

**GENETİK ALGORİTMA İLE ARAÇ ROTALAMA PROBLEMİNİN ÇÖZÜMÜ
VE BİR UYGULAMA DENEMESİ**

Yüksek Lisans Tezi

Ebru OKUR

Eskişehir 2019

**GENETİK ALGORİTMA İLE ARAÇ ROTALAMA PROBLEMİNİN ÇÖZÜMÜ
VE BİR UYGULAMA DENEMESİ**

EBRU OKUR

YÜKSEK LİSANS TEZİ

İşletme Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Mahmut ATLAS

Eskişehir

Anadolu Üniversitesi

Sosyal Bilimler Enstitüsü

Temmuz, 2019

JÜRİ VE ENSTİTÜ ONAYI

Ebru OKUR'un "Genetik Algoritma ile Araç Rotalama Probleminin Çözümü ve Bir Uygulama Denemesi" başlıklı tezi 10 Temmuz 2019 tarihinde, aşağıdaki jüri tarafından Lisansüstü Eğitim Öğretim ve Sınav Yönetmeliğinin ilgili maddeleri uyarınca toplanan İşletme Anabilim Dalı Sayısal Yöntemler Bilim Dalında, yüksek lisans tezi olarak değerlendirilerek kabul edilmiştir.

İmza

Üye (Tez Danışmanı) : Dr.Öğr.Üyesi Mahmut ATLAS

Üye : Doç.Dr.Namık Kemal ERDOĞAN

Üye : Dr.Öğr.Üyesi Levent TERLEMEZ

Prof.Dr.Bülent GÜNŞOY
Anadolu Üniversitesi
Sosyal Bilimler Enstitüsü Müdürü

ÖZET

GENETİK ALGORİTMA İLE ARAÇ ROTALAMA PROBLEMİNİN ÇÖZÜMÜ VE BİR UYGULAMA DENEMESİ

Ebru OKUR

İşletme Anabilim Dalı

Anadolu Üniversitesi, Sosyal Bilimler Enstitüsü, Temmuz 2019

Danışman: Dr. Öğr. Üyesi Mahmut ATLAS

Bu çalışmada, çözülmesi zor problemler sınıfında yer alan araç rotalama problemi ele alınmıştır. Araç rotalama problemi kısaca, müşteri taleplerini karşılayacak ürün dağıtımına ait optimum rotaların planlanması problemidir. Toplam kat edilen mesafenin en küçüklenmesi problemin temel amacını oluşturmaktadır. Günümüz iş hayatında ürün dağıtım maliyetleri toplam lojistik maliyetlerinin önemli bir kısmını oluşturmaktadır. Çalışmada, meta-sezgisel çözüm tekniklerinden birisi olan genetik algoritma kullanılmıştır. Çalışmanın uygulama kısmında, Eskişehir Halk Ekmek A.Ş.'nin müşteri taleplerini karşılayacak araç rota mesafesini iyileştirmek amaçlanmaktadır. Bu amaçla, Eskişehir Halk Ekmek A.Ş.'nin dağıtım araçlarının, satış büfelerine en kısa mesafeden rota hesaplamasına optimum çözüm aranmıştır. Sonrasında, mevcut durumun araç rotasyonu ile genetik algoritma ile bulunan araç rotasyonu karşılaştırılmıştır.

Anahtar Sözcükler: Genetik Algoritma, Araç Rotalama Problemi, Meta-Sezgisel Teknikler

ABSTRACT

SOLUTION OF VEHICLE ROUTING PROBLEM WITH GENETIC ALGORITHM AND AN APPLICATION STUDY

Ebru Okur

Business Administration Department

Anadolu University, Graduate School of Social Sciences, July 2019

Supervisor: Assist. Prof. Dr. Mahmut ATLAS

In this study, vehicle routing problem which is in the class of hard to solve problems is addressed. The vehicle routing problem is, in short, the problem of planning optimal routes for product distribution to meet customer demands. The main objective of the problem is to minimize the total distance traveled. In today's business world, product distribution costs constitute an important part of total logistics costs. In this study, one of the meta-heuristic techniques, genetic algorithm, is used. In the application part of the study, it is aimed to improve the vehicle route distance to meet customer demands of Eskişehir Halk Ekmek (Public Bread) Company. For this purpose, the optimum solution for the calculation of the route of the distribution vehicles of Eskişehir Halk Ekmek from the shortest distance to sales points was sought. Then, the vehicle rotation of the current situation and the vehicle rotation found by genetic algorithm were compared.

Keywords: Genetic Algorithm, Vehicle Routing Problem, Meta-Heuristic Techniques

ÖNSÖZ

“Genetik Algoritma İle Araç Rotalama Probleminin Çözümü Ve Bir Uygulama Denemesi” isimli çalışma Anadolu Üniversitesi Sosyal Bilimler Enstitüsü İşletme Anabilim Dalı Yüksek Lisans Programı’nda tez olarak hazırlanmıştır.

Yüksek lisans öğrenimim boyunca ve özellikle tez çalışması süresince göstermiş olduğu büyük desteği, özverisi, ilgisi, değerli zamanı, katkıları ve sabrı için değerli danışmanım Sayın Dr. Öğr. Üyesi Mahmut ATLAS’a,

Değerli görüş ve önerileriyle tez çalışmasının programlama ve yazılım kısmında önemli destek ve katkılarından dolayı Öğr. Gör./Kıdemli Yazılım Mühendisi Onur BAŞTÜRK’e,

Her konuda ve her zaman bana destek olan, teşvik eden ve güvenen annem Hülya TÜCCAR’a ve ablam Burcu ÖRDEK’e ,

En içten teşekkürlerimi sunarım.

Ebru OKUR

10/07/2019

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; bu çalışmamın Anadolu Üniversitesi tarafından kullanılan "bilimsel intihal tespit programı"yla tarandığını ve hiçbir şekilde "intihal içermediğini" beyan ederim. Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykır bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçları kabul ettiğimi bildiririm.

Ebru OKUR

İÇİNDEKİLER

	<u>Sayfa</u>
BAŞLIK SAYFASI	i
JÜRİ VE ENSTİTÜ ONAYI.....	ii
ÖZET	iii
ABSTRACT.....	iv
ÖNSÖZ	v
ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ.....	vi
İÇİNDEKİLER	vii
TABLolar DİZİNİ.....	ix
ŞEKİLLER DİZİNİ.....	x
SİMGE VE KISALTMALAR DİZİNİ	xii
GİRİŞ.....	1

BİRİNCİ BÖLÜM

1. ARAÇ ROTALAMA PROBLEMİ	3
1.1. Araç Rotalama Probleminin Tanımı.....	3
1.2. Araç Rotalama Probleminin Bileşenleri	5
1.3. Araç Rotalama Probleminin Modeli	8
1.4. Araç Rotalama Probleminin Çeşitleri.....	10
1.4.1. Kapasite kısıtlı araç rotalama problemi.....	10
1.4.2. Mesafe kısıtlı araç rotalama problemi.....	11
1.4.3. Zaman pencereci araç rotalama problemi.....	12
1.4.4. Geri toplamalı araç rotalama problemi	13
1.4.5. Dağıtım ve toplamalı araç rotalama problemi.....	13
1.5. Araç Rotalama Problemi İçin Çözüm Teknikleri.....	14
1.5.1. Kesin teknikler.....	14
1.5.2. Sezgisel teknikler	16
1.6. Araç Rotalama Probleminin Uygulama Alanları.....	20

İKİNCİ BÖLÜM

2. GENETİK ALGORİTMA	21
2.1. Genetik Algoritmanın Tanımı, Tarihçesi ve Önemi	21
2.2. Genetik Algoritmanın Temel Kavramları	23
2.2.1. Gen.....	23
2.2.2. Kromozom (Birey).....	23

2.2.3. Popülasyon	24
2.2.4. Uygunluk fonksiyonu ve değeri.....	24
2.2.5. Genetik operatörler	25
2.2.6. Sonlandırma kriteri.....	26
2.2.7. Genetik algoritmanın aşamaları ve akış şeması	26
2.3. Genetik Algoritmanın Adımları.....	28
2.3.1. Kodlama	28
2.3.2. Başlangıç popülasyonunun oluşturulması.....	30
2.3.3. Uygunluk fonksiyonunun belirlenmesi ve uygunluk değerinin hesaplanması.....	31
2.3.4. Seçim operatörü.....	32
2.3.5. Çaprazlama operatörü.....	35
2.3.6. Mutasyon operatörü.....	42
2.3.7. Sonlandırma kriteri ve algoritmanın sona ermesi.....	43
2.4.Genetik Algoritmanın Özellikleri	44
2.4.1.Genetik algoritmanın avantajları	45
2.4.2.Genetik algoritmanın dezavantajları.....	45
2.4.3.Genetik algoritmanın kullanım nedenleri	46
2.5.Genetik Algoritmanın Uygulama Alanları	46
ÜÇÜNCÜ BÖLÜM	
3. GENETİK ALGORİTMA KULLANILARAK ESKİŞEHİR HALK EKMEK İŞLETMESİ İÇİN ARAÇ ROTALAMA PROBLEMİ ÇÖZÜMÜ	49
3.1. Eskişehir Halk Ekmek İşletmesi ve Problem Tanımı	49
3.2.Uygulama Problemi Verileri	52
3.3.Uygulama Probleminin Genetik Algoritma ile Çözümü	58
3.3.1.Kromozom yapısı ve başlangıç popülasyonu	59
3.3.2.Uygunluk fonksiyonu	61
3.3.3.Genetik operatörler	63
3.3.4.Sonlandırma kriteri.....	68
3.4. Analiz Sonuçları	69
DÖRDÜNCÜ BÖLÜM	
4.SONUÇ VE ÖNERİLER	72
KAYNAKÇA.....	74
ÖZGEÇMİŞ	

TABLULAR DİZİNİ

	<u>Sayfa</u>
Tablo 3.1. Eskişehir Halk Ekmek İşletmesi bir günlük sipariş miktarları	53
Tablo 3.2. Genetik algoritmalar için önerilen parametre değerleri	68
Tablo 3. 3. Eskişehir Halk Ekmek İşletmesi'nin kullandığı mevcut rota	70
Tablo 3. 4. Genetik algoritma ile çözüm sonucu oluşturulan optimum rota.....	71

ŞEKİLLER DİZİNİ

	<u>Sayfa</u>
Şekil 1.1. Araç rotalama problemi örnek gösterimi.....	5
Şekil 1.2. Araç rotalama problemi türleri ve birbiri ile ilişkisi.....	10
Şekil 1.3. Zaman pençeleri araç rotalama problemi örneği	12
Şekil 1.4. Araç rotalama problemi çözüm teknikleri.....	14
Şekil 1.5. Dal-sınır algoritması örneği.....	15
Şekil 1.6. Tasarruf algoritması	17
Şekil 1.7. Gerçek karınca davranışları	19
Şekil 2.1. Gen, kromozom ve popülasyonu gösteren yapı	24
Şekil 2.2. Genetik algoritmanın genel yapısı.....	27
Şekil 2.3. Genetik algoritma akış şeması.....	28
Şekil 2.4. İkili kodlama gösterimi	29
Şekil 2.5. Değer kodlama gösterimleri	29
Şekil 2.6. Permutasyon kodlama gösterimi	30
Şekil 2.7. Rulet tekerleği Seçimi	33
Şekil 2.8. Turnuva seçimi	34
Şekil 2.9. Tek noktalı çaprazlama	37
Şekil 2.10. İki noktalı çaprazlama	38
Şekil 2.11. Çok noktalı çaprazlama	38
Şekil 2.12. Düzenli Çaprazlama	39
Şekil 2.13. Dairesel Çaprazlama.....	40
Şekil 2.14. Sıralı Çaprazlama	41
Şekil 2.15. Kısmi eşleştirmeli çaprazlama.....	42
Şekil 2.16. Mutasyon işlemi	43
Şekil 3.1. Eskişehir Halk Ekmek İşletmesi'ne ait ürünler	50
Şekil 3.2. Eskişehir Halk Ekmek İşletmesi üretim tesisi ve satış noktalarının konumları	51
Şekil 3.3.Eskişehir Halk Ekmek İşletmesi üretim tesisi ve satış büfeleri arasındaki mesafe matrisi (km)	56
Şekil 3.4. Yazılımın ara yüzündeki kısımların görseli	59
Şekil 3. 5.Uygunluk fonksiyonunun geliştirilen yazılımdaki ara yüz görseli	62

Şekil 3.6. Başlangıç popülasyonunun geliştirilen yazılımdaki ara yüz görseli	63
Şekil 3.7. Seçim işleminin geliştirilen yazılımdaki ara yüz görseli.....	64
Şekil 3.8. Eskişehir Halk Ekmek İşletmesi ekmek dağıtım probleminde mutasyon işlemi örneği	67

SİMGE VE KISALTMALAR DİZİNİ

Simgeler

- C_{ij}** : i ile j düğümü arasındaki mesafe
 $P(t)$: t anındaki popülasyon büyüklüğü
 $Enkz$: En küçüklenmeye çalışılan amaç fonksiyonu

Kısaltmalar

- DP** : Doğrusal Programlama
DTARP : Dağıtım Ve Toplamalı Araç Rotalama Problemi
GTARP : Geri Toplamalı Araç Rotalama Problemi
KARP : Kapasite Kısıtlı Araç Rotalama Problemi
MARP : Mesafe Kısıtlı Araç Rotalama Problemi
PMX : Kısmi Eşleştirmeli Çaprazlama
TA : Tabu Arama
TB : Tavlama Benzetim
YSA : Yapay Sinir Ağları
ZPARP : Zaman Pencereci Araç Rotalama Problemi

GİRİŞ

Optimum güzergahın belirlenmesi günümüzde gittikçe gelişmekte ve önem kazanmaktadır. Bir işletmenin başlıca lojistik amacı, hedeflere en kısa yoldan ulaşmasını sağlayacak bir sistem kurmaktır. Dağıtım faaliyetlerinin etkin ve verimli yönetilmesi maliyet düşüşlerini de doğrudan etkilemektedir. Uygun dağıtım rotalarının elde edilmesi sonucu oluşan kazanımlar, işletmelerin karlılığını arttırmaktadır.

