

46961

YILDIZ TEKNİK ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ



**MOTOR HATA TESPİT VE ANALİZİ İÇİN  
YAPAY SİNİR AĞLARININ TASARIMI**

**Elk. Müh. Kayhan GÜLEZ**

**F.B.E. Elektrik Mühendisliği Anabilim Dalında  
hazırlanan**

**YÜKSEK LİSANS TEZİ**

**Tez Danışmanı : Doç. Dr. Hacı BODUR**

**İSTANBUL, 1995**

**T.C. YÜKSEKÖĞRETİM KÜLTÜR VE  
BİLİM BAKANLIĞI**

## **TEŐEKKÜR**

Bu tezin hızlı bir Őekilde ilerlemesi ićin gereken kolaylıđı gosteren Y.T.Ü. Elektrik Mühendisliđi Elektrik Makinaları Ana Bilim Dalı öğretim üyesi Sayın Hocam Doć. Dr. Hacı BODUR'a teŐekkürlerimi sunarım.

Ayrıca tezin hazırlanmasında ve ćalıŐmalarım sırasında yönlendirme, teŐvik ve her türlü desteđi ile bana yardımcı olan C.B.Ü. Elektrik-Elektronik Mühendisliđi Bölüm Başkanı Sayın Hocam Yrd. Doć. Dr. Bekir KARLIK'a teŐekkürü bir borć bilirim.

Gerekli deneysel deđerlerin alınması konusunda ilgi ve yardımını esirgemeyen Y.T.Ü. Elektrik Mühendisliđi Elektrik Makinaları Ana Bilim Dalı öğretim üyelerinden Sayın Yrd. Doć. Dr. İbrahim ŐENOL'a teŐekkürlerimi sunarım.

Teorik ćalıŐmalar ile ilgili olarak yardımlarımı esirgemeyen İ.T.Ü. Ućak ve Uzay Bilimleri Fakültesi Ućak Mühendisliđi öğretim üyelerinden Sayın Doć. Dr. Ramazan TAŐALTIN'a teŐekkürlerimi sunarım.

Bilgisayar ile yaptığım ćalıŐmalarda her türlü yardımı sađlayan C.B.Ü. araştırma görevlilerinden baŐta İnŐ. Yük. Müh. Sayın Adem EREN olmak üzere, End. Müh. Sayın Muhterem ćÖL'e ve Elek. Yük. Müh. Sayın RaŐit ATA'ya teŐekkürü bir borć bilirim.

## **İÇİNDEKİLER**

Özet	V
Summary	VI
1. GİRİŞ	1
1.1. Konunun Tanıtılması	1
1.2. Mevcut Çalışmaların İncelenmesi	2
2. MOTOR HATA TESPİTİ	6
2.1. Döner Makinalarda Başlangıç Hataları	6
2.2. Döner Makinalarda Başlangıç Hata Tespiti İçin YSA'nın İncelenmesi	7
2.3. Araştırması Yapılan Başlangıç Hatalarının Türleri	8
2.4. Stator Akımı ve Rotor Hızı Maliyet Fonksiyonları	11
3. İNDÜKSİYON (ASENKRON) MOTORLARDA HATA TESPİTİ	13
3.1. İndüksiyon Motoru	13
3.1.1. İndüksiyon Motorunun Yapısı	13
3.1.2. Stator	13
3.1.3. Rotor	13
3.1.3.a. Sincap Kafesli Rotor (Kısa Devreli Rotor)	14
3.1.3.b. Sargılı Rotor (Bilezikli Rotor)	14
3.1.4. İndüksiyon Motorunun Çalışma Prensipleri	14
3.1.5. Performans karakteristikleri: Hız-Moment ve Hız-Akım Eğrileri	15
3.1.6. İndüksiyon Motor Dinamikleri	16
3.1.7. İndüksiyon Motorlarında Hız Kontrolü	17
3.1.7.a. Kutup Değiştirme Kontrolü	18
3.1.7.b. Hat Frekans Kontrolü	19
3.1.7.c. Hat Gerilim Kontrolü	19
3.1.7.d. Rotor Direnç Kontrolü	19
3.1.7.e. Yardımcı Aletler İle Kayma Kontrolü	20
3.1.8. Tek Fazlı İndüksiyon Motorları	21
3.1.9. İndüksiyon Motorları Test Prosedürleri	23
3.2. Sincap Kafesli İndüksiyon Motorlarında Başlangıç Hata Tespiti	24
3.3. Motorlarda Hata Tespiti İçin Matematiksel Tanımlama	24
3.4. Hata Tespitine Diğer Tekniklerin Yaklaşım Şekli	29
3.5. Hata Tespitinde YSA'nın Öğrenme Kabiliyetleri	30
3.6. Hata Tespiti İçin İleri-Besleme Ağ Modeli ve Geriye-Yayımlı Öğretme Algoritmasına Bakış	31

3.6.1. Çok Katmanlı İleri-Besleme YSA'nın Tanımı	31
3.6.2. Sınır Ağını Öğretmenin (Öğrenme) Matematiksel Formülasyonu	33
4. YAPAY SİNİR AĞLARI	38
4.1. Giriş	38
4.2. Tarihsel Gelişimi	38
4.3. YSA'nın Tanım ve Modeli	39
4.3.1. YSA'nın Tanımı	39
4.3.2. Nöronun Biyolojik Yapısı ve Nöron Modeli	40
4.4. YSA'nın Yapısı ve İşlem Elemanı	42
4.4.1. Giriş İşaret Sınıfları	43
4.4.2. Bağlantı Geometrieleri	45
4.4.3. Ağ Tipleri	46
4.4.4. Eşik Fonksiyonları	46
4.4.5. Ağırlık Uzayları	47
4.5. YSA'da Eğitim (Training)	49
4.5.1. Eğitim Algoritmaları	49
4.5.2. Bellek	50
4.5.3. Hata Toleransı	51
4.5.4. YSA Kullanımının Sebepleri	51
4.5.5. YSA'nın Klasik Yazılımlar İle Karşılaştırılması	52
4.6. Öğrenme Kuralları	52
4.6.1. Perceptron (İdrak Almaç)	53
4.6.2. Çok Katmanlı Perceptron (Multi-Layer Perceptron)	55
4.6.3. Hatanın Geriye-Yayılması Algoritması ve Genelleştirilmiş Delta Kuralı	56
4.6.4. Öğrenme ve Momentum Katsayıları	60
5. BAŞLANGIÇ HATA DETEKTÖRÜ VE YAPILAN ÇALIŞMA	61
5.1. Başlangıç Hata detektörünün Yapısı	61
5.2. Başlangıç Hata Detektörünün YSA İle Tasarımı ve Öğretim Konsiderasyonları	61
5.2.1. Öğrenme Parametrelerinin Probleme Bağlı Olarak Seçimi	62
5.2.2. Ağ konfigürasyonunu Basitleştirmek İçin Ağ Bilgisi	63
5.2.3. Ağ Öğretim Bilgi Grubunun Büyüklüğü: Öğrenme Teorisinin İncelenmesi	64
5.2.4. Ağ Sağlamlık Konsiderasyonları	66
5.3. İndüksiyon Motorlarında Yüksek Mertebeden Başlangıç Hata Detektörünün YSA İle Tasarımı	67
5.4. Yapılan Çalışma	68
5.4.1. Ölçme Sonuçlarının Alınması	68
5.4.2. Deneyler İle Elde Edilen Ölçüm Sonuçları	70
5.4.3. Simülasyon Sonuçları ve Akış Diyagramı	71

5.4.4. Sonuların Deęerlendirilmesi	77
5.5. Yapılan alıřmanın Mevcut alıřmalarla Karřılařtırılması	77
KAYNAKLAR	78



## ÖZET

Yapay sinir ağları (YSA) günümüz teknolojisinde hata tespiti, kontrol, işaret işleme gibi farklı birçok alanlarda başarı ile kullanılmaktadır. Mühendislik alanı olarak ele alırsak, başta bilgisayar ve elektrik-elektronik mühendisliği olmak üzere, biyolojiksel ve fiziksel kapsam dahilinde çok çeşitli yerlerde pratik uygulama alanı bulmuştur. Herşeyden önce matematiksel formüllerden bağımsızlığı ve sonuca etkili derecede kısa bir zaman dilimi içinde yaklaşması , YSA'yı diğer çözüm metodlarına göre tercih sebebi yapmıştır.

Bu çalışmada da, motorlarda hata tespitini gerçekleştirmek için ileri-besleme ağı ve hatanın geriye-yayılımı öğretme algoritması kullanılmaktadır. Yine, konuya bağlı kalınarak diğer yaklaşım metodları (parametre tahmin yaklaşımı v.b.) ile arasında hangi açıdan fark bulunduğu incelenmektedir. Son olarak ise, alınan deneysel ölçümlere göre öğretilen sinir ağına , farklı bir model için verilen sayısal değerlerin iterasyonel sonuçları yer almaktadır. Burada YSA'ya uygulanan değerler farklı tipte split(tek) fazlı sincap kafesli indüksiyon motorlarından alınmıştır.

Motor hata tespiti için , YSA'ya  $I$  ve  $w$  değerlerini girerek 30000 iterasyon sonucunda %95'in üzerinde bir başarı ile izolasyon hatası  $N_c$  ve yatak aşınma hatası  $B_c$  için elde edilmiştir. Özellikle  $N_c$  için elde edilen %99.2'lik hata tanıma oldukça büyük bir başarıdır.

## ***SUMMARY***

Artificial Neural Networks (ANN) is successfully used in a lot of areas such as fault detection, control and signal processing in our daily technology. If we pay attention for engineering areas to this subject like firstly computer engineering and electrical-elektronik engineering, ANN have found plenty of practical application areas, under biological and physical subjects. Before all of this things, ANN's matematical indepedence and its effectiveness in a very short time to approaching to the results, it was a very preference reason according to the other approaching metods.

In this thesis, feedforward network and error backpropagation training algorithm is used to perform the motor fault detection. In addition, there is a comparisn between the fault detection and the other approach methods (i.e. , parameter estimation approach) in therms of the way which is better. Finally, accrding to the values which were taken from the experimental measurements, there are iterasyonel results of the numerical values which is given for a different paradigm to the trained neural network. In that place, the values which were used for ANN were taken from split-phase squirral-cage induction motors in different models.

In this study, motor fault detection with ANN, as the inputs I and w, as the outputs  $N_c$  (insulation condition) and  $B_c$  (bearing wear condition) are made and discriminated with success rate above 95% for 30000 iterations. It is considerably important success to access this fault detection being 99.2% for  $N_c$ .

## 1. GİRİŞ

### 1.1. Konunun Tanıtılması

Son zamanlarda yapay sinir ağları (YSA) ile tasarım teknolojisi başarılı bir şekilde hata tespiti (fault detection), kontrol ve işaret işleme (signal processing) gibi birçok farklı alanlarda kullanılmaktadır. Yapay sinir ağlarının ilk kullanımı 1940'lara kadar götürülebilir. Ancak YSA çalışmalarında son on yılda ciddi bir ilerleme kaydedilmiştir. Yapılan çalışmaların büyük modellerinden (major paradigm) biri ileri-besleme (feed-forward) ağıdır. Kohonen ağı, Hamming ağı, Hopfield ağı, adaptif rezonans teorisi v.b. çalışmalar diğer ağ modellerine iyi birer örnek teşkil etmektedir. Her bir çalışma kendine özgü ağ yapısına, uygunluklarına ve öğretme algoritmalarına sahiptir. Bu konu içinde yer alan bazı yöntemler, karşılaşılan problemlerin ancak belirli türlerini çözmek için daha uygundur. Karşılaşılan problemleri çözmeye aşamasında, değişik yöntemler içinde muhtemelen ileri-besleme ağı en çok kullanılan ağ modelidir. Günümüzde YSA'ların % 80'ninden fazlasını ileri-besleme ağları oluşturur. Yapılan çalışma, ileri-besleme ağ yönteminin kullanımından hareketle, motor hata tespitini yapmak için YSA teknolojisinin niçin ve nasıl uygulanacağını anlatmağa çalışmaktadır.

Bu çalışma genel manada ele alındığında, YSA robotik, biyomedikal mühendislik ve haberleşme alanlarında çok geniş bir şekilde kullanılmakta iken, elektrik makinelerinde (tüm döner makineler için), kontrol olayını da kapsam dahilinde tutarak, böyle bir çalışma alanına uygulanması dikkate değer önemli bir olaydır. Yapılan çalışmalar başlı başına üzerinde çalışılan konuda bir problem teşkil etmekte iken, elektrik mühendisliğinde bu problemlerin herhangi birinin çözümüne YSA ile yapılacak iterasyonel bir yaklaşım çok fazla iterasyon sayısı içermesine rağmen çok kısa sürede ve verilen girişlere çok düşük hata payıyla cevap vermesinden dolayı (yani doğruluğunun yüksek olmasından dolayı) en iyi çalışma tekniklerinden biri olarak kabul edilmektedir.

Yapılan çalışmada özellikle prototip olarak ele alınan indüksiyon (asenكرون) motorlarında hata tespit ve analizi üzerinde durulmaktadır. Hata tespit konusu bu noktadan hareketle, diğer motor türlerine de rahatlıkla uygulanabilmektedir. İndüksiyon motorlarında hata türleri olarak en çok motorun sarım izolasyonu (winding insulation)  $N_c$ , ve motorun yatak durumu (bearing condition)  $B_c$  ile karşılaşılmaktadır. Bunlar daha sonraki bölümlerde ayrıca ele alınacaktır. Özellikle 1990'dan sonra bu konu üzerine yapılan çalışmalar dikkate alındığında, yeni bir sayısal çalışma yapılmak amacı ile belirli motor değerleri, YSA ile gerçekleştirilen simülasyona dayalı tasarımda kullanılmaktadır. İndüksiyon motorları için, yapılan bazı kabuller çerçevesinde, ileri-besleme ağı ve geri yayılım (backpropagation) öğretme algoritması kullanılarak sonuca ulaşmak istenmektedir.

Bunlardan başka, ana konunun kapsamı dahilinde olmak üzere , yaptığımız çalışmaya parametre tahmin yaklaşımının ve uzman (yani bu konu üzerinde yaptığı çalışmalarla belirli bir seviyeye gelmiş kişinin) yaklaşımının -motorun iç yapısına inmeden- nasıl uygulandığı ya da bu teknikleri YSA ile karşılaştırma açısından nasıl bir şey olduğunun kısa bir incelemesi ile YSA'nın tarihi gelişimi,yapısı ve uygulama alanları ileriki bölümlerde verilmektedir.

Ayrıca, prototip olarak ele alınan indüksiyon motoru üzerinde özel bir çalışma yapıldığından yine bundan sonraki bölümlerde indüksiyon motorları ile ilgili gerekli bilgi verilmekte ve bu bilgi verilirken bu tür motorlarda hatanın tespitinin nasıl yapıldığı hakkında yine incelemenin yapıldığı konu kapsamında kalınarak gerekli açıklama yapılmaktadır.

Son bölümde, geriye-yayılım öğretme algoritması kullanılarak bilgisayarda uygulaması yapılan çalışma yer almaktadır. Buradan elde edilen sonuçlar mevcut çalışmalarla karşılaştırılmakta ve doğruluk açısından yeni yaklaşımlara ulaşılmaya çalışılmaktadır. Bu amaçla ağ kabiliyeti , ağ oluşturma, mevcut çalışmakta olan bilgi (data) setinin boyutu, çalışan parametre değerlerinin görevi, durma kriteri gibi sinir ağıyla ilgili tasarıma ait ifadelerde incelenmiştir. Zaten bu tür tasarıma ait ifadeler sinir ağ yapısında önemli yer işgal etmektedir.

Yukarıda da belirttiğimiz gibi , YSA'lar yaklaşım olarak motor hata tespit konusunun yanısıra birçok mühendislik alanı ve bu alanlar içindeki farklı konularda rahatlıkla uygulama alanı bulur hale gelmiştir. Yapılan benzer çalışmalarda , iterasyon miktarının fazlalığına rağmen zamandan büyük miktarda kazanç sağlanması diğer yaklaşım metodlarına göre büyük bir üstünlüğü temsil etmektedir. Bu sebep dahilinde, diğer konular üzerinde bu konu ile bağlantılı olarak çalışılması konunun önemini dile getirmektedir.

### *1.2 Mevcut Çalışmaların İncelenmesi*

Motor hata tespiti konusu üzerindeki çalışmaları dikkatle inceleyecek olursak YSA gibi büyük çalışma alanına sahip bir konunun arka planda kalmış gibi görünen konulara bir basamak teşkil ettiği açıkça görülmektedir. Bu da bize şunu ifade etmektedir ki: YSA konusu ana başlık olarak ele alındığında çok büyük bir çalışma dairesini kapsamaktadır, halbuki bu konuya basamak yapılacak diğer çalışmalar YSA'nın büyük dairesi içinde küçük bir yeri işgal etmektedir. Yani her alana uygulanabilirlik özelliği her ana bilim dalındaki konuların çabuk bir şekilde çözümüne zaman açısından büyük kolaylık sağlamaktadır.

Bu noktada biraz konuyu açacak olursak , özellikle YSA'ların her şeyden önce diğer konulara çözüm sağlayabilmesi için hızlı öğrenme algoritmalarına ihtiyaç vardı. Bu konuda 1992 yılında yapılan bir çalışmada sinir ağları için hızlı öğrenme algoritmalarının bir ailesine yeni bir yaklaşım yapıldı (Karayiannis et al 1992). Bu çalışmada amaç ileri-besleme sinir ağlarını öğretme olayında genelleştirilmiş yeni bir kriter kullanmaktır.

Bu bahsettiğimiz kriter tek ve çok katmanlı sinir ağları için kullanılan optimizasyon stratejisine bağlı olarak hızlı öğrenme algoritmaları ortaya çıkarır. Genelleştirilmiş bu kriterin temelinde kurulmuş en basit algoritma , tek katmanlı sinir ağlarını öğretmek için sunulan hızlı delta kuralı (fast delta rule) algoritmasıdır. Ortaya konulan bu genelleştirilmiş kriterle bağlantılı olarak çok katmanlı sinir ağlarına , benzer bir optimizasyon strateji yaklaşımını, hızlı geri-yayılım (fast backpropagation) algoritması sağlar.

Bunlardan başka , aynı stratejinin temeline dayanılarak birbirine daha iyi uyumlu hızlı algoritmaların başka bir ailesi , Etkili Öğrenme Algoritmaları , geliştirilmiştir (Efficient LEarning Algorithms for Neural NETWORKS, ELEANNE) (Karayiannis et al 1992).

Sinir ağları 80'li yılların başlarında tekrar popüler olmaya başladığında araştırmacılar arasında, sinir ağı kullanımına karşı aşırı derecede bir talep olmasından dolayı, hakim bir eğilim vardı. Bu eğilim uygulamaya hızlı üretim ve sinir ağı sistemlerinin pazarlanması şeklinde aktarılmak istendi. Bu şekildeki eğilim asıl olarak konuşma ve görüntü tanıma sistemlerinin gelişimi ve de öğretilbilir kontrol sistemleri için geçerli ve gerçekçi bir alternatif olarak sinir ağlarını tesis etmek isteği ve amacıyla adapte edildi ve de bu anlamda ortaya çıktı. Yeni bir yapılanmağa giren sinir ağları için yapılan araştırmaların ilk çalışmaları boyunca mevcut öğrenme algoritmaları yeterince doyurucuydu. Fakat sinir ağları ile ilgili araştırmalar sanatsal (sadece teoriksel) çalışmalardan , gerçek dünya problemlerine çözüm bulma şeklini almağa başladığı andan itibaren , hem öğretme zamanı hem de programlama gereksinimleri altında sinir ağları , programlama tekniklerine bir alternatif olarak ortaya çıktı. Her ikisini karşılaştırma durumunda hız, yapacağımız tercihi etkileyen önemli bir faktör olmaktadır. Hızlı ve etkili öğrenme algoritmalarının varlığı gelecekte bu araştırma alanında ortaya çıkacak tüm yenilikler için çok önemlidir.

Bu noktada kısada olsa hızlı öğrenme algoritmalarının özelliklerinden bahsetmeğe ihtiyaç vardır. Bir öğrenme algoritması basitlik, esneklik ve etkinlik gibi oldukça kesin, kompleks ve çeşitli gereksinimler içeren bir temele dayanır. Bu koşullar altında bir öğrenme algoritmasının basitliği , uzman veya bir programcı gereksizsin belirli bir ortalama ile algoritmağa başvurma ve yeniden üretim için algoritmada ihtiyaç duyulan çabadır. Yine bu öğrenme algoritmasının esnekliği , sinir ağlarına çeşitli yapıları (mimari açıdan) öğretmek için algoritmanın başvurusu halinde gereksinim duyulan çözümsel ve programsal çabadır. Son olarak yine bu öğrenme algoritmasının etkinliği, bir sinir ağını öğretme ve de ağı sonuçlandırmanın performansı için sayısal ve zaman gereksinimlerine bakarak ağın bu gereksinimlere gösterdiği değersel (iyi,kötü gibi) çabadır.

Biraz önce tek katmanlı sinir ağlarını öğretmek için delta kuralından bahsetmiştik. Bununla ilgili olarak bilgi verecek olursak , Delta Kuralı B. Widrow tarafından tasarlanıp , yapay sinir ağları konusuna yaklaşık yirmi yıl önce bir hızlı öğrenim algoritması olarak sunuldu (Widrow et al 1970s). Delta Kuralı YSA araştırmalarını önemli derecede etkileyen ve bu konunun ilerlemesine büyük yardım sağlayan faktörlerden biriydi. Delta Kuralı adaptif filitrelemede geniş bir şekilde kullanıldığı halde , basitliği ve esnekliği bu kuralı, sinir ağlarının öğretilmesi için etkin bir şekilde kullanışlı hale getirdi.

Yukarıdaki açıklamalarımızda YSA'lar da en etkili algoritma olarak geriye-yayılım algoritmasının kullanıldığından bahsetmiştik. Bu konuya da kısaca temas edecek olursak, günümüzde çok katmanlı sinir ağlarını öğretmek için en popüler, algoritma hatanın geriye yayılımı (error backpropagation) algoritmasıdır. Bu algoritma orijinalinde (kuruluş temelinde) Werbos (1974) tarafından icat edilmiş ve Rumelhart (1986) tarafından da günümüzdeki etkili ve hızlı kullanım haline getirilmiştir.

Sinir ağları için hızlı ve etkili öğrenim algoritmalarındaki gelişme , ancak son beş yılda gösterilen ilgi ve bu konu üzerine düşünce bazında ciddi olarak düşülmesi sonucu olmuştur. Bunun doğal bir sonucu olarak , birkaç yeni algoritma çeşidi, mimari yapılarıyla (giriş katmanı, gizli katman, çıkış katmanı sıralamasıyla) beraber sinir ağlarını hızlı ve etkili öğretmek için geliştirildi.

Sinir ağları için öğrenme algoritmalarının gelişimini destekleyen iki büyük olgu vardır: ilki, üzerinde çalışılan öğretme probleminin formüle edilmesi ve diğeri buradaki optimizasyonu gerçekleştirmede kullanılan optimizasyon stratejisi. Sinir ağlarını öğretmek için ikinci mertebeden metodlar ise Parker (1987) tarafından geliştirildi. İkinci mertebeden metodlar , hatanın geriye yayılımı algoritmasındakinden daha hızlı, birbirine yaklaşan ileri-besleme sinir ağları için öğrenim algoritmalarının gelişiminde, diğeri araştırmacılar tarafından da geniş bir şekilde kullanıldı.

Motor hata tespiti için bahsedilen algoritmaların kullanımı dahilinde yapılan çalışmalara bakacak olursak, özellikle 1991 yılında içerik olarak kapsamı geniş tutan Mo-Yuen Chow ve Sui Oi Yee' nin döner makineler üzerine yaptıkları çalışma ilk planda dikkatimizi çekmektedir (Mo-Yuen Chow et al, 1991). Bu çalışmada, döner makinelerdeki başlangıç hataları için bir yaklaşım yapılmış olup, aynı çalışma içinde daha sonraki bölümlerde motorların özel bir kısmı prototip olarak ele alınıp hata tespiti yapmak amacıyla çözüm yolları aranmıştır. Yine aynı çalışmada tasarlanan yapay sinir ağından tek-fazlı sincap kafesli indüksiyon (asenkron ) motorları için alınan simülasyon sonuçları yeterince doyurucu bir performans göstermiştir. Bu çalışmanın prototip olarak ele alınan tek-fazlı sincap kafesli indüksiyon motoru için yapılması herhangi bir yanlış anlamaya sebebiyet vermemelidir. Şöyleki: yapılan çalışmada bu motor türünün prototip incelenmesinden genelleştirmeye gidilerek döner makinelerin diğeri tüm türleri içinde yapay sinir ağları yöntemi rahatlıkla uygulanabilmektedir. Burada uygulama açısından dikkat edilmesi gereken tek ve en önemli şey her makine çeşidine göre uygun modifikasyonlara ihtiyaç olduğudur.

Aynı bilim adamları daha sonra yaptıkları başka bir çalışmada diğeri yaklaşım metodlarını da dikkate almışlardır (yani diğeri yaklaşım metodları ile, YSA metodunu karşılaştırmaya gitmişlerdir). Bu çalışmada ileri-besleme ağları ve geriye yayılım öğretme algoritması kullanarak daha iyi bir sonuç elde etmeğe çalışılmıştır.

Yine W.T. Miller, R.S.Sutton ve P.J. Werbos'un yaptıkları, kontrol için sinir ağlarının kullanımı çalışması, elektrik ve elektronik mühendisliğinde çoğu alanı kaplayan kontrol konusıyla ilgili

bir kısım problemin - burada motor hata tespiti konusunda olduğu gibi- çözümüne büyük katkıda bulunacak gerekli açıklamaları içermektedir (W.T.Miller et al, 1990).

Kontrol için sinir ağıları konusundan başka R.P. Lippmann' ın sinir ağlarının programlanması ile ilgili yaptığı çalışma da bilgisayardan simülasyon sonuçlarının alınmasına (şekil olarak düşünüldüğünde ) büyük ölçüde yardımcı olmaktadır ( R.P. Lippmann, 1987) .

Ayrıca M.Y. Chow ve S.O. Yee' nin 1991 yılında yaptıkları, yapay sinir ağıyla gerçekleştirilmiş başlangıç hata dedektörünün sağlamlık testi çalışması da bu konu kapsamı içinde önemli bir yer teşkil etmektedir (M.Y. Chow et al, 1991).

Görüldüğü gibi, yapılan bir çalışmanın gerçekliği (hata payının çok düşük olması ), sağlamlığı , uygulanabilirliği, kontrol edilebilirliği, gözlenebilirliği gibi şartlarda o çalışma için büyük değer taşımaktadır.



## **2. MOTOR HATA TESPİTİ**

### **2.1 Döner Makinalarda Başlangıç Hataları**

Döner makineler büyüklük olarak çok geniş bir sahada üretilmektedir. fakat küçük (10 B.G' den daha az) ve artan büyüklükte (100 B.G' den daha az ) makineler özel bir ilgi alanı içinde yer almaktadır. Çünkü büyük makineler (100 B.G veya daha fazla) periyodik bakıma ve özel koruyucu aletlere sahiptir (direnc sıcaklık dedektörleri gibi). Bunlar meydana gelebilecek hatalara karşı uyarıda bulunmak için bu makinelerin sargılarında ve yatak kısımlarında yer almaktadır. Yapılan çalışmayı bilgisayar monitöründe kontrol etme (izleme) gibi geliştirilen bazı başlangıç hata tespiti planları oldukça pahalıdır. Çünkü bunları sadece pahalı, büyük makinelerle kullanmak uygundur. Küçük ve orta büyüklükteki makineler ekonomik nedenlerden daha az periyodik bakıma ihtiyaç duyarlar ve çok fazla koruyucu özelliklere sahip değildir.

Başlangıç hata tespitinin önemi, meydana gelmeden önce potansiyel makina başarısızlıklarını kontrol ederek, bu sayede sağlanan maliyet tasarrufları ile tanımlanabilir. Makinalar, ani hata akımlarını kesmek üzere devre kesicileri veya sigortalar tarafından korunulmağa ihtiyaç duyar. Aşırı akım yüklenmesine karşı koruma sağlayan motoru harekete geçiriciler (ateşleyiciler) ve yerleştirilmiş bulunan anahtarlar da motoru uzun süre aşırı yüklenme koşullarından korumak için kullanılır. Ancak , bu aletler (ateşleyiciler, anahtarlar) emniyet aletleri olarak motora tesis edilir. Aslında bunlar motoru ve bir hatadan dolayı meydana gelebilecek zarardan, yani motordaki hasardan , yakınında bulunan bir elemanı koruyabilir.fakat bunlar potansiyel hatalar meydana gelmeden önce uyarı vermez. Diğer taraftan , başlangıç hata tespiti genellikle uygulama açısından iyi olmayan makinalarda, programlanmak şartı ile önleyici bakıma izin verir. Bu da geniş çapta makina başarısızlığı tarafından sebep olunan düşüş zamanının (down-time) genişletilmiş bir periyodunu önleyebilir.

Döner makineler genellikle iyi tasarlanmış veya tesis edilmiş ve sağlam (robust) oldukları halde , başlangıç hatalarının mümkünatı elektrik enerjisinin mekanik enerjiye dönüşmesinde veya mekanik enerjinin elektrik enerjisine dönüşmesinde ortaya çıkan titreşimlerden dolayı doğaldır.Makinanın birkaç parçası başarısızlığa özellikle hassastır.Örneğin, stator sargılarının durumu mekaniksel titreşim, ısı, devir, yüklenme (belirli bir yükte çalışma) boyunca hasar ve de yağ tarafından kirlenme ile sebep olunan izolasyon başarısızlıklarına bağlıdır.Rotordaki, sargıları eşdeğer olan dökme alüminyum çubuklar bile makinadaki mekaniksel titreşimlerden etkilenir. Makinanın yataklarında aşırı yıpranma veya yetersiz yağlama, motor gereksiniminin dışında aşırı yağlama, yanlış yükleme veya ayarsızlıktan oldukça etkilenmektedir.

Daha sonraki bölümde görüleceği üzere geniş kullanım alanlarından dolayı, prototip inceleme olarak bu çalışmada tek-fazlı sincap kafesli motorun göz önüne alınarak genelleştirme çerçevesinde tüm

döner makinalar için sonuca ulaşılmaya çalışıldığından bahsetmiştik. Çünkü böyle bir alt kümeden (indüksiyon motoru) olayı ele alarak başlangıç hata tespiti projesi için YSA'ların bu tür bir konu üzerindeki kavramlarını geliştirerek , büyük kümeye (döner makinalar) ulaşmak daha sağlıklı bir çalışmayı ortaya çıkaracaktır.Yapılan çalışmadaki uygulamaya göre sinir ağı hata tespit metodolojisi sadece prototip olarak ele alınan tek bir motor türüyle sınırlı kalmayacak, aynı zamanda gereken değişikliklerle döner makinaların diğer birçok türlerine uygulanabilecektir.

