

T.C.
MARMARA ÜNİVERSİTESİ
BİLİMSEL ARAŞTIRMA PROJELERİ KOMİSYONU BAŞKANLIĞI
PROJE RAPORU

WEB TABANLI YAPAY ZEKA TEKNİKLERİ
EĞİTİM SİMÜLATÖRLERİNİN HAZIRLANMASI
(PROJE NO : FEN-E-050608-138)

Proje Yürütücüsü
Yrd. Doç. Dr. Mehmet TEKTAŞ

Yardımcı Araştırmacılar
Yrd. Doç .Dr. Necla TEKTAŞ
Yrd. Doç. Dr. Nevzat ONAT
Yrd. Doç. Dr. Gökhan GÖKMEN
Arş. Gör. Dr. Gökhan KOÇYİĞİT
Arş. Gör. Dr. Tahir Çetin AKINCI

İSTANBUL, 2010

1. BÖLÜM.....	7
1.1. Eğitim	7
1.1.1 Uzaktan Eğitim Kavramı	7
1.1.2 E-Öğrenme.....	9
1.1.3 Uzaktan Eğitimde Önemli Literatür Çalışmaları	11
1.2. Web Tabanlı Eğitim Kavramı.....	20
1.2.1 Web Tabanlı Eğitimin Avantajları	21
1.2.2 Web Tabanlı Eğitimin Dezavantajları	22
1.2.3 WTE Önündeki Engeller	23
1.3. Web Tabanlı Eğitimin Tasarlanması.....	24
1.4. Web Tabanlı Öğretimde Etkileşim	25
1.5. Türkiye’de Uzaktan Eğitim.....	26
1.6. Web Tabanlı Eğitim ve Sanal Laboratuvarlar.....	27
1.6.1 Sanal Laboratuvarlar	28
1.6.2 Sanal Laboratuvar Yapıları.....	28
1.6.3 Türkiye’de Sanal Laboratuvarların Geleceği	33
2. BÖLÜM.....	35
2.1. Yapay Zekâ	35
2.2. Yapay Zekâ Nedir?	35
2.2.1 İnsan gibi düşünen sistemler	36
2.2.2 İnsan gibi davranan sistemler	37
2.2.3 Rasyonel düşünen sistemler	37
2.2.4 Rasyonel davranan sistemler	37

2.3. Yapay Zekânın Amaçları	38
2.4. Yapay Zekânın Tarihçesi	38
2.5. Yapay Zekânın Alt Alanları.....	40
2.5.1 Yapay Zekâda Önemli Teknikler	41
2.6. Yapay Zekânın Kullanım Alanları Ve Kullanılan Teknikler.....	47
2.7. Endüstride Yapay Zekâ	48
2.7.1 Otomotivde Yapay Zekâ Teknolojileri.....	48
2.7.2 Kullanım Amaçlarına Göre Yapay Zekâ Teknolojileri	49
2.7.3 Askeri Alanda Yapay Zekâ Uygulamaları.....	56
2.7.4 Tıp Ve Sağlık Alanında Yapay Zekâ Uygulamaları	58
2.7.5 Akıllı Ev Teknolojileri ve Akıllı Ev Aletlerinde Yapay Zekâ Uygulamaları	61
2.8. Eğitimde Yapay Zekâ.....	64
2.9. Teknoloji Eğitiminde Yapay Zekâ	67
2.9.1 Robotik.....	70
2.10. Yapay Zekânın Ticari Uygulamaları.....	73
3. BÖLÜM : Bulanık Mantık (Fuzzy Logic)	76
3.1. Giriş	76
3.1.1 Bulanık Mantık ‘ın Tarihçesi ve Kurucusu Lotfi Zadeh	77
3.2. Bulanık Kümeler.....	78
3.2.1 Bulanık Kümelerde İşlemler.....	80
3.2.2 Üyelik İşlemleri.....	81
3.3. Bulanık Mantığın Kullanım Alanları ve Günümüzde Uygulama Örnekleri	85
3.4. Proje Kapsamında Hazırlanan Bulanık Mantık Simülatörünün Bölümleri	88

4. BÖLÜM: Yapay Sinir Ağları	91
4.1. Genel Tanım.....	91
4.2. YSA'ların Genel Özellikleri.....	91
4.3. YSA'ların Avantaj ve Dezavantajları	92
4.4. YSA'ların Kullanıldığı Alanlar	95
4.5. YSA'ların Tarihçesi	97
4.6. YSA'nın Yapısı, Elemanları ve Mimarisi.....	99
4.7. Yapay Nöron	101
4.8. YSA Çeşitleri	116
4.9. Sinir Sistemi ve Ysa'nın Benzerlikleri.....	119
4.10. Yapay Sinir Ağlarının Genel Özellikleri.....	119
4.11. YSA'nın Avantajları	119
4.12. YSA'nın Dezavantajları.....	120
4.13. Ysa'ların Sınıflandırılması.....	120
4.13.1 Mimari Yapılarına Göre.....	120
4.13.2 Öğrenme Yaklaşımlarına Göre.....	120
4.14. Öğrenme Kuralları	123
4.15. Arakat Bağlantılarına Göre Sınıflandırılması	124
4.15.1 Katlar Arası Bağlantılar	124
4.15.2 Diğer Katlarla Bağlantılar.....	124
4.16. Yaygın Olarak Kullanılan Öğrenme Algoritması.....	124
4.16.1 Geri Yayılım Algoritması	125
4.16.2 Delta-Bar-Delta	126
4.16.3 Multi-Layer Perception (MLP)	127

4.16.4	Learning Vector Quantisation Ağ	128
4.16.5	Hopfield Ağ	129
4.16.6	Elman Ağ	129
4.16.7	Kohonem Ağ	129
5.	BÖLÜM : Genetik Algoritma (Genetic Algorithm)	131
5.1.	Genetic Algoritmalar ve Evrimsel Hesaplama	133
5.2.	Rasgele Arama Algoritması	136
5.3.	İkili kodlu genetik algoritmalar	136
5.3.1	İkili kodlu genetik algoritmaların operatörleri	138
5.4.	Amaç fonksiyonu ve parametreler	140
5.4.1	Parametrelerin tanımlanması ve kodlama	141
5.4.2	Başlangıç Popülasyonu	145
5.4.3	Doğal seçim	146
5.4.4	Eşleştirme	148
5.4.5	Çaprazlama	151
5.4.6	Mutasyon	152
5.4.7	Gelecek nesil	153
5.4.8	Yakınsama	156
5.5.	Gerçek kodlu genetik algoritma	158
5.5.1	Amaç fonksiyonu ve parametreler	159
5.5.2	Başlangıç Popülasyonu	160
5.5.3	Doğal seçim	162
5.5.4	Eşleştirme	163
5.5.5	Çaprazlama	163

5.6. Mutasyon.....	166
5.7. Örnekler.....	170
5.7.1 Bir Açıklayıcı Örnek – GA Nasıl Çalışır.....	170
5.7.2 Bir GAnın Nasıl Çalıştığını Gösteren Diğer Bir Örnek.....	173
5.7.3 İki Değişken İçeren Bir Başka Örnek.....	176
5.7.4 İki değişken içeren diğer bir örnek	178
5.7.5 Gezgin Satıcı Probleminin Genetik Algoritmalarla Uygulaması....	192
5.8. Sonuç	195
5.9. Proje Kapsamında Gerçekleştirilen Genetik Algoritma Simülatörü	196
5.9.1 Temel Özellikler.....	196
5.9.2 Gereksinimler	196
5.9.3 Menüler.....	197
5.9.4 Temel Kavramlar	197
5.9.5 Simülasyonlar	198
6. Kaynaklar	200

1. BÖLÜM

1.1. Eğitim

Bilginin güçle eşdeğer görüldüğü günümüz bilgi toplumlarında eğitim; bilgi teknolojilerini rahatlıkla kullanan, bilgiyi üreten, sınıflandıran, sunan ve paylaşan bireyler yetiştirmeyi amaçlamaktadır. Bu belirtilen durumların gerçekleşmesi; öğrenme-öğretme süresi boyunca bireyin ön plana çıkarılması, öğretme ve öğrenmenin öğrenci merkezli olması, bu süre boyunca uygulanacak tekniklerin çağdaş bir anlayışla zamanın gerekliliklerine uygun biçimde tasarlanması, uygulanması, değerlendirilmesi ve geliştirilmesi ile mümkün olacaktır. Belirtilen gereklilikleri karşılayabilecek olan eğitim teknolojisi; öğretme-öğrenme süreçlerini etkili kılarak öğrenmenin kolay, somut, zengin, anlamlı, güdüleyici, teşvik edici, verimli ve kaliteli etkinliklere dönüştürülmesi için insan gücü ve onun dışındaki kaynakların amaca yönelik olarak uygulanmaya konulmasını içermektedir [1].

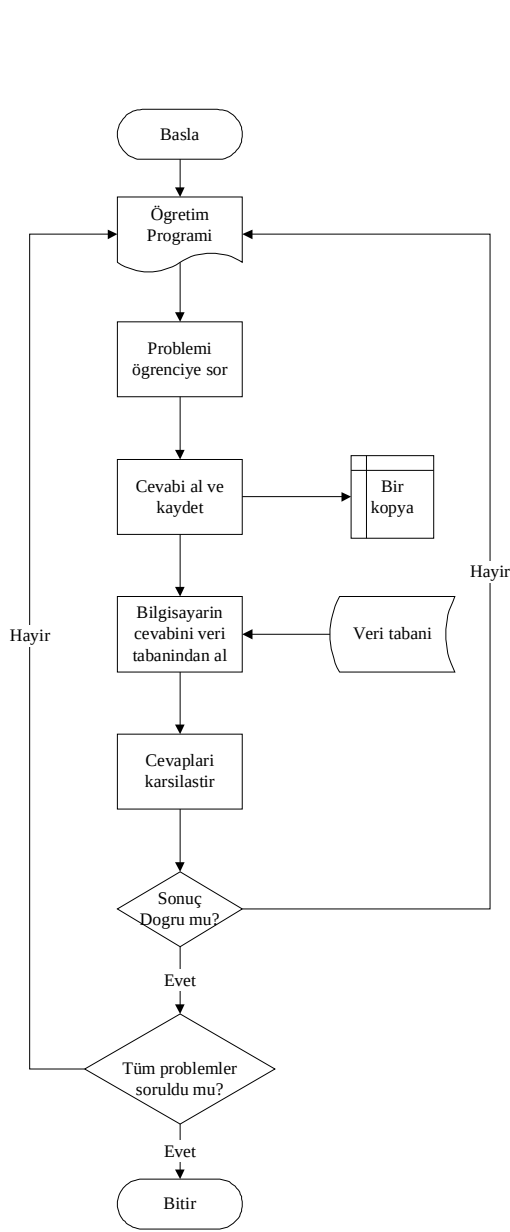
1.1.1 Uzaktan Eğitim Kavramı

Uzaktan eğitim, bireylere kendi kendilerine öğrenme imkanının sağlandığı, geleneksel eğitime göre daha esnek ve birey koşullarına uyarlanabilir bir eğitimidir. Uzaktan eğitimle eğitim hizmeti götürmekteki sınırlılıkların kısmen yada tümüyle ortadan kaldırılarak eğitim imkanlarının daha geniş kitlelere ulaştırılması amaçlanmaktadır. Çok ortamlı araçların ve sunu sistemlerinin işe koşulması, uzaktan eğitim tanımının yapılmasını güçleştirmektedir. Ancak, kısaca öğretmen ve öğrencinin zaman ve mekan bakımından birbirinden ayrıldığı ortamlar üzerine yapılandırılan eğitim uygulamalarının hepsi uzaktan eğitim olarak adlandırılmaktadır [2]. ABD Uzaktan Eğitim Derneği, uzaktan eğitimi “bilgi teknolojileri ve altyapılarının öğrenciler arası ve öğrenci öğretmen arasındaki aktiviteleri ve eğitimsel faaliyetleri sağlamak amacıyla kullanılmasıdır” olarak tanımlamaktadır. Bir başka tanıma göre uzaktan eğitim, uydu, video, ses, grafik, bilgisayar, çoklu ortam teknolojisi gibi araçların yardımıyla, eğitimin uzaktaki öğrencilere ulaştırılmasıdır [3].

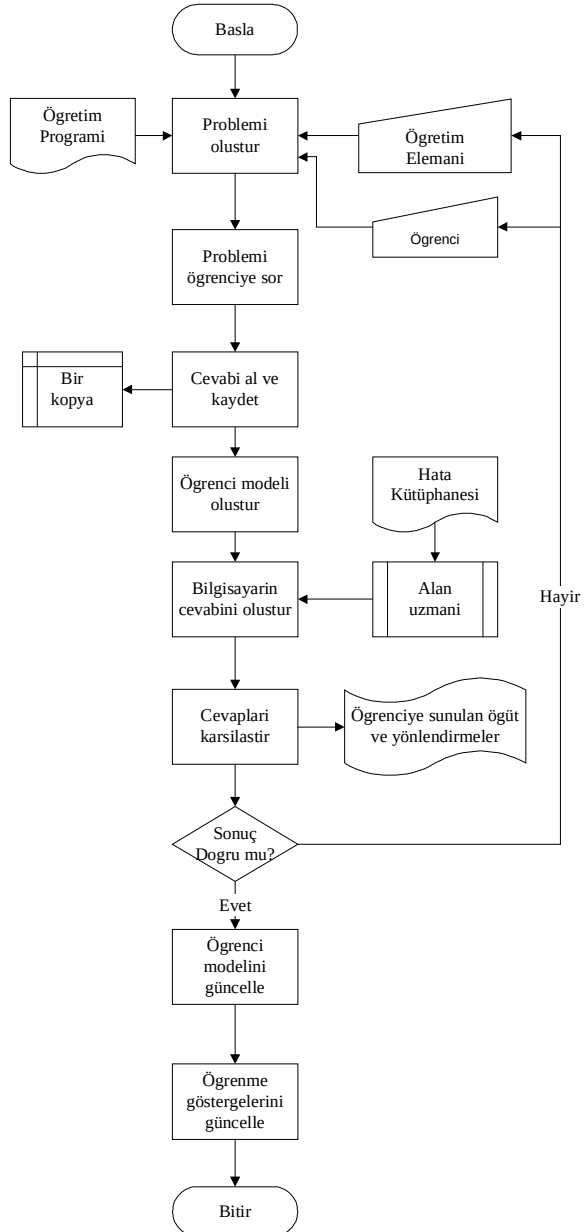
Bilgisayar ve internet teknolojilerindeki hızlı gelişmeler öğretme ve öğrenme yöntemlerini de çok hızlı bir şekilde değiştirmektedir. Bu gelişmeler çeşitli yollarla yapıla

gelen uzaktan eğitim modellerinin web tabanlı eğitime (WTE) kayması sonucunu doğurmuştur. WTE ise, uzaktan eğitimi desteklemek için, Web üzerinden verilmek üzere hazırlanmış, zamandan ve yerden (mekândan) bağımsız olarak erişim olanakları sunan, erişimin bir ağ üzerinden (internet ya da intranet olarak) yapıldığı eğitim şekline denilmektedir [4]. Bu tanımdan da anlaşılacağı üzere WTE'nin temel amacı uzaktan eğitim yapabilmektir.

WTE'de kullanılan yöntemler arasında eğitici bilgisayar yazılımları ve simülasyonlar olmak üzere iki temel yaklaşım ortaya çıkmaktadır. Eğitici bilgisayar yazılımları genelde konuların tekrarlanması ve bu konular hakkında çok sayıda problemin çözülmesi esasına dayanmaktadır. Bu tip sistemlerde özellikle sürekli tekrarlar sistemin verimliliğini ve öğrencilerin dikkatlerini olumsuz etkilemektedir. Simülasyon yazılımları ise öğrencilerin kendi oluşturdukları problemleri çözebilme, deneyler yapabilme gibi daha doyurucu bir etkileşim ortamı sunmaktadır. Bu sistemlerin en önemli dezavantajı, tasarım ve hazırlama aşamalarının daha zor ve uzun bir süreç olmasıdır. Simülasyon yazılımlarında çoğu zaman yapay zeka tekniklerinin en az bir tanesinin kullanılması zorunludur. Bir çok sistemde ise bir kaç tekniğin bir arada kullanıldığı melez (hybrid) yapılar bulunmaktadır. Literatürde, WTE kavramı bilgisayar destekli eğitim (*Computer Aided Instruction - CAI*), yapay zeka destekli simülasyona dayalı WTE sistemlerine de zeki bilgisayar destekli öğretim sistemleri (*Intelligent Computer Aided Instruction - ICAI*) denilmektedir [5], [6].



a-Geleneksel bilgisayar destekli eğitim modeli örneği



b-Zeki bilgisayar destekli eğitim modeli örneği

Şekil 1 Uzaktan eğitimde WTE modelleri.

1.1.2 E-Öğrenme

Uzaktan eğitimin bir alt kümesi olan e-öğrenmenin çeşitli kaynaklarda farklı tanımları mevcuttur. Bu tanımlardan bazıları şunlardır: İnternet veya bir bilgisayar ağı bulunan platform üzerinde sunulan, öğrenci ile öğretim elemanının birbirlerinden fiziksel olarak ayrı olmalarına rağmen, eş zamanlı (senkron) veya ayrı zamanlı (asenkron) çoklu

ortam teknolojisi yardımıyla iletişim kurdukları, öğrenme hızına göre öğrenmenin gerçekleştirildiği öğretim sürecidir [7]. E-öğrenme, internet teknolojileri aracılığıyla, öğretmen ve öğrencinin aynı ortamda ve aynı anda bulunmalarına gerek kalmadan gerçekleştirilen eğitim faaliyetleridir [8]. E-öğrenme, eğitim içeriğinin, internet, intranet, uydu yayını, etkileşimli TV, görsel/işitsel teyp, CD-ROM, vb. yoluyla elektronik ortamda aktarımıdır[8]. E-öğrenme, internet, bir ağ veya sadece bilgisayar yoluyla gerçekleşen öğrenmelerdir[9]. Internet, intranet veya bir bilgisayar ağı bulunan platform üzerinde sunulan, web tabanlı bir eğitim sistemidir [10]. Bu tanımlar ışığında e-öğrenme, yaşam boyu öğrenmeyi destekleyen, farklı niteliklere sahip fakat aynı hedefe yönelmiş bireyleri bir araya getiren, çeşitli bilgisayar ve iletişim teknolojilerini eğitimin hizmetine sunabilen, öğrenci merkezli, özgün kurumsal ve yönetsel yapılanmayı gerektiren planlanmış öğrenme olarak ifade edilebilir [11].

E-öğrenme karmaşık psikolojik bir olgudur. Geleneksel eğitim sistemlerinde bilgi kayıtlı olarak vardır ve konuşma vasıtasıyla (ders vererek) veya kağıtlarla (kitap okuyarak) öğrenciye aktarılır. Ancak e-öğrenme sistemiyle bilgi sunulur, depolanır ve elektronik araçlarla gönderilir (dersin web-tabanlı olarak işlenmesi gibi). Sonuç olarak, e-öğrenme bilginin, çok yönlü tiplerini (sözlü ve resimli), çok yönlü işlem modlarını (işitsel ve görsel) ve çok yönlü öğrenme durumlarını (ard arda ve aynı anda) içermesiyle son derece karmaşık hale gelir [12]

E-öğrenme, yaygınlaşmaya başladığı ilk yıllarda çevrimiçi kitaplar oluşturma olarak algılansa da, son zamanlarda e-öğrenmeden beklenenler artmıştır. İçeriklerin internette çevrimiçi kitap formatında sunulmasına nazaran e-öğrenmenin en belirgin özelliği içerik sunumunda kullanılan etkileşimlerdir. E-öğrenmede etkileşim konusu üzerinde yapılan araştırmalar incelendiğinde etkileşimin farklı türleri olduğu görülmüştür. E-öğrenme ile ilişkili olan üç farklı etkileşim vardır. Bunlar: öğrenci-içerik, öğrenci-öğretmen ve öğrenci-öğrenci etkileşimidir [11], [12]. Bununla birlikte bu etkileşim türlerine ek olarak, öğrenenin teknoloji ile içerik sunumu esnasında olan etkileşimini de ekleyerek öğrenci-ara yüz adında bir dördüncü etkileşim türü ortaya çıkmıştır. Öğrenci-içerik etkileşimi denilince akla ilk gelen, öğrencinin içerikle karşılıklı iletişim kurmasıdır. Bu iletişim sayesinde, öğrenci, kendisine kazandırılmak istenen bilgi ve becerilere bilgisayar ortamındaki bireysel çalışma ve çabası ile sahip olur. Bu çalışma sürecinde öğrenci içeriği okur, izler, uygulama yapar, soru sorar ve aldığı cevaplara göre

hatalarını, eksiklerini görür. Böyle bir yapının gerçekleşmesi, birçok fonksiyonun bir arada kullanımı ile mümkün olabilmektedir. E-öğrenme sistemlerindeki öğrenci-içerik etkileşimi için kullanılan fonksiyonlar şöyle sıranabilir:

- Sık sorulan sorular (SSS),
- ilgili eğitsel siteler bağlantılar,
- ilgili öğrenme materyallerine bağlantılar,
- multimedya sunumu (yazı, grafikler, animasyon ve sesler),
- sistem üzerinde kullanıcı rehberliği,
- online sınav veya kişisel değerlendirme,
- medyanın ileri seviyede kullanımı,
- içerikle ilgili online yardım,
- öğrenilen materyal üzerinde öğrenene destek, bireyselleştirilmiş öğrenme veritabanı,
- bireyselleştirilmiş eğitim, çalışma rehberliği,
- şakalar, çekiliş ve eğitsel oyunlar.

Bu fonksiyonlar sayesinde öğrenci, içerik ile iletişim kurabilir ve karşısında eğitmen varmış gibi davranışlar sergileyebilir. Özellikle e-öğrenmenin yapısı gereği, içeriğin öğrenciyi adapte etmesi, çalışmaya teşvik etmesi, bireysel aktivitelerle öğrenmeyi gerçekleştirmesi, başarı için en önemli etkenlerin başında gelmektedir. Bu nedenle öğrenci-içerik etkileşimi, e-öğrenme sistemi yönetici ve eğitmenlerin üzerinde hassasiyetle durması gereken ciddi bir konudur [11].

1.1.3 Uzaktan Eğitimde Önemli Literatür Çalışmaları

Bu bölümde kapsamlı araştırma sonuçlarını içeren iki önemli çalışmanın kısa özetleri verilmiştir.

1.1.3.1 Online Derslerde Kullanılan Teknolojilerin Etkinliği

2005 yılında ABD’de yapılan kapsamlı bir çalışmada web tabanlı MBA derslerinde kullanılan etkileşim teknikleri ile öğrenci ve öğretmenlerin bu tekniklere bakış açıları incelenmiştir. Yapılan çalışmanın ayrıntıları aşağıdaki tablolarda özetlenmiştir.

Tablo 1 Araştırmada kullanılan soru ve teknikleri [13].

Araştırma Soruları	Metodlar	Veri kaynağı
1- Hangi eğitimsel aktivite ve teknolojiler online ders etkileşimin sağlamak için kullanılmaktadır	Online derslerin içerik analizi Alan araştırması	27 online ders 102 öğrenci
2 -Öğrenci ve öğretmenler online ders etkileşimlerini nasıl algılamaktadırlar?	Öğrenci ve öğretmen röportajları	26 öğrenci, 10 öğrenci ile bireysel röportajlar ve 2 öğrenci grubu röportajı
3 - Bu algılama hangi noktalarda farklılık göstermektedir.	Alan araştırması	102 öğrenci

Tablo 2 Çalışmada dikkate alınan MBA derslerinde kullanılan eğitim teknolojileri ve kullanım oranları [13].

Teknolojiler	Kullanan	Kullanmayan	Kullanım Oranı
Ders Kitabı (Yazılı metin)	27	0	100%
E-posta	26	1	96%
Ders kitabına dayalı iki yönlü iletişim ve tartışma	25	2	93%
Asenkron (tartışma forumları v.b.) iletişim	23	4	85%
Senkron (sohbet v.b.) iletişim	11	16	41%
Etkileşimli ara sınav yöntemleri	18	9	67%
PowerPoint Slaytları	15	12	56%
Web Sayfaları	13	14	48%
Ses ve Görüntü klipleri	12	15	44%
Telefon	8	18	30%
Ses ve görüntü tabanlı iki yönlü iletişim (sesli posta, anlık mesajlaşma, video konferans v.b.)	0	27	0%

Tablo 3 Çalışmada dikkate alınan MBA derslerinde kullanılan öğretim aktiviteleri ve kullanım oranları [13].

Öğretim aktiviteleri	Kullanan	Kullanmayan	Kullanım Oranı
Öğretmen sorularına soru-cevap	27	0	100%
Görevler açısından geribildirim	27	0	100%
Dersin (sınıfın) ana konularının veya en önemli noktalarının özetlenmesi	26	1	96%
Sınıf tartışmalarına öğretmen katılımı	25	2	93%
Takım tabanlı öğrenme aktiviteleri	22	5	81%
Online tartışmaların değerlendirilmesinin bir parçası olarak katılım	18	9	67%
Küçük grup tartışmaları	11	16	41%
Grup tartışmalarına öğretmen katılımı	1	26	4%
Sanal ofis Saatleri	3	24	11%
Grup içi geribildirim/kritik yapma	4	23	15%
Karşılaştırmalı (dikkatli) değerlendirme	5	22	19%
Öğrencilerle online sohbet (coffee house)	2	25	7%
Öğrencilerin başlattığı forumlar	2	25	7%
Öğrencilerin beklentilerini ifade edebildikleri bülten panosu	4	23	15%
Haber hattı (Newslane)	2	25	7%

Tablo 4 Yapılan araştırmalar sonucunda öğrenci ve öğretmen algılamalarının özeti [13]

	Öğretmenler	Öğrenciler
Online Etkileşimler	Öğretmenler online etkileşimi öğretme ve öğrenmenin başarısı için önemli bir faktör olarak algılamaktadırlar.	Öğrenciler online etkileşimin önemi konusunda farklı algılamalara sahiptirler.
Yeterince online etkileşimin olmama nedenleri	Online etkilişimin sağlanması konusunda deneyim ve bilgi eksikliği	Öğrenme metodlarındaki farklılıklar veya kişisel nedenlerle tüm öğrencilerin fazla online etkileşim istememesi
	Öğretmenler arasındaki fikir farklılıkları ve verilen önemdeki eksiklikler	Online öğrenmede online etkileşimden beklentinin az olması

Tablo 4’de görüldüğü gibi online ders veren öğretmenler etkileşimin başarılı öğrenmenin önemli bir parçası olduğu konusunda genellikle hemfikirdirler. Genellikle ders etkileşimlerini sağlamak için çok sayıda teknoloji ve öğretim aktiviteleri sunulmaktadır. Ancak öğretmenler arasında bu teknolojilerin veya aktivitelerin kullanımı

noktasında, bu teknoloji ve aktivitelerin özel teknik uzmanlık gerektirmesi veya bunların verimli şekilde nasıl kullanılacağıının öğrenilmesinin önemli bir zaman gerektirmesi nedeniyle farklar oluşmaktadır. Ayrıca öğretmenler derslerini kendi istedikleri gibi etkileşimli yapmanın zor olduğunu kabul etmektedirler. Etkileşimli online ders sayısında istenilen düzeye ulaşmanın önündeki engel, sadece fakülte sınırlamaları veya teknik uzmanlık ya da zaman eksikliği değil, aynı zamanda yıllarca sürdürülen yüz yüze eğitim düzenlemelerin kafalarda oluşturduğu düşünce kalıpları ve alışkanlıklardır.

Öğrenciler ise kendi derslerinde online etkileşim konusunda farklı tercihlere sahiptirler. Kişisel nedenler veya farklı öğrenme metodları bu tercihi etkilemektedir. Bazı öğrenciler ise online etkileşimin çok fazla gerekli olmadığını, fazla bir şey kazandırmayacağını düşünmektedirler. Buna rağmen öğretmenler verimli öğretim stratejilerini araştırmalı ve etkileşimli online öğrenimin önündeki engelleri kaldırma yönünde çalışmalar yapmalıdırlar.

Bu çalışma sadece MBA dersleri için yapılmıştır. Daha fazla veri toplanabilmesi için çalışmaların sayısının artması ve aynı zamanda farklı alanlardaki online derslerin incelenmesi gereklidir [13].

1.1.3.2 WTE'in Uluslararası Analizi ve Gelecek için Tavsiyeler

2000 yılında tamamlanan ve AB projeleri kapsamında desteklenen Web Tabanlı Eğitimin Uluslararası Analizi ve Online Eğitimin Gelecekteki Gelişimi için Stratejik Tavsiyeler başlıklı bir sonuç raporu hazırlanmıştır. Bu raporda online eğitime verilen önemin küresel, kurumsal ve yönetsel sonuçları tartışılmaktadır. Bu rapor reklam ve mali konularda ilave bilgiler içermekte, akreditasyon, değerlendirme, kayıt ve gelişme esenliği ile birlikte pedagojik konuları da analiz etmektedir. Online eğitimin gelecekteki gelişimi ve bu gelişimin önündeki engeller de ele alınmaktadır. Sonuç olarak, politikacılar, eğitim yöneticileri ve online eğiticiler için stratejik tavsiyeler sunulmuştur.

Bu çalışma Internet üzerindeki derslerin uluslararası analizini sunmakta ve online eğitimin önemine yönelik stratejik tavsiyelerde bulunmaktadır. Sonuç ve tartışmalar literatür incelemelerine, 26 ülkede 130 kuruluş tarafından sağlanan catalog girişlerine ve bu kurumlarda söz sahibi 72 kişiyle yapılan röportajlara dayanmaktadır. Katalog verileri Mart98-Şubat99 arası toplanmış ve röportajlar 1999 baharında gerçekleştirilmiştir.

Web üzerinde sunulan derslerin durumunu profesyonel bir gözle analiz etmek, dengeli değerlendirme ve AB’de mesleki ve teknik eğitim konusuna yeni bir bakış açısı getirecek uzun vadeli tavsiyeler sağlamaktır [14].

1.1.3.3 Kurumsal Yayınlar

Çalışmada temas kurulan 130 kuruluşun önemli bir kısmı sadece birkaç web dersi sunmaktadır. Kuruluşların %23.1’i sadece bir web dersi, %46.2’si ise 5 dersten daha az web tabanlı ders sunmaktadır. Sadece dört kuruluş 100 veya daha üzeri ders vermektedir. Buradan aktivitelerin çoğunun deneme açmaçlı ve kuruluş için ana eksen olmadığı sonucu çıkarılabilir. Anket ayrıca kuruluşların %29.3’ünün 100 veya daha az öğrenciye sahip olduğunu göstermektedir. Sadece dört kuruluş 5000’den fazla öğrencisi olduğunu rapor etmiştir. Bu rakamlar nispi olarak oldukça düşüktür ve web tabanlı eğitimin kurumlar açısından temel eksen olarak kabul edilmediği anlamına gelmektedir.

Çıkan bir diğer sonuç, web derslerinde bilgisayar ve bilgi teknolojileri alanlarının eğitim amaçlı ders açısından üstünlük sağladıklarıdır. Bu iki alanın haricinde web dersleri çok geniş bir konu aralığını kapsamaktadır. Konu ve ders alanları kurumdan kuruma oldukça değişmektedir. Yaklaşık olarak kurumların yarısı sadece bir kategoride dersler sunmaktadır. Sadece beş kuruluş beş ve üzeri kategoride derslere sahiptir.

Ankete katılan ve incelenen kuruluşların hiç biri gerçek veya online üniversite olarak karakterize edilebilecek düzeyde yeterli online kurs ve destek servisi sağlamamaktadırlar. Anket 130 kuruluşun %60’ının üniversite ve yüksekokullara, %10’unun geleneksel açık öğretim üniversiteleri ve uzaktan eğitim sektörüne ve %9.2’sinin de limitet veya anonim şirketlere ait olduğunu göstermektedir.

1.1.3.4 Yönetimsel Yayınlar

Bir yönetim sistemi öğrenciler, öğretmenler, dersler ve ders materyallerini idare etmeyi gerektirmektedir. Online bir yüksekokul, binlerce öğrenci, yüzlerce öğretmen, şifre korumalı web sayfaları içeren çok sayıda ders, tartışma forumları, dağıtım listeleri, sınıf listeleri ve öğrenci sunumlarına sahip olabilir. Ayrıca aynı zamanda, ders kitaplarının gönderilmesi, okul ve sınav ücretlerinin kayıtlarının tutulması ve yerel

sınavların organizasyonunun yapılmasını da sağlamak zorundadır. Tüm bu servisler bir çok geleneksel kuruluş için oldukça büyük bir külfet oluşturmaktadır.

Büyük ölçekli ve profesyonel online eğitim sunmayı planlayan kuruluşlar web ile entegre olmuş bir yönetim sistemine ihtiyaç duymaktadır. Bunu başarmak için muhtelif stratejiler takip edebilirler. En basit yöntem hali hazırda çalışan bir yönetim sistemine sahip bir kuruluşla işbirliği yapmak olabilir. Daha fazla teknik beceri gerektiren bir başka çözüm, genel internet servislerine dayalı bir ev-içi sistemi geliştirmektir. Üçüncü seçenek ise online eğitim için geliştirilmiş standart bir sistemi satın almaktır. Bu standart sistemler sürekli olarak geliştirilmektedir. Fakat, hala yerel şartlara göre adaptasyon gerektirmektedirler. Sadece bazı yönetsel adaptasyon ihtiyaçları olabilir veya derslerde bazı pedagojik sınırlamalar yapılması gerekebilir.

Hayal kırıklığı yaratan fakat önemli bir tespit de bir çok kuruluşun web ortamını yönetsel amaçlar için kullanmadığıdır. Dış kaynak bir seçenektir ve çok kullanılmadığı tespit edilmiştir. Sadece bir örnek tanımlanabilmiştir. Röportajlarda bahsedilen standart yönetim sistemleri FirstClass, WebCT, ve Lotus Notes olmuştur..

1.1.3.5 Reklam ve Finansal Yayınlar

Röportajların bir çoğunda web hizmetlerinin önemli bir fonksiyonu program ve derslerin reklamlarının yapılması olarak ön plana çıkmıştır.

Web derslerinin öğrenim ücretleri kuruluşlar ve dersler arasında önemli farklar içermektedir. Bazı dersler ücretsiz ve herkese açık iken, bazıları tamamen veya kısmen dış desteklere sahiptirler. Öğrenim ücreti olarak çalışan kuruluşlarda ücretler geleneksel ders ücretleri ile aynı veya çok farklı değildir. Bu çalışmada öğrenci ödemelerinden sağlam gelir kazanan birkaç kuruluş örneği ortaya çıkmıştır. Aynı şekilde eğer dış kuruluşlar tarafından sağlanan araştırma geliştirme bursları ihmal edilirse, web tabanlı ders sunumunun ekonomik olarak başarılı olabileceğine inanan birkaç kuruluş ortaya çıkmıştır [14].

1.1.3.6 Pedagojik Sonuçlar

Eğiticiler sadece online kurslarla meşgul olan part-time öğretmenler ve norma görevlerinin yanı sıra bazı online dersler veren tam zamanlı öğretmenler olarak iki gruba

ayrılmaktadır. Öğretme faaliyetleri içinde öğrenci ve uzmanların dağılımlarının da bir bölüm oluşturduğunu görmek ilginç olmuştur.

Röportajlardan elde edilen analiz sonucu öğretmenlerin en azından aşağıdaki fonksiyonları gerçekleştirdiğini göstermektedir.

Organizasyon fonksiyonları : yapı tartışmalarında adımları atan ve ilk olarak başlatan kişi

Sosyal fonksiyonlar : Takip etme grupları

Entelektüel fonksiyonlar: soru cevaplama, öğrenciler internette klavuzluk etme.

Değerlendirme fonksiyonları (Assessment functions): geri besleme verme görevleri, önerileri düzeltme

İnsan öğretimi makine öğretime göre daha yaygın bir uygulamadır. Fakat bazı kuruluşlar insan eğitime ilave olarak makine eğitimi de içermektedir. Çoğu kuruluş bireysel eğitimlerle grup eğitimlerini birleştirme eğilimindedir. Bu ikisi arasındaki odaklanma miktarı da değişmektedir. Online öğretim bir çok derste yüz yüze toplantıları, video veya sesli konferans veya telefon konuşmalarını tamamlayıcı unsurdur.

Bazı kuruluşlar ders geliştirme ekiplerine sahiptirler. Diğerleri ise öğretmeni bir dersin tek tasarımcısı olarak kullanmaktadır. Farklı uygulamalar kalite kontrol ve geliştirme zamanı açısından farklı sorunlar içermektedir [14].

1.1.3.7 Akreditasyon

Bu röportajlar sonunda online ders ve program akreditasyonlarının kurumların geleneksel akreditasyon işlemleriyle çok benzer olduğu görülmüştür. Tamamlanan ders ve programları tanımlamada, dereceler, diplomalar, sertifikalar ve raporlar verme uygulaması çok büyük ölçüde kullanılmaktadır. Akreditasyon işleminin kurs veya programın online olması ya da olmamasından bağımsız olduğu görülmektedir. Bir çok kuruluş temelde akreditasyon işleminin kampus programları için izlenen yöntemle aynı olduğunu ifade etmektedir.

Akreditasyon önemli bir rekabet avantajı sağlamaktadır. Gerekli akreditasyonu sağlamak için muhtelif yöntemler takip edilebilir. Diğer ülkelerdeki kurumlarla işbirliği sonucunda iki taraflı akreditasyon işlemleri gerçekleştirilebilir.

1.1.3.8 Değerlendirme

Online derslerin toplam değerlendirmesinde çok geleneksel ve sıklıkla yüz yüze eğitim aktivitesi içeren bölümler içerdiği görülmekle birlikte, Oluşuma ya da gelişimle ilgili format ıve değerlendirme daha deneysel ve online aktivitelere dayalı olarak gerçekleştirilmektedir.

Kurumların çoğu bir ders veya program içinde çeşitli değerlendirme metotları uygulamaktadır. Öğretici hocaların değerlendirmesi en genel değerlendirme biçimi olarak ön plana çıkmaktadır. Kendi kendine değerlendirme örnekleri ile de bir çok yerde karşılaşlmıştır. Fakat bilgisayar değerlendirmesi yöntemi yok denecek kadar az kullanılmaktadır. Ancak, muhtelif sayıda online yapılan kısa sınavlar, çoktan seçmeli testler ve enteraktif alıştırma örnekleri mevcuttur. Çok dikkatli değerlendirmeler çok azdır. Bazı kurslarda hiç değerlendirme yapılmadığı görülmektedir. Bunun nedeni hoca içermeyen kendi kendine çalışmaya dayalı kurslar olmasıdır [14].

1.1.3.9 Kayıt ve İlerleme Esnekliği

Kayıt ve ilerlemenin her ikisi de çok veya az esnek olabilir. Ancak, grup halinde veya bireysel kayıt ve ilerleme modeli olmak üzere iki temel yaklaşım söz konusudur. Bu iki model, pazarlama, yönetim sistemleri ve pedagojik yaklaşımlar açısından önemli sonuçlar içeren iki farklı stratejiyi temsil etmektedir.

Çalışma sonucunda grup halinde kayıt ve ilerleme modelinin bireysel modele nazaran daha fazla kullanıldığını göstermektedir. 46 kurum grup modelini, 12 kurum bireysel modeli ve 11 kurum her iki modeli aynı anda kullanmaktadır.

Grup modelinin bu kadar baskın olmasının sebebi geleneksel eğitim sisteminde yer alan sömestr mantığına dayanan klasik düşünce sisteminin hala devam etmesidir. Diğer bir olası sebep, kurumlar tarafından çok dikkatli bir şekilde gözlemlenen bireysel kayıt ve ilerleme yönteminde takım çalışması ve işbirlikçi öğrenim imkanlarını sağlamanın çok zor olmasıdır. Ancak bir diğer sebep de birçok öğrencinin bireysel

esnekliđi tercih edeceđi göz önüne alınarak, kurumların sistem olarak ve yapısal eksiklikleri ile bireysel kayıt ve ilerleme yöntemini gerçekleştirecek düzeyde yeterli elemana sahip olmaması da olabilir. Eğer böyleyse, açık öğretim üniversiteleri ve uzaktan eğitim kurumlarının klasik üniversite ve yüksekokullara nazaran bireysel esnekliğe daha fazla istekli olması gerektiđi tezi ortaya atılabilir. Ancak bu çalışmada bu tezi destekleyecek kanıtlar bulunamamıştır [14].

1.1.3.10 Gelecekteki Gelişmeler

Online eğitime olan ilgi oldukça fazladır ve bu eğitimi uygulayan kurumlar hızla ve dünya çapında çoğalmaktadır. Kanada’da yapılan ciddi bir analiz, öncelikli büyüme stratejileri, daha çok ve çeşitli programlar açmak, uluslararası öğrencilere hitap etmek ve şirket eğitimi gibi yeni ve çekici pazarlar bulmak olması gerektiđi sonucuna varılmıştır. Bu çalışma sonucunda da ilerde, derslerin kalitelerinin daha iyi olacağı, daha çok ders açılacağı, ilave online hizmetler verilerek derslerin çekici hale getirileceđi, öğretmen eğitimi üzerine bir yönelme olacağı, diğer kurumlarla daha fazla işbirliği yapılacağı ve ilave organizasyona dayalı sonuçlar ortaya çıkacağı öngörülmektedir [14].

1.1.3.11 Engeller

Online eğitimde büyük ölçekli başarı beklenebilmesi için üstesinden gelinmesi gereken çok sayıda engel mevcuttur. Bunlar arasında, finansal engeller, deđişime karşı gösterilen direnç, bant genişliği sınırlamaları, erişim sınırlamaları, yetersiz araştırma araçları, telif hakları konuları ve online deđerlendirme önündeki engeller vardır.

Finansal engeller önemlidir. Çalışma sonucunda çok az sayıda kuruluşun öğrenci ücretlerinde gelen sağlam bir gelirinin olduğunu göstermektedir. Aynı zamanda geliştirme ve bakım maliyeti de yüksek olabilir. Ayrıca, bazı ülkelerdeki ulusal düzenlemeler kurumların öğretme ücretini ödemeyi reddetmektedir.

Online deđerlendirmenin efektif kullanımı önünde de çok sayıda engel vardır. Bunlardan bazıları, kamusal ve kurumsal düzenlemeler, fiziksel devamlılık gelenekleri, teknik imkansızlıklar, öğrenci tanımlanması, çalıntı dijital metaryellerin tespiti gibi konular bulunmaktadır [14].

1.1.3.12 Stratejik Öneriler

Sonuç olarak, online eğitimciler, eğitim yöneticileri ve politikacılar için aşağıdaki sekiz öneri sıralanmıştır

- Öğrencilerin online hareketliliklerini kolaylaştırmak için, derecelerin, sertifikaları, kredilerin ve notların ulusal ve uluslararası uyumunu arttırmak gerekir.
- Kurumları öğretim ücretlerini ödemesini engelleyen ulusal düzenlemelere karşı çıkmak gerekir.
- Maliyet olarak verimli olan online eğitim üzerine yoğunlaşmak gerekir.
- Online eğitimin yönetimi için daha iyi sistemler geliştirmek gerekir.
- Online eğitici, yönetici ve kanun yapıcıların eğitimi için yapılan teşvikleri desteklemek gerekir.
- Online değerlendirmeyi engelleyen tutum ve düzenlemelere karşı çıkmak gerekir.
- Online pedagoji ve didaktik üzerine daha fazla araştırma çalışmasını desteklemek gerekir.
- Online öğretmenlerin iş yüklerini azaltmak için yöntemler geliştirmek ve uygulamak gerekir [14].

1.2. Web Tabanlı Eğitim Kavramı

Web Tabanlı Eğitim (WTE), İnternet araçlarını kullanan bir uzaktan eğitim modelidir. WTE uzaktan eğitim sistemleri arasında en yenisidir. WTE; uzaktan eğitim, bilgisayar destekli eğitim ve İnternet'in bir arada kullanıldığı bir sistem olarak ortaya çıkmıştır.

Genel olarak, WTE'in bir tanımı yapılırsa, Web üzerinde uzaktan eğitimi desteklemek amacıyla hazırlanmış kişisel sayfalara ve bu sayfaların oluşturduğu eyleme "WTE (WTE)" denir. WTE İnternet'in gelişmeye ve yaygınlaşmaya başlamasıyla birlikte uzaktan eğitime açılmış yeni bir kapıdır. Uzaktan eğitim WTE ile yeni bir boyut

kazanmıştır. WTE ile birlikte uzaktan eğitimin enteraktifliği artmış, öğrenci ve öğretmenin senkron veya asenkron olarak fikir tartışması mümkün hale gelmiştir. WTE hızla ilerleyen teknolojinin yakalanması ve takip edilmesi adına en büyük imkandır. Çünkü her gün yeni birçok yeni teknolojinin ortaya çıktığı dünyamızda bu yenilikleri takip etmenin tek yolu Web tabanlı eğitimidir.

WTE insanların öğrenme şekillerini değil, öğretmenlerin öğretme yöntemlerini değiştirmiştir. WTE internetin ve bilgisayarın kolay elde edilir olmasıyla her evi bir okul, bir fakülte, bir kurs haline getirmiştir. WTE’de internetin ve bilgisayar teknolojisinin tüm nimetlerinden faydalanılarak pedagojik açıdan çok yüksek kalitede dersler hazırlanabilir. Dersin anlatımında İnternet ve bilgisayarlar tarafından desteklenen yazı, ses, görüntü, animasyon, simülasyon kullanılabilir. Başka bir deyimle dersin öğrenciler tarafından anlaşılabilmesi için gerekli bütün şartlar sağlanabilir. Bu da Web Tabanlı Eğitimin eğitim kabiliyetini göstermektedir [15].

1.2.1 Web Tabanlı Eğitimin Avantajları

- WTE’de öğrenciler, çalışmak istediklerinde ister evlerinden, isterse de İnternet’e bağlanılabilen başka mekanlardan eğitimlerini alırlar. Böylece “her zaman her yerde eğitim” olgusu gerçekleşmiş olur. Bununla birlikte teknolojik imkanların elverdiği ölçüde kameralardan da faydalanılabilir. Bunun sonucunda da “yüz-yüze eğitim” olgusu gerçekleştirilir.
- WTE ile kurumlar ve bölgeler arası dengesizlikler kaldırılır veya en aza indirilerek eğitimde fırsat eşitliği sağlanır.
- WTE’de öğrenciye klasik eğitimde sunulan kaynaklardan çok daha büyük ve geniş kaynak sunulur. WTE içerisinde hazırlanacak olan web sayfaları ile öğrenciler konular ile ilgili bağlantılar yapması ve gerekli bilgileri bizzat bilginin kaynağından alması sağlanabilir
- WTE klasik eğitim sistemine göre %40 - %60 daha ucuzdur. WTE de klasik eğitim sisteminde temin edilmesi gereken bina, sınıf, sıra, tahta gibi birçok araç kullanılmaz. Ders için gerekli olan, öğrencilerin evlerinden bilgisayarları ile

dersin İnternet sitesine bağlanmalarıdır. Deneylerde ve uygulamalarda kullanılan araç ve gereç yerine onun animasyonu, simülatörü kullanılabilir.

- WTE’de ders içeriklerinin anlatılmasında sayfaların gerekli yerlerine resim, ses ve görüntülerin eklenmesiyle canlandırılıp öğrencilerin derse olan ilgi ve motivasyonları arttırılabilir
- WTE’de ders anlatımları animasyonlarla ve simülasyonlarla zenginleştirilerek dersin eğitim kalitesi klasik sisteme oranla çok arttırılabilir.
- WTE’de öğrenci ders içerikleri içindeki daha önceden bildiği konuları atlama, anlamadığı konuları ise istediği kadar tekrar etme şansına sahiptir. Eğer varsa o konu hakkındaki linkleri tarayarak konu hakkında daha fazla bilgiye de ulaşabilir.
- WTE ders materyalleri öğretmen tarafından istenildiği anda güncellenebilir veya değiştirilebilir.
- WTE öğrencilerin düşünme kabiliyetini geliştirir. Öğrenciler eğitim sırasında WTE içerisinde bulunan “Tartışma Grupları” ya da “Mail Grupları” sayesinde sorulan sorular ya da karşılaşılan problemler hakkında sürekli tartışacaklardır.
- WTE geleneksel sınıf ortamında soru soramayan veya grup içinde katılım yetisine ulaşamayan adayların, elektronik ortamda özgüven kazanmalarını sağlar [12].

1.2.2 Web Tabanlı Eğitimin Dezavantajları

- WTE bir kursun hazırlanması için daha fazla zaman, daha fazla çaba gerektirir. WTEsayfası tasarlayan eğitimciler normal eğitime göre %40-50 daha fazla çaba harcarlar.
- WTE öğrencilerin de daha fazla çalışmasını ve çaba sarf etmesini gerektirmektedir. Web Tabanlı Eğitim, öğrencilerin oldukça iradesine sahip, kendi başına çalışmayı seven bir yapıda olmasını gerektirir.
- WTE öğrencilerinin dersi takip etmek için gerekli temel bilgisayar bilgilerini almış, İnternet teknolojilerini bilen ve bu bilgilerinin ışığında herhangi bir sorun çıktığında kendi başına üstesinden gelebilen yapıda olmaları gerekmektedir.

- WTE öğrencileri bilgisayarlarının kapasitelerinden dolayı bazı programların çalıştırılmaması, kullanılan programların versiyonlarının aynı olmaması ile oluşabilir ve üstesinden gelinmesi vakit ve para gerektirebilir.
- WTE yapan öğretmenlerin İnternet tabanlı eğitim araçlarının kullanımındaki bilgi eksiklikleri ve öğretmenlerin pedagojik açıdan eğitim materyallerinin hazırlanması konularında yetersiz kalmalarından dolayı ders içerikleri eğitim açısından gerektiği kadar güçlü olmayabilir. Ders anlatımları gerektiği kadar animasyonlarla, simülasyonlarla desteklenmeyebilir.
- Dersi takip eden öğrencilerin esastan çok teknoloji ile ilgilenmeleri sonucunda derse olan ilgi azalabilir.
- Kendi kendine çalışma alışkanlığı gelişmemiş öğrenciler dersi takipte zorlanırlar ve başarılı olamayabilirler.
- Web Tabanlı Eğitimde uygulamalar ve deneyler için her ne kadar simülörler kullanılsa da kabiliyet gerektiren eğitimlerde WBE nin başarı yüzdesi düşüktür [16].

1.2.3 WTE Önündeki Engeller

Özellikle devlet üniversitelerinde fakültelerin direnmesi WTE'nin ilerlemesini geciktirmektedir. WTE klasik ders sistemlerine göre çok daha fazla efor sarf etmeyi gerektirmektedir. Ülkemizde ve dünyada bir çok ülkede akademik ilerleme için verilen dersler ve içeriği neredeyse hiç önemli değildir. Tüm akademik kariyer çeşitli indekslerde taranan dergilerde yapılan yayınlar anlamına gelmektedir. Dolayısıyla WTE'nin getirdiği ilave iş yükü yayın sayı ve kalitesini düşürmektedir. Web tabanlı eğitimin gerçeği olarak, eğer sınıfın yönetimini ve hazırlıklarını öğretici yapıyorsa, her sınıf için normalin en az iki katı zaman ayırmak gerekmektedir. Öğrenci ile öğretici arasındaki iletişim çok daha fazla olmakta ve geniş bir zaman yayılabilmektedir. Ayrıca, hazırlık aşaması çok komplekstir. WTE'de tüm ders notlarını kapsayan PowerPoint sunumlarına, her sunumun video kayıtlarına, çok geniş kapsamlı ve objektif, aynı zamanda doğru ve yanlış cevapları yorumlayabilen sınav materyallerine ve tartışma ortamı için bir forum düzenine ihtiyaç duyulmaktadır. Ders hazırlığı normal derslerin en az iki katı zaman gerektirdiğinden

akademik yayın çalışmalarına ayrılacak zaman kısıtlanmak zorundadır. Bu nedenle WTE, yeni akademisyenlerin günümüz üniversitelerinde yerleşmeye başlayan “yayın yap ya da öl” yaklaşımına göre ölme risklerini arttırmaktadır [17].

WTE sistemine geçiş daha fazla çalışma, ve daha yeni alanlarda uzmanlaşma gerektirmekte ve ne öğretildiği noktasındaki sorumluluğu arttırmaktadır. Ders web ortamında bir kez yayınlandığında, tüm denetim ve eleştirilere açık hale gelmektedir. Bu durum bir çok akademisyenin istemediği bir çalışma şekli olabilir.

Yöneticiler web tabanlı eğitimi, teknik destek ve ekipman gerektirmesi nedeniyle tüm akademik bölüm yapısını değiştirmek orunda kalabilecekleri endişesiyle çok iddialı bulunmaktadır. Ayrıca sistemin il yatırım maliyetinin yanı sıra bakım ve işletme maliyetleri de oldukça yüksektir. Geleneksel eğitimde üniversiteler, kütüphanelerinde literatür yayınlarını bulundurmak, öğrencilere sınıf tahsis etmek ve çeşitli yazışmaları yürütecek sekreteryalar birimlerine ihtiyaç duymaktadırlar. Web tabanlı eğitimde ise tüm bunların yanı sıra ders hazırlama materyallerini almak ya da kiralamak, çok sayıda asistan, tekniker, bilgisayar uzmanı ve daha birçok ilave ekipmana ihtiyaç duyacaklardır [18].

1.3. Web Tabanlı Eğitimin Tasarlanması

WTE tasarlanmaya başlanmadan önce tespit edilmesi gereken bazı noktalar vardır. Bunlar aşağıda kısaca özetlenmiştir.

- Hazırlanacak Web Tabanlı Eğitimin amacı, sınırları tayin edilmelidir.
- Web Tabanlı Eğitimin hedef kitlesi ve bu hedef kitlesinin bilgi seviyesi iyi belirlenmelidir.
- Uygulanacak eğitim süreci sonunda öğrencilere kazandırılmak istenen hedef iyi belirlenmelidir.
- WTE için kullanılacak eğitim materyalleri belirlenmeli, Bu materyallerin web ortamına uygunluğu araştırılmalıdır.
- Belirlenen eğitim amacına ve hedef kitlenin bilgi seviyesine uygun pedagojik yaklaşım belirlenerek eğitim kalitesi yüksek tutulmalıdır.

- Öğrencilerin derse olan motivasyonunu yüksek seviyede tutmak için gerekli tedbirler alınmalıdır.
- Öğrencinin eğitimciyle etkileşim ve iletişim kurması için gerekli tedbirler alınmalıdır [19].

1.4. Web Tabanlı Öğretimde Etkileşim

Öğretimsel web sitelerinin tasarım ilkelerine uygun etkileşimli çoklu ortam uygulamalarını içermesi, konunun yüksek bir yüzde ile öğrenilebilmesi için oldukça önemlidir. Çoklu ortam destekli eğitimsel web siteleri özel uygulamalar tarafından oynatılan küçük ses ve video dosyalarını içermektedir.

Uzaktan eğitimde üç etkileşim biçimi vardır: Öğrenen-öğretmen, öğrenen-içerik ve öğrenen-öğrenen.

Öğrenen-öğretmen etkileşimi; öğretmeni sunuda, anlatımda, dönüt sağlamada ve öğrenciye yardım etmede bir öğretim biçimi şeklinde olmaktadır. Öğrenen de soru sorarak, ödev vererek, problemleri öğretmenle tartışarak etkileşime girebilmektedir.

Öğrenen-içerik de etkileşimin temel bileşenidir. İçerik; kitaplarda, çevredeki nesnelerde, soyut düşüncelerde, video teyplerde, bilgisayar programlarında, Web sitelerinde bulunabilmektedir.

Öğrenen-öğrenen etkileşiminde ise öğrenciler, kendi aralarında projelerini değerlendirmelerini, tartışmalarını, düşüncelerini. paylaşabilmekte, birbirlerine yardım edebilmektedir.

Etkileşim öğrenme sürecinin en önemli bileşenidir [20]. Web-tabanlı eğitimde öğretici ile öğrenci arasındaki etkileşim, sanal sohbet ortamları ve sanal tartışma grupları ile sağlanmaktadır. Sanal tartışma gruplarına tüm öğrenciler ve eğitimci katılmaktadır. Eğitimci veya öğrenciler var olan problemlere ilişkin mesajları panoya göndermektedir. Tartışma konusu eklendikten sonra kişiler görüşlerini bu ortama asenkron (eş zamansız) olarak ekleyebilmektedirler.

Eğitiminin diğer derslerle çakışmayacak şekilde belirlediği bir saatte öğrencilerle sohbet etmesi de yararlı etkileşim faaliyetlerinden biridir. Bu etkileşim yöntemi senkron (eş zamanlı) bir yöntemdir.

Elektronik-posta diğer bir etkileşim yöntemidir. Öğrenciler, eğiticilere, sistem yöneticisine veya diğer öğrencilere e-posta yoluyla ulaşabilmektedirler [15].

1.5. Türkiye’de Uzaktan Eğitim

Türkiye’de 1982 yılında yürürlüğe giren “Açık Yükseköğretim Yönetmeliği” kapsamında Anadolu Üniversitesi bünyesinde TRT aracılığıyla yapılan açık öğretim uygulamalarının detayları bilinmektedir [21]. Bu Kanun’da diğer üniversitelerin de açık öğretim yapabilecekleri hükmü getirilmişse de, bu madde mali ve teknolojik nedenlerle hayata geçememiştir. 1990’lı yılların başından itibaren özel radyo ve televizyonların sayısındaki artış bu konu ile ilgili yeni bir fırsat sunmuş ancak kanuni yetersizlikler ve engeller nedeniyle televizyon aracılığıyla açık öğretim faaliyeti halen başladığı gibi devam etmektedir.

İnternet tabanlı uzaktan eğitimin yolu ise 1999 yılının sonunda yürürlüğe giren “Üniversitelerarası İletişim ve Bilgi Teknolojilerine Dayalı Uzaktan Yükseköğretim Yönetmeliği” ile açılmıştır [22]. Bunu takiben 2000 yılında çıkarılan “Enformatik Milli Komitesi Yönetmeliği” ile başvuruların değerlendirileceği bir merkez oluşturulmuştur [23]. Aradan geçen 10 yıl içerisinde bir çok üniversite kendi uzaktan eğitim yönetmeliklerini çıkarmıştır. İlk olarak Sakarya ve Mersin üniversiteleri bünyesinde faaliyet gösteren uzaktan eğitim merkezleri açılarak öğrenci kabulüne başlamışlardır. Bu gün Yüksek Öğretim Kurumu (YÖK) tarafından öğrenci almasına onay verilmiş yedi üniversitede (Anadolu, Çukurova, Doğu Akdeniz, İstanbul Bilgi, Mersin, Ortadoğu Teknik ve Sakarya üniversiteleri), 12 ön lisans, 6 yüksek lisans olmak üzere 18 birim mevcuttur [24].

Son yıllarda YÖK tarafından üniversitelerin belli birimlerinde web tabanlı uzaktan eğitime geçilmesi yönünde telkinler yapılmaktadır. Uzaktan eğitimle kazanılmış yeterlilik ve derecelerin tanınması ile ilgili çalışmalar yürütülmektedir. Bu konuda bazı yasal boşluklar bulunsa da kanun ve yönetmeliklere uygun olarak açılan merkezlerden mezun olanların diploma denkliklerinin verilmesi de önemli bir teşvik mekanizması

olarak işletilmektedir [25]. Giderek daha çok öğrencinin yararlandığı web tabanlı öğretim sistemlerinin, artan öğrenci sayısı karşısında kalitelerini garanti edip edemedikleri, yüksek kaliteli bir uzaktan eğitimde dikkate alınması gereken pedagojik deneyimlerin neler olacağı gibi yeni sorular gündeme gelmekte ve bu soruların cevabının uzun bir süreç sonunda alınabileceği görülmektedir [13].

1.6. Web Tabanlı Eğitim ve Sanal Laboratuvarlar

Bilimsel hipotezlerin analizi ve matematiksel modellerinin elde edilmesi öncelikle fiziksel olayın kesin ve organize bir şekilde gözlemlenmesine bağlıdır. Dolayısıyla özellikle mesleki ve teknik eğitimde deney ve gözlem tekniği öğrencilerin anlama ve gerçek sistemlerle matematiksel modelleri karşılaştırmalarında çok önemli bir işlevi yerine getirmektedir. Bilimsel bir deney, genellikle, tahmin etme, gözlemlene ve sonucu açıklama aşamalarını içerir. Deneysel çalışmaların sağladığı yararlar üç temel maddede incelenebilir:

1. Öğrencilerin konu ile ilgili kavramsal bilgisini derinleştirerek bilimsel teoriler ile gerçek dünya arasında bağlantı kurmasını sağlar.
2. Öğrenci deney esnasında ilgili alanda kullanılan teknikler, aletler ve teçhizat hakkında uygulama aşinalığı kazanır ve bir prosedürün yürütülmesi ile ilgili alışkanlıkları gelişir.
3. Özenli seçilmiş uygulama örnekleri öğrencinin daha çekici bir ortamda tecrübelerini ve motivasyonunu arttırıcı bir öğrenim yapmasına ve kazandığı alışkanlıkları sergileyebilmesine olanak verir [26].

Bu önemli işlevleri yerine getiren uygulama derslerinin mesleki eğitimden ayrılmasının düşünülemeyeceği açıktır. Ancak günümüz koşullarında klasik yöntemlerle laboratuvar ortamında uygulama dersleri yapmanın bir çok zorlukları vardır. Öğrenciler için en ideal yöntem olan klasik deney yönteminde her öğrenci için bir deney seti kurmak gibi bir zorunluluk oluşmaktadır. Bunun çoğu zaman imkansız olması, gruplar halinde deney yaptırılması sonucunu doğurmakta ve bunun sonucunda öğrenme kalitesi ve öğrencilerin derslere ilgileri azalmaktadır.

1.6.1 Sanal Laboratuvarlar

Dijital elektronik alanında son yirmi yıldaki gelişmeler, yaşadığımız zaman diliminin bazı kişilerce “dijital çağ” olarak adlandırılmasına neden olacak kadar hızlı olmuştur. Bu gelişim, analog-dijital dönüştürücüler, örnekleme teknikleri ve dijital sinyal işleme özelliklerine sahip ölçme sistemlerinin ucuzlaması ve daha kolay erişilebilir olması sonucunu doğurmuştur. Uzaktan eğitimle birlikte sanal laboratuvar kavramının da yaygınlaşmasının temelinde de bu gelişmeler yer almaktadır [27]. Sanal laboratuvarların avantajları aşağıdaki gibi özetlenebilir:

- Tasarımları ve kurulumları ucuzdur. Çoğunlukla, karmaşık, pahalı cihazlar ve masraflı gerçek testler içermezler.
- Tasarım, örnekler ve modellerin önceden hazırlanmasıyla çok hızlı bir şekilde yapılabilir.
- Bir ekipman ile aynı anda birden fazla uygulama örneği geliştirilebilir ve bu örnekler arasından maliyet, boyut, performans, güvenilirlik gibi önemli parametrelerde kesin kriterlere dayanan optimal tasarımlar seçilebilir,
- Bir projede coğrafi şartlardan bağımsız bir çok bilim adamı çalışabilmektedir. Bu da günümüzde çok önemli olan bilgi ve tecrübe paylaşımlarının artmasına katkıda bulunabilir.
- Tasarımda kullanılan bilgisayar destekli tasarım (CAD) materyalleri, bu konudaki uzmanların eğitiminde kullanılabilir.
- Tasarım ve eğitim aşamasında çok geniş bilgi ve veri tabanları oluşturma olanağı sağlarlar.
- Bu veri tabanları ile hesaplama birimleri bilimsel araştırmalarda da kullanılabilir [28][29][30].