“Araç Rotalama Problemi” başlıklı ilk bölümünde, araç rotalama problemi ele alınarak ayrıntılı bir şekilde tanıtılmıştır. Bu bölümde, araç rotalama probleminin tanımı, araç rotalama probleminin bileşenleri, araç rotalama probleminin çeşitleri ve araç rotalama problemi için çözüm teknikleri hakkında bilgi verilmeye çalışılmıştır.

Araç rotalama probleminin çözümünde optimum rotaların belirlenebilmesi için birçok teknik geliştirilmiştir, fakat her durumda en iyi sonucu verebilecek kesin bir model ortaya konulamamıştır. Bu sebeple, kesin tekniklerin yetersiz kaldığı durumlar için sezgisel ve meta-sezgisel çözüm teknikleri geliştirilmiştir. Bazı durumlarda ise, kesin çözüm teknikleri ile optimum sonuç sağlansada, çözüme ulaşmak uzun sürebilmektedir. Günümüz iş dünyasında işletmelerin hızlı kararlar alabilmesi için, optimum sonuçlara ulaşılmasa bile, kısa sürede optimuma yakın sonuçlar veren sezgisel ve meta-sezgisel teknikler tercih edilmektedir.

“Genetik Algoritma” başlıklı ikinci bölümünde, araç rotalama problemi çözüm yaklaşımlarından meta-sezgisel çözüm teknikleri sınıfında yer alan genetik algoritmanın detaylarına yer verilmiştir. Bu başlık altında genetik algoritmanın tanımı, tarihçesi, önemi, temel kavramları ve işleyişine yönelik bilgi verilmesi amaçlanmaktadır. Meta-sezgisel algoritmaların araç rotalama probleminin çözümünde sıklıkla kullanılan tekniklerden olması, en iyiye yakın çözüme diğer algoritmalara göre daha kısa sürede ulaşabilmeleri, genetik algoritmanın bunlar arasında en etkin kullanılanı olup üstünlüklerinin bulunması gibi nedenlerden dolayı, çalışmada genetik algoritma tercih edilmiştir.

Genetik algoritma, evrim sürecinden ilham alınarak oluşturulmuş olup, en iyinin hayatta kalma temeline dayanmaktadır. Burada amaçlanan, doğal sistemlerin hayatta kalma özelliğinin taklit edilerek yapay sistemlere uygulanmasıdır.

Çalışmanın üçüncü bölümünde ise, uygulama çalışması yer almaktadır. Bu bölümde, Eskişehir Halk Ekmek İşletmesi'nin ekmek dağıtım problemine genetik algoritma kullanılarak çözüm aranmıştır. Uygulamanın amacı; müşteri taleplerinin tümü

karşılarken, en kısa rotayı oluşturabilmek için işletmeye ait hangi aracın hangi siparişleri nasıl bir sırayla taşınması gerektiğinin genetik algoritma ile belirlenmeye çalışılmasıdır. Eskişehir Halk Ekmek İşletmesi, homojen yapıda olan 3 araca sahiptir. Tek bir üretim tesisine(depo) sahip olan işletme, dağıtıma üretim tesisinden başlayarak 51 satış büfesinin tüm talebinin karşılanmasını sağlayacak bir dağıtım gerçekleştirmelidir. Araçlar, rotasyonundaki tüm satış büfelerini ziyaret ettikten sonra rotasyonunu yine üretim tesisinde sonlandırmalıdır. Eskişehir Halk Ekmek İşletmesi'nde araç rotasyonu planlamasında bugüne kadar ilgili çalışanlarının 17 yıllık tecrübelerinden yararlanılarak bir rota planlaması yapılmıştır. Bu sebeple tez çalışması kapsamında, işletmenin rotalama planlaması için genetik algoritma tekniği ile çözüm bulunmaya çalışılmıştır.

Eskişehir Halk Ekmek İşletmesi'nin dört farklı ürün çeşitliliğinin bulunması ve kullandığı araçların belli bir kapasite kısıtlaması olması problemi karmaşık hale getirmektedir. Bir ekmek kasasında yer alan ekmeklerin sayısı ekmeğin türüne göre değişiklik göstermektedir. Araçlar, günlük sipariş miktarına göre rotasındaki satış büfelerini ziyaret etmektedir. Araçların kapasitesi ise, araç rotasyonunda bulunan tüm satış büfelerinin talebini karşılayacak düzeyde değildir. Bu sebeple, gün içinde aracını doldurabilmek için yakın ya da uzak fark etmeksizin sık sık üretim tesisine geri dönme mecburiyetindedir.

Genetik algoritma kullanılarak elde edilen çözümde, Eskişehir Halk Ekmek İşletmesi'nin araç rotasyonu için mevcut sistemden daha iyi bir çözüm belirlenmeye çalışılmıştır. Son olarak mevcut sistemin araç rotasyonu ve genetik algoritma ile bulunan çözüm sonuçları karşılaştırılmıştır.

BİRİNCİ BÖLÜM

1. ARAÇ ROTALAMA PROBLEMİ

1.1. Araç Rotalama Probleminin Tanımı

Araç rotalama problemi, müşterilerin araçlar aracılığı ile ziyaret edilmesini içeren tüm problem sınıfına verilen genel bir addır. Bu problemlerde amaç, araçların gittiği toplam mesafenin (veya harcanan zaman, yapılan maliyet vb.) minimum düzeyde olmasını sağlamaktır (Christofides, 1976).

Araç rotalama problemi, hem pratik önemi hem de zor problemler sınıfında yer alması nedeniyle, optimizasyon problemleri arasında en çok incelenen problemlerden biridir. Araç rotalama problemi, bir dizi müşteriye hizmet verebilmek için araç filosu aracılığıyla optimum rotaların belirlenmesiyle ilgilenir (Toth ve Vigo, 2002a).

Araç rotalama probleminde iki temel amaç vardır (Christofides, 1976):

- i. Müşteri taleplerini, toplam değişken maliyetini minimum yapacak şekilde mevcut araçlar ile en uygun rotaların bulunması,
- ii. Müşteri gereksinimlerini karşılayabilecek şekilde en az sayıda taşıt ve en kısa yolun bulunmasıdır.

Klasik araç rotalama problemi, talepleri bilinen müşterilere hizmet veren bir grup aracın dağıtımını gerçekleştirdikten sonra başlangıç noktasına geri dönmesini sağlayacak en uygun güzergâhın belirlenmesi esasına dayanmaktadır. Burada amaç, bu dağıtımını gerçekleştirirken aynı zamanda oluşan tüm taşıma maliyetlerinin minimize edilmesini sağlamaktır. Klasik araç rotalama probleminin çözümü, aynı tip ve kapasiteye sahip homojen bir araç filosu için bir depoda başlayıp yine aynı depoda biten ve her müşteriye yalnızca bir kez hizmet verilmesine olanak sağlayan bir dizi rotanın oluşturulmasıdır (Aksaraylı vd., 2017).

Araç rotalama problemi, ilk olarak Dantzig ve Ramser (1959) tarafından genelleştirilmiş gezgin satıcı problemi olarak ifade edilmiştir. Günlük yaşam problemi için modellenerek özgün bir algoritmaya dayalı çözüm yaklaşımı elde edilmiştir. 1964'te Clarke ve Wright, etkili bir sezgisel algoritmayla farklı araç kapasiteleri ve müşteri talepleri için çözüm yaklaşımını geliştirmişlerdir. Literatürde bu çalışmaları, farklı türdeki araç rotalama problemi modelleri için kesin ya da yaklaşık çözüm sunan başka

algoritmalar takip etmiştir. Detaylı literatür çalışması, Eksioğlu vd. (2009) tarafından sistematik bir şekilde yapılmıştır.

Araç rotalama probleminin pratik uygulamalarında, rotalama sırasında birçok ek gereksinim ve operasyonel kısıtlamalar getirilmiştir. Bahsedilen ek gereksinim ve kısıtlamalar şu şekilde sıralandığında:

- Her güzergâhtaki yük araçların verilen kapasitesini aşmamalı,
- Her güzergahın toplam uzunluğu belirtilen sınırın üzerinde olmamalı,
- Müşterilerin hizmetleri belirli zaman pencerelerinde gerçekleşmeli,
- Araç filosu farklı tip araçlardan oluşabilir,
- Müşteriler arasında öncelik ilişkileri olabilir,
- Müşteri talepleri önceden tam olarak bilinemeyebilir,
- Bir müşterinin talebi farklı araçlar arasında bölünebilir,
- Talepler veya seyahat süreleri gibi bazı problem özellikleri dinamik olarak değişebilir (Toth ve Vigo, 2002a).

Araç rotalama probleminin günlük yaşam uygulamalarında karşılaşılan kısıtlar üç ana grupta toplanabilmektedir (Düzakın ve Demircioğlu, 2009):

1. Araçlarla ilgili kısıtlar

- Araç kapasite kısıtı (ağırlık veya hacim olarak)
- Toplam zaman kısıtı
- Sürücünün çalışma saatleri için yasal sınırlamalar

2. Müşteriler ile ilgili kısıtlar

- Her bir müşterinin bir tür ürün talep etmesi veya belirli çeşitte ürün dağıtılması; Lojistik firmaları buna örnek verilebilir.
- Dağıtımın yapılabilmesi için belirli zaman aralıklarının bulunması

3. Diğer kısıtlar

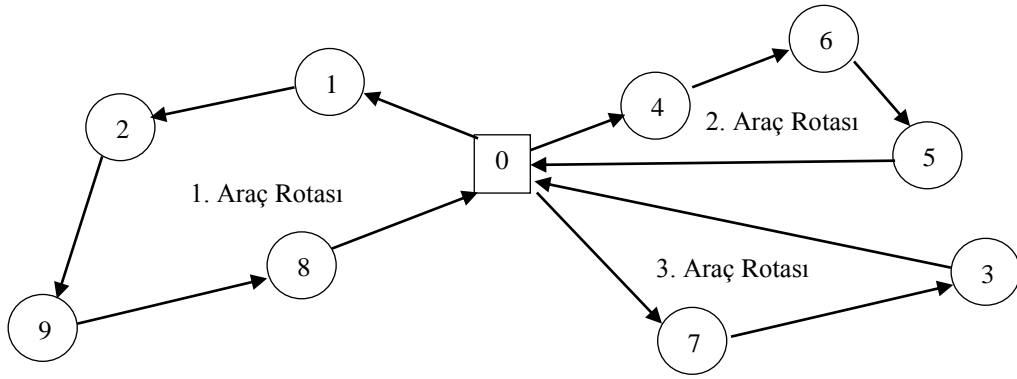
- Aynı araç ile aynı günde, aracın depoya dönerek tekrar yola çıkmasıyla, birden fazla tur yapılması
- Bir turun bir günden uzun olması
- Birden fazla depo olması

Araç rotalama probleminde rota oluşturulurken, şu hususlara dikkat edilmelidir (Çeyrekoğlu, 2017):

- Farklı konumlarda bulunan tüm müşterilerin talepleri karşılanmalıdır.

- Her müşteriye talebinin karşılanması için sadece bir araç bir defa uğramalıdır.
- Rotaya depodan başlayan bir araç mutlaka depoya geri dönmelidir.
- Uğruncak müşterilerin talep toplamları araç kapasitesini geçmemelidir.
- Her bir rota için sadece bir araç planlanmalıdır.
- Araç rotalamada temel amaç, her bir aracın kat edeceği mesafenin minimize edilmesi olmalıdır.

Şekil 1.1'de örnek bir araç rotalama problemi modeli gösterilmiştir. 1'den 9'a kadar numaralanmış düğümler sipariş noktalarını, 0 ise ana depoyu ifade etmektedir.



Şekil 1.1. Araç rotalama problemi örnek gösterimi (Şen, Cömert Ercan ve Yazgan, 2015)

Araç rotalama problemi, gezgin satıcı probleminin birden fazla araç ve eklenmiş kısıtlar ile geliştirilmiş halidir. Araç rotalama probleminin çözümü, aynı sayıda müşteri veya şehre sahip gezgin satıcı problemine kıyasla daha zordur (Demicioğlu ve Düzakın, 2009).

1.2. Araç Rotalama Probleminin Bileşenleri

Araç rotalama problemi, bir yol şebekesi üzerinden araçlar yardımıyla yapılan bir mal taşımacılığıdır. Bu yönden bakıldığında günlük yaşamda bu problemin bazı bileşenlerinden söz edilebilir. Bu bileşenler aşağıda verilmiştir (Paolucci, 2005):

- Yol şebekesi
- Müşteriler
- Araçlar
- Depolar

- Sürücüler
- Operasyonel kısıtlar
- Amaçlar

Yol ağı: Araç rotalama probleminde yol şebekesi, bir şebeke ile gösterilir. Şebekede düğümler ve oklar vardır. Şebekedeki oklar yollara, düğümler ise müşterilere karşılık gelmektedir. Yol şebekesini temsil eden şebeke yönlü, yönsüz veya hem yönlü hem yönsüz karışımı oklardan oluşabilir (Keçeci, 2008). Uygulamada yolların maliyet olarak düşünebileceğimiz bir değeri vardır, uzunluk birimi olarak mesafesi veya seyahat süresi gibi

Müşteriler: Araç rotalama probleminde müşteriler, hizmet bekleyen, yani depodan belirli miktarda ürün talep eden veya depoya belirli miktarda ürün arz eden birimlerdir. Bir şebekede müşteriler, düğümler ile temsil edilirler. Uygulamada müşterilerin talep miktarları, ürünlerin teslim edilmesi ya da toplanması şeklinde olabilir (Kiremitçi v.d., 2014).

Araçlar: Araç rotalama probleminin hareketli bileşenleri araçlardır. Kaç adet araç varsa o kadar rota olmalıdır. Araçların bir veya birden fazla depoda olduğu düşünülür. Her aracın bir taşıma kapasitesi mevcuttur. Taşıma kapasitesi ağırlık cinsinden olabileceği gibi hacim cinsinden de olabilir. Ayrıca her aracın taşıma kapasitesi aynı olabileceği gibi bazı problemlerde farklı taşıma kapasitesine sahip araçlarda kullanılabilir (Keçeci, 2008). Araç bileşeni için kapasite, hız, yakıt tüketim miktarı, birim zamanda oluşan araç kullanım maliyeti gibi özelliklerinden bahsedilebilir.

