

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DOĞRU AKIM MOTORLARININ HIZ DENETİMİNDE
YAPAY SİNİR AĞLARININ PERFORMANS ANALİZİ

Cafer BAL

116646

YÜKSEK LİSANS TEZİ
ELEKTRONİK - BİLGİSAYAR EĞİTİMİ
ANA BİLİM DALI

T.C. YÜKSEKÖĞRETİM KURULU
DOKÜMANTASYON MERKEZİ

ELAZIĞ

2002

116646

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DOĞRU AKIM MOTORLARININ HIZ DENETİMİNDE
YAPAY SİNİR AĞLARININ PERFORMANS ANALİZİ

Cafer BAL

YÜKSEK LİSANS TEZİ
ELEKTRONİK - BİLGİSAYAR EĞİTİMİ
ANA BİLİM DALI

Bu tez, 17-07-2002 tarihinde aşağıda belirtilen jüri tarafından oybirliği /oyçokluğu ile başarılı / başarısız olarak değerlendirilmiştir.

Danışman: Doç. Dr. Muammer GÖKBULUT

Üye: Doç. Dr. Z. Hakan AKPOLAT

Üye: Yrd. Doç. Dr. Fikret Hta

Bu tezin kabulü, Fen Bilimleri Enstitüsü Yönetim Kurulu'nun/...../..... tarih ve sayılı kararıyla onaylanmıştır.

ÖZET**YÜKSEK LİSANS TEZİ****DOĞRU AKIM MOTORLARININ HIZ DENETİMİNDE YAPAY SİNİR
AĞLARININ PERFORMANS ANALİZİ****Cafer BAL**

**Fırat Üniversitesi
Fen Bilimleri Enstitüsü
Elektronik Bilgisayar Eğitimi Ana Bilim Dalı**

2002, Sayfa: 82

Bu tez çalışmasında, uyarlanabilir aktivasyon fonksiyonlu yapay sinir ağları (YSA) ile doğru akım motorlarının hız kontrolü gerçekleştirilmiştir. Doğrusal olmayan aktüatör içeren bir doğru akım motorunun hız kontrolünde, saklı katmanında standart sigmoid, adaptif sigmoid, spline ve üstel sigmoid aktivasyon fonksiyonları kullanılan yapay sinir ağının performansı karşılaştırmalı olarak incelenmiştir. Böylece değişen yük şartlarında motorun kontrol performansını artıran ve YSA'nın eğitim süresini azaltan en uygun aktivasyon fonksiyonu belirlenmiştir. Doğru akım motorlarının hız denetiminde, uyarlanabilir aktivasyon fonksiyonlu YSA'nın etkinliğini gösteren simülasyon sonuçları verilmiştir.

Anahtar Kelimeler : Yapay sinir ağları, Uyarlanabilir aktivasyon fonksiyonları, Doğru akım motorlar

ABSTRACT

M.Sc.Thesis

**THE PERFORMANCE ANALYSIS OF ARTIFICIAL NEURAL NETWORK FOR
SPEED CONTROL OF DC MOTORS**

Cafer BAL

Firat University

Graduate School of Natural and Applied Sciences

Department of Electronics and Computer Science

2002, Page: 82

This thesis describes the speed control of DC motors using adaptive activation function neural networks. Comparative performance of the adaptive sigmoid, polynomial and spline activation functions used in the hidden layer of the multi-layer neural networks are tested for the control of DC motor including a nonlinear actuator. This comparison aims to determine the most suitable activation function, which reduces the training time of the artificial neural network and increases the control performance of the motor, under variable load conditions. Simulation results showing the effective of the adaptive activation function neural networks for speed control of DC motors, are given.

Keywords : Artificial neural networks, Adaptive activation functions, DC motors

TEŞEKKÜR

Bu tez çalışmam boyunca yardımlarını esirgemeyen başta danışmanım Sayın Doç.Dr. Muammer GÖKBULUT olmak üzere Sn. Doç. Dr. Hanifi GÜLDEMİR'e, Sn. Doç. Dr. Z. Hakan AKPOLAT'a, Sn. Öğr. Gör. Beşir DANDIL'a, Sn. Arş. Gör Erkan TANYILDIZI'na, bölümümüz öğretim elemanlarına, araştırma görevlisi arkadaşlarıma ve özellikle aileme gösterdikleri sabır ve güven için teşekkürlerimi borç bilirim.



İÇİNDEKİLER

ÖZET	II
ABSTRACT.....	III
TEŞEKKÜR.....	IV
İÇİNDEKİLER	V
ŞEKİLLER LİSTESİ	VII
SİMGELER LİSTESİ.....	IX
KISALTMALARLİSTESİ	X
1. GİRİŞ.....	1
2. DOĞRU AKIM MOTORLARI VE HIZ KONTROL YÖNTEMLERİ	4
2.1. Doğru Akım Motorlarının Yapısal Özellikleri.....	4
2.2. Doğru Akım Motorlarının Matematiksel Modeli.....	6
2.3 Doğru Akım Motorlarının Hız Kontrolü	12
2.3.1. Endüvi geriliminin kontrolü.....	12
2.3.2 Alan kontrolü	13
2.3.3 Endüvi gerilimi ve alan kontrolünün birlikte kullanılması ile hız kontrolü	13
2.4. Geleneksel kontrolörlerle doğru akım motorunun hız kontrolü	14
3. KONTROL SİSTEMLERİNDE YAPAY SİNİR AĞLARI.....	21
3.1. Yapay Sinir Ağlarının Tarihsel Gelişimi.....	21
3.2. Sinir Ağlarının Biyolojik Yapısı.....	22
3.3. Yapay Sinir Ağının Tanımı ve Modeli	23
3.4. Yapay Sinir Ağlarının Yapıları ve Çeşitleri	26
3.5. Yapay Sinir Ağlarında Öğrenme ve Öğrenme Algoritmaları.....	29
3.6. Yapay Sinir Ağlarının Temel Özellikleri	32
3.7 Yapay Sinir Ağlarının Avantajları.....	33
3.8. Yapay Sinir Ağları ile Sistemlerin Modellemesi ve Kontrolü.....	34
3.9. Sistemlerin Matematiksel Tanımı.....	35
3.10. Sistem Tanılama	36
3.10.1. Giriş Vektörlerine Göre Doğrusal Olmayan Dinamik Modeller	37
3.10.2. Doğrusal olmayan İlişkiye Göre Model Yapıları	38
3.11. Yapay Sinir Ağları ile Sistem Tanılama	39
3.11.1. Düz Modelleme	39
3.11.2. Ters Modelleme.....	40

3.11.3 Yapay Sinir Ağı Yapıları ile Sistem Kontrolünde Yeni Yaklaşımlar.....	41
3.12 Yapay Sinir Ağlarının kullanıldığı Kontrol Yapıları	42
3.12.1 Doğrudan Uyarlamalı Kontrol.....	43
3.12.2 Dahili Model (DM) Kontrol.....	44
3.12.3 Tahmini Kontrol	45
3.12.4 Öz Uyarlamalı Kontrol.....	46
3.12.5 Sabit Kontrolörlü Kontrol	47
4. ADAPTİF AKTİVASYON FONKSİYONLU YSA'LARI İLE DA	
MOTORLARIN HIZ KONTROLÜ.....	49
4.1. Sigmoid Aktivasyon Fonksiyonu.....	50
4.2. Üstel ve Sigmoid Aktivasyon Fonksiyonu	52
4.3. Spline Aktivasyon Fonksiyonu	55
4.4 DA Motorunun Kontrolünde YSA Yapısı ve Eğitimi.....	59
5. SİMÜLASYON SONUÇLARI	65
6. SONUÇ.....	75
KAYNAKLAR.....	77

ŞEKİLLER LİSTESİ

Şekil 2.1. Demir çekirdekli, kalıcı mıknatıslı bir doğru akım motorunun kesit görünümü	5
Şekil 2.2. 2 kutuplu serbest uyarımlı DA motorunun bağlantı şeması.....	5
Şekil 2.3. Kalıcı mıknatıslı doğru akım motorunun eşdeğer devresi	6
Şekil 2.4. Kalıcı mıknatıslı/serbest uyarımlı doğru akım motorunun blok diyagramı	9
Şekil 2.5. Doğru akım motorlarının hız-moment karakteristiği	11
Şekil 2.6. Endüvi gerilimi ve alan kontrolünün birlikte gerçekleştirilmesi.....	14
Şekil 2.7. DA motorunun PI kontrolü.....	15
Şekil 2.8. Kullanılan doğru akım motorunun blok diyagramı	16
Şekil 2.9. Kontrolör ilave edilmiş doğru akım motoru	17
Şekil 2.10. Farklı PI parametreleri için Root contour grafiği	18
Şekil 2.11. Kontrolör ve doğru akım motorunun blok diyagramı	18
Şekil 2.12. Nominal referans hız girişi için PI kontrollü motorun hızı	19
Şekil 2.13. PI kontrolörün aktüatör ile performansı	20
Şekil 3.1. Biyolojik sinir hücresi	23
Şekil 3.2. Yapay sinir hücresinin modeli.....	24
Şekil 3.3. Aktivasyon fonksiyonları.....	26
Şekil 3.4. İleri Beslemeli ağ yapısı	28
Şekil 3.5. Doğrusal olmayan model 3 yapısı.....	39
Şekil 3.6. Yapay sinir ağı ile düz modelleme (NARX model)	40
Şekil 3.7. Yapay sinir ağı ile düz modelleme (NOE model)	40
Şekil 3.8. Yapay sinir ağı ile ters modelleme	41
Şekil 3.9. Bellek hücreli yapay sinir ağı.....	41
Şekil 3.10. YGKİ Yapay sinir ağı	42
Şekil 3.11. Doğrudan uyarlamalı kontrol	43
Şekil 3.12. Dahili model kontrol.....	44
Şekil 3.13. Tahmini kontrol.....	45
Şekil 3.14. Öz uyarlamalı kontrol	47
Şekil 3.15. Sabit kontrolörlü kontrol	48
Şekil 4.1. Çift yönlü sigmoid fonksiyon	50

Şekil 4.2. Uyarlamalı karesel fonksiyon	52
Şekil 4.3. Catmull-Rom spline tabanlı aktivasyon fonksiyonunun sabit Δx adımlı kontrol noktalarının temsili.....	56
Şekil 4.4. Uyarlamalı spline yapay sinir ağı	58
Şekil 4.5. Yapay sinir ağları ile DA motorların hız kontrolü blok diyagramı.....	60
Şekil 5.1. Tristörlü doğrultucu ile beslenen DA motorun hız kontrolünde kontrolör YSA'ların eğitimi sırasında toplam karesel hataların değişimi.....	65
Şekil 5.2. Eğitim örnekleri (referans) ve sistem çıkışı	67
Şekil 5.3. Referans hız girişi ile motor hızı arasındaki hatalar	67
Şekil 5.4. Eğitilmiş kontrolör ağların basamak bir referans hız girişini genelleme performansları.....	68
Şekil 5.5. $f(.)=u^2$ aktüatörü ile DA motorun hız kontrolünde kontrolör YSA'ların eğitimi sırasında toplam karesel hataların değişimi	69
Şekil 5.6. $f(.)=u^2$ aktüatörü ile kontrolör ağların basamak bir referans hız girişini genelleme performansları.....	70
Şekil 5.7. Sigmoid aktüatör ile DA motorun hız kontrolünde kontrolör YSA'ların eğitimi sırasında toplam karesel hataların değişimi	71
Şekil 5.8. Sigmoid aktüatörü ile kontrolör ağların basamak bir referans hız girişini genelleme performansları.....	72
Şekil 5.9. Tristörlü doğrultucu ile beslenen DA motorun YSA ile hız kontrolünde YSA'ların eğitimi sonundaki uyarlanabilir aktivasyon fonksiyonları.....	74
Şekil 5.10. Tristörlü doğrultucu ile beslenen DA motorun YSA ile hız kontrolünde değişken yük torkunda performansı.....	75

SİMGELER LİSTESİ

x	YSA giriş vektörü.
u	Katmanlı YSA'da saklı katmandaki hücrelerin net girişi vektörü
v	Katmanlı YSA'da saklı katman çıkış vektörü.
y	YSA çıkış vektörü.
W_{ji}^1	Katmanlı YSA'da 1. katmandaki i. hücre çıkışından j. hücreye bağlantı ağırlığı.
b	Saklı katmandaki hücrelere uygulanan polarma ağırlığı.
q	Çıkış hücresine uygulanan polarma ağırlığı.
Δ	Ağırlıklara ve aktivasyon parametrelerine uygulanan düzeltme.
α	Öğrenme oranı.
β	Momentum katsayısı.
e	Örneksel hata.
e_c	Referan giriş ile motor çıkışı arasındaki hata
E	Örneksel hataların kareleri.
ω_{ref}	Referans hız girişi.
u_u	Sistem girişi.
$i_a(t)$	Endüvi akımı
L_a	Endüvi endüktansı
R_a	Endüvi direnci
v_a	Uygulanan gerilim
e_b	Zıt emk gerilimi
ϕ	Hava aralığı akısı
TL	Yük momenti
ω	Rotorun açısal hızı
T	Motorun ürettiği elektriksel moment
J	Motor eylemsizlik momenti
B	Viskoz sürtünme katsayısı
K_b	Zıt emk sabiti
K_i	Moment sabitidir
K_m	Motor sabiti

KISALTMALAR LİSTESİ

YSA	Yapar sinir ağıları
DA	Doğru akım
AA	Alternatif akım
BHYSA	Bellek hücreli yapay sinir ağıları
YGKİ	Yöresel geribeslemeli-küresel ileri beslemeli
DM	Dahili model
UPYSA	Uyarlanabilir Polinomlu Yapay Sinir Ağlarında



GİRİŞ

Elektrik motorları sürücü sistemler olarak endüstrinin vazgeçilmez elemanlarından biridir. Değişik özellikleriyle bir çok elektrik motoru bulunmasının yanında doğru akım (DA) motorları; ayarlanabilir hız, iyi hız regülasyonu, sık başlatma ve durdurmadaki başarımlarından dolayı önemli bir yere sahiptir (Dubey, 1989). Bütün sürücü sistemlerde olduğu gibi DA motorlarında da, parametre değişimleri, yük momentindeki değişimler ve bilinmeyen bozucu girişler gibi etkenlerden dolayı iyi bir hız yada konum kontrolüne ihtiyaç duyarlar. Günümüze değin DA motorlarının hız ve konum kontrolü için bir çok yöntem kullanılmıştır. Ancak son on yıllarda özellikle bilgi işleme teknolojisindeki gelişmelere paralel olarak yapay sinir ağlarında da (YSA) önemli ilerlemeler kaydedilmesiyle, bir çok alanda olduğu gibi kontrol alanında da çoğu sistemin kontrolünde YSA kullanılmıştır.

Biyolojik sınırlardan esinlenerek elde edilen YSA, doğrusal olmayan ve paralel bilgi işleme özellikleriyle; bulunduğu ortamın değişmesiyle cevaptaki davranışı değiştirebilme, giriş uyarılarındaki küçük değişimleri tolere edebilme, değişik bazı uyarılar karşısında daha önceki uyarılar arasında benzer özellikler keşfederek deneyimi olmadığı halde uyarıyı cevaplayabilme gibi üstün özelliklere sahiptir (Harvey, 1994), (Karna ve David, 1989). YSA bu yeteneklerinden dolayı, pek çok uygulamada olduğu gibi kontrol alanında da doğrusal olmayan, bilinmeyen yada yarı belirgin sistemlerin kontrolünde, parametre değişimi ve bozucu giriş gibi olumsuzlukların etkisini azaltmak amacıyla yaygın olarak uygulanma imkanı bulmuştur. Kontrol sistemlerinde, genellikle değişen dinamikler ve yük şartları, değişen eylemsizlik, parametre ve çevre şartları ile karşılaşılır. Bu nedenle, kontrol edilecek sistemin doğru bir matematiksel modelinin çıkarılmasına gereksinim duyan geleneksel kontrol yöntemleri, sistemin doğru bir matematiksel modeli çıkarılsa bile doyum, bozucu girişler, parametre değişimleri ve gürültü gibi bilinmeyen şartlardan dolayı etkin bir şekilde kullanılamazlar (Weerasooriya ve El-Sharkavi, 1991), (El-Sharkavi v.d., 1994). Geleneksel kontrol yöntemlerinin bütün bu sorunlarına karşın YSA yukarıda belirtilen genelleme yapabilme ve öğrenebilme yetenekleriyle bir kontrolörden beklenen pek çok özelliği yerine getirebilmektedir.

Sistemlerin modellenmesi ve kontrolünde, ileri beslemeli ve geri beslemeli çok katmanlı yapay sinir ağları yoğun ilgi görmüş ve bu ağların eğitimi için statik ve

dinamik geriye yayılım öğrenme algoritmaları geliştirilmiştir. İleri beslemeli yapay sinir ağları doğrusal olmayan bir dinamik işlevi gerçekleştirir ve dinamik geriye yayılım öğrenme algoritmaları ile eğitilirler. Dinamik geriye yayılım algoritmalarının ardışıl ve zamanda geriye yayılım uygulamalarının yanı sıra, geriye yayılım algoritmalarının toplu ve örneksel öğrenme çeşitleri de kontrol sistemlerine uygulanmaktadır. (Akın ve Gökbulut, 1997), (Koçak v.d., 1996), (Karahan v.d., 1997) Örnekse öğrenme algoritmasında her örnek teker teker işlenerek ağırlık değerleri her örnekte ayarlandığından gerçek zamanlı öğrenme, toplu öğrenmede ise örnekler toplu olarak değerlendirilerek ağırlık değerleri bir defada ayarlandığından gerçek zamanlı olmayan öğrenme olarak da adlandırılır.

YSA'nın eğitim algoritmaları ile birlikte yapısal özellikleri, YSA'lı kontrol sistemlerinin performansını etkileyen önemli faktörlerdir. Çok katmanlı ağ yapısının davranışı büyük ölçüde aktivasyon fonksiyonun şekline bağlı olduğu bilinmektedir (Guarnieri, 1999). Tang ve Kwan, (1992) eğimi uyarlanabilir sigmoidal aktivasyon fonksiyonun kullanıldığı çok katmanlı bir ağ yapısı sunmuştur. Chen ve Chang, (1996) iki parametrelili genelleştirilmiş standart sigmoid fonksiyonunun eğim ve doyum seviyelerinin uyarlanmasını önermiştir.

Cybenko, (1989) Chen ve Chen, (1993) ve Chen vd. (1995)'de aktivasyon fonksiyonlarının farklı sınıflarının davranışları, [7 –12] de birkaç farklı yazar tarafından ayrıntılı olarak çalışılmıştır.

Piazza, (1992) uyarlanabilir polinomlu aktivasyon fonksiyonlarına dayanan yapay sinir ağlarının yapısı vermiştir. Uyarlanabilir Polinomlu Yapay Sinir Ağlarında (UPYSA) yapısal karmaşıklık azaltılmış ve sınıflandırma problemlerinin çözümüne uygulanmıştır. Ancak, UPYSA'nın eğri uydurma problemlerindeki performansı yeterli görülmemekte ve ayrıca, kullanılan polinomların şekline de bağlı olarak özellikle öğrenme sürecinde sayısal hesaplamalarda kararsızlık ve yerel minimumlara yakalanma sorunları ortaya çıkmaktadır. Hwang vd., (1994) helmit polinomlarına dayanan yeni bir yaklaşım sunmuştur. Spline fonksiyonu gerekli olan düzgün karakteristiğe sahiptir (Güldemir, 1999), (Guarnieri vd., 1999). Guarnieri vd., (1999) spline aktivasyon fonksiyonlu hücrelerin özelliklerini araştırmış ve spline fonksiyonunun üstünlüklerini analiz ederek örüntü ve sistem tanılamaya uygulamıştır.

Xu, ve Zhang, (2000) standart sigmoid fonksiyonuna üstel fonksiyon (üstel ve sigmoid) ilave ederek elde edilen aktivasyon fonksiyonunu sınıflandırma problemine uygulamıştır.

Yapılan bu çalışmada, adaptif aktivasyon fonksiyonlu yapay sinir ağlarının sistem kontrolünde kullanımı incelenerek, yapay sinir ağlarının öğrenme ve genelleme yeteneklerinin belirlenmesi için YSA ile bir DA motorunun hız kontrolü gerçekleştirilmiştir. Bu amaçla;

2. bölümde, DA motorları ve matematiksel modellemesi hakkında genel bilgiler verilmiş ve DA motorunun geleneksel yöntemle hız kontrolü gerçekleştirilerek gelişmiş kontrol yöntemlerinin gerekliliği belirlenmiştir.

3. bölümde, yapay sinir ağları ve uygulamada sık olarak kullanılan ağ çeşitleri hakkında genel bilgiler verilmiş ve yapay sinir ağlarının, sistem modelleme ve kontrol uygulamalarında sık kullanılan ağ yapıları tanıtarak işleyişi hakkında genel bilgiler verilmiştir.

4. bölümde, DA motorunun adaptif aktivasyon fonksiyonlu YSA ile hız kontrolünün gerçekleştirildiği kontrol yapısının oluşturulması ve aktivasyon fonksiyonlarının parametrelerinin eğitim algoritması verilmiştir.

5. bölümde ise, ele alınan örnek DA motorunun yapay sinir ağları ile hız kontrolü, standart sigmoid, adaptif sigmoid, spline ve üstel sigmoid olmak üzere dört farklı aktivasyon fonksiyonlu yapay sinir ağı kontrolörle gerçekleştirilmiş ve yapay sinir ağlarının öğrenme ve genelleme yeteneklerinin belirlenebilmesi için yapılan çalışmalar ve elde edilen sonuçlar verilerek değerlendirilmiştir.

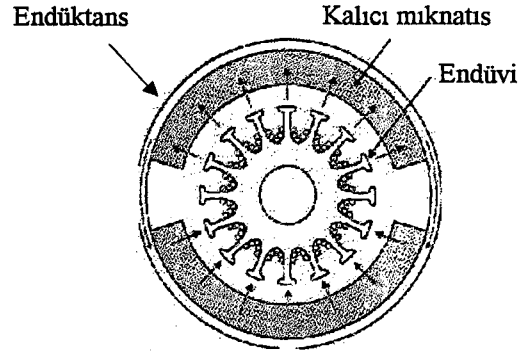
2. DOĞRU AKIM MOTORLARI VE HIZ KONTROL YÖNTEMLERİ

Doğru akım (DA) motorları, bir sürücü sistemi olarak endüstride yaygın olarak kullanılmaktadır. Sürücü sistemleri temelde elektrik enerjisini mekanik enerjiye çevirerek sürdüğü (tahrik ettiği) sistemde iyi bir hız regülasyonu, başlatma, frenleme ve devir yönü değişikliği sağlamak için kullanılır. DA motorları yüksek başlatma momentlerine sahip olmaları ve hız denetiminin geniş bir aralıkta yapılabilmesi nedeniyle birçok endüstriyel sistemde tercih edilmektedir (Gedikpınar ve Güldemir, 2001). Ancak parametre değişimleri ve yük momenti gibi bilinmeyen bozucu girişler altında iyi bir hız yada konum kontrolüne ihtiyaç duyan yüksek performanslı sürücü sistemlerinde dayanıklı kontrol yöntemini de seçmek gerekir. Bu bölümde, DA motorların yapısal özellikleri ve geleneksel hız kontrol yöntemleri incelenerek gelişmiş kontrol yöntemlerinin gerekliliği belirlenecektir.

2.1. Doğru Akım Motorlarının Yapısal Özellikleri

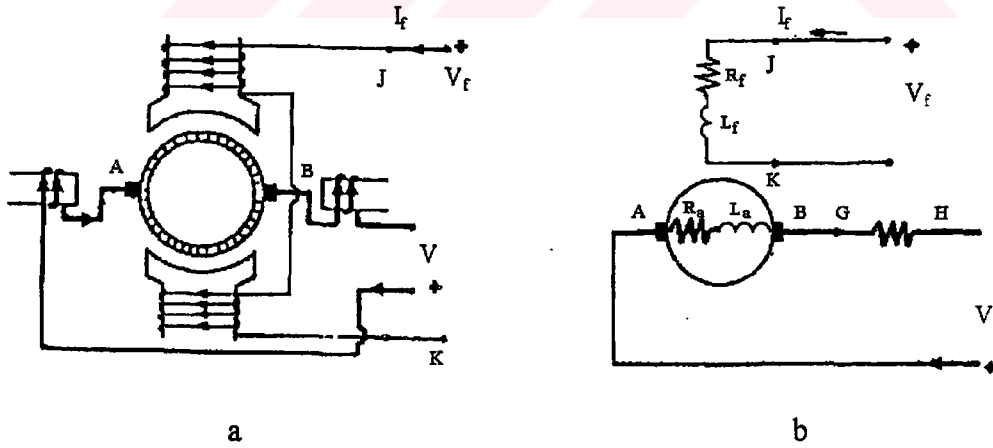
Doğru akım motorları, biri sabit ve diğeri hareketli olmak üzere iki temel parçadan oluşan ve bu iki parçadaki manyetik alanların etkileşimi sonucu hareket enerjisi üreten elektrik makineleridir.

Doğru akım motorları, uyarılma, komutasyon ve tasarım şekli gibi çeşitli kriterlere göre sınıflandırılır. Kalıcı mıknatıslı motorlarda endüvi tasarımı, demir çekirdekli, yüzey sargılı ve hareketli sargılı olmak üzere üç şekilde yapılabilir. Demir çekirdekli tasarım, yüksek eylemsizlik ve yüksek endüktansa sahip olmasına karşın, düşük maliyet ve yüksek güvenilirliğe sahiptir. Yüzey sargılı motorların endüktansı, demir çekirdekli motorlara göre daha düşüktür. Hareketli sargılı motorlar ise, çok düşük eylemsizlik momentine ve çok düşük endüktansa sahiptir. Şekil 2.1 de kalıcı mıknatıslı demir çekirdekli bir doğru akım motorunun kesit görünümü verilmiştir (Kuo, 1991).



Şekil 2.1 Demir çekirdekli, kalıcı mıknatıslı bir doğru akım motorunun kesit görünümü

Uyarma sargılarının beslenme şekline göre doğru akım motorları, serbest uyarımlı, seri uyarımlı, şönt uyarımlı ve kompund uyarımlı olmak üzere dört gruba ayrılır. Şekil 2.2 de serbest uyarımlı bir doğru akım motorunun uyarma sargılarının bağlantı şekli ve eşdeğer devresi verilmiştir. Şekilde görülen J-K uçlarıyla belirlenen uyarma sargıları motorun beslendiği kaynaktan bağımsız olarak beslenir. Buna karşılık uyarma sargıları, endüviye seri bağlandığında seri motor, paralel (şönt) bağlandığında paralel motor, hem seri hem de paralel iki uyarma sargısı bağlandığında ise kompund motor adını alır.



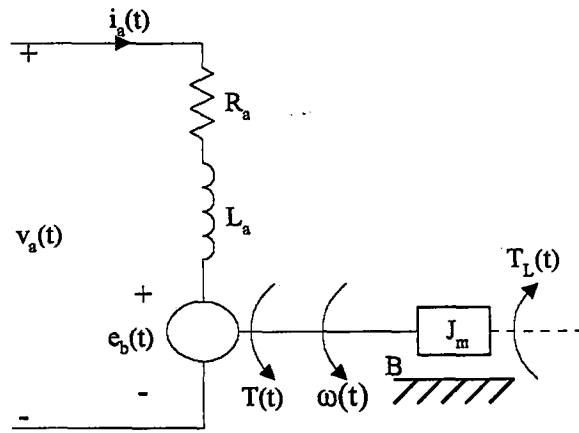
Şekil 2.2 2 kutuplu serbest uyarımlı DA motorunun
a) Bağlantı şeması b) Eşdeğer devresi

Doğru akım motorları, dönel parçalardaki sargıların akımını yön değiştirmek amacıyla kullanılan fırça ve komütatörler nedeniyle çok pahalıdır ve fazla bakım

gerektirir. Ayrıca, boyutlarının büyük olması ve boyutlarına göre ürettikleri momentin düşüklüğü de bu motorların önemli sakıncalarındandır. Ancak, gelişen kalıcı mıknatıs teknolojisiyle birlikte küçük boyutlu, yüksek moment üretebilen ve düşük maliyetli doğru akım motorları üretilebilmektedir. Ayrıca, kalıcı mıknatıs kutupları dönel kısma alınmak üzere sabit parça üzerindeki sargıların komutasyonunu elektronik olarak gerçekleştirildiği fırçasız doğru akım motorları da endüstriyel otomasyon sistemlerinde yaygın olarak kullanılır hale gelmiştir (Gedikpınar, 2002).

2.2. Doğru Akım Motorlarının Matematiksel Modeli

Kalıcı mıknatıslı bir doğru akım motorunun eşdeğer devresi, beslemesinin uygulandığı endüvi sargılarının elektriksel direnci R_a , endüktansı L_a olmak üzere seri RL devresi ile temsil edilebilir. Ancak, endüvide oluşan zıt elektromotor kuvveti, devreye aksi yönde bağlanan bir gerilim kaynağını temsil ettiği için, seri RL devresine ters yönde bağlı bir gerilim kaynağı elektriksel eşdeğer devreye eklenmelidir. Elektriksel devrenin eşdeğeri ile birlikte, ω açısal hızıyla dönen bir mil, açısal hızla aynı yönde üretilen moment, endüvinin eylemsizliğini (J), ve sürtünmesini (B) temsil eden mekanik sistemin oluşturduğu DA motorun eşdeğer devresi şekil 2.3 deki gibi çizilebilir (Tuncer, 1998).



Şekil 2.3 Kalıcı mıknatıslı doğru akım motorunun eşdeğer devresi.