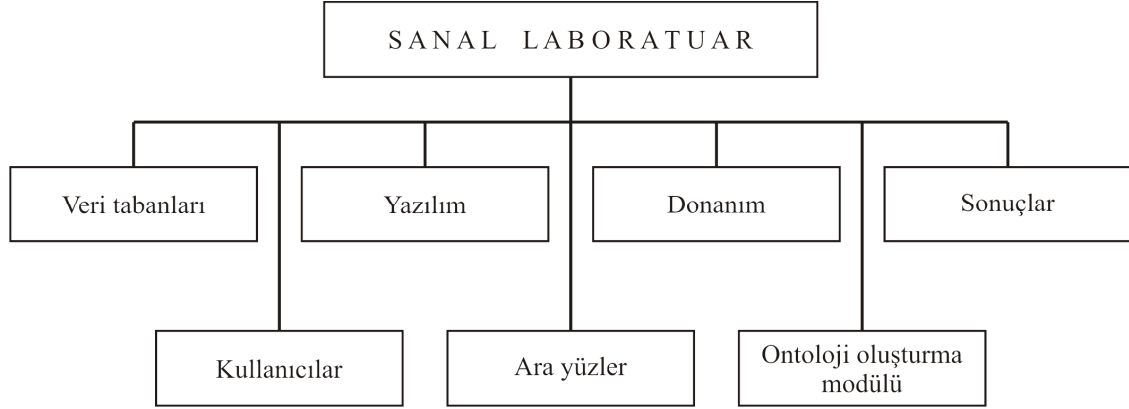
1.6.2 Sanal Laboratuvar Yapıları

Tipik bir sanal laboratuvar aşağıdaki temel bileşenleri içermelidir.

1. *Veri tabanı:* Sanal laboratuvarın fonksiyonlarını yerine getirmesi ve yönetimi için gerekli tüm bilgileri içeren birimdir. Bu veriler, elektronik elemanlar hakkındaki

bilgileri, elemanların ve birimlerin modellerini, tipik fonksiyonel çözümleri, sanal değerlendirme kurullarını, üreticiler ile ilgili bilgileri, araştırma nesnesi modellerini, enteraktif geliştirme araçlarını içermektedir. Ayrıca bu bilgi kütüphaneleri modelleri, araçları ve çözümleri de kapsamaktadır.

2. *Yazılım*: Sanal laboratuvar ve veri tabanının yönetimi, sanal araştırma ve deney hesaplamaları, girdi işleme ve tam kademeli cihaz geliştirme gibi fonksiyonları yerine getirmektedir.
3. *Donanım*: Yazılımın işlevliğini sağlamak amaçlı olarak veri girişi, depolama ve çıkış verme gibi işlevleri yerine getiren ekipmanlardan oluşur.
4. *Sonuçlar*: Gerekli cihaz listelerini de içeren tipik formlar, belirli fonksiyonel çözümler, tasarım dokümanları, ihtiyaçların özellikleri, kataloglar ve uygulama notları gibi bileşenlerdir. Bu formların bazıları standartlaştırılmıştır ve başka projelerden kopyalanabilir.
5. *Kullanıcılar*: Yukarıdaki bileşenleri kullanan kişilerdir. Genellikle yönetici, tasarım mühendisi ve müşteri (öğrenci) olmak üzere üç değişik kullanıcı vardır.
6. *Ara yüzler*: Sanal laboratuvar modülleri arasındaki ve sanal laboratuvar ile kullanıcılar arasındaki karşılıklı etkileşimi düzenleyen birimlerdir.
7. *Ontoloji oluşturma modülü*: Yapay zeka içeren özel bir yazılımdır. Sanal laboratuvarın işleyişini çeşitli tekniklerle bir insan-makine benzetimi içinde gerçekleştirmeyi amaçlar [28][29]. Şekil 2’de böyle bir birimi de içeren karmaşık yapılı bir sanal laboratuvara ait şema görülmektedir.



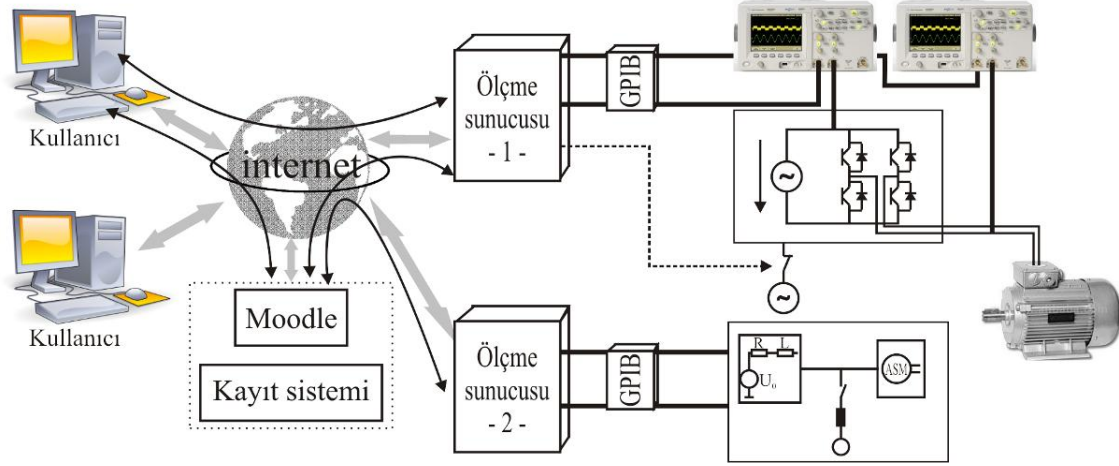
Şekil 2 Karmaşık bir sanal laboratuvar bileşenleri [28].

Uzaktan eğitimde sanal laboratuvar terimi iki farklı anlamda kullanılabilir. Her iki model aşağıda kısaca incelenmiştir.

1.6.2.1 Uzaktan Erişimli Deney Ekipmanları İçeren Sanal Laboratuvarlar

İdeal ve gerçekleştirilmesi daha zor olan bu tip sanal laboratuvarlarda öğrenciler uygulamada kullandıkları donanıma uzaktan erişerek gerçek bir deney seti üzerinde uygulama yapabilmekte ve sonuçları alabilmektedirler. Bu tip uygulamalarda laboratuvardaki donanımın güvenlik tedbirlerinin alınması gerektiğinden, en az bir görevlinin deneyler esnasında laboratuvarda bulunması zorunludur. Bu durumda öğrencilerin laboratuvarı ancak günün çalışma saatleri içerisinde kullanabilme olanakları olmaktadır [31]. İnternet teknolojisinin sağladığı olanakların artmasıyla, ülkenin ve dünyanın herhangi bir yerindeki bir çok laboratuvarın fiziki donanımları bir sanal laboratuvarın ara yüzü ile birleştirilebilmektedir. Böylece çok geniş imkanlara sahip sanal laboratuvarlar oluşturma imkanı doğmaktadır. Şekil 3’de örnek bir uzaktan erişimli sanal laboratuvar prensip şeması görülmektedir. Uzaktan eğitim yönetim sistemi (MOODLE:Modular Object-Oriented Dynamic Learning Environment) kullanıcılar ile sistem arasındaki trafiği denetlemektedir. Ölçme sunucusu ile ölçme elemanları arasındaki haberleşme ise IEEE-488 olarak bilinen genel amaçlı ara yüz (GPIB: General Purpose Interface Bus). ile sağlanmaktadır. Bu laboratuvar ara yüzü, Avrupa Birliği Leonardo da Vinci-II Programı kapsamında desteklenmekte ve Avrupa Birliği üyesi 14 ülkenin üniversitelerindeki fiziksel laboratuvarlar bir sanal laboratuvar oluşturmaktadır

[32]. Bu tip laboratuvar uygulaması gerek yüksek maliyet, gerekse uygun şartların hazırlanmasının zorluğu nedeniyle nadiren görülmektedir.



Şekil 3 Uzaktan erişimli laboratuvar prensip şeması örneği [32].

1.6.2.2 Benzetimlerden Oluşan Sanal Laboratuvarlar

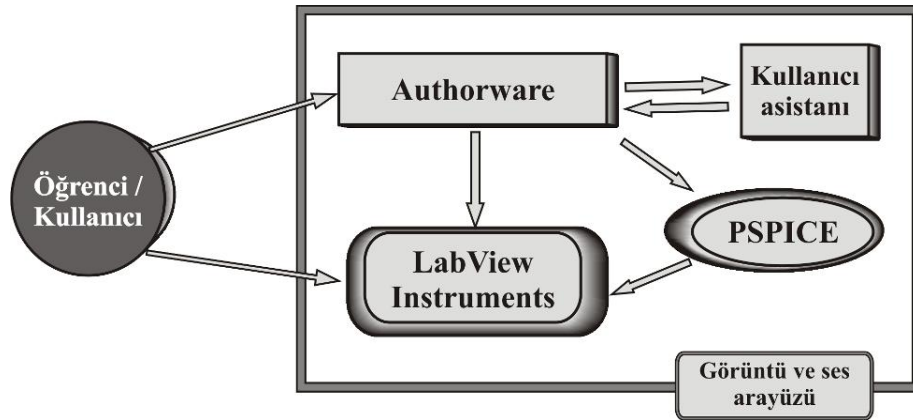
İkinci yöntemde ise, laboratuvar yapı ve işleyişinin bilgisayar ortamında benzetimi gerçekleştirilmekte ve öğrenciler bu benzetimleri çalıştıran ara yüzleri kullanarak deney yapmaktadırlar. Bu tip sanal laboratuvar uygulamalarında genellikle aşağıdaki paket programlar kullanılmaktadır.

- MATLAB (MATrix LABoratory); mühendislik alanında sayısal hesaplama, veri çözümleri ve grafik işlemlerinde kullanılabilecek genel amaçlı bir program olmakla beraber birçok özel amaçlı modüler paketlere de sahiptir. Ayrıca WINDOWS ortamında çalışan SIMULINK paketi ve grafiksel kullanıcı ara yüz (GUI) araçları etkileşimli benzetim programlarının hazırlanması ve çalıştırılmasında büyük kolaylıklar sağlamaktadır.
- LABVIEW (LABoratory Virtual Instrument Engineering Workbench); yüksek performanslı bilimsel ve mühendislik uygulamalarında ölçme ve otomasyon için tasarlanan grafiksel bir geliştirme ortamıdır. Bir ön panel ve blok diyagramı yapısından oluşur.

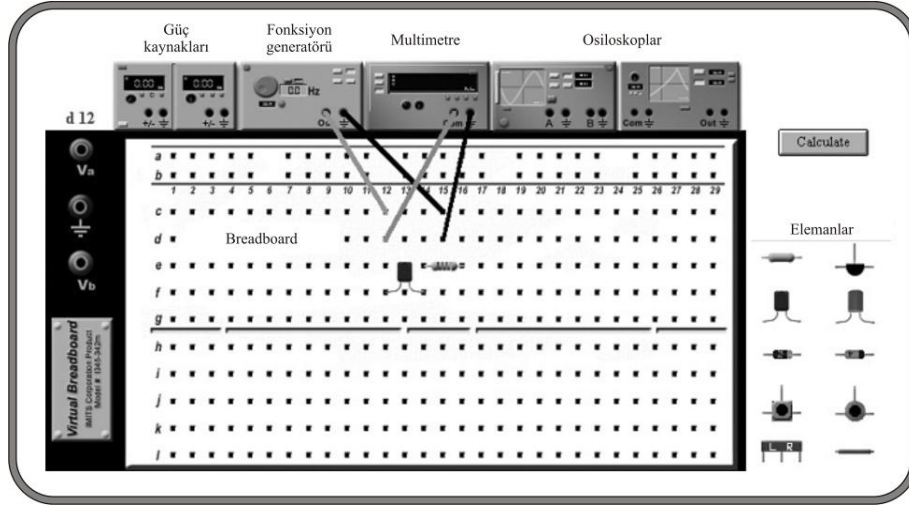
- EWB (Electronic Work Bench); elektrik ve elektronik eleman ve entegrelerini içeren bir paket program olup grafiksel bir ortam sunan başarılı bir simülasyon aracıdır.
- CISCO Lab Activity; Bilgisayar ağları üzerine dünya çapında kalitesi standart olan CISCO tarafından yapılan yazılım uygulamaları sanal laboratuvar oluşturmak için çok faydalı yazılımlardır.

Bunların dışında, PSPICE, VLEM, Mathcad, Ansys, Authorware, Mathematica gibi paket programlar da sanal laboratuvar oluşturmak için kullanılabilecek güçlü programlar arasındadır [33].

Şekil 4’de gerçek deney seti içermeyen sadece benzetimlerden oluşturulmuş bir sanal laboratuara ait prensip şeması ve Şekil 5’de kullanıcı ara yüzü örnekleri görülmektedir. Yukarıda sözü edilen programların bir kaçının bir arada kullanılmasıyla oluşturulan bu tip sanal laboratuvarların en önemli avantajları kurulumlarının daha kolay ve ucuz olmasıdır. Ayrıca kullanıcılar herhangi bir güvenlik endişesi duymaksızın her istediklerinde uygulama yapma imkanı bulabilmektedirler.



Şekil 4 Benzetimlerle oluşturulmuş bir sanal laboratuvar blok şeması [34].



Şekil 5 Sanal laboratuvar kullanıcı ara yüzü örneği [34].

1.6.3 Türkiye’de Sanal Laboratuvarların Geleceği

Sanal laboratuvarlar için durum belki de daha karmaşıktır. Çünkü yukarıda sayılan önemli avantajlarına rağmen laboratuvarları kullananların da kabul ettikleri, rekabet gücünü azaltan en önemli dezavantajı olarak, öğrencilerin gerçek cihaz ve aletlerle doğrudan temasının olmaması ve ölçme sistemindeki parçalar arasındaki fiziksel bağlantıları yapmamalarından kaynaklanan problemleri tamamıyla çözememiş olmaları gösterilmektedir [35]. Literatürde yapılan geleneksel-sanal laboratuvar karşılaştırma çalışmalarının çoğunda sanal laboratuvarların uygulamalara ayrılan zamanı ve öğrenci memnuniyetini arttırdığı ortaya konulmaktadır. Başarı noktasında da iki sistemin birbirine çok yakın olduğu öne sürülmektedir [36][37]. Ancak bu çalışmalarda sanal laboratuvar kullanan öğrencilerin, el becerisi ve kullanılan alet ve cihazları tanıma yetersiz kaldıklarının da altı çizilmektedir [38]. Bu önemli sorun çeşitli yollarla çözülmeye çalışılmaktadır. Örneğin, ülkemizde uzaktan eğitim veren bazı kurumlar uygulama derslerini belli zamanlarda okul laboratuvarlarında sıkıştırılmış olarak vermektedir. Bu uygulama uzaktan eğitim ve sanal laboratuvar uygulamasının mantığına ters düşmektedir.

Yukarıda sadece uygulama dersleri için sözü edilen olumsuzluklara rağmen, özellikle uzaktan eğitim sistemlerinin gelişimlerinin artarak devam etmesi beklenmektedir. Sanal laboratuvarların da el becerisinden çok muhakeme yeteneğinin ve

matematiksel modellerin ön plana çıktığı bilgisayar, kontrol ve tasarım mühendisliği alanlarında yüksek lisans ve doktora dersleri için uygulama alanlarını artması mümkün olabilir. Ayrıca Avrupa Birliği ve UNESCO tarafından 1997 yılında başlatılan ve ülkemizin de 2004 yılında imzaladığı Lizbon Süreci'nin hedefleri arasında mesleki ve teknik eğitimi geliştirmek ve yaşam boyu öğrenim uygulamalarını desteklemektir. Bunun sağlanabilmesi için uzaktan eğitim ve sanal laboratuvarlar önemli bir materyal olarak görülmektedir [39].

2. BÖLÜM

2.1. Yapay Zekâ

ZEKÂ NEDİR?

Zekâ iyi tanımlılık, sübjektiflik (özerklik) ve zamanla değişen bir kavramdır.

- ✚ Öğrenme veya anlama yeteneği veya yeni durumlarla başa çıkabilme; bilgiyi kullanma becerisi
- ✚ Çevreye bilgiyi uygulayabilme veya soyut olarak düşüncüyü amacına yönelik ölçebilme yeteneği [40]
- ✚ Elde etme, analiz yapma, anlama ve bunları bilgiye uygulayabilme yeteneği
- ✚ Sebepleri düşünebilme ve akıllıca bilgiyi tutabilme yeteneği .

2.2. Yapay Zekâ Nedir?

Yapay zekâ, doğadaki tüm canlıların davranışlarından ve en üstünü olan insanın davranış biçiminden esinlenerek böyle davranan sistemleri modelleme çalışmasının genel adıdır ve ismi 1950'li yıllarda 'Artificial Intelligence' olarak konulmuş bir olgudur. Yapay zekâ;

- İnsan gibi davranma: Turing test
- İnsan gibi düşünme: Bilişsel modelleme
- Rasyonel düşünme: Mantık
- Rasyonel davranma:
 - o İnsanlar gibi düşünen sistemler yapmak
 - o İnsanlar gibi düşünen sistemler yapmak

çerçevesinde, disiplinler arası bir kavram olarak ele alınabilir. Bu disiplinler felsefe, biyoloji, psikoloji, sosyoloji, bilgisayar, matematik, tıp ve bunların alt dalları olan muhakeme, makine öğrenmesi, doğal dil işleme, robotik ve benzeri alanlar olarak tanımlanır.

Yapay zekâ, genel görüşe göre, özellikle insan gibi davranan sistemlerdir. Ama diğer canlıların da model olarak alındığı göz önünde tutulursa genel anlamda doğadaki davranış biçimlerinin modellenmesi olarak tanımlanabilir [41].

İnsan gibi davranan sistemler denildiğinde bu davranış biçimi temelde 6 farklı şekilde karşımıza çıkar.Bunlar;

1. Kontrol: Bir sistemin kontrolü (Robot kontrolü,trafik kontrolü)
2. Karar Verme (Harp stratejileri, tıpta tanı, hukuk, uzman sistemler)
3. Tahmin (çıkarım) (Finans, trafik, görüntü işleme (image process))
4. Öğrenme,
5. Problem Çözme (NP, TSP, Pipeline)
6. Optimizasyon (En mâkul çözüm, en kısa yol, optimum süreç...)

Bütün bu çalışmaların sonunda yapay zekâ araştırmacıları iki gruba ayrıldılar. Bir grup insan gibi düşünen sistemler yapmak için çalışırken, diğer grup ise rasyonel karar verebilen sistemler üretmeyi amaçlamaktaydı. Aşağıda bu yaklaşımlar kısaca incelenmiştir.

2.2.1 İnsan gibi düşünen sistemler

İnsan gibi düşünen bir program üretmek için insanların nasıl düşündüğünü saptamak gerekir. Bu da psikolojik deneylerle yapılabilir. Yeterli sayıda deney yapıldıktan sonra elde edilen bilgilerle bir kuram oluşturulabilir. Daha sonra bu kurama dayanarak bilgisayar programı üretilebilir. Eğer programın giriş/çıkış ve zamanlama davranışı insanlarınkine eşse programın düzeneklerinden bazılarının insan beyninde de mevcut olabileceği söylenebilir.

İnsan gibi düşünen sistemler üretmek bilişsel bilimin araştırma alanına girmektedir. Bu çalışmalarda asıl amaç genellikle insanın düşünme süreçlerini çözümlemede bilgisayar modellerini bir araç olarak kullanmaktır.

2.2.2 İnsan gibi davranan sistemler

Yapay zekâ araştırmacılarının baştan beri ulaşmak istediği ideal, insan gibi davranan sistemler üretmektir. Turing zeki davranışı, bir sorgulayıcıyı kandıracak kadar bütün bilişsel görevlerde insan düzeyinde başarımlar göstermek olarak tanımlamıştır. Bunu ölçmek için de Turing testi olarak bilinen bir test önermiştir. Turing testinde denek, sorgulayıcıyla bir terminal aracılığıyla haberleşir. Eğer sorgulayıcı, deneğin insan mı yoksa bir bilgisayar mı olduğunu anlayamazsa denek Turing testini geçmiş sayılır.

Turing, testini tanımlarken zeka için bir insanın fiziksel benzetiminin gereksiz olduğunu düşündüğü için sorgulayıcıyla bilgisayar arasında doğrudan fiziksel temastan söz etmekten kaçınmıştır. Burada vurgulanması gereken nokta, bilgisayarda zeki davranışı üreten sürecin insan beynindeki süreçlerin modellenmesiyle elde edilebileceği gibi tamamen başka prensiplerden de hareket edilerek üretilmesinin olası olmasıdır.

2.2.3 Rasyonel düşünen sistemler

Bu sistemlerin temelinde mantık yer alır. Burada amaç çözülmesi istenen sorunu mantıksal bir gösterimle betimledikten sonra çıkarım kurallarını kullanarak çözümünü bulmaktır. Yapay zekâ'da çok önemli bir yer tutan mantıkçı gelenek zeki sistemler üretmek için bu çeşit programlar üretmeyi amaçlamaktadır.

Bu yaklaşımı kullanarak gerçek sorunları çözmeye çalışınca iki önemli engel karşımıza çıkmaktadır. Mantık, formal bir dil kullanır. Gündelik yaşamdan kaynaklanan, çoğu kez de belirsizlik içeren bilgileri mantığın işleyebileceği bu dille göstermek hiç de kolay değildir. Bir başka güçlük de en ufak sorunların dışındaki sorunları çözerken kullanılması gereken bilgisayar kaynaklarının üstel olarak artmasıdır.

2.2.4 Rasyonel davranan sistemler

Amaçlara ulaşmak için inançlarına uygun davranan sistemlere rasyonel denir. Bir ajan algılayan ve harekette bulunan bir şeydir. Bu yaklaşımda yapay zekâ, rasyonel ajanların incelenmesi ve oluşturulması olarak tanımlanmaktadır. Rasyonel bir ajan olmak için gerekli koşullardan biri de doğru çıkarımlar yapabilmek ve bu çıkarımların sonuçlarına göre harekete geçmektir. Ancak, yalnızca doğru çıkarım yapabilmek yeterli değildir. Çünkü bazı durumlarda doğruluğu ispatlanmış bir çözüm olmadığı halde gene

de bir şey yapmak gerekebilir. Bunun yanında çıkarımdan kaynaklanmayan bazı rasyonel davranışlar da vardır. Örneğin, sıcak bir şeye değince insanın elini çekmesi bir refleks harekettir ve uzun düşünce süreçlerine girmeden yapılır.

Bu yüzden yapay zekâyı rasyonel ajan tasarımı olarak gören araştırmacılar, iki avantaj öne sürerler. Birincisi “düşünce yasaları” yaklaşımından daha genel olması, ikincisi ise bilimsel geliştirme yöntemlerinin uygulanmasına daha uygun olmasıdır.

2.3. Yapay Zekânın Amaçları

İnsanlara karar vermede ve amaçlarına ulaşmada yardımcı olacak zeki sistemler sağlamak.

- Kavrayış (perception) – görme, dinleme, koklama, dokunma
- Muhakeme (reasoning) – Düşünce, plan, sorgu
- Eylem (Navigasyon, manevra.....)
- Öğrenme (Adaptasyon, keşif.....)
- Bilinç (içgüdünün hissedilmesi)

Bilimsel amaç: Zekânın temellerini (esaslarını) ve biyolojik sistemlerin fonksiyonlarını anlamaktır. Örneğin beyin nasıl çalışır?

Limitler ve sınırlar: Hangi işler mümkündür, hangi işler mümkün değildir? Bir işi düzenlemek için optimal (en uygun) yol nedir?

Mühendislik amaç: Gerçek ortamda zeki olarak hareket edebilen zeki makineler (programlar, özerk robotlar...) tasarlamaktır.

Yapay zekâ mühendislik açısından konuyla ilgilenir [41].

2.4. Yapay Zekânın Tarihçesi

Yapay zekâ (artificial intelligence) ismi ilk olarak Dartmouth konferansında ortaya atılmasına karşın, kökleri bilgi ve muhakemenin doğasına ait çalışmalar açısından binlerce yıl öncesine dayanır. Alan Turing ismi ile anılan hesabın ve hesaplamanın temel teorisi olarak bilinen Turing makinesini bulmuştur. Turing bu makinede modern bilgisayarlardaki iki önemli eleman olan programlar ve depolamayı bir araya getirmiştir.

Bu nedenle bilgisayar bilimindeki en prestijli ödöl onun adıyla anılan Turing ödölüdür. Bu Turing makinesi, Turing tarafından genelde kod çözme, matematik ve oyunlarda (satranç) kullanılmıştır. Yine adıyla anılan Turing Test çok popülerdir. II. Dünya savaşında müttefikler için kod çözücü olarak çalışan Alan Turing 1954' te intihar etmiştir.

1940→Sinir Ağları Teorisinin Doğumu (McCulloch & Pitts (1943), Hebb (1949))

1941→İlk elektronik bilgisayar

1948→İlk ticari bilgisayar

1956→Dartmouth konferansı

1956→Mantık teorileri geliştirildi

1958→LISP dili geliştirildi

1970→İlk uzman sistem

1969-1979→Bilgi Tabanlı Sistemler

1972→Prolog Geliştirildi

1980→Yapay zekâ endüstride kullanılmaya başlandı

1981→ Japonlar'ın 5. Kuşak projesi (CIRCA)

1986→Yapay zekâ temelli donanımların şirketlere 425 milyon\$' lık satış yapıldı

1986→ Yapay sinir ağlarına dönüş oldu

1988→ DEC 40 uzman sisteme sahipti

1991→ Yapay zekâ askeri sistemleri 1. Körfez Savaşı'nda etkili bir biçimde kullanıldı

1997→Deep Blue isimli satranç programı Kasparov' u yendi

2000→Robot hayvanlar vizyona çıktı.

2000→Kismet robotu (tebessüm eden robot)

2000→Nomad isimli robot göktaş örneklerini inceleyerek Antarktika'nın uzak bölgelerinin araştırılmasında kullanıldı.

2.5. Yapay Zekânın Alt Alanları

- **Problem Çözümü**
 - o Bulmaca-puzzle-satranç
 - o Matematik
 - o Arama
 - o Problem indirgeme teknikleri
- **Uzman Sistemler ve Uzmanlık**
 - o Bilgi toplama (bilgi nasıl elde edilir)
 - o Cevapların açıklanması
- **Planlama**
 - o Eylem planlaması
- **Robotik**
 - o Robot manipulator
- **Görme**
 - o Obje tanıma
- **Makine Öğrenmesi**
- **Sinir Ağları**
 - o Şimdiki veriden gelecek davranışlar hakkında tahmin
 - o Beynin fiziksel davranışını modelleme
- **Diller ve Ortamlar**
 - o LISP (1958 de gerçekleştirildi ,Çok basit fakat geliştirilemedi)
 - o PROLOG (İlk sürüm 1970 Fransa, 1980 Japonlar tarafından beşinci kuşak projesinde CIRCA' ya adapte edildi)

- o CLIPS
- o Nesne Tabanlı Programlama Teknikleri
- o Otomatik Programlama

2.5.1 Yapay Zekâda Önemli Teknikler

Bulanık mantık, yapay sinir ağları ve evrim algoritmalarının lineer olmayan dinamik sistemlerin kontrolü ve modellerin sentezi, analizi ve tasarımında kullanmak esas amaçtır.

- **YSA (Yapay Sinir Ağları)**
- **Uzman Sistemler**
- **Bulanık Mantık**
- **Genetik Algoritmaları**
- **Sürü Optimizasyon Teknikleri**

2.5.1.1 Yapay Sinir Ağları

Yapay sinir ağları (YSA), insan beyninin özelliklerinden olan öğrenme yolu ile yeni bilgiler türetebilme, yeni bilgiler oluşturabilme ve keşfedebilme gibi yetenekleri, herhangi bir yardım almadan otomatik olarak gerçekleştirebilmek amacı ile geliştirilen bilgisayar sistemleridir [40].

Yapay sinir ağları; insan beyninden esinlenerek, öğrenme sürecinin matematiksel olarak modellenmesi uğraşı sonucu ortaya çıkmıştır. Bu nedenledir ki, bu konu üzerindeki çalışmalar ilk olarak beyni oluşturan biyolojik üniteler olan nöronların modellenmesi ve bilgisayar sistemlerinde uygulanması ile başlamış, daha sonraları bilgisayar sistemlerinin gelişimine de paralel olarak bir çok alanda kullanılır hale gelmiştir.

İnsan beyninin çalışma prensibini taklit ederek çalışan bu sistemler, her ne kadar bilgisayar teknolojisi hızlı bir gelişim göstermiş, işlem hızları nano saniyeler mertebesine inmiş olsa da, bırakalım insan beynini, ilkel bir canlı beyninin fonksiyonları dahi baz alındığında, böyle bir organizmanın yanında çok ilkel kalmaktadır. Nano saniyeler

bazındaki işlem hızları ile YSA'lar, mili saniyeler mertebesindeki işlen hızları ile işlem yapan insan beyninin işlevselliğinin henüz çok uzağındadır.

Burada kısa bir hatırlatma yapmak gerekirse; insan beyinde yaklaşık 10^{11} sinir hücresinin varlığından bahsedilmekle birlikte, bu sayının bilgisayar ortamında modellenmesi şu an için mümkün görünmemektedir.

2.5.1.2 YSA' nın Kullanım Alanları

- Dil işleme
- Veri sıkıştırma
- Güvenlik
- Kontrol
- Robotik
- Tahmin (Piyasadaki en iyi stokları toplama – Hava Tahmini – Kansere Teşhisi)
- Kümeleme,
- Sınıflandırma
- Tanıma (El yazısı tanıma, konuşma tanıma)
- Sınıflandırma
- Veri Analizi
- Veri Filtreleme (telefon sinyalindeki gürültünün bastırılması)
- Finans – Piyasa (Stok piyasa tahmini – Stratejik planlama)
- İşaret İşleme (Hava tahmini – uydu görüntü analizi)
- Tahmin
- Bio informatik (protein ve genlerin fonksiyel analizi)
- Astronomi objelerin Sınıflandırılması (Astronomik verinin sınıflandırılması)

2.5.1.3 Uzman Sistemler

Yapay zekânın en önemli uygulama alanlarından biri *uzman sistemlerdir*. Bu tip sistem belli bir alanda uzman olan kişilerin uzmanlıklarına dayanarak çözüm arar. Bunu bir tür bilgisayarda düzenlenmiş danışma sistemi olarak düşünebiliriz. Uzman sistemler hem makine hem de insan müdahalesine ihtiyaç duyan uygulamalarda kullanılır.

2.5.1.4 Uzman sistemlerin uygulama alanları

- Tıp,
- finansal planlama,
- bilgisayar konfigürasyonu,
- gerçek zamanlı sistemler,
- trafik yönetimi ve kontrolü,
- sigortacılık ...

2.5.1.5 Uzman sistemlerin elemanları ve arayüzleri

- Bilgi tabanı (knowledge base): Uzmanın bildirilerinden oluşur. If-then kurallarıyla yapılandırılır,
- Çalışma alanı (working space): Problem çözümünde gerekli özel bilgileri bulundurur,
- Çıkarım birimi (inference engine): Bilgi tabanından ve problemin özel verilerinden gelen tavsiyelerle değişen sistemin merkezindeki kod.

Uzman sistem tasarımının anlaşılması için; sistem ile birbirini etkileyen kişisel rollerinde anlaşılmasına ihtiyaç vardır. Bunlar:

- Ana uzman (domain expert): Problemin çözüm yolunu tespit eden kişi veya kişiler.
- Bilgi Mühendisi (knowledge engineer): Uzmanın bilgisini çözerek uzman sistemin kullanabileceği şekle dönüştüren kişi.
- Kullanıcı (user): Uzman tarafından verilen bilgileri problem çözümünde kullanacak kişi.

2.5.1.6 Uzman Sistemlerin Özellikleri:

- Geriye zincirleme (backward chaning): If- then kuralları kullanılarak alt amaçlardan bir amaca varılır.
- Belirsizlik ile işleme (coping with uncertainty): Sistemin yeteneği, tam bilinmeyen kurallar ve verilere verdiği cevaplar ile muhakeme edilir.
- İleri zincirleme (forward chaning): Başlangıç verilerinden If-then kuralları kullanılarak problem çözümüne gidilir.
- Veri temsili (data represantation): Sistemde (erişilebilir ve depolanabilir) probleme özel veriler.
- Kullanıcı arayüzeyi (user interface): Sistem kullanılarak kolayca oluşturulan kod parçaları.

2.5.1.7 Bulanık Mantık

Bulanık mantık kavramı, ilk olarak 1965 yılında L. Zadeh tarafından kullanılmıştır. Bulanık mantık kavramı genel olarak insanın düşünme biçimini modellemeye çalışır. Klasik küme kavramında bir üye bir kümenin üyesidir veya üyesi değildir. Bulanık mantık kavramında bir üyenin bir kümenin üyesi olup olmadığı üyelik fonksiyonları ile belirlenir. Bu kavram ile bulanık mantığın kullandığı çıkarım yöntemleri kullanılarak olaylar hakkında yorum yapmaya çalışılır. Bulanık mantığın en güçlü tarafı var olan bir uzman bilgisinin kullanılmasıdır. Bu durum uzman bilgisinin tam olarak elde edilemediği durumlarda ise büyük bir dezavantaj oluşturur.

2.5.1.8 Bulanık Mantığın Kullanıldığı bazı uygulamalar

- o Hidroelektrik güç üniteleri için kullanılan Baraj kapılarının otomatik kontrolü (*Tokio Electric Pow.*)
- o Stok kontrol değerlendirmesi için bir uzman sistem(*Yamaichi, Hitachi*)
- o Klima sistemlerinde istenmeyen ısı iniş çıkışlarının önlenmesi
- o Araba motorlarının etkili ve kararlı kontrolü (*Nissan*)
- o Otomobiller için “Cruise-control” (*Nissan, Subaru*)

- o Dökümanların arşivleme sistemi (*Mitsubishi Elec.*)
- o Depremlerin önceden bilinmesi için Tahmin Sistemi (*Inst. of Seismology Bureau of Metrology, Japan*)
- o İlaç teknolojileri: Kanser teşhisi (*Kawasaki Medical School*)
- o Cep bilgisayarlarında el yazısı algılama teknolojisi (*Sony*)
- o Video Kameralarda hareketin algılanması (*Canon, Minolta*)
- o El yazısı ve ses tanımlama (*CSK, Hitachi, Hosai Univ., Ricoh*)
- o Helikopterler için uçuş desteği (*Sugeno*)
- o Çelik sanayinde makina hızı ve ısısının kontrolü (*Kawasaki Steel, New-Nippon Steel, NKK*)
- o Raylı metro sistemlerinde sürüş rahatlığı, duruş mesafisinin kesinliğini ve ekonomikliğin geliştirilmesi (1.Giriş ‘te bahsedilen metro hedefe 7 cm kala durabilmektedir)(*Hitachi*)
- o Otomobiller için gelişmiş yakıt tüketimi (*NOK, Nippon Denki Tools*)

2.5.1.9 Genetik Algoritma

Genetik algoritma (GA) süreci doğal evrime benzetilir. Bu nedenle *Üreme* (Reproduction), *Çaprazlama* (Crossover), *Mutasyon* (Mutation) gibi doğal evrimde kullanılan operatörleri içerir.

Üreme, uygunluk (fitness) değerlerine bakılarak stokastik yöntemlerle seçilen bireylerden yeni bir popülasyon oluşturma işlemidir. Bu işlem, ilerleyen generasyonlarda daha yüksek uygunluk değerlerine sahip bireylerin oluşmasına neden olur. Bu nedenle bu işleme en uygunun hayatta kaldığı test (survival of the fittest) adı verilir.

Çaprazlama, çoğunlukla rastgele olarak seçilen iki bireyin kromozomları çaprazlanarak gerçekleşir. Bu işlemde, bireylerin kromozomunu oluşturan dizilerin değişik kısımlar yer değiştirilerek yeni döl üretimi sağlanır. Bu döl popülasyonunda daha az uygunluk değerine sahip “zayıf” bireylerin yerine konabilir. Çaprazlama, genetik

algoritmada en önemli operatördür ve jenerasyonda yeni çözümlerinin üretiminden sorumludur.

Mutasyon, bireyin kromozomunu oluşturan dizideki tek bir elemanın değerinin rastgele olarak değişmesidir. Mutasyon, çözümün alt optimal noktalara takılmasını önleyen ve çok düşük olasılık değeri ile uygulanan operatördür.

Genel olarak genetik algoritma, çözüm bilgisinin hiç olmadığı veya çok az olduğu bir durumla aramaya başlar. Çözüm çevreden gelen etkileşime ve genetik operatörlere bağlıdır. GA, aramaya paralel bir şekilde, birbirinden bağımsız noktalardan başlar, bu nedenle alt optimal çözümlere takılma olasılığı azdır. Bu nedenle GA, karmaşık arama problemleri “birden çok alt çözüm kümesi olan” için en iyi optimizasyon tekniği olarak bilinir. GA ‘yı diğer Evrim Algoritması (EA) türlerinden farklı kılan özellikleri;

- Eşeyli üreme yöntemini,
- Mutasyon ve çaprazlama operatörlerini,
- Stokastik veya deterministik seçim yöntemlerini,
- Problemin çözümü için problemin kendisi yerine kodlanmış bir dizisini kullanması olarak gösterilebilir.

Bu şekilde GA diğer EA türlerinden daha esnek bir yapı sağlar. Bir evrim işlemi, potansiyel çözüm uzayında, popülasyonu oluşturan kromozomlar içinde en uygun kromozomu arama işlemidir. Böyle bir arama iki zıt amacı dengelemeyi gerektirir. Bu amaçlar; en iyi çözümlerin aranması (Exploit) ve arama uzayının genişletilmesidir (Explore) Genetik algoritma ile geleneksel optimizasyon teknikleri (özellikle nümerik metotlar) arasındada çok önemli farklılık vardır. Bu farklılıklar aşağıdaki gibi özetlenebilir:

i-) GA optimize edilecek olan parametrelerin kendileri ile değil kodlanmış dizileri üzerinde çalışır. Pekçok durumda ikili (binary) kodlama kullanılır. Fakat genetik algoritmalar için bu bir gereklilik değildir. Gerçek sayı kodlama, ağaç yapılı kodlama (tree coding) gibi farklı kodlama sistemleri de kullanılabilir.

ii-) GA, bir popülasyon içinde arama yapar. Bu popülasyon, problemin bütün olası çözümlerini temsil eden uzayı oluşturur. Başlangıç popülasyonu genellikle rastgele üretilen bireyleri içerir.

iii-) GA, problemin çözümünü belirlenen çözüm uzayında aramak için bir uygunluk fonksiyonu (fitness function) kullanır. Bu uygunluk fonksiyonu klasik optimizasyon tekniklerinde kullanılan amaç fonksiyonuna benzetilebilir.

iv-) GA, sonuca ulaşmak için stokastik yöntemler kullanır.

Genetik algoritmaları cazip kılan özelliklerinin bazıları şunlardır:

Öğrenme: Genetik algoritma global arama tekniklerinde yaygınca kullanılır ve en iyisi olarak bilinir. Mevcut performans ölçütlerini kullanarak verilen bir arama uzayında arama uzayını genişletme ve en iyi arama özelliklerini kullanma yeteneğine sahiptir. Bu özelliklerini çaprazlama, mutasyon ve üretim gibi genetik operatörlerle kullanarak öğrenme yeteneğine sahiptir.

Genetik kod yapısı: Genetik algoritma doğrudan parametrelerle değil, kodlanmış parametre dizisiyle çalışır. Bu, kullanıcıya problemleri bir değişken optimizasyon problemi gibi çözmesine imkan verir.

Çözümlerin optimalliği: Pek çok gerçek hayat problemlerinin *Çok Modellilik* (Multimodal) ve doğrusal olmama gibi özellikleri vardır. Geleneksel arama teknikleri böyle arama uzaylarında yetersiz kalır. Genetik algoritma ise böyle karmaşık arama uzaylarında optimale yakın çözümler bulma yeteneğine sahiptir. Genetik algoritma, mühendislik, bilim, ekonomi çok değişik alanlardaki problemler için gürbüz (Robust) bir optimizasyon aracı olarak son yıllarda büyük bir önem kazanmıştır.

Genetik algoritmanın uygulama alanlarından bazıları; haberleşme şebekleri tasarımı, elektronik devre dizaynı, gaz boruları şebekeleri optimizasyonu, görüntü ve ses tanıma, veri tabanı sorgulama optimizasyonu, uçak tasarımı, fiziksel sistemlerin kontrolü, gezgin satıcı problemlerinin çözümü, ulaşım problemleri, optimal kontrol problemleridir .

2.6. Yapay Zekânın Kullanım Alanları Ve Kullanılan Teknikler

Endüstri: Ev Temizlik Robotları, Otomatik araçlar için rehberlik, boru hattı denetimi, zeki evler, zeki ev teknolojileri, otomatik sürüş, konuşulan kelimeyi tanıma,

celestial nesnelerin sınıflandırılması, elektro-mekanik artecraftın kontrol yapısı ve morfolojisinin tasarımı

Savunma: Akıllı bombalar, hedef tayini.

Tıp: Kanseri teşhisi, DNA dizilerinde gen keşfi, adli tıp.

Güvenlik: Haberleşme ağlarının yönetimi, kriptoloji, kriminoloji)

Finans: Borsa tahmini, Vergi hazırlama yazılımı, kredi kartı sahtekarlığının ortaya çıkarılması.

İnternet Olayları: Web tabanlı arama motorları (Yahoo,Google...), haberleşme (e-mail, kelime işlemci)

Tasarım: Boeing 777, Pentium ,Space shuttle

İş Hayatı, postane, otomatik adres tanıma ve mektupların sıralanması;

Banka: Otomatik çek okuyucu, imza doğrulama sistemleri;

Telefon Şirketleri: Otomatik ses tanıma, telefon numaralarının gruplar içinde sıralanması;

Kredi Kartı Şirketleri: Sahtekarlık denetimi, uygulamaların otomatik görüntülenmesi;

Bilgisayar Şirketleri: Yardım masalarından otomatik teşhis uygulamaları; toplantı çizelgelerinin otomatik dağıtılması; apartmanlar için proje planlama)

Robotlar: Mars Rover, DSI, RoboCup, Oyuncak sektörü(Sony Aibo), eğlence sektörü

Oyun Oynama: Robocup, Sims,

Eğitim: Matematik problemlerin çözümü, teorem ispatı, dilbilgisi kontrolü, doküman düzenleme, simülasyonlar, planlama, müfredat geliştirme [40].

2.7. Endüstride Yapay Zekâ

2.7.1 Otomotivde Yapay Zekâ Teknolojileri

Otomotiv endüstrisinde artan rekabet ortamında geline son noktada; araçların ve

firmaların birbirlerine üstünlük sağlamaları kullandıkları yapay zekâ teknolojilerine bağlı olarak gerçekleşmektedir. Yapay zekâ teknolojileri akıllı sistemler olarak tanımlandığında; hayatımızı kolaylaştıran bu teknolojiler otomotiv endüstrisinde; akıllı hava yastıkları, akıllı sensörler, akıllı farlar, akıllı aynalar, akıllı fren sistemleri, akıllı klimalar ile donatılmış araçlar olarak günlük hayatımıza girmiştir. Son yıllarda, BM kontrol teknikleri otomotiv sektörünün çok geniş bir kısmında kullanılmaya başlanmıştır. Otomotiv endüstrisinde transmisyon uygulamaları, Motor kontrolü ve ABS fren sistemlerinde BM çok yaygın olarak kullanılmaktadır. BM kontrol uygulamaları geleneksel kontrol algoritmalarına göre daha üstün karakteristikler ortaya koymaktadırlar. Bu nedenle çalışmamızda, otomotiv endüstrisinde kullanılan yapay zekâ teknolojilerinin belirli uygulamaları ele alınmıştır. Yapay zekâ teknolojileri kullanım yeri ve amaçlarına göre tasnif edilmiştir.

2.7.2 Kullanım Amaçlarına Göre Yapay Zekâ Teknolojileri

- Güvenlik ve Emniyet Amaçlı Yapay Zekâ Teknolojileri
- Konfor Amaçlı Yapay Zekâ Teknolojileri
- Diğer Amaçlarla Kullanılan Yapay Zekâ Teknolojileri

2.7.2.1 Güvenlik Amacı İle Kullanılan Yapay Zekâ Teknolojileri

a Değişken Kodlu UK Immobilizer Sistemi

Araç güvenlik sistemlerinden biri olan immobilizer sistemi araç çalınmalarına karşı kullanılan bir sistem olup, araç elektronik kontrol ünitesi (ECU) ile araç immobilizer kontrol ünitesi arasında veri kontrol ve doğrulama esasına dayanmaktadır. Araç immobilizer kontrol ünitesi; aracın uzaktan kumanda ile açılması ve aracın çalıştırılmak istenmesi durumunda, anahtar içine yerleştirilmiş olan mikroçipte kayıtlı olan şifreyi tanımakta ve aracın çalışabilmesi için bu kodu tanıdığını ve kodun doğru olduğunu araç elektronik kontrol ünitesine iletmek sureti ile aracı çalışmaya hazır hale getirmektedir. Belirli bir algoritma ile değişen ve her seferinde farklı bir kod ile ilk çalışmayı gerçekleştiren yeni nesil immobilizer sistemleri günümüz gelişmiş elektronik teknolojilerine karşın, açma kodunun elektronik kaydı ve kopyalanması konusunda güvenlik açısından yeni bir uzman sistem uygulamasıdır.

b ABS Fren Sistemi

ABS, sert ve acil frenleme esnasında optimal araç kontrolü ve en küçük durma mesafesini garantiye almak için otomobillerde kullanılır. Günümüzde ABS araç güvenliğine hayati bir katkı olarak kabul edilmektedir[4]. Bir ABS Modülünün başlıca parçaları Elektronik Kontrol Ünitesi, (ECU), tekerlek hız sensörleri ve fren modülatörleridir. Tekerlek hız sensörleri ECU'ya tekerlek hızı ile orantılı frekanslarda elektronik palsler gönderir. ECU bu bilgileri değerlendirerek fren kuvvetini düzenler. ABS sistemleri yapıları gereği nonlineer ve dinamik olduklarından bulanık mantık kontrolünü için adaydır. Fuzzy ABS'ye giriş değerleri olan tekerlek hızı, hızlanma ve her bir tekerleğin kayması, tekerlekten gelen sinyallerin birleşimi ile hesaplanabilir. Bu sinyaller fuzzy-ABS sisteminde arzu edilen kontrolü gerçekleştirmek için işlenir. Bu değerlendirmeler ile aracın kayma yapmaksızın frenleme yapması ve emniyetli bir durma gerçekleşmesi sağlanmış olur [42].

c Elektronik Stabilite Programı

Klasik fren sistemlerinden sonra devreye giren ABS fren sistemleri ve daha sonra patinaj kontrol sistemleri (TC); elektronik stabilize programının (ESP) temelini oluşturmaktadır. ESP donanımına sahip bir araçta ABS ve TC de mevcut demektir. Mercedes, Bavarian Motor Works (BMW), Toyota, Honda, Ford ve Volvo gibi firmalar bulanık mantık konseptini kendi ürünlerinde birleştirmeye devam etmektedirler. Honda tarafından dizayn edilen Araç stabilite destek programı böyle bir örnektir. Bu sistem çok zekidir, sistem insan beyninden daha hızlı bir şekilde çalışır. VSA (Vehicle Stability Assist) fren basıncını, Motor gücünü veya her ikisini birden kontrol ederek çekişi en iyi şekilde devam ettirir. Siz hiçbir faaliyette bulunmasanız bile VSA motor gücünü veya frenleri ön veya arka taraftaki kaymayı azaltmak için kullanacaktır[42, 43].Sürüş ve frenleme güvenliğinde günümüz teknolojisinde son nokta olan bu sistem sayesinde araç stabilize en üst seviyelere çıkarılmıştır. Sistem devre dışı bırakılmadıkça otomatik olarak devrededir. Sistem bir çok sensörden bilgi almakla beraber özellikle direksiyon dönüş sensörü, kayma tespit sensörü ve ABS tekerlek sensörleri en önemli veri kaynaklarıdır. Aracın viraja girmesi anında tekerlek sensörlerinden gelen dönüş sayısı bilgileri, Direksiyon sensöründen gelen viraj acısı bilgisi ve Araç merkezindeki kayma

tespit sensöründen gelen kayma isteği bilgileri ve diğer sensörlerden gelen bilgilerde değerlendirilerek Elektronik Kontrol Modülü tarafından araç Fren sistemine komut verilmekte ve kaymanın olacağı tekerlek dışındaki tekerleklerle değişik basma kuvvetleri ile fren kuvveti uygulanarak araç ekseninde tutulmaya çalışılmaktadır. Bu suretle aracın yoldan çıkması riski büyük ölçüde engellenmiş olmaktadır.

d Hız Sabitleyici ve Hız Sınırlayıcı Sistemler

Sürüş konforu ile birlikte güvenlik ve yakıt tüketimi açısından düşünülerek tasarlanan bu sistemler sürücünün isteği doğrultusunda, aracı sabit bir hızda veya belirli bir limitin altında kullanmak ve araç önüne gelebilecek engellere otomatik yavaşlama tepkisi vererek güvenli bir sürüş sağlamak amacı ile araçlara uygulanmıştır. Sürücünün herhangi bir vesile ile frene veya debriyaja basması durumunda sistem devre dışı kalarak tekrar manuel kullanıma geçmekte fakat sabitlenen değer hafızada saklanmaktadır. Gerektiğinde tekrar kumanda panelinde ilgili butona basılarak hafızadaki değerde aracın sabit hızda seyretmesi sağlanabilmektedir. Uyarlanabilir Hız kontrol sistemi, aracın tamponunun arkasına yerleştirilmiş radar sistemini önündeki aracın mesafe ve hızını tespit etmek için kullanır. Bu durum bir gelişmiş radar sensörü, bir dijital sinyal işlemcisi ve bir uzunlama kontrol elemanı sayesinde gerçekleştirilir. Eğer öndeki araç yavaşlar ise veya başka bir nesne tespit edilir ise sistem motora veya fren sistemine bir bildirim sinyali gönderir. Yolda engel kalmadığında sistem ayarlı hız değerine aracın tekrar dönmesi için yeniden hızlanacaktır.

e Akıllı Hava Yastıkları

Pasif güvenlik sistemlerinden olan hava yastığı (airbag), kaza anında sürücü ve yolcularda meydana gelebilecek yaralanmaları en aza indirmek üzere kullanılmaktadır. Yeni nesil akıllı airbag sistemleri sayesinde koltukta oturan bir yolcunun olup olmadığı tespit edilmekte ve ona göre gerekmiyor ise ilgili hava yastığının açılmaması sağlanmaktadır. Aynı zamanda çift kademeli açılma prensibi ile, ön darbe algılayıcıları vasıtasıyla darbenin şiddeti tespit edilerek kademeli açılma sağlanmaktadır. Jaguar XK-S ve XJ modeli airbag sisteminde mevcut bazı sensörler vasıtası ile yolcunun pozisyonu tespit edilmekte ve airbag açma değerleri optimize edilerek; sürücünün pozisyonu,

ağırlığı, duruş açısı gibi ayrıntılar ölçümlenmekte ve tüm değerlere uygun açma zamanlaması ve açma hızı Yapay Sinir ağları kullanılarak uygulanmaktadır [44].

f Hidro-Aktif Sürüş Sistemi

Sistem istenilen durumlarda araç tekerlek-zemin mesafesinin değiştirilmesi mantığı ile çalışmaktadır. Sistem sürücü bölmesindeki bir ile aktif edilmekte ve aracın ön ve arka kısmında bulunan hidrolik pistonların üzerine etki eden basınç değerinin değiştirilmesi sonucu araç seviyesinde yükselme veya alçalma meydana getirmektedir. Araç seviyesindeki değişim miktarı sensörler vasıtası ile ölçülerek hidro-aktif sürüş sistem kontrol ünitesine iletilmektedir. Hidro-aktif sistem kontrol ünitesi CAN-BUS bilgi aktarım sistemi vasıtası ile diğer kontrol üniteleri ile de bilgi alış verişinde bulunmaktadır. Bu alış veriş sayesinde araç hızındaki değişime bağlı olarak araç yüksekliği sürücünün müdahalesi dışında otomatik olarak ayarlanabilmektedir. Özellikle yüksek pozisyonda bırakılan araç, hızlanma esnasında aerodinamik kaygılar ile otomatik olarak alçaltılmaktadır. Bu şekilde daha güvenli ve konforlu sürüş sağlanabilmektedir.

g Araç Takip Sistemleri

Bu sistem ile araç üzerine yerleştirilmiş olan ünite GPRS bağlantı ile bir alıcı üniteye SMS vasıtası ile veri aktarımı yapmakta ve bu veriler sayesinde alıcı ünite başında oturan operatör kullanıcının o andaki konumunu, hızını, yakıt durumunu, araç mevkisinin rakım değerini vs. tespit edebilmektedir. Ayrıca istenir ise kumanda ekranı başında oturan operatör tarafından araç bloke edilerek, araç alarımının çalışması sağlanabilmektedir. Bu sistem büyük şehirlerde araç takibini ve güvenliğini sağlamak için birçok büyük firma tarafından kullanılmaktadır.

h Şerit Değiştirme İkaz Sistemi

Bu sistem hem araç konforunu hem de emniyetini artıran bir uygulamadır. Bu sistem ile sürücü ani şerit değiştirmelerde koltuğun vibrasyonu ile ikaz edilmekte ve uyandırılmaktadır. Özellikle otoyolda uzun süreli kullanımlarda sürücünün dikkatinin dağılması ve uykusunun gelmesi tehlikeli bir durumdur. Bu durumun tespiti ve ikazı için kullanılan sistem araç altında şerit takip sensörleri ve sürücü koltuğunun altına yerleştirilmiş vibrasyon motorlarından oluşmaktadır. Sistem aktif edildiğinde, yol

şeritlerini kızıl ötesi ışınlarla takip eden sensörler sürekli olarak gövde kontrol ünitesine bilgi vermektedir. Araç istikametinde, sürücünün sinyal vererek şerit değiştirmesi durumu dışında meydana gelecek değişim, anında sensörler aracılığı ile tespit edilecek ve Kontrol ünitesinin komutu ile sürücü koltuğu altındaki vibrasyon motorları aktif edilecektir. Bu sayede sürücünün dikkati çekilecek ve olası kazanın önüne geçilecektir.

2.7.2.2 Konfor Amacı İle Kullanılan Yapay Zekâ Teknolojileri

a Akıllı Aynalar

Araç konfor sistemlerinden olan akıllı aynalar ışığa duyarlı olarak çalışmakta olup ortamdaki veya dış kaynaktan gelen ışık şiddetine bağımlı olarak renk değiştirmekte ve sürücü açısından ideal kullanım sağlamaktadır. Özellikle gece sürüşlerinde arkadan gelen araçların aydınlatma sistemlerindeki aksaklıklar sebebi ile meydana gelen göz rahatsızlıkları bazen kazaya bile sebebiyet vermekte olup, akıllı aynalar ile bunun önüne de geçilmektedir. Ayrıca geri yönde araç kullanımı esnasında sağ aynanın daha rahat park etmek için manuel ayarlanma zorluğu akıllı aynalar ile otomatik olarak kaldırılmıştır. Araç geri vitese takıldığında geri vites bilgisi gövde kontrol modülüne (BCM) iletilmekte ve ayna camı otomatik olarak ayarlanarak kolay park edilebilecek bir ayna görüntüsü sunmaktadır.

b Akıllı farlar

Klasik far sistemlerinde aydınlatmayı sağlayan uzun ve kısa farlar ile aydınlatma mesafesi değiştirilebilmekte fakat aydınlatma istikametine tesir edilememektedir. Yeni teknolojiler ile otomobillerde kullanıma giren akıllı far uygulamalarında xenon lambalar ve XY-doğrultusunda hareket eden ve görüş açısını değiştiren teknolojiler görülmektedir. Akıllı far sistemleri sayesinde direksiyonda bulunan dönüş açısı sensöründen alınan dönüş bilgisi ve aracın hız bilgisini de değerlendirerek, dönüş yapan aracın dönüş merkezi istikametinde kalan ölü bölgelerin de aydınlatılması sağlanabilmektedir. Bu sayede daha güvenli bir sürüş ortamı hazırlanmaktadır.

c Park Pilot Sistemi

Bir çok sürücü için araç park etmek oldukça zor ve riskli bir iştir. Özellikle estetik ve aerodinamik sebepler ile geri görüşün iyice zorlaştığı yeni nesil araçlarda, park pilot sistemi konfor artırıcı bir donanımdır. Ses ikazlı, metraj görüntülü veya kamera destekli gibi değişik çeşitleri olan park pilot sistemleri üst seviyede sürüş konforu sağlamaktadır. Bir buton ile devreden çıkarılabilen sistem normalde sürekli devrededir. Özellikle kameralı modelleri ile araç kullanıcılarına; geri vites kullanımında rahat bir sürüş sağlanmakta, diğer zamanlarda da sürücü isterse park pilot sistem ekranını DVD olarak kullanabilmektedir.

d Çoklu Kullanıma Açık Şanzıman

Klasik manüel şanzıman mantığı ile hazırlanmış olan şanzıman gövdesine, bazı elektronik adaptasyonlar vasıtası ile otomatik şanzıman kullanım rahatlığı katılmıştır. Sistem şanzıman üzerine yerleştirilen elektronik sensör ve adaptasyon parçaları ile Şanzıman kontrol ünitesi arasında elektronik veri alışverişi sağlanması esasına dayanır. Aynı anda şanzımanla birlikte motorun bir çok noktasından Motor kontrol ünitesine gelen bilgiler de Şanzıman kontrol ünitesi ve Motor kontrol ünitesi arasında transfer edilmektedir. Eğer sürücü şanzımanda otomatik kullanım pozisyonunu seçmiş ise, tüm veriler Şanzıman kontrol ünitesinde değerlendirilerek, o anki şartlar için en uygun vites konumu elektronik olarak şanzıman tarafından gerçekleştirilecektir.

2.7.2.3 Diğer Amaçlar İle Kullanılan Yapay Zekâ Teknolojileri

a .Motor Kontrol Sistemi

Yapay Sinir Ağları kullanılan ve NeuroDyne tarafından geliştirilen sistemlerin bazıları şunlardır. Dizel motorlar için PM emisyonu veya NOx emisyonu gibi parametrelerin ölçümünün zor olduğu veya ölçülemediği gerçek zamanlı değerlerin elde edilerek Yapay Sinir Ağları ile Motorun kontrol edilmesi. Emisyonların tahmini ve gerçek zamanlı ölçümü için sensörler kullanılır. Motor, motor kontrolü ve yakıt yanma modellemede de Yapay Sinir Ağları kullanılır [45].

Bunun yanı sıra yapay sinir ağlarına dayalı zeki performans ve emisyon tahmin sistemleri otomotiv sektörünün son uygulamalarından biridir. Bu sistem bir prediktif motor modelidir ve bir mikro işlemci ve motorun gerçek zamanlı olarak birlikte paralel

alışması esasına dayanır. Giriş sinyalleri, motorda her defasında aynı sensörlerden Elektronik kontrol ünitesi (Mikro işlemci) tarafından alınır. SI motorlarda yapay sinir ağıları; manifold hava basıncı ve sıcaklığı, motor hızı, yakıt enjeksiyon süresi, EGR valfi değerleri, egzoz gazı oksijen konsantrasyonu, egzoz gaz sıcaklığı, Motor soğutucu sıcaklığını, kelebek pozisyonu ve ateşleme avansını kullanır. Elektronik kontrol ünitesine (ECU) iletilen sistem bilgileri, ECU tarafından değerlendirilerek, motora bağılı aktörlere bu değerlere uygun çalışmayı sağlayacak bilgi ve komutları aktarır. Aynı zamanda ilettiğinin yerine doğru ulaşp ulaşmadığını da geri dönüş sinyali ile teyit eder. Bu sayede sistemde meydana gelebilecek hatalarda otomatik olarak tespit edilmiş olur. Her hangi bir sebep ile sensörlerden birinin çalışmaması veya sürekli hata vermesi durumunda sistem otomatik uyarı sistemini aktif eder. Aynı zamanda motor çalışmasında herhangi bir aksaklığa sebep olmamak için sistemin öğrenmiş olduğu normal çalışma durum değerlerini esas kabul ederek, tüm durumlar için ortalama değerler atamak sureti ile motor çalışmasının devamını sağlar.

b Ara Diagnostik Sistemi

Elektronik teknolojisi ile donatılmış günümüz araçlarında teşhis ve diagnostik yapmak son derece zor ve karmaşık bir hal almıştır. Bu sebeple birçok araç üreticisi firma tarafından diagnostik cihazları geliştirilmiş ve kullanılmaktadır. Bilgi tabanlı sistemler olarak adlandırılan bu yapay zekâ teknolojileri günümüz araçlarının çoğunda motor kontrol, Süspansiyon kontrol, Patinaj kontrol ve ABS gibi ekipmanlarda kullanılmaktadır. Literatürde bilinen ve Fiat, Alfa Romeo, Peugeot ,Citroen, Renault, Chrysler gibi firmaların kullandığı başlıca diagnoz sistemler şunlardır. IDEA (Integrated Diagnostic Expert and Assistance), VMBD (Vehicle model Based Diagnosis), FTA (Fault Tree Analysis), FMEA (Failure-Model and Effect Analysis).

Frenleme sistemindeki hata ve arıza durumunda sürücüyü uyaran araca monte edilmiş ve yapay sinir ağıları kullanan diagnos sistemler de vardır.Bu sistemler özellikle ticari araçlar ve otobüsler için yaygın olarak kullanılır. Bu ve benzeri yapay sinir ağıları kullanan erken uyarı sistemleri bütün frenleme sistemlerinde kullanılır [46-54}.

2.7.3 Askeri Alanda Yapay Zekâ Uygulamaları

Bu gün, birçok askeri alanda yapay zekâ uygulamaları başlatılmıştır. Bu alanlardan bazıları şunlardır: Askeri araştırmalar, askeri imalat, bakım-onarım, hareket planlaması, lojistik, eğitim, istihbarat toplama ve işleme, istihbarat analizi ve durum tespiti, sensör kaynaklarının dağıtımı, kuvvet dağıtımı, kuvvet komuta ve kontrolü, güzergah planlaması, muharebe taktikleri, otonom / yarı-otonom araçlar, aviyonik, elektronik harp, ve komuta kontrol istihbarat karşı-koyma, haberleşme, ağ kontrolü, ve enformasyon yönetimi ve ulaşımı.