Depolar: Araç rotalama probleminde depolar, çeşitli veya benzer tipte araçların buldukları ve dağıtım planının merkezini oluşturan birimlerdir. Verilecek kararlar, yapılacak planlar, araçların depodan çıkarak hangi noktalara uğrayıp geri tekrar depoya dönmesi gerektiği fikrine dayanır. Tek depo olabileceği gibi bazı problemlerde, birden fazla deponun da olması mümkündür. Depolara atanan araçların sayısı ve türleri, deponun müşterilere arz edebileceği toplam miktarlar da, depo bileşeninin özelliklerini oluşturmaktadır (Günay, 2013).

Sürücüler: Sürücüler, araç rotalama problemlerinde doğrudan dikkate alınmasa da dolaylı olarak göz önünde bulundurulmak zorundadır. Günlük yaşam uygulamalarında sendikal ve sözleşme şartları modellere yansıtılmalıdır. Yasalarda sürücülerin çalışma periyotları, vardiyaları, fazla mesai şartları ve vermesi gereken dinlenme araları

belirtildiğinden, oluşturulan dağıtım planlarının bu düzenlemelere göre yapılması zorunluluğu vardır.

Kısıtlayıcılar: Araç rotalama probleminde kısıtlayıcılar, yapılan taşımacılığın doğası ve gereklerine, verilen taşımacılık hizmetinin kalitesine ve sürücülerin çalışma sözleşmelerine bağlı olarak değişiklik göstermektedir. Ancak, genel olarak bir araç rotalama probleminde kısıtlayıcılar iki sınıfta toplanır. İlki, yerel kısıtlayıcılar ve tek bir rota için geçerli olan kısıtlardır. İkincisi ise bütünsel kısıtlayıcılar ve bütün rotalar için geçerli olan kısıtlardır (Keçeci, 2008).

Yerel kısıtlayıcılar ile şu kısıtlar:

- Araç kapasitesinin aşılmaması,
- Verilen maksimum rota uzunluğunun veya rota süresinin aşılmaması,
- Verilmesi durumunda rotadaki müşterilere belirli zaman pencerelerinde uğranılması,
- Hizmetin türüne göre sadece toplama,
- Sadece dağıtım veya her ikisinin birden yapılması,
- Müşteriler arasındaki öncelik ilişkisi sağlanmaktadır.

Bütünsel kısıtlayıcılarla ise şu kısıtlar:

- Araç sayısı kadar rotanın olması,
- Verilmesi durumunda araç veya depo için maksimum rota sayısının aşılmaması,
- Sürücüler arasında iş yükünün dengelenmesi,
- Çalışma periyotlarının ve vardiyaların,
- Rotalar arasında belirli bir asgari zaman aralığı olacak şekilde düzenlemesi sağlanır (Paolucci, 2005).

Amaçlar: Araç rotalama probleminde;

- Taşıma maliyetleri ve taşımada kullanılan araçların sabit maliyetleri toplamını minimize etmek,
- Araç ve/veya sürücü sayısını minimize etmek,
- Rota sürelerini, mesafelerini ve maliyetlerini dengelemek,
- Tamamen veya kısmen hizmet verilemeyen müşteriler için katlanılması gereken ceza toplamını minimize etmek,
- Toplam mesafeyi minimize etmek,
- Toplam süreyi minimize etmek şeklinde olabilir.

1.3. Araç Rotalama Probleminin Modeli

Araç rotalama probleminin genel matematiksel modeli aşağıda verilmektedir (Fisher ve Jaikumar, 1981):

Parametreler

C_{ij} = i ile j düğümü arasındaki uzaklık

m = araç sayısı

n = düğüm sayısı

S = düğümler alt kümesi

$|S|$ = S alt kümesindeki düğüm sayısı

Karar Değişkenleri

$$x_{ijk} = \begin{cases} 1, & k \text{ aracı } i \text{ düğümünden hemen sonra } j \text{ düğümüne giderse} \\ 0, & \text{diğer durumda} \end{cases}$$
$$y_{ik} = \begin{cases} 1, & k \text{ aracı } i \text{ düğümünü ziyaret ederse} \\ 0, & \text{diğer durumda} \end{cases}$$

Amaç Fonksiyonu

Toplam kat edilen mesafenin en küçüklenmesi, Denklem 1.1’de gösterilmektedir.

$$Enkz = \sum_{i,j} c_{ij} \sum_k x_{ijk} \quad (1.1)$$

Kısıtlar

Her müşteri bir araca atanmalıdır; Denklem 1.2’te gösterilmektedir.

$$\sum_k y_{ik} = \begin{cases} 1, & i = 2, \dots, n \\ m, & i = 1 \end{cases} \quad (1.2)$$

Bir araçla bir şehre en fazla bir kez uğranabilmelidir (i düğümüne k aracı ile gelindiye ve i düğümünden de j düğümüne geçilecekse, bu geçiş yine k aracı ile olmalıdır); Denklem 1.3’te gösterilmektedir.

$$\sum_k x_{ijk} = y_{ik} \quad \forall(i, k) \quad i = 1, \dots, n \quad k = 1, \dots, m \quad (1.3)$$

Alt rota oluşması engellenmelidir; Denklem 1.4'te gösterilmektedir.

$$\sum_{i,j \in S} x_{ijk} \leq |S| - 1 \quad (1.4)$$

$$\forall(k) \quad S \subseteq \{2, \dots, n\} \quad k = 1, \dots, m$$

Karar değişkenleri 0-1 tamsayı değişken olmalıdır; Denklem 1.5'te gösterilmektedir.

$$y_{ik} \in \{ 1, 0 \} \quad i = 1, \dots, n \quad k = 1, \dots, m \quad (1.5)$$

$$x_{ijk} \in \{ 1, 0 \} \quad i, j = 1, \dots, n \quad k = 1, \dots, m \quad (1.6)$$

Bu tanımlamalar altında araç rotalama probleminin genel matematiksel modeli aşağıdaki gibidir;

Amaç fonksiyonu:

$$Enkz = \sum_{i,j} c_{ij} \sum_k x_{ijk} \quad (1.1)$$

Kısıtlayıcılar:

$$\sum_k y_{ik} = \begin{cases} 1, & i = 2, \dots, n \\ k, & i = 1 \end{cases} \quad (1.2)$$

$$\sum_k x_{ijk} = y_{ik} \quad \forall(i, k) \quad i = 1, \dots, n \quad k = 1, \dots, m \quad (1.3)$$

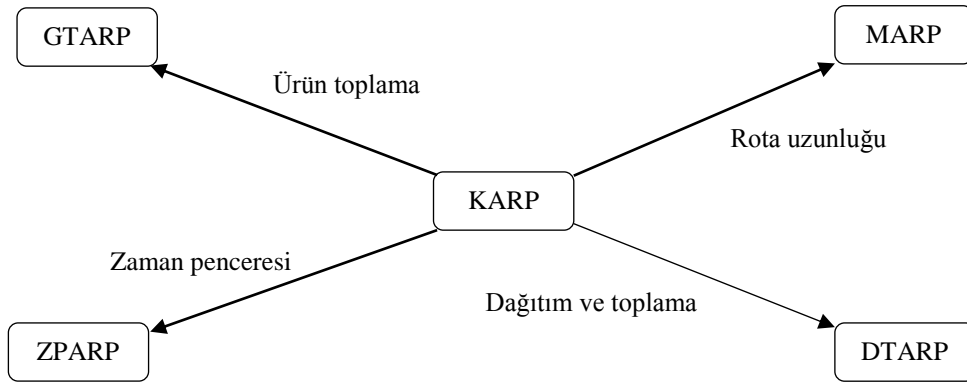
$$\sum_{i,j \in S} x_{ijk} \leq |S| - 1 \quad \forall(k) \quad S \subseteq \{2, \dots, n\} \quad k = 1, \dots, m \quad (1.4)$$

$$y_{ik} \in \{ 1, 0 \} \quad i = 1, \dots, n \quad k = 1, \dots, m \quad (1.5)$$

$$x_{ijk} \in \{ 1, 0 \} \quad i, j = 1, \dots, n \quad k = 1, \dots, m \quad (1.6)$$

1.4. Araç Rotalama Problemi Çeşitleri

Araç rotalama problemi günlük yaşamdaki bazı özel durumlardan kaynaklanan bazı kısıtlar nedeniyle çeşitli dallara ayrılır. Araç rotalama problemlerinin başlıca türleri: kapasite kısıtlı araç rotalama problemi (KARP), mesafe kısıtlı araç rotalama problemi (MARP), zaman pencereci araç rotalama problemi (ZPARP), geri toplamalı araç rotalama problemi (GTARP), dağıtım ve toplamalı araç rotalama problemi (DTARP) ve bu çeşitlerin birleşiminden oluşan diğer kombinasyonlardır. Şekil 1.2.'de araç rotalama problemi türleri ve aralarındaki ilişki gösterilmiştir.



Şekil 1.2. Araç rotalama problemi türleri ve birbiri ile ilişkisi

Literatürde araç rotalama problemleri birden çok temel türe göre sınıflandırılabilir. En önemlileri; kapasite kısıtlı, mesafe kısıtlı, zaman pencereci, geri toplamalı, dağıtım ve toplamalı araç rotalama problemleridir. Her bir temel türün, ilave kısıtlara ve farklı özelliklere sahip alt türleri de literatürde bulunmaktadır (Toth ve Vigo, 2002b).

1.4.1. Kapasite kısıtlı araç rotalama problemi

Araç rotalama problemin temel alınan ve en çok incelenen bir türü kapasite kısıtlı araç rotalama problemidir (Kek, Cheu ve Meng, 2008). Kapasite kısıtlı araç rotalama probleminde, tüm araçların belirli bir kapasitesi vardır. Bu sebeple müşteri taleplerinin önceden bilinmesi gerekmektedir. Talepler tek seferde karşılanmaktadır. Müşterilere hizmet vermek üzere merkez bir depoda bulunan araçlar homojen yapıdadır. Araçlar hareketine bir depodan başlayıp dönüşleri yine bu depoda sonlandırmaktadırlar. Bu problem türünde amaç, tüm müşterilerin taleplerinin en az maliyetle karşılanmasıdır.

Gezgin satıcı problemi, kapasite kısıtlı araç rotalama probleminin özel bir versiyonudur. Gezgin satıcı problemi, problem ağındaki tüm düğümleri ziyaret edecek en az maliyetli rotanın bulunmasını amaçlar. Kapasite kısıtlı araç rotalama probleminde kullanılacak araç sayısı tek olduğunda ve kapasite kısıtı kaldırıldığında, problem gezgin satıcı problemi olmaktadır (Günay, 2013).

KARP için, M adet rotanın (her biri bir araç rotasına karşılık gelmektedir) en az maliyetle belirlenmesi amaçlanmaktadır. Bununla birlikte dikkat edilmesi gereken hususlar aşağıdaki (Günay, 2013):

- Her rota depodan geçmektedir.
- Her müşteri yalnızca bir rotaya atanmalıdır.
- Bir rotada ziyaret edilecek tüm müşterilerin taleplerinin toplamı araç kapasitesi (Q)'ni geçemez, koşullarına uymalıdır.

1.4.2. Mesafe kısıtlı araç rotalama problemi

Kapasite kısıtlı araç rotalama probleminin her rotadaki kapasite kısıtının en fazla mesafe veya en fazla seyahat süresi kısıtı ile değiştirilmesiyle ortaya çıkmıştır ve mesafe kısıtlı araç rotalama problemi (MARP) olarak adlandırılmıştır. Bu mesafe kısıtı, her rotada hizmet verecek aracın kat edebileceği belirli bir mesafe veya belirli bir sürede depoya geri dönme sınırının olduğunu ifade etmektedir.

Mesafe kısıtının eklenmesi taşınan ürünlerin cinsine, sürücülerin ya da araçların özelliklerine bağlı olabilir. Örneğin, taşınan ürünün uzun süre taşınması bozulmalara sebep olabilir ya da sürücünün belirli bir süreden fazla araç kullanmaması gibi sınırlamalar olabilir. Bu gibi durumlarda probleme mesafe kısıtının eklenmesi gerekmektedir.

Bazı problemlerde kapasite kısıtı ile beraber mesafe kısıtının da eklenmesi gerekebilir. Bu durumda ortaya çıkan problem mesafe kısıtlı kapasiteli araç rotalama problemi şeklinde tanımlanır.

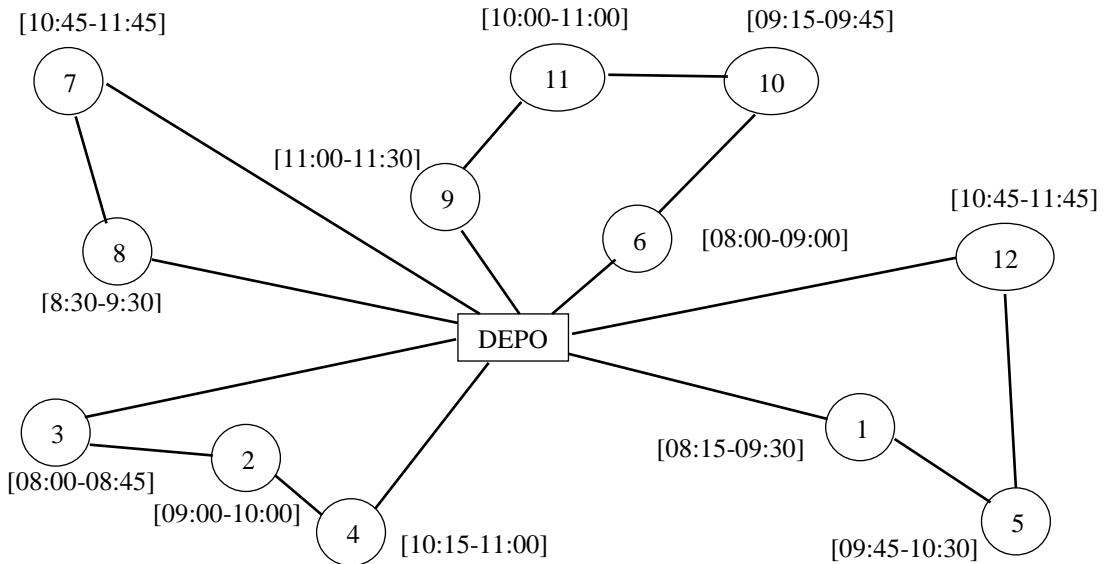
Problemin amacı, araçların kat edecekleri toplam seyahat mesafesini veya seyahat süresini minimize edecek tüm kısıtları sağlayan araç rotalarının belirlenmesini sağlamaktır. Problem şu şekilde tanımlanmıştır (Günay, 2013):

- Her müşteri bir defa ziyaret edilmektedir ve bir araç tarafından hizmet görmektedir.

