PF	verim (%)
traincgb	learnrdm	mse	86.4
traincgb	learnrd	mse	86.8
traincgb	learnrdm	msereg	85.0
traincgb	learnrd	msereg	86.8
trainlm	learnrdm	mse	84.2
trainlm	learnrd	mse	83.7
trainlm	learnrdm	msereg	82.9
trainlm	learnrd	msereg	83.7
trainrp	learnrdm	mse	86.5
trainrp	learnrd	mse	86.4
trainrp	learnrdm	msereg	84.3
trainrp	learnrd	msereg	85.0
trainoss	learnrdm	mse	85.0
trainoss	learnrd	mse	82.1
trainoss	learnrdm	msereg	76.9
trainoss	learnrd	msereg	82.2
traingdx	learnrdm	mse	85.9
traingdx*	Learnrd*	mse*	87.9*
traingdx	learnrdm	msereg	85.6
traingdx	learnrd	msereg	83.9

Çizelge 5.1p newelm

BTF	BLF	PF	verim (%)
traincgb	learnadm	mse	82.1
traincgb	learnad	mse	83.6
traincgb	learnadm	msereg	84.6
traincgb	learnad	msereg	83.9
trainoss	learnadm	mse	78.6
trainoss	learnad	mse	87.5
trainoss	learnadm	msereg	83.2
trainoss	learnad	msereg	84.3
traingdx	learnadm	mse	84.8
traingdx*	learnad*	mse*	87.9*
traingdx	learnadm	msereg	80.2
traingdx	learnad	msereg	78.5

Çizelge 5.2 Burg ön işleminden geçirilmiş verilerin YSA algoritmalarıyla eğitim sonuçları**Çizelge 5.2a traingda**

lr	Lr_inc	lr_dec	verim (%)
0.1	1.05	0.7	80.6
0.01	1.05	0.7	78.2
0.001	1.05	0.7	78.6
0.1*	1.2*	0.5*	81.4*
0.01	1.2	0.5	79.2
0.001	1.2	0.5	79.5
0.1	1.5	0.6	80.0
0.01	1.5	0.6	74.3
0.001	1.5	0.6	80.0

Çizelge 5.2b) traingdx

lr	Mc	verim (%)
0.1	0.9	78.8
0.01	0.9	78.0
0.001	0.9	78.5
0.1	0.55	76.6
0.01	0.55	78.5
0.001*	0.55*	82.0*
0.1	0.25	79.9
0.01	0.25	78.8
0.001	0.25	81.5

Çizelge 5.2c trainrp

lr	Delt_inc	delt_dec	verim (%)
0.1	1.2	0.5	78.0
0.01*	1.2*	0.5*	80.1*
0.001	1.2	0.5	77.8
0.1	1.5	0.6	79.4
0.01	1.5	0.6	75.0
0.001	1.5	0.6	75.8
0.1	1.7	0.8	71.4
0.01	1.7	0.8	79.0
0.001	1.7	0.8	76.8

Çizelge 5.2d traincgf

scal_tol	alpha	beta	delta	gama	verim (%)
20	0.001	0.1	0.01	0.1	81.4
10	0.001	0.1	0.01	0.1	80.4
40	0.001	0.1	0.01	0.1	81.4
20	0.01	0.1	0.01	0.1	82.4
10*	0.01*	0.1*	0.01*	0.1*	83.8*
40	0.01	0.1	0.01	0.1	78.4
20	0.001	0.5	0.01	0.1	80.9
10	0.001	0.5	0.01	0.1	81.7
40	0.001	0.5	0.01	0.1	82.9
20	0.001	0.1	0.1	0.1	80.5
10	0.001	0.1	0.1	0.1	82.3
40	0.001	0.1	0.1	0.1	79.7
20	0.001	0.1	0.01	0.5	81.4
10	0.001	0.1	0.01	0.5	82.8
40	0.001	0.1	0.01	0.5	80.5