Ayrıca harp oyunlarında bilgisayarlarca oluşturulan CGF adı verilir. Eğitim ve analiz amaçlı olarak kullanılması ABD'de 1980'lerin sonlarında ciddi bir şekilde ele alınmaya başlamıştır. Bundan sonraki önemli gelişme, bilgisayarlı harp oyunlarına, gene ABD'de CFOR programıyla komuta-kontrol yeteneğinin eklenmesi olmuştur. Mevcut bilgisayarca oluşturulan kuvvetlerin gelişmesi dört safhada incelenebilir [55]. Bu safhaları şu şekilde sıralayabiliriz:

Nesil	Bilişim Süreci
1	Yok
2	Hedef tespiti ve önleme
3	Görevin kesilmesi ve başlatılması
4	Çok katmanlı Komuta Kontrol
5	Amaç seçimi ve Öğrenme

Birinci nesil CGF'lerin en bariz özelliği bunlarda bilişim sürecinin bulunmamasıdır. Bu nedenle, kendileri için oluşturulmuş olan senaryodan dışarı çıkamazlar. İkinci nesil sistemler ise, planlanmış faaliyetlerle çatışmayacak şekilde senaryo elemanlarına, hedef tespiti, nişan alma, ve rakip kuvvetlerle çatışmaya girme gibi basit etkileşimli davranışlar yaptırabilme özelliğine sahiptir. Güzergahlar ve yollar kullanıcı tarafından önceden veya senaryo sırasında belirlenir ve etkileşimler de ancak bunlar üzerinde olabilir.

Üçüncü nesil sistemler genellikle Yarı Otomatik Kuvvetler (SAF) olarak isimlendirilir. Bu sistemler genellikle, önceden planlanmış, kural veya durum tabanlı modüllerden oluşan görevleri uygularlar. Bunlarda davranışlar görev çerçevelerine yerleştirilir ve bu çerçeveler de öteki görev çerçevelerine oturtulur. Ana amaçlar böylece bir görev hiyerarşisinden meydana gelir. Bu organizasyon, karmaşık görev ve davranışların oluşturulmasını kolaylaştırmaktadır.

Dördüncü nesil sistemler ise bu gelişmiş üçüncü nesil sistemler üzerinde komuta kontrol (C2) süreçlerine sahip sistemlerdir. Görünüşte basit sanılan bu görev, savaş alanının en zor görevlerinden biridir. Ancak bu sistemlerin çoğunda sadece komutanın görevi temsil edilmektedir. Dördüncü nesil CGF'lere örnek DARPA'nın Komuta Kuvvetleri (CFOR) programıdır. CFOR, C2 süreçlerini, komuta kademesinin bir dizi davranış ve etkileşimi şeklinde temsil eder. Dağıtık Etkileşimli Benzetim (DIS) ortamları dördüncü nesil sistemleri bireyler ve birlikler düzeyinde temsil edebilmektedir, fakat mevcut durumda bunlar taburdan daha yukarı kuvvetlerin temsilinde yetersiz kalmaktadır. Ordular düzeyinde durum bunun tam tersidir. Üst düzey birimler başarılı bir şekilde temsil edilebilmekte, ancak aynı sistem içinde daha alt birimlere ve senaryo elemanlarına doğru gidildiğinde sorunlar ortaya çıkmaktadır.

Bir CGF sisteminin gerçek değeri, bunun temsil ettiği davranışların tümü ile ölçülmektedir. Davranışların modellendirilmesi ise en zor işlerden biridir. Bunu kolaylaştırmak için davranış süreçlerinin ve modellerinin kodlanmasını standartlaştırmak gerekmektedir. Ortak ve uyumlu bir dil geliştirme, gelecek nesil CGF sistemlerinde davranış modellendirilmesi konusunda karşılaşılabilecek sorunların bir kısmına çözüm getirebilecektir.[58]

- a. Hedef Tipinin Belirlenmesi (Yapay sinir ağları): Gürültülü çevrelerde hedef tipinin belirlenmesi askeri amaçlı uygulamalarda çok önemli bir problemdir. Uçak, gemi, kara araçları gibi radar hedefleri elektromanyetik bir dalga tarafından uyarıldıkları zaman, saçılan alan ilgili hedeflerin karakteristik özelliklerini gösterir.
- b. Hedef tanıma ve takip sistemleri (Yapay sinir ağları)
- c. Yeni sensörlerin performans analizleri (Yapay sinir ağları)

- d. Radar ve görüntü sinyalleri işleme (Yapay sinir ağları)
- e. Sensör fizyonu (Yapay sinir ağları)
- f. Askeri uçakların uçuş yörüngelerinin belirlenmesi (Yapay sinir ağları)
- g. Mayın dedektörleri (Yapay sinir ağları)
- h. Silah kullanımı, hedef belirlenmesi, nesneler arasında en iyisini seçebilme yeteneği, sima belirleme, sensör, sonar, radar ve belgelerin karşılaştırmasını içeren görüntü sinyallerinin yeni çeşitleri, dış görünüşü ayırt etme ve gürültüyü sindirme , sinyal/görüntü tanımlama (Yapay sinir ağları)
- i. Uçaklardaki otomatik pilotların yüksek performansı, uçuş yolu taklitleri, uçak kontrol sistemleri, otomatik pilot özelliklerinin artması, uçak bölümlerinin taklitleri, uçak bölümlerinin hata dedektörleri (Yapay sinir ağları)
- j. Akıllı füzeler

2.7.4 Tıp Ve Sağlık Alanında Yapay Zekâ Uygulamaları

- a. **Tıpta Tanı (uzman sistemler):** Farklı konularda Tıbbi tanılar koyan sistemler (dahiliye, akciğer, bulaşıcı kan hastalıkları vb.).
- b. **YSA Tıp ve İlaç Sanayi:** YSA tıbbın ilgi alanlarına bugünlerde oldukça fazla girmektedir. Önümüzdeki birkaç yıl içinde YSA'nın biyomedikal sistemlerde yoğun bir şekilde uygulanacağına inanılmaktadır. Şu anda ki çalışmalar insan vücudunun parçalarını modelleme ve çeşitli taramalarla hastalık teşhisi üzerine yoğunlaşmıştır.YSA, taramalarla hastalık teşhisi yapmak için idealdir. Çünkü; hastalıkların nasıl tanımlanacağı hususunda özel algoritmalar üretmeye gerek yoktur. YSA örneklerle öğrendiği için hastalıkların ihtiyaç yoktur. Gerekli olan, bütün hastalık çeşitlerini gösterebilecek örnek bulmaktır.Örneklerin sayısı kalitesi kadar önemli değildir. Sistemin güvenilir ve etkili bir biçimde çalışabilmesi için örneklerin çok dikkatli seçilmesi gerekir.

- c. **Tıp Teşhisi Yardımcı Öğeleri (Medical Diagnostic Aides):** YSA'nın kalp krizi teşhisinde kullanılmasının doktorlardan daha iyi bir sonuç alması itibariyle Wall Street Journal 'da ele alınmıştır. Bu uygulama önemlidir çünkü doktorların yüklü miktarda bilgiyi yorumlaması gereken acil durum odasında kullanılmıştır. Piyasada bulunan ticari bir ürün, sıvı örneğini inceleyerek meme kanserinin teşhisinde YSA teknolojisini sunar.bu ürünün klinikteki kullanımında, sıvı örneği teşhisinde insanlardan çok daha üstün olduğu kanıtlanmıştır. İngiltere de, miyokart enfarktüsünün erken teşhisinde kullanılan YSA tekniğinin halen dört hastanede klinik testleri sürmektedir.arastırma düzeyinde, epilepsi, akciğer hastalıkları, koroner atardamar hastalıkları ve kalp spazmı gibi hastalıkların teşhisinde YSA kullanılmaktadır. Elektrokardiyografi (ECG) ve Elektroenseflografi (EEG) tekniklerinin yorumlanmasında bu teknoloji kullanılır.
- d. **Kalp ve Kan Damar Sistemlerini Modelleme (Modelling and Diagnosing Cardiovascular System):** YSA'lar, insanın kardiyovascular sistemini modellemede bilimsel olarak kullanılır. Teşhis; bireyin kardiyovascular sistem modelini inşa ederek ve bunu hastalardan alınan gerçek zamanlı psikolojik ölçülerle karşılaştırarak bu teşhis başarılı.Eğer bu işlem düzenli olarak yapılırsa potansiyel hastalık sebeplerine erken teşhis konulabilir.Ve bu hastalıklarla mücadele işlemini kolaylaştırır.
- e. **Elektronik Koku Alma (Electronic Noses):** Elektronik Koku Alma (EKA), YSA uygulama alanlarındandır. EKA'nın telemedicine'de uygulanır. Telemedicine; iletişim bağıyla uzak mesafeden tıp uygulamasıdır. EKA'lar uzaktan tedavi yapılan bir ortamda kokuları tanır.Ve tanınan bu kokular elektronik olarak kokunun tekrar tanınabileceği bir sistemin olduğu bir yere gönderilebilir. Çünkü koku algılama tıpta önemli bir yere sahiptir. Bu durumdan, koku algılama ile uzaktan tedavinin çok daha gelişeceğini söyleyebiliriz.
- f. **Anlık Doktor (Instant Physician):** 1980'lerin ortasında geliştirilmiş uygulamaya verilmiş bir isimdir. Bu uygulama da, çok büyük

miktarlardaki bilgiyi saklamak için verileri bilgileri birbirleriyle ilişkilendirebilen bir sinir ağı yapılmıştır. Bu sinir ağının her bir ağ katmanında bulgu, tanı ve bazı özel durumlar tedaviler hakkında bilgi içermektedir. Böyle bir sistem oluşturulduktan sonra, bulguları içeren bir ağ hazırlanabilir.

- g. **Biyokimyasal Analiz (Biochemical Analysis):** YSA çok çeşitli analitik kimya uygulamalarında kullanılmaktadır. Tıpta YSA'ları kan ve idrar örneklerini analiz etmede şeker hastalığında glikoz seviyesini belirlemede vücut sıvılarında iyon seviyelerini saptamada ve verem gibi patolojik durumları tespit etmede kullanılmaktadır.
- h. **Tıbbi İmaj Analizi (Medical Image Analysis):** YSA'lar çok çeşitli görüntüleme modellerindeki tıbbi imajları analiz etmede kullanılır. Bunalandaki uygulamalar ultrasonogram'da tümör tespiti, monogramlarda meme belirleme ve sınıflandırma, göğüs filmlerinde sınıflandırmada dahil olmak üzere pek çok amaç için kullanılır.
- i. **İlaç Geliştirme (Drug Development):** Milli sağlık kuruluşlarındaki araştırmalar AİDS ve Kanseri tedavi etmek amacıyla ilaç geliştirme sürecinde YSA'ları kullanılmaktadırlar. YSA'lar ayrıca biomolekülleri de modelleme sürecinde de kullanılır.
- j. **Solunum Hastalıklarının Teşhisi (Yapay Sinir Ağları)**
- k. **EEG ve ECG Analizleri (Elektrokardiyografi(EEG) ve Elektroenseflografi (EEG)) (yapay sinir ağları)**
- l. **Hastalıkların Teşhisi Ve Resimlerden Tanınması (Yapay Sinir Ağları)**
- m. **Tıbbi Resim İşleme (Yapay Sinir Ağları)**
- n. **Transplant Zamanlarının Optimizasyonu (Yapay Sinir Ağları)**
- o. **CTG İzleme (Yapay Sinir Ağları)**
- p. **Hamile Kadınların Karınlarındaki Çocukların Kalp Atışlarının İzlenmesi (Yapay Sinir Ağları)**

- q. **Üroloji Uygulamaları (Yapay Sinir Ağları)**
- r. **Organ Nakli Zamanını En İyi Düzeye Getirmek (Yapay Sinir Ağları)**
- s. **Hastane Ödemelerinin Azaltılması (Yapay Sinir Ağları)**
- t. **Hastanelerdeki Kalitenin Artması (Yapay Sinir Ağları)**
- u. **Acil Durumlarda Kullanılan Odalar için Test Tavsiyeleri (Yapay Sinir Ağları)**
- v. **Kanser Hücrelerinin Analizi (Yapay Sinir Ağları) Protez Dizaynları (Yapay Sinir Ağları) [20].**

2.7.5 Akıllı Ev Teknolojileri ve Akıllı Ev Aletlerinde Yapay Zekâ Uygulamaları

- a. **Bulaşık makinesi:** Bulaşıkların sayısı ve kirin miktarına göre yıkama ve parlatma stratejilerini belirlemek.
- b. **Buzdolabı:** Kullanıma göre soğutma ve dondurma sürelerini belirlemek. (Bir nöral ağ, kullanıcının alışkanlıklarına göre nispi kuralları tespit etmektedir).
- c. **Çamaşır makinesi:** Kirlilik seviyesi, çamaşır türü ve miktarı ile su seviyesine göre yıkama stratejisini belirlemek. Bazı modeller, kullanıcıların arzularına göre kuralları ayarlamak için nöral ağlar kullanmaktadır. Girdileri karıştırmak ve sıcaklıkla denetimi süreyi ayarlamak.
- d. **Duş sistemi:** Su sıcaklığındaki değişiklikleri denetlemek.
- e. **Elektrikli süpürge:** Toz miktarı ve zemin türüne göre motorun emme stratejisini tespit etmek.
- f. **Fotoğraf makinesi:** Görüntünün herhangi bir yerindeki nesneyi bulup oto-fokus yapmak.
- g. **Kamera:** Oto-fokus ve ışığın ayarlanması. Elin titremesinden kaynaklanan görüntü bozukluklarını gidermek ve oto-fokusu temin etmek.

- h. **Klima:** İstenilmeyen sıcaklık oskilasyonunu önlemek ve açma-kapamada daha az enerji sarfetmek.
- i. **Kurutucu:** Çamaşırların türü ve miktarına göre kurutma stratejilerini ve süresini belirlemek.
- j. **Mikrodalga fırın:** Enerji sarfiyatı ve pişirme stratejilerini belirlemek.
- k. **Nemlendirme:** Oda şartlarına göre nem nispetini ayarlamak.
- l. **Pirinç pişirme aleti:** Buhar, sıcaklık ve pirinç miktarına göre pişirme süresi ve metodunu belirlemek.
- m. **Televizyon:** Her bir çerçeve için renk ve dağılımını ayarlamak ve odanın durumuna göre sesi stabilize etmek.
- n. **Tercüme programı:** Kelimeleri tanıyıp tercüme etmek.
- o. **Tost makinesi:** Her bir ekmek türü için, tost yapma süre ve sıcaklığını sağlamak.
- p. **Fırın:** Yemek pişme süresini ayarlamak, yemek cinsine göre gerekli sıcaklığı sağlamak [55-64].

Bulanık mantık esasına göre üretilmiş bazı ticari ürünler ve üretici firmalar aşağıda listelenmiştir.

ÜRÜN

ŞİRKET

Asansör denetimi

Fujitec, Mitsubishi Elektrik, Toshiba

Avuç içi bilgisayar

Sony

Bulaşık makinesi

Matsushita

Buzdolabı

Sharp

Çamaşır makinesi

Daewoo (Kore), Goldstar (Kore), Hitachi, Matsushita, Samsung (Kore), Sanyo, Sharp

Duş sistem

Matsushita, Panasonic denetlemek

Elektrikli süpürge

Hitachi, Matsushita, Toshiba

Fotoğraf makinesi	Canon, Minolta
Fotokopi makinesi	Canon
Kamera	Canon, Sanyo, Matsushita, Panasonic
Klima	Hitachi, Matsushita, Mitsubishi, Sharp
Kimyevi karıştırıcı	Fuji
Kurutucu	Matsushita
Mikrodalga fırın	Hitachi, Sanyo, Sharp, Toshiba
Nemlendirme	Casio
Ocak denetimi	Mitsubishi
Pirinç pişirme aleti	Matsushita, Sanyo
Plazma islemesi	Mitsubishi
Sağlık idaresi sistemleri	Omron
Televizyon	Goldstar, Hitachi, Samsung, Sony
Tercüme programı	Epson
Tost makinesi	Sony [65].

Akıllı bina sistemleri de son yıllarda gelişmeler göstermektedir. Günümüzde akıllı bina sistemleri birkaç bileşenden meydana gelmektedir. Bu bileşenler:

- a. HVAC (Isıtma-Soğutma, Havalandırma Sistemleri) Otomasyon Sistemleri
- b. Oda Kontrol Sistemleri
- c. Yangın Algılama & Alarm Sistemleri
- d. Kapalı Devre Televizyon Sistemi
- e. Güvenlik ve Erişim Sistemleri
- f. Güç ve Enerji Otomasyonu Sistemleri
- g. Aydınlatma Otomasyonu Sistemleri
- h. Data ve Haberleşme Sistemleri [66]

- i. Akıllı Termostatlar: Değişimi algılayan ve duruma göre devreye giren veya çıkan bir termostat, sürekli on-off işlemi yapabilen klasik termostattan daha iyidir. Bununla birlikte böyle bir “Akıl” programı mikroişlemciye konularak monte edilebilir ve bu çok kolaydır. Algoritma ampirik ölçümlere dayanır ve bir çok faktörler ve şartları gerektirir. Dolayısı ile bunun matematik modellemesi oldukça zordur. Bulanık Mantık kontrolü 2 noktada araya girer. Bulanık Mantık sisteminin çıkışlarından biri set sıcaklığını düzeltir diğeri ise histerisizin girişimini adapte eder [67].

2.8. Eğitimde Yapay Zekâ

Günlük hayatımızda her alanda karşımıza çıkan yapay zekâ teknolojilerinin eğitimdeki yeri ve önemini vurgulamadan önce teknoloji eğitiminin, bu eğitimin unsurlarının ve yapay zekâ tekniklerinin eğitimdeki uygulamalarının ele alınması gerekir. Teknoloji Eğitimi (TE), mühendislik eğitimi ile teknik eğitimin iç içe olduğu bir eğitim türüdür. TE, müfredatın teknolojik gelişmelere göre güncellenmesi gereken dinamik bir eğitim türüdür. İdeal bir teknoloji eğitiminde temel teorik derslerin yanı sıra laboratuvar uygulamaları, simülasyonlar, proje çalışmaları, verimli bir staj dönemi ve en güncel teknoloji olan yapay zekâ teknolojileri (YZT)’nin bu müfredat kapsamına ne şekilde yansıtılacağı konusu çok önemlidir. Öncelikle bu eğitimi verecek öğretmenler ile yapay zekâ teknikleri ile ilgilenen veya YZT konusunda çalışan profesyonellerin bir program çerçevesinde TE müfredatını ülkemiz koşullarına göre yeniden gözden geçirmeleri ve dünya standartlarını yakalamak için bu eğitimi verecek tüm disiplinlere yaygınlaştırmaları gerekir. Eğitimin teknoloji ile entegrasyonu denince ilk akla gelen Amerika, Japonya ve Avrupa’nın gelişmiş ülkelerinin yer aldığı coğrafyada uygulanan teknoloji eğitimidir. Bu aynı zamanda gelişmiş ülke olmanın doğal bir sonucudur. Ülkemiz gibi gelişmekte olan ülkelerde ise hedeflenen bir eğitim seviyesidir. TE, araştırma-geliştirme yapabilen, teknolojik gelişmelere ayak uydurabilen ve bilgilerini bu yönde güncelleştirebilen bireyi yetiştirmeyi hedefleyen bir eğitimidir. TE bu anlamda müfredatın teknolojik gelişmelere göre güncelleştirildiği, eğitimcilerin öğretmenlerin buna ayak uyduracak şekilde kurslarla yetiştirildiği, oryantasyon programları ve seminerlerle

desteklendiği bir eğitim türüdür. Böyle bir eğitim anlayışında son 20 yılın en önemli konularından birisi olması ve gün geçtikçe artan bir ilgi odağı haline gelmesinde yapay zekâ teknolojilerinin payı büyüktür.

Yapay zekâ teknolojileri, endüstri, otomasyon, tıp, eğitim, finans, robotik, otomotiv, savunma, eğitim, ev teknolojileri, güvenlik., v.b. pek çok alanda uygulanan en gelişmiş teknolojilerdir. Böyle bir teknoloji eğitiminin gerçek anlamda ele alınabilmesi için yapay zekâ tekniklerinin eğitimde kullanılması zorunludur. Yapay zekâ denilince akla ilk gelen uzman sistemler ve bunun eğitimdeki ilk kullanımı zeki öğretim sistemleridir (ZÖS) (intelligent tutoring systems-ITS). ITS (ZÖS) yapıları ve amaçlarına göre pek çok araştırmacının ilgilendiği ve sürekli gelişen bir alan olmuştur. Literatürdeki bilinen en önemli ITS yazılımları WEST (Burton and Brown, 1982) , SOPHIE (Brown, Burton and deKleer, 1982) Sleeman ve ITS alanında çalışanlar Brown (1982), Wenger (1987), Psotka, Massey ve Mutter (1988), Ohlsson (1986), Schank ve Edelson (1990). ITS, yüksek matematik konularında (Du and McCalla, 1991), fen bilimlerinde (Lester and Porter, 1991), tarih-dilbilimi-sosyal bilimlerde (Bruneau, Chambreuil, Chambreuil, Chanier, Dulin, Lotin and Nehemie -1991) tarafından yardımcı olacak şekilde kullanılmıştır. (Frederiksen, Donin, DeCary and Edmond-1991) ITS ile ikinci bir dil öğrenimini ele almıştır. ITS yüksekokullarda müfredat kapsamına alınarak elektronik, tamir-bakım ve arıza tespitinde kullanılmıştır (Cooper, 1991; Frederiksen, White, Collins and Eggan, 1988; and Kurland and Tenny, 1988) ve SOPHIE kullanılarak (Brown, Burton and deKleer, 1982) tarafından seminerler verilmiştir. Çoğu ITS azda olsa halen kullanılmakta olsa bile sadece birkaçı geniş olarak test edilmiştir. Bunun nedeni öğrencilere sınıfta çözdürülen testlerden kaynaklanmaktadır. Başarılı olan ITS'ler genelde matematik, fen ve yabancı dil alanlarıyla kısıtlıdır. Bu alanlarda öğrenme performansını ölçmek ve ITS hazırlamak çok kolay olması nedeniyle uygulamada en çok yoğunlaşılan ve üzerinde çalışılan alanlar olarak tercih edilmişlerdir [84-96].

En başarılı ITS, Anderson'un Geometrisi ve Lisp öğreticisidir [1][97]-[1][99]. SHERLOCK ise elektronik tamir bakımda kullanılan diğer bir başarılı ITS'dir. [1][100], [1][101]. Teknoloji eğitiminde böyle bir ITS yaklaşımı Yüksek Matematik, Fen ve yabancı dil derslerinde kullanılarak ve CLIPS, SHELL gibi uzman sistemler bilgisayar ortamında bir sonraki aşamada yapay zekâ teknikleri derslerine bir hazırlık olur. Ayrıca,

her iki aşamada bizlere çok yardımcı olacak ve bir ders kapsamında iki dönemlik MATLAB dersinin uygulamalı olarak okutulması vazgeçilmez bir zorunluluktur. Teknoloji eğitiminde yapılması gereken temel aşamalar şunlardır:

- Öğrencilerin kazandıkları bilgiyi pratiğe aktaracak şekilde yeni bir öğrenim tarzı.
- Öğrenci merkezli bir eğitim yönteminin izlenmesi ve öğreticinin burada rehber olarak görev yapmasıyla öğrenciye önem verilmesi. Geleneksel öğrenim yönteminin bu anlayışa göre yeniden yapılandırılması.
- Öğreticinin tecrübelerini aktarmada uygulamalı olarak teknolojik cihazları derslerinde kullanması ve bunu sağlayacak ortamların hazırlanması.
- Proje ve ödevlerle öğrencinin düşünme, araştırma ve uygulama yeteneğinin geliştirilmesi ve aynı zamanda öğrendiklerini pratiğe aktarması teşvik edilmelidir.
- Öğrencilerin dersi aldıkları dönemde uygulama yapabilecekleri yerlerde verimli staj yapmalarının sağlanması.
- Yukarıda belirtilen hedeflere ulaşmak için teknolojik gelişmelere uygun olan ve YZT içeren bir müfredatın hazırlanması ve güncelleştirilebilecek şekilde dinamik olması.
- Yapay Zekâ disiplinler arası bir alan olmasına rağmen yeni bir bilim dalı olarak periyodikler, konferanslar ve organizasyonlara sahiptir. Bunların en önemlileri aşağıda sıralanmıştır.
- Journals-Artificial Intelligence
- Journal of AI Research (JAIR)
- Computational Intelligence
- J. of Experimental and Theoretical AI
- Conferences & organizations-AAAI, IJCAI, ECAI, ACM SIGART olarak verilebilir.

Yapay Zekâ ile ilgili detaylı bilgi için önerdiğimiz adresler:

- o MIT Artificial Intelligence Laboratory - <http://www.ai.mit.edu/>
- o Ai Research Project - <http://www.a-i.com/>
- o The homepage of Ray Kurzweil - <http://www.kurzweilai.net/>
- o The homepage of Kevin Warwick - <http://www.kevinwarwick.com/>
- o The homepage of Hans Moravec - <http://www.frc.ri.cmu.edu/~hpm/>
- o The homepage of Rodney A. Brooks - <http://www.ai.mit.edu/people/brooks/>

2.9. Teknoloji Eğitiminde Yapay Zekâ

TE, araştırma geliştirme yapabilecek, teknolojik gelişmelere ayak uydurabilen ve bilgilerini bu yönde güncelleştirebilen bireyi yetiştirmeyi hedef alan bir eğitimidir. TE bu anlamda müfredatın teknolojik gelişmelere göre güncelleştirildiği, eğitimcilerin öğrencilerin buna ayak uyduracak şekilde kurslar, oryantasyon programları ve seminerlerle desteklendiği bir eğitim türüdür. Böyle bir eğitim anlayışında son 20 yılın en önemli konularından ve gün geçtikçe artan bir ilgi odağı olan yapay zekâ teknolojileri önemli bir etkidir. Önemli olan bu eğitimi veren disiplinlerde bu teknolojileri anlamaya yönelik yapay zekâ Teknikleri dersinin teorik ve uygulamalı olarak nasıl verilmesi gerektiğidir. Hangi branşlarda hangi tekniklerin ağırlıklı olarak öğretilmesi tüm branşlarda öğretilecek Genel yapay zekâ dersinin içeriği YÖK, Tübitak ve üniversitelerin YZ konusunda çalışan akademisyenleri tarafından belirlenen bir müfredatta gerçekleştirilmesi ve bu müfredatın ilgili sektörlerdeki profesyonel kadrolar tarafından üniversitelerde verilecek seminer, konferans ve proje çalışmaları ile desteklenerek son dönemlerin ülkemizdeki en öncelikli hedeflerinden biri olan okul-sanayi işbirliğinin gerçekleştirilmesidir. Eğer, TE gereği gibi verilirse mezun olan teknik branşlardaki öğrencilerimiz (mühendisler, teknik fakülte mezunları,...) iş hayatına hazır olarak atılacak ve bunlar için tekrar eğitim programlarına harcanan para teknoloji eğitime destek kapsamında okullara aktarılacaktır. Yani özel sektördeki yeni mezun bir mühendise verilen eğitimi verilmesi gereken esas yer olan üniversitelere getirmek TE'nin hedeflerindendir. Bir başka deyişle, öğrenciye imkansızlıklar nedeniyle veremediğimiz

teknoloji desteğini bir eğitim çerçevesinde değerlendirerek adı üzerinde teknoloji eğitimi vermektir. Böyle bir eğitim anlayışında yapay zekâ teknolojileri en önemli ve en güncel bilgidir. Her gün medyadan takip ettiğimiz zeki çamaşır makineleri, buzdolapları, öğretim sistemleri gibi kavramların mantığını lisansta anlatmış olursak Yüksek lisans ve Doktora seviyesinde yada özel sektörde benzeri bir proje yada araştırma çalışmasında daha aktif rol almanın yanında ileri götürülmesi anlamında da çalışma yapabilecek bilgiye sahip olur.

Yapay zekâ teknolojilerinin günlük hayatımızda endüstriyel otomasyondan, otomotiv sektörüne, finans sektöründen, medikal sektöre, robotikten eğitim sektörüne kadar ülkemizde pek çok alanda da uygulanmaktadır. Bu tip uygulamalar genelde ya ithal teknoloji kullanılarak yada ilgili sektörlerdeki müteşebbisler tarafından Ar-Ge bölümlerinde özel eğitimlere tabi tutulan mühendisler, matematikçiler ve tasarımcılar tarafından gerçekleştirilmektedir.

Uzmanlık isteyen ve ülke ekonomisine büyük yük getiren yetişmiş insan ihtiyacının lisans ve lisans üstü düzeyinde yurt dışında olduğu gibi okul-sanayi işbirliği çerçevesinde giderilmesi çok önemlidir. Bu ihtiyacı karşılamada ilgili branşlarda lisans seviyesinde 3. ve 4. sınıflarda yapay zekâ tekniklerinin öğretilmesi ve lisans üstü seviyesinde de projeler geliştirebilecek şekilde yaygınlaştırılması, ülkemizin önümüzdeki yıllarda eğitimden endüstriye kadar en güncel yapay zekâ teknolojileri uygulayabilen modern ve gelişmiş bir ülke olmasını sağlayacaktır.

Bu ideal ancak yapay zekâ teknolojilerinin teknoloji eğitimindeki yeri ve öneminin anlaşılmasına ve bu konudaki çalışmaların yerel ve merkezi otoritelerce desteklenmesi ile gerçekleştirilebilecektir. Uygulama alanlarına göre yapay zekâ tekniklerinden hangisinin hangi branşlarda öğrenilmesi gerektiği, bu ders kapsamında öğrenilmesi gereken programlama dilleri ve gerekli yazılımlar yukarıda bahsedilen müfredata bir örnek teşkil etmesi açısından aşağıdaki tabloda verilmiştir. (Tablo 5)

Tablo 5 Uygulama alanlarına göre farklı branşlardaki yapay zekâ teknikleri

Branş	Genel YAPAY ZEKÂ	Lisansla önerilen YZT	Yazılımlar	Öğrenilecek
Elektronik	Tanım,Tarihçe,Uygulama Alanları,Amaçları, Kapsamı	Bulanık Mantık, Uzman Sistemler, Yapay Sinir Ağları	Matlab , CLIPS, Neuro Solutions, Fuzzy Tech.	Prolog , Lisp, C/C++, Pascal
Bilgisayar	Tanım,Tarihçe,Uygulama Alanları,Amaçları, Kapsamı	Sürü Algoritmaları, Yapay Sinir Ağları, Uzman Sistemler, Bulanık Mantık, Genetik Programlama	Matlab , CLIPS, Neuro Solutions, Fuzzy Tech.	Prolog , Lisp, C/C++, Pascal
Makine	Tanım,Tarihçe,Uygulama Alanları,Amaçları, Kapsamı	Genetik Algoritmalar, Uzman Sistemler, Bulanık Mantık	Matlab , Fuzzy Tech.	C/C++, Pascal
Biomedikal	Tanım,Tarihçe,Uygulama Alanları,Amaçları, Kapsamı	Yapay Sinir Ağları, Uzman Sistemler	Matlab , Neuro Solutions, CLIPS.	C/C++, Pascal
Jeoloji-Maden	Tanım,Tarihçe,Uygulama Alanları,Amaçları, Kapsamı	Yapay sinir ağları, Bulanık Mantık	Matlab, Neuro Solutions, GIS, Fuzzy Tech	C/C++, Pascal
Matematik	Tanım,Tarihçe,Uygulama Alanları,Amaçları, Kapsamı	Bulanık Mantık, Uzman Sistemler, Yapay sinir Ağları	Matlab , CLIPS, Neuro Solutions, Fuzzy Tech.	C/C++, Pascal
Finans	Tanım,Tarihçe,Uygulama Alanları,Amaçları, Kapsamı	Sürü Algoritmaları, Genetik Algoritma, Yapay Sinir Ağları	Matlab , CLIPS, Neuro Solutions, Fuzzy Tech.	C/C++, Pascal
Trafik	Tanım,Uygulama Alanları,Amaçları, Kapsamı	Bulanık Mantık, Uzman Sistemler,Yapay Sinir Ağları, Genetik Algoritma	Matlab,Vissim, Transyt, Fuzzy Tech., Neuro Solutions.	Matlab, C/C++, Pascal

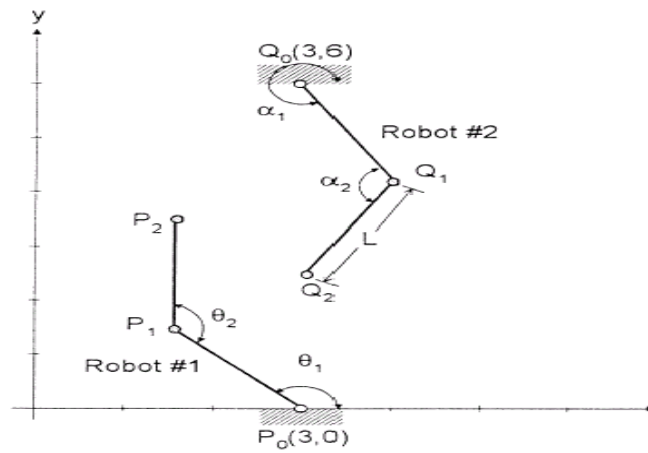
Yapay zekâ Teknikleri dersini 3 yıl boyunca Lisans ve Lisans üstü seviyesinde Matematik, Makine ve Elektrik branşlarında verdiğimiz dönemde öğrencilerden aldığımız izlenim şudur. Bu dersin daha detaylı ve uygulamalı bir şekilde branşa yönelik işlenmesi teknoloji eğitiminin adına uygun olacaktır. Bu eğitim simülasyonlarla, branşa özel yazılımlarla diğer teknolojik ekipmanlarla desteklenmesi teknoloji eğitimi açısından önemlidir. Bu kapsamda dersimizde delphi ile geliştirdiğimiz YSA Simülatörü, Genetik Simülatör ve MATLAB'ın demoları kullanılmıştır. Bu çalışmada vurguladığımız teknoloji eğitimi, yapay zekâ teknikleri dersinin simülatörler ve yazılımlarla desteklenmesi durumunda çok daha etkili olacaktır.

Bir elektronik, bilgisayar, makine...v.b. branşlarda eğitim alan bir mühendis veya teknik öğretmen adayı yapay zekâ teknikleri dersini lisan eğitiminin 3. ve 4. yılında uygun bir müfredat çerçevesinde alırsa, çalışacağı sektöre gitmeden önce ufku açılacaktır. Bu eğitimi lisansüstü ve doktora seviyesinde devam ettirirse çok güzel projelere ve akademik çalışmalara imza atabileceklerdir. Ülkemizin en önemli eksikliklerinden biri

teknoloji eğitimindeki müfredatın gelişen teknolojiye ayak uyduramamasıdır. Bu açığın, dünyada son yıllarda gerçek hayatta her alanda uygulama alanları bulan yapay zekâ teknolojilerinin ülkemizde de teknoloji ithali ile değil de bu alanda yetiştireceğimiz elemanlarla kapatılması elzemdir. Ülkemiz gibi gelişmekte olan ve çok büyük bir genç nüfus potansiyeli olan bir ülkede orta vadeli bir teknoloji eğitiminin yapay zekâ teknikleri dersinin lisansın 3. ve 4. sınıflarında ve lisansüstü seviyede teorik ve uygulamalı olarak okutulması gerçekleştirilmesi sadece önemli değil bir zorunluluktur.

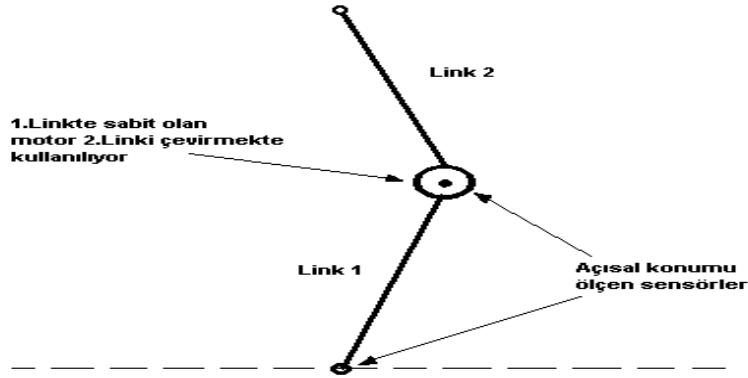
2.9.1 Robotik

- Duvarı Takip Eden Robot (genetik algoritma): Bu robotun amacı bir duvara rastlayınca kadar boş alanda hareket etmek ve duvara rastlayınca da belli bir mesafeden duvarı takip ederek ilerlemektir. Robotun başarılı sayılabilmesi için bulunduğu ortamın çevresini en azından bir kez çok yakın veya çok uzak kalmaksızın takip etmesi gerekir [60]
- Çok Kollu Robotların Çarpışmasız Hareketi (genetik algoritma): Bu çalışmada birden fazla koldan oluşan robot sistemin sabit engellere çarpmadan hareketinin yanı sıra hareketli çevre ve engellerle de çarpışmadan hareket etmesi sağlanmaya çalışılmıştır. Yörünge denklemleri dizilere çevrilmiş ve böylece GA çarpışmasız minimum hareket yolunu hesaplamak için kullanılmıştır.[61]



Şekil 6 Problemde Çarpışmasız Hareketi İncelenen İki Linkli Düzlemsel İki Robot

- Robot Eli (genetik algoritma): Bu çalışmada insan eli benzeri beş parmaklı bir robot elinin bir nesneyi kavraması için gereken hareketler incelenmiş ve bu karmaşık problemin çözümünde GA kullanılmıştır. Bu problem çarpışmasız hareket yörüngesi saptamaya benzemektedir. Her bir parmak diğeri için çarpırmaması gereken bir engeldir. Uyumluluk her bir parmağın kontak noktasına (nesneye dokunduğu nokta) olan uzaklığı, stabilite, manipulasyon ve çarpışmasız hareket göz önünde bulundurularak hesaplanmaktadır [62].
- Akrobat(genetik algoritma): Akrobat adından da anlaşıldığı gibi bir akrobatın hareketini simule etmektedir. Aslında iki linkli düzlemsel bir robottur. Elleriyle barfiks tutunup baş aşağı dengede duran bir akrobatın yaptığı hareket akrobata yaptırılmaya çalışılmıştır. Kullanılan tek motor birinci link ile sabittir ve 2. linki hareket ettirmektedir.[63]



Şekil 7 Akrobat

- Robot sensörleri (genetik algoritma)
- Yörünge kontrolü, ağır eşyaları kaldıran robotlar, yönetme kumandaları, görüntü sistemleri (yapay sinir ağları)
- Robotlarda görme sistemleri (yapay sinir ağları)

Robot teknolojisi çok hızlı bir gelişim göstermektedir. Asimo, aibo (oyuncak köpek) vb. gibi robotlar artık hayatımızın birer parçası olmaya başladılar. Robotların

mücadele ettiği Robocup isimli turnuvalar düzenleniyor. Bu ve buna benzer robot örneklerini aşağıda görebiliriz.

Stuttgart Üniversitesi'nin Paralel ve Dağıtılmış Yüksek Performans Bilgisayarları Enstitüsü'nde Prof. Paul Levi yönetiminde bir çalışma gurubu Aramis (adını monte edilmiş olan kolundan alıyor), Porthos (yük taşıyıcısı) ve Athos (bir stereo kameraya sahip ve gurubun gözcüsü) isimli üç robot üretmiştir. Bu robotlar küçük sorunlarını tek başlarına çözebilmektedir. Fakat bu robotlarda diğerlerinde olmayan bir özellik vardır, kooperasyon yeteneği. Şöyle ki; kimin hangi görevi hangi sırayla yapacağını aralarında kararlaştırıyorlar. Bunu konuşarak yapmaları teknik bir dayatmadan çok araştırmacıların oyun dürtüsüne işaret etmektedir. Aslında makineler bit ve byte'lar düzleminde anlaşmalarına rağmen, çalışma esnasında kadın ve erkek sesleriyle gerçekleşen sözlü diyaloglar ortaya çıkmaktadır. Prof. Levi'ye göre üç silahşorlar, günün birinde temizlik, nakliyat ve konstrüksiyon ile ilgili görevleri yürütecek bir robot kuşağının prototipleridir.

Bir başka örnek ise MIT'den Rodney Brooks'un tasarladığı ATTİLA isimli böcek robot. 30 cm. boyutundaki bu robot üzerinde 23 motor, 10 mikro işlemci ve 150 adet algılayıcı bulunuyor. Her bacağın üç bağımsız hareketi sayesinde engellerin üstüne tırmanıyor, dik inişler yapıyor ve tutunarak kendisini 25 cm. yüksekliğe çekebiliyor. Brooks'un yapay zekâ anlayışında izleme, avlanma, ileri gitme ve gerileme gibi bir takım ilkel içgüdü ve refleksler yer alıyor. Öte yandan onun robotlarında bunları seçen ve bu basit hareketleri yönlendiren bir beyin modeli yer almıyor. Bunun yerine, her davranış, robotun kontrolünde yarışan bireysel zekalar olarak işliyor. Kazananı, robotun alıcılarının o anda ne hissettiği belirliyor ve bu noktada diğer tüm davranışlar geçici olarak bastırılıyor. Kurulan mantıkta, “gerile” gibi tehlikeden sakınma davranışları, “avı izle” gibi daha üst seviyedeki fonksiyonları bastırıyor. Davranış hiyerarşisindeki her seviyenin gerçekleşmesi için bir alttakinin aşılması gerekiyor. Böylece bir böcek robot, örneğin “odadaki en uzak köşeyi belirle ve oraya git” gibi yüksek düzeyde bir komutu, bir yerlere çarpıp başına kaza gelme korkusu olmadan yerine getirebiliyor.

Mikro Robotlar: Robot teknolojisinde robotların fonksiyonları geliştikçe boyutlarının da küçüldüğü görülmektedir. Önceleri oldukça büyük boyutlarda olduğu halde bir çok fonksiyonu yerine getiremeyen robotlar var iken günümüzde robotların

boyutları küçülmekte ve fonksiyonları ise artmaktadır. Bu sürecin devam edeceğini görmekteyiz [25]. Önümüzdeki yıllarda robotların iyice küçülerek kibrit kutusu büyüklüğünde robotlara kadar inileceği görülmektedir. Robotların boyutlarının küçülmesinin ne gibi bir faydasının olacağı sorulabilir. Bu robotlar özel amaçlı işlerde kullanılabilir.

Asimo: Robotun hareket kabiliyeti çok ileri düzeyde. Akıllı, gerçek zamanlı, esnek yürüyüş teknolojisi Asimo'nun yön değiştirirken duraksamadan yürümeyi sürdürmesini ve ani hareketlerde dengesini korumasını sağlıyor. Hafızasında kayıtlı yürüyüş tarzlarının çeşitli kombinasyonlarıyla yürüyebilen Asimo, köşelere geldiğinde kendiliğinden yön değiştirebiliyor, yürüyüş hızını azaltıp, çoğaltabiliyor.



Şekil 8 Bir robot turnuvasından görüntü

2.10. Yapay Zekânın Ticari Uygulamaları

- a. Elektronik denetim sistemleri (bulanık mantık)
- b. Karar verme (bulanık mantık)
- c. Proses planlama (bulanık mantık)
- d. Fiber-optik kablo döşeme şehirlerarası (Genetik algoritma)
- e. Devre tasarımı: (Gen alg – gezgin satıcı problemi)

- f. Optimizasyon: Genetik algoritma, sayısal optimizasyon ve kombinatoral optimizasyon problemleri olan devre tasarımı, doğrusal olmayan denklem sistemlerinin çözümünde ve fabrika-üretim planlamasında kullanılır.
- g. Tren Frenleme: Örneğin, otomatik bir trenin istasyona ulaşması.Onu durdurmak için kesin mantık kullanılabilir.

Eğer<tren<istasyondan 50 m > ise <frene basar>

Ani bir fırlamayla 50 m ilerler.

Bulanık mantıkta;

Eğer<tren istasyona yakın> ise <tren yavaşlar> “Yakın” bulanık bir yüklem ve “yavaş” bulanık bir komuttur. Frene basınçlı basmak ise “yakın” ifadesinin doğruluk değerine, hıza ve yavaşlamaya bağlıdır.

Bu bulanık mantık, trenin durmasını ve istasyonu problemsiz terketmesini sağlar. (ilk Japanyo’da kullanılmıştır.)

- h. Karmaşık elektronik ve elektromekanik sistemlerin tamiri ve tasarımı, (uzman sistemler)
- i. Dizel ve elektrikli hareket eden sistemlerin tamiri (uzman sistemler)
- j. Bilgisayar ve iletişim sistemlerinin tasarımı ve hatalarının bulunması (uzman sistemler)
- k. Uzayla ilgili bir çok uygulama (planlama ve inceleme) (uzman sistemler. Çeşitli kontrol sistemleri (Demir yolları sinyalizasyonu ve hemzemin geçit kontrolü gibi)
- l. Web Tabanlı Arama Motorları (Yahoo,Google...) (uzman sistemler)
- m. Ev Temizlik Robotları (uzman sistemler)
- n. Telekomünikasyon alanında sinir ağı uygulaması
- o. Elektronik: Kod sıralaması tahmini, çip devrelerinin birleştirilmesi, yöntem kontrolü, çiplerdeki bozulmanın analizi, makinelerin görünümü, ses sentezleme, lineer olmayan modelleme

- p. Endüstri: Endüstride ilerleme kontrolü, ürün dizaynı ve analizi, yöntem ve makine tanınması, parça tanımlaması, görünüş kalitesini denetleme sistemleri, bira testleri, kaynak yapımında kalite analizleri, kağıt kalitesi tahmini, bilgisayar çipi kalite analizi, öğütme (parçalama) yöntemleri analizleri, kimyasal ürün dizayn analizleri, makine bakım analizleri, proje fiyatlandırması, planlama ve yönetme, kimyasal yöntemlerin sisteminin dinamik modellemesi
- q. Telekomünikasyon: Görüntü ve bilgi karşılaştırması, otomatik bilgi servisleri, konuşulan dili aynı anda başka bir dile çevirme, tüketici ödemeleri sistemleri
- r. Bir endüstriyel proseste fırınların ürettiği gaz miktarının tahmini
- s. İmalatta ürün tasarımı, proses ve makinelerin bakımı ve hataların teşhisi görsel kalite kontrolü
- t. Kimyasal proseslerin dinamik modellenmesi
- u. Cep telefonlarında ses ile çalışabilme
- v. Elektronik yonga hata analizleri
- w. Optimizasyon çalışmaları (üretim planlama ve kontrol çalışmalarında)
- x. Kömür güç istasyonları için çevrimiçi karbon akımı ölçülmesi [102-105].

3. BÖLÜM : Bulanık Mantık (Fuzzy Logic)

3.1. Giriş

Fiziksel sistemleri matematiksel olarak modellerken, transfer fonksiyonlarını çıkarırken, sistemlerin doğrusal ve zamanla değişmeyen sistemler olduğunu kabul ederiz. Oysa doğada doğrusal sistem pek yoktur. Bu kabullenmeyi belirli çalışma bölgeleri etrafında kabul edebiliriz. Bunların dışında matematiksel modelinin çıkarılması oldukça karışık hatta imkansız çok sayıda matematiksel işlemler kabullenmeler gerektirir [106]. Bir insanın zihnindeki düşünce dünyasının bile tomografisi çekilecek olursa, bunun çok renkli değişik hatta karmaşık motiflere sahip olduğu çok belirgin olmayan desenler içerdiği görülür. İşte bu belirsizliği, bulanıklılık (fuzzy) diye tanımlamak mümkündür [107]. Bulanık Mantık diğer adıyla “*Fuzzy Logic*” kuramı ilk kez 1965 yılında Lotfi Zadeh tarafından ortaya atılmıştır. Kümeler teorisinde bir eleman ya bir kümeye aittir yada değildir. Fakat bulanık kümelerde bir eleman birden fazla kümeye ait olabilmektedir. Bulanık kümelerde kesinlik kavramı yoktur[108]. Bulanık mantık (Fuzzy) karar verme mekanizması olarak ta tanımlanabilecek sözel ifadelerin bir uzman kişi tarafından belirtilen kesin olmayan sınırlar içindeki davranışını matematiksel olarak modellemeye yarar. Modelleme kesin olmayan bulanık kümelerden oluştuğundan *Bulanık* yada *Fuzzy* olarak ifade edilir. Bu ismi kişisel yada uzman kişinin kesin çizgilerle ifade edemediği ancak bölgesel olarak yaklaşık sınırlarının belli olduğu durumlarda anlamlı sonuçlar vermektedir.

Bulanık mantık, hesaplama tekniği bu tür sorunları büyük ölçüde çözebilmektedir. Bu nedenle bilinen geleneksel hesaplama yöntemlerine alternatif olarak ortaya çıkan bu yöntem, doğadaki işleyişi taklit ederek çözüme ulaşır. Bulanık mantık kavramı iki temel öğeden oluşur;

- 1- Bulanık kümeler ve bu kümeleri kullanarak bir dizi kural oluşturma
- 2- Karar verme süreci.

Bulanık mantık kuramının uygulamaları, günümüzün karmaşık problemlerinin çözümünde kullanışlı bir araç haline gelmiştir. İlk ortaya atıldığı tarihten bu yana konu,

matematikçiler, bilim adamları ve mühendisler tarafından birbirinden bağımsız pek çok çalışmaya konu olmuştur [106]. Bulanık Mantık'ın kurucusu Azeri asıllı Lotfi Zadeh'dir.

3.1.1 Bulanık Mantık 'ın Tarihçesi ve Kurucusu Lotfi Zadeh

Matematiğin doğruluğundaki ve bütünlüğündeki başarısı Aristoteles'in ve öğrencilerinin katkısıyla olmuştur. Onların mantık teorisini oluşturma çabaları ile matematik bilimi gelişmiş ve "Düşüncenin Yasaları" oluşturulmuştur. Bu yasalardan biri, her önermenin ya doğru yada yanlış olması gerektiğini belirtmektedir. Bu anlayışa geleneksel anlayış yada Aristo Mantığı denilmektedir [109, 111]. Bu yüzyılda matematik ve bilimdeki değişik fikirler belirsizlik kavramını da etkilemiştir [110, 111]. 1900'lerin başında, geleneksel anlayıştan farklı olarak Polonyalı mantıkçı Jan Lukasiewicz'in 3. bir değer olan "olası" kavramını ortaya atması, 1920'ler ve 1930'larda çok değerli mantık sisteminin gelişmesine yol açmıştır. 1930'larda Max Planck tarafından belirsizliği açıklayıcı ilk kavramlar geliştirilmiş ve 1965'de Azeri kökenli sistem bilimci Zadeh tarafından yayımlanan makale ile modern anlamda belirsizlik kavramı açıklığa kavuşmuştur [107, 109-112].



Bulanık mantık 'ın kurucusu, Lütü Zade, tam adıyla Lütü Rahim oğlu Askerzade (Bakü 4 Şubat 1921) dir. Lütü Zadeh; Azerbaycan asıllı matematik ve bilgisayar biliminde çalışan bulanık mantık teorisinin temelini atan bilim adamıdır. Kaliforniya Üniversitesi, Berkeley'nin Elektrik Mühendisliği ve Bilgisayar Bilimleri fakültesinde profesör olarak görev yapmaktadır.

Lütü Zadeh; İlköğrenimini Tahran'da, liseyi Alburz Koleji'nde tamamladı. Liseyi bitirdikten sonra Tahran Üniversitesi giriş sınavına katılıp, ikinciliği elde ederek Elektrik Mühendisliği Bölümü'nde öğrenime başladı. Üniversite öğrenimini bitirdikten daha sonra 1942 yılında Amerika Birleşik Devletleri'ne gidip orada Boston'daki Massachusetts Teknoloji Enstitüsü'nde Elektrik Mühendisliği yüksek lisans eğitimini, ardından New York'ta Columbia Üniversitesinden 1949 yılında doktora eğitimini tamamladı.

Prof. Dr. Lütü Askerzade, Sistem Teorileri üzerinde araştırmalarına Columbia Üniversitesi, Princeton Üniversitesi ve Kaliforniya (Berkeley) Üniversitesi devam etti.

1963 yılında Kaliforniya (Berkeley) Üniversitesi, Elektrik Mühendisliği Fakültesi dekanı oldu. Lütü A. Zade 1965 yılına kadar sistem teorisi ve karar teorilerin analizi üzerinde yoğunlaşmıştır, ancak bu yıldan sonra Bulanık Mantık üzerinde çalışmalarını başlayıp bu mantığın yapay zeka, dil, mantık, karar teorileri, kontrol teoriler ve sinir sistemleri şebekeleri üzerinde olan etkilerini araştırmıştır. Lütü A. Zade işten emekli olmasına rağmen Berkeley Üniversitesi Bilgisayar programlama merkezinin yönetimini üstelenerek çalışmalarına devam etmektedir. Prof. Dr. Lütü A. Zade'nin Bulanık Mantık'ı ortaya koyduğu andan itibaren günümüze değin dünyada 15000'nin üzerinde bilimsel makale yayınlanmıştır.

3.2. Bulanık Kümeler

Bulanık kümeler, kullandığımız sözel ifadeleri bilgisayara aktarabilmek için oluşturduğumuz matematiksel modellerdir. Bulanık Mantığın sağladığı avantajlar:

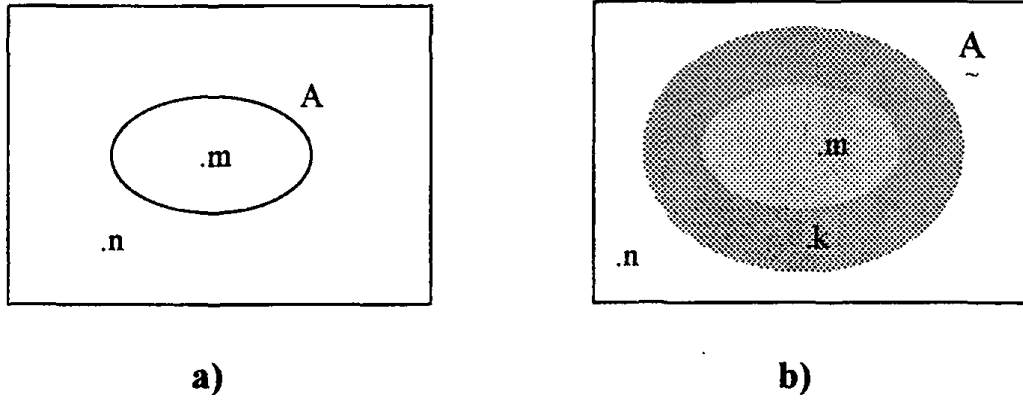
- İnsan düşünce sistemine ve tarzına yakındır.
- Uygulanmasında mutlaka matematiksel bir modele gereksinim duymaz.
- Yazılımın basit olması nedeniyle, sistem daha ekonomik olarak kurulabilir.
- Bulanık Mantık kavramını anlamak kolaydır.
- Üyelik değerlerinin kullanımı sayesinde, diğer kontrol tekniklerine göre daha esnektir.
- Kesinki arz etmeyen bilgiler kullanılabilir.
- Doğrusal olmayan fonksiyonların modellenmesine izin verir.
- Sadece uzman kişilerin tecrübeleri ile kolaylıkla bulanık mantığa dayalı bir model yada sistem tasarlanabilir.
- Geleneksel kontrol teknikleriyle uyum halindedir [113].

İsminin insanlarda çağrıştırdığı anlamın aksine bulanık mantık, belirsiz ifadelerle yapılan, belirsiz işlemler değildir. Modelleme aşamasında değişkenler ve kuralların esnek bir şekilde belirlenmesidir. Bu esneklik asla rastgelelik yada belirsizlik içermez. Nasıl bir

lastik içinde bulunduğu duruma göre şeklini değiştirirken, bütünlüğünü yapısını koruyabilirse, bir bulanık modelde değişen koşullara değişen cevaplar verirken özündeki yapıyı muhafaza eder [111].

Bir elemanın herhangi bir bulanık kümeye olan üyeliğini belirlemek için üyelik fonksiyonları kullanılır. Örneğin; x elemanının A Bulanık Kümesi'ne olan üyeliği Şekil 9'daki fonksiyon-teorisi formunda gösterilebilir [112-118].

$$\mu_{\tilde{A}}(x) \in [0,1] \quad (1)$$



Şekil 9 a) Klasik küme, b) Bulanık Küme [112]

$\tilde{A}$ bulanık kümesi ayrık X uzayının sonlu sayıda x elemanından ($x_1, x_2, x_3, \dots, x_n$) oluşuyor ise denklem (2)'deki gibi gösterilebilir.

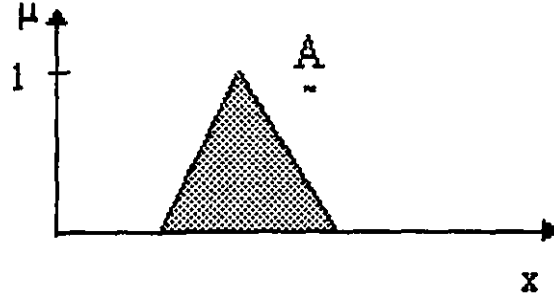
$$\tilde{A} = \left\{ \frac{\mu_{\tilde{A}}(x_1)}{x_1} + \frac{\mu_{\tilde{A}}(x_2)}{x_2} + \dots + \frac{\mu_{\tilde{A}}(x_n)}{x_n} \right\} = \sum \frac{\mu_{\tilde{A}}(x_i)}{x_i} \quad (2)$$

$\tilde{A}$ bulanık sürekli X uzayının sonsuz sayıda x elemanından oluşuyor ise denklem (3)'deki gibi gösterilebilir.

$$\tilde{A} = \int \frac{\mu_{\tilde{A}}(x)}{x} \quad (3)$$

(2) ve (3) denklemlerinin her ikisinde de kullanılan yatay çizgiler bölüm işareti olmayıp, x_i elemanı için ilgili üyelik fonksiyonun göstermektedir. Denklem(2)'deki “+”

işareti cebirsel toplama işlemini değil, küme işlemlerindeki birleşim işleminin fonksiyonel gösterimidir. Denklem 3'deki integral işareti de sürekli değişkenlerin birleşimi anlamında kullanılmaktadır [119]. Şekil 10'da bir A kümesi için üyelik fonksiyonu görülmektedir.



Şekil 10 A Bulanık Kümesi için Üyelik Fonksiyonu[112]

3.2.1 Bulanık Kümelerde İşlemler

Bulanık küme teorisi; sadece sözel değerlerin temsilini sağlamaz aynı zamanda bu değerlerin mantıksal bir yolla irdelenip sonuç çıkarılmasını da sağlar. Bulanık kümeler teorisinde sık kullanılan temel üç işlem aşağıda sıralanmıştır [119].

Birleşim işlemi,

$$\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x) \quad (4)$$

Kesişim işlemi

$$\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x) \quad (5)$$

Değilleme işlemi

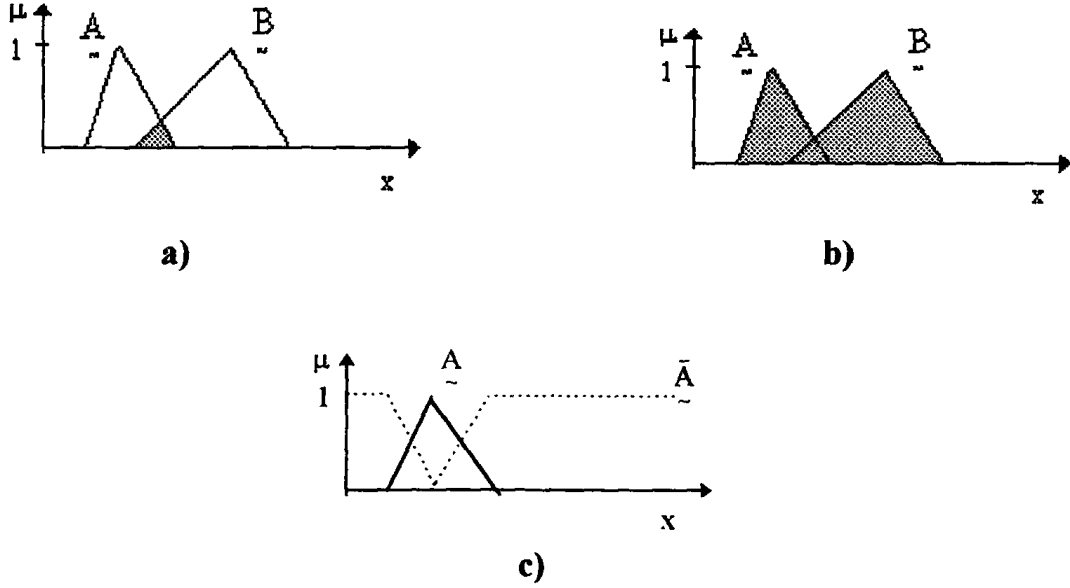
$$\overline{\mu_A}(x) = 1 - \mu_A(x) \quad (6)$$

Bu işlemler, klasik kümede kullanılan *Venn* diyagramına benzer bir yöntemle ifade edilirler. Bulanık kümeler için grafiksel gösterim Şekil 11'de verilmiştir. X uzayında tanımlanan A bulanık kümesi, o uzayın alt kümesidir. Klasik kümelerde olduğu gibi, bulanık kümelerde de, x elemanının boş küme'ye olan üyelik değeri 0 ve tüm küme

X'e olan üyeliği 1 dir. Bu ifadeler fonksiyon teorisi formunda aşağıdaki gibi gösterilebilir [119].

$$\underline{A} \subseteq X \Rightarrow \mu_{\underline{A}}(x) \leq \mu_X(x) \quad (7)$$

$$\forall x \in X \rightarrow \mu_{\emptyset}(x) = 0 \quad (8)$$



Şekil 11 : A Bulanık Kümelerde İşlemler a)Bileşim, b) Kesişim, c) Değil işlemi

Klasik kümeler için geçerli olan De Morgan kuralları bulanık kümeler için de geçerlidir.