- Her müşteri rotası depoda başlayıp depoda sonra ermektedir.
- Herhangi bir araç rotasının toplam uzunluğu, araçlar için belirlenen üst sınırı geçemez veya depodaki araç yükleme süresi, müşteriler arası seyahat süreleri ve müşterilerin servis sürelerinin toplamı olan rota süreleri, araç seyahat süresini aşamaz.

1.4.3. Zaman pencereci araç rotalama problemi

Kapasite kısıtlı araç rotalama problemine kapasite kısıtının yanı sıra her müşterinin hizmet görebileceği bir zaman aralığı kısıtının eklenerek genişletilmiş haliyle oluşturulan probleme zaman pencereci araç rotalama problemi adı verilmektedir. Bu zaman dilimi “Zaman Penceresi” olarak adlandırılmaktadır. Zaman penceresi, müşterilerin servisi kabul ettiği bir zaman dilimini ifade etmektedir ve her araç belirli bir zaman penceresinde hizmet vermek durumundadır. Bu zaman kısıtını sağlayabilecek şekilde rotalar oluşturulurken kısa mesafeli ve taleplerin araç kapasitesini aşmayacağı rotalar belirlenmelidir. Bu problem türünde amaç, öncelikle müşterilere istedikleri zaman aralığında hizmet vermek, sonrasında toplam rota uzunluğunu en küçüklemektir. Zaman pencereci araç rotalama problemi için bir örnek Şekil 1.3.’te verilmiştir.



Şekil 1.3. Zaman pencereci araç rotalama problemi örneği (Demircioğlu, 2009)

1.4.4. Geri toplamalı araç rotalama problemi

Geri toplamalı araç rotalama problemi, iade etme durumu gerektiren problemlere uygulanan araç rotalama problemidir. Ambalaj, depozito, yedek parçaların geri dönüşümü için fabrikaya geri gönderilmesi gibi işlemler bu problem türünün kapsamına girmektedir. Bu problem türünde, müşterilerden gelebilecek iade ürünler göz önünde bulundurularak araç kapasiteleri hesaplanmalıdır (Keskinürk vd., 2015).

GTARP, bütün müşterilere hizmet verecek araç rotalarını belirlenmesinde şu kısıtları sağlamalıdır (Toth ve Vigo, 2002b):

- Her araç bir rotaya atanmalıdır.
- Her rota depoda başlayarak depoda sonlanmalıdır.
- Müşteri talepleri ve müşterilerden geri toplanacak ürün miktarları araç kapasitesini aşamaz.
- Her rotada öncelikle ürün talepleri olan müşterilere hizmet verilir, sonrasında varsa ürün toplanacak müşterilere hizmet verilir.
- Her aracın toplam seyahat mesafesi minimize edilmelidir.

1.4.5. Dağıtım ve toplamalı araç rotalama problemi

Dağıtım ve toplamalı araç rotalama probleminde araçlar, müşterilere ürün dağıtımının yapılması ve müşterilerden geri alınacak ürünlerin toplanması olarak iki şekilde hizmet verilmektedir. Dağıtım için gidilen müşteri noktasında dağıtımdan sonra bir önceki dağıtım ürünlerinin kasalarının geri alınması gibi şekillerde karşımıza çıkabilmektedirler.

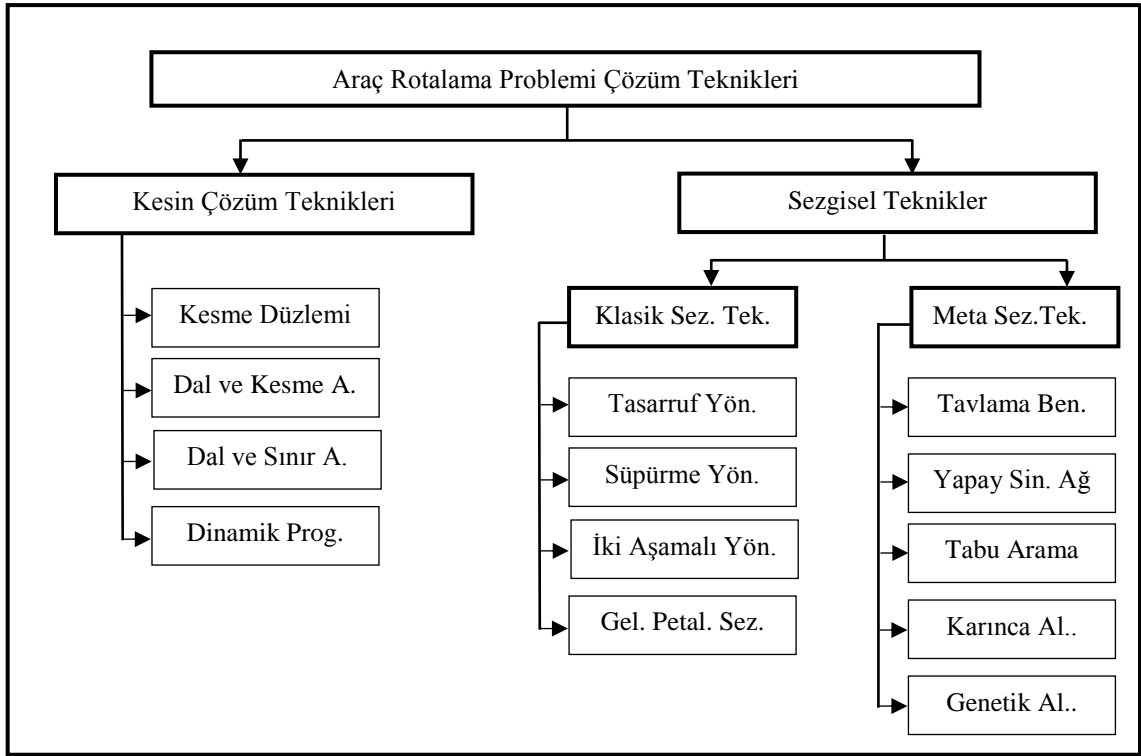
Geri toplamalı araç rotalama probleminden farklı olarak dağıtım ve toplama işlemleri sırasında bir öncelik sınırlaması bulunmadan eş zamanlı gerçekleştirilmektedir. Bu problem türünde dikkat edilmesi gereken nokta, dağıtım ve toplama işlemlerinin aynı müşteride aynı anda gerçekleştirilmesi sebebiyle dağıtılan ve toplanan ürün miktarına göre denge sağlanarak araç kapasitesinin hesaplanmasıdır.

Dağıtım ve toplamalı araç rotalama problem türünde amaç rotaların aşağıdaki kısıtlar sağlanarak bulunmasıdır (Yücel, 2016):

- Her bir rota depoyu ziyaret etmelidir.
- Her bir müşteri sadece bir rota tarafından ziyaret edilmelidir.
- Hiçbir aracın rota sırasındaki yükü araç kapasitesini geçmemelidir.

1.5. Araç Rotalama Problemi İçin Çözüm Teknikleri

Araç rotalama problemini çözmek için araştırmacılar tarafından pek çok teknik geliştirilmiştir. Bu çözüm teknikleri kesin çözüm teknikleri ve sezgisel teknikler olarak iki sınıfa ayrılır. Kesin çözüm teknikleri ile optimal çözüm garanti edilebilirken sezgisel teknikler ile optimuma yakın çözümler çok daha kısa sürede bulunabilmektedir (Keskintürk vd., 2015). Araç rotalama problemi için kullanılan başlıca çözüm teknikleri Şekil 1.4.'te gösterilmektedir.



Şekil 1.4. Araç rotalama problemi çözüm teknikleri (Demircioğlu, 2009 – Düzenleme yapılarak alınmıştır.)

1.5.1. Kesin teknikler

Araç rotalama problemi için kesin çözüm teknikleri gezgin satıcı problemi tekniklerinin geliştirilmesi ile oluşmuştur (Düzakın ve Demircioğlu, 2009). Optimum çözümler kesin çözüm teknikleriyle bulunur, fakat problemin büyüklüğüne göre problemin çözüm süresi üstel şekilde artar. Genel olarak küçük ve orta büyüklükteki problemlerin çözümünde daha etkindirler.

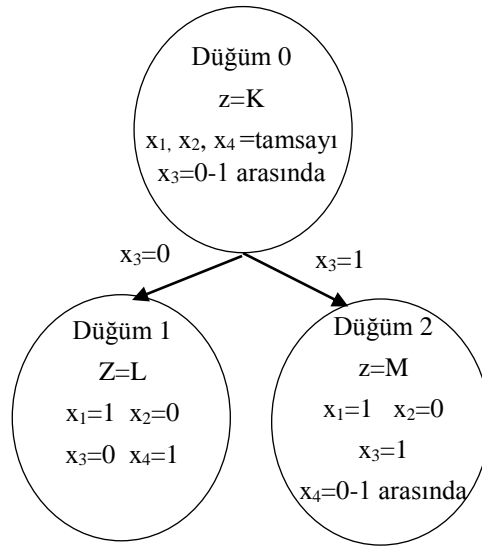
Kesme düzlemi: Kesme düzlemi tekniğinde ana fikir, doğrusal programlama ile elde edilmiş olan uygun çözüm alanındaki bazı parçaları kesip atmaktır, bu sayede tam

sayılı optimum çözüme simpleks tekniğiyle ulaşmaktır (Pan, 2015). Diğer bir ifadeyle tam sayıyı sağlamak için kısıtlar eklenir ve kısıtlar eklendiğinde elde edilen yeni çözüm tam sayılı ise optimum çözüme ulaşılmıştır. Eğer hala kesirli ise tam sayılı çözüm bulunana kadar kısıt eklemeye ve yeniden çözüm işlemine devam edilir (Keskindürk, vd., 2015).

Algoritmanın temelini oluşturan kesmenin iki önemli noktası vardır (Yücel, 2016):

- Tam sayılı programlama için uygun olan bir nokta kesmeyi sağlar.
- Gevşek biçim için en iyi olan nokta kesmeyi sağlamaz.

Dal ve sınır algoritması: 1960'ların başında önerilen bu teknik, olurlu çözümlerin sistematik bir şekilde sayılarak en iyi tam sayılı çözümü bulmayı amaçlayan doğrusal programlama (DP) tabanlı bir algoritmadır (Önsel, tarihsiz). Bu teknik, problemin optimum çözümünü araştırırken, problemle alakalı tüm aşamaları analiz etmektedir. Öncelikle, problemin olurlu çözümlerini daha küçük olan alt kümelere ayılırken her bir alt küme için alt ve üst sınırlar belirlenir. Sonrasında, bu değerlere bağlı olarak alt problemlerden bazıları çözümden atılır ve optimum çözüme ulaşılır (Öztürk, 2011). Dal-sınır algoritması örneği Şekil 1.5.'te gösterilmiştir.



Şekil 1.5. Dal-sınır algoritması örneği

Dal ve kesme algoritması: Dal-kesme tekniği, dal-sınır ve kesme tekniklerinin bir birleşimi şeklindedir. Öncelikle problemin DP modeli ile çözümü gerçekleştirilir. Bu aşamada amaç fonksiyonu ve kısıtlar yazılarak model oluşturulur. Çözüm esnasında

oluşabilecek alt rotalar sıfıra (0) eşitlenerek problem dallara ayrılır. Daha sonra karşılaşılabilecek dallarda ise araç adedi gibi kısıtları sağlayabilmek için model alt rota engelleme kısıtlayıcısı ilave edilerek optimum sonuca ulaşılmaya çabalanır (Keskintürk, vd., 2015).

Dinamik programlama: Dinamik programlama temel olarak, çözümü zor olan problemleri daha küçük alt problemlere ayırarak sonuca ulaşmaya çalışan bir optimizasyon tekniğidir.

1.5.2. Sezgisel teknikler

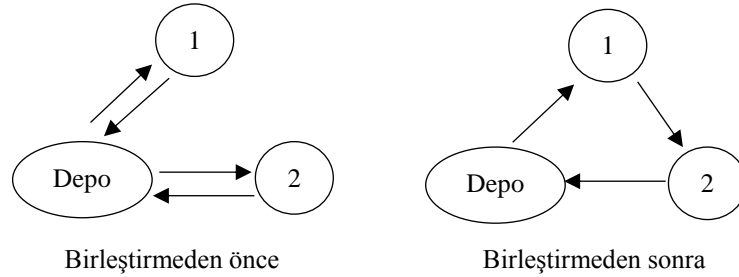
Araç rotalama probleminde kesin çözümünün belirlenmesi oldukça karmaşık bir iştir. Kısıtlı bir sürede iyi kalitede çözümlerin bulunabilmesi için sezgisel teknikler tercih edilebilmektedir. Sezgisel teknikler, herhangi bir hedefi gerçekleştirmek veya sonuca ulaşmak için çeşitli alternatiflerden etkili olanlara karar vermek amacıyla tanımlanan kriterler veya bilgisayar metotlarıdır. Bu tür teknikler en iyi çözümü garanti etmezler, fakat en iyi çözüme yakın bir çözüme ulaşabilmektedirler (Erol, 2006).

