



**FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ
MÜHENDİSLİK VE FEN BİLİMLERİ ENSTİTÜSÜ**

**GENETİK ALGORİTMA İLE MERKEZİ SINAVLARDA TEK VE ÇOK
BOYUTLU YAKINLIĞA GÖRE EN İYİ OTURUM PLANININ
OLUŞTURULMASI**

**YÜKSEK LİSANS TEZİ
Muhammed Fatih AĞALDAY**

Anabilim Dalı: Bilgisayar Mühendisliği

HAZİRAN 2018



**FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ
MÜHENDİSLİK VE FEN BİLİMLERİ ENSTİTÜSÜ**

**GENETİK ALGORİTMA İLE MERKEZİ SINAVLARDA TEK VE ÇOK
BOYUTLU YAKINLIĞA GÖRE EN İYİ OTURUM PLANININ
OLUŞTURULMASI**

**YÜKSEK LİSANS TEZİ
Muhammed Fatih AĞALDAY
(150221007)**

Anabilim Dalı: Bilgisayar Mühendisliği

Tez Danışmanı: Dr. Öğr. Üyesi Ali NİZAM

HAZİRAN 2018

TEZ ONAYI

FSMVÜ, Mühendislik ve Fen Bilimleri Enstitüsü'nün 150221007 numaralı Yüksek Lisans Öğrencisi **Muhammed Fatih AĞALDAY**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “**GENETİK ALGORİTMA İLE MERKEZİ SINAVLARDA TEK VE ÇOK BOYUTLU YAKINLIĞA GÖRE EN İYİ OTURUM PLANININ OLUŞTURULMASI**” başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile 11.06.2018 tarihinde savunmuş ve mezuniyeti hususunda enstitü için gerekli yeterlilikleri yerine getirmiştir.

Prof. Dr. M. Bülent ULUENGİN
Mühendislik ve Fen Bilimler Enstitüsü
Müdür

Tez Danışmanı: **Dr. Öğr. Üyesi Ali NİZAM**

Fatih Sultan Mehmet Vakıf Üniversitesi



Jüri Üyeleri: **Prof. Dr. Ali Yılmaz ÇAMURCU**

Fatih Sultan Mehmet Vakıf Üniversitesi



Dr. Öğr. Üyesi Buket DOĞAN

Marmara Üniversitesi



Savunma Tarihi: 11 Haziran 2018

ÖNSÖZ

Yapılan bu tez çalışması ile merkezi sınavların güvenilirliğini arttırmak için sınava giren bireylerden birbirini tanıyan kişilerin yan yana veya arka arkaya oturmalarını engelleyen optimizasyon işlemi genetik algoritma kullanılarak sağlanmıştır.

Araştırma konusunun belirlenmesi ve planlamasında, tez süresinin her aşamasında bilimsel katkısını, desteğini ve yardımlarını esirgemeyen, içtenliği ve samimi tavırlarıyla bana yol gösteren değerli danışman hocam Sayın Dr. Öğr. Üyesi Ali Nizam'a en derin saygı ve şükranlarımı sunarım. Başta Bilgisayar Mühendisliği Bölüm başkanı Prof. Dr. Ali Yılmaz ÇAMURCU olmak üzere diğer Fatih Sultan Mehmet Vakıf Üniversitesinin değerli hocalarına verdikleri kıymetli bilgiler için teşekkür ediyorum. Ayrıca tez yazımı boyunca bana desteklerini sunan, motive eden ve her zaman yanımda olan hayat arkadaşım Zeynep AĞALDAY' a çok teşekkür ederim.

Muhammed Fatih AĞALDAY

İÇİNDEKİLER

ÖNSÖZ	iii
İÇİNDEKİLER	iv
ŞEKİLLER LİSTESİ	vi
TABLO LİSTESİ	viii
KISALTMALAR	ix
ÖZET	x
SUMMARY	xi
1. GİRİŞ	1
1.1 Literatür Araştırması	2
1.2 Geliştirme Altyapıları	5
2. ÖLÇME ve DEĞERLENDİRME	7
2.1 Ölçme	7
2.2 Değerlendirme	8
2.3 Güvenirlik	9
3. OPTİMİZASYON TEKNİKLERİ	10
3.1 Optimizasyon Sınıfları	12
3.1.1 Deneme yanılma yolu ile optimizasyon	12
3.1.2 Tek ve çok parametrelili optimizasyon	12
3.1.3 Statik ve dinamik optimizasyon	13
3.1.4 Sürekli ve ayrık parametrelili optimizasyon	13
3.1.5 Sınırlı ve sınırsız optimizasyon	13
3.1.6 Rasgele ve minimum araştırma algoritmaları	13
3.2 Meta Sezgisel Algoritmalar	15
3.3 Yapay Zekâ	16
3.3.1 Evrimsel Hesaplama	16
4. GENETİK ALGORİTMA	17
4.1 Genetik Algoritmanın Tarihçesi	17
4.2 Genetik Algoritmalar	17
4.2.1 Rasgele Araştırma Algoritması	22
4.2.2 İkili kodlu genetik algoritmalar	23
4.2.3 Gerçek Kodlama	24
4.2.4 Değer Kodlama	25

4.3 Genetik Algoritma Yapısı	25
4.3.1 Basit Bir Genetik Algoritma ve Yapısı.....	26
4.3.2 Genel bir Genetik Algoritmanın adımları	27
4.4 Başlangıç Popülasyonun Oluşturulması	28
4.5 Amaç fonksiyonu ve parametreler.....	29
4.6 Uygunluk değeri ve hesaplanması.....	30
4.7 Doğal seçim	31
4.8 Kodlama (Kromozomların temsili)	32
4.9 Genetik Algoritmada Kullanılan Operatörler	33
4.9.1 Çaprazlama	33
4.10 Mutasyon	38
4.11 Rulet çemberi seçimi	39
4.12 Turnuva Seçimi	41
4.13 Elitizm	41
4.14 Genetik Algoritmaların Avantajları ve Dezavantajları.....	41
4.14.1 Avantajları	41
4.14.2 Dezavantajları	42
4.15 Genetik algoritmanın kullanıldığı bazı özel uygulama alanları	42
5. MERKEZİ SINAVLARDA OTURUM DÜZENLERİNİN GENETİK ALGORİTMA İLE OPTİMİZASYONU.....	43
5.1 Genetik Algoritmaya Dayalı Tek Boyutlu Sınav Yerleştirme Modeli.....	43
5.1.1 Kullanılan Veriler	44
5.1.2 Algoritma çalışma adımları	45
5.2 Tek Boyutlu Sınav Yerleştirme için İyileştirilmiş Genetik Algoritma Modeli	49
5.2.1 İyileştirilmiş Tek Boyutlu Yöntem	55
5.2.2 İyileştirilmiş Tek Boyutlu Yöntemde Karşılaştırmalı Sonuçlar	56
5.3 Çok Boyutlu Sınav Yerleştirme için Katsayı Belirlemeye Dayalı Genetik Algoritma Modeli	61
6. SONUÇ VE ÖNERİLER	63
7. KAYNAKLAR	68
8. ÖZGEÇMİŞ	72

ŞEKİLLER LİSTESİ

Şekil 3.1 Optimizasyon eğrisi.....	11
Şekil 3.2 Optimizasyon işlemlerinin temel aşamaları.....	12
Şekil 3.3 Optimizasyon algoritmaları kategorileri.....	14
Şekil 4.1 İkili kodun biyolojik yaşam ile genetik algoritmalar arasındaki benzetim	19
Şekil 4.2 Genetik algoritmaların çalışma prensibi.....	21
Şekil 4.3 İkili kodlu genetik algoritma.....	23
Şekil 4.4 Gerçek kodlu genetik algoritma akış diyagramı.....	24
Şekil 4.5 Gerçek kodlama.....	25
Şekil 4.6 Değer kodlama.....	25
Şekil 4.7 Genetik algoritmanın popülasyon yapısı.....	26
Şekil 4.8 Genetik algoritma akış diyagramı.....	27
Şekil 4.9 Gen Yapısı parametrelerin kodlanması.....	32
Şekil 4.10 Kromozomların temsili gösterimi.....	32
Şekil 4.11 Genetik algoritmalarda çaprazlama.....	33
Şekil 4.12 İkili kodlu genetik algoritmalarda tek noktalı çaprazlama.....	34
Şekil 4.13 Tek noktalı çaprazlama sonucu oluşan kromozomlar.....	34
Şekil 4.14 Gerçek Kodlu Genetik Algoritmalarda Tek noktalı Çaprazlama.....	34
Şekil 4.15 Genetik Algoritmalarda Çift Noktalı Çaprazlama.....	35
Şekil 4.16 İki nokta çaprazlama sonucu oluşan kromozomlar.....	35
Şekil 4.17 Genetik Algoritmalarda Çok Noktalı Çaprazlama.....	36
Şekil 4.18 Üç nokta çaprazlama sonucu oluşan kromozomlar.....	36
Şekil 4.19 Üniorm çaprazlama sonucu oluşan kromozomlar.....	37
Şekil 4.20 Mutasyon işlemi sonucu oluşan kromozomlar.....	38
Şekil 4.21 Genetik Algoritmalarda doğal Seçim ve Genetik Operatörler.....	39
Şekil 4.22 Rulet Tekerleği ve bireylerin yüzde oranları.....	40
Şekil 4.23 Rulet çemberi seçimi.....	40
Şekil 5.1 Sınıf oturum düzeni.....	43
Şekil 5.2 Klasik tek noktalı çaprazlamaya dayalı modelin Gen yapısı.....	43

Şekil 5.3 Klasik tek noktali çaprazlamaya dayali modelin Kromozom yapisi	44
Şekil 5.4 Klasik tek noktali çaprazlamaya dayali modelin kromozom 1 yapisi	46
Şekil 5.5 Klasik tek noktali çaprazlamaya dayali modelde C1 – C2 sinif düzeni	46
Şekil 5.6 Klasik tek noktali çaprazlamaya dayali modelde kromozomların gösterilişi	47
Şekil 5.7 Klasik tek noktali çaprazlamaya dayali modelde çaprazlama işleminin	48
Şekil 5.8 İyileştirilmiş Tek Boyutlu Modelin Gen Yapisi	50
Şekil 5.9 İyileştirilmiş tek boyutlu yöntemin kromozom yapisi.....	50
Şekil 5.10 İyileştirilmiş tek boyutlu modelin C1- C2 Sınıf düzenleri	51
Şekil 5.11 Tek Boyutlu Yöntemde kromozom ceza puanı belirlemede kullanılan öğrenci yakınlıkları	52
Şekil 5.12 Tek ve Çok boyutlu yöntemlerin akış diyagramı	54
Şekil 5.13 İyileştirilmiş tek boyutlu yöntemde kısmi eşleşmiş çaprazlama	55
Şekil 5.14 İyileştirilmiş tek boyutlu modelde ceza puanlarının parametrelere göre değişim grafiği	59

TABLO LİSTESİ

Tablo 3.1 Optimizasyon tekniklerine örnekler	15
Tablo 5.1 Genetik Algoritma Parametrelerin Analiz Sonuçlarına Etkisi	60
Tablo 5.2 Klasik Yöntem Hatalı Gen Analiz Sonuçları	49
Tablo 6.1 İyileştirilmiş Tek Boyutlu Yöntem Ceza Puanları Analiz Sonuçları	56
Tablo 6.2 Bulanık Mantığa Dayalı Çok Boyutlu Yöntem Ceza Puanları Analiz Sonuçları	62
Tablo 6.3 İyileştirilmiş Tek Boyutlu Yöntem ile Bulanık Mantığa Dayalı Çok Boyutlu Yöntemin Oransal Olarak Karşılaştırılması	64
Tablo 6.4 İyileştirilmiş Tek Boyutlu Yöntem ile Bulanık Mantığa Dayalı Çok Boyutlu Yöntemin Ceza Puanlarının Karşılaştırılması	65
Tablo 6.5 İyileştirilmiş Tek Boyutlu Yöntem Sonuçlarının Bulanık Mantığa Dayalı Çok Boyutlu Yönteme Göre Karşılaştırılması	66

KISALTMALAR

ÇN1 : Çaprazlama Noktası 1

ÇN2 : Çaprazlama Noktası 2

B1 : Birey 1

B2 : Birey 2

YB1 : Yavru Birey 1

YB2 : Yavru Birey 2

ÖSYM : ÖLÇME, SEÇME VE YERLEŞTİRME MERKEZİ BAŞKANLIĞI

ÖZET

Aynı okuldan, aynı sınıftan veya aynı mahalleden öğrencilerin yapılan oturma düzenlerinde yan yana veya arka arkaya gelmesi istenmeyen yardımlaşmalara ve kopya çekilmesine sebep olabilmektedir. Öğrencilerin sınıf ve salonlara en uygun şekilde yerleştirilmesi amacıyla çeşitli çalışmalar yapılmıştır. Ancak yapılan çalışmalar sadece sınıf düzeyinde kalmış olup sınırlı bir öğrenci gurubunun sınıf içi oturma düzenlerinin öğrencinin başarısı üzerine etkileri gibi konulara odaklanmıştır. Yüksek Lisans tez çalışması ile sınavların güvenilirliğini arttırmak amacıyla, birbirini tanıyan veya herhangi bir biçimde ilişkili öğrencilerin salonlara arka arkaya veya yan yana gelmeyecek şekilde yerleştirilmesi için Genetik Algoritma yöntemi kullanımı hedeflenmiştir. Genetik algoritmalar karmaşık düzeydeki problemlerin çözümü için kullanılan sezgisel yöntemlerinden biridir. Doğal yaşam örnek alınarak iyi nesillerin yaşatılması kötü nesillerin yok edilmesi üzerine çalışan bir algoritmadır. Belirleyeceğimiz uygunluk fonksiyonu sayesinde en iyi yapıyı koruyup bir sonraki nesle taşıyarak en iyi oturma düzeni sağlanır. Yapılan çalışmada Artuklu Üniversitesi Uzaktan Eğitim Uygulama ve Araştırma Merkezi sınav verileri ilgili birimlerden izin alınarak tez çalışmasında kullanılmıştır.

SUMMARY

Side-by-side or consecutively sitting of students from the same class or from the same neighborhood may result in unwanted communications or cheating. Various studies conducted to determine the most optimum seating arrangement but these studies focused on issues like the impact of seating arrangement on student success. These various studies mostly focused on physical classroom seating arrangement. The Aim of the following thesis is to prevent students that are related to each other in a way to sit next to or consecutively during the exams by using genetic algorithm. The reliability of the exams will also be increased with M.Sc. study. Genetic algorithms are one of the many heuristic algorithms used to solve complex problems. Genetic algorithms imitate natural selection by eliminating unfit generations so that the fitter generations sustain. The fitness function to be determined will ensure that each session is the better version of its predecessor and the final session is the most optimum session. Data acquired from Artuklu University Distance Learning Center examination has been used for the thesis study with permission from the related unit.

1. GİRİŞ

Eğitimde öğrencinin hedef davranışların kazanılıp kazanılmadığını ölçmenin farklı yolları vardır. En çok kullanılan yöntemlerden biri sınavlardır. Ölçme ve değerlendirme için yazılı sınavlar, sözlü sınavlar ve çoktan seçmeli sınavlar tercih edilmektedir. Yazılı sınavlar öğrencinin özgün ve yaratıcı düşünme gücünü, yazılı anlatım becerisini, bir konu hakkındaki görüşünü, ilgisini, tutumunu ve düşüncelerini ölçmek için yapılan pratik sınavlardan biridir (Tekin, 2004). Genellikle yazılı sınavların tercih sebebi küçük bir grup için ölçme ve değerlendirme yapmaktır. Kalabalık gruplarda ise yazılı sınavlar tercih edilmekle birlikte ölçme ve değerlendirme kısmı uzun sürdüğü için çoktan seçmeli sınavlar daha çok tercih edilebilmektedir. Binlerce öğrencinin katıldığı sınavları merkezi bir sınav ile yapmak ölçme ve değerlendirme açısından oldukça daha az zaman almaktadır.

Ülkemiz genelinde binlerce kişinin katıldığı çok fazla merkezi sınavlar mevcuttur. ÖSYM tarafından her sene YGS, YDS, ALES, KPSS, ÜDS ve TUS gibi birçok sınav yapılmaktadır. Ayrıca Anadolu üniversitesi açık öğretim sınavları ile bazı bakanlıkların yaptığı birçok merkezi sınavlar vardır. Son zamanlarda Açık öğretim fakültesi ile beraber birçok üniversite ön lisans ve lisans düzeyinde bölümler açarak merkezi sınavlar yapmaktadır.

Bu sınavlarda rastgele yapılan oturma düzenlerinde birbirini tanıyan kişilerin yan yana veya arka arkaya oturması ihtimal dâhilindedir. Bu oturma düzeninde öğrencilerin sınavda yardımlaşması muhtemeldir. Bu durum sınavların ölçme ve değerlendirme açısından güvenilirliğini kaybetmesine yol açar. Bu tezde amacımız birbirini tanıma ihtimali olan öğrencilerin aynı salonda yan yana veya arka arkaya oturmayacak şekilde en iyi sınav oturum düzeninin sağlanmasıdır. Kullandığımız yazılımda genetik algoritma metodu sayesinde birbirlerini tanıyan kişilerin yan yana veya arka arkaya oturmasını engellemektedir. Bu önerdiğimiz model sayesinde, birbirini tanıyan kişilerin sınav salonuna birbirlerine uzak yerleştirilmesi sağlanmış olacak ve daha büyük kapasiteye sahip birimlerde yapılan merkezi sınavların ölçme ve değerlendirme güvenilirliği artacaktır.

Bu tez çalışmasında amaç, merkezi yapılan sınavlarda kopya çekme ihtimalini mümkün olduğu kadar minimize etmektir. Yazılım içerisinde kullandığımız Genetik Algoritma sayesinde birbirini tanıma ihtimali olan kişilerin birbirlerine yakın oturmaları önlemektedir. Sınava girenlerin birbirini tanıma olasılığı farklı durumlar

dikkate alınarak hesaplanabilir. Öncelikle sınava giren kişiler aynı bölümden ya da okuldan mezun olabilir, bu kişiler akraba veya aynı semtte yaşıyor da olabilirler. Bu durum sınav ortamında muhtemel yardımlaşmalara yol açabilir ve sınavın ölçme ve değerlendirme güvenilirliğini azaltır. Yukarıda bahsedilen benzer özelliklere sahip bireyler yazılım sayesinde birbirlerinden uzak yerlerde oturmaları hızlı bir şekilde sağlanacaktır. Bu sayede sınava giren öğrencilerin birbirlerinden muhtemel yardımlaşma ihtimalini ortadan kaldıracaktır. Geçmiş dönemlerde yaşanan binlerce kişinin katıldığı merkezi sınavların iptal edilmelerinin önüne geçilmiş olacaktır.

Mevcut sınavlarda sınav salonlarına rastgele yerleştirilen kişilerin birbirleri ile yakınlıkları tespit edilmek istense, her salonda sınava giren kişilerin tek tek aralarındaki yakınlıkları kontrol edilmelidir. Bu inceleme tam kapsamlı arama yöntemi ile yapılırsa oldukça fazla işlem süresi gerekecektir. Her salon için tek tek kişiler arasındaki yakınlığı bulmanız gerekecek ve muhtemel yakınlık bulması durumunda bu kişiler başka salonlara alındığında başkaları ile yakın oturma ihtimali olduğundan sorgulama aynı şekilde tekrarlayacaktır. Tekrarlanan bu durum özyinelemeli bir şekilde devam edecektir ve oldukça fazla işlem süresine ihtiyaç duyacaktır. Önerdiğimiz model sayesinde büyük kitlelere yapılacak merkezi sınavlarda istenilen en iyi oturma düzenini sağlayacak optimizasyonu kısa bir sürede sağlanmış olacaktır. Teknolojinin özellikle de bilgisayar alanındaki hızlı gelişmeler optimizasyon alanındaki araştırmalara yöneltmiştir. Günümüzde araştırmacılar, optimizasyonları gerçekleştirmek ve optimal veya optimale yakın çözümler aramak için giderek daha fazla bilgi işlem gücüne ihtiyaç duymaktadırlar.

1.1 Literatür Araştırması

Sınavlarda katılımcılar için en iyi oturma düzenini sağlanması üzerine yapılmış literatürde çok fazla çalışma tespit edilememiştir. Yapılan çalışmalardan birisi Faruk Bulut ve Şeyda Subaşı'nın bir makalesidir. (Bulut ve Subaşı, 2015). Bu makalede bir model önerilmiş fakat makale incelendiğinde bazı hususların gözden kaçtığı tespit edilmiştir. Makalede Genetik Algoritma ile kromozomlar çaprazlama işlemine tabi tutulurken yavru bireylerde genlerin çaprazlama esnasında tekrarladığı tespit edilmiştir. Literatürde sınıflardaki belirli bir öğrenci grubunun en uygun sınıf içi oturma düzeni ile ilgili bazı çalışmalar vardır. Fakat sınav salonu düzeni oluşturmaktan çok öğrencilerin ders başarısını artırma üzerine çalışmalar yapılmıştır. Literatürde

ayrıca üniversitelerde öğrencilerin birden fazla derse ait sınavların çakışma olmadan yapılması üzerine önerilen farklı metotlar bulunmaktadır.

Shin-ike ve Iima sınıf oturma düzenine ait ilk çalışmayı yapmışlardır (Shin-ike ve Iima,2011). Çalışmalarında genetik algoritma yöntemi ile bir sınıftaki öğrencilerin sınıf sıralarına oturma düzenleri ile ilgi bir model önermişlerdir. Sınıf oturma düzenlerinin rastgele veya boy sırasına göre düzenlenmesi öğrencilerin başarısına olumsuz yönde etkisi olduğu ortaya koyulmuş ve öğrencilerin başarılarının artırılması ve aynı zamanda sınıf başarısının artırılması üzerine bir yaklaşım öne sürülmüştür. Sınıf oturma düzenleri belirlemek için öğrencilere yapılan anket çalışmasında görme ve işitme duyuları, sınıf arkadaş ilişkileri, bireysel özellikler, hafıza ve zekâ durumları üzerine bazı sorular sorulmuştur. Ankette cevaplanan sorular dikkate alınarak yerleşim sağlayan genetik algoritma tabanlı çalışmalar Japonya'daki devlet okullarında denenmiş ve başarılı sonuçlar alınmıştır.

Tez çalışması ile benzerlikleri olan bir konuda zaman çizelgeleme konusudur. Konusu itibari ile belli kıstasları bulunan zaman çizelgelemede aynı zaman dilimine birden fazla ders ataması yapılmaması gerekmektedir. Aynı şekilde Tez konusu olan çalışmada da ifade edilen oturma düzeni için aynı sıraya farklı iki kişinin oturması istenmeyen bir durumdur. Zaman çizelgesi günümüz akademik ortamda popüler bir konu haline almıştır. Çizelgeleme problemi araştırmacıların dikkatini uzun bir süredir çekmiş ve çizelgeleme hakkında oldukça araştırma yapılmıştır. Çizelgeleme problemi üzerine 1960'lı yıllardan itibaren birçok araştırmacı tarafından çalışılmaktadır. Zaman çizelgelemenin otomatikleştirme konusunda ilk çalışmalardan birisi 1963 Gotlieb tarafından yapılmıştır. Bu zamana kadar konu ile ilgili birçok konferans, bildiriler ve dergiler yayınlandı ve birçok yazılım geliştirildi. Zaman çizelgesi problemleri arasında en çok ders çizelgeleme ve sınav çizelgeleme problemleri incelenmektedir. Eğitim kurumlarında yaşanan problemi çözmek için çok sayıda çalışma yapılmış ve günümüzde araştırmacılar bu probleme farklı yöntemler geliştirmişlerdir. Eğitimle ilgili zaman çizelgelerinin yapıları birbirlerinden farklı olduğundan, çözüm yöntemleri de farklılaşmakta ve problemin büyüklüğüne bağlı olarak yöntemlerde farklılıklar olmaktadır. Eğitim Zaman Çizelgeleme Problemi üzerine etkili çalışmalar 1960-1970 yılları arasında yapılmış ve 1970'li yılların sonlarına doğru konuya olan ilgi düşerken 1980 sonrası tekrar ilgi artmaktadır. 1995 yılında 60 çalışma ile kısa adı PATAT (Practice and Theory of Automated Timetabling) olan Uluslararası Otomatik Zaman

Çizelgeleme Uygulaması ve Teorisi Konferansı düzenlenerek probleme olan ilgi zirve yapmıştır (Bardadym,1996; Burke ve Ross, 1996). Zaman çizelgeleme planlamasının günlük hayatta uygulama alanı oldukça fazladır.

Gerçek hayattaki birçok problemi zaman çizelgesi olarak ifade edilebilir. Birçok araştırmacı bu tür çizelgeleme problemini daha iyi bir sonuca ulaşmak için uğraşmaktadır. Zaman çizelgeleme problemleri deterministik olmayan polinomal NP problem olarak kabul edilmektedir. Zaman Çizelgeleme üretim ve hizmet endüstrisinde önemli bir karar verme süreci olup ulaşım, makine, sağlık hizmetleri, eğitim ve spor gibi farklı alanlarda görmekteyiz. Metronun gidiş geliş saatleri, uçak uçuş programları, şehirlerarası taşımacılık, otobüs tren seferlerinin planlanması, sağlık personellerin nöbetleri, spor müsabakalarının hazırlanması, üniversitelerde ders ve sınav programlarının hazırlanması, fabrikadaki işçi vardiyaları, televizyon programlarının yayın akışları için çizelgeleme kullanılmaktadır.

Üniversitelerde sınavların sınav salonlarına en uygun şekilde atanması üzerine benzer çalışmalar yapılmıştır. Sınavların düzenlenmesi ile ilgili zaman tablosu oluşturulması üzerine Genetik Algoritma tabanlı bir model öneren bir çalışma anlatılmıştır (Ross ve diğ, 2006). Bir öğrencinin aynı anda birden fazla dersin sınavına girme ihtimali yoktur. Bu durum dikkate alınarak en kısa sürede sınavların hepsinin yapılabilmesi için farklı yöntemler denenmiş ve bazı çözüm önerileri getirmişlerdir.

Zaman çizelgeleme üzerine Qu ve arkadaşları yaptıkları bir çalışmada literatürdeki yöntemleri kısaca anlatarak aralarındaki farkları göstermişlerdir (Qu ve diğ, 2009). Zamanlama problemini yedi farklı arama tekniğini kullanarak üç farklı üniversitedeki veri kümeleri üzerinde inceleyerek karşılaştırmışlardır.