3.2.2 Üyelik İşlemleri

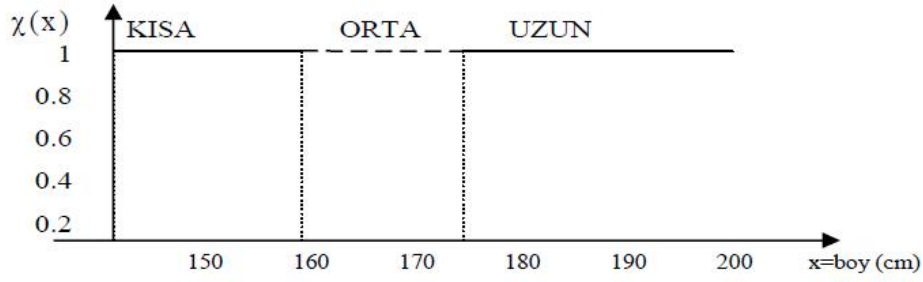
Bulanık kümeler $\mu_{\underline{A}}(x)$ üyelik fonksiyonu ile temsil edilir. $\mu_{\underline{A}}(x)$, üyelik fonksiyonundaki bir x noktasının $\underline{A}$ bulanık kümesindeki üyelik derecesidir. $\mu_{\underline{A}}(x) = 1$ konumunu, x'in $\underline{A}$ bulanık kümesinin kesin bir elemanı olduğunu tanımlamaktadır. Benzer şekilde $\mu_{\underline{A}}(x) = 0$, x'in $\underline{A}$ bulanık kümesi dışında olduğunu göstermektedir. $0 < \mu_{\underline{A}}(x) < 1$ arasındaki her değer, x'in $\underline{A}$ bulanık kümesindeki üyeliğinin belirsiz değerleridir. Bu yüzden kesin olmayan büyüklükler üyelik fonksiyonları tarafından belirtilmiş bulanık kümelerle temsil edilirler [120].

Önemli üyelik fonksiyonları Tablo 6’da gösterilmiştir.

Tablo 6 Önemli Üyelik Fonksiyonları[6]

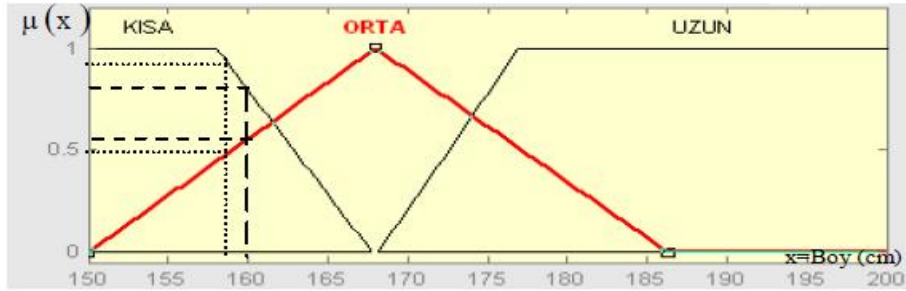
Adı	Denklemi	Grafığı
Üçgen Üyelik Fonksiyonu	$\mu(x) = \text{Max} \left\{ \min \left[\frac{x - x_1}{x_2 - x_1}, \frac{x_2 - x_1}{x_2 - x_1}, 0 \right] \right\}$	
Yamuk Üyelik Fonksiyonu	$\mu(x) = \text{Max} \left\{ \min \left[\frac{x - x_1}{x_{t1} - x_1}, 1, \frac{x_2 - x_1}{x_2 - x_{t2}}, 0 \right] \right\}$	
Gauss Üyelik Fonksiyonu	$\mu(x) = e^{-\frac{1}{2} \left(\frac{x - x_1}{\sigma} \right)^2}$	

Bir büyüklüğün niceliğini veya niteliğini anlatmak için günlük dilde kullandığımız bazı kavramlar, birden fazla özelliği aynı anda gösterebilir, açık ve net tek bir anlamı yoktur. Örneğin “iyi” sıfatı bir çekici nitelerken kullanışlı, fonksiyonel gibi anlamlara gelirken, bir öğrenci için örnek-çalışkan, bir çorba için lezzetli-güzel anlamları taşıyabilir. Kesin değerli kümelerle bulanık kümelerin farkını şu ifadelerle de açıklamak mümkündür. Birçok büyüklüğü veya ifadeyi kesin sınırlarla sınıflara ayırmak mümkündür. Bazı kavramlar ise, az ya da çok gibi birden fazla özelliği aynı anda gösterebilir. Bu kavramı, baskın özelliğini gösterdiği sınıfa dahil etmek, diğer özelliğini ihmal etmek doğru bir yaklaşım olmayabilir. Yaşlı-genç, ılık, soğuk, sıcak gibi kavramlar net olmayan ifadelerdir ve ifadeleri açıklayabilmek için kümeler kavramından yararlanılır [106].



Şekil 12 Klasik kümelerle boy sınıflaması [106]

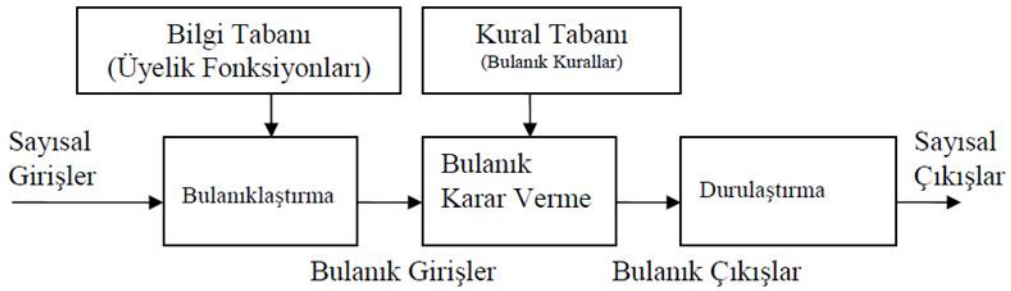
Şekil 12’de klasik kümelerle boy sınıflaması gösterilmiştir. Bu kümelerle bakıldığında 1.59 boyundaki biri “kısa” sınıfına/kümesine girerken 1.60 boyundaki arkadaşı “orta” boylu kabul edilmektedir. Aynı şey 1.74 boyundaki biri için de söylenebilir. Çünkü kendisi de hemen hemen aynı özellikleri göstermesine rağmen 1.75 boyundaki arkadaşı uzun boylu kabul edilirken kendi orta boylu gurubuna girmektedir. Oysa Şekil 10’daki gibi bir gösterim biçimi daha gerçekçi olacaktır. Şekil 13’de 1.59 boyundaki kişi “Boy” evrensel kümesi üzerinde bulunan “Kısa” kümesine %92 üye iken, aynı evren üzerinde bulunan “Orta” kümesine %50 üye olacaktır (Şekil 14). 1.60 boyundaki kişi de, “Kısa” kümesine %80 üye iken, “Orta” kümesine %58 üye olacaktır.



Şekil 13 Bulanık kümelerle boy sınıflaması [106]

Benzer şekilde 1.74 ve 1.75 boyundakiler ilk gösterimde çok farklı sınıflara düşerlerken, bu kişilerin birbirine yakın özellikler taşıdığı ikinci gösterimde daha anlamlı bir biçimde ifade edilebilmektedir. Bulanıklaştırma dediğimiz bu ifade biçiminde, bir büyüklüğü birden fazla sözel değer (veya önerme) ve bu önermelerin doğruluk dereceleriyle temsil edebiliriz. Bulanıklaştırma işleminde kullandığımız iç içe geçmiş kümelerle ise bulanık küme adını veriyoruz [106].

Sistemin yapısı karmaşık ta olsa, insanlar sistemin davranışı, eğilimi hakkında genel hatlarıyla bir yargıya varabilirler. Bu genel bilgi, insanların karmaşık sistemleri anlayıp kontrol edebilmesi için yeterlidir. Bulanık küme kuramın mühendislik veya diğer alanlarda uygulanmasındaki amaç, kesin olmayan bilgiler ışığında tutarlı sonuçlar çıkarabilmektir. Belleğimizde bilgi ve tecrübelerimiz sonucu pekiştirdiğimiz yorum, anlam ve değerlendirmelerden oluşan çok sayıda sözel kural kalıbı vardır. Karar verme süreci, karşı karşıya kalınan bir durumu bu kurallar doğrultusunda yapılan iç konuşmalar aracılığıyla değerlendirerek bir sonuca varma şeklindedir. Bilgisayarların bir durum karşısında bu tür bir muhakeme yapabilmesi için o durumla ilgili bilgi, tecrübe ve sezgilerimizden oluşan bir dizi kuralı bilgisayara aktarabilmemiz gerekir. “EĞER bu böyleyse VE şu da şöyleyse O HALDE şunu yap” gibi sözel kuralların matematiksel karşılığı ise bahsettiğimiz bulanık kümelerin birbiriyle uygun şekilde bağlanması ile oluşturulmaktadır. Buna bulanık karar verme süreci denir. Şekil 14’de bilgisayarlarda bulanık karar verme süreci bulanıklaştırma, karar verme ve durulaştırma işlemleri gösterilmiştir [106].



Şekil 14 Bulanık karar verme sisteminin yapısı [106]

Dış dünyadan bilgisayara ölçüm yoluyla alınan ve kesin bir nümerik (sayısal) değere sahip olan giriş verisi, bilgi tabanındaki üyelik fonksiyonları tarafından sözel ifadeler ve giriş verisinin bu ifadeyi ne oranda desteklediğini gösteren üyelik derecelerine dönüştürülür. Bu aşamaya *bulanıklaştırma* adı verilir. Bulanıklaştırma sonunda elde edilen sözel ifadeler, insanların karar verme sürecinde olduğu gibi, kural

tabanındaki önermelerle karşılaştırılır ve yine sözel yargı sonuçlarına varılır, bu sonuçların hangi oranda geçerli olduğunu yine girişteki

üyelik dereceleri belirler. Bu kısma bulanık karar verme süreci adı verilir. Bulanık karar verme sürecinin çıkışında yargı sonuçlarını ifade eden sözel ifadeler ve bunların destek dereceleri bulanık çıkışlar olarak adlandırılır. Eğer bilgisayar çıkışta bir makineye bilgi yolluyorsa, bulanık çıkışlar yine makinelerin anlayacağı dil olan sayısal çıkış değerlerine dönüştürülmelidir. Bu dönüştürme işlemi *durulaştırma* yada *berraklaştırma* katında yapılır.

Kural Tabanı; karar verme işleminde kullanılan bir çok paralel kurallardan ve sistem değişkenlerinden oluşmaktadır. Bu kurallar sistemin girişi ile çıkışı arasındaki ilişkiyi tanımlamaktadır.

3.3. Bulanık Mantığın Kullanım Alanları ve Günümüzde Uygulama Örnekleri

BM birçok kontrol uygulamasında başarıyla kullanılmıştır. Bulanık mantığın kullanım alanlarından bazıları:Trafik Sinyal Optimizasyonu (Kavşak ve Ana arterlerde) ,Katılım Denetimi Kontrolü,Robotik,Proses kontrol,Ev elektroniği,Trafik ,Görüntü işleme, Veri tabanı sorgulama, Arıza denetimidir.Buna ilaveten Bulanık Mantığın en yaygın kullanım alanlarının başında şu konular gelmektedir:

Yapay zeka, sistem analizi, karar analizi, nümerik analiz, veri işleme, mühendislik, Genetik algoritmalar , ekonomi, robotik

Bulanık mantık ilk kez 1973 yılında, Londra'ki Queen Mary College'da profesör olan Ebrahim H. Mamdani tarafından bir buhar makinasında uygulandı.

Ticari olarak ise ilk defa, 1980 yılında, Danimarka'daki bir çimento fabrikasının fırınıni kontrol etmede kullanıldı. Bulanık mantık ile hazırlanan bir sistem, bilgisayar desteğinde, sensörlerden ısı ve maddelere ait bilgileri alarak ve "feed-back" (geri besleme) metoduyla değişkenleri kontrol ederek, bu ayarlama işini çok hassas ölçümlerle gerçekleştirmiş ve büyük oranda enerji tasarrufu sağlamıştır.

1987'de, Uluslararası Bulanık Sistemler Derneği'nin Tokyo'da düzenlediği bir konferansta bir mühendis, bulanık mantıkla programladığı bir robota, bir çiçeği ince bir çubuğun üzerinde düşmeyecek şekilde bıraktırmayı başarmıştır. Bundan daha fazla ilgi

eken gerek ise, robotun bunu yaptığını gren bir seyircinin mhendise, sistemden bir devreyi ıkarmasını teklif etmesinden sonra grlmştr. Mhendis nce, devreyi ıkarırsam iek dřer diye bunu kabul etmemiř, fakat seyircinin ieğinin ne tarafa doėru dřtğn grmek istediğini sylemesi zerine devreyi ıkarmıřtır ve Robot beklenmedik bir řekilde yine aynı hassaslıkla ieğiyi dřrmeden ubuėun zerine bırakmıřtır. Kısacası bulanık mantık sistemleri, yetersiz bilgi temin edilse bile tıpkı insanların yaptığı gibi bir tr "saėduyu" kullanarak (yani mevcut bilgiler yardımıyla neticeye gtrc akıl yrtmeler yaparak) iřlemleri gerekleřtirebilmektedir.

Bulanık mantık kullanılarak retilen edilen fotoėraf makineleri, otomatik odaklama yapanlardan bile daha net bir grnt vermektedir.

Fotokopi makineleri ise bulanık mantıkla ok daha kaliteli kopyalar ıkarmaktadırlar. Zira odanın sıcaklığı, nemi ve orijinal kaėıttaki karakter yoėunluėuna gre deėiřen resim kalitesi, bu  temel unsur hesaplanarak mkemmele yakın hale getirilmektedir.

Kameralardaki bulanık mantık devreleri ise sarsıntılardan doėan grnt bozukluklarını asgariye indirmektedir. Bilindiėi gibi elde tařınan kameralar, ne kadar dikkat edilirse edilsin net bir grnt vermez. Bulanık mantık programları bu grntleri netleřtirmek iin řyle bir metod kullanır: Eėer grntdeki btn řekiller, aynı anda, bir tarafa doėru kayıyorsa bu, insan hatasından kaynaklanan bir durumdur; kayma gz nne alınmadan kayıt yapılır. Bunun dıřındaki řekiller ve hareketler ise normal ekim durumunda gerekleřtiėi iin mdahale edilmez.

Bira bulanık mantık sistemi ise, mekanik cihazlardan ok daha verimli bir řekilde bilgi deėerlendirmesi yapmaktadır. Japon Omron Grubu, byk firmalara saėlık hizmeti veren bir sisteme ait beř tıp veri tabanını, bulanık mantık teorileri ile kontrol etmektedir. Bu bulanık sistem, 10.000 kadar hastanın saėlık durumlarını ėrenmek ve hastalıklardan korunmalarına, saėlıklı kalmalarına ve stresten kurtulmalarına yardımcı olmak zere kiřiye zel planlar izebilen yaklaşık 500 kural kullanmaktadır.

Pilav piřirme aletlerinden asansrlere, arabaların motor ve sspansiyon sistemlerinden nkleer reaktrlerdeki soėutma nitelerine, klimalardan elektrikli sprgelere kadar bulanık mantığın uygulandıėı birok alan bulunmaktadır. Bu alanlarda

sağladığı enerji, iş gücü ve zaman tasarrufu ile "iktisat" açısından da önem kazanmaktadır.

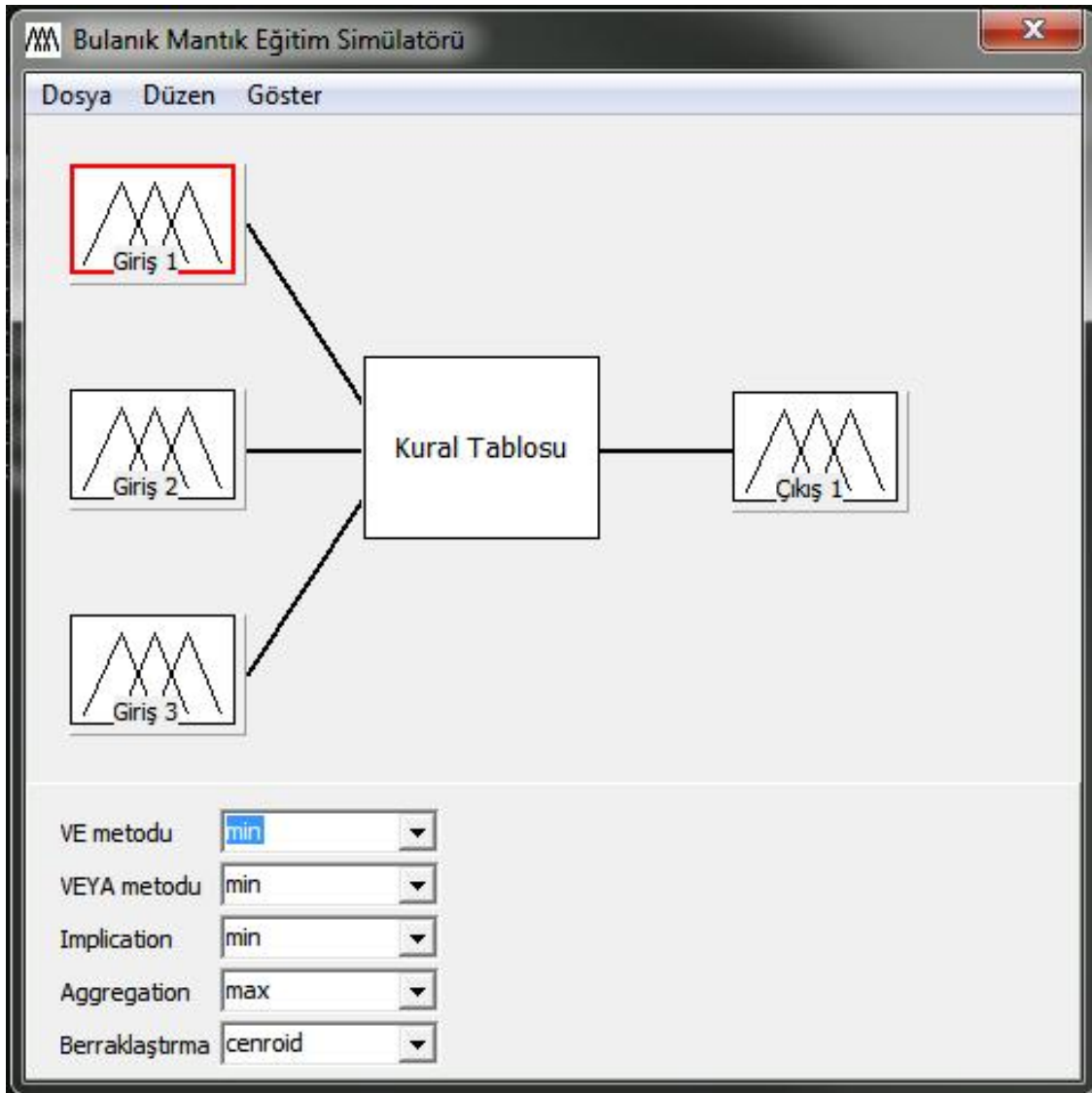
Bulanık mantığın gelecekteki uygulama sahaları, daha da genişleyecek gibi gözükmemektedir. Şeker hastaları için vücuttaki insülin miktarını ayarlayarak yapay bir pankreas görevi yapan minik yapıların üretiminde, prematüre doğumlarda bebeğin ihtiyaç duyduğu ortamı devam ettiren sistemlerin hazırlanmasında, suların klorlanmasında, kalp pillerinin üretiminde, oda içindeki ışığın miktarının ayarlanmasında ve bilgisayar sistemlerinin soğutulmasında bulanık mantık çok şeyler vaatetmektedir.

3.1. Bulanık Mantığın Kullanıldığı bazı Uygulamalar:

- Hidroelektrik güç üniteleri için kullanılan Baraj kapılarının otomatik kontrolü (*Tokio Electric Pow.*)
- Stok kontrol değerlendirmesi için bir uzman sistem (*Yamaichi, Hitachi*)
- Klima sistemlerinde istenmeyen ısı iniş çıkışlarının önlenmesi
- Araba motorlarının etkili ve kararlı kontrolü (*Nissan*)
- Otomobiller için "Cruise-control" (*Nissan, Subaru*)
- Dökümanların arşivleme sistemi (*Mitsubishi Elec.*)
- Depremlerin önceden bilinmesi için Tahmin Sistemi (*Inst. of Seismology Bureau of Metrology, Japan*)
- İlaç teknolojileri: Kanser teşhisi (*Kawasaki Medical School*)
- Cep bilgisayarlarında el yazısı algılama teknolojisi (*Sony*)
- Video Kameralarda hareketin algılanması (*Canon, Minolta*)
- El yazısı ve ses tanımlama (*CSK, Hitachi, Hosai Univ., Ricoh*)
- Helikopterler için uçuş desteği (*Sugeno*)
- Çelik sanayinde makina hızı ve ısısının kontrolü (*Kawasaki Steel, New-Nippon Steel, NKK*)

- Raylı metro sistemlerinde sürüş rahatlığı, duruş mesafesinin kesinliğini ve ekonomikliğin geliştirilmesi (1.Giriş 'te bahsedilen metro hedefe 7cm kala durabilmektedir) (*Hitachi*)
- Otomobiller için gelişmiş yakıt tüketimi (*NOK, Nippon Denki Tools*)

3.4. Proje Kapsamında Hazırlanan Bulanık Mantık Simülatörünün Bölümleri



Şekil 15 Bulanık mantık eğitim simülatörü ana penceresi

Kural Tablosu Düzenle

Eğer (Giriş 1 = Orta) VE (Giriş 2 = Yavaş) VE (Giriş 3 = Geniş) ise (Çıkış 1 = Hızlı)
Eğer (Giriş 1 = Parlak) VE (Giriş 2 = Yavaş) VE (Giriş 3 = Orta) ise (Çıkış 1 = Çok Hızlı)
Eğer (Giriş 1 = Karanlık) VE (Giriş 2 = Hızlı) VE (Giriş 3 = Geniş) ise (Çıkış 1 = Orta)
Eğer (Giriş 1 = Karanlık) VE (Giriş 2 = Hızlı) VE (Giriş 3 = Dar) ise (Çıkış 1 = Yavaş)
Eğer (Giriş 1 = Parlak) VE (Giriş 2 = Yavaş) VE (Giriş 3 = Dar) ise (Çıkış 1 = Hızlı)

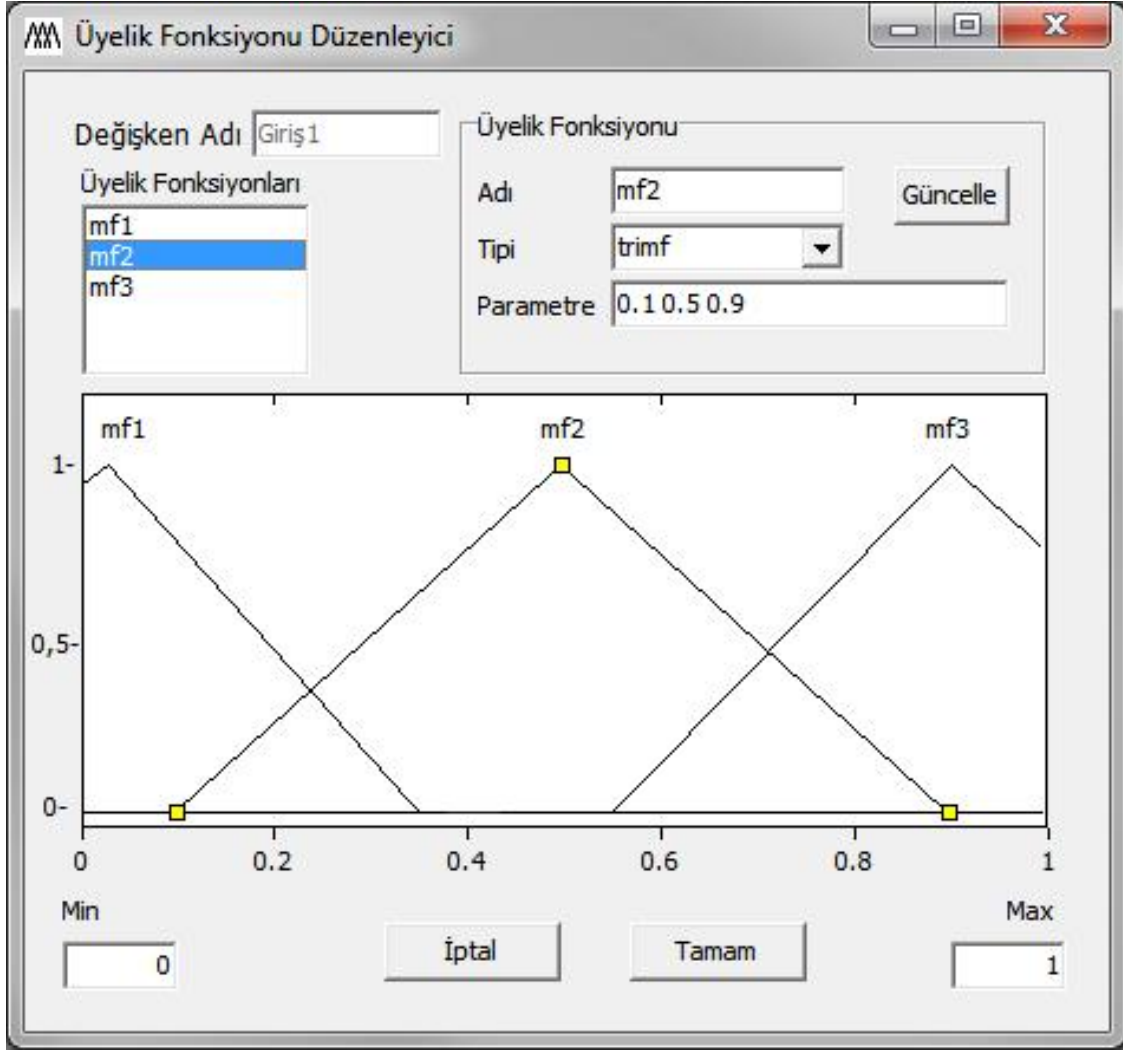
Giriş1	Giriş2	Giriş3	Çıkış1
Yok	Yok	Yok	Yok
Karanlık	Yavaş	Dar	Yavaş
Orta	Orta	Orta	Orta
Parlak	Hızlı	Geniş	Hızlı
☐ değil	☐ değil	☐ değil	☐ değil

Bağlantı
☒ ve
☐ veya

Ağırlık
1

Kural Ekle Kural Sil Kural Değiştir

Şekil 16 Kural tablosu oluşturma penceresi



Şekil 17 Üyelik fonksiyonu düzenleme penceresi

4. BÖLÜM: Yapay Sinir Ağları

4.1. Genel Tanım

Yapay sinir ağları (YSA), insan beyninin özelliklerinden olan öğrenme yolu ile yeni bilgiler türetebilme, yeni bilgiler oluşturabilme ve keşfedebilme gibi yetenekleri, herhangi bir yardım almadan otomatik olarak gerçekleştirebilmek amacı ile geliştirilen bilgisayar sistemleridir [121].

Yapay sinir ağları; insan beyninden esinlenerek, öğrenme sürecinin matematiksel olarak modellenmesi uğraşı sonucu ortaya çıkmıştır. Bu nedenledir ki, bu konu üzerindeki çalışmalar ilk olarak beyni oluşturan biyolojik üniteler olan nöronların modellenmesi ve bilgisayar sistemlerinde uygulanması ile başlamış, daha sonraları bilgisayar sistemlerinin gelişimine de paralel olarak bir çok alanda kullanılır hale gelmiştir.

İnsan beyninin çalışma prensibini taklit ederek çalışan bu sistemler, her ne kadar bilgisayar teknolojisi hızlı bir gelişim göstermiş, işlem hızları nano saniyeler mertebesine inmiş olsa da, bırakalım insan beynini, ilkel bir canlı beyninin fonksiyonları dahi baz alındığında, böyle bir organizmanın yanında çok ilkel kalmaktadır. Nano saniyeler bazındaki işlem hızları ile YSA'lar, mili saniyeler mertebesindeki işlen hızları ile işlem yapan insan beyninin işlevselliğinin henüz çok uzağındadır.

Burada kısa bir hatırlatma yapmak gerekirse; insan beyninde yaklaşık 10^{11} sinir hücresinin varlığından bahsedilmekle birlikte, bu sayının bilgisayar ortamında modellenmesi şu an için mümkün görünmemektedir. Fakat karar hızı açısından insan beyni ile henüz yarışamaları bile, YSA'lar yapısalıkları ve hassas eşleştirmelerin başarı ile gerçekleştirebilmeleri ile gün geçtikçe daha fazla uygulama alanı bulmaktadır.

4.2. YSA'ların Genel Özellikleri

YSA'lar, uygulanan ağ modeline göre değişik karakteristik özellikler göstermelerine karşın temel birkaç ortak özelliğe sahiptirler.

Birinci özellik; YSA'larda sistemin paralellliği ve toplamsal işlevin yapısal olarak dağılımlılığıdır [122]. YSA lar bir çok nörondan meydana gelir ve bu nöronlar eş zamanlı olarak çalışarak karmaşık işlevleri yerine getirir. Diğer bir deyişle karmaşık işlevler bir

çok nöronun eş zamanlı çalışması ile meydana getirilir. Süreç içerisinde bu nöronlardan her hangi biri işlevini yitirse dahi sistem güven sınırları içerisinde çalışmasına devam edebilir.

İkinci özellik ise genelleme yeteneği, diğer bir deyişle ağ yapısının, eğitim esnasında kullanılan nümerik bilgilerden eşleştirmeyi betimleyen kaba özellikleri çıkarsaması ve böylelikle eğitim sırasında kullanılmayan girdiler için de, anlamlı yanıtlar üretebilmesidir [123].

Üçüncü olarak; ağ fonksiyonları non-lineer olabilmektedir. Yapı üzerinde dağılmış belli tipteki non-lineer alt birimler özellikle, istenen eşleştirmenin denetim ya da tanımlama işlemlerinde olduğu gibi non-lineer olması durumunda işlevin doğru biçimde yerine getirilebilmesini matematiksel olarak olası kılarlar.

Dördüncü özellik ise; sayısal ortamda tasarlanan YSA'ların, donanımsal gerçekleştirilebilirlikleridir. Bu özellik beklide YSA'ların günlük hayatta daha da fazla yaşamımızın içine gireceğinin (girebileceğinin) göstergesidir.

4.3. YSA'ların Avantaj ve Dezavantajları

YSA'lar makina öğrenmesi gerçekleştirebilirler. Yapay sinir ağlarının temel işlevi zaten bilgisayarın öğrenmesini sağlamaktır. Olayları öğrenerek benzer olaylar karşısında mantıklı kararlar verebilirler.

Bilgi işleme yöntemleri geleneksel programlamadan farklıdır. Bu nedenle geleneksel programlamanın getirdiği bir çok olumsuzluk ortadan kaldırılabilir

Bilgiler ağın tamamında saklanır. Geleneksel programlamada olduğu gibi bilgiler veri tabanları yada dosyalarda belli bir düzende tutulmaz, ağın tamamına yayılarak değerler ile ölçülen ağ bağlantılarında saklanmaktadır. Nöronlardan bazılarının işlevini yitirmesi, anlamlı bilginin kaybolmasına neden olmaz.

Örnekleri kullanarak öğrenirler. YSA'nın öğrenebilmesi için örneklerin belirlenmesi, bu örneklerin ağa gösterilerek istenen çıktılara göre ağın eğitilmesi gerekmektedir. Ağın başarısı, seçilen örnekler ile doğru orantılıdır, ağa olay bütün yönleri ile gösterilemezse ağ yanlış çıktılar üretebilir.

Daha önce görülmemiş örnekler hakkında bilgi üretebilirler. YSA'lar eğitimleri sırasında kendilerine verilen örneklerden genellemeler çıkarırlar ve bu genellemeler ile yeni örnekler hakkında bilgi üretebilirler.

Algılamaya yönelik olaylarda kullanılabilirler. YSA'ların en başarılı oldukları alanlar, algılamaya yönelik uygulama alanlarıdır. Bu alanlarda başarıları kanıtlanmıştır.

Örüntü (pattern) ilişkilendirme ve sınıflandırma yapabilirler. YSA'lar kendilerine örnekler halinde verilen örüntüleri kendisi veya diğerleri ile ilişkilendirebilir. Ayrıca kendisine verilen örneklerin kümelenmesi ile, bir sonraki verinin hangi kümeye dahil olacağının karar verilmesi konusunda kullanılabilirler.

Örüntü tamamlama yapabilirler. Ağa eksik bilgileri içeren örüntüler verildiğinde eksik bilgilerin tamamlanması konusunda başarılıdır.

Kendi kendine öğrenebilme ve organize etme yetenekleri vardır. YSA'lar online olarak öğrenebilirler ve kendi kendilerini eğitebilirler.

Eksik bilgi ile çalışabilmektedirler. Geleneksel sistemlerin aksine YSA'lar eğitildikten sonra veriler eksik bilgi içerse dahi, çıktı üretebilirler. Bu durum bir performans kaybı yaratmaz, performans kaybı eksik bilginin önemine bağlıdır. Burada bilgilerin önem dereceleri eğitim sırasında öğrenilir.

Hata toleransına sahiptirler. YSA'ların eksik bilgilerle çalışabilmeleri ve bazı hücreleri bozulsa dahi çalışabilmeleri, onları hatalara karşı toleranslı yapar.

Dereceli bozulma (Graceful degradation) gösterirler. Bir ağ, zaman içerisinde yavaş ve göreceli bir bozulmaya uğrar. Ağlar problemin ortaya çıktığı anda hemen bozulmazlar.

Dağıtık belleğe sahiptirler. YSA'larda bilgi ağa dağılmış bir şekilde tutulur. Hücrelerin bağlantı ve ağırlık dereceleri, ağın bilgisini gösterir. Bu nedenle tek bir bağlantının kendi başına anlamı yoktur.

Burada çok temel bazı avantajlardan bahsedilmekle beraber, YSA'ların daha pek çok avantajı vardır.

YSA'ların, pek çok avantajın yanında bazı dezavantajları da vardır. Belli başlı dezavantajları;

Donanım bağımlıdır. YSA'ların en önemli sorunu donanım bağımlı olmalarıdır. YSA'ların en önemli özellikleri ve var oluş nedenlerinden birisi olan paralel işlem yapabilme yeteneği, paralel çalışan işlemciler ile performans gösterir.

Uygun ağ yapısının belirlenmesinde belli bir kural yoktur. YSA'larda probleme uygun ağ yapısının belirlenmesi için geliştirilmiş bir kural yoktur. Uygun ağ yapısı deneyim ve deneme yanılma yolu ile belirlenmektedir.

Ağın parametre değerlerinin belirlenmesinde belli bir kural yoktur. YSA'larda öğrenme katsayısı, hücre sayısı, katman sayısı gibi parametrelerin belirlenmesinde belirli bir kural yoktur. Bu değerlerin belirlenmesi için belirli bir standart olmamakla birlikte her problem için farklı bir yaklaşım söz konusu olabilmektedir.

Öğrenilecek problemin ağa gösterimi önemli bir problemdir. YSA'lar nümerik bilgiler ile çalışabilmektedirler. Problemler YSA'lara tanıtılmadan önce nümerik değerlere çevrilmek zorundadırlar. Burada belirlenecek gösterim mekanizması ağın performansını doğrudan etkileyecektir. Bu da kullanıcının yeteneğine bağlıdır.

Ağın eğitiminin ne zaman bitirilmesi gerektiğine ilişkin belli bir yöntem yoktur. Ağın örnekler üzerindeki hatasının belirli bir değerin altına indirilmesi eğitimin tamamlandığı anlamına gelmektedir. Burada optimum neticeler veren bir mekanizma henüz yoktur ve YSA ile ilgili araştırmaların önemli bir kolunu oluşturmaktadır.

Ağın davranışları açıklanamamaktadır. Bu sorun YSA'ların en önemli sorunudur. YSA bir probleme çözüm ürettiği zaman, bunun neden ve nasıl olduğuna ilişkin bir ipucu vermez. Bu durum ağa olan güveni azaltıcı bir unsurdur. Tablo 7’de geleneksel algoritmalar ile YSA’ların çeşitli özellikleri açısından karşılaştırması verilmiştir.

Tablo 7 Geleneksel algoritmalar ile YSA'ların karşılaştırması

Geleneksel Algoritmalar	Yapay Sinir Ağları
Çıkışlar, koyulan kurallara girişlerin uygulanması ile elde edilir.	Öğrenme esnasında giriş çıkış bilgileri verilerek, kurallar koyulur.
Hesaplama; merkezi, eş zamanlı ve ardışıldır.	Hesaplama; toplu, eş zamansız ve öğrenmeden sonra paraleldir.
Bellek paketlenmiş ve hazır bilgi depolanmıştır.	Bellek ayrılmış, ve ağı yayılmıştır. Dahilidir.
Hata toleransı yoktur.	Hata toleransı vardır.
Nisbeten hızlıdır.	Yavaş ve donanıma bağımlıdır.
Bilgiler ve algoritmalar kesindir.	Deneyimden yararlanır.

4.4. YSA'ların Kullanıldığı Alanlar

Yapay sinir ağları başlıca; **Sınıflandırma, Modelleme ve Tahmin** uygulamaları olmak üzere, pek çok alanda kullanılmaktadır. Başarılı uygulamalar incelendiğinde, YSA'ların çok boyutlu, gürültülü, karmaşık, kesin olmayan, eksik, kusurlu, hata olasılığı yüksek sensör verilerinin olması ve problemi çözmek için matematiksel modelin ve algoritmaların bulunmadığı, sadece örneklerin var olduğu durumlarda yaygın olarak kullanıldıkları görülmektedir. Bu amaçla geliştirilmiş ağlar genellikle şu fonksiyonları gerçekleştirmektedirler [124].

- Muhtemel fonksiyon kestirimleri
- Sınıflandırma
- İlişkilendirme veya örüntü eşleştirme
- Zaman serileri analizleri
- Sinyal filtreleme
- Veri sıkıştırma
- Örüntü tanıma
- Doğrusal olmayan sinyal işleme

- Doğrusal olmayan sistem modelleme
- Optimizasyon
- Kontrol

YSA'lar pek çok sektörde değişik uygulama alanları bulmuştur. Bunlardan bazıları;

Uzay: uçuş simülasyonları, otomatik pilot uygulamaları, komponentlerin hata denetimleri vs.

Otomotiv: otomatik yol izleme, rehber, garanti aktivite analizi, yol koşullarına göre sürüş analizi vs.

Bankacılık: kredi uygulamaları geliştirilmesi, müşteri analizi ve kredi müracaat değerlendirilmesi, bütçe yatırım tahminleri vs.

Savunma: silah yönlendirme, hedef seçme, radar, sensör sonar sistemleri, sinyal işleme, görüntü işleme vs.

Elektronik: kod sırası öngörüsü, çip bozulma analizi, non-lineer modelleme vs.

Eğlence: animasyonlar, özel efektler, pazarlama öngörüsü vs.

Finans: kıymet biçme, pazar performans analizi, bütçe kestirimi, hedef belirleme vs.

Sigortacılık: ürün optimizasyonu, uygulama politikası geliştirme vs.

Üretim: üretim işlem kontrolü, ürün dizaynı, makina yıpranmalarının tespiti, dayanıklılık analizi, kalite kontrolü, iş çizelgeleri hazırlanması vs.

Sağlık: göğüs kanseri erken teşhis ve tedavisi, EEG, ECG, MR, kalite artırımı, ilaç etkileri analizi, kan analizi sınıflandırma, kalp krizi erken teşhis ve tedavisi vs.

Petro kimya: arama, verim analizi vs.

Robotik: yörünge kontrol, forklift robotları, görsel sistemler, uzaktan kumandalı sistemler, optimum rota belirleme vs.

Dil: sözcük tanıma, yazı ve konuşma çevrimi, dil tercüme vs.

Telekomünikasyon: görüntü ve data karşılaştırma, filtreleme, eko ve gürültü sönmülendirilmesi, ses ve görüntü işleme, trafik yoğunluğunun kontrolü ve anahtarlama vs.

Güvenlik: parmak izi tanıma, kredi kartı hileleri saptama, retina tarama, yüz eşleştirme vs.

Bu örnekler çoğaltılabilir. Görüldüğü gibi YSA'lar günlük hayatımızda farkında olmadığımız pek çok alanda kullanılmaktadır. Gün geçtikçe uygulama alanları genişlemekte ve gelişmektedir.

4.5. YSA'ların Tarihçesi

YSA'lar ile ilgili çalışmalar 20. yy'nın ilk yarısında başlamış ve günümüze kadar büyük bir hızla devam etmiştir. Bu çalışmaları 1970 öncesi ve sonrası diye iki kısma ayırmak mümkündür. Zira 1970 yılları YSA için bir dönüm noktasını teşkil etmiş daha önce aşılması imkânsız görünen pek çok problem bu dönemlerde aşılmıştır.

İlk yapay sinir ağı modeli 1943 yılında, bir sinir hekimi olan **Warren McCulloch** ile bir matematikçi olan **Walter Pitts** tarafından gerçekleştirilmiştir. **McCulloch** ve **Pitts**, insan beyninin hesaplama yeteneğinden esinlenerek, elektrik devreleriyle basit bir sinir ağı modellemişlerdir.

1948 yılında "**Wiener "Cybernetics"**" isimi kitabında, sinirlerin çalışması ve davranış özelliklerine değinmiş, 1949 yılında ise **Hebb "Organization of Behavior"** adlı eserinde öğrenme ile ilgili temel teoriyi ele almıştır.

Hebb kitabında öğrenebilen ve uyum sağlayabilen sinir ağları modeli için temel oluşturacak "**Hebb kuralı**"nı ortaya koymuştur. **Hebb kuralı**; sinir ağının bağlantı sayısı değiştirilirse, öğrenebileceğini ön görmekteydi.

1957 yılında **Frank Rosenblatt**'ın **Perceptron**'u geliştirmesinden sonra, YSA'lar ile ilgili çalışmalar hız kazanmıştır. **Perceptron**; beyin işlevlerini modelleyebilmek amacıyla yapılan çalışmalar neticesinde ortaya çıkan tek katmanlı eğitilebilen ve tek çıkışa sahip bir ağ modelidir [125].

1959 yılında **Bernand Widrow** ve **Marcian Hoff** (**Stanford Üniversitesi**) **ADALINE (Adaptive Linear Neuron)** modelini geliştirmişler ve bu model YSA'ların mühendislik uygulamaları için başlangıç kabul edilmiştir. Bu model Rosenblatt'ın Perceptron'una benzemekle birlikte, öğrenme algoritması daha gelişmiştir. Bu model uzun mesafelerdeki telefon hatlarındaki yankıları ve gürültüleri yok eden bir adaptif filtre olarak kullanılmış, ve gerçek dünya problemlerine uygulanan ilk YSA olma özelliğini kazanmıştır. Bu yöntem günümüzde de aynı amaçla kullanılmaktadır.

1960 ların sonlarına doğru YSA çalışmaları durma noktasına gelmiştir. Buna en önemli etki; **Minsky** ve **Pappert** tarafından yazılan **Perceptrons** adlı kitaptır. Burada YSA'ların doğrusal olmayan problemleri çözemediği meshur **XOR problemi** ile ispatlanmış, ve YSA çalışmaları bıçak gibi kesilmiştir.

Tüm bunlara rağmen **Anderson, Amari, Cooper, Fukushima, Grossberg, Kohonen, Hopfield** gibi bilim adamları çalışmalarını sürdürmüşler, 1972 de **Kohonen** ve **Anderson** **Associative memory** konusunda benzer çalışmalar yayınlamışlardır. **Kohonen** daha sonra 1982 yılında **Kendi kendine Öğrenme Nitelik Haritaları (Self Organizing Feature Maps SOM)** konusundaki çalışmasını yayınlamıştır. 1960'ların sonlarına doğru **Grosberg Carpenter** ile birlikte **Adaptif Rezonans Teorisini (ART)** geliştirmiştir.

1970'lerin sonlarına doğru **Fukushima, NEOCOGNITRON** modelini tanıtmıştır. Bu model şekil ve örüntü tanıma amaçlı geliştirilmiştir.

1982 1984 yıllarında **Hopfield** tarafından yayınlanan çalışmalar ile YSA'ların genelleştirilebileceği ve çözümü zor problemlere çözüm üretebileceğini göstermiştir. Geleneksel gezgin satıcı problemini çözmüştür. Bu çalışmaların neticesi **Hinton ve arkadaşları**'nın geliştirdiği **Boltzman Makinası**'nın doğmasına yol açmıştır.

1988 yılında, **Broomhead** ve **Lowe** **radyal tabanlı fonksiyonlar modelini (Radial Basis Functions RBF)** geliştirmişler ve özellikle filtreleme konusunda başarılı sonuçlar elde etmişlerdir. Daha sonra **Spect**, bu ağların daha gelişmiş şekli olan **Probabilistik ağlar (PNN)** ve **Genel Regresyon Ağlarını (GRNN)** geliştirmiştir [126]

Bu tarihten günümüze kadar sayısız çalışma ve uygulama geliştirilmiştir.

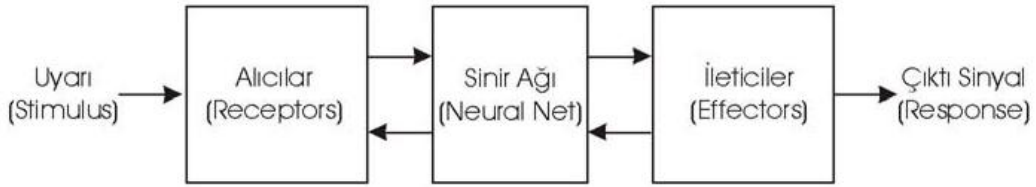
- **1890-** insan beyninin yapısı ve fonksiyonları ile ilgili ilk yayının yazılması
- **1911-** insan beyninin sinir hücrelerinden oluştuğu fikrinin benimsenmesi
- **1943-** yapay sinir hücrelerine dayanan hesaplama teorisinin ortaya atılması ve eşik değerli mantıksal devrelerin geliştirilmesi
- **1949-** öğrenme prosedürünün bilgisayarlar tarafından gerçekleştirilecek şekilde geliştirilmesi
- **1956 - 1962 ADALINE ve Widrow - Hoff** öğrenme algoritmasının geliştirilmesi
- **1957 - 1962 Perceptron'un** geliştirilmesi
- **1965-** ilk makina öğrenmesi kitabının yayınlanması
- **1967 - 1969** bazı gelişmiş öğrenme algoritmalarının geliştirilmesi (Grasberg vs.)
- **1969-** tek katmanlı algılayıcıların yetersizliklerinin ispatlanması
- **1969 - 1972** doğrusal ilişkilendiricilerin geliştirilmesi
- **1972-** korelasyon matris belleğinin geliştirilmesi
- **1974-** geriye yayılım modelinin geliştirilmesi
- **1978-** ART modelinin geliştirilmesi
- **1982-** çok katmanlı algılayıcıların geliştirilmesi
- **1984-** Boltzman Makinası'nın geliştirilmesi
- **1988-** RBF modelinin geliştirilmesi
- **1991-** GRNN modelinin geliştirilmesi [127]

4.6. YSA'nın Yapısı, Elemanları ve Mimarisi

Önceden de belirtildiği gibi, yapay sinir ağları insan beyninden esinlenilmiş yapılardır. Bu yüzden, YSA'ların yapısını anlamak için insan beyninin yapısını ve çalışma şeklini anlamak önemlidir. Bununla birlikte, tahmin edilebileceği gibi insan

beyni, karmaşık yapısı ile anlaşılması ve anlatılması zor bir konudur. Hatta, beynin çalışma şekli hala tam olarak anlaşılamamıştır. Çok farklı bir disiplinin konusu olan bu temanın burada açıklanması gereksiz kalmaktadır. Ancak, en azından konunun özünün anlaşılabilmesi açısından bazı yararlı bilgiler verilebileceği düşüncesi ile genel bir açıklama yapılmaktadır.

İnsan beyni, sinir sisteminin merkezini oluşturan temel elemandır. En basit şekilde, sürekli olarak iletilen bilgiyi alır, idrak eder (işler) ve uygun kararları vererek gerekli yerlere iletir. Çok basit görünmekle birlikte aslında oldukça karmaşık olan bu yapının Haykin (1999) tarafından kullanılan basit bir gösterimi Şekil 18’de sunulmaktadır [122].



Şekil 18 Sinir Sisteminin Blok Diyagramı

Şekilden takip edilebileceği gibi, dışarıdan veya başka bir organdan gelen sinyaller (uyarı) alıcılar yoluyla sinir ağına iletilir. Sinyaller burada işleminden geçirilerek çıktı sinyalleri oluşturulur. Oluşturulan çıktı sinyalleri ise ileticiler yoluyla dış ortama veya diğer organlara iletilirler. Şekil 18’de Sinir Ağı olarak gösterilen ortadaki bölüm, yani beyin, sinir sisteminin merkezi konumundadır. Beynin temel yapı taşları ise sinir hücreleri, diğer bir ifadeyle nöronlardır. Beyin, işlevini birbirleri arasında yoğun bağlantılar bulunan bu yapı taşları ile yerine getirir. Nöronların en belirgin özelliği, vücudun diğer bölümlerinin tersine yeniden üretilmeyen belirli bir hücre türü olmasıdır. Beynin diğer temel yapısal ve fonksiyonel birimleri olan bağlantılar (synapses) nöronlar arası etkileşimi sağlarlar. Dolayısıyla, beynin yapısında bu bağlantılar da önemli bir yer tutmaktadır. Zaten beynin oldukça etkin çalışan bir organ olmasının temel sebebi de bu yoğun bağlantılı yapısıdır. Bu yapı sayesinde beyin, bugünkü bilgisayar teknolojisinden kat kat daha etkin bir şekilde çalışabilmektedir.

Beynin bu yapısı etkinlik açısından oduđu kadar bir diđer önemli özellik açısından da önem taşımaktadır. Bu yoğun bağlantılı yapı beynin plastiklik (yenilenebilme) özelliđini de beraberinde getirmektedir. Plastiklik özelliđi, geliřmekte olan sinir sisteminin kendisini kuřatan çevreye adapte olmasını sađlamaktadır. Geliřmiř bir beyinde, plastiklik iki mekanizma ile izah edilebilmektedir.

Bunlar, nöronlar arasında yeni bağlantıların oluřturulması ve var olan bağlantıların modifiye edilmesidir. Ayrıca, bu özellik öğrenme kavramı açısından da çok büyük önem taşımaktadır.

Plastikliđin, nöronların iřlem yapabilmeleri için gerekli bir özellik olduđu görölmektedir. Çünkü, öğrenme süreci bağlantı ađırlıklarının deđiřtirilmesi veya yeni bağlantıların oluřturulması (hatta bazı bağlantıların iptal edilmesi) sayesinde gerçekeřmektedir. Bu iliřki, yapay nöronlar kullanarak beyinden esinlenilmiř sinir ađları oluřturulabilmesini sađlamaktadır.

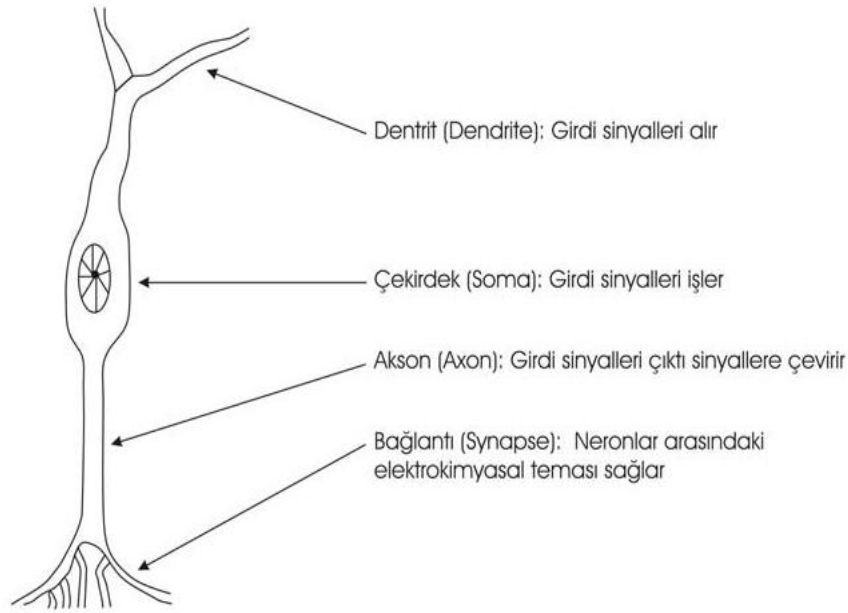
Yapay Sinir Ađları, beyinden esinlenildiklerinden dolayı benzer yapıya sahiptirler. Fakat, beynin yoğun bağlantılı ve komplike yapısının sadece beyine özel bir özellik olduđu belirtilmelidir. Bařka hiřbir yerde veya dijital bilgisayarda bulunmayan bu yapıya yakınsamak günümüz teknolojisine bile çok uzaktır. Yapay sinir ađlarını oluřturmak için kullanılan yapay nöronlar, beyindekilere kıyasla oldukça ilkel sayılırlar. Dolayısıyla, yapay nöronlar beynin yoğun bağlantılı ve komplike yapısından hala oldukça uzaktırlar ama genel yapı olarak tutarlıdırlar. Diđer bir ifadeyle, YSA'lar, komplike, çok yönlü ve güçlü bir organizma olan beynin sadece en temel elemanlarını kopyalamaya çalışmaktadır.

4.7. Yapay Nöron

Yapay sinir ađlarının genel yapı olarak beyinden esinlenildiđi daha önce belirtilmiřti. Dođal olarak, YSA'ların temel elemanları da yapay nöronlardır (bundan sonra yapay nöron yerine sadece nöron kelimesi kullanılmaktadır). Bu nöronlar, aralarındaki bağlantılar oluřturularak ve tabakalar halinde gruplandırılarak yapay sinir ađları oluřturulmaktadır.

İnsanın beyinsel gücünün bu yapı taşları birkaç genel işleve sahiptirler. Bir biyolojik nöron, temel olarak, diğer kaynaklardan girdiler alır, belirli bir şekilde bunları birleştirir, sonuç üzerinde bir işlem (genelde doğrusal olmayan) uygular ve nihai sonucu üretir. Şekil 19 bir nöronun dört temel elemandan oluşan genel yapısını ve bu dört eleman arasındaki ilişkiyi göstermektedir.

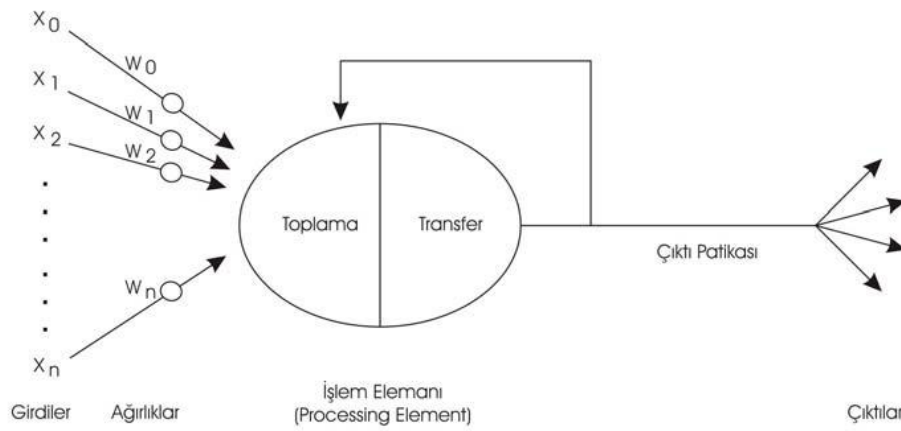
Şekilden de görüldüğü gibi, tüm doğal nöronlar dört temel bileşene sahiptir. Bu bileşenler biyolojik isimleri ile bilinirler: dentrit (dendrite), çekirdek (soma), akson (axon) ve bağlantı veya sinaps (synapse). Dentritler, çekirdeğin saça benzeyen uzantılarıdır ve girdi kanalları olarak işlev görürler. Bu girdi kanalları diğer nöronların sinapsları aracılığıyla girdilerini alırlar. Daha sonra çekirdek, gelen bu sinyalleri zaman içinde işler. Çekirdek, bu işlenmiş değeri bir çıktıya dönüştürdükten sonra bu çıktıyı akson ve sinapslar aracılığıyla diğer nöronlara gönderir.



Şekil 19 Biyolojik Nöronun Genel Yapısı ve İşlevleri

Yapay sinir ağlarının temel işlem elemanı olan yapay nöronlar, doğal nöronların dört temel fonksiyonunu simüle ederler. Bir yapay nöronun temel yapısı, genel haliyle,

Şekil 20’de sunulmaktadır. Burada girdi değerler $x(i)$ matematiksel sembolü ile gösterilmiştir ve bu gösterimde $i = 0,1,2,...,n$ değerlerini almaktadır. Bu girdi değerlerin her biri bir bağlantı ağırlığıyla çarpılmaktadır. Bu ağırlıklar ise $w(i)$ ile gösterilmektedir. En basit yapıda, bu çarpımlar toplanır ve bir transfer fonksiyonuna gönderilerek sonuç üretilir. Bu sonuç daha sonra bir çıktıya dönüştürülür. Bu elektronik uygulama değişik toplama fonksiyonları ve transfer fonksiyonları kullanabilir ve farklı ağ yapılarında uygulanabilir.



Şekil 20 Yapay Nöronun Genel Yapısı

Şekil 20 McCulloch ve Pitts (1943) tarafından tanımlanan, biyolojik nöronun basit matematiksel modelinin gösterimi olarak da düşünülebilir [128]. McCulloch-Pitts modeli incelendiğinde, toplama fonksiyonu olarak doğrusal bir fonksiyon ve transfer fonksiyonu olarak birim adım fonksiyonu kullanılmış olduğu görülmektedir. Temelde biyolojik nörona benzer bir şekilde, işlem elemanı toplama fonksiyonu sonucunun belirli bir eşik değerinin altında veya üstünde olmasına göre çıktısını iki ihtimal arasından seçerek oluşturmaktadır. Bu matematiksel modelin fonksiyonel gösterimi aşağıdaki gibidir.

$$y_i = \Psi(g(x)) = \Psi\left(\sum_{j=0}^n w_{ij} x_j\right); x_i = (x_0, x_1, \dots, x_n) \in \mathfrak{R} \quad (9)$$

Bu gösterimde, y_i çıktı değeri, $\Psi(\cdot)$ transfer fonksiyonunu, $g(\cdot)$ toplama fonksiyonunu, w_{ij} bağlantı ağırlıklarını ve Q_i ise i nolu nöron için eşik değerini göstermektedir. $\Psi(\cdot)$ transfer fonksiyonu birim adım fonksiyonudur.

$$\Psi(\cdot) = \begin{cases} 1; g(\cdot) \geq 0 \\ 0; g(\cdot) < 0 \end{cases} \quad (10)$$

$g(\cdot)$ toplama fonksiyonu ise x_j girdilerinin doğrusal bir bileşimi olarak düşünülmüştür:

$$g(\cdot) = \sum_{j=0}^n w_{ij} x_j - Q_i \quad (11)$$

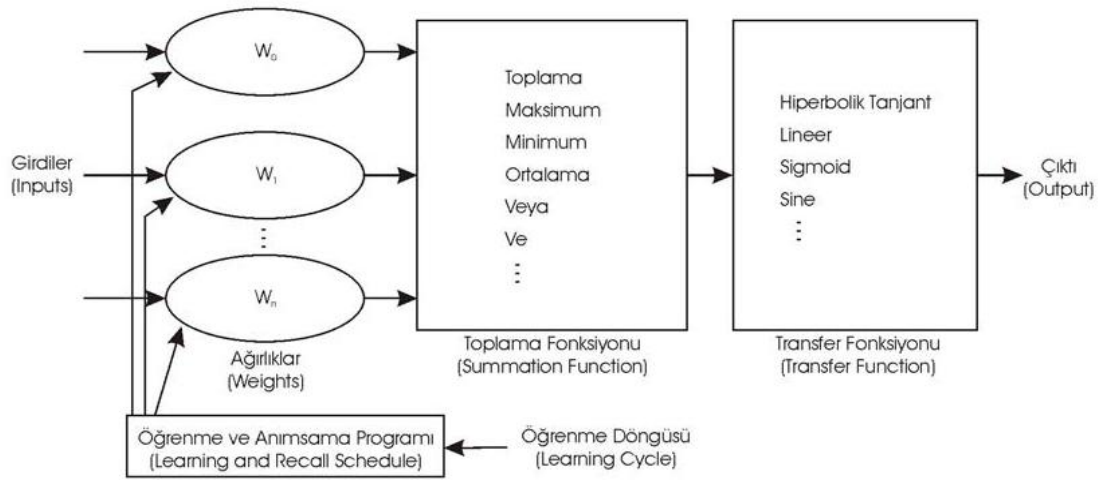
McCulloch-Pitts modelinde transfer fonksiyonu olarak birim adım fonksiyonu ve toplama fonksiyonu olarak doğrusal bir fonksiyon kullanılmakla birlikte YSA alanındaki gelişmelerle beraber farklı fonksiyonların kullanılabilmesi sağlanmıştır. Şekil 20 içinde YSA modellerinde kullanılabilecek farklı fonksiyonlara bazı örnekler sunulmaktadır. Basitçe görülebileceği gibi, hangi fonksiyon kullanılırsa kullanılsın, bir nöron modelinin matematiksel ve fonksiyonel gösterimi değişmeyecektir.

Bu yapıda kullanılan fonksiyonlar temelde uygulama konusuna bağlıdır. Bazı uygulamalar ikili (binary) veri seti gerektirir. Bu uygulamalara örnek olarak konuşma tanımlama ve metin tanıma uygulamaları verilebilir. Bu tür uygulamalar, doğal olarak, sadece toplama fonksiyonu içeren nöronlardan oluşan ağları kullanamaz. Bu ağlar mantıksal fonksiyonları kullanabilir ve bu fonksiyonlar toplama ve transfer fonksiyonlarına dahil edilebilir.

Veya bazı uygulamalar basit bir şekilde toplama ve belirli bir eşik değeri ile karşılaştırma şeklinde bir işleme gereksinim duyabilirler. Bu şekilde evet/hayır, doğru/yanlış veya 0/1 gibi iki olası sonuç arasından seçim yapılabilir. Bazı fonksiyonlar ise girdi verileri zamanla ilişkilendirebilir ve bu şekilde zamana bağlı ağlar oluşturulabilir.

Şekil 21’de, daha önce tanımlanan basit bir yapay nöron yapısının daha detaylı bir şeması sunulmaktadır. Şekilde, girdi değerler işlem elemanına üst sol bölümden girmektedir. İşlemde ilk adım, bu girdi değerlerin her birinin ilgili ağırlıklarla $w(i)$

ağırlıklandırılmalarıdır. Bir nöron genellikle, eşanlı olarak birçok sayıda girdi alır. Her girdinin kendi nispi ağırlığı vardır. Bu ağırlıklar, biyolojik nöronların değişen sinaptik etkililikleri ile aynı görevi üstlenirler. Her iki durumda da, bazı girdiler diğerlerine göre daha önemli hale gelirler. Bu sayede, işlem elemanın bir sinirsel tepki üretmesi işleminde daha fazla etkili olurlar. Ayrıca, ağırlıklar girdi sinyalin güçlülüğünü belirleyen adaptif katsayılarıdır. Yani, girdinin bağlantı gücünün bir ölçüsüdür. Bu bağlantı güçleri, çeşitli eğitim setlerine göre değiştirilebilirler.



