

PAMUKKALE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

**MAGNEZYUMDAN KAYNAKLI VE KAYNAKSIZ
YAPI ELEMANLARININ YORULMA DAVRANIŞLARININ
YAPAY SİNİR AĞLARI YÖNTEMİYLE BELİRLENMESİ**

**YÜKSEK LİSANS TEZİ
Ercüment BÜTÜNER**

Anabilim Dalı : Makine Mühendisliği

Tez Danışmanı: Prof. Dr. Alper GÜLSÖZ

EKİM 2011

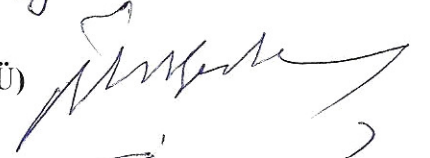
YÜKSEK LİSANS TEZ ONAY FORMU

Pamukkale Üniversitesi Fen Bilimleri Enstitüsü 091111016 nolu öğrencisi Ercüment BÜTÜNER tarafından hazırlanan “MAGNEZYUMDAN KAYNAKLI VE KAYNAKSIZ YAPI ELEMANLARININ YORULMA DAVRANIŞLARININ YAPAY SİNİR AĞLARI YÖNTEMİYLE BELİRLENMESİ” başlıklı tez tarafımızdan okunmuş, kapsamı ve niteliği açısından bir Yüksek Lisans tezi olarak kabul edilmiştir.

Tez Danışmanı : Prof. Dr. Alper GÜLSÖZ (PAÜ)



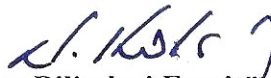
Jüri Üyesi : Yrd. Doç. Dr. Özler KARAKAŞ (PAÜ)



Jüri Üyesi : Yrd. Doç. Dr. Emre ÇOMAK (PAÜ)



Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 19/10/2011 tarih ve 28/11 sayılı kararıyla onaylanmıştır.


Fen Bilimleri Enstitüsü Müdürü
Prof. Dr. Nuri KOLSUZ

Bu tezin tasarımı, hazırlanması, yürütülmesi, arařtırmalarının yapılması ve bulgularının analizlerinde bilimsel etięe ve akademik kurallara özenle riayet edildiđini; bu alıřmanın dođrudan birincil ürünü olmayan bulguların, verilerin ve materyallerin bilimsel etięe uygun olarak kaynak gösterildiđini ve alıntı yapılan alıřmalara atfedildiđine beyan ederim.

İmza

: 

Öğrenci Adı Soyadı

: Ercüment BÜTÜNER

ÖNSÖZ

Bu çalışmada, yüksek maliyet ve uzun zaman gerektiren magnezyumdan kaynaklı ve kaynaksız yapı elemanlarına ait yorulma deneylerinden elde edilen sonuçlar ışığında, aynı sonuçlara yakın verilerin bilgisayar ortamında elde edilebilirliği araştırılmıştır. Bu amaçla, bilgisayar ortamında geliştirilmiş olan çok katmanlı algılayıcı yapay sinir ağı modelleri, eldeki veriler ile eğitilerek test edilmiştir. Elde edilen sonuçlar, yapay sinir ağlarının bu konuda da başarılı bir şekilde kullanılabileceğini göstermiştir.

Bu çalışma süresince benimle tecrübelerini paylaşan ve benden özverili yardımlarını esirgemeyen hocam Sayın Prof.Dr. Alper GÜLSÖZ'e, yaptığı doktora çalışmasında gerçekleştirdiği deneysel çalışmalardan elde ettiği değerleri kullanmama izin veren ve çalışma süresince bana destek olan hocam Sayın Yrd. Doç. Dr. Özler KARAKAŞ'a, yapay sinir ağları konusunda verdiği eğitim ve yardımlarıyla bu çalışmaya büyük bir katkıda bulunan hocam Sayın Yrd. Doç. Dr. Emre ÇOMAK'a çok teşekkür ederim.

Ayrıca, çalışmanın matematiksel konularında bana yardımcı olan ve her zaman manevi destekte bulunan arkadaşım Şerafettin Cem ÇINAR'a, bu çalışma süresince benden yardımlarını esirgemeyen ve manevi desteklerini her zaman yanımda hissettiğim anneme, babama, kardeşime ve nişanlıma çok teşekkür ederim.

Ekim 2011

Ercüment BÜTÜNER
(Makine Mühendisi)

İÇİNDEKİLER

Sayfa

ÖZET	xi
SUMMARY	xii
1. GİRİŞ	1
1.2 Literatür Özeti	2
2. YAPAY SİNİR AĞLARI	7
2.1 Akıl ve Zeka	7
2.2 Yapay Zeka.....	7
2.3 Yapay Sinir Ağları	9
2.4 Yapay Sinir Ağlarında Bilginin Depolanması ve Geri Alınması	11
2.5 Yapay Sinir Ağları ile Neler Yapılabilir?	12
2.6 Yapay Sinir Ağlarının Mühendislik Uygulamaları.....	14
2.6.1 Kimya mühendisliği uygulamaları.....	15
2.6.2 İnşaat mühendisliği uygulamaları	15
2.6.3 Elektrik ve elektronik mühendisliği uygulamaları.....	15
2.6.4 İmalat ve makine mühendisliği uygulamaları.....	16
2.6.5 Sistem ve kontrol mühendisliği uygulamaları	16
2.6.6 Harita mühendisliği uygulamaları.....	16
2.6.7 Bilgisayar mühendisliği uygulamaları	16
2.7 Yapay Sinir Ağlarının Geleceği	16
2.8 Biyolojik Bir Beyin Sinir Hücrelerinin Yapısı	17
2.9 Bir Yapay Sinir Ağının Ana Öğeleri	19
2.9.1 Girişler	20
2.9.2 Ağırlıklar	21
2.9.3 Toplama işlevi.....	21
2.9.4 Etkinlik işlevi.....	22
2.9.5 Ölçekleme ve sınırlama	24
2.9.6 Çıkış işlevi	24
3. ÇOK KATMANLI ALGILAYICI YSA MODELİ	27
3.1 Çok Katmanlı Algılayıcı (ÇKA).....	27
3.2 ÇKA Modelinin Yapısı	28
3.3 ÇKA Ağının Öğrenme Kuralı	30
3.3.1 İleri doğru hesaplama	30
3.3.2 Geriye doğru hesaplama	31
3.3.3 Ara katman ile çıktı katmanı arasındaki ağırlıkların değiştirilmesi.....	32
3.3.4 Ara katmanlar arası veya ara katman ile girdi katmanı arasındaki ağırlıkların değiştirilmesi.....	33
3.4 ÇKA Ağının Çalışma Prosedürü	34
3.5 Ağın Eğitilmesi.....	35
3.5.1 ÇKA ağının performansının ölçülmesi	39
3.6 Bir ÇKA Ağının Oluşturulmasında Dikkat Edilmesi Gereken Bazı Önemli Noktalar	40
3.6.1 Örneklerin seçilmesi.....	40
3.6.2 Girdi ve çıktılarının gösteriminin belirlenmesi.....	42
3.6.3 Başlangıç değerlerinin atanması	43
3.6.4 Öğrenme katsayısı ve momentum katsayılarının belirlenmesi.....	44
3.6.5 Örneklerin ağa sunulma şekli	45
3.6.6 Ağırlıkların değiştirilmesi zamanı	45

3.6.7 Örneklerin değerlerinin ölçeklendirilmesi.....	46
3.6.8 Durdurma kriterleri	46
3.6.9 Ara katman ve proses elemanları sayılarının belirlenmesi.....	47
3.7 ÇKA Ağı'nın Uygulama Alanları	47
4. MAGNEZYUMDAN YAPI ELEMANLARI ÜZERİNDE GERÇEKLEŞTİRİLEN YORULMA DENEYLERİNDEN ELDE EDİLEN VERİLERİN YSA YAKLAŞIMI İLE İNCELENMESİ	49
4.1 Çalışmada Kullanılan Deneysel Veriler	50
4.2 Kök Aralıksız Kaynaklı Numunelere Ait Yorulma Davranışlarının YSA Yöntemiyle İncelenmesi.....	50
4.2.1 Uygulamada kullanılacak olan YSA modelinin mimari yapısı	51
4.2.2 Problemin bilgisayar ortamına aktarılması.....	52
4.2.3 Verilerin programa okutulması.....	55
4.2.4 Yapay sinir ağı modelinin tasarımı, eğitimi ve simülasyonu	56
4.3 Köşe Dikişli Numunelere Ait Yorulma Davranışlarının YSA Yöntemiyle İncelenmesi.....	68
4.3.1 Problemin bilgisayar ortamına aktarılması.....	68
4.3.2 Verilerin programa okutulması.....	69
4.3.3 Yapay sinir ağı modelinin, eğitimi ve simülasyonu.....	70
4.4 Kaynaksız Numunelere Ait Yorulma Davranışlarının YSA Yöntemiyle İncelenmesi.....	80
4.4.1 Problemin bilgisayar ortamına aktarılması.....	81
4.4.2 Verilerin programa okutulması.....	81
4.4.3 Yapay sinir ağı modelinin, eğitimi ve simülasyonu.....	82
5. SONUÇLAR VE ÖNERİLER.....	86
5.1 Sonuçlar	86
5.2 Öneriler	88
KAYNAKLAR.....	90
ÖZGEÇMİŞ.....	92

KISALTMALAR

a	: YSA'nın simülasyonundan elde edilen normal değerler
ABD	: Amerika Birleşik Devletleri
an	: YSA'nın simülasyonundan elde edilen ölçeklendirilmiş değerler
ANN	: Artificial Neural Network (Yapay sinir ağı)
ASIC	: Application Specific Integrated Circuit (Uygulamaya özel tümleşik devre)
at	: YSA'nın test işlemi sonucunda elde edilen normal değerler
atn	: YSA'nın test işlemi sonucunda elde edilen ölçeklendirilmiş değerler
ÇKA	: Çok Katmanlı Algılayıcı
EM	: Esas Malzeme
KM	: Kaynak Malzemesi
NET	: Ara katmandaki proses elemanlarına gelen net girdi
p	: YSA'daki giriş matrisi
pctest	: YSA'nın test işlemi için oluşturulan giriş matrisi
t	: YSA'daki çıkış matrisi
TE	: Total Error (Çıktı katmanı için oluşan toplam hata)
ttest	: YSA'nın test işlemi için oluşturulan hedef matris
w	: YSA proses elemanları arası bağlantı ağırlık değeri
x	: YSA proses giriş değeri
y	: YSA proses çıkış değeri
YSA	: Yapay Sinir Ağları
z	: YSA'dan beklenen (hedeflenen) çıkış değeri

TABLO LİSTESİ

Tablolar

2.1: Geleneksel programlama ve yapay sinir ağlarının bazı özellikleri	11
2.2: Bazı toplama fonksiyonu örnekleri.....	22
2.3: Yapay sinir ile biyolojik sınırlar arasındaki benzerlikler.....	25
4.1: Nominal gerilme genliği 20 MPa ve gerilme oranı $R = 0$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler.....	53
4.2: Yaklaşık K_t değerleri hesaplanmış nominal gerilme genliği 20 MPa ve gerilme oranı $R = 0$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler	54
4.3: Gerilme oranı $R = -1$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile eğitilen YSA modelinden elde edilen değerler.....	59
4.4: Gerilme oranı $R = -1$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile test edilen YSA modelinden elde edilen değerler	61
4.5: Gerilme oranı $R = 0$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile eğitilen YSA modelinden elde edilen değerler.....	63
4.6: Gerilme oranı $R = 0$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile test edilen YSA modelinden elde edilen değerler.....	64
4.7: Gerilme oranı $R = 0,5$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile eğitilen YSA modelinden elde edilen değerler.....	65
4.8: Gerilme oranı $R = 0,5$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile test edilen YSA modelinden elde edilen değerler	66
4.9: Nominal gerilme genliği 30 MPa ve gerilme oranı $R = 0$ olan köşe dikişli kaynaklı numunelere ait deneysel veriler	69
4.10: Gerilme oranı $R = -1$ olan köşe dikişli kaynaklı numunelere ait deneysel veriler ile eğitilen YSA modelinden elde edilen değerler.....	71
4.11: Gerilme oranı $R = -1$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile test edilen YSA modelinden elde edilen değerler	73
4.12: Gerilme oranı $R = 0$ olan köşe dikişli kaynaklı numunelere ait deneysel veriler ile eğitilen YSA modelinden elde edilen değerler.....	74
4.13: Gerilme oranı $R = 0$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile test edilen YSA modelinden elde edilen değerler	76
4.14: Gerilme oranı $R = 0,5$ olan köşe dikişli kaynaklı numunelere ait deneysel veriler ile eğitilen YSA modelinden elde edilen değerler.....	77
4.15: Gerilme oranı $R = 0,5$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile test edilen YSA modelinden elde edilen değerler	78
4.16: Gerilme oranı $R = 0,5$ olan köşe dikişli kaynaklı numunelere ait iki farklı test işleminden elde edilen değerlerin karşılaştırılması	79
4.17: Kaynaksız numunelere ait deneysel verilerle programlanan YSA modelinden elde edilmiş olan değerler.....	82

ŞEKİL LİSTESİ

Şekiller

2.1: Von-Neumann mimari temeli.....	9
2.2: Basitleştirilmiş biyolojik bir beyin sinir hücresi	17
2.3: Temel nöron tipleri	18
2.4: Yapay bir sinir	20
2.5: Bazı etkinlik işlevi örnekleri	23
3.1: ÇKA modelinin yapısı	28
3.2: Hata – ağırlık eğrisi (basit).....	36
3.3: Hata – ağırlık eğrisi (karmaşık).....	37
3.4: Hata – iterasyon ilişkisi.....	39
3.5: Ağırlık başlangıç değerlerinin önemi	44
4.1: Kök aralıksız kaynaklı numunelerin kaynak dikiş şeklinin şematik gösterimi	50
4.2: Kullanılan YSA modeli.....	51
4.3: Yaklaşık çentik form sayılarının hesaplanması.....	54
4.4: Programa okutulmak üzere düzenlenen not defteri dosyası formatı	55
4.5: Verilerin program tarafından okunması, giriş ve çıkış matrislerinin oluşturulması	56
4.6: YSA modelinin çalışma şekli	57
4.7: YSA'nın tasarımı, eğitimi ve simülasyonu	58
4.8: Kök aralıksız kaynaklı numunelerin eğitim serisi ($R = -1$).....	60
4.9: YSA'nın test işlemi.....	61
4.10: Kök aralıksız kaynaklı numunelerin test serisi ($R = -1$)	62
4.11: Kök aralıksız kaynaklı numunelerin eğitim serisi ($R = 0$)	63
4.12: Kök aralıksız kaynaklı numunelerin test serisi ($R = 0$).....	64
4.13: Kök aralıksız kaynaklı numunelerin eğitim serisi ($R = 0,5$)	66
4.14: Kök aralıksız kaynaklı numunelerin test serisi ($R = 0,5$).....	67
4.15: Köşe dikişli kaynaklı numunelerin kaynak dikiş şeklinin şematik gösterimi	68
4.16: Verilerin program tarafından okunması, giriş ve çıkış matrislerinin oluşturulması	69
4.17: YSA'nın tasarımı, eğitimi ve simülasyonu	70
4.18: Köşe dikişli kaynaklı numunelerin eğitim serisi ($R = -1$).....	71
4.19: YSA sisteminin test işlemi	72
4.20: Köşe dikişli kaynaklı numunelerin test serisi ($R = -1$)	73
4.21: Köşe dikişli kaynaklı numunelerin eğitim serisi ($R = 0$)	75
4.22: Köşe dikişli kaynaklı numunelerin test serisi ($R = 0$).....	76
4.23: Köşe dikişli kaynaklı numunelerin eğitim serisi ($R = 0,5$)	77
4.24: Köşe dikişli kaynaklı numunelerin test serisi ($R = 0,5$).....	78
4.25: Gerilme oranı $R = 0,5$ olan köşe dikişli kaynaklı numunelere ait iki farklı test işleminden elde edilen değerlerin karşılaştırılması	79

4.26: Kaynaksız numunelerin şematik yapısı	80
4.27: Verilerin program tarafından okunması, giriş ve çıkış matrislerinin oluşturulması	81
4.28: Kaynaksız numunelerin eğitim serisi ($R = -1$)	83
4.29: Kaynaksız numunelerin eğitim serisi ($R = 0$).....	84

SEMBOL LİSTESİ

D	Test setinden doğru olarak hesaplanan örnek sayısı
E	Error (Hata)
E_m	Çıktı katmanındaki m. proses elemanı için oluşan hata
K_t	Çentik form sayısı
N_B	Kırılma çevrim sayısı
P	Performans oranı
R	Gerilme oranı
T	Test setindeki toplam örnek sayısı
W	Weight (Ağırlık)
W^*	Hata tolerans değerini elde etmek için hedeflenen ağırlık değeri
β_j	Ara katmanda bulunan j. elemana bağlanan eşik değer elemalarının ağırlık değeri
δ_m	m. çıktı ünitesinin hatası
ε	Hata tolerans değeri
v	Toplama işlevi
σ_a	Gerilme genliği
Θ	Eşik değer

ÖZET

MAGNEZYUMDAN KAYNAKLI VE KAYNAKSIZ YAPI ELEMANLARININ YORULMA DAVRANIŞLARININ YAPAY SİNİR AĞLARI YÖNTEMİ İLE BELİRLENMESİ

Magnezyum, düşük yoğunluğundan kaynaklanan hafiflik, özgül dayanımının yüksek olması ve kaynağındaki gelişmelerden dolayı son yıllarda endüstriyel uygulamalarda gittikçe artan bir şekilde kullanılmaktadır. Bu avantajlarından dolayı magnezyum, özellikle taşıt tekniği için büyük bir önem arz etmektedir. Buna paralel olarak kaynaklı ve kaynaklı olmayan magnezyum alaşımlarının yorulma davranışlarıyla ilgili, bilhassa gelişmiş ülkelerde araştırma-geliştirme faaliyetlerinin ve bu konuyla ilgili yayınların da arttığı görülmektedir. Bilindiği üzere yorulma davranışlarının incelenmesine ait araştırma-geliştirme faaliyetleri hem maliyeti yüksek hem de uzun bir zaman gerektiren çalışmalardır.

Yapay sinir ağları, günümüzde birçok mühendislik problemine hızlı ve ekonomik çözümler üretebilen sistemlerdir. Yapay sinir ağlarının hızlı ve ekonomik olmaları bu sistemlerin, uzun bir zaman dilimi gerektiren ve maliyetli çalışmalar olan yorulma deneyleri için kullanılabilmesini cazip hale getirmektedir. Yapılan literatür araştırmasında yapay sinir ağı uygulamalarının bu konuya paralel olan bir çok mühendislik uygulamasında hayat bulduğu görülmektedir.

Bu çalışmada elimizde hazır olarak bulunan gerçek deneysel çalışmalardan elde edilen veriler, Matlab R2009a bilgisayar programında tasarlanan, bu probleme en uygun olarak görülen ileri beslemeli çok katmanlı algılayıcı yapay sinir ağı modelinde kullanılmıştır. Yapay sinir ağı modelinden elde edilen sonuçlar ile deneysel sonuçlar daha iyi yorum getirebilmek için, karşılaştırmalı olarak, grafikler halinde gösterilmiştir.

Anahtar kelimeler: Magnezyum, yapay sinir ağları, yorulma, kaynak

SUMMARY

DETERMINATION OF FATIGUE BEHAVIOURS OF WELDED AND UNWELDED MAGNESIUM COMPONENTS BY ARTIFICIAL NEURAL NETWORKS PRACTICE

In recent years, Magnesium is increasingly used in industrial applications owing its low density, lightness, specific strength and developments of its welding techniques. By the virtue of these advantages, magnesium has importance especially in vehicle techniques. Besides, researches and publications about welded and unwelded magnesium components are observed particularly in developed countries. As it is known, research-development activities about fatigue behaviour are expensive studies and require long time periods.

Nowadays, artificial neural networks can produce quickly and affordable solutions for several engineering problems. Being quick and affordable make artificial neural networks attractive for fatigue tests which require long time periods and are costly. In literature survey, it can be observed that artificial neural networks spring to life within several engineering applications.

In this study, data sets which had been obtained from genuine experiments are used in multilayer feed forward perceptron model which is suitable for this problem and designed by Matlab R2009a computer programme. Results from artificial neural networks and genuine experiments are drawn comparatively.

Keywords: Magnesium, artificial neural networks, fatigue, welding

1. GİRİŞ

Magnezyum, düşük yoğunluğundan kaynaklanan hafiflik, özgül dayanımının yüksek olması ve kaynağındaki gelişmelerden dolayı son yıllarda endüstriyel uygulamalarda gittikçe artan bir şekilde kullanılmaktadır. Magnezyumun yoğunluğu $1,7 \text{ g / cm}^3$ (alüminyumun yoğunluğu = $2,7 \text{ g / cm}^3$, çeliğin yoğunluğu = $7,8 \text{ g / cm}^3$) olmasına rağmen alüminyum ile hemen hemen aynı dayanım değerlerine sahiptir (200 – 250 MPa). Hatta bazı magnezyum alaşımları, alüminyum alaşımlarına göre % 35 daha hafif olmasına rağmen dayanım/ağırlık yani özgül dayanım değerleri alüminyuma göre daha yüksek değerlere ulaşabilmektedir. Böylece magnezyum alaşımları hafifliğin önem kazandığı kara ve hava taşıtları, el aletleri, spor gereçleri, ev ve büro malzemeleri, elektronik birimlerde ve daha birçok makine ve cihazların imalinde gittikçe artan bir şekilde kullanılmaya başlanmıştır. Bu avantajlarından dolayı magnezyum, özellikle taşıt tekniği için büyük bir önem arz etmektedir. Buna paralel olarak kaynaklı ve kaynaklı olmayan magnezyum alaşımlarının yorulma davranışlarıyla ilgili, bilhassa gelişmiş ülkelerde araştırma-geliştirme faaliyetlerinin önem kazandığı ve arttığı görülmektedir. Bilindiği üzere yorulma ile ilgili bu araştırma-geliştirme faaliyetleri hem maliyeti yüksek hem de uzun bir zaman gerektiren çalışmalardır.

Yapay sinir ağları günümüzde doğrusal olmayan birçok mühendislik problemi için hızlı ve ekonomik bir şekilde, başarılı yorumlar ve çözümler üretebilen, insan beyninin çalışma prensibinden (paralel bilgi işleme) yola çıkılarak bilgisayar ortamında tasarlanan programlardır. Yapay sinir ağları, bu özelliklerinden dolayı yorulma deneyleri gibi zaman ve maliyet gerektiren çalışmalar için cazip bir uygulama olmaktadır.

Bu çalışmanın amacı, malzemelerin yorulma davranışlarını yorumlayabilecek yapay sinir ağı mimarilerinin tasarlanması, magnezyumdan kaynaklı ve kaynaklı olmayan yapı elemanlarına ait yorulma deneylerinden elde edilmiş verilerin, tasarlanmış bu ağların öğretmenli öğrenme prensibi ile eğitilmesinde kullanılması ve ardından test için ayrılan ve eğitimde kullanılmayan veriler ile ağların test edilmesi ve son olarak da

ağın kendi ürettiği sonuçlar ile gerçek verilerin karşılaştırılmasıdır. Başka bir deyişle, yapay sinir ağları yaklaşımının magnezyumdan kaynaklı ve kaynaksız yapı elemanlarına ait yorulma davranışlarının yorumlanabilirliğini ortaya çıkarmaktır.

Yapılacak çalışmaya ait detaylı çalışma adımları aşağıda gösterilmiştir.

1. Yapay sinir ağları ve kullanım alanları ile ilgili bir ön araştırma yapılacaktır.
2. Yapay sinir ağları çeşitlerinin kullanım alanları, eğitim şekilleri ve çalışma prensiplerine göre, bu problem için kullanılacak en uygun yapay sinir ağı modeli belirlenecektir.
3. Elimizde üç ayrı grup şekilde bulunan (kaynaksız, kök aralıksız ve köşe dikişli numuneler) numunelere ait deneysel sonuçlar, yapay sinir ağları için eğitim ve test grupları halinde ayrılacaktır.
4. Bütün veri gruplarının sadece bir yapay sinir ağı modelinde mi ya da her grup için ayrı ayrı hazırlanacak dört tane bağımsız yapay sinir ağı modellerinde mi kullanılacağına karar verilecektir.
5. Kullanılacak olan yapay sinir ağı modelleri, verilere uygun bir mimariye sahip olacak şekilde bilgisayar ortamında Matlab R2009a programı yardımıyla oluşturulacaktır.
6. Oluşturulan sinir ağları, önceden eğitim ve test grupları olarak dört ayrı gruba ayrılmış olan veriler yardımı ile eğitilecek ve test edilecektir. Bu aşamada yapay sinir ağlarına ait eğitim parametreleri ve ağın topolojik yapısı ağdan elde edilen sonuçlara göre değiştirilerek optimum bir model oluşturulacaktır.
7. Yapay sinir ağı modellerinden elde edilen veriler ile deneysel çalışmalardan elde edilen sonuçlar karşılaştırmalı olarak grafikler halinde gösterilecektir.

1.2 Literatür Özeti

Elmas Çetin'in hazırladığı "Yapay Sinir Ağları" adlı kitap yapay sinir ağları, yapay sinir ağlarının çeşitleri, çalışma şekilleri, oluşturulması, öğrenme parametreleri ve uygulamaları hakkında genel bilgiler içermektedir [1].

Ercan Öztemel'in hazırladığı "Yapay Sinir Ağları" adlı kitap yapay zeka teknolojileri, makine öğrenmesi, yapay sinir ağlarının yapıları ve elemanları, yapay sinir ağlarının tarihçesi, yapay sinir ağları çeşitleri, günümüzdeki yapay sinir ağları uygulamaları hakkında detaylı bilgiler vermektedir [2].

Mehmet Erler ve arkadaşlarının hazırlamış olduğu "Mühendislikte Yapay Zeka Uygulamaları – I" adlı kitap, yapay zeka kavramı ve teknikleri, yapay sinir ağları ve yapıları, öğrenme algoritmaları ve yapay sinir ağı tasarımında dikkat edilmesi gereken hususlar gibi konulara değinmiştir. Ayrıca mühendislik konusunda YSA uygulamaları ile ilgili bazı örnekler içermektedir [3].

Yapılan bu çalışmada, Özler Karakaş'ın, deneysel kısımlarını Almanya'nın Darmstadt kentinde bulunan Fraunhofer LBF İşletme Dayanımı ve Sistem Güvenirliği Enstitüsü'nde gerçekleştirdiği, Pamukkale Üniversitesi bünyesindeki "Biçimlenebilen Magnezyum Alaşımlarından Kaynaklı Yapı Elemanlarının Yorulma Dayanımı Değerlendirmelerinde Çentik Gerilmesi Yönteminin Uygulanması" başlıklı doktora çalışması kapsamında elde edilen, magnezyumdan kök aralıklı kaynaklı, kök aralıksız kaynaklı, köşe dikişli kaynaklı ve kaynaklı numunelere ait deneysel veriler kullanılmıştır [5].

Dongho Bae, ve arkadaşları farklı geometri ve ölçülerdeki nokta kaynağı bağlantıları için daha önceki yorulma testlerinde elde edilmiş olan değerler yardımıyla deneysel çalışmalardan daha ekonomik ve avantajlı yorulma dizayn metodunu yapay sinir ağları yöntemi ile geliştirmişlerdir [7].

P. Obranic ve M. Fajdiga alüminyum-çelik bağlantılarında fretting yorulmasını tanımlayabilmek amacıyla yapay sinir ağları yaklaşımını kullanmışlardır. Yapay sinir ağları yaklaşımı ile yapay sinir ağları, fretting yorulma olgusunu modellemek için eğitilebilmiştir. Bu çalışmada elde edilen en temel fayda eğitilmiş yapay sinir ağlarının fretting yorulması olayındaki farklı faktörlerin etkisini kesin olarak tanımlayabilmiş olmasıdır [8].

M.D. Mathew ve arkadaşları yapay sinir ağı modellemesini azot alaşımlı 316L paslanmaz çeliğin kısa zaman yorulma ömrü tahmini için uygulamışlardır. Bu amaçla, eldeki deney verileri kullanılarak 316L paslanmaz azot alaşımlı çeliğin yorulma ömrü tahmini için yapay sinir ağı modeli geliştirilmiştir [9].

F. Iacoviello ve arkadaşları, gerilme oranının sinterlenmiş dubleks paslanmaz çeliklerdeki çatlak yayılmasına etkisini deneysel ve yapay sinir ağı yaklaşımı yardımıyla incelemişlerdir. Yapılan çalışmada, PM dubleks paslanmaz çelikte R gerilme oranının yorulma çatlak yayılımına etkisi araştırılmış ve yapay sinir ağı tabanlı model yeni bir simülasyon aracı olarak optimize edilmiştir [10].

Jamal A. Abdalla ve Rami Hawileh yaptıkları çalışmada yapay sinir ağı modellemesini, çelik takviye çubukların kısa zaman yorulma davranışlarının tahmini için kullanmışlardır [11].

K. Genel yapay sinir ağlarının, çeliklerin gerilme-ömür yorulma özelliklerinin tahmininde kullanılabilirliğini çelik malzemelere ait çekme dayanımı verileri ile birbirinden ayrı dört yapay sinir ağı modelini kullanarak araştırmıştır. Yapay sinir ağı modelleri gerilme genliği ve yorulma ömrü eğrilerini birincil derecede karakterize eden yorulma dayanım katsayısı ve yorulma süneklik katsayılarını %98 ve %99 doğrulukta ayrı ayrı tahmin edilebildiğini göstermiştir. YSA yaklaşımının doğru bir şekilde kullanılması neticesinde, eğitilmiş YSA modellerinden elde edilen değerlerin, yaklaşık metotlara göre daha makul olduğu görülmektedir [12].

C. Marquardt ve H. Zenner, Miner ve Palmgren'e göre hasarın lineer hasar birikimine göre değerlendirildiği, yapı elemanları için genelde dış yüklemeler ile yapı elemanının dayanımının karşılaştırılmasına dayanan yorulma ömrü hesabı yöntemlerine alternatif olarak yapay sinir ağları ile, değişken genlikli yüklemeler altındaki yorulma ömrünü hesaplamışlardır [13].

J.R. Mohanty ve arkadaşları yapay sinir ağları yardımıyla serpiştirilmiş karma-modlu aşırı yüklemeler altında artık yorulma ömrü tahmininde bulunmuşlardır. Tahmin edilen sonuçların iyi bir uyum içerisinde olduğu ve artık yorulma ömrünün belirlenmesini kolaylaştırdığı görülmektedir [14].

İran'ın kuzeyinde bulunan Neka elektrik santralindeki yeniden ısıtma sistemine ait borularda belirli periyotlarla yapılan ölçümlerde et kalınlıklarının değiştiği tespit edilmiştir. A. Majidian ve M.H. Saidi, et kalınlığının zamana bağlı değişiminden yola çıkılarak, bu boruların işletme ömürlerini yapay sinir ağları, bulanık mantık ve genetik algoritma yöntemleriyle tahmin etmişlerdir. Sonuçlar, yöntemlerin doğruluğu ve geçerliliğini tanımlamak amacıyla literatürdeki hazır veriler ile karşılaştırılmıştır [15].

Kompozit malzemelerin spektrum yorulma ömürlerinin tahmini için A.P. Vassilopoulos ve arkadaşları iki farklı materyale ait deneysel veriler ışığında yapay sinir ağları yöntemini uygulamışlardır. Ayrıca geleneksel yöntemlere göre çok daha küçük deneysel düzenek ile değişken spektrum yüklemeler altındaki yapı elemanlarının dizaynı için oldukça kullanışlı olan yorulma ömür diyagramlarını verimli bir şekilde elde etmişlerdir [16].

Malzemelerdeki gerilme yoğunluğu faktörünün belirlenmesi için Ki-Bok Kim ve arkadaşları kompakt gerilme numunelerindeki mikro çatlaktan kırılmaya kadar olan akustik emisyon ölçümlerinden elde edilen veriler ışığında yapay sinir ağlarını modellemişlerdir [17].

A.Bezazi ve arkadaşları eğilme gerilmeleri altındaki sandviç kompozit yapıların yorulma ömürlerinin tahmininde Bayesian ifadesine göre eğitilmiş yapay sinir ağlarını kullanmışlardır [18].