Geçmişten bugüne kadar araç rotalama probleminin çözümü için pek çok sezgisel teknik geliştirilmiştir. Geliştirilen bu sezgisel teknikler, 1960-1990 yılları arasında geliştirilen klasik sezgisel teknikler ve son yıllarda büyük gelişim gösteren meta sezgisel teknikler olarak iki sınıfa ayrılabilir (Şahin ve Eroğlu, 2014). Meta sezgisel teknikleri klasik sezgisel tekniklerden farklı kılan temel nokta, kötüye giden çözümlerin veya uygun olmayan çözümlerin bazı olasılıklara göre arama sürecine dahil olmalarına izin vermesidir. Bu sayede, algoritmanın yerel optimuma takılmaması sağlanmaktadır (Günay, 2013).

Klasik sezgisel teknikler, rotaların yapımından ve geliştirilmesinden oluşmaktadır. Clark ve Wright (1964) tarafından önerilen tasarruf tekniği, Gillet ve Miller (1974) tarafından önerilen süpürme tekniği, Christofides ve arkadaşları (1979) tarafından geliştirilen iki aşamalı teknik ve Renaud ve arkadaşları (1996) tarafından önerilen petal tekniği klasik sezgisel tekniklerdir.

Tasarruf tekniği: Araç rotalama problemlerini çözmek için geliştirilen tekniklerden birisi olan ve Clarke ve Wright tarafından 1964 yılında geliştirilen rota oluşturma sezgiselidir. Kolay anlaşılabilmesi ve diğer araç rotalama tekniklerine göre daha esnek olması nedeniyle geniş bir kullanım alanına sahiptir (Düzakın ve Demircioğlu, 2009). Tasarruf tekniği, rotaların birleştirilmesiyle kazanılacak olan maliyet tasarrufuna dayanır

ve iki müşterinin farklı rotalar yerine aynı rotada sırayla hizmet görmesiyle maliyette sağlanacak azalmanın tahminidir (Günay, 2013). Şekil 1.6. tasarruf algoritmasını temsil etmektedir.



Şekil 1.6. Tasarruf algoritması

Süpürme tekniği: Gillet ve Miller (1974) tarafından geliştirilmiştir ve iki aşamalı bir tekniktir. İlk aşamada, her kümeye bir araç atanacak şekilde öncelikle müşteriler kümelere ayrılır. Sonrasında her kümedeki müşteri grubunun rotalarının oluşturulmasında gezgin satıcı problemi olarak yaklaşılır (Günay, 2013). Tekniğin uygulanmasında, her bir noktanın polar koordinatları $i = 1, \dots, n$ için (r_i, θ_i) belirlenir ve depo orijin noktası $r_0 = 0$ ve $\theta_0 = 0$ olarak kabul edilir. Koordinatlar θ_i temel alınarak artan sırayla dizilir (Düzakın ve Demircioğlu, 2009).

İki aşamalı teknik: Bu teknik, kapasite kısıtlı araç rotalama problemlerinin çözülmesi amacıyla geliştirilmiştir (Düzakın ve Demircioğlu, 2009). İlk aşamada, noktalar kapasiteyi aşmayacak şekilde araçlara atanır, sonrasında ise her bir araç için gezgin satıcı problemi sezgiselleri kullanılarak rota belirlenir (Keskintürk vd., 2015).

Geliştirilmiş petal tekniği: Petal sezgiseli, araç rotalama problemleri için ilk olarak Foster ve Ryan (1976) tarafından önerilmiş, sonrasında 1993 yılında Ryan ve arkadaşları tarafından geliştirilmiştir. Geliştirilmiş petal sezgiseli ise Renaud ve arkadaşları tarafından 1996 yılında önerilmiştir. Bu sezgisel, petal tekniği ile turların oluşturulması ve kolon yenileme işlemine göre en iyi seçimin yapılmasıdır. Bu sezgisel ile kısa sürede optimuma yakın sonuçlar elde edilmektedir (Düzakın ve Demircioğlu, 2009).

Genetik algoritma, karınca kolonisi, tavlama benzetim, yapay sinir ağları ve tabu arama teknikleri önde gelen meta-sezgisel tekniklerdir. Aşağıda bu tekniklerin detaylarına yer verilecektir.

Tavlama benzetim (TB): Bu teknik, katıların fiziksel tavlama sürecinden esinlenilerek, Kirkpatrick ve arkadaşları (1983) tarafından geliştirilmiş meta-sezgisel

tekniklerden biridir. Fiziksel tavlama işlemi, bir ısı banyosundaki bir katı maddenin eritildiği, daha sonra eritilmiş kristal maddeyi donma haline gelene kadar uygulanan soğutma tekniğidir. Bu soğutma işleminde sıcaklık azalışı ani bir şekilde gerçekleştirilip, sıcaklık hızlı bir şekilde sıfıra düşürülürse, oluşan kristalin yapısında bozukluklar olabilmektedir. Bunu engelleyebilmek için ortam sıcaklığı yavaşça, kademeli olarak düşürülmelidir (Osman ve Kelly, 1996).

Gerçekte, maddenin sıvı halinde parçacıkların rastgele hareketi söz konusudur. Katı halinde ise, bir kafese girmiş şekilde düzenlenir ve bu durumda katının enerjisi minimum olur. Bu duruma yer durumu denir. Bir katı için yer durumu, sıcaklık değeri yeteri kadar yükselttilip, düşürülmüş ve bu soğutma gerektiği gibi yavaş yapılmış ise söz konusu olabilmektedir. Aksi takdirde yarı kararlı bir yapıda donma gerçekleşir. En iyileme problemleriyle tavlama süreci arasında ilişki şu şekildedir (Güden v.d., 2005):

- Katının farklı olabilen fiziksel durumları, problemin mümkün çözümlerine,
- Sistemin enerjisi, problemin amaç fonksiyonuna,
- Bir durumun enerjisi, bir çözümün amaç fonksiyonu değerine,
- Yarı kararlı durum, problemin yerel optimum çözümüne
- Yer durumu, problemin global optimum çözümüne karşılık gelir.

Bu algoritma, bilgisayar tasarımı, moleküler fizik ve kimya, çizelgeleme ve araç rotalama gibi birçok probleme uygulanabilmektedir (Tüfekçier, 2008).

Yapay sinir ağları: Yapay sinir ağları (YSA), insan beyninin işleyişinden esinlenen meta-sezgisel bir tekniktir. Sinir ağları seti belirli bir ağ ile birleşmektedir. Sinir bir girdi olarak çıktıyı hesaplar ve sonrasında diğer sinirlere sinyal gönderir. Genel olarak, gezgin satıcı problemi için yapay sinir ağları iki sınıfa ayrılır. Birinci sınıfta sinirler tam sayılı programlama şeklinde organize edilir, ikinci sınıfta ise sinirler çözüm uzayında noktalar şeklinde görülür (Demircioğlu, 2009).

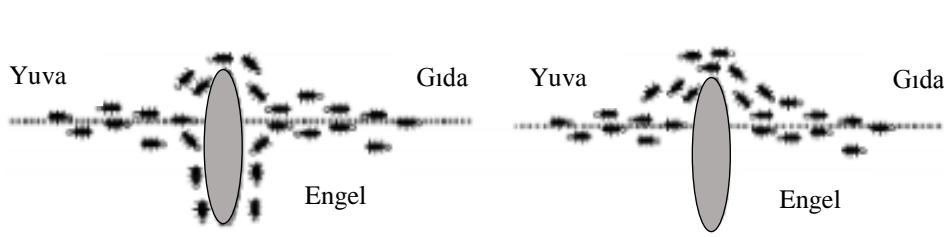
Tabu arama (TA): Bu teknik, optimizasyon problemlerinin çözümü için Glover (1989) tarafından geliştirilen meta-sezgisel bir tekniktir. Tekniğin yapısı akıllı problem çözme genel prensiplerine dayanmaktadır (Osman ve Kelly, 1996).

Tabu arama, kendi yapay hafızasında bulunan aramanın geçmişine ait bazı çözümleri hafızasında tutar ve bazı çözümlerin aramaya katılmasına yasak koyarak algoritmanın yerel optimuma takılmasını engellemektedir (Günay, 2013). Tabu arama yasaklı listeyi temel alarak çalışmaktadır. Bu liste düzenli olarak güncellenir ve

algoritmanın sonraki döngülerde tekrar etmemesi sağlanır. Genel olarak çözümler döngüsel olarak önce kötüleşir, sonra iyileşme gösterir (Keskintürk vd., 2015).

Çözüm kalitesi ve çözüme kısa sürede ulaşılabilmesi sebebiyle tabu arama algoritması araç rotalama problemlerinin bütün çeşitleri için en çok kullanılan tekniklerdendir (Şahin ve Eroğlu, 2014).

Karınca kolonisi: Bu algoritma, Marco Dorigo (1991) tarafından karıncaların yiyecek bulma mekanizmalarından esinlenilerek geliştirilen bir meta sezgisel tekniktir. Karıncaların belirli bir mantığa göre hareket ettiğine inanan Dorigo, karınca kolonisi algoritmasını gezgin satıcı problemi ve araç rotalama problemi gibi optimizasyon problemlerine uygulamıştır (Düzakın ve Demircioğlu, 2009). Bu algorithmada, karıncalar yuva ile yiyecek arasındaki en kısa mesafeyi belirlemek için kendi aralarında haberleşme imkânı sağlayan bir koku bırakırlar ve bu kokunun yoğun olduğu yerler en kısa mesafeyi ifade etmektedir. Arkadan gelen diğer karıncalar kokunun fazla olduğu yolları tercih etme eğilimindedirler ve bu ihtimal daha kuvvetlidir (Şahin ve Eroğlu, 2014). Şekil 1.7.'de karıncaların yuvaları ve yiyecekleri arasında bir süre sonra en kısa yolu belirlemesi örneği verilmiştir.



Şekil 1.7. Gerçek karınca davranışları

Genetik algoritma: Genetik algoritma doğadaki evrim sürecine benzer şekilde tasarlanan ve iyi olanın hayatta kalması ilkesi üzerine kurulmuş olan bir tekniktir. Genetik algoritmanın işleyişi de doğal evrim mekanizmasına benzer şekilde çalışır. Genetik algoritma, verilerin genler olarak kodlanması, bu genler üzerinde bilgisayar ortamında çaprazlama ve mutasyon işlemlerinin uygulanması ve bu işlemler sırasında istenilen sonuca uygun şekilde amaç fonksiyonunun verilmesi ile sonuca ulaşma işleyişine göre çalışmaktadır (Çalışkan, Yüksel ve Dayık, 2016).

1.6. Araç Rotalama Probleminin Uygulama Alanları

Araç rotalama problemi çoğunlukla bir ağ içerisindeki belirli noktalar arasında ürün ve hizmet dağıtımı ile uğraşmaktadır. Günümüzde ürün sevkiyatında, mal ve insan taşımacılığındaki problemler giderek artmaktadır. Örnek olarak:

- Ürün ve hizmetlerin bir ya da birden fazla sayıda depodan, çeşitli müşteri konumlarına dağıtımının yapılması,
- Üretim planlaması ve üretilen mamul, yarı mamul, hammaddelerin fabrikalar arasındaki sevkiyatı,
- Ürünlerin mağazalara taşınması ve stok planlaması,
- Para dağıtımı,
- Barlara ve lokantalara içecek dağıtımı,
- Süt dağıtımı ve toplanması,
- Akaryakıt dağıtımı,
- İnternette yapılan alışverişlerdeki teslimatlar,
- Çöplerin toplanması ve taşınması,
- Merkez depodan mağazalara ürün dağıtılması,
- Posta hizmetleri,

gibi günlük yaşamda çok sık karşılaşılan problemler verilebilir (Demircioğlu, 2009).

İKİNCİ BÖLÜM

2. GENETİK ALGORİTMA

Bu bölümde, meta-sezgisel çözüm tekniklerinden biri olan ve evrimsel süreci taklit ederek problemlere çözüm üretmeyi amaçlayan genetik algoritmanın detaylarına yer verilecektir. Genetik algoritmanın tanımı, tarihçesi, önemi, temel kavramları ve işleyiş sürecine yer verilerek, bir problemin genetik algoritma ile çözüm aşamaları anlatılmaya çalışılacaktır. Sonrasında genetik algoritmanın avantajları, dezavantajları, kullanım nedenleri ve uygulama alanlarına değinilecektir.

2.1. Genetik Algoritmanın Tanımı, Tarihçesi ve Önemi

İnsanoğlu tarihinin derinliklerinden bu yana doğa ile iç içe hayatını sürdürmektedir. Bu birliktelikte insanoğlunun doğayı ve doğadaki canlıları incelemesi sonucu birçok icat ve teknik geliştirilmiştir. Genetik algoritma bu tekniklerden biri olmakla birlikte en yenilerindedir. Genetik algoritmanın esası doğal seçim ve genetik kurallarına dayanmaktadır. Bu kurallar, iyi nesillerin kendi yaşamlarını korurken, kötü nesillerin yok olması ilkesine dayanmaktadırlar. Genetik algoritma bu iki kuralı bir arada kullanarak en iyiyi aramayı hedefleyen bir eniyileme tekniğidir (Şen, 2004).

“Genetik algoritma” ifadesi ilk defa Bagley’in bir çalışmasında kullanılmıştır (Bagley 1967). Daha sonra makine öğrenmesi konusunda çalışan Holland (1975), canlılardaki genetik işlemleri sanal ortamda gerçekleştirmeye çalışarak bu işlemlerin etkinliğini açıklamıştır. Aynı yıl içerisinde yapılan De Long’un (1975) çalışmasında, genetik algoritmanın eniyileme amaçlı olarak kullanılabilceği çeşitli deneyler üzerinde gösterilmiştir. Başlangıçta pratik bir yararı olmadığı düşünülen genetik algoritmaya ilgi, Holland’ın öğrencisi olan Goldberg’in yaptığı doktora teziyle National Science Foundation tarafından Genç Araştırmacı ödülü verilmesiyle artmıştır (Goldberg, 1989). Bu çalışmaların ardından Genetik algoritma çalışmaları hızlı bir artış göstermiştir. Son yıllarda da problemlerdeki karmaşıklığı azaltma ve kısa sürede iyi çözümler elde etme özelliği dolayısıyla etkin bir optimizasyon aracı haline gelmiştir (Gupta ve Imtavanich, 2010).