Şekil 21 Yapay Nöronun Detaylı Yapısı

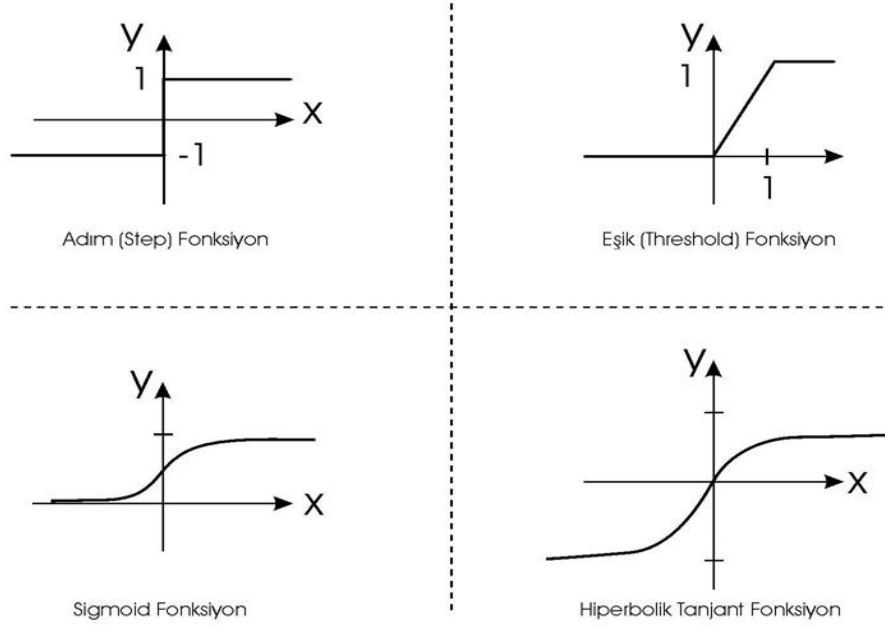
Ağırlıklandırmadan sonra, bu değiştirilmiş girdiler toplama fonksiyonuna gönderilirler. Toplama fonksiyonunda, adından da anlaşılacağı gibi, genelde toplama işlemi yapılmaktadır fakat bir çok farklı işlem çeşidi toplama fonksiyonu için kullanılabilir. Toplama fonksiyonu, bu basit çarpımlar toplamına ek olarak, minimum, maksimum, mod, çarpım veya çeşitli normalizasyon işlemlerinden birisi olabilir. Girdileri birleştirecek olan algoritma genellikle seçilen ağ mimarisine de bağlıdır. Bu fonksiyonlar farklı şekilde değerler üretebilir ve sonra bu değerler ileri doğru gönderilir. Ek olarak, uygulamacı kendi fonksiyonunu oluşturup toplama fonksiyonu olarak kullanabilir. Bazı toplama fonksiyonları, transfer fonksiyonuna iletmenden önce, sonuçları üzerinde ilave işlemler yaparlar. Bu işlem aktivasyon fonksiyonu olarak adlandırılan işlemdir. Bir aktivasyon fonksiyonu kullanmanın amacı toplama fonksiyonu çıktısının zamana bağlı

olarak deęişmesini saęlamaktır. Fakat, aktivasyon fonksiyonu literatürü henüz tam olarak gelişmemiştir. Bundan dolayı, çoęu yapay sinir aęında birim aktivasyon fonksiyonu kullanılmaktadır. Birim aktivasyon fonksiyonu ise bir aktivasyon fonksiyonu kullanılmaması ile aynı anlama gelmektedir.

Ayrıca, aktivasyon fonksiyonu, her işlem birimi için ayrı ayrı kullanılan bir bileşenden ziyade aęın genel bir bileşenidir. Yani, oluşturulan bir aę yapısında, tüm işlem elemanları aynı aktivasyon fonksiyonunu kullanırlar.

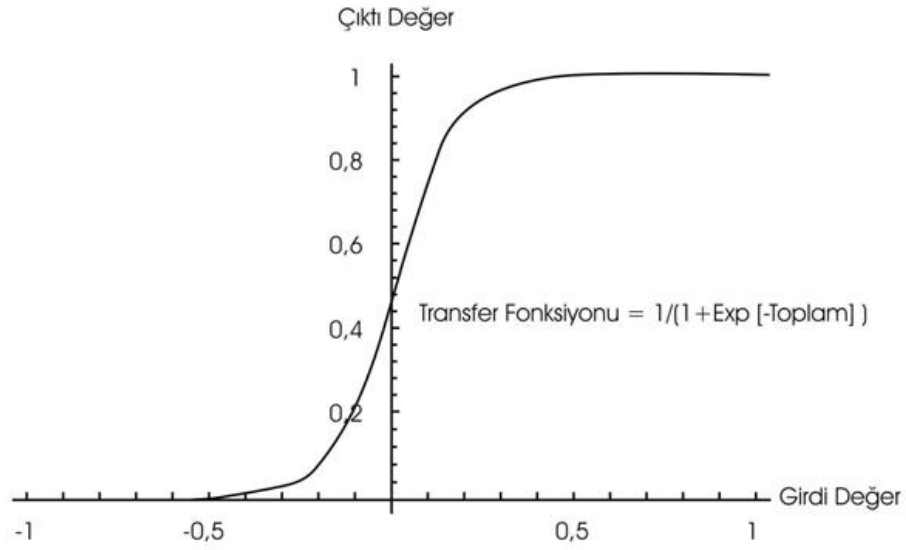
Sonraki aşamada toplama fonksiyonunun çıktısı transfer fonksiyonuna gönderilir. Bu fonksiyon, aldığı değeri bir algoritma ile gerçek bir çıktıya dönüştürür. Transfer fonksiyonu genellikle doğrusal olmayan bir fonksiyondur. Doğrusal fonksiyonlar genelde tercih edilmez çünkü doğrusal fonksiyonlarda çıktı, girdi ile orantılıdır. Bu durum, ilk YSA denemelerinin başarısızlıkla sonuçlanmasının temel nedenidir (Minsky ve Papert, 1969) [129]. Genellikle kullanılan transfer fonksiyonları eşik, sigmoid, hiperbolik tanjant vb. fonksiyonlardır. Bu fonksiyonlar arasında en çok kullanılanlar Şekil 22’da örneklenmiştir. Transfer fonksiyonunun sonucu genellikle işlem elemanının çıktısıdır. Transfer fonksiyonunun çalışma yapısı Şekil 23’de sigmoid fonksiyon kullanılarak örneklenmektedir. Sigmoid transfer fonksiyonu, toplama fonksiyonundan gelen ve şekilde ‘Toplam’ olarak gösterilen değeri alır ve sıfır ile bir arasında bir değere dönüştürür. Sıfır ile bir arasındaki bu değer transfer fonksiyonunun ve dolayısıyla işlem elemanının çıktısıdır ve dış ortama veya girdi olarak başka bir nörona iletilir.

Transfer fonksiyonu işlemi öncesinde, sisteme tekdüze dağılmış bir rassal hata eklenebilmektedir. Bu rassal hatanın kaynağı ve büyüklüğü, aęın öğrenme işlemi sürecinde belirlenir. Sisteme böyle bir hata teriminin eklenmesinin sebebi, insan beyninin işlevinin, içinde bulunduğu ortamın şartlarından (örnek olarak sıcak/soğuk olmasından) etkileniyor olmasıdır. Bu yüzden, YSA literatüründe rassal hata ekleme işlemi “sıcaklık ” olarak da adlandırılmaktadır. Günümüzde, rassal hata kullanımı fiilen tam olarak yerleşmemiştir ve halen bir araştırma süreci içerisinde. Ayrıca, bazı aęlarda, transfer fonksiyonunun çıktısı üzerinde başka işlemler, ölçeklendirme ve sınırlandırma, yapılabilmektedir.



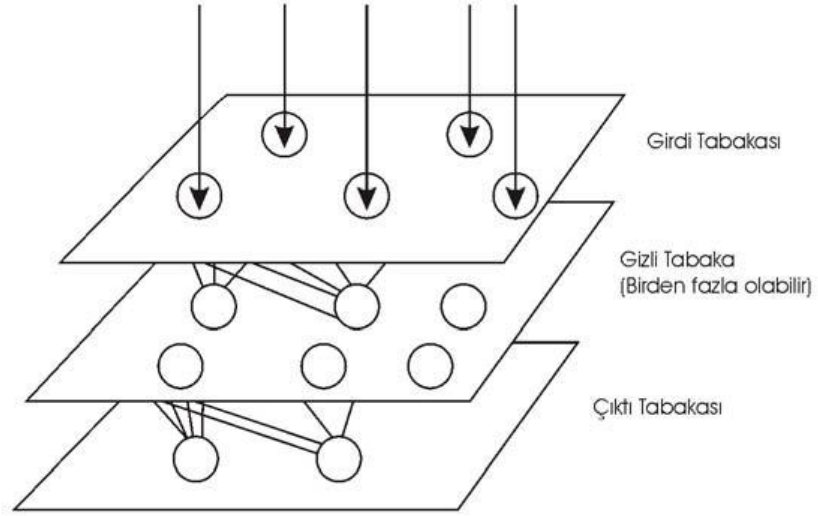
Şekil 22 En Çok Kullanılan Transfer Fonksiyonları

Transfer fonksiyonundan çıkan değer işlem elemanının da çıktısıdır. Fakat, bazı durumlarda işlem elemanının bu çıktıyı bir çıktı fonksiyonu ile bir dönüşüme uğratması gerekebilmektedir. Bu çıktı, ağı yapısına göre, girdi olarak başka bir işlem elemanına veya bir dış bağlantıya gönderilir. Tüm YSA'lar, yukarıda temel elemanları anlatılan bu temel yapı taşlarından yani nöronlardan oluşturulurlar. Bu yapı taşlarının dizaynı, sinir ağı sanatının, başka bir deyişle mimarisinin oluşturulmasının ilk bölümüdür. Bu sanatın ikinci bölümü ise bu işlem elemanlarının kümelenendirilmesi ve birbirleri arasındaki bağlantıların oluşturulmasını içerir. Beyinde kümelenendirme, bilginin dinamik, etkileşimli ve kendiliğinden organize bir şekilde işlenmesini sağlayacak şekildedir. Biyolojik sinir ağları üç boyutlu uzayda mikroskobik elemanlarla oluşturulur. Bu nöronlar hemen hemen sınırsız sayıda bağlantılar içerirler. Bu, yapay sinirler için mümkün değildir. Bugünkü teknoloji ile iki boyutlu ortamda ve belirli sayıda bağlantı içeren nöronlar oluşturulabilmektedir. Bu durum, YSA'ların yetenek ve çeşitlerini sınırlamaktadır.



Şekil 23 Transfer Fonksiyonunun Çalışma Yapısı

YSA’larda, yapay nöronlar basit bir şekilde kümelendirilmektedirler. Bu kümelendirme tabakalar halinde yapılmaktadır ve daha sonra bu tabakalar bir diğerine ilişkilendirilmektedir. Temel olarak, tüm YSA’lar benzer bir yapıya sahiptirler. Böyle bir genel yapı Şekil 24’de gösterilmektedir. Bu yapıda, bazı nöronlar girdileri almak için bazı nöronlar ise çıktıları iletmek için dış mekan ile bağlantılı haldedirler. Geri kalan tüm nöronlar ise gizli tabakalardadırlar, yani sadece ağ içinde bağlantıları vardır. Önemli olan nokta, bir sinir ağının, bir nöronlar yumağından daha komplike olması gerektiğidir. YSA’ların ilk yıllarında, bazı araştırmacılar, nöronlar arasındaki bağlantıları rassal olarak oluşturmuşlar ve olumsuz sonuçlarla karşılaşmışlardır. Bir yapıyı dizayn etmenin en kolay yolu elemanları tabakalandırmaktır. Burada tabakalandırmanın üç bölümü vardır. Bunlar, nöronları tabakalar halinde gruplandırmak, tabakalar arasındaki bağlantıları gruplandırmak ve son olarak ise toplama ve transfer fonksiyonlarını gruplandırmaktır.



Şekil 24 YSA'ların Genel Yapısı

Tek tabaka ya da tek eleman içeren bazı başarılı ağlar oluşturulabilmesine rağmen çoğu uygulamalar en az üç tabaka (girdi tabakası, gizli tabaka ve çıktı tabakası) içeren ağlara ihtiyaç duymaktadır. Girdi tabakası, dışarıdan girdileri alan nöronları içerir. Ayrıca, önemli olan bir nokta, girdi tabakasındaki nöronların girdi değerler üzerinde bir işlem uygulamamasıdır. Sadece girdi değerleri bir sonraki tabakaya iletirler ve bu yüzden de bazı araştırmacılar tarafından ağların tabaka sayısına dahil edilmezler. Çıktı tabakası ise çıktıları dışarı ileten nöronları içeren tabakadır. Girdi ve çıktı tabakaları tek tabakadan oluşurken bu iki tabaka arasında birden fazla gizli tabaka bulunabilir. Bu gizli tabakalar çok sayıda nöron içerirler ve bu nöronlar tamamen ağ içindeki diğer nöronlarla bağlantılıdır. Çoğu ağ türünde, gizli tabakadaki bir nöron sadece bir önceki tabakanın tüm nöronlarından sinyal alır. Nöron işlemini yaptıktan sonra ise çıktısını bir sonraki tabakanın tüm nöronlarına gönderir. Bu yapı ağın çıktısı için bir ileri besleme patikası oluşturur. Bu bir nörondan diğerine olan iletişim hattı, sinir ağları için önemli bir parçadır.

Bazı ağlarda, bir nöron aynı tabakadaki başka nöronlara engel (inhibit) oluşturabilir. Bu, yanıl engelleme (lateral inhibition) veya rekabet (competition) olarak adlandırılır ve en çok çıktı tabakasında kullanılır.

Nöronların diğer nöronlara bağlanma şekli ağı çalışmasını önemli derecede etkilemektedir. Bugün, büyük ve profesyonel yazılımlarda kullanıcı bu bağlantılar üzerinde istediği gibi ekleme, kaldırma ve kontrol işlemi yapabilmektedir.

Bu noktaya kadar sunulan bilgiler, YSA'ların matematiksel gösterimlerinin de daha net bir şekilde anlaşılmasını sağlayabilecektir. Daha önce, McCullogh-Pitts tarafından tanımlanan nöron modelinin fonksiyonel gösterimi (Denklem-9) verilmişti. Belirli bir fonksiyon varsayımı içerilmeden, Şekil 20'deki gibi basit bir yapay nöron için Denklem (9)'de verilen matematiksel notasyon şu şekilde genelleştirilebilir

$$y_i = \Psi(g(x)) = \Psi\left(w + \sum_{j=0}^n w_j x_j\right); x_i = (x_0, x_1, \dots, x_n) \in \mathfrak{R} \quad (11)$$

Burada değişken ve parametreler McCullogh-Pitts modelinde açıklandığı gibidir. Denklem (11) tek bir yapay nöronun matematiksel gösterimini vermektedir ama aynı zamanda tek nörondan oluşan bir sinir ağı gösterimi olarak da düşünülebilir. Oysa bir sinir ağı, tabakalar halinde sıralanmış bir çok nöron içermektedir. Dolayısıyla, tek nörona ait bu notasyonun bütün bir sinir ağının matematiksel gösterimini verecek şekilde düzenlenmesi gerekmektedir. $y=f(x_0, x_1, x_2, \dots, x_n)$ fonksiyonel ilişkisini modelleyen bir sinir ağı yapısı düşünülürse, girdi tabakasında $n+1$ adet nöron ve çıktı tabakasında bir adet nöron olacaktır. Sinir ağının gizli tabaka içermemesi durumunda ağın fonksiyonel gösterimi Denklem-4'deki tek nörona ait fonksiyonel gösterim ile aynı olacaktır. Çünkü, girdi tabakasındaki nöronlar girdi değerler üzerinde bir işlem uygulamazlar, sadece girdi değerleri bir sonraki tabakadaki nöronlara aktarırlar.

Bu yüzden, sinir ağları tanımlanırken tabaka sayısı bir eksik gösterilir. Diğer bir ifadeyle, bir girdi tabakası, bir gizli tabaka ve bir çıktı tabakasına sahip bir sinir ağı üç tabakaya sahip olmasına rağmen 2-tabakalı ağ olarak tanımlanır. Girdi tabakasında işlem yapılmamasından ve çıktı tabakasında da tek nöron olmasından dolayı, sinir ağı işlem yapılan tek nörona sahiptir ve doğal olarak fonksiyonel gösterimi Denklem (11)'deki gibi olacaktır.

Diğer taraftan, sinir ağının gizli tabaka içermesi durumunda fonksiyonel yapı değişecektir çünkü işlem yapılan nöron sayısı artacaktır. Bir gizli tabaka içerilmesi durumunda, fonksiyonel gösterim aşağıdaki gibi olacaktır:

$$y_k = f_k \left(\alpha_k + \sum_{j \rightarrow k} w_j f_j \left(\alpha_k + \sum_{j \rightarrow k} w_{ij} x_j \right) \right) \quad (12)$$

Denklem (12) daha önce açıklanmamış olan bazı parametre ve fonksiyonlar içermektedir. y_k çıktı değerleri gösterirken, f_k çıktı tabakası transfer fonksiyonunu göstermektedir. α_k çıktı tabakasına ait sapma değerini, W_j çıktı tabakasına ait ağırlıkları, f_j ve α_j sırasıyla gizli tabakaya ait transfer fonksiyonunu ve sapma değerini, x_i girdi değerleri ve w_{ij} ise i girdi elemanını j gizli elemanına bağlayan ağırlığı temsil etmektedir. İki gizli tabaka olması durumunda ise bu fonksiyonel gösterim Denklem (13) şeklinde olacaktır. Gizli tabaka sayısı arttıkça bu gösterim de benzer şekilde değişmeye devam edecektir.

$$y_l = f_l \left(\alpha_l + \sum_{k \rightarrow l} w_l f_k \left(\alpha_k + \sum_{j \rightarrow k} w_{ik} f_j \left(\alpha_j + \sum_{i \rightarrow j} w_{ij} x_j \right) \right) \right) \quad (13)$$

Belirli bir uygulamaya yönelik bir ağ yapılandırıldıktan sonra, bu ağ artık eğitime hazır durumdadır. Bu aşama, daha önce değinilmiş olan deneyim yoluyla öğrenme özelliği için kilit önem taşımaktadır. Çünkü bu, bağlantı ağırlıklarının belirlendiği aşamadır. Genel olarak, başlangıç ağırlıkları rassal olarak seçilir ve eğitime, ya da diğer bir ifadeyle öğrenme işlemi başlar. Eğitime işlemi için “yönlendirmeli (supervised)” ve “yönlendirmesiz (unsupervised)” olmak üzere iki yaklaşım vardır. Yönlendirmeli eğitime, ağın çıktı için istenilen veri değerleri verebilmesi için girdi-çıkı ilişkisini elde edebilmesini sağlayacak bir mekanizma içermektedir.

Yönlendirmesiz eğitime ise dış müdahale olmaksızın, girdilerin ağ tarafından analiz edilmesi ve bu analiz sonucunda bağlantıların oluşturulmasıdır. Kullanılan ağların büyük çoğunluğu yönlendirmeli eğitimi kullanır.

Yönlendirmesiz eğitime, girdiler için bazı karakter belirleme durumlarında kullanılır. Bununla beraber, kendi kendine öğrenme kavramı parlak bir gelişme potansiyeli taşısa da günümüzde tam olarak çalışmamaktadır.

Yönlendirmeli eğitimde hem girdi hem de çıktı veriler kullanılır. Öncelikle, ağ rassal olarak belirlenen başlangıç ağırlıklarını kullanarak girdileri işler ve çıktıyı istenilen çıktı ile karşılaştırır. Elde edilen hatalar sistem içinde geriye gönderilir ve bu hatalar

kullanılarak ağı kontrol eden bağlantı ağırlıkları güncellenir. Bu işlem defalarca tekrarlanır ve bağlantı ağırlıkları sürekli olarak ayarlanır (tweaked). Eğitim sırasında kullanılan veri seti “eğitme veri kümesi (training set)” olarak adlandırılır. Bir ağı eğitilmesi sırasında, aynı veri seti bağlantı ağırlıkları belirleninceye kadar defalarca işleminden geçirilir.

Bugün, yapay sinir ağı oluşturmaya yönelik programlar, bir ağı doğru cevabı öngörebilme yeteneğine nasıl yakınsadığını gözlemlemeye ve test etmeye yönelik araçlar sağlamaktadır. Diğer taraftan, sistem yalnızca (istatistiksel olarak) istenilen noktaya veya doğruluğa ulaşınca durdurulmaktadır ve bu durum da diğer araçlarla birleşince eğitim işlemleri günlerce sürebilmektedir. Bu arada, bazı ağlar öğrenme işlevini hiç yapamayabilir. Bunun sebebi girdi verinin istenen çıktıya ait belirli bir bilgi taşınamaması olabilir. Ayrıca, tam öğrenmeyi sağlayacak kadar yeterli veri olmaması durumunda ağ yakınsamayabilir. İdeal olarak, gerekli testlerin yapılabilmesi için veri setinin bir bölümünün ayrılabilmesi kadar geniş bir veri seti gereklidir. Fazla sayıda işlem elemanı içeren çok tabakalı ağlar veri için hafıza oluşturabilme yeteneğine sahiptirler. Ağı hafızaya alma sürecinin yeterli olup olmadığının gözlenebilmesi için ise, yönlendirmeli eğitimde veri setinin bir bölümü, eğitim sonrasında gerekli testlerin yapılabilmesi için ayrılmalıdır.

Diğer taraftan, bir ağı ilgili problemi çözmemesi durumunda kullanıcının yapabilecekleri iki gruba ayrılabilir. İlk grup ağı yapısının gözden geçirilmesini kapsar. Daha açık olmak gerekirse, girdi ve çıktı verileri, tabaka sayısı, her tabakadaki eleman sayısı, tabakalar arasındaki bağlantılar, toplama, transfer ve eğitim fonksiyonları ve hatta başlangıç ağırlıkları gözden geçirilmelidir.

Tüm bu kriterler, yapay sinir ağlarının sanatsal bölümü olan başarılı bir ağ oluşturmak için gereklidir. Diğer grup ise kullanıcının tercihi ve yaratıcılığına bağlı olan eğitim kurallarını içermektedir. Eğitim sırasında ağırlıkların ayarlanabilmesi için gerekli adaptif geri beslemeyi sağlayacak çok sayıda değişik eğitim kuralı (algoritma) vardır. En yaygın olan teknik geriye doğru hata beslemesi ya da bilinen ismiyle geri yayılmadır. Bu eğitim kuralları daha sonra ayrıntılı olarak açıklanacaktır.

Eğitime konusunda diğer bir önemli nokta ise öğrenmenin sürekli devam edeceğidir. Bir optimum noktaya gelindiğinde, YSA veri setinin genel istatistiksel trendine göre kendisini biçimlendirir. Bu noktadan sonra ise eğitmeye devam edilmesi durumunda ağ öğrenmeye devam edecektir. Bu aşamada, ağ veri setinden hatalı (spurious) ilişkiler çıkartmaya başlayabilir. Bu yüzden aşırı eğitime sorununa dikkat edilmeli ve eğitime uzunluğu iyi ayarlanmalıdır.

Eğitime işlemi doğru bir şekilde tamamlandığında, yani hem daha fazla öğrenmeye gerek kalmamış hem de aşırı eğitime yapılmamışsa, istenildiği takdirde ağırlıklar dondurulabilir. Bazen, ağın ortaya çıkan son hali bir donanıma (hardware) çevrilerek daha hızlı çalışması sağlanabilmektedir. Diğer sistemler ise kullanılırken de öğrenmeye devam edebilirler.

Diğer eğitime yaklaşımı olan yönlendirmesiz yaklaşım ayrıca adaptif eğitime olarak da adlandırılır. Bu eğitime yaklaşımında, ağa girdi sağlanır ama istenilen çıktı değerler sağlanmaz. Sistem girdi veriyi gruplandırmak için hangi özellikleri kullanacağına kendi kendisine karar verir ki bu yöntem kendi kendine öğrenme (self-organization) veya adaptasyon olarak bilinir. Günümüzde, yönlendirmesiz eğitime tam olarak anlaşılamamış durumdadır. Adaptif eğitmenin öncü araştırmacılarından birisi Tuevo Kohonen'dir. Kohonen, doğru cevabı bilmenin yararlarından yararlanmadan öğrenen bir ağ geliştirmiştir. Bu ağ, bir çok sayıda bağlantısı olan tek tabakaya sahip olması nedeniyle biraz sıra dışı sayılabilir.

Kohonen, daha sonra çalışmalarını bu standart yapının dışındaki ağlara, ileri besleme ve geri yayılma yaklaşımlarına yöneltmiştir. Kohonen'in çalışması nöronları belirli alanlara gruplandırmakla ilgilidir. Bir alanın içerdiği nöronlar topolojik olarak sıralanmıştır (topologically ordered). Topoloji, matematiğin bir dalıdır ve geometrik yapıyı değiştirmeden bir boyuttan diğerine haritalamanın (mapping) nasıl yapılacağı ile ilgilenir. Kohonen, sinir ağı modellerinde topolojik sıralamanın olmamasının bugünkü sinir ağlarını, beyin içindeki gerçek sinir ağlarının basit bir çıkarımsaması (abstraction) haline getirdiğini belirtmiştir. Bu araştırmalar devam ettikçe daha etkili kendi kendine öğrenen ağlar elde edilebilir. Fakat, şu an için bu alan sadece deney ortamlarında kalmaktadır.

Öğrenme (veya adaptasyon) YSA'ların yapısı içinde önemli bir yere sahiptir. Çünkü, YSA'ların bazı önemli özellik ve avantajlarının kaynağını oluşturmaktadır. Bu yüzden, YSA yapısı içindeki öğrenme sürecine yönelik elemanlar büyük önem taşımaktadır. Bu elemanların ilki öğrenme fonksiyonudur. Öğrenme fonksiyonunun amacı her işlem elemanının girdilerine ait değişken bağlantı ağırlıklarını ayarlamaktır.

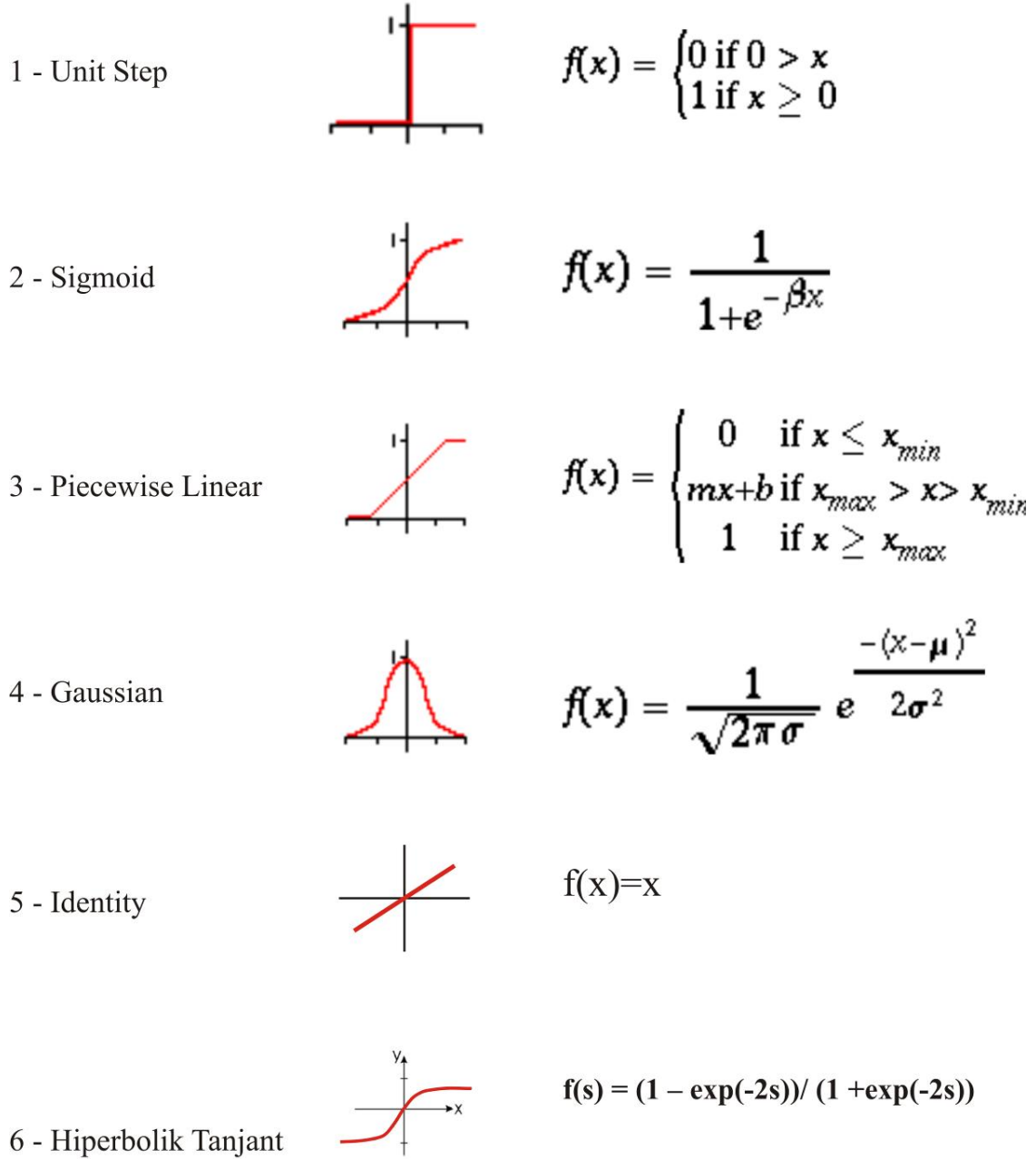
Girdi bağlantı ağırlıklarının, istenilen sonucu elde edecek şekilde değiştirilmelerini sağlayan bu işlem adaptasyon fonksiyonu olarak da adlandırılmaktadır.

İkinci eleman ise hata fonksiyonudur. Öğrenme fonksiyonunun gerekli ayarlamaları yapabilmesi için yanılma payının biliniyor olması gerekmektedir. Hata fonksiyonu, bu amaca yönelik olarak, o anki çıktı ile istenilen çıktı arasındaki farkı, hatayı, hesaplar ve gerekiyorsa bir transformasyon uygular. Bu hata, literatürde cari hata (current error) olarak adlandırılır ve bu hata veya transformasyonu sağlanmış hali genellikle önceki tabakaya geri yayılır.

Diğer bir eleman ise öğrenme oranıdır. Öğrenme oranı, öğrenme sürecinin hızı ve işlevi açısından önemlidir. Çünkü, YSA'ların öğrenme gücü ile hızı ters orantılıdır. Basit bir şekilde, bir adımda daha fazla öğrenme, daha düşük bir hız ve dolayısıyla daha fazla zaman anlamına gelmektedir. Diğer bir ifadeyle, daha fazla hız daha az öğrenme anlamına gelmektedir. Sonuç olarak, bir ağın ne kadar eğitileceği sorusu öğrenme oranına bağlıdır. Öğrenme oranının belirlenmesinde ise ağın karmaşıklık düzeyi, büyüklüğü, mimarisi, kullandığı öğrenme kuralı ve istenilen doğruluk derecesi gibi bir çok faktör rol oynar. Çoğu öğrenme fonksiyonu, öğrenme oranı için belirli standartlara sahiptir. Öğrenme oranı genellikle (0,1) gibi bir aralık içinde belirlenir. Bu aralıkta, öğrenme oranının küçük değer alması, yavaş bir öğrenme süreci getirecektir. Diğer taraftan ise, öğrenme sürecinin küçük adımlar halinde olması maksimum doğruluk derecesine yakınsamayı getirebilecektir.

Yapay Sinir Ağlarının çalışma şeklini basit bir şekilde anlatan bir örnek model çözümü sunulmuştur. YSA metodolojisi ve teorisi hakkında buraya kadar anlatılanların daha net anlaşılmasını ve daha sonra anlatılan uygulamalar ile kolaylıkla birleştirilebilmesini sağlayabileceği düşünülen bu örnek çözüm Balkin (2001)'den

alınmıştır [130]. Şekil 25’de YSA da kullanılan temel aktivasyon fonksiyonları görülmektedir.



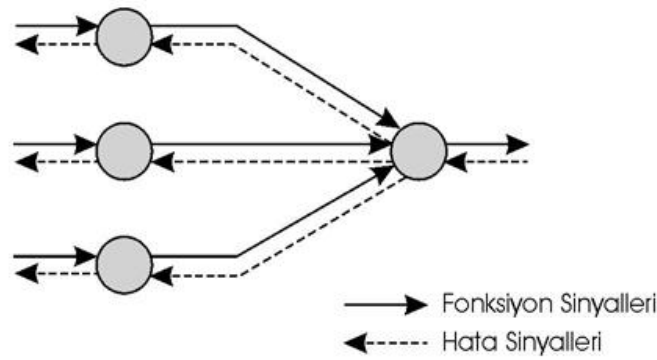
Şekil 25 Aktivasyon Fonksiyonları

4.8. YSA Çeşitleri

YSA'ların çok sayıda farklı çeşitleri vardır. Bu farklılıkların kaynağı mimarisi, öğrenme yöntemi, bağlantı yapısı vb. olabilmektedir. Genel olarak, YSA'lar üç ana kritere göre sınıflandırılmaktadırlar. Bu kriterlerden biri öğrenme yöntemidir. Önceki bölümde belirtildiği gibi, temel olarak iki çeşit öğrenme algoritması vardır: yönlendirmeli öğrenme ve yönlendirmesiz öğrenme. Her yöntemin kullandığı öğrenme kuralı değişebilmekteyse de, YSA'lar bu iki algoritmaya göre sınıflandırılırlar.

İkinci bir sınıflandırma, ağıın kullandığı veriye göre yapılmaktadır. Temel olarak, kalitatif ve kantitatif olmak üzere iki tür veri vardır. Kalitatif verilerle çalışan ağlar, ister yönlendirmeli ister yönlendirmesiz öğrenme kullansın, sınıflandırma ağları olarak bilinirler. Kantitatif veriler kullanan yönlendirmeli eğitime ise regresyon olarak adlandırılmaktadır.

Birbirine zıt yönde çalışan bu iki sinyal çeşidi fonksiyon sinyalleri ve hata sinyalleri olarak adlandırılmaktadırlar [131]. Şekil 26'de küçük bir ağ bölümü kullanılarak bu sinyallerin yönleri gösterilmektedir. Şekilde, sol taraf girdi veri tarafını, sağ taraf çıktı tarafını ve yuvarlaklar ise nöronları göstermektedir.



Şekil 26 Eğitime Sürecindeki Sinyal Çeşitleri

Fonksiyon sinyali, ağa girdi tabakasından giren, ağ içinde ileri doğru yayılan ve çıktı tabakasından çıktı sinyali olarak çıkan bir girdi sinyalidir. Bu tür bir sinyalin fonksiyon sinyali olarak adlandırılmasının iki temel sebebi vardır. İlk olarak, ağın çıktısı

için gerekli fonksiyonları uyguladığı kabul edilmektedir. İkinci sebep ise fonksiyon sinyalinin geçtiği her nöronda, sinyal girdilerin ve o nörona uygulanan ağırlıkların bir fonksiyonu olarak hesaplanır. Fonksiyon sinyalleri, girdi sinyalleri olarak da adlandırılmaktadırlar. Hata sinyali ise, fonksiyon sinyalinin tersine, çıktı tabakasından başlar ve tabaka tabaka geriye doğru yayılır. Hata sinyali olarak adlandırılmasının sebebi, tüm nöronlarda bir hata tabanlı fonksiyon ile hesaplanmasıdır.

Ağın genel yapısına dönersek, tabaka sayısı ve tabakaların içerdiği işlem elemanı sayısı ağın performansı açısından önemli ve zor kararlardır. Zor karar olmalarının sebebi ise herhangi bir uygulama için net bir seçim kriterinin olmamasıdır. Bunun yerine, uygulamalar sonucunda ortaya çıkmış ve araştırmacılar tarafından benimsenmiş bazı kurallar vardır. Bunlar:

Kural-1: Girdi ve çıktı veriler arasındaki ilişkinin karmaşıklık derecesi arttıkça, tabakaların içerdiği işlem elemanı sayısı da artmalıdır.

Kural-2: Modellenen konu değişik safhalara ayrılabilirse, tabaka sayısının artırılması gerekebilir.

Kural-3: Eldeki eğitme verisinin genişliği, gizli tabakalardaki toplam nöron sayısı için bir üst limit kriteri oluşturur.

Geri yayılma ağlarda çok çeşitli öğrenme kuralı, hata fonksiyonları ve transfer fonksiyonları kullanılabilir. Öğrenme kuralı olarak genellikle Delta Kuralı'nın bir varyantı kullanılmaktadır. Delta kuralı, ağın çıktısı ile istenilen çıktı arasındaki farkın hesaplanması ile başlar. Bu hata kullanılarak bağlantı ağırlıkları belirli bir doğruluk derecesi faktörüne göre güncellenir. Bu öğrenme mekanizmasının komplike olan tarafı, hatalı çıktı üretilmesinde hangi işlem elemanının daha etkili olduğunun belirlenmesi ve hatanın düzeltilmesi için bu işlem elemanının nasıl değiştirileceğidir. Bu noktada aktif olmayan bir nod hataya sebep olamaz ve dolayısıyla ağırlıklarını değiştirmeye gerek yoktur. Bu sorunun çözümü için, eğitme setine ait girdi veriler ağın girdi tabakasına sunulur ve istenilen çıktılarla karşılaştırma çıktı tabakasında gerçekleştirilir.

Öğrenme işlemi süresince, ağ içinde ileri doğru bir bilgi akışı vardır ve tabaka tabaka her işlem elemanının çıktısı hesaplanır. Çıktı tabakasına ulaşıldığında, bu

tabakanın çıktısı ile istenilen çıktı arasındaki fark hesaplanır ve bu hata önceki tabakalara iletilir (geri yayılma). Bu süreçteki önemli nokta ise, hata önceki tabakalara iletilirken, transfer fonksiyonunun türevi ile bir transformasyon işlemi uygulanmasıdır. Hatanın iletilmesi, tabaka tabaka geriye doğru olur ve bu süreçte Delta Kuralı ile bağlantı ağırlıkları ayarlanır. İşlem, girdi tabakasına ulaşılan kadar devam eder ve bu noktada yeni bir döngüye başlar.

Geri yayılma algoritmasında, Delta Kuralı ile bağlantıların ayarlanmasının matematiksel gösterimi şu şekilde özetlenebilir. Delta Kuralı, temel olarak, ilgili bağlantı ağırlığının ayarlanması için gerekli olan düzeltme miktarını formüller. Buna göre, nöron(i) ve nöron(j) arasındaki bağlantı için düzeltme miktarı şu şekilde hesaplanır.

$$\text{Ağırlık düzeltme miktarı} = \text{Öğrenme oranı} \times \text{Yerel değişim} \times \text{Nöron} \quad (14)$$

veya matematiksel formül olarak:

$$\Delta w_{ji}(n) = \eta \cdot \delta_j(n) \cdot Y_i(n) \quad (15)$$

Burada dikkat edilmesi gereken nokta, yerel değişimin hesaplanma şeklinin nöron'un çıktısı veya gizli nöron olmasına göre değişiklik gösterdiğidir.

Buna göre:

- • Nöron(j) bir çıktı nöronu ise, yerel değişim, nöron(j)'ye ait hata sinyali ve fonksiyon sinyalinin türevinden hesaplanmaktadır.
- • Nöron(j) bir gizli nöron ise, yerel değişim, fonksiyon sinyalinin türevi ve bir sonraki tabakadaki nöronlara ait değişimlerin ağırlıklı toplamı kullanılarak hesaplanır.

Burada, Delta Kuralının içerdiği düzeltme formülünün mantığı ve bileşenleri hakkında bilgi bir özet halinde sunulmuştur. Daha ayrıntılı bilgi içinse [122] ve [132]'e bakılabilir. Ayrıca, diğer öğrenme kuralları için bazı temel bilgilere yer verilmiştir.

Son olarak, geri yayılma ağların bazı kısıtlamalara da sahip olduğu belirtilmelidir. Geri yayılma mekanizması oldukça geniş bir girdi-çıkı veri seti ile geniş çaplı bir yönlendirmeli eğitmeye ihtiyaç duyar. Ek olarak, içsel haritalama yapısı tam olarak

anlaşılmadığından sistemin istenilen doğruluk derecesine yakınsayamaması ihtimali de mevcuttur.

4.9. Sinir Sistemi ve Ysa'nın Benzerlikleri

1- Sinir Sistemi	–	YSA Sistemi
2- Nöron	–	İşlem Elemanı
3- Dendrit	–	Toplam Fonksiyonu
4- Hücre Gövdesi	–	Transfer Fonsiyonu
5- Aksonlar	–	Eleman Çıkışı
6- Sinapslar	–	Ağırlıklar

4.10. Yapay Sinir Ağlarının Genel Özellikleri

- YSA'lar ani bozulma göstermezler. Yapay sinir ağlarının hata toleransına sahip olmaları dereceli bozulma gösterebilmelerini sağlar.
- Dağıtık bilgiye sahiptirler. Ağın bilgisi bağlantılara yayılmış haldedir.
- Sadece nümerik bilgiler ile çalışabilmektedirler.
- YSA'lar normal yollarla çözülmesi zor olan problemleri çözmek için tasarlanmışlardır. (sınıflandırma tahminleme gibi)
- Literatürde 100'den fazla yapay sinir ağı modeli vardır.

4.11. YSA'nın Avantajları

Lineer olmayan özelliğe sahip olmaları

Öğrenme kabiliyeti

Genelleme yapma

Adaptasyon kabiliyeti

Gürültüye karşı toleransı

Donanım olarak gerçekleştirilmesi

Paket yazılımların bulunması

4.12. YSA'nın Dezavantajları

- a. Donanım bağımlıdır.
- b. Uygun ağ yapısının belirlenmesinde belli bir kural yoktur.
- c. Ağın parametre değerlerinin belirlenmesinde belli bir kural yoktur.
- d. Öğrenilecek problemin ağa gösterimi önemli bir problemidir.
- e. Ağın eğitiminin ne zaman bitirilmesi gerektiğine ilişkin belli bir yöntem yoktur.
- f. Ağın davranışlarının açıklanamamaktadır.

4.13. Ysa'ların Sınıflandırılması

4.13.1 Mimari Yapılarına Göre

- a. Geri beslemeli (Elman, Jordan)
- b. İleri beslemeli (MLP, LVQ)

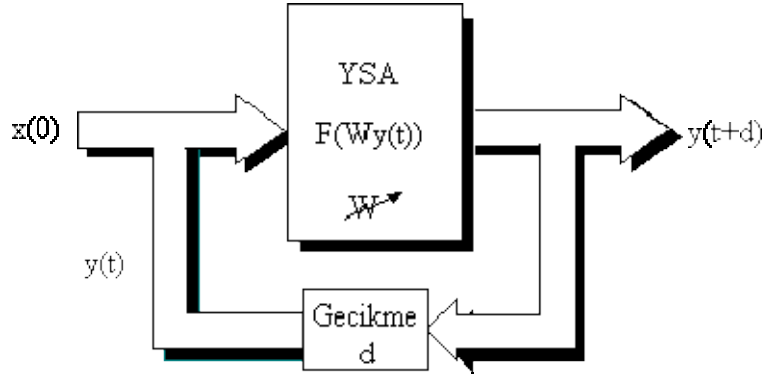
4.13.2 Öğrenme Yaklaşımlarına Göre

- a. Danışmanlı öğrenme (MLP-BP)
- b. Danışmansız öğrenme (ART, SOM)
- c. Takviyeli öğrenme (GA)

4.13.2.1 .Geri Beslemeli (ELMAN, JORDAN):

Bir geri beslemeli sinir ağı, çıkış ve ara katlardaki çıkışların, giriş birimlerine veya önceki ara katmanlara geri beslendiği bir ağ yapısıdır. Böylece, girişler hem ileri yönde hem de geri yönde aktarılmış olur. Şekil 27’de bir geri beslemeli ağ görülmektedir. Bu çeşit sinir ağlarının dinamik hafızaları vardır ve bir andaki çıkış hem o andaki hem de önceki girişleri yansıtır. Bundan dolayı, özellikle önceden tahmin uygulamaları için uygundur. Bu ağlar çeşitli tipteki zaman-serilerinin tahmininde oldukça başarılı

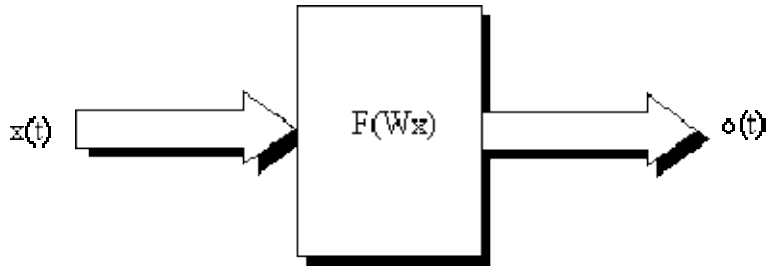
sağlamışlardır. Bu ağlara örnek olarak Hopfield, SOM (Self Organizing Map), Elman ve Jordan ağları verilebilir.



Şekil 27 Geri beslemeli ağ için blok diyagram

4.13.2.2 İleri Beslemeli (MLP, LVQ)

İleri beslemeli bir ağda işlemci elemanlar (İE) genellikle katmanlara ayrılmışlardır. İşaretler, giriş katmanından çıkış katmanına doğru tek yönlü bağlantılarla iletilir . İE'ler bir katmandan diğer bir katmana bağlantı kurarlarken, aynı katman içerisinde bağlantıları bulunmaz. Şekil 28'de ileri beslemeli ağ için blok diyagram gösterilmiştir. İleri beslemeli ağlara örnek olarak çok katmanlı perseptron (Multi Layer Perseptron-MLP) ve LVQ (Learning Vector Quantization) ağları verilebilir.



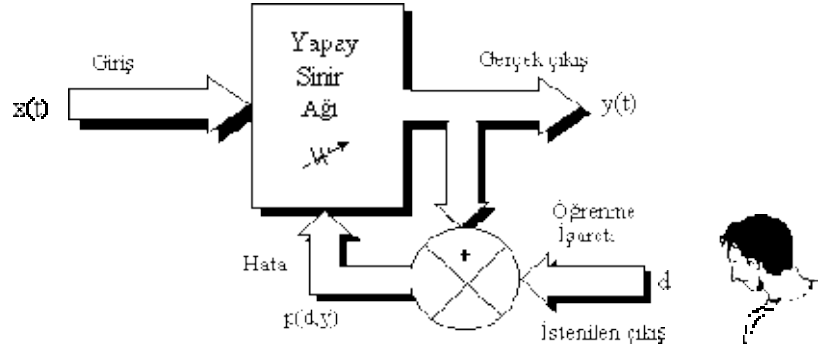
Şekil 28 İleri beslemeli ağ için blok diyagram

4.13.2.3 .Danışmanlı Öğrenme (MLP-BP)

Bu tip öğrenmede, YSA'ya örnek olarak bir doğru çıkış verilir. İstenilen ve gerçek çıktı arasındaki farka (hataya) göre İE'ler arası bağlantıların ağırlığını en uygun çıkışı elde etmek için sonradan düzenlenebilir. Bu sebeple danışmanlı öğrenme

algoritmasının bir “öğretmene” veya “danışmana” ihtiyacı vardır. Şekil 29’da danışmanlı öğrenme yapısı gösterilmiştir.

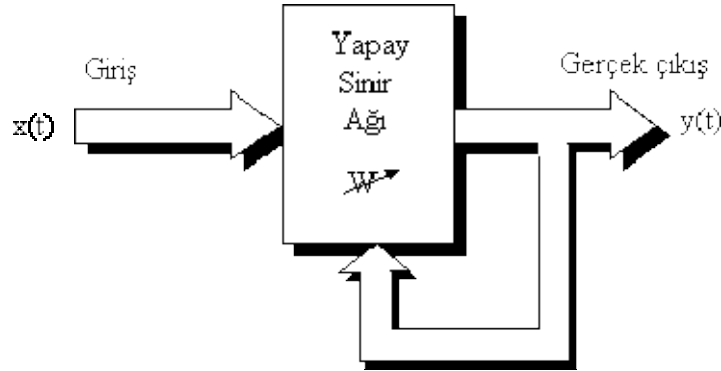
Widrow-Hoff tarafından geliştirilen delta kuralı ve Rumelhart ve McClelland tarafından geliştirilen genelleştirilmiş delta kuralı veya geri besleme (back propagation) algoritması danışmanlı öğrenme algoritmalarına örnek olarak verilebilir.



Şekil 29 Danışmanlı öğrenme yapısı

4.13.2.4 B.Danışmansız Öğrenme (ART, SOM)

Girişe verilen örnekten elde edilen çıkış bilgisine göre ağ sınıflandırma kurallarını kendi kendine geliştirmektedir. Bu öğrenme algoritmalarında, istenilen çıkış değerinin bilinmesine gerek yoktur. Öğrenme süresince sadece giriş bilgileri verilir. Ağ daha sonra bağlantı ağırlıklarını aynı özellikleri gösteren desenler (patterns) oluşturmak üzere ayarlar. Şekil 30’de danışmansız öğrenme yapısı gösterilmiştir. Grossberg tarafından geliştirilen ART (Adaptive Resonance Theory) veya Kohonen tarafından geliştirilen SOM (Self Organizing Map) öğrenme kuralı danışmansız öğrenmeye örnek olarak verilebilir.

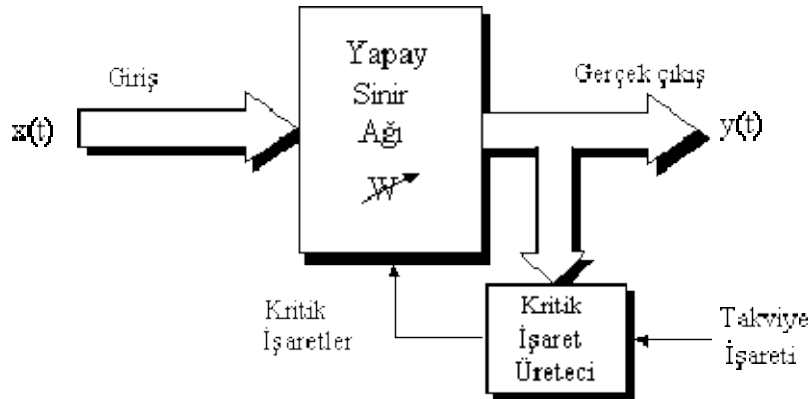


Şekil 30 Danışmansız öğrenme yapısı

4.13.2.5 .Takviyeli Öğrenme (GA)

Bu öğrenme kuralı danışmanlı öğrenmeye yakın bir metoddur. Denetimsiz öğrenme algoritması, istenilen çıkışın bilinmesine gerek duymaz. Hedef çıktıyı vermek için bir “öğretmen” yerine, burada YSA’ya bir çıkış verilmemekte fakat elde edilen çıkışın verilen girişe karşılık iyiliğini değerlendiren bir kriter kullanılmaktadır. Şekil 31’de takviyeli öğrenme yapısı gösterilmiştir.

Optimizasyon problemlerini çözmek için Hinton ve Sejnowski’nin geliştirdiği Boltzmann kuralı veya GA takviyeli öğrenmeye örnek olarak verilebilirler.



Şekil 31 Takviyeli öğrenme yapısı

4.14. Öğrenme Kuralları

- 1- **Hebb Kuralı:** İlk ve en iyi bilinen kuraldır.

2- **Hopfield Kanunu:** İstenilen çıkış ile giriş aktif veya aktif değil ise ağırlıklar öğrenme katsayısıyla arttırılır. Diğer durumlarda ise ağırlıklar öğrenme katsayısıyla azaltılır.

3- **Kohonen's Öğrenme Kanunu:** Nöronlar öğrenmek ve ağırlıklarını güncellemek için yarışır.

4- **Delta Kuralı:** Windrow-Hoff, Least Mean Square

4.15. Arakat Bağlantılarına Göre Sınıflandırılması

4.15.1 Katlar Arası Bağlantılar

Tamamen Bağlantılı

Kısmen Bağlantılı

İleri Beslemeli

Çift Yönlü

Hiyerarşik

Resonans

4.15.2 Diğer Katlarla Bağlantılar

Recurrent

On-center/off

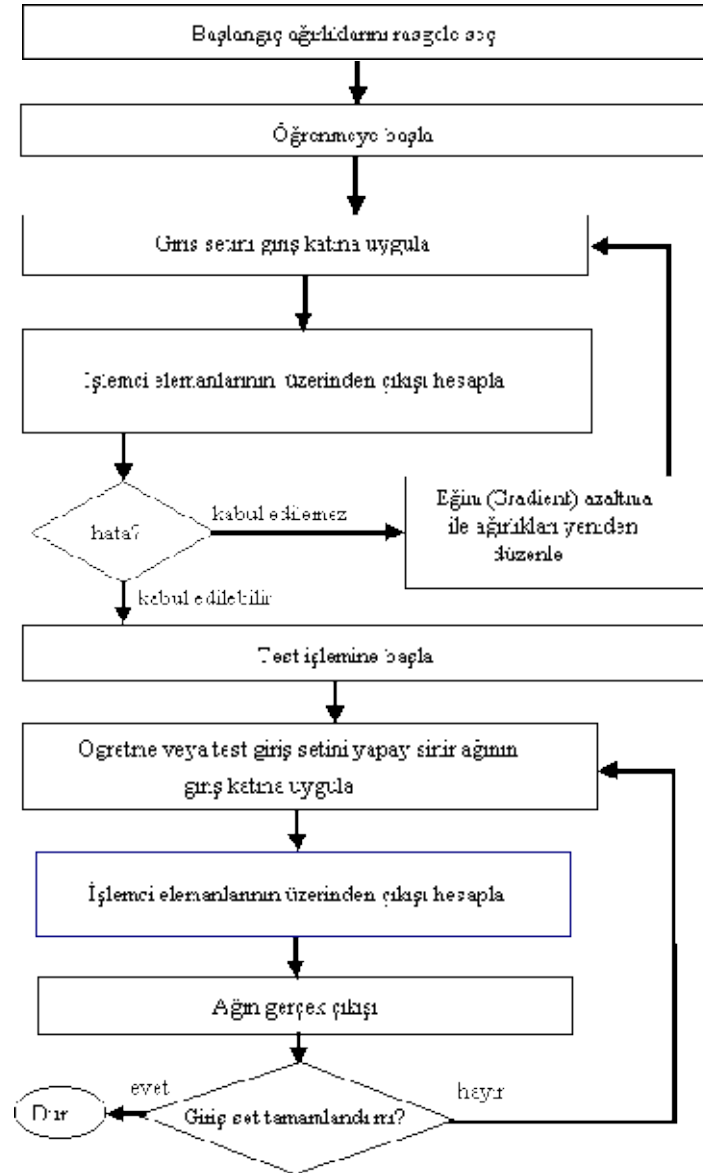
4.16. Yaygın Olarak Kullanılan Öğrenme Algoritması

1. Geri-yayılım
2. Delta-Bar-Delta
3. Extended DBD
4. Hızlı Yayılım
5. Momentumlu Geri-yayılım
6. Levenberg-Marquardt metot
7. Genetik algoritma

4.16.1 Geri Yayılım Algoritması

- En çok kullanılan algoritmalarından biridir.
- Dereceli azalır.
- MLP'lerine eğitiminde en sık kullanılan algoritmadır.
- i ve j katman arasında tanımlanan ağırlıklardaki $w_{ji}(t)$ değişikliği hesaplanır.

$$\Delta w_{ij}(t) = \eta \delta_j x_i + \alpha \Delta w_{ij}(t-1) \quad (16)$$



Şekil 32 Çok katmanlı bir perseptron geri yayılım akış şeması

4.16.2 Delta-Bar-Delta

Delta-Bar-Delta (DBD) çok katmanlı perseptronlarda bağlantı ağırlıklarının yakınsama hızını arttırmak için kullanılan bir sezgisel yaklaşımdır. Deneysel çalışmalar, ağırlık uzayının her boyutunun tüm hata yüzeyi açısından tamamen farklı olabileceğini göstermiştir. Hata yüzeyindeki değişimleri açıklamak için, özellikle, ağırlık her bağlantısı kendi öğrenme katsayısına sahip olmalıdır. Bu düşünce, tek ağırlık boyutu için uygun adım büyüklüğü, tüm ağırlık boyutları için uygun olmayabilir. Bununla birlikte, her bir bağlantıya bir öğrenme oranı atanarak ve bu öğrenme oranının zamanla değişmesine izin verirse, yakınsama zamanını azaltmak için daha çok serbestlik derecesi sağlanmış olur.

İleri beslemeli YSA yapıları çoğu zaman karmaşıktırlar. Ağdaki her bağlantı için en uygun öğrenme katsayıları kümesini belirlemek oldukça zaman alıcı olabilir. Eğimin geçmişteki değerlerini kullanarak, yerel hata yüzeyinin eğriliğini çıkarmak için sezgisellik uygulanabilir. Bir bağlantı için ağırlık değişimlerinin işareti, birkaç ardışık zaman adımları sırayla değiştiği zaman, bağlantı için öğrenme oranı azalmalıdır. Bağlantı ağırlık boyutu hata yüzeyi ile ilgili büyük bir eğriliğe sahiptir. Bağlantı ağırlık değişimleri bir kaç ardışık zaman adımları için aynı işarete sahip olduğundan bağlantı için öğrenme oranı artırılmalıdır. Ağırlık boyutu hata yüzeyi ile ilgili küçük bir eğriliğe sahiptir ve yüzey önemli bir mesafe için aynı doğrultudaki eğime göre devam eder. Standart geri yayılım algoritmasında eğim bileşeni aşağıdaki şekilde verilir.

$$\delta(k) = \frac{\partial E(k)}{\partial w(k)} \quad (17)$$

Burada $E(k)$, k anındaki hata değerini, $w(k)$ bağlantı ağırlığını ve (k) ise ağırlık değişiminin eğim bileşenini göstermektedir. Standart geri yayılım algoritmasında bağlantı ağırlığı,

$$w(k+1) = w(k) + \alpha \delta(k) \quad (18)$$

olarak güncelleştirilir. Burada sabit bir öğrenme oranıdır. DBD öğrenme kuralında, her bağlantı için değişken öğrenme oranı (k) atanır ve bağlantı ağırlığının güncelleşmesi

$$w(k+1) = w(k) + \alpha(k) \delta(k) \quad (19)$$

şeklinde yapılır. Jacobs, her bağlantıda öğrenme hızının azalması ve artması için sezgiselliğin gerçekleştirilmesinde, (k) eğim bileşeninin ağırlıklı ortalamasını kullanılmıştır. Bu ağırlıklı ortalama $\delta(k)$ 'dır. Özellikle, ağırlıklı ortalama aşağıdaki şekilde gösterilmiştir.

$$\delta(k) = (1 - \theta)\delta(k) + \theta\delta(k-1) \quad (20)$$

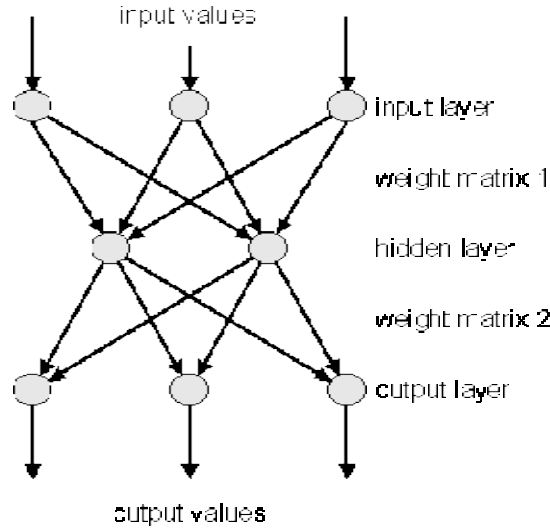
Burada, konveks ağırlık faktörüdür. Sezgisel açıdan, önceki eğim bileşeninin üstel artması ve şu anki eğim bileşeni aynı işaretli ise, öğrenme oranı sabiti ile artan ağırlıkla birleştirilir. Mevcut eğim bileşeni üstel ortalamadan farklı işaretli ise, öğrenme oranı mevcut değeri ile orantılı olarak azalır. Bu öğrenme oranının güncelleştirilmesi aşağıdaki eşitlikle tanımlanmıştır.

$$\Delta\alpha(k) = \begin{cases} k & \bar{\delta}(k-1)\delta(k) > 0 \\ -\varphi\alpha(k) & \bar{\delta}(k-1)\delta(k) < 0 \\ 0 & \text{diğer} \end{cases} \quad (21)$$

Burada, öğrenme katsayısı artma faktörü, öğrenme katsayısı azaltma faktörü, (k) k anındaki öğrenme oranını göstermektedir. Eşitlik yukarıdaki şekilde görülebileceği gibi algoritma, öğrenme katsayılarını lineer olarak artırmakta, fakat geometrik olarak azaltmaktadır [133 – 137].

4.16.3 Multi-Layer Perception (MLP)

- Çoksayıda danışmalı öğrenme algoritması kullanılabilmekte
- Kullanımı yaygın
- Bir çok probleme uygulanabilmesi
- Topolojik yapısı ve matematiksel çalışma metodolojisi basittir.



Şekil 33 Multi-Layer Perception prensip şeması

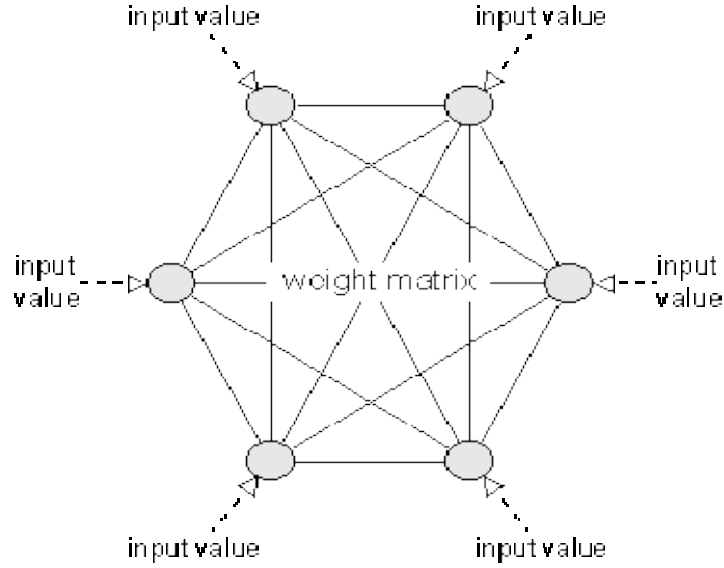
4.16.4 Learning Vector Quantisation Ağ

- LVQ ağı genel sınıflandırma problemlerinin çözümünde kullanılmaktadır.
- Kullanılan öğrenme stratejisi destekleyici (reinforcement learning) öğrenmedir. Yani ağa sadece üretilen çıktının doğru olup olmadığı bilgisi verilir.
- Çıktı değerlerinin belirlenmesinde ise kazanan herşeyi alır (winner takes all) stratejisi uygulanmaktadır.
- Sadece girdi vektörüne en yakın olan vektörün (kazanan vektör) değerleri (ağın bu vektöre ait ağırlıkları değiştirilir.)
- Kohonen Katmanına ara katman da denmektedir.
- Girdi katmanı ile ara katman = Tam bağlantılı
- Ara katman ile çıktı katmanı = Kısmi bağlantılı
- Kohonen katmanı ile çıktı katmanı arasındaki
- ağırlıklar sabit olup 1'e eşittir.
- G-A katman arasındaki her bir ağırlık vektörüne referans vektörü denmektedir.