316L(N) çeliğinin kısa zaman ve sürünme yorulma davranışlarının etkileşimi ve ömür tahminlerini V.S. Srinivasan ve arkadaşları elde edilen deneysel veriler ışığında modelledikleri yapay sinir ağları ile elde etmişlerdir [19].

Raimundo Carlos Silverio Freire Jr. ve arkadaşları yeni ağ yapılarının sürekli ömür diyagramlarının elde edilmesindeki kullanımını belirlemek amacıyla, daha önceden yayınlanmış olan ileri beslemeli yapay sinir ağı modelinin devamı olarak, az sayıdaki S – N eğrilerinden elde edilmiş olan değerlerle eğitilmiş yeni sinir ağlarını kullanmışlar ve bu çalışma neticesinde daha iyi sonuçlar elde etmişlerdir [20].

A. Fathi ve A.A. Aghakoucak, kaynaklı boru bağlantılarındaki yorulma çatlak yayılması tahmininde yapay sinir ağı yöntemini kullanmışlardır. T kaynak bağlantılarındaki kaynak dikişi çatlaklarının büyüme faktörünün tahmini için sonlu elemanlar metodundan elde edilen veriler ile eğitilen 4 adet MLP ağı geliştirilmiştir [21].

Özler Karakaş yaptığı çalışmada, alüminyumdan kaynaklı 4 farklı alaşım üzerinde yorulma ömürlerinin yapay sinir ağı yöntemiyle tahmin edilebileceğini göstermiştir [22].

2. YAPAY SİNİR AĞLARI

2.1 Akıl ve Zeka

Akıl kelimesi toplumda genellikle insanların zeka düzeyini ifade etmek amacıyla kullanılmaktadır. Sıklıkla akıl kavramı zekayla karıştırılmaktadır. Oysa akıl, düşünme, anlama, kavrama, idrak etme, karar verme ve önlem alma yetenekleridir. Akıl aynı zamanda muhakeme ve bilgi elde etme gücü olarak da tanımlanabilir. Zeka, gerçekleri algılama, yargılama ve sonuç çıkarma yeteneklerinin tamamıdır.

Akıl, genetik yoldan intikal eden sevgi, korku, kıskançlık, doğal savunma güdülerinin yanı sıra bulunduğumuz çevreden aldığımız etkileşimlerden ve toplumun şartlandırmalarından etkilenecek gelişmektedir. Dolayısıyla akıl sabit değil, aksine insanın hayatının sonuna kadar artabilen ve gelişebilen bir yetenektir. Akıl, makine, bilgisayar, yazılım veya başka bir yolla taklit edilemez.

Her insan doğuştan belli bir zekaya sahiptir. Zeka, belirli bir konuda çalışılarak, öğretilerek, eğitilerek, edinilen bilgi ve birikimlerle, deneyimlere dayalı becerilerle geliştirilebilir. İlk kez karşılaşılan ya da ani olarak gelişen bir olaya uyum sağlayabilme, anlama, öğrenme, analiz yeteneği, beş duyunun, dikkatin ve düşüncenin yoğunlaştırılması zeka ile gerçekleştirilebilmektedir. Zeka, yazılım veya tümleşik yongalarla (entegre devre, çip) taklit edilebilmektedir. Bu durumda zeka “yapay zeka” olarak adlandırılmaktadır.

2.2 Yapay Zeka

İnsan beyni dünyanın en karmaşık makinesi olarak kabul edilebilir. İnsan beyni sayısal bir işlemi birkaç dakikada yapabilmesine karşın, idrak etmeye yönelik olayları çok kısa bir sürede yapar. Örneğin yolda giden bir şoför, yolun kayganlık derecesinin, önündeki tehlikeden ne kadar uzak olduğunu, sayısal olarak değerlendirmese dahi geçmişte kazanmış olduğu tecrübeler sayesinde aracın hızını azaltır. Çünkü o saniyelerle ölçülebilecek kadar kısa bir sürede tehlikeyi idrak etmiş ve ona karşı koyma gibi bir tepki vermiştir. Bu noktada akla gelen ilk soru şu olmaktadır: Acaba

bir bilgisayar yardımı ile böyle bir zeka üretmek mümkün olabilir mi? Bilgisayarlar çok karmaşık sayısal işlemleri anında çözümleyebilmelerine karşın, idrak etme ve deneyimlerle kazanılmış bilgileri kullanabilme noktasında çok yetersizdirler. Bu olayda insanı ya da insan beynini üstün kılan temel özellik, sinirsel algılayıcılar vasıtası ile kazanılmış ve görelî olarak sınıflandırılmış bilgileri kullanabilmesidir. Uzman sistemler (US), bulanık mantık (BM), genetik algoritma (GA) ve yapay sinir ağları (YSA) gibi yapay zeka alt dalları özellikle son yıllarda, geniş bir araştırma ve uygulama alanı bulmaktadırlar.

Uzman sistem, temelde insan düşüncelerini gerçekleştirmek amacıyla bilgisayar tarafından işlenen bir yazılımdır. Uzman sistem geliştirilirken, uzmanların belli bir konudaki bilgi ve deneyimlerinin bilgisayarlara aktarılması amaçlanmaktadır.

Bulanık mantık, bulanık küme teorisine dayanan bir matematiksel disiplindir. Bulanık mantık insan mantığında olduğu gibi, Uzun-Kısa, Sıcak-Soğuk, Hızlı-Yavaş, Uzun-Ortadan Uzun-Orta-Ortadan Kısa-Kısa, Sıcak-Ilık-Az Soğuk-Soğuk-Çok Soğuk vb. gibi ara değerlere göre çalışmaktadır.

Genetik algoritmalar yapay zekanın gittikçe genişleyen bir kolu olan evrimsel hesaplama tekniğinin bir parçasını oluşturmaktadır. Genetik algoritma Darwin'in evrim kuramı (doğada en iyinin yaşaması) kuralından esinlenerek oluşturulan, bir veri öbeğinden (küme, grup) özel bir veriyi bulmak için kullanılan bir arama yöntemidir.

Genetik algoritma, geleneksel yöntemlerle çözümü zor veya imkansız olan problemlerin çözümünde kullanılmaktadır.

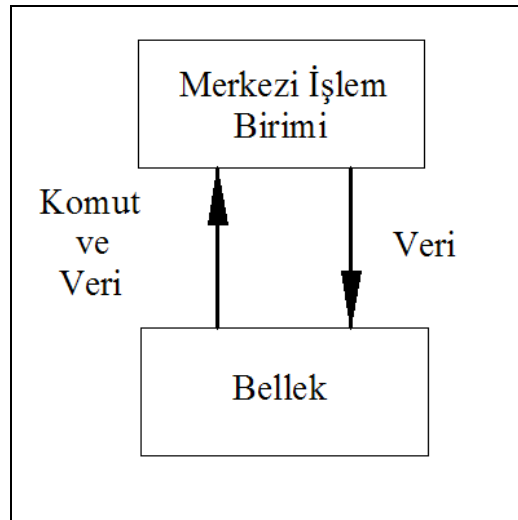
Herhangi bir problemin genetik algoritma ile çözümü, problemi sanal olarak evrimden geçirerek yapılmaktadır.

Genel olarak yapay sinir ağları insan beyninin sinir ağlarını taklit eden bilgisayar programlarıdır. Yapay sinir ağları, bir anlamda paralel bilgi işleme sistemi olarak düşünülebilir. Yapay sinir ağlarına bu bilgiler ilgili olaya ait örnekler üzerinde eğitilerek verilir. Böylelikle, örnekler sayesinde açığa çıkarılmış özellikler üzerinde çeşitli genelleştirmeler yapılarak daha sonra ortaya çıkacak ya da o ana kadar hiç rastlanmamış olaylara çözümler üretilmektedir.

2.3 Yapay Sinir Ağları

Günümüzde bilgi işleme büyük çoğunlukla sayısal bilgisayarlar ile gerçekleştirilmektedir. Bu durum, bilgi işlemenin sayısal bilgisayarlara bağımlı olduğu gibi yaygın bir yanılgıya yol açmıştır. Fakat bilgi biliminin temelini oluşturan sibernetik ve diğer disiplinlere baktığımızda bilgi işlemenin kendi ortamlarında hayatta kalmak için mücadele eden canlılarla birlikte ortaya çıktığını ve bugün bilgisayarlar tarafından işlenen bilginin bunun sadece küçük bir parçasını oluşturduğunu görürüz. Gerçekte bilgi işlem makinelerinin değişik türlerini ortaya çıkarmayı amaçlayan araştırmalar, sayısal bilgisayar dünyasındaki baş döndürücü gelişmelerin gölgesinde kalmasına rağmen devam etmektedir. Bu araştırmaların bir yönünü insanların ve diğer canlıların sahip olduğu yapıları, işletim ilkelerini taklit eden bir bilgi işletim sistemini geliştirmek oluşturmaktadır.

Sayısal bilgisayarlar 1940'ların sonlarından günümüze değin hızla gelişmiştir. Önceleri matematik hesaplamalarda kullanılan sayısal bilgisayarlar, daha sonra metin, sembol, resim ve ses işlemeyi de kapsayan geniş bir uygulama alanı bulmuştur. Bir sayısal bilgisayarın işlevsel özelliği Şekil 2.1'de görülmektedir. Sayısal bilgisayarlar Von-Neumann mimari temeli üzerine kurulmuşlardır.



Şekil 2.1: Von-Neumann mimari temeli

Son zamanlardaki sinirsel ve sinirsel-psikolojik deneyler beyin yapısını önemli ölçüde aydınlatmıştır. İnsanın bilgi işlem sürecini geniş açıdan inceleyen bilişsel (Cognitive) bilim gibi alanlarda paralel çalışma için birleştirilmiş çoklu işlem elemanlarını içeren modeller önerilmiştir. Ayrıca matematik ve fizik alanındaki

arařtırmalar deęiřik yollarla birleřtirilen çoklu elemanları ieren matematik analizleri zerine daha ok yoęunlařmaktadır. Bu etmenler insanların ve hayvanların bilgi iřlem sistemlerinde grlen alıřma ilkelerini ve yapılarını aıęa ıkarmayı amalayan, bu yapı ve alıřma ilkelerini temel alan bir bilgi iřlem sistemini inřa etme yolunda arařtırmaları artırmıřtır. Sinirsel-bilgi-iřlem (Neuro-computing) terimi bu arařtırmanın bilgi mhendislięiyle ilgili ynlerini ifade etmek iin kullanılan terimdir.

Yapay sinir aęları (YSA), insan beyninden esinlenerek geliřtirilmiř, aęırlıklı baęlantılar aracılıęıyla birbirine baęlanan ve her biri kendi belleęine sahip iřlem elemanlarından oluřan paralel ve daęıtılmıř bilgi iřleme yapılarıdır. Yapay sinir aęları, bir bařka deyiřle, biyolojik sinir aęlarını taklit eden bilgisayar programlarıdır. Yapay sinir aęları zaman zaman baęlantıcılık (connectionism), paralel daęıtılmıř iřlem, sinirsel-iřlem, doęal zeka sistemleri ve makine ęrenme algoritmaları gibi isimlerle de anılmaktadır.

Yapay sinir aęları bir programcının geleneksel yeteneklerini gerektirmeyen, kendi kendine ęrenme dzenekleridir. Bu aęlar ęrenmenin yanı sıra, ezberleme ve bilgiler arasında iliřkiler oluřturma yeteneęine de sahiptir.

Yapay sinir aęları insan beyninin bazı organizasyon ilkelerine benzeyen zelliklerini kullanmaktadırlar. Yapay sinir aęları bilgi iřleme sistemlerinin yeni neslini temsil ederler. Genel olarak yapay sinir aęları model seimi, sınıflandırılması, iřlev tahmini, en uygun deęeri bulma ve veri sınıflandırma gibi iřlerde bařarılıdır. Geleneksel bilgisayarlar ise zellikle model seme iřinde verimsizdir ve sadece algoritmaya dayalı hesaplama iřlemleri ile kesin aritmetik iřlemlerde hızlıdırlar.

Bir ok yapay sinir aęı tipi bulunmakla birlikte bazılarının kullanımını dięerlerinden daha yaygındır. En ok kullanılan yapay sinir aęı geri yayılımlı yapay sinir aęı olarak bilinendir. Bu tip yapay sinir aęı tahmin ve sınıflandırma iřlemlerinde ok iyi sonular vermektedir. Bir bařkası Kohonen zrgtlemeli Harita'dır. Bu tip sinir aęları, karıřık bilgi kmeleri arasında iliřki bulma konusunda bařarılı sonular vermektedir.

Belirsiz, grltl ve eksik bilgilerin iřlenmesinde yapay sinir aęları bařarıyla kullanılmaktadır.

Yapay sinir aęları geleneksel programlamalardan ok farklı zellikler gsterirler. Geleneksel programlama ve yapay sinir aęlarının bazı zellikleri Tablo 2.1'de verilmiřtir.

Tablo 2.1: Geleneksel programlama ve yapay sinir ağlarının bazı özellikleri [1]

Geleneksel Programlama	Yapay Sinir Ağları
Tümdengelimli usavurma: Çıkış üretmek için giriş bilgilerine bilinen kurallar uygulanır.	Tümevarımlı usavurma: Giriş ve çıkış bilgileri (eğitilen örnekler) verilir, kuralları biz koyarız.
Hesaplama merkezi, eşzamanlı ve ardışıktır.	Hesaplama toplu, eşzamansız ve öğrenmeden sonra paraleldir.
Bellek paketlenmiş, hazır bilgi depolanmış ve yer adreslenebilir.	Bellek ayrılmıştır, dahilidir ve içerik adreslenebilir.
Hata toleransı yoktur.	Eğer bilgi, gürültülü veya kısmi ise kurallar bilinmiyorsa ya da karışıkta hata toleransı uygulanabilir.
Hızlıdır.	Yavaştır.
Bilgiler ve Algoritmalar kesindir.	Yapay sinir sistemleri deneyimden yararlanır.

2.4 Yapay Sinir Ağlarında Bilginin Depolanması ve Geri Alınması

Yapay sinir ağlarında herhangi bir örnek giriş verisinin tanınabilmesi ve bunun daha sonra kullanılabilmesi için verinin ağda nasıl temsil edildiğinin, nerede saklandığının ve nasıl geri alındığının bilinmesi gerekmektedir.

Klasik bilgisayarlarda bilgi 1 ve 0 serileri ile temsil edilirken, yapay sinir ağlarında matematiksel işlev ile temsil edilir. İşlem elemanları arasındaki bağlantının ağırlıkları bu işlevin değişkenleri olarak görev yaparlar. Burada ağırlıklar saklanan bilginin ne olduğuna karar verir, fakat kendilerinin bir anlamı yoktur.

Sinir ağlarındaki bilgi, ağ içindeki bağlantılarda ve birçok ağırlıklar yoluyla dağılmaktadır. Klasik bilgisayar bilgiyi belleğinde belirli bir yerde saklar, sinir ağları ise bilgiyi tüm ağ boyunca dağıtır. Bu durumda dağıtılmış bellek olarak bilinir.

Klasik bilgisayarlarda bilgi, belirli bir bellek bölgesine ulaşılarak alınır. Sinir ağlarında ise, bilgi muhtemelen görüntü ya da tamamlanmamış giriş örneği olarak ağa sunulur. Ağ, giriş ile tüm ihtimalleri birleştirerek en iyi ve en uygun örneği çıkış olarak seçer. Bu çıkış örneği, ağ tarafından giriş örneği hakkında eldeki bilgi olarak temsil edilir. Bu durum ise birleştirilmiş bellek olarak bilinir [1].

2.5 Yapay Sinir Ağları ile Neler Yapılabilir?

Yapay sinir ağları günümüzde geliştirilmiş en güncel ve en mükemmel örüntü (gözlenebilir veya ölçülebilir bilgi) tanıyıcı ve sınıflandırıcılardan sayılabilirler. Bu ağları bu kadar güncel yapan da, eksik bilgilerle çalışabilme ve normal olmayan verileri işleyebilme yetenekleridir. Özellikle çok sayıda veriyi işleme gerektiren (radar verileri gibi) işlerde çok avantajlı sonuçlar üretebilmektedirler. Günümüzde birçok problem aslında şekil tanıma problemi haline getirilmekte ve ondan sonra çözülmektedir. Bu nedenle yapay sinir ağlarının kullanılabilceği birçok alan vardır. Endüstriyel ve sosyal hayatta görülen binlerce örnek ile başarılı oldukları gösterilmiştir. Fakat her problemi yapay sinir ağı ile çözmek mantıklı olmayabilir. Eğer herhangi bir problemin çözümü için yeterli etkinlikte ve verimlilikte çözüm yöntemi söz konusu ise yapay sinir ağı kullanılmasının bir anlamı yoktur. İlgili olay hakkında örneklerin olmayışı da bu ağları kullanmamak için önemli bir nedendir. Bir problemin yapay sinir ağı ile çözülmesi için şu şartlardan birini sağlaması gerekir.

- Sadece yapay sinir ağları ile problemlere pratik çözümler üretebilme durumunun söz konusu olması gerekir.
- Başka çözüm yolları olmasına rağmen yapay sinir ağlarının daha kolay ve etkin çözümler üretebilmesinin sağlanması gerekir.

Başarılı uygulamalar incelendiğinde yapay sinir ağlarının, doğrusal olmayan, çok boyutlu, gürültülü, karmaşık, kesin olmayan, eksik, kusurlu, hata olasılığı yüksek sensör verilerinin olması ve problemin çözümü için özellikle bir matematik modelin ve algoritmanın bulunmaması hallerinde yaygın olarak kullanıldıkları görülmektedir.

Bu amaçla geliştirilmiş ağlar genel olarak şu fonksiyonları yerine getirmektedir:

- Olasılık fonksiyon kestirimleri
- Sınıflandırma
- İlişkilendirme veya örüntü eşleştirme
- Zaman serileri analizleri
- Sinyal filtreleme
- Veri sıkıştırma
- Örüntü tanıma
- Doğrusal olmayan sinyal işleme
- Doğrusal olmayan sistem modelleme
- Optimizasyon
- Zeki ve doğrusal olmayan kontrol

Yukarıda listelenen konularda teorik uygulamaların ötesinde günlük hayatta kullanılan finansal konulardan mühendisliğe ve tıp bilimine kadar birçok uygulamadan bahsetmek mümkündür. Bunlardan bazıları şöyle sıralanabilir:

- Veri madenciliği
- Optik karakter tanıma ve çek okuma
- Bankalarda kredi kartı isteyen müracaatları değerlendirme
- Ürünün pazardaki performansını tahmin etme
- Kredi kartı hilelerini saptama
- Zeki araçlar ve robotlar için optimum rota belirleme
- Güvenlik sistemlerinde konuşma ve parmak izi tanıma
- Robot hareket mekanizmalarının kontrol edilmesi
- Mekanik parçaların ömürlerinin ve kırılmalarının tahmin edilmesi
- Kalite kontrolü
- İş çizelgeleme ve iş sıralaması

- İletişim kanallarındaki geçersiz ekoların filtrelenmesi
- İletişim kanallarındaki trafik yoğunluğunu kontrol etme ve anahtarlama
- Radar ve sonar sinyalleri sınıflandırma
- Üretim planlama ve çizelgeleme
- Kan hücreleri reaksiyonları ve kan analizlerini sınıflandırma
- Kanserin saptanması ve kalp krizlerinin tedavisi
- Beyin modellemesi çalışmaları

Bunların çoğaltılması mümkündür. Yukarıdakiler yalnızca genel olarak hangi alanlarda kullanılabileceklerini göstermek amacıyla verilmiştir. Yoksa hemen hemen her alanda örneklerini görmek mümkündür. Çünkü gerçek hayatta kullanılan sistemlerin çoğu doğrusal olmayan modellemeler gerektirmektedir. Bu ise geleneksel yöntemler ile çözüm üretilmesini zorlaştırmakta bazen de imkansızlaştırmaktadır [2].

2.6 Yapay Sinir Ağlarının Mühendislik Uygulamaları

YSA uygulamaları temel olarak tahmin, sınıflandırma, veri yorumlama, veri filtreleme ve veri ilişkilendirme olmak üzere beş sınıfa ayrıştırılabilir. Bunlar:

- **Tahmin (prediction):** Uygulanan giriş değerlerinde bazı çıkış değerleri bulmaya çalışılır. Hava durumu tahmini, kanser riskini belirleme bunlara örnek olarak verilebilir.
- **Sınıflandırma (classification):** Giriş değerleri kullanılarak sınıflandırma yapılır. Arıza sınıflandırma, karakter tanıma, hastalık teşhis etme bunlara örnek olarak verilebilir.
- **Veri ilişkilendirme (data association):** Bu sınıflandırmaya benzer bir yaklaşım gibi görünse de buna ilave olarak hatalı olan verileri tanımlar. Örnek olarak taranan bir dokümandaki karakterleri algılamanın yanında tarayıcının düzgün olarak çalışmadığını da algılayabilir.
- **Veri yorumlama (data conceptualization):** Giriş verisindeki gruplar arasındaki ilişkileri analiz etme işlemidir. Bir veri tabanındaki birbirine benzeyen verileri gruplandırma buna örnek olarak verilebilir.

- **Veri filtreleme (data filtering):** Giriş sinyalleri içerisindeki uygun olmayan verilerin ayıklanmasını sağlar. Bir telefon sinyali içerisinde bulunan gürültüyü ayıklama, bir resim üzerindeki istenilmeyen parazitleri temizleme buna örnek olarak verilebilir.

YSA'lar birçok mühendislik problemlerinin çözümünde kullanılmışlardır. Aşağıda kimya mühendisliği, inşaat mühendisliği, elektrik ve elektronik mühendisliği, imalat ve makine mühendisliği, sistem ve kontrol mühendisliği alanlarında bulunan uygulamalardan bazıları verilmiştir.

2.6.1 Kimya mühendisliği uygulamaları

Kimyasal reaktör seçiminde, dinamik işlemlerde hata belirlenmesinde, endüstriyel polimerasyonda eritme akış indisi tahmininde, endüstriyel mayalama işleminin modellenmesinde, biyokimyasal işlemlerde mikrobik konsantrasyonların tahmininde uygulanmıştır.

2.6.2 İnşaat mühendisliği uygulamaları

Konstrüksiyon projelerinde kaynak seviyelerini belirlemede, rezervuar çıkış kontrolünde, biyolojik bilgiler yardımıyla nehir suyu kalitesinin sınıflandırılmasında, nehir akışının tahmin edilmesinde, sonlu-eleman-temelli yapısal analiz işleminin modellenmesinde, yapı malzemelerinin iç yapılarındaki çatlakların tespit edilmesinde, depreme maruz betonarme çerçevelerde emniyetli yatay taşıyıcı tahmininde uygulanmıştır.

2.6.3 Elektrik ve elektronik mühendisliği uygulamaları

Hastaların alarmla kontrolünde, resimlerden gürültüleri temizlemede, görüntülerin sıkıştırılmasında, gürültü filtreleme ve resimlerdeki kenar bilgisinin çıkarılmasında, güç sisteminde harmoniklerin tahmininde, gezgin haberleşme sisteminde kanal dağıtımında, ultrasonik müziklerden cisimlerin sınıflandırılmasında, optik okuyucu sistemler için resimlerin ön işleme alanlarında, anten dizi elemanlarından elde edilen işaretler arasındaki faz farklılıklarının karşılaştırılmasıyla radar izlemede, mikrodalga parlaklığına bağlı jeofiziksel parametrelerin tayininde, uzaktan kontrollü görüntü sınıflama işleminde, anten dizi analizleri ve tasarımında, elektromagnetik işaretlerin etkileşim modellerini oluşturmada, mikroşerit hatların tasarımı ve analizlerinde kullanılmıştır.

2.6.4 İmalat ve makine mühendisliği uygulamaları

Metal kesme tezgahının kontrolünde, akustik salınım ve iş parçası kuvvetiyle iş parçası yatağının kontrolünde, güç tüketimi ve iş parçası ivmesi, hücreli imalat için grup teknolojisi parça gruplarının tasarımında, hareket eden nesnelere için engelsiz yol planlamasında, makine parametrelerinin optimizasyonunda, makine arızalarının sınıflandırmasında, malzemelerin ısı transferinin belirlenmesinde, uçak kanat kutu yapılarının tasarımında kullanılmıştır.

2.6.5 Sistem ve kontrol mühendisliği uygulamaları

Esnek kollu robotun kontrolünde, bir model helikopterin havada kontrolü, çok değişkenli (mafsallı) robotun yörünge koordinasyonunda, iki sıvı tank sisteminin akış seviye kontrolünde, anestezi derinlik ölçümü ve kontrolü, bir robot için optimal yolun bulunmasında, banyo suyu sıcaklığının kontrolünde, endüstriyel robot kontrolünde, sistem kimliklendirme gibi birçok alanda uygulanmıştır.

2.6.6 Harita mühendisliği uygulamaları

Görüntü sınıflandırmada, fotogrametrik veya uzaktan algılama verilerinin yorumlanmasında, görüntü ortamında, örnek eşlemede, 3D yüzey hesaplamalarında, gerçekçi görüntüler için ışın izleme algoritmalarının geliştirilmesinde kullanılmıştır.

2.6.7 Bilgisayar mühendisliği uygulamaları

Görüntü işleme, sınıflandırma, filtreleme ve yorumlamada, sistem modelleme, kimliklendirme ve kontrol işlemlerinde istatistiksel proses kontrol uygulamalarında, ağ güvenliği, şifreleme ve şifre çözmede, sinyal işlemede kullanılmıştır [3].

2.7 Yapay Sinir Ağlarının Geleceği

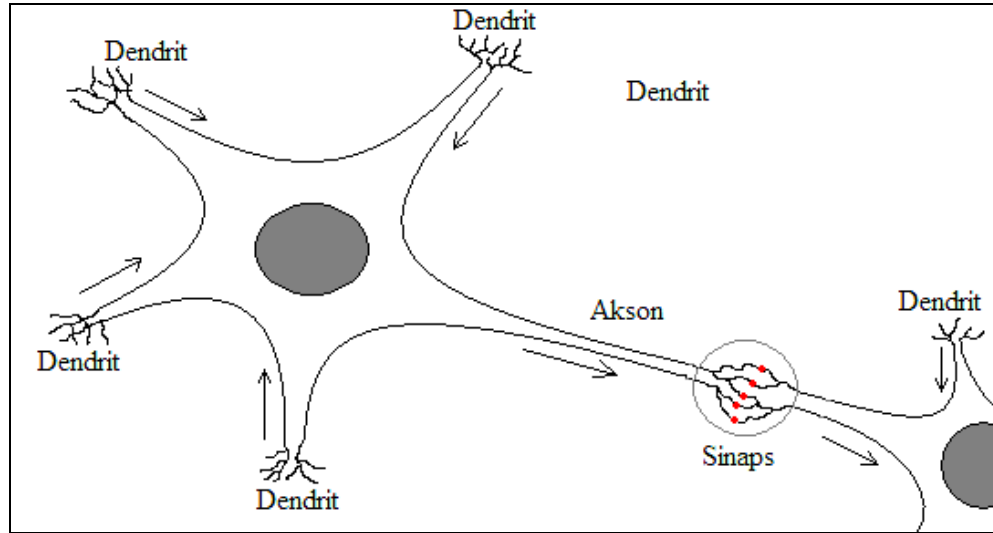
Bugün, tüm dünyada sinir ağları ile ilgili birçok araştırma yapılmaktadır. Yapay sinir ağlarını eğitmek uzun süreler aldığından başlıca araştırmalar bu konuya odaklanmış görünmektedir. Burada amaç yeni ve daha verimli öğrenme algoritmaları, zamana bağlı olarak değişen modellere karşılık verebilen ağlar ve silikon sinir ağları geliştirmek şeklinde özetlenebilir.

Yapay sinir ağlarını kullanıma sunmak için özel yongalar gerekmektedir. Edinburg Üniversitesinde bir sinir ağı yongası geliştirilmiştir. Özel bazı şirketler sayısal, analog ve optik olmak üzere üç tip sinir yongası üzerinde çalışmaktadırlar. Bazı şirketler, özel derlenmiş devreler (ASIC) ile bir sinir ağı uygulaması yaratmak için silikon malzemeler üzerinde çalışmaktadırlar. ASIC'ların ve sinire benzer sayısal yongaların yakın gelecekte büyük bir başarıyla kullanılacağı gözükmektedir. Son olarak optik yongalar çok ümit verici gözükmektedir.

2.8 Biyolojik Bir Beyin Sinir Hücresinin Yapısı

Yapay sinir ağlarının daha iyi anlaşılması için örnek teşkil eden doğal sinir ağı olan biyolojik sinir ağlarının yapısının ve çalışma ilkelerinin iyi bilinmesi gerekmektedir.

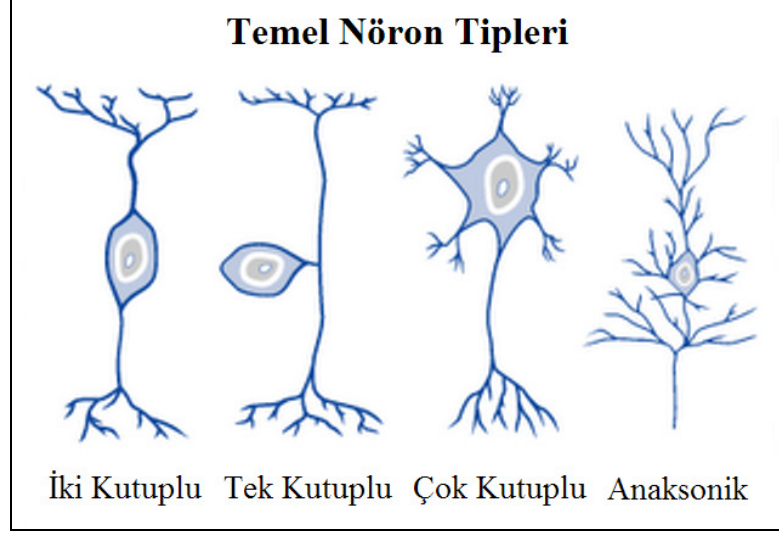
İnsan beyninin en temel parçası, hatırlama, düşünme, her harekette daha önceki deneyimlere başvurma yeteneğini sağlayan kendine özgü sinir hücreleridir. Beynimizde 10^{10} adet sinir hücresi ve bunların da 6×10^{13} 'ten fazla sayıda bağlantısının olduğu söylenmektedir [2]. İnsan beyninin çalışma frekansı 100Hz'dir, ağırlığı yetişkin bir insanda yaklaşık 1,3 kg'dır. Şekil 2.2'de basitleştirilmiş biyolojik bir beyin sinir hücresi gösterilmiştir.



Şekil 2.2: Basitleştirilmiş biyolojik bir beyin sinir hücresi

Sinir hücreleri kabaca çok kutuplu, iki kutuplu, tek kutuplu ve anaksonik olmak üzere gruplandırılabilirler (Şekil 2.3). Bizim örnek aldığımız sinir çok kutuplu tiptedir. Yani çok sayıda dendriti ve tek bir aksonu vardır. Bu sinirlerin hücre gövdeleri

omurilik (medulla sprialis)'in ön boynuzunda bulunur ve miyelin tabaka ile çevrelenmiştir. Miyelin tabaka stoplazma ile hücreler arası sıvı arasındaki kapasiteyi düşürerek, işaretlerin yayılma hızını arttırmaya yarayan yalıtım maddesidir. Miyelinli sinirlerle iletim, miyelinsizlere göre oldukça hızlıdır.



Şekil 2.3: Temel nöron tipleri [4]

Ayrıca miyelin tabaka ile kaplı akson üzerinde her birkaç mm'de bir yer alan ve işaretleri periyodik olarak yeniden üretmeye yarayan ranvier adı verilen boğumlar vardır. Bununla birlikte, tüm doğal sinirlerin dört temel bileşeni vardır. Bunlar, dendritler, soma, akson ve sinaps'tır. Temel olarak biyolojik sinir diğer kaynaklardan girişleri alır, soma girişleri genelde doğrusal olmayan bir şekilde işler. Akson işlenmiş girişleri çıkışı aktarır. Sinaps çıkışı diğer sinirlere gönderir.

Bir sinirden diğer bir sinapsa işaretlerin taşınması karmaşık bir kimyasal süreçtir. Burada özel verici maddeler, birleşmeden sonra giden son işaretle serbest kalır. Alınan işaret hücrenin içindeki elektrik potansiyelinin daha altındaysa etki ortaya çıkar. Eğer potansiyel bir eşik değerine ulaşırsa akson içine bir işaret gönderir. Buna hücrenin uyarılması denir.