Genetik algoritma genel anlamda, dizilerden oluşan bir popülasyona çoğalma, çaprazlama ve mutasyon operatörlerinin uygulanmasını içermektedir. Bu operatörlerin

uygulanmasından sonra yeni bir popülasyon oluşur. Yeni popülasyon eski popülasyon ile yer değiştirir. Her dizinin bir uygunluk değeri vardır. Diziler uygunluk değerine göre seçilirler. Ortalama uygunluk değeri üzerinde uygunluk değerine sahip olan kromozomların gelecek nesillere aktarılma olasılığı daha yüksektir. Evrim süreci, popülasyonun ortalama uyumunu giderek artırır ve ilerleyen nesillerde daha iyi uygunluk değerinin elde edilmesini sağlar (Taşkın ve Emel, 2009). Goldberg 'in tanımına göre, genetik algoritma rastlantısal arama tekniğini kullanarak çözüm bulmaya çalışan, parametre kodlama esasına dayanan sezgisel bir arama tekniğidir (Elmas, 2016).

Genetik algoritmanın önemini açıklayabilmek adına klasik eniyileme tekniklerine göre bazı üstünlükleri şu şekilde sıralanabilir:

Genetik algoritma problemlere tek bir çözüm üretmek yerine farklı çözümlerden oluşan bir çözüm kümesi üretir. Bu sayede, arama uzayında aynı anda birçok nokta değerlendirilmekte ve sonuç olarak bütünsel çözüme ulaşma olasılığı yükselmektedir. Çözüm kümesindeki çözümler birbirinden tamamen bağımsızdır. Her biri çok boyutlu uzay üzerinde bir vektördür (Özkan, 2008).

Genetik algoritmada çözümün uygunluğu onun seçilme şansını artırır ancak seçilmesini garanti etmez. Seçim işlemi de başlangıç popülasyonunun oluşturulması gibi rastgeledir, ancak bu rastgele işlemde çözüm adaylarının seçilme olasılıklarını çözümlerin uygunluk değerleri belirler (Özkan, 2008). Böylelikle iyi uygunluk değerine sahip çözümlerin seçilme olasılığı yüksekken, kötü diye ifade edilen uygunluk değerlerine sahip çözümlerin de değerlendirilmesi mümkün olmaktadır.

Genetik algoritma sadece bir çözüm noktası değil, belirli bir aday grup çözüm noktaları üzerinde çalışır. Genetik algoritmada olası çözümlerin tümü hemen yaratılmaz. Optimum ya da optimuma yakın çözüme, olası çözümlerin bir bölümü üzerinden gidilir. Genetik algoritma deterministik kuralları değil, olasılık üzerine kurulu kuralları kullanan tekniklerdir. Genetik algoritma paralel çalışır çünkü aynı anda problem uzayının birden çok noktası üzerinde arama yaparlar (Özkan, 2008). Bu durum klasik tekniklerde tek nokta ve yön olarak karşımıza çıkar.

Genetik algoritma arama uzayında ilerleyebilmek için sadece ilgili noktanın uygunluk değerini kullanırlar. Böylelikle sonuca ulaşmak için türev ve diferansiyel işlemler gibi yardımcı başka bilgiler kullanmaya gerek kalmaz. Genetik algoritma, ne yaptığı konusunda bilgi içermez, nasıl yaptığını bilir. Bu sebeple kör bir arama tekniğidir.

Diğer tekniklerde olduğu gibi doğrudan parametreler üzerinde çalışmaz. Genetik algoritma, optimize edilecek parametreleri kodlar ve bu kodlar üzerinde işlem yaparak parametre kodlama esasına göre çalışır.

Genetik algoritma, karmaşık problemlerin çözülmesinde etkilidir. Kesin çözümlere ulaşamayan veya matematiksel olarak modellenmesi yapılamayan problemlerde genetik algoritmadan yararlanır. En iyi çözümü garanti etmez ama kısa sürede optimuma yakın çözüme ulaşabilmektedir (Kırda, 2013).

2.2. Genetik Algoritmanın Temel Kavramları

Genetik algoritmanın amacı evrimsel süreci taklit ederek problemlere çözüm üretebilmek olduğu için genetik algoritmadaki birçok kavram genetik bilimindeki karşılıklarıyla benzerlik göstermektedir. Genetik algoritmanın çalışmasında ve başarılı çözüm değerlerine ulaşmasında kullanılan kavram ve bu kavram değerlerinin iyi belirlenmesi gerekmektedir. Aşağıda bu kavramların incelenmesine yer verilmiştir.

2.2.1. Gen

Gen, bir canlının (bireyin) kalıtsal özelliklerinden herhangi bir tanesini taşıyan parçadır. Her gen, her biri bir özelliği gösteren kromozomların, belirli konumlarında bulunan ve karakterlerin belirlenmesinde etkili olan birimlerdir. Genlerin bir araya gelmesiyle kromozomu yani olası çözümlerden birini oluştururlar (Yücel, 2016). Genler, genetik algortmada çözümün bir özelliğini temsil etmektedirler. Genetik algoritmanın kullanıldığı programlama yapısında, gen yapıları programcının tanımlamasına bağlıdır. Bu nedenle yazılan programa göre gen içeriği çok önemlidir (Elmas, 2007).

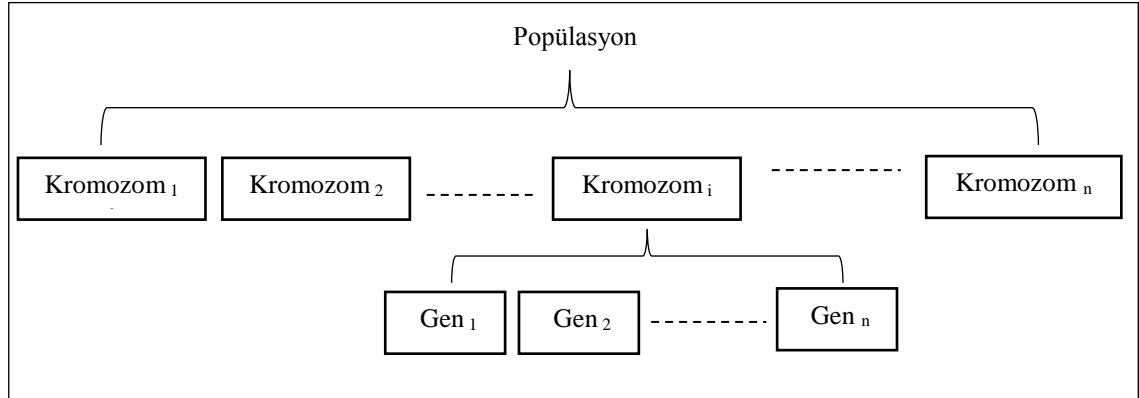
2.2.2. Kromozom (Birey)

Topluluğu oluşturan canlılardan her birine birey ya da kromozom denir. Kromozomlar, bir veya daha fazla genin bir araya gelmesiyle oluşurlar ve genlerinde bulunan özellikleri taşırlar. Kromozomların bir araya gelmesiyle de popülasyon oluşur. Genetik algortmada ise her kromozom problem için olası çözümleri temsil eder ve problemin çözümünde en önemli birim olduğu için bilgisayar ortamında iyi ifade edilmesi gerekmektedir.

2.2.3. Popülasyon

Aynı tür içinde yer alan canlıların oluşturduğu bireyler topluluğudur. Genetik algorithmada popülasyon kavramı, olası çözümlerden oluşan topluluğu temsil etmektedir. Popülasyondaki kromozom (birey) sayısı problemin yapısına göre programcı tarafından belirlenir ve nesiller boyunca sabit tutulur.

Popülasyon büyüklüğünün doğru belirlenmesi, genetik algoritmanın başarısında ve çözüm süresinde önemli etkisi olan faktörlerden biridir. Bu büyüklüğün gereğinden küçük olarak belirlenmesi iyi çözümlerin popülasyondan ayrılması riskini taşıırken, gereğinden büyük olarak belirlenmesi de hesaplamaların çok fazla zaman almasına neden olarak sonuca ulaşmayı zorlaştıracaktır. Bu nedenle, popülasyon büyüklüğü belirlenirken çözüme yavaş ulaşılmasını ve iyi çözümlerin popülasyondan ayrılmasını engelleyecek bir denge sağlanmalıdır. Yukarıda anlatılan kavramlar şematik olarak Şekil 2.1.'de gösterilmiştir.



Şekil 2.1. Gen, kromozom ve popülasyonu gösteren yapı (Pulat ve Karakoç, 2016)

2.2.4. Uygunluk fonksiyonu ve değeri

Genetik algorithmada, popülasyon içindeki iyi özelliklere sahip kromozomların bir sonraki nesile aktarılması istenir. Bunun için de problemin amacıyla orantılı olarak oluşturulan uygunluk fonksiyonu kullanılarak bir değerlendirme yapılır.

Uygunluk değeri, çözümün kalitesini belirler ve uygunluk fonksiyonu kullanılarak hesaplanır. Fonksiyonun sonucu olarak ortaya çıkan değer, ilgili kromozomun uygunluk değerini temsil etmektedir. Yüksek değerlere sahip kromozomların bir sonraki nesile aktarılma olasılığı daha fazladır. Yani, iyi olanların korunarak daha iyiye götürebilecek

olmasından dolayı, algoritmanın gücünü oluşturan en önemli faktörlerden biri olmaktadır (Yücel, 2016).

2.2.5. Genetik operatörler

Genetik algoritmanın en basit formu üç tip genetik operatör içerir: seçim, çaprazlama ve mutasyon.

Seçim: Bu operatör popülasyondaki kromozomları üreme için seçer. Genellikle seçim kriteri uygunluk değeridir. Yüksek uygunluk değerine sahip olan kromozomların üremek için seçilme ihtimali daha kuvvetlidir. Bu adımda kromozomlardan hangilerinin bir sonraki nesile aktarılacağına karar verilmektedir, bu nedenle önemlidir.

Çaprazlama: Bu operatör, eşleştirme havuzundaki kromozomlar arasında genlerin değişimidir. Aslında, çaprazlamanın birçok farklı yolu vardır. Bu, problemin yapısına göre özel olarak tasarlanmaktadır.

Algoritmanın başında belirlenen belirli bir oranda yapılan bu işlem sonucunda farklı özelliklere sahip yeni kromozomlar oluşmaktadır. Bu oranın çok yüksek olması nesiller içinde yeni dizilerin daha hızlı oluşmasını sağlarken yüksek performanslı dizilerin de hızlı bir şekilde çözümden ayrılmasına neden olabilir. Diğer yandan, çaprazlama oranı çok düşük olduğunda değişime uğrayacak dizi sayısı az olacağından problem durgunlaşır yavaşlayabilir (Kahvecioğlu, 2003). Bu sebeple, çaprazlama oranı uygun bir değerde seçilmelidir.

Çaprazlama olasılığı çaprazlamanın hangi sıklıkla yapılacağını ifade etmektedir. Eğer çaprazlama oranı %0 ise çaprazlama yapılmaz ve yeni kromozomlar eski kromozomların aynısı olur ama bu yeni kuşağın eskisiyle aynı olacağı anlamını taşımaz. Eğer bu oran %100 olursa, yeni kromozomlar tamamen çaprazlama ile elde edilir. Çaprazlama sonucunda yeni kromozomların daha iyi olması umut edilir (Kurt ve Semetay, 2001).

Mutasyon: Bu operatör kromozomdaki bazı genleri rastgele olarak ters çevirir. Gen diziliminde ufak değişimlerin meydana gelmesini sağlayan bu operatör önceki adımlarda kaybedilen yararlı genlerin kurtarılması ve popülasyonda çeşitliliğin sağlanması için tasarlanmıştır.

Mutasyon işlemi gerçekleştirilirken belirli bir oran doğrultusunda bir veya daha fazla genin değiştirilerek, farklı bir kromozom oluşturulması sağlanır. Mutasyon oranı %0 ise mutasyon olmaz ve yeni kromozom çaprazlama sonrasında olduğu gibi kalır. Eğer

mutasyon oranı %100 olursa popülasyon içindeki kromozomlar tamamen değişir (Kurt ve Semetay, 2001). Mutasyon oranı genelde 0,01 gibi çok düşük bir olasılıkla gerçekleştirilir. Eğer mutasyon oranı çok büyük olursa, kromozomların gereğinden fazla değiştirilmesine neden olur ve çözümün optimuma yaklaşması zorlaşabilir (Kırda, 2013).

2.2.6. Sonlandırma kriteri

Sonlandırma kriteri, genetik algoritma sürecinin hangi durum ya da durumlarda durması gerektiğini belirleyen bir koşuldur. Karar verici tarafından belirlenen sonlandırma kriteri, yineleme sayısı (iterasyon sayısı) olabileceği gibi probleme özgü başarı değerleri ya da arama süreside olabilmektedir.

Karar vericinin tercih bilgisinin çözüm sürecinin başında istendiği yaklaşımlar ile optimum çözümlere ulaşmak daha kısa zamanda ve daha kolay olmaktadır. Bu tür yaklaşımların en büyük sakıncası ise, karar vericinin tercih bilgisini belirlemede yaşadığı güçlülüdür. Bunun nedeni, bir anlamda belirsizlik altında tercih yapmaktır (Atlas, 2008).