Deris ve arkadaşları haftalık ders çizelgesinin oluşturulmasında kısıt tabanlı muhakeme tekniği kullanmışlar ve kısıt tatmin modeli olarak formüle edilen Üniversite Ders Zaman Çizelgeme Probleminin çözümü için bir algoritma önermişlerdir. Önerilen çizelge Malezya'daki üniversitelerin genel yapılarını yansıtacak şekilde ve özel durumları dikkate almaktadır (Deris ve diğ, 2000).

Yiğit, Meslek Liselerinde kullanılacak uygun ders çizelgelerinin hazırlanabilmesi için genetik algoritmaları kullanmıştır. Meslek liseleri için hazırladığı haftalık ders çizelgeleme problemi ile ilgili çalışmasında bazı zorunlu kısıtlar tanımlamıştır. Öğretim elemanının vermiş olduğu ders saatlerinin kesişmemesi, farklı şubelerin aynı derslikte ve aynı saate ders yapılmaması, hocanın belirlemiş olduğu bir günün veya

ders saatinin boş bırakılması, laboratuvar ve atölye gibi ihtiyaç duyulan derslerin uygun dersliklere atanması keskin kısıtlar olarak gösterilebilir. Çözüm yöntemi için genetik algoritmayı tercih etmiştir. Amaç fonksiyonu bir uygunluk değeri belirlenmiştir. Uygunluk değeri hesaplanırken kısıtları bozan atamalar için cezalandırma yöntemi kullanılmıştır. Uygunluk değerinin buradaki amacı cezayı minimize etmektir. Ders çizelgeleme için C++ programlama dilini kullanarak geliştirilen ara yüze veriler girilerek sonuçlar bir rapor şeklinde alınır. Yapılan deneysel çalışmalara göre geliştirilen algoritma yönteminin uygun sonuçlar verdiğini ifade etmiştir. Bu çalışma ile uygun ders çizelgeleri elde edilebileceği ispatlanmıştır (Yiğit, 2006).

MirHassani, Shahrood Teknoloji Üniversitesinde karşılaşılan Ders Çizelgeleme Problemini çözmek için ikili programlama yapısı üzerine bir model önermiştir. Amaç uygunluk fonksiyonunun elde edilen toplam cezanın minimizasyonunu sağlamaktır. Önerilen zaman çizelgeleme metodunun ilgili üniversitenin kullandığı metottan dahi iyi olduğunu ve değerlerin hesaplanma süresi önemli ölçüde değiştirebileceğini belirtmiştir (MirHassani, 2006).

Tezde önerilen model sayesinde binlerce kişinin katıldığı merkezi sınavlarda, aynı salonda birbirini tanıyan kişilerin yan yana veya arka arkaya oturmalarını engelleyecek şekilde yerleştirilmesi sağlanmaktadır. Bu problem optimizasyon problemi olarak ele alınmıştır. Türkiye Geneli yapılan birçok sınavda sınav güvenilirliğini artıracak olan bu model sayesinde tüm öğrencilerin aynı anda tek oturumla sınav olmalarına olanak sağlayacaktır.

Tez çalışması 5 bölümden oluşmaktadır. Diğer bölümde merkezi sınavlarda ölçme ve değerlendirme ile ilgili bilgiler mevcuttur. Ölçme ve değerlendirme konusunun güvenilirliği üzerinde durulmuştur. Üçüncü bölümde optimizasyon modelleri Dördüncü bölümde tez çalışmasında kullanılan algoritma modeli ve önerilen yöntemler anlatılmıştır. Beşinci bölümde yazılım modelinin araştırma bulguları ile sonraki çalışmaların neler olabileceği konusundaki öneriler ve son bölümde tez çalışması ile ilgili değerlendirmeler yer almaktadır.

1.2 Geliştirme Altyapıları

Tez çalışmasını hazırlamak için Python programlama dili tercih edilmiştir. Bunun en önemli nedeni hesaplamalı bilimin uygulama ve öğretimi noktasında mükemmelle

yakın bir dile sahip olmasıdır. Python Guido Van Rossum tarafından 1991 yılında geliştirilen yüksek seviyeli ve nesne yönelimli bir programlama dilidir. Python platformdan bağımsız bir programlama dil olmasından dolayı Windows, Linux, Mac OS X, Android ve Macintosh gibi tüm platform ve işletim sistemleri üzerinde çalışabilmektedir. Python programlama dilinin basit ve anlaşılır söz dizimine sahip olması birçok programcı ve özellikle Google tarafından tercih edilen bir dil haline getirmiştir. Ruby ve Perl gibi programlama dillerinin önüne geçmiştir. Python dilinin söz diziliminin basit ve anlaşılır olması sayesinde hem program kodunu yazmak, hem de başkası tarafından yazılmış bir program kodunu okumak, diğer dillere göre daha kolaydır (Özgül, 2013).

Python kullanımı açık kaynaklı olduğu için bir şirkete bağlı değildir. Python dili için herhangi bir ücret ödenmez. Çok karmaşık işlemleri basit komutlarla yapılabilir. Basit işlemler için sınıf oluşturmak zorunluluğu yoktur. Genel kullanım alanı gittikçe genişliyor ve oldukça seviliyor. Bu sayede sürekli geliştiriliyor. Yazılımı öğrenmek için kaynak bulmak oldukça kolaydır. Bilimsel araştırmalarda yaygın olarak kullanılmaktadır. Farklı bilim dallarında kullanılmak üzere belli bir çalışma için özel hazırlanmış kütüphaneleri vardır. Bu kütüphaneler sadece python üzerine kendini yetiştirmiş profesyonel yazılım uzmanları tarafından hazırlanmaktadır. Python her ne kadar mükemmel yakın olan bir dil olsa da dinamik ve yorumlanan bir dil olması sebebiyle çok hızlı çalışmamaktadır. Çok yoğun hesaplama gerektiren uygulamalar için C veya Fortran'la yazılmış programlar tercih edilmelidir. Python bazı ara çözümler ile C programlama dili ile yazılmış bir programı python koduna entegre edilebilmektedir. Bilimsel hesaplamalar için geliştirilen SciPy Kütüphanesi buna örnek verilebilir. Birçok durumda C programlamaya göre yavaş çalışmasına rağmen C programlama ile bir haftada yazılacak programı Python ile bir günde yazmanız mümkün olabilmektedir. Python programlama dilini birçok ünlü firma tarafından kendi uygulamalarında tercih ettiğini görmekteyiz. Google arama motorları Python programını kullanmaktadır. Video paylaşım sitesi olan YouTube python ile yazılmıştır. Çok kullanılan BitTorrent paylaşım sitesi Python ile geliştirilmektedir. Animasyon Film Endüstrisi python programlama tercih etmektedir. Ve yapılan incelemede birçok uygulamanın python tarafından yazıldığını görmekteyiz.

2. ÖLÇME VE DEĞERLENDİRME

Ölçme ve değerlendirme eğitim sürecinin sürekli birlikte söylendiği iki kavramdır. Eğitim sürecinin çıktılarının doğru ölçülmesi ve değerlendirilmesi çok önemlidir. Ölçme ve değerlendirmenin doğru yapılabilmesi her şeyden önce sınavların güvenilirliğine ve sağlıklı yapılabilmesine bağlıdır.

2.1 Ölçme

Ölçme kavramının tanımlarına bakıldığında çok geniş bir yelpazesi olduğunu görürüz. Ölçme işlemi tanım olarak “Ölçme, bir niteliğin gözlenip gözlem sonucunun sayılarla veya başka sembollerle gösterilmesidir.” (Turgut, 1992) şeklinde tanım yapılabilir. Eğitimde ölçme, bireyin bilgi ve becerilere sahip olma derecesini sayı ve sembollerle ortaya çıkarma işidir. Ölçülmek istenen özelliklerin bazıları doğrudan bazıları ise dolaylı gözlenebilmektedir.

Araştırma yapmak aynı zamanda ölçme işlemi yapmaktır. Araştırma yapmak için ölçme kaçınılmazdır. Ölçme en genel anlamda olaylara ve nesnelere uygun sayılar vermektir. Ölçme nesne olaylar arasında farklı düzeylerde ayırt edebilme işlemi olarak söylenebilir. Bir olay veya nesne diğerlerinden ayırt edilebiliyorsa ölçülebilmiş demektir. Ölçme temelde bir tasvir olayıdır; belli bir objenin belli bir özelliğe sahip olup olmadığının gözlemlenerek sonuçlarının sembollerle, özellikle de sayılarla ifade edilmesidir (Tekin, 1977). Objenin ya da bireylerin belirli bir özelliğe sahip olma derecesi duruma veya objeye göre değişiklik arz edebilir. Ölçme farklılık ve değişkenlik kavramında odaklaşmaktadır. Bir şey var olandan farklıysa bir önem arz eder ve değişkenlik araştırılmaya değer görülür. Bundan dolayı araştırmada değişkenlik çok önemlidir. Bir öğrenci gurubuna aynı dersi öğretmede iki yaklaşım uygulanıyor olsun, fakat bu öğrencilerin başarıları arasında bir farklılık yoksa hangi yöntemin daha başarılı olduğunun bir anlamı olmaz. Fakat üçüncü bir yaklaşım öğretme ve öğrenme bakımından farklılık yaratacak derecede bir etkisi varsa bu yöntemin araştırmaya değer bir tarafı olur. Bir ölçme olayı analiz edildiğinde şunlar görülür:

$$\begin{array}{ccccccc}
\text{Gözlenen} & = & \text{Gerçek} & + & \text{Sistemli} & + & \text{Rasgele} \\
\text{Ölçüm} & & \text{Ölçüm} & & \text{Hata} & & \text{Hata} \\
& & (\text{Ölçülmek} & & (\text{Geçersizlik} & & (\text{Güvenilir} \\
& & \text{istenen} & & \text{de denir}) & & \text{Olmayan}) \\
& & \text{yapı}) & & & &
\end{array}$$

Yukarıdaki ifade edilmek istenen bir öğrenciye ait matematik testinin başarı puanı, gözlenen bir ölçümdür. Öğrencinin gerçek matematik bilgisi, yapının gerçek ölçümünü verir. Öğrencinin ilgisi, güdülenmesi gibi sistemli hataları bize gösterir. Öğrencinin soruları rastgele cevaplama yansıması yansız rasgele hatayı belirtir. Sonuçta sistemli ve sistemsiz random hatalar yüzünden gözlenen ölçümler daima gerçek yapıyı yansıtamazlar. Ölçmenin bu eleman ya da parçalarından gerçek ölçüm ile sistemli ölçümün-puanın sabit, yansız hatanın ise değişmesi beklenir (Balcı, 2010).

2.2 Değerlendirme

Değerlendirme, bir karar verme işlemidir, bu yönüyle ölçmeden ayrılır. Değerlendirme kavramı birçok ölçme veya gözlemler neticesinde varılan yargıyı belirten genel bir terimdir. Ölçme sonuçlarından zihinsel yargılarla öznel sonuçlar çıkarmadır. Eğitim ve öğretimde amaçlara ne derece yaklaşıldığının bir ölçüsünü vermektedir. (Binbasıoğlu, 1983). Ölçme işleminde, var olanın, olanaklar ölçüsünde aslına uygun olarak tasvirine çalışılır. Değerlendirmede ise ölçme sonuçları alınır; ölçütle karşılaştırılır ve ölçme sonucunun, ölçütle belirlenen koşulu karşılayıp karşılamadığına bakılır. Değerlendirme, ölçülen özelliğe ilişkin bir kararla sonuçlanır. Bu karar, ölçülen özelliğin belli bir amaçla işe yarayıp yaramadığını gösterir.

Ölçüt ölçme sonuçlarının karşılaştırıldığı, değerlendirme için değişmeyen değerler ve karar da karşılaştırmanın sonucudur. Ölçme işleminin gerçekleşmesi uygun bir ölçme aracının seçilmesine bağlıdır. Değerlendirme için ölçme işlemine ve ölçüte ihtiyacımız vardır. Çünkü değerlendirme “Ölçme sonuçlarını bir ölçüt (kriter veya referans çerçevesi) ile karşılaştırarak bir değer yargısına ulaşma sürecidir.” (Turgut, 2012).

Belli bir alanda kişilerin ne derecede bir öğrenme gücüne ya da düzeyine sahip oldukları ölçmeler yapılarak belirlenir. Ölçme eğitimde vazgeçilmez bir işlem olarak karşımıza çıkmaktadır. Öğrenci seçiminde, eğitim ve öğretim güçlüklerinin ortaya çıkarılmasında ve öğretimin değerlendirilmesinde, öğrencileri güdüleyip başarılarının

yükseltilmesinde, öğrencilerin rehberlikle ilgili sorunlarının çözümünde, öğrencilerin zeka seviyelerini belirlemede ölçme işleminden yararlanırız (Binbasıoğlu, 1983).

Kişilerin öğrenme düzeyleri ölçme araçları kullanılarak ölçülebilir ve sonuçta değerlendirme yapılır, yani bir yargıya varılır. Bir eğitim sisteminin başarısı sistemin felsefesine uygun değerlendirme yöntemleriyle paralellik teşkil etmektedir (Balcı ve Tekkaya, 2000). Ölçme işlemi ancak değerlendirme yapılarak anlam kazanır. Çünkü değerlendirmeler sonucunda öğrenciye davranışını nasıl değiştireceği veya geliştireceği hakkında yol gösterir, yeterince başarılı olan öğrenci güdülenir, gelecekteki öğrenimin planlanmasına ve meslek seçimine yardımcı olunur (Turgut, 1992).

Ölçme işleminin gerçekleşmesi uygun bir ölçme aracının seçilmesine bağlıdır. Bir ölçme aracının geliştirilmesinde bazı temel ilkeler vardır. İyi bir ölçme aracında Güvenirlik, Geçerlik, Kullanışlılık, Objektiflik, Ayırt edicilik, Örnekleyicilik olmalıdır.

2.3 Güvenirlik

Bir ölçümün rasgele hatadan arınmış olmasıdır. Eğitimde kullanılan ölçme araç ve yöntemlerin güvenilirliğini artırmak için alınabilecek bazı tedbirler bulunmaktadır. Bir sınavda soruların gayet anlaşılır ve kesinlikle cevaplandırılabilir olması gerekir ki, o sınavdan elde edilen puanın güvenilirliği artsın. Sınav sorularının sınava girecek öğrencilerin en az yarısı tarafından doğru cevaplandırılabilir güçlükte olmalıdır. Her sınav objektif yollarla puan verilebilmelidir. Ölçme sonuçlarının objektifliği arttıkça güvenilirliği artar. Sınav süresinin öğrencilerin tamamına yakınının tüm soruları cevaplandırmalarına yetecek uzunlukta olmalıdır. Objektif testlerle yapılan sınavlarda sürenin oldukça kısa verilmesi öğrencilerin testin sonundaki bazı soruları okumadan ve gelişi güzel cevaplandırmalarına sebep olur ve bu durumun güvenilirliği sıfıra yakındır.

3. OPTİMİZASYON TEKNİKLERİ

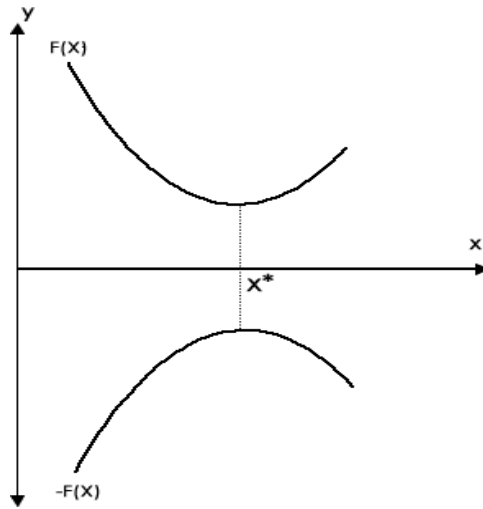
Optimum kelime anlamı ideal olan manasına gelen Latince bir kelimedir. Optimizasyon herhangi bir işlevin daha iyisini elde eden veya işlevin en iyi sonucunu üretmeyi gerçekleştiren süreçtir. Optimizasyon gerçek hayattaki problemin modellenerek ve matematiksel olarak tanımlanarak en iyi çözüm veya çözümlerini bulan işlemler bütünüdür. Optimizasyon bilim insanların ortaya koydukları yeni fikirlerinin geliştirilmesine yardımcı olur. Optimizasyon teorisi optimal değerlerin hesaplanmasında kullanılan metotlar, teknikler ve algoritmaların bütünüdür. Üzerinde çalışılan problemin veya fikrin en iyi sonuca ulaşması ya da daha iyi bir sonuca yaklaşması için probleme ait verilerin değiştirilmesi işlemi optimizasyon sayesinde gerçekleşir. Belirli sınırlamaları sağlayarak matematiksel kurallara dayalı algoritmalarla optimizasyon probleminin çözümü mümkün olmaktadır. Optimizasyonun işlevi probleme ait birçok çözüm arasından en iyi çözümü bulmaktır. Optimizasyon işlemi gerçekleştirmek için kullanılan verileri sürekli olarak değiştirilmesi gerekmektedir. Elde edilen çözümlerin değerlendirilmesi optimizasyon ile gerçekleşir. Bu işlemler ancak bir bilgisayar ortamında yapılırsa optimizasyon kısa sürede gerçekleşir ve mükemmel sonuçlara ulaşır (Çunkaş, 2006).

Optimizasyon belirtilen kısıtlamalar altında en iyi çözümü arar. Bu yaklaşım göreceli bir durum olup her problemin en iyi sonucunu bulamayabilir bazı problemlerin sonucu maksimum ya da minimum olarak verilebilir. Bu bakış açısına göre optimum kavramı her iki durumu kapsamaktadır. Bu nedenlerden dolayı sonuç göreceli olmaktadır. Kullanılan metotlar optimizasyon problemlerinin çözümünde her zaman etkili olmayabilir ve istenen çözüme ulaşmayı garanti etmezler. Genel olarak makul bir hızda ve ölçüde en iyi çözüme ulaşmada başarılıdırlar. Optimizasyonun amacı, yapılacak bir işin, sermayenin, emeğin veya kullanılacak teknolojinin minimumda tutulması ya da elde edilecek olan kar oranının maksimum yapılmasıdır. Genel olarak optimizasyon, hedeflenenin maksimum ya da minimum olması olarak tanımlanabilir. Hayatımızda birçok optimizasyon problemleri ve çözümleri ile karşılaşmaktayız. Örneğin bir yere gitmek istediğimiz zaman en iyi ve en kısa yolu düşünerek yola çıkarız. Bir ürün satın almak istediğimiz zaman en kaliteli ve fiyatı en uygun ürünün hangisi olduğunu karar verir, ona göre hareket ederiz.

Optimizasyon problemi bilgisayar, elektrik, makine, inşaat ve kimya gibi farklı mühendislik alanlarında, matematikte, fizikte, ekonomide, sosyal bilimlerde çok geniş bir çalışma alanı bulunmaktadır. Araçların dizayn edilmesi, inşaat ve yapıların dizaynı, proses kontrolü, kalite ve kontrol işlemleri, üretim planlama aşaması, stok kontrolü gibi çok farklı alanlarda optimizasyon işlemleri yapılmaktadır. Optimizasyon sayesinde gerçek hayattaki birçok problemin çözümünde performans kriteri kullanılarak en iyi çözüm bulunabilir.

Optimizasyon matematiksel olarak sayısal bir fonksiyonu maksimize veya minimize etme probleminin çözümünde izin verilen değerlerini sistematik bir şekilde kullanarak arama işlemi olarak da tanımlanabilir. Bir denklemin köklerini bulmak veya optimizasyonu benzer anlamlar taşır. Kökün belirlenmesi fonksiyonun sıfır olduğu noktaların aranmasıdır. $f(x)$ gibi bir fonksiyonun türevinin sıfır olduğu nokta x değerinin optimum noktasıdır. Ayrıca $f(x)$ in ikinci türevi alındığında optimumun maksimum veya minimum olduğunu belirtir. Eğer $f(x)$ ikinci türevi 0'dan küçük ise nokta maksimumdur, $f(x)$ ikinci türevi 0'dan büyükse nokta minimumdur. Şayet $f(x)$ ikinci türevi 0 değerine eşit olursa kök problemini çözerek optimumu bulmaya yarar.

Optimizasyon metotlarının çoğu kök veya sıfır işlemini kullanır. Kök belirlenirken fonksiyonun sıfır olduğu yerlere bakılırken, optimizasyon işlemlerinde fonksiyonun türevinin sıfır olduğu değerler araştırılır. Birçok işlemde fonksiyonun türevini bulmak kolay olmamaktadır. Fonksiyon lineer değilse bulunan kökün optimum olup olmadığı sonucunu bulmak zorlaşmaktadır. Bu tip fonksiyonlar için optimizasyon bölgesi küçültülür ve sınırlandırılarak sonuca ulaşılır.



Şekil 3.1 Optimizasyon eğrisi.

Şekil 3.1. de gösterilen x^* noktası $F(x)$ eğri fonksiyonun minimum noktası iken $-F(x)$ eğri fonksiyonunun maksimum noktasıdır. Bir problemin birden fazla çözümü olabilir. Fakat önemli olan en iyi çözümü bulmaktır. Bazı problemlerin kesin bir çözümü bulunurken, bazı problemlerin optimal noktalar olarak bilinen farklı minimum ve maksimum noktalarına sahiptir. Optimizasyon işlemi ile bu farklı noktalar bulunabilir. Optimizasyon işlemlerinin temel aşamaları Şekil 3.2. de gösterilmiştir.



Şekil 3.2 Optimizasyon işlemlerinin temel aşamaları (Çunkas, 2006).

3.1 Optimizasyon Sınıfları

Optimizasyon algoritmaları kategorize edilerek Şekil 3.3'te verilmiştir. Optimizasyon algoritmaları aşağıdaki gibi sınıflandırılabilir.

Deneme yanılma yolu ile optimizasyon

Tek ve çok parametrelili optimizasyon

Statik ve dinamik optimizasyon

Sürekli ve ayrık parametrelili optimizasyon

Sınırlı ve sınırsız optimizasyon

Rasgele ve minimum araştırma algoritmaları

3.1.1 Deneme yanılma yolu ile optimizasyon

İşlem sürecinde hakkında bilgi olmadan çıkışı etkileyen değerlerin değiştirilmesiyle sonucun elde edilmesidir. Örneğin televizyon antenin en iyi ses ve görüntüyü yakalayabilmesi için yapılan deneme yanılma gibi.

3.1.2 Tek ve çok parametrelili optimizasyon

Optimizasyon, bir parametreden oluşuyorsa, tek boyutludur. Birden fazla parametresi varsa çok boyutlu optimizasyona ihtiyaç vardır. Çok boyutlu optimizasyonlarda boyut sayısı artıka zorluk derecesi de artmaktadır.

3.1.3 Statik ve dinamik optimizasyon

Statik optimizasyonlar zamandan bağımsız olmasına karşın, dinamik optimizasyonlar zamana bağlı olarak çıkış üretirler. Örneğin işine hızlı bir şekilde gitmek isteyen bir kişinin birçok alternatifi olduğu kabul edilirse en iyi yol mesafe açısından bakılırsa statik bir problem iken, hava şartlarına, yol durumuna göre bakılırsa dinamik bir probleme dönüşür. En kısa mesafe en hızlı olan yol olmayabilir.

3.1.4 Sürekli ve ayrık parametrelili optimizasyon

Parametreler sonsuz değer alıyorsa sürekli parametre olarak adlandırılırken, sınırlı değer alıyorsa da ayrık parametreler olarak adlandırılırlar. Örneğin liste halinde verilen yapılacak olan işlerin birbirinden bağımsız olması ayrık parametrelili olarak değerlendirilebilir. Bir fonksiyonun optimum değerini bulma işlemi sürekli parametrelili olarak örnek verilebilir.

3.1.5 Sınırlı ve sınırsız optimizasyon

Parametreler bir tanım aralığında tanımlanmış ise sınırlı optimizasyon olarak adlandırılır. Eğer parametreler herhangi bir değerde tanımlanmış ise sınırsız optimizasyon adı verilir. Sınırlı parametreler değişkenlerin sınırları kaldırılarak sınırsız parametrelere çevrilirler. Örneğin $f(x)$ fonksiyonunun sınırları $-1 \leq x \leq 1$ arasında olursa bu fonksiyonun $x = \sin(u)$ tanımı sınırlı olan bir fonksiyonu sınırsız bir optimizasyona dönüştürür. Değer aralığı ne olursa olsun $x - 1$ ile $+1$ aralığında değer alacaktır.

3.1.6 Rasgele ve minimum araştırma algoritmaları

Bazı algoritmalarda uygunluk değerlerini minimize etmek için parametrelerin başlangıç değerleri ayarlanır. Bu algoritmalar hızlı olmak birlikte yerel minimum değere ulaşabilir. Bu algoritmalar sayısal metotlara dayalı klasik optimizasyon olarak adlandırılır. Bir parametre kullanılarak diğer parametreyi tespit etmek, bazı belirli adımlarla olmaktadır. Rasgele metotlar ise parametrelerin optimum çözümünü bulmada ihtimal hesaplarını kullanarak optimizasyon gerçekleştirilir. Bu yöntem uzun sürede sonuca ulaşmasına rağmen global minimum değeri tespit etmede daha başarılıdır.



Şekil 3.3 Optimizasyon algoritmaları kategorileri (Haupt Randly ve diğ, 1998)

Bu gruplara bakıldığında optimizasyon metotlarını deterministik metotlar ve İstatistiksel metotlar olmak üzere iki ana gruba ayırabiliriz.

1-Deterministik Metotlar

2-İstatistiksel metotlar

Deterministik optimizasyon metotları, lokal maksimum veya minimuma yakınsayan algoritmalarıdır. Türevsel hesaplamalar veya türevsel yaklaşımlar deterministik metotlara örnek verilebilir (Haataja,1994). Yerel maksimum veya minimum sonuca yakınsayan algoritmalar deterministik optimizasyon algoritmaları olarak kabul edilir. İstatistiksel algoritmalar ise rasgele araştırma algoritmalarında olduğu gibi global maksimum veya minimum sonucu bulmak için rasgele sayılar ve bazı stratejiler kullanırlar (Palko, 1996).

Optimizasyon aracının kullanılabilmesi için öncelikle sistemin performansını sayısal olarak ölçülebilen bir amacın belirlenmesine ihtiyaç vardır. Bu amaç bir sayısal veri içeren kâr, zaman, miktar gibi ifadeler olabilir. Sistemin değişken veya bilinmeyen olarak ifade edilen belirli karakteristiklerine bağlıdır. Bu değişkenler sürekli olarak sınırlandırılabilir veya kısıtlanabilir. Problem için amaç, değişken ve kısıtların belirlenmesi ile modelleme yapılır. Belirli bir optimizasyon algoritması mevcut olmadığından benzer problemlere uyarlanmış algoritmalar uyarlanmaktadır. Problemin doğru ve hızlı çözümünün ilk koşulu doğru algoritmayı seçmektir.