Eşdeğer devrede kullanılan değişkenler ve parametreler;

$i_a(t)$	endüvi akımı (A)	L_a	endüvi endüktansı (H)
R_a	endüvi direnci (Ohm)	$v_a(t)$	uygulanan gerilim (V)
$e_b(t)$	zıt emk gerilimi (V)	ϕ	hava aralığı akısı (Weber)
$T_L(t)$	yük momenti (Nm)	$\omega(t)$	rotorun açısal hızı (rad/sn)
$T(t)$	motorun ürettiği elektriksel moment (Nm)	J	motor eylemsizlik momenti (kgm^2)
$\theta(t)$	rotorun açısal konumu (rad)	B	viskoz sürtünme katsayısı ($\text{N}/(\text{m}/\text{sn})$)
$K_i=K_m\phi$	moment sabiti (Nm/A) dir.	$K_b=K_m\phi$	zıt emk sabiti ($\text{V}/\text{rad}/\text{sn}$)

Doğru akım motorunun dinamik analizi için gerekli matematiksel modeli, Şekil 2.3'deki eşdeğer devreden çıkarılabilir. Elektriksel devrenin denklemi,

$$v_a(t) = L_a \frac{di_a(t)}{dt} + R_a i_a(t) + e_b(t) \quad (2.1)$$

şeklinde yazılır. Endüvinin açısal hızıyla orantılı olarak oluşan zıt elektromotor kuvvet,

$$e_b(t) = K_m \phi \frac{d\theta(t)}{dt} = K_m \phi \omega(t) \quad (2.2)$$

dir. ϕ sabit olduğundan $K_m \phi = K_b$ olarak alınır ve denklem (2.1) de yerine yazılırsa,

$$v_a(t) = L_a \frac{di_a(t)}{dt} + R_a i_a(t) + K_b \omega(t) \quad (2.3)$$

olur. Endüvi akımı ile orantılı olarak motorun ürettiği elektriksel moment,

$$T(t) = K_m \phi i_a(t) \quad (2.4)$$

Burada ϕ sabit olduğundan $K_m\phi=K_i$ olarak alınabilir. Motorun üretmiş olduğu moment, motorun eylemsizliğini, yük momentini ve viskoz sürtünmeyi karşılamak zorundadır. Buna göre mekanik sistemin denklemi,

$$T(t) = J \frac{d\omega(t)}{dt} + T_L(t) + B\omega(t) \quad (2.5)$$

Motorun ürettiği elektriksel moment ifadesi, denklem (2.5) te yerine yazılırsa,

$$K_i i_a(t) = J \frac{d\omega(t)}{dt} + T_L(t) + B\omega(t) \quad (2.6)$$

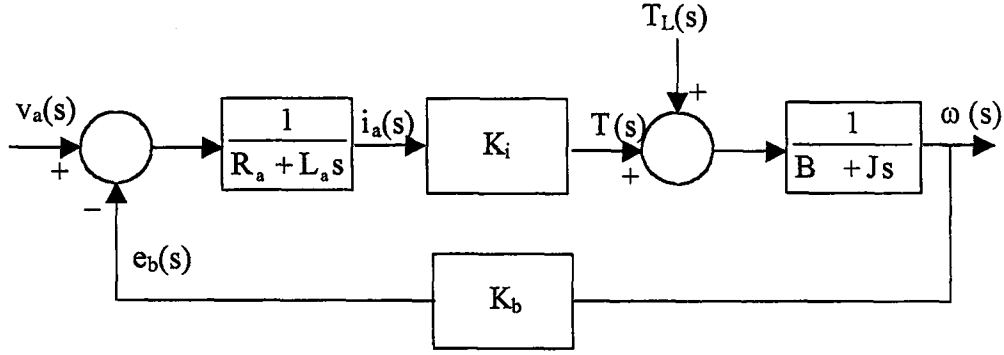
olur. Sistemin durum değişkenleri $i_a(t)$, ve $\omega(t)$ olmak üzere, denklem (2.3) ve (2.6) vektör-matris biçiminde düzenlenirse;

$$\begin{bmatrix} \frac{di_a(t)}{dt} \\ \frac{d\omega(t)}{dt} \end{bmatrix} = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{K_b}{L_a} \\ \frac{K_i}{J} & -\frac{B}{J} \end{bmatrix} \begin{bmatrix} i_a(t) \\ \omega(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{L_a} & 0 \\ 0 & -\frac{1}{J} \end{bmatrix} \begin{bmatrix} v_a(t) \\ T_L(t) \end{bmatrix} \quad (2.7)$$

elde edilir. Motorun açısal hızı ile giriş gerilimi arasındaki transfer fonksiyonu;

$$\frac{\omega(s)}{v_a(s)} = \frac{K_i}{JL_a s^2 + (R_a J + BL_a)s + (K_i K_b + R_a B)} \quad (2.8)$$

olarak elde edilir. Burada $T_L(t)$ sıfır olarak alınmıştır. Şekil 2.4'de, denklem (2.3) ve (2.6) kullanılarak çizilen, kalıcı mıknatıslı doğru akım motorunun blok diyagramı görülmektedir.



Şekil 2.4 Kalıcı mıknatıslı/serbest uyarımlı doğru akım motorunun blok diyagramı

Doğru akım motoru gerçekte açık çevrimli bir sistem olmasına rağmen, Şekil 2.4'de görüldüğü gibi zıt elektromotor kuvvet nedeniyle motorun içinde bir geri besleme mevcuttur. Zıt elektromotor kuvvet fiziksel olarak motor hızıyla orantılı negatif bir geri besleme işaretidir. Zıt elektromotor kuvvet, bir elektriksel sürtünme olarak, sistem kararlılığını artıran bir işarete karşılık gelir.

Gerçekte K_i moment sabiti ve K_b zıt elektromotor sabiti, bir motor için iki farklı parametre olsa da aralarında kuvvetli bir ilişki vardır. Bu ilişki güç denklemlerinden görülebilir. Buna göre endüvi de oluşan elektrik gücü;

$$P(t) = e_b(t) i_a(t) \quad (2.9)$$

olarak yazılır. Mekanik güç;

$$P(t) = T(t) \omega(t) \quad (2.10)$$

şeklinde ifade edilebilir. Buna göre güçlerin eşitliği gerektiğinden

$$P(t) = T(t) \omega(t) = K_b \omega(t) \frac{T(t)}{K_i} \quad (2.11)$$

ilişkisinden;

$$K_b (V/(rad/s)) = K_i (N-m/A) = K_m \quad (2.12)$$

olmak durumundadır. Bir doğru akım motorunun dinamiklerini ortaya koyan yukarıdaki matematiksel modeller, kontrol sistemlerinin analiz ve tasarımı için gereklidir. Ancak L_a 'nın ve J 'nin etkisinin görülmediği sürekli rejim durumunda, özellikle motorun hız-moment karakteristiğinin incelenmesi gerekebilir. Bu amaçla Şekil 2.3 deki doğru akım motorunun eşdeğer devresinden sürekli rejim denklemleri elde edilirse, motorda oluşan zıt elektromotor kuvveti,

$$E_b = K_m \phi \omega \quad (2.13)$$

olur. Elektriksel devrenin denklemi,

$$V = E_b + I_a R_a \quad (2.14)$$

olur. Motorun ürettiği moment ifadesi,

$$T = K_m \phi I_a \quad (2.15)$$

olduğundan sürekli rejim denklemleri kullanılarak motor hızı,

$$\omega = \frac{V}{K_m \cdot \Phi} - \frac{R_a \cdot I_a}{K_m \cdot \Phi} = \frac{V}{K_m \cdot \Phi} - \frac{R_a \cdot T}{(K_m \cdot \Phi)^2} \quad (2.16)$$

olarak yazılır.

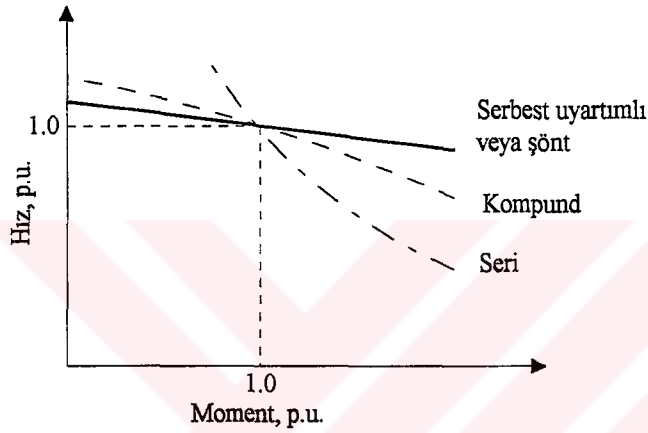
Denklem (2.13) – (2.16) üç tip doğru akım motorları (serbest uyarımlı, şönt ve kompund motor) için de geçerlidir. Serbest uyarımlı doğru akım motorlarında alan gerilimi sabit tutulduğunda Φ akısı moment değişimlerine karşılık sabittir (Dubey, 1989).

$$K_m \Phi = K \text{ (sabit)} \quad (2.17)$$

Buna göre denklem (2.16) tekrar düzenlenirse,

$$\omega = \frac{V}{K} - \frac{R_a \cdot I_a}{K} = \frac{V}{K} - \frac{R_a \cdot T}{K^2} \quad (2.18)$$

elde edilir. Denklem (2.18) den görüldüğü gibi serbest uyarımlı bir doğru akım motorunun hız-moment karakteristiği bir doğru şeklindedir. Yüksüz durumdaki hız, endüvi gerilimi ve alan sargısı tarafından saptanmaktadır. Şekil 2.5'ten görüldüğü gibi, moment artışına karşılık olarak motor hızı azalmaktadır. Hız regülasyonu (2.18)'deki gibi endüvi devresi direncine bağlıdır.



Şekil 2.5 Doğru akım motorlarının hız-moment karakteristiği

Pratikte alan akımı sabit olsa bile endüvi reaksiyonu nedeni ile moment artışına karşılık olarak akı da azalmaktadır. Bunun sonucunda hızdaki azalma (2.18) denklemi ile hesaplanandan daha az olmaktadır.

Seri motorlarda alan akısı Φ , endüvi akımının bir fonksiyonudur. Doyum bölgesi dışında alan akısı ile endüvi akımı arasındaki ilişki şu şekilde ifade edilebilir;

$$\Phi = K_f \cdot I_a \quad (2.19)$$

Seri motorlar için (2.15) ve (2.16) denklemlerinden yararlanarak moment ve hız denklemlerini aşağıdaki gibi düzenlenebilir,

$$T = K_m \cdot K_f \cdot I_a^2 \quad (2.20)$$

$$\omega = \frac{V}{K_m \cdot K_f \cdot I_a} - \frac{R_a}{K_m \cdot K_f} = \frac{V}{\sqrt{K_m \cdot K_f}} \cdot \frac{1}{\sqrt{T}} - \frac{R_a}{K_m \cdot K_f} \quad (2.21)$$

Burada R_a ; endüvi sargısı ve alan sargısı dirençlerinin toplamına eşittir. Doğru akım seri motorun hız-moment karakteristiği şekil 2.5’de gösterilmiştir. Seri motorlarda momentteki artış endüvi akımındaki artışı oluşturur. Akımın artışına bağlı olarak Φ akısı da artar ve motor hızını düşürür. Aşırı yüklenme durumlarındaki moment değerlerinde manyetik devre doyuma gider ve hız-moment karakteristiği bir doğruya yaklaşır. Seri motorlar başlatma momentlerinin yüksek olması istenen yerlerde ve büyük yük momentlerinde etkin olarak kullanılmaktadırlar.

Kompund motorun hız-moment denklemleri (2.13) ve (2.15) kullanılarak türetilir. Hız-moment karakteristiği şekil 2.5’de verilmiştir.

2.3 Doğru Akım Motorlarının Hız Kontrolü

Denklem (2.16)’ dan bir doğru akım motorunun hız kontrolünün üç yöntemle yapılabileceği görülmektedir. Bunlardan biri olan endüvi direncinin değiştirilmesi ile yapılacak hız kontrolü, endüvi devresine ilave edilecek dirençlerle sağlanırken, verimi önemli oranda düşürmesinden ve hız kontrolünün geniş sınırlar arasında yapılamamasından dolayı, tercih edilen bir yol değildir. Bu nedenle doğru akım motorunun hız kontrolü genellikle iki şekilde yapılır:

Endüvi gerilimi ile hız kontrolü

Alan akısının (yada akımı) ile hız kontrolü

2.3.1. Endüvi geriliminin kontrolü

Denklem (2.14) ve (2.15) den görüldüğü gibi, doğru akım motorunun endüvi gerilimi azaltılırsa, endüvi akımı ve üretilen momentin azalacağı görülür. Böylece motor momenti yük momentinden daha düşük olur ve zıt elektromotor kuvveti ile beraber hız da azalır. Bu durumda, daha düşük bir hız değerinde, motor tarafından üretilen moment yük momentine eşit olur. Denklem (2.14) ve (2.15)’e göre, doğru akım

motorunun endüvi gerilimi artırıldığında motor tarafından üretilen moment ve buna bağlı olarak endüvi akımı artar. Sonuç olarak motor daha yüksek bir hızda yükü tahrik eder (Dubey,1989).

2.3.2 Alan kontrolü

Belirli bir hızda doğru akım motorunun alan akımı azaltıldığı takdirde denklem (2.13)'den görüldüğü gibi, zıt elektromotor kuvvette beraberinde azalır ve endüvi devre direncinin küçük olması nedeniyle, denklem (2.14)'den görüldüğü gibi, endüvi akımı büyük bir oranda artar. Alan zayıflatılması durumunda denklem (2.15)'e göre, yük momenti göz önüne alınmazsa üretilen moment artar. Böylece motor hızlanmaya başlar ve denklem (2.13)'de görüldüğü gibi, zıt elektromotor kuvvetini artırır. Sonuç olarak motor bir öncekinden daha büyük yeni bir hıza erişmiş olur ve motor tarafından üretilen moment bu yeni hız değerinde yük momentine eşit olur.

Doğru akım motorlarının alan akımı artırıldığı taktirde ise, denklem (2.13)'de görüldüğü gibi, zıt elektromotor kuvvet alan şiddetine bağlı olarak artar. Bazı durumlarda besleme geriliminden daha büyük olabilir. Bunun sonucunda endüvi akımı azalmaya başlar ve hatta ters yönde artmaya devam eder. Motor bir jeneratör olarak çalışır ve kaynağa enerji aktarır. Denklem (2.15)'de görüldüğü gibi, akım azalınca üretilen moment azalır ve denklem (2.16)'dan görüldüğü gibi hız azalır. Hızdaki azalma ile motor bir önceki hızından daha düşük bir hızda dönmeye başlar ve motor tarafından üretilen moment yeni bir hız değerinde yükü tahrik eder.

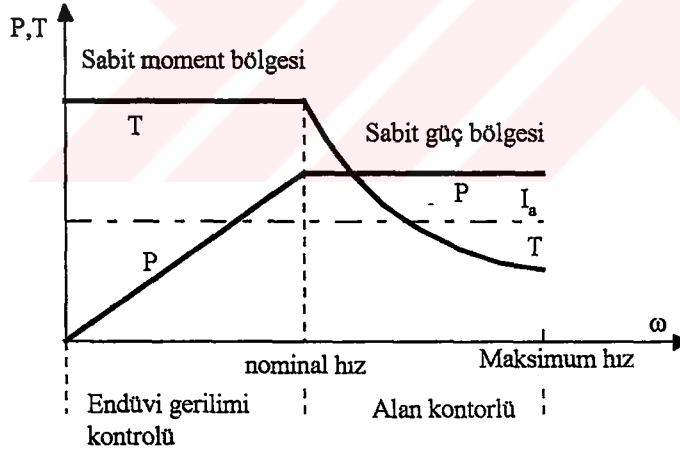
2.3.3 Endüvi gerilimi ve alan kontrolünün birlikte kullanılması ile hız kontrolü

Sürücü sistemlerinde endüvi gerilimi ve alan kontrolü birlikte gerçekleştirildiği taktirde, hız daha geniş bir aralıkta denetlenebilir. Endüvi geriliminin denetlenmesi durumunda tüm hızlarda maksimum moment kabiliyeti sağlanmaktadır. Alan kontrol metodu ise, endüvi geriliminin kontrolü ile erişilemeyen hız değerleri için kullanılır. Nominal hızın altındaki hız değerlerinde alan akımı sabit tutularak sadece endüvi gerilimi değiştirilmesi ile hız kontrolü yapılır. Nominal hızın üstünde bir hız kontrolü, endüvi geriliminin kontrolü ile sağlanamaz. Çünkü besleme gerilimi belirli bir değerde

sabittir. Bu yüzden nominal hızın üzerindeki hızlar için, endüvi gerilimi ve alan kontrolü birlikte yapılır (Dubey,1989).

Denkelem (2.13) ve (2.15)'den görüldüğü gibi, alan akısı azaltıldığında üretilen moment düşer, buna karşılık, i_a ve v_a sabit kabul edilirse, zıt elektromotor kuvvet artar ve dolayısıyla hız artar. Öyleyse alan kontrolü ile hız kontrolü yapmak, motorun moment kabiliyetini azalmaktadır. Yani nominal hızın üzerindeki hız değerlerinde hız ile moment ters orantılı olarak değişir.

Nominal hıza kadar, denklem (2.15)'de görüldüğü gibi, alan akısı ve endüvi akımı sabit olursa üretilen moment sabit olur. v_a değişimiyle orantılı hız değişimi olduğundan dolayı güç hız ile orantılı olarak artış gösterir. Nominal hız üstündeki hız değerleri için, denklem (2.14) ve (2.15)'den görüldüğü gibi, endüvi akımı ve endüvi gerilimi sabit olursa, moment hız ile ters orantılı değişir. Buna karşılık v_a ve i_a sabit olduğundan güç sabit kalır. Alan ve endüvi geriliminin birlikte kontrolü yapılması şekil 2.6'da moment ve güç değerleri ile verilmiştir. Şekilde gösterildiği gibi endüvi akımı tüm hızlar için sabit kabul edilmiştir.

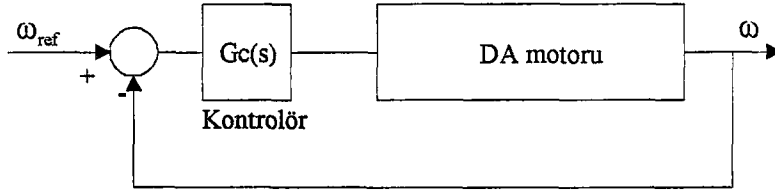


Şekil 2.6 Endüvi gerilimi ve alan kontrolünün birlikte gerçekleştirilmesi

2.4. Geleneksel kontrolörlerle doğru akım motorunun hız kontrolü

Geleneksel kontrol mantığı; arzu edilen çıkışı elde etmek için, çıkış ile arzu edilen değer arasındaki hata işaretinin türevi ve/veya integrali alınarak, oransal kontrolöre eklenmesinden oluşmaktadır. Bu şekilde, sistem çıkışının arzu edilen değeri

takip etmesi amacına ulaşmak için, kontrolörün hata işaretine göre üretmiş olduğu işaret sistem girişini oluşturur. Yani kontrolör arzu edilen sistem çıkışı için, sisteme uygulanması gereken giriş işaretini hesaplar. Geleneksel kontrolörler, oransal kontrol+integral kontrol (PI), oransal kontrol+türev kontrol (PD) veya üçünün birlikte kullanıldığı kontrolör (PID) yapılarıdır.



Şekil 2.7 DA motorunun PI kontrolü

Geleneksel kontrol yapılarında kullanılan kontrolörlerin transfer fonksiyonları ifade edilirse PD kontrolör için,

$$G_c(s) = K_p + K_D \cdot s \quad (2.22)$$

PI kontrolör için,

$$G_c(s) = K_p + \frac{K_I}{s} \quad (2.23)$$

PID kontrolör için,

$$G_c(s) = K_p + K_D \cdot s + \frac{K_I}{s} \quad (2.24)$$

olarak elde edilir. Transfer fonksiyonlarından görüldüğü gibi, kontrolör hata işaretinden sistem girişini üretmek için, belirlenmesi gereken parametreler (K_p , K_D , K_I) kontrolörün tasarım problemini ifade eder. Bu parametreler sabit katsayılar olmak üzere, K_p oransal

kontrolörün, K_D türev kontrolörün, K_I integral kontrolörün belirlenmesi gereken parametreleridir.

Geleneksel kontrolörlerin sistem performansı üzerine etkileri kısaca şöyle ifade edilebilir. PD kontrolör, sisteme sönüm eklerken sistem duyarlılığını artırır ve sistem cevabını hızlandırır. Ancak, sürekli rejim hatasını gidermez. PI kontrolör, sürekli rejim hatasını giderir. Yükselme ve yerleşme zamanını kötüleştirir. Bu problemleri azaltmak için PID (oransal+türev+integral) kontrolör kullanılır. (Ogata, 1990)

Bu alt bölümde, bir DA motorunun hız kontrolü için PI kontrolör tasarımı verilecek ve motorla doğrusal olmayan bir aktüatör olması durumunda kontrol sisteminin performansındaki bozulmalar incelenecektir. Motorun parametreleri

$$V = 220V$$

$$P = 3,3kW$$

$$n = 1500 \text{ dev/dak}$$

$$I_a = 17A$$

$$R_a = 0,6\Omega$$

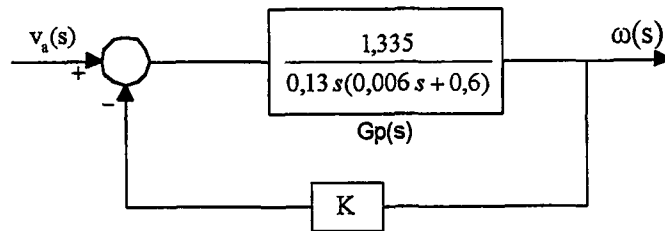
$$L_a = 6mH$$

$$J_m = 0,13 \text{ kgm}^2$$

$$B = 0$$

$$K = 1,335$$

Bu parametrelere göre, doğru akım motorunun blok diyagramı şekil 2.8 deki gibi çizilebilir.

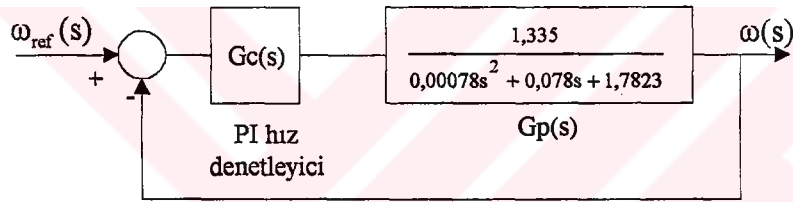


Şekil 2.8 Kullanılan doğru akım motorunun blok diyagramı

Motora kontrolör eklenmediği durum için kapalı çevrim transfer fonksiyonu,

$$\frac{\omega(s)}{v_a(s)} = \frac{K}{JL_a s^2 + (R_a J + BL_a)s + (K^2 + R_a B)} = \frac{1,335}{0,00078s^2 + 0,078s + 1,7823} \quad (2.25)$$

şeklinde olur. Motorun açık çevrim transfer fonksiyonunun s düzlemi üzerinde $s_1 = -35.33$ ve $s_2 = -64.69$ olan iki kutbu mevcuttur. Açık çevrim transfer fonksiyonundan, sistemin sönüm oranı $\zeta=1$ olarak bulunur ($\zeta = \cos\theta$). Sönüm oranı, kararlı sistemler için genellikle 0.707 civarında seçilmektedir (Kuo, 1991). Sönüm oranını uygun bir değere çekip, maksimum aşmanın aşırı olmadığı ve referans giriş değerine daha kısa sürede yerleşmesini sağlamak için doğru akım motoruna PI kontrolör eklenir. PI hız kontrolör ile birlikte doğru akım motorunun blok diyagramı şekil 2.9'da verilmiştir.



Şekil 2.9 Kontrolör ilave edilmiş doğru akım motoru

Geleneksel kontrolörlerin tasarımı için "root-locus", "bode diyagramları", "nichols eğrileri",...vs gibi teknikler kullanılmaktadır. Bu çalışmada root-locus tekniği kullanılmıştır. Root locus tekniği için açık çevrim sistemin transfer fonksiyonu;

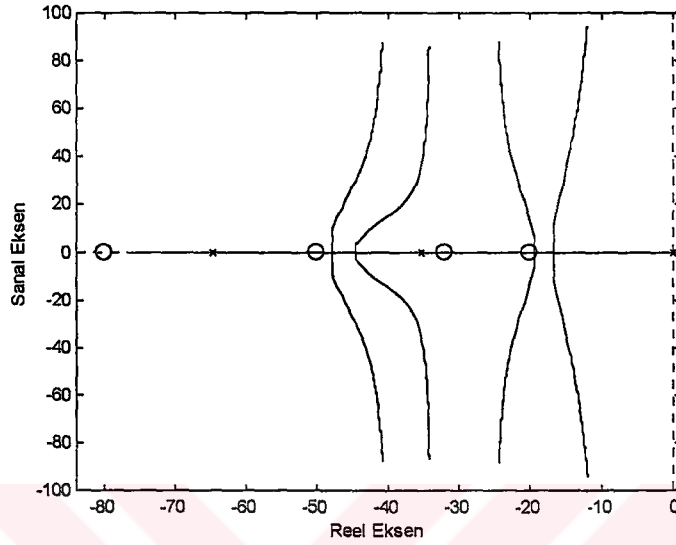
$$G(s) = G_c(s)G_p(s) = \frac{K_p \left(s + \frac{K_i}{K_p} \right)}{s} \left(\frac{1,335}{0,00078s^2 + 0,078s + 1,7823} \right) \quad (2.27)$$

şeklindedir.

Burada $a = \frac{K_i}{K_p}$ olarak PI kontrolörün sisteme eklediği sıfırı ifade etmektedir.

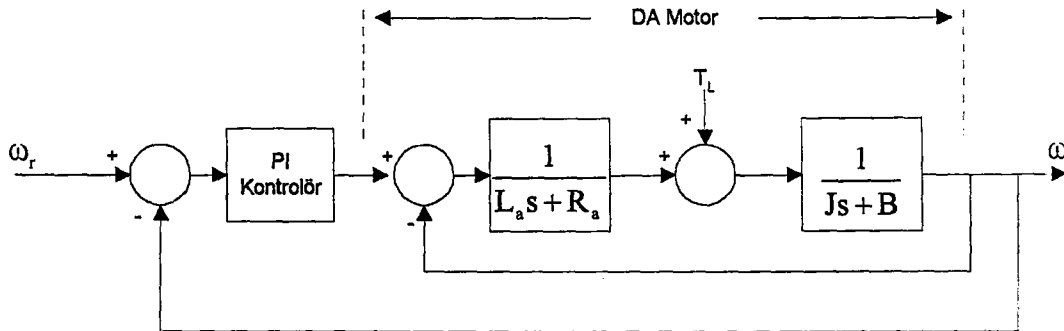
Denklem 2.27'den PI kontrolörle birlikte sistemin 3. dereceden olduğu görülmektedir.

Bu sistem için istenilen ζ ve ω_n değerlerini karşılayacak bir PI tasarlamak oldukça zordur. Bu durum şekil 2.10'da verilen sistemin root kontur grafiğinden de anlaşılmaktadır.



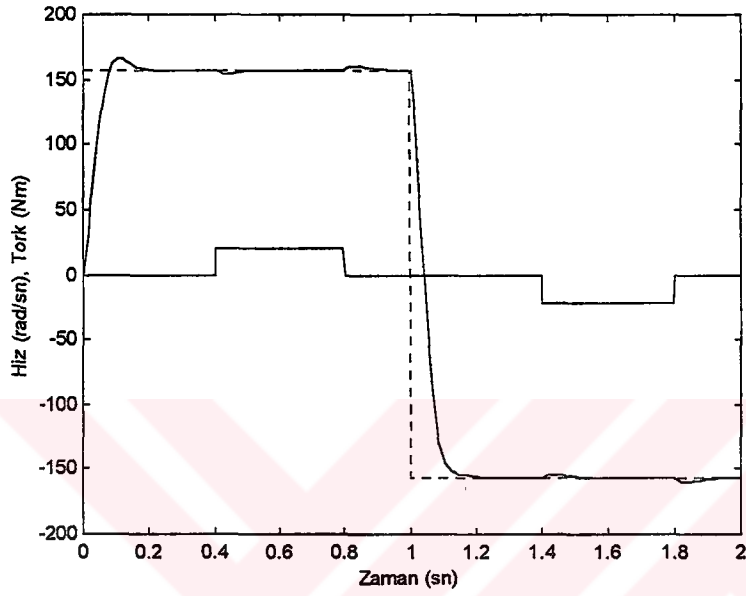
Şekil 2.10 Farklı PI parametreleri için Root contour grafiği

Root cotour grafiğinden $\zeta = 0,707$ olacak şekilde bir PI tasarlamak için, kontrolörün sisteme eklemiş olduğu sıfırın yerleştirileceği yerin sistem cevap hızına (ω_n) önemli bir etkisinin olacağı görülmektedir. En büyük ω_n değeri $a=20$ (yada $a \ll 20$) için elde edilebilmektedir. Ancak bu durumda, arzu edilen kutupların dışında oldukça baskın bir reel kutup olacağından sistemden beklenen davranış elde edilemeyecektir. Sonuç olarak, $s_1=-35,33$ deki sistem kutbunu iptal eden (yada kutbun sağına en yakın) bir PI kontrolör sıfırı ile, en uygun kontrol performansı sağlanabilecektir. $a=32$ alınarak root locus tekniği ile kontrolör parametreleri $K_p = 1,3361$ ve $K_I = 42,8372$ olarak bulunur. Tasarlanan PI kontrolörle DA motorunun blok diyagramı Şekil 2.11'de verilmiştir.