2.2.7. Genetik algoritmanın aşamaları ve akış şeması

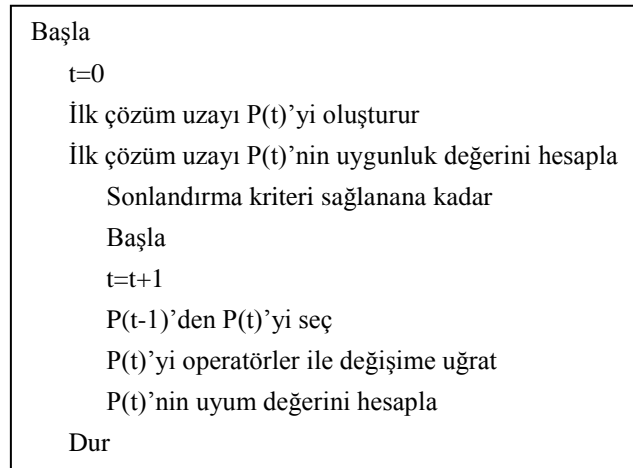
Genetik algoritma genel olarak aşağıdaki çözüm aşamalarını gerçekleştirir:

1. n adet kromozomdan oluşan rastgele bir popülasyon oluşturulur.
2. Popülasyondaki her kromozomun uygunluk değeri hesaplanır.
3. İlk popülasyon büyüklüğüne (n adet kromozom) eşit olan yeni popülasyon oluşturuluncaya kadar aşağıdaki adımlar tekrar edilir.
 - a) Mevcut popülasyondan seçim teknikleri yardımıyla çaprazlama uygulanacak kromozomlar seçilir.
 - b) Belirlenmiş olan çaprazlama oranı ile çaprazlama işlemi gerçekleştirilip seçilen kromozomların yerine yeni kromozomlar oluşturulur.
 - c) Mutasyon oranı ile yeni oluşturulan kromozomlara mutasyon uygulanır. Mutasyon uygulanmış olan yeni kromozomlar yeni popülasyonda yer alır.
4. Yeni popülasyon mevcut popülasyon ile yer değiştirilir.
5. 2.aşamaya geri dönülür.

Yukarıda anlatılan aşamalar genetik algoritmanın çoğu uygulaması için temel niteliğindedir. Bu aşamaların uygulanması sırasında dikkate alınması gereken bir takım ayrıntı vardır: Çözümlerin nasıl kodlanacağı, popülasyon büyüklüğünün ne olacağı,

seçim, çaprazlama ve mutasyon operatörlerinin ayrıntıları ve oranları, sonlandırma kriteri gibi. Algoritmanın başarısı çoğunlukla bu ayrıntılara bağlıdır (Mitchell and Taylor, 1999).

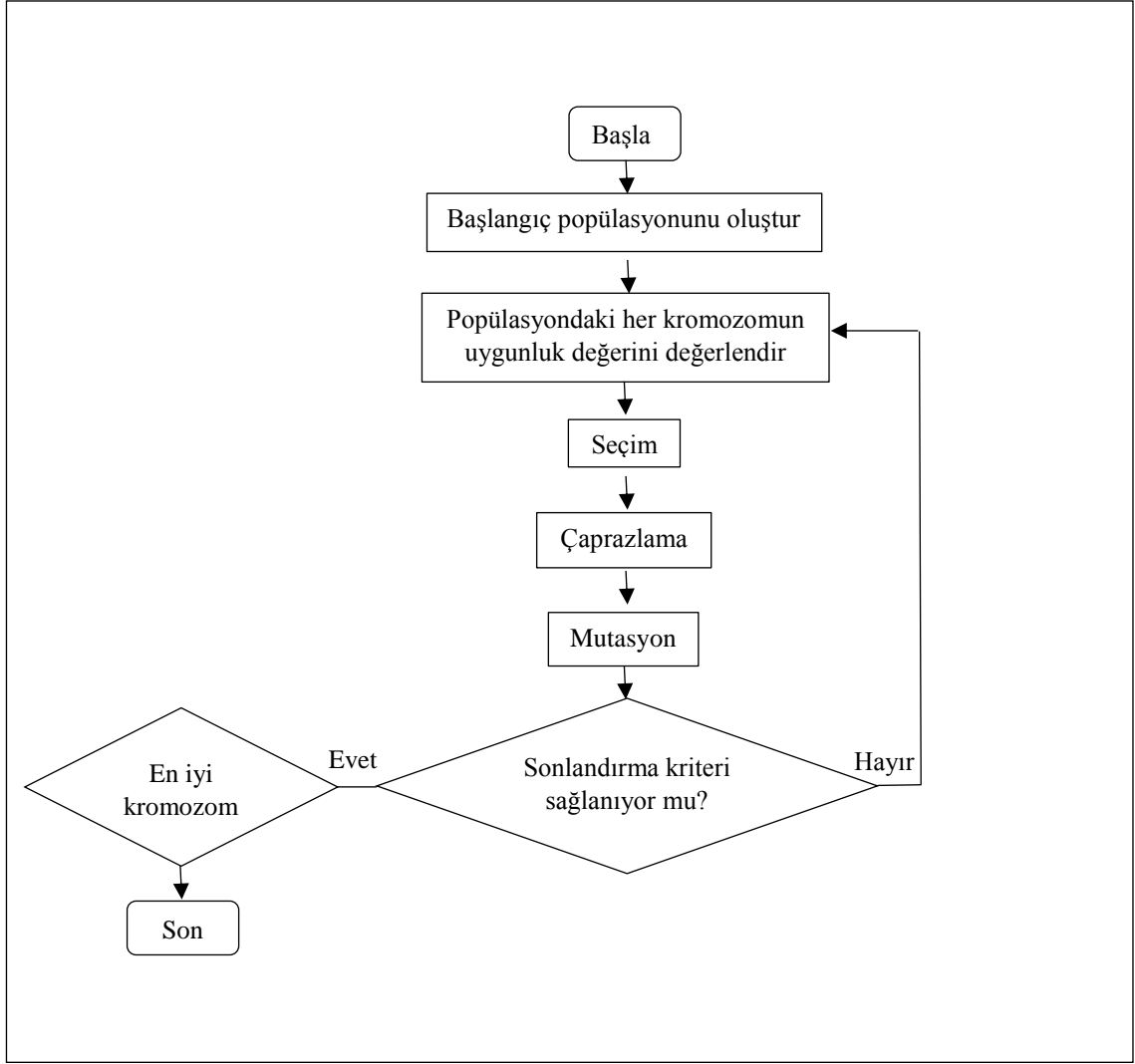
Genetik algoritma süreci için bilgisayar programlarından yararlanılmaktadır. Basit genetik algoritma genel yapısının bilgisayar programında kodlanmasına yönelik yapay kodu Şekil 2.2.'de verilmiştir.



Şekil 2.2. Genetik algoritmanın genel yapısı (Dasgupta ve Michalewicz, 1997)

Yapay kodda öncelikle başlangıç popülasyonunun oluşturulması gerektiği görülmektedir. Popülasyonun ilerleyen yinelemelerde değiştiğini ifade etmek için t sembolü kullanılmıştır. Başlangıç popülasyonu $P(0)$, t anındaki popülasyon da $P(t)$ olarak gösterilmiştir. Sonraki adımda her kromozomun uygunluk değeri hesaplanır. Uygunluk değerine göre kromozomlar seçilerek yeni bir popülasyon oluşturulur. Oluşturulan yeni popülasyonun bazı kromozomları genetik operatörler yardımıyla değişime uğratılır. Yeni popülasyonun uygunluk değeri hesaplanır ve tüm bu adımlar sonlandırma kriteri sağlanana kadar sürdürülür.

Genetik algoritmanın temel işleyişine ait akış şeması ise Şekil 2.3.'de verilmiştir. Akış şemasında görüldüğü gibi genetik algoritmanın ilk adımı uygun kodlamanın yapılmasıdır. Sonraki adımlar ise uygunluk değerinin hesaplanması ve temel genetik operatörlerin uygulanması şeklinde devam etmektedir. Sonlandırma kriteri sağlandığında algoritma sona erdirilir. Bu adımlar genetik algoritmanın yapay kodu ile benzer olduğu için tekrar değinilmeyecektir.



Şekil 2.3. Genetik algoritma akış şeması

2.3. Genetik Algoritmanın Adımları

2.3.1. Kodlama

Genetik algoritmada kullanılan temel mekanizmalardan biri, problemin olası çözümlerini temsil eden kromozomların kodlanmasıdır. Kodlama ile probleme özgü bilgilerin genetik algoritmanın kullanabileceği bir şekle dönüştürülmesi sağlanır (Günay, 2013).

Bir problemin çözümü için genetik algoritma uygulamaya başlamadan önce problem değişkenlerinin aynı boyutlara sahip diziler şeklinde kodlanması gerekmektedir. Dizilerden her biri çözüm uzayında olası rassal bir noktayı temsil etmelidir. Kodlama,

problemin yapısına göre değişiklik göstermektedir. İkili kodlama, değer kodlama ve permutasyon kodlama kullanılmaktadır. Kodlamanın doğru seçilmesi genetik algoritmanın performansı üzerinde oldukça etkilidir, aynı zamanda diğer genetik operatörlerin uygulanabilirliğini de etkilediği için ayrı bir önem taşımaktadır. (Rothlauf, 2006).

İkili (Binary) kodlama: İlk kez Holland (1975) tarafından önerilen ikili kodlama tekniği, genetik algoritma uygulamalarında en eski ve en yaygın olarak kullanılan gösterimlerden biridir. Çözüm uzayındaki değerlerin ikili sayma sistemine geçirilmesiyle gerçekleştirilir (Yücel, 2016). İkili kodlamada, her kromozom 0 veya 1 karakter dizilerinden oluşmaktadır. Tüm değerlerin temsil edilebileceği şekilde, sabit bir dizi uzunluğu belirlenmelidir. İkili kodlama ile gerçekleştirilen örnek bir kromozom gösterimi Şekil 2.4.'te verilmiştir.

Kromozom A	10110010110010101110
Kromozom B	11111110000011000001

Şekil 2.4. İkili kodlama gösterimi

İkili kodlama tekniği, çok sık kullanılan bir teknik olmasına rağmen bazı sakıncaları vardır. Örneğin, çok değişkenli fonksiyon eniyilemesi için değişkenlerin alt ve üst sınırlarına bağlı olarak elde edilen dizi çok uzun olabilir. Aynı zamanda gezgin satıcı, çizelgeleme, kareli atama gibi kombinatoryal eniyileme problemlerinde ikili düzende kodlama, araştırma uzayını tam olarak temsil edememektedir (Duman, 2007).

Değer kodlama: Problemin gerçek değerlerinin doğrudan kullanıldığı kodlama şeklidir. Kodlanan karakterler tamsayılar, ondalıklı sayılar, harfler ya da ilgili problem için anlamlı sözcüklerden oluşabilir. Bu tip problemler için ikili kodlama çok zordur ve hesaplama karmaşıklığının artmasına neden olmaktadır. Şekil 2.5.'te değer kodlamanın alabileceği değerleri gösteren örnek kromozom yapıları verilmiştir.

Kromozom A	ADBHNJMKLERTY
Kromozom B	(1,234), (5,678), (4,678)
Kromozom C	(ön), (sağ), (ön), (arka)

Şekil 2.5. Değer kodlama gösterimleri

Değer kodlama gösterim şeklinin karmaşık değerler içeren bazı problemlerde kullanımı çok uygun olmasına rağmen, kullanıldığı probleme özgü çaprazlama ve mutasyon operatörleri tanımlamak gerekmektedir (Sivanandam ve Deepa, 2008).

Permutasyon kodlama: Genellikle sıralama problemlerinde permutasyon kodlama kullanılır. Bu kodlamada her kromozom dizideki bir sırayı temsil eden sayılardan oluşur. Permutasyon kodlama, genellikle iş çizelgeleme, gezgin satıcı problemi, araç rotalama problemi gibi sıralamanın önemli olduğu problemlerin çözümde kullanılır. Şekil 2.6.'da permutasyon kodlama örneği verilmiştir.

Kromozom A	135792468
Kromozom B	123498765

Şekil 2.6. Permutasyon kodlama gösterimi

Permutasyon kodlama yaklaşımda dikkat edilmesi gereken husus, kromozomlar için uygun çaprazlama ve mutasyon operatörlerinin geliştirilmesi olacaktır. Kromozom diziliminde değişkenlerin birbirini tekrar etmemesi ve problemin niteliğine göre gerçek bir sıralama içermesi gerekir. Kromozomlara uygulanacak genetik operatörler ile de bu kuralın bozulmaması sağlanmalıdır (Günay, 2013).

2.3.2. Başlangıç popülasyonunun oluşturulması

Problemin olası çözümlerini temsil eden kromozomların nasıl kodlanacağı belirlendikten sonra, bu kromozomlardan oluşan rassal bir çözüm kümesi ile aramaya başlanır. Genetik algoritma, çözüm aramaya tek bir noktadan değil popülasyon olarak tanımlanan bir çözüm kümesi ile aramaya başlar. Popülasyonu oluşturan kromozomlar, problemin çözüm uzayındaki farklı noktaları temsil ettiği için çözüm uzayının farklı noktaları eş zamanlı olarak taranmaktadır.

Başlangıç popülasyonu belirlenirken dikkat edilmesi gereken noktalar başlangıç popülasyonunun nasıl belirleneceği ve popülasyon büyüklüğünün ne olacağıdır. Başlangıç popülasyonu çoğunlukla rastgele olarak belirlenir. Popülasyondaki kromozomların başka sezgisel tekniklerle belirlendiği yaklaşımlarda mevcuttur (Sivanandam ve Deepa, 2008). Başlangıç popülasyonunun sezgisel teknikler kullanılarak oluşturulması optimum çözüme ulaşmayı hızlandırabilecekken, bazı durumlarda

çeşitliliği azaltarak yerel optimuma takılmaya neden olabilmektedir. Başlangıç popülasyonunun rastgele oluşturulması ise çözüme daha geç ulaşılmaya sebep olabileceken, çeşitliliği arttırdığı için çözüm uzayının farklı noktalarının taranmasına olanak sağlayabilir ve global optimuma ulaşma konusunda daha başarılıdır (Gomez ve Hougen, 2007).

Popülasyondaki kromozom sayısı programcı tarafından belirlenir ve sonraki nesiller boyunca sabit tutulur. Popülasyon büyüklüğü genetik algoritmanın performansı ile doğru orantılıdır. Popülasyon boyutunun büyük olması, çözüm uzayının daha geniş bir bölümünde arama yapılmasını sağlarken, çözüme ulaşma süresini uzatacaktır. Popülasyon büyüklüğünün küçük olması ise algoritmanın çalışma hızını arttırırken, çözüm uzayının görece daha az bir kısmını tarayacağı için en iyi çözüme ulaşılma olasılığını azaltacaktır.