Çizelge 5.2e traincgp

scal_tol	alpha	beta	delta	gama	Verim (%)
20	0.001	0.1	0.01	0.1	84.5
10	0.001	0.1	0.01	0.1	81.4
40	0.001	0.1	0.01	0.1	80.5
20	0.01	0.1	0.01	0.1	79.3
10	0.01	0.1	0.01	0.1	78.6
40	0.01	0.1	0.01	0.1	84.1
20	0.001	0.5	0.01	0.1	82.5
10	0.001	0.5	0.01	0.1	83.5
40	0.001	0.5	0.01	0.1	81.4
20*	0.001*	0.1*	0.1*	0.1*	85.0*
10	0.001	0.1	0.1	0.1	82.5
40	0.001	0.1	0.1	0.1	83.5
20	0.001	0.1	0.01	0.5	82.6
10	0.001	0.1	0.01	0.5	81.5
40	0.001	0.1	0.01	0.5	84.3

Çizelge 5.2f trainbfg

scal_tol	alpha	beta	delta	gama	verim (%)
20	0.001	0.1	0.01	0.1	83.4
10	0.001	0.1	0.01	0.1	78.5
40	0.001	0.1	0.01	0.1	79.6
20	0.01	0.1	0.01	0.1	83.5
10	0.01	0.1	0.01	0.1	78.6
40	0.01	0.1	0.01	0.1	82.8
20	0.001	0.5	0.01	0.1	81.5
10	0.001	0.5	0.01	0.1	84.4
40	0.001	0.5	0.01	0.1	80.5
20	0.001	0.1	0.1	0.1	81.7
10	0.001	0.1	0.1	0.1	83.8
40*	0.001*	0.1*	0.1*	0.1*	84.5*
20	0.001	0.1	0.01	0.5	82.5
10	0.001	0.1	0.01	0.5	83.4
40	0.001	0.1	0.01	0.5	83.3

Çizelge 5.2g traincgb

scal_tol	alpha	beta	Delta	gama	verim (%)
20	0.001	0.1	0.01	0.1	82.1
10	0.001	0.1	0.01	0.1	83.5
40	0.001	0.1	0.01	0.1	81.5
20*	0.01*	0.1*	0.01*	0.1*	84.3*
10	0.01	0.1	0.01	0.1	80.1
40	0.01	0.1	0.01	0.1	84.2
20	0.001	0.5	0.01	0.1	82.2
10	0.001	0.5	0.01	0.1	79.5
40	0.001	0.5	0.01	0.1	82.1
20	0.001	0.1	0.1	0.1	83.7
10	0.001	0.1	0.1	0.1	78.4
40	0.001	0.1	0.1	0.1	79.3
20	0.001	0.1	0.01	0.5	82.4
10	0.001	0.1	0.01	0.5	81.4
40	0.001	0.1	0.01	0.5	79.6

Çizelge 5.2h trianoss

Scal_tol	alpha	beta	Delta	gama	verim (%)
20	0.001	0.1	0.01	0.1	78.6
10	0.001	0.1	0.01	0.1	81.4
40	0.001	0.1	0.01	0.1	84.6
20	0.01	0.1	0.01	0.1	84.5
10	0.01	0.1	0.01	0.1	83.2
40	0.01	0.1	0.01	0.1	85.4
20	0.001	0.5	0.01	0.1	85.7
10*	0.001*	0.5*	0.01*	0.1*	86.8*
40	0.001	0.5	0.01	0.1	82.8
20	0.001	0.1	0.1	0.1	81.2
10	0.001	0.1	0.1	0.1	83.3
40	0.001	0.1	0.1	0.1	83.5
20	0.001	0.1	0.01	0.5	84.4
10	0.001	0.1	0.01	0.5	81.4
40	0.001	0.1	0.01	0.5	84.3

Çizelge 5.2j trainscg

Sigma	Lambda	verim (%)
5.0e-5	5.0e-7	71.4
5.0e-6	5.0e-7	73.5
5.0e-7	5.0e-7	70.5
5.0e-4	5.0e-7	69.5
5.0e-3	5.0e-7	78.6
5.0e-5	5.0e-6	73.5
5.0e-5	5.0e-5	74.7
5.0e-5*	5.0e-8*	81.4*
5.0e-5	5.0e-9	77.5