## ***2.2 Döner Makinalarda Başlangıç Hata Tespiti İçin Yapay Sinir Ağlarının İncelenmesi***

YSA ile hata tespiti konusunda olduğu gibi , çözüme belirli yaklaşım sağlayan çalışmalar genelde makinanın durumunu belirtebilen , makina parametrelerini tahmin etmek için yapılmıştır.Bu noktada makinanın parametrelerine başvurma yani parametre tahmin yaklaşımının büyük bir dezavantajı , doğru olması bir zorunluluk olan -ilgi odağımız altındaki- sistem dinamiklerini anlamağa (teorik ve pratik olarak yorumlamağa) ihtiyaç duyulmasıdır. Çünkü , bir sisteme parametre tahmin yöntemi ile yaklaşım yaptığımızda o sistemin modellenmesini yapıp bunun üzerinde sonuca gitmek gerekmektedir. Ancak , YSA'lar başlangıç hata tespit projesini oluşturmak için kullanıldığında, makinanın modellenmesi zorunluluğu ortadan kalkmakta ve bunun bir ihtiyaç olmasına son verilmektedir. Sinir ağlarının herhangi bir konuda istenilen haritalamağı öğrenme kabiliyetine sahip olduğu da unutulmamalıdır. Bu kısmı biraz açacak olursak, girişler olarak motor parametreleri ve çıkışlar olarak motor durumlarını düşündüğümüzde, girişlere bağlı olarak çıkışların tam matematiksel ilişkisini bilmeğe ihtiyaç duyulmaksızın sonuç sadece giriş-çıkış örneklerine (sayısal değerlerine) dayandırılarak bulunur.Yani YSA bu sayısal değerlerden yararlanarak belirli bir hata payıyla bize sonuçları verir. Uygun şekilde öğretilen sinir ağının, içindeki hatlar veya bağlantılar istenen haritalamanın lineersizliğini içerecektir. Eğer matematiksel modellemeğe gidilseydi bu lineersizlikle karşılaşılacaktı. YSA sayesinde bu tür modelleme zorluğundan kaçınılmıştır.

Giriş bölümünde, özellikle son zamanlarda sinir ağlarının hata teşhisi, sistem dinamiklerini modelleme, robot kontrolü ve diğer birçok farklı alanlarda geniş bir şekilde kullanılmaya başlandığından bahsetmiştik. Bunun en büyük sebeplerinden biri olarak, sinir ağlarının sistem girişine ve sistemden kaynaklanan gürültülere karşı oldukça sağlam yapıda olmaları , öğrenme kabiliyetlerine sahip olmaları ve gerçek zamanda da (real time) (YSA gibi sistemsel veya programsal bir işlemi ya da hesaplamayı ilgili fiziksel olayın meydana geldiği anda yürütme ile ilgili işlem) çalışabilir olması gösterilebilir.

YSA'lar başlangıç hata tespitini gerçekleştirmek için uygulamaya aktarıldığında , kompleks nonlineer denklemler , matematiksel modelleme zorluğu ve modelleme hataları gibi sıkıntuların tümünden sakınılabılır. Sinir ağları verilen öğretme bilgisine dayanan giriş-çıkış haritalamalarını gerçekleştirebildiğinden, aşırı derecede açık bir şekilde matematiksel modellerin belirtilmesine ihtiyaç

yoktur. İstenen haritalamanın yüksek lineersizliği dolayısı ile, bu modelleme kendi kendine ağ içinde iyice yerleştirilmiş olur, yani bu ağ katmanları arasındaki hatları (veya bağlantıları) kast etmektedir. Gerçekten, sinir ağları tanımlama işlemi için örnek tanımlamağa bağlı olan nonlineer adaptif sistem tanımlama üniteleri olarak tetkik edilebilir.

Sayısal değerlere dayanan hata koşullarının yorumu genellikle zor bir olaydır. Son zamanlarda yapay sinir ağı konusunun haricinde başka bir konu fuzzy sayı (lojik) yorumu, hata yorumunda gündeme geldiğinden beri, hataya sebep olan koşula anlam vermek için ciddi bir matematiksel formülasyon kullanmak genelde yanlış ve pratik olmayan bir durumdur. Bu yüzden karşılaşılan birçok problemde, sonuçların çözümlerden ve formüllerden daha çok sebep ve geçmiş tecrübeye dayalı yorumu (yani sadece insanların yapma kapasitesinde olma hali) gerekli olur. Bir sinir ağı direkt olarak bir uzmanın, ölçüm ile elde edilmiş bilgiyi kullanarak geçmiş deneyimine dayalı yorumu tarafından öğretildiğinden, hata yorumu fuzzy lojik dolayısı ile sinir ağı içine bu çerçevede iyice yerleştirilmiş olur.

Aynı zamanda YSA'lar gerçek zaman uygulamaları için çok uygun olan çok-girişli paralel işleme kabiliyetlerinden dolayı daha geniş bir alanda kullanılmaktadır. Giriş değerlerinin büyük bir sayısı aynı zamanlı olarak çok-girişli bir sinir ağını besleyebilir. Giriş düğümlerinin sayısında artış olduğu halde ağı hesaplama zamanı aynı kalır. Çünkü sinir ağları paralel işlem uygulaması yapar ve bu kabiliyete sahiptir. Bu da sinir ağının kullanımı açısından çok önemlidir. Böylece bu durum bize göstermektedir ki: giriş düğümlerinin sayısındaki artış sinir ağının işleme hızını etkilememektedir. Bunlardan başka giriş düğümlerinin sayısındaki bu artış ölçme gürültüsünden kaynaklanan etkiye bakışla sinir ağının sağlamlığını artırır.

Bunlara ilave olarak, birkez tasarımı gerçekleştirilmiş yapay sinir ağlarında, değişiklikler veya eklemeler yapılmağa ihtiyaç duyulursa bu tür şeyler kolaylıkla gerçekleştirilebilir. Ağ yapısında değişiklik yapıldıktan sonra sadece birkaç kez daha ağ yeniden öğretilerek çok fazla zorluk çekilmeksizin ihtiyaçlara uygun hale getirilebilir.

Bugüne kadar üzerinde çalışılan konularda yapılan analizler, simülasyon sonuçları ve karşılaştırmalar sinir ağlarının performansının istatistiksel ayırt etme (discriminant) analizi gibi klasik istatistiksel metodların performansından üstün olduğunu göstermiştir. Bu da genel manada sinir ağı yorumunu klasik istatistiksel yaklaşımlardan daha doğru ve güvenilir bir hata tespit planı olarak gösterir.

Buradan bir alt sonuç çıkaracak olursak, sinir ağları başlangıç hata tespitini gerçekleştirmede kendilerini cazip hale getiren birçok avantajlara ve yararlı özelliklere sahiptir.

### **2.3 Araştırması Yapılan Başlangıç Hatalarının Türleri**

Daha önceden yapılacak olan çalışma için prototip olarak indüksiyon motorlarının inceleneceğini söylemiştik. Bu hatırlatmanın ışığı altında başlangıç hata türleri konusu ile ilgili açıklamalarımıza

geçebiliriz. Bir indüksiyon motorunda başlangıç hatalarının farklı birçok türleri meydana gelebilir. Genel anlamda başlangıç hata tespiti için YSA'ların yararlılığını kanıtlamak üzere ön hazırlık çalışmasının bir parçası olarak , bu çalışma içinde araştırmalarımızı indüksiyon motorlarında başlangıç hatalarının çok yaygın iki şekli üzerinde yoğunlaştırmaktayız. Yalnız burada bir kabul yapmak zorunluluğu vardır. Bu kabul de belirli bir tahmin çerçevesinde motordaki diğer tüm elemanların çalışır durumda olduğu varsayımdır. Hata türleri sırasıyla.

1- Bir sarımdan diğer sarıma stator izolasyon hatası.

2- Motorun yatak aşınma hatası.

iki tür olarak ele alınacaktır. Motorun sıcaklık şiddeti , motor çalışmaya başladığı andan kalıcı hale gelinceye kadar aşırı derecede çok fazla değiştiği halde, tasarımı düşünülmüş başlangıç hata detektörü sadece sürekli hal altında tanımlanmaktadır.

Bunlardan başka, yanlış alarmlara sebep olabilen rotor çubuğunun kırılması gibi , motor hatalarının diğer birçok türleri vardır. Bir motorda karşılaşılan olması mümkün her tür hatayı %100 doğrulukla denetleyebilen bir hata detektörüne sahip olmak biraz gerçek dışıdır. Gelecekte bu konu üzerinde yürütülecek araştırmalar için, çok yönlü (bütün amaçlara cevap veren) bir hata detektörünü yapmakta mevcut bütün doyurucu hata tespit tekniklerini birleştirme durumu gerçekten önemli bir yer teşkil etmektedir. Fakat biz, bu çalışmayı başlangıç hata tespitinde sinir ağlarını kullanmanın esnekliğini göstermek amacıyla, bir tek-fazlı sincap kafesli indüksiyon motorunda sarımdan sarıma stator izolasyonu ve yatak aşınma hatasını araştırmakla sınırlayacağız.

Bir sarımdan diğerine stator izolasyon hatası ve yatak aşınmasının analizine dayanan yüksek mertebeden bir YSA, hata tespiti operasyonunun gerçekleştirilmesi için gerektiği şekilde tasarlanmıştır. Motorda nadiren parazitler görmek muhtemel olduğunda , hata tespit projesi gerçek zamanda uygulanmak için tasarlandığından , bu motor daima kalıcı (sürekli ) halde olmayabilir. Hata detektörüne uygunsuz girişlerin gelmesi durumunda (parazitler boyunca ve parazitlerden sonra motorun geçici halinin ölçümleri gibi ) sinir ağı yanlış alarmlar verebilir. Bunun bir sonucu olarak , başka bir yapay sinir ağı, biraz değiştirilmiş hızlı bir öğrenme algoritması kullanarak, motor kalıcı hal ölçümlerini kaydediyor iken geçici hal ölçümlerini filitrelemek için geliştirilir. Bu kalıcı hal ölçümleri motor içinde mümkünatı bulunan başlangıç hatalarının tespiti için hata detektörüne bir besleme şeklinde gönderilir.

Sinir ağı hata detektörünü tasarlamamanın ve öğretmenin ilk deneyleri için, ayrıca, sunulan çalışmanın esnekliğini göstermek için , bir doğru (çalışma açısından doğruluk ) kontrol edilebilir bilgi kaynağına ihtiyaç duyulur. Bu bilgi kaynağını sağlamak için bir indüksiyon motorunun dinamiklerine benzetme yapılarak bir bilgisayar programı geliştirilir. Bu program, farklı hata şartları altında sinir ağı detektörünü test etmek ve öğretmek için gerekli bilgiyi genelleştirerek sabit bir yük altında çalıştığı varsayılan motorun performansının simülasyonunu gerçekleştirebilir. Yine bu bilgisayar programı

endüktans kaçağı, doyma etkileri (saturation effects) , yatak aşınması ve stator sargı izolasyon hataları gibi faktörleri içerir.

Yapılmış olan çalışmalarda bu program, sinir ağını öğretmek ve test etmekte sadece ihtiyaç duyulan giriş-çıkış ölçümlerini elde etmek için kullanılır. Bu çalışma içinde, bir tek-fazlı sincap kafesli indüksiyon motor dinamiklerinin simülasyonunu yapmakta kullanılmıştır.

Üzerinde çalıştığımız problemi fazla ayrıntıya girmeden formüle etmeğe çalışırsak , incelenmekte olan hata türleri sarımdan sarıma stator izolasyon hatası ve yatak aşınması olmak üzere , motorun diğer tüm parçalarının iyi bir durum içinde olduğunu varsayıyoruz. Bu kabul dahilinde , motor ölçümlerinin sahip oldukları başlangıç hatalarının etkilerini anlamak için ele alınan motor analiz edilir. Daha önce yapılmış mevcut çalışmalardan da yararlanarak, inceleme kapsamında bilgiyi bilgisayara aktarmak için, başlangıç hatalarını denetlemekte kullanılan değişkenler belirlenebilir.

Bu tür analiz tam doğru motor modellerine ihtiyaç duymaz. Şimdi motor parametreleri arasındaki ilişkileri kabataslak inceleyelim.

Stator sargı akımı değerlerini  $I$  ve rotorun hızını  $w$  ile gösterelim. Kalıcı halde,

$$\frac{dI}{dt} = \frac{dw}{dt} = 0 \text{ 'dır.} \quad (2.1)$$

Nonlineer cebirsel denklemleri bir sistem olarak ifade edersek , stator sargısının eşdeğer sarımlarının sayısı  $N$ , motorun sönüm katsayısı  $B$ , değişkenler olarak , kalıcı halde akım  $I$  ve rotor hızı  $w$  ile temsil edilmek üzere,

$$f = [f_1 \ f_2]^T \quad (2.2)$$

'den hareketle

$$f(I, w, N, B) = 0 \text{ 'dır.} \quad (2.3)$$

Bu denklem sargı ve yatakların durumunun belirtilerini stator akım ve rotor hızının ölçümlerinden elde edilebileceğini ileri sürer. Asıl sargıdaki , sargıdan sargıya izolasyon başlangıç hatası asıl sargının yerine geçen eşdeğer sargıların ,  $N$  , sayısını azaltacaktır. Bunun bir sonucu olarak, sargı direnç matrisi  $r$  ve sargı endüktans matrisi  $L$  ve diğer buna yakın motor parametreleri bundan dolayı değişecektir. Keza motorun yatak aşınması , sönüm katsayısı  $B$  'nin sayısal değerinde bir artışa sebep olacaktır.

Gerçekten yapılan analizden stator akımı ve rotor hızı , stator sargısının ve yataklarının değişme şartlarına çok duyarlı bir yaklaşımla bulunur. Bu yüzden, stator akımı  $I$  ve rotor hızı  $w$  sarım izolasyonu ve yatak hataları için ölçülmüş değişkenler olarak seçilir.

#### 2.4 Stator Akımı Ve Rotor Hızı Maliyet Fonksiyonları

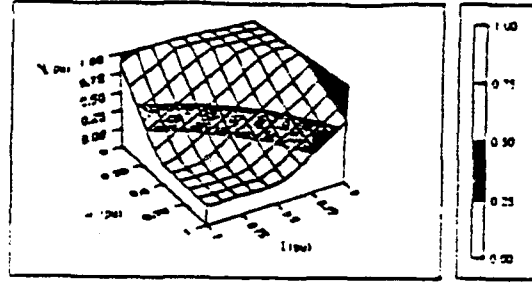
Birçok çalışma için, motorun niteleyici durumu yani doğrudan aritmatiksel değerleri içermeyen hali, parametrelerin sayısal değerlerinden daha çok, ilgi kapsamı içindedir. Motorun bu niteleyici durumu genellikle ihtiyaç duyulan operasyon durumunun maliyet fonksiyonuna dayandırılır. Burada sıcaklık modellenmesi ve motorun, motor olarak etkinliği böyle maliyet fonksiyonları için kullanılır.

Aşırı akım çekilmesi yüzünden sebep olunan aşırı sıcaklık, izolasyon hatalarının en büyük sebebidir (sınıf A'ya (A motorun tasarım sınıfını belirtmektedir) ait olan araştırma altındaki motor izolasyon türü). Uygun bir termal (ısı) model stator sargısında üretilen stator akımı ve ısıyla ilişki kurmak için kullanılır. Böylece akımla üretilen ısıya dayanan stator sargısının ömrü sargının durumunu belirlemek için bir maliyet fonksiyonu olarak kullanılır. Benzer şekilde, yatak durumu, motor etkinliği ile ilgili başka bir maliyet fonksiyonuyla belirlenir. Maliyet fonksiyonu haritalamaları yüksek derecede nonlineerdır ve sinir ağı hata tespiti kapasitesinin esnekliğini göstermek için seçilir.

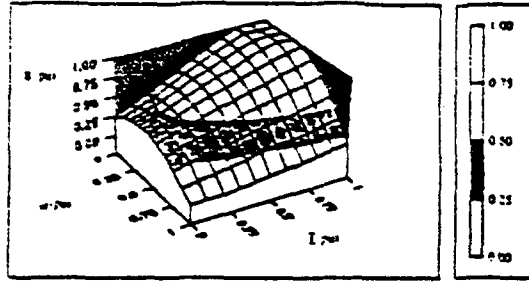
Hataların sınıflanmasının bu konu dahilinde çok hayali olduğu söylenebilir. Yine bu işlem operasyon durumu, ekonomik faktörler gibi birçok faktörlere bağlı olduğundan bu artık bir fuzzy lojik problemi olarak çözüme kavuşturulabilir. Birçok durumlarda, hataların durum bilgisi çalıştığı alanda kendisinin keşifsel ve deneysel tecrübesine dayanan bir uzman tarafından sağlanabilir (yani, uzman bir mühendis analog olarak yük hızını genellikle bir yük tahmin etme programından daha iyi tahmin edebilir). Sinir ağı, tekrar maliyet fonksiyonlarının nasıl seçildiğini bilmeğe ihtiyaç duymaz. Bu sadece sargıyı ve yatak durumunu (çıkış olarak) içeren öğretim modellerini bilmeğe ihtiyaç duyar (stator akımı ve rotor hızı ölçümleri (giriş olarak)).

Eşdeğer sargı sayısı  $N$  ve sönüm katsayısı  $B$  'ye bakışla sarımdan sarıma izolasyon hatası ve yatak aşınma hatası arasındaki ilişkiye dayanan motorun durumu üç hal seviyesinde sıralanabilir - iyi, orta, kötü.

Aşağıda verilen şekil 2.1 ve şekil 2.2 yapılmış bir çalışmanın motor dinamiklerinin bilgisayar simülasyonunda alınmış sonuçlarından elde edilmiştir. Burada yapılan çalışmanın  $N$  ve  $B$  hata bölgelerine göre şematik olarak gösterimi mevcuttur. Giriş olarak  $I$  ve  $w$  alındığını daha önceden belirtmiştik. Bu sonuçlandırma şemasında,  $g: R * R \rightarrow Z^2 \times N_{a,s}$  (statorun a fazı) ve  $B$  gerçek sayı uzayı  $R$ 'nin elemanıdır.  $Z$  ise yapılan çalışmada,  $Z = [0.9, 0.5, 0.1]$  olmak üzere göz önüne alınmış durum uzayıdır. Yani iyi, orta, kötü, olarak motor durumunu temsil eder (bu  $Z$  değeri yapılmış olan bir çalışmadan örnek olarak alınmıştır). Bu şekiller bize başlangıç hata tespitini yapmada bir sinir ağını tasarlamak için kılavuzluk etmektedir.



Şekil 2.1 Faz akımı I ve rotor hızı  $w$ 'nın ölçme zamanı üzerinden eşdeğer sarımlar  $N_{2s}$ 'in sayısal haritalaması



Şekil 2.2 Faz akımı I ve rotor hızı  $w$ 'nın ölçme zamanı üzerinden sönüm katsayısı B'nin haritalaması

Şekil 2.1 ve 2.2 . seçilen maliyet fonksiyonlarına dayanan ' iyi, orta, kötü ' bölgelerine ayrılmada motor için operasyon rejimlerinin bir süresini (anını ) göstermektedir.

Buradan hareketle bir takım sonuçlara varacak olursak , yine bu çalışma dahilinde yapılan yorumlardan da yararlanarak , 1 rpm (dakikadaki devir sayısı ) 'den daha iyi bir ayırt etme ile motor hızının ölçümü gibi tam doğru ölçümleri, ucuz malzeme ile başarmak oldukça zordur. Ancak gerekli koşulları sağlanmış ( öğrenme ve öğretme gibi ) sinir ağı hata tespit projesi kullanılırsa tam ölçme ve pahalı malzemeye ihtiyaç duyulmaz. Burada ölçme yanlışları, ölçme gürültüsü sıfır ortalamağa sahip ise, ortalama bir filtre kullanılarak , filtre edilebilen ölçme gürültüsü olarak, işlemden geçirilebilir. Sinir ağı performansının önemsiz miktardaki kaybı yapılan simülasyonlarda gözlenebilir. Yine yapılan bu çalışmada 10 büyüklüğünde bir örnekleme penceresi kullanarak % 0.05 ölçme yanlışlığıyla-yani yaklaşık olarak I için 0.035 A, w için 16 rpm değerleri elde edilmiştir. Bu göstermektedir ki, pahalı olmayan ölçü aletleri oldukça uygundur.

Bu kısmın bir sonucunu çıkaracak olursak, ağ girişleri olarak sinir ağı tam doğru ölçümlere ihtiyaç duymadığından , tam doğruluk verecek pahalı ölçüm aletlerine ihtiyaç yoktur. Böylece bu maliyetten kaçınılabılır. Daha büyük sinir ağını yaparak gereksiz bir maliyet yerine , ağ girişlerini arttırmakla, hem gürültü filitrelenebilir hem de hata tespiti yapılabilir.

### **3. İNDÜKSİYON (ASENKRON) MOTORLARDA HATA TESPİTİ**

#### **3.1 İndüksiyon Motoru**

Yaptığımız incelemede prototip olarak indüksiyon motorlarını ele aldığımızdan çalışmamızın bu kısmına, bu konuda bilgi vererek devam edeceğiz.

##### **3.1.1 İndüksiyon Motorlarının Yapısı**

İndüksiyon motorları endüstride en fazla kullanılan motorlardır. İndüksiyon motorlarının devir sayıları yükte çok az değişir, bu motorlar sabit devirli motorlar sınıfına girer. Doğru akım şönt motorlarının devir sayısı büyük sınırlar içinde değiştirilebildiği halde, indüksiyon motorunun devir sayısı sınırlı olarak bir veya iki kademe arttırılabilir veya azaltılabilir. Bu motorların tüm yönleri ele alındığında,

1. İndüksiyon motorları daha ucuzdur.
2. İndüksiyon motorları periyodik bakıma daha az ihtiyaç gösterir.
3. İndüksiyon motorlarının çalışması sırasında elektrik arki meydana gelmez, (D.A. motorları çalışırken kollektör dilimleri ile fırçalar arasında kıvılcımlar çıkar).

Bu özellikler, indüksiyon motorlarının endüstride en çok kullanılan motorlar olmalarına sebep olmuştur.

İndüksiyon motorları genel olarak stator ve rotor olmak üzere iki kısımdan yapılmıştır.

##### **3.1.2 Stator**

Stator, indüksiyon motorunun duran kısmıdır. Statorun oluşturulması: 0.4-0.5 veya 0.8 mm kalınlığında silisyumlu saçların özel kalıplarla preste basılır. Daha sonra üç fazlı sargıların yerleştirileceği stator nüvesi stator saçların paketlenmiş durumu denilen şekle getirilerek elde edilir. Gövdeye motor kapakları takılır, motor kapaklarının ortasındaki bilyalı yataklar statorun ortasında dönecek olan motora yataklık eder.

##### **3.1.3 Rotor**

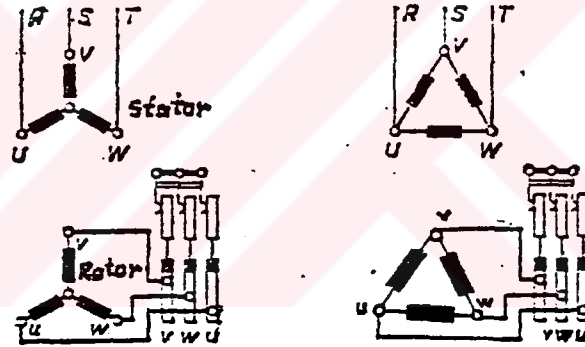
İndüksiyon motorunun dönen kısmına rotor denir. Sincap kafesli (kısa devreli rotor) rotor ve sargılı rotor (bilezikli rotor) olmak üzere iki çeşit rotor vardır.

### 3.1.3.a Sincap Kafesli Rotor (Kısa Devreli Rotor)

Stator gibi silisyumlu saçlar kalıpla preste kesilerek palet edildikten sonra rotor kanalları içine alüminyum eritilerek pres döküm ile kısa devre kafes sargıları meydana getirilir. Rotorun iki tarafında rotor çubuklarını kısa devre eden halkalarda küçük kanatçıklar oluşturulur. Bu kanatçıklar pervane ödevi görerek motorun soğumasını sağlar. Bazı büyük rotora sahip motorlarda kanallara bakır çubuklar yerleştirilir. Rotorun iki tarafına konan bakır halkalara bakır çubuklar kaynak edilerek sincap kafes oluşturulur.

### 3.1.3.b Sargılı Rotor (Bilezikli Rotor)

Saçları paketleyerek silindirik şekilde yapılan rotorun üzerindeki oluklara 3 fazlı alternatif akım sargısı yerleştirilir. 120 derece faz farklı olan üç fazlı sargılar yıldız veya üçgen bağlandıktan sonra çıkarılan 3 uç, rotor miline yalıtılarak yerleştirilen 3 bileziğe bağlanır.



Şekil 3.1 Sargı uçlarının bileziklere bağlanması ile 3 fazlı reostanın devreye sokulması

### 3.1.4 İndüksiyon Motorunun Çalışma Prensibi

3 fazlı, 2 kutuplu bir indüksiyon motoruna şebekenin RST faz emk'lerini uygulayalım. Statordaki sargılardan geçen alternatif akımlar, dönen NS kutuplarını meydana getirirler. Stator sabit olduğu halde, dönen NS kutupları ortadaki kısa devreli rotorun çubuklarını keserek çubuklarda emk'leri indükler. Kısa devreli rotor çubuklarından endüksiyon akımları geçer. Döner olan (NS) kutupları saat ibresi yönünde döndüğüne göre rotor çubuklarından geçen endüksiyon akımlarının yönleri sağ el kuralı ile rahatlıkla bulunur. Bu akımlar rotorun NS kutuplarını meydana getirirler. Döner stator kutupları rotorun kutuplarını etkileyerek (benzer kutuplar birbirini iter, zıt kutuplar birbirini çeker prensibinden hareket ile) rotoru saat ibresi yönünde döndürürler.

N kutbunun altındaki rotor çubukları bir yöne, S kutbunu altındaki rotor çubukları diğer yöne doğru itilirler. Bu itme kuvvetlerinin meydana getirdiği döndürme momenti rotoru saat ibresi yönünde, döner alanın yönünde döndürür.

Rotorun devir sayısı ( $n_r$ ) arttıkça, döner alanın rotor çubuklarını kesmesi azalacağından, rotor çubuklarında endüklenen emk'ler ve kısa devre çubuklarından geçen endüksiyon akımları azalır. Dolayısı ile, rotoru döndüren moment azalır. Rotorun devir sayısında artış olmaz. Motor boştta çalışırken rotorun devir sayısı senkron devir sayısına (döner alanın devrine) yaklaşır. Döner alanın devir sayısı (senkron devir)  $n_s$  ile rotor devir sayısı  $n_r$  arasındaki farka "Rotorun Kayması" denir. Diğer bir ifade ile, rotor devrinin senkron devirden geri kalmasına "Kayma" denilmektedir. Kaymanın senkron devir sayısı ile ifadesi,

$$s = \frac{n_s - n_r}{n_s} \cdot 100 \quad (3.1)$$

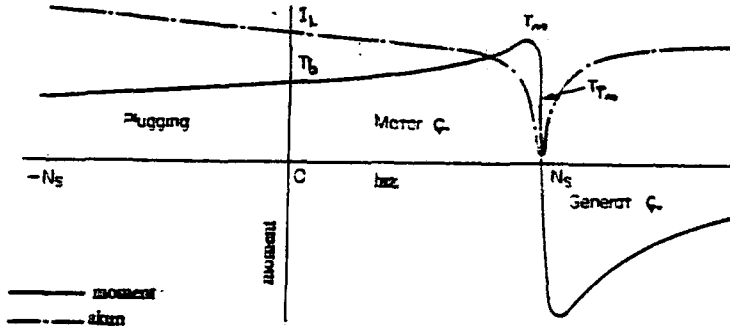
$$n_r = (1 - s) \cdot n_s \quad \text{dir.} \quad (3.2)$$

Bu formülün ışığı altında rotorun devir sayısı  $n_r$ , hiçbir zaman döner alanın devir sayısına yani senkron devire ( $n_s$ ) eşit olmaz. Bu da rotorun senkron devirden daha az bir devirle döndüğünün göstergesidir.

Bu arada her faz sargısında endüklenen emk'nun o sargıya uygulanan şebekenin faz emk'sına zıt yönde olduğunu, bu zıt emk'nun transformatörlerde primer sargıdan geçen alternatif akımın meydana getirdiği, çekirdekte dolaşan ve değişmekte olan manyetik akının primer sargı üzerinde endüklediği zıt emk'ya benzetilebildiğini hatırlatmakta yarar vardır.

### 3.1.5 Performans Karakteristikleri : Hız-Moment ve Hız-Akım Eğrileri

Şekil 3.2 de çok fazlı bir indüksiyon motoru için hız-moment ve hız-akım eğrileri verilmiştir. Bu eğrilerde üç büyük önemli bölge: motor çalışma, motorun bağlantılarını motorun zıt yönde çalışması eğilimi göstereceği biçimde düzenlemek suretiyle motoru frenleme ve generator çalışma.



Şekil 3.2 Çok fazlı bir indüksiyon motoru için hız-moment ve hız-akım eğrileri

Bu eğriler incelendiğinde, motorun çalışma prensibinden bahsederken ifade edilmeye çalışılan senkron hız  $n_s$ , kayma  $s$  gibi sayısal değerleri içeren kısımları da anlamak daha kolay olmaktadır. Yine bu eğri ile karşımıza çıkan kritik moment ifadesi, motor çalışma bölgesinde meydana gelen maksimum momenttir. Bu şekilde  $T_m$  ile gösterilmiştir. Rotorun bloke edilmesi ile ortaya çıkan moment ise şekilde  $T_b$  ile ifade edilmektedir. Bu da motor sıfır hızda (durma hali) iken motor tarafından üretilen momenttir. Aynı zamanda bu yük tarafından zorla gönderilen herhangi dağılık kuvvetlerin üzerinden gelmesi gereken önemli bir moment ifadesidir. Normalde bir motor maksimum moment  $T_m$  ve senkron hız  $n_s$  arasında çalışır. Yani egride  $T_{Tm}$  ile gösterilen nominal tam yük momenti ile hız ekseninde kalan yer çalışma bölgesidir.

### 3.1.6 İndüksiyon Motor Dinamikleri

Burada hata tespitinin yapılabilmesi için daha sonraki bölümler içinde karşımıza çıkacak olan motor dinamiklerini biraz tanıtmak amacı ile çok fazla kapsamlı olmayan bilgi verilecektir.

Bir bütün olarak beygir güçlü indüksiyon motorları arasında, yani güce ihtiyaç duyulan çalışma koşullarındaki motorlarda, en yaygın dinamik problemler motoru harekete geçirme (başlatma), motoru hareket dışı hale getirme (durdurma) ve mevcut sistemin ciddi parazitleri boyunca çalışma işlemine devam etmek için motorun çalışma kabiliyeti ile yakından ilgilidir. Örneğin, bir endüstri kuruluşunda karşılaşılan tipik bir problem, çalıştırılmaya başlanmış bir motora aşırı bir ani akımın gelmesi ile sebebiyet verilen gerilim azalmasından dolayı normal çalışmayı durdurmakta olup, diğer paralel bağlı motorlara herhangi bir müdahale de bulunmaksızın bahsetmekte olduğumuz büyük motorun çalışma kabiliyetine nasıl tesir edilebileceğidir.