Farklı optimizasyon problemleri için geliştirilen farklı çözüm yöntemlerinden bazıları Deterministik, Stokastik, İstatistiksel, Matematiksel, Sezgisel optimizasyon çözüm

yöntemleridir (Haataja, 1994). Optimizasyon problemlerinin çözümü için Tablo 3.1’de farklı optimizasyon tekniklerin sınıflandırılması verilmiştir.

Tablo 3.1 Optimizasyon tekniklerine örnekler (Bal, 1995; Karaboğa, 2004)

Matematiksel Programlama Teknikleri	Stokastik Süreç Teknikleri	Yapay Zeka Optimizasyon Teknikleri
Klasik Analiz	İstatistiksel Karar Teorisi	Genetik Algoritma
Doğrusal Olmayan Programlama	Markow Süreçleri	Tavlama Benzetim
Doğrusal Programlama	Yenilenme Teorisi	Tabu Araştırması
Dinamik Programlama	Benzetim Metodu	Karınca Koloni Algoritması
Oyunlar Teorisi	Güvenilirlik Teorisi	Diferansiyel Gelişim Algoritması
Kareli Programlama		Yapay Bağışıklık Algoritması

3.2 Meta Sezgisel Algoritmalar

Birçok optimizasyon problemi pratikte NP-Hard problemi olarak kabul edilir. Bu tür problemlerde optimal sonucu bulmayan fakat makul bir sürede iyi sonuçlar elde edebilen algoritmalar ile çözümler sunulmuştur. Bu yaklaşım meta sezgisel yöntemleri ortaya çıkarmıştır. Sezgisel yöntemler deneme yöntemi ile çözüme ulaşan yöntemlerdir.

Meta sezgisel algoritmaların amacı en iyi çözüme ulaşmaktır. Genellikle mevcut çözüm üzerinde bir değişikliğe bakılarak farklı çözümler aranır. Bu metodun yaptığı değişiklikler bazen olumlu bazen de olumsuz olabilmektedir. Sezgisel yöntem birçok parametreye bağımlı olabilir. Problemin çözümünde bağımlı olan parametrelerden dolayı meta sezgisel yöntemin başarılı olma durumunu belirsizleştirebilir. Meta sezgisel yöntemler çözüm yaklaşımları ortaya koyar. Bu yöntemler herhangi bir probleme ait çözüm ortaya koyamadığı için metodun probleme uyarlanması gerekir. Genetik Algoritma, Tavlama benzetimi, karınca kolonisi ve yapay sinir ağları bilinen meta sezgisel yöntemlerdendir.

3.3 Yapay Zekâ

Yapay zekâ optimizasyonu meta sezgisel algoritma olarak da bilinmektedir. Yapay zekâ, insan zekâsı taklit edilerek öğrenme ve öğrendiklerine göre davranma insan faktöründen bağımsız olarak hareket edebilen ve gelişme sağlayan tasarımlar sistemler ortaya koyar. Yapay zekâ, düşünme, öğrenme, akıl yürütme ve bunlardan bir sonuç çıkarma gibi insani kabiliyetlerin yapay sistemler ile gerçekleştirilmesidir. Yapay zekâ tekniklerinde insan beynin çalışması ve biyolojik evrim model alınarak insan gibi düşünmesi esas alınmaktadır. Yapay zekâ çözüm algoritmaları kesin çözümü garanti edemezler fakat uygun bir çözüme ulaşabilmeyi garanti ederler. Bir problem için başarılı sonuçlar elde ederken başka bir problem için aynı şekilde başarılı sonuçlar elde edemeyebilir. Yapay zekâ optimizasyon algoritmalarına yapay ısıl işlem algoritmaları, genetik algoritmalar, tabu araştırma algoritmaları örnek verilebilir (Ulusoy, 2002).

3.3.1 Evrimsel Hesaplama

Evrimsel hesaplama tekniğinin bir dalı olan Genetik Algoritma yapay zekânın bir kolu olarak genişledi. Evrimsel optimizasyon işlemlerinde amaç hayatta kalan yeni bireylerin bir sonraki nesilde daha yetenekli bireylerin oluşmasını sağlamaktır. Doğal yaşamda her canlının hayatta kalabilmek ve çoğalmak için yeteneklerini etkileyen karakteristikler canlının kromozomlarında uzun stringler ile ifade edilirler. Bir çiftten gelen bilginin kombinasyonundan oluşan yavru kromozomlar yeni nesillerde tekrar eşleşerek en iyi karakteristikleri içeren yavru kromozomları oluşturacaktır. Uygunluk değeri en yüksek bireyler seçilmesi ile doğal seçim işlemi gerçekleşir.

Doğal seçim anlaşılmadan Genetik Algoritma anlaşılabilir. Etrafımızdaki birçok olup biten olayda doğal seçim vardır. Dünyamıza baktığımızda muhteşem karmaşıklığı gözlemleriz. Bugün dünyadaki canlıları göz önüne aldığımızda çok büyük optimizasyonlar oluşturulmuş ve binler ile ifade edilebilecek iterasyonlar sonucunda dünyada optimum çözümler elde edilmiştir. İnsanoğlu yaşam mücadelesini yani amaç fonksiyonunu maksimize etmeyi ister (Grant, 1985). Evrim süreci doğal yaşama en uygun olanın hayatta kalmasını sağlayan bir algoritmadır. Doğal yaşamda çevreyi değiştirebilecek zekâ ve yeteneğe sahip olunursa, yaşamda global maksimuma ulaşılabilir (Haupt Randy ve diğ., 1998).

4. GENETİK ALGORİTMA

4.1 Genetik Algoritmanın Tarihçesi

Evrimsel hesaplama çok büyük bir geçmişe sahip olmasına rağmen son 20 yıl içinde dikkatleri üzerine çekebilmiştir. Evrimsel hesaplama I. Rechenberg tarafından 1960'larda yayınlanan "Evrimsel Stratejileri" isimli eserinde tanıtılmış ve sonra gelen araştırmacılar bu fikri geliştirmişlerdir. Bu süreci gelişimi taklit eden Genetik Algoritma ilk olarak Michigan Üniversitesinde görev yapan John Holland tarafından ortaya konmuştur. John Holland bir bilgisayar yardımı ile anlayamadığı çözüm yöntemlerinin öğretilebileceğini düşündü. Bu yaklaşım John Holland öğrencileri ve arkadaşları tarafından geliştirildi. John Holland bu sayede "Doğal ve Yapay Sistemlerde Adaptasyon (Adaption in Natural and Artificial Systems)" adlı kitabını 1975 yılında yayınladı (Holland,1992). Genetik algoritmalar alanında esas gelişim 1985 yılında John Holland'ın doktora öğrencisi David E. Goldberg adlı inşaat mühendisi tarafından hazırlanan "Gaz Boru hatlarının Genetik Algoritma Kullanılarak Denetlenmesi" konulu doktora tezi ile sağlanmıştır. İlk dönemlerde, genetik algoritmanın bir yararı olmadığı düşünülüyordu. National Science Foundation tarafından verilen genç araştırmacı ödülünü alması ve bu tezdeki uygulamalardan 4 yıl sonra Goldberg "Makine Öğrenmesi, Arama ve Optimizasyon için Genetik Algoritma" adlı 1989 yılında yayınladığı kitabı (Goldberg, 1989) ile çalışmalara yeni bir boyut kazandırmış ve yayınlanması ile genetik algoritmaya olan ilgi artmıştır (Şen,2004). Günümüzde genetik algoritma konusunda en kapsamlı referans olma özelliğini korumuştur (Körez,2005).

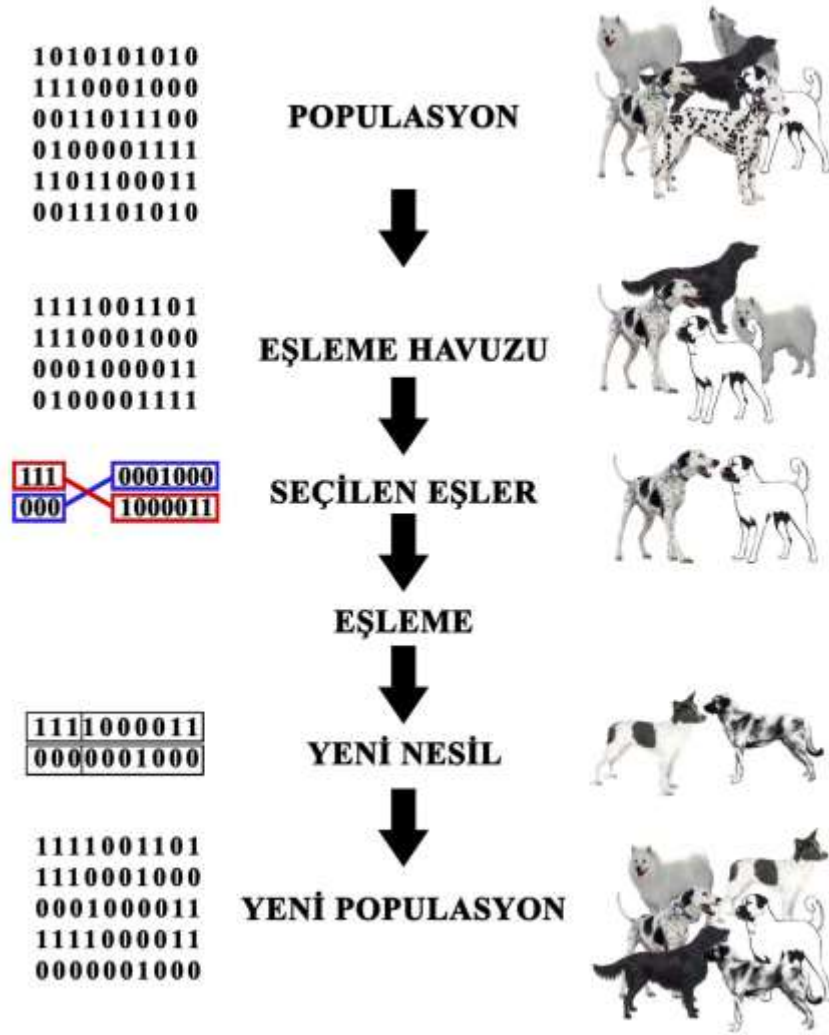
4.2 Genetik Algoritmalar

Yapay zekâ tekniklerinin bir alt kolu olan evrimsel hesaplamalar altında Genetik algoritmalar yer almaktadır. Karmaşık çözülemeyen problemler bilimin gelişmesi ile beraber yeni çözümler üretilmiş ve tüm optimizasyon çözümleri gibi genetik algoritmada da önemli katkılar sağlamıştır (Çunkaş, 2006). Genetik algoritmalar, matematikçi ve aynı zamanda psikolog olan John Holland tarafından 1970'li yılların başında ortaya atılmıştır. Bu algoritma bir veri aralığı arasından istenilen veriyi bulmak için kullanılmaktadır. Genetik algoritmalar tekrarlayan ihtimalli arama metodudur

(Holland, 1975). Bir problemin genetik algoritma ile çözümü, problemin sanal olarak biçimsel ve yapısal olarak dönüştürülmesi ile gerçekleşir (Adewuya,1996; Eiben ve Smith,2003). Genetik algoritma, büyük ve lineer olmayan arama aralığında problemlerin geleneksel hesaplama metotları ile sonuca ulaşmadığı durumlarda doğal genetik yöntemleri kullanan arama algoritması olarak tanımlanmaktadır (Louis, 2000). Genetik algoritma doğanın gelişim mekanizmasını model alınan bir çözüm tekniğidir. Genetik algoritmalarda problemin çözümü binary ya da binary olmayan sistemler üzerinde dizi yapısında kodlanmaktadır ve önemli bilgileri tutmak için bu dizilerde bazı işlemler uygulanmaktadır. Evrim teorisi esasına göre çalışan bu yöntem bir problem için en iyi çözüm ya da çözümleri arar. Bu yaklaşım doğal ortama uyum sağlayan canlıların hayatta kalmaları ve uyum sağlayamayan canlıların elenmesi olarak anlaşılmasıdır. Genetik Algoritmalar bu kuralları kullanarak en iyi çözümü aramayı hedefler (Goldberg, 1989; Şen, 2004).

Genetik algoritmalar ismini biyolojide geçen gen kavramdan esinlenerek doğadaki gelişim modeli alınarak meydana getirilmiş bir çözüm tekniğidir. Biyoloji biliminde genler temel kalıtımın yapı birimleridir. Genler DNA yapısı içerisinde bir çift kromozomdan oluşmaktadır. Her bir organizmanın hücresinde kromozom sayıları farklıdır. Örneğin insanda 46, kurbağada 26, sivrisinekte 6 kromozom mevcuttur. Bunların hepsi “allel” olarak bilinir (Curtis, 1975). Kromozom yapısındaki allel kombinasyonları bireylerin kişisel özelliklerini belirler. Bireyler üzerinde allel’lerden biri baskın diğeri pasiftir.

Genetik algoritmalarda biyolojik doğal yaşam arasındaki benzerlik şekil 4.1 de görülmektedir. Her ikisi de popülasyondaki bireylerin rasgele seçimi ile başlar. Örnekleme için köpek türünün özelliklerinin genetik kodu sol tarafta ikili kodlarla temsil edilmiştir.



Şekil 4.1 İkili kodun biyolojik yaşam ile genetik algoritmalar arasındaki benzetim (Haupt Randy ve diğ., 1998)

Her bir köpek ikilik sayılarla kodlanmıştır. Yeni bireylerin üretilmesi için popülasyondan rasgele iki birey seçilir ve eşleştirilir. Yeni oluşan bireylerde iyi havlayan köpeğin çıkma olasılığı oldukça yüksektir. Oluşan yeni bireyler istenmeyen bireylerin yerine geçer ve yine eşleştirme havuzuna aktarılarak işlemler tekrarlanır. Bu işlem en iyi birey seçilene kadar ya da döngünün bitişi sağlanana kadar devam eder (Goldberg, 1993).

Biyolojik sistemi modelleyerek yapıyı optimize eden evrim algoritmaları arasında yer alan genetik algoritmalarda biyolojide geçen gen kavramı genetik algoritmaların parametrelerini oluşturmaktadır. Bu parametreler yine biyolojide kavram olarak geçen kromozomu oluşturmaktadır. Genetik algoritmalarda tüm kromozomlar popülasyonu oluşturur. Belirli kurallar çerçevesinde popülasyonun uygunluğu optimize edilir. Uygunluk değerleri daha iyi olan çözümler bir sonraki kuşağa aktarılır. Sürekli bu

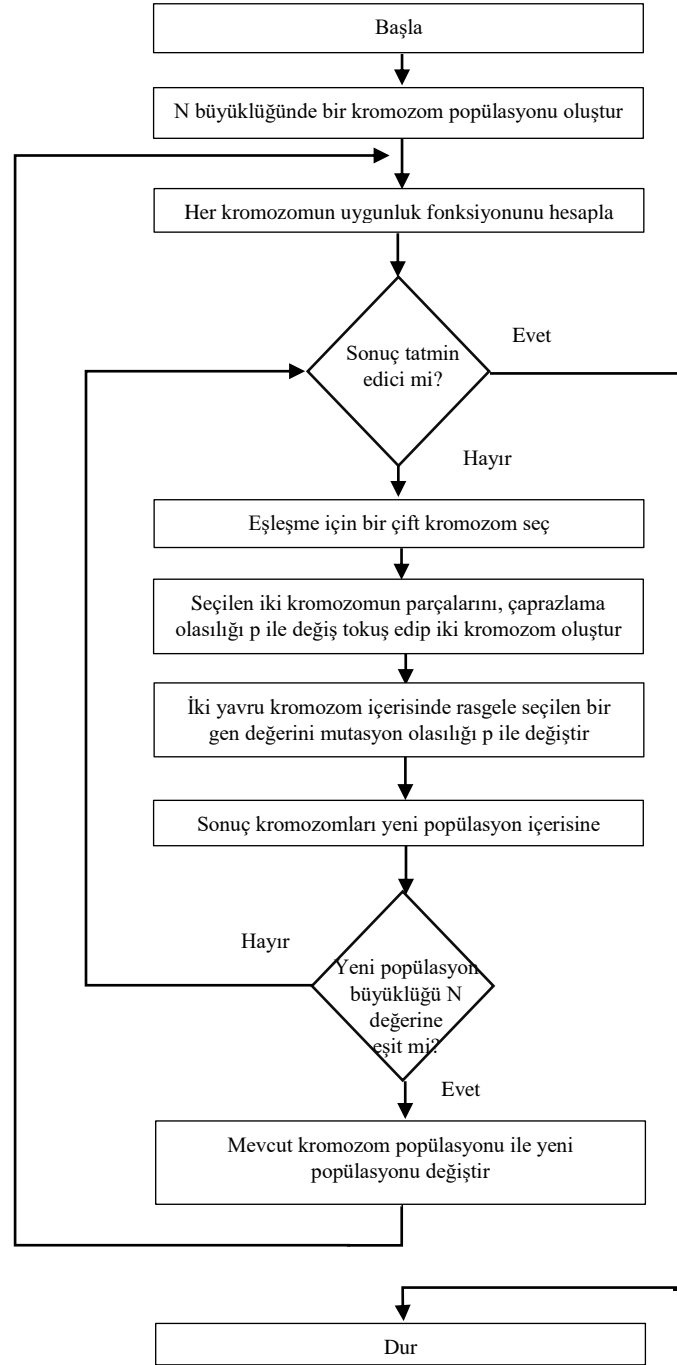
şekilde gerçekleşen bir problemde iyi çözümler veren bireyler olacağı için istenilen sonuca ulaşılacaktır (Goldberg, 1993; Negnevitsky, 2006). Rasgele bilgi değişimi sayesinde oluşturulan kromozomlar içinde hayatta kalanların birleştirilmesi ile yeni nesiller elde edilir (Angeline,1995).

Genetik Algoritmalarda popülasyonun her çözümü diğer çözümlerle karşılaştırarak daha iyi bir çözüm olduğunu ya da olmadığını gösteren uygunluk değeri atanır. Doğal yaşamda uygunluk değeri bireyin yaşam içerisinde mücadelesini ne kadar iyi olduğunu gösterir. Uyum değerinin yüksek olması bireyin sonraki nesilde yaşama şansını ve özelliklerini diğer nesillere aktarma ihtimalini artırır. Algoritmanın amacı popülasyonu geliştirmek ve en uygun bireyleri oluşturmaktır. Uyum değeri yüksek bireyler popülasyondaki diğer bireyler ile eşleşerek uyum değeri yüksek yeni bireylerin oluşması sağlanır. Bireylerin taşıdığı özellikler eşleme yöntemi ile diğer nesillere aktarılır. Uyum değeri düşük bireyler dolayısıyla sonraki nesillerde temsil edilme ihtimalleri azdır. Her iterasyonda en iyi bireyler seçildiğinden özellikleri iyi olan bireylerin oluşturduğu yeni bir popülasyon meydana gelir. Eğer genetik algoritma iyi tasarlanırsa popülasyon en iyi çözüme yakınsar (Beasley, 1993). Genetik algoritma genellikle optimizasyon probleminde başarılıdır. Genetik Algoritma problemlerinde ilk adım, aday çözümleri oluşturmaktır. Genetik algoritmalarda problemin çözüm uzayı olan kromozomların her birinde belirtilen değişkenlerine gen adı verilir ve genin olası değerlerine alleller ve değişkenlerin kromozom içerisindeki yerlerine lokus adı verilir. Değişkenin kromozom üzerindeki yeri basit problemlerde çok önemli değildir (Reeves, 1993).

Genetik algoritmalarda döngü içerisinde rastlantısal olarak en iyi çözüm bulan optimizasyon sistemidir. En iyi çözümü belirleyen uygunluk fonksiyonu yeni çözümler arasından seçim yapar ve daha iyi bir sonuç üretmek için çaprazlama operatörü kullanılır. Bu işlem süreci bir yerde tıkanıp aynı çözüm etrafında dolaşır. Bu süreci değiştirmek için mutasyon operatörü kullanılır. Döngü işlemi bittiğinde optimum sonuca ulaşılır.

Problemin çözümünde kullanılan bireylerin gösterimi problemin çözümünde önemli katkı sağlayan etkenlerdendir. Probleme göre bireylerin gösteriminde farklılıklar olabilir. Bir bireyin popülasyonda istenilen çözüm olup olmadığına uygunluk fonksiyonu karar verir. Buradan dönecek değere göre popülasyondaki başka bireyler ile çoğalmaları için seçim kümesine dâhil olurlar. Burada seçilen rasgele iki birey

çaprazlama ve mutasyon işlemine tabi tutularak çoğalma işlemi gerçekleşir. Bu oluşan popülasyondan kaliteli bireyler ebeveyn olarak seçilir böylece istenen kalitesiz bireyler popülasyona dahil edilmezler böylece kaliteli bireylerin sayısı artar. Bu metotla kaliteli bireylere sahip nesil oluşmaktadır. Uygunluk değeri optimum olan çözümler arttıkça uygunluk değeri daha iyi olan bireyler elde edilecek ve daha bir çalışma uzayı elde edilecektir.



Şekil 4.2 Genetik algoritmaların çalışma prensibi (Negnevitsky, 2006)

Şekil 4.2’de bitirme kriterine ulaşıncaya kadar popülasyonun döngü içinde tekrar tekrar yenilendiği görülmektedir. Bir sonraki döngüde bulunan uygunluk fonksiyonu en uygun değerden daha iyi değilse bulunan optimum değer mevcut uygun değer ile değişmez. Bazen optimum değer algoritma çalıştığı halde uzun süre değişmeyebilir. Bunun sebebi genetik algoritma rasgele olarak çalışmaktadır. En iyi sonucun arandığı problemlerde en iyiyi bulmak bazen uzun sürebilir. Yeni popülasyon bir önceki popülasyondan daha iyi olmayabilir. Böyle bir durumda sınırlı bir sürede optimum değer değişmez ise algoritma tekrar başa dönerek yeniden işlemlerin başlanması sağlanmalıdır. Bu tercih döngü içerisindeki tıkanıkların önüne geçerek daha kısa bir sürede optimum değere ulaştırmasına yardımcı olacaktır.

Problemin optimize edilmeden yapılması gereken parametrelerin dönüştürülmesi yani kodlama işlemi yapılması gerekmektedir. Kodlama Genetik Algoritmanın en önemli ayağını oluşturur. Gen olarak ifade edilen her bir eleman değişkenler string olarak tanımlanır. Bu değişkenler istenirse ikili kodla ya da sayılar ile ifade edilebilir ve probleme özel olarak istenilen aralık tanımlanır.

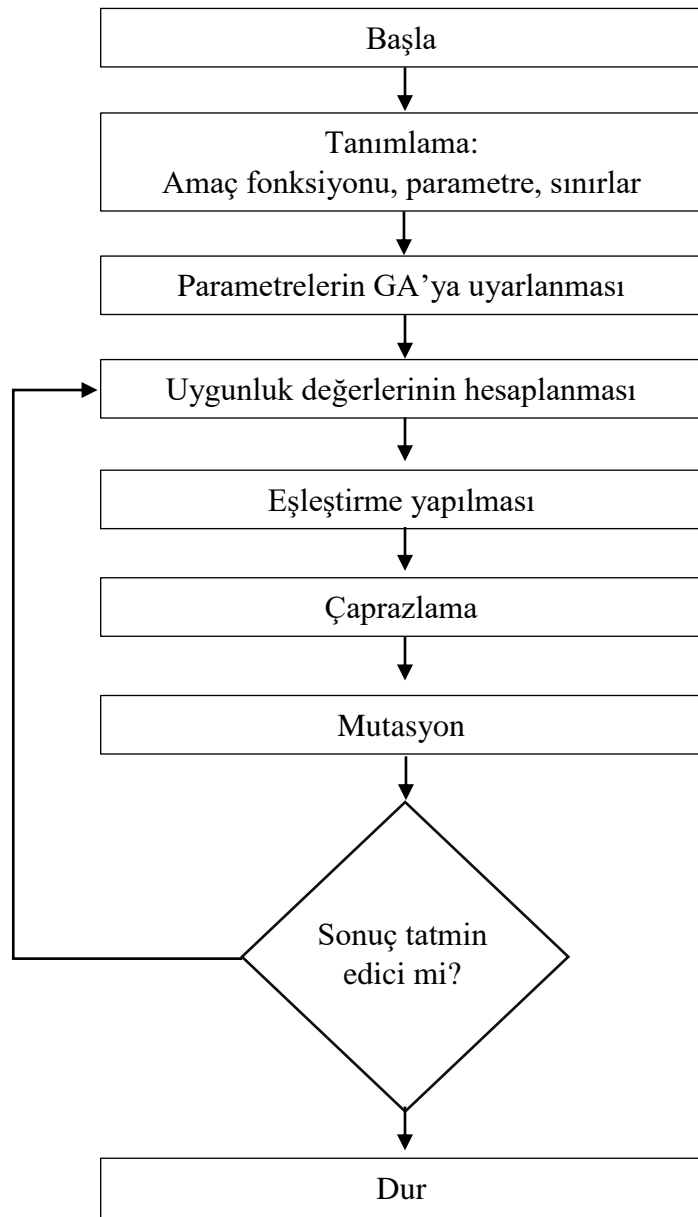
Optimizasyon ve arama işlemleri için sezgisel yöntemlerden olan genetik algoritmalar en iyiyi seçmek için geniş algoritmalar gibi tüm farklı durumları üretmez. Bundan dolayı mükemmel bir çözüm bulamayabilir. Genetik Algoritmalar hiçbir durumu önceden bilmemesine karşın bilgiyi öğrenme ve toplama yeteneği vardır. Genetik Algoritmalar her problemde iyi bir yöntem olmayabilir. Birkaç parametreden oluşan fonksiyonların çözümünde klasik metot daha kullanışlı ve hızlıdır. Diğer bölümde Genetik algoritmalar dışında rasgele araştırma algoritması kısaca tanıtılacaktır. Ayrıca ikili kodla çalışan Genetik Algoritma, Gerçek kodlu çalışan Genetik Algoritma, Permutasyon Kodlama ve Değer Kodlama işlenecektir.

4.2.1 Rasgele Araştırma Algoritması

En basit arama algoritmasıdır. Arama uzayında rasgele noktaları araştırır ve kabul gören bir sonuca ulaşıncaya kadar veya maksimum iterasyon sayısına ulaşıncaya kadar devam eder. Rasgele aramayı yapmak oldukça basit fakat bazı problemlerin çözümünde verimsiz olabilmektedir. Uygun çözüme ulaşmak uzun zaman alabilir.