- Diğer LVQ ağları LVQ2 ,Cezalandırılmalı LVQ ve LVQ-X'tir.

4.16.5 Hopfield Ağ

- Tek katmanlı ,geri dönüşümlü bir ağdır.
- Proses elemanlarının tamamı hem girdi hemde çıktı elemanıdır.
- Ağın bağlantıları enerji fonksiyonudur.
- Satıcı gezgin problemini çözer.



Şekil 34 Hopfield ağı

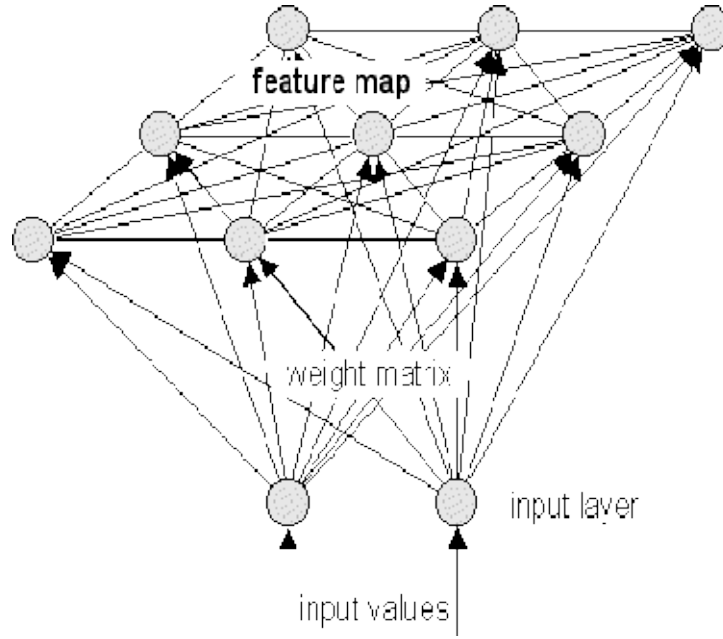
4.16.6 Elman Ağ

- Bilinen en basit geri dönüşümlü ağıdır.
- Girdi elemanları içerik elemanları ve girdi elemanlarından oluşur.
- Elman ağı kısmi geri dönüşümlü bir ağıdır.
- İçerik elemanları aktivasyon değerini tutarlar

4.16.7 Kohonem Ağ

- Danışmansız ağ

- Giriş ve çıkış katmanı
- Etiketleme işlemi
- Sınıflandırma problemleri



Şekil 35 Kohohem ağ yapısı

5. BÖLÜM : Genetik Algoritma (Genetic Algorithm)

Genetik algoritma (GA) süreci doğal evrime benzetilir. Bu nedenle *Üreme* (Reproduction), *Çaprazlama* (Crossover), *Mutasyon* (Mutation) gibi doğal evrimde kullanılan operatörleri içerir. Üreme, uygunluk (fitness) değerlerine bakılarak stokastik yöntemlerle seçilen bireylerden yeni bir popülasyon oluşturma işlemidir. Bu işlem, ilerleyen generasyonlarda daha yüksek uygunluk değerlerine sahip bireylerin oluşmasına neden olur. Bu nedenle bu işleme en uygunun hayatta kaldığı test (survival of the fittest) adı verilir.

Çaprazlama, çoğunlukla rastgele olarak seçilen iki bireyin kromozomları çaprazlanarak gerçekleşir. Bu işlemde, bireylerin kromozomunu oluşturan dizilerin değişik kısımlar yer değiştirilerek yeni döl üretimi sağlanır. Bu döl popülasyonunda daha az uygunluk değerine sahip “*zayıf*” bireylerin yerine konabilir. Çaprazlama, genetik algorithmada en önemli operatördür ve generasyonda yeni çözümlerinin üretiminden sorumludur. Mutasyon, bireyin kromozomunu oluşturan dizideki tek bir elemanın değerinin rastgele olarak değişmesidir. Mutasyon, çözümün alt optimal noktalara takılmasını önleyen ve çok düşük olasılık değeri ile uygulanan operatördür.

Genel olarak genetik algoritma, çözüm bilgisinin hiç olmadığı veya çok az olduğu bir durumla aramaya başlar. Çözüm çevreden gelen etkileşime ve genetik operatörlere bağlıdır. GA, aramaya paralel bir şekilde, birbirinden bağımsız noktalardan başlar, bu nedenle alt optimal çözümlere takılma olasılığı azdır. Bu nedenle GA, karmaşık arama problemleri “birden çok alt çözüm kümesi olan” için en iyi optimizasyon tekniği olarak bilinir. GA’yı diğer Evrim Algoritması (EA) türlerinden farklı kılan özellikleri;

- Eşeyli üreme yöntemini,
- Mutasyon ve çaprazlama operatörlerini,
- Stokastik veya deterministik seçim yöntemlerini,
- Problemin çözümü için problemin kendisi yerine kodlanmış bir dizisini kullanması olarak gösterilebilir.

Bu şekilde GA diğer EA türlerinden daha esnek bir yapı sağlar. Bir evrim işlemi, potansiyel çözüm uzayında, popülasyonu oluşturan kromozomlar içinde en uygun kromozomu arama işlemidir. Böyle bir arama iki zıt amacı dengelemeyi gerektirir. Bu amaçlar; en iyi çözümlerin aranması (Exploit) ve arama uzayının genişletilmesidir (Explore) Genetik algoritma ile geleneksel optimizasyon teknikleri (özellikle nümerik metotlar) arasında da çok önemli farklılık vardır. Bu farklılıklar aşağıdaki gibi özetlenebilir:

- i. GA optimize edilecek olan parametrelerin kendileri ile değil kodlanmış dizileri üzerinde çalışır. Pek çok durumda ikili (binary) kodlama kullanılır. Fakat genetik algoritmalar için bu bir gereklilik değildir. Gerçek sayı kodlama, ağaç yapılı kodlama (tree coding) gibi farklı kodlama sistemleri de kullanılabilir.
- ii. GA, bir popülasyon içinde arama yapar. Bu popülasyon, problemin bütün olası çözümlerini temsil eden uzayı oluşturur. Başlangıç popülasyonu genellikle rastgele üretilen bireyleri içerir.
- iii. GA, problemin çözümünü belirlenen çözüm uzayında aramak için bir uygunluk fonksiyonu (fitness function) kullanır. Bu uygunluk fonksiyonu klasik optimizasyon tekniklerinde kullanılan amaç fonksiyonuna benzetilebilir.
- iv. GA, sonuca ulaşmak için stokastik yöntemler kullanır.

Genetik algoritmaları cazip kılan özelliklerinin bazıları şunlardır:

Öğrenme: Genetik algoritma global arama tekniklerinde yaygınca kullanılır ve en iyisi olarak bilinir. Mevcut performans ölçütlerini kullanarak verilen bir arama uzayında arama uzayını genişletme ve en iyiyi arama özelliklerini kullanma yeteneğine sahiptir. Bu özelliklerini çaprazlama, mutasyon ve üretim gibi genetik operatörlerle kullanarak öğrenme yeteneğine sahiptir.

Genetik kod yapısı: Genetik algoritma doğrudan parametrelerle değil, kodlanmış parametre dizisiyle çalışır. Bu, kullanıcıya problemleri bir değişken optimizasyon problemi gibi çözmesine imkan verir.

Çözümlerin optimalliği: Pek çok gerçek hayat problemlerinin *Çok Modellilik* (Multimodal) ve doğrusal olmama gibi özellikleri vardır. Geleneksel arama teknikleri böyle arama uzaylarında yetersiz kalır. Genetik algoritma ise böyle karmaşık arama uzaylarında optimale yakın çözümler bulma yeteneğine sahiptir. Genetik algoritma, mühendislik, bilim, ekonomi çok değişik alanlardaki problemler için gürbüz (Robust) bir optimizasyon aracı olarak son yıllarda büyük bir önem kazanmıştır.

Genetik algoritmanın uygulama alanlarından bazıları; haberleşme şebekleri tasarımı, elektronik devre dizaynı, gaz boruları şebekeleri optimizasyonu, görüntü ve ses tanıma, veri tabanı sorgulama optimizasyonu, uçak tasarımı, fiziksel sistemlerin kontrolü, gezgin satıcı problemlerinin çözümü, ulaşım problemleri, optimal kontrol problemleridir .

Esnek Programlama (Soft Computing)

Uzman sistemler, bulanık mantık, yapay sinir ağları ve genetik algoritma uygulamalarda tek başlarına kullanılabildikleri gibi birçok uygulamada her bir yöntemin avantaj ve dezavantajları göz önüne alınarak birlikte kullanılır. İşte böyle bir yaklaşıma esnek programlama adı verilir. Bu şekilde çok daha etkin yöntemler geliştirilmiştir.

5.1. Genetic Algoritmalar ve Evrimsel Hesaplama

Evrimsel hesaplama bir optimizasyon işlemidir. Öyle ki, amaç, hayatta kalacak bireylerin yeteneklerini geliştirmektir. Evrimsel hesaplama (EC), bir arama işleminde doğal seçimin benzetimidir. Doğada, organizmalar hayatta kalmak ve çoğalmak için yeteneklerini etkileyen belli karakteristiklere sahiptir. Bu karakteristikler organizmanın kromozomlarında içerilen bilginin uzun stringleri şeklinde temsil edilirler. Eşlenerek çoğaltmadan sonra, yavru kromozomlar her çiftten gelen bilginin bir kombinasyonundan oluşur. Umut verici bir biçimde sonuç, her çiftin en iyi karakteristiklerini içeren yavru kromozomlar olacaktır. Doğal seçim işlemi, uygunluk değeri en yüksek bireyin seçilmesini sağlar.

Evrimsel doğal seçim aracılığıyla, popülasyon içinden rasgele bir şekilde seçilmiş bireyler, uygun kromozom değerleri arama olarak düşünülebilir. Bu amaçla, bir evrimsel algoritması (EA) verilen bir probleme en uygun çözüm için olasılıklı bir aramadır.

Evrım algoritmaları; genetik algoritmalar (GA), genetik programlama (GP), evrimsel programlama (EP), evrimsel strateji (EV) ve benzerlerini içeren alt bölümlere ayrılabilir.

Genetik algoritma (GA)'nın terminolojisinin anlaşılması için “doğal seçim” in (seleksiyonun) anlaşılması gerekir. Dünyayı gözlemleyecek olursak, olup biten olaylarda doğal seçim göze çarpar. Birbirinden ayrı muazzam organizmalar ve bu organizmalardaki karmaşıklık, inceleme ve araştırma konusudur. Organizmaların niçin böyle olduğu ve nasıl bu aşamaya geldiği sorgulanabilir. Bir başka açıdan meseleye şöyle bakılabilir. Sanki bu günün dünyasında, çok büyük optimizasyon algoritmaları oluşturulmuş ve binlerce iterasyon sonunda yer yüzünde optimum çözüm elde edilmiştir. Amaç fonksiyonu, yaşam mücadelesini temsil eder. İnsanoğlu bunu maksimize etmek ister [139]. Adaptasyon ve uygunluğun seviyesi, dünyada uzun süre yaşayabilmenin göstergesi haline gelmiştir. Evrim süreci, hayat şartlarına en uygun olanın yaşamasını sağlayan büyük bir algoritmadır. Eğer çevreyi değiştirme zeka ve yeteneğine sahip olunursa hayatta, global maksimum elde edilebilir [139].

Gen, temel kalıtım birimidir. Organizmanın genleri; DNA (Deoxyribo Nucleic Acid) formunda bir çift kromozomdur. Organizmanın her bir hücresi benzer miktarda kromozomlar içerirler. Her bir vücut hücresindeki kromozom miktarı farklıdır. Örneğin; sivrisinek de 6, kurbağada 26, insanda 46 ve alabalıkta 94 adet kromozom vardır. Genler, her biri farklı karakteristiği temsil eden iki fonksiyonel formda meydana gelir. Bu formların her biri “allel” olarak bilinir [140]. Örneğin allel değerine göre insanın gözünün biri mavi olurken diğeri kahverengi olmaktadır. Kromozomlardaki allel’lerin kombinasyonları, fertlerin kişisel özelliklerini belirler. Fert üzerinde allel’lerden biri baskın olurken diğeri pasif olmaktadır.

Genetik algoritmalar, biyolojik süreci modelleyerek fonksiyonları optimize eden evrim algoritmalarıdır. GA parametreleri, biyolojideki genleri temsil ederken, parametrelerin toplu kümesi de kromozomu oluşturmaktadır. GA’ların her bir ferdi kromozomlar (bireyler) şeklinde temsil edilen popülasyonlardan oluşur. Popülasyonun uygunluğu, belirli kurallar dâhilinde maksimize veya minimize edilir. Her yeni nesil,

rasgele bilgi deęiřimi ile oluřturulan diziler içinde hayatta kalanların birleřtirilmesi ile elde edilmektedir [141].

Bu metot, uzun alıřmaların neticesinde ilk defa John Holland (1975) tarafından uygulanmaya bařlandı. En son onun ğrencisi olan David Goldberg popler oldu. David Goldberg tezinde; gaz boru hattının kontroln ieren bir problemin zmn genetik algoritma ile gerekleřtirdi [142].

GA'ların avantajları;

- Srekli ve ayrıık parametreleri optimize etmesi
- Trevsel bilgiler gerektirmemesi
- Ama fonksiyonunu geniř bir spektrumda arařtırması
- ok sayıda parametrelerle alıřma imknı olması
- Paralel PC 'ler kullanılarak alıřtırılabilmesi

Karmařık ama fonksiyonu parametrelerini, lokal minimum veya maksimumlara takılmadan optimize edebilmesi

Sadece tek zm deęil, birden fazla parametrelerin optimum zmlerini elde edebilmesi olarak sıralanabilir.

GA'lar arama ve optimizasyon iin sezgisel yntemlerdir. Geniř arama algoritmalarının aksine, genetik algoritmalar en iyiye semek iin tm farklı durumları retmez. Bundan dolayı, mkemmel zme ulařamayabilir. Fakat zaman kısıtlamalarını hesaba katan en yakın zmlerden biridir. GA lar řartlara uyum saęlayabilir. Bunun anlamı, nceden hi bilgisi olmamasına karřın, olayları ve bilgiyi ęrenme ve toplama yeteneęine sahip olmasıdır.

Her problemin zm iin GA kullanmak iyi bir yol deęildir. Birka parametrelili analitik fonksiyonun zmnde klasik metotlar daha hızlıdır. Byle durumlarda, nmerik metotlar tercih edilmelidir. Paralel bilgisayarlar kullanılırsa GA daha hızlı sonu verebilir.

Gelecek bölümde GA'larla benzerlik gösteren Saf Rasgele Araştırma Algoritması (pure random search) kısaca anlatılacaktır. GA'lara özel operatörler tanıtılacak ve birkaç örnekle çalışması gösterilecek. Ayrıca, ikili kodlarla çalışan (ikili kodlu) GA ve gerçek kodlarla çalışan (gerçek kodlu) GA'ların çalışma ve işleyişi anlatılacaktır.

5.2. Rasgele Arama Algoritması

Rasgele Arama, belki de en basit arama işlemidir. Bir başlangıç arama noktasından veya başlangıç noktalarının kümesinden başlayan arama işlemi, arama uzayında rasgele noktaları araştırır ve bu kabul edilebilir bir çözüme ulaşıncaya veya maksimum iterasyon sayısı ulaşıncaya kadar devam eder. Rasgele aramayı gerçekleştirmek son derece basitken, verimsiz olabilir. Eğitim zamanı kabul edilebilir çözüm elde edilmeden önce çok uzun olabilir.

Rasgele araştırma için bir algoritma çalışması ve işleyişi aşağıda sunulmuştur.

Adım 1. N başlangıç arama noktaları kümesini seç. $C_g = \{ C_{g,n} = | n=1,2,...,N \}$. Burada, $C_{g,n}$ I değişkenlerinin vektörü ve $g=0$ dır. Her bir $C_{g,n}$ elemanı, $U(\min, \max)$ değişken değerlerin sınırı olmak üzere, verilen aralıklarda üretilir.

Adım 2. Her bir $C_{g,n}$ vektörünün (“uygunluk”) $F(C_{g,n})$ doğruluğunu değerlendir.

Adım 3. En iyi noktayı bul $C_{g, best} = \min\{ F(C_{g,n}) \}$

Adım 4. if $C_{g, best} < C_{best}$ then $C_{best} = C_{g, best}$ tümünün en iyi çözümü

Adım 5. if C_{best} kabul edilebilir bir çözüm ise veya maksimum iterasyon sayısı aşılmış ise o zaman “dur” ve çözüm olarak C_{best} “dön”

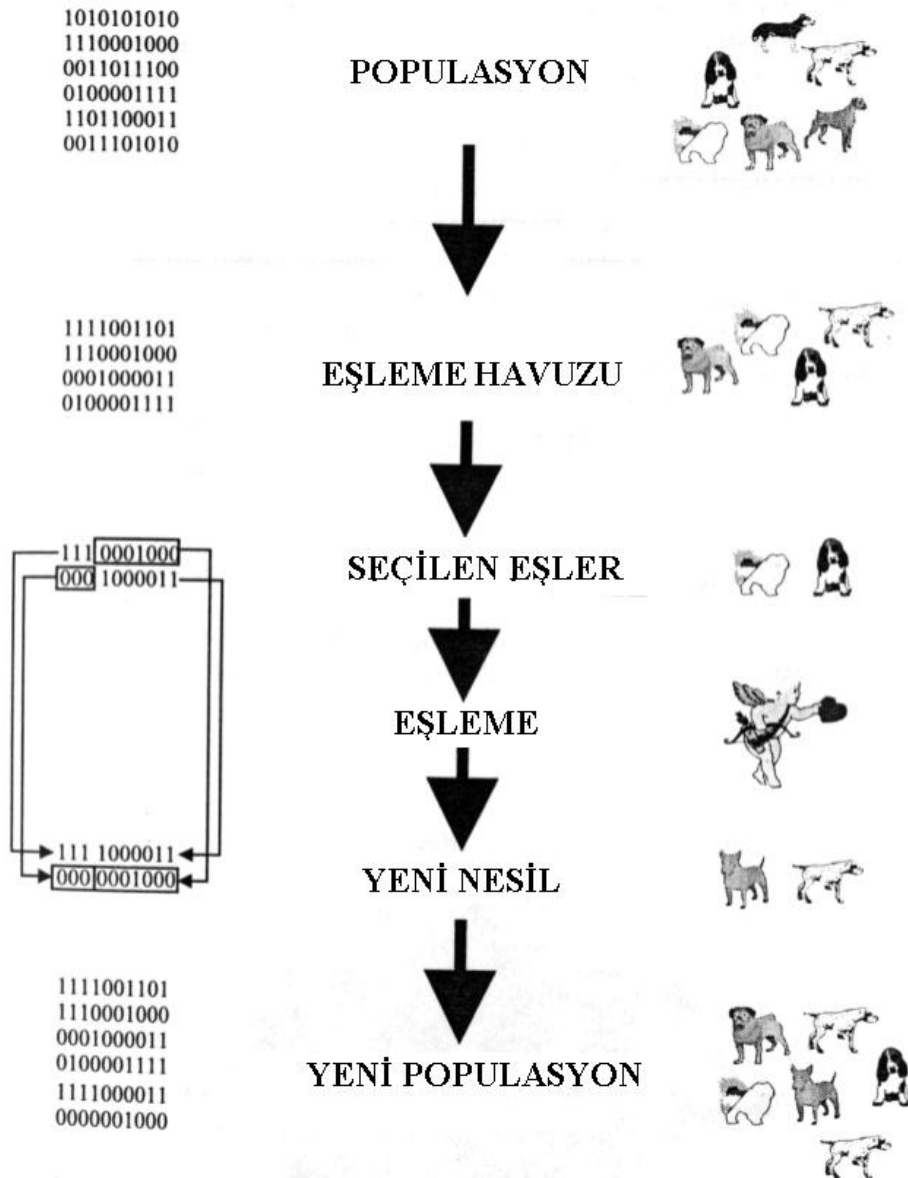
Adım 6. Her bir $C_{g,n}$, $\Delta C_{g,n}$ ile karıştır. Burada, $\Delta C_{g,n} \approx N(0, \sigma^2)$ ve σ^2 li küçük bir değişimi ifade eder.

Adım 7. $g = g + 1$ artır ve adım 2 ye git.

5.3. İkili kodlu genetik algoritmalar

Genetik Algoritmalar (GAs), verilen bir veri kümesi için en iyi çözümü (parametreler kümesini) bulacak, başka bir doğrusal olmayan optimizasyon aracıdır. GA

algoritma, bir mümkün çözümler kümesinin rasgele üretilmesiyle başlar. Parametreleriyle her bir çözüm, arama uzayında (kromozom veya nesil uzayı), uygunluk fonksiyonunun özel bir noktasını üretir. Her iterasyondaki bu farklı nesiller kümesi popülasyon olarak adlandırılır. Netice olarak, bir popülasyonun en iyi çözümünün bir parçasından (yarısı veya dörtte biri denebilir), çocuklar (yeni nesil) üretilir. Bu yeni nesil eskilerden daha iyi olması beklenir.

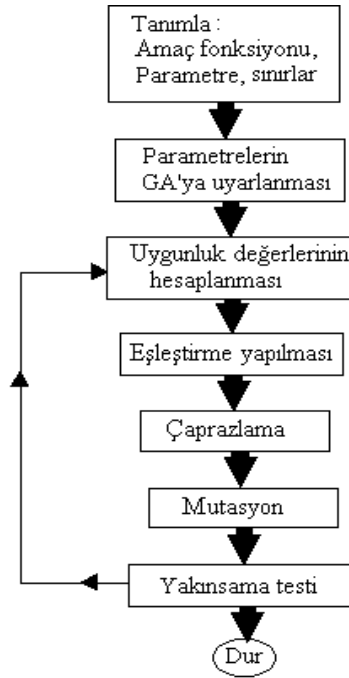


Şekil 36 İkili kodlu GA ile biyolojik evrim arasındaki benzetim [139]

Biyolojik evrim ile ikili kodlarla çalışan GA arasındaki benzerlik Şekil 36’da görülmektedir. Her ikisinde de popülasyonun üyeleri rasgele başlar. Köpeklerin her birinin karakteristik özellikleri, sol taraftaki satırlarda verilmiştir. En iyi havlayan köpekler üretilmek istenirse sadece birkaç köpeğin elde tutulması gerekir. Her bir köpek ikili sayı sistemiyle kodlanır. Yeni yavruların üretilmesi için bu popülasyonda rasgele iki adet köpek seçilir ve eşleştirilir. Eşleştirmenin sonucunda iyi havlayan köpeğin çıkma olasılığı yüksektir. Oluşan yeni nesil, eşleştirme havuzuna tekrar atılır. Başa dönülerek yeniden eşleştirme prosedürü tekrarlanır [143]. Bu işleme, en iyi havlayan köpek elde edilene kadar devam edilir.

5.3.1 İkili kodlu genetik algoritmaların operatörleri

Diğer optimizasyon metotlarında olduğu gibi ikili kodlu GA’da da amaç fonksiyonu, parametreler ve sınırlar tanımlanır. Aynı şekilde yakınsama kontrol edilerek algoritma son bulur. İkili kodlu GA’nın akış diyagramı Şekil 37’de verilmektedir.



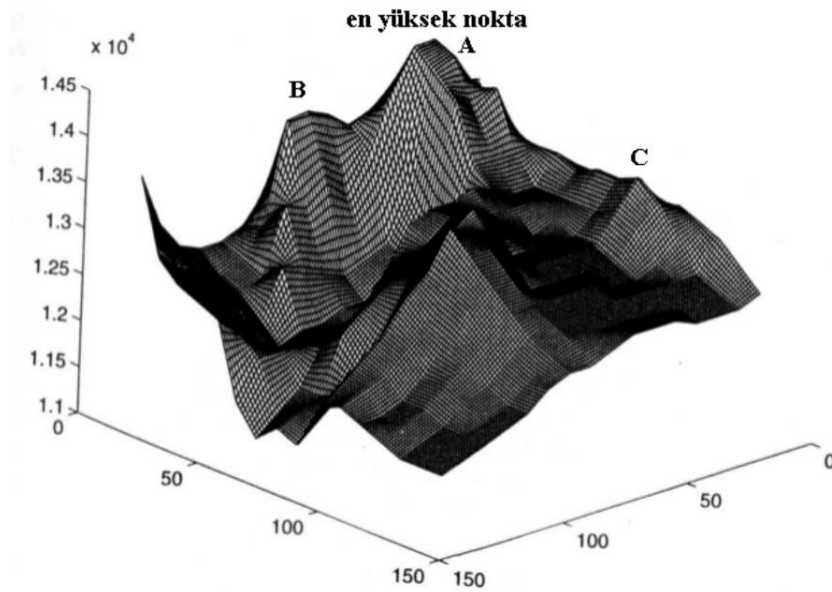
Şekil 37 İkili kodlu GA’nın akış diyagramı

Genel bir GA aşağıdaki program kod ile özetlenebilir.

1. $y = 0$ ata.

2. C_g ilk neslini oluřtur.
3. While(yakınsama olmazken)
 - a. Her bireyin uygunluęunu deęerlendir.
 - b. $g = g + 1$
 - c. C_{g-1} den ebeveynleri seę.
 - d. Yavru O_g yi řekillendirmek ięin ęaprazlama seęilen ebeveynleri yeniden birleřtir.
 - e. O_g deki yavruyu mutasyona uęrat.
 - f. nceki nesil C_{g-1} ve yavru O_g den yeni nesli seę.

GA ile topoęrafik haritadaki ykseltiler veya vadiler arařtırılabilir. Optimizasyon algoritması vadiyi bulmak ięin, amaę fonksiyonunun minimum deęerini ve en yksek blgeyi bulmak ięin amaę fonksiyonunun maksimum deęerini arar. řekil 38’de en yksek tepe, deniz seviyesinden 14255 m ykseklіğinde olan A noktasıdır. B noktası 13281 m ve C noktası 11800 m ykseklіindedir. A noktasına ulařılıncaya kadar arada noktalar vardır. Bu yzden klasik metotlarda en yksek nokta olan A’yı bulmak zordur. GA ise bu noktayı rahatlıkla bulabilir.



řekil 38: Daęın  boyutlu grnm [139]

5.4. Amaç fonksiyonu ve parametreler

Amaç fonksiyonu, giriş parametrelerine göre çıkış üreten bir fonksiyondur. Bu fonksiyon matematiksel veya deneysel olabilir. Burada amaç, giriş parametreleri için uygun değer bulununcaya kadar çıkış üretmektir. Örneğin banyo küveti doldurulurken farkında olmadan optimizasyon yapılır. Giriş parametreleri, sıcak ve soğuk su çeşmeleridir. Amaç fonksiyonu deneysel sonuçlardır. Çünkü el, suyun içine sokularak suyun sıcaklığı ayarlanır [139].

GA'ya, optimize edilecek parametre dizileri ve kromozomlar (bireyler) tanımlanarak başlanır. Kromozom, N_{par} adet parametreye sahip ise optimizasyon problemi N_{par} boyutlu olur. Parametreler $P_1, P_2, P_3, \dots, P_{N_{par}}$ ise;

Kromozom = $[P_1, P_2, P_3, \dots, P_{N_{par}}]$ olarak tanımlanır.

Örneğin; topoğrafik bir haritayı ele alalım. Amaç fonksiyonu giriş parametreleri, enlem (x) ve boylam (y) kabul edilirse, kromozom = $[x, y]$ şeklinde tanımlanır. Burada $N_{par}=2$ dir. Her bir kromozom; P_1, P_2 parametrelerinin değerlerine göre hesaplanır.

$$F(\text{kromozom}) = F(P_1, P_2, \dots, P_{N_{par}})$$

Çoğu zaman amaç fonksiyonu oldukça karmaşıktır. Parametrelerin hangisinin önemli olduğuna karar verilmelidir. Parametreler çok fazla olursa GA, iyi sonuç vermeyecektir. Örneğin; arabalarda yakıt tasarrufu için önemli parametreler araba boyutları, motor boyutları ve malzeme ağırlıkları şeklinde belirlenebilir. Boya rengi veya koltuk tipi gibi diğer parametreler ya çok az etkili ya da hiç etkili değildir. Bazen parametrelerin seçimi ve değer tespiti, tecrübeyle veya deneme-yanılma yoluyla yapılır. Örneğin; $F(w, x, y, z) = 2x + 3y + z/100000 + \sqrt{w}/9876$ şeklinde analitik bir fonksiyonun parametrelerinin, 0–10 arasında değiştiğini kabul edelim. Amaç fonksiyonu içerisinde w ve z parametreleri yüksek değerlere bölündüğünden 0–10 arasında değişmesi, sonucu pek fazla etkilemeyecektir. Eğer w ve z parametreleri, ihmal edilirse, 4 boyutlu problem 2 boyutlu olarak modellenir.

Optimizasyon problemlerinin çoğunda parametrelerin belirli sınırlar içerisinde kalması istenir. Örneğin; otomobilin ağırlığı sıfır olmayacak, boyu 10 m'yi geçmeyecek vb. sınırlamalar getirilebilir. Sınırsız parametreler ise herhangi bir değer alabilir. İki türlü

sınır belirlenebilir. Birincisi; $0 \leq x \leq 10$ arasında değişen bir parametrede, eğer $x=11$ değerini alırsa $x=10$ yapılır ve $x= -1$ değerini alırsa $x=0$ yapılır. İkincisi; $x = 5\sin y + 5$ olarak tanımlanırsa, y 'nin herhangi bir değerinde $0 \leq x \leq 10$ arası değişecektir. Böyle bir dönüştürmeyle sınırlı optimizasyon problemi sınırsız optimizasyon problemine çevrilmiş olur [144].

Bağımlı parametreler, optimizasyon algoritmaları için özel problemler meydana getirir. Çünkü bir parametrenin değişimi diğer parametreleri de etkileyecektir. Örneğin; arabanın boyunu artırmak, ağırlığını da artıracaktır. Fourier serisi katsayılarında olduğu gibi bağımsız parametreler birbirlerini etkilemezler. GA literatüründe parametre etkileşimi, “epistasis” olarak adlandırılır. Biyolojide epistasis, genlerin birbirini etkilemesi olarak tanımlanır. Epistasis çok az olursa, minimum araştırma algoritmaları iyi sonuçlar üretir. GA’da Epistasis orta ve yüksek derecede olursa iyi sonuç vermektedir, çok yüksek veya çok düşük olursa iyi sonuç vermemektedir [139].

5.4.1 Parametrelerin tanımlanması ve kodlama

Şekil 37’den, GA nın üç işlem için döngüye girdiğini görebiliriz.

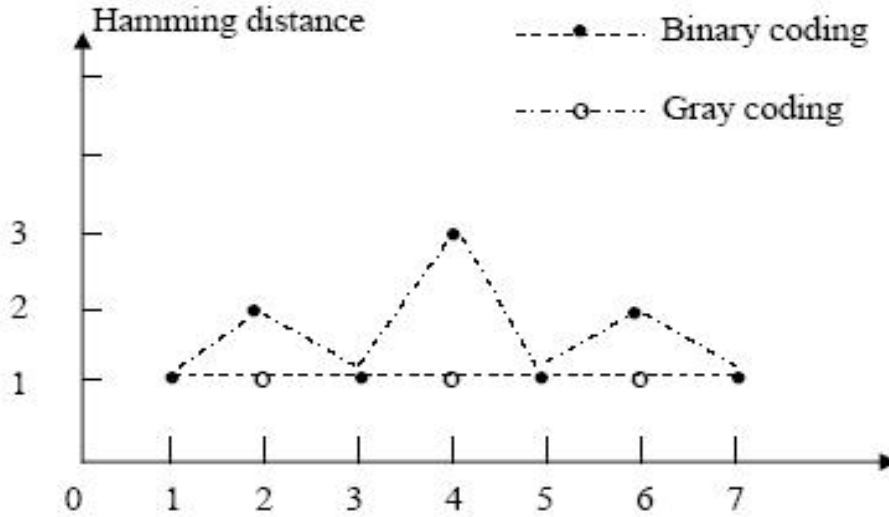
- En iyi gen stringinin seçimi
- Genetik operasyon (çaprazlama, mutasyon)
- Yeni gen stringleri (çocuklar) ile eski popülasyonun kötü gen stringlerinin yer değiştirilmesi.

Optimizasyon döngüsü başlamadan önce, optimize edilmesi gereken parametreler istenilen şekle dönüştürülmek zorundadır. Buna kodlama (encoding) denir. Kodlama GA için önemli bir konudur. Çünkü sistemden gözlemlenen bilgiye bakış açısı büyük ölçüde sınırlandırılabilir. Gen stringi probleme özel bilgiyi depolar. Gen olarak adlandırılan her bir öge, genellikle değişkenler stringi olarak ifade edilir. Değişkenler ikili veya reel sayı şeklinde gösterilebilir ve aralığı probleme özel olarak tanımlanır.

1	1	0	1	0	0	1	0	→ İkili kodlama
43	4	9	16	30	22	59	12	→ Reel sayı kodlaması

Şekil 39 Gen stringlerinde parametrelerin kodlanması

İkili kodlama sıklıkla kullanılırken, Şekil 40’da gösterildiği gibi Hammimg cliffs dezavantajına sahiptir. Bir Hamming cliff, iki bitişik sayısal değer ayrı bit gösterimlerine sahip olduğunda şekillenir. Mesela onluk sayılar 7 ve 8 i düşünün. İkili gösterimlerle ilgili olarak (4 bit gösterimini kullanan) 7=0111 ve 8=1000 4 hamming mesafesidir. (Hamming mesafesi birbirine benzemeyen ilgili bitlerin sayısıdır.)



Şekil 40 ikili ve gray kodlama için Hamming mesafesi

Bu, değişkenlerdeki küçük bir değişimin uygunluktaki küçük bir değişiklikte sonuç vermesi gerektiği zaman, bir problemi ortaya koyar. Eğer örneğin, uygun çözüm 7 olsun. Fakat yeni en iyi çözüm 8 olduğunda; 8 i elde etmek için birçok bitin değiştirilmeye ihtiyaç duyulduğu aşikârdır. Hâlbuki uygunluk değerinde küçük bir değişim söz konusudur.

Alternatif bir bit gösterimi Gray Kodlama kullanmaktır. Ardışık sayısal değerlerin gösterimleri arasındaki Hamming Mesafesi bunlardan biridir. (Tablo 8’de görülen) Tablo 1 ikili ve Gray kodlamayı 3-bit için karşılaştırır.

Tablo 8 İkili ve Gray kodlama

	Binary	Gray
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

İkili sayılar dönüşüm kullanılarak kolaylıkla Gray kodlamaya çevrilebilirler.

$$\begin{aligned} g_1 &= b_1 \\ g_k &= b_{k-1} \bar{b}_k + b_{k-1} \bar{b}_k \end{aligned} \quad (22)$$

b_k da k ikili sayıdaki bitin sırasıdır. b_1 en anlamlı biti(yani sol en baştaki biti), $\bar{b}_k$; b_k nin değilini, + lojik OR lamayı ve çarpımlar da lojik AND'i gösterir.

GA'lar ayrıca integer veya reel değerleri temsil eder ve bir kromozomda değişkenlerin sırasının önemli bir rol oynadığı yerdeki komut tabanlı temsiller kullanılarak geliştirilmiştir.

Gaz için klasik temsil şeması, sabit uzunluklu ikili vektörlerdir. Bir I-boyutlu arama uzayının durumunda, her bir birey bir bit stringi olarak kodlanan her bir değişkenle I değişkenlerinden oluşur. Eğer değişkenlerin ikili değeri varsa, her bir kromozomun uzunluğu I bittir. Nominal değerli değişkenlerin durumunda, her bir nominal değer D boyutlu bit vektörleri olarak kodlanır. Bu değişken için 2D, her bir D-bit stringi farklı bir nominal değerle temsil edilir. Ayrık nominal değerlerinin toplam sayısıdır. Sürekli değerli değişkenlerin durumunda, her bir değişken D-boyutlu bir bit vektöre haritalanmalıdır.

$$\phi = R \rightarrow \{0,1\}^D \quad (22)$$

Sürekli uzay aralığı, sonlu bir $[\alpha, \beta]$ aralığına sınırlandırılması gerekir. Standart ikili kod çözme yöntemi kullanarak her bir kromozom C_n in sürekli değişkeni $C_{n,i}$ sabit uzunluklu bir bit stringi kullanılarak kodlanır. Örneğin, eğer $z \in [z_{\min}, z_{\max}]$ bir 30-bit gösterimine dönüştürülmesi gerekirse, aşağıdaki formül kullanılabilir.

$$(2^{30} - 1) \frac{z - z_{\min}}{z_{\max} - z_{\min}} \quad (23)$$

Amaç fonksiyonunun uygunluk değerinin hesaplanması için her iterasyonda kromozomun 10'luk sayı sistemine çevrilmesi gerekir. On bitlik genlerden oluşan bir kromozom örneği aşağıda verilmiştir.

$$\text{Kromozom} = [\underbrace{1111001001}_{\text{gen1}} \underbrace{0011011111}_{\text{gen2}} \dots \dots \dots \underbrace{0000101010}_{\text{genNpar}}]$$

Bu kromozomun bit sayısı, toplam $N_{\text{bit}} = N_{\text{gen}} \times N_{\text{par}}$ bit olarak bulunur. Şekil 5'de verilen topoğrafik harita, 128x128 değerlendirme noktasından oluşmaktadır. Eğer x ve y , iki gen olarak kodlanırsa her bir gen, "7" bittir ve x, y parametreleri, 2^7 'ye kadar değer alabilir. Bu değer aralıkları $(40^0.15') \leq y \leq (40^0.16')$ ve $(105^0.37'.30'') \geq x \geq (105^0.36')$ 'dir. Böylece kromozom aşağıdaki gibi ifade edilir (Haupt 1998).

$$\text{Kromozom} = [\underbrace{1100011}_x \underbrace{0011001}_y]$$

Genler, aşağıdaki denklem kullanılarak onluk sayı sistemine çevrilir. Buradaki ikili sayı, her bir gen için üretilen ikili sayıdır.

$$X_i = a_i + \text{decimal}(1001 \dots 001_2) (b_i - a_i) / (2^{m_i} - 1) \quad (24)$$

Burada;

a_i = Parametrelerin alt sınır değeri

b_i = Parametrelerin üst sınır değeri

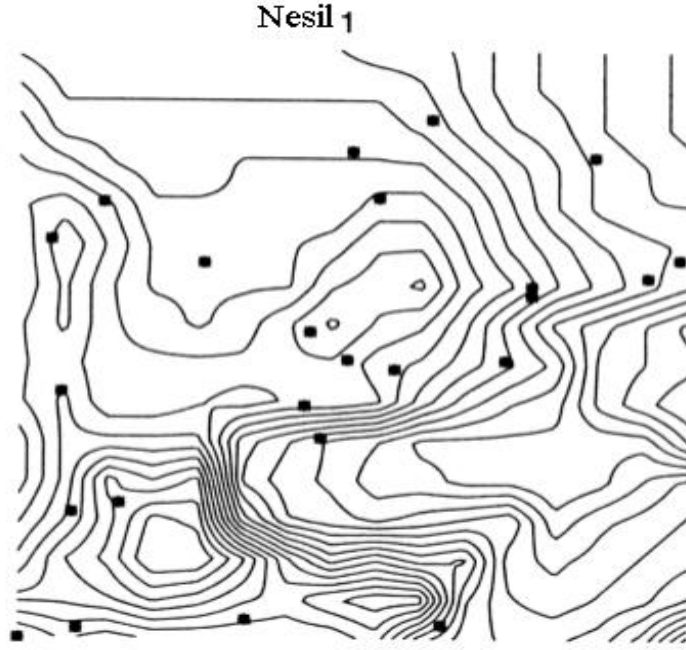
m_i = Gen'in bit sayısı'dır

5.4.2 Başlangıç Popülasyonu

Kromozom, başlangıçta verilen popülasyon miktarına göre rasgele üretilir. Bu durumda $N_{ipop} \times N_{bit}$ adet bitlerle matris oluşturulur. N_{ipop} , başlangıç popülasyonunun toplam sayısıdır, N_{bit} 'de kromozomun bit sayısıdır. Matrisin her bir satırı, bir kromozomu temsil eder. Kromozomlar ise, enlem ve boylamları temsil eder. Rasgele üretilen popülasyondaki parametreler, amaç fonksiyonunda yerlerine konularak ve uygunluk değerleri hesaplanarak değerlendirilir. Yirmi dört adet kromozomun uygunluk değerleri Tablo 9'da verilmiştir. Kromozomların yerleşimleri Şekil 41'de topoğrafik harita üzerinde gösterilmiştir.

Tablo 9 Başlangıç popülasyonunun sıralanışı [139]

Kromozomlar	Uygunluk değerleri
00000000000000	-13000
11111011010010	-11800
00010110000010	-13255
11000011001010	-12347
01111111101001	-12560
01000111010001	-12700
01010110000100	-13338
11101111001110	-11890
01111100111100	-12953
00100001011110	-12891
10001110111010	-12759
10111000111100	-12320
11011011101000	-11797
00100110011101	-13778
00010100011011	-13360
01110010101011	-12220
11000011001100	-12452
10011101110000	-12335
10100000000011	-12857
00001101010110	-13166
00010000110101	-13164
01101100110010	-12927
01101111000010	-13079
10001001011111	-12756



Şekil 41 Başlangıç popülasyonunun topoğrafik harita üzerinde dağılımı [139]

5.4.3 Doğal seçim

Başlangıç popülasyonu fazla ise bazı iteratif adımların uygulanması gerekir. Kromozom olarak en uygun olanı hayatta kalmalı diğerleri de yok olmalıdır. Doğal seçim, algoritmanın her bir iterasyonunda (generasyonunda) meydana gelir. Burada ne kadar kromozomun elde tutulacağına karar vermek biraz keyfidir. Kromozomlarda gelecek nesil içerisinde yer alma mücadelesi başlamaktadır. Popülasyon içerisinde bütün kromozomların seçilmesi gelecek nesile çok fazla katkısı olmamaktadır. GA’da değişik seçim metotları vardır.

- a. *Ağırlıklı seçim:* İlk olarak amaç fonksiyonunda, kromozomların uygunluk değerleri hesaplanır. Hesaplanan uygunluk değerleri en küçükten en büyüğe doğru sıraya konur. Tablo 10’da görüldüğü gibi $N_{pop} \leq N_{ipop}$ ise tutulur, geriye kalanlar atılır. Burada N_{pop} değeri N_{ipop} ’a kadar olabilir. Genelde popülasyonun %50’sinin seçilmesi ($N_{pop} = N_{ipop}/2$) uygun seçenektir. Seçilen N_{pop} ’un yarısı N_{iyi} , yarısı da $N_{kötü}$ olarak ayrılır. N_{iyi} olanlar eşleştirme havuzuna konurken $N_{kötü}$ olanlar eşleştirme havuzundan atılır. Toplam popülasyon sayısı $N_{ipop} = 24$ ’dür. GA’nın her bir iterasyonunda popülasyonun 12’si tutulur ve bu kromozomların altı tanesi

eşleştirme havuzuna atılır. Doğal seçim sonuçları Tablo 10’da gösterilmiştir. Bunlar amaç fonksiyonunda hesaplanan uygunluk değerlerine göre sıralanmış olup ilk altı tanesi gelecek nesil için kopyalanacaktır.

- b. Eşik değer seçim:** Eşik değerden küçük olan kromozomlar yaşamaya devam eder, eşik değeri aşan kromozomlar ölür. Bu tekniğin avantajı, popülasyonun sıralanmasına gerek olmamasıdır.
- c. Rulet tekerleği seçim:** Rulet tekerleği seçim metodunun uygulanması için, ilk önce kromozomların toplam uygunluk değeri hesaplanır. Kromozomlar, toplam uygunluk değerine bölünerek her bir kromozom için 0–1 arasında değişen seçim ihtimalleri bulunur. Daha sonra, kümülatif ihtimaller hesaplanır. Popülasyon sayısı kadar “rasgele” 0-1 arasında sayılar üretilir. Üretilen rasgele sayı, birinci kromozomun kümülatif seçim ihtimalinden küçük ise, birinci kromozom seçilir. Eğer değilse, ikinci kromozomun veya diğerlerinin kümülatif ihtimalleriyle karşılaştırılarak hangisinden küçükse o kromozom seçilir. Böylece Rulet seçim metodu gerçekleştirilmiş olur.
- d. Turnuva Seçimi:** Bu yöntemde bir grup k bireyi rasgele olarak seçilir. Bu k bireyleri daha sonra bir turnuvaya katılır ve en iyi uygunluklu birey seçilir. Çaprazlama için iki turnuva tutulur. Bunlardan biri her bir ebeveyni seçmek içindir. Bir kereden daha fazla üretmek için seçilebilecek bir ebeveyn mümkündür. Turnuva seçiminin avantajı, popülasyonun daha kötü bireylerinin seçilmeyecek olması ve bundan dolayı sonraki neslin genetik yapısına katılmayacak olmasıdır. Ayrıca en iyi bireyler tekrar üretim sürecinde baskın olmayacaktır.
- e. Seçkinlik(elitism):** Bu işlem, şu anki nesilden sonraki nesle aktarılan bir grup bireyin seçimini kapsar. Mutasyona uğratılmaksızın, sonraki nesle aktarılan bireylerin sayısı, nesil boşluğu olarak işaret edilir. Eğer nesil boşluğu sıfırsa; yeni nesil, yeni bireylerin girişleriyle oluşur. Pozitif nesil boşlukları için, diyelim ki k , k bireyleri sonraki nesle aktarılır.

Seçilen genler sonraki yeni nesli oluşturmak için çaprazlama ve mutasyon genetik operasyonlarından geçer.

Tablo 10 Kromozomların uygunluk değerlerine göre sıralanışı [139]

Kromozomlar		Uygunluk Değerleri
N_{ipop}	N_{iyi}	N_{pop}
	0010011001 1101	-13778
	0001010001 1011	-13360
	0101011000 0100	-13338
	0001011000 0010	-13225
	0000110101 0110	-13166
	0001000011 0101	-13164
	0110111100 0010	-13079
	0000000000 0000	-13000
	0111110011 1100	-12953
	0110110011 0010	-12927
	0010000101 1110	-12891
	1010000000 0011	-12857
	1000111011 1010	-12759
	1000100101 1111	-12756
	0100011101 0001	-12700
	0111111110 1001	-12560
	1100001100 1100	-12452
	1100001100 1010	-12347
	1001110111 0000	-12335
	1011100011 1100	-12320
	0111001010 1011	-12220
	1110111100 1110	-11890
	111101101 0010	-11800
	1101101110 1000	-11797

5.4.4 Eşleştirme

İki adet yeni nesil üretmek için N_{iyi} kromozomların bulunduğu eşleştirme havuzundan iki tane kromozom seçilir. Eşleştirme, seçilen kromozomlar arasında gerçekleştirilir. Eşleştirmenin değişik yöntemleri vardır.

- Yukarıdan aşağıya doğru eşleştirme: Listenin en üstünden başlanır ve eşleştirme için seçilen N_{iyi} kromozomların sayısına ulaşana kadar devam edilir. Böylece kromozom_{2i-1} ile kromozom_{2i} $i=1,2,3$ için eşleştirilir. Bu örnekte kromozom₁ ile kromozom₂, kromozom₃ ile kromozom₄ ve kromozom₅ ile kromozom₆ eşleştirildi. Bu metot, doğal bir yöntem olmamakla beraber yeni başlayanlar için kolay bir yoldur.
- Rasgele eşleştirme: Kromozomların eşleştirilmesinde rasgele sayı üretici kullanılır. Kromozomlar 1'den başlayarak N_{iyi} 'ye kadar sıralanır. Birinci eşleştirmeyi bulmak için iki adet rasgele sayı üretilir.

Kromozom=roundup{ N_{iyi} x rasgele sayı}

Burada roundup fonksiyonu, sayıları en yüksek sayıya yuvarlar. Örneğin; rasgele olarak üretilen 6 sayı 0.1535, 0.6781, 0.0872, 0.1936, 0.7021 ve 0.3933 ise bu sayılar, 6 ile çarpılıp bir üst tamsayıya yuvarlanarak; 1, 5, 1, 2, 5, 3 değerleri elde edilir. Buna göre kromozom₁-kromozom₅, kromozom₁-kromozom₂ ve kromozom₅-kromozom₃ eşleştirilecektir.

- c. Ağırlıklı rasgele eşleştirme: Bu yöntem, eşleştirme havuzundaki kromozomların, amaç fonksiyonunda hesaplanan uygunluk değerlerine göre seçilme ihtimallerini dikkate alır. En düşük uygunluk değerine sahip kromozom, eşleştirme ihtimali en yüksek kromozom olurken en yüksek uygunluk değerine sahip kromozom eşleştirme ihtimali en düşük olarak düşünülür. Üretilen rasgele sayı, hangi kromozomun seçileceğini belirler. Bu yöntem kendi arasında ikiye ayrılır.
- d. Sıra Ağırlık: Kromozomlar büyükten küçüğe doğru sıralanır ve denklem (25) kullanılarak kromozomların sıralamasındaki yerlerine göre P_n olasılıkları hesaplanır.

$$P_n = \frac{N_{iyi} - n + 1}{\sum_{n=1}^{N_{iyi}} n} = \frac{6 - n + 1}{1 + 2 + 3 + 4 + 5 + 6} = \frac{7 - n}{21} \quad (25)$$

Burada n, sıra sayısını göstermektedir. Bu yöntemle uygunluk değerlerinden bağımsız olarak, sadece sıralamayı dikkate alan P_n normalize değerleri elde edilmektedir. Bu metot uygulanarak alınan sonuç Tablo 11 'de gösterilmiştir.

Tablo 11 Eşleştirme olasılıkları [139]

n	Kromozom	P_n	$\sum_{i=1}^n P_i$
1	00100110011101	0.2857	0.2857
2	00010100011011	0.2381	0.5238
3	01010110000100	0.1905	0.7143
4	00010110000010	0.1429	0.8572
5	00001101010110	0.0952	0.9524
6	00010000110101	0.0476	1.0000

Kromozomların seçilmesinde kümülatif ihtimaller kullanılır. Rasgele olarak 0–1 arasında bir sayı üretilir. Bu sayı, kümülatif ihtimallerin en başından başlanarak sırayla karşılaştırılır. Kümülatif ihtimal değeri, üretilen rasgele sayıdan büyükse, eşleştirme havuzu için kromozom seçilir. Benzer tarzda 6 tane rasgele sayı üretilerek eşleştirme gerçekleştirilir. Üretilen rasgele sayılar 0.1535, 0.6781, 0.0872, 0.1936, 0.7021 ve 0.3933’dir. Bu sayılara göre kromozom₁-kromozom₃, kromozom₁-kromozom₁ ve kromozom₃-kromozom₂ eşleştirilirler. Kromozomlar kendisi ile eşleştirilirse, bu kromozomun gelecek nesilde daha etkin olarak yer aldığı söylenebilir. Bu yaklaşım tarzı doğaya daha uygundur.

Değer ağırlık: Kromozomların uygunluk değerleri amaç fonksiyonundan hesaplanır. Eşleştirme havuzunda N_{iyi+1} ’inci sırada bulunan kromozomun amaç fonksiyonunda hesaplanan uygunluk değeri, bütün eşleştirme havuzundaki kromozomlardan çıkartılır. N_{iyi+1} ’inci, sıradaki kromozomun uygunluk değeri – 13079’dur. Eşleştirme havuzundaki her bir kromozomun normalize uygunluk değerleri ve seçilme ihtimalleri aşağıdaki gibi hesaplanır.

$$C_n = f(\text{kromozom}_n) - f(\text{kromozom}_{N_{iyi+1}}) \quad (26)$$

$$P_n = \left| \frac{C_n}{\sum_{p=1}^{N_{iyi}} C_p} \right| \quad (27)$$

Buna göre Tablo 12’de üstteki kromozomların daha ağırlıklı olarak sıralamada yer aldığı görülmektedir. Daha önce üretilen rasgele sayılar kullanılırsa kromozom₁-kromozom₃, kromozom₁-kromozom₁ ve kromozom₃-kromozom₁ eşleştiği görülür.

Tablo 12 Eşleştirme havuzunda kromozomların sıralanışı [139]

n	Kromozom	C_n	P_n	$\sum_{i=1}^n P_i$
1	0010011001 1101	-13778+13079= -699	0.4401	0.4401
2	00010100011011	-13360+13079= -281	0.1772	0.6174
3	01010110000100	-13338+13079= -259	0.1632	0.7805
4	00010110000010	-13255+13079= -176	0.1109	0.8915
5	00001101010110	-13166+13079= -87	0.0547	0.9461
6	000100001101 01	-13164+13079= -85	0.0539	1.0000

5.4.5 Çaprazlama

İki adet yeni nesil elde etmek için kromozomların bulunduğu eşleme havuzundan iki adet kromozom seçilir. Eşleme sürecinde, seçilen kromozomlardan bir ve birden fazla yeni nesil oluşturma olayına “çaprazlama” denir. En yaygın olarak kullanılan iki kromozomdan iki tane yeni nesil elde edilmesidir. Öncelikle kromozomun ilk biti ile son biti arasında çaprazlama noktası rasgele seçilir. Birinci kromozomun çaprazlama noktasının solundaki bitlerin hepsi ikinci kromozoma geçerken, benzer tarzda ikinci kromozomun çaprazlama noktasının solundaki bitler de birinci kromozoma geçerler. Sonuçta meydana gelen yeni kromozomlar, her iki kromozomun karakterlerini içermektedirler. Çaprazlama oranı, bir popülasyonda ne kadar kromozomun çaprazlamaya tabi tutulacağını belirler.

$$\begin{array}{lcl}
 \text{Kromozom}_1 = \overbrace{00100110011}^{\text{yer degisecek}} 101 & \text{Nesil}_1 = 01010110000101 \\
 \downarrow \uparrow \\
 \text{Kromozom}_2 = \overbrace{01010110000}^{\text{yer degisecek}} 100 & \text{Nesil}_2 = 00100110011100i
 \end{array}$$



Şekil 42 Tek noktalı çaprazlama



Şekil 43 Gerçek kodlu GA’da çaprazlama

Gerçek kodlu GA’da sonraki neslin reel sayılarını değıştirmedigine dikkat edin (Şekil 43). Çünkü çaprazlama noktası her zaman reel sayılar arasındadır. (1,5,10 ve 14) Sonuçta meydana gelen nesiller her iki kromozomun karakterlerini içermektedirler. Bu durum Tablo 13’de gösterilmiştir.

Tablo 13 Eşleştirmesi yapılan kromozom çiftleri [139]

İşlem Sırası	6xrasgele(2-1)	Kromozom
1	2-1	1
2	5-5	5
3	6-3	3
4	4-5	4
5	1-1	1
6	4-5	4

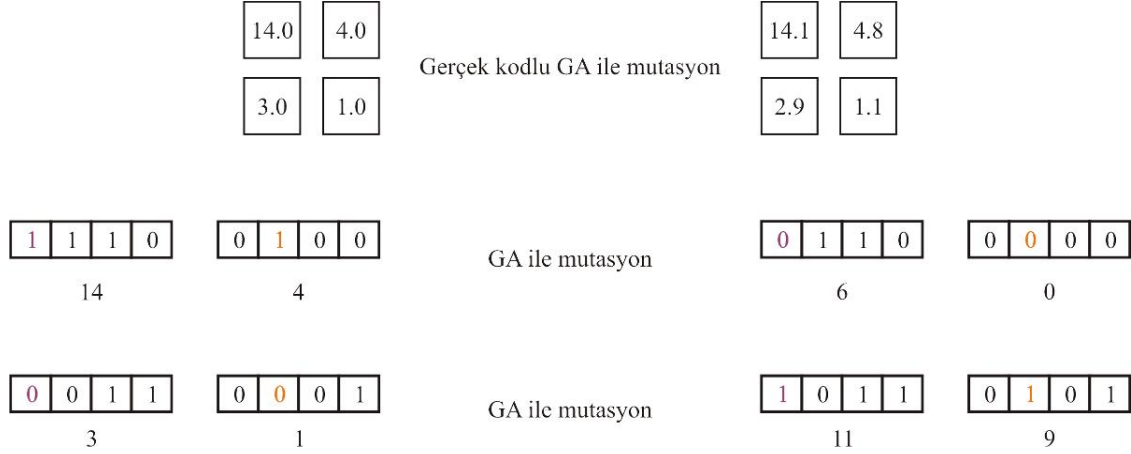
5.4.6 Mutasyon

GA operatörlerinin ikincisi olan mutasyon, kromozomdaki bitlerin küçük bir yüzdesini değıştirir. Mutasyon ile kromozomdaki bitler “1” ise “0”, “0” ise “1” yapılır. $N_{pop} \times N_{bit}$ ’lik bir popülasyon matrisinde, mutasyon noktaları rasgele seçilir. Mutasyonun % değeri artırılırsa, algoritma daha geniş parametre uzayını tarama yapar. Son iterasyonda mutasyon meydana gelmez. Verilen örneğe, en iyi kromozom hariç tutularak %5’lik mutasyon değeri uygulanırsa, rasgele sayı üretici 7 adet kromozom için, satır ve sütunlara karşılık gelecek tam sayılar üretir. Böylece üretilen tamsayılara göre dördüncü satır ve on birinci sütundaki (4–11) bit “0” iken “1” yapılır.

00010110000010 → 00010110001010

(5-3), (2-2), (2-1), (5-14), (8-10) ve (5-8) satır ve sütun numaralarına göre altı kez meydana gelir. Çoğu mutasyonlar, kromozomların uygunluk değeri artırarak daha iyi sonuç elde edilmesine yardımcı olur.

Çaprazlamada olduğu gibi, mutasyonda gerçek kodlu ve ikili kodlu GA’da farklı etkilere sahiptir. ES mutasyonunda, gen stringlerinin değeri, normal bir şekilde dağıtılmış küçük bir sayı ekleyerek değıştirilmesiyle iyi bir ayarlama olarak anlaşılabilir. GA da bir bitin terslenmesinin geniş bir etkisi olabilir,



Şekil 44 GA’da mutasyon işlemleri

5.4.7 Gelecek nesil

İterasyon sonucunda, çaprazlamaya ve mutasyona uğramış kromozomların uygunluk değerleri hesaplanarak Tablo 14’de verilen yeni nesil elde edilir. İtalik yazılan bitler, mutasyona uğrayanlardır. Tablo 15’de görüldüğü gibi bir sonraki adımda kromozomlar uygunluk değerlerine göre yeniden sıralanır ve eşleştirme havuzuna konur. Başlangıç popülasyonun ortalama değeri, –12738 iken birinci nesilden sonra bu değer, –13334’e ulaşmıştır. Tablo 15’deki kromozomların topoğrafik harita üzerinde yerleşimi Şekil 45’de verilmiştir.

Eşleştirme havuzunda kromozomların sadece altı tanesi tutulmaktadır. Tekrar çaprazlama, mutasyon ve sıralamadan sonra elde edilen üçüncü nesil Tablo 16’da gösterilmektedir. Üçüncü neslin sonunda popülasyon genel ortalaması –13403 olmaktadır. Dördüncü neslin sonunda ise –13676 ve yerleşimi Şekil 46’da görülmektedir.