Dinamik analizlerdeki indüksiyon motorunu ifade metodları, düşünülebilir bir kapsam dahilinde problemin yapısının ve kompleksliğinin tam veya kesin olarak tanımlanmış veya ifade edilmiş olma niteliğine bağlıdır. Yine bu analizler dahilinde elektriksel geçici rejimler makina dinamikleri için önemli bir yer işgal etmektedir. Eğer elektriksel geçici rejimleri ihmal edilebilir olarak düşünürsek olayı bir boyutta basitleştirmiş oluruz. Çünkü elektriksel geçici rejim isminden de anlaşılacağı gibi, çok hızlı bir şekilde çalışmakta olduğumuz elektriksel olayın dışına çıkmakta yani yatışmaktadır (hareketli rejimin sürekliliği ile karşılaştırıldığında geçici rejim çok kısa bir zaman diliminde çok hızlı olarak kaybolmaktadır). Bu durum için kabulumuz 3600 devir/dakika dan büyük motorlardır.

Bu şartlar altındaki motorun ifadesi kalıcı hal teorisine dayandırılabilir. Karşılaşılan problem daha sonra dinamik analizi, özellikle lineersizliklerin içinden uygunsuz bir hale getirmeyecek olan oldukça etkili derecede basit fakat oldukça mantıklı derecede gerçekçi ifade etmeyi amaçlar. Nispeten basit problemler için yapılan bir yaklaşım ise grafiksel yöntemdir. Hem motor tarafından üretilen hem de

yükü döndürmek için ihtiyaç duyulan moment, eğrisel formda verilen bilgi için hızın lineer olmayan fonksiyonları olarak düşünülür.

Yine motor dinamikleri içinde zamana karşılık düşen motor hızı bilinirse yine zamana karşılık gelen akım karakteristiği ise, ya hızı karşı bir akım eğrisi ya da analitik ifadelerle elde edilen bir sonuçtur. Endüvi akımının büyüklüğü, mıknatıslama reaktansının düzeninde etkin rotor empedansı olduğu noktada hız artuncaya kadar sayısal değer olarak büyük kalır. Bu noktada akım kalıcı hal çalışma değerine düşmeye başlar. Yüksüz halde bu akım motorun mıknatıslama akımıdır. Tam yük altında ise, bu akım mıknatıslama akımı ve stator yük akımının toplamını içerir.

Bazı durumlarda, nominal gerilim altındaki motoru başlatma akımı istek doğrultusunda aşılabilir. Böyle şartlar altında bir çok opsiyonlar mevcuttur.

Bu anlatılanlara belirli sınırlamalar getirmek ve motoru istenilen şekilde çalıştırmak için motorda seri empedans kullanımı, motora nominal gerilim uygulama, Y- $\Delta$  anahtarlama ve bölünmüş-sargı kullanımı (motoru harekete geçirme) gibi çalışmalar yapılarak değerce azaltılmış başlatma akımları ve momentleri elde edilir.

### 3.1.7 İndüksiyon Motorlarında Hız Kontrolü

Yapay sinir ağları ile hata tespiti olayı başlı başına bir kontrolü içerdiğinden bu noktada kısaca indüksiyon motorların kontrol olayından bahsedilecektir.

Basit bir indüksiyon motoru sabit hız sürücülerinin gereksinimlerini önemli ve yeterli derecede karşılar. Ancak bir çok motor uygulamaları bir kaç hızı veya kademeli olarak ayarlanabilir hızın sürekli değişimine de ihtiyaç duyar. AC güç sistemlerinin ilk tasarımı olduğu dönemlerde mühendisler ayarlanabilir hızı sahip ac motorların gelişimi ile geniş bir şekilde ilgilendiler.

Yapılan çalışmalar sonucunda bir indüksiyon motorunun senkron hızı,

1. Kutup sayılarının değişimi ile
2. Hat frekansının çoğaltılması ile

Kayma ise:

3. Hat gerilimini değiştirmekle
4. Rotor direncini değiştirmekle
5. Rotor devrelerinde uygun frekans gerilimlerini koymakla değiştirilebilir

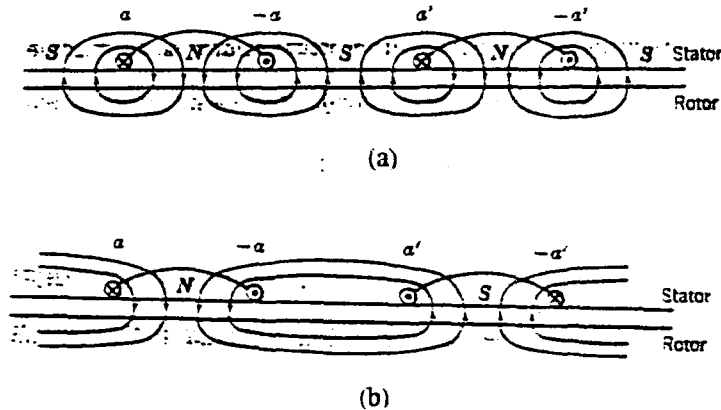
Hız kontrol metodları yukarıda verilen bu beş özelliğe dayanır.

### 3.1.7.a Kutup Değiştirme Kontrolü

Bu yöntemin temelinde, bir indüksiyon motorundan iki veya daha fazla devir elde etmek yatar. Buradan kısaca konuyu açıklayacak olursak, stator sargısı, kutup sayısını bobin bağlantılarında basit değişimler ile ikiye bir oranında değiştirebilmesi için tasarlanabilir. Ya da iki senkron hız seçilebilir. Burada rotor hemen hemen daima sincap kafeslidir. Bir kafes sargısı, daima stator alanı oluşturarak, aynı sayıda kutuplara sahip bir rotor alanı üretmek için reaksiyon gösterir. Bir sargılı rotor kullanılırsa ilave zorluklarla karşılaşılır. Çünkü, rotor sargısının böyle bir durumda kutup değişimi için yeniden düzenlenmesi gerekmektedir. Bu kutup değişimi için her biri düzenlenmiş iki bağımsız stator sargı grubu ile dört senkron hız, bir sincap kafesli motorda elde edilebilir (örneğin 600, 900, 1200, 1800).

Kutup değiştirme sargısının temel prensipleri şekil 3.3'de gösterilmiştir. Burada aa ve a'a'. a fazı stator sargısının iki bobinini kapsayan kısmıdır. Gerçek bir sargı elbette her bir grupta birkaç bobinden meydana gelecektir. Diğer stator fazları için sargılar benzer şekilde düzenlenir. Bobinler bir 4-kutup alanı üretecek şekilde birbirine bağlanır. a'a' bobinindeki akım bir kontrolör aracılığı ile ters yönde çevrilmiştir. Bunun sonucunda bir 2-kutup alanı oluşur. Kontrolör a'a' bobinlerinin ters yönde çevirdiği andaki iki bobin grubunun bağlantıları seriden paralele ve fazlar arasında Y'den  $\Delta$ 'ne veya tersi durumda  $\Delta$ 'den Y'a değiştirilebilir. Bu açıklama dahilinde, hava boşluğu akı yoğunluğu bu iki bağlantıda istenilen moment-hız karakteristiklerini üretmek için ayarlanabilir.

Bu kontrol türü farklı durumlarda motorun değişik hız ve momentlere ayarlanmasını sağlamaktadır. Yani aynı maksimum momentte değişik iki hızın kontrolü (sürtünmeli yükler), düşük hızda, maksimum momentin yaklaşık iki katında, sabit güce ihtiyaç duyulduğunda (vinçler), ve düşük hızda, maksimum momentten daha az bir değere ihtiyaç duyulduğunda (fanlar) bu kontrol türü rahatlıkla kullanılabilir.



Şekil 3.3 Kutup değiştirme sargısının prensipleri

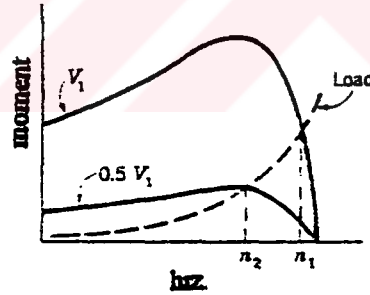
### 3.1.7.b Hat Frekans Kontrolü

Bir indüksiyon motorunun senkron hızı hat frekansını değiştirerek kontrol edilebilir. Yaklaşık olarak sabit akı yoğunluğunu muhafaza etmek için hat gerilimi frekans ile direkt olarak değiştirilebilir. Böylece maksimum moment hemen hemen sabit kalır. Bu şekilde kullanılan bir indüksiyon motoru sabit akı ve değişken endüvi gerilimi ile ayrı olarak tahrik edilmiş bir dc motor ile benzer karakteristiklere sahiptir.

Buradaki problem, en etkin ve ekonomik ayarlanabilir frekans kaynağını belirlemektir. Uygulanabilecek bir metod, frekans değiştirici olarak bir sargılı rotora sahip indüksiyon motoru kullanmaktır. Diğer bir yol ise, katı-hal (güç elektroniği) frekans dönüştürücülerinin kullanımınıdır.

### 3.1.7.c Hat Gerilim Kontrolü

Bir indüksiyon motoru tarafından üretilen iç moment motorun primer girişindeki gerilimin karesi ile orantılıdır. Yük, şekil 3.4'te gösterildiği gibi bir moment-hız karakteristiğine sahip olup, hız  $n_1$ 'den  $n_2$ 'ye düşürülecektir. Hız kontrolünün bu metodu yaygın olarak fanları süren küçük güçlü sincap kafesli motorlarda kullanılmaktadır.

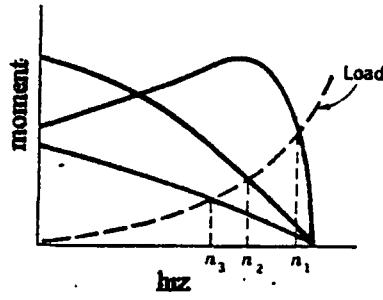


Şekil 3.4 Hat gerilimi ile hız kontrolü

### 3.1.7.d Rotor Direnç Kontrolü

Rotor direncini üç farklı değerleri için moment hız karakteristiği şekil 3.5'te gösterilmiştir. Yük, bu şekildeki gibi moment-hız karakteristiğine sahip ise, rotor direnç değerlerinin herbirine karşılık gelen hızlar  $n_1$ ,  $n_2$  ve  $n_3$ 'tür. Hız kontrolünün bu metodu endüvi ile seri olan direnç vasıtası ile dc şönt motor hız kontrolüne benzer karakteristiklere sahiptir.

Hem hat gerilimi hem de rotor direnç kontrolünün mevcut dezavantajları, yükü değiştirmeye göre düşük hızlarda ve zayıf hız regülasyonunda etkinliklerinin düşük olmasıdır.

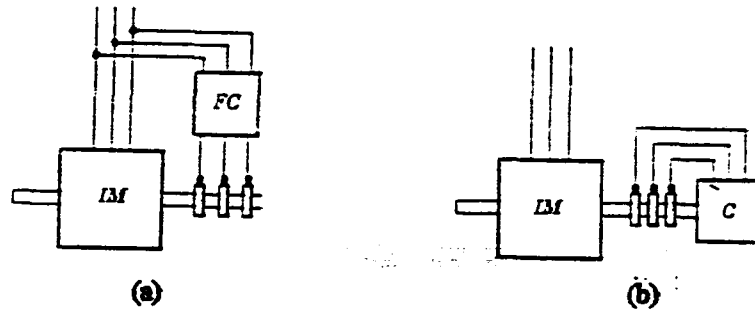


Şekil 3.5 Rotor direnci ile hız kontrolü

### 3.1.7.e Yardımcı Aletler İle Kayma Kontrolü

Kaymayı değiştirecek hız kontrolü için düşünülen çalışmalarda indüksiyon motorlarında güç akışını anlatan temel kanunlar akılda (insan proje üretme hafızasında) doğmalıdır. Statordan çekilen gücün  $s$  (kayma ile ilgili olan) kısmı rotor devrelerinde elektromanyetik indüksiyonla elektrik gücüne çevrilir. Rotor devreleri kısa devre ise bu güç rotorda  $I^2 R$  kaybı olarak harcanır ve düşük hızlarda bu çalışma etkisiz bir çalışma şeklidir.

Birçok proje bu kayma frekansı elektrik gücünü düzeltmek için üretilir. Bunların bazıları detayda oldukça komplike olduğu halde, bunların tamamı sargılı rotora sahip bir indüksiyon motorunun rotor devrelerine, kayma frekansının ayarlanabilir gerilimlerine ulaşmak için bir sebep teşkil etmektedir. Bunlar şekil 3.6'da görüldüğü gibi iki başlık altında sınıflandırılır. Bu şekilde İM, 3 fazlı sargılı rotorlu, hızı regüle edilmiş indüksiyon motorunu temsil etmektedir.



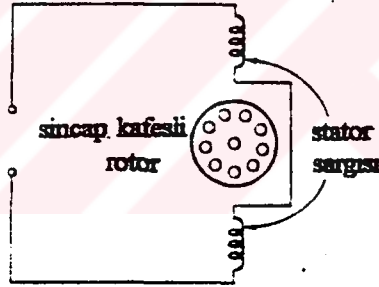
Şekil 3.6 Yardımcı aletler ile indüksiyon motoru hız kontrolü için iki temel şema

Şekil 3.6.a'daki İM'nin rotor devreleri yardımcı frekans değiştirme devresine (FC ile temsil edilen blok) bağlıdır. FC asıl motorun rotorunda meydana getirilen kayma frekans elektrik gücünü hat frekansındaki elektrik gücüne dönüştürür ve hatta geri gönderir. Şekil 3.6.b'de İM'nin rotor devreleri yardımcı devreye bağlıdır (C ile gösterilen blok). C kayma frekans elektrik gücünün mekaniksel güce

dönüştürüldüğü ve asıl motor tarafından geliştirilen mil gücüne eklenildiği elektriksel bir devredir. Her iki projedeki, asıl motor hız ve güç faktörü, yardımcı makinalarda kayma-frekans gerilimlerinin büyüklüğüne göre kontrol edilerek ayarlanabilir. Yardımcı devre, döner makinaların veya ayarlanabilir oranlı transformatörlerin tasarımılanmış oldukça komplike bir sistemi olabilir. Ya da katı-hal (güç elektroniği) frekans dönüştürücüsünden meydana gelebilir.

### 3.1.8 Tek-Fazlı İndüksiyon motorları

Yapısal olarak, tek-fazlı indüksiyon motorlarının en yaygın çeşitleri stator sargıları düzenli olanlar hariç, çok fazlı sincap kafesli indüksiyon motorlarına benzer. Bu tür kafesli bir rotor ve tek bir faza sahip stator sargısının durumu şekil 3.7'de gösterilmiştir. Yoğun bir bobin olduğu halde (yani tek bir bobin parçası şeklinde olması halinde), gerçek stator sargısı, manyeto motor kuvvetin yaklaşık olarak bir sinüsoidal uzay dağılımını üretmek için yarıklara dağıtılır. Böyle bir motor başlangıç momentine sahip değildir. Fakat harekete bir alet yardımıyla başlatılırsa çalışmaya devam edecektir.



Şekil 3.7 Tek-fazlı temel yapıda bir indüksiyon motoru

Şekil 3.7'deki tek fazlı stator sargısı için, stator akımı zamanın kosinüs fonksiyonu şeklinde ise manyeto motor kuvvet, boşluk ve zamanın bir fonksiyonu olarak,

$$f_1 = F_{1,\max} \cos \omega t \cos \theta \quad (3.3)$$

yazılabilir.

Burada pozitif ve negatif eşit büyüklükteki mmf (manyeto motor kuvvet) dalgasını da,

$$f^- = \frac{1}{2} F_{1,\max} \cos(\theta - \omega t) \quad (3.4)$$

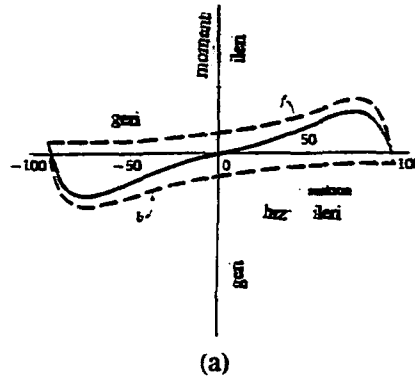
$$f^- = \frac{1}{2} F_{1,\max} \cos(\theta + \omega t) \quad (3.5)$$

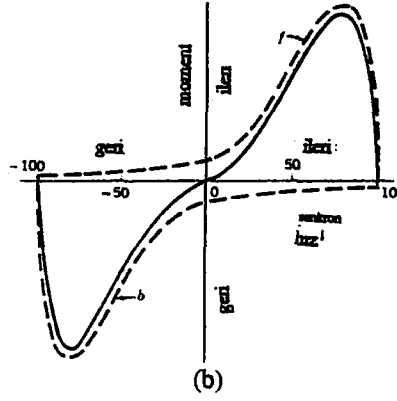
şeklinde yazabiliriz.

Manyeto motor kuvvet dalgalarının her bir ögesi indüksiyon motor hareketini üretir. Fakat buna karşılık gelen momentler ters yöndedirler. Hareketsiz rotorla, stator ve rotor akımlarının manyeto motor kuvvetinin birleşimi ile üretilen ileri ve geri hava boşluğu akı dalgaları birbirine eşittir. Yine burada işlemin elemanı durumunda bulunan momentler de birbirine eşittir, ve başlangıç momenti üretilmez. İleri ve geri hava boşluğu akı dalgaları rotor dönerken aynı kaldığı takdirde, elemansal alanların her biri ihmal edilebilir stator kaçak empedansına sahip çok fazlı bir motora, benzer bir moment-hız karakteristiği üretecektir. İki ögesel eğrinin matematiksel özeti olan sonuç, moment-hız karakteristiği, motor yardımcı bir alet tarafından harekete geçirilse, harekete başlatıldığı herhangi bir yönde moment üreteceğini gösterir.

Rotor hareket halinde iken hava boşluğu dalgalarının eşit kaldığı şekilde ileri sürülecek bir varsayım olayların gerçek durumunun oldukça zor bir ifade tarzı olacaktır. İlk bölgede, stator kaçak empedansının etkileri ihmal edilir. Fakat, rotor akımları ile sebep olunan etkiler bu şart altında uygun şekilde açıklanamaz. Aşağıda kısaca verilen açıklama, bir tek-fazlı indüksiyon motorunun performansının, birbirine eşit ileri ve geri akı dalgalarının temelinde, tahmin edilecek olandan daha iyi olduğunu gösterir.

Rotor hareket halinde iken, geri alan tarafından ortaya çıkarılan rotor akımları, durma ve güç faktöründe bulunduğu yerden değerce daha büyüktür. Bu, stator akımına karşı koyan manyeto motor kuvvet geri akı dalgasını küçültme ile sonuçlanır. Buna tezat olarak, ileri alan tarafından üretilen ögesel akımların manyetik etkisi durma anındakinden daha azdır. Çünkü rotor akımları daha az, fakat güç faktörü daha yüksektir. Bu yüzden hız arttığandan, geri akı azalırken ileri akı artar. Bundan dolayı, rotorun hareketi ile ileri alan momenti daha büyük, geri alan momentinde ise bu durum, şekil 3.8.a ' da gösterildiğinden daha azdır. Bu olayın doğru bir yaklaşım durumu şekil 3.8.b ' de gösterilmiştir. Yüzde birkaç kayma değerinde, normal çalışma bölgesinde ileri alan geri alandan birkaç kez daha büyüktür ve akı dalgası, dengeli çok fazlı bir motorun hava boşluğundaki döner alan sabit genişliğinden çok farklı olmaz. Bu nedenle, normal çalışma bölgesinde bir tek-fazlı motorun moment-hız karakteristikleri aynı rotora sahip ve aynı maksimum hava boşluğu akı yoğunluğu ile çalışmakta olan çok fazlı bir motordan aşırı derecede kötü durumda değildir.





**Şekil 3.8** Tek-fazlı bir indüksiyon motorunun moment-hız karakteristiği:(a) ileri ve geri sabit akı değişikliklerine göre (b) akıdaki sayısal değişikliklere göre

### 3.1.9 İndüksiyon Motorların Test Prosedürleri

İndüksiyon motorları için Amerikan test standardı, IEEE 112 6a dır. Bu standart genellikle testlerin nasıl yürütüleceğini tanımlar fakat makinanın karşılaşması gereken değerleri sağlamaz. IEEE test standardının amacı sadece test yapma prosedürlerini tanımlamaktır. National Electrical Manufacturers Association (NEMA) ve diğer endüstri standartları eğer bir motor özel bir standart türü ile kullanılmak için sınırlanursa bu durumda test sonuçlarının kesinlikle belirtir. Örneğin NEMA standardı bloke edilmiş rotor şartları altında maksimum akımı kesinlikle belirtmektedir fakat IEEE 112 6a ise bloke edilmiş rotor testinin sadece nasıl belirleneceğini tanımlar. IEEE'nin bu standardında tanımlanan testlerin bazıları,

1. Verim ve güç faktörünü belirlemek için performans testleri
2. Hız-moment ve hız-akım karakteristiklerini ölçmek için testler
3. Yüksüz haldeki test
4. Bloke edilmiş rotor testi
5. Yük altında çalışan makinanın sıcaklık artışı belirlemek için sıcaklık testleri
6. İzolasyon direnci
7. Yüksek potansiyel testi
8. Mil akımı ve yatak izolasyon testi
9. Titreşim testleri
10. Gürültü testleri.

Döner makinalar için bir çok uluslararası test vardır. Fakat Amerika'da bunlardan iki tanesi önemli yeri işgal etmektedir. Her ikisinin kullanımındaki önemi uluslararası ticarete bakış açısından ve uluslararası standartlardan kaynaklanmaktadır. Bu iki standart uluslararası Electrotechnical Commission Standard (IEC 34-2) ve Japon standardı JEC 37 dir.

Genelde bu testler motorların tasarımı için yapılmaktadır. Bu konunun kapsamı dahilinde, sadece motorların tasarım testlerinde izolasyon direnci ve yatak izolasyon testinin de yapıldığını göstermek için test prosedürlerinden burada bahsedilmiştir.

### ***3.2 Sincap Kafesli İndüksiyon Motorlarında Başlangıç Hata Testi***

İkinci bölümde döner makinaların genellikle iyi tasarmlandığından ve oldukça sağlam yapıda olduğundan söz etmiştik. Buna rağmen elektrik enerjisinin mekanik enerjiye veya mekanik enerjinin elektrik enerjisine dönüşümünde oluşan titreşimlerden dolayı bu makinalarda başlangıç hatalarının meydana gelme ihtimalinin makinanın yapısında doğal olarak varolduğundan da ayrıca bahsetmiştik.

Uygun sistem izleme ve hata tespit planları ile, bakım maliyetleri azaltılabilir ve makinaların güvenilirliği artırılabilir. Kendi alanında tecrübeli bir mühendis genellikle motorun çalışma performansını gözleyerek motor hatalarını keşfedebilir ve tanımlayabilir. Ancak, tecrübeli mühendislerin böyle bir iş için kullanımı maliyet açısından pahalı ve yetiştirilmeleri oldukça uzun zaman sürecini kapsamaktadır ve oldukça zordur. Buna ilaveten, sistem izleme ve hata tespit planlarını otomatikleştirmek, sürekli hattı (on-line) izleme işlemini yapan birkaç tecrübeli mühendise güvenmekten daha iyidir.

Diğer taraftan hata tespiti için yapay sinir ağlarının kullanımı motora dışarıdan müdahale ile, karşılaşılan problemlere çözüm bulan bir tekniktir. Fakat, parametre yaklaşım tekniğinden farklı olarak, sinir ağları ile ölçüm sonuçlarına dayanan kompleks ve kesin matematiksel modellere ihtiyaç duymaksızın çalışma yaparak hata tespiti yapılabilir. Ayrıca bazan öyle motor durumları ortaya çıkarkı sadece insanlar bu problemlerin tespit edilmesi ve giderilmesi işini yapabilirler. Halbuki bu zor motor şartları altında dahi karşılaşılan sorunlar, gerekli çalışmalar yapılarak sinir ağlarında kolaylıkla tanımlanabilir. Birçok hata tespit yöntemi için gereğinden fazla bilgiye ihtiyaç duyulur ve bu bilgiler bu yöntemlerin içinde yapılan araştırma sonucunda mevcut hale gelir, ve sonra bu bilgiler daha doğru sonuç elde etmek için matematiksel işlemler içinde kullanılır. Bu kavram, sinir ağı performansının gücünü (sağlamlığını) arttırmak için çok-girişli paralel işlem özellikleri ile beraber sinir ağını meydana getirmede kolaylıkla uygulanabilir.

### ***3.3 Motorlarda Hata Tespiti İçin Matematiksel Tanımlama***

Başarılı olarak hata tespitini gerçekleştirme olayında bir motorun farklı çalışma koşullarındaki durumunu tanımlamak için farklı kriterlere ihtiyaç duyulur. Burada bir split (ana sargı ile paralel bağlı yardımcı bir sargısı bulunan tek fazlı bir indüksiyon motoru: bu sargı ilk hareket için gerekli döner manyetik alanı oluşturmak üzere manyetik konumu bakımından ana sargıya göre deplase olmuş durumdadır. Yardımcı devre genellikle motor önceden belirlenen bir devre ulaştığında açılmaktadır.) fazlı sincap kafesli indüksiyon motorunun ulaşılmak istenen hedefleri göstermek için prototip olarak kullanıldığını daha önceden söylemiştik. Bu konu üzerinde gelecekteki tartışmaları aydınlatmak ve

kolaylaştırmak için matematiksel ifadelerle hata tespit problemini tanımlamak bu iş için harcanan zamana değer hale gelmiştir. aşağıda açıklamağa başlayacağımız kısım matematiksel yönden ortaya çıkan karmaşıklığa kısa bir tanımlama getirir.

Bu ön açıklamadan sonra motor dinamiklerinin matematiksel tanımına geçebiliriz. Motorun bilinen bir yük momenti ile sürekli (kalıcı) halde çalışıyor olduğunu kabul ettiğimizi tekrar belirtelim. Bir split (tek)-fazlı sincap kafesli indüksiyon motorunun stator akı bağlantıları ,

$$\lambda_s = [\lambda_{as}, \lambda_{bs}]^T \quad (3.6)$$

ve rotor akı bağlantıları,

$$\lambda_r = [\lambda_{ar}, \lambda_{br}]^T \quad (3.7)$$

durum değişkenleri olsun (a ve b burada sırası ile a ve b fazlarını temsil etmektedir ). Yine bu ifadelerde s indisi stator devreleri ile özdeşleştirilmiş değişken ve parametreleri, r indisi ise rotor devreleri ile özdeşleştirilmiş değişken ve parametreleri temsil etmektedir. Split (tek)-fazlı sincap kafesli indüksiyon motorunun dinamikleri aşağıda durum denklemleri ile beraber verilmiştir.

$$\dot{\lambda}_s = R_s i_s - V_s \quad (3.8)$$

$$\dot{\lambda}_r = R_r i_r - V_r \quad (3.9)$$

$$R_s = \begin{bmatrix} r_{as} & 0 \\ 0 & r_{bs} \end{bmatrix} \quad (3.10)$$

$$R_r = \begin{bmatrix} r_{ar} & 0 \\ 0 & r_{br} \end{bmatrix} \quad (3.11)$$

$r_{as}, r_{bs}, r_{ar}, r_{br}$  a ve b fazı stator sargı direnci ve rotor eşdeğer sargı direncidir.  $\lambda$  akı bağlantısı  $\lambda$  'nın zaman değişkenindeki türevidir. Stator sargı akımlarını,

$$i_s = [i_{as}, i_{bs}]^T \quad \text{ile} \quad (3.12)$$

rotor sargı akımlarını,

$$i_r = [i_{ar}, i_{br}]^T \quad \text{ile} \quad (3.13)$$

stator sargı gerilimlerini,

$$V_s = [V_{as}, V_{bs}]^T \quad \text{ile} \quad (3.14)$$

rotor sargı gerilimlerini,

$$V_r = [V_{ar}, V_{br}]^T \text{ ile} \quad (3.15)$$

gösterirsek ( rotor sargı gerilimleri bir indüksiyon motoru için sıfırdır.) kalıcı halde veya pertürbasyon (tedirgi) koşullarında motorun akı bağlantıları akımlara göre bir lineer ilişkiye yaklaştırılabilir. Bu yaklaşımı belirtirsek,

$$\begin{bmatrix} \lambda_s \\ \lambda_r \end{bmatrix} = \begin{bmatrix} L_s(\theta) & L_{sr}(\theta) \\ L_{sr}(\theta)^T & L_r(\theta) \end{bmatrix} \begin{bmatrix} i_s \\ i_r \end{bmatrix} \quad (3.16)$$

$L_s$  : stator endüktansı,

$L_r$  : rotor endüktansı,

$L_{sr}$  :stator rotor arasındaki karşılıklı endüktans,

$\theta$  : açısız olarak rotorun pozisyonudur.

Temel elektromanyetik teoriden , bir sargının akı bağlantısı bu sargının eşdeğer sarımlarının sayısının bir fonksiyonudur. a ve b fazları için eşdeğer sargılar, stator sargılarında,

$$N_s = [N_{as}, N_{bs}]^T \text{ ile} \quad (3.17)$$

ve rotor sargılarında,

$$N_r = [N_{ar}, N_{br}]^T \quad (3.18)$$

ile belirtilir.

Sargı direnci ve endüktansı gibi motor parametreleri eşdeğer sargıların değişen değerlerinden dolayı değişecektir. Eşdeğer sarımların farklı değerleri için aynı motor yapısı farklı bir çalışma üretecektir. Sincap kafesli bir indüksiyon motoru için , rotor oldukça güçlü ve sağlam yapıdadır ve  $N_s$  stator sargısındaki bozucu etkiden dolayı değerini değiştiriyor iken  $N_r$  'nin genellikle sabit olduğu varsayılır.  $N_s$  bir değişken olduğunda  $R_s$  ,  $L_s$  ,  $L_{sr}$  ise  $N_s$  'e bağlı fonksiyonlar olur ve  $R_s(N_s)$  ,  $L_s(N_s)$  ve  $L_{sr}(N_s)$  şeklinde belirtilirler. Böylece bir indüksiyon motorunun performansı bu motorun stator eşdeğer sargı sarımları  $N_s$  'in bir fonksiyonu olarak belirtilebilir.

Motorun elektrik momenti,  $T_e$

$$T_e = i_s^T \frac{\partial}{\partial \theta} L_{sr} i_r \quad (3.19)$$

denkleminde de görüldüğü gibi motor parametrelerinin ve durum değişkenlerinin bir fonksiyonudur. Buradan, motor  $N_s$  'in bir fonksiyonu olduğu görülmektedir ve bağımlılığını belirtmek için  $T_e(N_s)$  şeklinde ifade edilecektir.

Motor için hareket denklemi,

$$T_e(N_s) = J \dot{\omega} + B \omega + T_l \quad (3.20)$$

$\omega$  : rotor hızı  $\omega$ 'nin zaman değişkenindeki türevidir.

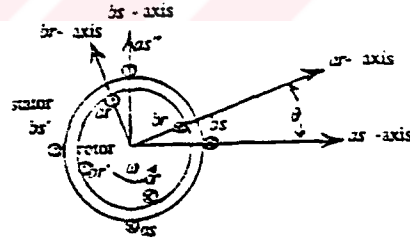
J: rotorun ataletidir ve yüke bağlıdır.

B: motorun sönüm katsayısıdır,

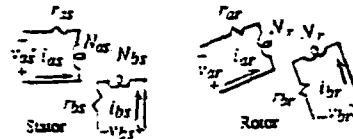
$T_l$ : bilindiği varsayılan yük momentidir.