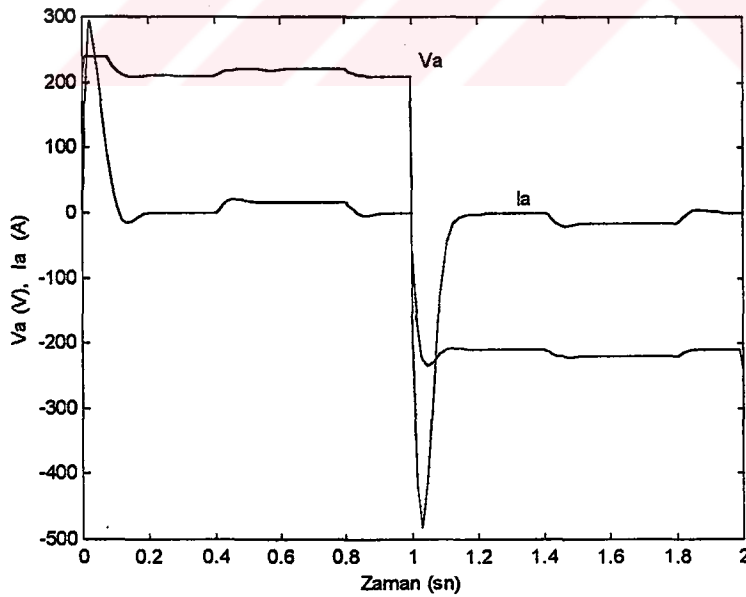


Şekil 2.11 Kontrolör ve doğru akım motoru blok diyagramı

Nominal hızda (157 rad/sn) bir referans giriře karşılık motordan alınan hız çıkış grafiği şekil 2.12 de görölmektedir. 2 saniyelik simülasyon süresinin 0,4. ve 1,4. saniyelerinde 0,4 saniye süreyle, nominal (21 Nm) yük momenti uygulanmıştır. Ayrıca, PI kontrolör tarafından üretilerek motora uygulanan giriş gerilimi de motorun nominal çalışma geriliminin yüzde on fazlası ile sınırlandırılmıştır.



(a)

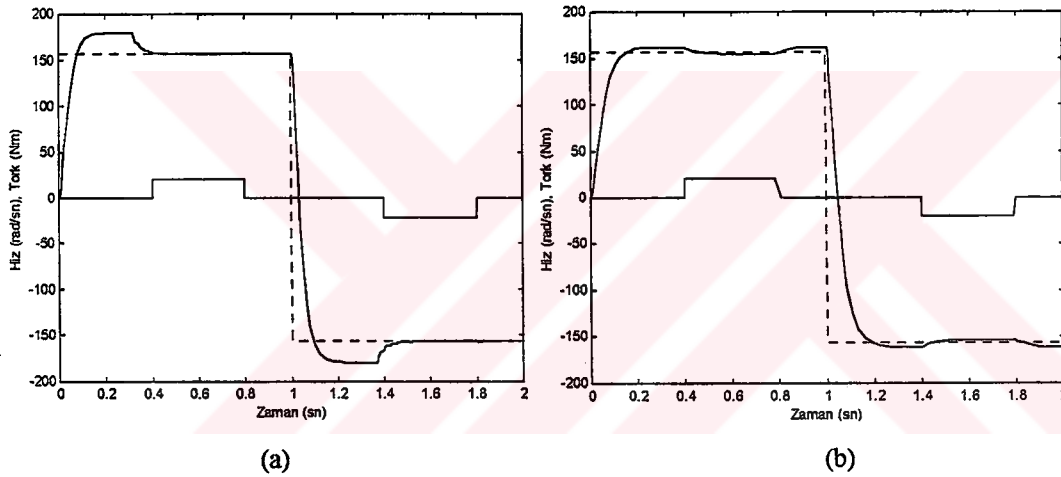


(b)

Şekil 2.12 Nominal referans hız girişi için PI kontrollü motorun
a) Hızı b) motor akımı (i_a) ve gerilimi (v_a)

Motor hızının nominal hız referansına ulaşabilmesi için, endüviye uygulanan nominal gerilimden büyük bir gerilim PI tarafında üretilmektedir. Ancak motora uygulanabilecek değerler sınırlı olduğundan, motorun referans hıza ulaşmasını geciktirmiştir. Şekil 2.12-(b)'deki grafikler ise motora uygulanan ve gerilimini ve motorun çekmiş olduğu ia akımını göstermektedir. Şekilden de görüldüğü gibi, motorun ani hız değişimlerinde çekilen akım artarken, hız istenilen değere ulaştığında, akımda nominal değerine ulaşmıştır.

Şekil 2.13'de doğrusal olmayan iki farklı aktüatörle kontrol sisteminin performansı görülmektedir. Şekilden de görüleceği gibi sistem cevabı oldukça bozuktur. Kontrolör çıkışında doğrusal olmayan aktüatör bulunduğundan uygun PI parametrelerinin analitik olarak tasarımı imkansızdır. Bu durumda PI parametreleri ancak deneme yanılma yoluyla seçilebilir, bu da bir tasarımcı için arzu edilmeyen bir durumdur.



Şekil 2.13 PI kontrolörün aktüatör ile performansı
a) $f(.)=\tanh$ aktüatör ile b) Tristörli doğrultucu kullanıldığında

Doğru akım motorlarının hız kontrolünde, tasarım kolaylığından dolayı PI, PD, PID gibi geleneksel kontrolörler sıklıkla kullanılır. Ancak Şekil 2.13'deki sonuçlardan görüldüğü gibi, sistemde doğrusal olmayan bir eleman olduğunda sistemin geleneksel yöntemlerle kontrolü oldukça güçtür. Bu yüzden yeni kontrol yöntemleri geliştirilerek istenen performans sağlanmaya çalışılmaktadır. Bu yeni kontrol yöntemlerinden biri, yapay sinir ağları (YSA) ile sistem kontrolüdür. Bu tez çalışmasında, doğru akım motorların hız kontrolünde adaptif aktivasyon fonksiyonlu YSA'nın kontrol performansının araştırılması amaçlanmıştır.

3. KONTROL SİSTEMLERİNDE YAPAY SİNİR AĞLARI

3.1. Yapay Sinir Ağlarının Tarihsel Gelişimi

İnsan davranışlarını taklit etmek amacıyla geliştirilen yapay zeka uygulamalarının bir alt dalı olan yapay sinir ağları, ortaya çıkışından günümüze değin bir çok aşamalardan geçmiş ve son yıllardaki teknolojik gelişime paralel olarak gelişimini çok hızlı bir şekilde devam ettirmektedir.

Yapay sinir ağları, anatomi, psikoloji ve nörobiyoloji alanlarındaki araştırmalarla paralel gelişmektedir. 1943 yılında McCulloch ve Pitts'in yayınladıkları makale yapay sinir ağları konusundaki ilk makale sayılmaktadır. Bu makalede sinirlerin giriş faaliyet seviyesi, belirli bir eşik değerine eriştiğinde aktif hale geçen açık/kapalı anahtar gibi davrandıklarını ve mantık fonksiyonlarını hesaplamak için çeşitli şekillerde bir araya getirilebileceklerini göstermiştir. 1949'da nöropsikolog Hebb yayınladığı makalede beynin öğrenme yeteneğini bir modelle açıklamıştır. 1954'te M. Minsky yapay sinir ağları konusunda doktora yapmış ve 1960'lı yıllarda yapay sinir ağlarına ilgi büyük ölçüde artmıştır. Bu yıllarda F. Rosenblatt, McCulloch ve Pitts'in eşik değeri yaklaşımını kullanarak tecrübeye dayanan, öğrenebilen eleman ağlarını meydana getirmiştir. Yine bu yıllarda S. Grossberg, verbal öğrenmede deney sonuçlarıyla ifade edilen ağların diferansiyel denklemlerini kurmuş; daha sonra gerçek biyolojik sinir ağlarını ifade eden genel denklem setini çıkarmış ve yaptığı araştırmaların sonunda "Adaptif Rezonans Teorisi"ni geliştirmiştir. 1969 yılında M. Minsky ve S. Papert yapay sinir ağlarındaki sınırlamaları inceledikleri "perceptrons" adlı kitabı yayınlamışlardır. Bu kitapta tek katmanlı ağların ayrıcalıklı veya işlemini gerçeklemek gibi basit sorunları çözemeyecekleri ispatlanmıştır.

Yapay sinir ağlarındaki büyük gelişmeler 1980'li yıllarda gerçekleşmiştir. 1980'lerin başında J. Hopfield, gelişmeyle birlikte rasgele bulunan etkenlerin probleme karıştığını ve her çözüm yolunda problemin başlangıç halinde bulunduğu işaret etmiş, bu nedenle her hesaplamanın doğru çözüm yolu için sürekli yenilenen bir mekanizmaya ihtiyaç olduğunu belirterek bu iş için geliştirdiği stratejiyi yapay sinir ağlarında kullanmıştır. 1986 yılında D. Rumelhart, G. Hinton ve R. Williams geriye yayılım öğrenme kuralıyla çok tabakalı ağ sistemlerinin bir çok problemi çözmek için eğitilebileceğini göstermiştir. Son yıllarda bilgisayar teknolojisindeki büyük

gelişmelerden dolayı, yapay sinir ağlarının yazılım ve donanım uygulamaları hızla devam etmektedir.

Tüm bu gelişmelere rağmen yapay sinir ağlarının sadece bağlantılar arasındaki ağırlık matrislerinin değiştirilebilmesi ve yapının sabit kalması yapay sinir ağlarının yeteneklerini önemli ölçüde sınırlamaktadır. Yapay sinir ağlarının dinamik ve öğrenme özelliklerinin matematiksel tanımı ve modellenmesi konusundaki ilerlemeler bu konudaki ümit verici gelişmeler olarak gösterilebilir.

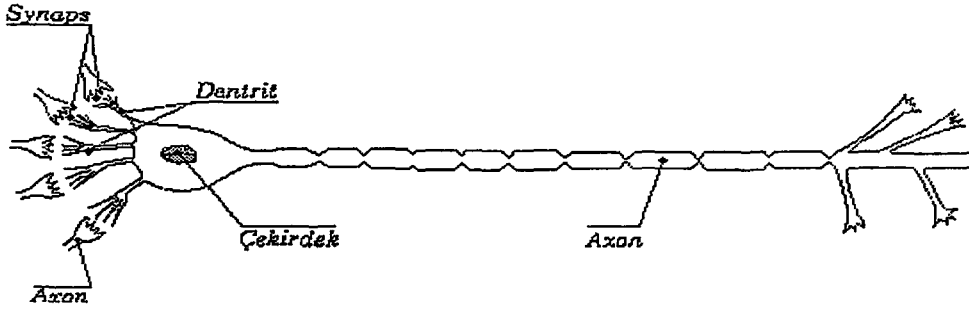
3.2. Sinir Ağlarının Biyolojik Yapısı

Canlıları dış ortamdaki değişikliklere karşı kendi iç ortamını belirli sınırlar içerisinde koruyan sinir sistemi, sinir hücrelerinden meydana gelmiş bir yapıdır. Sinirler vücuttaki diğer hücrelerle birçok ortak özelliklere sahiptir, fakat ilave olarak elektro kimyasal işaretleri algılama, işleme ve iletme özellikleri de vardır. Canlı bir sinir sisteminin temelini oluşturan bir sinir hücresi şekil 3.1’de görülmektedir.

Bir sinir hücresi, bir hücre gövdesi ile dendrid ve akson adı verilen uzantılardan meydana gelir. Sinir hücresi, dendrid adı verilen bir kısım uzantılarla diğer sinir hücrelerinden aldığı işaretleri hücre gövdesine taşır ve hücre gövdesinde toplanan bu işaretler değerlendirilerek bir çıkış işareti üretilir ve bu işaretler akson adı verilen uzantılar vasıtasıyla diğer sinir hücresine gönderilir. Bir sinir hücresinde birçok dendrid olmasına karşın tek bir akson vardır. Yani bir sinir hücresinde birçok giriş bulunmasına rağmen tek bir çıkış vardır.

Sinir hücrelerinin en büyük özelliği ise, sinir hücresinin çevreden gelen işaretlere ya cevap vermesi ya da vermemesidir. Eğer sinir hücresine gelen işaret, eşik değerini aşarsa hücre tarafından kabul edilir ve cevaplandırılır, eşik değerini aşamıyorsa yok kabul edilerek hiçbir cevap üretilmez.

İki sinir hücresi arasında birinin dendridi ile diğerinin aksonu arasında bir bağlantı vardır ve bu bağlantı yerine sinaps adı verilir. Bu bağlantı yerinde sinirler birbirine bağlı değildir. İki hücre arasında küçük bir bağlantı bulunmamasına rağmen iki sinir hücresi darbeleri iletme için birbirine yeterince yakındır.



Şekil 3.1 Biyolojik sinir hücresi

Birinci sinir hücresinin aksonu, sinir hücresinden aldığı işaret sonucunda şişer ve bu şişlik ikinci sinir hücresinin dendridinde bulunan algılayıcılar tarafından kaydedilerek hücre zarının Na^+ ve K^+ iyonları geçirebilme yeteneğini değiştirme yönünde etkileyen kimyasal reaksiyon başlar. Böylece hücre zarı iyonların zarı geçmesine izin verir ve zar potansiyelini geçici olarak değiştirir (Harvey,1994).

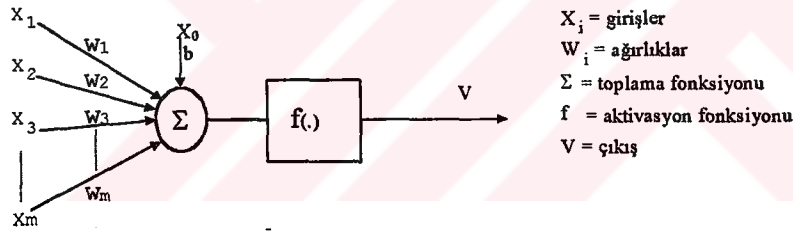
Biyolojik beynin en önemli özelliklerinden birisi de, öğrenme olayıdır. İnsanlar ve hayvanlar, sürekli olarak içerisinde bulundukları çevre ile ilişki neticesinde bir öğrenme işlemi içerisindeyler. Öğrenilen her yeni bilgi, beynin fonksiyonlarını hemen etkileyerek, yapılan davranışlara yansır. Yapay sinir ağlarının gerçekleştirilmesinde bu özellik esas teşkil eder.

3.3. Yapay Sinir Ağının Tanımı ve Modeli

Yapay sinir ağları genel bir deyişle, insan beyninin çalışma prensibini taklit etme esası üzerine kurulmuş bilgi işleme yöntemleridir. Bu ağlar birbirine paralel olarak bağlanmış işlem elemanlarından (yapay sinir ağı hücresi, nöron, ünite, birim, düğüm) ve onların hiyerarşik bir organizasyonundan oluşurlar. Yapay sinir ağlarının paralel yapıları, bilgisayarları geleneksel yöntemlerden çok daha farklı kullanarak özellikle seri bilgisayarlarda bilinen yöntemlerle yapılması mümkün olmayan ve çok zor olan bir takım işlevleri (ses ve görüntü tanıma gibi) rahatlıkla yapmaları, yapay sinir ağlarını üstün kılmıştır. Yapay sinir ağının temel düşüncesiyle insan beyninin fonksiyonları arasında benzerlikler vardır. Yapay sinir ağları, her ne kadar temel yapı itibariyle bir kısım özellikleri insan beyninin fiziki özelliklerinden esinlenerek ortaya atılmış ise de, kesinlikle şu andaki halleri ile insan beyninin ne tam ne de yaklaşık bir modeli olarak değerlendirilemezler (Özmetel, 1996).

İnsan beyninin ne olduğu ve nasıl çalıştığı henüz kesinlik derecesinde keşfedilmiş sayılmaz. Günümüzde her ne kadar karmaşık matematiksel hesaplamaları ve hafıza işlemlerini eldeki mevcut bilgisayarlarla hızlı ve doğru yapmak mümkün ise de, aynı bilgisayarlarla beynin birçok basit fonksiyonunu (görmek, duymak koklamak, ...vs.) yerine getirmek ya mümkün olmamakta veya oldukça zor olmaktadır. Aynı şekilde biyolojik beyin, tecrübe ile öğrenme ve bilgiyi kendi kendine yorumlama, hatta eksik bilgilerden sonuçlar çıkartma kabiliyetine sahiptir. Bu, daha çok biyolojik sistemlerin, hücreler üzerinde dağıtılmış bilgiyi paralel olarak işleme özelliklerinden kaynaklanır. Hücreler birbirine bağlı ve paralel çalıştıklarından bazılarının işlevini yitirmesi halinde, diğerleri çalıştığı için sinir sistemi, fonksiyonunu tamamen yitirmez. İşte yapay sinir ağları, bu özellikleri bünyesinde toplayacak şekilde geliştirilmektedir.

Yapay sinir ağları, biyolojik sinir ağları gibi yapay sinir hücrelerinin veya işlem elemanlarının bir araya gelmesinden oluşur. Her bir işlem elemanı şekil-3.2 'da görüldüğü gibi beş parçadan oluşur. Bunlar; girişler, ağırlıklar, toplama fonksiyonu, aktivasyon fonksiyonu ve çıkıştır.



Şekil 3.2. Yapay sinir hücresinin modeli.

Tıpkı biyolojik sinir hücresi gibi, işlem elemanına birden fazla giriş gelmekte ve sadece bir çıkış üretilmektedir. Girişler, dendritlere benzer şekilde diğer yapay hücrelerden bağlantılar vasıtasıyla işlem elemanına bilgi gelmesini sağlarlar. Bazı durumlarda bir işlem elemanı kendisine de bilgi geri gönderebilir (feedback). Bahsedilen bu bilgiler elemanlar arasında bulunan bağlantı hatları üzerinde depolanır. Her bağlantının bir ağırlığı vardır. Bu ağırlık bir işlem elemanının diğer üzerindeki etkisini gösterir. Ağırlık büyüdükçe etki de büyür. Ağırlığın sıfır olması hiç bir etkinin olmaması, negatif olması ise etkinin ters yönde olması demektir. Bu ağırlıklar sabit olabildikleri gibi değişken de olabilirler. Toplama fonksiyonu, bir işlem elemanına gelen net girişi hesaplayan bir fonksiyondur. Net giriş genellikle gelen bilgilerin ilgili

bağlantıların ağırlıkları ile çarpılıp toplanması ile belirlenir. Bu nedenle adı, toplama fonksiyonu olarak verilmiştir. Aktivasyon fonksiyonu da, toplama fonksiyonu tarafından belirlenen net girişi alarak, işlem elemanının çıkışı belirleyen fonksiyondur. Genel olarak türevi alınabilen bir fonksiyon olması tercih edilir (şekil 3.3). Toplama ve çıkış fonksiyonları, ilgili probleme bağlı olarak farklı şekiller alabilirler. İşlem elemanının çıkış ünitesi ise çıkış fonksiyonunun ürettiği dürtüyü diğer işlem elemanlarına veya dış dünyaya aktarma işlevini yapar. İşlem elemanları ağırlık topolojik yapısına bağlı olarak tamamen birbirinden bağımsız ve paralel olarak çalışabilirler. Şekil-3.2’de verilmiş olan yapay sinir hücresine $x_1, x_2, x_3 \dots x_m$ giriş işaretleri, uygulanmaktadır. Her giriş işareti, $w_1, w_2, w_3 \dots w_m$ ilgili ağırlık değerleriyle çarpılırken basit hücre modelinde, v çıkış ifadesi aşağıdaki şekilde ifade edilebilir.

$$u = \sum_{i=0}^m x_i w_i - b \quad (3.1)$$

$$v = f(u) \quad (3.2)$$

Burada, girişler x , ağırlıklar w vektörüyle toplu olarak gösterilir. Giriş işaretleri biyolojik hücrelerin sinapsislerindeki işaretlere karşı düşen giriş hattındaki elektriksel etkinliği, ağırlık vektörüyle biyolojik sinirlerdeki sinaptik gücü ifade etmektedir. b , ise eşik değeri olarak adlandırılır. $f(.)$ fonksiyonu, bir lineer veya lineer olmayan fonksiyondur. Şekil 3.3’te $f(.)$ fonksiyonu için kabul görmüş gerçek biyolojik sinirlerin transfer fonksiyonuna yakın lineer olmayan fonksiyonlar görülmektedir. Bu şekilde v çıkış etkinliği VAR/YOK olarak iki durum arasında sınırlandırılmış olmaktadır. Bu iki durum arasındaki durum değiştirmenin ayrıntıları $f(.)$ fonksiyonunun ayrıntılarıyla belirlenir. Yani v hücre çıkış etkinliği, hücreye girişlerden gelen toplam etkinliğe, bir $f(.)$ aktivasyon fonksiyonu üzerinden bağlıdır. Şekil 3.3-b’de verilen lineer aktivasyon fonksiyonu için,

$$f(.) = u \quad (3.3)$$

Şekil 3.3-a’da verilen aktivasyon fonksiyonu için,

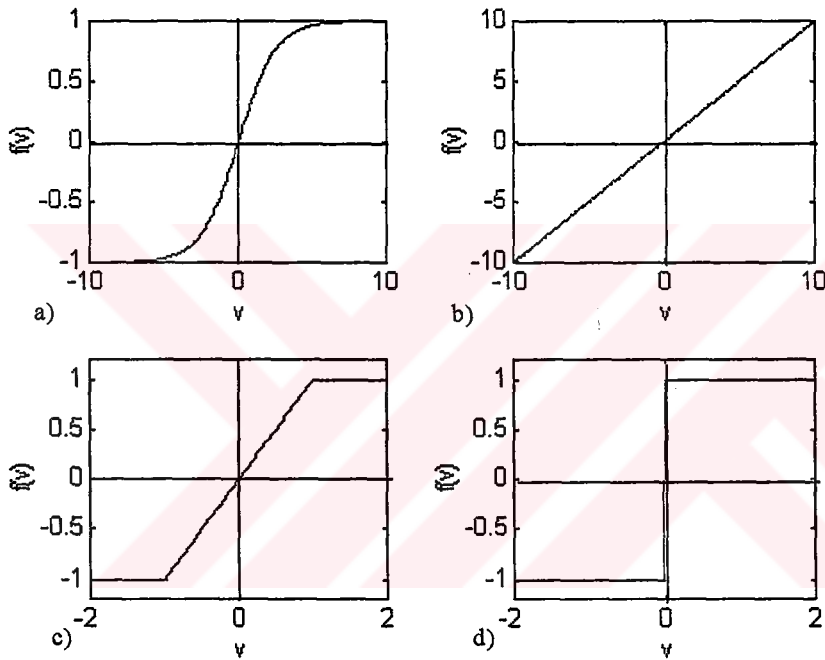
$$f(.) = \tanh(u) \quad (3.4)$$

Bunlardan başka tanımlanabilecek farklı bir aktivasyon fonksiyonu ise,

$$f(u)=v \quad (3.5)$$

ile ifade edilebilir.

Aktivasyon fonksiyonları, işlem elemanlarının sınırsız sayıdaki girişini önceden belirlenmiş sınırdaki çıkış olarak düzenler. En çok kullanılan dört tane aktivasyon fonksiyonu vardır. Bunlar, lineer, rampa, basamak ve sigmoid fonksiyonudur. Şekil 3.3’de bu fonksiyonlar gösterilmiştir.



Şekil 3.3. Aktivasyon fonksiyonları.

3.4. Yapay Sinir Ağlarının Yapıları ve Çeşitleri

Yapay sinir ağlarının görevlerini gerçekleştirmede, sahip oldukları fiziksel yapının da önemi vardır. Bugün 50'ye yakın farklı yapılanma görülmekte ve bu sayı her geçen gün artmaktadır (Özmetler, 1989). Farklı yapılaşma, işlem elemanlarının birbirleri ile olan bağlantılarından ve uygulanan öğrenme kuralından kaynaklanmaktadır. İşlem elemanları ya tamamen birbirleri ile bağlantılı veya yerel olarak gruplar halinde bağlantılı olabildikleri gibi değişik şekilde de birbirleri ile bağlanabilmektedirler. Bilgi akışı bu bağlantılar üzerinden tek yönlü olduğu gibi, çift

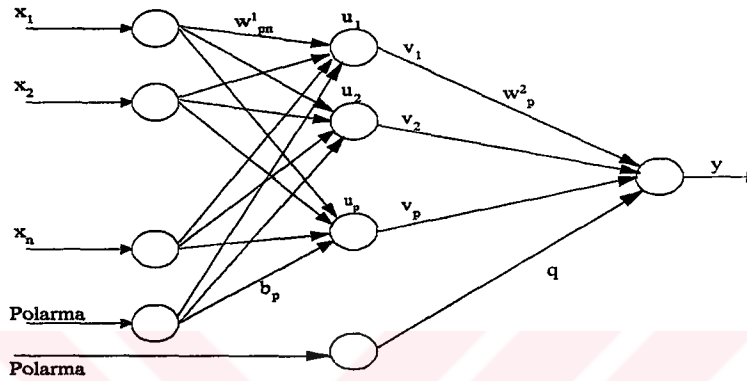
yönlü de olabilir. Bir gurup işlem elemanı bir araya gelerek bir katman oluştururlar. Genel itibariyle yapay sinir ağlarında 3 tür katman bulunur. Sinir ağının dış dünya ile bağlantısını kuran girdi katmanı, gelen bilgileri işleme kabiliyetine sahip ara katmanlar ve sinir ağının kararlarını dış dünyaya aktaran bir çıkış katmanı (Şekil 3.4). Giriş katmanında çoğu zaman bilgi işleme söz konusu olmaz. Bu katmandaki işlem elemanları aldıkları bilgiyi herhangi bir değişikliğe uğratmadan ara katmandaki işlem elemanlarına aktarırlar. Burada sözü edilen bilgi, sinir ağının işlem elemanları arasındaki bağlantı hatları üzerinde bulunan ağırlıklarla ifade edilir. Dolayısı ile bilgi, bütün ağa dağıtılmış demektir. Bu bilgiler, yapay sinir ağının belleğidir veya diğer bir deyişle yapay sinir ağları eğitildikten sonra kullanılacağı sistem için bir veri tabanıdır.

İşlem elemanları, onların birbirleri ile ilişkileri ve katmanlar arası ilişkiler değişik yapısal modellerin oluşmasına neden olmaktadır. Bu modelleri burada tek tek izah etmek oldukça fazla yer ve zaman gerektireceğinden, sadece örnek bir yapay sinir ağı modeline değinilecektir. Şekil 3.4’de tek yönlü bağlantıların olduğu, bir katmandaki bilgilerin sadece bir sonraki katmana geçtiği ve yine bir katmandaki işlem elemanlarının bir sonraki katmana tamamen bağlandığı bir ağ yapısı gösterilmiştir. Bilinen meşhur geriye yayılım ağı (ileri beslemeli, katmanlı ağ) modeli bu tür yapıya örnek olarak verilebilir.

Araştırmacılar birçok farklı yapay sinir ağ modelleri yapmışlar ve bunları eğitmek için birçok algoritma geliştirmişlerdir. Bu ağ modellerinin birçoğu biyolojik sinirlerden esinlenerek veya en azından biyolojik olarak geçerli kabul edilmiştir. Bu modellerin her biri belli uygulama alanlarında başarılı olmuşlardır. Bu bölümde sistem modelleme ve kontrol alanında en çok kullanılan ileri beslemeli yapay sinir ağı yapısı hakkında detaylı bilgi verilmiştir.

Yapay sinir ağlarının en tanınmış olan ileri beslemeli ağ, çok katmanlı bir ağıdır. Widrow-Hoff öğrenme kuralının bir uzantısı olarak Rumelhart, Hinton ve Williams tarafından 1986 yılında gerçekleştirilmiştir (Rumelhart v.d., 1986). Giriş ile çıkış katmanları arasında, birden fazla katmanın kullanıldığı yapay sinir ağı sistemidir. Gizli katman olarak isimlendirilen bu ara katmanlarda, aracısız giriş alamayan ve aracısız çıkış veremeyen işlem elemanları vardır. Katmanlardaki hücreler arasında ve bir katmandan önceki katmana geriye doğru bir bağlantı yoktur. Şekil 3.4’de ileri beslemeli çok katmanlı yapay sinir ağının genel yapısı verilmiştir.

Eğitici öğrenme kullanarak eğitilen ileri beslemeli ağda, girişe bir giriş vektörü uygulanır ve çıkıştan bir çıkış vektörü elde edilir. Amaç her giriş vektörü için, ağı üreteceği cevabın istenilen çıkış vektörünün aynısı veya yeterince yakın değerlerde olmasını sağlayacak şekilde ağırlık vektörleri oluşturmaktır. Yakın zamana kadar ileri beslemeli ağlar, eğitim algoritmalarının biyolojik sistemlere uygun olmamasına rağmen, son zamanlarda geliştirilen öğrenme algoritmalarıyla yaygın bir şekilde kullanımı sağlanmıştır.



Şekil 3.4 İleri Beslemeli ağ yapısı

İleri beslemeli ağın çıkışını elde etmek için, ileri yönde (girişten çıkışa) hesaplamalar yapılır. j . Düğümün net girişi u_j , girişlerin bir lineer fonksiyonudur ve w_{ji} ağırlıklarıyla birlikte,

$$u_j = \sum_i x_i w_{ji} + b_j \quad (3.6)$$

şeklinde ifade edilir. Net girişin lineer olmayan fonksiyonu olan v_j ,

$$v_j = \frac{1 - e^{-u_j}}{1 + e^{-u_j}} \quad (3.7)$$

ile ifade edilir. Saklı katman çıkışı v_j ile çıkış düğümü arasındaki ilişki doğrusaldır. Ağ çıkışı,

$$y = \sum_j v_j w_j^2 + q \quad (3.8)$$

şeklinde elde edilir. Giriş değerlerinin polarma ile birleştirilmesi, öğrenmeyi kolaylaştırmaktadır. Başlangıçta tüm ağırlıklara küçük rasgele değerler atanır. Bu ağın simetrisini bozar ve ağırlıkların büyük değerlerle dolmuş hale gelmesini engeller. Giriş tabakasına $x_1, x_2 \dots x_n$ giriş vektörü uygulanır (3.6) - (3.8) ifadeleri ile ağ çıkışları $y_1, y_2 \dots y_m$ hesaplanır. (Aktivasyon fonksiyonu olarak sigmoid fonksiyonu kullanıldığı kabul edilmiştir.)