4.2.2 İkili kodlu genetik algoritmalar

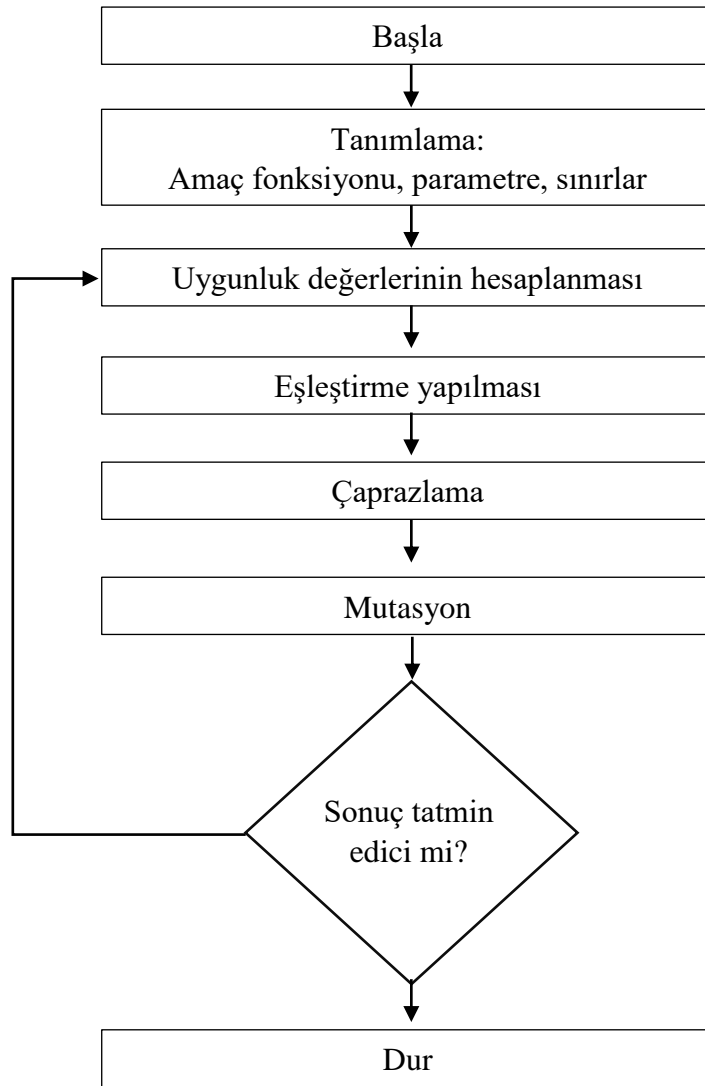
Bir veri kümesi için en iyi sonucu bulacak, lineer olmayan optimizasyon aracıdır. Genetik Algoritmada amaç fonksiyonu ve parametreleri tanımlanır. İlk çözümler rasgele üretilmesi ile başlar. Her bir çözüm uygunluk fonksiyonunun özel bir noktasını oluşturur. Her iterasyondaki farklı bireyler kümesi popülasyon olarak isimlendirilir. Popülasyonun en iyi bireyleri üretilir. Yeni üretilen neslin bir öncekilerden daha iyi olması beklenir. Netice olarak, bir popülasyonun en iyi çözümünün bir parçasından, yeni yavrular üretilir. Üretilen yeni nesil çocukların eski nesilden daha iyi olması beklenir. İkili kodlu genetik algoritmanın gösterimi şekil 4.3' te verilmiştir.



Şekil 4.3 ikili kodlu genetik algoritma (Çunkaş, 2006)

4.2.3 Gerçek Kodlama

İkili kodlama Genetik Algoritmalarda 1 ve 0 ile ifade edilen parametrelerin kromozomun boyutunu oldukça arttırmaktadır. Bunun yerine sayılar ile kodlama yapmak daha avantajlıdır. Gerçek kodlama sayesinde bilgisayar belleğinde veriler daha az yer kaplamaktadır. Mehmet Çunkaş'ın bu iki algoritmayı farklı test fonksiyonları kullanarak temel farklılıklarını test etmiştir (Çunkaş, 2006). Gerçek kodlu Genetik Algoritmaların ikili kodlu Genetik Algoritmaya göre daha hızlı çalıştığı ve optimum değere daha kısa sürede ulaştığını göstermiştir. Literatürde Sürekli Parametrelili olarak geçtiği bilinen gerçek kodlu genetik algoritma ikili kodlu algoritmaya çok benzemektedir. İkili kodlu genetik algoritma parametre olarak 1 ve 0 ları kullanırken gerçek kodlamada parametreler gerçek rakamlar ile ifade edilir. Gerçek kodlamanın akış diyagramı şekil 4.4. te verilmiştir.



Şekil 4.4 Gerçek kodlu genetik algoritma akış diyagramı (Çunkaş, 2006)

Gerçek kodlamada diziler arka arkaya sıralanan rakamlardan oluşmaktadır. Bazı durumlarda genetik operasyonlar tamamlandıktan sonra bazı düzeltmelerin yapılmasına ihtiyaç vardır. Gerçek kodlamada her bir kromozom rakamların arka arkaya sıralandığı, tam sayılı gerçek değerlerden oluşan bir dizidir. Kromozom yapılarının gösterimi şekil 4.5 te verilmiştir.

Kromozom 1	Kromozom 2
4 7 9 8 1 5 3 2 6	7 2 3 1 4 9 8 5 6

Şekil 4.5 Gerçek kodlama

Gerçek kodlama yalnızca problemleri sıralamada yarar sağlar. Bu durumda bile kromozom uyumluluğundan kaçınmak için bazı çaprazlama ve mutasyon düzeltmelerinin yapılması zorunlu olmaktadır. Bu kodlama fonksiyon optimizasyon problemlerinin çözümünde ikili ve gri kodlamaya göre daha etkin olmaktadır (Gen ve Cheng, 2000).

4.2.4 Değer Kodlama

Her bir kromozom yapısında bir değer dizisinden meydana gelir ve bu değerler ile ilişkili rakam, reel sayı veya özel bazı karakterler biçiminde olabilirler. Özel bazı problemler için bu kodlama en iyi sonucu verir. Bu tür problemlerde kendine özgü genetik operatörlerin geliştirilmesini zorunlu kılar. Değer kodlama biçimi aşağıdaki şekil 4.6. daki gibidir (Sivanandam ve Deepa, 2008).

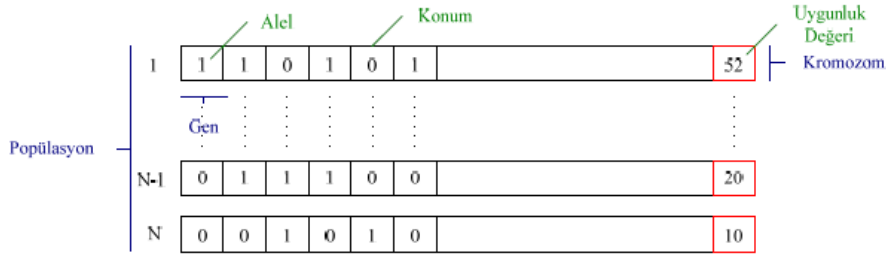
Kromozom A	1.2324 5.3243 0.4556 2.3293 2.4545
Kromozom B	ABDJEIFJDHDIERJFDLDFLFEGT
Kromozom C	C (geri), (geri), (sağ), (ileri), (sol)

Şekil 4.6 Değer kodlama

4.3 Genetik Algoritma Yapısı

Genetik algoritmada problem gösterimi için kromozom olarak adlandırılan yapı kullanılır. Her kromozom yapısı problemin çözüm uzayındaki rasgele bir çözümü (x) ifade eder. En çok bilinen kromozom yapısı binary (ikili bit) dizisi şeklindedir. Bu yaklaşımda kromozom büyüklüğü probleme göre değişir. N adet değişkeni olan bir

kromozom $[p_1, p_2, p_3, p_4, \dots, p_N]$ şeklinde gösterilir. p değerleri binary kodlamada 0 yada 1 değerini almaktadırlar. Popülasyon yapısı Şekil 4.7. de gösterilmiştir.



Şekil 4.7 Genetik algoritmanın popülasyon yapısı

Problemin en iyi çözüm kümesinin bulunabilmesi için;

- Bireylerin gösterimi doğru bir şekilde yapılmalıdır.
- Uygunluk fonksiyonu etkin bir şekilde oluşturulmalıdır.
- Doğru genetik işlemler seçilmelidir.

4.3.1 Basit Bir Genetik Algoritma ve Yapısı

Genetik algoritma yenilemeli ve rastlantısal bir süreçtir. Genetik Algoritma popülasyon esaslı optimizasyon yöntemidir. Popülasyon belirli bir bireyi ifade eder ve her birey çözümü temsil eder. Genetik Algoritmanın temel özellikleri arasında başlangıç popülasyonunun rasgele oluşturulması ve en iyi çözüm kümesini oluşturması için süreci devam ettirilmesidir. Rasgele seçilmiş bir popülasyon ile çözüme başlanır. Popülasyondan bireyler seçilir çaprazlama ve mutasyon işlemlerine tabi tutulurlar. Matematikçi J. Hadmard "Gerek matematikte gerek başka alanlarda buluş ve icatlar farklı alanlardan düşüncelerin bir araya gelmesiyle gerçekleşir" demiştir (Alander, 1997). Bu sözü ile Genetik Algoritmanın gelişim sürecini özetlemiştir. Aşağıda basit bir genetik algoritmanın temel aşamaları verilmiştir.

Begin

Baslangic populasyonunu olusturulmasi

while Sonlandirmakriteri saglanincaya kadar

Bireylerin uygunluk degerini hesaplanmasi

Bireylerin secimi (Çaprazlama İşlemi için)

Çaprazlama işlemi (Seçilen Bireylere Uygula)

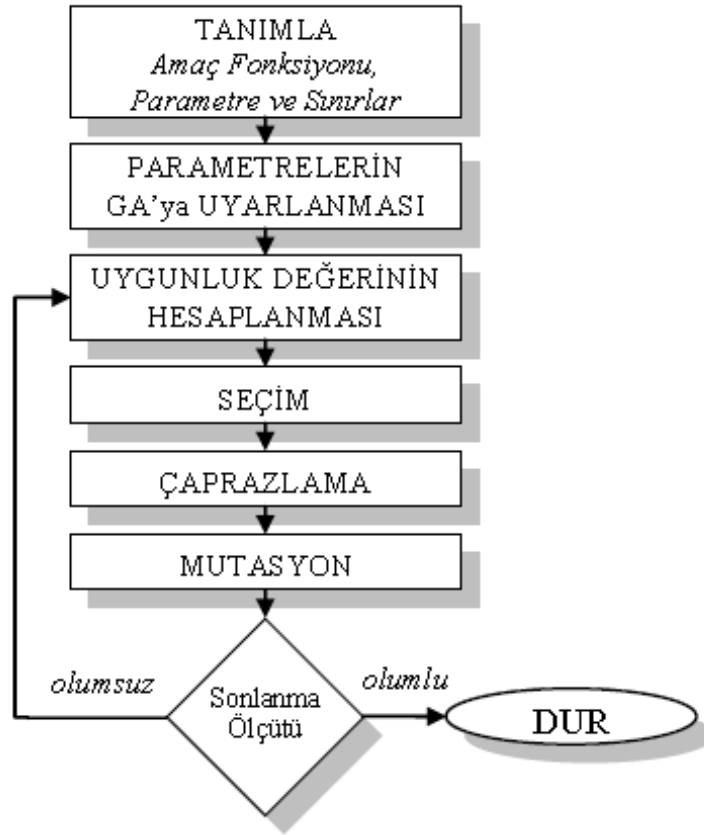
Mutasyon işlemi (Yeni Oluşan Bireylere Uygula)

Secim işlemi uygula (Bir Sonraki Popülasyon İçin)

end-while

End

Genetik Algoritma ile çözülen bir problemin çözümü için kullanılan akış şeması Şekil 4.8 de verilmiştir.



Şekil 4.8 Genetik algoritma akış diyagramı

4.3.2 Genel bir Genetik Algoritmanın adımları

1. $t=0$ ilk iterasyonunda N adet kromozoma sahip popülasyon rasgele oluşturulur (Her bir kromozom yapısı problemin olası çözüm kümeleridir).
2. Başlangıç Popülasyonu için rasgele büyüklükte kromozom seç
3. Her kromozom yapısı için $f(x)$ uygunluk değerini bulunur.
4. $t=t+1$. Bir sonraki popülasyon oluşturulur. Bunun için aşağıdaki adımlar gerçekleştirilir.
 - a. Kromozomlar bir önceki popülasyonun uygunluk değerlerine göre seçilme şansını kazanırlar. Yüksek uygunluk değerlerine sahip olan kromozomların olduğu popülasyonun seçilme ihtimali daha yüksektir. Düşük uygunluk değerine sahip

kromozomlar yeni popülasyona geçmeyebilirler. Yüksek uygunluk değerine sahip kromozomlar birkaç defa aynı popülasyona geçerler.

b. Seçim sonucunda meydana gelen popülasyonundaki kromozomlar belli bir olasılıkla çaprazlanırlar. Çaprazlama sonucunda meydana gelen yeni kromozomlar eski kromozomların yerine geçerler.

c. Çaprazlama işlemi sonucunda meydana gelen popülasyonda genetik çeşitliliği arttırmak amacıyla belli bir olasılık değeri ile mutasyon işlemi uygulanır.

5. Mutasyon işlemi sonucunda oluşan popülasyondaki yeni kromozomlar eski kromozomların yerine geçer.

6. Popülasyona ait tüm kromozomların uygunluk değerleri bulunur.

7. İlk başta belirlenmiş olan iterasyon sayısı tamamlanmamışsa 4. Adıma geri dönülür.

8. Tüm iterasyonlar sonucunda en iyi uygunluk değerine sahip olan kromozom problemin optimal çözümünü olur. Algoritma en uygun sonuca ulaşması ile veya belirlenen iterasyonda süreç duracaktır (Tabak, 2008).

Genetik Algoritmaların diğer algoritmalarından 4 farkı vardır (Goldberg, 1993).

- Genetik Algoritmalarda çözümler kodların değiştirilmesi ile aranır. Çözümün üretilmesi parametrelerin kodlanmasına bağlıdır.
- Genetik Algoritmalarda çözüm aralığından rasgele seçilen değerlerin operatörler ile daha iyi bir noktaya ulaşması amaçlanır.
- Genetik Algoritmalarda uygunluk fonksiyon değeri tercih edilir.
- Genetik Algoritmalarda olasılıksal kurallar kullanılır.

4.4 Başlangıç Popülasyonunun Oluşturulması

Genetik Algoritmalarda yapılması gereken ilk aşama belirlen sayı kadar kromozomdan oluşacak bir popülasyon oluşturmaktır. Genel görüş popülasyon büyüklüğünün 100-300 arasında bir sayı olarak belirlenir. İşlemlerin karmaşık olması ve arama derinliği büyüklük seçiminde önemlidir. Üretilen başlangıç popülasyonu rasgele değerlerden oluşturulmaktadır (Karaboğa, 2004). Bu durum uygun olmayan çözümlere neden olabilir. Bu durumdan kurtulmanın yolu probleme özel sezgisel

yöntemlerden yararlanılır (Koza, 1992). Oluşan çözüm gurubunun biyolojideki benzerliği nedeniyle popülasyon, oluşan çözüm kodlarına kromozom adı verilmektedir. Başlangıç popülasyonu ve her kromozomda oluşturulan bitlerle $N_{pop} \times N_{bit}$ gibi bir matris hazırlanır. N_{pop} başlangıç popülasyon sayısını, N_{bit} ise kromozomdaki bit sayısını ifade eder. Matriste her bir satır kromozomu ifade eder. Popülasyon büyüklüğünün fazla olması daha çok çözüm elde edilmesini sağlar. Popülasyon büyüklüğü arttıkça çözüme ulaşma süreside doğrusal olarak artacaktır. Popülasyonun az olması optimum çözüm süresini kısaltabilir fakat daha iyi elde edilebilecek sonuçlara ulaşamayabilir.

4.5 Amaç fonksiyonu ve parametreler

Problemin giriş parametre değerlerine göre bir çıkış değer üreten fonksiyonlardır. Bu fonksiyonun amacı problemin giriş değerlerine göre uygun değere ulaşmaya kadar çıkış üretmektir. Örneğin banyo kovasını doldururken bir optimizasyon gerçekleşir. Sıcak su ve soğuk su çeşmeleri bizim giriş parametremiz olur. Bu örnekte elimizle suyun sıcaklık ayarını kontrol ederiz. Burada amaç fonksiyonu deneysel sonuçlar üretir (Haupt Randy ve diğ, 1998).

Genetik Algoritmalarda amaç fonksiyonu bazı durumlarda karmaşık olabilmektedir. Problem için hangi parametrenin daha önemli olduğuna dikkat edilmelidir. Genetik Algoritmalarda giriş parametrelerin çok olması istenen bir durum değildir çünkü bu durumda iyi bir sonuca ulaşamayabilir. Örneğin araba tasarımlarında yakıttan tasarruf etmek için motor, ağırlık ve araba boyutları gibi parametreler olarak belirlenebilir. Fakat arabanın boyası ve koltuk tipi rengi gibi benzer parametrelerin neredeyse hiç etkili değildir. Bu nedenlerden dolayı parametre seçimi, tecrübeye dayanarak ve deneme yanılma ile tespit edilebilir.

Optimizasyon problemleri genellikle belirli sınırlar içerisinde parametre değerleri alması istenir. Örneğin araba ağırlığı sıfır olamaz veya boyu 10 metreyi geçemez gibi değerler alınması istenebilir. Sınırlamalar iki şekilde belirlenebilir. Örneğin bir malzemenin boyu 0 ile 10 arasında bir değer olarak belirlenebilir. Eğer boyu 11 değeri girilirse boy=10 yapılır. Yine aynı şekilde -1 değer alacak olursa boy=0 olarak değiştirilir. Diğer bir sınırlama $x=5\sin y+5$ olarak belirlenirse y 'in herhangi bir

değerinde x 0 ile 10 arası değer alacaktır. Böyle bir yöntem ile sınırlı bir optimizasyon problemini sınırsız bir optimizasyon problemine dönüştürebiliriz (Holland, 1992).

Optimizasyon işlemlerinde birbirlerine bağlı parametreler bazı sorunları da beraberinde getirmektedir. Bir parametrenin değişimi başka bir parametreyi doğrudan etkileyecektir. Örneğin arabanın boyutu değişirse arabanın ağırlığı değişir. Bağımsız parametreler ise birbirlerini etkilemezler. Genetik Algoritmada parametre etkileşimi “epistasis” olarak adlandırılan kavram çok düşük olursa, minimum araştırma algoritmaları iyi çözümler üretir. Bu değer orta ve yüksek değerde veya çok düşük olursa iyi sonuç vermemektedir. Epistasis biyolojide genlerin birbirlerini etkilemesi olarak tanımlanır (Haupt Randy ve diğ, 1998).

4.6 Uygunluk değeri ve hesaplanması

Genetik algoritmanın performansı etkileyen en önemli unsur çözümlerin değerlendirilmesidir. Bu değerlendirmeyi yapabilmek için dikkat edilmesi gereken husus uygunluk fonksiyonudur. Problem çözümünde genetik algoritma kullanılmak isteniyorsa uygunluk fonksiyonuna ihtiyaç vardır. Uygunluk fonksiyonun üzerinde çalışılan probleme uygunluğu Genetik Algoritmanın sağlıklı çalışması ve doğru sonuçlar vermesini etkileyecektir. Uygunluk fonksiyonu problem için bulunan her çözümde kromozomun ne kadar iyi bir çözüm olduğunu bulan fonksiyondur. Bu fonksiyonun kromozomlarının uygunluk değerlerini bulmasına hesaplama (evaluation) denir (Painton ve Campbell, 2000). Uygunluk fonksiyonu sayesinde popülasyonun en iyi bireyleri bir sonraki nesle aktarılacaktır. Popülasyon içerisindeki bütün bireylerin uygunluk değerleri hesaplanır ve en iyi bireylerin bir sonraki nesle aktarılması sağlanır. Üretilen yeni bireyler başarısız bireylerin yerini alır. Böylece bireyler her nesil ile yenilenecek en iyi sonuca ulaşmaya yaklaşarak optimum çözüme ulaşırlar.

Uyum fonksiyonunda problemde en yüksek değer yani maksimumu aranıyorsa tüm x çözümleri için $f(x) \geq 0$ biliniyorsa

$f(\hat{x}) = \max f(x)$ eşitliğini sağlayan x değerleri bulunur.

Bu problem en alt değeri yani minimumu aranıyorsa

$f(\hat{x}) = \min f(x)$ eşitliğini sağlayan x değerleri bulunur.

Problemin uygunluk deęerine gre minimizasyon yapılacaksa uyum deęeri dşk, maksimizasyon yapılacaksa yksek uygunluk deęeri aranır.

4.7 Doęal seęim

Bireylerin oęalma iin seildięi bir iřlemdir. Uygunluk deęerlerine bakılarak poplasyonun en uygun kromozomlarının seilme srecine doęal seim denir. Dięer bir ifade ile poplasyon havuzundan yeni zm bireylerinin oluřturulmasındaki srecin adıdır. Sonraki nesil iin yeni birey meydana getirecek ebeveyn bireylerin ne řekilde seileceęi ve her bir ebeveyn bireyin ne kadar yeni bireye sahip olacaęı belirlenmelidir (Mitchell, 1996). Bir sonraki nesil iin yeni birey oluřturacak ebeveyn bireyler ile reme havuzu yani ara poplasyon oluřturulur. Poplasyonun oęalması iin havuzdan rasgele ebeveyn bireyler seilir. Uygunluk deęerine bakılarak deęeri kk olan kromozomlar bir sonraki kuřaęa aktarılması engellenmiř olacak ve yeni kuřak kromozomlar istenilen dzeye yaklařacaktır. Doęal seim iřlemleri sayesinde farklı yapıda kromozomlar meydana getirebilmek ve kromozom zenginlięini arttırmak iin kromozomlar aprazlama iřlemine sokulurlar. Genetik Algoritmada ilk poplasyondan bir sonraki kuřaęa aktarılacak ebeveynlerin seiminde ortalama uygunluk deęerinin arttırılması amalanmaktadır. Her kuřakta ortalama uygunluk deęeri arttıka istenen duruma daha uygun poplasyonlar oluřacaktır. İlerleyen kuřaklarla beraber uygunluk deęeri aynı kalitedeki ve birbirine benzer bireylerin aprazlanması ile homojen bireyler oluřacaktır. Ama kk poplasyonlar ile alıřarak optimum zme yakın verimli bir poplasyon elde etmek okta mmkn deęildir. Doęal seim teknikleri bu problemlerin nne gemek iin geliřtirilmiřtir. Bu seim yntemi tekniklerinden rulet tekerleęi (Ranking Scheme) yntemi, turnuva seim (Tournament Scheme) yntemi ve elitist seim yntemleri en ok kullanılanlardır. Bařlangı poplasyonu oęullama yntemi ile oluřturulur. Daha sonra yeni poplasyonun oluřturulabilmesi iin seim yntemine karar verilmesi gerekir. Yksek uyumlu bireylerin seilme olasılıęı daha fazla olduęundan bir sonraki nesile kopyalanması amalanır (Blicle,1995). Uyum deęerine gre bir sonraki nesile aktarılacak bireyleri seen yntemlere uyuma orantılı seim yntemleri denir (Tomassini, 1996). Doęal seim tekniklerinden en ok tercih edilen rulet emberi ve turnuva seimidir.

4.8 Kodlama (Kromozomların temsili)

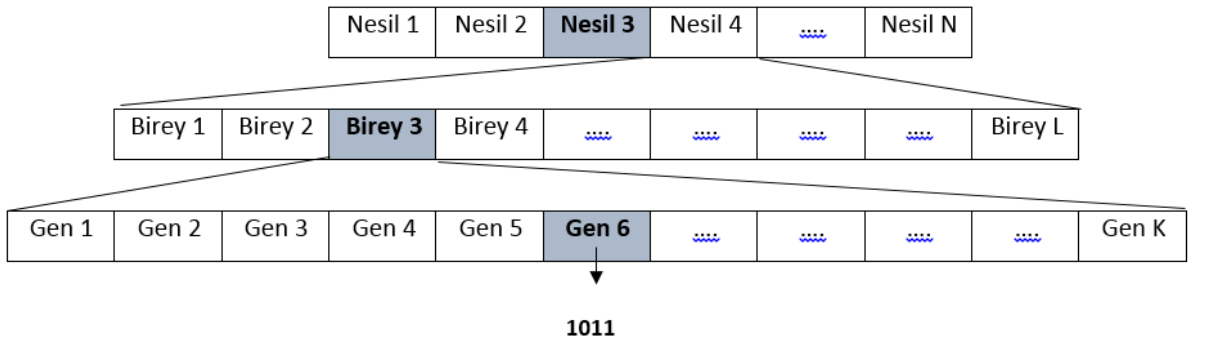
Genetik algoritma problemlerinin çözümünde ilk olarak yapılması gereken problemin değişkenlerini temsil eden kromozomların her bir geni üzerinde kodlanmasıdır. Belli bir çözüm ile ilgili tüm bilgileri taşıyan kromozom oluşturulur. Kodlama yapısı, probleme ait değişken yapısına bağlıdır ve kodlama yöntemi performans açısından önemlidir. En çok kullanılan kodlama yapısı ikili kodlamadır. Basit bir genetik algoritma ikili diziden oluşan bir yapı üzerinde çalışır. Burada kromozomlar, 0 ve 1'lerden oluşan diziler halindedir. Her bir parametre n adet kadar bit kullanır. Her bir parametrenin n değeri farklı olabilir (Bridges & Goldberg, 1987). Kromozomların kodlanmasında tam sayı dizileri veya kurallar dizisi olarak farklı gösterim biçimleri mevcuttur. Genetik Algoritma her çalıştırıldığında popülasyondaki dizilerin uygunluk değerleri değerlendirme fonksiyonu ile hesaplanır. Uygunluk fonksiyonu sayesinde kromozomlar problemin parametresi olurlar ve bu parametrelere göre hesaplama yaparlar. Uygunluk fonksiyonun verimli ve hassas olması genetik algoritmanın başarısını artırır. Bağımsız parametrelerin kromozom içinde kodlanması gerekir (Davis, 1991). 8 bitlik genlerden oluşan bir kromozom örneği aşağıdaki şekil 4.9 da verilmiştir. Tüm popülasyonların temsili gösterimi ise şekil 4.10'da verilmiştir.