Sinirin işlevi sinaps boyunca dendrite ulaşan uyarıları alarak, bunlardan aksiyon potansiyeli oluşturma veya oluşturmamaktır. Aksiyon potansiyeli genel olarak 0,1 mV genliğinde 1 milisaniyelik elektriksel nabzı alır ve akson boyunca 120 m/s hızda hareket eder.

Hareket potansiyeli, akson ucundaki sinapsa ulaştığında, elektriksel uyarıdan kimyasal uyarıya dönüşerek, sinaps aralığından postsinaptik sinire geçer. Postsinaptik

sinirin zarında kimyasal uyarı elektriksel uyarıya dönüştürülür ve dendritler boyunca somaya ulaştırılır. Bir sinaps ya uyarıcı ya da inhibe edici olabilir. Uyarıcı sinapslar uyarıcı sinir aksiyon potansiyelini artırır, inhibe edici sinapslar ise uyarıcı sinir aksiyon potansiyelini azaltır.

Bir sinir hücresinin yüzey zarının iç ve dış tarafında potansiyel fark vardır. Bitişik sinirin sinapsından gönderilen nörotransmitter kimyasal işaretler sinirin zar potansiyelini ayarlar. Sinapslar, darbe-kodlu elektriksel sinirsel işaretleri nörotransmitter işaretlere dönüştürür.

Her aksonun üzerinden geçen işaretleri değerlendirebilecek yetenekte olduğu varsayılmaktadır. Bu değerlendirme bir işaretin belli bir sinir için ne kadar önemli olduğunu göstermektedir.

Etkinlik " $x(t)$ " ve işaretin " $S(x(t))$ " nörofizyolojik yorumu potansiyel farkın elektriksel darbelerini ve zamansal toplamlarını içermektedir. Etkinlikler küçük zar darbelerini içerir. İşaretler ise geniş aksonal darbeleri veya elektrik potansiyellerini içerir.

Matematiksel olarak gerçek değerli etkinlik ($x(t)$) sinirinin yüzey zarında t zamandaki potansiyel farkını (gerilim) temsil eder. Etkinlik pozitif veya negatif olabildiği gibi teorik olarak sonsuz da olabilir [1].

2.9 Bir Yapay Sinir Ağının Ana Öğeleri

Yapay sinir ağları, birbirine bağlı çok sayıda işlem elemanından oluşmuş, genellikle paralel işleyen yapılar olarak adlandırılabilir. Yapay sinir ağlarındaki işlem elemanları (düğümler) basit sinirler olarak adlandırılır. Bir yapay sinir ağı, birbirleriyle bağlantılı, çok sayıda düğümlerden oluşur.

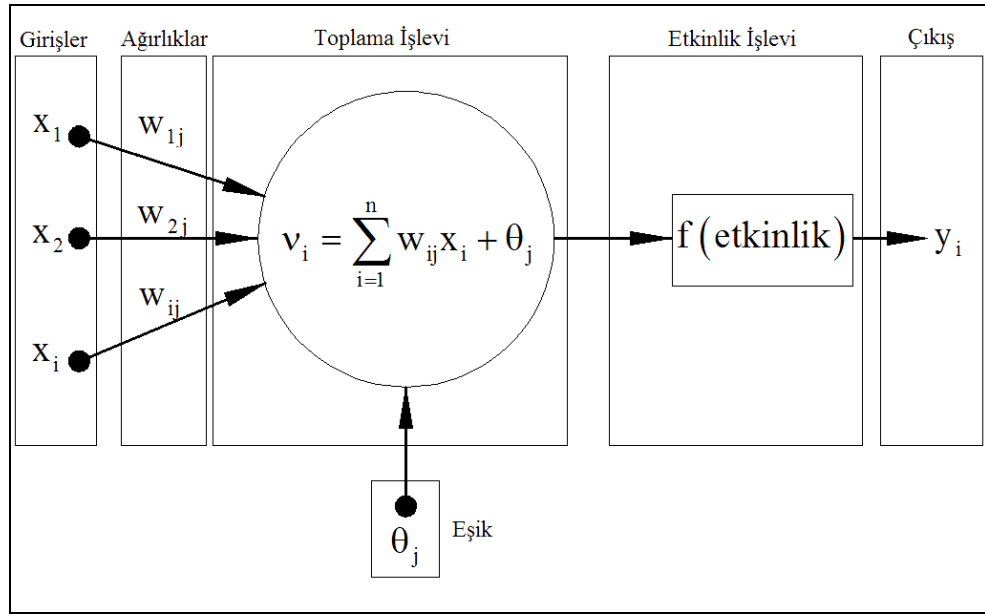
Yapay sinir ağları, insan beyni gibi öğrenme, hatırlama ve genelleme yeteneğine sahiptirler.

İnsan beyninde öğrenme üç şekilde olur;

- Yeni aksonlar üreterek
- Aksonların uyarılmasıyla
- Mevcut aksonların güçlerini değiştirerek.

Her aksonun, üzerinden geçen işaretleri değerlendirebilecek yetenekte olduğu savunulmaktadır. Aksonun bu özelliği, bir işaretin belli bir sinir için ne kadar önemli olduğunu göstermektedir.

Yapay sinir ağlarının temel birimi işlem elemanı ya da düğüm olarak adlandırılan yapay bir sinirdir. Bir yapay sinir, biyolojik sinirlere göre daha basit olmasına karşın, biyolojik sinirlerin dört temel işlevini taklit ederler. Şekil 2.4'te yapay bir sinir (düğüm) gösterilmiştir.



Şekil 2.4: Yapay bir sinir [1]

Girişler x_i sembolüyle gösterilmiştir. Bu girişlerin her biri ağırlık w ile çarpılır. Basitçe, bu ürünler eşik değeri θ_j ile toplanır ve sonucu oluşturmak için etkinlik işlevi ile işlem yapılır ve y_i çıkışı alınır.

Tüm yapay sinir ağları bu temel yapıdan türetilmiştir. Bu yapıdaki farklılıklar yapay sinir ağlarının farklı sınıflandırılmalarını sağlar. Bir yapay sinirin öğrenme yeteneği, seçilen öğrenme algoritması içerisinde ağırlıkların uygun bir şekilde ayarlanmasına bağlıdır [1].

2.9.1 Girişler

Girişler (x_1, x_2, \dots, x_n) çevreden aldığı bilgiyi sinire getirir. Girişler kendinden önceki sinirden veya dış dünyadan sinir ağına gelebilir. Bir sinir genellikle gelişigüzel birçok girdileri alır [1].

2.9.2 Ağırlıklar

Ağırlıklar (w_1, w_2, \dots, w_i) , yapay sinir ağı tarafından alınan girişlerin sinir üzerindeki etkisini belirleyen uygun katsayılarıdır. Her bir giriş kendine ait bir ağırlığa sahiptir.

Bir ağırlık değerinin büyük olması, o girişin yapay sinire güçlü bağlanması ya da önemli olması, ağırlık değerinin küçük olması zayıf bağlanması ya da önemli olmaması anlamına gelmektedir [1].

2.9.3 Toplama işlevi

Toplama işlevi v_i , sinirde her bir ağırlığı ait olduğu girişlerle çarpımının toplamlarını eşik θ_j değeri ile toplayarak etkinlik işlevine gönderir. Bazı durumlarda toplama işlevi bu kadar basit bir işlem yerine, en az (min), en çok (max), çoğunluk veya birkaç normalleştirme algoritması gibi çok daha karmaşık olabilir [1]. Tablo 2.2'de bazı toplama fonksiyonu örnekleri gösterilmiştir [2].

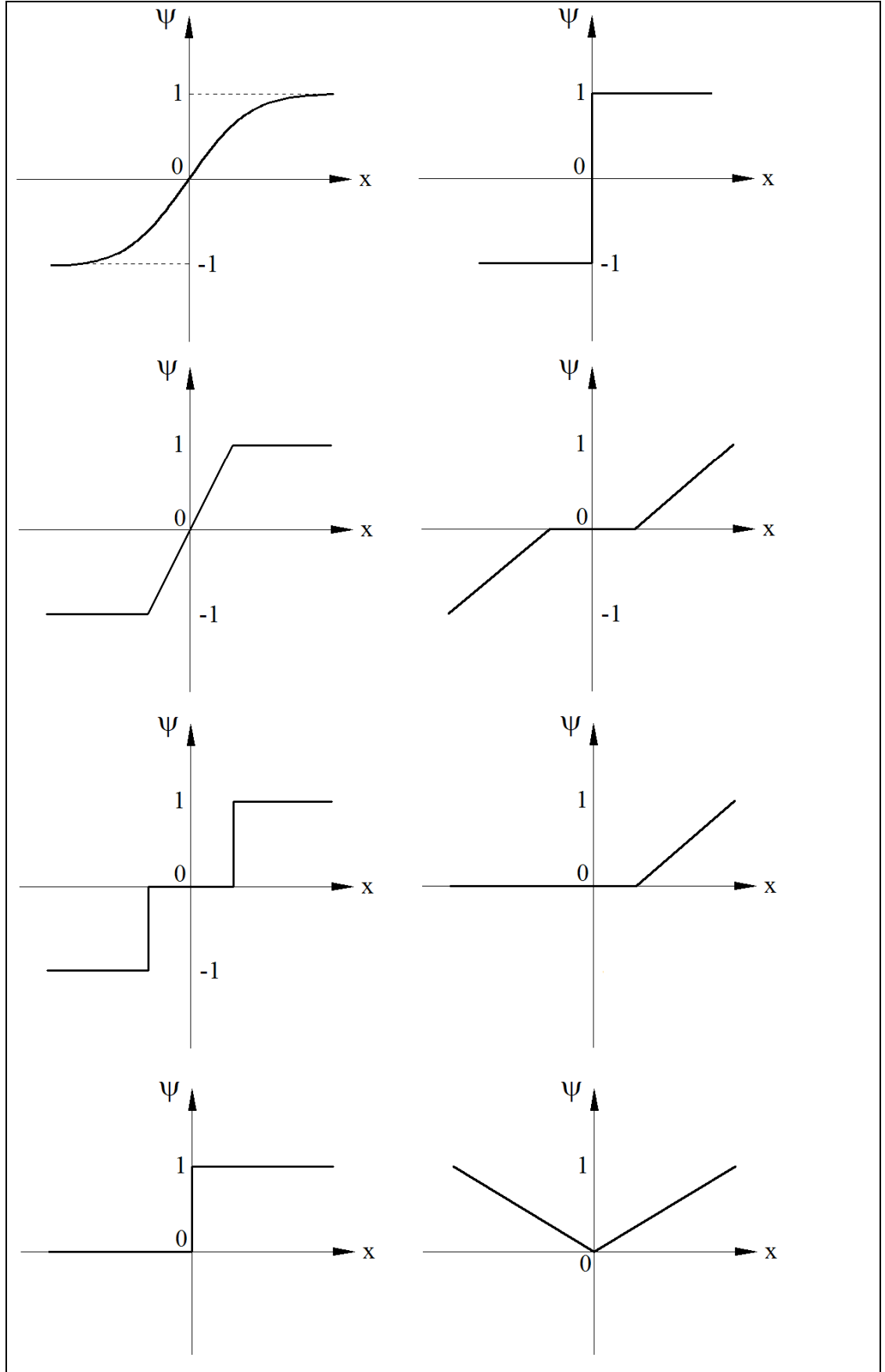
Tablo 2.2: Bazı toplama fonksiyonu örnekleri [2]

Net Giriş	Açıklama
<p>Çarpım:</p> $\text{Net Girdi} = \prod_{i=1}^n w_{ij} x_i$	Ağırlık değerleri girdiler ile çarpılır ve daha sonra bulunan değerler birbirleri ile çarpılarak net girdi hesaplanır.
<p>Maksimum:</p> $\text{Net Girdi} = \text{Max}(w_{ij} x_i), i = 1, \dots, n$	n adet girdi içinden ağırlıklar ile çarpıldıktan sonra en büyüğü yapay sinir hücresinin net girdisi olarak kabul edilir.
<p>Minimum:</p> $\text{Net Girdi} = \text{Min}(w_{ij} x_i), i = 1, \dots, n$	n adet girdi içinden ağırlıklar ile çarpıldıktan sonra en küçüğü yapay sinir hücresinin net girdisi olarak kabul edilir.
<p>Çoğunluk:</p> $\text{Net Girdi} = \sum_{i=1}^n \text{sgn}(w_{ij} x_i)$	n adet girdi içinden ağırlıklar ile çarpıldıktan sonra pozitif ve negatif olanların sayısı bulunur. Büyük olan sayı hücrenin net girdisi olarak kabul edilir.
<p>Kümülatif Toplam:</p> $\text{Net Girdi} = \text{Net}(\text{eski}) \sum_{i=1}^n (w_{ij} x_i)$	Hücreye gelen bilgiler ağırlıklı olarak toplanır ve daha önce gelen bilgilere eklenerek hücrenin net girdisi bulunur.

2.9.4 Etkinlik işlevi

Toplama işlevinin sonucu, etkinlik işlevinden f (etkinlik) geçirilip çıkışa iletilir. Bir etkinlik işlevinin kullanım amacı, zaman söz konusu olduğunda toplama işlevinin çıkışının değişmesine izin vermektir. Etkinlik işlevi, Şekil 2.5'te gösterilen eğrilerin biri şeklinde olabilmektedir.

Bu eğrilerden eşik veya basamak işlevleri etkinlik işlevinin nasıl çalıştığını basit bir şekilde açıklamaktadır. Sinir, etkinlik işlevinin eşik seviyesinin altında çıkış üretmez. Sinir, etkinlik seviyesinin üzerinde çıkış üretir.



Şekil 2.5: Bazı etkinlik işlevi örnekleri [1]

Etkinlik işlevinin çıkışı y_i , giriş vektörleri x_i tarafından uyarıldığında, Eşitlik 2.1'deki gibi tanımlanır.

$$y_i = \begin{cases} 1 & \text{eğer } w_1x_1 + w_2x_2 + \dots + w_nx_n \geq T \\ 0 & \text{eğer } w_1x_1 + w_2x_2 + \dots + w_nx_n < T \end{cases} \quad (2.1)$$

İkili girişlerin bir örneği verildiğinde, etkinlik işlevi 0 ya da 1 çıkış verecektir.

2.9.5 Ölçekleme ve sınırlama

Düğümde, etkinlik işlevinin sonuçları ölçek veya sınır işlemlerinden geçebilir. Bu ölçeklendirme basitçe bir ölçek etmeni ile etkinlik değerinin çarpımının sonucudur. Sınırlandırma ise, ölçeklenmiş sonuçların en az ve en çok sınırlarını aşmamasını sağlamaktır [1].

2.9.6 Çıkış işlevi

Çıkış $y_i = f(s)$, etkinlik işlevi sonucunda dış dünyaya veya diğer sınırlara gönderildiği yerdir. Bir sinirin bir tek çıkışı vardır. Sinirin bu çıkışı, kendinden sonra gelen herhangi bir sayıdaki diğer sınırlara giriş olabilir.

Her bir düğümde bir çıkış işaretine izin verilir. Bu işaret diğer yüzlerce sinir hücrelerinin girişi olabilir. Bu durum biyolojik sinirde olduğu gibidir. Biyolojik sinirde de birçok giriş varken sadece bir çıkış etkinliği vardır. Düğüm çıkışı etkinlik işlevinin sonucuna eşdeğerdir. Fakat bazı ağ yapıları, komşu düğümler arasında yarışma oluşturmak için etkinlik sonuçlarını düzenleyebilir. Böylece yarışmacı girişler hangi düğümün öğrenme ya da uyma işlemine katılacağına karar vermesinde yardımcı olur.

Bütün bu anlatılanların ışığında yapay sinir ile biyolojik sinirler arasındaki benzerlik Tablo 2.3'deki gibi gösterilebilir [1].

Tablo 2.3: Yapay sinir ile biyolojik sinirler arasındaki benzerlikler [1]

Biyolojik Sinir	Yapay Sinir
Sinir Sistemi	Sinirsel Hesaplama Sistemi
Sinir	Düğüm (Sinir, İşlem Elemanı)
Sinaps	Sinirler Arası Bağlantı Ağırlıkları
Dendrit	Toplama İşlevi
Hücre Gövdesi	Etkinlik İşlevi
Akson	Sinir Çıkışı

2.9.7 Öğrenme

Öğrenme kuralı Hebbian öğrenme kuralı denilen basit bir modele dayanır. Hebbian öğrenme kuralı temel olarak “eğer iki düğüm aynı zamanda etkin ise aralarındaki bağ gücü artar” kuramına dayanmaktadır. Öğrenmenin amacı her bir düğüm girişlerindeki değişken bağlantı ağırlıklarını derlemektir. İstenen bazı sonuçları elde etmek için, giriş bağlantılarının ağırlıklarını değiştirme işlemi uyma işlevi olarak adlandırıldığı gibi öğrenme kipi olarak ta adlandırılabilir.

Öğretmenli ve öğretmensiz olmak üzere iki tip öğrenme türü vardır. Öğretmenli öğrenmede bir öğretmene ihtiyaç vardır. Öğretmen, bir veri alıştırma kümesi veya ağ sonuçlarının performansını derecelendiren bir gözlemci olabilir. Öğretmenli öğrenmede eğitilmiş sinirlere öğretme işaretini gönderecek sinirler eğitilir. Bu işaretin bağlantısındaki ağırlıkları ayarlamakta kullanılır. Önemli öğrenme kurallarından bir kaçını aşağıda örnek olarak verilmiştir [1].

- **Hebb Kuralı**

İlk ve en iyi bilinen öğrenme kuralı Donald Hebb tarafından tanıtılmıştır. Tanımlama 1949’da yazdığı “The Organization of Behaviour” adlı kitabında görünür. Temel kural: Eğer bir sinir başka bir sinirden bir giriş alırsa ve her ikisi de yüksek aktif ise (matematiksel olarak aynı işaretli), sinirler arasındaki boyut kuvvetlendirilir.

- **Hopfield Kuralı**

Bu kural, kuvvetlendirme veya zayıflatmanın genliğini belirleyebilmesi istisnası haricinde Hebb kuralıyla benzerdir. Buna göre, “eğer istenilen çıkış ve girişin her ikisi de aktif veya her ikisi de durgun ise, bağlantı boyutlarını öğrenme oranı kadar arttır, aksi halde boyutu öğrenme oranı kadar azalt”, (Öğrenme fonksiyonlarının çoğunun öğrenme oranı veya öğrenme sabiti için bazı koşulları vardır).

- **Delta Kuralı**

En çok kullanılan kurallardan biri olan Delta kuralı, Hebb kuralının daha geliştirilmiştir. Bu kural bir sinirin gerçek çıkışı ile istenilen çıkış değeri arasındaki farkı azaltmak için giriş bağlantı güçlerinin sürekli olarak geliştirme fikrine dayanır. Bu kural ağ hatasının karesini minimize etmek için bağlantı boyutlarını değiştirir. Hata bir önceki katmana geri çoğaltılır. Her bir zaman dilimi için bir hata şeklinde bu geri çoğaltma işlemi ilk katmana ulaşınca kadar devam eder. Bu tip ağ “İleri Beslemeli Ağ” olarak isimlendirilir. Geri yayılım adını bu hata terimlerini toplama yönteminden türetir.

Bu kural ayrıca Windrow-Hoff öğrenme ve en küçük ortalama kareler (Least Mean Square) kuralı olarak da adlandırılır.

- **Eğimli İniş Kuralı**

Bu kural Delta kuralına benzer. Çünkü transfer fonksiyonunun türevi, bağlantı ağırlıklarına uygulamadan önce, Delta hatası düzeltmek için kullanılır. Bu kural durağan bir noktaya birçok şekilde yaklaşmasına rağmen sıkça kullanılır. Bir ağın farklı katmanları için öğrenme oranları, öğrenme işleminin daha hızlı olmasına yardımcı açıklamıştır. Bu test işleminde çıkışa yakın olan katmanların öğrenme oranından daha düşüktür. Giriş verilerinin güçlü bir modelden çıkarılmadığı uygulamalarda, bu işlem özellikle önemlidir.

- **Kohonen Öğrenme Kuralı**

Kohonen tarafından geliştirilen bu yöntem biyolojik sistemlerdeki öğrenmeden esinlenilmiştir. Bu yöntemde sinirler öğrenmek için elverişli durum veya ölçülerini güncellemek için yarışır. En büyük çıkış ile işlenen sinir, kazanımı ilan eder ve komşularına bağlantı boyutlarını güncellemeleri için izin verilir [1].

3. ÇOK KATMANLI ALGILAYICI YSA MODELİ

Yapay sinir ağlarının ilk modellerinin en temel özellikleri, doğrusal olan olayları çözebilme yeteneklerine sahip olmalarıdır. Bu ağlar ile doğrusal olmayan ilişkiler öğrenilememektedir. Bu sorunu çözmek için çok katmanlı algılayıcılar geliştirilmiştir [2].

3.1 Çok Katmanlı Algılayıcı (ÇKA)

Bir yapay sinir ağının öğrenmesi istenen olayların girdi ve çıktıları arasındaki ilişkiler doğrusal olmayan ilişkiler olması durumunda daha gelişmiş modellere ihtiyaç vardır. Çok katmanlı algılayıcı modeli bunlardan birisidir. Olayın doğrusal olup olmaması ne demektir? Bu konuyu iyi anlayabilmek için ünlü XOR problemine bakmak gerekir. Bu problemin özelliği, doğrusal olmayan bir ilişkiyi göstermesidir. Yani çıktıların arasına bir doğru veya doğrular çizerek onları iki veya daha fazla sınıfa ayırmak mümkün değildir. Basit algılayıcı ve ADALINE ile bu problemi çözmek mümkün olmamıştır.

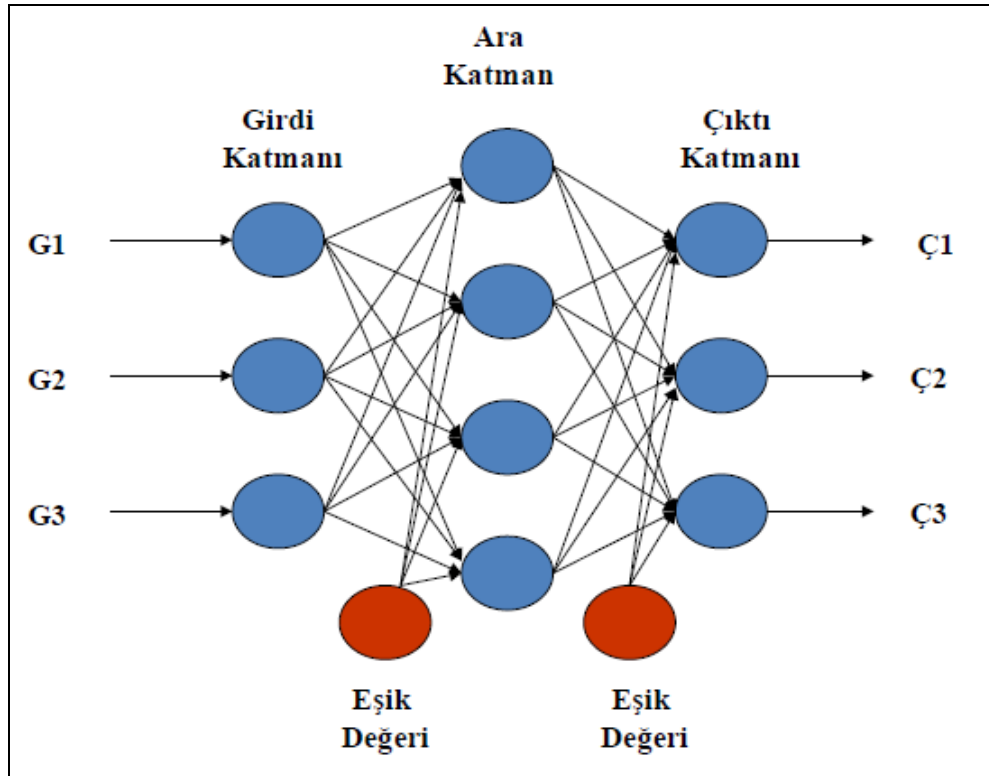
Bu problem hemen hemen yapay sinir ağının anlatıldığı her kitapta örnek olarak verilmektedir. Bundan çok yaygın olarak bahsedilmesinin nedeni şöyle açıklanabilir. Minsky özellikle basit algılayıcı modelinin bu probleme çözüm üretemediğini göstermiş ve yapay sinir ağlarının doğrusal olmayan problemlere çözüm üretemediğini iddia ederek bilimsel araştırmaların durmasına neden olmuştur. Çünkü günlük olayların çoğu (hemen hemen hepsi) doğrusal olmayan bir nitelik taşımaktadır. XOR probleminin çözülememesinden sonra neredeyse bütün çalışmalar durmuş, sadece birkaç araştırmacı çalışmalara devam etmiştir. Bu problemi çözerek yapay sinir ağlarına tekrar dikkatleri çekmeyi başarmışlardır. O nedenle bu problem yapay sinir ağı araştırmalarında bir kilometre taşı olarak görülmektedir.

XOR problemini çözmek amacı ile yapılan çalışmalar sonucu çok katmanlı algılayıcı modeli (ÇKA) geliştirilmiştir. Rumelhart ve arkadaşları tarafından geliştirilen bu modele hata yayma modeli veya geriye yayım modeli (backpropagation network) de denmektedir. ÇKA modeli yapay sinir ağlarına olan ilgiyi çok hızlı bir

şekilde arttırmış ve yapay sinir ağları tarihinde yeni bir dönemin başlamasına neden olmuştur. Bu model günümüzde mühendislik problemlerinin hemen hemen hepsine çözümler üretebilecek güce sahiptir. Özellikle sınıflandırma, tanıma ve genelleme yapmayı gerektiren problemler için çok önemli bir çözüm aracıdır. Bu model Delta Öğrenme Kuralı denilen bir öğrenme yöntemini kullanmaktadır. Bu kural aslında ADALINE ve basit algılayıcı modellerin öğrenme kurallarının geliştirilmiş bir şeklidir. Temel amacı ağın beklenen çıktısı ile ürettiği çıktı arasındaki hatayı en aza indirmektir. Bunu hatayı ağa yayarak gerçekleştirdiği için bu ağa hata yayma ağında denmektedir [2].

3.2 ÇKA Modelinin Yapısı

ÇKA ağlarının yapısı Şekil 3.1’de gösterildiği gibidir.



Şekil 3.1: ÇKA modelinin yapısı [6]

Şekilde de görüldüğü gibi ÇKA ileriye doğru bağlantılı ve 3 katmandan oluşan bir ağdır. Bunlar:

- **Girdi Katmanı**

Dış dünyadan gelen girdileri (x_1, x_2, \dots, x_n) olarak ara katmana gönderir. Bu katmanda bilgi işleme olmaz. Gelen her bilgi geldiği gibi bir sonraki katmana gider. Birden fazla girdi gelebilir. Her proses elemanının sadece bir tane girdisi ve bir tane çıktısı vardır. Bu çıktı bir sonraki katmanda bulunan bütün proses elemanlarına gönderilir. Yani, girdi katmanındaki her proses elemanı bir sonraki katmanda bulunan proses elemanlarının hepsine bağlıdır.

- **Ara Katmanlar**

Ara katmanlar girdi katmanından gelen bilgileri işleyerek bir sonraki katmana gönderir. Bir ÇKA ağında birden fazla ara katman ve her katmanda birden fazla proses elemanı olabilir. Ara katmandaki her proses elemanı bir sonraki katmandaki bütün proses elemanlarına bağlıdır.

- **Çıktı Katmanı**

Ara katmandan gelen bilgileri işleyerek ağa girdi katmanından verilen girdilere karşılık ağın ürettiği çıktıları (y_1, y_2, \dots, y_n) belirleyerek dış dünyaya gönderir. Bir çıktı katmanında birden fazla proses elemanı olabilir. Her proses elemanı bir önceki katmanda bulunan bütün proses elemanlarına bağlıdır. Her proses elemanının sadece bir tane çıktısı vardır.

ÇKA ağında bilgiler girdi katmanından ağa sunulur ve ara katmanlardan geçerek çıktı katmanına gider ve ağa sunulan girdilere karşılık gelen ağın cevabı dış dünyaya iletilir.

ÇKA ağı öğretmenli öğrenme stratejisini kullanır. Ağa, hem örnekler hem de örneklerden elde edilmesi gereken çıktılar (beklenen çıktı) verilmektedir. Ağ kendisine gösterilen örneklerden genellemeler yaparak problem uzayını temsil eden bir çözüm uzayı üretmektedir. Daha sonra gösterilen benzer örnekler için bu çözüm uzayı sonuçlar ve çözümler üretebilmektedir [2].

3.3 ÇKA Ağının Öğrenme Kuralı

ÇKA ağları öğretmenli öğrenme stratejisine göre çalışırlar. Yani; bu ağlara eğitim sırasında hem girdiler hem de o girdilere karşılık üretilmesi gereken (beklenen) çıktı değerleri gösterilir. Ağın görevi her girdi için o girdiye karşılık gelen çıktıyı üretmektir. ÇKA ağının öğrenme kuralı en küçük kareler yöntemine dayalı Delta Öğrenme Kuralının genelleştirilmiş halidir. O nedenle öğrenme kuralına Genelleştirilmiş Delta Kuralı da denmektedir. Ağın öğrenebilmesi için eğitim seti adı verilen ve örneklerden oluşan bir sete ihtiyaç vardır. Bu set içinde her örnek için ağın hem girdiler hem de o girdiler için ağın üretmesi gereken çıktılar belirlenmiştir. Genelleştirilmiş “Delta Kuralı” iki safhadan oluşur.

- Safha 1 – ileri doğru hesaplama: Ağın çıktısını hesaplama safhasıdır.
- Safha 2 – geriye doğru hesaplama: Ağırlıkları değiştirme safhasıdır.

3.3.1 İleri doğru hesaplama

Bu safhada bilgi işleme eğitim setindeki bir örneğin girdi katmanından (x_1, x_2, \dots, x_n) ağa gösterilmesi ile başlar. Daha önce belirtildiği gibi, girdi katmanında herhangi bir bilgi işleme olmaz. Gelen girdiler hiçbir değişiklik olmadan ara katmana gönderilir. Yani girdi katmanındaki k. proses elemanının çıktısı y_k^i Eşitlik (3.1) olarak belirlenir.

$$x_k^i = y_k^i \quad (3.1)$$

Ara katmandaki her proses elemanı girdi katmanındaki bütün proses elemanlarından gelen bilgileri bağlantı ağırlıklarının (w_1, w_2, \dots) etkisi ile alır. Önce ara katmandaki proses elemanlarına gelen net girdi (NET_j^a) Eşitlik (3.2) kullanılarak hesaplanır.

$$NET_j^a = \sum_{k=1}^n w_{kj} y_k^i \quad (3.2)$$

Burada w_{kj} k. girdi katmanı elemanını j. ara katman elemanına bağlayan bağlantısının ağırlık değerini göstermektedir. j. ara katman elemanının çıktısı ise bu net girdinin aktivasyon fonksiyonundan (genellikle sigmoid fonksiyonundan) geçirilmesiyle hesaplanır. Uygulamada genellikle bu fonksiyon kullanılmakla beraber,

kullanılması zorunlu değildir. Önemli olan burada türevi alınabilir bir fonksiyon kullanmaktır. Diğer aktivasyon fonksiyonlarından herhangi birisini burada kullanmak mümkündür. Yalnız geriye doğru hesaplamada burada kullanılan fonksiyonun türevinin alınacağını unutmamak gerekir. Sigmoid fonksiyonunun kullanılması halinde çıktı, Eşitlik (3.3)'e göre hesaplanır.

$$y_k^i = \frac{1}{1 + e^{-(NET_j^a + \beta_j^a)}} \quad (3.3)$$

Burada β_j , ara katmanda bulunan j. elemana bağlanan eşik değer elemanının ağırlığını göstermektedir. Bu eşik değeri ünitesinin çıktısı sabit olup 1'e eşittir. Ağırlık değeri ise sigmoid fonksiyonunun oryantasyonunu belirlemek üzere konulmuştur. Eğitim esnasında ağ bu değeri kendisi belirlemektedir.