Motorun geometrik sargı yerleşimi şekil 3.9.a ' da ve bunu temsil eden devre şeması da şekil 3.9.b ' de gösterildiği gibidir. Bu şekillerdeki as, bs, ar ve br eksenleri stator ve rotor kutuplarının a ve b fazları için manyetik akı eksenleridir.

Bir bölünmüş fazlı indüksiyon motoru için, motor genel bir bölünmüş fazlı kaynaktan gelen enerji ile enerjilenmiş her iki stator sargıları ile asıl (faz a) ve başlangıç (faz b) sargıları, harekete geçirilir. Rotor senkron hızın % 60 - %80 'ine ulaştığında başlangıç sargısı santrifüj tipinde bir ayırıcı anahtarla kaynaktan ayrılır. Bu yüzden, başlangıç sargısı kalıcı hal çalışmada, yani  $N_{bs}$ , kalıcı haldeki hata tespitinde yer almaz. Böylece notasyonun basitleştirilmesi için N.  $N_{bs}$ 'in ihmal edildiği yerde  $N_{as}$  'i yerleştirmek için kullanılır. Çünkü sadece kalıcı hal çalışmadaki hata tespitinde bu durum gözönüne alınır.



(a)



(b)

Şekil 3.9 (a) Tek-fazlı sincap kafesli bir indüksiyon motorunun geometrik sargı yerleşimi

(b) Geometrik sargı yerleşimine karşılık gelen akım diyagramı

$I, i_{as}$  'in değerlerini aldığını ve  $w$  'nın rotor hızının değeri olduğunu kabul edelim kalıcı halde 2.1, 3.8 - 3.13 (5 denklem) denklemlerini birleştirerek ve çalıştırarak değişkenler olarak da  $N$  ve  $B$  ile, kalıcı hal akımı  $I$  ve rotor hızı  $w$  'yı alarak bu lineer olmayan cebirsel denklemlerle denklem 2.2 ve 2.3 'deki gibi bir ifade elde edilebilir.

Biraz önce motor parametrelerini lineer olmayan cebirsel denklemlerle ifade edilebileceğini söylemiştik. Bu ilişki çerçevesinde denklem 3.20' deki indüksiyon motor dinamikleri ile, denklem 3.21 de gösterildiği gibi  $(I, w)$  ve  $(N, B)$  arasında bir ilişki meydana gelir.

$$M_1 : (I, w) \rightarrow (N, B) \quad (3.21)$$

İndüksiyon motorundaki mevcut lineersizliklerden dolayı  $M_1$ , yüksek derecede bir lineersizliğe sahiptir. Burada  $M_1$  'in tam doğru bir matematiksel modelini elde etmek zordur. Daha önceden de belirtildiği gibi asıl sargının koşulları ve motorun yatak aşınmaları, asıl sargı eşdeğer sınımlar  $N$  ' in ve sönüm katsayısı  $B$  ' nin sayısal değerlerinde ifade edilir. Yaptığımız inceleme için, nicelik olarak motoru tanımlayan  $N$  ve  $B$  değerleri, motorun durumunu bize niteleyici olarak belirleyen  $N_c$  ve  $B_c$  ' yi vermek için üç durum seviyesine (iyi, orta, kötü) ayrılır. Motorun durumunun bu şekilde niteleyiciyi tanımlaması hata tespit amaçları için oldukça uygundur. İkinci bir ilişki olarak  $M_2$  niceleyici tanımlama  $(N, B)$  ' den niteleyici tanımlama  $(N_c, B_c)$  ' ye belirtmek için kullanılır. Aşağıdaki denklem bu durumu en iyi şekilde belirtmektedir.

$$M_2 : (N, B) \rightarrow (N_c, B_c) \quad (3.22)$$

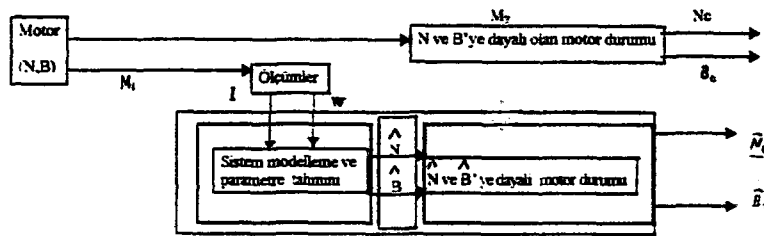
Bir sonuç olarak  $(I, w)$  ' dan  $(N_c, B_c)$  ' ye ilişki  $M_1$  ve  $M_2$  ' nin bir kompozisyonu olarak şu şekilde yazılabilir.

$$M = M_1 \cdot M_2 : (I, w) \rightarrow (N_c, B_c) \quad (3.23)$$

Az önce yukarıda belirtmiş olduğumuz notasyonları kullanarak bu yaklaşımı şu şekilde özetleyebiliriz: İlk önce  $M_1$  'e dayanan  $N$  ve  $B$  değerlerinin tahmini yapılır, sonra model  $M_2$  'ye ve sırasıyla  $\hat{N}$  ve  $\hat{B}$  şeklinde gösterilen  $N$  ve  $B$  ' nin tahmin değerlerine dayanan motor durumlarını belirler.

Daha sonra niteleyici motor durumu  $(\hat{N}_c, \hat{B}_c)$ ,  $(\hat{N}, \hat{B})$  ve  $M_2$  temeline dayandırılarak tahmin yapılır.

Bu çalışma şeklinin ayrıntılı planı şekil 3.10 ' da gösterilmiştir. Motor dinamiklerinin lineersizliğinin yüksek derecesi  $M_1$  ve ayrıklık  $M_2$  'den dolayı bu sistem çok kompleksdir. Böylece verilen bir indüksiyon motoru için böyle bir yaklaşımdan doğru bir matematiksel ifade elde etmek oldukça zordur.



Şekil 3.10 Motor hata tespitinde parametre tahmini / teşhis işlemi

### 3.4 Hata Tespitine Diğer Tekniklerin Yaklaşım Şekli

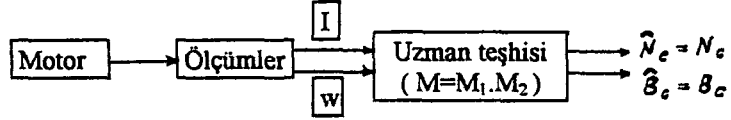
Bugüne kadar birbirini destekleyen veya birbirine karşıt olan bir kaç hata tespit metodu geliştirilmiştir. Bu tekniklerden bazılarında pahalı hata tanımlama aletlerine veya motor durumunu belirlemek için hat dışı (off-line) hata analizine gereksinim duyulmaktadır. Örneğin, radyo frekansı test planı önce bir makinanın stator sargısına radyo frekans sinyalleri gönderir ve sargı indüktansının hatalar içerip içermediğini belirlemek için sinyal dalgasındaki değişimleri ölçer. Bu teknikte öncekinde olduğu gibi pahalı aletlere ihtiyaç duyar ve sadece büyük ve pahalı makinalarda kullanılır. Parça analizi gibi, (motorun durumunu belirlemek için analizde kullanılmak üzere motor yağı örneklerinin bir laboratuara getirilmesine gereksinim duyan) diğer popüler teknikler, iyice inceleme ve periyodik kontrol için hatlı izleme ve hata tespitinden çok daha uygundur.

Başka bir hata tespit tekniği, parametre tahmin yaklaşımı dışarıdan bir motorun iç yapısına girmeden hata tespit planıdır. Böyle hata tespit planları motorun yapısını bozmaksızın motorun durumunu tahmin etme olayında pahalı olmayan ölçmelere dayanır. Bu planlar hatlı izleme ve hata tespit amaçları için uygundur. Ekonomik ve zararlı olmayan özelliklerinden dolayı, dışarıdan müdahaleye izin veren teknikler birçok mühendis tarafından oldukça sık tercih edilir. Ancak, parametre tahmin yaklaşımı doğru bir matematiksel modele ve sistem parametrelerinin yerleşimine dayandırılan sistem dinamiklerinin ayrıntılı bir şekilde anlaşılmasına ihtiyaç duyar. Parametreler genellikle motorun içinde bulunduğu durumları yansıtmak için seçilir. Örneğin, motor yatak milinin durumu, motor mekanik denkleminin sönüm katsayısını etkileyecektir. Yatak mili aşındığından sönüm katsayısı artar. Böylece, parametre tahmin yaklaşımı motorun mekanik denklemine ve sönüm katsayısının değerini tahmin etmek için ölçümlere ( rotor hızı gibi) dayandırılabilir. Seçilen parametrenin sayısal değerini tahmin ettikten sonra (yani motorun sönüm katsayısı) tahmin edilen sayısal değeri bir nicelik tanımlamaya (iyi veya kötü bir yataklama gibi) dönüştürmek için bir alete gereksinim duyulur. İşte bu noktada karşılaşılan sıkıntı doğru bir matematik sistem modeline (yani motorun mekanik dinamik denklemini) gereksinim duyulduğu ve bunu elde etmenin zor olduğudur. Ayrıca doğru matematiksel formüller kullanan ve bir fuzzy kavram olan hata durumlarının yorumu genellikle pratik ve doğru değildir.

Parametre tahmin tekniğinin performansı gerçek durumları yansıtmak üzere aşırı derecede biraz önce belirtilen  $M_1$  ve  $M_2$ ' nin doğruluğuna bağlıdır.

Yine bundan önceki bölümlerde, sayısal değerlere dayandırılmış bir motorun durumuna getirilecek bir yorumun, genellikle zor bir iş olduğundan bahsetmiştik. Çünkü böyle bir durumda hata açıklama, yani sayısal olarak belirtme fuzzy bir içeriktir ve genellikle tecrübe gerektirir. Bu yüzden birçok durumlarda sonuçların keşifsel yorumu (sadece insanların yapma kabiliyetinde olduğu) gerekli olur. Tecrübeli bir mühendis motorun tam bir matematiksel modelini bilmeksizin çalışma koşullarına ve ölçümlerine dayandırılan motorun durumunu teşhis ve tespit edebilir (şekil 3.11 ' de gösterildiği gibi). Bu

yaklaşım şekli basit ve güvenilirdir, ve 3.23 denklemindeki komplike ilişki M kesin olarak motor hakkında mühendisin bilgisine dayanır. Ancak, tecrübeli bir mühendis burada bu işi yaparken sebep göstermesini ve basit olarak kararları vermek için kullandığı mantığı belirtmesini gerektiren detaylı açıklamaları veremeyebilir, çünkü tecrübe bu alan dahilinde fuzzy lojik alana aittir ve tam matematiksel modeli doğru olarak tanımlaması oldukça zordur.



Şekil 3.11 Uzmanların motor hata tespit yaklaşımı / teşhis işlemi

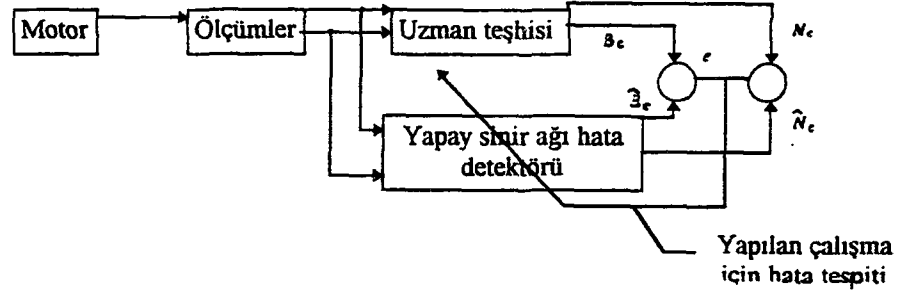
İnsanın tecrübe yaklaşımı parametre tahmin yaklaşımının üstünde birçok avantajlara sahiptir. Ancak tecrübe yaklaşımının büyük dezavantajı tecrübe dediğimiz şahsa ait bir birikimin, yani uzun sürece dayalı mesleki çalışmanın bir yerden bir yere transfer edilme durumunun hemen hemen hiç denecek kadar az olduğudur. Hem araştırmacılar hemde mühendisler bazen zaman kaybederek bazen de yanlış olarak deneyim ve bilgilerini uluslararası diller ve matematik içerisinde transfer ederler (tıpkı parametre tahmin tekniğinde olduğu gibi). Gerçekte, motor hata tespiti ve teşhisi yapmak için uzman mühendisler tarafından kullanılan tecrübe ve bilgi araştırmacılar tarafından üzerinde çalışılan bu konunun tarihsel gelişimine de bağlıdır.

Sinir ağlarından başka diğer metodlar da başlangıç hata tespiti için kullanılabilir. Ancak, çalışma bölgelerini ayıran hatlar şekil 3.1 ve 3.2 de gösterildiği gibi lineer değildir ve kolaylıkla belirlenemez. Birçok yönden karşılaşılan durumlara göre planlanmış bir sistem motorun çalışma bölgelerini ayırmak için kullanılamaz.

### 3.5 Hata Tespitinde Yapay Sinir Ağlarının Öğrenme Kabiliyetleri

Günümüzde ortaya çıkan sinir ağı teknolojisiyle, uzman yaklaşımı taklit edilebilir ve otomatikleştirilebilir hale getirilmiştir. Yapay sinir ağları temsil ettiği bir motor bilgi grubunu kullanan uzmanların bilgisini öğrenerek motor hata tespitini yapmak için eğitilebilir. Böyle bir plan şekil 3.12 'de gösterilmektedir. Yapay sinir ağının öğrenim ve öğretim olayının başında sinir ağı hata detektörü tarafından yapılan motorun durumunun denetlenmesi ve teşhisi - bizim özel incelememiz için  $\hat{N}_c$  ve  $\hat{B}_c$  - doğru olmayacaktır. Uzman tarafından yapılan doğru karar  $y = [N_c, B_c]^T$  ve sinir ağı tarafından yapılan  $\hat{y} = [\hat{N}_c, \hat{B}_c]^T$  arasındaki farka dayandırılan, bir hata miktarı

$e = (N_c - \hat{N}_c)^2 + (B_c - \hat{B}_c)^2$  genelleştirilir ve doğru karara yakın olan daha iyi bir  $\hat{y}$  vermek için sinir ağının giriş parametrelerini ayarlamakta kullanılır. Matematiksel modellere ihtiyaç duymaksızın sadece giriş-çıkış örneklerine dayandırılmış hata tespitini öğrenmek için bir sinir ağını öğreterek, parametre tahmin yaklaşımının karmaşıklığından kaçınılabilir. Sinir ağı uygun olarak öğretilirse, ağı ağırlıkları hata tespitini yapmak için ihtiyaç duyulan bilgiyi içerecektir.



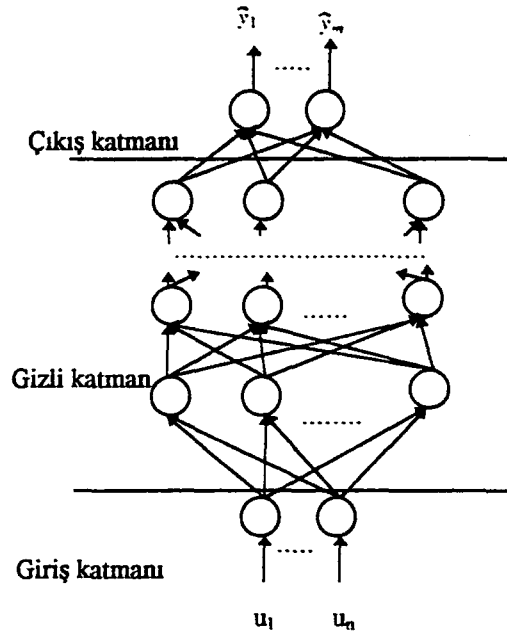
Şekil 3.12 Yapay sinir ağı hata detektörünü öğretmenin şematik diyagramı

### 3.6 Hata Tespiti İçin İleri-Besleme Ağ Modeli ve Geriye-Yayılım Öğretme Algoritmasına Bakış

Sinir ağlarının uzmanların bilgisinden yararlanıp nasıl öğrenme yapacağını teknik olarak tanımlayacak olursak, bir sinir ağının yapısını matematiksel terimlerle öğretmeyi açıklamak daha doğrudur. 4. bölümde ayrıntılı olarak, ileri-besleme ağlarının matematiksel tanımı ve geriye-yayılım (back propagation) öğretme algoritması yer almaktadır.

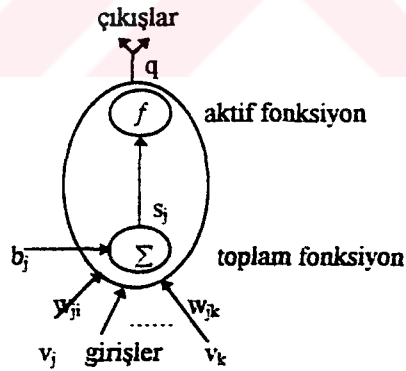
#### 3.6.1 Çok Katmanlı İleri-Besleme Yapay Sinir Ağlarının Tanımı

Temel çok katmanlı ileri-besleme ağı üç katman içerir: Giriş katmanı, gizli katman, çıkış katmanı (şekil 3.13). Bu çeşit sinir ağı bir giriş katmanı, bir çıkış katmanı ve herhangi bir sayıda bu ikisi arasında yer alan gizli katmanlara sahiptir. Her bir ağ katmanı düğümler ve sinirler denilen işleme üniteleri içerir. Bir ağ tabakasındaki her bir düğüm bir sonraki katmanın bütün düğümlerine kendi çıkışını gönderecektir. Giriş katmanında bulunan düğümler dış dünyadan gelen sinyalleri kabul eder, motorun akımı ve hız ölçümleri gibi (dikkate aldığımız örnekte görüleceği gibi) akım transdüserleri ve takometreleri vb. kullanır. Sinir ağını giriş katmanı bir ara yüzey olarak görev yapar ve dış dünyadan gelen bilgiyi alarak bunu ağı içindeki işleme ünitelerine transfer eder. Bu bir insanın gözünün retinası ve parmaklarımızın hissetme hücreleri gibi ara yüzey parçalarına benzer. Benzer şekilde, sinir ağının dış katmanı sinir ağının işleme ünitelerinden gelen bilgiyi dış dünyaya gönderen bir ara yüzey olarak çalışır. Gizli katmanlardaki ara yüzeyler sinir ağının işleme üniteleridir.



Şekil 3.13 Çok katmanlı bir ileri-besleme yapay sinir ağının temel yapısı

Genellikle ileri-besleme sinir ağlarında gizli ve çıkış katmanlarındaki tüm düğümler aynı yapıya sahiptir. Gizli ve çıkış katmanındaki her bir düğüm önceki katmanın düğümlerinden gelen  $v=[v_1, v_2, \dots, v_k]^T$  sinyallerini kabul eder. ağırlıklar ise  $w_j=[w_{j1}, w_{j2}, \dots, w_{jk}]^T$  tarafından ölçeklendirilmiş 1 nolu katmandaki  $j$ . düğüm  $j=1, \dots, k$  'ya takip eden niceliği sayar (şeki! 3.14).



Şekil 3.14 J. sinir ağı düğümünün temel yapısı

$$s_j = \sum_{i=1}^k w_{ji} v_i + b_j = w_j^T v + b_j \quad (3.24)$$

$b_j$ :  $j$ . nöronun relaksiyonu (manyetik kuvvet).

$s_j$ :  $j$ . nöronu  $o_j$  çıkışını vermek için genellikle bir non-lineer çalışma fonksiyonu.

$$o_j = f(s_j) \quad (3.25)$$

Çalışma fonksiyonu monoton olarak artan diferansiyel herhangi bir fonksiyon olabilir. Sigmoid fonksiyonu birçok biyolojik sınırların hareketine benzediğinden belkide en çok kullanılan çalışma fonksiyonudur. Buradan sigmoid çalışma fonksiyonunu yazacak olursak,

$$o_j = \frac{1}{1 + e^{-s_j}} \quad (3.26)$$

$s_j$ , 3.24 nolu denklemde belirtildiği gibidir ve  $o_j$ , j. düğümün çıkışıdır. 3.26 nolu denklemde tanımlanan girişe bakışla tipik bir sinir çıkışı sigmoid bir hareket fonksiyonu için şekil 4.7 de gösterilmiştir. Gizli ve çıkış katman düğümlerinden farklı olarak, giriş katmanındaki düğümler klasik olarak sigmoid bir fonksiyon yerine lineer bir hareket fonksiyonu kullanır ve giriş sınırlarına yapılan girişler  $v = u'$  dur, bunlar dış dünyadan gelen ölçülmüş sinyalleri temsil eder. Giriş-çıkış bilgisi sayısal kararlılık mantığı için [0.1] arasında klasik olarak normalleştirilir.

L ağdaki katmanların sayısını temsil etmek üzere girişler sıfır katmanı olarak sayılır yani .

$$o_i^0 = u_i \quad (3.27)$$

Üç katmanlı ileri-besleme ağlarının ( bir giriş katmanı, bir gizli katman, bir çıkış katmanı ) herhangi bir komplike değişen nonlinear fonksiyonları öğrenebildiği ispat edilebilir. Ancak, verilmiş bir problemi çözmek için gereksinim duyulmuş düğümlerin tam sayısını belirtmekte herhangi bir teori veya öğretim algoritması yoktur.Şu anda, özel bir mühendislik problemini çözmek için ağ konfigürasyonları tasarımı hala bir deneme yanılma işlemidir. Ancak, bu ispatın ifadeleri ileri-besleme ağının kabiliyetlerine teoriksel destek sağlar.

Farklı bilim adamları kendi çalışmalarını yürütmek için farklı ağ konfigürasyonları kullanmağı tercih eder. Bununla birlikte, üç-katmanlı bir ağ konfigürasyonu en popüler olandır ve bu göz önüne aldığımız çalışmalar içinde en çok kullanılandır. Üç katmanlı bir ağ  $Mn_i, n_h, n_o$  olarak gösterilebilir.  $n_i$  giriş düğümlerinin,  $n_h$  gizli düğümlerinin,  $n_o$  çıkış düğümlerinin sayısıdır.

### 3.6.2 Sinir Ağı Öğretmenin ( Öğrenme ) Matematiksel Formülasyonu

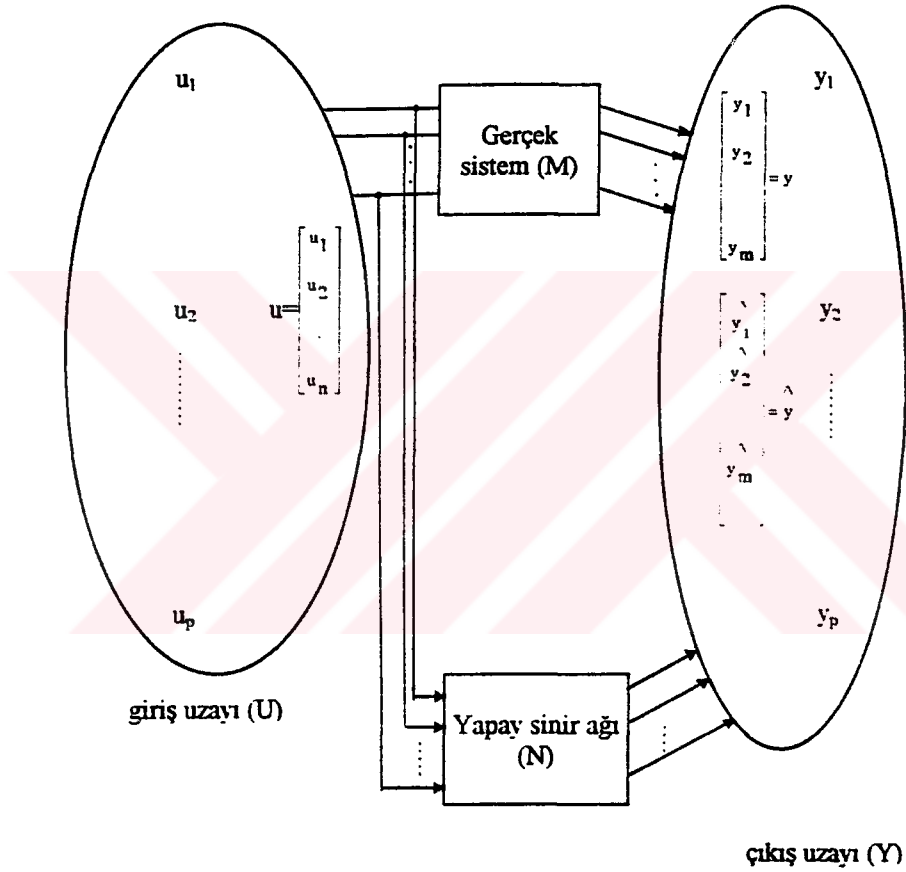
Bir yapay sinir ağını incelemenin birçok farklı yolu vardır. Sistem mühendisliği açısından bakacak olursak, bir sistemin giriş-çıkış haritalandırmasını öğrenme olarak bir sinir ağını göz önünde canlandırmak daha kolay olabilir. Örneğin, bu öğrenme kabiliyeti hata tespitini yapmak için kullanılabilir. Bu bölüm verilen bir sistemin giriş-çıkış ilişkisini öğrenmek için bir yapay sinir ağını öğretim işleminin matematiksel tanımını verir, sonra bir indüksiyon motorundaki hata tespitini yapmak

için tekniği genişletir. Böyle bir teknik uygun düzeltmelerle diğer mühendislik uygulamalarına genişletilebilir.

Şekil 3.15'de gösterilen bir sistemi ele alalım. Bu sistem  $u = [u_1, u_2, \dots, u_n]^T$  şeklinde girişlere ve  $y(u) = [y_1, y_2, \dots, y_m]^T$  şeklinde çıkışlara sahiptir. Giriş boşluğu  $U: \{ u \in U \mid u \text{ sisteme giriştir} \}$ 'dan çıkış boşluğu  $\eta: \{ y(u) \in \eta \mid y \text{ u girişine sistemin çıkışıdır} \}$ 'ya ilişki  $M$  olarak sistemi temsil etmesi uygundur.

$$M: U \rightarrow \eta$$

(3.28)



Şekil 3.15 Gerçek bir sistem ve bir yapay sinir ağının giriş-çıkış ilişkisinin şematik diyagramı

Karmaşıklığa sebep olmaksızın aynı notasyon  $M$  . denklem 3.23 ve 3.28'de giriş-çıkış haritalamasını belirlemek için kullanılır. Şimdi, bir yapay sinir ağının böyle bir ilişkisini ( $M$ ) öğrenelim. Yapay sinir ağının gerçek çıkışı . çıkış boşluğu  $\eta$  .  $w \in M$  ağırlıkla parametreleştirilmiş ağırlık uzayı, ve verilen giriş  $u$  . ait olan  $y$  ( $u \mid w$ ) olarak belirlenir.

Ağ öğretme işlemi matematiksel olarak temsil edilen ağ giriş parametreleri  $w$ 'yı ayar ederek mümkün olduğu kadar giriş-çıkış ilişkisi içinden  $M$ 'i temsil etmek için ağ öğretme olarak düşünülebilir. Bütün  $u \in U$  için matematiksel olarak  $w$ ,

$$\min_{w \in M} \|\hat{y}(u, w) - y(u)\|_2 \quad (3.29)$$

şeklinde yazılabilir.

Zorla yüklenmiş öğretilmedeki sınırlama ile ve böylece verilen aynı  $u$  girişi ile ,

$$\|\hat{y}(u, w) - y(u)\| \leq \varepsilon_a \quad (3.30)$$

bu denklem bazı uygun normda ve  $\varepsilon_a$  olduğu yerde ağ çıkışı  $\hat{y}(u, w)$  ve gerçek çıkış  $y(u)$  arasındaki farkın bir preset hata toleransıdır. Eğer  $\varepsilon_a$  çok küçük ise  $\hat{y}$  yaklaşık olarak  $y$ 'yi temsil eder.

Sistem  $M(u)$  'nun tam sürekli ilişkisi genellikle tam bilinmez. Eğer  $M(u)$  tam bilinse ilişkiyi öğrenmek için bir sinir ağını öğretmeye ihtiyaç yoktur. Ancak,  $u$ 'nun  $[(u_1, y_1), (u_2, y_2), \dots, (u_p, y_p)]$  ölçümlerinin ayrıık yerleşimleri ve  $y(u) = M(u)$  genellikle tam olarak elde edilir. Ağ öğretme işlemi , bir grup ayrıık giriş-çıkış ölçümlerinin  $M$ 'i öğrenmek için, ağ öğretmeden meydana gelir. Yapay sinir ağı terminolojisi bakımından,  $u_p$  p'nci giriş modeli,  $y_p = y(u_p)$  çıkışın yerini tutan model ve  $(u_p, y_p)$  p'nci öğretme modelidir. İndüksiyon motoru hata tespit incelemesi için .

$$u = [I, w]^T = [\text{satorakımı}, \text{rotorhızı}]^T \quad (3.31)$$

$$y = [N_c, B_c]^T = [\text{sarıım izolasyon durumu}, \text{motor yatak durumu}]^T \text{ 'dur.} \quad (3.32)$$

Sinir ağını öğretmek için  $p$  tane giriş-çıkış öğretme modelleri var olduğunu kabul edelim. Öğretme işlemi matematiksel olarak şu şekilde ifade edilebilir.

$$\min_{w \in M} \|\hat{y}(u_p, w) - y(u_p)\|_2 = \min_{w \in M} \quad (3.33)$$

$$\sqrt{\frac{1}{P} \sum_{p=1}^p (y(u_p, w) - y(u_p))^2} \quad . \quad p = 1, 2, \dots, P \quad (3.34)$$

böylece denklem 3.34  $\leq \varepsilon$  ve  $\varepsilon$  bir preset hata toleransıdır. Eğer denklem 3.34 biraz önce söylediğimiz gibi,  $\varepsilon$  'dan küçük ise, ağın gerçek sistem haritalamasını öğrenmiş olduğu söylenirse yani, denklem 3.30'daki , öğretme bilgisinin sistem hareketini temsil edebildiğidir. İki norm ölçü, sinir ağı öğretilmede

kullanılan en yaygın ölçümlerden biri, 3.34 nolu denklemde kullanılmaktadır. Ancak, diğer uygun norm ölçümleri bazı özel uygulamalara göre kullanılabilir.

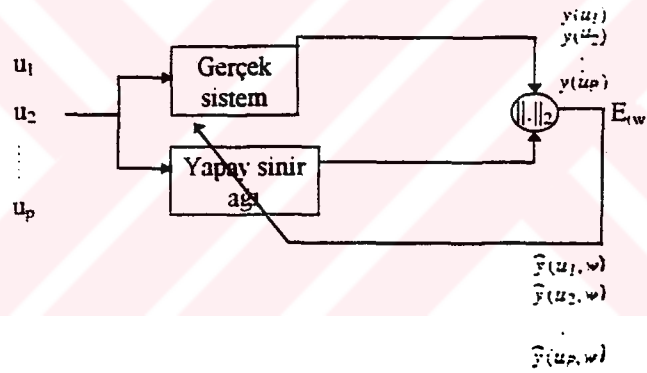
Öğretme modeli  $p$ 'nin öğretme hatası olarak  $e_p$ 'yi belirleyelim, yani  $u_p$  girişi ve geçerli akım ağırlığı ile, ağınc çıkışı ve sistemin çıkışı arasındaki fark,

$$e_p = e(u_p, w) = \| \hat{y}(u_p, w) - y(u_p) \|. \quad (3.35)$$

Ağı, öğretme boyunca, bütün öğretme modelleri için hatayı minimize edecektir (tek bir öğretme modeli için hatayı minimize yapmaktan daha çok). Bu yüzden ağınc öğretme hatasını bir nicelik  $E$  olarak belirleyelim,

$$E(w) = \frac{1}{P} \sum_{p=1}^P e(u_p, w) \quad (3.36)$$

Öğretme işleminin şematik diyagramı ise şekil 3.16'da gösterildiği gibidir.