Kromozom=[Gen1 , Gen2 ,, GenN]

Kromozom=[11001110, 10110110,, 10100110]



Şekil 4.9 Gen Yapısı parametrelerin kodlanması



Şekil 4.10 Kromozomların temsili gösterimi

4.9 Genetik Algoritmada Kullanılan Operatörler

Genetik Algoritmalarda problemin çözümü ile ilgili bilgiyi seçim işlemi verebilir. Ancak kesin bir çözüme ulaştırmaz. Genetik Algoritma popülasyon olarak ifade edilen bir çözüm seti yani kromozomlar ile başlatılır. Bir popülasyon sonuçları bir önceki popülasyon daha iyi olacağı düşünülen yeni popülasyon oluşturmak için kullanılır. Popülasyon büyüklüğüne ulaşıncaya kadar seçim işlemi devam eder ve her birey seçildikten sonra üreme havuzuna aktarılır. Üreme havuzuna katılan bireyler yeni neslin ebeveynleri olurlar. Seçim işlemlerinden sonra eşleşen ebeveyn kromozomlar ile bu eşleme sonucu oluşan yavru kromozomlar üretmek için genetik operatörler kullanılmaktadır. Problemin en uygun çözümünü için kromozom çeşitliliği arttırılacaktır. Uyumluluklarına göre seçilen çözümlerle yeni popülasyon oluşturulur. Çünkü uyumlu seçimlerin daha iyi sonuçlar vermesi olasıdır. Bu durum istenen çözüme ulaşıncaya kadar tekrarlanır (Kurt, 2001). Genetik operatörler sayesinde çözüme ulaşmayı sağlayacak yeni bireyler elde edilirler. Yeni çözüm oluşturmak için kullanılan iki önemli genetik operatör vardır. Genetik algoritmalarda kullanılan operatörler çaprazlama ve mutasyon işlemidir (Negnevitsky, 2006).

4.9.1 Çaprazlama

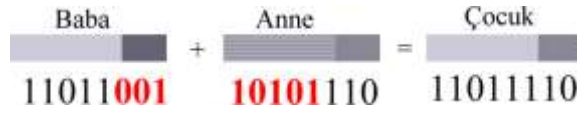
Genetik algoritma ile çözülen birçok uygulamada en önemli operatör çaprazlama operatörüdür. Bireylerin iyi özelliklerini çaprazlama operatörü ile birleştirerek daha iyi bireylerden oluşan çözümler oluşturması beklenir. Popülasyondan yeni iki kromozom elde etmek için havuzdan iki adet kromozom seçilme işlemi gerçekleşir. Bu süreçte seçilen kromozomlardan yeni kromozom oluşturma işlemine çaprazlama denir (Şen, 2004). Bu işlemde öncelikle ilk kromozomun çaprazlama noktasının en solundaki bitler eşleşen diğer kromozoma geçer, diğer kromozomun aynı şekilde en solundaki bitler ilk kromozoma geçer. Yeni bireylerin her biri bir önceki neslin genlerini taşır. Bir popülasyonda ne kadar çaprazlama işlemi olacağı çaprazlama oranına bağlıdır. Çaprazlama operatörlerinden en çok kullanılan tek, iki, çok nokta ve üniform çaprazlamalardır. Şekil 4.11 de genel çaprazlama operatörü kuralı verilmiştir.



Şekil 4.11 Genetik algoritmalarda çaprazlama

4.9.1.1 Tek nokta çaprazlama

Genetik algoritmanın en basit çaprazlama örneğidir. Çaprazlama sonucu eşlemeye girecek olan popülasyonun bireyleri muhtemel ebeveynleri içeren $N/2$ oranında bir popülasyon kümesi elde edilir. Çaprazlama olasılığı sayesinde çaprazlama uygulanacak çözüm bireyleri belirlenir. Genellikle 0,6 - 1 aralığında P olasılığı kullanılmaktadır (Man ve diğ.,1995). Rasgele seçilen 0 ile 1 arasındaki sayı P olasılığından küçük ise bu çözüm çiftine çaprazlama uygulanır. Değilse bu çözümler bireyler aynen kalır. Her iterasyonda $P(\text{olasılık}) \times N(\text{Birey Sayısı})$ kadar bireye çaprazlama uygulanır (Grefenstette, 1986). Aşağıda verilen şekillerde tek noktalı çaprazlama örnekleri sunulmuştur.



Şekil 4.12 İkili kodlu genetik algoritmalarda tek noktalı çaprazlama



Şekil 4.13 Tek noktalı çaprazlama sonucu oluşan kromozomlar



Şekil 4.14 Gerçek Kodlu Genetik Algoritmalarda Tek noktalı Çaprazlama

İlk olarak kromozom uzunluğu kadar rasgele bir tam sayı yani bir çaprazlama noktası belirlenir. Örnekte çaprazlama noktası için 4 olarak verilmiştir. Bu aşamadan sonra iki

kromozom çaprazlama noktasından sonraki ikinci kısımlar yer değiştirerek, yeni çözüm bireyleri elde edilir.

Elde edilen bireyler ebeveynlerinden farklı olabilirler, ancak ebeveynlerin özelliklerini korur. Eğer iki ebeveyn yüksek uyum değerlerine sahip ise, bireylerden en az birinin ebeveynlerinden daha iyi olma olasılığı yüksektir. Böyle bir seçim yavruların üremesini sağlarken aksi durumda yavruların yok olmaları sağlar. Çaprazlama sadece yüksek uyuma sahip bireyler üremeyebilir. Bazen bir kısmı ya da çoğu düşük uyuma sahip yavrular meydana gelebilir. Bu olasılığı yok etmek için P çaprazlama olasılığından yararlanır.

4.9.1.2 İki nokta çaprazlama

İki farklı çaprazlama noktası vardır. Bu noktalar arasında kalan alt dizinler diğer kromozomun alt dizinleri ile yer değiştirmesi ile yeni bireyler elde edilir (Tomassini, 1996). Aşağıda verilen diziler arasında iki nokta çaprazlama işlemini uygulanış şekli gösterilmektedir. Çaprazlama noktası 1 ile dizinin elemanı N-1 arasından rasgele sayı seçilmektedir. Aşağıda verilen şekillerde çift noktalı çaprazlama örnekleri sunulmuştur.



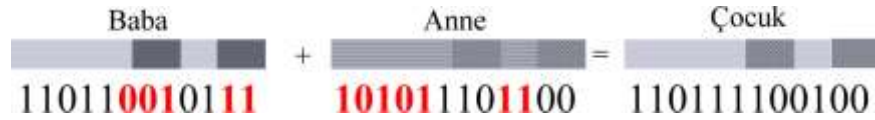
Şekil 4.15 Genetik Algoritmelerde Çift Noktalı Çaprazlama



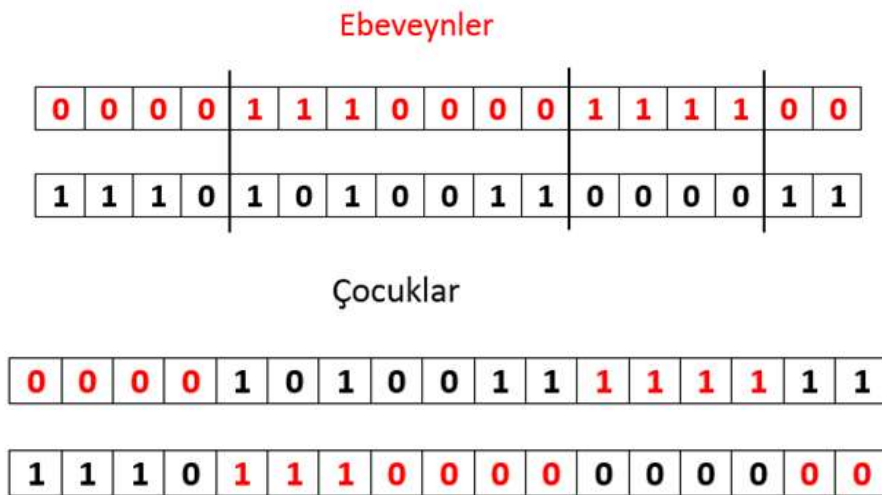
Şekil 4.16 İki nokta çaprazlama sonucu oluşan kromozomlar

4.9.1.3 Çok Nokta Çaprazlama

İki nokta çaprazlamanın farklı bir versiyonu ve eklentili halidir. Her bir kromozom çözümü n çaprazlama noktası ile n parçaya ayırılır. Bir sıra atlanarak elde edilen allel blokları çiftler arasında değiştirilerek yeni bireyler oluşturulur (Srinivas ve Patnaik, 1994). Çok nokta çaprazlama operatörü testleri daha fazla alt dizi değiştirildiğinde performansta gerileme kaydedildiği sonucuna varılmıştır. Performansı arttırmak için bazı değişiklikler kazandırmak gerekir. Ancak, yapılan bu değişiklikler fazla olursa, bir çözüm bireylerinin iyi özelliklerine karar verme olasılığını arttırır. Bu konuda yapılan çalışmalar çok noktalı çaprazlamaların kromozomdaki iyi özellikleri birleştirerek daha uygun olacağını ortaya koymuştur. Etkin bir genetik Algoritma aramasında mevcut çözümün iyi özelliği işletmek ve yeni özellikler kazanılması arasındaki denge çok önemlidir (Tomassini, 1995). Aşağıda verilen şekillerde çok noktalı çaprazlama örnekleri sunulmuştur.



Şekil 4.17 Genetik Algoritmalarda Çok Noktalı Çaprazlama



Şekil 4.18 Üç nokta çaprazlama sonucu oluşan kromozomlar

4.9.1.4 Üniform çaprazlama

İki ebeveyn ata verildiğinde ilk yavrunun her geni, rasgele olarak oluşturulan bir çaprazlama maskesi ile birinci veya ikinci ebeveyn atanın karşılık gelen genin kopyalanması ile oluşturulur. Çaprazlama maskesinde 1 (bir) o genin ilk ebeveyn atadan kopyalanacağını, 0 (sıfır) ise o genin ikinci ebeveyn atadan kopyalanacağını söyler. Şekil 4.19’da üniform çaprazlamanın nasıl yapıldığı anlatılmaktadır.



Şekil 4.19 Üniform çaprazlama sonucu oluşan kromozomlar

Gerçek uygulamaların birçoğu için probleme özgü çözüm gösterimi ve çaprazlama operatörü geliştirilebilmesi Genetik Algoritmanın avantajlarından biridir. Çaprazlama ile üreyen bireyler popülasyona ait olmayan bilgileri alamazlar. Yani var olan popülasyonun tüm bireyleri, dizinin 1.pozisyonunda 1 (bir) içeriyorsa, çaprazlama bu pozisyonda 0(sıfır) olan bir bireyi hiçbir zaman oluşturmaz (Odetayo, 1995).

Genetik Algoritmanın bu teknikleri karşılaştırılmıştır. Üniform çaprazlama bitlerinin pozisyonları dikkate alınmaksızın değiştirmektedir. Ancak bu yapı çok fazla sorun olmaktadır. Tek nokta ve iki nokta çaprazlamalarda popülasyon homojen bir yapıda olduğunda aramaya daha çok yardımcı olmaktadır (Beasley, 1993; Goldberg, 1991).

Popülasyon büyük olması çaprazlama türünü doğal olarak etkilemektedir. Elde edilen sonuçlar küçük popülasyon için üniform çaprazlama daha başarılı iken, büyük popülasyonlar için iki nokta çaprazlama daha başarılı olduğunu göstermektedir. Küçük popülasyonlarda keşifsel bir aramaya yaparken üniform çaprazlamanın bozuculuğu oldukça yardımcı olmaktadır. Büyük popülasyonlarda mevcut çeşitlilik keşif gereksinimini azaltmakta ve iki nokta çaprazlamayı büyük popülasyonlar için daha uygun hale getirmektedir (Srinivas ve Patnaik, 1994).

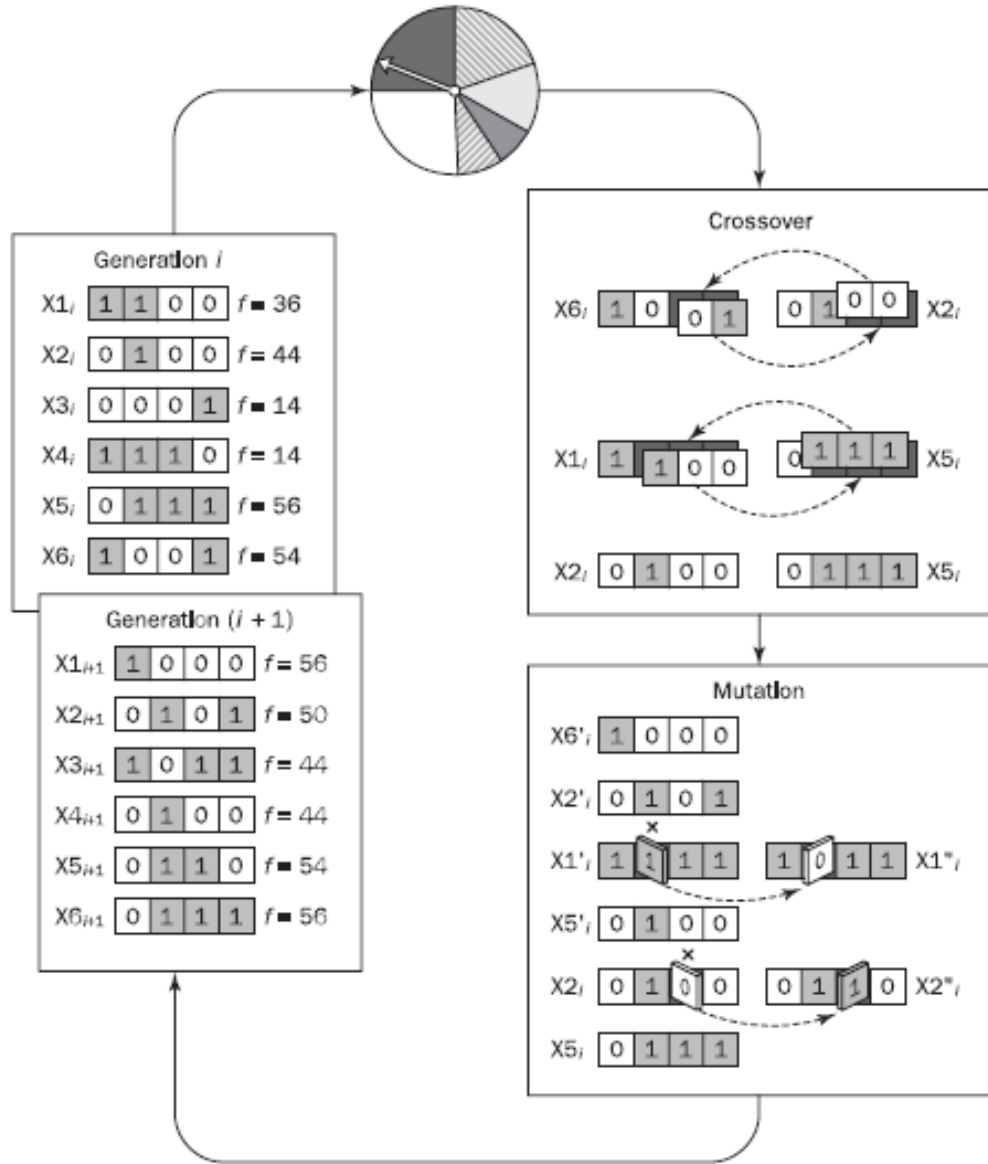
4.10 Mutasyon

Kromozom üzerindeki rasgele seçilen genin şekil 4.20'deki gibi değiştirilmesidir. Mutasyon işlemi uygunluk değerine ciddi bir etkisi olmasa da, çoğu durumda olumsuz olarak etkileyebilmektedir. Bu nedenlerden dolayı mutasyon işleminin gerçekleştirme frekansı olan mutasyon oranı düşük olarak seçilmektedir. Çaprazlama oranı kadar yüksek değer girilmesi popülasyonun olumsuz olarak etkilenmesine neden olabilir. Genetik Algoritmalarda mutasyon frekansı için 0.01 ile 0.001 arasında değerler tercih edilmektedir (Çunkaş, 2006; Aksu, 2008). Genetik algoritma operatörünün çalışması Şekil 4.21. ile gösterilmiştir. Genetik algoritmalarda mutasyon operatörünün görevi arama işlemlerinde uygun bir çözümde sabit kalmadığını ve arama uzayında kısır döngüden kurtarmayı sağlar. Bütün bu işlemler ile beraber birbirine benzer kromozomların olduğu popülasyonda çeşitlilik azalabilmektedir. Böylece tüm kromozomlar birbirinin aynısı ve birbirine benzer yapıda olacak ve optimum ortalama uygunluk değerine ulaşamayabilir. Arama işlemleri devam etmeyerek uygun çözüm bulamayacaktır. Bu süreci değiştirmenin yolu mutasyon operatörünü kullanarak farklı kromozom yapıları elde etmek ve böylece popülasyondaki kromozom çeşitliliğini arttırmaktır. Mutasyon ve çaprazlama işlemi sonucunda yavru bireylerden oluşturulan popülasyon mevcut popülasyon sayısına ulaştığında yeni popülasyon mevcut popülasyon değiştirilerek bir sonraki süreçler tekrarlanır.



Şekil 4.20 Mutasyon işlemi sonucu oluşan kromozomlar

Doğal seçim, çaprazlama ve mutasyon işlemleri sonunda üretilen yavru kromozomların oluşturduğu yeni popülasyonun büyüklüğü mevcut popülasyonun büyüklüğüne eşit olduğunda, yeni popülasyon mevcut popülasyon ile değiştirilerek bu evrim devam ettirilmektedir. Probleme verilen kriter kadar süreç tekrarlanır ve genetik algoritma en uygun çözümü uygunluk değeri en yüksek olan kromozom olarak sonuçlandırmaktadır.



Şekil 4.21 Genetik Algoritmelerde doğal Seçim ve Genetik Operatörler (Çunkaş, 2006)

4.11 Rulet çemberi seçimi

Bu yöntem stokastik örneklem olarak da bilinmektedir (Goldberg, 1989). İlk olarak holland tarafından ortaya atılmıştır. Her bireyin P seçilme olasılığı hesaplanır ve bir tablo da muhafaza edilir. Tablodaki bu olasılık değerleri biri diğerine eklenerek ilerler. Belirli bir rasgele sayıya ulaşıncaya da bu sayıyı aştığında sayının ait olduğu birey seçilir. Bu işlemler popülasyondaki kromozomların sayısı kadar tekrarlanır. Bir dairenin, bireylerin uyum değerlerine orantılı olarak dilimlenmesi ile çevrildiğinde elde edilen sonuçların benzetiminden dolayı rulet tekerleği adı verilmiştir.

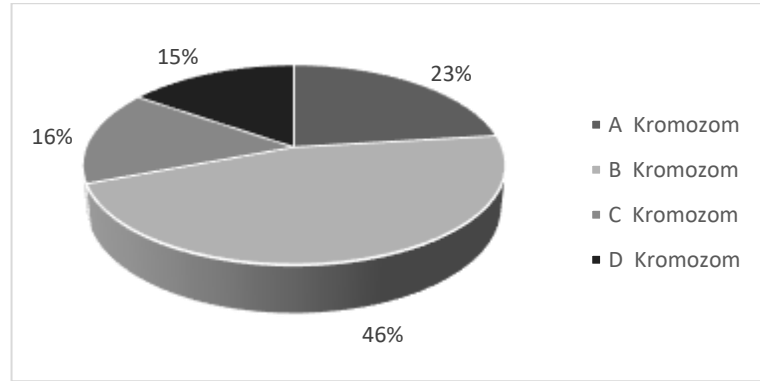
Popülasyon genişliği=6

A için üyelik değeri=3

B için üyelik değeri=6

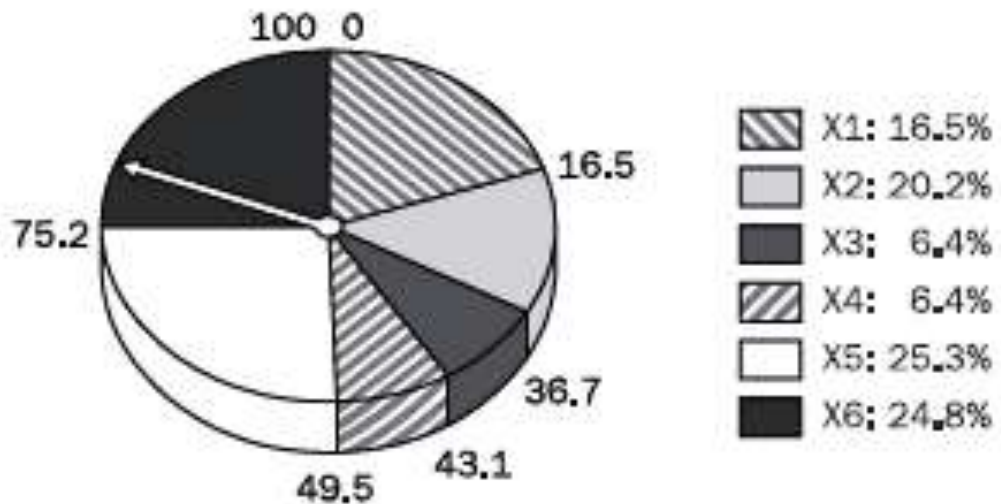
C için üyelik değeri=2

D için üyelik değeri=2



Şekil 4.22 Rulet Tekerleği ve bireylerin yüzde oranları

Bu tekniğin uygulanabilmesi için popülasyonun tüm kromozomlarının uygunluk değerleri hesaplanır. Her kromozom toplam uygunluk değerine bölünerek 0 ile 1 aralığında bir seçim ihtimali hesaplanır. Bundan sonra kümülatif değerler hesaplanır. Popülasyonun büyüklüğü kadar 0 ile 1 arasında rasgele sayı üretilir. Üretilen rasgele sayı hangi kümülatif değerler arasında ise o kromozom seçilir. Böylece rulet yöntemi kullanılmış olur. Rulet çarkı yöntemi şekil 4.22 ve şekil 4.23'te verilmiştir.



Şekil 4.23 Rulet çemberi seçimi (Turgut, 1992)

4.12 Turnuva Seçimi

Turnuva seçimi sıralama seçimine benzer bir yöntem olup daha verimli ve uygulamaya daha çok yatkındır. Turnuva seçiminde rasgele bir sayı kadar bireylerden oluşan grup seçilir. Rasgele seçilen sayı büyüklüğüne turnuva genişliği denilir. Seçilen bireyler daha sonra bir turnuvaya katılır. Bu gruptaki en iyi birey yeni popülasyona eklenir ve işlem N kez tekrarlanır. Turnuva genişliğinin büyük olması yöntemin seçicilik etkisini arttırır (Michalewich, 1994). Genellikle turnuva yönteminde iki birey seçilir ve ikili turnuva olarak adlandırılır (Blicle, 1995). Çaprazlama işlemi için iki kez turnuva yapılır. İlk seçimde her bir ebeveyn seçilir. Turnuva seçiminin avantajı, popülasyondaki zayıf bireylerin seçilmeyerek diğer kuşağa aktarılmamasını sağlayarak kötü bireylerin popülasyonun genetik yapısını etkilenmesine engel olmaktadır.

4.13 Elitizm

Algoritma içerisinde kullanıcı tarafından kodlanırken belirli sayıda uygunluk fonksiyon değerinden en iyi çözüme sahip bireyler hiçbir işleme girmeden yeni popülasyona aktarılarak elit bireyler tutulur. Uygunluk değeri kötü olan bireyler bir sonraki işlemlerde diğer nesillere aktarılmazlar. Elit bireylerin sayısının çok olması popülasyonun çeşitliliğini azaltırken aynı zamanda kromozom yapılarının değişmesine sebep olur. Eski nesildeki bireylerden uygunluk değeri yüksek olanların yeni nesile aktarılmaması durumunda yeni nesildeki bireylerden uygunluk değeri en yüksek değere sahip bireyin bile eski nesilden kötü olma ihtimali vardır.

4.14 Genetik Algoritmaların Avantajları ve Dezavantajları

4.14.1 Avantajları

- ✓ GA türevsel bilgiler içermez,
- ✓ Sürekli ve ayırık parametreleri optimize eder,
- ✓ Amaç fonksiyonunu geniş bir aralıkta arar,
- ✓ Çok sayıda parametre ile çalışabilir,
- ✓ Karmaşık amaç fonksiyonlarını yerel minimum ya da maksimumlara takılmadan optimize eder,
- ✓ Birden fazla parametrelerin optimum çözümleri elde edebilir (Tomassini, 1996)

4.14.2 Dezavantajları

- ✓ Teorik alt yapıları basit olup problem uzayının özelliklerinden istenilen kadar yararlanamazlar.
- ✓ Seçimler rasgele bir sisteme dayalı arama gerçekleştiği için problemin çözümü uzun sürebilmektedir.
- ✓ Mutlak sonuca ulaşılması istenen problemlere uygun değildir.
- ✓ Sınırlı bir arama işlemi gerçekleştiği için en ideal bir sonuca ulaşmayı garanti etmez.
- ✓ Lokal optimizasyon işlemlerinde etkili değildir.

4.15 Genetik algoritmanın kullanıldığı bazı özel uygulama alanları

(Çunkas, 2006 ; Negnevitsky, 2006).

- ✓ Gezgin Satıcı Problemi
- ✓ Araç Rotalama Problemi
- ✓ Çizelgeleme Problemi
- ✓ Montaj Hattı Dengeleme Problemi
- ✓ Baskı Devre Kartlarında İşlem Sırası Belirleme Problemi
- ✓ Tesis Yerleşim Problemi
- ✓ Atama Problemi
- ✓ Hücresel Üretim Problemi
- ✓ Sistem Güvenilirliği Problemi
- ✓ Optimizasyon (Bakım, servis, depo toplama).
- ✓ Görüntü işleme.
- ✓ Makine öğrenmesi (Yapay Sinir Ağlarında öğrenmeyi sağlayan ağırlık hesaplamaları, robotlar).
- ✓ Sosyal sistemler (Böcek kolonileri, Çok etmenli sistemlerde iş birliği).
- ✓ Yapay sinir ağları

5. MERKEZİ SINAVLARDA OTURUM DÜZENLERİNİN GENETİK ALGORİTMA İLE OPTİMİZASYONU

5.1 Genetik Algoritmaya Dayalı Tek Boyutlu Sınav Yerleştirme Modeli

Bu bölümde genetik algoritma kullanılarak merkezi sınavlarda en iyi oturma düzenini sağlayan modeli tanımlanmış ve tez çalışmasında "*Genetik algorithmada tek noktali çaprazlama kullanımına dayalı model*" veya kısaca "*Klasik tek noktali çaprazlamaya dayalı model*" olarak isimlendirilmiştir. Bu modelin nasıl tasarlandığı bir örnek üzerinden verilmektedir. X adet öğrenci ve farklı kapasitelerdeki Y adet sınıfa yerleştirilecektir. Birbirlerini tanıyan öğrenciler yan yana veya arka arkaya gelmeyecek şekilde yerleştirilecektir. Biri 12 diğeri 8 kişi kapasiteli farklı iki sınıf ve toplamda 20 öğrenci olduğu kabul edildi. Şekil 5.1’de C1 ve C2 sınıf ID numaraları olmak üzere sınıf oturma planı bir matris düzeninde verilmiştir.