Ara katmanın bütün proses elemanları ve çıktı katman proses elemanlarının çıktıları aynı şekilde kendilerine gelen NET girdinin hesaplanması ve sigmoid fonksiyonundan geçirilmesi sonucu belirlenirler. Çıktı katmanından çıkan değerleri, yani çıktıları (y_1, y_2, \dots, y_n) bulunca ağırlık ileri hesaplama işlemi tamamlanmış olur [2].

3.3.2 Geriye doğru hesaplama

Ağa sunulan girdi için ağırlık ürettiği çıktı ağırlık beklenen çıktıları (z_1, z_2, \dots) ile karşılaştırılır. Bunların arasındaki fark hata olarak kabul edilir. Amaç bu hatanın düşürülmesidir. O nedenle geriye hesaplamada bu hata ağırlık değerlerine dağıtılarak bir sonraki iterasyonda hatanın azaltılması sağlanır. Çıktı katmanındaki m. proses elemanı için oluşan hata (E_m) , Eşitlik (3.4) kadar olacaktır.

$$E_m = z_m - y_m \quad (3.4)$$

Bu bir proses elemanı için oluşan hatadır. Çıktı katmanı için oluşan toplam hatayı (TE) bulmak için bütün hataların toplanması gerekir. Bazı hata değerleri negatif olacağından toplamın sıfır olmasını önlemek amacı ile ağırlıkların kareleri hesaplanarak sonucun karekökü alınır. ÇKA ağırlığının eğitilmesindeki amaç bu hatayı azaltmaktır. Toplam hata Eşitlik (3.5) ile bulunur.

$$TE = \frac{1}{2} \sum_m E_m^2 \quad (3.5)$$

Toplam hatayı en aza indirmek için bu hatanın kendisine neden olan proses elemanlarına dağıtılması gerekmektedir. Bu ise proses elemanlarının ağırlıklarını değiştirmek demektir. Ağırlıkların değiştirilmesi için iki durum söz konusudur.

- Ara katman ile çıktı katmanı arasındaki ağırlıkların değiştirilmesi
- Ara katmanlar arası veya ara katman girdi katmanı arasındaki ağırlıkların değiştirilmesi

3.3.3 Ara katman ile çıktı katmanı arasındaki ağırlıkların değiştirilmesi

Ara katmandaki j. proses elemanını çıktı katmanındaki m. proses elemanına bağlayan bağlantının ağırlığındaki değişim miktarına Δw^a denirse; herhangi bir t zamanında (t. iterasyonda) ağırlığın değişim miktarı Eşitlik (3.6) ile hesaplanır.

$$\Delta w_{jm}^a(t) = \lambda \delta_m y_j^a + \alpha \Delta w_{jm}^a(t-1) \quad (3.6)$$

Burada λ öğrenme katsayısını, α momentum katsayısını göstermektedir. Öğrenme katsayısı ağırlıkların değişim miktarını, momentum katsayısı ise ÇKA ağırlığının öğrenmesi esnasında yerel bir optimum noktaya takılıp kalmaması için ağırlık değişim değerinin belirli bir oranda bir sonraki değişime eklenmesini sağlar. Bu konu aşağıda tekrar tartışılacaktır. Eşitlikteki δ_m ise m. çıktı ünitesinin hatasını göstermektedir ve Eşitlik (3.7) yardımıyla hesaplanır.

$$\delta_m = f'(NET) \cdot E_m \quad (3.7)$$

Buradaki $f'(NET)$ aktivasyon fonksiyonunun türevidir. Sigmoid fonksiyonunun kullanılması durumunda Eşitlik (3.8) ile hesaplanır.

$$\delta_m = y_m (1 - y_m) \cdot E_m \quad (3.8)$$

Değişim miktarı hesaplandıktan sonra ağırlıkların t. iterasyondaki yeni değerleri Eşitlik (3.9)'a göre hesaplanır.

$$w_{jm}^a(t) = w_{jm}^a(t-1) + \Delta w_{jm}^a(t) \quad (3.9)$$

Benzer şekilde eşik değer ünitesinin de ağırlıklarını değiştirmek gerekmektedir. Onun için öncelikle değişim miktarını hesaplamak gerekir. Eğer çıktı katmanında

bulunan proses elemanlarının eşik değer ağırlıkları β^y ile gösterilirse; bu ünitenin çıktısının sabit ve 1 olması nedeni ile değişim miktarı Eşitlik (3.10) ile hesaplanır.

$$\Delta\beta_m^y(t) = \lambda\delta_m + \alpha\Delta\beta_m^y(t-1) \quad (3.10)$$

Eşik değerlerin t. iterasyondaki ağırlığının yeni değeri ise Eşitlik (3.11) yardımıyla hesaplanır [2].

$$\beta_m^y(t) = \beta_m^y(t-1) + \Delta\beta_m^y(t) \quad (3.11)$$

3.3.4 Ara katmanlar arası veya ara katman ile girdi katmanı arasındaki ağırlıkların değiştirilmesi

Dikkatli incelenirse, ara katman ile çıktı katmanı arasındaki ağırlıkların değişiminde her ağırlık için sadece çıktı katmanındaki bir proses elemanının hatası dikkate alınmıştır. Bu hataların oluşmasında girdi katmanı ve ara katman arasındaki ağırlıkların (varsa iki katman arasındaki ağırlıkların) payı vardır. Çünkü, en son ara katmana gelen bütün bilgiler girdi katmanı veya önceki ara katmandan gelmektedir. O nedenle girdi katmanı ile ara katman arasındaki (veya iki ara katman arasındaki) ağırlıkların değiştirilmesinde çıktı katmanındaki proses elemanlarının hepsinin hatasından payını alması gerekir. Bu ağırlıklardaki değişimi (mesela girdi katmanı ile ara katman arasındaki ağırlıkların değişimi) Δw^i ile gösterilirse değişim miktarı;

$$\Delta w^i = \Delta w_{kj}^i \lambda \delta y_k^i + \alpha \Delta w_{kj}^i(t-1) \quad (3.12)$$

Eşitlik (3.12) yardımıyla hesaplanır. Buradaki hata terimi δ^a ise Eşitlik (3.13)'e göre hesaplanacaktır.

$$\delta_j^a = f'(NET) \sum_m \delta_m w_{jm}^a \quad (3.13)$$

Aktivasyon fonksiyonu olarak sigmoid fonksiyonu düşünülürse bu hata değeri Eşitlik (3.14) ile hesaplanacaktır.

$$\delta_j^a = y_j^a (1 - y_j^a) \sum_m \delta_m w_{jm}^a \quad (3.14)$$

Hata değeri hesaplandıktan sonra yukarıda verilen Eşitlik (3.14) ile değişim miktarını bulmak mümkün olur. Ağırlıkların yeni değerleri ise Eşitlik (3.15) yardımıyla hesaplanır.

$$w_{kj}^i = w_{kj}^i(t-1) + \Delta w_{kj}^i(t) \quad (3.15)$$

Benzer şekilde, eşik değeri ünitesinin yeni ağırlıkları da yukarıdaki gibi hesaplanır. Ara katman eşik değeri ağırlıkları β^a ile gösterilirse değişim miktarı Eşitlik (3.16)'ya göre hesaplanır.

$$\Delta \beta_j^a(t) = \lambda \delta_j^a + \alpha \Delta \beta_j^a(t-1) \quad (3.16)$$

Ağırlıkların yeni değerleri ise t. iterasyonda Eşitlik (3.17) olarak hesaplanacaktır.

$$\beta_j^a(t) = \beta_j^a(t-1) + \Delta \beta_j^a(t) \quad (3.17)$$

Böylece ağırlıklarının hepsi değiştirilmiş olacaktır. Bir iterasyonda hem ileri hem de geriye hesaplamalar yapılarak tamamlanmış olacaktır. İkinci bir örnek verilerek sonraki iterasyona başlanır ve aynı işlemler öğrenme tamamlanmaya kadar tekrarlanır [2].

3.4 ÇKA Ağının Çalışma Prosedürü

ÇKA ağlarının çalışması şu adımları içermektedir.

1. Adım - Örneklerin toplanması: Ağın çözmesi istenilen olay için daha önce gerçekleşmiş örneklerin bulunması adımıdır. Ağın eğitilmesi için örnekler toplandığı gibi (eğitim seti) ağın test edilmesi için de örneklerin (test seti) toplanması gerekmektedir. Ağın eğitilmesi sırasında test seti ağa hiç gösterilmez. Eğitim setindeki örnekler tek tek gösterilerek ağın olayı öğrenmesi sağlanır. Ağ olayı öğrendikten sonra test setindeki örnekler gösterilerek ağın performansı ölçülür. Hiç görmediği örnekler karşısındaki başarısı ağın iyi öğrenip öğrenmediğini ortaya koymaktadır.

2. Adım - Ağın topolojik yapısının belirlenmesi: Öğrenilmesi istenen olay için oluşturulacak olan ağın topolojik yapısı belirlenir. Bu aşamada, girdi katmanındaki proses elemanlarının, ara katmanların, ara katman veya katmanlardaki proses elemanlarının ve çıktı katmanındaki proses elemanlarının sayıları belirlenmektedir.

3. Adım - Öğrenme parametrelerinin belirlenmesi: Ağın öğrenme katsayısı proses elemanlarının toplama ve aktivasyon fonksiyonları, momentum katsayısı gibi parametreler bu adımda belirlenmektedir.

4. Adım - Ağırlıkların başlangıç değerlerinin atanması: Proses elemanlarını birbirlerine bağlayan ağırlık değerlerinin ve eşik değer ünitesinin ağırlıklarının başlangıç değerlerinin atanması yapılır. Başlangıçta genellikle rastgele değerler atanır. Daha sonra ağ uygun değerleri öğrenme sırasında kendisi belirler.

5. Adım - Öğrenme setinden örneklerin seçilmesi ve ağa gösterilmesi: Ağın öğrenmeye başlaması ve yukarıda anlatılan öğrenme kuralına uygun olarak ağırlıkları değiştirmesi için ağa örnekler (girdi / çıktı değerleri) belirli bir düzeneğe göre gösterilir.

6. Adım - Öğrenme sırasında ileri hesaplama yapılması: Yukarıda anlatıldığı şekilde sunulan girdi için ağın çıktı değerleri hesaplanır.

7. Adım - Gerçekleşen çıktının beklenen çıktı ile karşılaştırılması: Ağın ürettiği hata değerleri bu adımda hesaplanır.

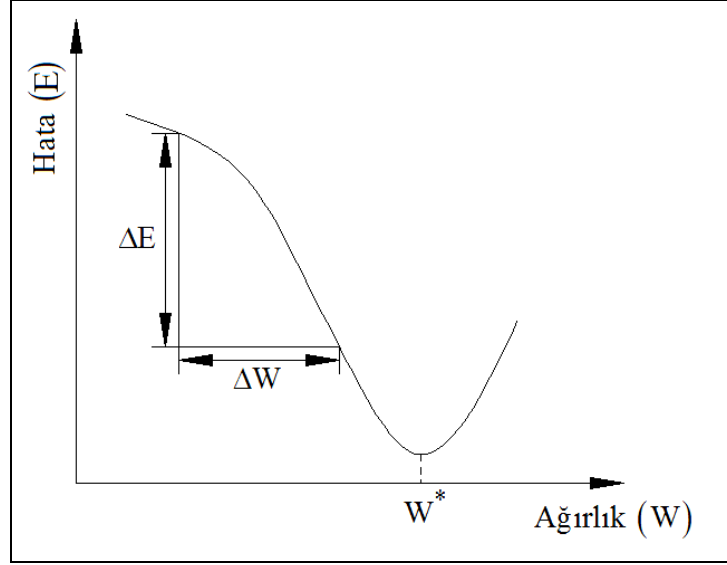
8. Adım - Ağırlıkların değiştirilmesi: Yukarıda anlatıldığı gibi geri hesaplama yöntemi uygulanarak üretilen hatanın azalması için ağırlıkların değiştirilmesi yapılır.

Yukarıdaki adımlar ÇKA ağının öğrenmesi tamamlanmaya, yani gerçekleşen çıktılar ile beklenen çıktılar arasındaki hatalar kabul edilir düzeye ininceye kadar devam eder. Ağın öğrenmesi için bir durdurma kriterinin olması gerekmektedir. Bu ise genellikle üretilen hatanın belirli bir düzeyin altına düşmesi olarak alınmaktadır [2].

3.5 Ağın Eğitilmesi

ÇKA ağlarının eğitilmesi felsefesi diğer ağlarınkinden farklı değildir. Ağın kendisine gösterilen girdi örneği için beklenen çıktıyı üretmesini sağlayacak ağırlık değerleri bulunmaktadır. Başlangıçta bu değerler rastgele atanmakta ve ağa örnekleri gösterdikçe ağın ağırlıkları değiştirerek zaman içerisinde istenen değerlere ulaşması sağlanmaktadır. İstenen ağırlık değerlerinin ne olduğu bilinmemektedir. Bu nedenle yapay sinir ağlarının davranışlarını yorumlamak ve açıklamak mümkün olmamaktadır. Zaten bu ağları diğer yapay zeka tekniklerinden mesela uzman sistemlerden ayıran en önemli farkı davranışlarını açıklayamamasıdır. Bunun temel nedeni de bilginin ağ üzerine dağıtılmış olması ve ağırlık değerlerinin kendi başlarına herhangi bir anlam

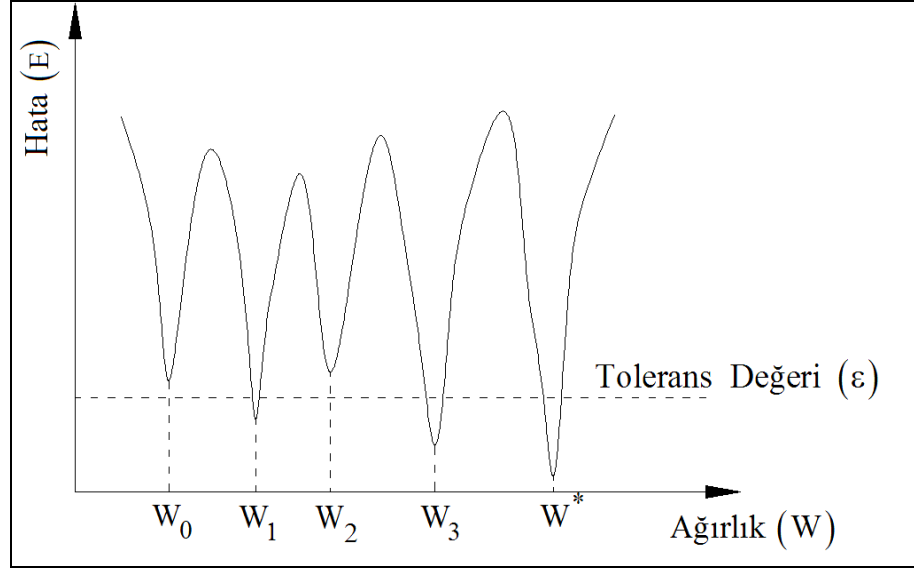
göstermemeleridir. Ağ ile ilgili bilinen konu, problem uzayında en az hata verebilecek ağırlık değerlerinin bulunmasıdır. Hatanın en az değeri kavramını anlatabilmek için basit bir problem düşünülürse, ağıın öğrenmesi istenen olayın (problem uzayının) Şekil 3.2’de gösterildiği gibi bir hata uzayı olduğu varsayalım. Şekil 3.2’de W^* en az hatanın olduğu ağırlık vektörünü göstermektedir.



Şekil 3.2: Hata – ağırlık eğrisi (basit)

Ağın W^* değerine ulaşması istenmektedir. Bu ağırlık değeri problem için hatanın en az olduğu noktadır. O nedenle her istasyonda ΔW kadar değişim yaparak hata düzeyinde ΔE kadar bir hatanın düşmesi sağlanmaktadır. Burada bir noktaya dikkatleri çekmek gerekir. Problemin hata düzeyi her zaman böyle basit ve iki boyutlu olmayacaktır. Daha karmaşık bir hata düzeyini gösteren Şekil 3.3’te görüldüğü gibi problemin çözümü için en az hatayı veren ağırlık vektörü W^* olmasına rağmen pratikte bu hata değerini yakalamak çoğu zaman pek mümkün olmayabilmektedir. Bu çözüm ağın sahip olabileceği en iyi çözümdür. Fakat bu çözüme nasıl ulaşılacağı konusunda elimizde bir bilgi yoktur. Ağ, eğitim sırasında kendisi bu çözümü yakalamaya çalışmaktadır. Bazen farklı bir çözüme takılabilmekte ve performansı daha iyileştirmek mümkün olmamaktadır. O nedenle kullanıcılar ağların problemlere ürettikleri çözümlerde (performanslarında) ϵ (tolerans değeri) kadar hatayı kabul etmektedirler. Tolerans değeri aralığındaki herhangi bir noktada olay öğrenilmiş kabul edilmektedir. Şekildeki W_0 ve W_2 çözümlerinin hataları, kabul edilebilir hata düzeyinin üzerinde olduğundan bu çözümler kabul edilemez çözümlerdir. Bunlara yerel çözümler denilmektedir. W_1 ve W_3 çözümleri en iyi çözüm olmamalarına

rağmen kabul edilebilir hata düzeyinin altında bir hataya sahiptirler. Bunlar da yerel çözümler olmalarına rağmen kabul edilebilir çözümlerdir. Görüldüğü gibi bir problem için birden fazla çözüm üretilebilmektedir. Bu nedenle yapay sinir ağlarının her zaman en iyi çözümü ürettikleri söylenemez. Kabul edilebilir bir çözüm ürettiklerini söylemek daha doğru olur. Üretilen çözüm en iyi çözüm olsa bile bunun bilinmesi zordur. Çoğu durumda bilinmesi mümkün değildir.



Şekil 3.3: Hata – ağırlık eğrisi (karmaşık)

Şekil 3.3 aynı zamanda başka bir gerçeği daha göstermektedir. Neden en iyi sonuç bulunamamaktadır? Bunun başka nedenleri de olabilir. Örneğin,

- Problem eğitilirken bulunan örnekler problem uzayını %100 temsil etmeyebilir.
- Oluşturulan ÇKA ağı için doğru parametreler seçilmemiş olabilir.
- Ağın ağırlıkları başlangıçta tam istenildiği şekilde belirlenmemiş olabilir.
- Ağın topolojisi yetersiz seçilmiş olabilir.

Bu ve benzeri nedenlerden dolayı ağ, eğitim sırasında, hatayı belirli bir değerin altına düşüremeyebilir. Mesela W_1 ağırlıklarını bulur, hatayı da aşağıya düşürmez. Bu aslında yerel bir çözümdür. En iyi çözüm değildir. Hata kabul edilebilir düzeye indiğinden yerel en iyi bir çözüm olarak görülebilir. Global çözümün bulunması da mümkün olabilir. Bu tamamen ağın tasarımına, örneklerin niteliğine ve eğitim sürecine bağlıdır.

Bazı durumlarda ağıın takıldığı yerel sonuç kabul edilebilir hata düzeyinin üstünde kalabilir (Mesela şekildeki W_0 ağırlıklarının bulunması ve hatanın daha fazla azaltılmasının mümkün olmaması). Bu durumda ağıın olayı öğrenmesi için bazı değişiklikler yapılarak yeniden eğitilmesi gerekir.

Bu değişiklikler aslında şunlar sayılabilir:

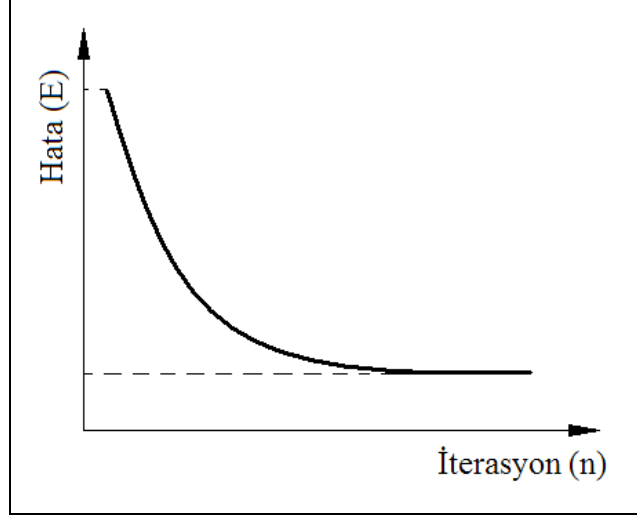
- Başka başlangıç değerleri kullanılabilir
- Topolojide değişiklikler yapılabilir (ara katman sayısını artırmak, proses elemanı sayısını artırmak veya azaltmak gibi)
- Parametrelerde değişiklik yapılabilir (fonksiyonların başka seçilmesi, öğrenme ve momentum katsayılarının değiştirilmesi gibi)
- Problemin gösterimi ve örneklerin formülasyonu değiştirilerek yeni örnek seti oluşturulabilir
- Öğrenme setindeki örneklerin sayısı artırılabilir veya azaltılabilir
- Öğrenme sürecinde örneklerin ağı gösterilmesi

ÇKA ağılarının yerel sonuçlara takılı kalmaması için momentum katsayısı geliştirilmiştir. Bu katsayının iyi kullanılması yerel çözümleri kabul edilebilir hata düzeyinin altına çekebilmektir.

ÇKA ağılarının eğitilmesinde diğer önemli bir sorun ise öğrenme süresinin çok uzun olmasıdır. Ağırlık değerlerinin başlangıçta büyük değerler olması durumunda, ağıın yerel sonuçlara düşmesi ve bir yerel sonuçtan diğerine sıçramasına neden olmaktadır. Eğer ağırlıklar küçük aralıkta seçilirse o zaman da ağırlıkların doğru değerleri bulması uzun zamanlar almaktadır. Bazı problemlerin çözümü sadece 200 iterasyon sürerken bazıları 5 – 10 milyon iterasyon gerektirmektedir. Bu konuda da elimizde bilimsel bir yaklaşım yoktur. Tamamen deneme yanılma yolu ile en uygun başlama koşullarının belirlenmesi gerekmektedir.

Ağıın öğrenmesinin gösterilmesinin en güzel yolu hata grafiğı çizdirmektir. Öğrenen bir ağı için her iterasyonda oluşan hatanın grafiğı çizilirse Şekil 3.4'te gösterildiğı şekilde benzer bir hata grafiğı oluşur. Burada hatanın zaman içerisinde düştüğü görülür. Belirli bir iterasyondan sonra hatanın daha fazla azalmadığı görülür. Bu ağıın öğrenmesini durdurduğı ve daha iyi sonuç bulunamayacağı anlamına gelir.

Eğer elde edilen çözüm kabul edilemez ise o zaman ağ yerel bir çözüme takılmış demektir. Bu durumda ağın yukarıda önerilen değişiklikleri yaparak yeniden eğitilmesi gerekir [2].



Şekil 3.4: Hata – iterasyon ilişkisi

3.5.1 ÇKA ağının performansının ölçülmesi

Bir yapay sinir ağının performansı denilince öğrenme yeteneğinin ölçülmesi anlaşılır. Ağ eğitim sırasında kendisine gösterilen bütün örnekler doğru cevaplar üretiyor diye performansı iyidir denemez. Bu ağın öğrenip öğrenmediğini gösterebilir ama iyi öğrenip öğrenmediğini göstermez. O nedenle, öğrenen ağların daha önce görmedikleri örnekler karşısında da beklenen performansı gösterip gösteremeyeceklerinin ölçülmesi gerekmektedir. Bunun için genellikle ağın eğitildiği problem üzerinden hem eğitimde kullanılacak hem de test esnasında kullanılacak örnekler gösterilir. Ağ bunları öğrenince (hepsi için veya kabul edilebilir oranda hepsi için doğru cevaplar üretmeye başlayınca) ağa hiç görmediği test setindeki örnekler gösterilir. Ağın performansı Eşitlik (3.18) yardımıyla, görmediği örnekler karşısında ürettiği doğru cevaplar oranı şeklinde ölçülür.

$$P = \frac{D}{T} \times 100 \quad (3.18)$$

Burada D test setinden doğru olarak cevaplanan örnek sayısını, T test setinde bulunan toplam örnek sayısını, P ise performans oranını göstermektedir. Eğer performans oranı istenilen düzeyde veya kabul edilebilir bir değerde değil ise ağ eğitim setindeki bütün örnekleri doğru cevaplasa bile iyi öğrenmiştir denemez.

O zaman eğitime biraz daha devam etmek gerekebilir. Eğitim iterasyonlarını artırmaya rağmen hala performans artmıyor ise o zaman örneklerin problem uzayını iyi temsil edemedikleri veya ağın parametrelerinin veya topolojisinin iyi seçilemediği anlaşılır. Daha önce belirtilen değişiklikleri yaparak ağı yeniden eğitmek gerekebilir. Ağı eğitilmesi sırasında dikkat edilmesi gereken konulara aşağıda da değinilecektir [2].

3.6 Bir ÇKA Ağına Oluşturulmasında Dikkat Edilmesi Gereken Bazı Önemli Noktalar

Yapılan araştırmalar ve tecrübeler bir ÇKA ağına performansını etkileyen unsurların şunlar olduğunu göstermektedir:

- Örneklerin seçilmesi
- Girdi ve çıktılarının ağa gösterimi
- Girdilerin nümerik gösterimi
- Çıktıların nümerik gösterimi
- Başlangıç değerlerinin atanması
- Öğrenme ve momentum katsayılarının belirlenmesi
- Örneklerin ağa sunulması
- Ağırlıkların değiştirilme zamanları
- Girdi ve çıktılarının ölçeklendirilmesi
- Durdurma kriterinin belirlenmesi
- Ara katmanların ve her katmandaki proses elemanlarının sayısının belirlenmesi
- Ağların büyütülmesi veya budanması [2].

3.6.1 Örneklerin seçilmesi

Örneklerin seçilmesi ağına performansını yakından ilgilendiren bir konudur. Çünkü ağ bu örnekleri dikkate alarak ağırlıkları değiştirmektedir. Seçilen örneklerin problem uzayını temsil edebilecek nitelikte olması çok önemlidir. Bazı ÇKA ağı tasarımcıları

problem uzayının sadece bir dilimini gösteren örnekleri ağa göstermekte fakat tamamı ile ilgili yorumlar yapmasını beklemektedir. Bu mümkün değildir. Bazıları ise elmayı gösterip portakalı sormaktadır. Unutulmaması gereken şudur ki ağa ne gösterilirse ağ ancak o konularda yorumlar yapabilir ve ancak o konuda, görmediği örneklere çözümler üretebilir. $2 \times 2 = 4$ diye ağa öğretirseniz 3×3 kaç eder diye soramazsınız. Çünkü ağ bu konuda bir örnek görmemiş ve genellemeleri yapacak durumda olmamıştır.

Bazı durumlarda ise ağa problem uzayının sadece uç değerleri gösterilmekte ve bütün problem uzayını öğrenmesi istenmektedir. Bu durumda da ağın problem uzayını öğrenmesi beklenemez. ÇKA ağı tasarımcılarının problem uzayının her bölgesinden ve uzayı temsil eden örnekleri seçmesi gerekmektedir.

Hangi örneklerin problem uzayını temsil ettiğini belirlemek bazen kolay olmayabilir. Belirlenen örneklerin uzayı temsil etme yeteneklerini ölçebilecek bir yöntem de şu ana kadar geliştirilmemiştir. Fakat tasarımcı kendisi örnekleri seçerken ekstrem uçlardan ve sadece belirli bölgelerden örnekler almaktan kaçınmalıdır. Örneğin kadın ve erkek resimlerini birbirinden ayıran bir ÇKA modeli tasarlanması istenirse erkek resimlerini temsilen sadece gür saçlı erkek resimlerini kullanıp ondan sonra saçlı olmayan bir erkek resmini tanımasını istemek doğru olmaz. Bıyıklı ve sakallı erkek resimlerinin de tanınması söz konusu olacak ise o zaman onların örneklerinin eğitim seti içinde yer alması gerekmektedir. Aksi takdirde ya ağ başarısız olacak ya da problemin tanımı sadece gür saçlı erkekleri kadınlardan ayıran bir ÇKA ağı olacaktır. Bu konuya tasarımcı mümkün olduğu kadar dikkat etmelidir. Bu konuda önerilen, öncelikle bütün örnekleri belirleyip onları eğitim ve test seti olarak ikiye bölmektir. Ağ eğer test setinde başarılı sonuçlar üretiyorsa öğrenmiş demektir. Çünkü test setindeki örnekler ağ öğrenirken görmemektedir. Test setinde başarılı olan fakat günlük kullanımda sorunlar gösteren bir ağ için başarısız demek doğru değildir. Problem ağa iyi gösterilmemiş ve iyi örnekler seçilmemiş olması olasıdır diye bakmak lazımdır. Onun için bu gibi durumlarda öncelikle örnek setinin gözden geçirilmesi gerekmektedir. Ağın başarısız olduğu örnekler belirlenip toplanarak eğitim setine katılır ve ağ yeniden eğitilerek performansı artırılır. Böylece, zaman içinde problem uzayını gösteren eğitim setine ve ağa kavuşmak mümkün olabilir. Yalnız böyle bir uygulama ilgili ağın test seti üzerindeki performansının yüksek olması durumunda yapılması gerekir. Test setine iyi cevaplar üretemeyen ağ zaten iyi öğrenememiş

demektir. Orada örnek seti kadar başka sorunlarda söz konusu olabilir. Bu durumda; aşağıda anlatılanlar ışığında örnek setini de içerecek bir çalışma ile düzenlemeler yapılarak ağın performansının artırılması gerekir.

Bu konuda diğer önemli konu da seçilen örneklerin ilgili problem uzayında gerçekleşmiş ve çözülmüş gerçek örnekler olmasıdır. Hayali çözümler veya varsayım örnekler başarılı sonuçların doğmasını önleyebilirler. Bazı durumlarda örnek setini oluşturmak çeşitli nedenlerden dolayı mümkün olmayabilir. Bu durumlarda ilgili problem uzayının bir benzetim modeli oluşturularak gereken örnekleri üretmek mümkün olabilir. Bu durumda benzetim modelinin gerçeğe ne kadar yakın tasarlandığı önemlidir. Ağın bu benzetim modeli üzerinden gelen örnekler ile öğrenip günlük kullanımda sonuçlar üretememesi ağın başarısız olduğu anlamına gelmez. Tasarımcılar, başarısız sonuçlar elde edildiği zaman öncelikle benzetim modelinin iyi tasarlandığından emin olmalıdırlar. Benzer şekilde benzetim modelinin de yine bütün problem uzayını gösteren örnekler üretecek şekilde tasarlanması gerekmektedir.

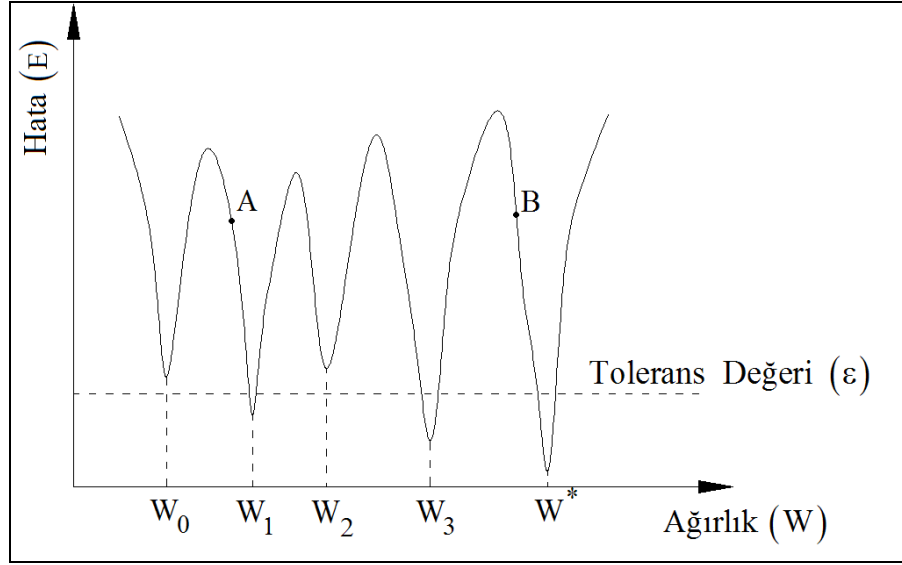
Örneklerin seçiminin önemini kavrayamayan bazı tasarımcılar ağıları eğitememekte ve öğrenmiyor diye şikayet etmektedirler. Örneklerini inceleyip gerekli düzenlemeleri yapsalar belki de bir ya da iki saat içinde öğrenecek bir ağ tasarlayabileceklerdir. Kısaca ağa ne gösterilirse karşılığında onun benzeri sonuçlar alınır. Bu gerçeği göz ardı etmeden eğitim ve test seti oluşturulmalıdır.