Katmanların boyutu hakkında genelleme yaparak bir değer vermek mümkün değildir. Giriş ve çıkış katmanlarının boyutu uygulamaya göre tespit edilir. Ancak saklı katmanın boyutu belirsiz olmakla birlikte genellikle giriş katmanından daha büyük olmalıdır. Eğer saklı katmandaki hücre sayısı az seçilirse, giriş değerleri arasından gerekli özellikleri tanımlamayabilir; gereğinden fazla olması halinde ise, genelleme ve soyutlama hatasına düşebilir. Ortaya çıkan bu sorunun en iyi çözümü ortalama bir değer almaktır.

3.5. Yapay Sinir Ağlarında Öğrenme ve Öğrenme Algoritmaları

Daha önce belirtildiği gibi bir yapay sinir ağının sahip olduğu bilgi, işlem elemanları arasındaki bağlantı hatları üzerinde saklanır ve ağırlıklar vasıtası ile gösterilir. Ağ, olaylar hakkında giriş ve çıkışlar arasındaki ilişkiyi, elde bulunan mevcut örneklerden genellemeler yaparak öğrenir ve bu genelleme ile yeni oluşan ve ortaya çıkan daha önce hiç görülmemiş olaylar hakkında karar verir. Yani ağa, bir örnek olay gösterildiğinde, giriş katmanından alınıp ara katmanlarda işlenerek, ağın o olay hakkında ürettiği sonuç, çıkış katmanına sunulur. Bu bilgiyi işleme, ağın sahip olduğu tecrübeye göre, bilginin ara katmanlarda çağrıştırılması ile gerçekleştirilir. Bu çağrıştırma olayı modelden modele değişmektedir. Mesela, ara katmanlardaki işlem elemanları sahip oldukları bağlantılar ile kendi kararlarını üretir ve çıkış katmanındaki işlem elemanlarına gönderirler. Çıkış katmanındaki işlem elemanları da, yine ilgili ağırlıkları kullanarak ağın en son kararını oluştururlar. Bu ağırlıklar, tıpkı ilgili olayın belirli özelliklerini hafızada saklayan elemanlar gibi düşünülebilirler. Bilgi işleme ise (örüntü tanıma gibi), bir olay gösterildiğinde hafızadan ilgili özellikleri çağırarak ve bunlar ile ilgili girişleri birlikte analiz ederek karar vermek şeklinde yorumlanabilir.

Yapay sinir ağının öğrenme işlemini gerçekleştirebilmesi için, sahip olduğu bütün ağırlıkların, ilgili problemde öğrenilmesi istenen özellikleri genelleştirecek

şekilde doğru değerlere sahip olması gerekir. Bu doğruluk ne kadar artarsa ağıın öğrenme işlemi o kadar iyi olur. Doğru ağırlık değerleri, bir öğrenme kuralına göre tespit edilirler. Çoğunlukla bağlantılara başlangıç değerleri olarak rasgele ağırlıklar atanır ve bu ağırlıklar eldeki örnekler incelendikçe bir kurala göre değiştirilerek doğru ağırlık değerleri bulunmaya çalışılır. Kısaca belirtmek gerekirse öğrenme kuralları, bir işlem elemanının örnekleri gördükçe kazandığı tecrübeye göre, ilgili bağlantı ağırlıklarını nasıl değiştireceğini belirleyen algoritmalarıdır. Öğrenme yöntemleri temelde üç grupta toplanır (Zurada, 1992).

Eğitici Öğrenme : Bu yöntemde, dışardan bir eğitici sinir ağına müdahalesi söz konusudur. Eğitici, sinir ağına ilgili giriş için üretmesi gereken sonucu sinir ağı sistemine verir. Yani yapay sinir ağına giriş/çıkış ikilisinden oluşan örnekler sunulur. Bu ikili, ağıın öğrenmesi gereken özellikleri temsil eder. Ağ giriş kısmını alır ve o anki bağlantı ağırlıklarının temsil ettiği bilgi ile bir çıkış oluşturur. Bu çıkış, olması gereken çıktı ile mukayese edilir ve aradaki hata tekrar ağa aktararak ağırlıklar bu hatayı azaltacak şekilde değiştirilir.

Eğitici Öğrenme : Bu durumda hiç bir eğiticiye ihtiyaç yoktur. Bu nedenle çoğu zaman buna kendi kendine organize olma (self-organized learning) da denilmektedir. Ağ, kendine gösterilen örnekleri alır ve belli bir kritere göre sınıflandırır. Bu kriter önceden bilinmeyebilir. Ağ, kendi öğrenme kriterlerini kendisi oluşturmaktadır.

Takviyeli Öğrenme : Bu yöntemde de yine eğiticiye ihtiyaç vardır. Yukarıdaki yöntemden farkı ise, bu durumda eğitici sinir ağına, ağıın üretmesi gereken sonuç yerine, onun ürettiği sonucun sadece doğru veya yanlış olduğunu söylemesidir. Bu ise ağa bir takviye sinyalinin gönderilmesi ile gerçekleştirilir. Bu yöntem, ilgili örnek için beklenen çıktının oluşturulamadığı durumlarda çok faydalıdır.

Öğrenme işleminin, başlangıçtaki rasgele seçilmiş ağırlıkların, belirli bir kurala göre değiştirilmesine dayandığı belirtilmişti. Bu öğrenme kurallarının ikisi aşağıda yer almaktadır. Bu algoritmalarda, $w_j = [w_{j1} \ w_{j2} \ ... \ w_{jp}]$ vektörü ağıdaki i . hücreye gelen n tane girişin ağırlık katsayılarını, $x = [x_1 \ x_2 \ ... \ x_n]$ vektörü ağa gelen n tane girişi, y_j ve d_j 'de sırasıyla i . hücrenin hesaplanan ve istenen çıkışını temsil etmektedirler.

Delta Kuralı (Eğimdüşme) Öğrenme Algoritması : Eğimdüşme (gradient descent) öğrenme algoritması veya “delta kuralı” öğrenme algoritması, sadece türetilbilir fonksiyonlara sahip olan ağlara uygulanabilmektedir. Eğiticili bir öğrenme algoritmasıdır. w değişimi, hatanın eğiminin tersi yöndedir. Böylece, hata fonksiyonunun minimum olduğu yere doğru hareket edilmektedir. k . örnek için, ağ çıkışı ile arzu edilen çıkış arasındaki fark (hata)’a göre yeni ağırlık değerleri,

$$w_j(k+1) = w_j(k) - \alpha \cdot \frac{\partial E(k)}{\partial w_j(k)} \quad (3.9)$$

ile elde edilir. $E(k)$, hata fonksiyonunu temsil etmektedir ve aşağıdaki gibi tanımlanmıştır.

$$E(k) = \frac{1}{2} (d(k) - y(k))^2 \quad (3.10)$$

ağ çıkışı, hata fonksiyonunda yerine yazılırsa,

$$E(k) = \frac{1}{2} [d(k) - f(w(k).x(k))]^2 \quad (3.11)$$

olur. Buradan yeni ağırlıkları bulmak için, hata fonksiyonunun ağırlıklara göre kısmi türevleri aşağıdaki gibi hesaplanır.

$$\frac{\partial E(k)}{\partial w_i(k)} = -[d(k) - f(w(k).x(k))] \frac{\partial f(w(k).x(k))}{\partial (w(k).x(k))} . x_i(k) \quad (3.12)$$

Bilinen geriye yayılma algoritmasının esası da, bu öğrenme kuralına dayanmaktadır.

Geriye Yayılım Öğrenme Algoritması : İleri beslemeli ağ yapılarında kullanılan bir algoritmadır. İleri beslemeli bir yapay sinir ağı için arzu edilen çıkış ile ağ çıkışı arasındaki fark (hata), çıkıştan girişe doğru yansıtılır. Bunun için, k anındaki hatanın ağı her bir ağırlığa göre eğimi hesaplanarak, ilgili ağırlığa uygulanacak değişim miktarı bulunur. Katmanlı bir yapay sinir ağı yapısı için bu algoritmanın uygulanışı bölüm 4’de verileceğinden burada verilmemiştir.

3.6. Yapay Sinir Ağlarının Temel Özellikleri

Yapay sinir ağlarını hem popüler yapan, hem de geleneksel bilgi işleme metotlarından ayıran özellikleri vardır. Yapay sinir ağlarının bir takım özellikleri, her ne kadar ilgili problemin yapısına ve kullanılan sinir ağı modeline bağlı olsa da, genel özellikleri aşağıdaki gibi verilebilir (Karlık, 1995).

Öğrenme : Daha önce söz edildiği gibi, yapay sinir ağları, olaylar hakkındaki ilişkiyi belirli bir algoritmaya dayanarak çözmek yerine o ilişkiyi gösteren örnekleri incelemek suretiyle çözümler üretmeyi sağlarlar. Burada, ilgili olay ile alakalı sinir ağına örneklerden başka hiç bir ön bilginin verilmemiş olması dikkat çekicidir. Ağ, kendisine gösterilen örnekleri tekrar tekrar inceleyerek aradaki ilişkiyi kavramaya çalışır. Her yeni örnek, ağın sahip olduğu bilgiye bir yenisini ekler ve bu işlem tekrar ettikçe ilgili problem hakkında bazı genellemeler yapılır.

Genelleme : Alışagelmiş bilgisayarların bir karakteristiği, istenen çıkışı üretmek için, tam olarak doğru girişlere ihtiyaç göstermeleridir. Öte yandan yapay sinir ağları, girişlerinde değişimler olsa bile doğru çıkışı üretebilirler. Mesela, basılı harfleri tanımak için eğitilmiş bir sistem, gürültüden dolayı giriş karakterlerinin bozulması halinde bile doğru olarak çalışmaktadır. Yani sistem, daha önce o tipten hiç bir şey görmemiş olmasına rağmen, insanlar gibi tamam olmayan veya kısmen hatalı girişlerle bile, harfleri tanıyabilmektedir. Çalışmalar, insanların çoğunun bir metindeki harflerin yarısından fazlası silinmiş olsa bile, o metni okuyabildiklerini göstermiştir.

Yapay sinir ağları, kendilerine gösterilen bir şekli daha önce öğrendikleri ile mukayese ederek ve aradaki benzerlikleri ortaya koyarak, belirli sınıflara ayırma özelliklerine sahiptirler.

Yapay sinir ağları tam doğru olmayan bir eğitime kümesinden, tam doğruyu çıkarabilirler. Ses tanımak için eğitilmiş bir yapay sinir ağına, gürültü tarafından bozulmuş ses verilebilir. Buna rağmen eğitimden sonra, sistem girişi bozuk ses olmasına rağmen, çıkışta ses mükemmel bir şekilde oluşturulabilir. Yani sistem, eğitim kümesinin özünü çıkarıp saklamıştır. Böylece, eksik veya gürültülü girişlere karşın uygun şekilde cevap verebilmektedir.

Hata Toleransı : Verilerde eğer bir eksiklik söz konusu olursa, geleneksel yöntemler çalışmazlar. Daha önce belirtildiği üzere, iyi eğitilmiş ve genelleme kapasitesi yüksek bir sinir ağı, kendisine takdim edilen veriler eksik olsa da karar verme işlemine devam eder. Aynı şekilde, yapay sinir ağı üzerinde birtakım problemler ve bozukluklar olabilir. Geleneksel sistemlerin tersine yapay sinir ağları, bu durumda da çalışmalarına devam ederler. Verilerdeki eksiklik veya yapay sinir ağındaki yapısal bozukluk arttıkça yapay sinir ağının performansı yavaş yavaş azalmaya başlar. Fakat sistem fonksiyonunu tamamen durdurmaz ve mutlaka bir sonuç üretilir. Bu özellikler yapay sinir ağının yapısından kaynaklanmaktadır. Çünkü ağın sahip olduğu bilgi, ağ üzerindeki hücrelerin birbiri ile olan bağlantıları üzerine dağıtılmıştır. Zaten böyle bir durumda tek bir bağlantı ve onun üzerindeki bilgi, başlı başına hiçbir zaman bir mana ifade etmez. Ancak, bir gurup halinde ve tam olarak bağlantıların birlikte düşünülmesi sonucu anlamlı bilgiler üretilir. Bundan dolayı birkaç bağlantının etkisiz hale gelmesi, sonucu ya etkilemez veya performansı yavaş yavaş düşürür. Yapay sinir ağı sahip olduğu diğer bağlantılar nedeniyle işlevine devam eder. Geleneksel sistemlerin ardışıl çalışmalarından dolayı, sistemdeki en küçük bir hatanın veya bozulmanın ulaşabileceği boyutlar düşünülürse bu özelliğin ne kadar önemli olduğu ortaya çıkar.

Hız : Bilgi işleme hızı bilgisayar teknolojisinde önemli bir yer teşkil eder. Sistemlerin her geçen gün biraz daha karmaşık olduğu, dolayısı ile daha fazla hacimde veriyi daha verimli bir şekilde işleme gerekliliği, yeni yazılım/donanım sistemlerinin zorunluluğunu ortaya çıkarmıştır. Diğer yandan insan beyni oldukça fazla bilgiyi gerçek zamanlı (real-time) olarak, oldukça hızlı bir şekilde işleyebilmektedir. Yapay sinir ağlarının da, yine birbirlerine bağlı ve paralel işlem elemanlarından oluştuğundan böyle hızlı işleyebilmeleri bu ağlara özellikle endüstriyel hayatta çok önemli olan gerçek zamanlı çalışma kabiliyeti kazandırır.

3.7 Yapay Sinir Ağlarının Avantajları

Sinir ağları sahip oldukları avantajlardan dolayı mevcut klasik tek işlemlerle çözümlerle çalışmanın zor olduğu pek çok konuda uygulanmaktadır. Aşağıda bu avantajlardan bazıları verilmiştir.

1-Öncelikle ilgili faktörleri belirlemeye daha az ihtiyaç duyarlar. Genelde ilk yapılacak iş ilişkili faktörlerin istatistiksel modelini çıkarmaktır. Sinir ağları veri sunmada ve ilgili verileri saptamada üstünlüğe sahiptir.

2-Sinir ağları genelde yüzlerce faktöre sahiptir; fakat bütün giriş faktörlerinin kabul edilen etkisi, modelin herhangi bir istatistik modelden çok daha kesin olarak zor problemlere cevap bulmasını sağlar.

3-İstatistik modeller, dolaylı yoldan ilişkileri öğrenme yoluna gider. Halbuki sinir ağları yaklaşımıyla, problemi doğrudan modellemek mümkündür.

4-Giriş faktörlerinin çok olmasından ve verideki bozucu etkiden veya donanımın hatasından kaynaklanan sorunlar azdır.

5-Sinir ağ modellerinde bulunan her sinapsis kendi işlemine sahiptir. Aynı katmanda bulunanlar arasında zaman bağımlılığı olmayıp, tamamen senkron olarak çalışırlar.

3.8. Yapay Sinir Ağları ile Sistemlerin Modellemesi ve Kontrolü

Zamanla değişmeyen ve parametreleri bilinen doğrusal sistemler için, sistemin ihtiyaç duyulan matematiksel modelini çıkarmak ve kontrol algoritması geliştirmek artık oldukça kolaydır. Parametreleri bilinmeyen doğrusal sistemler için, sistem tanınması ve kontrolünde geliştirilen yöntemlerle kararlı parametre uyarlaması gerçekleştirilmiştir (Narendra vd., 1989, Aström vd., 1995). Ancak sistemin karmaşıklığının artması gibi nedenler ihtiyaç duyulan modelin elde edilmesini engellemektedir.

Doğrusal olmayan sistemlerin tanınması ve kontrolünde, sistem teorisi ve kararlılık şartları yeterince geliştirilememiştir. Bununla birlikte sistemlerdeki belirsizlikler ve dinamik davranışlarındaki değişiklikleri tanılamak için akıllı kontrol yöntemlerine yönelik çalışmalar artmıştır (Khalil vd., 1992).

Akıllı kontrol, insanın yaratıcı ve uyarlayıcı davranışlarını taklit edebilecek niteliğe sahip sistemler geliştirmeyi hedeflemektedir. Akıllı kontrol, zamanla değişen iyi tanımlanmamış ortamlarda çalışan, ortamda ve sistem dinamiklerindeki değişimlere kendini uyarlayan, bilgileri kararlı bir yöntemle öğrenen, sistemin dinamikleri üzerine çok az kısıtlama gerektiren algoritmalara ihtiyaç duyar (Brown vd., 1994).

Son yıllarda uzman sistemler ve bulanık mantık gibi denetim yöntemlerinin yanı sıra yapay sinir ağları, doğrusal olmayan sistemlerin uyarlamalı tanınması ve denetiminde yoğun ilgi görmüştür. Yapılan çalışmalarda yeni teoriler ve modeller geliştirilerek gerçek uygulamalarda kullanmak üzerine yoğunlaşmıştır. Özellikle geleneksel yöntemlerle çözümü güç problemler için yapay sinir ağının uygulanması önemlidir. Kararlılık şartları henüz kesin olarak belirlenememekle birlikte yapay sinir ağı, doğrusal olmayan sistemlerin denetiminde en etkin yöntem olarak görülmekte ve gerçek dünyadaki sistemler için yapay sinir ağının modelleme ve denetim yeteneklerinin incelenmesine ihtiyaç duyulmaktadır.

Kontrol sistemlerindeki etkinliği açısından yapay sinir ağının en önemli özelliği, doğrusal olmayan herhangi bir fonksiyonu yaklaştırabilmesidir. Gerçek zamanda öğrenme, uyarlama ve genelleme yeteneği ile yapısı gereği çok girişli, çok çıkışlı sistemlere de uygulanabilmesi nedeniyle kontrol alanında tercih sebebi olmuştur. Ayrıca paralel yapısı nedeniyle hatayı tolere etme ve hızlı bilgi işleme yeteneği de arzu edilen bir özelliktir.

3.9. Sistemlerin Matematiksel Tanımı

Yapay sinir ağı ile sistem tanılama ve denetim algoritmaları, sürekli zamanda ifade edilebilmekle birlikte denetim uygulamalarının çoğunluğu örneklenmiş sistemlerdir. Doğrusal olmayan sistemlerin sürekli zamanda durum denklemleri,

$$\dot{x}(t) = \Phi(x(t), u(t)) \quad (3.13)$$

$$\dot{y}(t) = \Psi(x(t), u(t)) \quad (3.14)$$

$u(t)$ -sistemin giriş vektörü, $x(t)$ -durum değişkenleri vektörü ve $y(t)$ -sistem çıkış vektörü olmak üzere sistemin ayrık zamanda durum denklemi aşağıdaki gibi yazılır.

$$x(k+1) = \Phi(x(k), u(k)) \quad (3.15)$$

$$y(k) = \Psi(x(k), u(k)) \quad (3.16)$$

Sistemlerin tanı modellemelerinde, sistem çıkışının sistemin bütün durum değişkenleri hakkında yeterince zengin bilgiyi ihtiva ettiği varsayılır (Brown vd., 1994, Aström vd., 1995). Buna göre bir girişli bir çıkışlı doğrusal olmayan bir sistemin ayrık zamanda giriş-çıkış modeli aşağıdaki gibi yazılır.

$$y(k)=f(y(k-1),\dots y(k-n),u(k),\dots u(k-m)) \quad (3.17)$$

Yapay sinir ağı ile sistem tanılama ve denetimde, sistemin derecesinin (n,m) bilinmesi gerekir. Sistemin derecesinin küçük seçilmesi, modellenemeyen dinamikler nedeniyle denetim sisteminin kararlılığını etkilerken büyük seçilmesi modelin yakınsaması ve genelleme yeteneğini kötüleştirir. Sistemin tanı modeli düşük dereceden seçildiğinde, uyarlamalı yapay sinir ağı-Bulanık mantık sistemlerinin kullanılması önerilmektedir, ancak tanı modelinin giriş vektöründe kararlılığı kesinleştiren bilgi olmadığı sürece denetim sisteminin kararlılığını kesin olarak belirlemek olanaksızdır (Brown vd., 1994).

3.10. Sistem Tanılama

Sistem tanılama problemi, uygun bir model yapısı seçerek, seçilen yapıdaki bilinmeyen parametrelerin hesaplanmasını gerektirir. Parametre hesaplamada temel kural, bilinen parametrelerin model yapısını seçerken bir ön bilgi olarak kullanılmasıdır. Bilinen ilk bilgilere göre 3 çeşit model yapısı seçilebilir (Sjoberg vd., 1995).

Açık modeller: Sistemin ön bilgileri ve fiziksel yapısından yararlanarak tam olarak bilinen modelleri ihtiva eder.

Yarıaçık modeller: Sistemin fiziksel özelliklerinden bazı ön bilgilerin elde edilebildiği ve diğer parametrelerin sistemden ölçülen verilerle belirlendiği modellerdir. Model yapısı, sistemin fiziksel yapısına ve derecesine uygun olarak seçilebilir ve model parametreleri sistemden ölçülen verilerle belirlenir.

Kapalı modeller: Sistem hakkında hiçbir fiziksel ön bilginin bilinmediği fakat seçilen modelin, etkinliği belirlenmiş bir yapıda seçilen modellerdir. Doğrusal sistemler için FIR, ARX, ARMAX gibi çeşitli model yapıları ve parametre hesaplama teorileri geliştirilmiştir. Doğrusal olmayan sistemler için de benzer model yapıları seçilebilir ancak tanılama problemi karmaşık matematiksel teorilere ihtiyaç duyar.

Sistem tanılama problemi, $k=1,2,...N$ için sistemden ölçülen $u(k)$ ve çıkış $y(k)$ 'lerden yararlanarak sistemin gelecekteki çıkışını veren doğrusal olmayan ilişkiyi belirlemektir.

$$y(k)=f(y(k-n),u(k-m))+e(k) \quad (3.18)$$

$e(k)$ -hata terimi olarak adlandırılır ve amaç hatayı sıfır yaparak sistemin gelecekteki çıkışını doğru olarak elde etmektir. Doğrusal olmayan $f(.)$ fonksiyonu, parametre vektörüne göre aşağıdaki gibi yazılabilir.

$$y(k)=f(y(k-n),u(k-m),w) \quad (3.19)$$

Denklem 3.19'daki genel model yapısı, sistemden ölçülen verilerin doğrusal olmayan dönüşümüne ve ağırlıklara göre;

$$y(k)=f(y(k-n),u(k-m),w)=f(\phi(k),w) \quad (3.20)$$

$$\phi(k)=\phi(y(k-n),u(k-m))$$

Burada, $\phi(k)$ model giriş vektörü, elemanları ise model girişleri olarak söylenir. Bu durumda sistem tanımlama problemi, model giriş vektörünün $(\phi(k))$ seçimi ve doğrusal olmayan ilişkinin $f(\phi)$ belirlenmesi gibi alt problemlere ayrıştırılır.

3.10.1. Giriş Vektörlerine Göre Doğrusal Olmayan Dinamik Modeller

Doğrusal model yapılarına benzer şekilde doğrusal olmayan modeller giriş vektörlerine göre aşağıdaki isimlerle söylenir. $x(k)$ -giriş vektörlerini göstermek üzere,

- NFIR Model girişleri olarak sistem girişlerini kullanan modellerdir.

$$x(k)=(u(k),...u(k-m)) \quad (3.21)$$

- NARX Model girişleri olarak sistem giriş ve çıkış verilerini kullanan modellerdir. yapay sinir ağı modellerde ileri beslemeli yapay sinir ağı yapısı gerektirir ve seri-paralel model yapısı olarak ta söylenir (Narendra vd.,1990).

$$x(k) = (y(k-1), \dots, y(k-n), u(k), \dots, u(k-m)) \quad (3.22)$$

• NARMAX Model girişleri olarak sistem giriş ve çıkış verileri ile birlikte model hatasını da kullanan modellerdir. İleri beslemeli yapay sinir ağı ile modellenebilir.

$$x(k) = (y(k-1), \dots, y(k-n), u(k), \dots, u(k-m), e(k), \dots, e(k-n)) \quad (3.23)$$

• NOE Model girişi olarak, sistem girişi ile model çıkış verilerini kullanan modellerdir. Model çıkışı y_m ile gösterilirse model giriş vektörü

$$x(k) = (y_m(k-1), \dots, y_m(k-n), u(k), \dots, u(k-m)) \quad (3.24)$$

NOE modeller, geri beslemeli yapay sinir ağı yapısı gerektirir ve model yapısı olarak ta söylenir (Narendra vd., 1990).

• NBJ Model girişleri olarak, sistem girişi model çıkışı ve modelleme hatalarını kullanan modellerdir ve geri beslemeli yapay sinir ağı yapısı gerektirir.

$$x(k) = (y_m(k-1), \dots, y_m(k-n), u(k), \dots, u(k-m), e(k), \dots, e(k-n)) \quad (3.25)$$

Bu yapıların dışında durum değişkenleri ve sistem girişini model giriş vektörü olarak kullanan doğrusal olmayan durum uzay modelleri sayılabilir.

3.10.2. Doğrusal olmayan İlişkiye Göre Model Yapıları

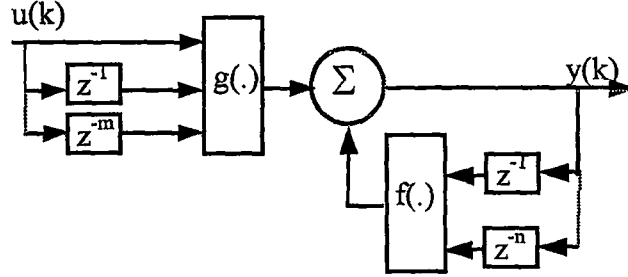
Model giriş vektörlerine göre belirlenen bu model yapılarının dışında sistemin doğrusal olmayan ayırık zaman giriş-çıkış denkleminde göre 4 çeşit model yapısı incelenmiştir (Narendra vd., 1990, Narendra vd., 1995).

$$1. y(k) = \sum_{i=1}^n a_i y(k-i) + g(u(k), u(k-1), \dots, u(k-m)) \quad (3.26)$$

$$2. y(k) = f(y(k-1), \dots, y(k-n)) + \sum_{i=0}^m b_i u(k-i) \quad (3.27)$$

$$3. \quad y(k) = f(y(k-1), \dots, y(k-n)) + g(u(k), \dots, u(k-m)) \quad (3.28)$$

$$4. \quad y(k) = f(y(k-1), \dots, y(k-n), u(k), \dots, u(k-m)) \quad (3.29)$$



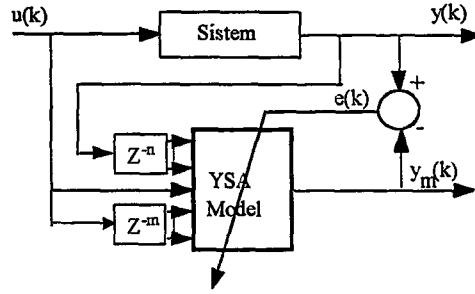
Şekil 3.5 Doğrusal olmayan model 3 yapısı

3.11. Yapay Sinir Ağları ile Sistem Tanılama

Yapay sinir ağının sistemin giriş-çıkış verileri kullanarak eğitilmesi, doğrusal olmayan bir fonksiyonu yaklaştırma problemi olarak incelenir. Doğrusal olmayan tek gizli katmanlı bir yapay sinir ağının herhangi bir doğrusal olmayan sürekli fonksiyonu arzu edilen doğrulukta yaklaştırabileceği belirlenmiştir. Yapay sinir ağının yapısına göre, herhangi bir fonksiyon yaklaştırma problemi için orta katmandaki hücre sayısı yada RTF ağlarında taban fonksiyonu sayısı ve merkezlerin yeri gibi yeterince ispat edilememiş bilgilere rağmen yapay sinir ağı, sistem tanılama ve denetiminde yaygın olarak kullanılmaya başlanmıştır.

3.11.1. Düz Modelleme

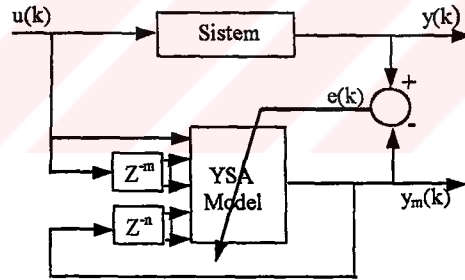
Bir sistemin düz yön dinamiklerini gösterecek şekilde yapay sinir ağı modelin seçilmesi ve eğitilmesi düz modelleme olarak söylenir (Brown vd., 1994, Hunt vd., 1992). Şekil 3.6'da NARX yada seri-paralel model yapısına göre bir sistemin yapay sinir ağı tanı modeli verilmiştir.



Şekil 3.6 Yapay sinir ağı ile düz modelleme (NARX model)

Yapay sinir ağı ile bir sistemin NARX modeli, ileri beslemeli katmanlı yapay sinir ağı yapısı ile elde edilir ve yapay sinir ağı, sistem çıkışı ve model çıkışı arasındaki hataya göre geriye yayılım algoritması ile eğitilir (Narendra vd., 1990, Tanumaru vd., 1992, Nguyen ve Widrow, 1990).

Sistemlerin denetimi açısından önemli bir konu, denetlenen sistemin dinamik davranışlarının model ve denetleyici yapay sinir ağına tam olarak temsil edilmesidir. Bu ancak Dinamiklerin yapay sinir ağı yapısında tanımlanması ile sağlanabilir. Bu durumda geri beslemeli yapay sinir ağı yapılarına ihtiyaç duyulur ve NOE modeli olarak bilinen düz yapay sinir ağı model yapısı Şekil 3.7’de görülmektedir.



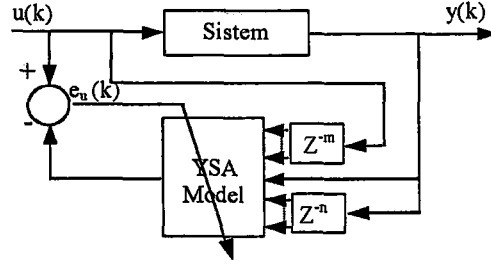
Şekil 3.7 Yapay sinir ağı ile düz modelleme (NOE model)

Sistem tanılamada NARX model, kararlılık açısından ve statik geriye yayılım algoritması ile yapay sinir ağı eğitilebileceğinden tercih edilir. Ancak, gürültülü sistemlerde, gerçek sistem çıkışı üzerindeki gürültünün neden olduğu polarma problemini önlemesi nedeniyle NOE model kullanılır.