Çizelge 5.2k trainlm

mu	mu_dec	mu_inc	mem_reduc	verim (%)
0.001	0.1	10	1	81.5
0.01	0.1	10	1	83.3
0.0001	0.1	10	1	81.2
0.005	0.1	10	1	83.2
0.001	0.01	10	1	83.5
0.001	0.001	10	1	79.7
0.001*	0.5*	10*	1*	84.1*
0.001	0.1	5	1	80.8
0.001	0.1	20	1	82.4
0.001	0.1	15	1	83.0

Çizelge 5.2l trainbr

mu	mu_dec	mu_inc	mem_reduc	verim (%)
0.005	0.1	10	1	81.4
0.05	0.1	10	1	78.2
0.0005	0.1	10	1	82.6
0.001	0.1	10	1	79.8
0.005	0.01	10	1	81.0
0.005*	0.001*	10*	1*	83.7*
0.005	0.5	10	1	82.1
0.005	0.1	5	1	78.3
0.005	0.1	20	1	76.2
0.005	0.1	15	1	78.3

Çizelge 5.2m newrb

spread	MN	DF	verim (%)
1	70	5	65.5
1	100	1	68.4
0.5	100	1	65.5
0.25	100	1	64.8
0.125	100	1	58.2
0.0125	100	1	71.6
0.0060	100	1	67.4
0.00125	100	1	49.5
0.0124*	100*	1*	79.6*
1.5	100	1	65.6
0.02	100	1	69.5
2	100	1	58.5
3	100	1	72.6
0.0129	100	1	73.7

Çizelge 5.2n newrbe, newgrnn, newpnn

newrbe		Newgrnn		newpnn	
spread	verim (%)	spread	verim (%)	spread	verim (%)
1	65.5	1	48.5	1	48.7
0.5	68.5	0.5	42.1	0.5	53.4
0.25	62.1	0.25	55.2	0.25	63.4
0.125	63.3	0.125	59.2	0.125	58.8
0.0125	58.7	0.0125	63.4	0.0125	78.5
0.0060	61.7	0.0060*	75.4*	0.0060	77.8
0.00125	45.8	0.00125	74.4	0.00125	81.4
0.0124	68.3	0.0124	73.4	0.0124	80.5
1.5	78.3	0.0123	74.4	0.0123	77.7
0.02	62.2	1.5	42.4	0.0122	71.5
2	53.4	0.02	74.4	1.5	65.4
3	55.0	2	41.5	0.02	68.2
0.0129*	82.5*	3	44.3	2	65.4
0.0135	65.4	0.0129	63.2	3	67.3
2500	80.8	0.0135	66.3	0.0129	80.3
2480	80.2	2500	46.9	0.0135*	81.3*
2510	80.2	1	48.5	2500	40.5

Çizelge 5.2o newcf

BTF	BLF	PF	verim (%)
traincgb	learngdm	mse	81.4
traincgb	learngd	mse	80.9
traincgb	learngdm	msereg	79.2
traincgb	learngd	msereg	83.4
trainlm	learngdm	mse	82.8
trainlm	learngd	mse	80.8
trainlm	learngdm	msereg	81.4
trainlm	learngd	msereg	81.7
trainrp	learngdm	mse	83.5
trainrp	learngd	mse	82.8
trainrp	learngdm	msereg	83.5
trainrp	learngd	msereg	84.2
trainoss	learngdm	mse	82.2
trainoss	learngd	mse	74.5
trainoss	learngdm	msereg	73.4
trainoss	learngd	msereg	77.1
traingdx	learngdm	mse	78.5
traingdx	learngd	mse	82.4
traingdx*	learngdm*	msereg*	84.4*
traingdx	learngd	msereg	83.5

Çizelge 5.2p newelm

BTF	BLF	PF	verim (%)
traincgb	learngdm	mse	79.7
traincgb	learngd	mse	77.4
traincgb	learngdm	msereg	82.6
traincgb	learngd	msereg	83.2
trainoss	learngdm	mse	77.2
trainoss	learngd	mse	79.5
trainoss*	learngdm*	msereg*	83.4*
trainoss	learngd	msereg	82.4
traingdx	learngdm	mse	79.6
traingdx	learngd	mse	77.6
traingdx	learngdm	msereg	78.4
traingdx	learngd	msereg	82.1