Şekil 3.16 Yapay sinir ağınc öğretme

Çok katmanlı ileri-besleme ağınclarının nasıl öğretileceği yani 3.36 nolu denklemdeki  $E$ 'nin nasıl minimum yapılacağı, yakın zamana kadar sinir ağıncın başarılı bir şekilde incelenmesi için büyük bir engeldi. 1970'lerde çok katmanlı ileri-besleme ağınclarında karşılaşılan bu tür birkaç engeli ortadan kaldıran öğretme algoritmaları geliştirildi. Şimdilik, sinir ağıncını öğretmek için mevcut geriye-yayılım (backpropagation) algoritması ve simüle edilmiş uygunlaştırma arasında birkaç iyi bilinen öğretme algoritması vardır. Bu metodlar yineleyici olarak ağınc ağırlıklarını ayarlayarak  $E$ 'yi minimum yapmak için ağınc öğretir. Geriye-yayılım algoritması ileri-besleme ağıncı için büyük avantajla en uygun ve en iyi öğretme algoritmasıdır ve özet olarak daha sonraki bölümde ayrıntılı olarak kendisinden bahsedilmektedir.

Geriyeye yayılım öğretme algoritması temel olarak denklem 3.36'daki  $E$  hatasını minimum yapmak için optimal bir  $w$ 'yi araştıran en hızlı bir iniş metodudur. Ağ ağırlıkları iterasyona tabii tutulmuş olarak şu şekilde güncelleştirebiliriz.

$$w(\text{itera} + 1) = w(\text{itera}) - \eta \left. \frac{\partial E(w)}{\partial w} \right|_{w=w(\text{itera})} \quad (3.37)$$

Burada itera akmakta olan öğretme iterasyonunun sayısı ve  $\eta$  öğrenme oranıdır ( genellikle küçük pozitif bir rakamdır, 0.1 gibi).

Çıkış katmanı için, 3.37 nolu denklemdaki nicelik  $\frac{\partial E(w)}{\partial w}$ , gizli katmanlardaki ağırlıklar için  $\frac{\partial E(w)}{\partial w}$  gizli katmanlara çıkış katmanında elde edilmiş hata sinyalini geri göndermek için zincir kuralının kullanımına gerek duyuyor iken, çıkış katmanının  $w$  ağırlıklarına göre  $E$ 'nin türevini alarak kolaylıkla toplanabilir. Ağ ağırlıklarını adapte etmek için kullanılan delta kuralından daha önce kısaca bahsedilmişti.

$$\delta_j = o_j(1 - o_j)(y_j - o_j) \quad \text{çıkış katmanı düğümü } j \text{ için.} \quad (3.38)$$

$$\delta_j = o_j(1 - o_j) \sum_k \delta_k w_k \quad \text{gizli katman düğümü } j \text{ için} \quad (3.39)$$

$$\Delta w_{ji}(\text{itera} + 1) = \eta \delta_j o_i + \alpha \Delta w_{ji}(\text{itera}) \quad (3.40)$$

$$w_{ji}(\text{itera} + 1) = w_{ji}^n + \Delta w_{ji}(\text{itera} + 1) \quad (3.41)$$

$$\delta_j o_j = - \left( \frac{\partial E(w)}{\partial w} \right), \quad 3.37 \text{ deki iterasyon işleminin iniş eğimidir. Denklem 3.40 da, } \eta \text{ öğrenme oranı}$$

( en hızlı iniş algoritmasındaki adım büyüklüğüne eşdeğer) ve  $\alpha$  momentum oranıdır. Denklem 3.40 'a eklenmiş fazla terim  $\alpha \Delta w_{ji}(\text{itera})$  denklem 3.37 ile karşılaştırıldığında momentum terimidir ama ilave olarak yerel minimumdan kaçınmak için ağırlık denkleminde yer almaktadır.  $\eta$ ,  $\alpha$ 'nın seçenekleri (sayısal değerleri) daha sonra sunulur.

## 4. YAPAY SINIR AĞLARI

### 4.1 Giriş

Yapay sinir ağları ya da kısaca YSA: insan beyninin çalışma sisteminin yapay olarak benzetimi çabalarının bir sonucu olarak ortaya çıkmıştır. En genel anlamda bir YSA insan beynindeki birçok nöronun, ya da yapay olarak basit işlemcilerin birbirlerine değişik etki seviyeleri ile bağlanması ile oluşan karmaşık bir sistem olarak düşünülebilir. Önceleri temel üp birimlerinde insan beynindeki nöronların matematiksel madelleme çabaları ile başlayan çalışmalar, geçtiğimiz on sene içerisinde, disipline bir şekil almıştır. YSA bugün fizik, matematik, elektrik ve bilgisayar mühendisliği gibi çok farklı bilim dallarında araştırma konusu haline gelmiştir. YSA'nın pratik kullanımı genelde, çok farklı yapıda ve formlarda bulunabilen bilgi verilerini hızlı bir şekilde tanımlama ve algılama üzerinedir. Aslında mühendislik uygulamalarında YSA'nın geniş çaplı kullanımının en önemli nedeni, klasik tekniklerle çözümü zor problemler için etkin bir alternatif oluşturmasıdır.

BİLGISAYAR	İNSAN BEYİNİ
Sayısal	Analog
Seri	Paralel
Komut kümeli	Bilgiye adapte olma
Yanlış hesaplamalar sonucu etkiler	Birimlerin ana işlemlere etkisi azdır
Giriş verilerindeki hatalar sonucu etkiler	Giriş verilerindeki hatalara her zaman duyarlı değil

### 4.2 Tarihsel Gelişimi

1940'larda Mc. Culloch ve Pitts nöronun lojik fonksiyonlarını sağlayan basit bir eşik cihazı olarak modellenebileceğini gösterdi (Mc. Culloch et al, 1943). Aynı zaman aralığında mühendislik temelleri geri besleme ve beyin fonksiyonlarından faydalanan Wiener, sibernetikini temelini atıyordu. 1949'da Donald Hebb hücrese seviyede beynin öğrenme mekanizması konusunda bazı çalışmalar yapmıştır (1949). Hebb'ın biyolojik öğrenme kuralı, bir nöronun dentrit yoluyla gelen bir aksonal girişinin bir darbe üretmesine sebep olur. Sonraki aksonal girişlerin darbe üretmesi olasılığı artar. Böylece yapılan davranışın mükafatı ortaya çıkar. Hızlı hesaplama yönelik ilk YSA çalışmaları 1950'li yıllarda başlamış ve basit nöron modellerine dayalı bir hesaplama modeli 1950'lerde Rosenblatt tarafından ortaya

atılmıştır (1958). 1960'lı yıllarda Widrow ve Hoff, bu basit nöron modellerini kullanarak ilk öğrenebilen adaptif sistemler üzerinde çalışmıştır (Widrow et al, 1960). Ancak 1969'da Minsky ve Papert yayınladıkları Perceptron adlı bir kitapta YSA yardımı ile öğrenmede ve hesaplamada aşılması zor engeller olduğunu iddia etmişler ve bu iddia YSA konusundaki çalışmalarını büyük ölçüde yavaşlatmıştır (Minsky et al, 1969).

1982 yılında artık YSA teknolojisinin teori kısmı yapılanmasını hemen hemen tamamlamıştı ve 1982'de J.J. Hopfield tarafından yayınlanan "Neural Networks and Physical systems" adlı çalışma ile çağdaş YSA devri başlamış oluyordu (1982). Bu çalışmada Hopfield nöronların karşılıklı etkileşimlerine dayanan bir nöral hesaplama modeli önermiştir. Model bir enerji fonksiyonunu alabileceği en az değerine indiren 1. mertebe lineer olmayan diferansiyel denklemlerden oluşmuştur. Hopfield ağ seviyesinde, tek tek nöron seviyesinde varılmayan hesaplama kapasitesinin bulunduğunu öne sürdü. Bu tür YSA'na Hopfield ağı denmektedir. Hopfield'in geri beslemeli YSA modelini ortaya atması ve bunun pratik optimizasyon problemlerinde kullanılabilirliğini göstermesi YSA konusundaki çalışmalarını bu noktada yeniden hızlandırmıştır.

1976'da Grossberg ART(Adaptive Resonance Theory) adında bir YSA yapısı geliştirdi(1976). ART çok gelişmiş bir YSA'dır ve henüz çok fazla probleme uygulanmamıştır. O sıralarda Kohonen de (Self- Organizing Maps) "kendini düzenleyen nitelik haritasını" geliştireyordu(1982). Bu YSA nümerik aerodinamik akış hesaplamaları için çoğu algoritmik yöntemden daha etkili olmuştur.

1986'da Rumelhart ve arkadaşlarının "Parellel Distributed Processing" grubu ileri beslemeli modellerde yeni öğrenme modeli olan hatanın geriye yayılması algoritmasını (error back propagation algorithm) geliştirerek, bu konudaki daha önce iddia edilen aksaklıkların aşılabileceğini göstermişlerdir (Rumelhart et al, 1986). Bugün endüstride birçok YSA uygulamasında bu öğrenme yöntemi ile bunun değişik varyasyonları kullanılmaktadır. Back-propagation algoritması, çok kullanılan, öğrenilmesi kolay bir ağıdır.

Widrow ve öğrencileri ise ADALINE'den (ADaptive LINEar Element) sonra MADALINE'yi geliştirip bu YSA'ya uygulama alanları buldular. MADALINE I'i 1987'de MADALINE II, 1988'de (David Andes'in keşfettiği) MADALINE III takip etti (Widrow et al, 1990).

Günümüzde YSA'nın teorik çalışmaları büyük ölçüde tamamlanmış olup, 1986'dan bu yana uygulamaya yönelik çalışmalar son derece yoğun bir şekilde devam etmektedir.

### ***4.3 YSA'nın Tanımı ve Modeli***

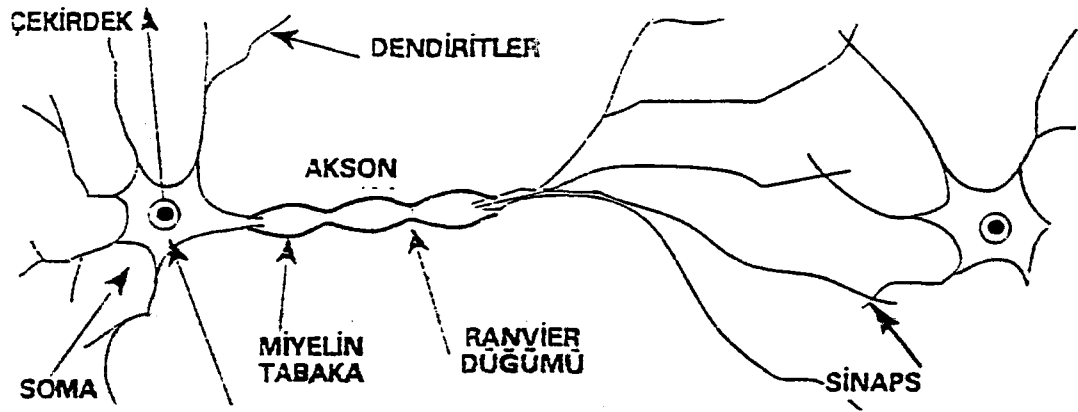
#### ***4.3.1 YSA'nın Tanımı***

YSA paralel dağılmış bir bilgi işleme sistemidir. Yani, YSA'nın temelinde, zeka gerektiren

işlemlerden oluşan bilgi işleme işlevi vardır. Bu sistem tek yönlü işaret kanalları (bağlantılar) ile birbirine bağlanan işlem elemanlarından oluşur. Çıkış işareti bir tane olup isteğe göre çoğaltılabilir. YSA yaklaşımının temel düşüncesiyle, insan beyninin fonksiyonları arasında benzerlik vardır. Bu yüzden YSA sistemine insan beyninin modeli denilebilir. YSA çevre şartlarına göre davranışlarını şekillendirebilir. Girişler ve istenen çıkışların sisteme verilmesi ile kendisini farklı cevaplar verebilecek şekilde ayarlayabilir. Ancak son derece karmaşık bir içyapısı vardır. onun için bugüne kadar gerçekleştirilen YSA; biyolojik fonksiyonların temel nöronlarını örnek alarak yerine getiren kompozit elemanlardır.

#### 4.3.2 Nöronun Biyolojik Yapısı ve Nöron Modeli

İnsanın bilgi işleme olayı beyinde gerçekleşir. Gerçekte en karmaşık sinir ağı Cerebral Cortex denilen "beyin"dir. Sinir sisteminin en basit yapısı nöronlardır. Beyinde yaklaşık olarak  $10^{10}$  sinir hücresi vardır. Yine hücre başına bağlantı sayısı ise  $10^4$  mertebesindedir. Beyin için çalışma frekansı 100 Hz'dir. Fiziksel boyutları ise 1.3 kg ve  $0.15 m^2$  kesitlidir. Vücudun değişik yerleri ile bilgi alışverişi yapan nöron hücresidir. Bir biyolojik sinir hücresinde, stoplazma dengede iken hücreler arası sıvıya göre  $-85 mV$ 'luk bir gerilim sınırına kadar polarizedir. Bu potansiyel farkı  $50 \text{ \AA}$  kalınlığındaki fosfolipid moleküllerinin iki katmanlı olarak düzenlenmesinden oluşmuş hücre zarı üzerindedir. Şekil 4.1 de basit bir nöron hücresi görülmektedir.

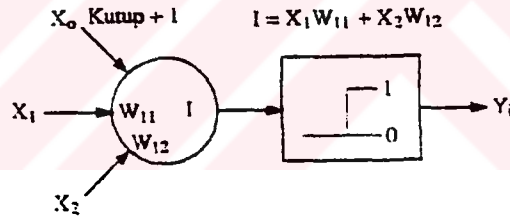


Şekil 4.1 Basit bir nöron yapısı

Nöron, soma adı verilen hücre gövdesi dendrit denilen kıvrımlı uzantılar ve somanın dalları sayesinde nöronu dallarına bağlayan tek sinir fiberli aksondan oluşur. Dendrit'ler hücreye gelen girişleri toplarlar. Dendrit tarafından alınan işaretler hücrede birleştirilerek bir çıkış darbesi üretilip üretilmeyeceğine

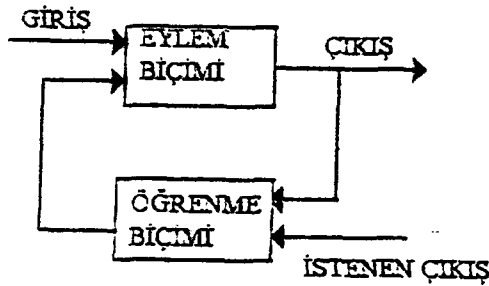
karar verilir. Eğer bir iş yapılacaksa üretilen çıkış darbesi aksonlar tarafından taşınarak diğer nöronlarla olan bağlantılara veya terminal organlara iletilir (akson tarafından bir elektriksel darbenin üretilmesi için,  $-40$  mV'luk eşik gerilimini aşacak ve hücre zarını depolarize edecek yönde stoplazmaya akım gönderilmelidir. Bu darbenin biçimi darbeyi üreten sistemden bağımsızdır). Beyindeki kortekste her nöronun bir karşılığı vardır. Bir nöronun çıkışı ona bağlı olan bütün nöronlara iletilir. Fakat korteks, işin yapılabilmesi için hangi nöron harekete geçirilecekse, sadece ona komut gönderir.

Somanın içinde ve çevresinde sodyum, kalsiyum, potasyum ve klor iyonları vardır. Potasyum yoğunluğu nöronun içinde, sodyum yoğunluğu dışındadır. Somanın zarı elektriksel olarak uyarılınca (söz konusu uyarı genellikle bir gerilim düşmesidir) zar, Na ve Ca gibi diğer iyonların içeri geçmesine izin verir ve somanın iç durumunu değiştirir nöronlar arasındaki bağlantılar hücre gövdesinde veya "sinaps" adı verilen dendritlerdeki geçişlerde olur. Yardımcı bir benzetme aksonlarla, dendritleri elektrik sinyallerini nörona ileten değişik empedansdaki yalıtılmış iletken olmasıdır. Sinir sistemi milyarlarca nöron ile tek bir nörondan çıkan aksonun 10000 kadar diğer nöronu bağlayan bir ağdır. Sinapslarla düzeltilen işaretleri taşıyan aksonlar ve dendritlerle içiçe geçmiş önörönlere bir sinir ağı oluştururlar. Şekil 4.2'de en basit formda gösterilen nöron modeli, bir eşik birimi olarak algılanabilir.



Şekil 4.2 Nöron modeli

Şekil 4.3'de ise YSA'nın genel blok şeması gösterilmektedir.

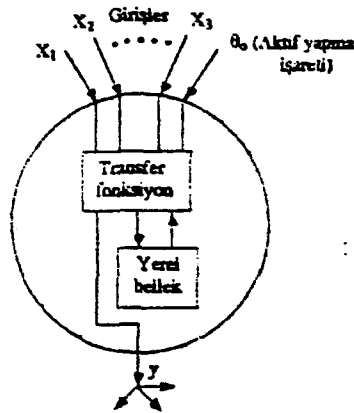


Şekil 4.3 YSA'nın genel blok şeması

Eşik birimi, çıkışları toplayan ve sadece girişin toplamı iç eşik değerini aştığında bir çıkış üreten işlem elemanıdır. Bir eşik birimi olarak nöron sinapslarındaki işaretleri alır ve hepsini toplar. Eğer toplanan işaret gücü eşiği geçecek kadar güçlü ise diğer nöronları ve dendritleri uyaran akson boyunca bir işaret gönderilir. Kesişen dendritlerden gelen sinapslarla kapılanan bütün işaretleri soma toplar. Toplam işaret daha sonra nöronun iç eşik değeri ile karşılaştırılır ve eşik değerini aşmışsa aksona bir işaret yayar. YSA, bu basit nöronların (düğümün ya da ünitelerin) bağlanarak bir ağ'a dönüştürülmesiyle meydana getirilir.

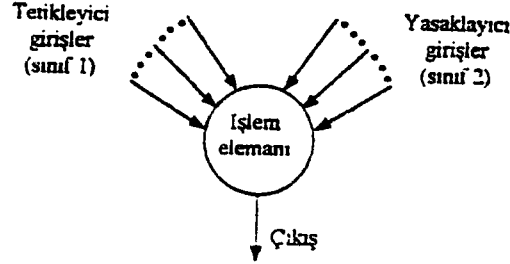
#### 4.4 YSA'nın Yapısı ve İşlem Elemanı

YSA temel olarak basit yapıda ve yönlü bir graf biçimindedir. Her bir düğüm hücre denilen n. dereceden lineer olmayan bir devredir. Düğümler işlem elemanı olarak tanımlanır. Düğümler arasında bağlantılar vardır. Her bağlantı tek yönlü işaret iletim yolu (gecikmesiz) olarak görev yapar. Her işlem elemanı istenildiği sayıda giriş bağlantısı ve tek bir çıkış bağlantısı alabilir. Fakat bu bağlantı kopya edilebilir. Yani bu tek çıkış birçok hücreyi besleyebilir. Ağ'daki tek gecikme, çıkışları ileten bağlantı yollarındaki iletim gecikmeleridir. İşlem elemanının çıkışı istenilen matematiksel tipte olabilir. Kısmen sürekli çalışma konumunda "aktif" halde eleman bir çıkış işareti üretir. Giriş işaretleri YSA'na bilgi taşır. Sonuç ise çıkış işaretlerinden alınabilir. Şekil 4.4 'de genel bir işlem elemanı (nöron, düğüm) gösterilmiştir.



Şekil 4.4 Genel işlem elemanı yapısı

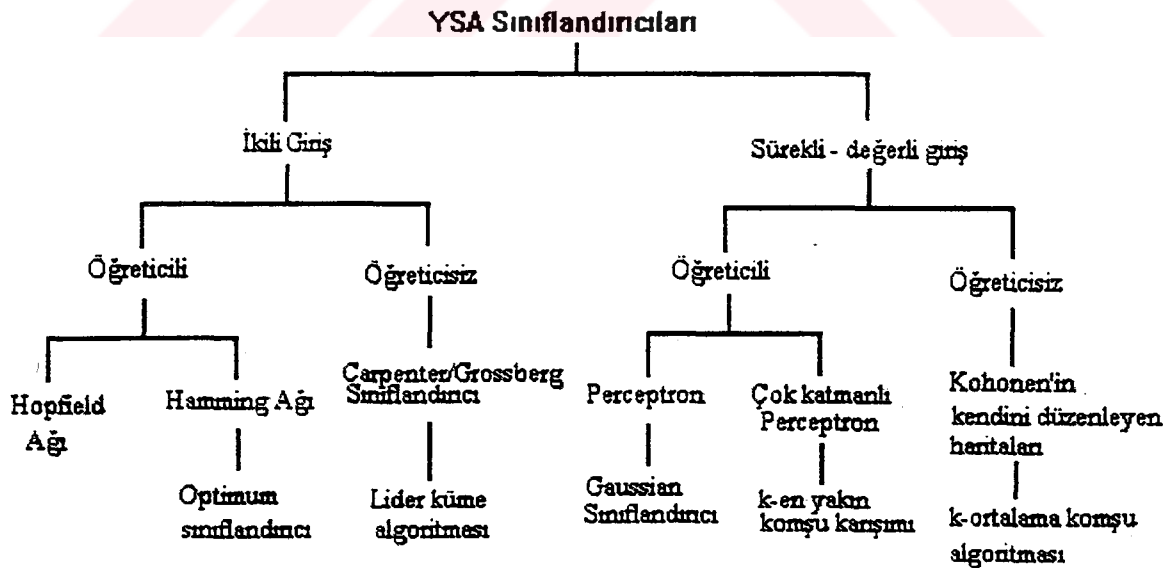


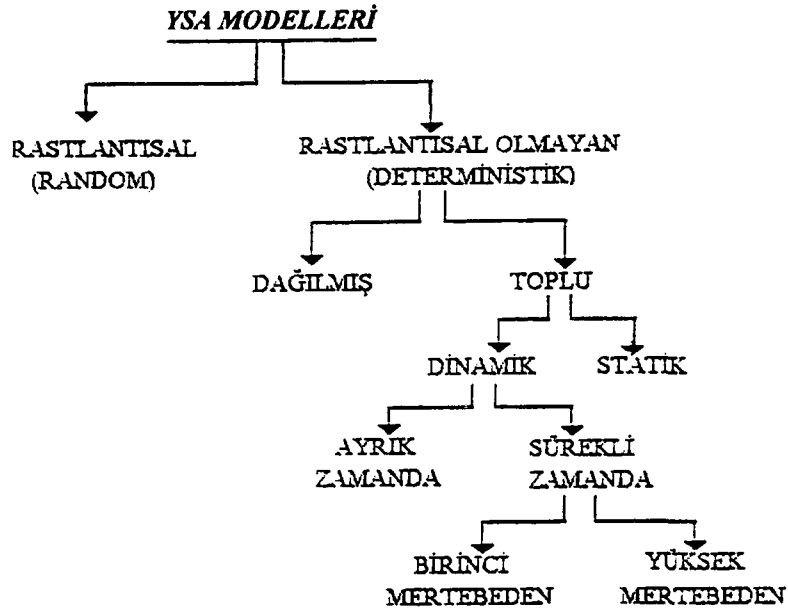


Şekil 4.6 Tetikleyici ve yasaklanan girişlere sahip bir işlem elemanı

Bir işlem elemanına gelen girişler matematiksel tiplerine göre etiketlenilerek sınıflandırılır. YSA, giriş veri tiplerine göre ikili giriş (0,1) ve sürekli değerli giriş olmak üzere aşağıdaki gibi sınıflandırılır.

Burada giriş işareti olarak seçilen  $I$  ve  $w$ 'nin sayısal değerleri bu değerlerin ölçümler boyunca okunması sırasında sürekli-değerli (reel sayı) olduğundan dolayı, sınıflandırıcı olarak öğreticili öğrenmeye sahip olan çok katmanlı perseptrona bağlı olarak ileri-besleme sinir ağı ve hatanın geriye-yayılımı algoritması kullanılmıştır.





#### 4.4.2 Bağlantı Geometrileri

Bağlantılarda taşınan işaret verisinin cinsi tanımlanmalıdır. Bağlantı geometrisi YSA için çok önemlidir. bağlantı işareti her cinsten olabilir. Bağlantının nerede başlayıp nerede bittiğini bilmesi gerekir. 1'den N'e kadar olan bir işlem elemanı kümesinin bağlantıları aşağıda tanımlandığı gibi  $N \times N$  boyutlu matris biçiminde gösterilebilir.

$$\begin{bmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \dots & w_{nn} \end{bmatrix} =$$

$w_{ij} = w_{ji} = 1 \Leftarrow i, j$  işlem elemanı  $j$  işlem elemanına bağlı

$w_{ij} = w_{ji} = 0 \Leftarrow$  bağlı değil

En fazla  $N^2$  bağlantı olur. Bağlantılar çeşitli geometrik bölgeler arasında demetler halinde düşünülebilir. Bu bağlantı demetlerinin uyması gereken kurallar şunlardır.

- 1- Bağlantı demetini oluşturan işlem elemanları aynı bölgeden çıkmalıdır.
- 2- Bağlantı demetinin işaretleri aynı matematiksel tipten olmalıdır.
- 3- Bağlantı demetinin işaretleri aynı sınıftan olmalıdır.
- 4- Bağlantı demetinin bir seçim fonksiyonu ( $\sigma$ ) olmalıdır.

$$\sigma : T \rightarrow 2^S \quad T:\text{Hedef belgesi} \quad S:\text{kaynak bölgesi}$$

Hedef bölgesindeki her işlem elemanı kaynak bölgesindeki her elemana giderse "tam" (full) bağlıdır. (örn:çok katmanlı perceptron). Eğer her hedef bölgesi elemanı N kaynak bölgesi elemanına bağlı ise "düzgün dağılmış" (uniform) olasıdır. Ayrıca her bir elemana, yine bir kaynak elemanı bağlı ise buna "bire-bir" bağlı denir.

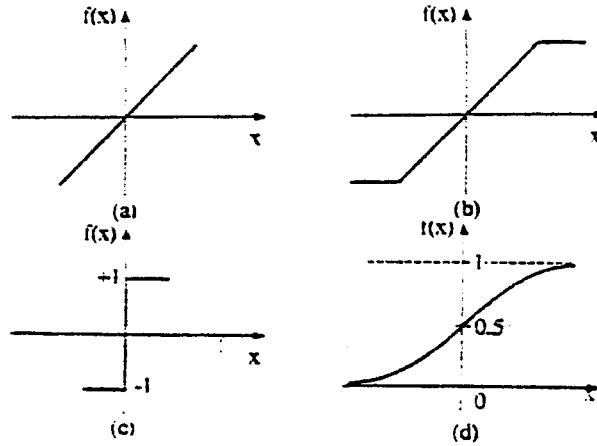
#### 4.4.3 Ağ Tipleri

Üç Çeşit Ağ Tipi Vardır

- 1- İleri beslemeli ağ: Her bir katmandaki hücreler sadece bir önceki katmanın hücrelerinden beslenir.
- 2- Kaskat bağlantılı ağ: Hücreler sadece önceki katmanlardaki hücrelerce beslenir.
- 3- Geri beslemeli ağ: En az bir hücre sonraki katmanlardaki hücrelerce beslenir. (Bu çalışmada hem ileri, hemde geri beslemeli ağ tipi birlikte uygulanmalıdır).

#### 4.4.4 Eşik Fonksiyonları

Transfer veya işaret fonksiyonları olarak da adlandırılan eşik fonksiyonları, muhtemel sonsuz domen girişli işlem elemanlarını önceden belirlenmiş sınırdaki çıkış olarak düzenler. Dört tane yaygın eşik fonksiyonu vardır. Bunlar, rampa, basamak ve sigmoid fonksiyonudur. Şekil 4.7'de bu fonksiyonlar gösterilmiştir.



Şekil 4.7 Sıkça kullanılan dört eşik fonksiyonu

Şekil 4.7 (a)'da gösterilen lineer fonksiyonun denklemi aşağıdaki gibidir.

$$f(x) = \alpha \cdot x$$

$\alpha$ : işlem elemanının  $x$  aktivitesini ayarlayan reel değerli bir sabittir. Lineer fonksiyon  $[-\tau, +\tau]$  sınırları arasında kısıtlandığında (b)'deki rampa eşik fonksiyonu olur ve denklemini:

$$f(x) = \begin{cases} \tau & : \text{eğer } x \geq \tau \text{ ise} \\ x & : \text{eğer } |x| < \tau \text{ ise} \\ -\tau & : \text{eğer } x \leq -\tau \text{ ise} \end{cases} \quad \text{şeklini alır.}$$

$+\tau$  ( $-\tau$ ) işlem elemanının maksimumu (minimumu) çoğu zaman doyma seviyesi olarak adlandırılan çıkış değeridir. Eğer eşik fonksiyonu bir giriş işaretine bağlı ise yaydığı  $+\tau$  giriş toplamı pozitif, bağlı değilse eşik basamak fonksiyonu  $[-\delta]$  olarak adlandırılır. Şekil 4.7 (c), basamak eşik fonksiyonunu gösterir ve denklemini:

$$f(x) = \begin{cases} \tau & : \text{eğer } x > 0 \text{ ise} \\ -\delta & : \text{diğer durumlar} \end{cases} \quad \text{şeklindedir.}$$

Son ve en önemli eşik fonksiyonu (bu çalışmada kullanılan) sigmoid fonksiyonudur. Şekil 4.7 (d) de gösterilen S biçimindeki sigmoid fonksiyonu; seviyeli, lineer olmayan çıkış veren, sınırlı, monoton artan fonksiyondur. Denklemi:

$$f(x) = \frac{1}{1 + e^{-x}} \quad \text{biçimindedir}$$

Her işlem elemanı kendisine verilen yerel veriye göre, kendisini ayarlayacak bütün YSA'nın enformasyon bölgesinin öğrenmesini sağlar. (Enformasyon bölgesi olasılık-yoğunluk fonksiyonu ile de tanımlanabilir). Enformasyon bölgesi birçok uygulamada, gerçek değerler "0" ile "1" arasında normalize edilmesi gerekir. (Normalize etmek: gerçek değeri 85 olan bir girişi 0.85 şeklinde ağa uygulamaktır.) Normalizasyon aynı anda bütün girişlere uygulanabilir.