3.11.2. Ters Modelleme

Yapay sinir ağılı denetim sistemlerinde, genellikle sistemin ters modeli denetleyici olarak kullanılır ve bir sistemin ters yön dinamiklerini $u(k)=f(y(k),u(k-m))$

verecek şekilde yapay sinir ağı modelin seçilmesi ve eğitilmesi ters modelleme olarak söylenir.



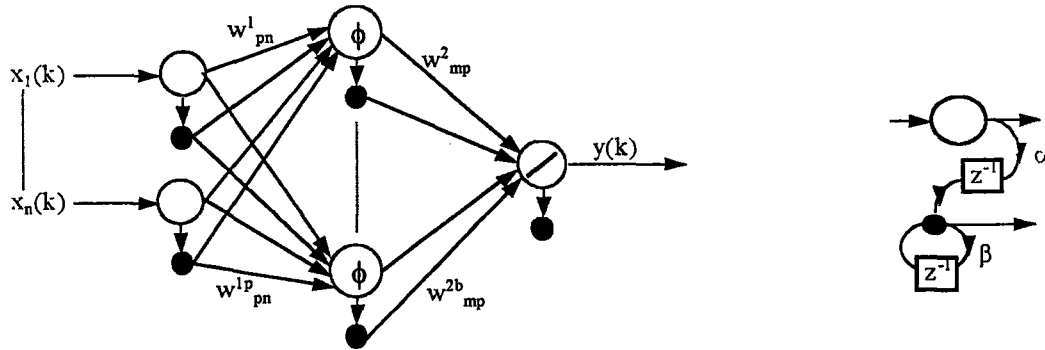
Şekil 3.8 Yapay sinir ağı ile ters modelleme

Doğrudan ters modelleme olarak ta bilinen şekil 3.8'deki modelde sistemin o anki ve önceki çıkışları ile sistem girişinin önceki örnekleri yapay sinir ağı ters modelin girişleri (regresörleri) olarak alınır ve yapay sinir ağı model sistem girişini verecek şekilde eğitilir. Ancak doğrudan ters modelleme yönteminde sistemin doğru bir ters modeli elde edilemeyebilir.

3.11.3 Yapay Sinir Ağı Yapıları ile Sistem Kontrolünde Yeni Yaklaşımlar

Yapay sinir ağı ile sistem tanılamada, doğru model yapısının seçilebilmesi ve regresörlerinin belirlenebilmesi için sistemin girişinin ve çıkışının en büyük gecikme derecelerinin bilinmesi gerekir. Sistemin derecesinin doğru belirlenememesi, modelde temsil edilemeyen dinamikler nedeniyle kararlı ve değişen dinamik şartlarda doğru bir model elde edilmesini etkiler. Bu nedenle geri beslemeli yapay sinir ağı yapıları kullanılarak sistemin derecesine ihtiyaç duymayan tam modelleri geliştirilmiştir.

Şekil 3.9'da Bellek hücreli yapay sinir ağları (BHYSA) olarak söylenen ve ağdaki her bir hücre için bir bellek hücresinin kullanıldığı katmanlı ağ yapısı verilmiştir.

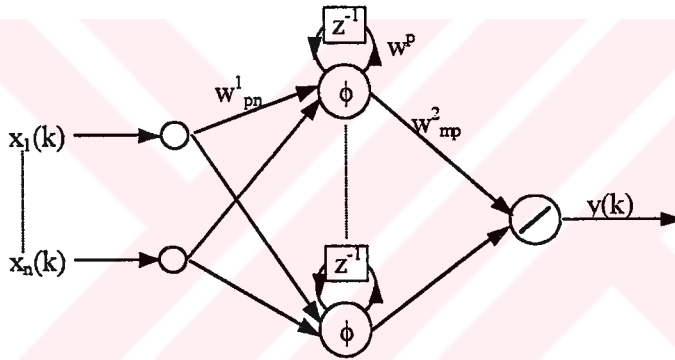


Şekil 3.9 Bellek hücreli yapay sinir ağı

Bellek hücreli yapay sinir ağında her temel hücreye ait bellek hücresi bir ağırlıklı (α) üzerinden öz geri beslemeye ve ait olduğu hücrenin yine bir ağırlık üzerinden (β) gecikmiş girişe sahiptir. Bellek hücreli yapay sinir ağının doğrusal olmayan bir sistemi modelleme yeteneği, sadece sistemin o an ki giriş ve bir önceki çıkış verileri regresör alınarak incelenmiş ve tatmin edici sonuçlar alınmıştır (Sastry vd., 1994).

İleri beslemeli katmanlı yapay sinir ağının sadece bir gizli katmanında bellek hücreleri kullanarak ilgili temel hücre çıkışının geçmiş örneklerini tutan ve zaman gecikmeli yapay sinir ağı olarak söylenen ağlarda sistem tanılamada incelenmiştir (Fukado vd., 1992).

Sistem tanılamada diğer bir yapay sinir ağı yapısı, gizli katman hücrelerinde öz geribeslemenin kullanıldığı ve yöresel geribeslemeli-küresel ileri beslemeli (YGKİ) yapay sinir ağıdır (Şekil 3.10).



Şekil 3.10 YGKİ Yapay sinir ağı

3.12 Yapay Sinir Ağlarının kullanıldığı Kontrol Yapıları

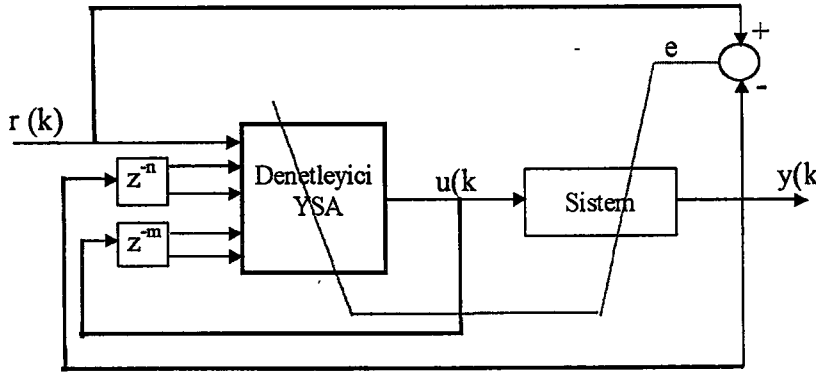
Dinamik sistemlerin belirlenen değişkenlerinin arzu edilen değerlerde tutulmasını sağlamak için kullanılacak kontrol sisteminin analiz ve tasarımı, kontrol sistemlerinin teorisini teşkil eder. Arzu edilen çıkışı elde etmek için, sistem gerekli girişi uygulayacak kontrolör tasarımı, kontrol problemini oluşturur. Doğrusal sistemler için zaman domeninde ve frekans domeninde çeşitli kontrol sistemi analiz ve tasarım yöntemleri geliştirilmiştir. Parametreleri bilinmeyen doğrusal sistemlerin kontrolü için çeşitli uyarlamalı kontrol yapıları incelenmiş ve bütün bir sistemin kararlılığını sağlayacak şekilde kontrol parametrelerini ayarlama kuralları belirlenmiştir (Narendra vd., 1995, Aström vd., 1995). Sonuçta, sistemin modeli ile bazı ön bilgiler olması

durumunda kararlı uyarlama kurallarının belirlenebileceği gösterilmiş ve çalışmalar, çeşitli bozucu şartlar altında kontrol sisteminin öz yeteneğini koruma ve geliştirmesini sağlayacak şartların belirlenmesine doğru yönelmiştir.

Doğrusal olmayan sistemlerin kontrolü için bazı yöntemler geliştirilmekle birlikte, sistemlerin uyarlamalı kontrolünde genel bir teori yeterince geliştirilememiştir. Buna mukabil yapay sinir ağının doğrusal olmayan sistemlerin uyarlamalı kontrolündeki başarısı, çalışmaların yapay sinir ağı ile kontrol alanında yoğunlaşmasına neden olmuştur. Yapay sinir ağı ile sistem kontrolünde, doğrusal sistemlerin kontrolünde kullanılan uyarlamalı yapılar esas alınmış ve doğrusal olmayan sistemler için yapay sinir ağının kontrol yeteneği incelenmiştir (Narendra vd., 1990, Boskovic vd., 1995, Rovithakis vd., 1994). Burada literatürde karşılaşılan çeşitli yapay sinir ağı ile kontrol yapılarının genel özellikleri verilmiştir.

3.12.1 Doğrudan Uyarlamalı Kontrol

Doğrudan kontrolde temel amaç, sistemin doğru bir ters modelinin belirlenmesidir. Sistemin ters yapay sinir ağı modeli kontrolör olarak kullanılır. Yapay sinir ağı modeli, sistemin tersini tam olarak temsil edebilirse sistemin çıkışı, referans girişe eşit olacaktır. Şekil 3.11 de, doğrudan uyarlamalı denetim yapısı görülmektedir.



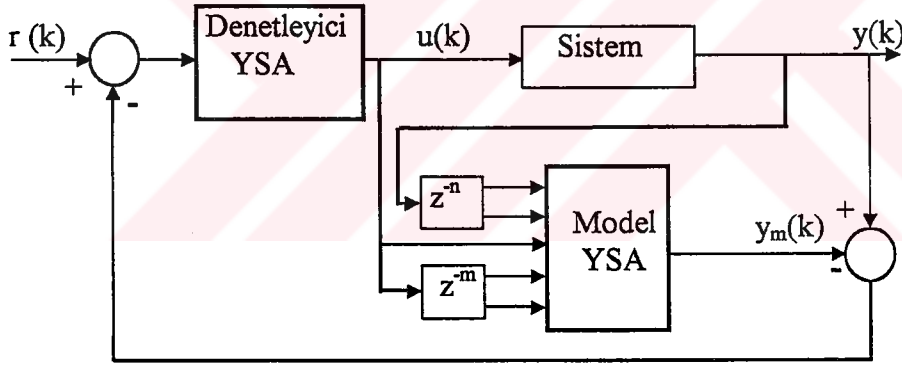
Şekil 3.11 Doğrudan uyarlamalı kontrol

Kontrolör yapay sinir ağı, referans giriş ile sistem çıkışı arasındaki hataya göre eğitildiğinden sistem çıkışının, sistem girişine göre doğru eğiminin hesaplanması gerekir. Sistemin doğrusal olmayan ilişkisi bilinemediğinde bu eğimi yaklaşık olarak hesaplamak ancak sayısal çözümleme yöntemleri ile mümkündür. Kontrol edilen

sistemin zaman sabitesi çok küçük yada çok büyük olduğunda ve sisteme bozucu girişler olması durumunda bu eğimi doğru olarak hesaplamak oldukça zordur. Bu şartlarda bulunan eğim, denetleyici yapay sinir ağının eğitiminde belirsizliklere neden olabilir. Yapay sinir ağının eğitimi için eğimin gerçek değeri yerine işareti de yeterli olduğundan eğimin işareti belirlenerek de yapay sinir ağı eğitilebilmektedir (Zhang vd., 1995).

3.12.2 Dahili Model (DM) Kontrol

DM kontrol, doğrusal olmayan sistemlerin kontrolünde etkin bir yaklaşım olarak görülmekte ve sistemin düz ve ters modelleri geri besleme elemanı olarak kullanılmaktadır. DM denetimde sistem çıkışı ile düz yapay sinir ağı modeli çıkışı arasındaki hata, geri beslemeyi oluşturur ve kontrolör yapay sinir ağı, sistemin ters modelini verecek şekilde eğitilir, Şekil 3.12.



Şekil 3.12 Dahili model kontrol

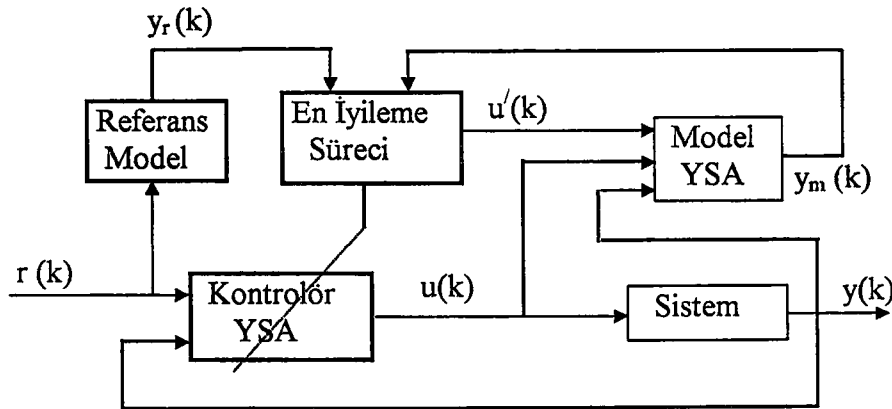
Kontrol sisteminin kararlılığı açısından DM kontrol yapısının aşağıdaki özellikleri belirlenmiştir (Hunt ve Sbrabaro, 1991).

- Kontrolör ve sistem giriş-çıkış kararlı ve düz model, sistemi tam olarak temsil ediyorsa kapalı çevrim sistem giriş-çıkış kararlıdır.
- Sistem modelinin tersi varsa ve bu ters model kontrolör olarak kullanılırsa kapalı çevrim sistem giriş-çıkış kararlıdır ve sistem çıkışı, referansı tam olarak izleyebilir.
- Sistemin sürekli rejimde doğru bir ters modeli kontrolör olarak kullanılırsa sistem giriş-çıkış kararlıdır ve sabit girişler için de hatasız bir kontrol elde edilebilir.

DM kontrolün bu özelliklerinden, sistemin doğru modelleri elde edilebilirse ölçülemeyen bozucu girişlere rağmen kapalı çevrim sistemin doğru bir izleme kontrolü sağlayacağı söylenebilir. Ancak pratikte, tam doğru modeller elde edilemez ve tam doğru bir denetim için gereken çok yüksek kazançlar nedeniyle modellenemeyen dinamikler olması durumunda duyarlılık problemleri ortaya çıkar. Bu nedenle, geri beslemeli sistemin kazancını azaltacak bir filtre kullanılır ve tam doğru bir kontrolörden uzaklaşarak sistemin bozucu girişlere karşı davranışı düzeltilir. Filtrenin diğer bir görevi, hata sinyalini denetleyici için uygun bir giriş uzayına dönüştürmektir (Hunt ve Sbrabaro, 1991, Hofer vd., 1993).

3.12.3 Tahmini Kontrol

Tahmini kontrol algoritmasının esası, sistem çıkışının ve kontrol girişinin n adım gelecekteki değerini tahmin ederek seçilen bir amaç ölçütünün en azlanmasıyla en iyi kontrol girişinin belirlenmesidir. En iyileme sürecinde bir dizi kontrol girişi belirlenir. Ancak, dizideki ilk kontrol sinyali, o anda sisteme uygulanır. Yapay sinir ağı ile tahmini kontrolde, sistemin düz ve ters modelleri kullanılır ve düz model çıkışı, referans girişle birlikte bir en iyileme algoritmasından geçirilerek seçilen bir amaç ölçütünün en azlanması ile en uygun kontrol girişi belirlenir. Şekil 3.13’de tahmini kontrol yapısı görülmektedir.



Şekil 3.13 Tahmini kontrol

Farklı amaç ölçütlerini en azlayan farklı tahmini kontrol algoritmaları geliştirilmekle birlikte genellikle aşağıdaki amaç ölçütü seçilir (Aström vd., 1995, Hofer vd., 1991, Khalid vd., 1995).

$$J = \sum_{j=N_1}^{N_2} Q [y_r(k+j) - y_m(k+j)]^2 + \sum_{j=1}^{N_u} R [u'(k+j-1) - u'(k+j-2)]^2 \quad (3.30)$$

Burada N_1 ve N_2 en iyileme algoritmasının gerçekleştirildiği sistem çıkışının gelecekteki zaman dilimini, N_u ise kontrol girişinin gelecekteki zaman dilimini gösterir. Q ve R , ağırlık katsayılarıdır. En iyileme sürecinde kontrolör yapay sinir ağı, en iyileme algoritmasının dinamiklerini yakınsayacak şekilde eğitilir ve sonuçta kontrolör yapay sinir ağı, en iyileme sürecinde elde edilen aynı kontrol girişini üretir. Tahmini denetimin önemli bir üstünlüğü kontrolör yapay sinir ağının eğitiminden sonra sistemin modeli ve en iyileme sürecine ihtiyaç duymamasıdır.

3.12.4 Öz Uyarlamalı Kontrol

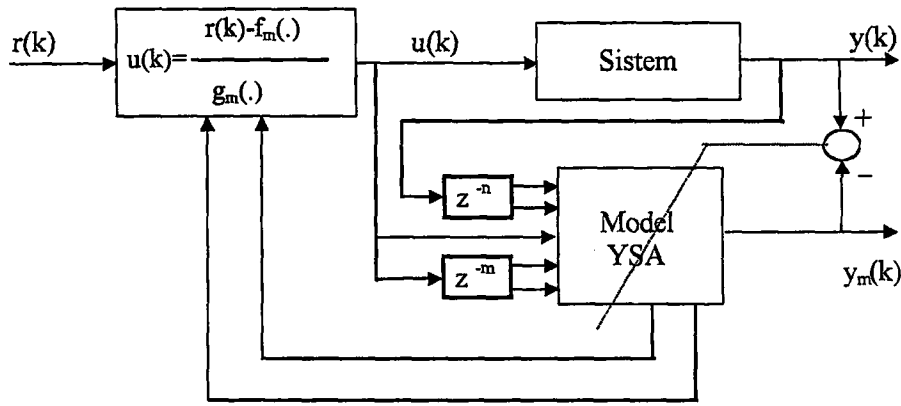
Kontrol edilebilir kanonik formda durum - uzay denklemleri ya da Denklem 3.31 formunda giriş - çıkış denklemleri ile modellenen sistemler geri besleme ile doğrusallaştırılabilir sistemler olarak bilinir (Chen, 1990, Chen ve Khalil, 1992). Doğrusal olmayan ve geri besleme ile doğrusallaştırılabilen tek giriş-tek çıkışlı sistem, d - sistemin bağıl derecesi, $i=0,1, n-1$ ve $j=0,1, m-1$ sistemin giriş ve çıkışındaki gecikme derecelerini göstermek üzere Denklem 3.39 de verilmiştir.

$$y(k+1) = f(y(k-i) \dots u(k-i+j)) + g(y(k-i) \dots u(k-i+j)) \cdot u(k-i+j+1) \quad (3.31)$$

Sistemin doğrusal olmayan $f(\cdot)$ ve $g(\cdot)$ fonksiyonları belirlenebilirse Denklem 3.32 de verilen kontrol girişi ile sistem doğrusallaştırılarak öz-uyarlamalı kontrolü gerçekleştirilebilir. Bağıl derecesi $d=1$ olan bir sistem için kontrol girişi,

$$u(k) = \frac{r(k) - f_m(\cdot)}{g_m(\cdot)} \quad (3.32)$$

Burada, r - referans giriş ve $f_m(\cdot)$, $g_m(\cdot)$ ise sistemin $f(\cdot)$ ve $g(\cdot)$ fonksiyonlarının tanılanan yaklaşığıdır.



Şekil 3.14 Öz uyarlamalı Kontrol

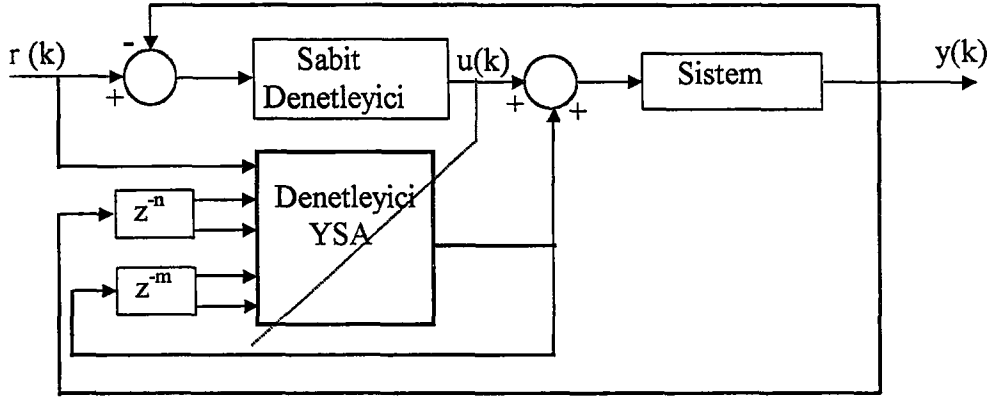
Denklem 3.31’de, sistemin bağıl derecesi $d > 1$ ise nedensel bir kontrol algoritması elde edilemez. Bu durumda, sistemin gelecekteki çıkışları önceki çıkışları ile ifade edilerek bulunan yeni doğrusal olmayan fonksiyonlarla Denklem 3.33 deki gibi nedensel bir kontrol gerçekleştirilebilir (Chen 1990, Chen ve Khalil 1992).

$$u(k) = \frac{r(k) - F_m(.)}{G_m(.)} \quad (3.33)$$

Geri besleme ile doğrusallaştırılabilir sistemlerin geleneksel yöntemlerle öz-uyarlamalı kontrolünde, doğrusal olmayan $f(.)$ ve $g(.)$ fonksiyonlarının $f(.) = \theta_i \cdot f_i(.)$ formunda parametrelerine göre doğrusal fonksiyonlarla modellenebileceği varsayılmıştır. $f(.)$ ve $g(.)$ fonksiyonları, yapay sinir ağı ile modellenerek bilinmeyen ve parametrelerine göre doğrusal olmayan dinamik sistemlerin de öz - uyarlamalı kontrolleri gerçekleştirilebilir.

3.12.5 Sabit Kontrolörlü Kontrol

Şekil 3.15 de verildiği gibi sistemin ters modelini verecek şekilde eğitilen kontrolör yapay sinir ağı ile birlikte doğrusal ve kararlılığı sağlayıcı sabit bir kontrolör, en basit doğrudan kontrol yapısını oluşturur.



Şekil 3.15 Sabit kontrolörlü kontrol.

Sabit kontrolör, kapalı çevrim sistemin kararlılığını sağlayacak şekilde tasarlanır ve çıkışı, kontrolör yapay sinir ağı için eğitim sinyalini oluşturur. Kontrol sisteminin başarısı, o anki çalışma noktasına bağlıdır ancak, gerçek zamanda sistemin ters dinamiklerini öğrenen kontrolör yapay sinir ağının eğitimi sürdükçe sistemin davranışı iyileşir. Yapay sinir ağı, sistemin ters dinamiklerini tam olarak öğrendikten sonra sabit kontrolörün bir etkisi kalmaz ve doğrudan uyarlamalı kontrol yapısına dönüşür. Sabit kontrolördeki integral etkisi ile sistemin sürekli rejim hatasının önlendiği belirlenmiştir.

Sabit kontrolörlü kontrol algoritması basit olmakla birlikte sabit kontrolörün tasarımı, sistemin başarısı ile yakından ilgilidir. Sabit kontrolör iyi tasarlanmadığında yapay sinir ağının eğitim sinyali etkileneneğinden yapay sinir ağının öğrenme oranı ve yakınsaması kötüleşecektir. Algoritmanın sabit kontrolörün tasarımına bağlı olarak bozucu girişlerde davranışının iyi olduğu belirtilmiştir (Hofer vd., 1993).

4. ADAPTİF AKTİVASYON FONKSİYONLU YSA'LARI İLE DA MOTORLARIN HIZ KONTROLÜ

Geleneksel kontrol yöntemlerinde, kullanılacak kontrolörün seçimi, parametrelerinin tespiti ve sisteme uygulanma şekli tasarımcı için önemli bir sorundur. Böyle bir kontrolör parametrelerinin belirlenmesi sistem parametrelerinin bilinmesine bağlıdır. Bununla birlikte kontrolör, sistemdeki modelleme hataları, sistemin dinamik parametrelerindeki değişimler ve dışarıdan sisteme etkiyen bozucu girişler gibi değişken durumlarda sistem kararlılığını koruyamaz. Geleneksel yöntemlerdeki bu tip sorunlara rağmen, yapay sinir ağları, öğrenme ve genelleme yeteneğinin oldukça iyi olması ve sistemin matematiksel denkleminin ihtiyaç duymaması nedeniyle kontrol alanında da yoğun olarak kullanılmaya başlanmıştır.

Yapay sinir ağlarının kontrol uygulamalarında, çoğunlukla katmanlı ağlar kullanılmaktadır. Bu ağların eğitiminde ise, belirlenen bir performans kriterinin parametrelere göre kısmi türevlerinin alınarak geriye yayındığı, geriye yayının algoritması yoğun ilgi görmüştür. Kontrol yapıları içinde en çok kullanılan yapılar ise doğrudan ve dolaylı kontrol yapılarıdır.

Yapay sinir ağlarının sistem tanılama ve kontrolde çeşitli sistemler uygulanmış ve oldukça başarılı sonuçlar elde edilmiştir. Bu nedenle, yapay sinir ağları üzerine yapılan çalışmalar yoğunluk kazanmıştır. Yapılan çalışmalarda, eğitim algoritmalarının yanında yapısal özelliklerinin yapay sinir ağının performansını etkileyen faktörler olduğu tespit edilmiştir. Böylece, çalışmalar yapay sinir ağlarının performansını artırmaya ya da sorunlarını gidermeye yönelmiştir. Sinir hücrelerinde kullanılan aktivasyon fonksiyonlarının problemin niteliğine göre çeşitlilik göstermesi, bu konuya yönelik çalışmalara neden olmuştur. Aktivasyon fonksiyonlarının, eğitim hızına ve genellemeye etkisini araştıran çalışmalar literatürde yer almıştır.

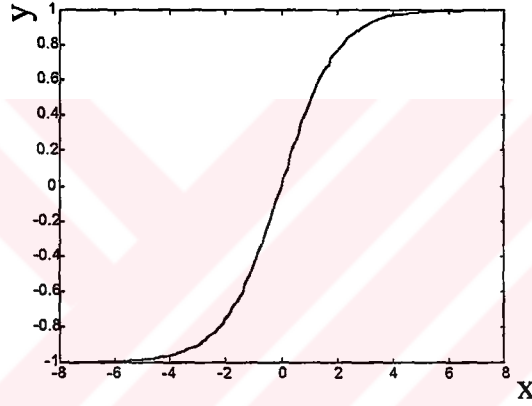
Bu bölümde, ikinci bölümde parametreleri verilen doğru akım motorunun, yine aynı bölümde verilen matematiksel modeli kullanılarak, uygun yapay sinir ağı kontrol yapısı tespit edilmiş ve bu yapıda kullanılan dört farklı aktivasyon fonksiyonu ve bunların adapte edilme yöntemleri incelenerek, DA motorunun yapay sinir ağları ile hız kontrolünün nasıl yapıldığı açıklanmıştır.

4.1. Sigmoid Aktivasyon Fonksiyonu

En çok kullanılan aktivasyon fonksiyonu sigmoid aktivasyon fonksiyonudur. Denklem 4.1 den görüldüğü gibi bu aktivasyon fonksiyonunun şeklini belirleyen iki parametre vardır. Bunlar, eğrinin eğimini belirleyen b ve genliğini belirleyen a parametreleridir.

$$y = a \frac{1 - e^{-bx}}{1 + e^{-bx}} \quad (4.1)$$

Bu aktivasyon fonksiyonu şekil 4.1 de görülmektedir.



Şekil 4.1 Çift yönlü sigmoid fonksiyon

Yapay sinir ağının eğitimi yapılırken aktivasyon fonksiyonunun parametreleri çoğunlukla sabit olarak alınır. Ancak, problemin niteliğine göre farklı başarımlar gösteren aktivasyon fonksiyonlarının, yapay sinir ağının eğitimi sırasında aktivasyon fonksiyonunun parametrelerinin eğitilmesi öğrenmeyi hızlandıracağı sonucuna götürür. Sigmoid aktivasyon fonksiyonu için, eğrinin eğimi ve genliğini belirleyen, eğitilebilecek iki parametre vardır. Bu parametrelerin eğitimi, yapay sinir ağının ağırlıklarının eğitiminde kullanılacak olan geriye yayılım algoritması ile gerçekleştirilir. Buna göre parametreye uygulanacak olan düzeltme miktarı toplam karesel hatanın ilgili parametreye göre türevinin alınmasıyla gerçekleştirilir. 1. katmandaki j . hücrenin genliğini belirleyen parametreye uygulanacak düzeltme,

$$a_j(k+1) = a_j(k) + \Delta a_j(k) \quad (4.2)$$

$$\Delta a_j(k) = -\alpha \frac{\partial E(k)}{\partial a_j(k)} \quad (4.3)$$

$$b_j(k+1) = b_j(k) + \Delta b_j(k) \quad (4.4)$$

$$\Delta b_j(k) = -\alpha \frac{\partial E(k)}{\partial b_j(k)} \quad (4.5)$$

S ağırlıkları uyarlama iterasyonu olmak üzere, N adet örneğin uygulanmasından sonra parametrelerin toplu olarak uyarlanmasında ise,

$$\Delta a, b(S) = -\alpha \frac{\partial E}{\partial a, b(S)} \quad (4.6)$$

$$a, b(S) = a, b(s-1) + \Delta a, b(S) \quad (4.7)$$

bağıntıları kullanılır.