3.6.2 Girdi ve çıktuların gösteriminin belirlenmesi

Örneklerin belirlenmesi kadar belki ondan da daha önemlisi örneklerin gösteriminin nasıl olacağını belirlenmesidir. Girdi / çıktı çiftlerinden oluşan örnekler ağa nasıl gösterilecektir? Yapay sinir ağıları sadece rakamlar ile çalışmaktadırlar. Eğer problem uzayında sayısal (nümerik) olmayan faktörleri dikkate almak gerekiyor ise o zaman onların rakamlar ile temsil edilebilmesi gerekmektedir. Bu dönüştürme çeşitli şekillerde olabilmekte ve bu da ağın performansını etkilemektedir. Hem girdi değerlerinin hem de beklenen çıktı değerlerinin nümerik olarak gösterilmesi gerekmektedir [2].

3.6.3 Başlangıç değerlerinin atanması

ÇKA ağının proses elemanlarını birbirine bağlayan bağlantıların ağırlıklarının başlangıç değerlerinin atanması da ağın performansı ile yakından ilgilidir. Genel olarak ağırlıklar belirli aralıklarda atanmaktadır. Bu aralık eğer büyük tutulursa ağın yerel çözümler arasında sürekli dolaştığı, küçük olması durumunda ise öğrenmenin geç gerçekleştiği görülmüştür. Bu değerlerin atanması için henüz belirlenmiş standart bir yöntem yoktur. Ağırlıkların başlangıç değerlerinin rastgele atanmaları istenmektedir. Tecrübeler 1,0 ile 0,1 arasındaki değerlerin başarılı sonuçlar ürettiğini göstermektedir. Fakat bu tamamen öğrenilmesi istenen problemin niteliğine bağlıdır. Sadece ÇKA ağı kullanıcılarının bunu bilmeleri ve tecrübelerden faydalanarak denemeleri yapmaları için burada belirtilmektedir. Bu değerler arasında ağırlıkları atayıp hiç öğrenmeyen ağlar da olabilir. Çünkü ağın öğrenmesi daha bir çok parametreye ve öğreneceği olayın karmaşıklığına göre değişmektedir. Özellikle girdi / çıktı arasındaki ilişkinin belirlenmesinin zor olduğu durumlarda öğrenme de güçleşmekte hatta başarısızlıkla sonuçlanabilmektedir. Başlangıç değerleri problemin özünün aranmasına başlandığı noktaları göstermektedir. Öğrenemeyen bir ağın başlangıç değerlerinin değiştirilmesi ağın öğrenmesine neden olabilir. Bazı durumlarda ağın öğrenmesi zor bir olay üzerinde eğitildiğinde, ağın öğrenmesi başlangıç noktalarının değiştirilmesi ile de mümkün olmayabilir. O nedenle her olumsuz eğitimde, başlangıç değerlerini suçlamamak lazımdır. Diğerleri ile birlikte düşünmek gerekir. Şekil 3.5 başlangıç noktalarının önemini göstermektedir.



Şekil 3.5: Ağırlık başlangıç değerlerinin önemi

Şekil 3.5'te görüldüğü gibi eğer bir ÇKA ağı öğrenmeye A noktasından başlar ise yerel bir çözüme (W_1) takılabilme olasılığı var iken B noktasından başlarsa en iyi çözümü (W^*) bulması daha kolay olmaktadır [2].

3.6.4 Öğrenme katsayısı ve momentum katsayılarının belirlenmesi

Başlangıç değerleri kadar öğrenme ve momentum katsayılarının belirlenmesi de ağı öğrenme performansı ile yakından ilgilidir. Öğrenme katsayısı ağırlıkların değişim miktarını belirlemektedir. Eğer büyük değerler seçilirse o zaman yerel çözümler arasında ağı dolaşması ve osilasyon yaşaması söz konusu olmaktadır. Küçük değerler seçilmesi ise öğrenme zamanını arttırmaktadır. Tecrübeler genellikle 0,2 – 0,4 arasındaki değerlerin kullanıldığını göstermektedir. Fakat bu tamamen ilgili probleme bağlıdır. Bu değerler iyidir demek doğru olmaz. Bazı uygulamalarda öğrenme katsayısının 0,6 değerini aldığı zaman en başarılı sonuçları verdiği görülmektedir.

Benzer şekilde momentum katsayısı da öğrenmenin performansını etkiler. Momentum katsayısı bir önceki iterasyon değişiminin belirli bir oranının yeni değişim miktarına eklenmesi olarak görülmektedir. Bu özellikle yerel çözümlere takılan ağların bir sıçrama ile daha iyi sonuçlar bulmasını sağlamak amacı ile önerilmiştir. Bu değer küçük olması yerel çözümlerden kurtulmayı zorlaştırabilir. Çok büyük değerlerde ise tek bir çözüme ulaşmada sorunlar yaşanabilir. Tecrübeler bu değer 0,6 – 0,8 arasında seçilmesinin uygun olacağını göstermektedir. Fakat bu da kesindir

denilemez. Problemin niteliğine göre kullanıcının belirlemesinde fayda vardır. Daha küçük değerler ile başarılı sonuçların alındığını gösteren örnekleri görmek de mümkündür.

3.6.5 Örneklerin ağı sunulma şekli

Örneklerin ağı sunulma şekli de öğrenme performansını etkileyebilir. Genel olarak örnekler ağı iki türlü sunulabilirler. Bunlar:

- Sıralı sunum,
- Rastgele sunum.

Sıralı sunumda örnek setindeki birinci örnek ağı sunulur. Bir sonraki iterasyonda ise sırası ile ikinci, üçüncü ve en sonuncu örneğe kadar örneklerin tamamı ağı sunulur. Sonra tekrar başa dönerek örnek setindeki örnekler tek tek sıra ile ağı tekrar sunulur. Bu işlem öğrenme sağlanıncaya kadar devam eder. Bu tür bir sunuşta örnek setindeki bütün örneklerin ağı gösterilme şansları eşittir.

Rastgele sunumda ise örnekler eğitim seti içinden rastgele seçilirler. Burada da iki durum söz konusudur.

- Seçilen bir örnek tekrar set içine atılıp rastgele yeniden seçim yapılır. Bu durumda bir örneğin peş peşe birden fazla seçilme şansı vardır. Öğrenme gerçekleşene kadar böyle devam edilir. Örneklerin ağı gösterilme şansları eşit değildir.
- Rastgele seçilen örnek eğitim içine tekrar atılmaz. Kalanlar arasından rastgele tekrar yeni örnek seçilerek ağı sunulur. Bütün örnekler ağı gösterilince eğitim seti tekrar içinden seçilen rastgele örnekler seçilerek ağı gösterilir. Bir gösterilen örnek bütün set ağı gösterilinceye kadar bekler. Öğrenme sağlanıncaya kadar bu işlem aynı şekilde tekrar eder. Örneklerin ağı gösterilme şansları bu durumda da eşittir.

3.6.6 Ağırlıkların değiştirilmesi zamanı

Ağırlıkların değiştirilmesi öğrenme kuralına göre yapılmaktadır. Genel olarak üç durumda ağırlıkların değiştirilmesine izin verilmektedir. Problemin durumuna göre ağırlıkların ne zaman değiştirileceğine karar vermek gerekir. Doğru zamanlama, ağı öğrenme performansını etkilemektedir. Bu üç durum şöyle özetlenebilir;

- **Her örnek ağı gösterildiğinde (pattern based learning):** Bu durumda ağı her örnek gösterildiğinde beklenen çıktı ile ağı gerçekleştirdiği çıktı arasındaki hata bulunur ve bu hata, ağırlıklara öğrenme kuralı gereğince dağıtılır. İkinci örnek ağı sunulduğunda çıktının hatası hesaplanır ve ağırlıklar değiştirilir. Her örnek gösterimi sonucu ağırlıklar değiştirilir.

- **Belirli sayıda örnek ağı gösterildiğinde (batch based learning):** Bu durumda ağı her örnek gösterildiğinde hatası hesaplanıp ağırlıklar değiştirilmez. Belirli sayıda örnek tek tek ağı gösterilir ve hatalar toplanır. İstenen sayıdaki örneğin ağı gösterilmesi bitince toplanan hata ağırlıklara dağıtılır. Genellikle 5 – 10 örnekten oluşan örnek grupları oluşturulmaktadır. Yani beş örnek peş peşe ağı gösterilmekte, hatalar hesaplanıp toplanmakta ve toplam hata öğrenme kuralına göre ağırlıklara dağıtılmaktadır. Aynı işlemler her örnek grubundaki örneklerin tamamı ağı gösterildikçe tekrarlanmaktadır.

- **Bütün örnek seti ağı gösterildiğinde (epoch based learning):** Bu durumda örnek setindeki bütün örnekler ağı tek tek gösterilir. Hatalar hesaplanır ve eğitim setindeki örneklerin tamamının hataları toplandıktan sonra bu hata ağırlıklara dağıtılır. Yani; ağı ağırlık değerleri örneklerin tamamı ağı gösterilmedikçe değiştirilmez. Örnek sayısının az olduğu durumlarda önerilmektedir.

3.6.7 Örneklerin değerlerinin ölçeklendirilmesi

ÇKA ağılarında girdi ve çıktıların ölçeklendirilmesi de ağı performansını yakından etkilemektedir. Çünkü ölçeklendirme örneklerin değerlerinin dağılımını düzenli hale getirmektedir. Bazı durumlarda problem uzayı ile ilgili örnekler farklı ölçekler kullanılan ortamlardan toplanmış olabilir. Bazı durumlarda da problemin girdileri arasında aşırı büyük ve küçük değerler görülebilir. Bu durumda bütün girdilerin belirli aralıkta ölçeklendirilmesi (genellikle 0 – 1 aralığında) hem farklı ortamlardan gelen bilgilerin aynı ölçek üzerine indirgenmesine hem de yanlış şekilde girilen çok büyük ve küçük değerlerin etkisinin ortadan kalkmasına neden olur.

3.6.8 Durdurma kriterleri

ÇKA modelinde ağı eğitilmesi kadar gereğinden fazla eğitilmemesi de önemlidir. Çünkü eğitilmek istenen bir ağı problem uzayına çözüm üretecek ağırlıkları bulduktan sonra eğitime devam edilirse bu ağı ağırlıklarında daha fazla değişikliklere neden

olur. Böylece en iyi çözümlü üreten bir ağ tekrar performansı daha düşük veya öğrenemeyen ağlara dönüşebilir. O nedenle ağın eğitiminin ne zaman durdurulması gerektiği konusunda karar vermek gerekmektedir. Pratikte genel olarak iki türlü durdurma kriteri kullanılmaktadır. Bunlar;

- Hatanın belirli bir değerin altına düşmesi halinde eğitimi durdurma
- Ağın belirli bir iterasyon sayısını tamamlaması sonucu eğitimi durdurmadır

3.6.9 Ara katman ve proses elemanları sayılarının belirlenmesi

ÇKA modelinde herhangi bir problem için kaç tane ara katman ve her ara katmanda kaç tane proses elemanı kullanılması gerektiğini belirten bir yöntem şu ana kadar bulunmuş değildir. Eğer girdilerin hepsi ikili olursa o zaman bazı yöntemler önerilmekle birlikte bu konudaki çalışmalar deneme yanılma yönteminin etkin olarak kullanıldığını göstermektedir. Ara katman sayısı ve proses elemanı sayıları da ağın performansını yakından ilgilendirmektedirler. Tasarımcılar kendi tecrübelerine dayanarak bunları belirler. O nedenle bir problem herhangi bir ağ ile kabul edilebilir hata altında çözüm üretse bile daha iyi bir ağ olur mu diye farklı sayıdaki ara katman ve her ara katmanda farklı sayıda proses elemanları ile denemeler yapmak gerekir. Böylece performansı daha yüksek bir ağ bulmak mümkün olabilir. Bazı durumlarda başlangıçta bir ağ oluşturulup zaman içinde büyütülerek veya küçültülerek istenen ağa ulaşılır.

3.7 ÇKA Ağının Uygulama Alanları

ÇKA ağları hayatın hemen hemen her alanında örnekleri görülen bir modeldir. Günümüzde uygulamaların sayısını dahi bilmek mümkün değildir. Genel olarak;

- Sınıflandırma,
- Tahmin etme,
- Tanıma,
- Yorumlama,
- Teşhis etme, problemlerinde başarı ile kullanılmaktadır.

Değişik alanlarda görülen ÇKA ağının uygulamalarına bazı örnekler vermek gerekirse;

- **Yatırımların Planlanmasında**

Şu ana kadar bu tür problemlerin çözülmesinde daha çok doğrusal tahmin yöntemleri kullanılır iken yapay sinir ağları ile borsadaki bazı şirketlerin durumları takip edilmektedir. Geleceğe yönelik tahminler yapılmasında yapay sinir ağlarından özellikle ÇKA ağlarından faydalanılmaktadır.

- **İmza Analizinde**

Özellikle bankalarda bulunan müşterilerin imzaları ile yeni işlemlerde karşılaşılan imzaların karşılaştırılmasında özellikle ABD’de yaygın olarak ÇKA ağlarının kullanıldıkları belirtilmektedir. Çünkü ÇKA ağları otomatik hesap işlemlerini yaygın olarak gerçekleştirmeyi sağlamaktadır. Bu konuda yapay sinir ağları donanımlarının bile üretildiği ve ilk yapay sinir ağı çipinin bu amaçla kullanıldığı rapor edilmiştir.

- **Proses Kontrolünde**

ÇKA ağları proseslerin kalite kontrolünde yaygın olarak kullanılmakta ve prosesin çalışması sırasında davranışlarını sürekli gözetleyerek anormal durumların olması halinde operatörleri uyarabilmektedir.

- **Makinelerin İzlenmesinde**

Bu konuda çok sayıda örnek vermek mümkündür. Mesela, ÇKA ağları, uçakların motorlarının davranışlarının izlenmesinde titreşim ve ses analizleri durumlarından olası hataları erkenden haber verebilmektedir. Benzer şekilde tren motorlarının izlenmesinde de kullanıldıkları bilinmektedir. İmalatta makinelerin hata teşhisinde kullanılmakta oldukları rapor edilmektedir.

- **İmalat Sektöründe**

Bu sektörde de ÇKA ağlarının uygulamalarına sayısız örnek vermek mümkündür. İşlerin makinelere çizelgelenmesinden kalite kontrole kadar bir çok alanda örnekler görülmektedir [2].

4. MAGNEZYUMDAN YAPI ELEMANLARI ÜZERİNDE GERÇEKLEŞTİRİLEN YORULMA DENEYLERİNDEN ELDE EDİLEN VERİLERİN YSA YAKLAŞIMI İLE İNCELENMESİ

Yapay sinir ağları daha önce de bahsedildiği gibi günümüzde birçok güncel uygulama alanında verimli bir şekilde kullanılabilir. Bu alanlardan bir tanesi olan, maliyetli ve uzun bir zaman dilimi gerektiren deneysel çalışmalardan oluşan malzemelerin yorulma davranışlarının incelenmesi uygulamalarının, yapay sinir ağları yöntemi ile modellenebilmesinin, bu maliyetleri ve gerekli zaman dilimini azaltabilecek bir uygulama olduğu düşünülmektedir. Mühendisliğin esaslarından olan ekonomik ve hızlı olma ihtiyacı, birçok mühendislik uygulamasında başarılı bir şekilde kullanılabilen deneysel çalışmalara göre daha ekonomik ve hızlı bir sistem olan yapay sinir ağları yöntemini, bu uygulama alanında da cazip kılmaktadır.

Magnezyumdan kaynaklı ve kaynaksız yapı elemanlarının yorulma davranışlarının yapay sinir ağları yöntemi ile incelenmesi işleminde aşağıdaki adımlar izlenmiştir.

- Daha önceden laboratuvar ortamında gerçekleştirilmiş magnezyumdan kaynaklı ve kaynaksız yapı elemanlarına ait verilerin toplanması. Bu veriler içerisinden eğitim ve test setlerinin oluşturulması.
- Bu uygulama için uygun olan yapay sinir ağı modelinin seçilmesi ve topolojik yapısının (mimarisinin) belirlenmesi.
- Ağın öğrenme parametrelerinin belirlenmesi.
- Ağın deneysel verilerden hazırlanmış setlerdeki eğitim ve test verilerine göre eğitimi ve test edilmesi.
- Ağın topolojisinde deneme yanılma yöntemine dayalı bir şekilde değişikliklere gidilmesi.
- Yapay sinir ağından elde edilmiş olan veriler ile gerçek deneysel çalışmalardan elde edilen verilerin [5] karşılaştırılması.

4.1 Çalışmada Kullanılan Deneysel Veriler

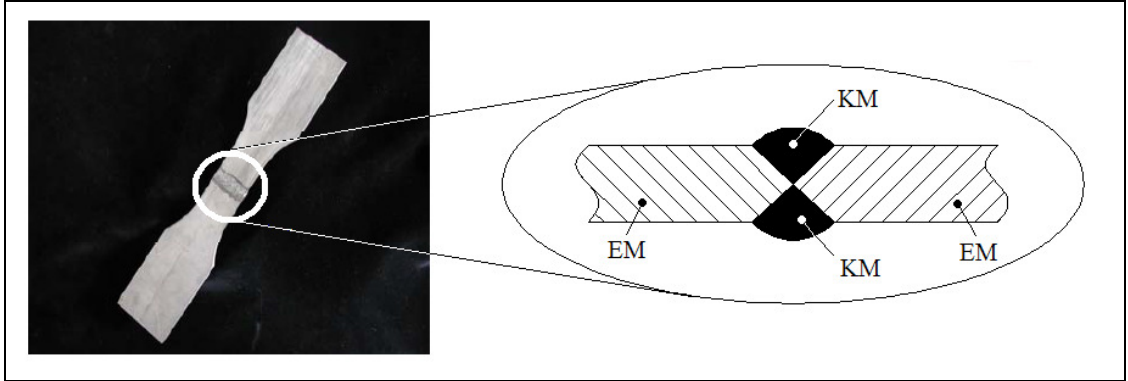
Yapılan çalışmada kullanılan veriler, magnezyumdan kaynaklı ve kaynaklı yapı elemanlarına ait üç farklı gruptan oluşan numuneler üzerinde yapılan deneysel çalışmalardan elde edilmiş verilerdir [5]. Bu üç farklı numune grupları şunlardır;

- Kök aralıksız kaynaklı numuneler
- Köşe dikişli kaynaklı numuneler
- Kaynaklı numuneler

Bu üç farklı deney grubu için optimum sonuçları verebilecek düzeyde geliştirilmiş bir yapay sinir ağı üzerinde çalışılmıştır. Yapılan çalışmalara ait detaylı anlatım ve bu çalışmalardan elde edilen sonuçlar aşağıda sırası ile gösterilmiştir.

4.2 Kök Aralıksız Kaynaklı Numunelere Ait Yorulma Davranışlarının YSA Yöntemiyle İncelenmesi

İlk örnek serisi olan kök aralıksız kaynaklı numunelerin şematik gösterimi Şekil 4.1’de gösterilmiştir.



Şekil 4.1: Kök aralıksız kaynaklı numunelerin kaynak dikiş şeklinin şematik gösterimi [5]

Kök aralıksız numuneler üzerinde gerilme oranı $R = -1$, $R = 0$ ve $R = 0,5$ olmak üzere üç farklı kategoride toplamda 42 adet deneysel veri bulunmaktadır. Bu birleştirmenin yorulma davranışlarını incelemek için tasarlanan YSA modelinin eğitimi ve test işlemleri bu veriler ile gerçekleştirilmiştir.

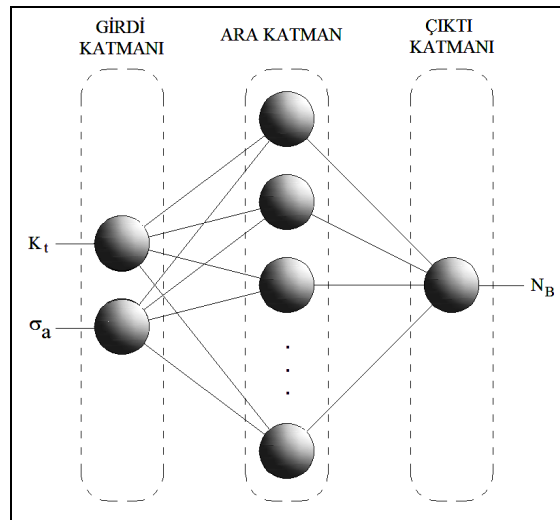
4.2.1 Uygulamada kullanılacak olan YSA modelinin mimari yapısı

Problemin analizinde kullanılacak olan ağ çok katmanlı algılayıcıdır. Çünkü analiz etmek istediğimiz problem doğrusal bir problem değildir ve bu nedenle, problemin çözümü için sadece doğrusal olan problemlere çözüm getirebilen ilk yapay sinir ağı modellemelerinden daha gelişmiş bir yapay sinir ağı modelinin kullanılması gerekmektedir. Çok katmanlı algılayıcılar daha önce de bahsedildiği gibi doğrusal olmayan problemlerin analiz ve çözümünü başarılı bir şekilde gerçekleştirebilecek yeterliliğe sahip olan bir yapay sinir ağı modelidir.

Oluşturmamız gereken ağ iki girdiye ve bir çıktıya sahip olacaktır. Bu girdiler;

- **1. Girdi:** Gerilme genliği (σ_a)
- **2. Girdi:** Çentik form sayısı (K_t)

olmaktadır. Yaptığımız sinir ağı modellemesinden bu girdilere karşılık elde edeceğimiz çıktı ise çevrim sayısı (N_B) olacaktır. Bu nedenle oluşturacağımız YSA modeli Şekil 4.2’de görüldüğü gibi iki adet nörondan oluşan bir giriş katmanı, ağdan elde edilen cevaplar ışığında deneme yanılma yöntemine bağlı olarak nöron sayısı değişecek olan bir ara katman ve bir nörondan oluşan çıkış katmanı şeklinde inşa edilecektir.



Şekil 4.2: Kullanılan YSA modeli

4.2.2 Problemin bilgisayar ortamına aktarılması

Yapılan çalışmada problemin bilgisayar ortamına aktarılmasındaki en önemli aşama verilerin doğru bir şekilde sisteme girilmesi ve çok uç değerlerin veya yetersiz veri gruplarının ayıklanmasıdır. Buradaki amaç ağı eğitimi esnasında aykırı veya yetersiz verilerin eğitim işlemini negatif bir şekilde etkilemesine ve sonuçların başarı oranlarının düşmesine neden olmasını önlemektir. Çünkü YSA eğitimlerinde kullanılan verilerin tutarlılığı, ağı eğitimi doğrudan etkilemektedir.

K_t çentik form sayıları her numune için ayrı ayrı bilinmemekte ve kök aralıksız kaynaklı numuneler için 1,79 – 4,15 değerleri arasında olmaktadır [5]. Malzemelerin yorulma davranışlarında çentik form sayısı K_t değerlerinin, kırılma çevrim sayısı (N_B) üzerinde birincil derecede rol oynadığı bilinmektedir. Bu nedenle, bu değerleri net olarak bilemesek de bu değerlere yakın değerler elde etmeye çalışarak yaptığımız YSA modelinde bu değerleri yaklaşık olarak kullanabilmek amaçlanmıştır.

K_t çentik form sayılarının yaklaşık değerlerinin elde edilmesi için ilk olarak Matlab programında Eşitlik (4.1)'de verilen matematiksel denklemi çözebilen bir sistem oluşturulmuştur. Elimizde K_t değerlerine ait uç noktalar haricindeki ara değerlere ait bir verinin bulunmaması nedeniyle konveksliği açısından en uygun olarak Eşitlik (4.1) uygun görülmüştür.

$$K_{ti} = \frac{a}{N_{Bi}^2} + b \quad (4.1)$$

Bu çözümün anlaşılabilmesi için daha önceden, kök aralıksız kaynaklı numuneler ile ilgili yapılmış olan deneysel çalışmalardan elde edilen veri grubundaki nominal gerilme genliği (σ_a) 20 MPa ve gerilme oranı $R = 0$ olan değerler için K_t değerlerinin nasıl elde edildiğini bir örnek olarak incelemek faydalı olacaktır. Tablo 4.1'de nominal gerilme genliği (σ_a) 20 MPa ve gerilme oranı $R = 0$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler gösterilmiştir [5].

Tablo 4.1: Nominal gerilme genliđi 20 MPa ve gerilme oranı R = 0 olan kök aralıksız kaynaklı numunelere ait deneysel veriler

Nominal Gerilme Genliđi (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı (N_B)
20	1,79 – 4,15	430687
20	1,79 – 4,15	478114
20	1,79 – 4,15	320450
20	1,79 – 4,15	378586
20	1,79 – 4,15	610932
20	1,79 – 4,15	353648
20	1,79 – 4,15	308655

Bu deneysel veri grubundaki K_t üst sınır deđerine $K_{t,max}$, K_t alt sınır deđerine $K_{t,min}$ diyelim. Ayrıca en büyük kırılma çevrim sayısına $N_{B,max}$, en küçük kırılma çevrim sayısına $N_{B,min}$ diyelim. Buna göre Eşitlik (4.1)'deki a ve b deđerlerimiz Eşitlik (4.2)'ye göre hesaplanır.

$$a = \frac{(K_{t,min} - K_{t,max})}{\left(\frac{1}{N_{B,max}^2} - \frac{1}{N_{B,min}^2} \right)} \quad (4.2)$$

$$b = K_{t,min} - \frac{a}{N_{B,max}^2}$$

Seçtiđimiz veri grubundaki deđerler için Eşitlik (4.1) ve Eşitlik (4.2)'yi Şekil 4.3'de gösterildiđi gibi matlab programında programladığımızda elde ettiğimiz K_t deđerleri Tablo 4.2'de gösterilmiştir.

```

1 -   clc
2 -   clear all
3
4   %degerlerin not defteri dosyasindan okunmasi
5   load ('cent1.txt');
6
7   %okunan degerlerin matrise çevrilmesi
8   x(1,19)=0.0;
9   for i=1:19,
10  x(1,i)=cent1(i,1);
11  end
12
13  %hiperbolik denklem degiskenlerinin belirlenmesi
14  y1 = 1.79; %deney serisindeki minimum centik faktor degeri
15  y2 = 4.15; %deney serisindeki maksimum centik faktor degeri
16  x1 = 610716; %deney serisindeki maksimum kirlma çevrim sayisi
17  x2 = 308655; %deney serisindeki minimum kirlma çevrim sayisi
18
19  % (a / x) + b = y seklindeki hiperbolik denklem katsayilarinin hesaplanmasi
20  a = (y1-y2) / ((1/x1^2)-(1/x2^2));
21  b = y1 - (a/x1^2);
22
23  %centik faktor degerlerinin hesaplanmasi
24  y = (a ./ x.^2) + b;

```

Şekil 4.3: Yaklaşık çentik form sayılarının hesaplanması

Tablo 4.2: Yaklaşık K_t değerleri hesaplanmış nominal gerilme genliği 20 MPa ve gerilme oranı $R = 0$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler [5]

Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı (N_B)
20	2,61	430687
20	2,30	478114
20	3,92	320450
20	3,09	378586
20	1,79	610932
20	3,40	353648
20	4,15	308655

Tablo 4.2’de görüldüğü üzere kırılma çevrim sayısı (N_B) ile ters orantılı olan çentik form sayısı (K_t) değerlerimiz bu örnek serisi için minimum kırılma çevrim sayısında ($N_{B,min} = 308655$) üst sınır değerini ($K_{t,max} = 4,15$), maksimum kırılma çevrim sayısında ise ($N_{B,max} = 610932$) alt sınır değerini almaktadır ($K_{t,min} = 1,79$). Ara değerler ise kırılma çevrim sayısı değerlerine göre hesaplanmıştır.

K_t değerlerinin hesaplanmasında dikkat edilmesi gereken husus, yaklaşık K_t değerlerinin bulunmasında kullanılacak olan verilerin bir saçılma aralığı göstermesidir. Ayrıca aynı gerilme oranı ve nominal gerilme genliğine sahip, kırılma çevrim sayıları içerisinde bulunan çok büyük veya çok küçük değerlerin kullanılması

durumunda K_t değerlerinde bu uç değerler yüzünden yığılmalar meydana gelmektedir. Örneğin, ortalama olarak 400000 kırılma çevrim sayısına sahip bir deney serisinde bulunan 10000000 gibi, beklenmeyen şekilde hasara uğramış bir numuneye ait bu değerlerin kullanılması, K_t alt sınır değerinin bu aykırı sonuca eşitlenmesine, diğer ortalama kırılma çevrim sayıları için de yaklaşık K_t değerlerini K_t üst sınır değeri ve buna çok yakın değerlere eşitleyerek bir yığılma göstermesine neden olacaktır.

4.2.3 Verilerin programa okutulması

Matlab programında oluşturulacak yapay sinir ağı modellemesi için kullanılacak verilerin programa matrisler şeklinde tanıtılması gerekmektedir. Eldeki deney serisine ait verilerin tek tek matris şeklinde yazılması zaman kaybı olarak değerlendirilebilir. Bu nedenle bu verilerin programa bir ortamdan (bir dosyadan) okularak programda matrise çevrilmesi daha hızlı ve doğru bir şekilde gerçekleştirilebilmektedir. Bunun için kök aralıksız kaynaklı numunelere ait nominal gerilme genliği (σ_a), çentik form sayısı (K_t) ve kırılma çevrim sayısı (N_B) değerleri YSA sisteminde eğitim ve test işlemleri için kullanılmak üzere, gerilme oranı $R = -1$, $R = 0$ ve $R = 0,5$ için ayrı ayrı Şekil 4.4'de görülen not defteri dosyası formatında hazırlanmıştır.

Dosya	Düzen	Biçim	Görünüm	Yardım
σ_a1	kt1	NB1		
σ_a2	kt2	NB2		
σ_a3	kt3	NB3		
σ_a4	kt4	NB4		
σ_a5	kt5	NB5		
σ_a6	kt6	NB6		
σ_a7	kt7	NB7		
σ_a8	kt8	NB8		
σ_a9	kt9	NB9		
σ_a10	kt10	NB10		
σ_a11	kt11	NB11		
σ_a12	kt12	NB12		
σ_a13	kt13	NB13		
σ_a14	kt14	NB14		
σ_a15	kt15	NB15		
σ_a16	kt16	NB16		
σ_a17	kt17	NB17		

Şekil 4.4: Programa okutulmak üzere düzenlenen not defteri dosyası formatı

Bu işlemin ardından yapay sinir ağı modeli olarak kullanılacak olan Matlab program dosyası (Matlab M-file) ile bilgisayarda aynı adreste (Örneğin: C:\Program Files\Matlab\ANN) bulunması gereken ve deney verilerini içeren not defteri dosyası “load” komutu ile programa okutulmuştur. Programın belleğine aldığı bu bilgiler “for” döngüsü yardımı ile not defteri dosyasındaki σ_a ve K_t değerlerini içeren ilk iki sütun satır satır okutularak, not defteri dosyasındaki her satırın birer sütun olarak bulunduğu p matrisine dönüştürülmüştür (Şekil 4.5).