	1.sütun	2.sütun	3.sütun
1.satır			
2.satır			
3.satır			
4.satır			

C1 Sınıfı

	1.sütun	2.sütun
1.satır		
2.satır		
3.satır		
4.satır		

C2 Sınıfı

Şekil 5.1 Sınıf oturma düzeni

Yukarıda matris düzeninde verilen sınıf oturma planında her bir sıraya yerleştirilecek öğrenci ID numarası bir gen olarak temsil edilmektedir. Öğrencilerin ID numaraları S1, S2, S3,... , S10 şeklinde kabul edildiğinde kromozom yapısı Şekil 5.2’deki gibi modellenmiştir.

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	→ Öğrenci ID
C1	C1	C2	C1	C2	C1	C2	C2	C1	C1	→ Sınıf ID
2	4	1	3	3	3	2	2	3	4	→ Satır No
2	1	2	2	2	3	2	1	1	2	→ Sütun No

Şekil 5.2 Klasik tek noktali çaprazlamaya dayalı modelin Gen yapısı

Öğrenci ID numaraları S1, S2..., S20 olmak üzere modelin kromozom gen yapısına Şekil 5.3'te örnek sunulmuştur. Kromozom ilk popülasyonda her bir sıraya bir öğrenci yerleştirilecek şekilde modellenmiştir.

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
C1	C1	C2	C1	C2	C1	C2	C2	C1	C1	C1	C1	C2	C2	C1	C1	C1	C2	C1	C2
2	4	1	3	3	3	2	2	3	4	2	2	4	4	1	1	1	3	4	1
2	1	2	2	2	3	2	1	1	2	3	1	2	1	3	2	1	1	3	1

Şekil 5.3 Klasik tek noktali çaprazlamaya dayalı modelin Kromozom yapısı

5.1.1 Kullanılan Veriler

En iyi oturma planının hazırlanmasında gerçek veriler kullanılmıştır. Mardin Artuklu Üniversitesi Uzaktan Eğitim Uygulama ve Araştırma Merkezinden gerekli izinler alınarak gerçek veriler tez çalışmasında kullanılmıştır. Bu veriler arasında öğrencinin adı soyadı, TC numarası, öğrenci numarası, cinsiyet, doğum yeri, doğum tarihi, yaş, uyruk, bölümü, programı, ikametgâh adresi gibi öğrenciye ait birçok bilgi yer almaktadır. Bu bilgilerden öğrencinin doğum yeri, okuduğu program ve ikametgâh adresi esas alınarak öğrenciler arasındaki yakınlık ilişkisine bakılmaktadır. Bu bilgiler excel formatında hazırlanarak tarafımıza ulaştırılmıştır. Excel verisi 68 farklı bölüm, 157 farklı doğum yeri ve 138 farklı ikamet adreslerinden oluşuyor. Bu veriler ilk 3 sıralamaya göre incelenmiştir. Doğum Yerine göre 118 öğrenci Mardin, 113 öğrenci Diyarbakır ve 111 öğrenci Kızıltepe nüfusuna kayıtlıdır. İkamet adreslerine göre 132 öğrenci Artuklu, 97 öğrenci Kızıltepe, 45 öğrenci Midyat ilçelerinde ikamet etmektedir. Ön lisans ve Lisans programa göre 127 öğrenci İslami İlimler, 54 öğrenci ilk ve acil yardım, 47 öğrenci Mimarlık Programına kayıtlıdır. Tez çalışmasında her bir analiz 10 kez tekrar edilerek analiz sonuçlarının ortalaması alınmıştır. Bu sayede analiz sonuçlarında hata payının en fazla %10 olduğu düşünülmektedir. Genel hatalardan sonuç çıkarma üzerine yazılan makale bu durumu desteklemektedir (Nadeau ve Bengio,2003).

Python programlama dilinde öncelikle hem öğrencilerin bulunduğu veriler hem de sınav salonlarına ait verilerin bulunduğu excel dosyası okutulur. Bu dosyada öğrencilerin bilgileri veri tabanlarında olduğu gibi sütunlar halindedir. Aynı şekilde sınav salonlarına ait excel dosyasında da sınav merkezi binalar, sınav salonları, sınıf mevcutları ve her sınıfın oturma düzeni bulunmaktadır. Bu sayede öğrencilerin hangi

sınav salonunda nasıl oturacaklarını karar veren yazılım içinde bir kod yapı yer almaktadır. Excel verileri okunarak bir kromozomda oluşması gereken gen yapısı da belli olur.

Yazılım içerisinde öğrenciye ait kimlik bilgileri dışında program bilgisi, doğum yeri bilgisi ve ikamet bilgisi kullanılmaktadır. Ayrıca üniversitedeki birimlerin sınıf bilgileri ve mevcutları sisteme dâhil edilmiştir. Öğrencilerin yakınlık ilişkisini tespit etmede bu veriler kullanılarak ceza puanları belirlenmektedir. Yan yana veya arka arakaya oturan öğrencilerden aynı bölümde okuyan, aynı ikamet adresinde oturan veya doğum yeri aynı olan öğrencilerin bulunduğu sınıf düzenlerine ceza puanı verilmektedir. Öğrencilerin birbirlerini tanıma senaryoları için birkaç örnek verilebilir.

S2 - S9 ' in aynı bölümde okuyabilir.

S5 – S18 aynı ikamette oturuyor olabilir.

S6 – S19 doğum yeri aynı olabilir.

Bu ve benzeri senaryo örnekleri çoğaltılabilir. Bu senaryo dikkate alınarak en iyi oturma düzeni oluşturmak için örnek bir çalışmanın adımları şu şekilde olur.

5.1.2 Algoritma çalışma adımları

Adım 1.

İlk olarak 4 kromozomdan oluşan rastgele bir popülasyon oluşturacağız. Oluşturduğumuz her kromozomun uygunluk değeri hesaplanacak. Denklem 5.1 de her bir kromozoma ait uygunluk değeri f ile gösterilmiştir. Uygunluk değeri, sisteme dâhil olan verilerden elde edilerek bir fonksiyon yardımıyla sıralı olarak belirlenmektedir. Yaptığımız uygulamada amacımız problemde en iyi olanı seçmek olduğu için tasarladığımız uygunluk fonksiyonu ceza eksenli bir sonuç sistemi olarak çalışmaktadır. Kromozom yapısında hedef çözüm ceza puanının 0 olması yâda 0'a yaklaşması yani minimum değeri bulunmasıdır. Bir kromozom yapısında birbirleri ile ilişkisi olan genleri tespit etmek yani aynı salonda arka arakaya veya yan yana oturan kişilerin birbirleri ile olan tanışıklıkları bulunan öğrencilerin tespiti durumunda bir ceza puanı belirlenmektedir. Her bir tanışıklık için aynı ceza puanı verilmemiştir. Aynı salonda fakat birbirlerine yakın oturmeyen tanışık olan öğrencilerin bulunması durumunda aralarındaki sıra sayısına göre ters orantılı farklı bir ceza puan sistemi geliştirilebilir.

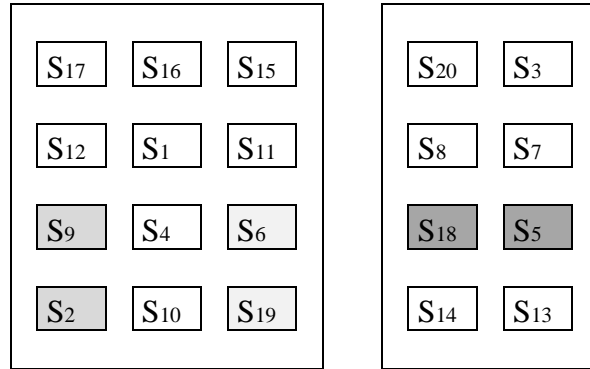
$$f = 1/1 + \sum_n^S(\text{Ceza Puanı}) \quad (5.1)$$

Kromozom1

S ₁	S ₂	S ₃	S ₄	S ₅	S ₆	S ₇	S ₈	S ₉	S ₁₀	S ₁₁	S ₁₂	S ₁₃	S ₁₄	S ₁₅	S ₁₆	S ₁₇	S ₁₈	S ₁₉	S ₂₀
C ₁	C ₁	C ₂	C ₁	C ₂	C ₁	C ₂	C ₂	C ₁	C ₁	C ₁	C ₁	C ₂	C ₂	C ₁	C ₁	C ₁	C ₂	C ₂	C ₁
2	4	1	3	3	3	2	2	3	4	2	2	4	4	1	1	1	3	4	1
2	1	2	2	2	3	2	1	1	2	3	1	2	1	3	2	1	1	3	1

Şekil 5.4 Klasik tek noktalı çaprazlamaya dayalı modelin kromozom 1 yapısı

Kromozom1 yapısı verilen Şekil 5.4'te uygunluk değeri hesaplandığında ($f=1/1+\sum\text{Ceza Puanı}$) üç çift arasında birbirleri ile olan ilişkisi tespit edilir ve $f=1/4$ değerini bulunur. Kromozom 1 öğrenciler mevcut salonlara rastgele yerleştirildiğinde aşağıdaki Şekil 5.5. deki gibi bir düzen meydana gelir.



Şekil 5.5 Klasik tek noktalı çaprazlamaya dayalı modelde C1 – C2 sınıf düzeni

Rastgele belirlenen başlangıç popülasyonu ile birlikte kromozom yapısında belirtildiği üzere sınıf düzeninde dağılım gerçekleşir. Kromozomların uygunluk değerlerine göre çaprazlama işlemleri gerçekleştirilmektedir. Çaprazlanan kromozomlardan yeni yavru bireyler elde edilir. Oluşan bireylerin hangilerinin doğru bir çözüm olup/olmadığına karar vermek için her işlemin sonunda bir fonksiyon yardımıyla karar verilir. Kromozomlarda verilen oturma düzeninde kişinin hemen sağındaki ve arkasındaki kişi ile bir yakınlıkları kontrol edilerek belirlen ceza puanını alır. Kromozom1'e ait oturma düzeni incelendiğinde C1 sınıfında 2 farklı benzerlik ve C2 sınıfında ise 1 benzerlik olmak üzere toplam 3 benzerlik tespit edilmiştir.

Kromozom 2 ($f=1/3$)

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
C1	C1	C2	C1	C2	C1	C2	C2	C1	C1	C1	C1	C2	C2	C1	C1	C1	C2	C2	C1
3	2	3	3	3	4	2	4	1	4	3	1	1	1	1	2	4	2	4	2
2	2	1	3	2	1	1	1	2	2	1	1	1	2	3	3	3	2	2	1

Kromozom 3 ($f=1/2$)

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
C1	C1	C2	C1	C2	C1	C2	C1	C1	C1	C2	C2	C1	C2	C1	C1	C1	C1	C2	C2
4	3	3	3	3	1	2	4	4	2	2	4	1	4	1	3	2	2	1	1
2	1	2	2	1	3	1	3	1	3	2	2	2	1	1	3	2	1	1	2

Kromozom 4 ($f= 1/2$)

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
C2	C2	C2	C1	C2	C1	C1	C1	C1	C1	C2	C1	C1	C2	C2	C1	C1	C1	C1	C2
3	3	1	3	2	2	1	3	2	4	2	3	4	1	4	1	1	4	2	4
1	2	2	3	2	3	2	1	1	2	1	2	3	1	2	1	3	1	2	1

Şekil 5.6 Klasik tek noktali çaprazlamaya dayali modelde kromozomların gösterilişii

Adım 2.

Şekil 5.6' da görüldüğü gibi Kromozom 3 ve Kromozom 4 çaprazlama işleminin sonucunda Yeni Birey 1 görülmektedir. Çaprazlama işleminin hangi noktadan yapılacağına rastgele bir sayı üreterek belirlenmektedir. Belirlenen rastgele noktayı 5 kabul edelim. Kromozom 3'ün ilk 5 geni ile Kromozom 4'ün son 15 geni birleşerek Yeni Birey 1'i oluşturmaktadır. Aynı şekilde Kromozom 4'ün ilk 5 geni Kromozom 3'ün son 15 geni birleşerek Yeni Birey 2 oluşmaktadır.

Kromozom 3

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
C1	C1	C2	C1	C2	C1	C2	C1	C1	C1	C2	C2	C1	C2	C1	C1	C1	C1	C2	C2
4	3	3	3	3	1	2	4	4	2	2	4	1	4	1	3	2	2	1	1
2	1	2	2	1	3	1	3	1	3	2	2	2	1	1	3	2	1	1	2

Kromozom 4

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
C2	C2	C2	C1	C2	C1	C1	C1	C1	C1	C2	C1	C1	C2	C2	C1	C1	C1	C1	C2
3	3	1	3	2	2	1	3	2	4	2	3	4	1	4	1	1	4	2	4
1	2	2	3	2	3	2	1	1	2	1	2	3	1	2	1	3	1	2	1

Yavru Birey 1

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
C1	C1	C2	C1	C2	C1	C1	C1	C1	C1	C2	C1	C1	C2	C2	C1	C1	C1	C1	C2
4	3	3	3	3	2	1	3	2	4	2	3	4	1	4	1	1	4	2	4
2	1	2	2	1	3	2	1	1	2	1	2	3	1	2	1	3	1	2	1

Yavru Birey 2

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
C2	C2	C2	C1	C2	C1	C2	C1	C1	C1	C2	C2	C1	C2	C1	C1	C1	C1	C2	C2
3	3	1	3	2	1	2	4	4	2	2	4	1	4	1	3	2	2	1	1
1	2	2	3	2	3	1	3	1	3	2	2	2	1	1	3	2	1	1	2

Şekil 5.7 Klasik tek noktalı çaprazlamaya dayalı modelde çaprazlama işlemi

Şekil 5.7’de verilen çaprazlama işlemi sonucunda oluşan yeni yavru bireylerde kromozomun tamire ihtiyacı olacaktır. Sınav salonların bazılarında aynı sıraya farklı iki öğrencinin oturması söz konusudur. Yukarıdaki örnekte oluşan yeni Yavru Birey1 de 1 ve 10 numaralı öğrenciler C1 sınıfının 2.sütun 4.satırında oturduğu görülmektedir. Aynı şekilde 2 ve 8 numaralı öğrenciler C1 sınıfı 1.sütun 3.satırda ve 4 ve 12 numaralı öğrenciler C1 sınıfı 2.sütun 3.satırda aynı sıra denk gelmiştir. Bu durumun önüne geçmek için bir yöntem geliştirilmiştir; bkz. Bölüm 5.2.1. Çaprazlanacak ikinci kromozomun her bir geni karşı taraftaki genin yerini alır ve karşıdaki gen ise kendi yerine geçen karşı genin kendi kromozom içindeki yeri ile yer değiştirir. Böylece çaprazlama sonucu aynı kromozom içinde birden fazla aynı gen bulunmasını engellenir.

Merkezi Sınavlarda en iyi oturum düzeninin sağlanması ile ilgili yayınlayan Klasik tek noktalı çaprazlamaya dayalı modelde eksik olarak görülen sentetik veriler yerine gerçek verilerden yararlanılmıştır. Tablo 5.1’ de gerçek verilerden yararlanılarak yapılan klasik tek noktalı çaprazlamaya dayalı modelin analiz sonuçlarına göre verinin %20’nin üzerinde genlerin tekrarladığı gözlemlenmiştir. Klasik tek noktalı çaprazlamaya dayalı modelde kromozom yapısında tekrarlanan gen olması durumunda dikkate alınmadan çaprazlama işlemi yapılmaktadır. Klasik tek noktalı çaprazlamaya dayalı modelde bu kromozomların çaprazlanması sonucu tekrarlanan genler olması durumunda popülasyona dahil edilmeyeceği ifade edilmesine rağmen bunun nasıl engelleneceği ile ilgili herhangi bir çalışmaya rastlanmamıştır. Klasik tek noktalı çaprazlamaya dayalı modelde kromozomların çaprazlanmasından kaynaklı

tekrarlanan genler dikkate alınmamıştır. Klasik tek noktali çaprazlamaya dayali modeldeki çaprazlama biçimi ile yapıldığında karşımıza çikan hatali gen sayilari Tablo 5.1’ de verilmiştir. Tablodaki sonuçlara göre nesil sayısının artması ile hatali gen üretimi artmıştır. Yapılan tez çalışmasında tekrarlanan yapının oluşmaması için bir çözüm önerisi olan crossover Şekil 5.9’ da sunulmuştur. Hatali genlerden dolayı Klasik tek noktali çaprazlamaya dayali yöntemin sonuçları karşılaştırmak için hiçbir tabloya dahil edilmemiştir.

Tablo 5.1 Klasik Tek Noktalı Çaprazlamaya Dayalı Yöntem Hatalı Gen Analiz Sonuçları

Nesil Sayısı	Mutasyon Oranı (%)		
	%0,5	%1	%5
500	185	211	269
1000	128	275	291
2000	191	178	352

5.2 Tek Boyutlu Sınav Yerleştirme için İyileştirilmiş Genetik Algoritma Modeli

Genetik Algoritmanın kullanıldığı tez çalışmasında parametrelerin belirlenmesi gerekmektedir. Öncelikle popülasyon büyüklüğü, çaprazlama oranı ve mutasyon oranının değerleri belirlenir. Popülasyonun aktif olarak görev alması için toplam gen sayısı kadar ilk değerlerini varsayılan olarak alması sağlanır. Popülasyonun her geni öğrencinin ID sini temsil etmektedir. Sınav salonlarına sırasıyla rastgele öğrenci ID si alınarak kromozom oluşturulur. Kromozom yapısının sınıf bilgileri kromozom yapısı içinde yer almaz. Kodlama içinde rastgele seçilen öğrenciler sırasıyla döngü içinde salonlara ve sıralara tek tek atanarak işlemlerden geçer. Böylece sınıf bilgileri kodlama ile kromozom yapısından çıkarılmış olur. İyileştirilmiş tek boyutlu modelde kromozom yapısında küçük değişikliğe gidilerek öğrencileri sınıflara yerleştirme yerine işlemleri yerine sınav salonlarına öğrencileri yerleştirme yoluna gidilmiştir. Şekil 5.8 de kromozomun gen yapısı verilmiştir.

C1	C1	C1	C1	C1	C1	C2	C2	C2	C2	→ Sınıf ID
1	1	1	2	2	2	1	1	2	2	→ Sütun No
1	2	3	1	2	3	1	2	1	2	→ Satır No
S9	S7	S2	S5	S6	S10	S8	S1	S4	S3	→ Öğrenci ID

Şekil 5.8 İyileştirilmiş Tek Boyutlu Modelin Gen Yapısı

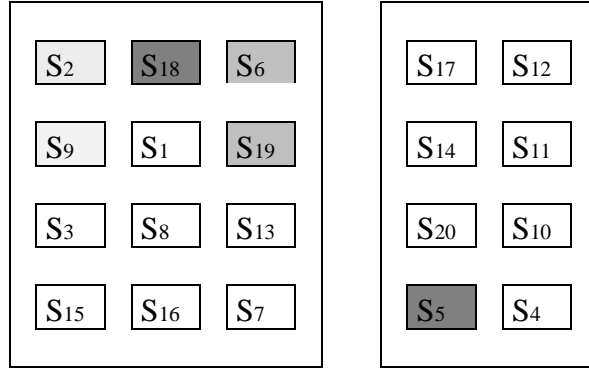
Kromozom yapısı öğrenci sayısından dolayı oldukça geniş bir söz dizimi sahiptir. Bu söz dizimin de ayrıca sınıf bilgilerinin eklenmemesi için programın kod kısmında her sınav salonuna sırasıyla rastgele öğrenciler atanmıştır. Bu sayede kromozomun gen yapısı içinde ek veri tutmadan kodlama içinde ek bir işlem yaparak kromozom yapısı sadece öğrenci numaralarının temsil edildiği benzersiz numaralardan oluşan bir yapı halini almıştır. Bir matris düzeni oluşturularak her öğrenci numarası için birden fazla bilgi tutulması sağlanmıştır. Bu matris düzeni ile sağlanan yapı bir veri tabanını andırmaktadır. Her bir öğrencinin T.C. kimlik numarasının olduğu, doğum yeri, ikametgâh adresi ve okuduğu bölüm/program yer almaktadır. Uygulamada kromozom yapısının nasıl oluşturulacağı, gen yapısının nasıl belirleneceği çok önemlidir. Sistem kaynağını en az kullanarak doğru hesaplama yapıyor olması belirlenecek olan yapıya bağlıdır. Gen yapısı belirlenirken sınıf matris düzeni dikkate alınarak öğrenci sayısından kaynaklı çok büyük boyutlu bir kromozom yapısının oluşmaması için öğrenci ID si gen olarak temsil edilmiştir. Sınıf numarası ve öğrencinin hangi satır ve sütunda oturduğunu bildiren sayılar kromozom yapısını oldukça genişleteceğinden yazılım içerisinde kodlar sayesinde çözümlenerek kromozom yapısında indis bilgisi şeklinde düzenlenmiştir. İyileştirilmiş tek boyutlu modelin örnek kromozom yapısı şekil 5.9’da verilmiştir.

C1	C1	C1	C1	C1	C1	C1	C1	C1	C1	C1	C1	C2	C2	C2	C2	C2	C2	C2	C2
1	1	1	1	2	2	2	2	3	3	3	3	1	1	1	1	2	2	2	2
1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
S2	S9	S3	S15	S18	S1	S8	S16	S6	S19	S13	S7	S17	S14	S20	S5	S12	S11	S10	S4

Şekil 5.9 İyileştirilmiş tek boyutlu yöntemin kromozom yapısı

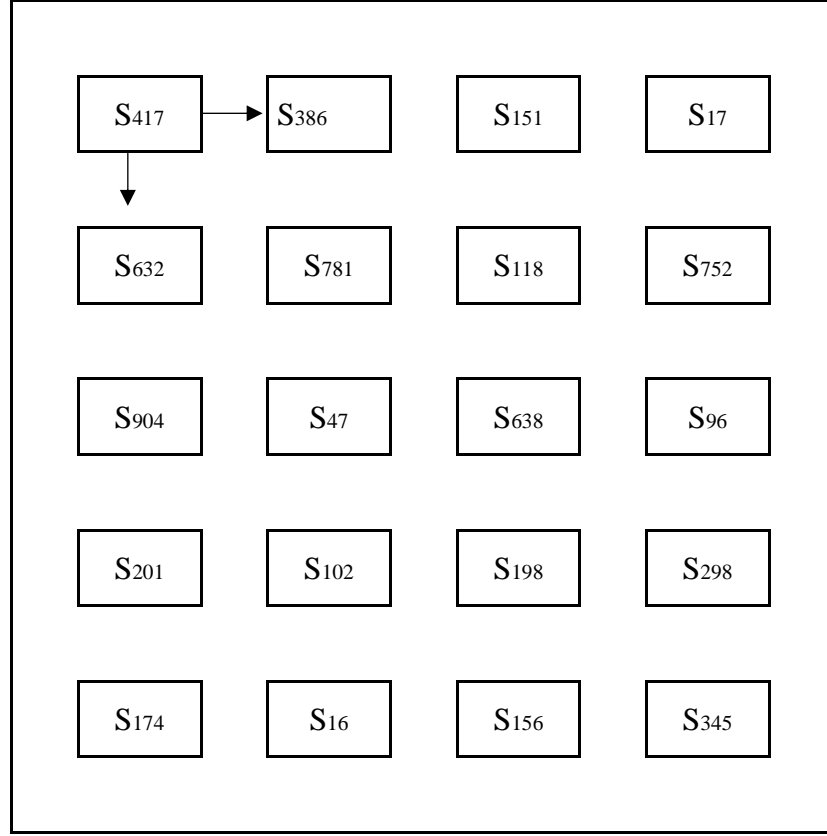
Öncelikle sınıf bilgilerinin olduğu veriler çekilerek sınıf mevcutları ve oturma düzenlerine göre kodlamada rastgele seçilen öğrencinin sırasıyla ilk salon ve sıraya oturması sağlanır. Salonlara yerleştirilen rastgele seçilen öğrencilerin tekrar seçilme ihtimali vardır. Bu seçimi göz önüne alarak kromozom yapısı içinde aynı genin yani

öğrencinin tekrar seçilmemesi sağlanmıştır. Aynı öğrenci tekrar seçilmesi durumunda işlem pas geçilir ve tekrar rastgele seçilmesi sağlanır. Bu sayede ilk neslin tüm kromozom yapısı içinde öğrenciler bütün sıralara tek tek yerleştirilmiş olur. Popülasyonun büyüklüğü kadar ilk nesil rastgele oluşturulur. Tek boyutlu kromozom yapısının sınıf düzeni şekil 5.10’da verilmiştir. Yakınlık ilişkisi bulunan kişilerin arka arakaya oturduğu görülmektedir.



Şekil 5.10 İyileştirilmiş tek boyutlu modelin C1- C2 Sınıf düzenleri

Öğrencilerin bilgilerinin bir veri tabanı gibi kullanıldığı matris üzerinden ilk neslin kromozom yapısı aynı düzende geçici bir matrise alınır. Sınıf bilgilerinin bulunduğu verilerden yararlanarak sırasıyla yerleştirilen her bir genin yakınındaki genler ile kontrolü sağlanır. Bu genler arasında bir yakınlık ilişkisi bulunursa ceza puanı verilmektedir. Kontrol edilen genin oturma düzeninde hemen arkasındaki ve sağındaki gen ile yakınlık ilişkisi kontrol edilir ve bir yakınlık tespit edilmesi durumdan ceza puanı alır. Tüm kromozom yapısı içinde bulunan genler sınıf oturma düzeninde hazırlanmış kodlar sayesinde tek tek kontrol edilir ve kromozoma ait toplam bir ceza puanı belirlenir. Her kromozom yapısına ait bir uygunluk değeri hesaplanır. Uygunluk değeri için her kromozomun genler arasındaki yakınlıklara verilen ceza puanlarına bakılır. Her kromozomun uygunluk değeri toplam ceza puanına göre belirlenir. Kromozom ceza puanı belirlenirken sınıflara rastgele yerleşen her bir öğrencinin hemen arkasındaki ve sağındaki kişi ile arasında bulunan yakınlıkları kontrol ediliyor. Şekil 5.11’de verilen sınıfın ilk sırasında oturan 417 numaralı öğrencinin hemen arkasında 632 numaralı öğrenci ve hemen sağ tarafında bulunan 386 numaralı öğrenci arasında bir ilişkinin olup olmadığına bakılıyor. Sınıftaki tüm öğrenciler bu yöntemle kontrol ediliyor. Sınıfın en sağında kalan kişiler sadece arkasında bulunan öğrenciler ile yakınlıklarına bakılıyor.