O halde, geriye yayılım algoritmasında, denklem 4.3 ve 4.5 deki ifadelerindeki

$\frac{\partial E(k)}{\partial a_j(k)}$ ve $\frac{\partial E(k)}{\partial b_j(k)}$ denklemlerinin kısmi türevleri ilk parametre için alınır;

$$\frac{\partial E(k)}{\partial a_j(k)} = \frac{\partial E(k)}{\partial e(k)} \frac{\partial e(k)}{\partial y(k)} \frac{\partial y(k)}{\partial v_j(k)} \frac{\partial v_j(k)}{\partial a_j(k)} \quad (4.8)$$

elde edilir ve eğitim işaretinin kısmi türevleri sırasıyla;

$$\frac{\partial E(k)}{\partial a_j(k)} = e(k)(-1)w_j^2 \phi(a(k)) \quad (4.9)$$

diğer parametre için eğitim işaretinin kısmi türevleri alınır;

$$\frac{\partial E(k)}{\partial b_j(k)} = \frac{\partial E(k)}{\partial e(k)} \frac{\partial e(k)}{\partial y(k)} \frac{\partial y(k)}{\partial v_j(k)} \frac{\partial v_j(k)}{\partial b_j(k)} \quad (4.10)$$

ifadesinin sonucu sırasıyla,

$$\frac{\partial E(k)}{\partial b_j(k)} = e(k)(-1)w_j^2 \phi(b(k)) \quad (4.11)$$

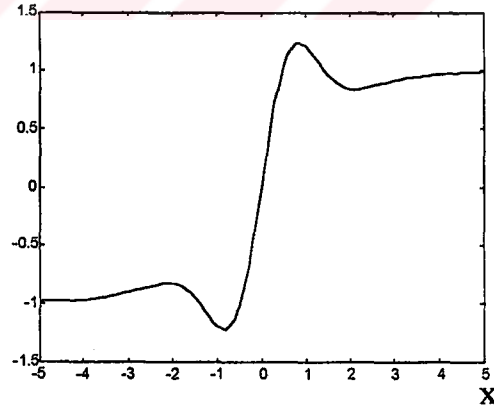
olarak hesaplanabilir ve bu bağıntılarla çift yönlü sigmoid aktivasyon fonksiyonunun parametreleri yenilenmiş olur. Tez içinde, parametreler sabit alındığında standart sigmoid; parametreler uyarlandığında adaptif sigmoid olarak adlandırılmıştır.

4.2. Üstel ve Sigmoid Aktivasyon Fonksiyonu

Bu tez çalışmasında kullanılan uyarlamalı aktivasyon fonksiyonlarından bir diğeri ise denklem (4.12) ile verilen fonksiyondur. Sigmoid fonksiyonuna karesel bileşen eklenmesiyle elde edilen bu aktivasyonunu Zhang, (2000) incelemiş ve fonksiyon yaklaştırmadaki başarısını tartışmıştır. Fonksiyonun şekli sigmoide benzemesine rağmen uyarlanabilecek parametre sayısı standart sigmoid fonksiyonuna göre daha fazladır.

$$y = f(x) = a_1 \cdot e^{-b_1 x^2} + a_2 \cdot \frac{1 - e^{-b_2 x}}{1 + e^{-b_2 x}} \quad (4.12)$$

Bu aktivasyon fonksiyonu Şekil 4.2 de verilmiştir.



Şekil 4.2 Üstel ve sigmoid fonksiyon

Denklem 4.12 ile verilen aktivasyon fonksiyonun parametrelerinin uyarlanmasında aşağıdaki ifadeler kullanılır.

$$a1_j(k+1) = a1_j(k) + \Delta a1_j(k) \quad (4.13)$$

burada $\Delta a_{1j}(k)$ parametreye uygulanacak düzeltme miktarını belirtir ve performans kriterinin ilgili parametreye göre kısmi türeviyle hesaplanır.

$$\Delta a_{1j}(k) = -\alpha \frac{\partial E(k)}{\partial a_{1j}(k)} \quad (4.14)$$

burada α öğrenme oranını temsil eder. a_2 parametresine uygulanacak düzeltme,

$$a_{2j}(k+1) = a_{2j}(k) + \Delta a_{2j}(k) \quad (4.15)$$

burada $\Delta a_{2j}(k)$ parametreye uygulanacak düzeltme miktarını belirtir ve performans kriterinin ilgili parametreye göre kısmi türeviyle hesaplanır.

$$\Delta a_{2j}(k) = -\alpha \frac{\partial E(k)}{\partial a_{2j}(k)} \quad (4.16)$$

burada α öğrenme oranını temsil eder. b_1 parametresine uygulanacak düzeltme,

$$b_{1j}(k+1) = b_{1j}(k) + \Delta b_{1j}(k) \quad (4.17)$$

burada $\Delta b_{1j}(k)$ parametreye uygulanacak düzeltme miktarını belirtir ve performans kriterinin ilgili parametreye göre kısmi türeviyle hesaplanır.

$$\Delta b_{1j}(k) = -\alpha \frac{\partial E(k)}{\partial b_{1j}(k)} \quad (4.18)$$

burada α öğrenme oranını temsil eder. b_2 parametresine uygulanacak düzeltme,

$$b_{2j}(k+1) = b_{2j}(k) + \Delta b_{2j}(k) \quad (4.19)$$

burada $\Delta b_{2j}(k)$ parametreye uygulanacak düzeltme miktarını belirtir ve performans kriterinin ilgili parametreye göre kısmi türeviyle hesaplanır.

$$\Delta b_{2j}(k) = -\alpha \frac{\partial E(k)}{\partial b_{2j}(k)} \quad (4.20)$$

burada α öğrenme oranını temsil eder.

S ağırlıkları uyarlama iterasyonu olmak üzere, N adet örneğin uygulanmasından sonra parametrelerin toplu olarak uyarlanması ise,

$$\Delta_{aj, bj}(S) = -\alpha \frac{\partial E}{\partial a_{j, bj}(S)} \quad (4.21)$$

$$a_{j, bj}(S) = a_{j, bj}(s-1) + \Delta_{aj, bj}(S) \quad (4.22)$$

bağıntıları kullanılır.

O halde, geriye yayılım algoritmasında, denklem 4.14 ve 4.16 ifadelerindeki

$\frac{\partial E(k)}{\partial a_{1j}(k)}$ ve $\frac{\partial E(k)}{\partial a_{2j}(k)}$ kısmi türevleri alınır;

$$\frac{\partial E(k)}{\partial a_{1j}(k)} = \frac{\partial E(k)}{\partial e(k)} \frac{\partial e(k)}{\partial y(k)} \frac{\partial y(k)}{\partial v_j(k)} \frac{\partial v_j(k)}{\partial a_{1j}(k)} \quad (4.23)$$

$$\frac{\partial E(k)}{\partial a_{2j}(k)} = \frac{\partial E(k)}{\partial e(k)} \frac{\partial e(k)}{\partial y(k)} \frac{\partial y(k)}{\partial v_j(k)} \frac{\partial v_j(k)}{\partial a_{2j}(k)} \quad (4.24)$$

elde edilir ve eğitim işaretinin kısmi türevleri sırasıyla;

$$\frac{\partial E(k)}{\partial a_{1j}(k)} = e(k)(-1)w_j^2 \phi(a_{1j}(k)) \quad (4.25)$$

$$\frac{\partial E(k)}{\partial a_{2j}(k)} = e(k)(-1)w_j^2 \phi(a_{2j}(k)) \quad (4.26)$$

denklem 4.18 ve 4.20'de $\frac{\partial E(k)}{\partial b_{1j}(k)}$ ve $\frac{\partial E(k)}{\partial b_{2j}(k)}$ kısmi türevleri alınır;

$$\frac{\partial E(k)}{\partial b_{1j}(k)} = \frac{\partial E(k)}{\partial e(k)} \frac{\partial e(k)}{\partial y(k)} \frac{\partial y(k)}{\partial v_j(k)} \frac{\partial v_j(k)}{\partial b_{1j}(k)} \quad (4.27)$$

$$\frac{\partial E(k)}{\partial b_{2j}(k)} = \frac{\partial E(k)}{\partial e(k)} \frac{\partial e(k)}{\partial y(k)} \frac{\partial y(k)}{\partial v_j(k)} \frac{\partial v_j(k)}{\partial b_{2j}(k)} \quad (4.28)$$

elde edilir ve eğitim işaretinin kısmi türevleri sırasıyla;

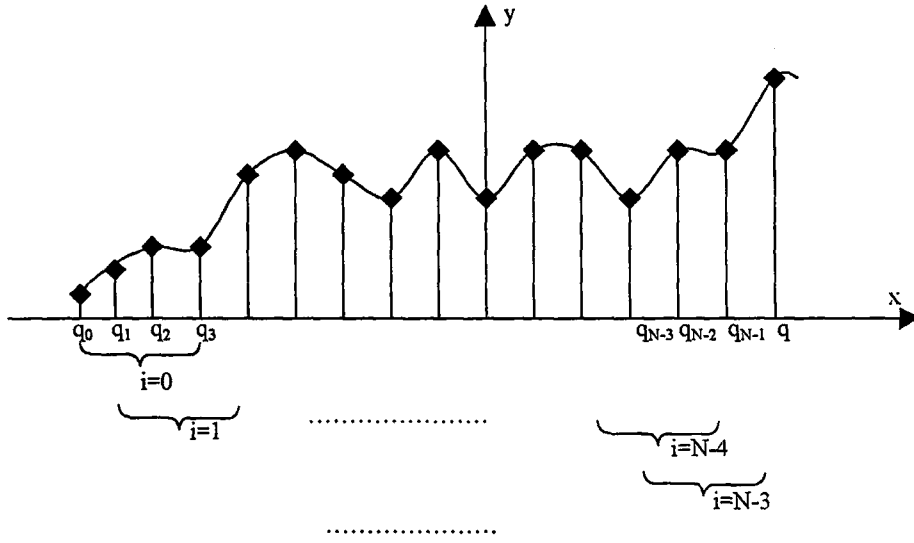
$$\frac{\partial E(k)}{\partial b_{1j}(k)} = e(k)(-1)w_j^2 \phi(b_{1j}(k)) \quad (4.29)$$

$$\frac{\partial E(k)}{\partial b_{2j}(k)} = e(k)(-1)w_j^2 \phi(b_{2j}(k)) \quad (4.30)$$

olarak hesaplanabilir. Böylece aktivasyon fonksiyonuna ait tüm parametreler yenilenmiş olur.

4.3. Spline Aktivasyon Fonksiyonu

Yukarıda anlatılan aktivasyon fonksiyonlarında, yapay sinir ağının etkinliğinin artırılması için ağı ayarlanabilecek parametre sayısının fazlalığı, ağı temsil yeteneğini arttıracak varsayımından hareket edilerek öne sürülmüştür. Bu yaklaşımla spline aktivasyon fonksiyonu, yapay sinir ağında kullanılan bir aktivasyon fonksiyonunu parçalara ayrılarak her bir noktayı uyarılma şansını verir. Ayrıca, yapay sinir ağının donanımsal olarak gerçekleştirilmesi durumunda bu aktivasyon fonksiyonunun sadece gerekli olan yada aktivasyon fonksiyonunu temsil eden aralıktaki değerleri bir hafıza ünitesine yüklenerek daha hızlı bir ulaşım sağlanmış olur. Yapay sinir ağının hem eğitim hızını artırması hem de donanımsal gerçeklemenin yanında yazılımda ve ağ yapısında sadeleştirmeye imkan tanınması yönüyle önemli bir üstünlüğü vardır. Şekil 4.3' te görüldüğü gibi x eksenini boyunca eşit aralıklarla örneklenen aktivasyon fonksiyonunun kontrol noktaları geriye yayılım algoritması ile uyarlanabilir. Burada x eksenini hücreye uygulanan net girişi ifade etmektedir. Net girişin aldığı değere karşılık, toplam karesel hataya göre i. eğri aralığının noktaları uyarlanır. Örnekler arasında rastlayan ara değerler en yakın alt kontrol noktası indisine yuvarlanarak uyarılma gerçekleştirilir.



Şekil 4.3. Catmull-Rom spline tabanlı aktivasyon fonksiyonunun sabit Δx adımlı kontrol noktalarının temsili

Denklem 4.31 de Catmull-Rom kübik spline tabanlı fonksiyon verilmiştir.

$$F_i(u) = \sum Q_{i+j} C_j(u) \quad (4.31)$$

burada Q kontrol noktalarını ifade ederken, C kübik polinomu kontrol noktalarının oluşturduğu eğri hareketini tanımlar. Eğri ara noktalarını bulur veya sadece denetim noktalarını yaklaştırır. Catmull-Rom kübik spline tabanlı fonksiyon aşağıdaki gibi tanımlanır.

$$\begin{aligned} C_0(u) &= 1/2(-u^3 + 2u^2 - u) \\ C_1(u) &= 1/2(3u^3 - 5u^2 + 2) \\ C_2(u) &= 1/2(-3u^3 + 4u^2 + u) \\ C_3(u) &= 1/2(u^3 - u^2) \end{aligned} \quad (4.32)$$

burada $0 \leq u \leq 1$ aralığında değişir. Denklem 4.31 ile verilen ifade matris formunda tekrar yazılırsa,

$$F_i(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \frac{1}{2} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} Q_i \\ Q_{i+1} \\ Q_{i+2} \\ Q_{i+3} \end{bmatrix} \quad (4.33)$$

şeklinde elde edilir. Kontrol noktaları eşit Δx aralıklarla kullanılırsa 4.33 denklemini aşağıdaki gibi tekrar yazılabilir.

$$F_i(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \frac{1}{2} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} q_i \\ q_i + \Delta x \\ q_i + 2\Delta x \\ q_i + 3\Delta x \end{bmatrix} \quad (4.34)$$

bu ifadenin doğrusallaştırılmış çözümü ise

$$F_i(u) = u\Delta x + q_i + \Delta x = u\Delta x + q_{i+1} \quad (4.34)$$

şeklinde elde edilir. Şekil 4.4'de genel bir spline hücresinin grafiksel gösterimi verilmiştir.

Bu ifadelerin ardından yapay sinir ağının ileri doğru çıkışının bulunması aşağıdaki şekilde yapılır.

$$z(k) = \frac{f_{net}(k)}{\Delta x} + \frac{N-2}{2}$$

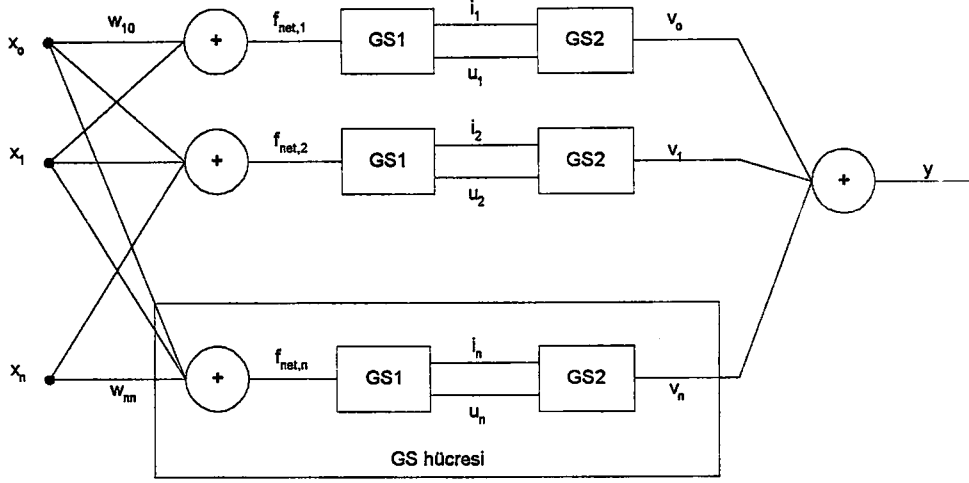
$$i(k) = \lfloor z(k) \rfloor$$

$$u(k) = z(k) - \lfloor z(k) \rfloor \quad (4.35)$$

burada $k=1,2,\dots,n$ yapay sinir ağına uygulanan giriş örneklerinin sayısını ifade eder. $\lfloor \cdot \rfloor$ floor operatörü olarak tanımlanır. z geçici olarak tanımlanan bir değişkendir. Ağ çıkışı ise,

$$y(k) = F_{k,i_k}(u_k) = \sum_{m=0}^3 q_{k,(i_k+m)} C_{k,m}(u_k) \quad (4.36)$$

olarak elde edilir. Burada 4.35 denklemini GS1 bloğunun çıkışını, 4.36 ise GS2 bloğunun çıkışını hesaplar.



Şekil 4.4 Uyarlamalı spline yapay sinir ağı

Toplam karesel hatanın geriye yayılımı ile ağırlıklara uygulanacak düzeltme miktarı $\frac{\partial E(k)}{\partial w_{ji}(k)}$ ifadesiyle hesaplanır. Kontrol noktalarına uygulanacak düzeltme miktarı ise $\frac{\partial E(k)}{\partial q_j(k)}$ ile hesaplanır. Elde edilen sonuçlarla ağırlıklar ve kontrol noktaları,

$$w_{ji}(k+1) = w_{ji}(k) + \Delta w_{ji}(k) \quad (4.37)$$

$$q_j(k+1) = q_j(k) + \Delta q_j(k) \quad (4.38)$$

şeklinde yenilenir. Burada,

$$\Delta w_{ji}(k) = -\alpha_w \frac{\partial E(k)}{\partial w_{ji}(k)} \quad (4.39)$$

ve

$$\Delta q_j(k) = -\alpha_q \frac{\partial E(k)}{\partial q_j(k)} \quad (4.40)$$

olur. α_w ve α_q öğrenme oranı olmak üzere, uygulanacak düzeltme miktarlarını bulmak için, 4.39 ve 4.40 denklemlerinin kısmi türevleri alınır. Bunun için,

$$\frac{\partial E(k)}{\partial w_{ji}(k)} = \frac{\partial E(k)}{\partial e(k)} \frac{\partial e(k)}{\partial y(k)} \frac{\partial y(k)}{\partial F(k)} \frac{\partial F(k)}{\partial u(k)} \frac{\partial u(k)}{\partial f_{net}(k)} \frac{\partial f_{net}(k)}{\partial w_{ji}(k)} \quad (4.41)$$

kısmi türevinin sonucu

$$\frac{\partial E(k)}{\partial w_{ji}(k)} = e(k)(-1)(1) \frac{\partial F(k)}{\partial u(k)} \frac{1}{\Delta x} x_i \quad (4.42)$$

olarak bulunur. Burada 4. terimin türevi

$$\frac{\partial F(k)}{\partial u(k)} = \left[3u^2 \quad 2u \quad 1 \right] \frac{1}{2} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} Q_i \\ Q_{i+1} \\ Q_{i+2} \\ Q_{i+3} \end{bmatrix} \quad (4.43)$$

olur. Kontrol noktalarının kısmi türevleri

$$\frac{\partial E(k)}{\partial q_j(k)} = \frac{\partial E(k)}{\partial e(k)} \frac{\partial e(k)}{\partial y(k)} \frac{\partial y(k)}{\partial F(k)} \frac{\partial F(k)}{\partial q(k)} \quad (4.43)$$

denklemleri ile aşağıdaki gibi bulunabilir.

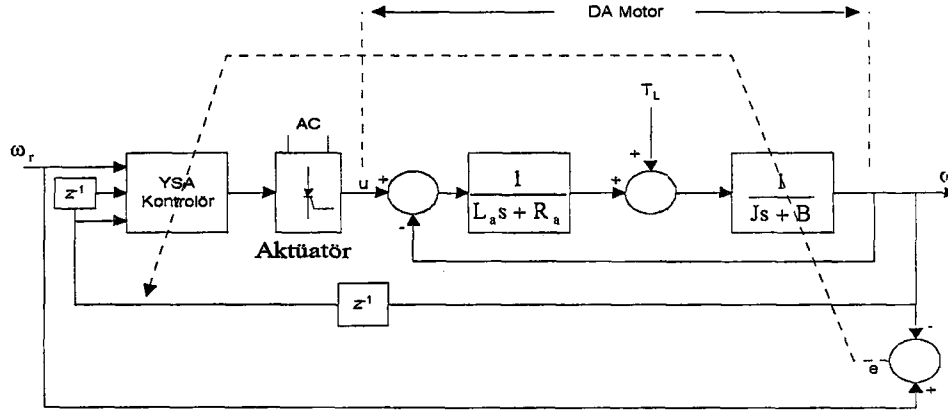
$$\frac{\partial E(k)}{\partial q_j(k)} = e(k)(-1)(1) \frac{\partial F(k)}{\partial q(k)} = -e(k)C_{k,m}(u_k) \quad m=0,...,3 \quad (4.44)$$

elde edilen bu ifadeler kullanılarak kontrol noktaları yenilenir.

4.4 DA Motorunun Kontrolünde YSA Yapısı ve Eğitimi

Bir sistemin ters yön dinamiklerini gösterecek şekilde yapay sinir ağı modelin seçilmesi ve eğitilmesi ters modelleme olarak söylenir. Şekil 4.5'te bu çalışmada da kullanılan özelleştirilmiş ters model yapısına göre DA motorunun yapay sinir ağı ters model/kontrol yapısı verilmiştir.

Yapay sinir ağının kontrolör olarak kullanıldığı ters model ağı, katmanlı yapay sinir ağı yapısı ile elde edilir ve yapay sinir ağı, sistem çıkışı ve referans giriş arasındaki hataya göre geriye yayılım algoritması ile eğitilir.



Şekil 4.5. Yapay sinir ağları ile DA motorların hız kontrolü blok diyagramı

Ters model yapısında yapay sinir ağı sistem girişi ve çıkışı zaman gecikme operatörü ile birlikte geri besleme yapılarak ağ girişine uygulanır. Yani model ağın girişleri sistem girişi, girişin gecikmişleri ile çıkışın gecikmişlerinden oluşturulur. Buna göre denklem (2.7) ile verilen DA motorunun durum denklemleri, T örnekleme periyodunu göstermek üzere, ileri farklar yöntemiyle ayrık zamana dönüştürülürse,

$$\begin{aligned} i_a(k+1) &= i_a(k) + T \left(-\frac{R_a}{L_a} i_a(k) - \frac{K_b}{L_a} \omega(k) + \frac{1}{L_a} v_a(k) \right) \\ \omega(k+1) &= \omega(k) + T \left(\frac{K_t}{J} i_a(k) - \frac{B}{J} \omega(k) - \frac{1}{J} T_L(k) \right) \end{aligned} \quad (4.45)$$

olarak bulunur. Burada $T_L=0$ alınarak ayrık zaman giriş çıkış modeli,

$$\omega(k) = k_1 \omega(k-2) + k_2 \omega(k-1) + k_3 v_a(k) \quad (4.46)$$

olur. Burada $k_1 \dots k_3$ katsayıları,

$$k_1 = -\frac{JL_a}{JL_a + (R_a J + BL_a)T + (K_m^2 + R_a B)T^2}, \quad k_2 = \frac{2JL_a + JR_a + BL_a}{JL_a + (R_a J + BL_a)T + (K_m^2 + R_a B)T^2}$$

$$k3 = \frac{K_m T^2}{JL_a + (R_a J + BL_a)T + (K_m^2 + R_a B)T^2}$$

olarak bulunur.

YSA ile kontrolörü gerçekleştirmek için, sistemin fark denklemleri elde edildikten sonra sistemin derecesine uygun çok katmanlı bir yapay sinir ağı oluşturulması gerekir. Böylece oluşturulan yapay sinir ağı sistem çıkışıyla referans giriş arasındaki hataya göre eğitilerek ağı ters sistemi öğrenmesi sağlanır.

Eğitim işlemi yapay sinir ağının giriş örneklerinin her birinin uygulamasında ağırlıklar ayarlanırsa örneksel öğrenme algoritması (çevrim içi = on line öğrenme), bütün örnekler uygulandıktan sonra bir defa ayarlanırsa toplu öğrenme (çevrim dışı = of line öğrenme) algoritması olarak tanımlanır. Eğitim işaretinin ağırlıklara göre gerçek gradyanı toplu öğrenmeyle elde edilir. Örneksel öğrenme gerçekte toplu öğrenmenin yani gerçek gradyanın bir yaklaşığıdır ve toplu öğrenmeye göre başarı şansı daha düşük olabilmektedir (Gökbulut, 1996).

Eğitimin amacı, yapay sinir ağının sistemin ters dinamiklerini taklit edecek şekilde öğrenmesini sağlamaktır. Bu nedenle yapay sinir ağının sistemin tersini temsil edebilecek şekilde doğru olarak oluşturulması gereklidir.

Ağın oluşturulması, sistem giriş çıkışı ve sistem derecesine göre seçilir. Ağ girişleri, sistemin ayrık zaman denklemindeki gecikmelerden ve sisteme uygulanan giriş fonksiyonundan oluştuğu için, ağ girişindeki hücre sayısı sistemin derecesine göre seçilmelidir. Fakat, fark denklemlerindeki bazı zaman gecikmelerinin girişe uygulanmaması ile ağ girişlerinin sayısı düşürülebilir. Dolayısı ile sistemin derecesi düşürülmüş bir modeli elde edilebilir ve böylece eğitim süresi düşürülmüş olur. (Bates v.d., 1993) Bu şekilde bir eğitimle, eğitim süresi düşürülmesine karşın ağın genelleştirme özelliği de önemli ölçüde düşmektedir. Bu sebeple, ağ oluşturulurken mümkün olduğunca, ağ girişindeki hücre sayısının sistem derecesine göre seçilmesine gayret edilmelidir.

Kontrol ağının, saklı katmanındaki hücre sayısının seçimi ise tamamen tecrübeyle belirlenir. Genel olarak, M giriş hücrelerinin sayısı, N çıkış hücrelerinin sayısını göstermek üzere, saklı katmandaki hücre sayısı $N(M+1)$ olarak belirlenebilir. (ZHANG v.d., 1995) Ama bu seçimde bir kesinlik yoktur. Saklı katmandaki hücre sayısı bu değerin üzerinde veya altında da seçilebilir, fakat saklı katmandaki hücre

sayısının giriş katmanındaki hücre sayısından çok yukarıda veya çok altında seçilmemesine özen gösterilmelidir. Zira saklı katmandaki hücre sayısının fazla seçilmesi halinde ağırlık taşması denilen, ağın öğrenmiş gibi görüldüğü halde öğrenememe durumu ortaya çıkabilmektedir.

Çıkış katmanındaki hücre sayısı ise tamamen sistem çıkışına göre seçilmelidir. Ağının da çıkış katmanında sistemin çıkış sayısı kadar hücre bulunmalıdır.

Ağ eğitimi hatanın geriye yayılımı kullanılarak yapılır. Bu amaçla eğitim işareti E , çıkıştaki hataların karelerinin toplamı, ω_{ref} arzu edilen çıkış (referans hız) ve ω sistem çıkışı olmak üzere k . örnekteki çıkış hatası,

$$e_c(k) = \omega_{ref}(k) - \omega(k) \quad j=1,2,...,m \quad (4.47)$$

ve eğitim işareti,

$$E(k) = \frac{1}{2} \sum_{k=1}^N e_c^2(k) \quad (4.48)$$

olarak hesaplanır. DA motorun girişi u_u olmak üzere kontrolör yapay sinir ağının çıkışı ile motora uygulanması gereken kontrol sinyali arasındaki hataya yöresel hata denir. Bu yöresel hata, arzu edilen bir kontrol sinyali belirli olmadığı için çıkış hatasının girişine göre sayısal türeviyle hesaplanır. Böylece yöresel hata,

$$e(k) = e_c(k) \frac{\omega(k) - \omega(k-1)}{u_u(k) - u_u(k-1)} \quad (4.49)$$

olarak bulunabilir. Ancak bu şekilde bulunan yöresel hata ters dinamiklerini tam olarak yansıtamaz. Bu nedenle, eğitim işaretinin ağırlıklara göre gradyanın işareti kontrolör yapay sinir ağının eğitiminde kullanılmak üzere yeterlidir.

Kontrolör ağın girişten çıkışa doğru hesaplamalarda kullanılan denklemler (3.7)-(3.9) ile verilmiştir. Buna göre, ağın giriş katmanını saklı katmana bağlayan ağırlıklar w_{ji}^1 , saklı katmanı çıkış katmanına bağlayan ağırlıklar w_{mj}^2 olmak üzere, k . örnekte kontrol ağının saklı katmanındaki ağırlıkların uyarlanması,

$$w_j^2(k+1) = w_j^2(k) + \Delta w_j^2(k) \quad (4.50)$$

kullanılır. Burada,

$$\Delta w_{ji}^2(k) = -\alpha \frac{\partial E(k)}{\partial w_{ji}^2(k)} \quad (4.51)$$

ağırlıklara uygulanacak düzeltme miktarını ifade eder. Giriş katman ağırlıklarının uyarlanmasında,

$$w_{ji}^1(k+1) = w_{ji}^1(k) + \Delta w_{ji}^1(k) \quad (4.52)$$

denklemini kullanılır. Burada,

$$\Delta w_{ji}^1(k) = -\alpha \frac{\partial E(k)}{\partial w_{ji}^1(k)} \quad (4.53)$$

giriş katmanındaki ağırlıklara uygulanacak düzeltme miktarını ifade eder.

S ağırlıkları uyarlama iterasyonu olmak üzere, N adet örneğin uyarlanması sona ağırlıkların toplu olarak uyarlanması ise,

$$w(s) = w(s-1) + \Delta w(s) \quad (4.54)$$

denklemini kullanılırken Δw ağırlıklara uygulanacak düzeltmeyi ifade eder ve aşağıdaki denklem ile hesaplanır.

$$\Delta w(s) = -\alpha \frac{\partial E}{\partial w(s)} \quad (4.55)$$

O halde, geriye yayılım algoritmasında, denklem 4.51 ve 4.53'deki kısmi türevler alınırsa;

$$\frac{\partial E(k)}{\partial w_{ji}^2(k)} = \frac{\partial E(k)}{\partial e(k)} \frac{\partial e(k)}{\partial \omega(k)} \frac{\partial \omega(k)}{\partial u_u(k)} \frac{\partial u_u(k)}{\partial w_{ji}^2(k)} \quad (4.56)$$

elde edilir ve türevler sırasıyla,

$$\frac{\partial E(k)}{\partial w_{ji}^2(k)} = e_r(k)(-1) \frac{\omega(k) - \omega(k-1)}{u_u(k) - u_u(k-1)} v_j(k) \quad (4.57)$$

saklı katmanı çıkışa bağlayan ağırlıklara uygulanması gereken düzeltme miktarı olarak bulunur. Giriş katmanını saklı katmana bağlayan ağırlıklar için eğitim işaretinin kısmi türevleri alınır;

$$\frac{\partial E(k)}{\partial w_{ji}^1(k)} = \frac{\partial E(k)}{\partial e(k)} \frac{\partial e(k)}{\partial \omega(k)} \frac{\partial \omega(k)}{\partial u_u(k)} \frac{\partial u_u(k)}{\partial v_j(k)} \frac{\partial v_j(k)}{\partial u_j(k)} \frac{\partial u_j(k)}{\partial w_{ji}^1(k)} \quad (4.58)$$

ifadesinin sonucu sırasıyla,

$$\frac{\partial E(k)}{\partial w_{ji}^1(k)} = e(k)(-1) \frac{\omega(k) - \omega(k-1)}{u_u(k) - u_u(k-1)} w_{ji}^2 \phi(u_j(k)) x_i(k) \quad (4.59)$$

olarak hesaplanabilir ve bu bağıntılarla kontrolör ağı ağırlıkları yenilenmiş olur.