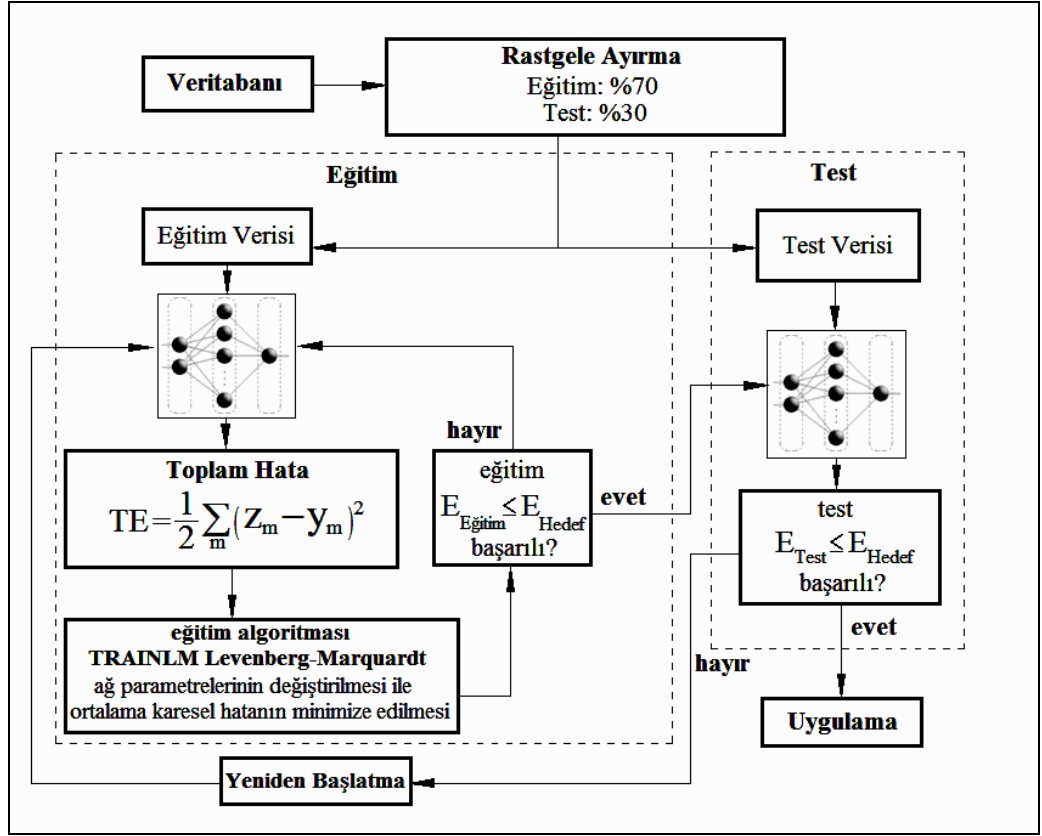
3. sütundaki veriler bizim çıktılarımız olacak olan N_B çevrim sayılarıdır. Bu nedenle son sütun aynı yöntemle p giriş matrisindeki değerlere karşılık gelen çıktı değerlerini içeren t çıkış matrisine dönüştürülmüştür (Şekil 4.5).

```
1 - clear all
2 - clc
3 - load ('network1.txt');
4
5 - %giris matrisinin olusturulmasi
6 - p(2,14)=0.0;
7 - for i=1:14,
8 -     for j=1:2,
9 -         p(j,i)=network1(i,j);
10 -     end
11 - end
12
13 - %cikis matrisinin olusturulmasi
14 - t(1,14)=0.0;
15 - for i=1:14,
16 -     t(1,i)=network1(i,3);
17 - end
```

Şekil 4.5: Verilerin program tarafından okunması, giriş ve çıkış matrislerinin oluşturulması

4.2.4 Yapay sinir ağı modelinin tasarımı, eğitimi ve simülasyonu

Verilerin ağına tanıtılabilecek bir forma getirilmesinden sonra bu veriler eğitim ve test olmak üzere iki gruba ayrılmıştır. Kök aralıksız kaynaklı numuneler üzerinde yapılmış olan deneysel çalışmalardan elde edilmiş $R = -1$, $R = 0$ ve $R = 0,5$ gerilme oranlarına sahip üç gruba ayrılmış toplamda 44 adet deneysel veri, 29 adet eğitim ve 15 adet test verisi olmak üzere tasarlanan YSA modelinin eğitim ve test işlemleri için kullanılmıştır. Tasarlanmış olan YSA modelinin çalışma şeklini daha iyi anlayabilmek için Şekil 4.6’da bulunan diyagramı incelemek faydalı olacaktır.



Şekil 4.6: YSA modelinin çalışma şekli

Şekil 4.6’da görüldüğü üzere veritabanımız yaklaşık %70 eğitim %30 test verisi olarak ayrılmıştır. Eğitim belirlenen eğitim algoritmasına göre toplam hata, hedeflenen tolerans değerinin altına düşünceye kadar devam eder. Eğitimin başarılı olması durumunda yani ağdan beklenen çıktı ile ağın üretebildiği çıktı arasındaki hata değerinin, istenilen hata tolerans değerine ulaşılması durumunda ayrılan %20’lik veri dilimiyle sistemin test aşamasına geçirilir ve tasarlanan YSA modeli burada eğitim esnasında kendisine gösterilmemiş olan değerler ile simülasyon işlemine tabi tutulur. Eğitim aşamasında olduğu şekilde toplam hata, tolerans değerine eşit veya bu değer altında ise tasarlanan YSA, istenilen kriterlere göre başarılıdır denebilir.

Bu problemin çözümünde kullanılacak olan YSA modeli, daha önce de bahsedildiği gibi çok katmanlı algılayıcı olan ileri beslemeli bir yapay sinir ağı modelidir. Oluşturulacak olan YSA modeli iki adet düğümden meydana gelen giriş katmanı, düğüm sayısı ağın eğitim ve test işlemlerindeki performansına bağlı olarak deneme yanılmaya dayalı şekilde belirlenecek olan bir ara katman ve bir adet düğümden meydana gelen çıkış katmanından oluşmaktadır.

Bu ađın oluřturulması iin ilk olarak programda “net = newff” komutu kullanılarak bir ileri beslemeli ađ yapısı oluřturulur. Aynı komut ierisinde kullanıma hazır hale getirdiđimiz eđitim seti (p ve t matrisleri) ađa tanıtılır ve gizli katmandaki dűđüm sayıları deneme yanılma yűntemiyle belirlenir. Ađırlık deđerleri eđitim iřleminin bařlangıcında program tarafından otomatik olarak atanır ve eđitim esnasında evrimler boyunca aynı řekilde program tarafından deđiřtirerek ayarlanır. Ađın eđitim iřlemi, bizim durdurma kriterlerimizden herhangi bir tanesi sađlanıncaya kadar sűrdűrűlűr. Őđrenme katsayımız 0,05 ve ađın eđitim esnasında yerel minimum sonulara takılmasını Őnlemek iin momentum sayısı 0,9 olarak ayarlanmıřtır. Durdurma kriterlerimiz maksimum dűngű sayısı (5000) ve ortalama hata olarak (0,01) belirlenmiřtir. Bu iřlemlerin ardından zaten bizim eđitim iin ayırdıđımız verilerinin YSA sisteminde eđitim, onay ve test řeklinde tekrar ayrılmaması iin eđitim serisine ayrılacak olan veri oranı řekil 4.7’de programın 26. satırında gűrűldűđű gibi, 1 olarak ayarlanmıř (net.divideParam.trainRatio = 1) ve neticede bu deđerlerin tamamının YSA sisteminin eđitiminde kullanılması sađlanmıřtır.

```
19 %YSA'nin tasarimi, egitimi ve simulasyonu
20 - net = newff(p,t,5,{'tansig' 'purelin'},'trainlm');
21 - net.trainParam.epochs = 5000;
22 - net.trainParam.goal = 0.01;
23 - net.trainParam.lr =0.05;
24 - net.trainParam.mc = 0.9;
25 - net.divideFcn = 'dividerand';
26 - net.divideParam.trainRatio = 1;
27 - net.divideParam.valRatio = 0.0;
28 - net.divideParam.testRatio = 0.0;
29 - net.trainParam.mu_max = inf;
30 - net.trainParam.max_fail = inf;
31 - net = train(net,p,t);
32 - a = sim(net,p);
33
```

řekil 4.7: YSA'nın tasarımı, eđitimi ve simulasyonu

alıřma esnasında bizim setiđimiz eđitim fonksiyonu (Levenberg-Marquardt eđitim algoritması, “trainlm” komutu), Őđrenme katsayısı (0,05) ve ara katmandaki dűđüm sayısı (5 adet) tamamen deneme ve yanılma yűntemiyle belirlenmiřtir. Yapay sinir ađları bűtűn parametrelerin aynı olması durumunda bile her eđitimde aynı

sonucu vermeyebilirler. Bu nedenle ara katmandaki düğüm sayısının neden beş olarak alındığını farklı düğüm sayıları ile deneme yaparak sonuçları karşılaştırmalı olarak verebilmek pek mümkün olamamaktadır.

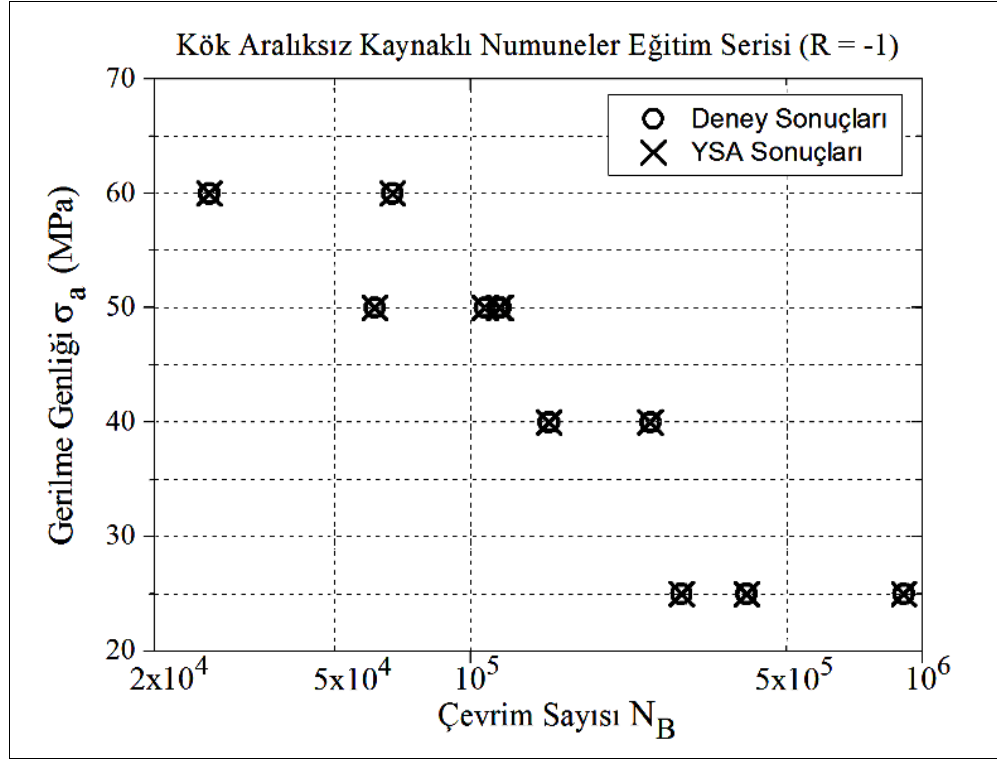
Şekil 4.7’de görüldüğü gibi, yukarıda bahsedilen işlem basamaklarından sonra “sim” komutu ile yapay sinir ağı modelinin, giriş değerleri ile oluşturulan p matrisindeki değerlere göre simülasyonu gerçekleştirilmiş ve elde edilen sonuçlar sistemde a olarak atanmıştır.

Kök aralıksız kaynaklı numunelere ait, gerilme oranı $R = -1$ olan deney grubu için yukarıda anlatılan şekilde, gerçekleştirilmiş eğitim ve simülasyon işlemlerinden elde edilmiş olan kırılma çevrim sayısı (N_B) değerleri Tablo 4.3’de gösterilmiştir.

Tablo 4.3: Gerilme oranı $R = -1$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile eğitilen YSA modelinden elde edilen değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
-1	25	4,15	292651
-1	25	2,88	407567
-1	25	1,79	909206
-1	40	4,15	149093
-1	40	1,79	250355
-1	50	4,15	61328
-1	50	1,80	115826
-1	50	1,95	107465
-1	50	1,79	116686
-1	60	1,79	67122
-1	60	4,15	26433

Elimizde bulunan, kök aralıksız kaynaklı numuneler üzerinde gerçekleştirilmiş deneysel çalışmalardan elde edilmiş, gerilme oranı $R = -1$ olan 17 adet veriden 11 tanesi eğitim, 6 tanesi ise test işlemi için kullanılmıştır. Bu deney grubu için eğitim işleminden sonra elde edilen değerler ile deneysel çalışmalardan elde edilmiş değerler karşılaştırmalı bir grafik olarak Şekil 4.8’de gösterilmiştir.



Şekil 4.8: Kök aralıksız kaynaklı numunelerin eğitim serisi (R = -1)

Şekil 4.8 incelendiğinde, gerilme oranı $R = -1$ olan kök aralıklı kaynaklı numuneler ile yapılmış deneysel çalışmalardan elde edilen değerler ile bu değerler kullanılarak eğitimi gerçekleştirilmiş YSA modelinden elde edilen değerlerin gerçek değerlere çok yakın olduğu açıkça görülmektedir. Fakat bu durum YSA modelinin eğitim işleminin başarılı bir şekilde gerçekleştirildiği anlamına gelmemektedir. Çünkü bu yargıya ulaşabilmek için ağırlık eğitim aşamasında verdiği cevaplar değil eğitim esnasında görmediği değerler için ürettiği cevapların söz konusu olduğu test aşamasında elde ettiği değerlerin incelenmesi gerekmektedir.

Eğitim işleminin gerçekleştirilmesinden sonra, eğitim esnasında ağırlık gösterilmeyen 6 adet test verisi ağırlık simülasyon işleminde kullanılmıştır. Gerilme oranı $R = -1$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile Şekil 4.4'de gösterilen not defteri dosyası formatında hazırlanmış olan not defteri dosyasının son 6 satırı test için ayrılan veri kümesini içermektedir. Eğitim işleminde olduğu gibi bu değerler programda test işlemi için matris formatına dönüştürülmüştür. Matrislere dönüştürülen bu 6 adet test verisi, tasarlanan YSA

modelinde sadece simülasyon işlemine tabi tutulmuştur (Şekil 4.9). Simülasyon sonrasında ağdan elde edilen cevaplar ile gerçek deneysel değerler Tablo 4.4'de gösterilmiştir [5]. Gerçek değerler ile YSA modelinden elde edilmiş olan değerler ayrıca Şekil 4.10'da karşılaştırmalı grafik olarak gösterilmiştir.

```

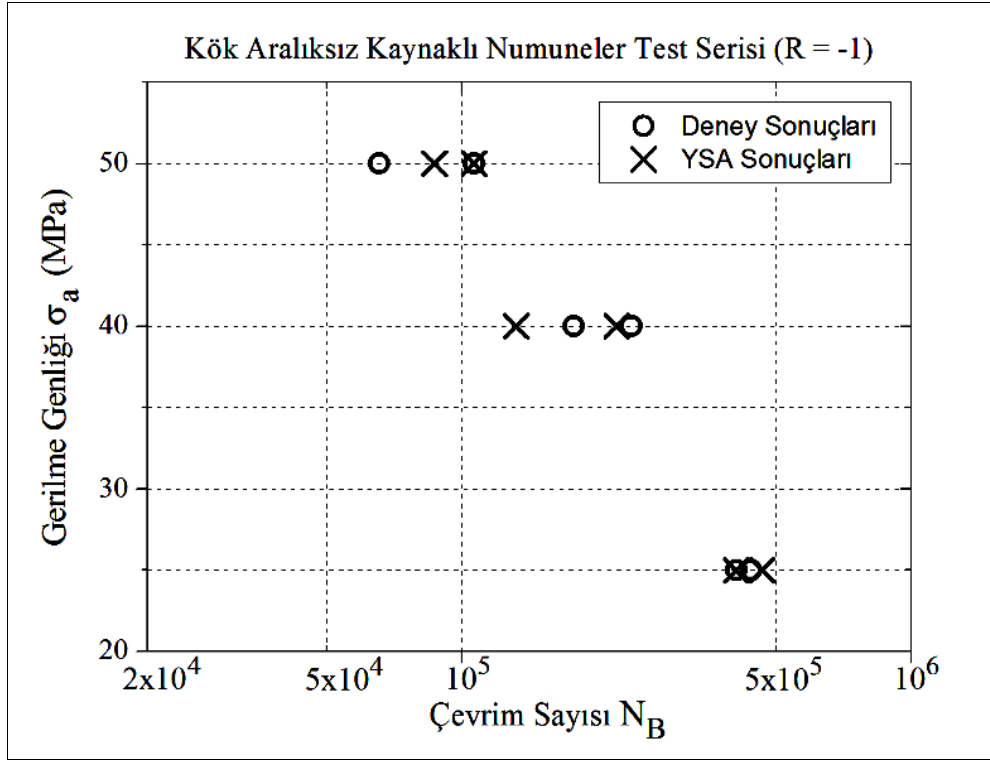
42 %YSA'nin test islemi
43 - ptest(2,6)=0.0;
44 - for i=1:6,
45 -     for j=1:2,
46 -         ptest(j,i)=network1(i+11,j);
47 -     end
48 - end
49
50 - ttest(1,6)=0;
51 - for i=1:6,
52 -     ttest(1,i)=network1(i+11,3);
53 - end
54
55 %Simulasyon
56 - at = sim(net,ptest);
57

```

Şekil 4.9: YSA'nın test işlemi

Tablo 4.4: Gerilme oranı $R = -1$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile test edilen YSA modelinden elde edilen değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,GERÇEK}$)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
-1	25	2,88	407432	407313
-1	25	2,68	440010	465119
-1	40	3,08	177354	132146
-1	40	1,92	238381	220760
-1	50	3,75	65535	87189
-1	50	1,97	106435	106458



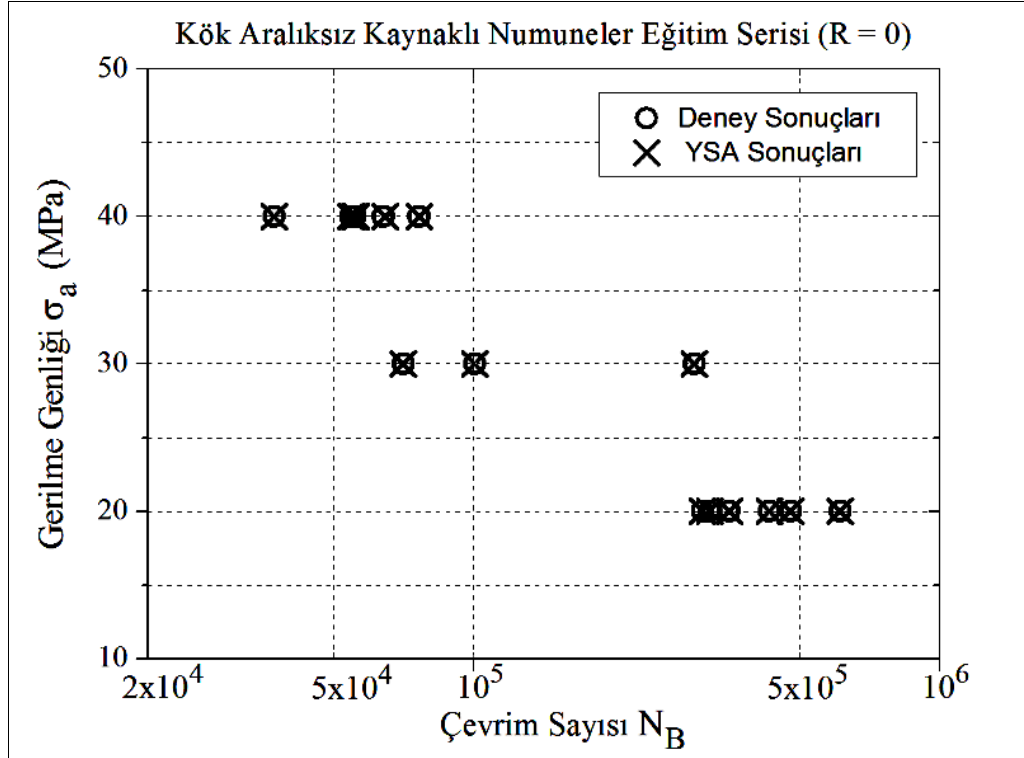
Şekil 4.10: Kök aralıksız kaynaklı numunelerin test serisi (R = -1)

YSA modelinin eğitim esnasında görmediği 6 adet deneysel veri ile gerçekleştirilen test işleminden sonra elde edilen değerler ile gerçek deneysel değerleri içeren Tablo 4.4 ve Şekil 4.10'daki grafik incelendiğinde, ağıın daha önce görmediği değerler için ürettiği cevapların gerçek değerlere yakın olduğu görülmektedir. Bu noktada ağıın eğitiminin bu grup için başarılı olduğunu söyleyebilmek mümkün olmaktadır.

İkinci deney grubu, gerilme genliği R = 0 olan kök aralıksız numunelere ait 19 adet deneysel veri, 13 adet eğitim ve 6 adet test verisi olarak ayrılmıştır. Uygun formatta düzenlenen not defteri dosyası sisteme okutularak eğitim için ayrılan değerler giriş ve çıkış matrislerine dönüştürülmüştür. Değerlerin ağına doğru bir şekilde sunulmasından sonra ağıın eğitim işlemi gerçekleştirilmiştir. Gerilme oranı R = 0 olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile bir önceki deney grubundan bağımsız bir şekilde eğitim işlemi gerçekleştirilen YSA modelinden elde edilen değerler Tablo 4.5'de gösterilmiştir. Ayrıca bu değerler ile gerçek değerler karşılaştırmalı grafik olarak Şekil 4.11'de görülmektedir.

Tablo 4.5: Gerilme oranı $R = 0$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile eğitilen YSA modelinden elde edilen değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
0	20	2,61	430687
0	20	2,30	478114
0	20	3,92	320450
0	20	1,79	610932
0	20	3,40	353648
0	20	4,15	308655
0	30	1,79	296979
0	30	4,15	70778
0	30	2,89	100492
0	40	1,79	76373
0	40	2,50	54598
0	40	4,15	37390
0	40	2,45	55609



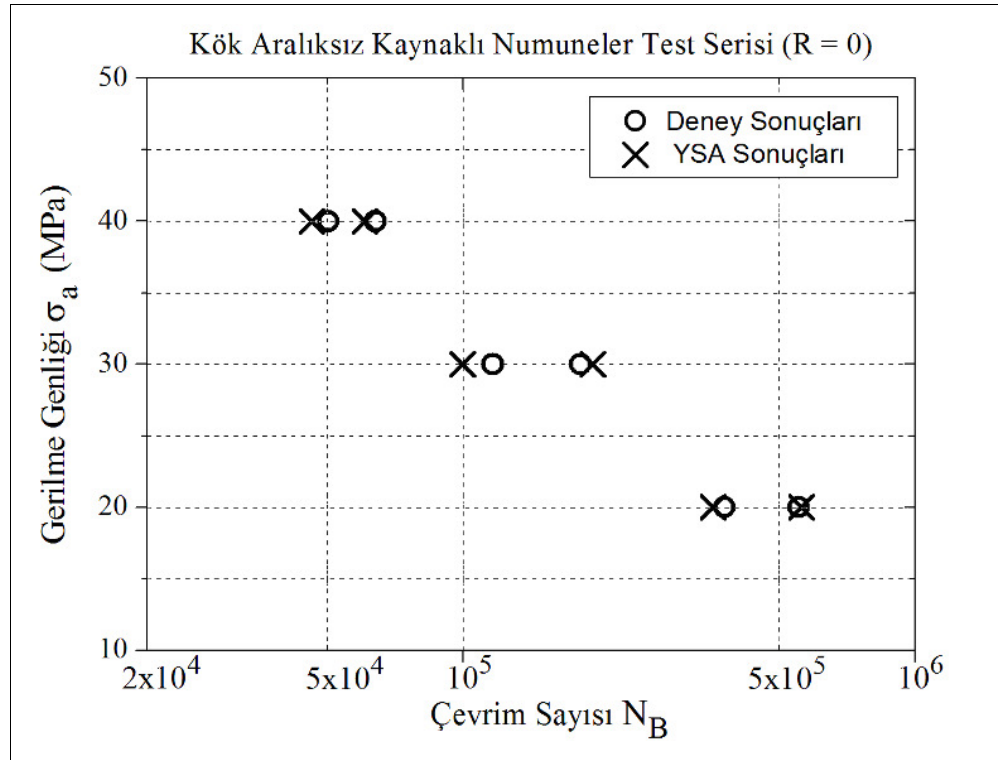
Şekil 4.11: Kök aralıksız kaynaklı numunelerin eğitim serisi (R = 0)

Eğitim işleminden elde edilen değerler ile gerçek değerlerin karşılaştırılması bir grafik olarak gösterildiği Şekil 4.11’de, eğitim işleminden elde edilen değerler ile gerçek değerlerin aynı olduğu görülmektedir. Bu işlemin ardından YSA modelinin test işlemi için ayrılan 6 adet veri, bir önceki çalışmada olduğu gibi matrislere

dönüştürülerek sistem için uygun formata getirilmiştir. Eğitim işlemi gerçekleştirilen YSA modeli bu değerler ile simülasyon işlemine tabi tutulmuştur. Simülasyon işleminden elde edilen değerler ile gerçek değerler [5] Tablo 4.6'da gösterilmiştir. Ayrıca bu değerler Şekil 4.12'de karşılaştırmalı grafik olarak gösterilmiştir.

Tablo 4.6: Gerilme oranı $R = 0$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile test edilen YSA modelinden elde edilen değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,GERÇEK}$)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
0	20	1,97	552848	560646
0	20	3,09	378586	356615
0	30	2,03	181788	193682
0	30	2,58	116272	99591
0	40	2,11	63970	60557
0	40	2,76	50291	46234



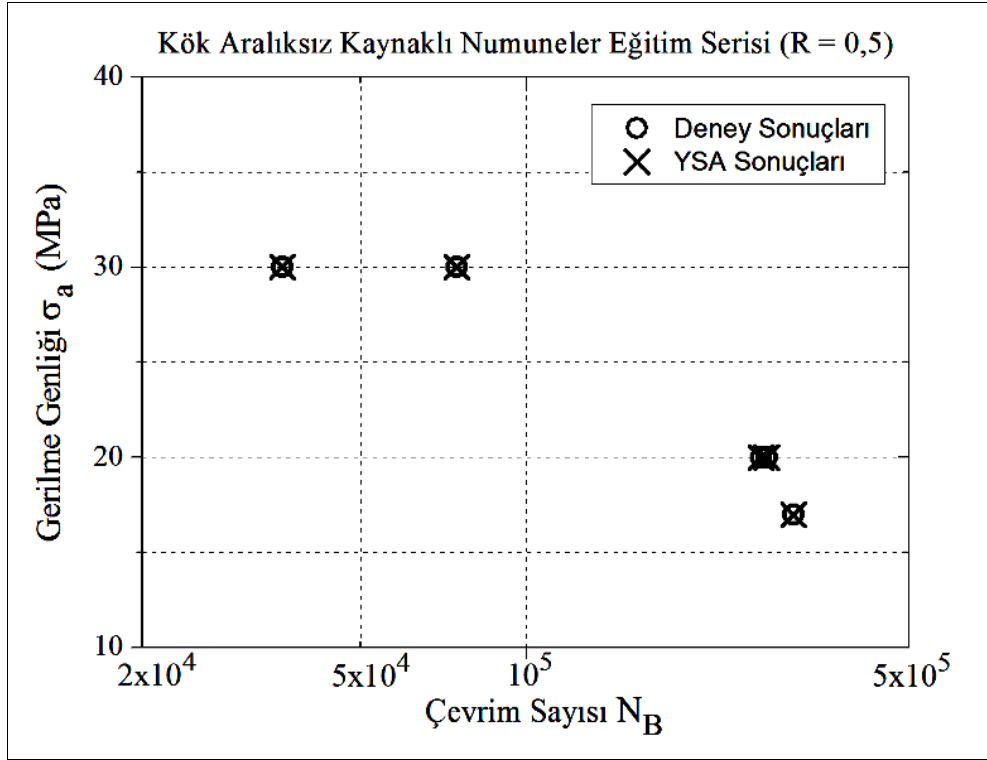
Şekil 4.12: Kök aralıksız kaynaklı numunelerin test serisi ($R = 0$)

Bu gruba ait 19 deneysel veriden 13'ü ile eğitilen ve 6'sı ile test işlemi gerçekleştirilen YSA modelinden elde edilen değerleri içeren Tablo 4.6 ve Şekil 4.12 incelendiğinde, YSA modelinden elde edilen değerler ile gerçek değerlerin birbirlerine oldukça yakın değerler olduğu açıkça görülmektedir. YSA modelinin eğitim işleminin bu grup için de ($R = 0$) başarılı bir şekilde gerçekleştiğini söyleyebiliriz.

Gerilme oranı $R = 0,5$ olan kök aralıksız kaynaklı numunelere ait elimizde 8 adet deneysel veri bulunmaktadır. Bu deneysel verilerin 5 adedi eğitim, 3 adedi test işlemi için ayrılmıştır. Uygun formatta hazırlanan not defteri dosyası vasıtasıyla programa okutulan bu veriler, YSA sisteminin eğitim işlemi için giriş ve çıkış matrislerine dönüştürülerek hazır hale getirilmiştir. YSA modelinin giriş ve çıkış matrislerine göre eğitim işlemi gerçekleştirilmiştir. Bu işlemden sonra ağ simülasyon işlemine tabi tutulmuş ve elde edilen değerler Tablo 4.7'de verilmiştir. Bu değerler aynı zamanda Şekil 4.13'de, gerçek değerler ile karşılaştırmalı bir grafik olarak gösterilmiştir.

Tablo 4.7: Gerilme oranı $R = 0,5$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile eğitilen YSA modelinden elde edilen değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
0,5	17	4,15	307545
0,5	20	3,31	274627
0,5	20	4,15	268418
0,5	30	4,15	35986
0,5	30	1,79	74830

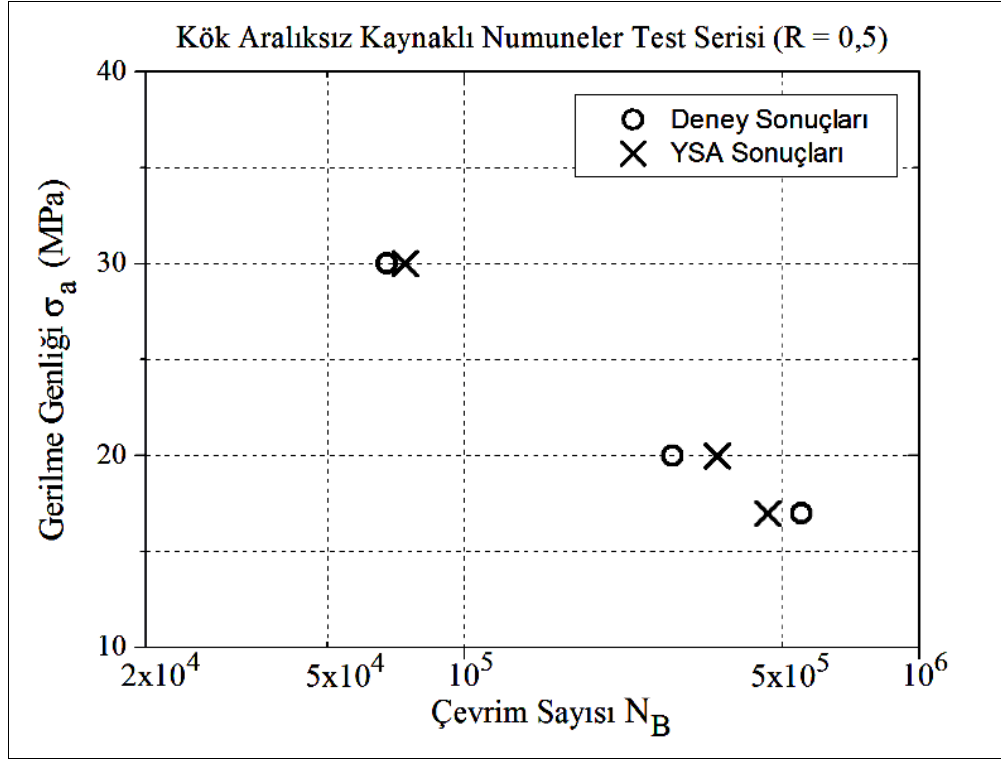


Şekil 4.13: Kök aralıksız kaynaklı numunelerin eğitim serisi (R = 0,5)

Gerilme oranı $R = 0,5$ olan kök aralıksız kaynaklı numunelere ait gerçek deneysel değerler ile YSA modelinden elde edilmiş değerlerin karşılaştırmalı bir grafik olarak gösterildiği Şekil 4.13 incelendiğinde, YSA modelinin eğitim işleminden sonra elde ettiği değerler ile deneysel değerlerin aynı olduğu net bir şekilde görülmektedir. Eğitim işleminden sonra, test işlemi için ayrılmış olan değerler matrislere dönüştürülerek ağı simülasyonu için uygun bir format haline getirilmiştir. Bu işlemden sonra bu değerler ile ağı simülasyonu gerçekleştirilmiştir. Ağı simülasyon işleminden elde edilen değerler ile gerçek değerler [5] Tablo 4.8'de verilmiştir. Tablo 4.8'de verilen değerler karşılaştırmalı bir grafik olarak Şekil 4.14'de gösterilmiştir.