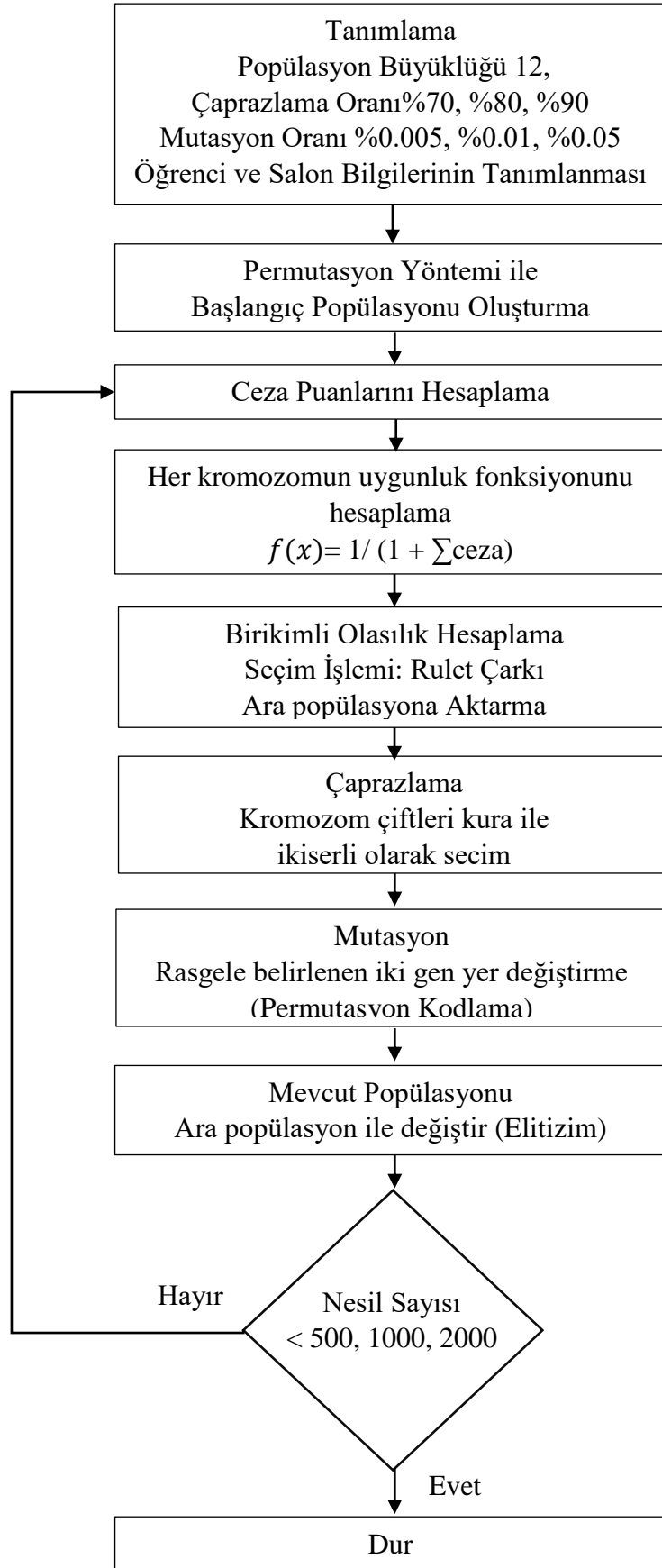
Şekil 5.11 Tek Boyutlu Yöntemde kromozom ceza puanı belirlemede kullanılan öğrenci yakınlıkları

İlk neslin herhangi bir kromozomun uygunluk değeri oranı tüm kromozomlarının uygunluk değerine bakılarak belirlenir. Tüm kromozomların uygunluk değerleri toplanır ve her kromozomun tüm kromozomlar arasındaki uygunluk değerinin oranı hesaplanır. Böylece ilk nesilde tüm kromozomların uygunluk değeri oranı kümülatif değerlerin bulunması için toplanırlar. Kümülatif yapının 0 ile 1 arasında bir değeri olur. Kümülatif değerler bir sonraki aşamada hayatta kalacak kromozomların seçiminde belirleyici olacaktır. Rulet çarkı yöntemi kullanılarak popülasyondaki hangi kromozomun hayatta kalacağı kümülatif değerlere diğer adıyla birikimli oran değerlerine göre belirlenir. Bu seçim 0 ile 1 arasında rastgele üretilen bir sayı üzerinden belirlenerek kromozomların bazıları hayatta kalacaktır. Kromozomların bazılarının ise yaşam döngüsü sona erecektir. Hayatta kalan kromozomların bir çaprazlama noktası belirlenerek yavru bireyler elde edilir. Çaprazlama noktası için rastgele bir seçim yapılması gerekir. Kromozom yapısında bulunan gen sayısı kadar rastgele bir değer üreterek çaprazlama noktası seçim işlemi yapılır. Hangi kromozom çiftinin çaprazlanacağı da önemli bir konudur. Bu işlemi gerçekleştirmek için bir

dizide saklanmak üzere popülasyon büyüklüğü kadar yani popülasyonda kaç kromozom varsa her kromozom sırası bir sayı ile temsil edilmek üzere rastgele sayı üretilir. Dizi içinde rastgele üretilen bu sayılar sırasıyla ikişerli bir şekilde olmak üzere çaprazlamaya tabi tutulacak kromozom çiftinin adresini ya da yerini bize verir. Örneğin popülasyon büyüklüğü 8 kromozomdan oluşan bir yapıda rastgele 7-2-5-8-1-4-3-6 sayılar üretildiğini kabul edelim. Çaprazlanacak kromozomların ilk çifti 7 ile 2 numaralı kromozomlar olur. Üretilen rastgele sayılar sırasıyla her bir kromozom çiftinin yerini bize söyler. Bu sayede hangi kromozomun hangi kromozom ile çaprazlanacağı belirlenir. Çaprazlanacak olan kromozom çiftinin çaprazlama noktasına göre ilk kromozomun çaprazlanacak noktaya kadar olan genler ile ikinci kromozomun çaprazlama noktasından olan genlerden yavru bireyler elde edilir. Aynı şekilde ilk kromozomun çaprazlama noktasından sonraki genler ile ikinci kromozomun ilk genleri birleşerek ikinci bir yavru birey elde edilir.

Daha sonraki aşamada rulet çarkı yöntemi ile birkaç nesil sonra istenen kromozomların seçilmesi sağlanır fakat kısır bir döngü olarak devam eder. Çünkü aynı özelliklerde kromozomlar meydana gelir. Genetik Algoritmalarda bu kısır döngünün kırılması için kromozom yapısı mutasyon işlemine tabi tutulur. Mutasyon işlemine tabi tutulacak kromozom rastgele bir sayı ile belirlenir. Rastgele üretilen sayı belirlenen mutasyon oranından daha yüksek ise kromozom mutasyon işlemine tabi tutulmaz. Mutasyon oranı oldukça düşük tercih edilmektedir. Tüm kromozomlar bu işlemlerden geçtikten sonra bir sonraki nesile aktarılırlar. Uygunluk değeri daha iyi olan kromozomlar bir sonraki nesilde tekrar tekrar aynı işlemlere tabi tutularak varlığını sürdürür.

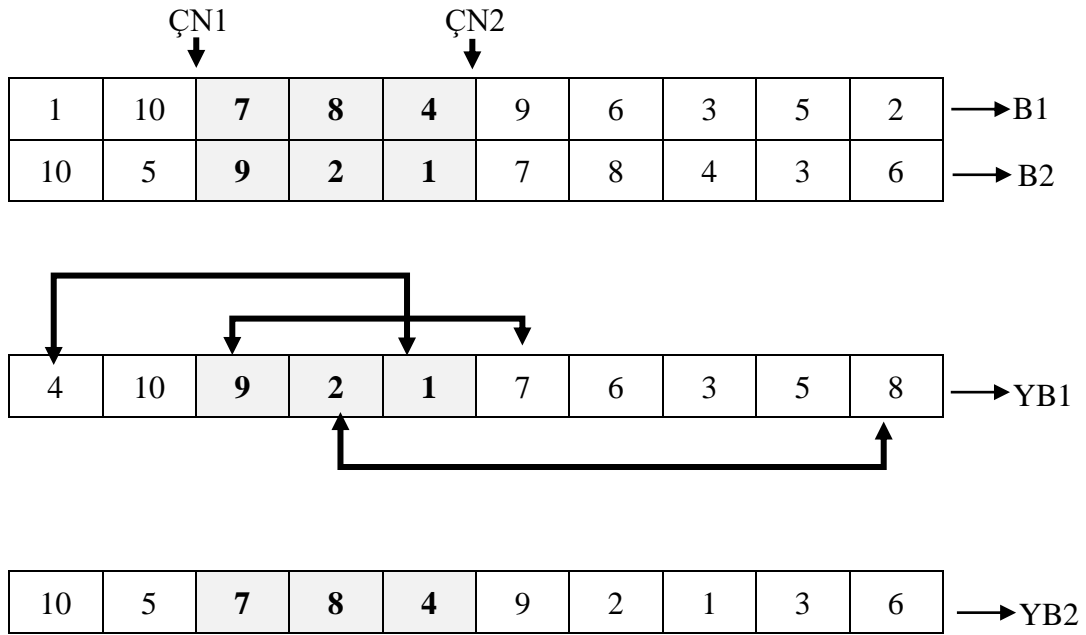
Bütün bu işlemlere tabi olmayan tek bir kromozom vardır. Bu kromozom hiçbir çaprazlama mutasyon işlemine tabi tutulmadan sürekli olarak her nesilde varlığını sürdürür. Neslin en iyi kromozomu yani uygunluk değeri en iyi olan kromozom yapısı elit olarak seçilir ve her nesilde en iyi elit yapı sürekli olarak hayatta kalır. Belirlenen nesil sayısı kadar işlemler tekrarlanır ve en iyi sonucu veren sınıf oturma düzenin ekrana verir. Tez çalışması için genetik algoritma kullanılarak geliştirilen yazılımın akış diyagramı şekil 5.12'de verilmiştir.



Şekil 5.12 Tek ve Çok boyutlu yöntemlerin akış diyagramı

5.2.1 İyileştirilmiş Tek Boyutlu Yöntem

Genetik Algoritmanın sınavlarda öğrenci yerleşim problemine doğrudan uygulanmasında çaprazlama sonucunda kromozom içerisinde bazı genler tekrarladığı görülmektedir (Bulut ve Subaşı, 2015). Tekrarlama sonucunda aynı kişinin birden fazla noktaya yerleştirilmesi gibi gerçek dünyaya uygulanması mümkün olmayan yerleşim planları ortaya çıkmaktadır. Veri kümesi karmaşıklıktıkça bu yanlış yerleşim sayısı da çok artmaktadır. Bu tekrarlanan genlerin önüne geçmek için ikinci kromozomun çaprazlanacak geni birinci kromozom içindeki yeri bulunarak ilk kromozomun çaprazlanacak genine aktaran iyileştirilmiş bir çaprazlama modeli geliştirilmiştir. Bu sayede aynı kromozomda aynı genin tekrarlamasının önüne geçilmiş olur. Şekil 5.12’de çaprazlama sonucu kromozomda oluşan gen tekrarlarının nasıl önüne geçildiği anlatılmaktadır. Geliştirilen yönteme kısaca *iyileştirilmiş tek boyutlu yöntem* ismi verilmiştir.



Şekil 5.13 İyileştirilmiş tek boyutlu yöntemde kısmi eşleşmiş çaprazlama

Yukarıdaki şekil 5.13’te iki kromozom karşılıklı çaprazlama işlemine tabi tutulacaktır. Birey1 de yer alan 7,8 ve 4 nolu genler Birey2 de yer alan 9,2 ve 1 nolu genler ile çaprazlanacaktır. Birey2 de bulunan 9 Birey1 deki 7’nin yerine geçecektir. Birey1 de 9’un bulunduğu yere Birey1 deki 7 gelecektir. Aynı işlemler diğerleri içinde söz konusudur. Böylece kromozom yapısı içinde aynı genlerin tekrarlanmasının önüne geçilir.

5.2.2 İyileştirilmiş Tek Boyutlu Yöntemde Karşılaştırmalı Sonuçlar

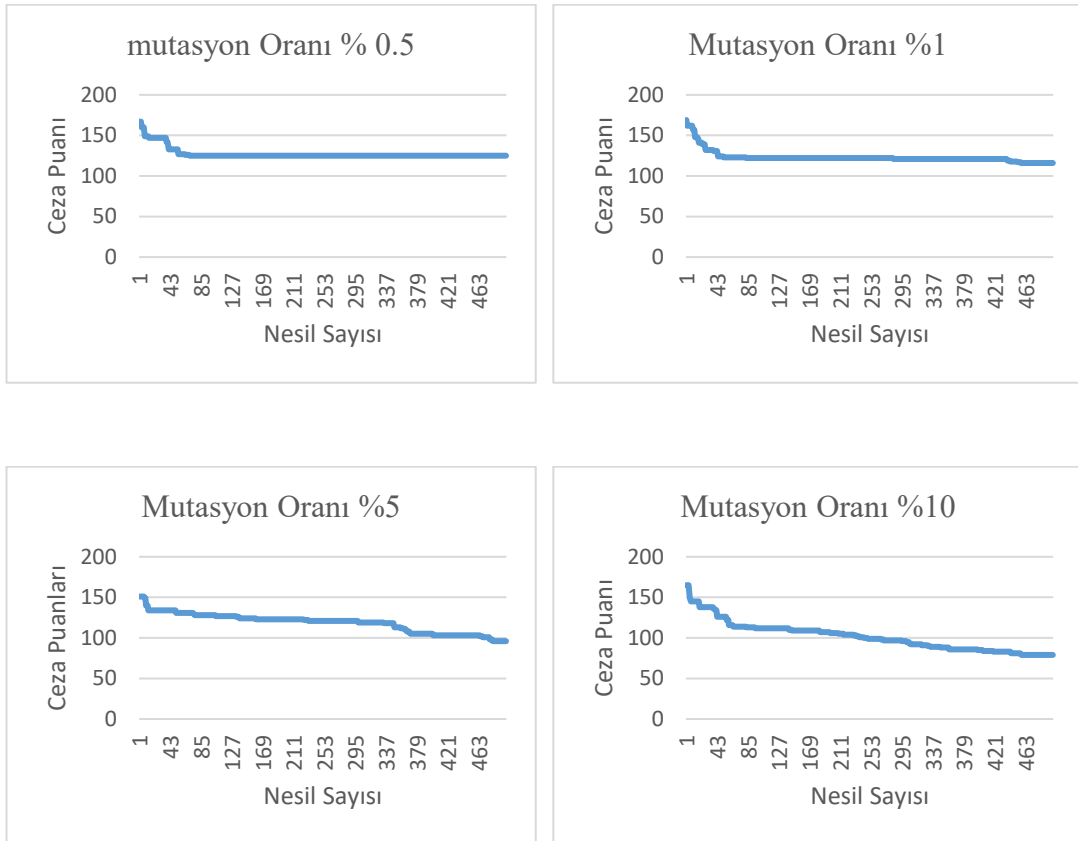
Genetik algoritma yöntemi kullanılarak yapılan iyileştirilmiş tek boyutlu yöntem için nesil sayısı, popülasyon büyüklüğü, çaprazlama oranı ve mutasyon olasılığı gibi her bir parametre değeri için analiz sonuçları karşılaştırılmıştır. Bu yöntemde de klasik tek noktalı çaprazlamaya dayalı modelde de kullanılan öğrenci ve salon bilgileri verileri kullanılmıştır. Genetik algoritma parametre değerleri üzerinde en iyi ve en uygun parametre değeri üzerinde ortak bir yargı bulunmamakla beraber David Schaffer genetik algoritmanın performans üzerindeki etkisi üzerine yaptığı deneysel çalışmalarda çaprazlama oranı için 0.75-0.95 aralığını, mutasyon oranı için 0.005-0.01 aralığını ve popülasyon büyüklüğü için 10-20 değerini önermiştir (Aksu,2008). Tez çalışmasında popülasyon büyüklüğü olarak 12 değeri en uygun değer olarak belirlenmiştir. Çaprazlama oranı %70 ile %90 arasında ve mutasyon olasılığı ise %0,5 ile %1 arasında farklı değerler tercih edilmiştir. Klasik tek noktalı çaprazlamaya dayalı modelde kullanılan öğrenci ve salon bilgileri aynı şekilde iyileştirilmiş tek boyutlu modelde de kullanılmıştır. Tablo 5.2' de İyileştirilmiş tek boyutlu modelin belirtilen parametrelere göre analiz sonuçları listelenmiştir.

Tablo 5.2 İyileştirilmiş Tek Boyutlu Yöntemin Ceza Puanları Analiz Sonuçları

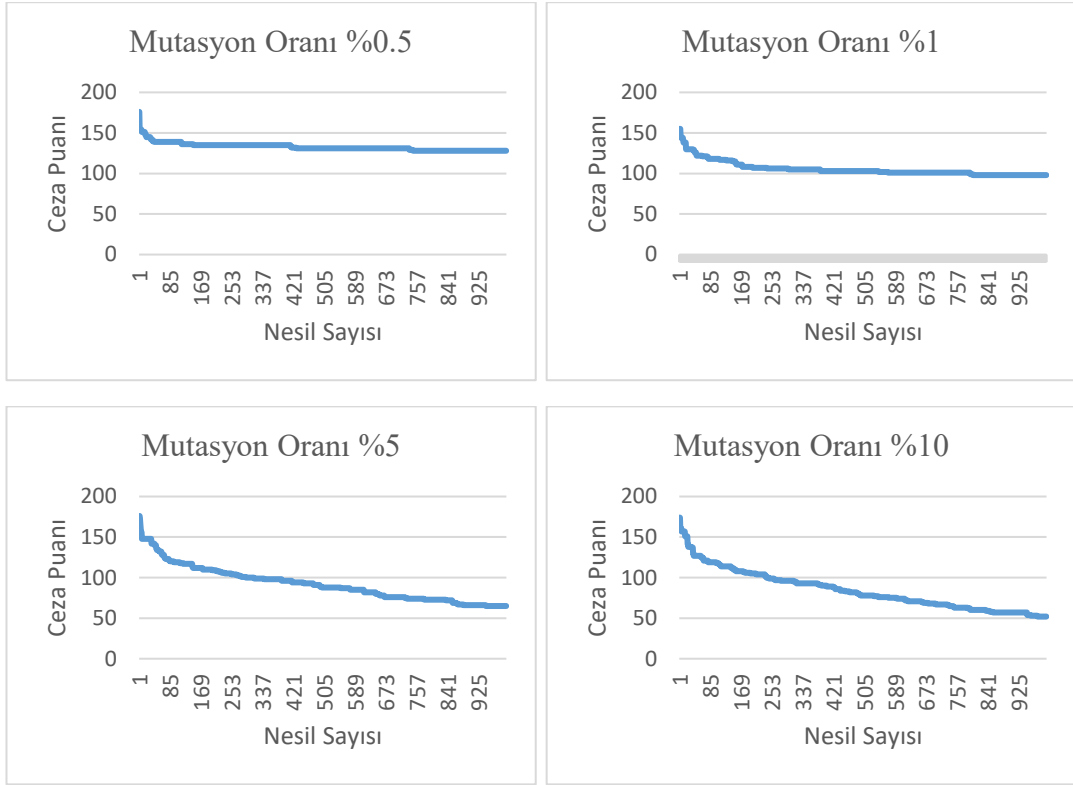
Çaprazlama Oranı(%)	Mutasyon Oranı (%)	Nesil Sayısı		
		500	1000	2000
70%	0.5	120	121	97
	1	113	109	110
80%	0.5	119	115	104
	1	96	98	86
90%	0.5	115	105	100
	1	105	107	91

Genetik algoritma rasgele aramaya dayalı bir algoritma olduğu için çaprazlama oranının artması her zaman için daha iyi bir sonuç vermeyebilir. Kromozom yapısı uzun olan problemlerde bu oran düşük bir değer olarak tercih edildiğinde algoritma kısır bir döngüye girmektedir. Mutasyon olasılığı arttıkça problemin çözümüne olumlu bir katkısı olmasına karşın genetik arama rastsal bir aramaya dönüşmektedir. İyileştirilmiş tek boyutlu yöntem ile test edilen algoritmanın kısır döngüsünü bozmak için mutasyon oranını literatürde belirtilen olasılıktan oldukça yüksek tutulmuş ve sonuçlar karşılaştırılmıştır. Ayrıca nesil sayısının artması uygunluk fonksiyonunu istenen en iyi değere ulaşmasını sağlamaktadır. Yapılan araştırmalar sonucunda kullanılan parametrelere göre aşağıdaki gibi grafikler elde edilmiştir.

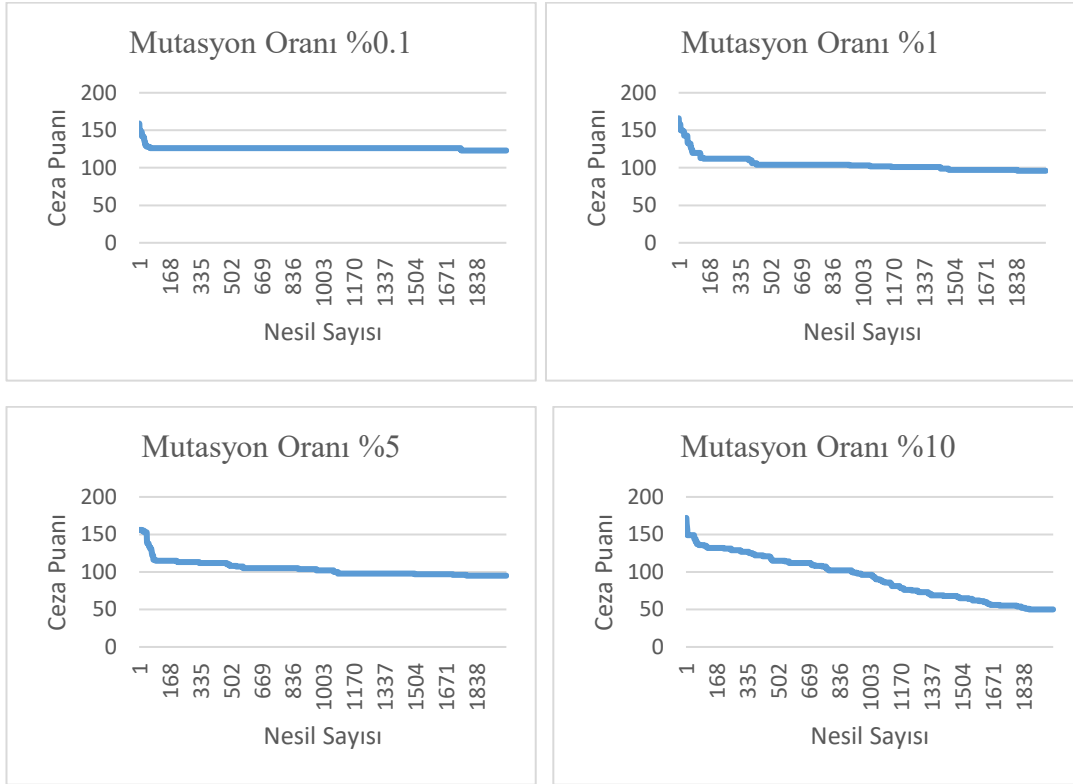
A. Nesil Sayısı 500 Çaprazlama Oranı 90



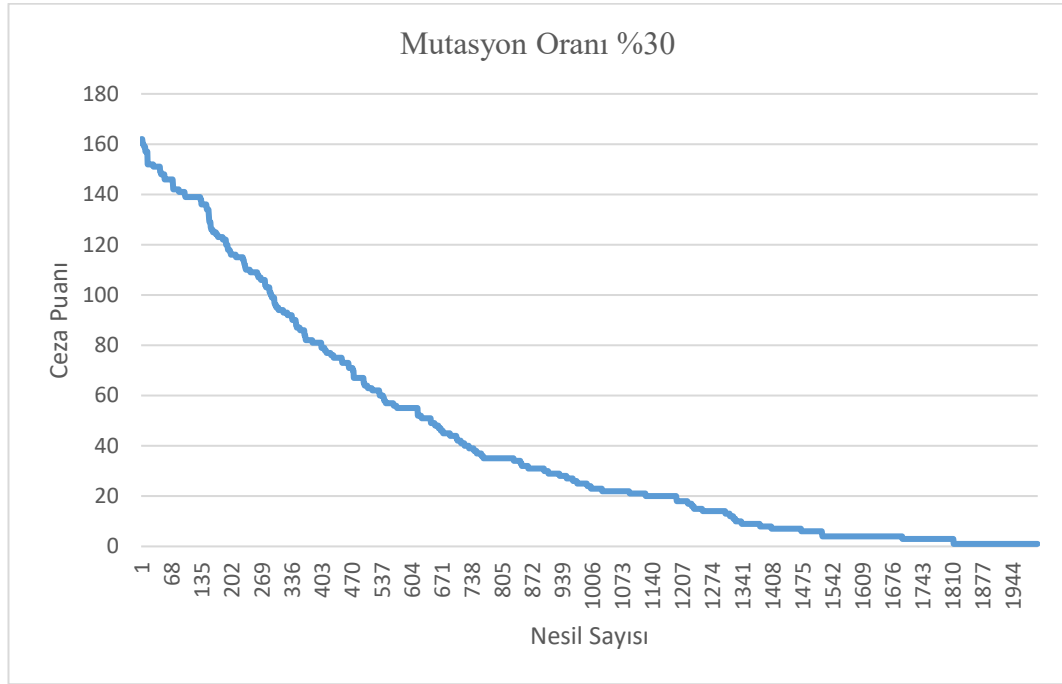
B. Nesil Sayısı 1000 Çaprazlama Oranı 90



C. Nesil Sayısı 2000 Çaprazlama Oranı 90



D. Nesil Sayısı 2000 Çaprazlama Oranı %90 Mutasyon Oranı %30



Şekil 5.14 İyileştirilmiş tek boyutlu modelde ceza puanlarının parametrelere göre değişim grafiği

Grafikler incelendiğinde tüm analizlerde yapılan çaprazlama oranı %90 ve popülasyon büyüklüğü 12 olarak belirlenmiştir. Nesil sayısı 500 iken mutasyon oranının artması ile en iyi ceza puanları 80-100 arasında bir değer almaktadır. Nesil sayısı 1000 olduğu durumda en iyi ceza puanı 52-65 arasına olduğu görülmektedir. En iyi sonucu nesil sayısı 2000 olduğu durumda ancak mutasyon oranı %30 iken alabilmektedir. İyileştirilmiş tek boyutlu yöntem ile hazırlanan algoritma düşük mutasyon olasılığı ile test edildiğinde olumlu sonuçlar alınmasına rağmen istenen ve beklenen sonuca ulaşamamıştır. Tez çalışmasında kullanılan genetik algoritmanın kromozom yapısı uzun olduğu için ancak mutasyon olasılığı literatürde geçen oranların oldukça üstünde seçildiğinde problemin çözüme ciddi bir etkisi olmuştur. Genetik algoritma problemlerinde mutasyon olasılığı %0,5 ile %1 arasında olumlu sonuçlar alınırken iyileştirilmiş tek boyutlu yöntemde %30 civarında oldukça yüksek bir mutasyon olasılığı ile ancak başarılı bir sonuç elde ettiğini görüyoruz. İyileştirilmiş tek boyutlu yöntemin yüksek mutasyon oranları ile test edilmesinin nedeni problemin çözümüne olan katkısıdır. Bu yöntemde popülasyon büyüklüğü için 12 değeri ve çaprazlama oranı %90 olarak seçildiğinde daha iyi sonuçlar alınmaktadır. Yapılan analiz sonuçları

bu durumu kanıtlar niteliktedir. Şekil 5.14'te verilen parametreler ile ancak en iyi uygunluk değerine ulaşılmaktadır.