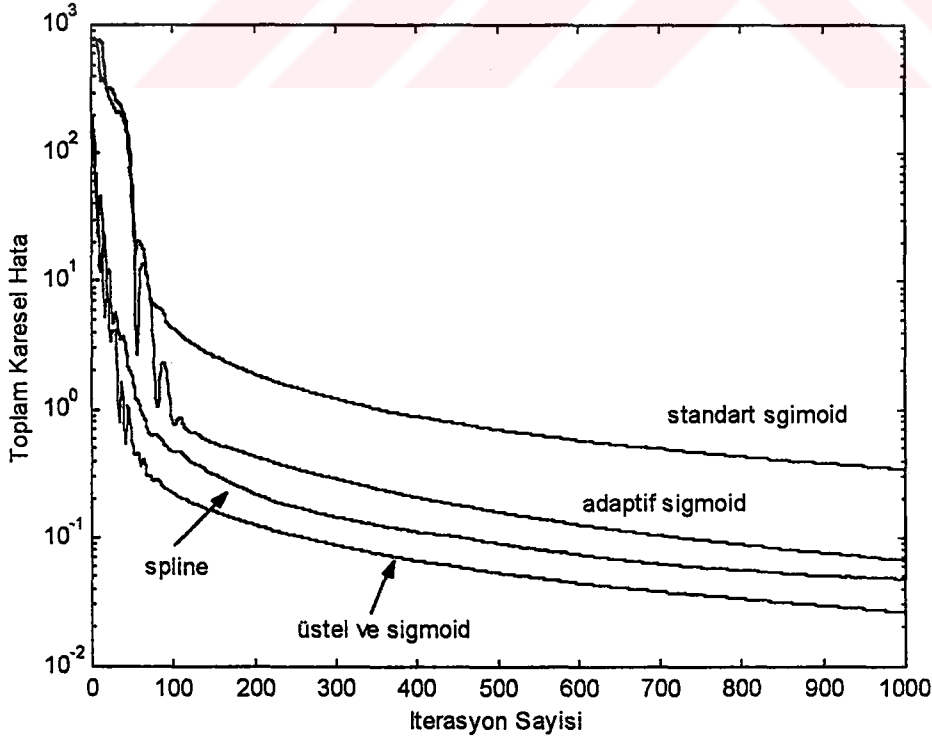
Eğitime başlamadan önce, ağı ağırlıkları ilk halde rasgele seçilerek simetrisi bozulur. Bu durumda kontrolör yapay sinir ağına, referans giriş sinyali uygulandığında, ağı çıkışından elde edilen $u_u(k)$ kontrol sinyaline göre sistem çıkışından referans girişe göre hatalı bir $y(k)$ değeri elde edilir. Elde edilen bu hatalı $y(k)$ değeri referans işareti ile karşılaştırılır. Ortaya çıkan hata işareti, geriye yayılım algoritması kullanılarak yukarıdaki bağıntılara göre hesaplanıp ağı ağırlık değerlerinin ayarlanması sağlanır. Bu şekilde tekrarlanan eğitimle ağ, giriş işaretine nasıl doğru cevap vereceğini öğrenir. Eğitim sayısı artırıldıkça ağ daha da iyi öğrenerek çıkış hatasını azaltır. Hatanın istenilen değerin altına düşmesi sağlanınca ağı giriş ve saklı katmanındaki ağırlıklar saklanarak eğitim işlemi kesilir. Eğitimin kesilmesine, toplu öğrenmede hataların karelerinin toplamının istenilen değere düşmesi ile, örneksel öğrenmede ise program zaman zaman durdurularak sistem çıkışındaki düzeltme kontrol edilerek karar verilir.

5. SİMÜLASYON SONUÇLARI

Bölüm 4.4’de adaptif aktivasyon fonksiyonlu YSA’ları ile kontrol yapısı çıkarılan DA motorunun, MATLAB programı kullanılarak simülasyonu gerçekleştirilmiştir. Simülasyon çalışmasında kullanılan motorun parametreleri bölüm 2.4’te verilmiştir. Motorun nominal giriş gerilimine göre hızı normalize edilmiştir. Buna göre verilen grafiklerdeki 0.7136 değerleri motorun nominal hız değerine karşılık gelmektedir. Ayrıca uygulamada motora nominal gerilimden fazlası uygulanamayacağı dikkate alınarak kontrolör çıkışında sınırlandırıcı kullanılmıştır. Şekil 5.1’de saklı katmanda 3 hücrenin kullanıldığı kontrolör yapay sinir ağı ile kontrol edilen ve tristörlü doğrultucu ile beslenen doğru akım motorunun sinüsoidal bir referans girişle eğitimi sırasında hataların kareleri toplamının düşüşü görülmektedir. Bir aktüatör olarak görev yapan kontrollü doğrultucunun doğrusal olmayan bağıntısı;

$$f(.) = \frac{u_{\max}}{\pi} (\cos(\alpha) + 1) \quad (5.1)$$

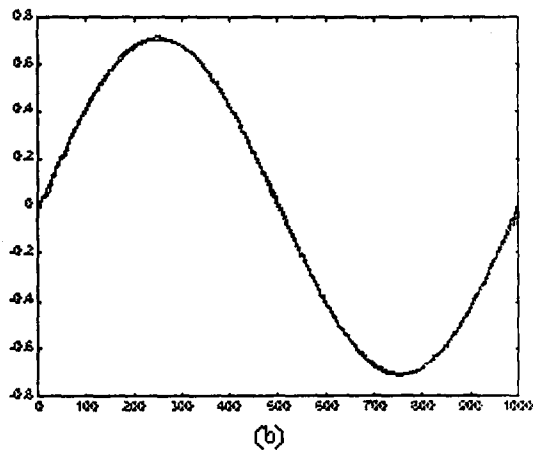
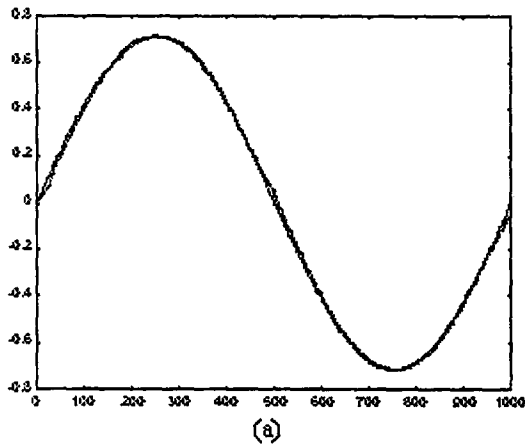
kullanılmıştır. Burada, gerilimin açığa dönüşümü $\alpha = -\frac{\pi}{u_{\max}}u + \pi$ şeklinde doğrusaldır.

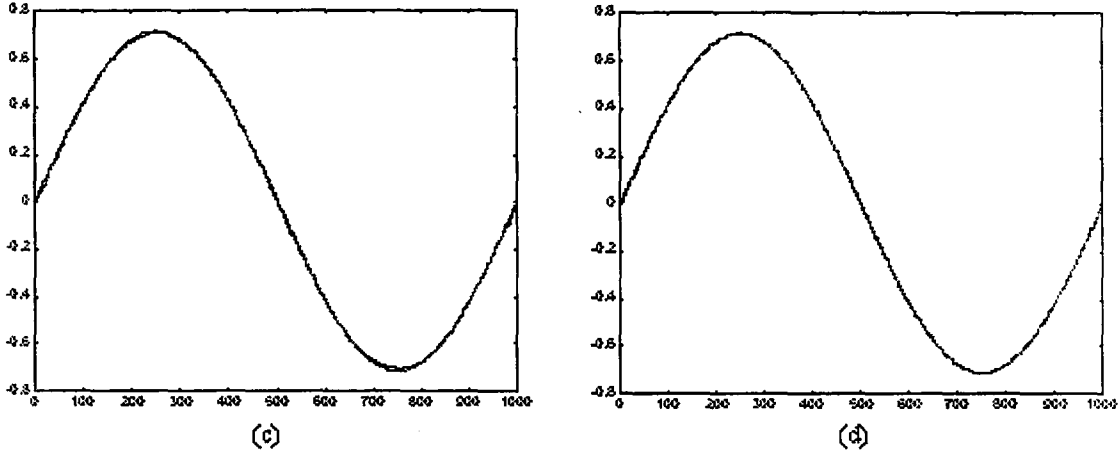


Şekil 5.1. Tristörlü doğrultucu ile beslenen DA motorun hız kontrolünde kontrolör YSA’ların eğitimi sırasında toplam karesel hataların değişimi

Şekil 5.1’de standart sigmoid aktivasyon fonksiyonu kullanılan kontrolör yapay sinir ağı için, toplam karesel hatanın düşüş hızının diğerlerine göre daha yavaş olduğu görülmektedir. Burada, her bir aktivasyon fonksiyonunun kullanıldığı kontrolör yapay sinir ağı, aynı öğrenme oranı ve aynı momentum katsayıları ile aynı başlangıç ağırlıklarından başlatılarak bin iterasyon için toplu öğrenme ile eğitilmiştir. Şekil 5.1’de görüldüğü gibi yaklaşık aynı performansı gösteren dört kontrolör ağından adaptif sigmoid aktivasyon fonksiyonu, 1000 iterasyonluk eğitim süreci sonunda toplam hatayı diğerleriyle yaklaşık aynı değere kadar indirmesine rağmen, diğer aktivasyon fonksiyonlarının kullanıldığı kontrolör ağlarının toplam hatayı minimize etme hızı daha yüksektir. En iyi performansı ise üstel ve sigmoid aktivasyon fonksiyonunun kullanıldığı kontrolör yapay sinir ağı gerçekleştirmiştir.

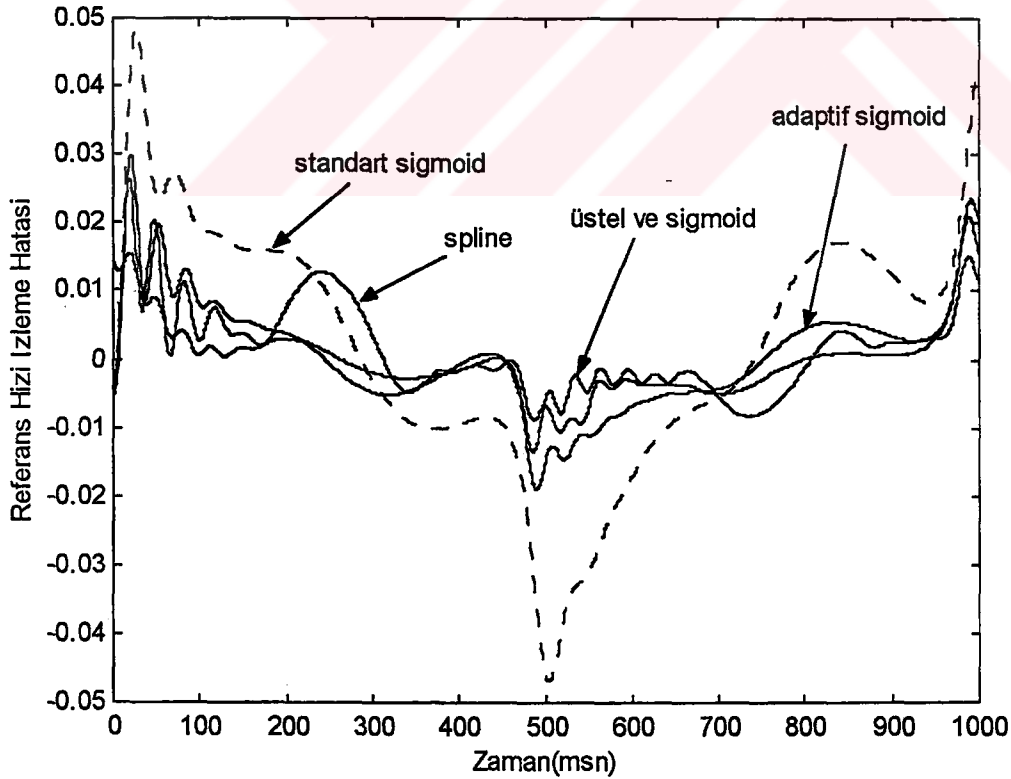
Saklı katmanda üç hücre kullanılarak eğitilen kontrolör ağların 1000 iterasyonluk eğitim sonucunda sinüsoidal bir referans hız girişi için DA motorunun kontrol performansları Şekil 5.2’de verilmiştir. Simülasyon sonuçlarına göre, Şekil 5.1’de verilen toplam hatanın düşüş hızlarına uyumlu olarak standart sigmoid aktivasyon fonksiyonunun kullanıldığı kontrolör ağının performansı diğerlerine göre kotüdür. Yine toplam karesel hatanın diğer aktivasyon fonksiyonları için birbirine yakın olmasına paralel olarak, birbirlerine yakın derecede öğrenme gerçekleşmiştir. Ancak, küçük farkların bu sonuçlardan gözlenmesi güçtür. Bu nedenle, Şekil 5.3’te izleme hatası verilmiştir. Şekil 5.2-c-d’de görüldüğü gibi spline ve üstel ve sigmoid aktivasyon fonksiyonlu kontrolör ağların referans hızı izleme performansı oldukça iyidir. Aynı gözlem şekil 5.3’deki izleme hatalarından da rahatlıkla fark edilebilir.





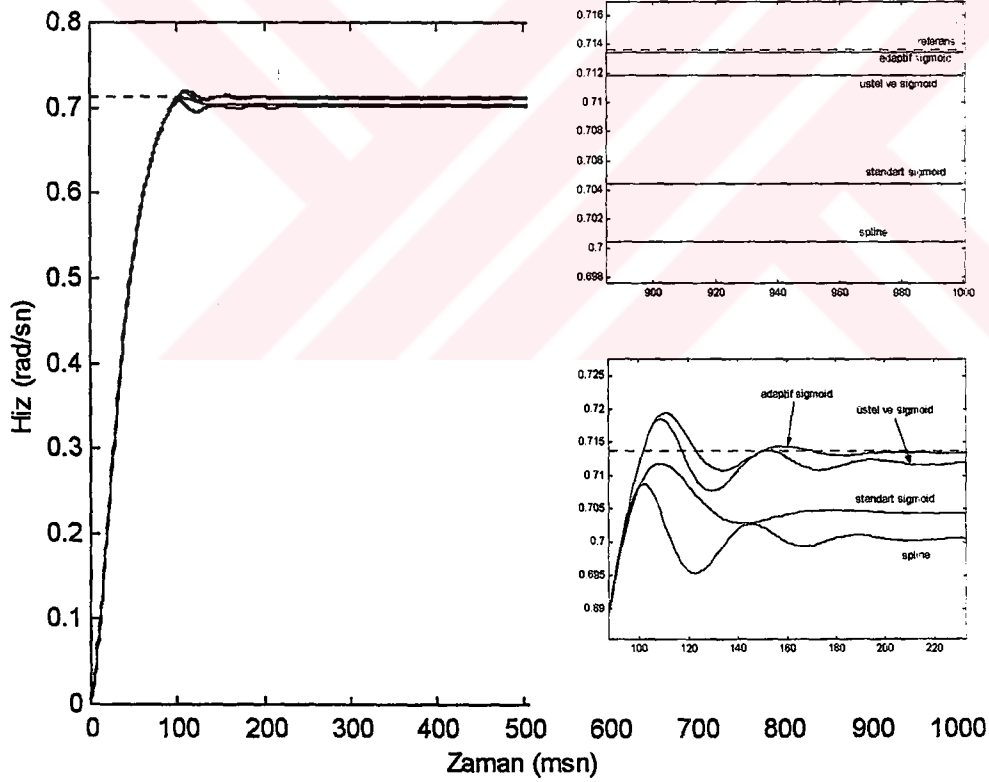
Şekil 5.2 Eğitim örnekleri (referans) ve sistem çıkışı a) standart sigmoid'li kontrolör ağı b) adaptif sigmoid'li kontrolör ağı c) spline'li kontrolör ağı d) üstel ve sigmoid'li kontrolör ağı performansları

Şekil 5.3'de kontrolör ağların sinüsoidal bir referans hız girişini izleme hataları görülmektedir. Bin iterasyonluk eğitim sonunda aynı referans girişi yaklaştırma hataları, farklı aktivasyon fonksiyonlarından oluşturulan kontrolör ağlarının, standart sigmoid dışında, oldukça küçük olduğu görülmektedir.



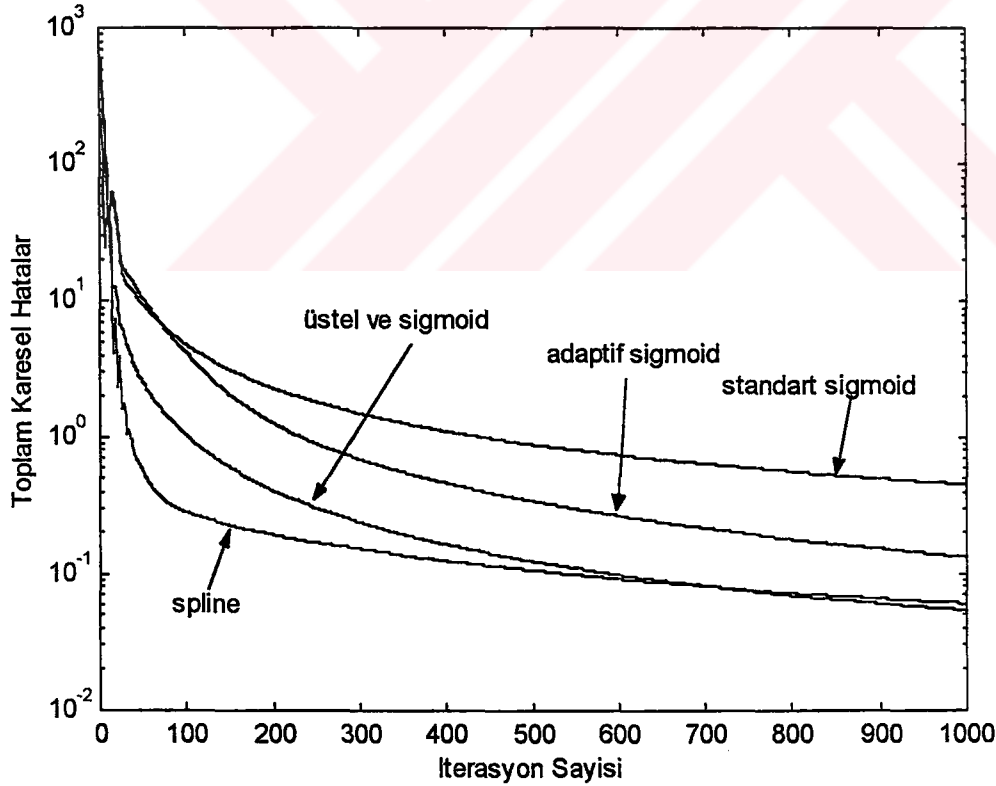
Şekil 5.3 Referans hız girişi ile motor hızı arasındaki hatalar

Şekil 5.4'de 1000 iterasyonluk eğitim sonunda eğitilerek ağırlıkların ve aktivasyon fonksiyonlarının parametreleri saklanan kontrolör YSA'larının motorun basamak hız referansını genellemesi karşılaştırmalı olarak verilmiştir. Toplam karesel hatanın son değerinin oldukça düşük olduğu spline aktivasyon fonksiyonun kullanıldığı kontrolör ağılar genellemede aynı performansı gösterememiştir. En iyi performans adaptif sigmoid aktivasyon fonksiyonun yer aldığı kontrolör ağıının olmuştur. Buna karşın toplam karesel hatanın düştüğü son değer açısından en iyi performansı gösteren üstel ve sigmoid aktivasyonlu kontrolör ağı genellemede ancak ikinci en iyi performansı göstermiştir. Burada dikkati çeken spline aktivasyon fonksiyonlu kontrolör ağı olmuştur. Bu kontrolör ağı, şekil 5.1'de görüldüğü gibi toplam karesel hatanın değişiminde ikinci en iyi performansı gösterirken genelleme performansı en kötü olan kontrolör ağı olmuştur.



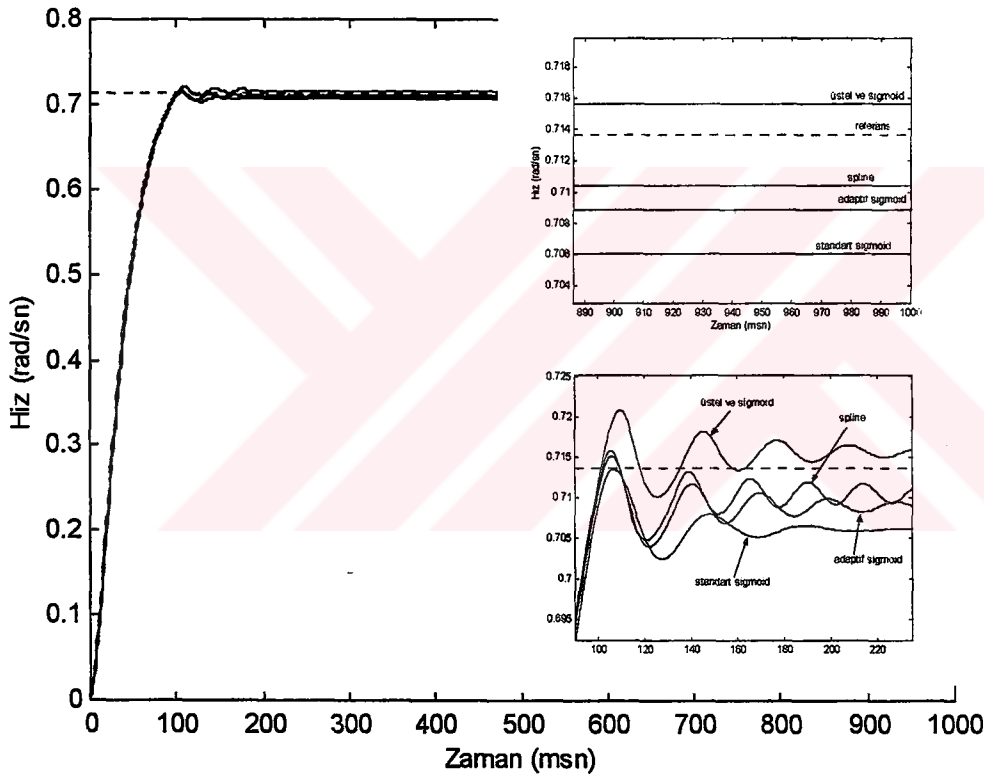
Şekil 5.4 Eğitilmiş kontrolör ağların basamak bir referans hız girişini genelleme performansları

Şekil 5.5’de doğru akım motorunun önüne girişin karesini alan ($f(.)=u^2$) bir aktüatör kullanarak eğitilen kontrolör ağlar için toplam karesel hatanın değişiminin verildiği grafik görülmektedir. Burada, kontrolör ağlarının saklı katmanında 5 hücre kullanılmıştır. Her bir aktivasyon fonksiyonunun kullanıldığı kontrolör ağları, aynı rasgele başlangıç ağırlıkları ile eğitime başlatılarak, 1000 iterasyon boyunca yine aynı öğrenme oranıyla (0,001) eğitilmiştir. Eğitim sonunda, sonuç ağırlıkları ve aktivasyon fonksiyonunun eğitilen parametreleri saklanarak motorun nominal hız değeri basamak referans giriş olarak uygulanıp genelleme performansı incelenmiştir. Şekil5.5’de görüldüğü gibi standart sigmoid aktivasyon fonksiyonunun kullanıldığı kontrolör ağının performansı diğer kontrolör ağlara göre daha kötüdür. Diğer üç aktivasyon fonksiyonunun performansları ise birbirine oldukça yakındır. Ancak, dikkat edilirse hatanın düşüş hızları karşılaştırıldığında, spline aktivasyon fonksiyonunun kullanıldığı kontrolör ağı diğerlerine göre hatayı daha hızlı bir şekilde küçültmüştür. Yine ikinci olarak en hızlı hatayı düşüren kontrolör ağı üstel ve sigmoid aktivasyon fonksiyonunun kullanıldığı kontrolör ağı olmuştur.



Şekil 5.5 $f(.)=u^2$ aktüatörü ile DA motorun hız kontrolünde kontrolör YSA’ların eğitimi sırasında toplam karesel hataların değişimi

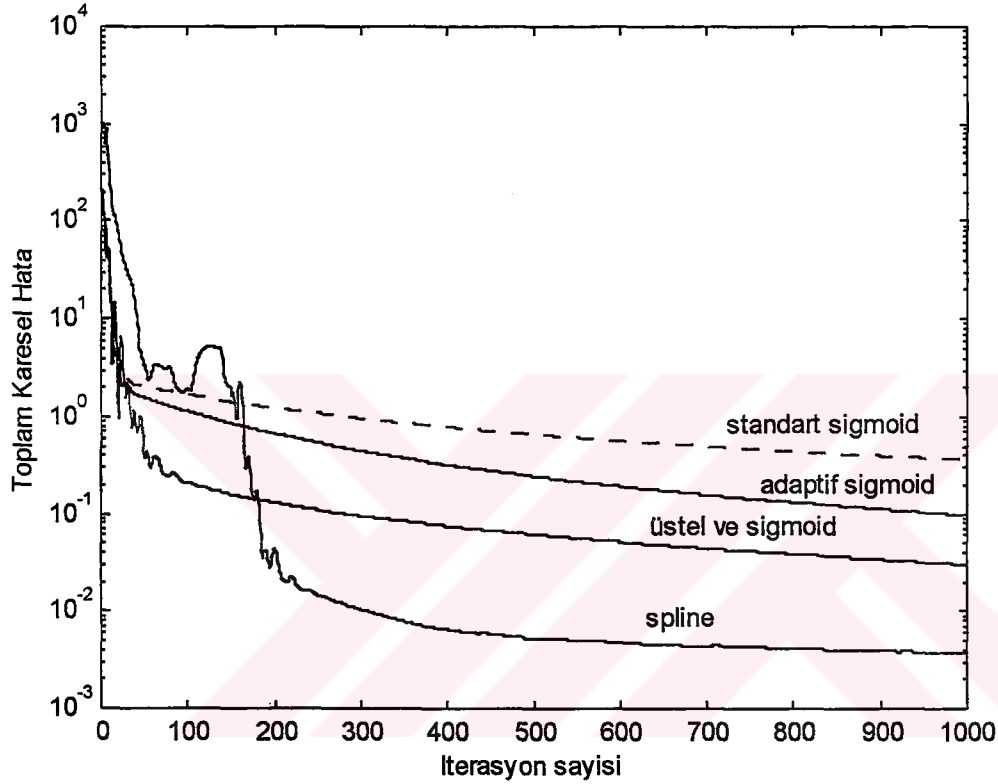
Şekil 5.6'da sistem girişinde $f(.)=u^2$ aktüatörü bulunan doğru akım motorunun YSA ile kontrolünde, 1000 iterasyonluk eğitim sonunda kontrolör YSA'ların basamak referans hız girişini genelleme performansları karşılaştırılmıştır. Şekilden de görülebileceği gibi en iyi performansı üstel ve sigmoid aktivasyon fonksiyonunun yer aldığı kontrolör ağı gösterirken, en kötü performansı yine standart sigmoid aktivasyon fonksiyonunun kullanıldığı kontrolör ağı göstermiştir. Spline ve adaptif sigmoid aktivasyon fonksiyonlarının kullanıldığı kontrolör ağlar birbirlerine oldukça yakın performans göstermiştir.



Şekil 5.6 $f(.)=u^2$ aktüatörü ile kontrolör ağların basamak bir referans hız girişini genelleme performansları

Şekil 5.7'de doğru akım motorunun önüne sigmoid aktivasyon fonksiyonu aktüatör olarak kullanıldığında, kontrolör ağların 1000 iterasyon için toplam karesel hataların değişimi görülmektedir. Burada kontrolör ağlarının saklı katmanında 7 hücre kullanılmıştır. Aynı yapıya sahip kontrolör ağları yine aynı başlangıç şartları ile 1000 iterasyon boyunca aynı öğrenme oranıyla (0,001) eğitilmiştir. Önceki kontrolör ağlarının performanslarına benzer şekilde performans sergileyen kontrolör ağların

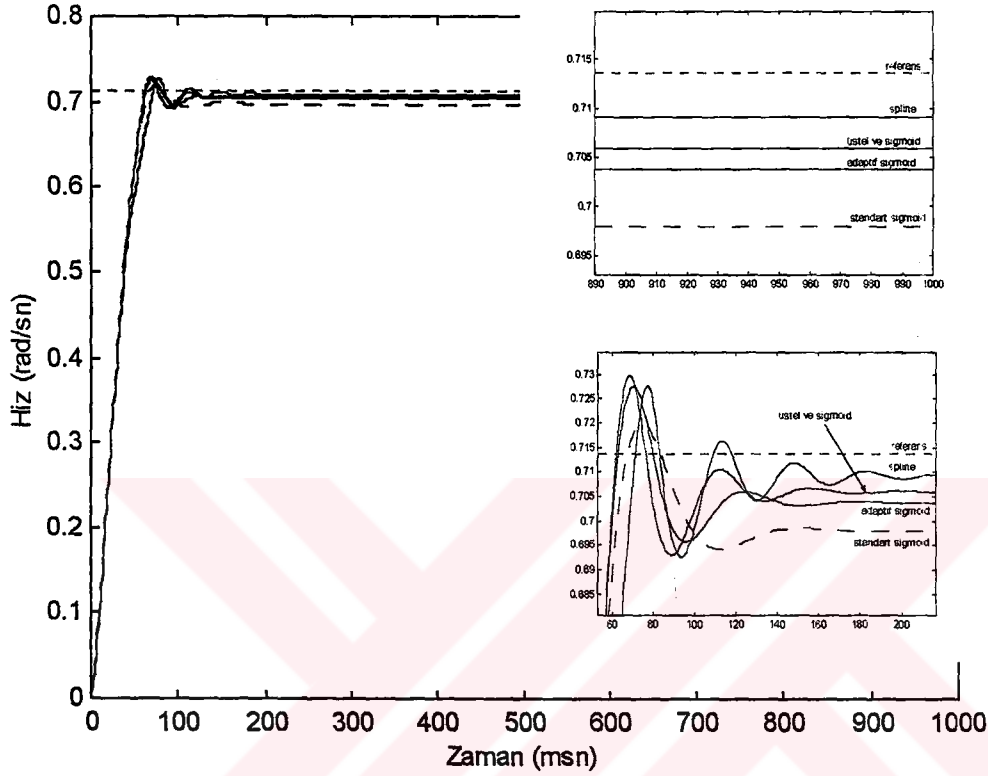
toplam karesel hataların düşüş eğrileri Şekil 5.7’de verilmiştir. Şekilden görüldüğü gibi toplam karesel hatanın düşüş hızı spline aktivasyon fonksiyonu için oldukça iyidir. İkinci en iyi performansı üstel ve sigmoid aktivasyon fonksiyonlu kontrolör ağı göstermiştir. Standart sigmoid aktivasyon fonksiyonunun kullanıldığı kontrolör ağı ise daha kötü bir performans göstermiştir.