#### 4.4.5. Ağırlık Uzayı

Bir çok YSA öğrenme işlemi, işlem elemanlarının ağırlığı değiştirilerek sağlanır. Böylece tanımlanan ağırlık değiştirilerek öğrenmede iyi bir model kullanıp, ağırlıkların bu modele göre değiştirilmesi esastır. Basit bir matematiksel model olarak herbir işlem elemanının "n" adet gerçek ağırlığı olduğu düşünülerek ve N adet işlem elemanı gözönüne alınırsa:

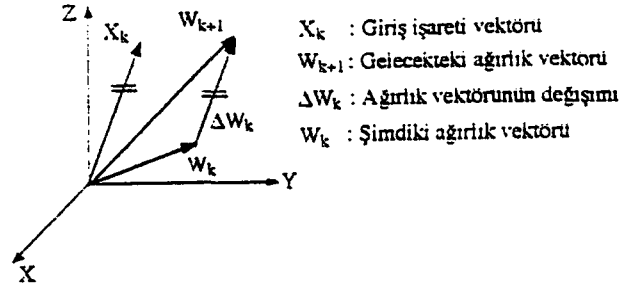
$$w = (w_{11}, w_{12}, \dots, w_{1n}, w_{21}, w_{22}, \dots, w_{2n}, \dots, w_{N1}, w_{N2}, \dots, w_{Nn})^T$$

$$w = (w_1^T, w_2^T, w_3^T, \dots, w_N^T)$$

$w_1, w_2, \dots, w_N$ : işlem elemanlarının ağırlık vektörleridir.

$$w_1 = \begin{pmatrix} w_{11} \\ w_{12} \\ \vdots \\ w_{1n} \end{pmatrix} \quad w_N = \begin{pmatrix} w_{N1} \\ w_{N2} \\ \vdots \\ w_{Nn} \end{pmatrix}$$

YSA ağırlık vektörü  $N$ ,  $n$  boyutlu orkid uzayında yayılır. YSA'nın enformasyon işleme performansı, ağırlık vektörünün belirli bir değeri ile bulunacaktır. Aşağıda şekil 4.8' de ağırlıkların düzeltiliminin vektörel çizimi verilmiştir.



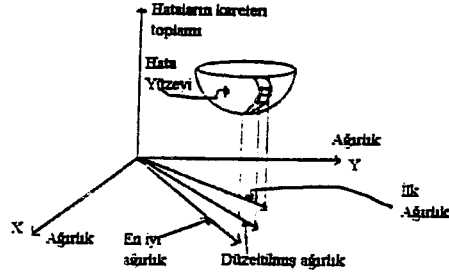
Şekil 4.8 Üç boyutlu uzayda ağırlık vektörünün değişimi

Şekilde görüldüğü gibi  $\Delta \vec{W}_k \cdot \vec{X}_k$  ile aynı doğrultuda olduğunda istenen hata düzeltilimini en küçük ağırlık değişimi ile elde etmek mümkündür. Böylece yeni bir giriş örüntüsü uygulandığında önceki eğitim örüntülerinin cevabı en az bozulmuş olur. Hata değişimini inceleyen iki çeşit kural vardır.

- 1- Hata düzeltme kuralları .
- 2- Gradyen kuralları.

**Hata düzeltme kuralları:** Her bir giriş örüntüsünde ağırlıkları yeniden ağırlayarak çıktı hatasını en aza indirmeye çalışırlar. Gradyen kurallarında ise, ağırlıklar yeniden ayarlanarak ortalama karesel hatayı (MSE) en aza indirilmeye çalışılır.

Bu noktada gradyen kuralından kısaca bahsedecek olursak, hatayı düzeltmenin (yani minimize etmenin) geometrik bir yorumunu yapmak mümkündür. Bunu yapabilmek için ağırlıkların mümkün olan tüm değerleri, hataların kareleri toplamına karşı gelecek şekilde üç boyutlu koordinat sisteminde çizilir. Bu çizim sonunda hata yüzeyi küresel bir top şeklindedir. Bu yüzeyi bir tasa da benzetmek mümkündür. Tasa'nın en alt kısmı hataların kareleri toplamının en küçük değerlerine karşı gelmektedir. Eğitim sırasında amaç ağırlıklar kümesinin en iyisini bulmak olan, en alt kısmına ulaşmaktır. Geriye-yayılım algoritması o andaki ağırlıklar yerine, yüzey hatasının eğimini hesaplayarak amacına ulaşır. Daha sonra da bu ağırlıkları tasa'nın alt kısmına doğru artımsal olarak değiştirir. İşte bu artımsal olarak tasa'nın üst kısmından alt kısmına doğru ilerleme işlemine "gradyen iniş" denir. Şekil 4.9'da gradyen inişin geometrik yorumu görülmektedir.



Şekil 4.9 Gradyen inişin geometrik yorumu

Ağırlık vektörü ile çalışan YSA'da önemli noktalardan birisi, bir öğrenme kuralı geliştirip, enformasyon bölgesi kullanarak (eşik fonksiyonu ile) ağırlık vektörü "w" iyi istenilen YSA performansı verecek noktaya yönlendirmektir. Genellikle öğrenme kuralı için bir performans ya da maliyet fonksiyonu tanımlanır. Minimizasyon veya maksimizasyon ile "w" vektörü bulunur. Bir performans çeşidi olarak bilinen, MSE (karesel ortalama hata) şu şekilde tanımlanır.

$$F(w) = \int_{\mathcal{A}} |f(x) - G(x, w)|^2 \rho(x) dv(x)$$

Amaç F'i küçültmeye çalışmaktır.

$y=G(w,x)$ : sistemin giriş çıkış fonksiyonu.

y: çıkış işaret vektörü

x: giriş işaret vektörü

w: ağırlık vektörü

$\rho(x)$ : olasılık yoğunluk fonksiyonu

#### 4.5 YSA'da Eğitim (Training)

##### 4.5.1 Eğitim Algoritmaları

Eğitim algoritmaları YSA'nın ayrılmaz bir parçasıdır. Eğitim algoritması eldeki problemin özelliğine göre öğrenme kuralını YSA'na nasıl adapte edeceğimizi belirtir. Üç çeşit eğitim algoritması yaygın olarak kullanılmaktadır.

1- Öğreticili eğitim (supervised training).

2- Skor ile eğitime (graded training).

3- Kendini düzenleme ile eğitime (self-organization training)

Öğreticili eğitimde, elimizde doğru örnekler vardır. Yani  $(X_1, X_2, \dots, X_n)$  şeklindeki giriş vektörünün,  $(y_1, y_2, \dots, y_n)$  şeklindeki çıkış vektörü, tam ve doğru olarak bilinmektedir. Herbir  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$  çifti için ağ doğru sonuçları verecek şekilde seçilen bir öğrenme kuralı ile beraber eğitilir.

Skor ile eğitimde giriş işaretlerine karşılık gelen çıkış işaretleri tam olarak bilinmemektedir. Çıkış işareti yerine skor verilir ve ağın değerlendirilmesi yapılır. Özellikle kontrol uygulamaları için idealdir. Çeşitli maliyet (cost) fonksiyonları kullanılır.

Kendini düzenleyen ağ, giriş işaretine göre kendini düzenleyerek organize eder. Olasılık yoğunluk fonksiyonlarına, sınıflandırma ve şekil tanıma problemlerine uygulanabilir.

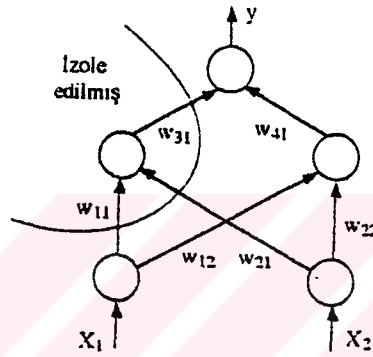
Ne tür eğitime yöntemi kullanılırsa kullanılsın, herhangi bir ağ için gerekli karakteristik özellik, ağırlıkların verilen eğitime örneğine nasıl ayarlanacağını belirttikten sonra öğrenme kuralının oluşturulmasıdır. öğrenme kuralının oluşturulması için bir örneğin, ağa defalarca tanıtılması gerekebilir. Öğrenme kuralı ile ilişkili parametreler ağın zaman içinde gelişme kaydetmesiyle değişebilir. Hangi YSA algoritmasında ne tür bir eğitime kullanıldığı bu bölümün giriş işaretlerinin sınıflandırılması kısmında gösterilmiştir.

#### 4.5.2 Bellek

YSA'nın önemli bir özelliği bilgiyi saklama şeklindedir. YSA'da bellek dağılır. Bağlantı ağırlıkları YSA bellek biçimleridir. Ağırlıkların değerleri ağın o anki bilgi durumunu temsil eder. Mesela: bir giriş/istenen çıkış çiftinin belirtilen bilgi parçası ağın içinde birçok bellek biçimine dağıtılmıştır. Bellek üniteleri ile diğer saklı bilgiler, bu bilgiyi paylaşırlar. Bazı YSA bellekleri ilişkilidir. Öyleki eğitilen ağa bir kısmı uygulanırsa, ağ bu girişe belleğindeki en yakın çıkışı bu giriş için seçer ve tam girişe bağlı çıkış ortaya çıkar. Eğer YSA oto-ilişkili ise, kısmi giriş vektörlerinin ağa verilmesi bu girişlerin tamamlanması ile sonuçlanır. YSA belleğinin yapısı; eksik, gürültülü ve tam seçilemeyen bir giriş uygulandığı zaman bile mantıklı çıkış üretmeye uygundur. Bu kurala "genelleme" adı verilir. Bir genellemenin kalitesi ve anlamı, uygulama çeşidine, ağın tipine ve karmaşıklığına dayanır. Lineer olmayan çok katmanlı ağlar (özellikle geriye yayılım ağları) gizli katmandaki özelliklerden öğrenirler ve bunları çıkışlar üretmek için birleştirirler. Gizli katmandaki bilgi, yeni giriş örüntülerine akılcı çözümler oluşturmak için kullanılabilir.

### 4.5.3 Hata Toleransı

Klasik hesaplama sistemleri çok az bir zarardan bile etkilenir. YSA için durum farklıdır. Bu farklılık YSA'nın hata toleranslı olmasıdır. İşlem elemanlarının az da olsa zarar görmesi sistemin bütününe etkiler. YSA paralel dağılmış parametrelili bir sistem olduğundan her bir işlem elemanı izole edilmiş bir ada olarak düşünülebilir. Şekil 4.10 'da çok katmanlı perseptron (MLP) için bu durum gösterilmiştir.



Şekil 4.10 MLP'nin izole edilmiş hali

Daha çok işlem elemanın zarar görmesi ile sistemin davranışı biraz daha değişir. Performans düşer ama sistem hiç bir zaman durma noktasına gelmez. YSA sistemlerinin hata toleranslı olmasının nedeni bilginin tek bir yerde saklanmayıp, sisteme dağıtılmasıdır. Bu özellik sistemin durmasının önemli bir zarara neden olacağı uygulamalarda önem kazanır.

### 4.5.4 YSA Kullanımının Sebepleri

- 1- YSA'lar verilerden hareketle, bilinmeyen ilişkileri akıllıca hemen ortaya çıkarabilmektedir. Bu özellikleri, uygulama açısından son derece önemlidir. Ayrıca veri toplama için bir ön sorgulama ya da açıklama gerekmemektedir.
- 2- YSA'lar çözüm olarak geliştirilebilir. Bir örnekten hareketle, diğer örneklerdeki benzerlikleri doğru olarak anlayabilirler. Genelleştirme yapılabilmesi bu bakımdan çok iyi bir özelliğidir, çünkü gerçek dünya verilerinde sürekli olarak gürültü ve bozucu etkiler mevcuttur.
- 3- YSA'lar non-lineerdir. Bu özellikleri nedeni ile daha karmaşık problemleri lineer tekniklerden daha doğru çözerler. Non-lineer davranışlar hissedilir, algılanır, bilinebilir, ancak bu davranışları ya da problemleri matematiksel olarak çözmek zordur.

4- YSA'lar son derece paralellige sahiptir. Bağımsız işlemleri aynı anda çok hızlı yürütebilirler. Paralel donanımlar yapıları gereği YSA'lara uygun olduğundan kendisine alternatif çözüm metodlarından daha elverişlidir.

#### 4.5.5 YSA'nın Klasik Yazılımlar İle Karşılaştırılması

YSA'lar, ısrarla belirtildiği gibi önceden tahmin, örnek değerlendirme ve gruplama işlemlerinde etkilidir. Aynı işlemleri klasik bir bilgisayar programı ile yapmak da mümkündür. YSA'lar, açıkça kuralları bulunmayan veya anında optimizasyon kısıtlamaları koyan uygulamalar için idealdir. YSA için endüstriyel kontrol işlemleri oldukça yaygın uygulama alanlarıdır. Burada kurallar çok sık değişmez ve üstelik iyi bir tarafı da öteki çalışma koşullarına ait verilerin bol oluşudur.

Klasik programlar da belirli bir görev için yazılmış bir yazılım yıllarca aynı tip işi yapar. Örneğin, bir mühendislik programı olan Autocad ve benzerleri ile sürekli aynı hizmetler yapılabilir.

YSA'ların uygulamadaki dezavantajlarını sıralayacak olursak,

- 1- Bir problemin çözümünde çok uygun bir çözüm bulamayabilirler ve çözümde hata yapabilirler. Buna sebep ise eğitecek bir fonksiyonun bulunamamasıdır. Fonksiyon bulunsa bile yeterli veri sağlanamayabilir.
- 2- Sonuç almak yüzlerce giriş örneğinin hesaplanmasına bağlı olabilir. Ayrıca hangi ağırlığın sonucu nasıl etkileyeceğini tahmin etmek zordur.
- 3- YSA'larla bir dizi işlem yapmak, bunları eğitmek yavaş ve pahalı olabilir. Maliyeti artıran sebeplerden ilki eğitime verilerinin toplanması ve değerlendirilmesidir. Doğru değerleri bulmak için deneyler yapmak gerekebilir.
- 4- Bir YSA'nın kalitesi ve kapasitesi uygulamadaki hızı ile orantılıdır. Öyleki düğüm sayısındaki az bir artış bile yürütme zamanında çok daha fazla artışa sebep olur. Örneğin 100 düğümde 10 000 bağlantı var ise, standart bir mikroişlemci bunu 10 000 000 çarpma, saklama işlemi yaparak hesaplanır. Böylece ağdan saniye de 1000 geçiş olur. Eğer 300 düğüm var ise aynı işlemci ancak 100 kere geçiş yapmayı sağlayabilir. Kısacası düğüm sayısı 3 kat arttığında cevap süresi 10 kat azalır.

Yinede YSA'ların diğer çözümlerden daha doğru çözümler ürettikleri de bir gerçektir. Çünkü bu sakıncalı durumlar teorik olarak söz konusudur.

#### 4.6 Öğrenme Kuralları

Bilginin kurallar şeklinde açıklandığı klasik uzman sistemlerin tersine, YSA gösterilen örnekten öğrenerek kendi kurallarını oluşturur. Öğrenme: giriş örneklerine veya (tercihen) bu girişlerin

çıkışlarına bağlı olarak ağırlık bağlantı ağırlıklarını değiştiren veya ayarlayan öğrenme kuralı ile gerçekleştirilir.

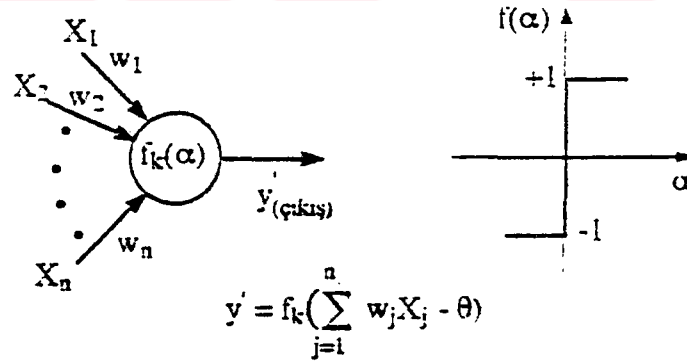
Öğreticisiz öğrenmede her giriş işareti için istenen çıkış sisteme tanıtılır ve YSA giriş/çıkış ilişkisini gerçekleştirene kadar kademe kademe kendini ayarlar. Günümüzde kullanılan birçok öğrenme kuralı vardır. Bilinen en çok kullanılan öğrenme kuralları şunlardır.

- Raslantusal (Hebb) öğrenme kuralı
- Performans (Widrow ve ADALİNE) öğrenme kuralı
- Kompetitif (Kohonen) öğrenme
- Filtreleme (Grossberg)
- Spotitemporal öğrenme
- Genelleştirilmiş Delta Kuralı Öğrenme

Burada bütün öğrenme kuralları incelenmeyecektir. Sadece burada etkinliğinden bahsedilen "Genelleştirilmiş Delta Kuralı" ,ilk bölüm de kısaca anlatılmıştı.

#### 4.6.1 Perceptron (ıdrak, almaç)

Perceptron ağı, ilk 1943 yılında Mc Culloch ve Pitts tarafından saptandı (Mc culloch et al. 1943). Onların bahsettikleri YSA tipi aşağıda şekil 4.11'da gösterildiği gibidir.

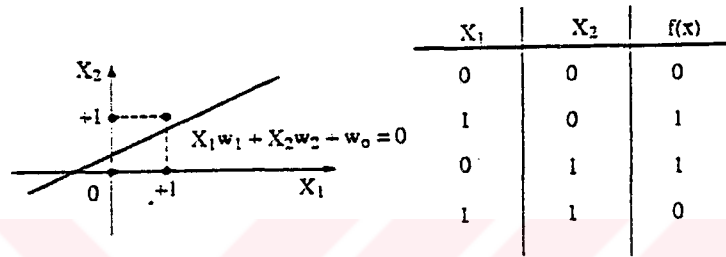


Şekil 4.11 Tek katmanlı perceptron yapısı

$\sum_{j=1}^n w_j x_j - \theta = 0$  n- boyutlu uzayda n-1 boyutlu bir düzlem belirler.

Yani bu ilk perkeptron modeline göre, giriş bilgisinin mevcut iki sınıftan hangisine eşit olabileceğini bulacak şekilde eğitilen basit bir ağıdır. Daha sonra 1960 yıllarında F.Rosenblatt yukarıdaki ağ tipini

biraz daha geliştirdi (Hebb, 1949). Ama Minshy ve Papert bu tek katmanlı perceptronun XOR (ayrıcalıklı veya) işlemini gerçekleştiremediğini ispatladılar(Minsky et al, 1969). Şekil 4.12 , şekil 4.13 ve şekil4.14'den anlaşılacağı gibi 0'ların bir tarafta 1'lerin bir tarafta ayıracak şekilde bir bölge oluşmuyor. XOR gibi 3 veya daha fazla sınıfa ihtiyaç duyulan problemleri çözmek için yapılması gereken işlem: YSA yeni katmanlar eklemektir. Eşik bağlarıyla oluşturulan karar bölgesi şeklinin karmaşıklığı sadece eklenmiş olan katmanların sayısı ile sınırlıdır. Şekil 4.13'de gösterildiği gibi içbükey ayrılabilir fonksiyonlar gerçekleştirilebilir.

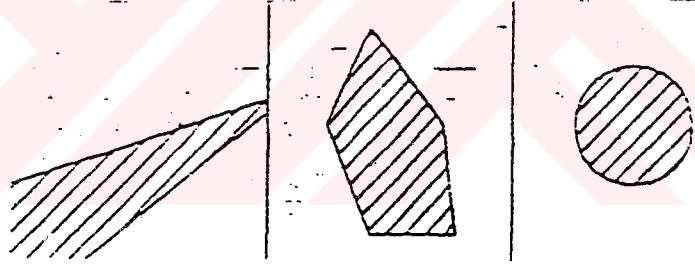


Şekil 4.12 Lineer ayrılabirliğin gösterimi

YAPI	Karar Bölgeleri tipi	XOR Problemi	Bölgelere Dayalı sınıflar	En iyi ayırdığı bölge şekilleri
Tek katman 	Doğrusu ayrıştırılmazdır. Ayrıştırmayı veya işlevini gerçekleştiremez.			
İki katman 	Konveks (iç bükey) açık veya kapalı bölgeler.			
Üç katman 	İç bükey olmayan hatırı bağlantılı olmayan bölgeler			

Şekil 4.13 Çok katmanlı perceptronda gizli katmanın rolü

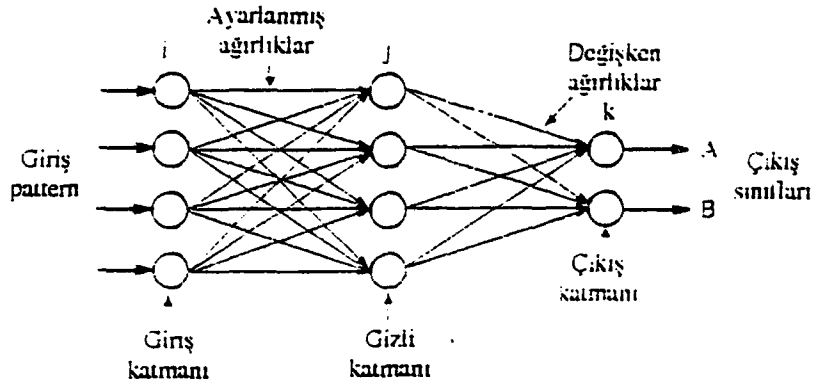
Bilgi lineer yayılamıyorsa 2.katmanın çıkışı konveks bölgededir bunun neticesi olarak 3.katmandan gelecek olan çıkış bilgisinin şekli, herhangi bir bölgenin şeklinde olabilir. Bu sebeple ihtiyaç duyulan katman sayısı üç olmaktadır. şekil 4.13'den anlaşılacağı gibi iç-bükey olmayan, hatta basit bağlantılı olmayan bölgeleri kümelemek ancak 3- katmanlı ağ ile mümkündür. 3-katmanlı perceptronun 2. katmanında ihtiyaç olan düğümlerin sayısı, bir karar bölgesinin birleştirilmemiş hali veya bir ağ gözünün bir dış-bükey alandan meydana gelemediğinin birinden büyük olması lağzımdır. 2. katmandaki düğüm sayısı en kötü durumda giriş bilgilerinin dağılımını yapan bölgenin bağlanmamış sayısına eşit olması gerekir. Birinci katmandakilerin sayısı her iki katmandaki değişim ile 3 yada 4 köşeli dışbükey bir alan oluşturmaya yeterli seviyede olmalıdır. Bunun tipik bir sonucu olarak da en az 1. katmandakinden üç kat fazla miktarda olması gerekmektedir. Bununla beraber Gutierrez ve arkadaşları değişik perceptron ağlarının ihtiyacı olan düğüm sayıları hakkında çalışma yaptılar ve çok fazla düğümünde, çok az sayıda olduğu gibi zararlı etkisi olduğunu buldular (Gutierrez et al, 1989). Tek katmanlı perceptron uygulanan her eğitimin seti modelinin en önemli özelliği, lineer biçimde dağılmak zorunda olmasıdır. Şayet bu doğruysa, Rosenblatt perceptron ağının kararlı olacağını gösterdiler (mesela ağırlıklar iterasyonla yakınsar).



Şekil 4.14 Açık ve kapalı konveks karar bölgelerinin örnekleri

#### 4.6.2 Çok Katmanlı Perceptron (Multi-Layer Perceptron)

Çok katmanlı perceptron giriş ve çıkış katmanları arasında birden fazla katmanın kullanıldığı YSA sistemleridir. Gizli katman (hidden layer) olarak isimlendirilen bu katmanlarda , düğümleri aracısız giriş olmayan ve aracısız çıkış veremeyen üniteler vardır. Şekil 4.15 'de çok katmanlı perceptronun genel yapısı verilmiştir.



Şekil 4.15 Çok katmanlı perceptron yapısı

İki katmanlı ağlarda veriler giriş katmanı tarafından kabul edilirler. Ağ içinde yapılan işlemler sonucunda çıkış katmanında oluşan sonuç değer işlenen cevap ile karşılaştırılır. Bulunan cevap ile istenen cevap arasındaki herhangi bir ayrılık varsa ağırlıklar bu farkı azaltacak şekilde yeniden düzenlenir. Girişteki değer, ağırlıklar uygun noktaya ulaşana kadar değişmez. Hesaplanan çıkışlar istenilen cevaplarla karşılaştırılarak sonuçta gerekirse hata işaret belirtilir. Hata işareti gizli birimlerden çıkış birimine olan ağırlıkları değiştirmekte kullanılır. Ama bunu yaparken giriş katmanından gizli katmana gelenin değiştirilip değiştirilemediğini düşünmek gerekir. Gizli birimlerden ne tür bir çıkış istendiği bilinmeyeceği için gizli birimlerin çıkışında hata işareti verilmesi kolay bir şey değildir. Bunun yerine her bir birimin, çıkış biriminin hatalarına olan etkisi bilinmelidir. Bu hatalı birim için gizli birime bağlı olan çıkış birimlerinin hata işaretlerinin ağırlıkları toplamı alınarak yapılır. Çok gizli katmana sahip sistemlerde her sistemin hata işaretleri, bir önceki katmanın düzeltilmiş işaretlerinden çıkartılarak işlem tekrarlanır. Sonuç olarak ağırlık düzeltme işlemi çıkış seviyesine bağlı ağırlıklardan başlar ve işlem ters yönde, giriş seviyesine varana kadar devam eder. Sonuçta sistem hatalar yapar, ama bu hatalardan birşeyler öğrenip isteneni bulana kadar işleme devam eder. Bu yöntem "hatanın geriye yayılması algoritması" (Error back-propagation algorithms) denir. Şimdi bu algoritmayı inceleyelim.

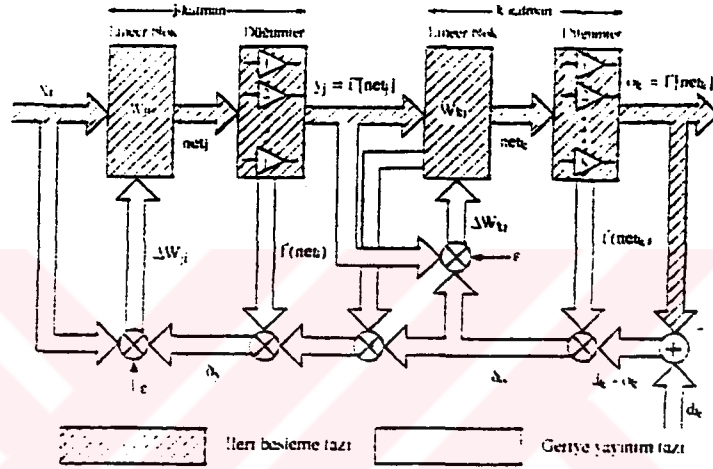
#### 4.6.3 Hatanın Geriye Yayılması Algoritması ve Genelleştirilmiş Delta Kuralı

Hatanın geriye yayılması algoritması, karesi alınmış hata fonksiyonunu minimize eden kodlu bir algoritma olup ve genelleştirilmiş delta kuralını eğitime için kullanılır. Şekil 4.16'de mimarisi gösterilen algoritma, ana hatlarıyla şöyledir:

Her bir  $j$  biriminin çıkışı  $o_j$  şu şekilde tanımlanır:

$$o_j = f(\text{net}_j) = f(x) \quad \text{ise} \quad \text{net}_j = \sum_i w_{ji} o_i + \theta_j \quad (4.1)$$

Burada  $o_j$ ; i. biriminin çıkışı  $w_{ji}$ ; i biriminden j birimine bağlantının ağırlığı,  $\theta_j$ ; j biriminin kutbu (bias)  $\{\sum_i$ ; çıkışı j birimine akan her i biriminin toplamıdır.  $f(x)$  bir monoton artan ve türevi alınabilen fonksiyondur. Pratikte bir lojistik aktivasyon fonksiyonu olarak  $f(x)=1/1 + e^{-x}$  (sigmoid) daha çok kullanılır.



Şekil 4.16 Hatanın geriye yayılması algoritmasının blok diyagramı

m-boyutlu giriş örüntüleri set edildiğinde  $\{i_p = (i_{p1}, i_{p2}, \dots, i_{pn}) : p \in P\}$ 'dir. Benzer şekilde istenilen n- boyutlu çıkış örüntüleri  $\{t_p = (t_{p1}, t_{p2}, \dots, t_{pn}) : p \in P\}$  belirtir. Burada: P:YSA uygulanan işaret şekilsel v.b. örüntüleri verir.

Bir görüntü için karesel hata (MSE) fonksiyonu  $E_p$  şu şekilde tanımlanır.

$$E_p = \frac{1}{2} \sum_{j \in \text{çıkış katmanı}} (t_{pj} - o_{pj})^2 \quad (4.2)$$

Amaç uygun  $w_{ji}$  ve  $q_j$  seçimiyle,  $E = \sum_p E_p$  toplam hatayı yeterince küçük yapmaktır. Bu amacı gerçekleştirmek için, bir  $p \in P$  örüntüsü ard arda ve rasgele biçimde seçilir. Daha sonra  $w_{ji}$  ve  $q_j$  şöyle değiştirilir.

$i_{pi}$  : Giriş işaretinin i bileşeni;

$t_{pj}$  : Çıkış vektörünün j bileşeni;

$o_{pj}$  : YSA uygulanan P örüntü setinin ürettiği çıkış ise;

$$\delta_{pj} = (t_{pj} - o_{pj}) \quad (4.3)$$

$$\Delta_p w_{ji} = -\varepsilon \left( \frac{\partial E_p}{\partial w_{ji}} \right) \quad (4.4)$$

$$\Delta_p \theta_j = -\varepsilon \left( \frac{\partial E_p}{\partial \theta_j} \right) \quad (4.5)$$

Burada  $\varepsilon$  : öğrenme oranı adı verilen küçük bir pozitif sabit sayıdır. Şayet gizli katman yok ise: (4.4) ve (4.5)'in sağ tarafı hesaplanır, o zaman:

$$\frac{\partial E_p}{\partial w_{ji}} = \frac{\partial E_p}{\partial o_{pj}} \frac{\partial o_{pj}}{\partial w_{ji}} \quad (4.6)$$

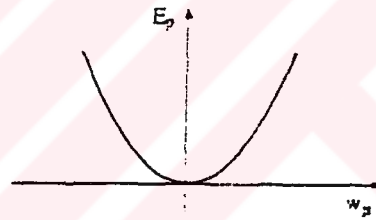
$$\frac{\partial E_p}{\partial o_{pj}} = -(t_{pj} - o_{pj}) = -\delta_{pj} \quad (4.7)$$

$$o_p = \sum_i w_{ji} \cdot i_{pi} \quad (4.8) \text{ ise: } \frac{\partial o_{pj}}{\partial w_{ji}} = i_{pi} \quad (4.9)$$

elde edilir. (4.7) ve (4.9) ifadelerini (4.6)'da yerine koyarsak:

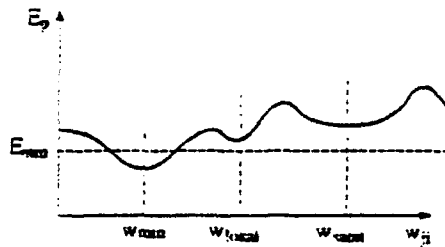
$$-\frac{\partial E_p}{\partial w_{ji}} = \delta_{pj} i_{pi} \quad (4.10)$$

olur. Hata fonksiyonu Şekil 4.17 de gösterildiği gibidir.



Şekil 4.17 Gizli katmanı olmayan ağın hata fonksiyonu

Gizli katman olduğu zaman: Hata düzeyi şekil 4.17'de olduğu gibi sadece bir minimumdan oluşmuyor. Şekil 4.18'deki gibi çeşitli minimumlar oluşur. Öğrenmede en küçük minimuma ulaşılmak istenir.