Çizelge 5.3 Verimin örnekleme süresine göre değişimi

algoritmalar	örnekleme süresi (s)	ortalama verim (%)
traingdx	0.05	75.4
traingdx	0.1	78.2
traingdx	0.2	78.5
traingdx	0.5	81.4
traincgb	0.05	79.5
traincgb	0.1	82.4
traincgb	0.2	82.8
traincgb	0.5	85.6
trainoss	0.05	78.3
trainoss	0.1	81.4
trainoss	0.2	83.2
trainoss	0.5	84.8
trainlm	0.05	75.5
trainlm	0.1	78.0
trainlm	0.2	82.4
trainlm	0.5	83.4
newpnn	0.05	77.8
newpnn	0.1	80.9
newpnn	0.2	83.4
newpnn	0.5	84.7
newcf	0.05	76.5
newcf	0.1	78.5
newcf	0.2	80.4
newcf	0.5	82.5
newelm	0.05	77.4
newelm	0.1	81.4
newelm	0.2	84.3
newelm	0.5	84.5

6. SONUÇLAR VE ÖNERİLER

Beyinsel aktiviteleri yansıtan EEG yapısal olarak kaotik parçacıklar barındırır. Bu parçaların bir kısmı nefes alma yutkunma, göz kırpmaya gibi fiziksel faaliyetleri yansıtırken, büyük bir bölümü ise ihtiyaç duyulan düşünsel işlemlerin çıktısı niteliğindedir. Gerekli ayıklama işlemlerinin ardından bu ham bilgiyi temsil edecek bir metot geliştirmemiz zorunludur. Bu tezde ve bir çok uygulamada önce EEG datasının uygun AR derecesinde güç spektral yoğunluğu hesaplanmıştır. Böylelikle ham bilgi yapay sinir ağının daha kolay başa çıkabileceği bir formata indirgenmiştir. İki gizli katmana sahip ileri beslemeli bir YSA, karesel hata istenen belirli bir değerin altına düşünceye kadar eğitilmiştir. Elde edilen sonuçlar göstermiştir ki, 'traincgb' algoritmasıyla eğitilen bir MLP en yüksek verimi vermektedir. RBF ağının verdiği sonucun güvenilir olmadığı gözlenmiştir. Zira giriş bilgisine, ön işleme ve verinin incelenen kesimine fazlasıyla bağımlıdır. Bu nedensellik göz önüne alındığında optimal çalışma 'traingdx' ile sağlanmıştır.

Gerçek bir kontrol sisteminde 1.7s'lik bir inceleme aralığına imkan olmadığı için bu süre aşağıya çekilmeye çalışılmış, verimin süre ile birlikte azaldığı gözlenmiştir. Bunun sebebi yararlı bilginin zamanın o fraksiyonunda YSA'nın başa çıkabileceği kadar yer alamamasında saklıdır. YSA gerekli yakınsama işlemini tam olarak yerine getiremediği için verimin düşmesine sebep olmaktadır. Ama gene de sonuçlar umut vaat edici niteliktedir.

Düşüncelerin karmaşık koridorları, bilgisayarlar bir yana dursun, insanlar için bile bir muamma niteliğindedir. Kararsız bir yapıyı anımsatan bir takım kaotik kırıntılar EEG'yi bilinen yöntemlerle sınıflandırmayı zorlaştırmaktadır. Nedensellik ilişkisini fark etmek aynı rasgelelikle kutsanmış YSA için bile zor bir uğraştır. Bu noktadan sonra araştırmaya değer diğer bir konu genetik algoritma (GA) ve GA'nın YSA'ya uygulanması olarak düşünülebilir. Belki GA'nın popülasyonlar aracılığıyla evrimleştirdiği bilgi sayesinde daha üstün bir başarı elde edilebilecektir (Schröder, 2003). Diğer bir atılabilecek adım ise SVM (Support Vector Machines, Destek Vektör Makineleri) adı altında toplanmış olan bir başka örüntü tanıyıcılarıdır. Kısaca SVM olarak bahsedeceğimiz bu konu kökleri Vapnik'in İstatistiksel Öğrenme Teorisine (Vapnik, 1995) uzanan bir takım sınıflandırıcı ve regresyon yöntemlerini barındırmaktadır. Sınıflandırma tekniği optimal ayırım yasasına dayandırılarak, örüntü sınıflarının maksimum şekilde birbirinden ayrılabilmesi mümkün kılınmaktadır. Bu durum YSA'nın arzu edileni karşılayamadığı durumlarda başvurulabilecek alternatif bir yöntemin varlığına işaret etmektedir.