Tablo 4.8: Gerilme oranı $R = 0,5$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile test edilen YSA modelinden elde edilen değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,GERÇEK}$)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
0,5	17	1,79	549536	464618
0,5	20	1,79	286971	359641
0,5	30	1,95	67752	74340



Şekil 4.14: Kök aralıksız kaynaklı numunelerin test serisi (R = 0,5)

Gerilme oran $R = 0,5$ olan kök aralıksız kaynaklı numunelere ait 8 adet deneysel veriden 5 tanesi ile eğitim işlemi gerçekleştirilen ve bu işlemten sonra, eğitim esnasında kendisinde gösterilmeyen 3 tane veriyle test edilen YSA modelinin Şekil 4.14 ve Tablo 4.8 incelendiğinde bu değerlere oldukça yakın değerler üretebildiğini söylemek mümkündür.

Kök aralıksız kaynaklı numuneler için tasarlanan ve gerçek deneysel veriler ile eğitimi gerçekleştirilen YSA modeli, eğitim esnasında sisteme gösterilmemiş bazı deneysel veriler ile test işlemine tabi tutulmuştur. Gerilme oranı $R = -1$, $R = 0$ ve $R = 0,5$ olmak üzere üç farklı grup olarak ayrı ayrı eğitimi ve hemen ardından simülasyonu gerçekleştirilen YSA modeli, nominal gerilme genliği σ_a ve çentik form sayısı K_t değerlerine karşılık N_B kırılma çevrim sayılarını tahmin etmiştir. Tahmin edilen bu değerler ile olması gereken gerçek kırılma çevrim sayılarının karşılaştırmalı olarak gösterildiği grafikler (Şekil 4.10, Şekil 4.12 ve Şekil 4.14) incelendiğinde, oluşturulan YSA mimarisinin magnezyumdan kök aralıksız kaynaklı numunelerin yorulma davranışları ile ilgili gerçek değerlere oldukça yakın değerler ürettiği görülmektedir. Test için kullanılan bu değerleri, üzerinde daha önceden deney yapılmamış numunelere ait değerler olarak kabul edecek olursak, YSA yaklaşımı ile magnezyumdan kök aralıksız kaynaklı numunelere ait bir örnek

Tablo 4.9: Nominal gerilme genliđi 30 MPa ve gerilme oranı $R = 0$ olan köşe dikişli kaynaklı numunelere ait deneysel veriler [5]

Nominal Gerilme Genliđi (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı (N_B)
30	4.50	207707
30	2.31	295339
30	1.62	359078
30	1.58	364448
30	2.68	271990
30	2.08	312472

4.3.2 Verilerin programa okutulması

Köşe dikişli kaynaklı numunelere ait toplam 64 adet veri bir önceki bölümde Şekil 4.4'de gösterilen not defteri dosyası formatında, gerilme oranlarına göre üç farklı grup olarak ($R = -1$, $R = 0$ ve $R = 0,5$) satır ve sütunlar şeklinde hazırlanmıştır. Tasarlanmış olan YSA modeli ile bilgisayarda aynı dosya adresinde bulunan bu dosyalar matlab programına okutularak program belleđine alınmıştır. Belleđe alından bu dosyalar YSA modelinin eğitimi için kullanılacak p giriş ve t çıkış matrislerine dönüştürülerek eğitime hazır forma getirilir (Şekil 4.16).

```
1 -   clc
2 -   clear all
3 -   load ('network1.txt');
4
5 -   p(2,13)=0.0;
6 -   for i=1:13,
7 -       for j=1:2,
8 -           p(j,i)=network1(i,j);
9 -       end
10 -   end
11
12 -   t(1,13)=0.0;
13 -   for i=1:13,
14 -       t(1,i)=network1(i,3);
15 -   end
16
```

Şekil 4.16: Verilerin program tarafından okunması, giriş ve çıkış matrislerinin oluşturulması

4.3.3 Yapay sinir ağı modelinin, eğitimi ve simülasyonu

Gerilme oranı $R = -1$, $R = 0$ ve $R = 0,5$ olmak üzere üç gruptan oluşan toplamdaki 64 adet veriden 42 tanesi eğitim, 22 tanesi test işlemi için ayrılmıştır. Gerilme oranlarına göre gruplandırılmış veriler YSA sisteminde kök aralıksız numunelerde olduğu gibi, her ayrı gerilme oranı grubu için birbirlerinden bağımsız olarak eğitim ve test işlemlerinde kullanılmıştır. Tasarlanmış olan YSA modelinin eğitimi sırasıyla gerilme oranı $R = -1$, $R = 0$ ve $R = 0,5$ olan gruplar için p giriş ve t çıkış matrislerine göre gerçekleştirilmiştir. YSA modelinin eğitiminden sonra, simülasyon işlemi gerçekleştirilmiş a değerleri elde edilmiştir (Şekil 4.17). YSA modelinden gerilme oranı $R = -1$ olan köşe dikişli kaynaklı numuneler için elde edilen bu a değerleri (N_B kırılma çevrim sayıları) Tablo 4.10'da verilmiştir.

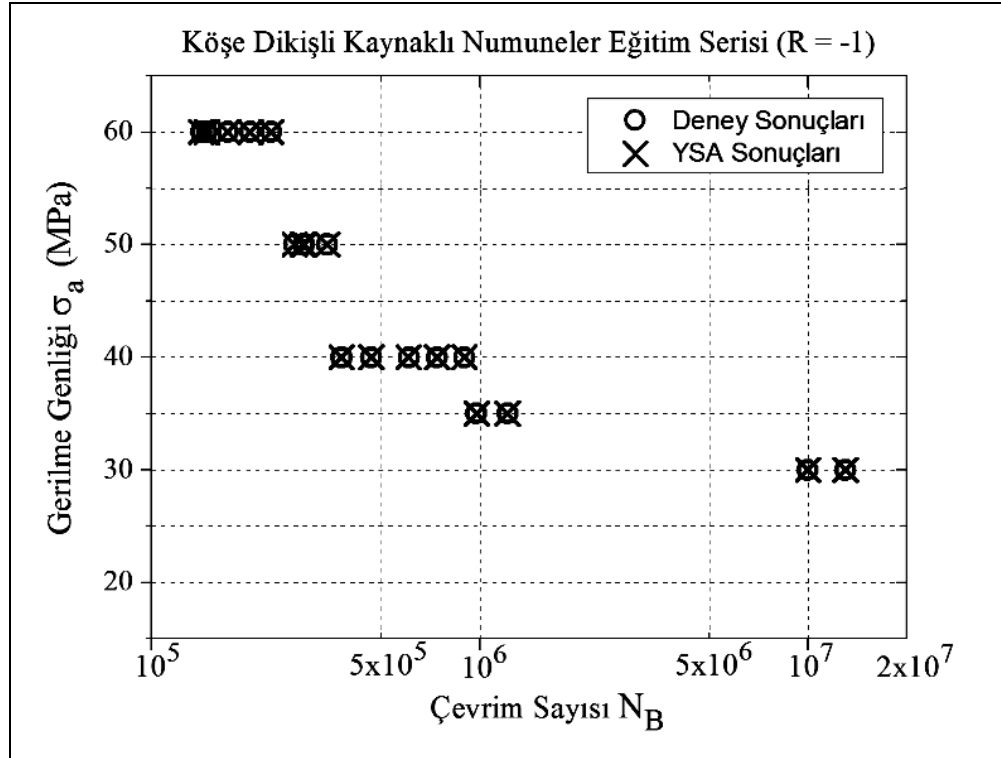
```
19 %YSA'nın tasarımı, eğitimi ve simülasyonu
20 - net = newff(p,t,6,{'tansig' 'purelin'},'trainlm');
21 - net.trainParam.epochs = 10000;
22 - net.trainParam.goal = 100;
23 - net.trainParam.lr =0.05;
24 - net.trainParam.mc = 0.9;
25 - net.divideFcn = 'dividerand';
26 - net.divideParam.trainRatio = 1;
27 - net.divideParam.valRatio = 0.00;
28 - net.divideParam.testRatio = 0.00;
29 - net.trainParam.mu_max = inf;
30 - net.trainParam.max_fail = inf;
31 - net = train(net,p,t);
32 - a = sim(net,p);
33
```

Şekil 4.17: YSA'nın tasarımı, eğitimi ve simülasyonu

Gerilme oranı $R = -1$ olan köşe dikişli kaynaklı numunelere ait 28 adet deneysel veriden 18 tanesi ile Şekil 4.17'de gösterildiği gibi eğitim ve simülasyon işlemi gerçekleştirilen YSA modelinden elde edilen değerler Tablo 4.10'da verilmiştir. Bu değerler ile gerçek deneysel değerler ayrıca karşılaştırmalı bir grafik olarak Şekil 4.18'de gösterilmiştir.

Tablo 4.10: Gerilme oranı $R = -1$ olan köşe dikişli kaynaklı numunelere ait deneysel veriler ile eğitilen YSA modelinden elde edilen değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
-1	30	4,49	10000000
-1	30	1,58	13000000
-1	35	4,49	975846
-1	35	1,58	1216796
-1	40	3,27	466924
-1	40	2,32	607835
-1	40	1,87	742042
-1	40	1,58	896369
-1	40	4,49	378372
-1	50	4,49	273030
-1	50	3,51	291645
-1	50	1,58	343357
-1	50	3,64	289010
-1	60	4,49	141731
-1	60	4,12	147661
-1	60	3,05	170440
-1	60	2,18	199367
-1	60	1,58	231013



Şekil 4.18: Köşe dikişli kaynaklı numunelerin eğitim serisi (R = -1)

Gerilme oranı $R = -1$ olan köşe dikişli kaynaklı numuneler için gerçekleştirilmiş eğitim ve simülasyon işlemi sonucunda elde edilen değerler ile eğitimde kullanılan gerçek değerler arasında ilişki Şekil 4.18'de açıkça görülmektedir. Tasarlanan YSA modeli, eğitim aşamasında gerilme oranı $R = -1$ olan köşe dikişli kaynaklı numuneler için eğitim değerlerine oldukça yakın değerler elde etmiştir. Fakat ağı test işlemine tabi tutmadan eğitimin başarısı hakkında bilgi vermek mümkün değildir.

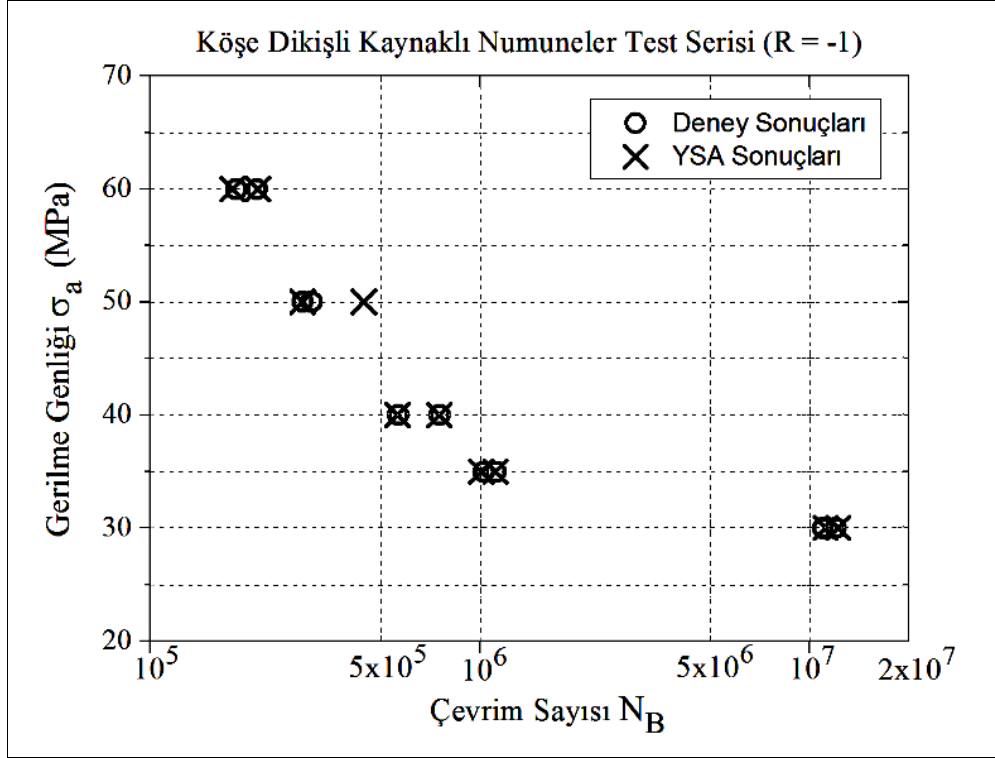
Tasarlanan YSA modelinin, gerilme oranı $R = -1$ olan köşe dikişli kaynaklı numuneler için toplamda 28 adet veriden 18 tanesi ile gerçekleştirilen eğitim işlemi sonrasında, eğitim esnasında görmediği 10 adet veri ile test işlemi gerçekleştirilmiştir. Test işlemi, eğitim işleminde olduğu gibi değerlerin matrislere dönüştürülerek ağa sunulmak için uygun bir forma getirilmesinden sonra, YSA modelinin simülasyon işlemine tabi tutulması şeklinde gerçekleştirilmiştir (Şekil 4.19). YSA modelinden elde edilen değerler ile gerçek deneysel çalışmalardan elde edilen değerler Tablo 4.11'de verilmiştir. Bu değerler ayrıca karşılaştırmalı bir grafik olarak Şekil 4.20'de gösterilmiştir.

```
42 %YSA'nin test islemi
43 - ptest(2,10)=0.0;
44 - for i=1:10,
45 -     for j=1:2,
46 -         ptest(j,i)=network1(i+18,j);
47 -     end
48 - end
49
50 - ttest(1,10)=0;
51 - for i=1:10,
52 -     ttest(1,i)=network1(i+18,3);
53 - end
54
55 %Simulasyon
56 - at = sim(net,ptest);
57
```

Şekil 4.19: YSA sisteminin test işlemi

Tablo 4.11: Gerilme oranı $R = -1$ olan köşe aralıksız kaynaklı numunelere ait deneysel veriler ile test edilen YSA modelinden elde edilen değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,GERÇEK}$)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
-1	30	2,31	12000000	12222217
-1	30	3,25	11000000	11187674
-1	35	2,59	1114680	1113514
-1	35	3,66	1029596	1010992
-1	40	1,84	754843	754703
-1	40	2,53	566193	562642
-1	50	2,76	308821	444862
-1	50	3,63	289282	289219
-1	60	2,64	182590	177756
-1	60	1,95	210126	211699



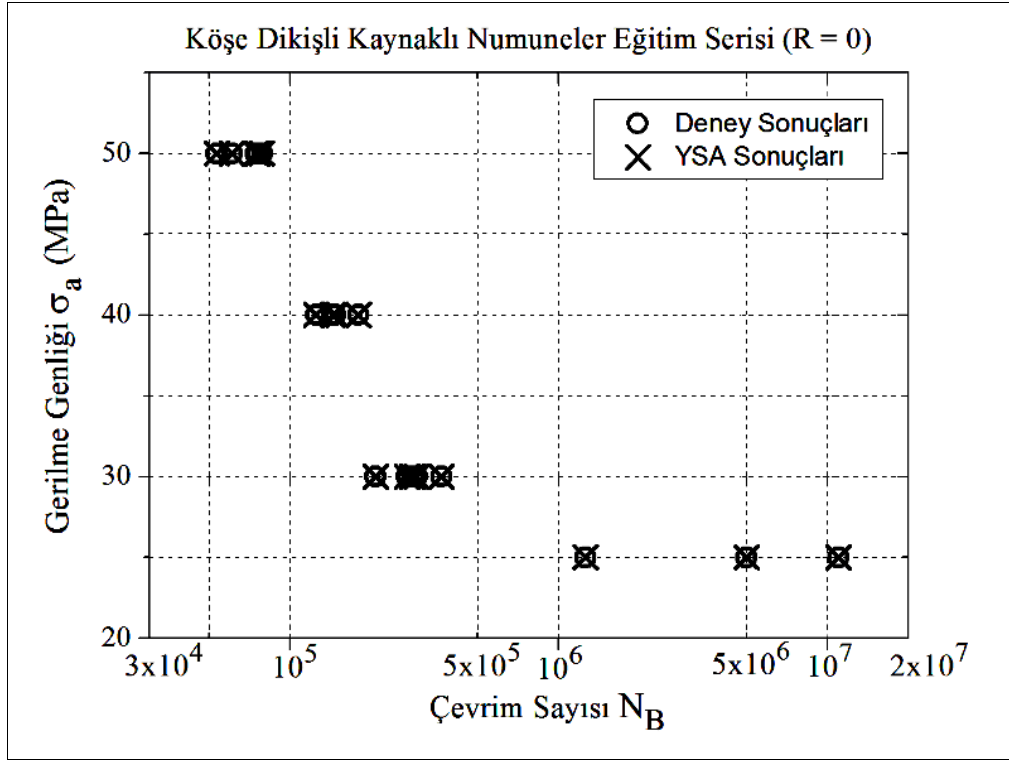
Şekil 4.20: Köşe dikişli kaynaklı numunelerin test serisi (R = -1)

Gerilme oranı $R = -1$ olan köşe dikişli kaynaklı numuneler için test işlemi gerçekleştirilen YSA modelinden elde edilen değerler ile gerçek deneysel değerleri içeren Tablo 4.11 ve Şekil 4.20'deki grafik incelendiğinde, test işleminden elde edilen değerlerin gerçek değerlere oldukça değerler olduğu görülmektedir. Sonuç olarak, birinci grup olan ($R = -1$) bu deney serisi için, tasarlanan YSA modelinin başarılı sonuçlar üretebildiği söylenebilir.

Köşe dikişli kaynaklı numunelere ait gerilme oranı $R = 0$ olan ikinci deney grubuna ait toplamda 25 adet deneysel veri 17 adet eğitim, 8 adet test verisi olarak ayrılmıştır. İlk olarak ağ eğitim için ayrılan 17 adet veri ile eğitilmiş ve ardından simülasyon işlemine tabi tutulmuştur. Simülasyon işleminden sonra elde edilen değerler Tablo 4.12’de verilmiş ve eğitimde kullanılan gerçek deneysel verilerle karşılaştırmalı bir grafik olarak Şekil 4.21’de gösterilmiştir.

Tablo 4.12: Gerilme oranı $R = 0$ olan köşe dikişli kaynaklı numunelere ait deneysel veriler ile eğitilen YSA modelinden elde edilen değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
0	25	1,73	5000000
0	25	4,49	1255713
0	25	1,58	11000000
0	30	4,49	207707
0	30	2,31	295338
0	30	1,58	364448
0	30	2,69	271989
0	30	2,48	284490
0	40	4,49	125092
0	40	3,10	144131
0	40	2,91	147356
0	40	1,58	179568
0	50	4,49	53223
0	50	1,85	74698
0	50	3,24	60790
0	50	1,77	75750
0	50	1,58	78628



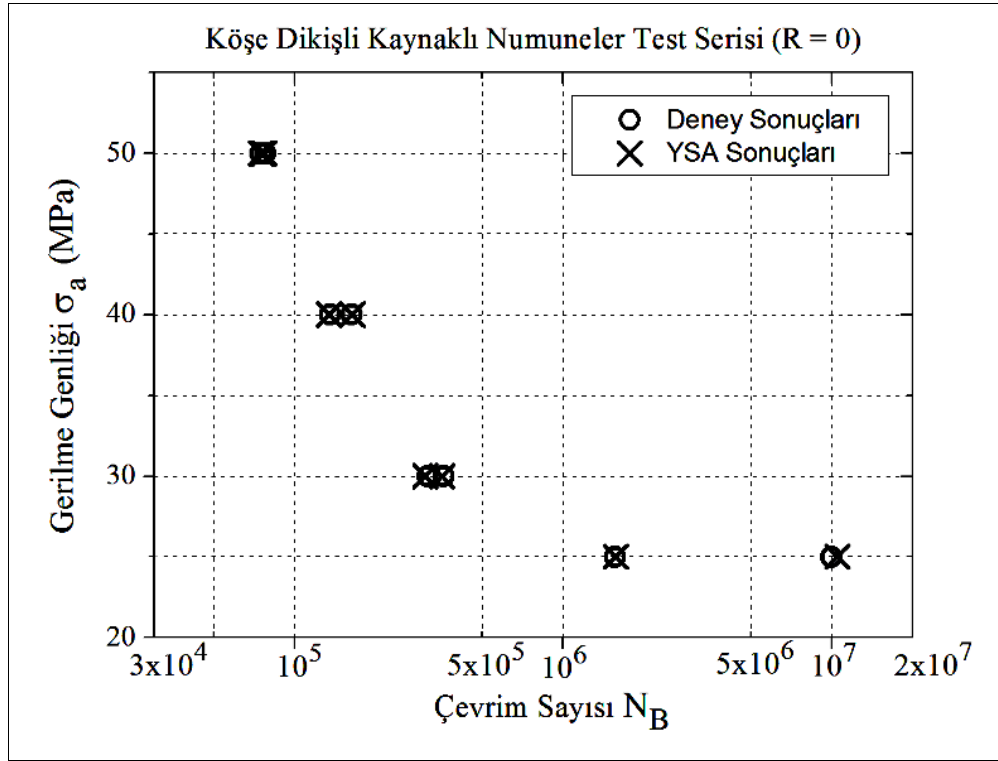
Şekil 4.21: Köşe dikişli kaynaklı numunelerin eğitim serisi (R = 0)

Gerilme oranı $R = 0$ olan köşe dikişli kaynaklı numunelerden elde edilmiş deneysel veriler ile bu deneysel veriler vasıtasıyla eğitilen YSA modelinden elde edilen değerleri Şekil 4.21 yardımıyla karşılaştıracak olursak, sonuçların bu deney serisi için de doğru bir şekilde elde edilebildiğini söyleyebiliriz. Grafikte de açıkça görüldüğü gibi YSA sistemi, eğitim esnasında kullanılan verilere oldukça yakın değerleri elde edebilmiştir. Bu YSA sisteminin, daha önce görmediği değerlere yorum getirebilme yeterliliğinin söz konusu olduğu test aşaması sonrasında eğitimin başarısı daha net ve doğru bir şekilde yorumlanabilir.

YSA modeli, gerilme oranı $R = 0$ olan köşe dikişli kaynaklı numuneler için toplamda 25 adet veriden 17'si ile eğitim işlemi gerçekleştirildikten sonra eğitim esnasında görmediği 8 adet veri ile test işlemine tabi tutulmuştur. Test işlemi sonrasında YSA modelinden elde edilen değerler ile gerçek deneysel çalışmalardan elde edilen değerler Tablo 4.13'de verilmiş ve karşılaştırmalı bir grafik olarak Şekil 4.22'de gösterilmiştir.

Tablo 4.13: Gerilme oranı $R = 0$ olan köşe aralıksız kaynaklı numunelere ait deneysel veriler ile test edilen YSA modelinden elde edilen değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,GERÇEK}$)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
0	25	3,46	1558574	1576339
0	25	1,59	10000000	10510224
0	30	1,62	359078	353873
0	30	2,08	312472	307021
0	40	2,17	162932	164064
0	40	3,67	135295	134531
0	50	1,63	77821	76786
0	50	1,86	74532	74942



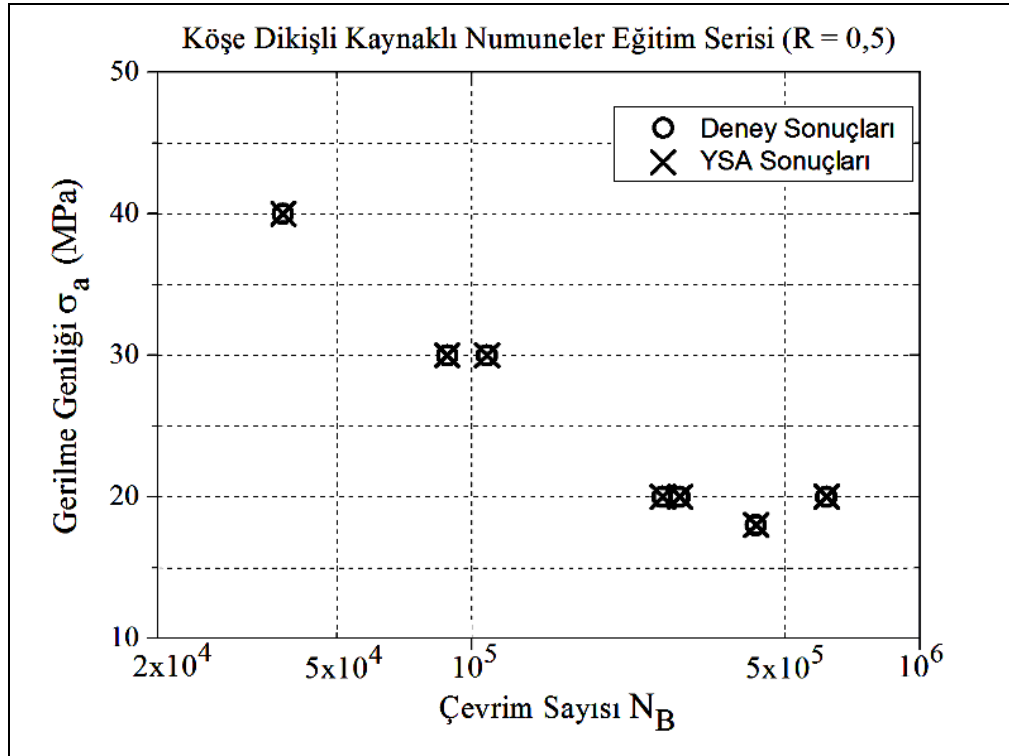
Şekil 4.22: Köşe dikişli kaynaklı numunelerin test serisi ($R = 0$)

Tablo 4.13 ve Şekil 4.22 incelendiğinde, gerilme oranı $R = 0$ olan köşe dikişli kaynaklı numunelere ait deneysel veriler ile eğitim ve test işlemi gerçekleştirilen YSA modelinden elde edilen değerlerin, deneysel değerlere çok yakın olduğu görülmektedir. İkinci grup olan bu deney serisi için de, tasarlanan YSA modelinin eğitim işleminin başarılı olduğu söylenebilir.

Son olarak 11 adet veriden oluşan gerilme oranı $R = 0,5$ olan köşe dikişli kaynaklı numunelerden elde edilmiş verilerin 7 tanesi eğitim, 4 tanesi test işlemi için ayrılmıştır. Tasarlanmış olan YSA modelinin eğitim işlemi gerçekleştirildikten sonra ağ simülasyon işlemine tabi tutulmuştur. Ağın simülasyonundan elde edilen değerler ile eğitimde kullanılan değerler Tablo 4.14’de verilmiş ve bu değerler karşılaştırmalı bir grafik olarak Şekil 4.23’de gösterilmiştir.

Tablo 4.14: Gerilme oranı $R = 0,5$ olan köşe dikişli kaynaklı numunelere ait deneysel veriler ile eğitilen YSA modelinden elde edilen değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
0,5	18	4,49	430569
0,5	20	1,58	618922
0,5	20	4,49	266730
0,5	20	3,92	291014
0,5	30	1,58	108154
0,5	30	4,49	88134
0,5	40	1,58	38071



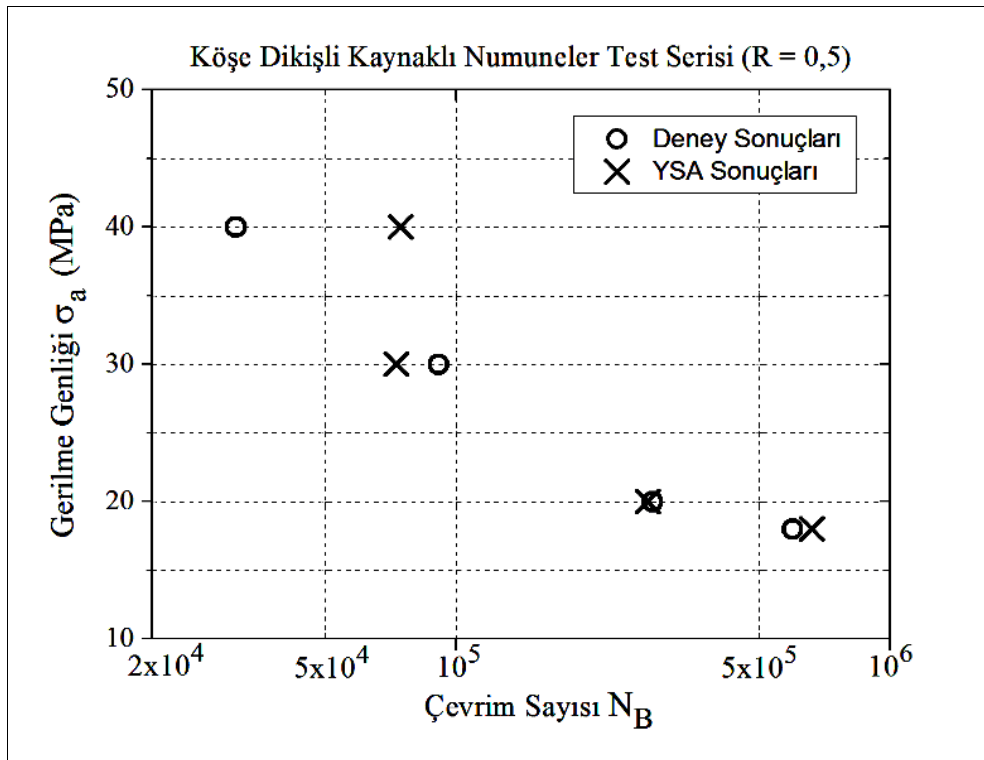
Şekil 4.23: Köşe dikişli kaynaklı numunelerin eğitim serisi ($R = 0,5$)

Gerilme genliği $R = 0,5$ olan deneysel veriler ile eğitimi gerçekleştirilen YSA sisteminden elde edilen değerler ile eğitim işlemi kullanılan değerlerin karşılaştırmalı olarak gösterildiği Şekil 4.23 incelendiğinde, YSA sisteminden elde edilen cevapların köşe dikişli kaynaklı numunelere ait diğer iki deney grubu ile aynı şekilde sonuçlandırıldığı görülmektedir.

Köşe dikişli kaynaklı numunelere ait deneysel verilerden gerilme oranı $R = 0,5$ olan son grup için bu gruba ait 11 adet deneysel veriden 4 tanesiyle test işlemi gerçekleştirilen YSA modelinden elde edilen değerler ile gerçek deneysel değerler Tablo 4.15’de verilmiştir. Bu değerler ayrıca karşılaştırmalı bir grafik olarak Şekil 4.24’de gösterilmiştir.