Şekil 4.18 Gizli katmana ait ağın hata fonksiyonu

Bu durumda j. düğümün lineer olmayan çıkışı:

$$o_{pj} = f_j(\text{netp}_j) \Rightarrow \text{netp}_j = \sum_i w_{ji} o_{pi} \quad (4.11)$$

şeklindedir. Bu durumda:

$$\frac{\partial E_p}{\partial w_{ji}} = \frac{\partial E_p}{\partial \text{netp}_j} \frac{\partial \text{netp}_j}{\partial w_{ji}} \Leftrightarrow \frac{\partial \text{netp}_j}{\partial w_{ji}} = \frac{\partial}{\partial w_{ji}} \sum_k w_{jk} o_{pk} = o_{pj} \quad (4.12)$$

$$\delta_{pj} = -\frac{\partial E_p}{\partial \text{netp}_j} = -\frac{\partial E_p}{\partial o_{pj}} \frac{\partial o_{pj}}{\partial \text{netp}_j} = -\frac{\partial E_p}{\partial o_{pj}} f'_j(\text{netp}_j) \quad (4.13)$$

İki durum var:

1-  $o_{pj}$  YSA'nın çıkışı ise:

(4.7) ifadesini (4.13)'de yerine koyarsak.

$$\delta_{pj} = (t_{pj} - o_{pj}) f'_j(\text{netp}_j) \quad (4.14)$$

bulunur.

2- Eğer gizli katmanların çıkış işaretinden bahsediliyorsa yani eleman çıkış elemanı değilse:

$$\sum_k \frac{\partial E_p}{\partial \text{netp}_k} \frac{\partial \text{netp}_k}{\partial p_j} \quad (4.15)$$

şeklinde ise.

$$\sum_k \frac{\partial E_p}{\partial \text{netp}_k} \frac{\partial}{\partial o_{pj}} \sum_i w_{ki} o_{pi} = -\sum_k \delta_{pk} w_{kj} \quad (4.16)$$

olur. Bulduğumuz son işlemi (4.13)'de yerine koyarsak:

$$\delta_{pj} = f'_j(\text{netp}_j) \sum_k \delta_{pk} w_{kj} \quad (4.17)$$

elde edilir. (4.16) denklemindeki (-) işareti, ağırlıkların ters yönde değiştiğini belirtir. Bütün yaptığımız işlemleri kısaca özetleyecek olursak:

1. Genelleştirilmiş  $\Delta$  (delta) kuralı:

$$\Delta_p w_{ji} = \varepsilon \delta_{pj} i_{pi}$$

2. Çıkış katmanı elemanları için:

$$\delta_{pj} = (t_{pj} - o_{pj}) f'_j(\text{netp}_j)$$

3. Gizli katman elemanları için:

$$\delta_{pj} = f'_j(\text{netp}_j) \sum_k \delta_{pk} w_{kj}$$

olur. İşlem elemanında, transfer (eşik) fonksiyonu olarak "sigmoid" fonksiyonu kullanılırsa:

$$o_{pj} = \frac{1}{1 + e^{-\left(\sum_i w_{pi} o_{pi} + \theta_j\right)}} \quad (4.18)$$

$$(\text{netp}_j) = \sum_i w_{ji} o_{pi} + \theta_j$$

ifadesinin türevi alınır ve gerekli kısaltmalar yapılırsa:

$$\frac{\partial o_{pj}}{\partial \text{netp}_j} = o_{pj} (1 - o_{pj}) \quad (4.19)$$

bulunur. Bunu (4.14) de yerine koyarsak, çıkış elemanı için:

$$\delta_{pj} = (t_{pj} - o_{pj}) o_{pj} (1 - o_{pj}) \quad (4.20)$$

elde edilir. (4.19)'u (4.17) de yerine koyarsak, gizli katman elemanı için:

$$\delta_{pj} = o_{pj} (1 - o_{pj}) \sum_k \delta_{pk} w_{kj} \quad (4.21)$$

bulunur. Yukarıda toplam içerisinde gösterilen k'nın, j çıkış birimine akan herbirim k olduğuna dikkat edilmelidir. Hesaplamayı hızlandırmak için momentum terimleri ( $\alpha$ ) eklenirse, en genel halde çıkış ve

gizli katman ifadeleri şu şekilde olur:

$$\Delta_p w_{ji}(t+1) = \varepsilon \delta_{pj} o_{pi} + \alpha \Delta_p w_{ji}(t) \quad (4.22)$$

$$\Delta_p \theta_j(t+1) = \varepsilon \delta_{pj} + \alpha \Delta_p \theta_j(t) \quad (4.23)$$

Burada; t: öğrenme saykılarının sayısını gösterir. ( $\alpha$ ) küçük pozitif bir sayıdır.

#### 4.6.4 Öğrenme ve Momentum Katsayıları

YSA ile ilgili bir başka sorunda, düzgün bir öğrenme katsayısının ( $\varepsilon$ ) ayarlanmasıdır. Ağırlıkları çok yüksek tutmak davranışın bozulmasına neden olabilir. O nedenle öğrenme katsayısını böyle bir davranışı önlemek için küçük tutmak gereklidir. Öğrenme katsayısı,  $0.01 < \varepsilon < 10$  aralığında seçilen sabit bir sayıdır. Öte yandan çok küçük bir öğrenme oranında, öğrenme işleminin yavaşlamasına yol açar.

Momentum ( $\alpha$ ) fikirsel olarak bu noktadan hareketle ortaya atılmıştır. Momentum mevcut delta ağırlığı üzerinden önceki delta ağırlığının belli bir kısmını besler. Böylece daha düşük öğrenme katsayısı ile daha hızlı öğrenme elde edilir. Momentum katsayısı genellikle  $0 < \alpha < 1$  aralığında değişen sabit bir sayıdır.

## 5. BAŞLANGIÇ HATA DETEKTÖRÜ VE YAPILAN ÇALIŞMA

### 5.1 Başlangıç Hata Detektörünün Yapısı

Burada bahsedeceğimiz, hattı kontrol hata detektörü iki kısımdan meydana gelmektedir.

- 1- Bir parazit ve gürültü filitresi - yapay sinir ağı (PGF-YSA),
- 2- İndüksiyon motorları için bir başlangıç hata detektörü- yapay sinir ağı (BHDIM-YSA).

Detektör, motorun sarımdan sarıma izolasyon hatasını ve yatak aşınmasını denetleyebilir ve aynı zamanda bu detektör motor milindeki parazitlere ve de ölçmedeki gürültülere karşı oldukça dayanıklıdır. Başlangıç hata tespit projesinin bütününde şemalandırılmış hali şekil 5.1'de gösterilmiştir. Bu şekilde.

$T_d(t)$  : motor milinin parazit momentini.

$n(t)$  : ölçme gürültüsü.

$v(t)$  : motorun giriş gerilimi.

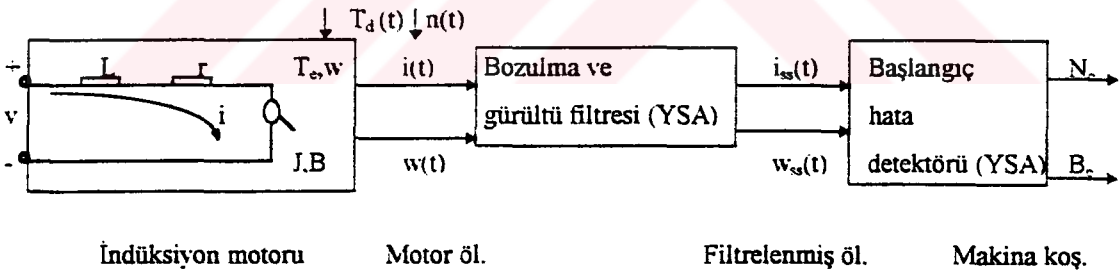
$L, r, J$  ve  $B$  motor parametreleri.

$T_e$  : motor momentini.

$i(t)$  ve  $w(t)$  motorun efektif stator akım ve ortalama rotor hızının gerçek zaman ölçümleri.

$i_{ss}(t)$  ve  $w_{ss}(t)$  .  $i(t)$  ve  $w(t)$ 'nin filitre edilmiş (sürekli-hal) versiyonları.

$N_c$  ve  $B_c$  sırası ile stator sargı ve yatak durumları.



Şekil 5.1 Yapay sinir ağları ile yürütülen indüksiyon motorları için başlangıç hata tespit planının detektörü içeren bir planı

### 5.2 Başlangıç Hata Detektörünün YSA İle Tasarım Ve Öğretme Konsiderasyonları

İlk bölümlerde yapay sinir ağlarının motor hata tespitinde kullanımı hakkında gerekli tanımlamağı vermiştik. Bu noktada ise hata tespit amaçları için bir yapay sinir ağının tasarım konsiderasyonlarını inceleyeceğiz. Motor hata tespitini gerçekleştirmek için iyi bir sinir ağı konfigürasyonu seçmekte düşünülmesi gereken birçok faktör vardır.

- 1- Ağ doğruluğu, ağ sağlamlığı ve alet uygulanabilirliği gibi pratiğe yönelik konsiderasyonlar.

2- Giriş ve çıkış değişkenlerini belirleme, öğretme bilgi setinin büyüklüğünü seçme, ağ ağırlıklarını başlangıç durumuna getirme, öğretme parametre değerlerini seçme ( öğrenme oranı ve momentum oranı gibi ) ve öğretmeği durdurma kriterini seçme gibi ağ öğretme konsiderasyonları.

3- Kullanılacak giriş ve çıkış düğümlerinin sayısını belirleme , ağdaki gizli katmanların sayısını ve her bir gizli katmandaki kullanılmış gizli düğümlerin sayısını belirleme gibi ağ tasarım konsiderasyonları.

Bu parametreleri değişkenlerin seçiminde kullanabilen bazı sıkı kurallar vardır. Bunların genel tavsiyelerinin bazısını ve sonra ağ tasarım konsiderasyonlarının tümünü optimize eden veya belirli bir yaklaşımla bunu yapan bir sinir ağını otomatik olarak gerçekleştirmek için üzerinde çalışılan konuya bir fuzzy lojik yaklaşımda yapılabilir.

Bir ağın performansını test etmek için , konsiderasyon altında sistemi temsil eden ve istatistiksel olarak önemi olan öğretme bilgi grubunu ve test etme bilgi grubunu seçmek yaygın bir uygulamadır. Test etme bilgi grubu ağ başarılı olarak öğretildikten sonra ağ performansını test etmek için kullanılıyor iken, öğretme bilgi grubu sinir ağını öğretmek için kullanılır. Giriş ve çıkış bilgisi genellikle [0.1] arasında normalize edilir. Çıkış normalizasyonu önemlidir ve gereklidir. çünkü ağ çıkışları sigmoid fonksiyon kullanımından dolayı [0.1] arasında sınırlandırılır. Giriş normalizasyonu ise ağ bilgi işleminin sayısal kararlılığını arttıracaktır.

### 5.2.1 Öğrenme Parametrelerinin Probleme Bağlı Olarak Seçimi

Ağ ağırlıkları , öğrenme oranı , momentum oranını ve ağ öğretmesini durdurmak için belirli bir kriter içeren öğretme kriterini ve öğretme parametrelerini seçmek ihtiyacı vardır. Bu değerlerin seçimi bazen ağ öğretmenin başarısına bağlıdır. Ancak, bu değişkenlerin seçimi genellikle problem bağımlı olmaktadır, yani parametre değerlerini seçmek için kullanılabilen genel formül yoktur. Yine de , aşağıda özet olarak tanımlanan bazı yol gösterici uyarılar ve tavsiyeler bir giriş denemesi olarak izlenebilir. Birkaç denemeden sonra , ağ tasarımcısı, verilen bir probleme uygun hale getiren , uygun kriteri oluşturmak için yeterince tecrübeye sahip olmalıdır.

Sinir ağının başlangıç ağırlıkları ağın öğretme işleminde önemli rol oynar. Çıkış (sonuç) ağırlıkları hakkında bir giriş bilgisi olmaksızın , [ -0.1 , 0.1 ] arasında küçük tam değerlerle bütün ağırlıkları tesadüfen başlatmak yaygın uygulamadır ve de , daha hızlı ve daha etkili ağ öğretme için geriye-yayılım (backpropagation) algoritmasını geliştirmek ve optimize etmek için bir metod üzerinde çalışıldı ve geliştirildi. En popüler metodlardan biri , yerel minimuma yakalanmaktan sakınmak için öğrenme kuralında ilave bir momentum terimi, 3.40 nolu denklemdeki  $\alpha \Delta w_{ji}$  (itera)'yı içerir.

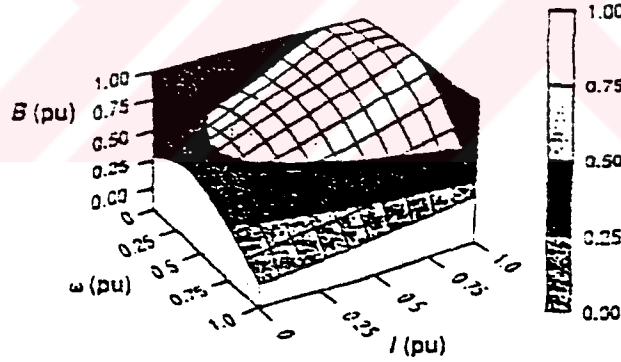
Öğrenme oranı ve oranının genelde değerleri sırası ile 0.1 ve 0.7 'dir. Ağ öğretmeyi durdurmak için kullanılan popüler kriter , yeterince küçük karasel öğretme hatası ve ağ ağırlıklarında yeterince küçük değişimlerdir. Derece olarak yeterince ne kadar küçük olduğu genellikle ağ

tasarımcısına maksimum oranda bağlıdır ve sinir ağının arzulanan doğruluk seviyesine dayandırılır. İndüksiyon motorunun hata tespit incelemesi için ya öğretim setinin (grubunun) ortalama karekök hatası ya da ağ ağırlıklarındaki değişim yeterince küçük olduğunda (0.005'den daha az ) ağ öğretmeyi durdururuz.

### 5.2.2 Ağ Konfigürasyonunu Basitleştirmek İçin Öğretim Bilgisi

Hata tespitini yapmakta bir sinir ağını başarılı olarak öğretmek için , giriş-çıkış öğretim modelleri doğru olarak seçilmelidir. Ortaya konmuş olan problemde , motor ölçümlerinin kalıcı hal akımı  $I$  ve rotor hızı  $w$  , motorun sarım izolasyonu  $N_c$  ve yatak durumu  $B_c$  'nin durumunu sezmek için kullanılabilir.  $I$  ve  $w$  değerleri kolay bulunurluğu , gerçekliği ve duyarlılığından dolayı seçilmektedir.  $I$  ve  $w$  'nın herbir kombinasyonu için  $N_c$  ve  $B_c$  değerleri ölçülmüş bilginin (datanın) bir uzmanın keşifsel yorumu ile elde edilebilir.

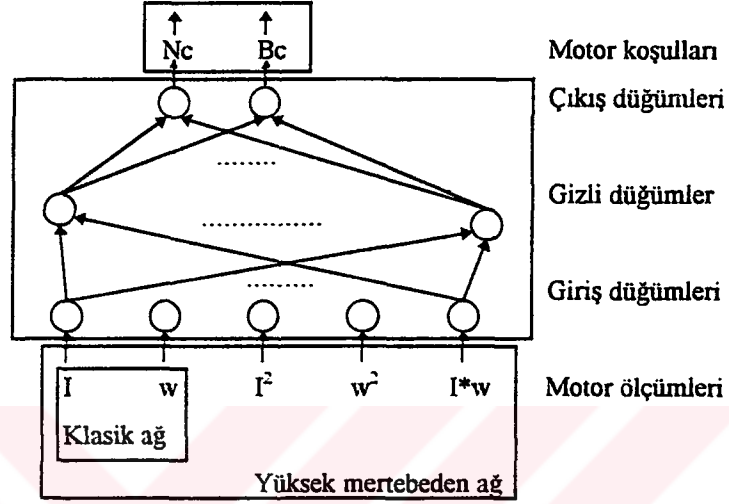
Girişler ve çıkışların ilişkisi hakkında temel bir bilgi hazır ve doğru olarak kullanılırsa ağ yapısı ve öğretim zamanı azaltılabilir ve ağ doğruluğu önemli derecede yükseltilebilir. Örneğin , girişler  $[ I, w ]$  'ya karşı  $B_c$  'nin planı  $[ I, w ]$  'nın ikinci mertebeden fonksiyonları olmak için meydana çıkan farklı yatak koşullarını ayıran sınırlamaları gösterir ( şekil 5.2 ).



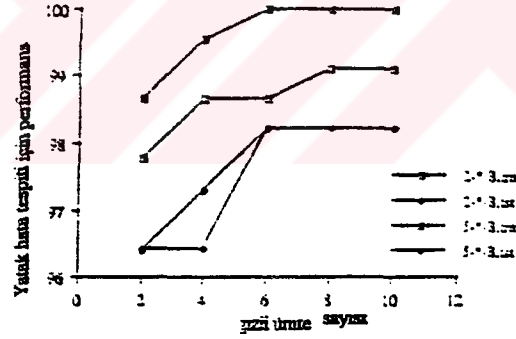
Şekil 5.2 I ve w'ya göre B'nin üç boyutlu diyagramı

Bu yüzden giriş alanı (boşluğu)  $[ I, w ]$  'dan yüksek mertebe  $[ I, w, I^2, w^2 ]$  'ya genişletildi ve sinir ağına giriş olarak kullanıldı (şekil 5.3). sonra hem kullanılan gizli düğümlerin sayısı hem de ağın öğretim zamanı azalıyor iken ağ doğruluğu artırılabilir. Giriş  $[ I, w ]$  ile klasik bir ağ ve giriş  $[ I, w, I^2, w^2 ]$  ile yüksek mertebeden bir ağ kullanımının sonuçları şekil 5.4'te gösterilmiştir. 2-\*-Btrn, 2-\*-Btst, 5-\*-Btrn, 5-\*-Btst sırası ile klasik ağ öğretim bilgi grubu, klasik ağ test etme bilgi grubu, yüksek mertebeden ağ öğretim bilgi grubu ve yüksek mertebeden test etme bilgi gruplarını temsil eder. Bu sonuçlar çok açık bir şekilde yüksek mertebeden ağın klasik ağdan daha iyi çalıştığını belirtir.  $[ I, w ]$  'dan  $[ I, w, I^2, w^2 ]$  'ya giriş boşluğunun genişliği ilave motor ölçümlerine ihtiyaç

duymaz, fakat  $[I, w]$ 'nin mevcut ölçümlerinin sadece basit bir şekilde işlenmesine gereksinim duyar. Ağdaki doğruluğu geliştirme sebebi, denklem 3.36'daki hata fonksiyonu  $E(w)$ 'nin giriş boşluğunda yüksek mertebeden bir konveks fonksiyon olmasındandır. Böylece  $w$ 'yı öğretmek (temel olarak iyi bir öğretim algoritması) yüksek mertebeden girişler kullanıldığında çok daha hızlı ve kolaydır. Ağ performansını geliştirmek için temel bir bilgi kullanmak sinir ağı incelemelerinde hem faydalı hem de popülerdir.



Şekil 5.3 Başlangıç hata tespiti için yüksek mertebeden bir sinir ağının şeması

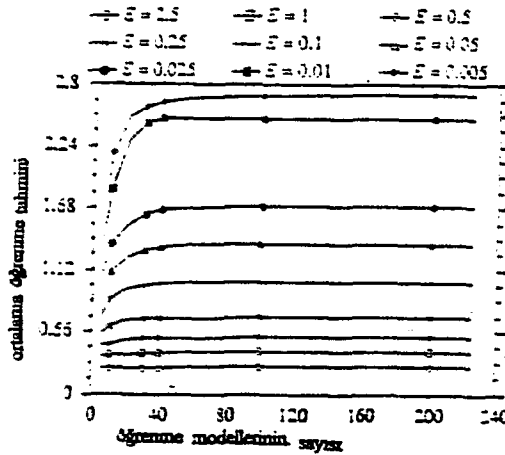


Şekil 5.4 Yatak hata tespiti için yüksek mertebeden ve klasik sinir ağlarının performans karşılaştırması

### 5.2.3 Ağ Öğretim Bilgi Grubunun Büyüklüğü : Öğretim Teorisinin İncelenmesi

Bir sinir ağını öğretmek için kullanılan öğretim örneklerinin sayısı bazen öğretim işleminin başarısını kritik derecede etkilemektedir. Öğretim örneklerinin sayısı yeterli değil ise tasarımı yaptığımız ağ sistemin gerçek giriş-çıkış ilişkisi  $M$ 'yi doğru olarak öğrenemez. Öğretim örneklerinin sayısı aşırı derecede fazla ise ağ öğretim zamanı daha uzun olacaktır. N. Tishby, E. Levin, ve S. A. Solla 1989'da yayımladıkları kitaplarında ve M. Y. Chow, G. Bilbro, ve S. O. Yee ise 1991'de yaptıkları bir

çalışmada öğretim örneklerinin sayısını tahmin etmek için öğrenme teorisinin kullanımını tanımlamaktadırlar. Yani bu özel bir inceleme için bir yapay sinir ağını öğretmede etkilidir. Bu teori verilen bir ağ konfigürasyonu için bilginin maksimum miktarını vermekte ihtiyaç duyulan öğretim örneklerinin ortalama sayısını tahmin etmek için Maksimum Entropy Prensibine dayanır (N. Tishby et al, 1989; M. Y. Chow et al, 1991). Yataklama durumu  $B_c$ 'yi tahmini öğrenmek için gereksinim duyulan öğrenme modellerinin ortalama sayısını tahmin etmek öğrenme teorisini kullanmak için sonuçlar öğretmenin sonucu için farklı ortalama karekök hata değerleri  $E$ 'ye bakışla şekil 5.5'te gösterilmiştir. Ortalamada, giriş boşluğu içinden muntazam olarak dağıtılmış 80 öğretim modeli kullanımını belirten öğrenme teorisi motor yatağı  $B_c$ 'nin durumunu doğru olarak tahmin etmek için ağı öğretmekte etkindir. Öğrenme teorisinin teoriksel sonuçları M. Y. Chow ve arkadaşlarının 1991 yılında yaptıkları çalışmada deneyler ile çeşitlendirilmiştir (M. Y. Chow et al, 1991). Böylece, gereksiz öğretim modelleri yani, 80 öğretim modelinden daha fazla, ve gereksiz öğretim zamanından sakınılabılır. Hata tespiti yapmak için ağı öğretim gibi bazı incelemeler için öğretim çevrim dışı yapılabilir, ve daha fazla öğretim bilgisi daha büyük ağ doğruluğunu başarmak için yetersiz ağ bilgisi kullanmağa tercih edilir. Biraz önce bahsettiğimiz 1991 yılında yapılan çalışmada 224 öğretim modeli, indüksiyon motoru hata tespitini gerçekleştirmek için ağı öğretmekte kullanılmıştır, ve 223 test etme modeli ağ performansını test etmek için bu çalışma içinde yer almıştır. Öğretim örneklerinin bu sayısı bu tür çalışmalarda tüm ağ konfigürasyonlarını öğretmekte fazlası ile yeterlidir (M. Y. Chow et al, 1991).



Şekil 5.5 Yatak durumunun ağ öğrenme eğrileri

#### 5.2.4 Ağ Sağlamlık Konsiderasyonları

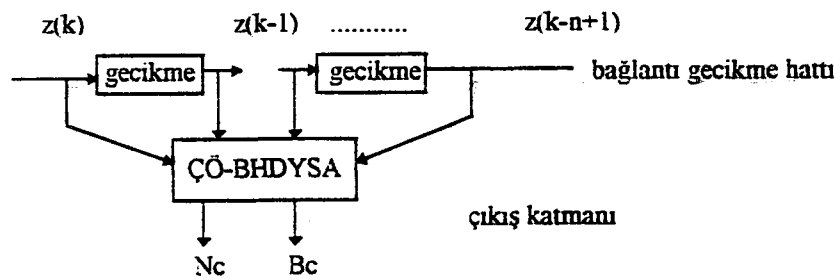
Sinir ağının hata tespit performansında % 95'in üzerinde kesin bir seviyeye ulaşırsa bunun tatmin edici bir sonuç olduğu rahatlıkla söylenilebilir. Bu durum karşılaşılan bazı özel hata tespit incelemeleri içinde geçerlidir. Çoğu hata tespit planları gerçek zaman incelemeleri için olduğundan gürültü konsiderasyonları ve küçük parazitler önemli problemler olur. Gürültü hata detektörlerinin bir baştan bir başa kapsayıcı olarak ağın performansını arttırmak için olduğu bilinir. Bu yüzden , gürültüleri bastırmak için metodlara , gürültünün mevcut olduğu durumlarda sinir ağlarını idare eden hata detektörünün doğruluğunu yükseltmek için ihtiyaç duyulur. Böyle yapmak için bir metod, sinir ağına girişlerin sayısını arttırmak için kol-gecikme hatları kullanmaktır. Şekil 5.6'da gösterildiği gibi, k örnek olarak deneme zaman indeksi ve n sinir ağı hata detektörüne girişler olarak kullanılan z'nin örneklenmiş ölçümlerinin sayısıdır. Klasik ağ için,

$$z=[L,w] \quad (5.1)$$

yüksek mertebeden ağ için,

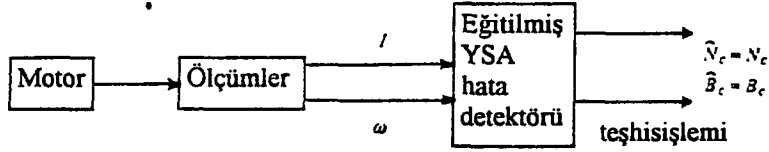
$$z=[L,w,I^*w,I^2,w^2] \quad (5.2)$$

Başlangıç hatası kullanılmış örnekleme zamanı ölçümü ile karşılaştırıldığında yavaş bir zamanla değişen olay olduğundan , gecikmiş girişlerin tümü yaklaşık olarak aynı değerde olmalıdır. Bu metod gürültüye karşı sinir ağının dayanıklılığını arttırmak için yardım eder, böylece hata denetemesinin yapılışında sinir ağı için fazlalık bilgi üretir ve bu ağın güvenilirliğini artırır (J. C. Hung et al. 1973). Kol-gecikme hatları ile ortaya çıkan yeni sinir ağı yapısına ÇÖ-BHDYSA (Çok örnekli Başlangıç Hata Detektörü Yapay Sinir Ağı) denilmektedir.



Şekil 5.6 Çok örnekli başlangıç hata detektörünün genel yapısı

YSA ile motor hata tespitinin genel şeması ise şekil 5.7'de gösterilmektedir. Ağa giriş olarak daha fazla bilgi verildiğinde ağın doğruluğunun artan yönde geliştiği bilinmektedir. Bu gelişme büyük gürültü değişimlerinin mevcudiyetinde bile vardır.



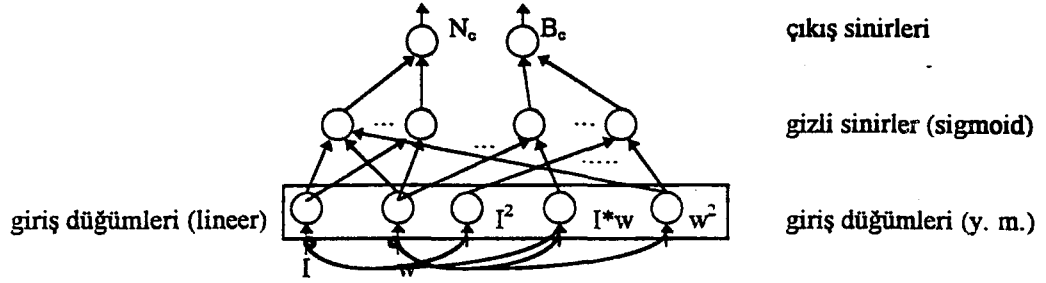
Şekil 5.7 YSA motor hata tespiti / teşhis işlemi.

### 5.3 İndüksiyon Motorlarında Yüksek Mertebeden Başlangıç Hata Detektörünün YSA İle Tasarımı

Bundan önce bu konu üzerinde yapılan çalışmalar dikkate alınarak başlangıç hata detektörünün tasarımına ve yapısına kısa bir bakış yapılacağını hatırlatmakta yarar vardır. Şimdi bu açıklamayı yaptıktan sonra konunun içeriğine geçebiliriz.

Şekil 2.2 ve şekil 5.1'de gösterildiği gibi boşluk  $I$  ve  $w$  ölçüsü üzerinden  $N$  ve  $B$ 'nin çevre çizgisi planları, sarımdan sarıma izolasyon hatası ve yatak aşınması durumlarının oluşturduğu sınıfların herbirine üç bölgeye ayrılmayı sağlar. Bu bölge sınırlarının her biri  $I$  ve  $w$ 'nın cebirsel fonksiyonları ile yaklaştırılabilir. Bir sonuç olarak, giriş boşluğunu iki boyuttan  $[I, w]$ , beş boyuta  $[I, w, I \cdot w, I^2, w^2]$ , nispeten basit bir yüksek mertebeden sinir ağı yapısı doyurucu başlangıç hata tespit performansı vermek için tasarlanabilir.

Tek-fazlı bir indüksiyon motorunun stator sargı izolasyonu ve yatak aşınması birbirinden bağımsızdır. Bu yüzden, bunların kontrolü ikiye katlanabilir. Sargı izolasyonu kontrol eden nöronlar (sinirler) yatak aşınmasını kontrol eden nöronlardan ayrılabilirler. Yani, bütün gizli nöronlar tam olarak gizli katmandaki nöronlardan elde edilen bilgiden  $N$  ve  $B$ 'nin durumlarını çözen son çıkış nöronlarına bağlanamaz. Burada sigmoid fonksiyon bütün nöronlar için (giriş katmanındakiler hariç) nonlineer hareket fonksiyonu olarak kullanılır ve ağa en çok kullanılan geriye-yayılım (backpropagation) algoritması öğretilir. Tasarmlanan ağ yapısı şekil 5.8'de görüldüğü gibidir.



Şekil 5.8 Başlangıç hata detektörünün konfigürasyonu

Farklı yatak aşınması ve stator sargı izolasyon hataları için endüktans kaçağı, doyum etkilerini içeren bir indüksiyon motorunun dinamiklerinin detaylı bir sayısal simülasyonundan elde edilen öğretim örneklerinin bir grubu sinir ağını öğretmek için kullanılır. Motorun sabit bir yükte çalıştığı varsayılır. Öğretim bilgisi grubu konsiderasyon altında tüm hata oranını kapsar. Sinir ağına girişler motorun kalıcı hal stator akımı  $I$  ve rotor hızı  $w$ 'nın sayısal değerleridir. Çıktılar seçilen maliyet fonksiyonlarına karşılık motorun sargı ve yatak durumlarına bağlıdır.

Burada basit olarak başlangıç hata tespit projesini gerçekleştirmek için yapay sinir ağlarının esnekliğini göstermek istediğimizden gerçek bir motor yerine tek-fazlı sincap kafesli bir indüksiyon motorunun bilgisayar simülasyonunun kullanıldığını bir kez daha vurgulamakta fayda vardır. Sinir ağına girişlerin bir bilgisayar simülasyonundan veya gerçek motor ölçümlerinden gelip gelmemesinin giriş bilgisinin kaynağı açısından önemi yoktur. Motor dinamiklerini analiz etmekteki tek sebebimiz konsiderasyon altında hataları denetlemekte izlenilmek için uygun durum değişkenleri bulmaktır.