Her ne kadar bilgisayar, bir takım bilgilerin prosesinde çok hızlı olsa da insan beyninin esnekliğine sahip değildir. Özellikle bir takım önemsizmiş gibi gözükten işlemlerin (evde tanıdık olmayan bir lamba veya mobilyanın fark edilmesi) yapılması bir bilgisayar için hazmedilmesi güç bir olgudur. Ayrıca beynin öğrenme ve hatırlama kabiliyeti, beş yaşındayken edinilmiş olan ufak bir ayrıntıyı anımsayacak kadar kuvvetlidir [14]. Öte yandan bilgisayar bazlı EEG örüntü tanıma tekniklerinin, sözü geçen bu insani işlevleri yerine getirebilmesi için daha uzun yılların geçmesi gerektiği su götürmez bir gerçektir. Gene de edinilmiş olan verilerin ışığında yapılan tüm uğraşlara degecek bir umudun olduğu da yadsınamaz. Vücutsal işlevlerini yerine getiremeyen insanların tüm isteklerini, henüz karşılayamasa bile bir takım basit emirlerin bahsi geçen stratejilerle tanınabileceği ve kontrole yönelik uygulamalarla bütünleştirilerek kullanabileceği vurgulanmaya çalışılmıştır. Temennimiz, üçüncü nesil bilgisayarlar olarak tasvir edilen Quantum Bilgisayarların, bir an önce bilim camiasının hizmetine girmesi ve böylelikle edinilecek olan muazzam işlemsel yetenekle, YSA, GA, SVM gibi teknikleri çok daha yüksek hızlarda, kabul edilebilir tanıma verimlerinde çalıştırabilmesidir.

KAYNAKLAR

Baillet, S., Mosher, J.C., Leahy R.M.(2001),"Electromagnetic Brain Mapping", IEEE Signal Processing Magazine, 14-30.

Blanchard, G. ve Blankertz, B.(2004),"BCI Competition 2003:Data Set IIa - Spatial Patterns of Self-Controlled Brain Rhythm Modulations", IEEE Transactions On Biomedical Engineering, 100-104.

Blankertz, B., Müller, K., Curio, G., Theresa, M., Vaughan ve Schalk, G.(2004),"The BCI Competition 2003:Progress and Perspectives in Detection and Discrimination of EEG Single Trials", IEEE Transactions On Biomedical Engineering, XX, NO. Y, 100-106.

Chiappa, S. ve Bengio S.(2004),"HMM and IOHMM Modeling of EEG Rhythms for Asynchronous BCI Systems", IDIAP, CH-1929 Martigny, Switzerland, 1-6.

Cincotti, F., Scipioni, A., Timperi, A., Mattia, D.(2003),"Comparison of Different Feature Classifiers for Brain Computer Interfaces", Proceedings of the 1st International IEEE EMBS Conference on Neural Engineering, Italy, 645-647.

Culpepper, B.J. ve Keller, M.Robert (2003),"Enabling Computer Decisions Based on EEG Input",IEEE Transactions on Neural Systems and Rehabilitation Engineering,Vol. 11, No.4, 354-360.

Curran, E., Sykacek, P., Stokes, M., Roberts, S.J., Penny,W., Johnsrude, I., Owen, A.M.(2003), "Cognitive Tasks for Driving a Brain-Computer Interfacing System: A Pilot Study", IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol.12, No.1, 48-54.