Tablo 4.15: Gerilme oranı $R = 0,5$ olan kök aralıksız kaynaklı numunelere ait deneysel veriler ile test edilen YSA modelinden elde edilen değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,GERÇEK}$)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
0,5	18	1,58	596345	659254
0,5	20	4,08	283500	276256
0,5	30	3,91	91266	72875
0,5	40	4,49	31164	74736

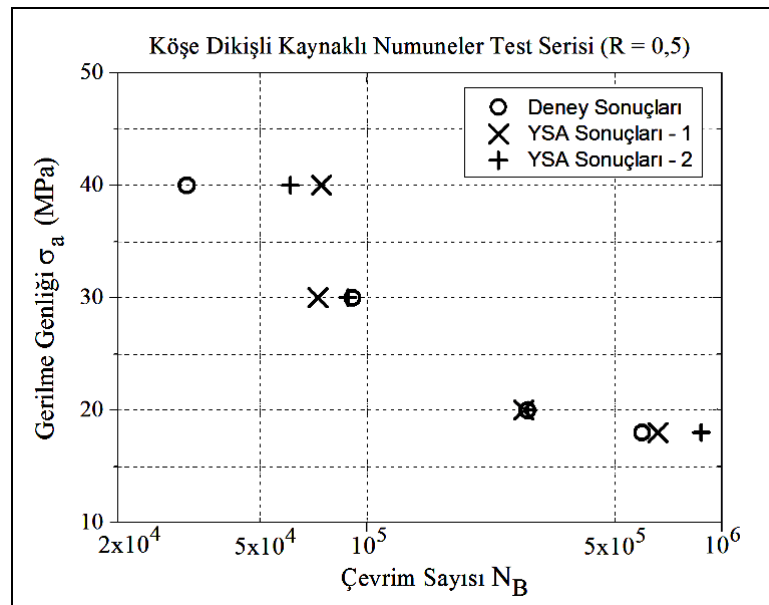


Şekil 4.24: Köşe dikişli kaynaklı numunelerin test serisi (R = 0,5)

Grafik ve ilgili tablo (Tablo 4.15) incelendiğinde, gerilme oranı $R = 0,5$ olan köşe dikişli kaynaklı numunelerin test işleminden elde edilen gerilme genliği 40 MPa olan değer, deneysel değerden diğerlerine göre daha uzak olduğu görülmektedir. Bunun nedeninin eğitim esnasında ağa, nominal gerilme genliği 40 MPa olan sadece bir adet verinin gösterilmiş olması olarak yorumlanabilir. Fakat 18 MPa olan değer için de aynı durum söz konusu iken, test işleminde bu değer gerçek değere yakın çıkmasının nedenini izah etmek gerekirse; tekrar tekrar yapılan test işlemlerinde gözlemlenen bir durumdan söz etmek açıklayıcı olacaktır. Bazı test işlemlerinde nominal gerilme genliği 40 MPa olan test değerleri gerçek değerlere yaklaşırken, nominal gerilme genliği 18 MPa olan test değerlerinin de gerçek değerlerden uzaklaştığı gözlemlenmiştir. Buna örnek olarak Tablo 4.16'da iki farklı test işlemi için karşılaştırmalı sonuçlar verilmiştir. Ayrıca örnek teşkil eden bu iki farklı test işleminden elde edilen sonuçlar karşılaştırmalı bir grafik olarak Şekil 4.25'de gösterilmiştir.

Tablo 4.16: Gerilme oranı $R = 0,5$ olan köşe dikişli kaynaklı numunelere ait iki farklı test işleminden elde edilen değerlerin karşılaştırılması

Nominal Gerilme Genliği (σ_a)	Çentik Form Sayısı (K_t)	Kırılma Çevrim Sayısı ($N_{B,GERÇEK}$)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
18	1,58	596345	659254	874652
40	4,49	31164	74736	60894



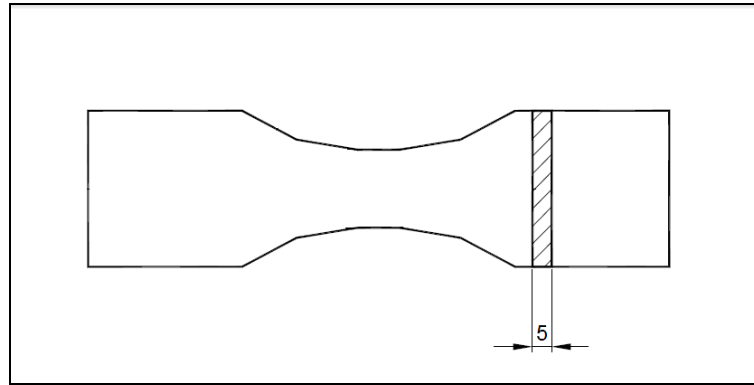
Şekil 4.25: Gerilme oranı $R = 0,5$ olan köşe dikişli kaynaklı numunelere ait iki farklı test işleminden elde edilen değerlerin karşılaştırılması

Şekil 4.25 incelendiğinde, anlatılan durum açıkça görülmektedir. İkinci test işleminden elde edilen, nominal gerilme genliği 40 MPa olan numune için kırılma çevrim sayısı gerçek değerine yaklaşırken, nominal gerilme genliği 18 MPa olan numune için tam tersi bir durum söz konusudur. Tablo 4.16'daki değerler ile yapılan matematiksel hesaplama neticesinde, nominal gerilme genliği 40 MPa olan numune için kırılma çevrim sayısı gerçek değerine %44,4 yaklaşırken, nominal gerilme genliği 18 MPa olan numune için kırılma çevrim sayısı gerçek değerinden %36,1 oranında uzaklaşmıştır.

Köşe dikişli kaynaklı numunelere ait deneysel veriler ile birbirlerinden bağımsız olarak üç grup şeklinde ($R = -1$, $R = 0$, $R = 0,5$) gerçekleştirilen eğitim ve test işlemleri neticesinde, ağı eğitiminin başarısı ile ilgili olumlu bir yargıya varılabilir. Çünkü test için kullanılan veriler daha önce üzerinde deneysel çalışma yapılmamış numunelere ait veriler olarak ele alındığında, köşe dikişli kaynaklı numuneler için, tasarlanan YSA modelinin yapılması planlanan ve şartları bilinen deneylere yakın sonuçları başarılı bir şekilde üretebildiği açıkça görülmektedir.

4.4 Kaynaksız Numunelere Ait Yorulma Davranışlarının YSA Yöntemiyle İncelenmesi

Üçüncü ve son örnek serisi olan kaynaksız numunelerin şematik yapısı Şekil 4.26'da gösterilmiştir.



Şekil 4.26: Kaynaksız numunelerin şematik yapısı [5]

Kaynaksız numunelere ait, gerilme oranı $R = -1$ ve $R = 0$ olmak üzere iki grup halinde 28 adet veri bulunmaktadır. Bu verilerdeki çentik form sayıları önceki iki çalışmadan farklı olarak sabit ($K_t = 1$) olmaktadır. Bu da elimizdeki en önemli değişkenlerden bir tanesinin olmaması anlamına gelmektedir. Tasarlanan YSA modelinin eğitim işlemi sadece nominal gerilme genliği (σ_a) değerine bağlı olacaktır. Bu noktada açıkcası kullanılacak olan girdi değerlerinin yetersiz olduğu düşünülmektedir. Çünkü aynı nominal gerilme genliğinde oldukça büyük farklılık gösteren kırılma çevrim değerleri mevcuttur ve tasarlanan YSA modelinin bu değer ile eğitiminden elde edilecek sonuçlar merak konusu olmaktadır.

4.4.1 Problemin bilgisayar ortamına aktarılması

Kaynaksız numunelere ait veriler için değişken bir K_t çentik form sayısı söz konusu olmadığı için daha önceki diğer üç örnekteki gibi hesaplama işlemine gerek duyulmamaktadır. Bu nedenle değerler direkt olarak not defteri dosyasında daha önceki çalışmalarda olduğu gibi satır ve sütunlar şeklinde uygun formatta hazırlanarak programa sunulabilecek bir hale getirilir.

4.4.2 Verilerin programa okutulması

Uygun formatta hazırlanan not defteri dosyasındaki veriler program tarafından okularak program belleğine alınır. Önceki çalışmalarda olduğu gibi bu veriler p giriş ve t çıkış matrislerine dönüştürülerek YSA modelinin eğitim işleminde kullanılmak üzere uygun bir formata getirilmiştir (Şekil 4.27).

```
1 -   clc
2 -   clear all
3 -   load ('network1.txt');
4
5 -   p(2,13)=0.0;
6 -   for i=1:13,
7 -       for j=1:2,
8 -           p(j,i)=network1(i,j);
9 -       end
10 -    end
11
12 -   t(1,13)=0.0;
13 -   for i=1:13,
14 -       t(1,i)=network1(i,3);
15 -   end
16
```

Şekil 4.27: Verilerin program tarafından okunması, giriş ve çıkış matrislerinin oluşturulması

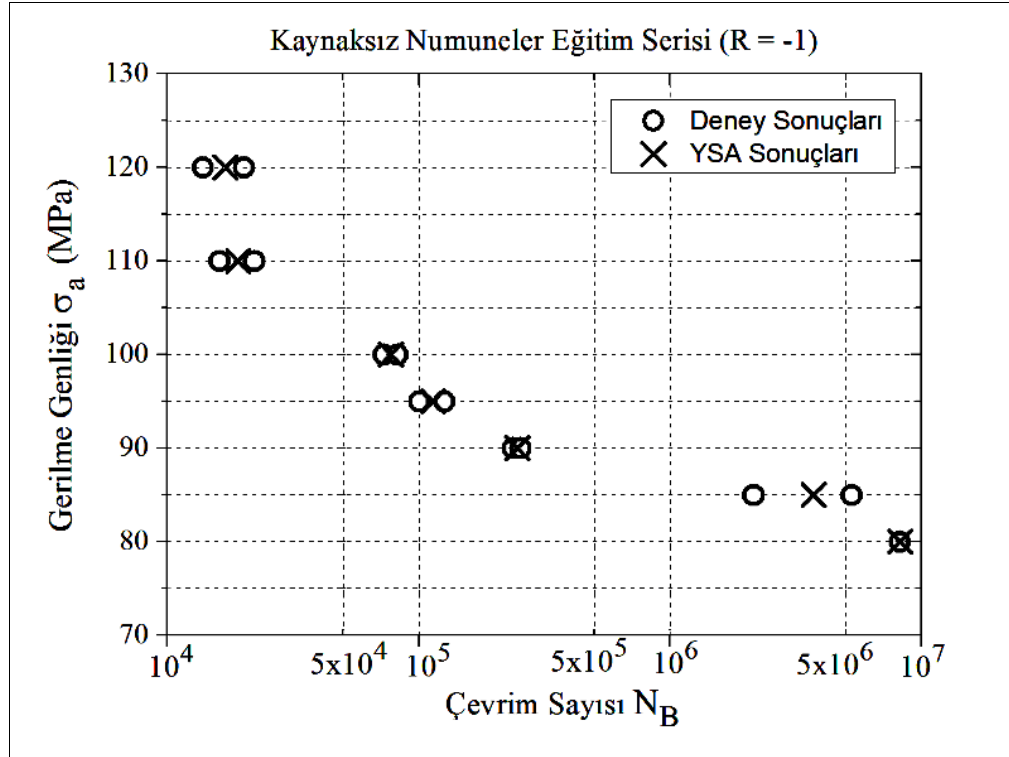
4.4.3 Yapay sinir ağı modelinin, eğitimi ve simülasyonu

Gerilme oranlarına göre iki gruptan oluşan ($R = -1$, $R = 0$) kaynaklı numunelere ait deneysel veriler, tasarlanmış olan YSA modelinde birbirlerinden bağımsız olarak eğitim işlemine tabi tutulmuştur. Kaynaklı numuneler için YSA modelinden elde edilen değerler Tablo 4.17’de gösterilmiştir.

Tablo 4.17: Kaynaklı numunelere ait deneysel verilerle programlanan YSA modelinden elde edilmiş olan değerler

Gerilme Oranı (R)	Nominal Gerilme Genliği (σ_a)	Kırılma Çevrim Sayısı ($N_{B,YSA}$)
-1	80	8239720
-1	85	3717440
-1	85	3717440
-1	90	246236
-1	90	246236
-1	95	114018
-1	95	114018
-1	100	77558
-1	100	77558
-1	110	19152
-1	110	19152
-1	120	17011
-1	120	17011
0	60	5120045
0	60	5120045
0	65	5038620
0	70	2433839
0	70	2433839
0	80	159564
0	80	159564
0	90	48336
0	90	48336
0	100	36135
0	100	36135
0	100	36135
0	100	36135
0	100	36135
0	100	36135

Tablo 4.13'deki gerilme oranı $R = -1$ olan kaynaklı numunelere ait 16 adet veri ile gerçekleştirilmiş eğitim işleminden elde edilen sonuçlar incelendiğinde diğer iki çalışmadan çok daha farklı bir durum göze çarpmaktadır. Dikkat edilirse, aynı nominal gerilme genlik değerleri σ_a için ağın verdiği cevaplar aynı olmaktadır. Bu duruma bir yorum getirmeden önce gerilme genliği $R = -1$ olan numunelere ait gerçek deneysel veriler ile YSA sisteminden elde edilmiş olan değerlerin karşılaştırmalı bir grafik olarak gösterilmesi faydalı olacaktır (Şekil 4.28).

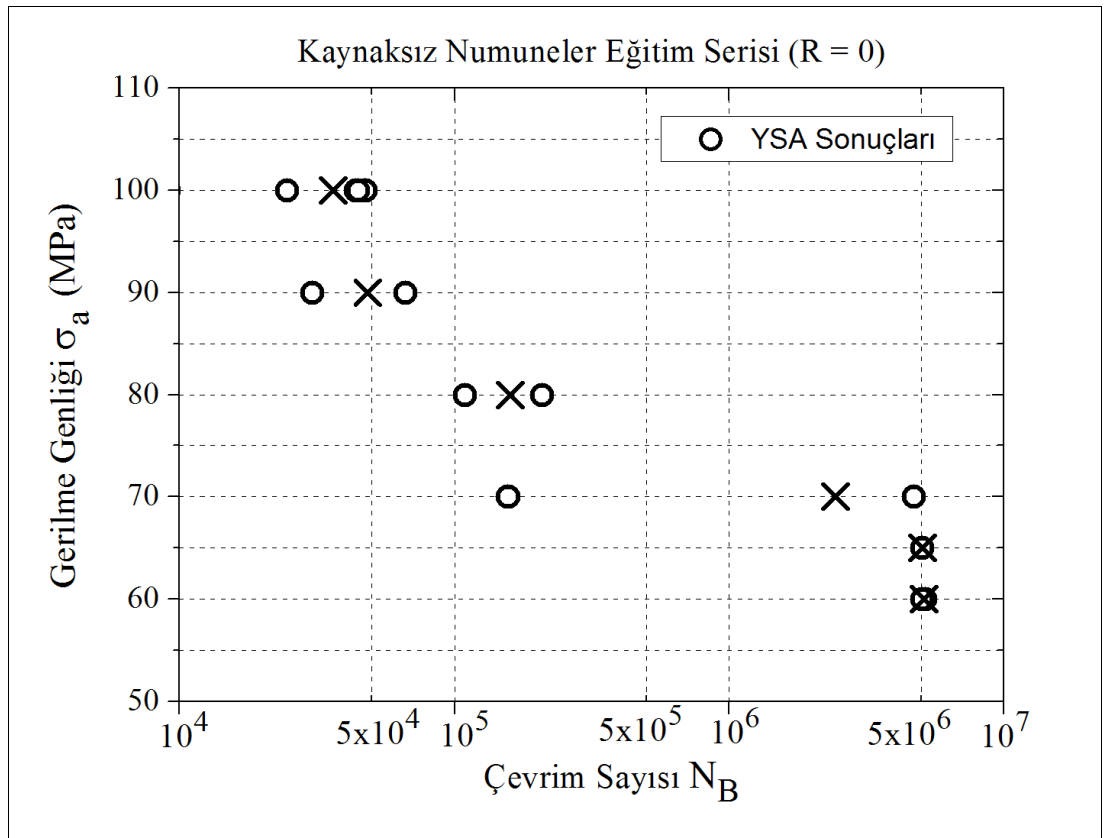


Şekil 4.28: Kaynaklı numunelerin eğitim serisi ($R = -1$)

Tablo 4.10'da görülen, aynı σ_a nominal gerilme genlik değerlerine sahip numunelerin aynı N_B kırılma çevrim sayılarına sahip olmalarının nedenini daha sağlıklı bir şekilde yorumlayabilmek için çizilen Şekil 4.28'deki grafik incelendiğinde YSA modelinin, eğitimde kullanılan N_B kırılma çevrim sayısı değerlerinin ortalamalarını aldığı net bir şekilde görülmektedir. Bunun nedeni, eğitilmek istenilen YSA modelinin N_B kırılma çevrim sayılarının belirlenmesinde bir tek değişken olan nominal gerilme genliğinden başka daha önceki çalışmalarda olduğu gibi etkin bir değişken olan K_t gibi bir değişkene ihtiyaç duymasındır. Çünkü YSA modeli matematiksel işlemler yaparak istenilen değerlere yakın değerleri elde

eder. Eğer tasarladığımız modelde, istenilen değere etki eden birincil faktörleri sisteme gösteremezsek, elde edeceğimiz sonuç yukarıda görüldüğü gibi olacaktır.

Aynı durumun söz konusu olduğu gerilme oranı $R = 0$ olan kaynaklı numuneler için de bu işlemi gerçekleştirdiğimiz zaman durumda bir değişiklik görülmemektedir. Tablo 4.10'da görüldüğü gibi, $R = 0$ koşullarında da, aynı nominal gerilme genliği (σ_a) değerlerine sahip örneklerin N_B kırılma çevrim sayıları eşit olmaktadır. Gerilme oranı $R = 0$ olan numunelerden elde edilen değerler ile YSA modelinden elde edilen değerleri aynı şekilde bir grafik üzerinde gösterecek olursak, vardığımız yargının doğruluğu netlik kazanacaktır (Şekil 4.29).



Şekil 4.29: Kaynaklı numunelerin eğitim serisi ($R = 0$)

Tasarlanan YSA modelinin gerilme oranı $R = 0$ olan kaynaklı numunelere ait veriler ile gerçekleştirilen eğitim işleminden elde ettiği sonuçlar bir önceki gerilme oranı $R = -1$ olan kaynaklı numunelere ait veriler ile gerçekleştirilen eğitim işleminden elde ettiği sonuçlarla aynı nitelikte olmaktadır. Sonuç olarak, elde etmek istediğimiz değerler için birincil derecede rol oynayan değişkenleri (örneğin kaynaklı numuneler için K_t değerleri) yeterli bir şekilde sisteme girmezsek, tasarlayacağımız

YSA modeli Şekil 4.28 ve Şekil 4.29'da görüldüğü gibi, istediğimiz değerlerden uzak ve kayda değer olmayan sonuçlar üretecektir.

Kaynaksız numuneler için gerçekleştirilen eğitim aşamasında, eğitim algoritması, öğrenme katsayısı, ara katmandaki düğüm sayısı gibi değişkenlerin farklı kombinasyonlarının denenmesine rağmen bu sonuçlardan farklı sonuçlar elde edilememiştir. Bu nedenle kaynaklı numuneler için bu çalışmada bir test işlemi gerçekleştirilmemiştir. Çünkü eğitim işlemi dahi tamamlanamayan bir YSA modelinin test işlemine tabi tutulması gibi bir durum söz konusu olmamalıdır.

Gerilme oranları $R = 0$ ve $R = -1$ olan kaynaklı numunelere ait iki deney grubu ile gerçekleştirilen eğitim ve simülasyon işlemlerinden elde edilen sonuçlar, kaynaklı numuneler için YSA yaklaşımının kullanışlı bir yöntem olmadığını ve YSA sisteminden elde edilmesi istenen değerler için birincil derecede rol oynayan değişkenlerin, YSA tasarımında çok büyük bir önemi olduğunu göstermektedir.

5. SONUÇLAR VE ÖNERİLER

Yapılan çalışma hem sonuç olarak hem de çalışma içerisinde karşılaşılan bazı güçlükler ve bunların çözümleri açısından tecrübeye dayanan deneysel paylaşımların önemli olduğu bir çalışmadır. Çünkü YSA tasarımlarında birçok parametre ve mimari düzen deneme yanılma yöntemiyle belirlenmektedir. Fakat benzer problemlerin çözümlerinde kullanılan YSA modelleri, büyük benzerlikler gösterebilmektedir. Bu nedenle elde edilen sonuçlar ve bu çalışma süresince karşılaşılan bazı tecrübeler paylaşarak bu konuda çalışma yapan kişilere yardımcı olabilmek amaçlanmıştır.

5.1 Sonuçlar

Tasarlanan YSA modeli üç ayrı deney serisi ve bu serilerdeki farklı gerilme oranları için bağımsız bir şekilde eğitilmiş ve test edilmiştir. Öğretmenli öğrenme prensibiyle eğitilen YSA modeline, eğitim esnasında elde etmesi gereken N_B kırılma çevrim sayıları, bu değerlerin elde edilmesinde kullanılan girdiler olan σ_a nominal gerilme genlikleri ve K_t çentik form sayıları ile birlikte gösterilmiştir. Test işlemi için ayrılmış ve eğitim esnasında sisteme gösterilmemiş olan değerler, eğitimi tamamlanmış YSA modelinde simülasyon işlemine tabi tutulmuştur. Test işleminin sonucunda, YSA modelinden elde edilen değerler ile gerçek deneysel çalışmalardan elde edilen değerler karşılaştırmalı tablolar ve grafikler halinde gösterilmiştir.

Kök aralıksız ve köşe dikişli kaynaklı numunelerin yorulma ömürlerinin tahmininde kullanılmak üzere tasarlanan YSA modelinin test işleminden elde edilen değerler ile bu numunelere ait gerçek deneysel değerler karşılaştırıldığında, tasarlanan YSA modelinin eğitim aşamasından sonra, bu test değerlerine oldukça yakın ve değerlendirilebilir sonuçlar üretebildiği net bir şekilde görülmektedir. Daha önceden görmediği deneysel çalışmalara ait parametreler ile bu deneysel çalışmalardan elde edilmiş değerlere oldukça yakın değerler elde edebilen bir YSA modeli, deneysel çalışma sayısı ihtiyacının YSA yaklaşımıyla azaltılabileceğinin mümkün olduğunu göstermektedir. Eğer test için ayırdığımız verileri, daha önceden üzerinde deney

yapmamış olduğumuz numunelere ait değerler olarak ele alırsak, uygun şekilde eğitilmiş bir YSA modeli sayesinde, deney yapmadan deney yapmış gibi sonuçlar üretebilmemizin mümkün olduğunu söyleyebiliriz.

Bilindiği üzere malzemelerin yorulma davranışlarının incelenmesi amacıyla gerçekleştirilen yorulma deneyleri, oldukça maliyetli ve uzun bir zaman gerektiren çalışmalardır. Bu çalışmalardaki deney sayısını azaltabilecek bir YSA modelinin tasarlanabilmesi ekonomik açıdan bir tasarruf anlamına gelmektedir. Yapılan bu çalışma, deneysel çalışmalardan elde edilen verilerin YSA yaklaşımıyla kullanılarak, bilgisayar ortamında ve yüksek bir oranda doğruluğa sahip deney sonuçlarının üretilebileceğinin mümkün olduğunu göstermektedir.

Kaynaksız numuneler ile ilgili sonuçlar kaynaklı birleştirmelere ait diğer iki deney grubuna ait sonuçlar ile kıyaslandığında, kaynaklı numunelerin yorulma davranışlarının incelenmesinde YSA yaklaşımının uygun bir yöntem olmadığı açıkça görülmektedir. Bunun nedeni, kırılma çevrim sayısını etkileyen değişkenlerin kaynaklı numuneler için sadece nominal gerilme genliği (σ_a) olmasıdır. Diğer bir ifade ile çıkış değerlerini etkileyen giriş değerlerinin yetersiz olmasıdır. Çünkü YSA modeli, döngüler boyunca gerçekleştirdiği matematiksel işlemler ile istenilen değerlere ulaşmaya çalışan bir sistemdir ve her döngü sonrası hata değerini azaltmaya çalışarak istenilen değere yani bizim belirttiğimiz tolerans değerine yaklaşmaya çalışır. Fakat bu işlemleri gerçekleştirirken bizim girdi olarak verdiğimiz değişkenleri kullanır. Kaynaksız numuneler için değişken sadece nominal gerilme genliğidir. Bu değer, tek başına istediğimiz kırılma çevrim sayılarını elde edebilmek için yeterli olmamaktadır. Diğer kaynaklı numunelere ait örneklerdeki kırılma çevrim sayılarının tahmininde elde edilen başarılı sonucun nedeni, bu çalışmalarda geometrik yapıya bağlı çentik form sayılarının da giriş değerleri olarak kullanılmasıdır. Bu nedenle tasarlanan YSA modeli istenilen kırılma çevrim sayısı değerlerine oldukça yakın değerler üretebilmektedir.

Deneysel çalışmalardan elde edilen sonuçların YSA yaklaşımıyla incelenebilmesi için tasarlanan YSA modelinin eğitiminde deneysel çalışmalardan elde edilmiş değerlere ihtiyaç duyulmaktadır. Yani hiç deneysel çalışma yapmadan YSA yaklaşımında bulunabilmek şu anda mümkün olmamaktadır. Ortalama olarak bu

çalışmada kullanılan deneysel verilerin %70'i eğitim, %30'u test işleminde kullanılmıştır. Bu oranlar eldeki verilerin tutarlılığı veya problemin bağlı olduğu değişkenlere göre değişebilmektedir.

Sonuç olarak YSA yaklaşımı, düzgün bir YSA modelinin tasarlanabilmesiyle kaynaklı malzemelerin yorulma davranışlarını yorumlamak için oldukça başarılı bir yöntem olabilmektedir. Fakat bu yöntem kaynaksız malzemelerin yorulma davranışlarının incelenmesinde yetersiz kalmaktadır.

5.2 Öneriler

Bu çalışmada olduğu gibi, yapay sinir ağları yaklaşımı kullanılarak daha az deneysel çalışma ile daha çok deney sonucunu elde etmeyi amaçlayan çalışmalarda, yapılacak deneysel çalışmalarla YSA sisteminin eğitimi için yeterli sayıda verinin elde edilmesi gerekmektedir. Bu çalışmadan elde edilen sonuçlar incelendiğinde, aynı nominal gerilme genliğindeki numunelerin yaklaşık kırılma çevrim sayısının tahmini için YSA eğitiminde kullanılabilir nitelikte en az 3 adet deneysel verinin yeterli olabileceği gözlemlenmiştir. Aynı nominal gerilme genliğine ait 2 adet deneysel veri ile eğitimi gerçekleştirilmiş YSA modelinin test işleminde bu değerler için verdiği cevaplardaki sapmanın diğerlerine oranla daha fazla olduğu görülmektedir. Bu nedenle yapılacak olan deneysel çalışmalarda bu şartların sağlanması önemli olmaktadır.

Ayrıca, eğitim esnasında gereksiz veya yetersiz veri gruplarının kullanımından kaçınılmalıdır. Tahmin edilecek olan deney sonuçlarını doğrudan etkileyen değerlerin, kesinlikle YSA modelinin eğitiminde kullanılması gerekmektedir. Aksi takdirde YSA modelinden elde edilen tahmini değerler hiçbir şekilde kullanıma elverişli olmamaktadır. Bu durum yapılan çalışmadaki kaynaksız numunelerin kırılma çevrim sayılarının tahmininde açıkça görülmektedir. Bunun nedeni, YSA modelinin eğitiminde ağın tahmin etmesi istenen N_B kırılma çevrim sayılarını doğrudan etkileyen K_t çentik form sayılarının kullanılmamış olmasıdır.

Elde edilen tecrübeler ışığında, yapay sinir ağları ile yapılan çalışmalarda dikkat edilmesi gereken hususlardan bahsetmek gerekirse; öncelikle elimizde bulunan verilerin sayı olarak yeterli ve nitelik olarak uygun olması gereklidir. Ardından YSA modelini tasarlayacak kişiye kabaca yol gösterebilmesi için, çalışılacak konuya

benzer, üzerinde daha önceden çalışılmış konuların incelenmesi ve başlangıç olarak benzer parametre ve mimariye sahip bir YSA modeli ile işe başlanması hem zaman hem de emek tasarrufu açısından daha uygun olacaktır. Ayrıca YSA modelinin tasarımı, eğitimi, simülasyonu ve test aşamaları esnasında ara katman sayısı, ara katmandaki veya katmanlardaki düğüm sayıları, öğrenme katsayısı, momentum katsayısı gibi parametrelerin birçok farklı kombinasyonlarının denenmesi ve hatta bu denemelerin birçok kez tekrarlanması gerekebilir.

KAYNAKLAR

- [1] **Çetin, E.**, 2003: *Yapay Sinir Ağları*, 1. Baskı, Seçkin Yayıncılık, Ankara.
- [2] **Öztemel, E.**, 2003: *Yapay Sinir Ağları*, Papatya Yayıncılık, Papatya & Kelebek Tasarım, İstanbul.
- [3] **Erler, M., Sarıoğlu, Ş., Beşdok, E.**, 2003: *Mühendislikte Yapay Zeka Uygulamaları – I*, Ufuk Yayıncılık, Kayseri.
- [4] **Url-I** <<http://charailovers.blogspot.com/2010/03/engelompokan-sel-saraf.html>>, alındığı tarih 22.08.2011.
- [5] **Karakaş, Ö.**, 2006: *Biçimlenebilen Magnezyum Alaşımlarından Kaynaklı Yapı Elemanlarının Yorulma Dayanımı Değerlendirmelerinde Çentik Gerilmesi Yönteminin Uygulanması*, Doktora Çalışması, Pamukkale Üniversitesi, Denizli.
- [6] **Url-II** <<http://members.comu.edu.tr/boraugurlu/courses/bm434/week4/hafta4.pdf>> alındığı tarih 16.08.2011.
- [7] **Bae, D., Sohn, I., Jung, W., Song, S.**, 2002: A fatigue design method for spot-welded lap joints, *Proceedings of the Institution of Mechanical Engineers*, vol 216 Part D, 181 – 189.
- [8] **Obranic, P., Fajdiga, M.**, 2003: A neural network approach to describing the fretting fatigue in aluminium-steel couplings, *International Journal of Fatigue*, 25: 201 – 207.
- [9] **Mathew, MD., Dae, K., Woo-Seog, R.**, 2008: A neural network model to predict low cycle fatigue life of nitrogen-alloyed 316L stainless steel, *Materials Science & Engineering*, A 474: 247 – 253.
- [10] **Iacoviello, F., Iacoviello, D., Cavvalini, M.**, 2004: Analysis of stress ratio effects on fatigue propagation in a sintered duplex steel by experimentation and artificial neural network approaches, *International Journal of Fatigue*, 26: 819 – 828.
- [11] **Abdalla, J., Hawileh, R.**, 2010: Modeling and simulation of low-cycle fatigue life of steel reinforcing bars using artificial neural network, *Journal of The Franklin Institute*, 1 – 20.

- [12] **Genel, K.**, 2004: Application of artificial neural network for predicting strain-life fatigue properties of steels on the basis of tensile tests, *International Journal of Fatigue*, 26: 1027 – 1035.
- [13] **Marquardt, C., H Zenner, H.**, 2005: Lifetime calculation under variable amplitude loading with the application of artificial neural Networks, *International Journal of Fatigue*, 27: 920 – 927.
- [14] **Mohanty, J., Parhi, D., Ray, P., Verma, B.**, 2009: Prediction of residual fatigue life under interspersed mixed-mode (I and II) overloads by artificial neural network, *Fatigue & Fracture of Engineering Materials & Structures*, 32: 1020 – 1031.
- [15] **Majidian, A., Saidi, M.**, 2007: Comparison of fuzzy logic and neural network in life prediction of boiler tubes, *International Journal of Fatigue*, 29: 489 – 498.
- [16] **Vassilopoulos, A., Georgopoulos, E., Dionysopoulos, V.**, 2007: Artificial neural Networks in spectrum fatigue life prediction of composite materials, *International Journal of Fatigue*, 29: 20 – 29.
- [17] **Kim, K., Yoon, D., Jeong, J., Lee, S.**, 2004: Determining the stress intensity factor for a material with an artificial neural network from acoustic emission measurements, *NDT&E International*, 37: 423 – 429.
- [18] **Bache, M., Tasleem, M.**, 1999: Fatigue life prediction techniques for notch geometries in titanium alloys, *International Journal of Fatigue*, 21: S189 – S197.
- [19] **Srinivasan, V., Valsan, M., Rao, K., Manan, S., Raj B.**, 2003: Low cycle fatigue and creep-fatigue interaction behaviour of 316L(N) stainless steel and life prediction by artificial neural network approach, *International Journal of Fatigue*, 25: 1327 – 1338.
- [20] **Silverio Freire, R., Doira Neto, A., Freire de Aquino, E.** 2007: Use of modular networks in the building of constant life diagrams, *International Journal of Fatigue*, 29: 389 – 396.
- [21] **A Fathi, A., Aghakoucak, A.**, 2007: Prediction of fatigue crack growth rate in welded tubular joints using neural network, *International Journal of Fatigue*, 29: 261 – 275.
- [22] **Karakaş, Ö.**, 2011: Estimation of fatigue life for aluminium welded joints with the application of artificial neural Networks, *Mat.-wiss.u.Werkstofftech*, vol. 42, 10: 888 – 893.

ÖZGEÇMİŞ



Ad Soyad: Ercüment BÜTÜNER

Doğum Yeri ve Tarihi: İzmir / 4 Ekim 1987

Adres: Siteler Mah., Site:3, No: 62 / 1, Aliğa – İZMİR

Lisans Üniversitesi: Pamukkale Üniversitesi, Makine Mühendisliği (2005 – 2009)