Yukarıda biraz önce tanımlanan BHDİM- YSA (Başlangıç Hata Detektörü İndüksiyon Motoru -Yapay Sinir Ağı ) kalıcı hal bilgisi kullanılarak öğretilmiştir. Ancak, hata tespit projesi başvurularında rotor milinde parazitlerin meydana gelebildiği yer olan hat içinde kullanım için tasarlanır. Sinir ağına giriş bilgisi bir parazit boyunca veya sonra motorun geçici halinde toplanırsa bu uygunsuz ölçümler yanlış alarmlara sebep olan , kontrol edilme durumunda gerçekte var olmayan hatalara sebep olabilir. Bunlardan başka , motor ölçüleri hat üzerinden alındığından ve BHDİM-YSA çoğu gerçek durumlarda bir A/D dönüştürücü tarafından beslenildiğinden bunlar muhtemelen ayrık zamanlı gürültü içeren ölçme gürültüsü ile kirlenirler. Bu tür motor ölçmelerini filtre etmek için bu bilgiler BHDİM-YSA 'ya girilmeden önce çoğunlukla Parazit ve Gürültü Filtresi ( PGF-YSA) denilen bir filtreden geçirilirler.

## 5.4 Yapılan Çalışma

### 5.4.1 Ölçme Sonuçlarının Alınması

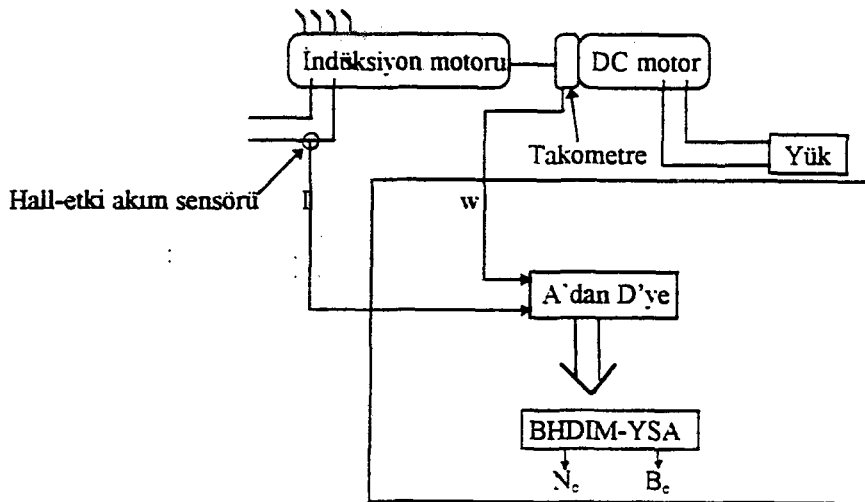
Başlangıç hata detektörü temeline dayanan sinir ağının performansı Türk Elektrik Motorları A.Ş.'nin laboratuvarından alınan değerler kullanılarak test edilmiştir. Ağı öğretmekte kullanılan

değerler ve sistemsel hareket noktasında ise daha önceden gerçekleştirilen bir çalışmadan yararlanılmıştır (M. Y. Chow et al. 1991). Test işleminin şematik olarak gösterimi şekil 5.9'da yer almaktadır. Test için yapılan ölçümlerde birkaç tip split (tek)-fazlı sincap kafesli indüksiyon motorları mekaniksel olarak daimi bir magneto dc motora akuple edilmiştir. Şematik diyagramdan da görüldüğü gibi, bir takometre yardımı ile devir sayıları okunmakta, bunlar daha sonra  $w$  ifadesinde kullanılarak, YSA'nın girişindeki değişkenlerden birinin sayısal değerleri elde edilmektedir. Benzer şekilde bir akım sensörünün yardımı ile  $I$  akım değerleri okunmaktadır. Bu da bilgisayara girilecek YSA değerlerini oluşturmaktadır.

Yapılan çalışma da ölçüm değerleri alınan tek-fazlı motorlarda istenilen ölçümleri sağlamak üzere gerekli bazı değişiklikler yapılmıştır. Giriş bağlantılarının bir veya birkaç çiftini kısa devre ederek stator sargı hatalarını izlemek mümkündür. Motor normal çalışmada iken sargı ilişkilerini belirleme metodu giriş bağlantı noktaları arasındaki gerilim düşümü ölçümünü içerir. Burada dikkat edilmesi gereken nokta, çoklu-sarım kısa devreleri, akım atlama ihtimaline karşı - akım ölçümlerinin yüksek seviyesinden dolayı- kullanılmalıdır. Yani, stator sargısının sadece tek sarımı için kısa devreler uygulanmalıdır. Bu yüksek akım seviyelerinin stator sargısına kolaylıkla hasar vermemesi içindir.

Deneyde dc motorun kullanılmasında iki temel amaç vardır. Bunların birincisi, dc motoru generatör gibi çalıştırarak ac motora sabit bir yük akımı, yani, sabit bir yük momenti vermektir. İkinci olarak ise, normal çalışma noktasında sabit yük akımını artırarak, ac motorda aşınmış bir yatak tarafından sebep olunan ilave yüke karşılık gelen, değişken bir yük momenti sağlamaktır.

İndüksiyon motorunun rotor hızının ise, bir takometre yardımı ile ölçüldüğünü söylemiştik. Bu sayısal değerler bilgisayarın genel amaçlı board kısmına bir frekans girişi ile verilir. İndüksiyon motoru stator sargı akımı bir Hall-etki aleti ile ölçülür. İndüksiyon motoruna asıl güç kaynağının bir teli bu Hall-etki aleti içinden bağlanır. Bu aletin çıkışı da bilgisayarın board kısmındaki analog kısma verilir. Böylece hata tespitinde kullanılacak değerler ölçülmüş olur.



Şekil 5.9 Başlangıç hata tespiti için motorun deney setinde yerleşiminin şematik gösterimi

#### 5.4.2 Deneyler İle Alınan Ölçüm Sonuçları

Aşağıdaki tablolarda YSA eğitildikten sonra test işleminde kullanılmak üzere deneysel olarak Türk Elektrik Motorları A.Ş. laboratuvarından alınan I ve w (n'e bağlı olarak hesaplanan) değerleri görülmektedir.

**Tablo 5.1**

Motor tipi: 6K624B	2p=2
I (A)	w (d/d)
2.9	179.49
3.2	461.54
4.7	254.61
6.2	820.51
7.6	461.54

**Tablo 5.2**

Motor tipi: 5KCP87A150	2p=2
I (A)	w (d/d)
4.16	310.71
4.77	307.77
5.76	304.83
7.01	300.54
8.38	297.4

**Tablo 5.3**

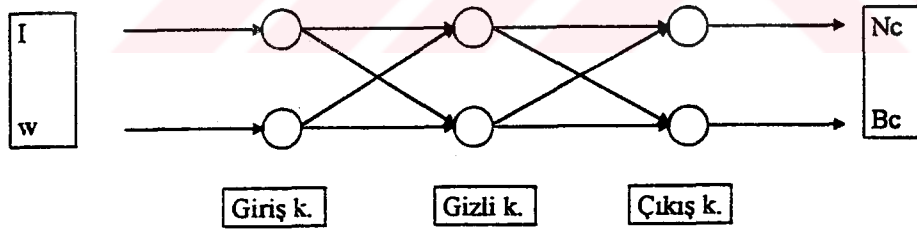
Motor Tipi: 5KCP87A150	2p=2
I (A)	w (d/d)
4.17	310.59
4.71	307.66
5.59	304.62
6.85	301.48
8.29	295.93

Tablo 5.4

Motor tipi: 5KC97C100	$2p=4$
I (A)	w (d/d)
5.66	309.76
6.00	307.45
6.68	303.26
7.46	298.66
8.48	292.79

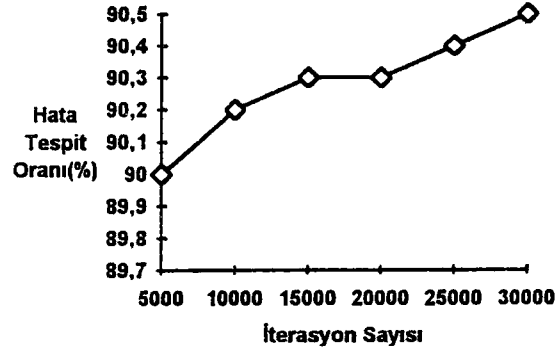
### 5.4.3 Simulasyon Sonuçları ve Akış Diyagramı

İlk bölümler debu konu ile ilgili gerekli formülasyonu tanımlamıştık. Fakat, daha önceden de belirttiğimiz gibi, YSA'nın matematiksel formülasyondan bağımsızlığı ve kısa bir zaman sürecinde bizi sonuca götürmesi YSA için bir üstünlük teşkil etmektedir. Şekil 5.10'da I ve w çok katmanlı YSA için giriş değerlerini oluşturmaktadır. Ağ, öğrenme işleyişi boyunca girişler ve arzu edilen (ulaşılmak istenen) çıkışlar bilinecek şekilde öğreticili eğitime ile öğretilir. Öğrenme bilgisi, bütün giriş bilgileri arzu edilen çıkışları üretene kadar ağda var olur. YSA'larda öğrenmenin, bağlantı ağırlıklarını ve işlem elemanı eşik değerlerini değiştirerek olduğunu daha önce kısaca belirtmiştik. Bu sebep ve sonuc ilişkisi dahilinde algoritma olarak hatanın geriye yayılması algoritması kullanılmaktadır (error back propagation).

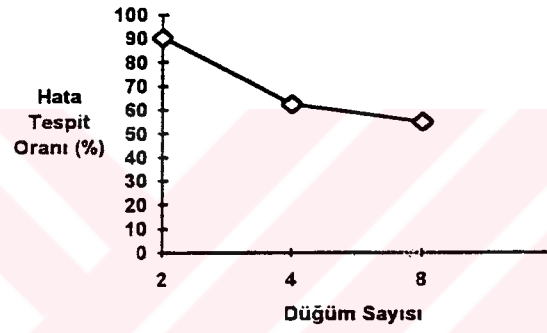


Şekil 5.10 İlk olarak eğitilen 2:2:2 mimarisindeki çok katmanlı ağ

Kullanılan ağ mimarisi ilk olarak 2:2:2 düzenindedir yani 2 giriş ünitesine karşılık olarak 2 gizli ünite ve 2 çıkış ünitesi yer almaktadır. Bu mimari için en iyi yakınsamanın  $\epsilon = 0.9$  ve  $\alpha = 0.7$  de gerçekleştiği görülmüştür. Şekil 5.11'de 2:2:2 mimarisi için iterasyona bağlı hata tesbitindeki başarı oranı görülmektedir. 30000 iterasyon sonucunda, %90.5'lik bir başarı ile hata tespitini tanıma gerçekleşmiştir. Şekil 5.12'de ise gizli katmandaki düğüm sayısının değişimine göre hata tespit oranı gösterilmiştir.

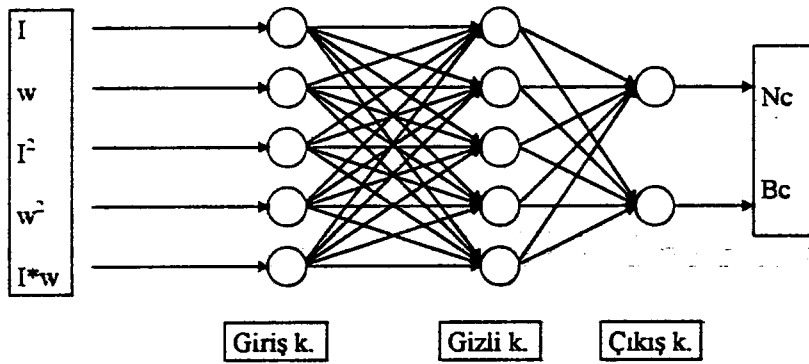


Şekil 5.11 2:2:2 mimarisi için iterasyona bağlı hata tespitindeki başarı oranı



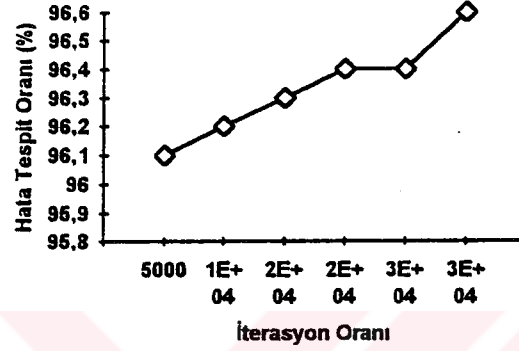
Şekil 5.12 Düğüm sayısına bağlı hata tespit oranı

Şekil 5.13 de ise 5:5:2 mimarisine göre (yüksek mertebeden ) hata tespitinin YSA ile gerçekleştirilmesinin modeli yer almaktadır.



Şekil 5.13 5:5:2 mimarisine göre çok katmanlı ağ

Bu ağ mimarisi 5:5:2 düzenindedir. 5 giriş ünitesine karşılık ,5 gizli ünite ve 2 tane de çıkış ünitesi yer almaktadır. Şekil 5.14 ve şekil 5.15 de ise hata tespit oranlarının iterasyon sayısına ve düğüm sayısına bağlı olarak değerleri gözükmektedir.



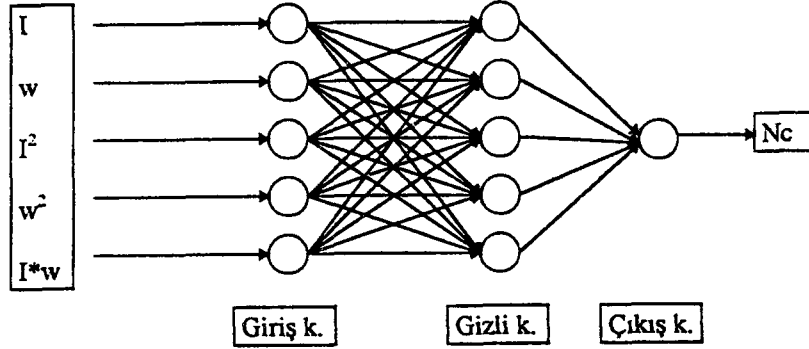
Şekil 5.14 5:5:2 mimarisi için hata tespitindeki başarı oranı .



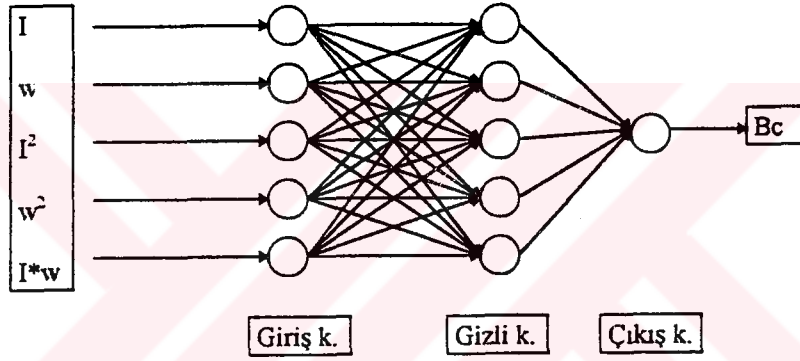
Şekil 5.15 5:5:2 mimarisi için düğüm sayısına bağlı hata tespiti

5:5:2 ağ mimarisi için  $\varepsilon = 0.7$  ve  $\alpha = 0.9$  olmak üzere % 96.6 lük bir başarı ile hatayı tanıma sağlanmıştır.

Yine 5:5:1 mimarisine göre hem  $N_c$  hemde  $B_c$  'nin tespiti için bir çalışma yapılmıştır. Bunların ağ mimarileri Şekil 5.16 ve şekil 5.17 de gösterilmiştir.

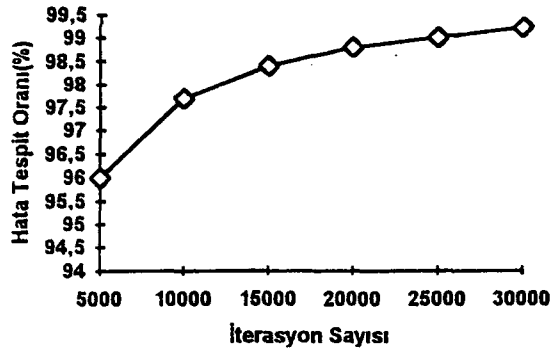


Şekil 5.16 5:5:1 mimarisine göre çok katmanlı ağ ( $N_c$  için)



Şekil 5.17 5:5:1 mimarisine göre çok katmanlı ağ ( $B_c$  için)

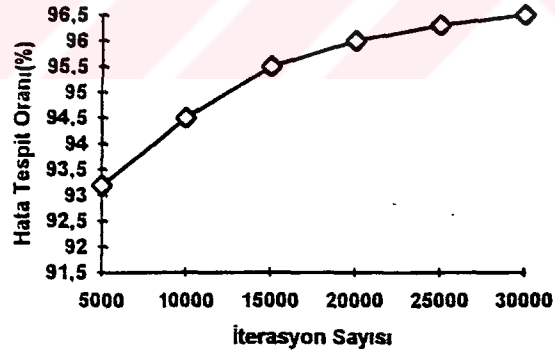
$\varepsilon = 0.7$  ve  $\alpha = 0.9$  için 30000 iterasyon sonucunda  $N_c$  'de % 99.2 ve  $B_c$  'de yine aynı iterasyon sayısı ve aynı  $\varepsilon$  ve  $\alpha$  değerleri için % 96.5'lik bir başarı ile hata tespitini tanıma sağlanmıştır . Şekil 5.18 ve şekil 5.19 iterasyon ve düğüm sayısına göre  $N_c$  için, şekil 5.20 ve şekil 5.21 ise yine aynı durumlara bağlı olarak  $B_c$  için olan değişim şemalarıdır. Şekil 5.22 ise karşılaştırma açısından yararlı olmak amacıyla bu 4 durumun iterasyon sayısına göre değişimini toplu olarak göstermektedir.



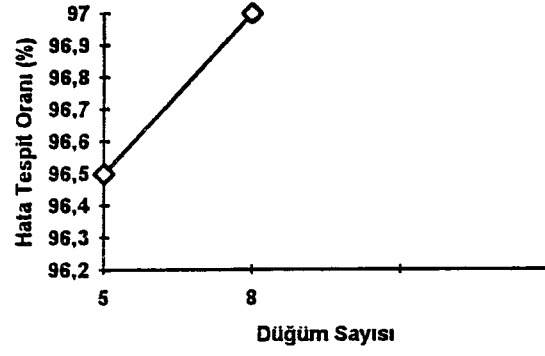
Şekil 5.18 5:5:1 mimarisine göre (Nc) hata tespitindeki başarı oranı



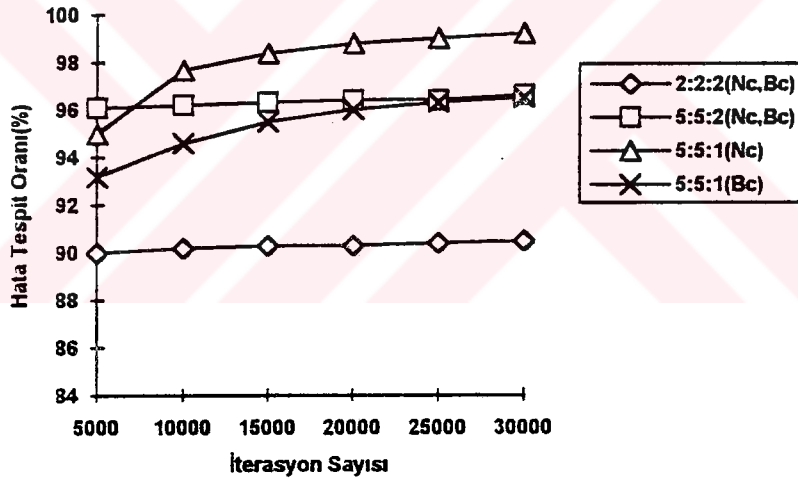
Şekil 5.19 5:5:1 mimarisinde düğüm sayısına göre(Nc) hata tespit oranı



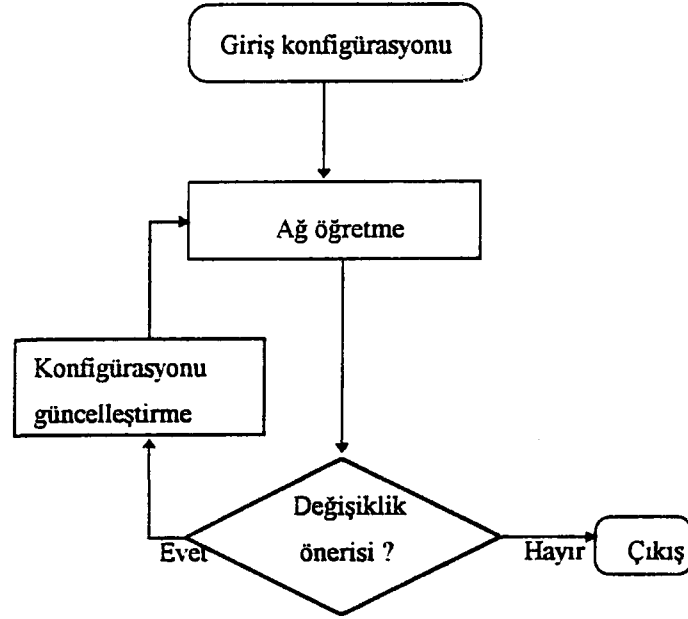
Şekil 5.20 5:5:1 mimarisine göre(Bc) hata tespitindeki başarı oranı 0



Şekil 5.21 5:5:1 mimarisinde düğüm sayısına göre(Bc) hata tespit oranı



Şekil 5.22 4 durumun iterasyon sayısına göre değişiminin karşılaştırılması



Şekil 5.23 Akış diyagramı

#### 5.4.4 Sonuçların Değerlendirilmesi

YSA'nın bu konuya uygulanmasında da geliştirilmiş delta kuralından yararlanılmıştır. Tez'de ağı öğretmekte kullanılan yazılım programı Bekir Karlık tarafından yapılan Doktora Tez'inde kullanılan programdır.(Karlık Bekir, 1994) yazım dilinde herhangi bir değişiklik yapılmamış olup, Turbo PASCAL'dır. Fakat programın C programlama dili ile komutsal açıdan daha basit olacağı noktasından hareketle, üzerinde böyle bir çalışma düşünülmektedir. Alınan sonuçların daha iyi görülmesi için farklı mimarilerin iterasyon sayısına göre hata tespit oranları tablo 5.4'te sunulmuştur.

Tablo 5.5

İterasyon sayısı	2:2:2	5:5:2	5:5:1	5:5:1
	Nc.Bc	Nc.Bc	Nc	Bc
5000	% 90.0	% 96.1	% 96.0	% 93.2
10000	% 90.2	% 96.2	% 97.7	% 94.6
15000	% 90.3	% 96.3	% 98.4	% 95.5
20000	% 90.3	% 96.4	% 98.8	% 96.0
25000	% 90.4	% 96.4	% 99.0	% 96.3
30000	% 90.5	% 96.6	% 99.2	% 96.5
	$\varepsilon = 0.9$	$\varepsilon = 0.7$	$\varepsilon = 0.7$	$\varepsilon = 0.7$
	$\alpha = 0.7$	$\alpha = 0.9$	$\alpha = 0.9$	$\alpha = 0.9$

### *5.5. Yapılan Çalışmanın Mevcut Çalışmalarla Karşılaştırılması*

Çok katmanlı YSA kullanılarak yapılan hata tespiti çalışmasında yüksek mertebeden çok katmanlı ağ için çıkış düğüm sayısının durumuna göre oldukça iyi bir oranda, hata tespitini tanıma sağlanmıştır. Ayrıca, YSA sınıflayıcıları uygulamada normal olarak gereken öğrenmenin miktarını azaltmak için de oldukça elverişlidir.

Mo-yuen Chow'un yapmış olduğu çalışmalardaki 2 giriş değişkeni (klasik) için elde ettiği başarı değerine (Nc için % 98.7, Bc için % 94.7) ulaşamamasına rağmen bu iki değişken için % 90.5 ile hata tespiti tanıma gerçekleştirilmiştir. Fakat, yüksek mertebeden ( 5 giriş için ) hata tespitinde Nc için hata tanımada % 99.2'ye Bc için ise % 96.5'a ulaşılmıştır. Mo-yuen Chow'un yapmış olduğu çalışmada ise yüksek mertebeden ağ için Nc'de % 98.7 Bc'de ise % 97.3 ile hata tanıma gerçekleştirilmiştir.

Yapılan çalışma bu anlamda kapsam olarak motor hata tespiti konusundaki sayılı çalışmalardan birini temsil etmektedir.

Bu sonuçlardan da anlaşılacağı üzere hızlı öğrenme algoritmalarının hedef değer yaklaşımına oldukça iyi bir etkisi vardır.

İnsan beynindeki sinir hücrelerinden hareketle tasarlanan YSA motor hata tespiti konusunda da rahatlıkla uygulanabilmektedir. Bu da sinir ağlarının matematiksel ve zaman açısından birçok zorluğa sahip problemleri çözmede kullanılabileceğini ve üstünlüğünü göstermektedir.

**KAYNAKLAR**

- 1- Alger, P. L. , " The nature of polyphase induction machines. " Wiley, NewYork, 1951
- 2- " Byte Türkiye. " Şubat 1994
- 3- Charles. S. . " Electrical control systems in industry. " Substitutes NewYork Mc-Graw Hill, 1963
- 4- Chow, M. Y. and Yee, S. O. . " Real time application of artificial neural networks for incipient fault detection of induction machines. " in Proc. 3rd Int. Conf. Ind. Eng. Appl. Artificial Intell Expert Sys. , S. C. . July 1990
- 5- Chow, M. Y. and Yee, S. O. . " Application of neural networks to incipient fault detection in induction motors. " J. Neural Network Computing, vol. 2, no. 3, 1990
- 6- Chow, M. Y. and Yee, S. O. . , " Robustness test of an incipient fault detector artificial neural network. " in Proc. IJCNN'91, Seattle WA, July 1991
- 7- Chow, M. Y. and Yee, S. O. . " Methodology for on-line incipient fault detection in single-phase squirrel-cage induction motors using artificial neural networks. " IEEE Trans. Ener. Convers. vol. 6, no. 3, Septem. 1991
- 8- Chow, M. Y. . Mangum, P. M. and Yee, S. O. . " A neural network approach to real time condition monitoring of induction motors. " IEEE Trans. Ind. Elektron , Decem. 1991
- 9- Chow, M. Y. . Bilbro, G. and Yee, S. O. . " Application of learning theory to a single-phase induction motor incipient fault detection artificial neural network. " Int. J. Neural Syst. , vol. 2, no. 1 and 2, 1991
- 10- Chow, M. Y. . Sharpe, R. N. and Hung, J. C. . " On the application and design of artificial neural networks for motor fault detection-Part I. " IEEE Trans. Ind. Elektron. Apr. 1993
- 11- Chow, M. Y. . Sharpe, R. N. and Hung, J. C. . " On the application and design of artificial neural networks for motor fault detection-Part II. " IEEE Trans. Ind. Elektron. Apr. 1993
- 12- Culloch, S. M. and Pitts, W. . " A logical calculus of the ideas immanent in nervous activity. " Bulletin of Mathematical Biophysics, vol. 5, pp. 115-133, 1943
- 13- " Elektrik makinaları-asen kron makinalar. " Ders Notları, Y.T.Ü. , 1991
- 14- Fitzgerald, A. E. . Kingsley, C. Jr. and Umans, S. D. . " Elektric machinery. " Substitutes NewYork Mc-Graw Hill, 1985; 2nd printing 1988, chapter :9,10 and 11.
- 15- Frank, P. M. . " Fault diagnosis in dynamic systems using analytical and knowledge based redundancy - a survey and some new results. " Automanca, vol. 26, no. 3, pp. 459-474, 1990.
- 16- Grossbery, S. . " Adaptive pattern classification and universal recoding: Part I. parallel development and coding of neural feature detectors. " Biological Cybernetics, 23, 121-134, 1976.
- 17- Hebb, O. . " The organization of behavior. " Introduction and Chapter 4, pp. XI-XIX, and 60-78, Wiley, New York, 1949.
- 18- "IEEE Standart Test Procedure for Polyphase Induction Motors and Generators." IEEE Standart 112, 1984.

- 19- Karayiannis N. B. and Venetsanopoulos A. N. , " Fast learning algorithms for neural networks. " IEEE Trans. Cir. and Sys. -II Analog and Digital Sys. Proc. vol. 39, no. 7, July 1992.
- 20- Karlık, B. , " Çok fonksiyonlu protezler için YSA kullanarak miyoelektrik kontrol. " Doktora Tezi, Y.T.Ü. , 1994.
- 21- Keyhani, A. and Miri, S. M. , " Observers for tracking of synchronous machine parametres and detection of incipient faults. " IEEE Trans. Energy Conver. vol. EC-1, June 1986.
- 22- Mageed, M. F. A. , Sakr A. F. and Bahgat A. , " Fault detection and identification using a hirearchical neural network- based system. " IEEE pp. 338-342, 1993.
- 23- Minsky and Papert S. , " Perceptrons. " Cambridge, MA:MIT Press, Introduction, pp. 1-20 and 73, 1969.
- 24- Nasar, S. A. , " Handbook of electric machines, " Substitutes New York: Mc-Graw Hill, 1987, chapter:3 and 4
- 25- Naylor, D. C. J. , " Artificial neural network review. " Department of Electrical and Electronic Engineering, University of Nottingham, 1990
- 26- Rosenblatt, " the perceptron: a probabilistic model for information storage and organization in the brain. " Psychological Review vol. 65, pp. 386-408, 1958
- 27- Rumelhart, E. , Hinton, G. E. and Williams, R. J. , " Learning representations by back-propagation errors. " Nature 323: 533-536, 1986
- 28- Sood, A. K. , Friedlander, C. B. and Fahs A. A. , " Engine fault analysis: Part I-statistical metods. " IEEE Trans. on Ind. Electron. , novem. 1985
- 29- Sood, A. K. , Friedlander, C. B. and Fahs A. A. , " Engine fault analysis: Part II-statistical metods. " IEEE Trans. on Ind. Electron. , novem. 1985
- 30- Timperley, J. E. , " Incipient fault identification through neutral R. F. monitoring of large rotating machines. " IEEE Trans. Power App. Sys. , vol. PAS-102, Mar. 1983
- 31- Werbos, P. J. , " Backpropagation through time: what it does and how to do it. " Proc. IEEE, vol. 78, pp. 1550-1560, Oct. 1990
- 32- Widrow, B. and Hoff, M. E. , " Adaptive switching circuits. " IRE WESCON Convention Record, New York : IRE pp. 96-104, 1960

## ÖZGEÇMİŞ

**Kayhan GÜLEZ**

Elektrik Mühendisi  
(1992 Yıldız Teknik Üniversitesi)

Doğum Tarihi ve Yeri

12.09.1970.İstanbul

1988-1992

Yıldız Teknik Üniversitesi  
Elektrik-Elektronik Fakültesi  
Elektrik Mühendisliği Bölümü

1984-1987

İstanbul Bayrampaşa A. Rifat Canayakın Lisesi

1981-1984

İstanbul Bayrampaşa Fetihstepe Ortaokulu

1976-1981

İstanbul Bayrampaşa Şair Baki İlkokulu

### GÖREVİ

1994

Araştırma Görevlisi  
Celal Bayar Üniversitesi  
Mühendislik Fakültesi  
Elektrik-Elektronik Mühendisliği Bölümü  
Devreler ve Sistemler Ana Bilim Dalı

Yabancı Dil

İngilizce

İSTANBUL  
YILDIZ TEKNİK ÜNİVERSİTESİ  
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ BÖLÜMÜ  
KAYHAN GÜLEZ