Delsanto, S., Lamberti, F., Montrucchio, B.(2003),"Automatic Ocular Artifact Rejection based on Independant Component Analysis and Eyeblink Detection", Proceedings of the 1 st International IEEE EMBS Conference on Neural Engineering, Italy, 309-312.

Demuth, Howard ve Beale Mark (2002), Neural Network Toolbox, The MathWorks, Inc., 36-39,42-45,52,139,144-151,245-249.

Fabiani, G.E., McFarland, D.J., Wolpaw, J.R., Pfurtscheller, G.(2004),"Conversion of EEG Activity Into Cursor Movement by a Brain-Computer Interface (BCI)", IEEE Transactions on Neural Systems and Rehabilitation Engineering,Vol.12, No.3, 331-340.

Felzer, T. ve Freisleben B.(2003),"Analyzing EEG Signals Using the Probability Estimating Guarded Neural Classifier", IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol.11, No.4, 361-371.

Freeman, W.J.(2003),"A Neurobiological Theory of Meaning in Perception", IEEE, Uni.of Cal.USA, 1373-1378.

Guger, C., Edlinger, G., Harkam, W., Niedermayer, I., Pfurtscheller, G.(2003),"How Many People are Able to Operate an EEG-based Brain-Computer Interface (BCI)", IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol. 11, No.4, 145-147.

Gurney, K. (2002), "Computers and Symbols versus Nets and Neurons", Dept. Human Sciences. Brunel University Uxbridge. Middx., 1-3.

Haselsteiner, E., Pfurtscheller, G.(2000), "Using Time-Dependent Neural Networks for EEG Classification",IEEE Transactions on Neural Systems and Rehabilitation Engineering,Vol.8, No.4, 457-464.

Hinterberger, T., Mellinger, J., Birbaumer N.(2003), "The Thought Translation Device: Structure of a Multimodal Brain-Computer Communication System", Proceedings of the 1 st International IEEE EMBS Conference on Neural Engineering, Italy, 603-606.

- Krose, B. ve Patrick, Van der S. (1996), "An Introduction to Neural Networks", University of Amsterdam & German Aerospace Research Establishment, 13-15.
- Lem, S., Schafer, C. ve Curio, G. (2004), "BCI Competition 2003—Data Set III: Probabilistic Modeling of Sensorimotor u Rhythms for Classification of Imaginary Hand Movements", *IEEE Transactions On Biomedical Engineering*, VOL.51, NO.6, June 2004, 1077-1080.
- Liu, H. S., Gao, X., Yang F., Gao, S.(2003), "Imagined Hand Movement Identification Based on Spatio-temporal Pattern Recognition of EEG", *Proceedings of the 1st International IEEE EMBS Conference on Neural Engineering*, Italy, 509-602.
- Lo, P., Wu, S., Wu, Y.(2004), "Mediation Training Enhances the Efficacy of BCI System Control", *Proceedings of the 2004 IEEE International Conference on Networking, Sensing & Control*, Taiwan, 825-828.
- Maiorescu, V.A., Şerban, M., Lazar, A. M.(2003), "Classification of EEG Signals Represented by AR Models for Cognitive Tasks-A Neural Network Based Method", *UTI &UMF, Romania*, 441-444.
- Mason, S.G., Moore, M.M., Birch, G. E.(2003), "Designing Pointing Devices Using Brain-Computer Interface Technology", *Proceedings of the 1 st International IEEE EMBS Conference on Neural Engineering*, Italy, 610-613.
- Matlab Help, "Spectral Analysis", "Spectral Estimation Method", "Parametric Methods, Yule-Walker AR Method"
- Millan, J.R., Mourino, J., Franze, M.(2002), "A Local Neural Classifier for the Recognition of EEG Patterns Associated to Mental Tasks", *IEEE Transactions on Neural Networks*, Vol 13, No.3, 678-686.
- Müller, K.R., Anderson, C.W., Birch, G.E.(2003), "Linear & Nonlinear Methods for Brain-Computer Interfaces", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol.11, No.2, 165-169.
- Obermaier, B., Müller, G.R., Pfurtscheller, G.(2003), "Virtual Keyboard Controlled by Spontaneous EEG Activity", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol.11, No.4, 422-426.
- Pang, C.C.C., Upton, A.R.M., Shine, G., Kamath, M.V.(2003), "A Comparison of Algorithms for Detection of Spikes in the Electroencephalogram", *IEEE Transactions on Biomedical Engineering*, Vol.50, No.4, 521-526.
- Pardey, J., Roberts, S., Tarassenko, L. (1996), "A Review of Parametric Modeling Techniques for EEG Analysis", *Med. Eng. Phys.*, Vol. 18, No. 1, 2–11.
- Schalk, G., McFarland, D.J., Hinterberger, T., Birbaumer, N. ve Wolpaw, J.R.(2004), "BCI2000: A General-Purpose Brain-Computer Interface (BCI) System", *IEEE Transactions On Biomedical Engineering*, VoL.51, No.6, June 2004, 1034-1043.
- Schlögl, A.(2003), "Data Set: BCI-Experiment", Department of Medical Informatics, Institute for Biomedical Engineering, University of Technology Graz., 1-2 .
- Schlögl, A., Keinrath, C., Scherer, R., Pfurtscheller (2003), "Information Transfer of an EEG-based Brain Computer Interface", *Proceedings of the 1st International IEEE EMBS Conference on Neural Engineering*, Italy, 641-644.
- Schröder, M., Bogdan, M., Rosenstiel, W., Hinterberger, T., Birbaumer, N.(2003), "Automated EEG Feature Selection for Brain Computer Interfaces", *Proceedings of the 1 st International IEEE EMBS Conference on Neural Engineering*, Italy, 626-629.