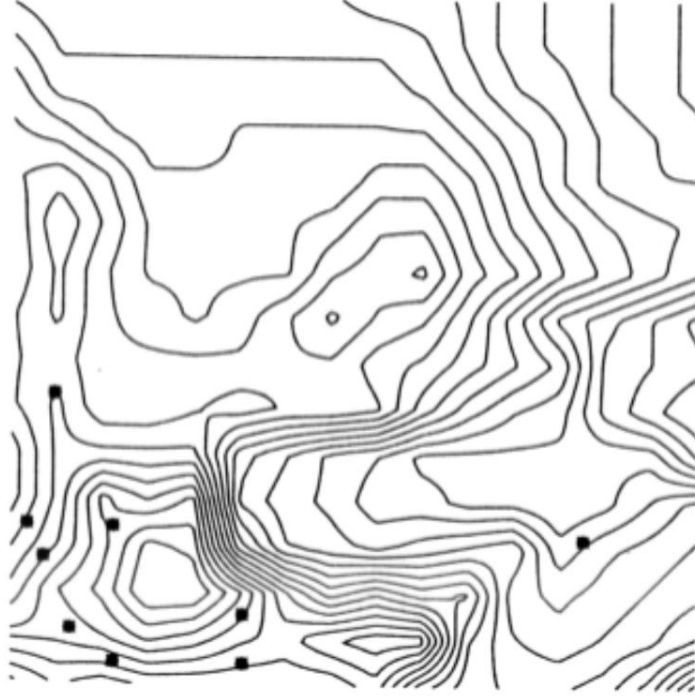
Tablo 14 Çaprazlamaya ve mutasyona uğramış yeni nesil [139]

Kromozomlar	Uygunluk değerleri
00100110011101	-13778
11 010100011011	-11956
01010110000100	-13338
0001011000 1 010	-13553
0000110 0 01011 1	-13289
00010000110101	-13164
00100110000100	-13372
010101100 0 1101	-13632
00 0 00110011101	-13036
00100110011101	-13778
00100110011101	-13778
01010110000100	-13338

Tablo 15 İkinci nesilden sonra sıralama [139]

Kromozomlar	Uygunluk değerleri
00100110011101	-13778
00100110011101	-13778
00100110011101	-13778
010101100 0 1101	-13632
0001011000 1 010	-13552
00100110000100	-13372
01010110000100	-13338
01010110000100	-13338
00 0 00110011101	-13036
0000110 0 01011 1	-13289
00010000110101	-13164
11 010100011011	-11956

Nesil - 2

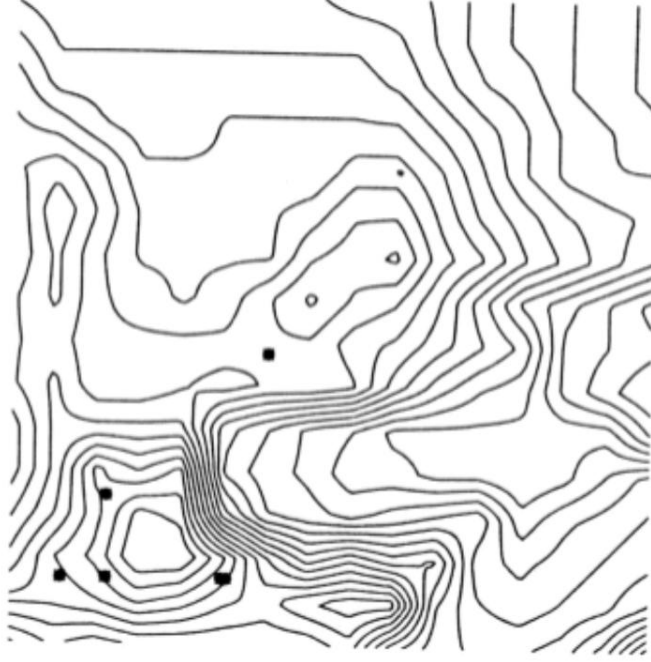


Şekil 45 İkinci neslin topoğrafik harita üzerinde dağılımı [139]

Tablo 16 Üçüncü nesilden sonra sıralama [139]

Kromozomlar	Uygunluk Değerleri
00100110011101	-13778
00100110011101	-13778
00100110011101	-13778
00100110011101	-13778
00100110011101	-13778
01010110001101	-13632
00010110001100	-13584
00010110001010	-13553
01010110001010	-13539
00100111010100	-13921
01101111011101	-13602
01100110011101	-13255

Nesil - 4



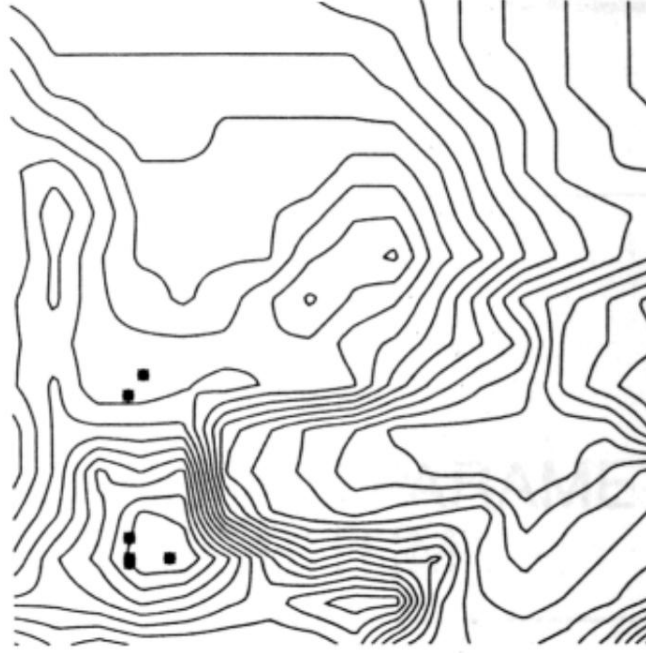
Şekil 46 Dördüncü neslin topoğrafik harita üzerinde dağılımı [139]

5.4.8 Yakınsama

İterasyon sayısına veya uygun bir çözüme ulaşıncaya kadar işleme devam edilir. Bütün kromozomların uygunluk değerleri değişmiyorsa algoritma durdurulmalıdır. Çoğu GA'lar genel ortalama, standart sapma ve minimum uygunluk değeri gibi istatistikleri tutarlar. Bu istatistikler, yakınsama testi için bir ölçü olabilir.

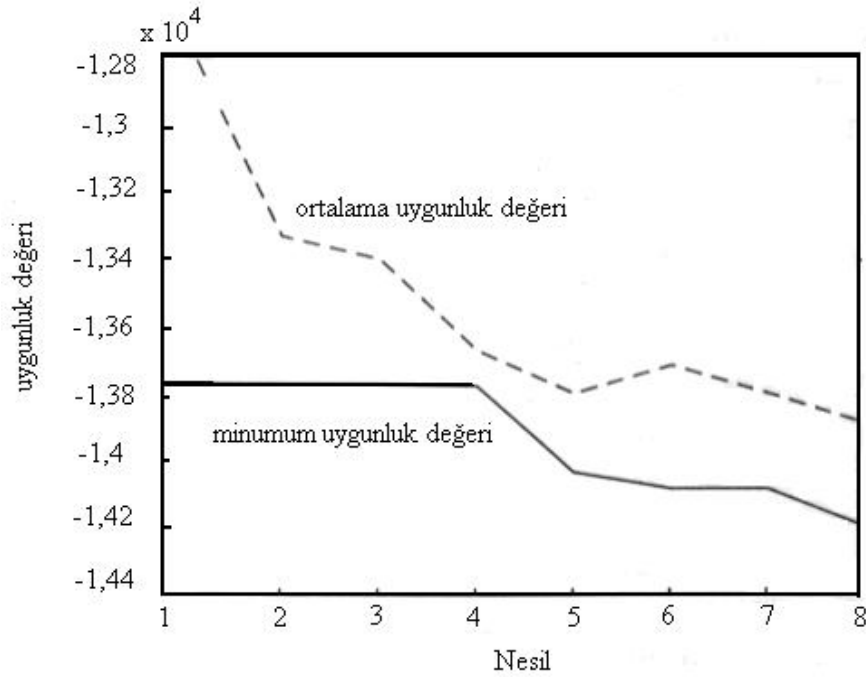
Son popülasyondaki kromozomların yerleşimi Şekil 3.9'da gösterilmektedir. Sekizinci neslin sonunda topoğrafik harita üzerinde iki minimum noktası etrafında yoğunlaşma görülmektedir. Örnekte dokuzuncu iterasyonun sonunda global minimum – 14199 olarak elde edilmiştir.

Nesil - 8



Şekil 47 Sekizinci neslin topoğrafik harita üzerinde dağılımı [139]

Algoritmanın minimuma yakınsaması ve her bir neslin genel ortalaması Şekil 48’de verilmektedir. Popülasyonun ortalama uygunluk değeri, algoritmanın yakınsaması hakkında bilgi vermesi bakımından önemli bir kriterdir. Örneğin maksimizasyon probleminde, farklı iki seçim metodu karşılaştırılıyor ise ortalama uygunluk değeri yüksek olan seçim metodu daha iyi sonuç vermektedir denilebilir.

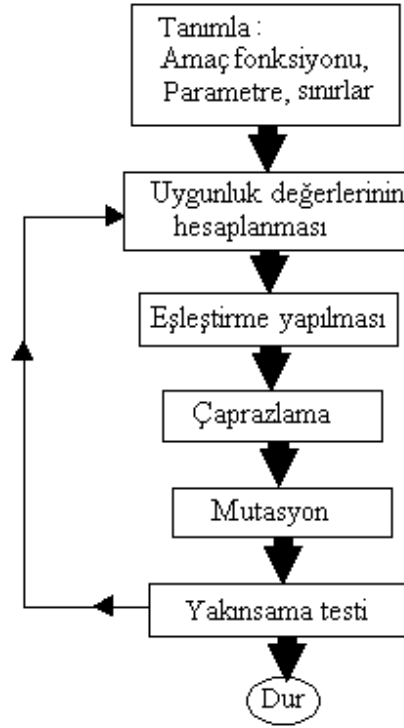


Şekil 48 İlk sekiz neslin ortalama ve minimum uygunluk değerleri [139]

5.5. Gerçek kodlu genetik algoritma

Problemlerin çözümünde rakamların hassasiyeti önemli bir yer tutmaktadır. İkili kodlu GA'da, parametrelerin "1" ve "0" larla ifade edilmesi, kromozomların boyutlarını oldukça artırdığından sınırlı hassasiyete sahiptir. Bunun yerine gerçek rakamlarla kodlama yapabilen, gerçek kodlu GA'yı kullanmak avantajlıdır. Gerçek kodlu GA, hem daha hassas hem de PC belleğinde daha az yer kaplamaktadır. Çunkaş ve Akkaya, değişik test fonksiyonları kullanarak ikili kodlu GA ve gerçek kodlu GA'nın temel farklılıklarını irdemişler. Gerçek kodlu GA 'ların ikili kodlu GA 'ya göre hızlı çalıştığını ve global optimumu daha kısa sürede bulduğunu göstermişlerdir [145].

Gerçek kodlu GA bazı literatürlerde "Sürekli Parametrelili GA" olarak da bilinmektedir. Bu çalışmada "Gerçek Kodlu GA" tanımlaması kullanılmıştır. Gerçek kodlu GA, ikili kodlu GA 'ya çok benzemektedir. Aralarındaki en önemli farklılık parametrelerin "1" ve "0" 'lar yerine gerçek rakamlarla ifade edilmesidir. Gerçek kodlu GA'nın akış diyagramı Şekil 49 'da verilmiştir.



Şekil 49 Gerçek kodlu GA’nın akış diyagramı

5.5.1 Amaç fonksiyonu ve parametreler

Optimizasyonda amaç, parametrelerin en uygun değerlerini bulmaktır. Optimize edilecek parametre değerleri, bir dizi haline getirilip kromozom şekline sokularak GA ile çözülür. N_{par} parametreye sahip bir kromozom $1 \times N_{par}$ boyutlu matris biçiminde yazılır.

$$\text{Kromozom} = [P_1, P_2, P_3, \dots, P_{N_{par}}]$$

Gerçek kodlu GA’da parametrelerin hepsi “ondalık” rakamlarla ifade edilir ve her bir kromozomun bir uygunluk değeri (C) vardır.

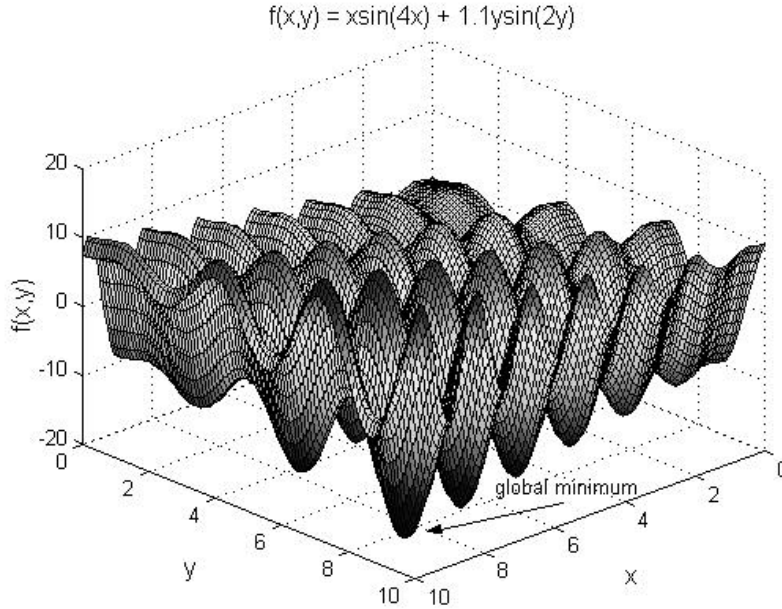
$$C = f(\text{kromozom}) = f(P_1, P_2, P_3, \dots, P_{N_{par}})$$

Örneğin; amaç fonksiyonu

$$C = f(x, y) = x \sin(4x) + 1.1y \sin(2y)$$

ve parametre sınırları $0 \leq x \leq 10$, $0 \leq y \leq 10$ olarak tanımlanırsa Kromozom=[x,y] şeklinde ifade edilebilir. Buradaki problem iki parametrelili bir optimizasyon problemi

olup Şekil 50’de grafik olarak gösterilmiştir. Bu problem, daha zor bir optimizasyon problemidir ve klasik metotlarla bu fonksiyonun global minimumunu bulmak zordur. GA kullanılırsa global minimumu bulmak kolaylaşır.



Şekil 50 $f(x, y) = x \sin(4x) + 1.1y \sin(2y)$ ’nin üç boyutlu görünümü

5.5.2 Başlangıç Popülasyonu

GA’nın çalışmaya başlayabilmesi için, N_{ipop} kromozomdan oluşan başlangıç popülasyonuna ihtiyaç vardır. Popülasyonun kromozomları $N_{par} \times N_{ipop}$ ’luk bir matris ile temsil edilir. Kromozomlar denklem (28) kullanılarak rasgele üretilir.

$$\dot{I}POP = (P_H - P_L) \times \text{rasgele}\{N_{ipop}, N_{par}\} + P_L \quad (28)$$

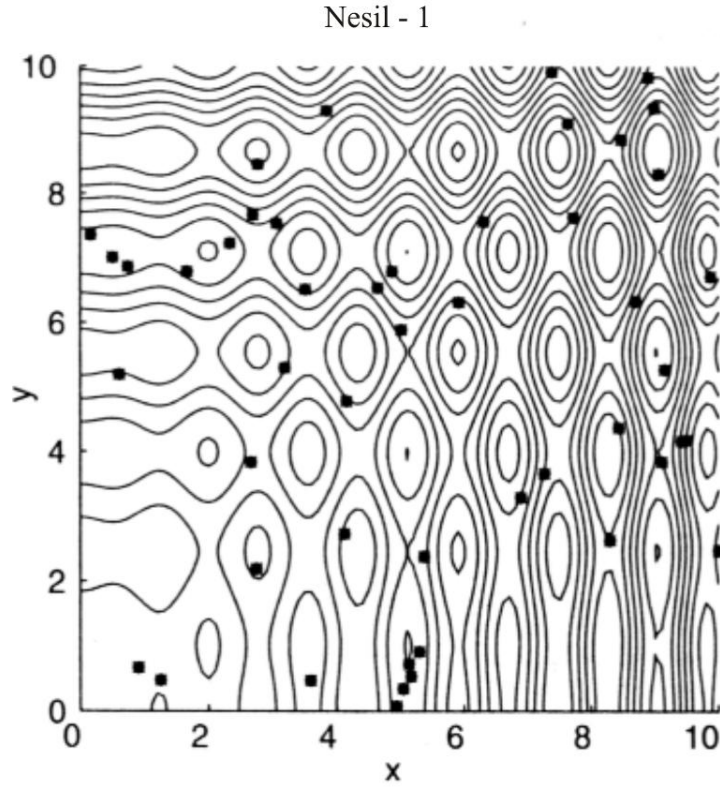
Burada;

$P_H \rightarrow$ Parametrenin üst sınır değeri

$P_L \rightarrow$ Parametrenin alt sınır değeri

$\text{rasgele}\{N_{ipop}, N_{par}\} \rightarrow N_{ipop} \times N_{par}$ olacak şekilde matris formunda 0-1 arasında üretilen rasgele sayıdır.

Bu tanımlamalar her parametre için kullanılabilir. Parametrelerin bu şekilde değerleri belirlendikten sonra amaç fonksiyonunda yerlerine konularak fertlerin uygunluk değerleri hesaplanır. x ve y den oluşan iki tane parametre vardır. Sınır değerleri $P_L = 0$ ve $P_H = 10$ dur. Amaç fonksiyonu biraz karmaşık olduğundan, başlangıç popülasyonunun yüksek tutulması iyi sonuç vermektedir. Başlangıç popülasyon sayısı $N_{ipop}=48$ seçilir ve bu durumda popülasyon matrisi 48×2 'dir. Başlangıç popülasyonun büyüklüğü araştırma uzayının daha geniş seçilmesini sağlar. Başlangıç popülasyonunun amaç fonksiyonunda hesaplanan uygunluk değerlerinin topoğrafik harita üzerinde dağılımı Şekil 51'de gösterilmiştir.



Şekil 51 Birinci neslin topoğrafik harita üzerinde dağılımı [139].

5.5.3 Doğal seçim

Gelecek nesilde, hangi kromozomun yer alacağını belirlemek gerekir. İkili kodlu GA'da yapıldığı gibi kromozomlar, büyükten küçüğe doğru sıralanır. Gerçek kodlu GA'da gelecek iterasyonda kullanılmak üzere N_{pop} kadar kromozom tutulur ve geri kalanı dikkate alınmaz. Doğal seçim, en iyi kromozom uygunluk değeri bulununcaya kadar devam etmelidir. N_{pop} kromozomları, kendi aralarında N_{iyi} ve $N_{kötü}$ olarak büyükten küçüğe doğru sıralamaya tabi tutulur. N_{iyi} , eşleştirme havuzuna atılırken, $N_{kötü}$ hariçte tutulur. Örnekte 48 adet kromozomun ortalama uygunluk değeri 0.9039 ve en iyi kromozomun uygunluk değeri -16.26'dır. Alt sıradan başlanarak 24 tanesi atılırsa, geri kalan kromozomların ortalama uygunluk değeri -4.27 olmaktadır. Tablo 17'de gösterildiği gibi her bir iterasyonda $N_{pop} = 24$ kromozom kullanılır ve bunun 12 tanesi N_{iyi} , 12 tanesi $N_{kötü}$ olarak tanımlanır.

Tablo 17 Yirmidört adet kromozomun sıralanması (Haupt 1998)

No	x	y	Uygunluk Değerleri
1	9.0465	8.3097	-16.2555
2	9.1382	5.2693	-13.5290
3	7.6151	9.1032	-12.2231
4	2.7708	8.4617	-11.4863
5	8.9766	9.3469	-10.3505
6	5.9111	6.3163	-5.4305
7	4.1208	2.7271	-5.0958
8	2.7491	2.1896	-5.0251
9	3.1903	5.2970	-4.7452
10	9.0921	3.8350	-4.6841
11	0.6056	5.1942	-4.2932
12	4.1539	4.7773	-3.9545
13	8.4598	8.8471	-3.3370
14	7.2541	3.6534	-1.4709
15	3.8414	9.3044	-1.1517
16	8.6825	6.3264	-0.8886
17	1.2537	0.4746	-0.7724
18	7.7020	7.6220	-0.6458
19	5.3730	2.3777	-0.0419
20	5.0071	5.8898	0.0394
21	0.9073	0.6684	0.2900
22	8.8857	9.8255	0.3581
23	2.6932	7.6649	0.4857
24	2.6614	3.8342	1.6448

5.5.4 Eşleştirme

Kromozomların eşleştirilmesinde ikili kodlu GA'da tanımlanan Ağırlıklı eşleştirme metodu kullanılmış ve uygunluk değerlerine göre kromozomların seçilme ihtimalleri Tablo 18'de verilmiştir. Çizelgede görüldüğü gibi eşleştirme çoğunlukla yukarıdan aşağıya doğru yapılmaktadır. Çünkü sıralamanın altındaki kromozomların uygunluk değerleri küçüktür. Rasgele üretilen 6 tane sayı aşağıya sıralanmıştır. (0.4679,0.5344), (0.2872,0.4985), (0.1783,0.9554), (0.1537,0.7483), (0.5717,0.5546), (0.8024,0.8907). Üretilen bu sayılar kullanılarak eşleştirme aşağıdaki gibi gerçekleştirilir.

Anne=[3, 2, 1, 1, 4, 5]

Baba=[3, 3, 10, 5, 3, 7]

Tablo 18 Kromozomların uygunluk değerlerine göre seçilme ihtimalleri

n	P_n	$\sum_{i=1}^n P_i$
1	0.2265	0.2265
2	0.1787	0.4052
3	0.1558	0.5611
4	0.1429	0.7040
5	0.1230	0.8269
6	0.0367	0.8637
7	0.0308	0.8945
8	0.0296	0.9241
9	0.0247	0.9488
10	0.0236	0.9724
11	0.0168	0.9892
12	0.0108	1.0000

5.5.5 Çaprazlama

Çaprazlama için çok farklı yaklaşımlar vardır. Adewuya ve Michalewicz tarafından ilginç metotlar ortaya konulmuştur [146, 147]. Kromozomun bir veya birden fazla yerinden bölünerek işaretleme yapılması ve işaretlenen yerlere göre kromozomun karşılıklı genlerinin yerlerinin değiştirilmesi “tek düzen metot” olarak adlandırılır. Aşağıdaki iki adet kromozom ele alınırsa;

$$\text{Kromozom}_1=[P_{a1},P_{a2},P_{a3},P_{a4},P_{a5},P_{a6},\dots\dots\dots P_{aN_{\text{par}}}]$$

$$\text{Kromozom}_2=[P_{b1},P_{b2},P_{b3},P_{b4},P_{b5},P_{b6},\dots\dots\dots P_{bN_{\text{par}}}]$$

Çaprazlama noktaları rasgele seçilir ve parametrelerin genleri karşılıklı değiştirilir.

$$\text{Nesil}_1=[P_{a1},P_{a2},\mathbf{P_{b3}},\mathbf{P_{b4}},P_{a5},P_{a6},\dots\dots\dots P_{aN_{\text{par}}}]$$

$$\text{Nesil}_2=[P_{b1},P_{b2},\mathbf{P_{a3}},\mathbf{P_{a4}},P_{b5},P_{b6},\dots\dots\dots P_{bN_{\text{par}}}]$$

Burada parametrelerin değerleri değişmemektedir ve parametreler sadece gelecek nesil içerisinde farklı yerlerde yer almaktadır. Bu işlem, ikili kodlu GA için iyi bir yöntem kabul edilmesine rağmen gerçek kodlu GA için iyi sonuç vermez.

Karıştırma metotları, gerçek kodlu GA için daha iyi sonuç vermektedir. Burada iki parametrenin değerleri karıştırılır ve aşağıdaki formül kullanılarak yeni nesil elde edilir [148].

$$P_{\text{yeni}} = \beta P_{\text{an}} + (1-\beta)P_{\text{bn}} \quad (29)$$

$\beta = 0$ ve 1 arasında üretilen rasgele sayı

P_{an} = Anne kromozomun n. parametresi

P_{bn} = Baba kromozomun n. parametresi

İkinci nesil, β yerine $1-\beta$ konularak ve birinci neslin tümleyeni alınarak bulunur. $\beta=1$ ise, P_{an} baskın gelir ve P_{bn} ölür. $\beta=0$ ise P_{bn} baskın gelir ve P_{an} ölür. $\beta=0.5$ olduğu zaman [149], sonuç iki parametrenin ortalaması olarak ortaya çıkar. Bu metodun iyi işlediği [147] tarafından gösterilmiştir. Karıştırma işlemi için hangi parametrenin seçileceği belirlenmelidir. Bazen çaprazlama noktalarının solundan veya sağından başlanarak bütün parametreler için lineer bir kombinasyon işlemi yapılır. Çaprazlama noktası rasgele seçilebilir. Parametreler, farklı β değerleri kullanılarak karıştırılabilir. Parametrelerin sınır aralıkları kullanılırsa, iki kromozomun bilgileri etkili olarak karıştırılır. Gerçek kodlu GA karıştırma metotlarının en basiti, “lineer çaprazlama metodu” dur [150]. Bu metotla iki kromozomdan üç tane nesil üretilir.

$$P_{yeni1}=0.5P_{an}+0.5P_{bn} \quad (30)$$

$$P_{yeni2}=1.5 P_{an} -0.5 P_{bn} \quad (31)$$

$$P_{yeni3}= -0.5 P_{an} +1.5 P_{bn} \quad (32)$$

Parametre sınırları dışında kalan herhangi bir nesil dikkate alınmaz. Sonraki popülasyonda kromozomun hayatını diğer iki nesil devam ettirir. Dikkat edilecek olursa β pek fazla değişmemektedir.

Diğer bir çaprazlama metodu olan “Heuristic çaprazlama” da ise β , 0-1 arasında değer almaktadır ve yeni nesil aşağıdaki formüle göre elde edilir [147].

$$P_{yeni}=\beta(P_{an}-P_{bn})+P_{an} \quad (33)$$

Her parametre için farklı β değeri üretilerek yeni nesiller elde edilir. Denklem (33) göre, izin verilen bazı değerlerin dışında üretilen yeni nesiller ihmal edilir. Algoritma, yeni β değeri üreterek işleme devam eder.

“Karma çaprazlama metodu” (blend crossover, BLX- α) bazı parametrelerin α değerlerini belirleyerek işleme başlar [151]. α değeri, parametrelerin dış sınırlarını belirler ve bu sınırlara göre yeni nesiller üretilir.

“Kuadratik çaprazlama” gibi metotlar amaç fonksiyonu için sayısal uygunluk gerçekleştirmeye çalışırlar. Sayısal uygunluğun gerçekleştirilebilmesi için üç adet kromozoma ihtiyaç duyulmaktadır [146].

Heuristik çaprazlama metoduyla extrapolasyon metodunun kombinasyonundan oluşan metot, en çok kullanılanıdır [139]. Bu metotla, ikili kodlu GA’daki çaprazlamaya daha çok yaklaşılmaktadır. Burada ilk olarak çaprazlama noktası rasgele seçilir.

$$\alpha=\text{roundup}\{\text{rasgele} \times N_{\text{par}}\} \quad (34)$$

$$\text{Kromozom}_1=[P_{a1},P_{a2},\dots,P_{a\alpha},\dots,P_{aN_{\text{par}}}]$$

$$\text{Kromozom}_2=[P_{b1},P_{b2},\dots,P_{b\alpha},\dots,P_{bN_{\text{par}}}]$$

Burada m ve n indisleri anne ve baba kromozomlarını tanımlamak amacıyla kullanılmaktadır. Bu metotta, denklem (35) ve denklem (36) kullanılarak yeni nesiller elde edilir.

$$P_{yeni1}=P_{a\alpha}-\beta[P_{a\alpha}-P_{b\alpha}] \quad (35)$$

$$P_{yeni2}=P_{b\alpha}-\beta[P_{a\alpha}-P_{b\alpha}] \quad (36)$$

Burada β , 0-1 arasında bir değerdir. Üretilen bu yeni nesiller kromozomda yerlerine konularak işlem tamamlanır.

$$Nesil_1=[P_{a1}, P_{a2}, \dots, P_{yeni1}, \dots, P_{bN_{par}}]$$

$$Nesil_2=[P_{b1}, P_{b2}, \dots, P_{yeni2}, \dots, P_{aN_{par}}]$$

Eğer kromozomun birinci parametresi seçilirse, seçilen parametrenin sağındaki parametreler değiştirilir. Kromozomun en son parametresi seçilirse, seçilen parametrenin solundaki parametreler değiştirilir. Bu yöntemde $\beta > 1$ 'den büyük olmadıkça sınır değerleri aşan herhangi bir parametre üretilmemektedir. Örneğin;

$$Kromozom_2=[5.2693, 9.1382] \quad Kromozom_3=[9.1032, 7.6151]$$

kromozomlarını ele alalım. Rasgele sayı olarak üretilen $\alpha=1$ için çaprazlama noktası olarak 1. parametre seçilmektedir. İkinci rasgele sayı olarak $\beta=0.7147$ üretilmektedir. Denklem (3.12) ve (3.13)'göre yeni nesil aşağıdaki gibi bulunur.

$$Nesil_3=[5.2693-0.7147 \times 5.2693 + 0.7147 \times 9.1032, 7.6151]=[8.0094, 7.6151]$$

$$Nesil_4=[9.1032 + 0.7147 \times 5.2693 - 0.7147 \times 9.1032, 9.1382]=[6.3631, 9.1382]$$

Diğer kromozomlara da aynı kurallar uygulanarak yeni nesiller elde edilir.

5.6. Mutasyon

Genetik algoritmalar, bazen çok hızlı yakınsarlar. Yakınsama sonucunda global maksimum bulunursa sonuç iyi, lokal minimumlar bulunursa beklenen sonuç elde edilememiş demektir. Hızlı yakınsamadan kurtulmanın yolu, araştırma uzayında, mutasyon aracılığıyla yeni çözümler elde etmektir. İkili kodlu genetik algortmada'da, mutasyon oranı %1-%5 arasında değiştirilirse iyi sonuçlar elde edilir [145]. Gerçek kodlu

GA’da mutasyon oranı ise daha yüksektir. Parametrelerin toplam sayısı ile mutasyon oranı çarpılarak mutasyona girecek parametre sayısı tespit edilir. Bir matriste, satır ve sütunu ifade edecek şekilde rasgele rakamlar üretilerek mutasyon gerçekleştirilir. Örneğin; mutasyon değerini $\mu=0.04$ olarak alınırsa, mutasyona uğrayacak parametre sayısı $0.04 \times 24 \times 2 = 2$ olarak hesaplanır. Çizelge 10’daki kromozom₇’nin birinci parametresi ile kromozom₂₂’nin ikinci parametresi mutasyona uğrarsa, kromozom₇’nin birinci parametresi ile kromozom₂₂’nin ikinci parametresi silinir ve yerine 0–10 arasında yeni bir rasgele sayı üretilerek konur.

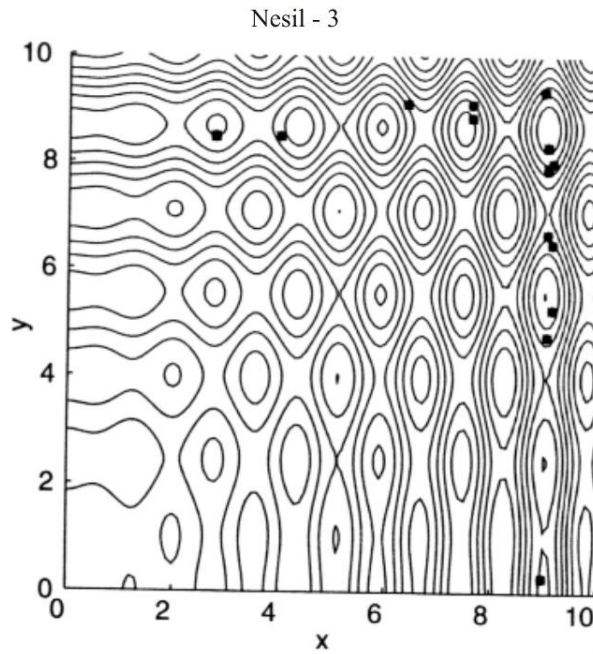
$$\text{Kromozom}_7 = [4.1208, 2.7271] \quad \Rightarrow \quad \text{Kromozom}_7 = [8.6750, 2.7271]$$

Mutasyon süreci tamamlandıktan sonra, yeni popülasyonun sıralaması Tablo 19’daki gibi olur. Algoritma, yedinci iterasyondan sonra minimum uygunluk değerini –18.5 olarak bulmaktadır. Popülasyon üyelerinin topoğrafik harita üzerinde dağılımı Şekil 52, Şekil 53 ve Şekil 54’de gösterilmiştir.

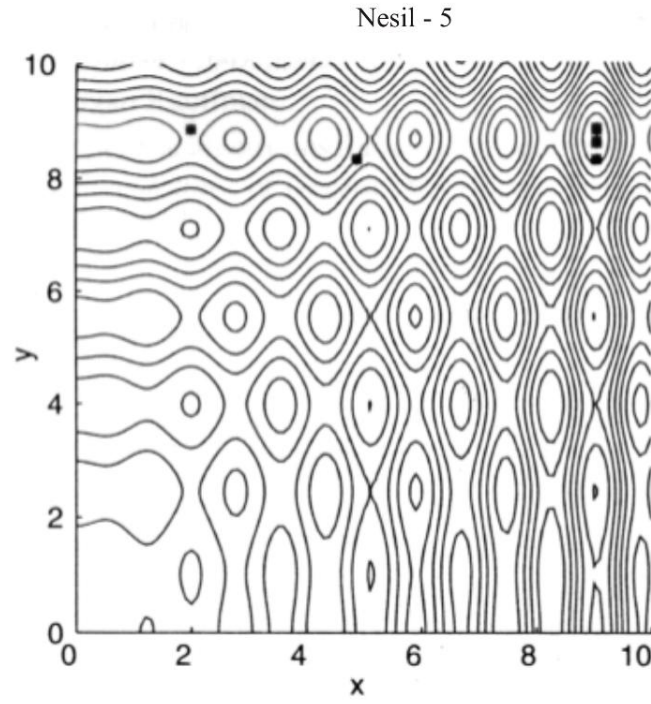
Şekil 52’de görüldüğü gibi popülasyon üyeleri üçüncü iterasyondan sonra iki minimum etrafında toplanmaya başlamaktadır. Şekil 53’de görüldüğü gibi beşinci iterasyondan sonra, bir minimum etrafında toplanmakta ve Şekil 54’daki gibi nihayet yedinci iterasyondan sonra global minimuma (-18.5) ulaşmaktadır. Şekil 55’de gerçek kodlu GA’nın yakınsama grafiği verilmiştir.

Tablo 19 Gerçek kodlu GA'nın ikinci nesilden sonraki sıralanışı (Haupt 1998)

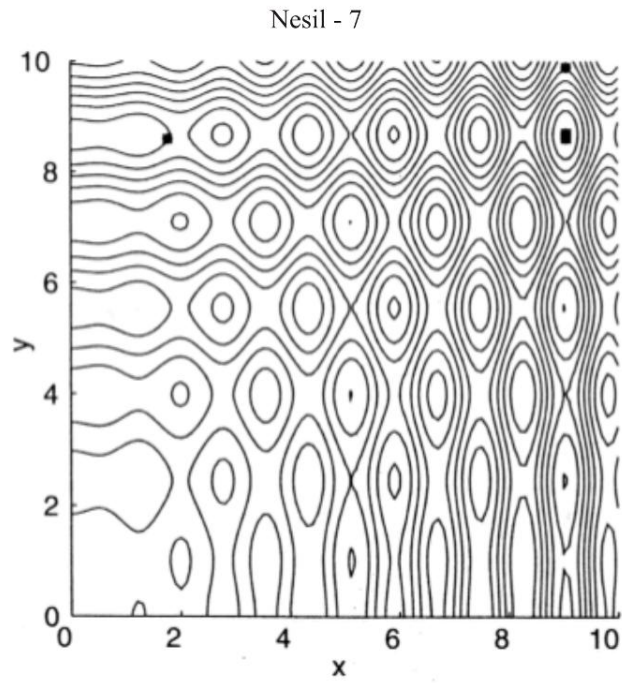
No	x	y	Uygunluk değerleri
1	9.0465	8.3128	-16.2929
2	9.0465	8.3097	-16.2555
3	9.1382	5.2693	-13.5290
4	7.6151	9.1032	-12.2231
5	7.6151	9.1032	-12.2231
6	7.6151	9.1032	-2.2231
7	2.7708	8.4789	-11.6107
8	2.7708	8.4617	-11.4863
9	9.1382	8.0094	-11.0227
10	8.9766	9.3438	-10.4131
11	8.9766	9.3469	-10.3505
12	9.0465	7.9346	-9.8737
13	1.5034	9.0860	-6.6667
14	4.4224	9.3469	-5.6490
15	5.9111	6.3163	-5.4305
16	7.6151	6.3631	-5.1044
17	9.0921	4.2422	-5.0619
18	2.7491	2.1896	-5.0251
19	3.1903	5.2970	-4.7452
20	9.0921	3.8350	-4.6841
21	0.6956	5.1942	-4.2932
22	4.1539	4.7773	-3.9545
23	8.6750	2.7271	-3.4437
24	4.1208	3.1754	-2.6482



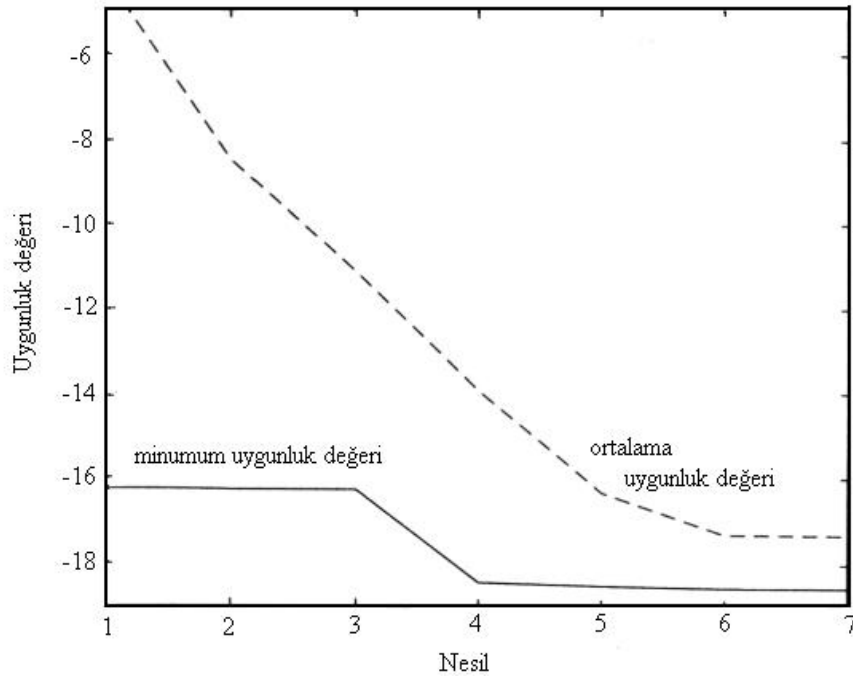
Şekil 52 Üçüncü neslin topoğrafik harita üzerinde dağılımı



Şekil 53 Beşinci neslin topoğrafik harita üzerinde dağılımı



Şekil 54 Yedinci neslin topoğrafik harita üzerinde dağılımı



Şekil 55 Yedi tane neslin ortalama ve minimum uygunluk değerleri [139]

5.7. Örnekler

5.7.1 Bir Açıklayıcı Örnek – GA Nasıl Çalışır

X parametresinin 0 ve 15 arasında değişken olduğu ($15x-x^2$) fonksiyonun maksimum değerini bulalım. Kolaylık için biz x'i sadece integer değerlerden alacağız. Böylece, kromozomlar dört genle ifade edilebilir (Tablo 20).

Tablo 20 Kromozomlar

Integer	Binary code	Integer	Binary code	Integer	Binary code
1	0001	6	0110	11	1011
2	0010	7	0111	12	1100
3	0011	8	1000	13	1101
4	0100	9	1001	14	1110
5	0101	10	1010	15	1111

Farz edelim ki; kromozom popülasyon boyutu 6, çaprazlama olasılığı 0,7 ve mutasyon olasılığı 0.001 dir.(seçilen bu olasılıklar GAlarda yeterince tipiktir.) Uygunluk fonksiyonu şöyle tanımlanır,

$$f(x) = 15x - x^2$$

GA rasgele bir şekilde üretilen birler ve sıfırlarla altı adet 4-bit string ile doldurularak kromozomların başlangıç popülasyonunu oluşturur. Başlangıç popülasyonu Tablo 21’de gösterilen gibi görünebilir. (Gerçek pratik bir problem, tipik olarak binlerce kromozomlu bir popülasyona sahip olacaktır.)

Sonraki adım her bir birey kromozomun uygunluğunu hesaplamaktır. Bu durumda başlangıç popülasyonunun ortalama uygunluğu 36 dır. Bunu geliştirmek için, başlangıç popülasyonu genetik operatörler kullanılarak değiştirilir: seçim, çaprazlama ve mutasyon.

Tablo 21 Rasgele bir şekilde üretilen başlangıç kromozomlarının popülasyonu

Chromosome label	Chromosome string	Decoded integer	Chromosome fitness	Fitness ratio, %	Cumulative ratio
x_1	1100	12	36	16.5 %	16.5
x_2	0100	4	44	20.2 %	36.7
x_3	0001	1	14	6.4 %	43.1
x_4	1110	14	14	6.4 %	49.5
x_5	0111	7	56	25.7 %	75.2
x_6	1001	9	54	24.8 %	100.0

Doğal seçimde, sadece en uygun türler hayatta kalabilir, üreyebilir ve sonraki nesle onların genleri ile geçilebilir. GA lar benzer bir yaklaşım kullanır. Fakat doğadan farklı bir şekilde, kromozom popülasyonunun boyutu bir nesilden sonrakine değiştirilmemiş kalır.

Tablo 21’deki son kolon, popülasyonun toplam uygunluğuna bireysel kromozomların uygunluk oranını gösterir. Bu oran, eşleşme için kromozomların seçilme şansını tanımlar. Böylece, x_3 ve x_4 kromozomlarının çok düşük bir seçilme olasılığı varken, x_5 ve x_6 kromozomlarının yeterli şansı vardır.

5.7.1.1 Seçim

Eşleşme için bir kromozom seçiminde, rulet tekerleği seçimi tekniğini kullanarak, [0,100] aralığında rasgele bir sayı üretilir ve kümülatif oranının rasgele sayıyı içerdiği kromozom seçilir. Bu, uygunluklarıyla orantılı olarak tekerlek üzerinde bir alana sahip her bir kromozomun yer aldığı bir tekerleğin dönmesi gibidir.

Örneğimizde, biz altı kromozomlu bir başlangıç popülasyonuna sahibiz. Böylece, sonraki nesilde aynı popülasyon boyutunu kurmak için, altı rasgele sayı üretilecektir.(bu, rulet tekerleğinin altı kez dönmesi gibidir.) İlk iki dönme, x_6 ve x_2 ebeveyn olmak için seçebilir; ikinci dönme çifti, x_1 ve x_5 kromozomlarını seçebilir ve son iki dönüş x_2 ve x_5 kromozomlarını seçebilir.

5.7.1.2 Çaprazlama

Bir çift ebeveyn kromozom seçildikten sonra, çaprazlama operatörü uygulanır. Öncelikle çaprazlama operatörü rasgele bir şekilde, iki ebeveyn kromozomu kırarak ve bu noktadan sonra kromozomların parçalarını değiştirecek bir çaprazlama noktası seçer. Sonuç olarak iki yeni yavru oluşturulur.

Mesela, iki kromozom x_6 ve x_2 ikinci genden sonra çaprazlanabilir. Her biri iki yavru üretir. Aşağıdaki gibi gösterilir:

x_6	<table border="1"><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr></table>	1	0	0	1	<table border="1"><tr><td>1</td><td>0</td><td>0</td><td>0</td></tr></table>	1	0	0	0	x_6'
1	0	0	1								
1	0	0	0								
x_2	<table border="1"><tr><td>0</td><td>1</td><td>0</td><td>0</td></tr></table>	0	1	0	0	<table border="1"><tr><td>0</td><td>1</td><td>0</td><td>1</td></tr></table>	0	1	0	1	
0	1	0	0								
0	1	0	1								
	Ebeveynler	Çocuklar									

Eğer bir çift kromozom çaprazlanmazsa, yavrular bireylerin tam kopyası olarak oluşturulur. 0.7 çaprazlama olasılığı genellikle iyi sonuç üretir.

5.7.1.3 Mutasyon

Doğada nadiren olan Mutasyon, gendeki bir değişikliği temsil eder. Bu, uygunlukta önemli bir gelişmeye neden olabilir. Fakat çoğunlukla, oldukça faydalı sonuçları vardır. Mutasyonun rolü, arama algoritmasının bir local optimuma takılmamasının garantisini sağlamaktır. Seçim sırası ve çaprazlama operatörleri herhangi bir homojen çözüm kümesinde durgunlaşabilir. Böyle şartlar altında, tüm kromozomlar özdeşdir ve bu yüzden popülasyonun ortalama uygunluğu geliştirilemeyebilir. Çözüm sadece optimal (veya local olarak oldukça optimal) olmak görünebilir. Çünkü arama algoritması daha fazla ilerlemeyebilir. Mutasyon rasgele bir aramaya eşdeğerdir ve genetik farklılıkların kaybının korunmasında bize yardım eder.

Mutasyon operatörü bir kromozomda rasgele bir şekilde seçilen geni çevirir. Mesela, aşağıda gösterildiği gibi, x_1 ' ikinci geninde ve kromozom x_2 de üçüncü geninde mutasyona uğratılmış olabilir.

x_1	<table><tr><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	1	<table><tr><td>1</td><td>1</td><td>0</td><td>1</td></tr></table>	1	1	0	1
1	1	1	1							
1	1	0	1							
x_2	<table><tr><td>0</td><td>1</td><td>0</td><td>0</td></tr></table>	0	1	0	0	<table><tr><td>0</td><td>1</td><td>0</td><td>0</td></tr></table>	0	1	0	0
0	1	0	0							
0	1	0	0							
	Parents	Children								

Mutasyon, bazı olasılıklı kromozomda, herhangi bir gende meydana gelebilir. Mutasyon olasılığı doğada oldukça küçüktür ve GAlarda oldukça küçük tutulur. Tipik olarak 0.001 ile 0.01 aralığında.

Genetik algoritmalar, popülasyonun ortalama uygunluğunun sürekli geliştirilmesini temin eder ve bir nesil sayısından sonra (tipik olarak birkaç yüz) popülasyon bir yakın-optimal çözümü geliştirir. Örneğimizde son çözüm sadece

0	1	1	1
---	---	---	---

 ve

1	0	0	0
---	---	---	---

 kromozomlarından oluşur.

Bu $f(x) = 15x - x^2 = 56$ dır.

Bu örnekte, problem sadece bir değişkene sahiptir. Çözümünü göstermek kolaydır.

5.7.2 Bir GAnın Nasıl Çalıştığını Gösteren Diğer Bir Örnek

$(30x - x^2)$ fonksiyonunun $[0, 31]$ aralığında maksimumunu bulalım. İlk adım değişken kümesi x 'i kodlamaktır. Örneğin, beş-dijit ikili string $[00000 - 11111]$ şeklindedir. Sonra başlangıç popülasyonunu üretiriz. Bu örnekte, aşağıda gösterildiği gibi popülasyon, dört potansiyel çözüm içerir.

Tablo 22 Nesil-0 kromozomları

Chromosome label	Chromosome string	Decoded integer	Chromosome fitness	Fitness ratio, %
1	01000	8	176	24.6
2	01101	13	221	30.8
3	11010	26	104	14.5
4	10010	18	216	30.1
			717	100.0

Başlangıç popülasyonu rasgele bir sayı üretici kullanılarak üretilir. Bu popülasyondan stringler x değişkeninin değerleriyle ilgilidir ve bu değerleri kullanarak biz $f(x)$ fonksiyonunun değerlerini değerlendiririz. Maksimizasyon problemi için, bu fonksiyon seçim işlemi yapabileceğimiz temelde uygunluk fonksiyonu olarak işleme geçirilebilir. Bu popülasyonun toplam uygunluğu 717 dir ve yukarıdaki tablonun son kolonunda her bir kromozomun uygunluğu yüzde olarak ifade edilir.

Seçim işlemi, kromozom çiftlerini bağıntılı kuvvetlerine göre eşleşme için değerlendirir ve seçer. Her bir kromozomun uygunluğuna dayanarak, bir seçim mekanizması, genetik manipülasyon işlemi için eşleşmeleri seçer. Seçim politikası, sonunda en iyi uygunluklu kromozomların(bireylerin) kurtarılması için sorumludur. Bu örnekte, ilk string bir kopyası, ikincinin iki kopyası ve dördüncü stringin bir kopyası seçilir. [ikinci string en yüksek uygunluğa sahiptir. Eşleşme için seçilmeyen üçüncü string ise en düşük uygunluğa sahiptir.] genellikle seçim işleminde ya orantılı seçim(rulet tekerleği seçimi), ya dereceli seçim ya da turnuva seçimi uygulanır.

Genetik operasyon(veya manipülasyon işlemi) genetik operatörleri kullanır: çaprazlama (Ayrıca, iki ebeveynin gen stringlerinin alt parçalarını birleştiren yeniden birleştirme operatörü de denir.) ve mutasyon, genetik bilgiyi işleyerek, yeni bir kromozom popülasyonu (yavrular) üretmek. Çaprazlamada, öncelikle, çaprazlamanın meydana gelip gelmeyeceğine karar verilir ve eğer yapılacaksa, rasgele sayı üreticisi tarafından tanımlanan çaprazlama noktasında stringler yer değiştirir. Aksi takdirde stringler basitçe kopyalanır. Örneğimizde, popülasyon boyutu dörtte korunur ve çaprazlama ilk çift için ikinci pozisyonda ve diğer çift için üçüncü pozisyonda meydana gelir.

Tablo 23 Nesil-0 çaprazlama

Chromosome label	Chromosome Strings (before)	Chromosome Strings (after)
1	01 000	01101
2	01 101	01000
2	011 01	01110
4	100 10	10001

Mutasyon operatörü bazı kromozomları rasgele bir şekilde değiştirerek popülasyonda yeni genetik yapıyı ortaya çıkarır. Arama algoritmasının lokal en iyiye takılmaktan kurtulmasına yardım eder. Mutasyon bazen rasgele bir string değiştirilerek gerçekleştirilir. Yukarıdaki tabloda ikinci kromozomun dördüncü biti mutasyona uğratılmıştır.

Tablo 24 Nesil-0 mutasyon

Chromosome label	Chromosome Strings (before)	Chromosome Strings (after)
1	01101	01101
2	01000	01010
2	01110	01110
4	10001	10001

Çaprazlama ve mutasyondan sonra yeni bir nesil oluşturulur ve sonlanma ölçütü istenilene ulaşıncaya kadar işlem devam eder. Sonlanma ölçütü simülasyon zamanının dolması, nesil sayısını aşması veya bir yakınsama ölçütünü sağlaması durumu olabilir.

Tablo 25 Nesil-1 kromozomları

Chromosome label	Chromosome string	Decoded integer	Chromosome fitness	Fitness ratio, %
1	01101	13	221	25.5
2	01010	10	200	23.1
3	01110	14	224	25.9
4	10001	17	221	25.5
			866	100.0

Tablo 26 Nesil-1 çaprazlama

Chromosome label	Chromosome Strings (before)	Chromosome Strings (after)
1	011 01	01110
3	011 10	01101
3	0111 0	01111
4	1000 1	10000

Tablo 27 Nesil-1 mutasyon

Chromosome label	Chromosome Strings (before)	Chromosome Strings (after)
1	0111 0	01100
3	01101	01101
3	01111	01111
4	10000	10000

Tablo 28 Nesil-2 kromozomları

Chromosome label	Chromosome string	Decoded integer	Chromosome fitness	Fitness ratio, %
1	01100	12	216	24.4
2	01101	13	221	24.9
3	01111	15	225	25.4
4	10000	16	224	25.3
			886	100.0

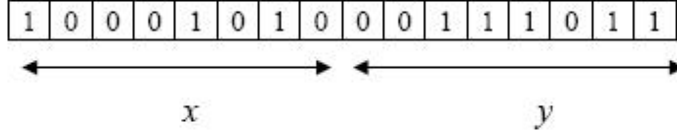
5.7.3 İki Değişken İçeren Bir Başka Örnek

Farz edelim ki, iki değişkenin “peak” fonksiyonunun maksimumunu bulunması istensin.

$$f(x, y) = (1 - x)^2 e^{-x^2 - (y+1)^2} - (x - x^3 - y^3) e^{-x^2 - y^2}$$

x ve y parametreleri -3 ve 3 arasındadır.

İlk adım kromozom olarak problem değişkenlerini temsil etmektir. x ve y parametrelerini birleştirilmiş bir ikili string olarak temsil ederiz:



Her bir parametre sekiz ikili bitlerle gösterilir. Sonra kromozom popülasyonunun boyutunu seçeriz. Örnek için 6 ve rasgele bir şekilde başlangıç popülasyonu üretilir.

Sonraki adım her bir kromozomun uygunluğunu hesaplamaktır. Bu iki seviyede yapılır. Önce, bir kromozom iki reel sayı içine dönüştürülerek kodlanır. x ve y , bunlar -3 ve 3 arasındaki aralıktadır. Sonra kodlanan x ve y peak fonksiyonunun içine koyulur.

5.7.3.1 Kodlama

Önce, 16 bit bir string olan bir kromozom iki 8 bit stringe parçalanır.



Sonra, bu stringler ikiliden (2 tabanından) onluğa (10 tabanına) çevrilir.

$$(10001010)_2 = (138)_{10}$$

ve

$$(00111011)_2 = (59)_{10}$$

Şimdi, 8-bit olarak ele alınabilen, 0'dan $(2^8 - 1)$ 'e, integer'ların aralığı x ve y parametrelerinin gerçek aralığına haritalanır. Bu ise -3 den 3 edir:

$$= 6/(2^8 - 1) = 0.0235294$$

x ve y nin gerçek değerlerini elde etmek için, onluk değerlerini 0.0235294 ile çarpar ve sonuçtan 3 ü çıkarırız:

$$x = (138)_{10} (0.0235294) - 3 = 0.2470588 \quad \text{ve}$$

$$y = (59)_{10} (0.0235294) - 3 = 1.6117647$$

Matematiksel fonksiyonda kodlanan x ve y değerlerini giriş olarak kullanarak, GA her bir kromozomun uygunluğunu hesaplar.

Maksimum “peak” fonksiyonunu bulmak için, 0.7 olasılıkla çaprazlama, 0.001 olasılıkla mutasyon ve 100 nesil sayısı kullanacağız. Bunun anlamı, GA durmadan önce 6 kromozomun 100 neslini oluşturacak olmasıdır.

GA’ların amacındaki en ciddi problem sonuçların kalitesiyle ilgilidir. Sonuçların özellikle en uygun çözüme ulaşıp ulaşılmamasıdır. Güvencenin bazı derecelerini sağlamanın bir yolu farklı mutasyon oranları altında elde edilen sonuçları karşılaştırmaktır. Sabit sonuçların olumlu olması için, biz ayrıca kromozom popülasyon boyutunu artırmalıyız.

5.7.4 İki değişken içeren diğer bir örnek

İkili kodlu bir GA’nın işlem basamakları şöyle tanımlanabilir.

1. Amaç (Objectif) fonksiyonunu tanımlayınız.
2. Parametrelerin sınırlarını tanımlayınız.
3. Popülasyon sayısı kadar rasgele kromozom üretiniz.
4. Parametrelerin ikilik değerlerini aşağıdaki formülü kullanarak 10’luk sisteme çeviriniz.
5. $X_i = a_i + \text{decima}(1001\dots\dots 001_2)(b_i - a_i)/2^{m_i} - 1$
6. Parametreleri amaç fonksiyonunda yerine koyarak uygun (maliyet) değerini hesaplayınız.
7. Popülasyon toplam uygunluk değerini bulunuz.
8. $F = \sum_{i=1}^{po-size} eval(v_i)$
9. Tabii seleksiyon ihtimalini her bir kromozom için hesaplayınız.
10. $P_i = eval(v_i)/F$
11. Her bir kromozom küme latif ihtimalini hesaplayınız.
12. $q_i = \sum_{j=1}^i p_j$
13. Selekse yön süreci Roulette tekerleğinin popülasyon sayısı(pop-size) kadar döndürülmesine dayandırılır.

14. Random olarak popülasyon sayısı kadar 0–1 arasında float bir sayı üretiniz.(r)
Hassasiyet virgülden sonra altı rakam olacaktır.
15. Üretilen random sayı $r < q_1$ ise 1. kromozomu seçiniz. Aksi takdirde V_i . Kromozom seçilir. ($2 \geq i \geq \text{pop-size}$) Üretilen random sayı $q_{i-1} < r \leq q_i$ olursa V_i seçilir.

5.7.4.1 Çaprazlama ve Mutasyon

1. djskd0-1 arasında random bir sayı float olarak üretiniz.(r)
2. Pc çaprazlama oranı, $r < P_c$ ise çaprazlama için verilen kromozomu seçiniz.
3. Çaprazlama için iki kromozom seçilir.
4. Çaprazlama noktasının belirlenmesi için Kromozom genişliğine göre bir sayı üretilir. (1.....m-1). m: kromozom genişliği
5. Üretilen bu sayıyla çaprazlama noktası belirlenmiş olur.
 $B_1b_2 \text{ bpos bpos}+1\text{.....bm}$
 $C_1c_2 \text{ cpos cpos}+1\text{.....cm}$
yeni nesil
 $b_1b_2 \text{ bpos cpos}+1\text{.....cm}$
 $c_1c_2 \text{ cpos bpos}+1\text{.....bm}$
6. $P_c \times \text{pop-size} = \text{Çaprazlamaya maruz kalacak kromozom sayısını verir.}$
 $P_m = \text{mutasyon yüzdesi}$
 $P_m \times \text{pop-size} = \text{Muatsyona maruz kalacak bit sayısını verir.}$
7. 0–1 arasında rasgele bir sayı üretiniz.(r)
8. Eğer $r < P_m$ ise biti mutasyona uğrattınız.

5.7.4.2 Uygulama

Amaç fonksiyonu:

$F(x_1, x_2) = 21.5 + x_1 \sin(4\pi x) + x_2 \sin(20\pi x_2)$ olarak tanımlansın.

Parametrelerin tanım aralığı: $-3 \leq x_1 \leq 12.1$ ve $4.1 \leq x_2 \leq 5.8$

Toplam popülasyon: pop-size=20

Çaprazlama ve mutasyon%leri: $P_c = .25$ $P_m = 0.01$

Amaç: Yukarıdaki iki değişkenli fonksiyonun maksimizasyonu

Her bir değişken için hassasiyet 4 decimal olarak kabul edilsin.

- $X_1 = [-3, 12.1]$ in toplam genişliği $(-3) - (12.1) = 15.1$ ve 15.1×10000 eşit değere bölünmesi gerekir.
 - Bunun anlamı $2^{17} < 151000 \leq 2^{18}$ dir. Buna göre x_1 i ifade etmek için 18 bit gereklidir.
- $X_2 = [4.1, 5.8]$ nin toplam genişliği $5.8 - 4.1 = 1.7$ ve 1.7×10000 eşit değere bölünebilmesi gerekir.
 - Bunun anlamı $2^{14} < 151000 \leq 2^{15}$ dir. Buna göre X_2 yi ifade etmek için 15 bit gereklidir.
- Buna göre iki parametrelili kromozom toplam genişliği $18 + 15 = 33$ bit dir. İlk 18 biti X_1 geri kalanı da X_2 yi temsil etmek amacıyla kullanılır.
- Rasgele üretilen bir kromozom ele alarak işleme devam edelim.
- $$\left(\underbrace{010001001011010000}_X \underbrace{111110010100010}_X \right)$$
- İlk 18 biti ele alırsak;
- $X_1 = -3 + \text{desimal}(010001001011010000)_2 \frac{12.1 - (-3)}{2^{18} - 1},$
- $X_1 = -3 + 70352 \times 15.1 / 262143 = 1.052426$
Aynı yöntemle X_2 şöyle bulunur.
- $X_2 = 4.1 + \text{desimal}(111110010100010)_2 \frac{5.8 - (-4.1)}{2^{15} - 1} = 5.755330$
Buna göre kromozomdaki genlerin değeri şöyledir.
- $[X_1, X_2] = [1.052426, 5.755330]$
- Kromozomun uygunluk değeri $f(X_1, X_2) = f(1.052426, 5.755330) = 20,252640$

Örneğimizde popülasyon sayısı 20 olduğundan, başlangıçta 1 ve 0 'lardan oluşan rasgele olarak 20 adet kromozom üretilir. Biz yukarıda sadece 1 kromozom için hesaplamanın nasıl yapılacağını gösterdik. Geriye kalan 19 adet kromozomun uygunluk değerleri de yukarıdaki gibi hesaplanmalıdır. Rasgele üretilen kromozomlar;

V1	=	(1001101000000001111111010011011111)
V2	=	(111000100100110111001010100011010)
V3	=	(000010000011001000001010111011101)
V4	=	(100011000101101001111000001110010)
V5	=	(000111011001010011010111111000101)
V6	=	(000101000010010101001010111111011)
V7	=	(001000100000110101111011011111011)
V8	=	(100001100001110100010110101100111)
V9	=	(010000000101100010110000001111100)
V10	=	(000001111000110000011010000111011)
V11	=	(011001111110110101100001101111000)
V12	=	(110100010111101101000101010000000)
V13	=	(111011111010001000110000001000110)
V14	=	(010010011000001010100111100101001)
V15	=	(111011101101110000100011111011110)
V16	=	(110011110000011111100001101001011)
V17	=	(011010111111001111010001101111101)
V18	=	(011101000000001110100111110101101)
V19	=	(000101010011111111110000110001100)
V20	=	(101110010110011110011000101111110)

- Uygunluk deęerleri ařaęıdaki gibi hesaplanmıřtır.

Eval(V ₁)	=	f(6.084492,5.652242)	=	26.019600
Eval(V ₂)	=	f(10.348434,4.380264)	=	7.580015
Eval(V ₃)	=	f(-2.516603,4.390381)	=	19.526329
Eval(V ₄)	=	f(5.278638,5.593460)	=	17.406725
Eval(V ₅)	=	f(-1.255173,4.734458)	=	25.341160
Eval(V ₆)	=	f(-1.811725,4.391937)	=	18.100417
Eval(V ₇)	=	f(-0.991471,5.680258)	=	16.020812
Eval(V ₈)	=	f(4.910618,4.703018)	=	17.959701
Eval(V ₉)	=	f(0.795406,5.381472)	=	16.127799
Eval(V ₁₀)	=	f(-2.554851,4.793707)	=	21.278435
Eval(V ₁₁)	=	f(3.130078,4.996097)	=	23.410669
Eval(V ₁₂)	=	f(9.356179,4.239457)	=	15.011619
Eval(V ₁₃)	=	f(11.134646,5.378671)	=	27.316702
Eval(V ₁₄)	=	f(1.335944,5.151378)	=	19.876294
Eval(V ₁₅)	=	f(11.089025,5.054515)	=	30.060205
Eval(V ₁₆)	=	f(9.211598,4.993762)	=	23.867227
Eval(V ₁₇)	=	f(3.367514,4.571343)	=	13.696165
Eval(V ₁₈)	=	f(3.843020,5.158226)	=	15.414128
Eval(V ₁₉)	=	f(-1.746635,5.395584)	=	20.095903
Eval(V ₂₀)	=	f(7.935998,4.757338)	=	13.666916

Yukarıdaki kromozomlar ierisinde uygunluk deęeri en zayıf kromozom V₂, en gl olanı da V₁₅ dir.

- řimdi bir seleksiyon sreci iin rolette tekerleęini uygulayalım.

- Popülasyonun toplam uygunluğu; $F = \sum_{i=1}^{20} eval(vi) = 387.776822$

- Her bir kromozomun seleksiyon ihtimali P_i (i.....20)

$$P_1 = eval(V_1)/F = 0.067099$$

$$P_2 = eval(V_2)/F = 0.019547$$

$$P_3 = eval(V_3)/F = 0.050355$$

$$P_4 = eval(V_4)/F = 0.044889$$

$$P_5 = eval(V_5)/F = 0.065350$$

$$P_6 = eval(V_6)/F = 0.046677$$

$$P_7 = eval(V_7)/F = 0.041315$$

$$P_8 = eval(V_8)/F = 0.046315$$

$$P_9 = eval(V_9)/F = 0.041590$$

$$P_{10} = eval(V_{10})/F = 0.054873$$

$$P_{11} = eval(V_{11})/F = 0.060377$$

$$P_{12} = eval(V_{11})/F = 0.038712$$

$$P_{13} = eval(V_{13})/F = 0.070444$$

$$P_{14} = eval(V_{14})/F = 0.051257$$

$$P_{15} = eval(V_{15})/F = 0.077519$$

$$P_{16} = eval(V_{16})/F = 0.061549$$

$$P_{17} = eval(V_{17})/F = 0.035320$$

$$P_{18} = eval(V_{18})/F = 0.039750$$

$$P_{19} = eval(V_{19})/F = 0.051823$$

$$P_{20} = eval(V_{20})/F = 0.035244$$

- Her bir kromozomun kümülatif ihtimali q_i ise; $q_i = \sum_{j=1}^i p_j$ kullanılarak;

$q_1=0.067099$, $q_2=0.086647$, $q_3=0.137001$, $q_4=0.181890$, $q_5=0.247240$,
 $q_6=0.293917$, $q_7=0.335232$, $q_8=0.381546$, $q_9=0.423137$, $q_{10}=0.478009$, $q_{11}=0.538381$,
 $q_{12}=0.577093$, $q_{13}=0.647537$, $q_{14}=0.698794$, $q_{15}=0.776314$, $q_{16}=0.837863$,
 $q_{17}=0.873182$, $q_{18}=0.912932$, $q_{19}=0.964756$, $q_{20}=1.000000$ hesaplanır.