Şekil 5.7 Sigmoid aktüatör ile DA motorun hız kontrolünde kontrolör YSA'ların eğitimi sırasında toplam karesel hataların değişimi

Doğru akım motorunun önünde sigmoid aktüatör kullanıldığı halde 1000 iterasyon eğitilen kontrolör ağlarının, doğru akım motorunun nominal hız değeri için genelleme performanslarının karşılaştırılması şekil 5.8’de verilmiştir. Şekilden de görüldüğü gibi, toplam karesel hatanın düşüş hızına benzer performans genelleme performanslarına yansımıştır. En iyi genelleme performansını spline aktivasyonu fonksiyonlu kontrolör ağı gösterirken, en kötü performansı yine standart sigmoid aktivasyon fonksiyonlu kontrolör ağı göstermiştir. Şekle dikkat edilirse toplam karesel hatanın eğitim sonundaki değerler spline dışında birbirine yakın olmasına rağmen

genelleme performanslarında standart sigmoid aktivasyon fonksiyonlu kontrolör ağı oldukça kötü bir performans sergilemiştir.



Şekil 5.8 Sigmoid aktüatörü ile kontrolör ağların basamak bir referans hız girişini genelleme performansları

Tablo 5.1’de, tristörlü doğrultucu ile beslenen DA motorunun hız kontrolünde, kontrolör YSA’ların eğitimi sonunda toplam karesel hataların aldığı son değerler, saklı katmanda farklı sayıda hücrenin kullanıldığı yapılar için verilmiştir. Tablodan görüldüğü gibi en iyi performansı, üstel ve sigmoid aktivasyon fonksiyonu ile gerçekleştirilen YSA’lı kontrollerler sağlanmıştır. Diğer yandan en kötü performans, standart sigmoid fonksiyonlu YSA’lı kontrolörün olmuştur. Buna karşın spline aktivasyon fonksiyonlu YSA’lı kontrolör en iyi performansa oldukça yakın değerlere ulaşmıştır. Tablodan görüldüğü gibi bu sıralama saklı katmandaki hücre sayılarının değişimiyle değişmemektedir. Ancak, bu performans sıralaması genellemede aynı değildir. Örneğin şekil 5.2’de verilen basamak genellemesine bakıldığında standart sigmoid aktivasyon fonksiyonlu kontrolör YSA, spline aktivasyon fonksiyonlu

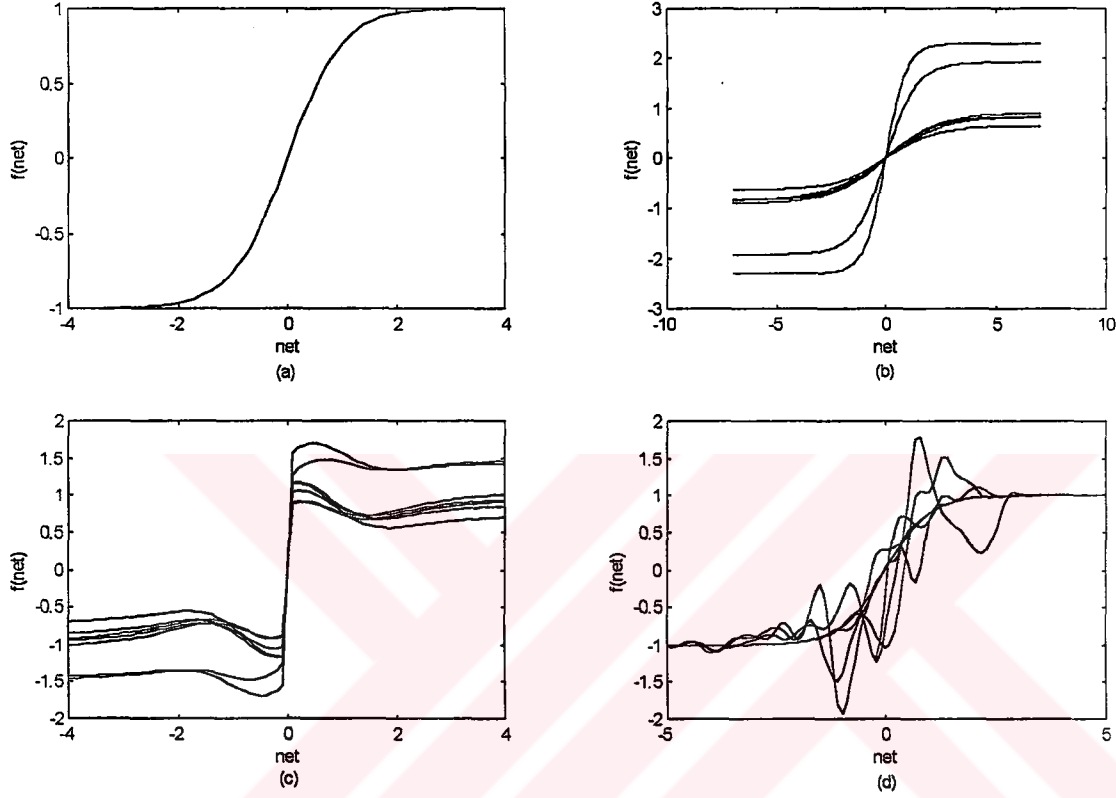
kontrolör YSA dan daha iyi genelleme yapmıştır. Diğer aktüatörlerin kullanıldığı kontrol yapılarında benzer sonuçlar alındığı için burada verilmemiştir.

Tablo 5.1 Tristörlü doğrultucu ile beslenen DA motorun hız kontrolünde YSA kontrolörlerin toplam karesel hatayı küçülttükleri son değerler

	N3_3_1	N3_5_1	N3_7_1
Sigmoid	0.3451	0.2199	0.3290
Adaptif sigmoid	0.0671	0.0969	0.1108
Üstel ve sigmoid	0.0264	0.0165	0.0323
Spline	0.0480	0.0455	0.0408

Şekil 5.9'da, denklem 5.1 aktüatör ile DA motorunun hız kontrolünde, saklı katmanında 7 hücrenin bulunduğu YSA'lı kontrolörler 1000 iterasyonluk eğitim sonunda, parametreleri eğitim esnasında adapte edilen aktivasyon fonksiyonlarının eğitim sonundaki şekilleri görülmektedir. Şekil 5.9-a'da, standart sigmoid aktivasyon fonksiyonunun parametreleri sabit olarak alındığı için saklı katmandaki her hücrede fonksiyonun şekli aynıdır. Şekil 5.9-b'de, iki parametresi başlangıçta 1 alınarak eğitilen standart sigmoid fonksiyonunun eğitim sonundaki şekilleri verilmiştir. Şekil 5.9-c'de, başlangıçta tüm parametreleri 1 olarak alınan, üstel ve sigmoid aktivasyon fonksiyonunun eğitim sonundaki şekli görülmektedir. Şekil 5.9-d'de ise, başlangıçta standart sigmoid fonksiyonu 0,3'lük sabit aralıkla örneklenerek 34 kontrol noktasının eğitimi sonucunda saklı katmandaki spline aktivasyon fonksiyonlarının şekli görülmektedir. Şekilden de görüldüğü gibi saklı katmanın her bir hücresindeki spline aktivasyon fonksiyonunun 34'er parametresi eğitildiği için, başlangıçta standart sigmoid aktivasyon fonksiyonundan örneklendiği halde oldukça farklı bir şekle dönüşmüştür. Aktivasyon fonksiyonunun eğitilebilir parametre sayısı ve bu parametrelerin eğitiminin kontrolör ağı kazandırdığı performans artışı dikkat çekicidir. Ancak, parametrelerin eğitim esnasında adapte edilmesi, hesaplamalara yeni bir yükte getirmekle birlikte ağ yapısında sadeleştirmeye imkan vermesi açısından önemsenmeyebilir. Çünkü YSA'lı kontrolörün sistem denetimini gerçekleştirirken göstereceği başarının büyük bölümünü öğrenme hızı oluşturur. YSA'nın eğitimi sistemlere göre farklı sürelerle sahip olması,

öğrenme hızının önemini ortaya koyması bakımından incelenmesi gereken bir çalışma alanı oluşturur. Burada verilen simülasyon sonuçlarından da görülmektedir ki, adapte edilebilen parametre sayısının mümkün olduğunca artması YSA'nın öğrenme hızını artırmaktadır.

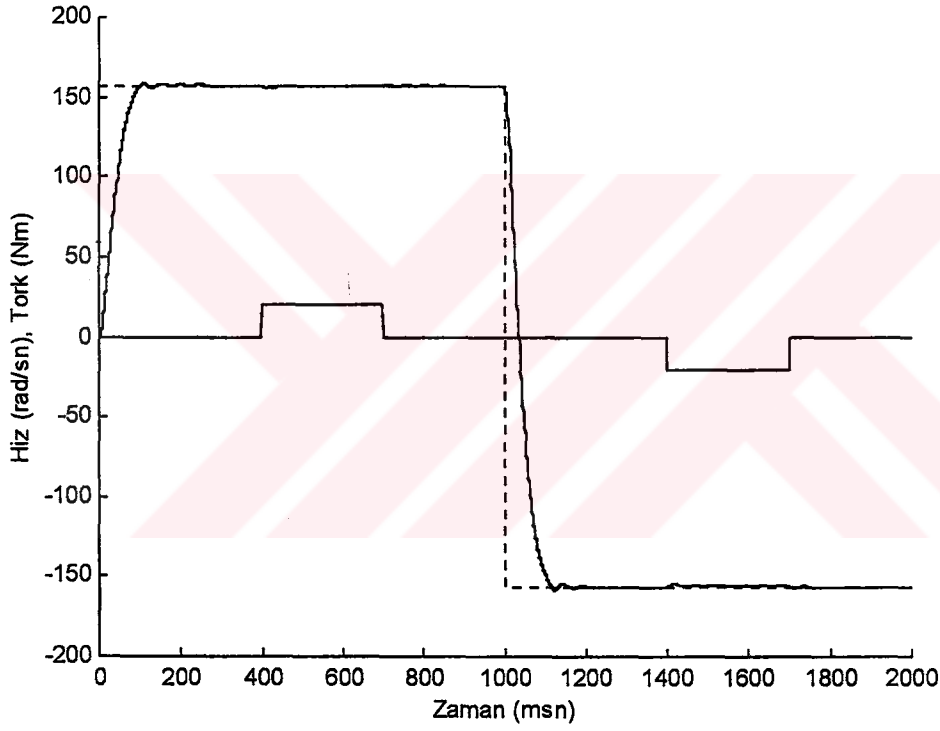


Şekil 5.9 Tristörlü doğrultucu ile beslenen DA motorun YSA ile hız kontrolünde YSA'ların eğitimi sonundaki uyarlanabilir aktivasyon fonksiyonları

a) standart sigmoid b) adaptif sigmoid c) üstel ve sigmoid d) spline

DA motorunun hız kontrolünde kullanılan adaptif aktivasyon fonksiyonlu YSA yapılarının incelenmesinde, kullanılacak aktivasyon fonksiyonu kontrolör ağının performansını artırdığı gibi, ağ yapısını da sadeleştirme imkanı verdiği görülmüştür. Üstel ve sigmoid aktivasyon fonksiyonlu kontrolör ağların performansı oldukça iyi olmakla birlikte fiziksel gerçeklemede, saklı katmandaki hücre çıkışlarında aktivasyon fonksiyonunun çıkışının hesaplanması gerekmektedir. Spline aktivasyon fonksiyonunun kullanılması durumunda, aktivasyon fonksiyonunu oluşturan kontrol noktaları bir tablodan okutularak hesaplamadan kurtulma şansını vermekle birlikte hız artışına da neden olur.

Şekil 5.10 'da tristörlü doğrultucu ile beslenen DA motorun hız kontrolünde YSA kontrolörün değişken yük momentinde performansı görülmektedir. Motora 0,4. saniyede ve 1,4. saniyede 0,3 saniye süreyle nominal yük momentini (21N) uygulanmıştır. Şekilden görüldüğü gibi motor hızı referansı oldukça iyi izlemektedir. Yapay sinir ağlarının öğrenme ve genelleme yeteneğinin açık bir şekilde görüldüğü kontrol performansı, tüm aktivasyon fonksiyonları ile gerçekleştirilen YSA kontrolörlerle elde edilmiştir. Ancak bazı aktivasyon fonksiyonları ile bu performansı elde etmek için eğitim süresini uzatmak gerekir.



Şekil 5.10. Tristörlü doğrultucu ile beslenen DA motorun YSA ile hız kontrolünde değişken yük momentinde kontrol performansı

6. SONUÇ

Bu tez çalışmasında, DA motorunun hız kontrolü adaptif aktivasyon fonksiyonlu YSA ile gerçekleştirilmiştir. Kullanılan YSA kontrolörler, geriye yayılım algoritması ve toplu öğrenme yöntemi ile eğitilmiştir. Kontrolör ağlar her bir aktivasyon fonksiyonu için, saklı katmanında 3, 5 ve 7 hücre kullanılarak DA motorunun hız kontrolündeki performansları karşılaştırılmıştır. Elde edilen sonuçlardan; DA motorunun hız kontrolünde motorun önüne üç farklı aktüatör kullanılarak eğitilen adaptif aktivasyon fonksiyonlu YSA kontrolörlerden, üstel ve sigmoid ile spline aktivasyon fonksiyonlu kontrolör ağının diğerlerine göre, öğrenme hızının daha iyi olduğu görülmüştür. En yavaş öğrenen aktivasyon fonksiyonu standart sigmoid aktivasyon fonksiyonu olmuştur. Adaptif sigmoid aktivasyon fonksiyonlu kontrolör ağı ise öğrenme hızıyla standart sigmoiden daha iyi sonuç vermiştir. Özellikle sigmoid aktüatör kullanılarak yapılan hız kontrolünde spline aktivasyon fonksiyonu diğerlerinden daha başarılı olmuştur.

Aktivasyon fonksiyonlarının genellemesinde, tristörlü doğrultucunun aktüatör olarak kullanıldığı kontrol yapısı için, spline fonksiyonlu kontrolör ağı, eğitim sonunda toplam karesel hatayı daha iyi minimize etmesine rağmen basamak genellemede en kötü performansı göstermiştir. Bu aktüatör için en iyi performansı üstel ve sigmoid aktivasyon fonksiyonu sağlarken, ikinci en iyi performansı adaptif sigmoid gerçekleştirmiştir. Standart sigmoid fonksiyonlu kontrolör ağı ise splinelı ağdan daha iyi sonuç vermiştir.

Aktüatör olarak girişin karesini alan eleman kullanıldığında; üstel ve sigmoid ve spline aktivasyon fonksiyonlu kontrolör ağlar hatayı minimize etme yönünden benzer performans göstermiştir. Ancak, genelleme yönünden üstel ve sigmoid fonksiyonlu kontrolör ağı daha iyi sonuç vermiştir. Bu aktüatör içinde en kötü performansı standart sigmoid aktivasyon fonksiyonu göstermiştir.

Sigmoid aktüatör ile yapılan hız kontrolünde, hatanın minimize edilmesinde ve genellemede splinelı kontrolör ağı oldukça başarılı olmuştur. Üstel ve sigmoid fonksiyonu her iki durum içinde ikinci sırayı almıştır. Bu aktüatör içinde en kötü performansı standart sigmoid aktivasyon fonksiyonlu kontrolör ağı göstermiştir.

Yapılan karşılaştırmalar için tüm ağ yapıları, aynı ağırlıklardan başlatılarak, aynı sayıda saklı katman hücresi ve aynı öğrenme oranı ile bin iterasyonluk eğitimi gerçekleştirilmiştir.

Sonuç olarak, uyarlanabilir aktivasyon fonksiyonlarının DA motorunun hız kontrolünde kullanımı hesaplama açısından ek yük getirirken, standart sigmoid aktivasyon fonksiyonuna göre, hem hatayı minimize etme hızı, yani eğitim süresini azaltma açısından, hem de genelleme performansı açısından üstünlük sağlamışlardır. Bunun yanında üstel ve sigmoid aktivasyon fonksiyonu diğer uyarlanabilir aktivasyon fonksiyonlarına göre, eğitim süresini kısaltma ve genelleme açısından daha iyi bir performans göstermiştir. Spline aktivasyon fonksiyonunun performansı da oldukça iyi olmasına karşın genel performansı üstel ve sigmoid aktivasyon fonksiyonundan kötüdür. Bu nedenle, hesaplama yoğunluğunun önemsiz olduğu durumlarda uyarlanabilir aktivasyon fonksiyonları tercih edilebilir.

KAYNAKLAR

- [1] Haykin, S., 1994, Neural Networks, A Comprehensive Foundation, Macmillan College Publishing Comp. Inc.
- [2] Narendra, K.S., and Parthasarathy, K., 1990, Identification and Control of Dynamical Systems Using Neural Networks, IEEE Transactions on Neural Networks, Vol.1, No:1, 4-27.
- [3] Hunt, K.J., Sbarbaro, D., Zbikowski R., and et all, 1992, Neural Networks for Control Systems – A Survey, Automatica, Vol.28, No:6, 1083-1112.
- [4] Tamura S., and Tateishi, M., 1997, Capabilities of a Four Layered Feedforward Neural Networks: Four Layars Versus Three, IEEE Trensactions on Neural Networks, Vol.8, No:2, 251-255.
- [5] Sastry, P.S., Santharam G., and Unnikrishnan, K.p., 1994, Memory Neuron Networks for Identification and Control of Dynamical Systems, IEEE Transactions on Neural Networks, Vvol.5, No:2, 306-319.
- [6] Fukado, T., Shibata, T., Tokita M., and et all, 1992, Neuromorpnic Control: Adaptation and Learning, IEEE Transactions on Industrial Electronics, Vol.39, No:6, 497-503.
- [7] Chen S., and Billings, S.A., 1992, Neural Networks for Nonlinear Dynamic System Modeling and Identification, Int. J. Control, Vol.56, No:2, 319-346.
- [8] Chen S., and Billings S.A., and Grant, P.M., 1992, Recursive Hybrid Algorithm for Non-Linear System Identification Using Radial Basis Function Networks, Int. J. Control, Vol.55, No:5, 1050-1070.
- [9] Bates, J., Elbuluk, M.E., and Zinger, D.S., 1993, Neural Network Control of a Chopper – Fed DC Motor, 8. APEC Confrence, 893-899.
- [10] Weerasooriya, S., and El-Sharkavi, M.A., 1991, Identification and Control of a DC Motor Using Back – Propagation Neural Networks, IEEE Transactions on Energy Conversion, Vol.6, No:4, 663-669.
- [11] McCulloch, W.S., and Pitts, W., 1943, A Logical Calculus of the Ideas Imminent in Nervous Activity, Bull. Math. Biophys., Vol.5, 115-133.

- [12] Rumelhart, D.E., McClelland, J.L., and the PDP Research Group, 1986, Parallel Distributed Processing (PDP): Exploration in the Microstructure of Cognition Vol. 1. Cambridge, MA: MIT Press.
- [13] Chen C.T., and Chang W.D., 1996, A Feedforward Neural Network with Function Shape Autotuning, Neural Networks Vol.9, No:4, 627-641.
- [14] Piazza, F., Uncini, A., and Zenobi, M., 1992, Artificial Neural Networks with Adaptive Polynomial Activation Function,” in Proc. IJCNN, Beijing, China, II-343-349.
- [15] Piazza, F., Uncini, A., and Zenobi, M., 1993, Neural Networks with Digital LUT Activation Function, in Proc. IJCNN, Nagoya, Japan, II-1401-1404.
- [16] Guarnieri, S., Piazza, F., and Uncini, A., 1999, Multilayer Feedforward Networks with Adaptive Spline Activation Function, IEEE Transactions on Neural Networks, Vol.10, No:3, 672-683.
- [17] Gökbulut, M., 1998, Fırçasız Doğru Akım Motorlarının Yapay Sinir Ağları ile Uyarlamalı Denetimi, Erciyes Ü. Fen Bil. Enst. Doktora Tezi.
- [18] Tuncer, S., 1998, Değişken Hızlı Sürücü Sistemleri İçin Fuzzy Denetleyicili Yeni Bir Algoritmanın Geliştirilmesi ve Uygulaması F. Ü. Fen Bil. Enst. Yüksek Lisans Tezi.
- [19] Dubey, G.K., 1989, Power Semiconductor Controlled Drivers, Prentice Hall, New Jersey.
- [20] Kuo, B.C., 1991, Automatic Control Systems, Prentice-Hall, Inc., New Jersey.
- [21] Brown, M., Harris, C., 1994, Neuro-Fuzzy Adaptive Modelling and Control. Prentice Hall Inc., London.
- [22] Tanumaru, J., and Omatu, S., 1992, Process Control by On-Line Trained Neural Controllers, IEEE Trans. On Industrial Electronics, Vol. 39, No:6,511-521.
- [23] Narendra, K.S., and Parthasarathy, K., 1991, Gradient Methods for the Optimization of Dynamical Systems Containing Neural Networks, IEEE Trans. On Neural Networks, Vol. 2, No2,252-262.
- [24] Narendra, K.S., and Annaswamy, A.M., 1989, Stable Adaptive Systems, Prentice Hall Inc., New York.
- [25] Åström, K.J., and Wittenmark, B., 1995, Adaptive Control, Addison Wesley Publishing Company Inc., New York.

- [26] Khalil, H.K., 1992, *Nonlinear Systems*, Macmillan Publishing Company, New York.
- [27] Sjöberg, J., Zhang, Q., Ljung, L., and et al, 1995, Nonlinear Black-Box Modeling in System Identification a Unified Overview, *Automatica*, Vol. 31, No:12,1691-1724.
- [28] Nguyen, D.H., and Widrow, B., 1990, Neural Networks for Self-Learning Control Systems, *IEEE Control System Magazine*, 18-23.
- [29] Boskovic, I.D., and Narendra, K.S., 1995, Comparison of Linear, Nonlinear and Neural Network-Based Adaptive Controllers for a Class of Fed-Batch Fermentation Processes, *Automatica*, Vol. 31, NO:6,817-840.
- [30] Rovithakis, G.A., and Christodoulou, M.A., 1994, Adaptive Control of Unknown Plants Using Dynamical Neural Networks, *IEEE Trans. On Systems, Man and Cybernetics*, Vol. 24, No:3,400-411.
- [31] Zhang, Y., Sen, P., and Hearn, G.E., 1995, An On-Line Trained Adaptive Neural Controller, *IEEE Control Systems*, 67-75.
- [32] Hunt, K.J., and Sbarbaro, D., 1991, Neural Networks for Nonlinear Internal Model Control, *IEE Proceedings-D*, Vol. 138, No:5,431-438.
- [33] Hofer, D.S., Neumerkel D., and Hunt, K., 1993, Neural Control of a Steel Rolling Mill, *IEEE Control Systems*, 65-75.
- [34] Khalid, M., Omatu, S., and Yusof, R., 1995, Temperature Regulation With Neural Network and Alternative Control Shemes, *IEEE Trans. On Neural Networks*, Vol. 6, NO:3,572-582.
- [35] Chen, F.C., 1990, Back-Proragation Neural Networks for Nonlinear Self-Tunning Adaptive Control, *IEEE Control Systems Magazine*, 44-48.
- [36] Chen, F.C., and Khalil, H.K., 1992, Adaptive Control of Nonlinear Systems Using Neural Network, *Int. J. Control*, Vol. 55, No:6,1299-1317.
- [38] Chen, T., and Chen H., 1993, Approximation of Continuous Functionals by neral Networks With Application to Dynamic Systems, *IEEE Trans. On Neural Networks*, Vol. 4, 910-918.
- [39] Chen, T., Chen, H., and Liu, R., 1995, Approximation Capability in $C(R^n)$ by Multilayer Feedforward Networks and Related Problems, *IEEE Tans. Neural Networks*, Vol. 6, 25-30.

- [40] Hornik, K., Stinchcombe, M., and White, H., 1989, Multilayer Feedforward Networks Are Universal Approximators, *Neural Networks*. Vol. 2, 359-366.
- [41] Hornik, K., Stinchcombe, M., and White, H., 1990, Universal Approximation of an Unknown Mapping and Its Derivatives Using Multilayer Feedforward Networks, *Neural Networks*, Vol. 3, 551-560.
- [42] Stinchcombe, M., and White, H., 1989, Universal Approximation Using Feedforward Networks With Nonsigmoid Hidden Layer Activation Functions, in *Proc. IJCNN*, Washington, D.C., 613-617.
- [43] Stinchcombe, M., and White, H., 1990, Approximating and Learning Unknown Mappings Using Multilayer Feedforward Networks With Bounded Weights, in *Proc. IJCNN*, San Diego, CA, III 7-16.
- [44] El-Sharkawi, M.A., El-Samahy, A.A., El-Sayed, M.L., 1994, High Performance Drive Of DC Brushless Motor Using Neural Networks, *IEEE Transaction On Energy Conversion*, Vol:9, No:2, 317-322
- [45] Karna, K.N., David, M.B., 1989, An Artificial Neural Networks Tutorial Part:1- Basics., *Neural Networks*, Vol:1, No:1, 5-23.
- [46] Harvey, R. L., 1994, *Neural Network Principles*, Prentice-Hall, Inc., New Jersey.
- [47] Tang, C.Z., and Kwan, H.K., 1992, Convergence and Generalization Properties of Multilayer Feedforward Neural Networks, in *Proc. IEEE Int. Symp. Circuits Syst.*, San Diego, CA, I 65-68.
- [48] Cybenko, G., 1989, Approximation by Superposition of a Sigmoidal Function, *Math. Contr. Signals Syst.*, No:2.
- [49] Hwang, J.N., Lay, S.R, Maechler, M., Martin, R.D., and Schimert, J., 1994, Regression Modeling in Bacpropagation and Projection Pursuit Learning, *IEEE Trans. Neural Networks*, Vol. 5, 342-353.
- [50] Xu, S., Zhang, M., 2000, Justification of a Neuron-Adaptive Activation Function, *IJCNN Proceedings*, Como, Italy, III-465.
- [51] Akın, E., Gökbulut, M., 1997, Fırçasız Doğru Akım Motorlarının YSA ile Tanımlanması Ve Öğrenme Algoritmaları, F.Ü. Fen Bil. Ens., Araştırma Raporu
- [52] Koçak, T., Cılız, K., Kaynak, O., 1996, Yapay Sinir Ağları ile Gerçek Zamanda Denetim, Otomasyon Teknolojileri Sempozyumu TOK'96, 23-30.

- [53] Karahan, Ö., Halıcı, U., Leblebicioğlu, K., 1997, Yapay Sinir Ağı İle Bütünleştirilmiş Model Öngörümlü Denetim, Otomasyon Teknolojileri Sempozyumu TOK'97, 29-34.
- [54] Dandıl, B., 1998, Yapay Sinir Ağları Yardımıyla Sistem Kontrolü, F.Ü. Fen Bil. Enst. Yüksek Lisans Tezi.
- [55] Özmetel E., 1996, Bilgisayarda öğrenme ve yapay nöral ağları, Otomasyon. mart, sayı.45, 134-140.
- [56] Özmeterler, E., 1989, Yapay Nöral Ağları, Yüksek Lisans Tezi, İ.T.Ü. Fen Bilimleri Enstitüsü, İstanbul.
- [56] Zurada, M.J., 1992, Introduction to Artificial Neural Systems, West Publishing Company, Inc. New York.
- [57] Karlık, B., 1995, Yapay Sinir Ağları, Yüksek lisans ders notları.
- [58] Türkoğlu, İ., 1996, Yapay Sinir Ağları ile Nesne Tanıma, F.Ü. Fen Bilim. Enst. Yüksek Lisans Tezi.
- [59] Gedikpınar, M., Güldemir, H., 2001, Fırçasız Doğru Akım Motorunun modellenmesi ve Hız Kontrolü, Elektrik-Elektronik ve Bilgisayar Müh. 9. Ulusal Kongr., Kocaeli-Sakarya Üniv. (1) 142-145.
- [60] Gedikpınar, M., 2002, Fırçasız Doğru Akım Motorlarının Kaymo Mod Gözlemleyicili Algılayıcısız Hız Kontrolü, Gazi Üniversitesi Fen Bil. Enst. Doktora Tezi.
- [61] Güldemir, H., 1999, Prediction of Induction Motor Line Current Spectra from Design Data, Universty of Notingham, PhD Thesis, UK.