İyileştirilmiş tek boyutlu modelde genetik algoritma parametrelerinin analiz sonuçlarına etkisi Tablo 5.3'te verilmiştir. Bu tablo bize çaprazlama oranının ve mutasyon olasılığının artmasının daha iyi sonuçlar verdiğini göstermiştir. Fakat genetik algoritmalarda her zaman mutasyon oranının artması olumlu sonuçlar vermemektedir. Bu yöntemde çaprazlama oranı ve mutasyon olasılığı arttıkça ceza puanlarının düştüğü görülmektedir. İyileştirilmiş tek boyutlu yöntem için en iyi oturma düzenini sağlayan parametreler incelendiğinde çaprazlama oranı %90 mutasyon olasılığının %30 ve nesil sayısı 2000 olduğu durumda en iyi oturma düzeni sağlanmaktadır. Tablo bize nesil sayısının artması ile istenen çözüme yaklaşıldığını gösteriyor. Nesil sayısı 2000 olduğu durumda neredeyse hiç kimse arasında yakınlık tespit edilmiyor. Mutasyon oranının artması sonuca olumlu katkısı olsa da işlem olarak rassal bir aramaya dönmektedir. Bu durum genetik algoritmanın temel mantığı ile uyumsuzdur.

Tablo 5.3 İyileştirilmiş tek boyutlu yöntemde genetik algoritma parametrelerinin analiz sonuçlarına etkisi

Nesil Sayısı	Çaprazlama Oranı (%)	Mutasyon Oranı (%)					
		0.1	0.5	1	5	10	30
500	80	118	119	96	99	83	48
	90	115	115	105	96	80	50
1000	80	121	115	98	67	40	11
	90	117	105	107	60	46	17
2000	80	112	104	86	36	9	1
	90	106	100	91	38	13	0

5.3 Çok Boyutlu Sınav Yerleştirme için Katsayı Belirlemeye Dayalı Genetik Algoritma Modeli

Sınav yerleşiminde öğrencinin okuduğu bölüm, doğum yeri ve ikamet adresi gibi birçok farklı boyut dikkate alınmalıdır. Ancak literatürde bu konuda yapılan çalışmalarda tek bir boyut kullanılmıştır; bkz. Bölüm 5.1. Tez kapsamında genetik algoritma üzerindeki ilk iyileştirmede de öğrenciler arasındaki tüm ilişkiler tek bir boyut gibi alınmıştır; bkz. Bölüm 5.2.

Yerleşim probleminin çözümünde farklı boyutların etkisini yansıtabilmek için katsayı belirlemeye dayalı bulanık mantıkta da kullanılan, bitişik kategori katsayı değerlerinin birbirinin en az iki katı olması kuralına dayalı olarak algorithmada iyileştirmeler yapılmıştır (McConnell S. 2006). k katsayı b boyut olmak üzere yakınlık hesabı denklem 5.2'deki formülle yapılmıştır.

$$yakınlık = k_1 * b_1 + k_2 * b_2 + \dots k_n * b_n \quad k_n = 2 * k_{n-1} \quad (5.2)$$

Ceza puanlarını belirlemede kullanılan parametrelerin etkisini yansıtmak için katsayı belirlemeye dayalı yöntem *Katsayı belirlemeye dayalı çok boyutlu yöntem* ismi verilmiştir.

İyileştirilmiş tek boyutlu yöntemde tüm yakınlık ilişkileri aynı benzerlikte kabul edilip sonuçlar incelenmiştir. Önerdiğimiz yöntemle birlikte katsayı belirlemeye dayalı çok boyutlu yöntem katsayısı modeli aynı Excel verileri ile uygulanarak sonuçlar tekrar analiz edilmiştir. Çoklu parametrelerin etkisini yansıtmak için veri kümelerini birbirinden ayırmada kullanılan katsayıları ikinin katı olarak seçme yöntemi kullanılmıştır. Yakınlık katsayıları aynı programda okuyan öğrenciler için çarpanı 4, aynı bölgede ikamet edenler için 2 ve aynı doğum yerine sahip olanlar için 1 kabul edilmiştir. Bu yöntem ile ceza puanı belirlemede kullanılan parametrelerin etkisini yansıtmak için katsayı belirlemeye dayalı çok boyutlu yöntemin analiz sonuçları Tablo 5.4' te verilmiştir.

Tablo 5.4 Katsayı Belirlemeye Dayalı Çok Boyutlu Yöntem Ceza Puanları Analiz Sonuçları

Çaprazlama Oranı(%)	Mutasyon Oranı (%)	Nesil Sayısı		
		500	1000	2000
70%	0.5	51	16	1
	1	51	13	0
80%	0.5	57	14	0
	1	51	12	0
90%	0.5	57	14	0
	1	49	11	0

Benzerlik katsayıları dikkate alındığında katsayı belirlemeye dayalı çok boyutlu yöntemin iyileştirilmiş tek boyutlu yönteme oranla oldukça daha iyi sonuçlar vermektedir. Bu durum dikkate alınarak analiz sonuçları yakınlık ilişkisine göre incelendiğinde sonuçların yaklaşık en az 2,33 oranında istenmeyen durumların bu yöntemle daha çok iyileştirildiği görülmektedir.

6. SONUÇ VE ÖNERİLER

Bu çalışmada optimizasyon problemlerinde kullanılan yöntemler tanıtılmış ve bir sonraki bölümünde genetik algoritmanın tarihçesi, temel yapısı ve çalışması hakkında bilgi verilmiştir. Genetik Algoritmanın adımları, seçim mekanizması, çaprazlama ve mutasyon operatörleri ve kullanılan parametrelerin öneminden bahsedilmiştir. Bir sonraki bölümde merkezi sınavlarda yakınlık ilişkisi bulunan öğrencilerin arasında istenmeyen yardımlaşmaların önüne geçmek için öğrencilerin birbirlerine uzak oturmalarını sağlamıştır. Bu tez çalışması ile en iyi oturma düzeninin Genetik Algoritma yöntemi kullanılarak hazırlanabileceği gösterilmiştir. Mardin Artuklu Üniversitesi Uzaktan Eğitim Uygulama ve Araştırma Merkezinden alınan öğrencilere ait doğum yeri, okuduğu program ve ikamet bilgilerinin bulunduğu veriler ile birlikte sınavların yapılacağı bina, salon ve oturma düzenlerine ait verileri kullanarak en iyi oturma düzenini sağlayacak yazılım python programlama dili kullanılarak geliştirilmiştir.

Klasik tek noktalı çaprazlamaya dayalı modelde çaprazlama sonucu oluşan tekrarlayan genler için çözüm önerisi sunulmuştur. Tez çalışmasında genetik algoritma parametrelerinin sonuca etkisi ayrıntılı olarak incelenmiştir. Bu parametrelerden mutasyon oranı %0,1 ile %1 arasında farklı oranlar verilerek karşılaştırılmış ve bu oranın artması ile çok daha iyi sonuçlar verdiği gözlemlenmiştir. İyileştirilmiş tek boyutlu yöntemde kromozom yapısının uzun olmasından kaynaklı mutasyon oranının yüksek seçilmesi rastsal bir aramaya dönüştürse de ceza puanını azalmıştır. Her iki modelde çaprazlama oranı %90 olarak seçildiğinde daha iyi sonuçlar verdiği gözlemlenmiştir. Nesil sayısı ilk olarak 500 seçilmiş fakat istenen sonuca ulaşamayınca artırılmıştır. Nesil Sayısı 1000 olarak arttırıldığında ise iyi sonuçlar almasına rağmen istenen ve beklenen duruma ulaşamamıştır. Çok boyutlu model yaklaşımında iterasyon sayısı 2000'e ulaştığında öğrenciler arasında yakınlık ilişkisi olan öğrencilerin yan yana oturmaması sağlanarak istenen sonuca ulaşılmaktadır.

Tek boyutlu modelde öğrencinin okuduğu bölüm, doğum yeri ve ikamet adresi gibi tüm benzerlikler aynı kabul edilmiş ve benzerlik katsayısı 1 olarak alınmıştır. Mutasyon olasılığı literatürde kabul gören değerler arasında tercih edildiğinde iyileştirilmiş tek boyutlu yöntem ile yapılan analiz sonuçlarına göre popülasyon

büyüklüğü 12, nesil sayısı 2000 ve çaprazlama oranı %80 olarak alındığında en iyi oturum düzenini sağlayacak parametreler olarak karşımıza çıkmaktadır. Nesil sayısı arttıkça probleme verilen ceza puanları azalmaktadır. Nesil sayısının artması tüm durumlarda problemin çözüme olumlu bir katkı sağlamaktadır.

Parametrelerin etkisini yansıtmak için katsayı belirlemeye dayalı uyguladığımız yöntem iyileştirilmiş tek boyutlu yöntemle oranla çok daha başarılı sonuçlar elde etmektedir. Çok boyutlu model tek boyutlu modele oranla yaklaşık 2.3 kat daha başarılı sonuçlar elde etmiştir. İyileştirilmiş tek boyutlu yöntem ile katsayı belirlemeye dayalı çok boyutlu yöntem oransal olarak Tablo 6.1’de karşılaştırmalı olarak verilmiştir.

Tablo 6.1 İyileştirilmiş Tek Boyutlu Yöntem ile Katsayı Belirlemeye Dayalı Çok Boyutlu Yöntemin Oransal Olarak Karşılaştırılması

Çaprazlama Oranı (%)	Mutasyon Oranı (%)	Nesil Sayısı 500	
		İyileştirilmiş Tek Boyutlu Yöntem	Katsayı Belirlemeye Dayalı Çok Boyutlu Yöntem
70	0.1%	128	55
80	0.1%	118	51
90	0.1%	115	49

Her iki modelde nesil sayısı 500 ve 1000 olduğu durumlar incelenmiş ve farklı mutasyon oranları ile sonuçlar analiz edilmiştir. İyileştirilmiş tek boyutlu yöntem ile katsayı belirlemeye dayalı çok boyutlu yöntem karşılaştırmalı analiz sonuçları Tablo 6.2’de verilmiştir.

Tablo 6.2 İyileştirilmiş Tek Boyutlu Yöntem ile Katsayı belirlemeye Dayalı Çok Boyutlu Yöntemin Ceza Puanlarının Karşılaştırılması

Çaprazlama Oranı(%)	Mutasyon Oranı(%)	Nesil Sayısı 500		Nesil Sayısı 1000	
		İyileştirilmiş Tek Boyutlu Yöntem	Katsayı belirlemeye Dayalı Çok Boyutlu Yöntem	İyileştirilmiş Tek Boyutlu Yöntem	Katsayı belirlemeye Dayalı Çok Boyutlu Yöntem
70	0.1%	128	51	132	16
	0.5%	120	51	121	13
80	0.1%	118	57	121	14
	0.5%	119	51	115	12
90	0.1%	115	57	117	14
	0.5%	115	49	105	11

İyileştirilmiş tek boyutlu yöntemin farklı parametrelerle oluşan popülasyondan alınan rasgele kromozom çözümleri ile yapılan test analizleri de bu oranları doğrulamaktadır. Örnekleme yapılan çalışmada alınan kromozom çözümleri python içerisinde yazılan küçük bir test kod yapısı ile iyileştirilmiş tek boyutlu yöntem ve katsayı belirlemeye dayalı çok boyutlu yöntem çözümleri test edilmiş ve sonuçları Tablo 6.3'te gösterilmiştir. Tek boyutlu yöntem sonucuna göre oluşmuş bir çözüm kümesini katsayı belirlemeye dayalı çok boyutlu yönteme göre hazırlanan kod ile test edildiğinde oldukça yüksek değerler vermektedir. Bu sonuçlar istenmeyen durumların yan yana geldiğini göstermektedir. Tablo 6.2'de verilen değerlere göre katsayı belirlemeye dayalı çok boyutlu modelin iyileştirilmiş tek boyutlu modele göre çok daha başarılı olduğu görülmektedir. Katsayı belirlemeye dayalı çok boyutlu yöntem katsayısı ile kişiler arasındaki yakınlık ilişkisi aynı kabul edilmeyerek istenmeyen durumların yan yana gelmesi engellenmektedir.

Tablo 6.3 İyileştirilmiş Tek Boyutlu Yöntem Sonuçlarının Katsayı belirlemeye Dayalı Çok Boyutlu Yönteme Göre Karşılaştırılması

A. Çaprazlama Oranı (%) 70 Mutasyon Oranı (%) 0.1

Kromozomlar	Nesil Sayısı	İyileştirilmiş Tek Boyutlu Yöntem Sonucu	Katsayı belirlemeye Dayalı Çok Boyutlu Yönteme Göre
Kromozom 1	500	131	309
Kromozom 2	1000	122	276
Kromozom 3	2000	114	261

B. Çaprazlama Oranı (%) 90 Mutasyon Oranı (%) 1

Kromozomlar	Nesil Sayısı	İyileştirilmiş Tek Boyutlu Yöntem Sonucu	Katsayı belirlemeye Dayalı Çok Boyutlu Yönteme Göre
Kromozom 4	500	120	270
Kromozom 5	1000	115	257
Kromozom 6	2000	74	157

Yaptığımız çalışma ile merkezi yapılan sınavların güvenilirliğini arttırmak için sınav esnasında birbirlerini tanıyan kişilerin olası yardımlaşmaların önlenmesi amaçlanmaktadır. Genetik Algoritma kullanılarak aynı anda binlerce kişinin yapıldığı sınavlar için önerilen model sayesinde en iyi oturum düzeni kısa bir sürede sağlanacaktır. Önerdiğimiz model daha çok üniversitelerin uzaktan eğitim yoluyla verilen öğretim programlarının merkezi sınavlarında çok rahat kullanılabilir. Aynı şekilde Milli Eğitim Bakanlığının veya ÖSYM'nin daha az katılımlı sınavlarında da kullanılacak tarza geliştirilmiştir. Tez çalışması ile sunduğumuz model Türkiye genelinde yapılan merkezi sınavlarda kullanılmak üzere geliştirilebilir.

Yapılan çalışma ile en iyi oturum düzeninin sağlanması problemine en etkin çözümler aranmıştır. Bu çalışma esas alınarak yapılabilecek araştırma önerileri şu şekildedir.

- ✓ Farklı bir algoritma yöntemi ile problemin çözümüne etkilerin araştırılması
- ✓ Geniş katılımlı sınavlar için daha hızlı işlem yapabilecek yöntemlerin araştırılması
- ✓ Yazılım üzerinde yapılabilecek iyileştirmelerin çözüme olan katkılarının araştırılması
- ✓ Öğrencilerin soyadına göre ayrı bir ceza puanı belirlenerek çözüme olan katkısının araştırılması
- ✓ Türkiye genelinde yapılan merkezi sınavlarda kullanılmak üzere geliştirilmesi

Önümüzdeki dönemde yukarıdaki maddeler ışığında tez çalışmasının kapsamının genişletilmesi planlanmaktadır.

7. KAYNAKLAR

- Adewuya, A.A. (1996). New methods in Genetic Search with Real Valued Chromosomes. Master's Thesis, Cambridge: Massachusetts Institute of Technology.
- Aksu,Ö.(2008). Yeni Bir Paralel Genetik Algoritma Modeli Ve Analog Devre Tasarımına Uygulanması Yüksek Lisans Tezi, Kayseri: Erciyes Üniversitesi, Fen Bilimleri Enstitüsü,38s.
- Alander, J.T. (1997). "An Indexed Bibliography of Genetic Algorithms", Dept. Of Information Technical and Production Economics, Univ. of Vaasa. Report Series. 94-101.
- Angeline, P.J. (1995). "Evolutionrevolution: An introduction to the special track on genetic and evolutionary programming" IEEE Expert Intelligent Systems and their Applications 10. June pp.6-10.
- Bal, H. (1995). "Optimizasyon Teknikleri". Ankara: Gazi Üniversitesi Yayınları, 3-7.
- Balcı, E. ve Tekkaya, C. (2000). Ölçme ve Değerlendirme Tekniklerine Yönelik Bir Ölçeğin Geliştirilmesi, Hacettepe Üniversitesi Eğitim Fakültesi Dergisi,18, 42-50
- Balcı, Ali. (2010). Sosyal Bilimlerde Araştırma Yöntem, Teknik ve İlkeler. Ankara: Pegem Yayınevi, s106-110.
- Bardadym, V. A. (1996). Computer Aided School and University Timetabling: The New Wave, in E. Burke and P. Ross (Eds.) The Practice and Theory of Automated Timetabling I (PATAT 1995, Edinburgh, Aug/Sept, selected papers) (Lecture Notes in Computer Science, Vol. 1153). Springer, 22-45.
- Beasley, D.,Bull, D.R., Martin, R.R. (1993). "An overview of Genetic Algorithms: Part1-Fundamentals", Evolutionary Computation. 15:4,170-181.
- Beasley, D.,Bull,D.R., Martin,R.R. (1993). "An Overview of Genetic Algorithm: Part1-Fundamentals", Univ. of Michigan. 15: 2 258-269.
- Binbasıoğlu, C. (1983). Eğitimde ölçme ve değerlendirme. Ankara: Binbasıoğlu Yayınevi.
- Bridges, C.L, Goldberg,D.E. (1987). "An Analıysis of Reproductionand in A Binary Coded Genetic Algorithm". Proceedings of the Second International Conference on Genetic Algorithms, Cambrige. MA. 9-13.
- Bulut,F. ve Subaşı,S. (2015). Merkezi Sınavlar İçin Genetik Algoritmalar İle En İyi Oturma Planı. İzmir: Dokuz Eylül Üniversitesi Mühendislik Fakültesi Dergisi,17, 122-137.
- Burke, E. ve Ross, P. (1996). The Practice and Theory of Automated Timetabling I (PATAT 1995, Edinburgh, Aug/Sept, selected papers) (Lecture Notes in Computer Science, Vol. 1153), Springer.
- Curtis, H. (1975). Biology. 2nd Ed. New York, Worth Publisher.

- Çunkaş, M. (2006). Genetik Algoritmalar ve Uygulamaları Ders Notları.
- Davis, L. (1991). "Handbook of Genetic Algorithms", Van Nostrand Reinhold, NY. 385-392.
- Deris, S., Omatu, S. ve Ohta, H. (2000). "Timetable Planning Using The Constraint-Based Reasoning", Computers & Operations Research, 27: 819-840.
- Eiben A.E. ve Smith J.E. (2003). Introduction to Evolutionary Computing (1st Edition). Springer. Natural Computing Series.
- Gen, M., Cheng, R. (2000). Genetic Algorithms & Engineering Optimization, John Wiley & Sons, Inc.
- Goldberg, D.E. (1989). Genetic Algorithms in Search, Optimization, and machine Learning, Addison-Wesley, Reading. MA,412-415.
- Goldberg, D.E., Deb,K. (1991). "A Comparative Analysis of Selection Schemes Used in Genetic Algorithms", Foundation of Genetic Algorithms, Morgan Kaufmann Pub. 69-93.
- Goldberg, D.E. (1993). "Making genetic algorithms fly: A lesson from the Wright brothers," Adv. Technol. Dev. 2, pp.1-8, Feb.
- Gotlieb, C. C. (1963). The Construction of Class-Teacher Timetables, IFIP congress, North-Holland, Bildiriler Kitabı, 73-77.
- Grant, V. (1985). The Evolutionary process. New York, Columbia University Press.
- Grefenstette, J.J. (1986). "Optimization of Control Parameters for Genetic Algorithm". IEEE Transactions on Systems and Cybernetics, 16:1. 122-128.
- Haataja, J. (1994). Solving Optimization Problems. CSC- Center for Scientific Computing Ltd, Yliopistopaino. ISBN 952-9821-02-6. Ed. 1. 232p.(InFinnish).
- Haupt Randy L., Haupt Sue E. (1998). Practical Genetic Algorithms, A Willey-Interscience Publication, USA.
- Holland, J. H. (1975). Adaption in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence.
- Holland, J. (1992). Adaption in Natural and Artificial.
- Karaboğa, D. (2004). "Yapay Zekâ Optimizasyon Algoritmaları". İstanbul: Atlas Yayınevi, 75-112.
- Koza, J. R. (1992). "Genetik Programming: On the Programming of Computers by Means Of Natural Selection". MIT Pres. Cambridge.
- Körez, M.T. (2005). Sıralı Akış Tipi Çizelgeleme Problemlerinde Genetik Algoritma Uygulaması Yüksek Lisans Tezi. İstanbul: Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 85s.
- Krauss B., Lee J., Newman D. (2013). Optimizing the Assignment of Students to Classes in an Elementary School, INFORMS Transactions on Education, Cilt 14, s. 39-44

- Kurt, M., Semetay, C. (2001). Genetik Algoritma ve Uygulama Alanları, M.M.O. Makine Mühendis Dergisi, 42 (501). 19-24.
- Louis, S. J. VE LI, G. (2000). "Case injected genetic algorithms for traveling salesman problems", Information Sciences, 122. 201–225.
- McConnell S. (2006). Software Estimation: Demystifying the Black Art (Developer Best Practices), Microsoft Press, pp 136.
- Man, K.F., Tang, K.S, Kwong, S. (1995). "Genetic Algorithms: Concepts and Applications". IEEE Transactions on Industrial Electronics. 43:5. 519-533.
- Michalewicz, Z. (1994). "Genetic Algorithms + Data Structures = Evolutions Programs", Springer-Verlag, Berlin. 340-353
- MirHassani, S.A. (2006). "A Computational Approach to Enhancing Course Timetabling with Integer Programming", Applied Mathematics and Computation, 175: 814-822.
- Mitchell, M. (1996). "An Introduction to Genetic Algorithms". MIT Press. Massachusetts. 205-213
- Nadeau, C. and Bengio, Y. (2003). "Machine Learning". Norwell: Kluwer Academic Publishers, 52: 239-281.
- Negnevitsky, M. (2006). Artificial Intelligence: A Guide to Intelligent Systems, Pearson Education Limited 2nd Edition.
- Odetayo, M.O. (1995). "Knowledge Acquisition and Adaptation: Genetic Approach", Expert Systems, 12:1 3-13.
- Özgül, F. (2013). Her Yönüyle Python, 2. Baskı, Kodlab, İstanbul.
- Painton, L. and Campbell, J. (2000). "Genetic Algorithms in Operation of System Reliability", IEEE Transactions on Reliability, 44. 172-178.
- Palko S. (1996). "Structural Optimization of Induction Motor using a Genetic Algorithm and a Finite Element Method". Acta Polytechnica Scandinavica, Electrical Engineering Series No. 4, Helsinki, pp. 10-12.
- Qu E., Burke E., Mccollum B., Merlot L. ve Lee S. (2009). A Survey of Search Methodologies and Automated System Development for Examination Timetabling, J. of Scheduling, Kluwer Academic Publishers, Cilt 12, No. 1, Şubat 2009, s. 55-89.
- Reeves, C. (1993). "Modern Heuristic Methods for Combinational Problems", Blackwell Scientific Publications, Oxford, 320-328.
- Ross P., Hart E. ve Corne D. (2006). Some Observations About Ga-Based Exam Timetabling, Practice and Theory of Automated Timetabling II, Toronto, Springer Berlin Heidelberg, s. 115-129.
- Shin-ike K. ve Iima H. (2011). A Method For Determining Classroom Seating Arrangements By Using A Genetic Algorithm, SICE Annual Conference (SICE), Tokyo, s. 161-166.

- Sivanandam, S.N., Deepa, S.N. (2008). Introduction to Genetic Algorithm, Springer. 1st Edition.
- Srinivas, M., Patnaik, L.M. (1994). "Genetic Algorithms:" A Survey, Evolutionary Computation. 27:6. 17-26.
- Şen,Z. (2004). Genetik Algoritmalar ve En İyileme Yöntemleri. İstanbul: Su Vakfı Yayınları.
- Tabak, Ö. (2008). Genetik Algoritma ile Kapasiteli Servis Güzergâhı Belirlenmesi ve Bir Uygulama Yüksek Lisans Tezi. Eskişehir: Anadolu Üniversitesi Fen Bilimleri Enstitüsü.
- Tekin, H. (1977). Eğitimde ölçme ve değerlendirme. Ankara: Mars Matbaası.
- Tekin, H. (2004). Eğitimde ölçme ve değerlendirme. Ankara: Yargı Yayınevi,110.
- Tomassini, M. (1995). "A Survey of Genetic Algorithms", Annual Reviews of Computational Physics, 3: 87-118.
- Tomassini, M. (1996). Evolutionary Algorithms, Lecture Notes in Computer Science. Springer-Verlag, Berlin, 19-47.
- Turgut, M.F. (1992). Eğitimde ölçme ve değerlendirme metotları. Ankara: Saydam Matbaacılık.
- Turgut, M. F., Baykul, Y. (2012). Eğitimde ölçme ve değerlendirme. Ankara: Pegem.
- Ulusoy,G. (2002). "Proje Planlamada Kaynak Kısıtlı Çizelgeleme". Sabancı Üniversitesi Mühendislik ve Doğa Bilimleri Fakültesi Dergisi. 8: 23-29.
- Yiğit, T. (2006). Meslek Liseleri Haftalık Ders Çizelgelerinin Genetik Algoritmalar Yardımıyla Oluşturulması. Gazi Üniversitesi Endüstriyel Sanatlar Eğitim Fakültesi Dergisi, 19, 25-39.