Vinther, M. (2002), "AEP Analysis in EEG from Schizophrenic Patients Using PCA, Orsted DTU, 3-4.

Vapnik, V. (1995), "The Nature of Statistical Learning Theory", New York: Springer-Verlag.

Wolpaw, J..R., McFarland, D.J., Vaughan, T.M., Schalk, G.(2003), "The Wadsworth Center Brain-Computer Interface (BCI) Research and Development Program", IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol. 11, No.2, 204-207.

Yazgan, E. ve Korürek, M.(1996), TIP Elektroniği, İTÜ, 6.6-6.9.

INTERNET SİTELERİ

[1] http://www.bcm.edu/neurol/struct/epilep/epilepsy_neurophys.html

[2] <http://personal.tmlp.com/Jimr57/textbook/chapter3/chapter3.htm>

[3] <http://www.emc.maricopa.edu/faculty/farabee/BIOBK/BioBookNERV.html>

[4] <http://ifcsun1.ifisiol.unam.mx/Brain/mempot.htm>

[5] http://www.brookscole.com/chemistry_d/templates/student_resources/shared_resources/animations/ion_pump/ionpump.html

[6] <http://ifcsun1.ifisiol.unam.mx/Brain/action.htm>

[7] <http://www.ebme.co.uk/arts/eegintro/index.htm>

[8] <http://www.ebme.co.uk/arts/eegintro/eeg4.htm>

[9] <http://www.ebme.co.uk/arts/eegintro/eeg2.htm>

[10] <http://www.ebme.co.uk/arts/eegintro/eeg3.htm>

[11] <http://www.ebme.co.uk/arts/eegintro/eeg5.htm>

[12] <http://www.statsoftinc.com/textbook/stneunet.html#index>

[13] <http://www.mle.ie/~rob/mindbalance/>

[14] <http://www.napa.ufl.edu/2004news/braindish.htm>

[15] <http://www2.ac.edu/faculty/PMarek/Physio/PowerPoint/2.1%20Nerve%20Cells.ppt>

[16] <http://ebrains.la.asu.edu/~nsfadp/presentations/SilviaFerrari.ppt>

ÖZGEÇMİŞ

Doğum tarihi 14.03.1979

Doğum yeri İstanbul

Lise 1994-1997 Kadıköy Anadolu Lisesi

Lisans 1997-2001 Kocaeli Üniversitesi Mühendislik Fak.
Elektrik Mühendisliği Bölümü

Yüksek Lisans 2002-2005 Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü
Elektrik Müh. Anabilim Dalı