- Rulette tekerleğini 20 kez döndürelim. Yeni popülasyon için her döndürüşümüzde bir kromozom seçeceğiz. 20 kez döndürmek demek 0-1 arasında 20 adet sayı üretmek anlamındadır.

- Random olarak 0-1 arasında 20 tane sayı üretildi.

$r_1=0.513870$, $r_2=0.175741$, $r_3=0.308652$, $r_4=0.534534$, $r_5=0.947628$,
 $r_6=0.171736$, $r_7=0.702231$, $r_8=0.226431$, $r_9=0.494773$, $r_{10}=0.424720$, $r_{11}=0.703899$,
 $r_{12}=0.389647$, $r_{13}=0.277226$, $r_{14}=0.368071$, $r_{15}=0.983437$, $r_{16}=0.005398$,
 $r_{17}=0.765682$, $r_{18}=0.646473$, $r_{19}=0.767139$, $r_{20}=0.780237$.

ilk sayı $r_1=0.513870$, q_{10} 'dan büyük q_{11} 'den daha küçüktür. Bunun anlamı v_{11} kromozomu yeni popülasyon için seçilir. İkinci sayı $r_2 = 0.175741$, q_3 ten büyük q_4 'ten küçüktür yani v_4 kromozomu yeni popülasyon için seçilir.

- Sonuç olarak yeni popülasyon aşağıdaki kromozomları içerir:

V'1=(011001111110110101100001101111000) (v11)
V'2=(100011000101101001111000001110010) (v4)
V'3=(00100010000011010111101101111011) (v7)
V'4=(011001111110110101100001101111000) (v11)
V'5=(000101010011111111110000110001100) (v19)
V'6=(100011000101101001111000001110010) (v4)
V'7=(111011101101110000100011111011110) (v15)
V'8=(000111011001010011010111111000101) (v5)
V'9=(011001111110110101100001101111000) (v11)
V'10=(000010000011001000001010111011101) (v3)
V'11=(111011101101110000100011111011110) (v15)
V'12=(010000000101100010110000001111100) (v9)
V'13=(00010100001001010100101011111011) (v6)
V'14=(100001100001110100010110101100111) (v8)
V'15=(101110010110011110011000101111110) (v20)
V'16=(100110100000001111111010000111011) (v1)
V'17=(000001111000110000011010000111011) (v10)
V'18=(111011111010001000110000001000110) (v13)
V'19=(111011101101110000100011111011110) (v15)
V'20=(110011110000011111100001101001011) (v16)

Şimdi çaprazlama operatörünü uygulayalım

- Çaprazlama operatörü $P_c=0.25$ verildiğinden ortalama olarak kromozomların %25 i çaprazlamaya maruz kalacaktır. Pop-size=20 olduğundan çaprazlamaya maruz kalan popülasyon sayısı 5'dir.
- Yeniden random olarak 0-1 arasında 20 tane sayı üretilir. Eğer $r < 0.25$ ise çaprazlama için o kromozomu seçeriz. Rasgele üretilen sayılar aşağıdaki gibi olsun.

$r_1=0.822951$	0.151932	0.625447	0.314685	0.346901
0.917204	0.519760	0.401254	0.606758	0.785402
0.031523	0.869921	0.166525	0.674520	0.758400
0.581893	0.389248	0.200232	0.355635	$r_{20}=0.826927$

- $P_c=0.25$ olduğuna göre; yukarıda üretilen rasgele sayılardan 4 tanesi 0.25 altında olduğu için bunlar çaprazlamaya tabii tutulacak kromozomlardır. (V2',V11',V13',V18')
- Şans eseri 4 tane kromozom seçildi. Yani çift sayılıdır. Eğer sayı tek olsaydı ya bir tane kromozom ilave edecektik Veya bir tane kromozom kaldıracaktık. Bu seçimde rasgele olarak yapılır.
- Rasgele eleştirelim (V2',V11'), (V3',V18')
- Bu iki çift için rasgele popülasyonun toplam uzunluğu(1–32) içinde kalacak şekilde bir tam sayı üretilir.(kromozom toplam uzunluğu=33).
- Pos sayısı çaprazlama (crossing) yapılacak pozisyonu belirtmektedir. İlk kromozom çifti için;

V'2 = (100011000101101001111000001110010)

V'11 = (111011101101110000100011111011110)

ve pos sayısı 9 olsun, kromozomlar 9. pozisyonundan itibaren kesilir ve yer değiştirilir. Ortaya çıkan kromozomlar şöyledir:

$$V''2 = (10001100 \text{ } 1101110000100011111011110)$$

$$V''11 = (111011100 \text{ } 0101101001111000001110010)$$

İkinci çift kromozom;

$$V'13 = (000101000010010101000000001000110)$$

$$V'18 = (11101111101000100011101011111011)$$

İkinci çift için üretilen pozitif tam sayı 20 dir. 20. bitten sonraki bitler karşılıklı olarak yer değiştirirler.

$$V''13 = (000101000100101010 \text{ } 1101011111011)$$

$$V''18 = (111011111010001000 \text{ } 0101011111011)$$

- Çaprazlamadan sonra ortaya yeni nesil çıkar. ($V'13''$, $V'2''$, $V'18''$, $V'11''$).
- Popülasyonun yeni versiyonu aşağıdaki gibi dizilirler.

$$V'1 = (011001111110110101100001101111000)$$

$$V''2 = (10001100 \text{ } 1101110000100011111011110)$$

$$V'3 = (00100010000011010111101101111011)$$

$$V'4 = (011001111110110101100001101111000)$$

$$V'5 = (000101010011111111110000110001100)$$

$$V'6 = (100011000101101001111000001110010)$$

$$V'7 = (111011101101110000100011111011110)$$

$$V'8 = (00011101100101001101011111000101)$$

$$V'9 = (011001111110110101100001101111000)$$

$$V'10 = (000010000011001000001010111011101)$$

$$V''11 = (11101110 \text{ } 0101101001111000001110010)$$

$$V'12 = (010000000101100010110000001111100)$$

$$V''13 = (000101000100101010 \text{ } 1101011111011)$$

$V'_{14} = (100001100001110100010110101100111)$
 $V'_{15} = (101110010110011110011000101111110)$
 $V'_{16} = (1001101000000001111111010000111011)$
 $V'_{17} = (000001111000110000011010000111011)$
 $V''_{18} = (11101111101000100001010101111011)$
 $V'_{19} = (111011101101110000100011111011110)$
 $V'_{20} = (110011110000011111100001101001011)$

Şimdide Mutasyon operatörünü uygulayalım.

- Her bir generasyonda; mutasyon operatörü $P_m=0.01$ olduğundan, kromozomun toplam bit sayısı $33 \times 20 = 660$ bitin mutasyona maruz kalacağı tahmin edilir. Mutasyon için her bir bit eşit şansa sahiptir.
- 0–1 arası kromozomun toplam bit sayısı=660 kadar rasgele sayı üretilir.(r).
- Kromozomda eğer $r_i < 0.01$ ise mutasyon gerçekleştirilir.
- Rastgele üretilen 660 tane sayının 5 tanesi 0.01 den küçüktür (Tablo 29).

Tablo 29

Bit pozisyonu	R_i
112	0.000213
349	0.009945
418	0.008809
429	0.005425
602	0.002836

Aşağıdaki tabloda da hangi kromozomun kaçınıcı biti mutasyona uğradığı mavi renk ile verilmiştir.

$V'_1 = (011001111110110101100001101111000)$
 $V''_2 = (10001100 \text{ 1101110000100011111011110})$

V'3 = (001000100000110101111011011111011)
V'4 = (011001111110110101100001101111000)
V'5 = (000101010011111111110000110001100)
V'6 = (100011000101101001111000001110010)
V'7 = (111011101101110000100011111011110)
V'8 = (000111011001010011010111111000101)
V'9 = (011001111110110101100001101111000)
V'10 = (000010000011001000001010111011101)
V''11 = (11101110010110100111000001110010)
V'12 = (010000000101100010110000001111100)
V''13 = (0001010001001010101101011111011)
V'14 = (100001100001110100010110101100111)
V'15 = (101110010110011110011000101111110)
V'16 = (100110100000001111111010000111011)
V'17 = (000001111000110000011010000111011)
V''18 = (1110111110100010000101011111011)
V'19 = (111011101101110000100011111011110)
V'20 = (110011110000011111100001101001011)

Tablo 30

Bit pozisyonu	Kromozom numarası	Kromozom içindeki bit numarası
112	4	13
349	11	19
418	13	22
429	13	33
602	19	8

Tablo 30'a göre 13. kromozomda iki defa mutasyon gerçekleştirilir. Buraya kadar sadece 1 iterasyon/generasyon tamamlandı. İkinci iterasyon için, popülasyonun uygunluk değerleri amaç fonksiyonunda yerine konularak hesaplanır. Yeni uygunluk değerleri aşağıdaki gibidir.

eval(v ₁)	=	f(3.130078,4.996097)	=	23.410669
eval(v ₂)	=	f(5.279042,5.054515)	=	182010839
eval(v ₃)	=	f(-0.991471,5.680258)	=	16.020812
eval(v ₄)	=	f(3.128235,4.996097)	=	23.412613
eval(v ₅)	=	f(-1.746635,5.395584)	=	20.095903
eval(v ₆)	=	f(5.278638,5.593460)	=	17.406725
eval(v ₇)	=	f(11.089025,5.054515)	=	30.060205
eval(v ₈)	=	f(-1.255173,4.734458)	=	25.341160
eval(v ₉)	=	f(3.130078,4.996097)	=	23.410669
eval(v ₁₀)	=	f(-2.516603,4.390381)	=	19.526329
eval(v ₁₁)	=	f(11.088621,4.743434)	=	33.351874
eval(12)	=	f(-0.795406,5.381472)	=	016.127799
eval(13)	=	f(-1.811725,4.209937)	=	22.692463
eval(14)	=	f(4.910618,4.703018)	=	17.959701
eval(15)	=	f(7.935998,4.757338)	=	013.666916
eval(16)	=	f(6.084492,5.652242)	=	26.019600
eval(17)	=	f(-2.554851,4.793707)	=	21.278435
eval(18)	=	f(11.134646,5.666975)	=	27.591064
eval(19)	=	f(11.059532,5.054515)	=	27.608441
eval(20)	=	f(9.211598,4.993762)	=	23.867227

Yeni popülasyonun toplam uygunluğu $F=447.049688$ dir. Önceki popülasyonun uygunluk değeri 387.776822 idi. aynı zamanda önceki popülasyondaki en iyi kromozom $v_{15}=30.060205$ ’dan daha iyi kromozom $v_{11}=33.351874$ olarak hesaplanmıştır.

Şu anda selection prosesi tekrar çalıştırılır ve genetik operatörler uygulanır, bir sonraki jenerasyonun uygunluğu hesaplanır. 1000 jenerasyondan sonra popülasyon;

V_1	=	(1110111101100110111001010111011)
V_2	=	(111001100110000100010101010111000)
V_3	=	(111011110111011011100101010111011)
V_4	=	(11100110001000110000101010111001)
V_5	=	(1110111101110110111001010110111011)
V_6	=	(111001100110000100000100010100001)
V_7	=	(11010110001001001000110010110000)
V_8	=	(111101100010001010001101010010001)
V_9	=	(1110011000100100110010001100010110001)
V_{10}	=	(111011110111011011100101010111001)
V_{11}	=	(110110110000010010001100010110000)
V_{12}	=	(110101100010010010001100010110001)
V_{13}	=	(1110111011101101110010101011101)
V_{14}	=	(11100110011000010000101010111011)
V_{15}	=	(111001101010111001010100110110001)
V_{16}	=	(111001100110000101000100010100001)
V_{17}	=	(111001100110000100000101010111011)
V_{18}	=	(111001100110000100000101010111001)
V_{19}	=	(11110110001000101000111000001001)
V_{20}	=	(11100110011000010000101010111001)

Uygunluk deęerleri:

Eval(v ₁)	=	f(11.120940,5.092514)	=	30.298543
Eval(v ₂)	=	f(10.588756,4.667358)	=	26.869724
Eval(v ₃)	=	f(11.124627,5.092514)	=	30.316575
Eval(v ₄)	=	f(10.574125,4.242410)	=	31.933120
Eval(v ₅)	=	f(11.124627,5.092514)	=	30.316575
Eval(v ₆)	=	f(10.588756,4.214603)	=	34.356125
Eval(v ₇)	=	f(9.631066,4.427881)	=	35.458636
Eval(v ₈)	=	f(11.518106,4.452835)	=	23.309078
Eval(v ₉)	=	f(10.574816,4.427933)	=	34.393820
Eval(v ₁₀)	=	f(11.124627,5.092514)	=	30.316575
Eval(v ₁₁)	=	f(9.623693,4.427881)	=	35.477938
Eval(v ₁₂)	=	f(9.631066,4.427933)	=	35.456066
Eval(v ₁₃)	=	f(11.124627,5.092514)	=	30.316575
Eval(v ₁₄)	=	f(10.588756,4.242514)	=	32.932098
Eval(v ₁₅)	=	f(10.606555,4.653714)	=	30.746768
Eval(v ₁₆)	=	f(10.588814,4.214603)	=	34.359545
Eval(v ₁₇)	=	f(10.588756,4.242514)	=	32.932098
Eval(v ₁₈)	=	f(10.588756,4.242410)	=	32.956664
Eval(v ₁₉)	=	f(11.518106,4.472757)	=	19.669670
Eval(v ₂₀)	=	f(10.588756,4.242410)	=	32.956664

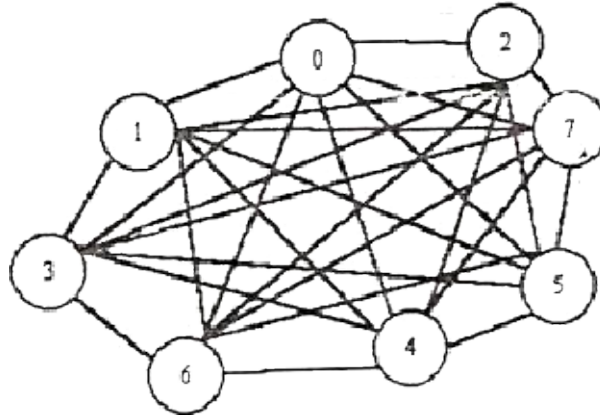
Bununla beraber eęer geliřmeye dikkatlice bakılırsa 1000.jenarasyondan sonra en iyi kromozom olan 35,477938 deęerinden daha iyi kromozomların uygun deęerlerini bulunduran önceki jenerasyonlar meydana gelmiş olabilir. Örneęin 396.jenerasyonundaki 38.827553 deęerlerine sahip kromozom en iyisidir.

Gelişim prosesindeki bir tek en iyi saklanabilir. O genetik algoritmalarındaki uygulamalarda alışılmış olarak daima en iyiyi saklar. Son popülasyondaki en iyi değerinin tersine tüm proseslerin sonunda aralarındaki en iyi değeri rapor eder.

- DİKKAT: En son iterasyonda mutasyon operatörü uygulanmamalıdır.

5.7.5 Gezgin Satıcı Probleminin Genetik Algoritmalarla Uygulaması

GA'ların uygulanmasına ilişkin incelenecek olan GSP'de 8 şehir olduğunu varsayalım. Hatırlatalım ki problemde amaç, satıcının her şehre yalnızca bir kere uğrayarak turunu en kısa yoldan tamamlamasıdır. Satıcının başladığı şehre geri dönme zorunluluğunun olmadığını varsayalım. (Yani burada amaç döngü değil, Hamilton yolunun bulunmasıdır.) Şehirler birbirine bağlı olmakta ve uzaklıklar verilmektedir bu şartlar içerisinde toplam alınan yolun minimum değerini bulmaya çalışacağız.[152].



Şekil 56 8 Şehirli GSP Şeması

5.7.5.1 Başlangıç popülasyonun oluşturulması:

GSP'de 8 şehir olduğunu kabul etmiştik. Şehirleri "0" dan "7" ye kadar numaralandırdığımızı düşünürsek, her gen bir şehre karşılık gelecektir. Bu durumda kromozomlarımız ve başlangıç popülasyonumuz rastlantısal olarak aşağıdaki gibi oluşturulabilir.(Popülasyon büyüklüğü "5" olarak seçilmiştir.)

10275463

71302564

24170536

41653720

60534217

5.7.5.2 *Populasyon içindeki her kromozomun amaç fonksiyonunun değerinin hesaplanması:*

İlk kromozomumuza bakarsak, bu kromozomun amaç fonksiyonu değeri, şartta verilen şehirlerarası yol uzunluklarına bağlı olarak aşağıdaki şekilde hesaplanır:

10275463

1'den 0'a 5 km

2'den 7'ye km

5'den 4'e 3 km

6'dan 3'e 5 km

0'dan 2'ye 3 km

7'den 5'e 4 km

4'den 6'ya 6 km

ise; bu kromozomun gösterdiği yolun toplam uzunluğu 27 km, yani amaç fonksiyonun değeri “27” olacaktır. Artık populasyonumuzun GA operatörlerinin uygulanması için hazırdır. Bu operatörleri uygulayarak populasyonumuzun her yeni nesilde evrilmesini ve daha iyi sonuçlar elde edilmesini sağlayabiliriz.

5.7.5.3 *Tekrar üretme, çaprazlama ve mutasyon operatörlerinin uygulanması:*

Goldberg probleminde uygulanan çaprazlama işlemini GSP'nin kromozomlarına uygularsak bazı güçlüklerle karşılaşabiliriz. Örneğin, rastgele bir şekilde üçüncü ve dördüncü kromozomları seçelim ve çaprazlama noktası 5 olsun. Bu durumda çaprazlama şu şekilde olacaktır.

Çaprazlama öncesi durum;

2 4 1 7 0 **5 3 6**

4 1 6 5 3 **7 2 0**

Çaprazlama sonrası durum;

2 4 1 7 0 **7 2 0**

4 1 6 5 3 **5 3 6**

Görüldüğü gibi yeni oluşan kromozomlar, GSP problemi için olumlu bir sonuç vermez. Çünkü satıcımız, tüm şehirleri gezmeli ve her şehre yalnızca bir kez uğramalıydı, fakat çaprazlama sonrası kromozomlarda bazı şehirlere hiç gidilmemekte yada şehirlere uygun gelen düğümlere iki kere başvurulmaktadır. Örneğin çaprazlama sonrası oluşan ilk kromozomumuza dikkat edecek olursak, 3., 5. ve 6. şehirlere hiç gidilmediğini ve 0., 2. ve 7. şehirlere iki defa uğrandığını gözlemliyoruz. Böyle bir olumsuz durumda ikinci bir işleme gerek duyulur. Yani kısıtlamaları sağlamayan uygunsuz kromozomların standartlaştırılması gerekmektedir. Standartlaştırma için aşağıdaki basit kural uygulanabilir: “Kromozom içinde tekrar eden ilk şehir ziyaret edilmeyen en küçük numaralı şehir ile değiştirilsin”. Bu standartlaştırma kuralı akla ilk gelen kurallardan biridir. Herhangi başka bir kural da standartlaştırma amacı için uygulanabilirdi. Evrimsel programlama böyledir; hemen hemen her şey plansız programsızdır. Tıpkı doğadaki evrim gibi... Sorunlu kromozomların standartlaştırılması ile şu sonucu elde ederiz:

Çaprazlama ve standartlaştırma sonrası durum;

3415**6**720

41**0**27536

Böyle bir durumda ikili düzende kodlanmış olan kromozomların daha avantajlı olabileceği görülmektedir. Ancak ikili düzende kodlama da kromozomların çok uzun olması, ya da problemin yapısına uygun olmaması gibi sorunlar çıkabilmektedir.

Şimdi de Goldberg problemindeki gibi mutasyon operatörüne geçelim. Ancak bu rasgele seçilen kromozomlar üzerindeki iki genin yerlerini değiştirmekle gerçekleştirilir. Rasgele olarak seçtiğimiz 5. kromozomumuzun ikinci ve yedinci genlerini seçtiğimizi varsayalım. Bu durumda; mutasyondan önceki durum mutasyondan sonraki durum

60534217

67534210

5.7.5.4 4. ve 5. Yeni kromozomların amaç fonksiyonu değerlerinin bulunması ve kötü kromozomların popülasyona çıkartılması:

Yeni oluşturulan fonksiyonların değerleri 2. adımda yapılanlara benzer olarak hesaplanır. Seçici bir model söz konusu ise, bu adımdan sonra popülasyonun içindeki en kötü değerli kromozom, popülasyondan çıkartılır.

5.7.5.5 2-5 arasındaki adımların tekrarı.

Konunun başında optimizasyon problemlerinde, optimuma her zaman ulaşamadığını ya da tüm olası sonuçların incelenmesini, bilgisayarlarla dahi mümkün olmadığını söylemiştik. Burada anlatmaya çalıştığımız GENETİK Algoritmalar, Evrim Kuramını temel olarak, bu tür problemler için optimum sonuçları ya da optimumlara yakın sonuçları elde etmek için kullanılmaktadır. Örneğin bu yöntemle 200–300 şehir için bile yaklaşık 20 dk içinde optimuma çok yakın sonuçlar elde edilebilmektedir.

5.8. Sonuç

Evrimin modellenmesiyle birlikte Genetik algoritmaların ele alınmaları çok yakın tarihe dayanmaktadır. Son zamanlarda bu konu ile ilgili çalışmalar oldukça artmaktadır. Bilgisayar teknolojisindeki gelişmelere paralel olarak, GA'ların pratikte uygulanabilirliği artmıştır. Her geçen gün GA'nın adına yeni sempozyumlar düzenlenirken yeni uygulama alanlarına tatbik edildiği de görülmektedir. GA'nın geliştirilmesi alanında; hibridizasyon, parametrelerin optimal seçimi ve adaptasyonu gibi konularda araştırmalar devam etmektedir [153].

İlk bölümde de anlatıldığı gibi klasik metotlar bazı sınırlamalara sahiptir. Bu metotlardaki ciddi sınırlamalardan biri optimizasyon algoritmasının seri çalışmasıdır. GA'da ise PC'leri paralel çalıştırılarak hızlı sonuçlar elde edilebilir. Birkaç tane algoritmanın aynı anda çalıştırılması ve popülasyonları arasında göç olabilmesi paralel işlemcilerle yapılır. Buna ek olarak popülasyondaki fertlerin maliyet değerleri aynı anda hesaplanarak GA'nın yakınsaması hızlandırılır.

GA karar mekanizmasında kullanılabilir mi? sorusu sorulduğunda subjektif maliyet fonksiyonlu GA akla gelmektedir. Bu algoritma market stoklarının

değerlendirilmesi, savaşta askeri stratejilerin geliştirilmesi gibi karar organlarında kullanılabilir.

GA geçici bir heves veya bir moda olmayıp uzun süre gündemde kalacak bir yöntemdir. Tabiatın hesaplama algoritmaları için iyi bir model olduğu bilinmektedir. Bütün bunlardan sonra tabiatta iyilikler ve güzellikler azalmaktadır ve belki belli bir müddet sonra tabiatta global optimuma ulaşacaktır.

5.9. Proje Kapsamında Gerçekleştirilen Genetik Algoritma Simülatörü

5.9.1 Temel Özellikler

Program Data Klasöründeki gerekli klasörlere atılan her türlü belgeyi otomatik olarak algılar ve programa dahil eder. Programın algıladığı dosya uzantıları:

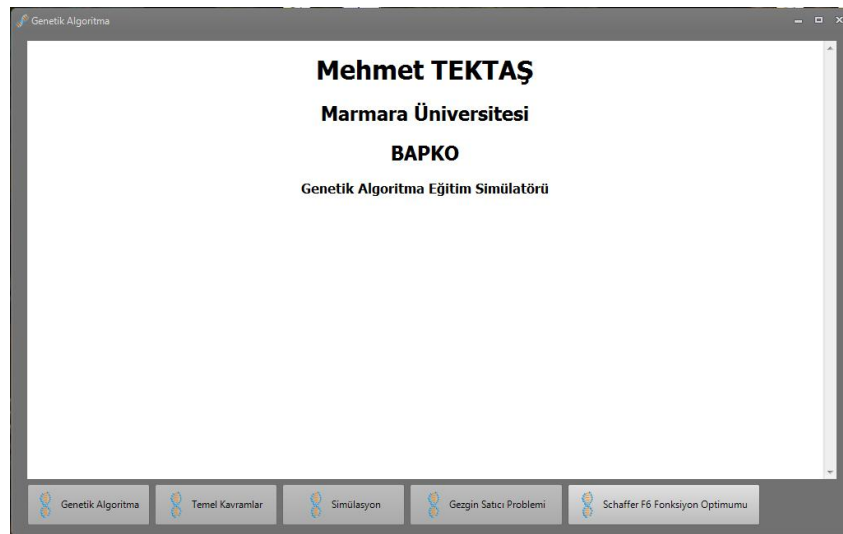
PPS ve PPT (PowerPoint Slaytları)

HTML ve HTM (Web Sayfaları)

.EXE (Çalıştırılabilir Programlar)

5.9.2 Gereksinimler

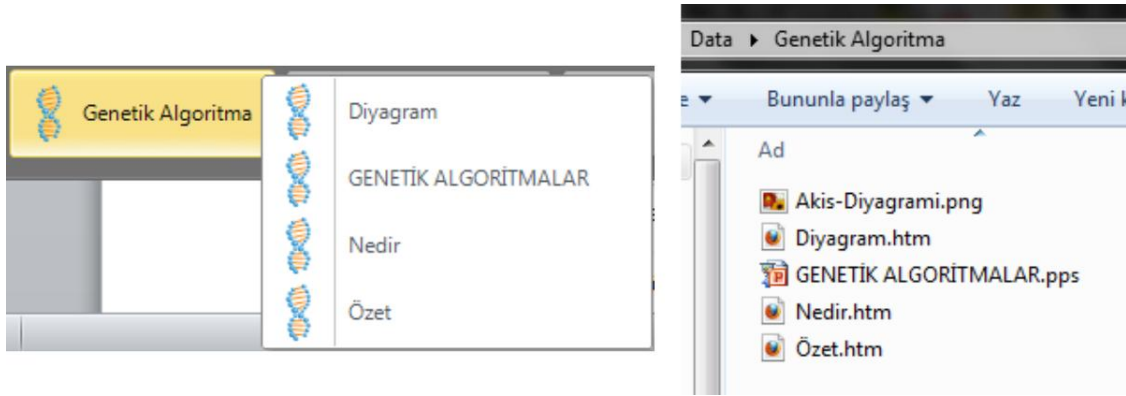
Programın çalışması için .NET Framework 2.0 Kurulması Gerekmemektedir.



Şekil 57 GA simülatörü ana penceresi

5.9.3 Menüler

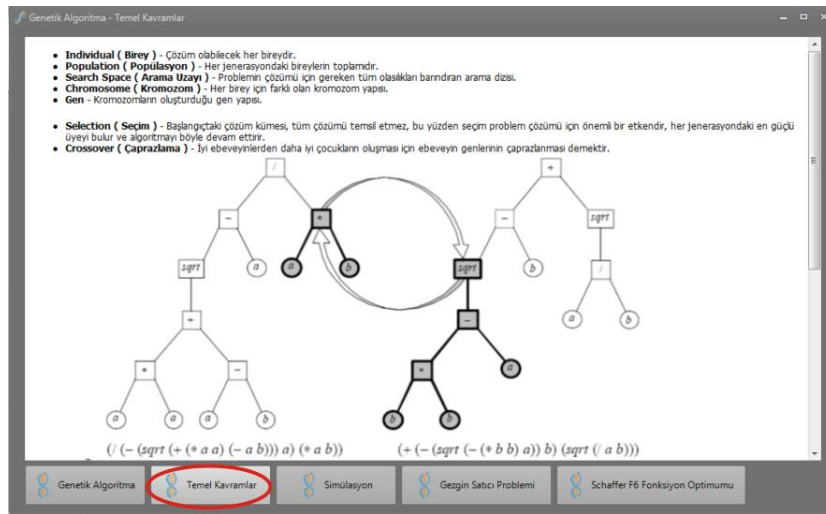
Genetik Algoritma: Data\Genetik Algoritma klasöründeki Power Point Slaytlarını, Html belgelerini ve Çalıştırılabilir dosyaları programa ekleyerek bir arada toplar. Programdan bu dosyalara ulaşılabilir.



Şekil 58 Menüler

5.9.4 Temel Kavramlar

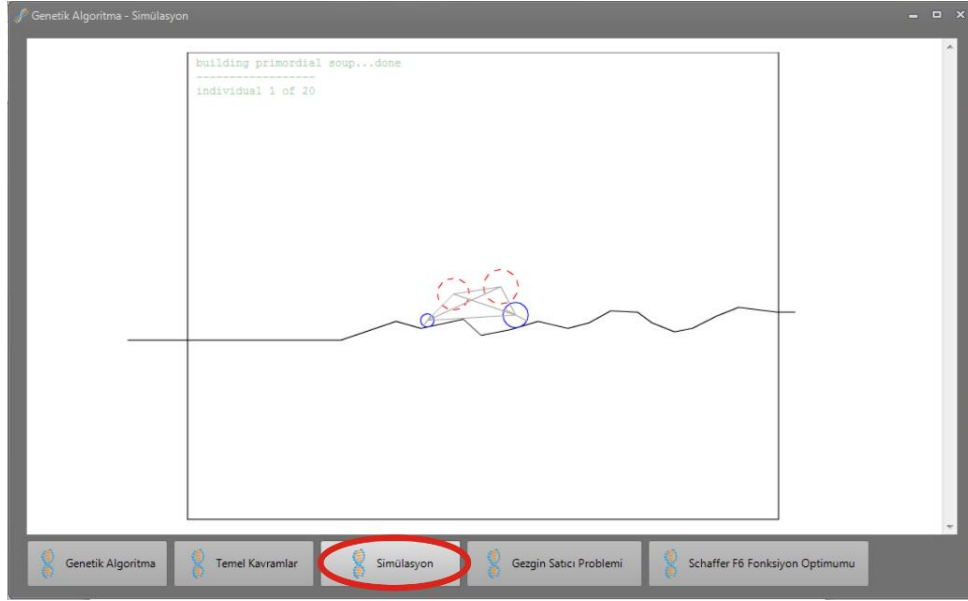
Data\Temel Kavramlar klasöründeki Power Point Slaytlarını, Html belgelerini ve Çalıştırılabilir dosyaları programa ekleyerek bir arada toplar. Programdan bu dosyalara ulaşılabilir.



Şekil 59 Temel kavramlar penceresi

5.9.5 Simülasyonlar

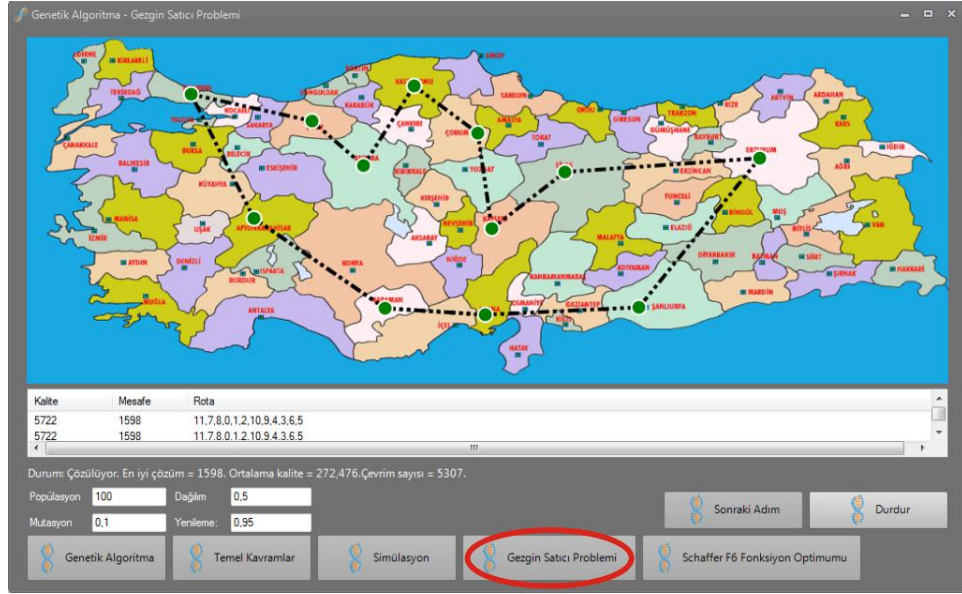
Data\Simülasyonlar klasöründeki Power Point Slaytlarını, Html belgelerini ve çalıştırılabilir dosyaları programa ekleyerek bir arada toplar. Programdan bu dosyalara ulaşılabilir.



Şekil 60 Simülasyon penceresi

5.9.5.1 Gezgin Satıcı Problemi

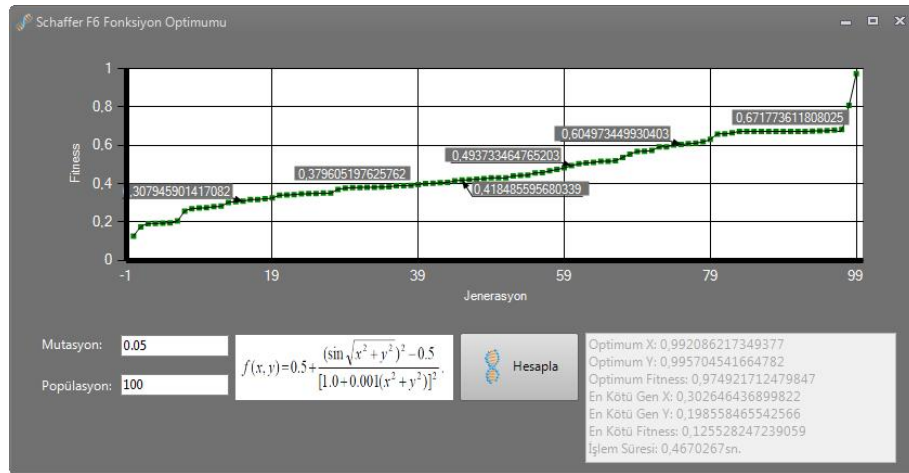
Programın kendi içindeki bir uygulamadır, gezgin satıcı problemini genetik algoritma ile çözüme kavuşturur, Harita üstünde istediğiniz şehirlere tıklayarak nodları oluşturun ve çöz düğmesine tıklamanız yeterlidir.



Şekil 61 Gezgin satıcı problemi ana penceresi

5.9.5.2 Schaffer F6 Fonksiyonu

Programın kendi içindeki bir uygulamadır, Schaffer F6 Fonksiyonunun çözümünü genetik algoritma kullanarak çözer ve grafiğini verir.



Şekil 62 Schaffer F6 fonksiyonunun çözümü

6. Kaynaklar

- [1] N. Taşkin, B. Kandemir, İ.İ. Aydın, “The Effects of Web Based Virtual Geometry Laboratory on Sixth Grade Elementary School Students’ Academic Achievements And Educational Attainments”, 3th International Computer&Instructional Technologies Symposium, October 07-09 2009, Trabzon, TURKEY, pp.409-415.
- [2] F. Uluğ, Z. Kaya, “Uzaktan Eğitim Yaklaşımıyla İlköğretim”, Ankara: Uzaktan Eğitim Vakfı, 1997.
- [3] United States Distance Learning Association, “Glossary of Terms”, http://www.usdla.org/Glossary_Distance.pdf, erişim tarihi: 24.12.2009.
- [4] W. Horton, Designing Web Based Training: How to Teach Anyone Anywhere Anytime, John Wiley&Sons. Inc., New York, USA, 2000
- [5] H.H. Önder “Yapay Zeka Programlama Teknikleri Ve Bilgisayar Destekli Eğitim”. Uluslararası Eğitim Teknolojileri Sempozyumu Bildirileri, Sakarya, Türkiye, 2001.
- [6] V.B. Devedzic “Key Issues in Next-Generation Web-Based Education” IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews, Vol.33, No.3, August 2003, pp.339-349.
- [7] N. Vural, “Uzaktan Eğitimde E-Pedagoji”, TBD Bilişim Zirvesi Bildirileri , 2002.
- [8] A. İşman, M. Barkan, U. Demiray, “Online Distance Education Book”, The Turkish Online Journal of Educational Technology, TOJET Press. 2003.
- [9] Türkiye Bilişim Vakfı (TBV), E-Öğrenme Klavuzu, 2003.
- [10] M. Özarslan, B. Kubat, Ö.F. Bay, “Uzaktan Eğitim İçin Entegre Ofis Dersi’nin Web Tabanlı İçeriğinin Geliştirilmesi ve Üretilmesi”, Akademik Bilişim 2007, Dumlupınar Üniversitesi, Kütahya 31 Ocak-2 Şubat 2007.
- [11] N. Tosun, H. Özgür, İ. Şahin, “E-Öğrenme Ortamlarında Öğrenci-İçerik Etkileşimi” 3th International Computer&Instructional Technologies Symposium, October 07-09 2009, Trabzon, TURKEY, pp.327- 331.

- [12] J. Bourne, J.C. Moore, “Elements of Quality Online Education into the Mainstream”, The Sloan Consortium, Vol.5, ABD, 2004.
- [13] B. Su, C.J. Bonk, R.J. Magjuka, X. Liu, S. Lee, “The Importance of Interaction in Web-Based Education: A Program-level Case Study of Online MBA Courses”, Journal of Interactive Online Learning, Vol.4, No.1, 2005, p.1-19.
- [14] N.K.I. Nettskolen, Online Education - An International Analysis of Web-based Education and Strategic Recommendations for Decision Makers, The Ivette Workshop, Barcelona, 17 November 2000.
- [15] H. Odabaş, “İnternet Tabanlı Uzaktan Eğitim ve Bilgi ve Belge Yönetimi Bölümleri”, Türk Kütüphaneciliği, Sayı: 17, No.1, s.22-36, 2003.
- [16] A.D. Carswell, V. Venkatesh, “Learner Outcomes in an Asynchronous Distance Education Environment”, International Journal of Human-Computer Studies, Vol:56, No.5, pp.475-494, 2003.
- [17] T.D. Lynch, C.E. Lynch, Web-Based Education, The Innovation Journal: An Issue on Processes and Tools, Vol.8, No.4, pp.1-28, 2003.
- [18] G. Bittencourt, E. Costa, M. Silva, E. Soares, “ A Computational Model for Developing Wemantic Web-Based Educational Systems”, Knowledge-Based Systems, Vol.22, pp.302–315, 2009.
- [19] Y. Başkaya, “Web Tabanlı Eğitim”, <http://www.genbilim.com>, 06 Mayıs 2009.
- [20] L.S. Vygotsky, “Mind in Society: The Development of Higher Psychological Processes, Harvard University Press, London, 1978.
- [21] T.C. Resmi Gazete, Sayı:17860, Ankara, 16 Kasım 1982.
- [22] T.C. Resmi Gazete, Sayı:23906, Ankara, 14 Aralık 1999.
- [23] T.C. Resmi Gazete, Sayı:23980, Ankara, 01 Mart 2000.
- [24] YÖK Uzaktan Eğitim Komisyonu Sitesi, <http://uek.aof.edu.tr/default.aspx>, erişim tarihi: Ocak 2010.
- [25] YÖK Başkanlığı’nın 15.07.2007 tarih ve 17626 sayılı yazısı.

- [26] S.C. Kong, Y.Y. Yeung, X.Q. Wub, “An Experience of Teaching for Learning by Observation: Remote-Controlled Experiments on Electrical Circuits”, *Computers & Education*, Vol.52, pp.702–717, 2009.
- [27] J.M.G. Palop, J.M.A. Teruel, “Virtual Work Bench for Electronic Instrumentation Teaching”, *IEEE Transactions on Education*, Vol.43, No.1, pp.15-18, February, 2000.
- [28] O. Palagin, V. Romanov, A. Sachenko I. Galelyuka, V. Hrusha, M. Kachanovska R. Kochan, “Virtual Laboratory for Computer-Aided Design: Typical Virtual Laboratory Structure and Principles of Its Operation”, *IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, pp.77-81, Dortmund, Germany, 6-8 September 2007,
- [29] S. Goldberg, W.F. Horton, T. Agayoff, “Teaching Power Conversion in a Virtual Laboratory”, *IEEE Power Engineering Society Summer Meeting*, Vol.3, pp.1515-1519, 2001.
- [30] M.W. Gertz, D.B. Stewart, P.K. Khosla, “A Human-Machine Interface for Distributed Virtual Laboratories”, *IEEE Robotics & Automation Magazine*, pp.5-13, December, 1994.
- [31] K.M. Carman Cheng, K.W. Eric Cheng, K.W. Chan, “Develop a Web Tool for Electrical Engineering Courses”, *Proceedings of the 2nd International Conference on Power Electronics Systems and Applications*, pp.161-165, Hong Kong, 2006.
- [32] P. Bauer, V. Fedak, O. Rompelman, “PEMCWebLab – Distance and Virtual Laboratories in Electrical Engineering: Development and Trends”, *Power Electronics and Motion Control Conference*, Vol.13, pp.2354 - 2359, 2008.
- [33] E. Akın, M. Karaköse, “Elektrik ve Bilgisayar Mühendisliği Eğitiminde Sanal Laboratuvarların Kullanımı”, http://www.emo.org.tr/..e7e0224018ab3cf_ek.doc, erişim tarihi : Ocak 2010.
- [34] M. Duarte, B.P. Butz, S.M. Miller, A. Mahalingam, “An Intelligent Universal Virtual Laboratory (UVL)” *IEEE Transactions on Education*, Vol.51, No.1, pp.2-9, February, 2008.

- [35] D. Grimaldi, S. Rapuano, "Hardware and Software to Design Virtual Laboratory for Education in Instrumentation and Measurement", Measurement, Vol.42, No.4, pp.485-493, May, 2009.
- [36] E. Bozkurt, A. Sarıkoç, "Fizik Eğitiminde Sanal Laboratuar Geleneksel Laboratuarın Yerini Tutabilir mi?", Selçuk Üniversitesi Ahmet Keleşoğlu Eğitim Fakültesi Dergisi, Sayı:25, S.89 -100, 2008.
- [37] M. Gündüz, Ö.K. Baykan, F. Yıldız, "Elektronik Deneyleri için Sanal Laboratuar Uygulaması", Selçuk Üniversitesi Teknik Bilimler MYO, Teknik-Online Dergi, Cilt:6, Sayı:2, S.61-74, 2007.
- [38] M. Magistris, "A MATLAB-Based Virtual Laboratory for Teaching Introductory Quasi-Stationary Electromagnetics", IEEE Transactions on Education, Vol.48, No.1, pp.81-88, February, 2005.
- [39] A. Eriş, "Bologna Sürecinin Öncesi ve Sonrasında Meslek Yüksekokulları", Marmara Üniversitesi Teknik Bilimler MYO-Sanayi İşbirliği Çalıştayı, İstanbul, Nisan, 2009.
- [40] Yapay Zekâ Nedir?, Temel Kavramlar, Uygulamalar. <http://members.tripod.com/~Bagem/bagem/index.html>
- [41] E. Öztemel "Yapay Zekâ Ne Kadar Yapaydır?" Otomasyon Dergisi, Sayı:126, Kasım 2002.
- [42] HARP Akademileri Komutanlığı Yayınlarından, Uzman Sistemler ve Yapay Zekâ, HARP Akademileri Basımevi, İstanbul, Mart-1996 .HARP Akademileri Komutanlığı Yayınlarından, Adı Geçen Eser, S. 49-51
- [43] Cogito 3 Aylık Düşünce Dergisi, Sayı: 13, Yıl: 1998
- [44] J.F. Baldwin, "Fuzzy Logic and Fuzzy Reasoning", Int. J.Man-Machine St., 11, 465-480, 1979.
- [45] D. Dubois, J. Lang, H. Prade, "Fuzzy Sets in Approximate Reasoning, Part 2 : Logical Approaches", Fuzzy Sets and Systems, 40, 203-244 1991.
- [46] A.A. Fraenkel, Set Theory and Logic, Addison-Wesley P.Co., 1966.

- [47] B.R. Gaines, "Foundations of Fuzzy Reasoning", Int. J.Man-Machine Studies, 8, 623-668, 1976.
- [48] S. Haack, Philosophy of Logic, Cambridge Un. Press., 1978.
- [49] P.E: Johnson, A History of Set Theory, Prindle, Weber, & Schmidt Inc., 1972.
- [50] B. Kosko, Neural Networks and Fuzzy Systems: A Dynamical Systems Approach To Machine Intelligence, Prentice-Hall Int., Inc., 1992.
- [51] E.J. Lemmon, Introduction to Set Theory, Routledge & Kegan Paul Ltd., 1968.
- [52] W. Pedryez, Fuzzy Control and Fuzzy Systems, Research Studies Press Ltd., (2nd Ext. ed.), John Wiley and Sons Inc., 1993
- [53] L.A. Zadeh, "Fuzzy Sets", Information and Control, 8, 338-353, 1965.
- [54] L.A. Zadeh, L.A., "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes", IEEE Trans. on Sys., Man, and Cybernetics, SMC-3, 28-44, 1973.
- [55] L.A. Zadeh, L.A., "Fuzzy Logic and Approximate Reasoning", Synthese, 30, 407-428, 1975.
- [56] M.S. Konuralp, A.H. Işık, E. Taçgın, "Salınan Kol-Kızak Mekanizmaların Kinematik Sentezini Genetik Algoritma Tekniğini Kullanarak Gerçekleştiren Bir Prototip Yazılım", 8. Uluslararası Makina Tasarım ve İmalat Kongresi, 9-11 Eylül 1998.
- [57] R.A. Mansfield, Genetic Algorithms, University of Wales College of Cardiff, 1990.
- [58] O. Gizolme, F. Thollon, "Shape Optimization of Synchronous Machine Rotor", Inter. Journal of Applied Electromagnetics and Mechanics, Vol 9, Issue 3, 1998.
- [59] H.L. Akın, H. Levent "Yapay Zekâda Vücut ve Beyin Problemi", Bilgisayar ve beyin, Nar yayınları, Mart 1997
- [60] R. Akyüz, Ömür "Zihnin Fiziği", Bilgisayar ve beyin, Nar yayınları, Mart 1997.
- [61] C.A. Bingöl, "Öğrenme ve Bellek", Bilgisayar ve Beyin, Nar yayınları., Mart 1997.
- [62] H. Bingöl, "Bilgisayar Üzerine", Bilgisayar ve beyin, Nar yayınları, Mart 1997.

- [63] N. Bozkurt, 20. Yüzyıl Düşünce Akımları, Sarmal yayınevi, Kasım 1995.
- [64] R. Canbeyli, “Psikoloji açısından beyin bilgisayar karşılaştırması”, Bilgisayar ve beyin, Nar yayınları, Mart 1997.
- [65] İ. Cin, “Yapay Zekâ ve Gelecek Korkusu”, Anahtar dergisi, Mayıs 1995
- [66] F. Crick, Şaşırtan varsayım, TUBİTAK yayınları, Nisan 1997
- [67] M.U. Çağlayan, “Beyin, Sinir Sistemi ve Bilgisayar Ağları”, Bilgisayar ve beyin, Nar yayınları, Mart 1997
- [68] Y. Denizhan, “Beyin Bağlamında Kaotik Sistemlere Bakış”, Bilgisayar ve beyin, Nar yayınları, Mart 1997
- [69] D.H. Freedman, “Yeni Nesil Robotlar”, Bilim teknik dergisi, Mayıs 1993
- [70] S. Gillmor, “İnsan sesini algılayan bir uygulama”, BYTE bilgisayar dergisi, Ocak 1998.
- [71] B. Gözkan, “Bilgi, Bilinç ve Yapay Zekâ”, Bilgisayar ve beyin, Nar yayınları, Mart 1997.
- [72] J.P. Haton, Yapay Zekâ, İletişim yayınları, Nisan 1991
- [73] T. Karşıdağ, “Fuzzy Logic: TAI’de Bir Uygulama”, internet adresi: www.kho.edu.tr/~btym/sistem
- [74] E. Öztemel, Yapay Sinir Ağları, Papatya Yayıncılık, Basım, Ağustos 2003.
- [75] A.D.Robert, “Developing Mobile Robot Wall-Following Algorithms Using Genetic Programing”, Applied Intelligence, 8, 33-41, 1998
- [76] A.S.Rana, A.M.S. Zalzala, “Collision-Free Motion Planning of Multiarm Robots Using Evolutionary Algorithm”, Proc. Instn. Mech. Engrs 1997, Vol.211, Part 1, 373-384.
- [77] I. Erkmen, A.M. Erkmen, H. Günver, “Robot Hand Preshaping and Regrasping Using Genetic Algorithms”, The International Journal of Robotics Research, Vol.19, No.9, September 2000, 857-874.

- [78] C.B. Scott, M.P. Kevin, “Intelligent Control for an Acrobat”, Journal of Intelligent and Robotic Systems, Vol 18, 209-248, 1997.
- [79] M. Tektaş, A. Akbaş, V. Topuz, “Yapay Zekâ Tekniklerinin Trafik Kontrolünde Kullanılması Üzerine Bir İnceleme”, Uluslararası Trafik ve Yol Güvenliği Kongresi ve Fuarı , Ankara, Ağustos -2002 .
- [80] V. Topuz, A. Akbaş, M. Tektaş, “Boğaz Köprüsü Yolunda Katılım Noktalarında Trafik Akımlarının Bulanık Mantık Yaklaşımı ile Kontrolü ve Bir Uygulama Örneği”, Uluslararası Trafik ve Yol Güvenliği Kongresi ve Fuarı , Ankara, Ağustos -2002.
- [81] M. Tektaş, N. Tektaş, V. Topuz, “Teknoloji Eğitiminde Yapay Zekâ”, IV.Uluslararası Eğitim Teknolojileri Sempozyumu,24-26 Kasım 2004,Sakarya.
- [82] C. Atkinson, M. Traver, T. Long, E. Hanzevack” This Real-Time, Neural Network-Based Intelligent Performance and Emissions Prediction System is Virtual Genius.” Smoke, June 2002.
- [83] www.automotivedesignline.com
- [84] J.S: Brown, R.R. Burton, and J. de Kleer, “Pedagogical, natural language and knowledge engineering and pedagogical techniques in SOPHIE I, II, and III. In D”, Intelligent Tutoring Systems, New York: Academic Press, 227-282, 1982.
- [85] R.R. Burton, J.S. Brown, (1982). “An investigation of computer coaching. In D. H.”, Intelligent Tutoring Systems (pp. 79-98). New York: Academic Press., 79-98, 1982.
- [86] D. Sleeman, J.S. Brown, Intelligent Tutoring Systems, New York: Academic Press., 1982.
- [87] E. Wenger, Artificial Intelligence and Tutoring Systems, Los Altos CA: Morgan and Kaufmann, 1987.
- [88] J. Psotka, D. Massey, S. Mutter, Intelligent Tutoring Systems: Lessons Learned, Hillsdale NJ: Lawrence Erlbaum, 1988.

- [89] S. Ohlsson, "Some Principles of Intelligent Tutoring", *Instructional Science*, 14, 293-326, 1986.
- [90] R. Schank, D. Edelson, "A Role for AI in education: Using Technology to Reshape Education", *Journal of Artificial Intelligence in Education*, 1 (2), 3-20, 1990.
- [91] Z. Du, G. McCalla, CBMIP -- A case-based mathematics instructional planner. In *Proceedings of the International Conference on the Learning Sciences*, Evanston IL, 1991.
- [92] J.C. Lester, B.W. Porter, "A Student-Sensitive Discourse Generator", In *Proceedings of the International Conference on the Learning Sciences*, Evanston, IL, 1991
- [93] J. Bruneau, A. Chambreuil, M. Chambreuil, M. Chanier, P.L. Dulin, P. Nehemie, "Cognitive Science, Artificial Intelligence, New Technologies: How to Cooperate for a Computer-Assisted Learning to Read System", In *Proceedings of the International Conference on the Learning Sciences*, Evanston, IL, 1991.
- [94] C. Frederiksen, J. Donin, M. DeCary, B. Edmond, "Discourse-Based Second-Language Learning Environments", In *Proceedings of the International Conference on the Learning Sciences*, Evanston, IL, 1991.
- [95] E.W. Cooper, "An Architecture for Apprenticeship: Collaboration with an Intelligent Tutoring System for Qualitative Electrical Troubleshooting", In *Proceedings of the International Conference on the Learning Sciences*, Evanston, IL, 1991.
- [96] J. Frederiksen, B. White, A. Collins, G. Eggan, "Intelligent Tutoring Systems for Electronic Troubleshooting" In J. Psotka, D. Massey, and S. Mutter. *Intelligent Tutoring Systems: Lessons Learned*. Hillsdale NJ: Lawrence Erlbaum, 1988.
- [97] J. Anderson, D.F. Boyle, G. Yost, "The Geometry Tutor" *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, 1985.
- [98] J.R. Anderson, E. Skwarecki, "The Automated Tutoring of Introductory Computer Programming", *Communications of the ACM*, Vol.29, 9, 842-849, 1986.

- [99] J.W. Schofield, D. Evans-Rhodes, B.R. Huber, 990). Artificial Intelligence in the Classroom: The Impact of a Computer-based Tutor on Teachers and Students, Social Science Computer Review, 8(1), 24-41, 1990.
- [100] A.M. Lesgold, S.P. Lajoie, M. Bunzo, G. Eggan, Sherlock: a Coached Practice Environment for an Electronics Troubleshooting Job”, In J. Larkin, R. Chabay, & C. Scheftic (Eds.), Computer assisted instruction and intelligent tutoring systems: Establishing communication and collaboration. Hillsdale, NJ: Lawrence Erlbaum Associates, 1993.
- [101] A.M. Lesgold, G. Eggan, S. Katz, G. Rao, “Possibilities for Assessment Using Computer-Based Apprenticeship Environments”, In W. Regian and V. Shute (Eds.), Cognitive approaches to automated instruction. Hillsdale, NJ: Erlbaum (1992).
- [102] N.M. Allahverdi, S. Yıldız, A. Ünüvar “Endüstride Uzman Sistem Uygulamaları”, 2. Endüstriyel Otomasyon'95 Sempozyumu Bildiriler Kitabı, 30-31 Mart, 1995, İstanbul, s.75-86, 1995.
- [103] C. Hall “The Intelligent Software Development Tools Market, Part 2”, Intelligent Software Strategies, V.12, No:3, pp. 1-16, 1996.
- [104] J.A. Booker, R.C. Kick, “Bringing Expert Systems Technology to the Accounting Classroom, Journal of Accounting and Computers, 3 Fall, pp. 47-57, 1987.
- [105] E. Bonabeau, M. Dorigo, G. Theraulaz, “Inspiration for Optimization from Social Insect Behaviour. Nature, 406, 39-42 -2000.
- [106] Serhat YILMAZ, KOÜ, Bulanık Mantık ve Mühendislik Uygulamaları Ders Notları, 2006.
- [107] Zekai ŞEN, Bulanık (Fuzzy) Mantık ve Modelleme İlkeleri, Bilge Sanat Yapım Yayınları, İstanbul, 2002.
- [108] Ulvi DAĞDELEN, Bulanık Mantık İle Adım Motor Kontrolü, Erciyes Üniversitesi Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 1996.
- [109] http://www.bumat.itu.edu.tr/dokuman_BULANIK_KUMELER.doc.

- [110] G.J. Klir, B. Yuan, Fuzzy Sets and Fuzzy Logic Theory and Applications, Prentice Hall PTR, New Jersey, 1995.
- [111] E. Kıyak, Bulanık Mantık Yöntemiyle Uçuş Kontrol Uygulamaları, Yüksek Lisans Tezi, Anadolu Üniversitesi, Eskişehir, 2003.
- [112] M. Yıldırım, Bulanık Mantık Yapay Sinir Ağı ile Doğrusal Olmayan Sistem Modelleme, Kocaeli Üniversitesi Fen Bilimleri Enstitüsü, Kocaeli, 1998.
- [113] H. Çiftçi, Fuzzy Logic Function Approximation for Some Mathematical Functions, Lisans Tezi, Osmangazi Üniversitesi Elektrik-Elektronik Mühendisliği, Eskişehir, 2002.
- [114] D. Nauck, F. Klawonn, R. Kruse, Fuzzy Sets, “Fuzzy Controllers, and Neural Networks”, Journal of the Humboldt-University of Berlin, Series Medicine 41, No:4, 1992.
- [115] D. Nauck, “Beyond Neuro-Fuzzy: Perspectives and Directions”, Proc. Third European Congress on Intelligent Techniques and Soft Computing, 1159-1164, Aachen, 1994.
- [116] D. Nauck, F. Klawonn, “Neuro-Fuzzy Classification Initialized by Fuzzy Clustering”, Proc. Fourth European Congress on Intelligent Techniques and Soft Computing, Aachen, 1996.
- [117] D. Nauck, R. Kruse, “Choosing Appropriate Neuro-Fuzzy Models”, Proc. European Congress on Intelligent Techniques and Soft Computing, Aachen, 1994.
- [118] D. Nauck, F. Klawonn, R. Kruse, “Combining Neural Networks and Fuzzy Controllers” FLAI’93 Linz, Austria, 1993.
- [119] T.J. Ross, Fuzzy Logic with Engineering Applications. Mc. Graw-Hill, Publishing Co. New York, 1995.
- [120] A.F. Baba, İTÜ Triga Mark-II Reaktörünün Bulanık Mantık Kontrolü, Marmara Üniversitesi Fen Bilimleri Enstitüsü, Doktora Tezi, İstanbul, 1995.
- [121] E. Öztemel, Yapay Sinir Ağları, sf 29, Papatya Yayıncılık, İstanbul, 2003

- [122] S. Haykin, Neural Networks, sf 41, Macmillan Collage Printing Company, New Jersey 1994
- [123] M.Ö. Efe, O. Kaynak, Yapay Sinir Ağları ve Uygulamaları”, sf 1, Boğaziçi Üniversitesi, İstanbul 2004
- [124] E. Öztemel, Yapay Sinir Ağları, sf 36, Papatya Yayıncılık, İstanbul, 2003
- [125] Ç. Elmas, Yapay Sinir Ağları Kuram, Mimari, Eğitim, Uygulama, sf 27-28, Seçkin Yayıncılık, Ankara 2003
- [126] E. Öztemel, Yapay Sinir Ağları, sf 40, Papatya Yayıncılık, İstanbul, 2003
- [127] Ş. Sağıroğlu, E. Beşdok, M. Erler, Mühendislikte Yapay Zeka Uygulamaları-1: Yapay Sinir Ağları, Ufuk Yayıncılık, Kayseri, 77-117, 340-369, 2003.
- [128] W.S. McCullogh, W. Pitts, “A Logical Calculus of the Ideas Immanent in Nervous Activity”, Bulletin of Mathematical Biophysics, 5, 115-133, 1943.
- [129] M.L. Minsky, S.A. Papert, Perceptrons, Cambridge, MA: MIT Press., 1969.
- [130] S.D. Balkin, “Using Recurrent Neural Networks for Time Series Forecasting,” Working Paper, 97-11, Department of Management Science and Information Systems, Pennsylvania State University, 1997.
- [131] D.B. Parker, “Optimal Algorithms For Adaptive Networks: Second Order Back Propagation, Second Order Direct Propagation, and Second Order Hebbian Learning”, IEEE 1st International Conference on Neural Networks, 2, 953-600, San Diego, CA., 1997.
- [132] J.M. Zurada, Introduction to Artificial Neural Systems, West St. Paul., 1992.
- [133] H. Zhimin, L.G. Maryellen, E.M. Charles, Effect of dominant features on neural network performance in the classification of mammographic lesions, Chicago, IL 60637, USA.
- [134] A.C. Yüçetürk “Yapay Sinir Ağları Kullanılarak Örüntü Sınıflandırma ve Tanıma”, Doktora Tezi, Ege Üniversitesi, 2000.)

- [135] G.F. Luger, Artificial Intelligence : Structures and Strategies for Complex Problem Solving, 4th edition, Addison-Wesley, 2002.
- [136] N.J. Nilsson, Artificial Intelligence : A New Synthesis, Morgan Kaufmann Publishers, 1998.
- [137] S.J. Russell, P. Norvig, Artificial Intelligence : A Modern Approach, Prentice-Hall, 1995.
- [138] V. Grant, The Evolutionary process, New York: Columbia University Press., 1985.
- [139] R.L. Haupt, S.E. Haupt, Practical Genetic Algorithms, A Willey-Interscience Publication, USA, 1998.
- [140] H. Curtis, Biology, 2nd Ed., New York: Worth publisher, 1975.
- [141] P.J. Angeline, "Evolution revolution: An Iintroduction to the Special Track on Genetic and Evolutionary Programming," IEEE Expert Intelligent Systems and their Applications 10, pp.6-10, 1995.
- [142] D.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, New York: Addison Wesley, 1989.
- [143] D.E. Goldberg, "Making Genetic Algorithms Fly: A Lesson from the Wright Brothers," Adv. Technol. Dev. 2, pp.1-8, 1993.
- [144] J.H. Holland, "Genetic algorithms,"" Sci. Am., pp. 66-72, 1992.
- [145] M. Çunkaş, Genetik Algoritmalar ve Uygulamaları, Ders Notları, Selçuk Ünivesitesi, 2006.
- [146] A.A. Adewuya, New methods in Genetic Search with Real Valued Chromosomes, Master's Thesis, Cambridge:Massachusetts Institute of Technology, 1996.
- [147] Z. Michalewicz, Genetic Algorithms + Data structures = Evolution Programs, 2nd ed., New York: Springer-Verlag, 1994.

- [148] N.J. Radcliff, "Forma Analysis and Random Respectful Recombination." In Proc. of Fourth International Conference on Genetic Algorithms, San Mateo, CA: Morgan Kauffman., 1991.
- [149] L. Davis, "Hybridization and Numerical Representation," in L.Davis(Ed.), The Handbook of Genetic Algorithms, New York: Van Nostrand Reinhold, pp. 61-71, 1991.
- [150] A. Wright, "Genetic Algorithms for Real Parameter Optimization," in G.J.E. Rawlins (Ed.), Foundations of Genetic Algorithms 2, San Mateo, CA: Morgan Kaufman, pp.205-218., 1991.
- [151] L.J. Eshelman, D.J. Shafter, "Real-Coded Genetic Algorithms and Interval-Schemata", in D.L. Whitley (Ed.), Foundations of Genetic Algorithms 2, San Mateo, CA: Morgan Kaufman, pp.187-20
- [152] D.A. Pierre, "Optimization" in McGraw-Hill Encyclopedia of Science and Technology 12, New York: McGraw-Hill, pp.476-482, 1992.
- [153] Wurtz F., Richomme M., Bignon J., Sabonnadiere J.C, 1997, "A Few Results for Using genetic algorithms in Design of Electrical Machines", IEEE Transactions on Magnetics, Vol.33, No.2