2.3.3. Uygunluk fonksiyonunun belirlenmesi ve uygunluk değerinin hesaplanması

Başlangıç popülasyonu oluşturulduktan sonraki adım, popülasyondaki her kromozomun uygunluk değerinin hesaplanmasıdır. Uygunluk değeri, uygunluk fonksiyonu ile belirlenir.

Genetik algoritmanın performansını etkileyen önemli unsurlardan biri de çözümlerin değerlendirilmesidir. Bu değerlendirmenin yapılabilmesi için uygunluk fonksiyonuna dikkat edilmelidir. Uygunluk fonksiyonu genetik algoritmanın sağlıklı çalışması ve doğru sonuçlar vermesini etkileyecektir. Genellikle uygunluk fonksiyonu problemin amaç fonksiyonu ya da onun bir dönüşümüdür (Acar,2014).

Belirli bir kromozomun uygunluk fonksiyonu, o kromozomun temsil ettiği çözümün kullanımıyla veya yeteneğiyle orantılı olan sayısal bir uygunluk değeri verir (Acar, 2014). Bir maksimizasyon problemi için yüksek uygunluk değeri o kromozomun ne kadar iyi olduğunu gösterir. Eğer minimizasyon problemi ise düşük uygunluk değeri o kromozomun ne kadar iyi olduğunu gösterir. Genetik algoritma her adımda daha iyi çözüm değerlerini bulmayı umarak çalışmaktadır. Bu sebeple, her nesilde daha uygun çözümlerin seçilmesi ve sonraki nesillere aktarılması için uygunluk değerinden yararlanılır.

2.3.4. Seçim operatörü

Kromozomların çoğalma için seçildiği bir işlemdir. Doğadaki doğal seçim işlemini genetik algoritma sürecine uygulayan operatördür. Uygunluk değerlerine göre değerlendirilerek popülasyonun en uygun kromozomlarının seçilmesi sağlanır. Sonraki oluşturulacak popülasyon için uygunluk değerine bakılarak değeri kötü olan kromozomların bir sonraki nesile aktarılması engellenmiş olur. Böylece, uygunluk değeri iyi kromozomların daha fazla kopya edilmesi mümkün olur. Seçim işlemiyle ortalama uygunluk değerinin iyiye gitmesi amaçlanmaktadır. Bu sayede istenen sonuca ulaşmak için daha uygun popülasyonlar oluşturulması sağlanacaktır.

Seçim işleminde popülasyondaki bütün kromozomların seçilme olasılığı vardır. Uygunluk değeri kötü olan kromozomların da görece daha az seçilme şansı ile üremelerine izin verilmesini sağlamak gerekir. Bu da problemin çözüm uzayında yerel optimum noktaları aşarak farklı bölgelerin taranması ve optimum çözüme ulaşılma şansının artırılması için gereklidir (Günay,2013).

Seçim operatörü, popülasyonun boyutuyla ilişkilidir. Topluluk boyutu küçük bir popülasyonla çalışılması durumunda topluluk çeşitlendirilmesinin olası iyi alternatiflerin oluşması için yetersiz kalması sorunu yaratabilir. Bu nedenle seçimde, popülasyondaki kromozomların çeşitlendirilmesini daraltan bir tekniğin uygulanması iyi sonuç vermeyebilir (Er, Çetin ve İpekçi Çetin, 2005).

Seçim sırasında dengenin gözetilmesi gerekmektedir. Çok güçlü bir seçim, o seviyede iyi uygunluk değerine sahip kromozomların popülasyonu kaplamasını ve çözüme ulaşmak için gereken çeşitliliğin azalmasını beraberinde getirir. Çok zayıf bir seçim ise aşırı yavaş bir evrime sebep olur (Cevre, 2008). Hangi kromozomların bir sonraki nesile aktarılacağını belirlemek için farklı seçim teknikleri geliştirilmiştir. Seçim tekniklerinden rulet tekerleği tekniği, turnuva seçim tekniği ve elit seçim tekniği en çok kullanılanlardır. Aşağıda bu tekniklerin detayları açıklanmıştır.

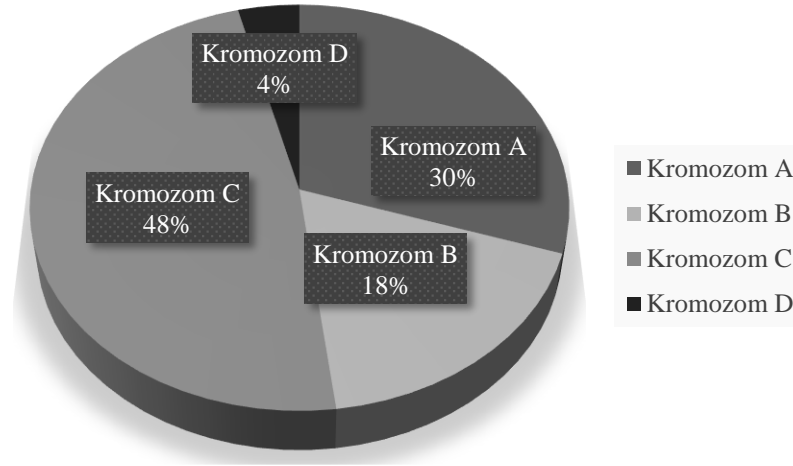
Rulet tekerleği tekniği: Bu seçim tekniği ilk kez Holland tarafından ortaya atılan ve sıklıkla kullanılan seçim tekniklerinden biridir (Sakawa, 2001). Rulet tekerleği seçiminde kromozomlar uygunluk değerlerine göre bir rulet tekerleğinde belli bir yüzde ile alan kaplarlar. Dolayısıyla tekerleğin döndürülmesiyle bir kromozomun sonraki nesillere aktarılma olasılığı tekerlek üzerinde kapladığı oran ile doğru orantılı olarak değişir. Bu teknik kötü uygunluk değerine sahip kromozomlara seçilme şansı tanırken, ortalama

uygunluk değerinden çok daha iyi uygunluk değerine sahip bir kromozomun tüm popülasyon üzerinde egemen olmasına sebep olur (Er, 2013).

Bu seçimde tekerlek n adet parçaya bölünür. Her aralık bir kromozomu temsil eder. Her kromozomun uygunluk değeri toplam uygunluk değerine bölünür. 1,2,3,...,n kromozomlarının uygunluk değerleri: $f_1, f_2, f_3, \dots, f_n$ ise,

$$i. \text{ kromozom için seçilme olasılığı } P_i: P_i = \frac{f_i}{\sum_{j=1}^n f_j} \text{ olacaktır.} \quad (2.1)$$

Böylece popülasyon içindeki her kromozomun çözüm kümesi içinde [0-1] arasındaki yeri bulunur. Kromozomlar yüzdeler olarak tekerlekte yer temsil ederler (Pulat ve Deveci, 2016). Şekil 2.7.'de dört kromozomlu bir popülasyondaki kromozomların rulet tekerleği üzerinde dağılımına bir örnek gösterilmiştir.



Şekil 2.7. Rulet tekerleği seçimi

Sonrasında seçim işlemini gerçekleştirmek üzere, genetik algoritma içerisinde bu teknik şu şekilde uygulanmaktadır (Kaya,2012).

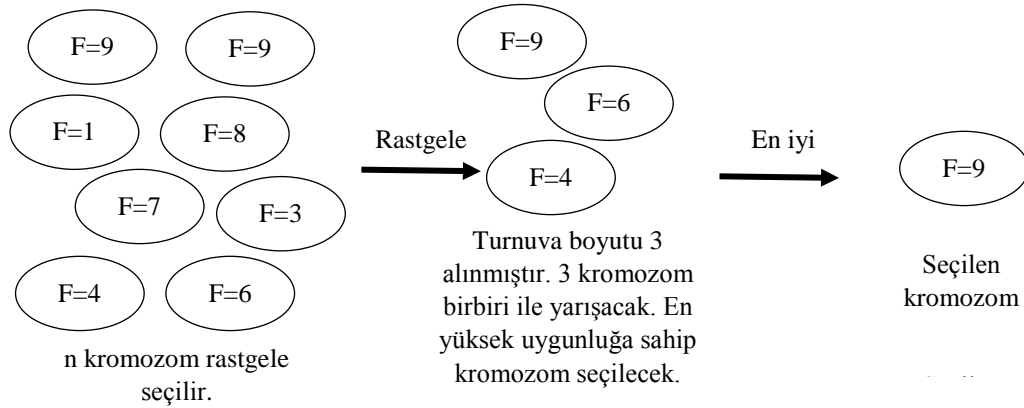
- 1) Popülasyondaki kromozomların toplam uygunluk değerleri (f_i) hesaplanır.
- 2) Her kromozomun uygunluk değeri, kendisi de dahil olmak üzere tüm kromozomların uygunluk değerleri toplamına bölünerek 0 ile 1 aralığında seçilme olasılıkları (formül 2.1'de gösterilmiştir.) belirlenir.
- 3) Kromozomların sahip oldukları seçilme olasılıkları birikimli olarak toplanır.

- 4) Seçim işlemi gerçekleştirilirken seçilecek kromozomu belirlemek için 0 ile 1 arasında rastgele bir sayı üretilir. Üretilen sayı hangi alana denk geliyorsa o alanla temsil edilen kromozom seçilir.

Rulet çemberi, popülasyon büyüklüğü kadar çevrilir. Her dönüşte çemberin durduğu dilime sahip olan kromozom bir sonraki nesli oluşturacak kromozomların bulunduğu ebeveyn havuzuna seçilmiş olur.

Turnuva seçim tekniği: Bu seçim tekniği, etkinliği ve basit uygulaması nedeniyle en popüler seçim tekniğidir. Turnuva seçiminde n kromozom popülasyondan rastgele seçilir ve seçilen kromozomlar birbiriyle yarışır. En iyi uygunluk değerine sahip kromozom galip gelir ve seçilir. Her turnuvada yarışan kromozom sayısı, turnuva büyüklüğü olarak adlandırılır. Turnuva büyüklüğü genellikle 2 alınır ve ikili turnuva olarak adlandırılır (Razali and Geraghty, 2011). Yeterli sayıda kromozom seçimi gerçekleşene kadar turnuva devam eder (Er, 2013).

Şekil 2.8.'de turnuva büyüklüğü 3 olarak belirlenen bir örnek ile turnuva seçim tekniği mekanizması gösterilmiştir.



Şekil 2.8. Turnuva seçimi (Razali and Geraghty, 2011)

Elit seçim tekniği (Elitizm): Kromozomlar üzerinde çaprazlama ve mutasyon operatörleri uygulanırken, popülasyonun sahip olduğu en iyi kromozom (çözüm) kaybolabilir. Bunu önlemek için popülasyondaki en iyi kromozom ya da belirlenen sayıdaki en iyi kromozomların doğrudan bir sonraki nesile aktarılması işlemine elit seçim tekniği adı verilir. Elit seçim tekniği genetik algoritmanın performansını artırıcı bir etkiye sahiptir. Böylece bir sonraki nesildeki en iyi kromozomun bir önceki nesildeki en

iyi kromozomdan kötü olma ihtimali kalmaz. Ancak yeni nesile aktarılacak kromozomların sayısı iyi ayarlanmalıdır. Yeni nesile aktarılacak en iyi kromozomların sayısı çok fazla olduğunda algoritma hep aynı çözümler etrafında dolaşacak ve global optimuma erişme olasılığı azalacaktır (Er, 2013).

2.3.5. Çaprazlama operatörü

Genetik algoritmanın performansını etkileyen önemli işlemlerden biridir. Seçilen iki veya daha fazla kromozomun genlerinin kullanılarak yeni kromozomların elde edilmesi işlemini gerçekleştirir. Çaprazlama sonrasında elde edilen yeni kromozomlar, ebeveynlerin genetik özelliklerini taşımaktadır. Başka bir ifadeyle, çaprazlama, ebeveynlerden bazı genleri alarak yeni kromozomlar oluşturduğundan, oluşan kromozomlar iki ebeveynin de bazı temel özelliklerine sahip olmaktadır (Yücel, 2016). Farklı çözümler arasında bilgi alış-verişi sağlar (Michalewicz, 1992).

Çaprazlama üç adımda gerçekleştirilen bir operatördür (Sivanandam and Deepa, 2008):

- i. Seçim operatörü rastgele eşleşme için iki ayrı diziden bir çift seçer.
- ii. Çaprazlama konumu dizi uzunluğu boyunca rastgele seçilir.
- iii. Son olarak, konum değerleri çaprazlama takiben iki dizi arasında değiştirilir.

Seçim işlemi ile belirlenen kromozomların yer aldığı bir çiftleşme havuzu oluşturulduktan sonra gerçekleştirilen çaprazlama işlemlerinin genellikle tümünde, iki ebeveyn, bir çiftleşme havuzundan alınarak karşılıklı genlerini değiştirirler. Bu sayede çaprazlama operatörünün çalıştırılmasıyla popülasyonda yer almayan yeni kromozomlar oluşturularak çeşitlilik sağlanır (Yücel, 2016).

Aynı kromozom çiftlerinden farklı çocuk kromozom meydana gelebilmesinden dolayı çaprazlama operatörü genellikle stokastiktir (Dreo vd., 2006).

Bir önceki nesilden daha iyi nitelikler içeren yeni kromozomlar yaratmak amacıyla çaprazlama operatörleri kullanılmaktadır (Pulat ve Deveci, 2017). Ancak iyi genetik özellikler taşıyan kromozomlardan yine iyi çocuklar oluşabileceği gibi daha kötü uygunluk değerine sahip çocukların meydana gelmesi de olasıdır. Uygunluğu ebeveynlerden daha kötü kromozomların elde edilmesine izin verilmesi, arama uzayının farklı ve keşfedilmemiş bölgelerine ulaşarak algoritmanın yerel optimum noktalarda tıkanmasına engel olacaktır (Günay,2013).

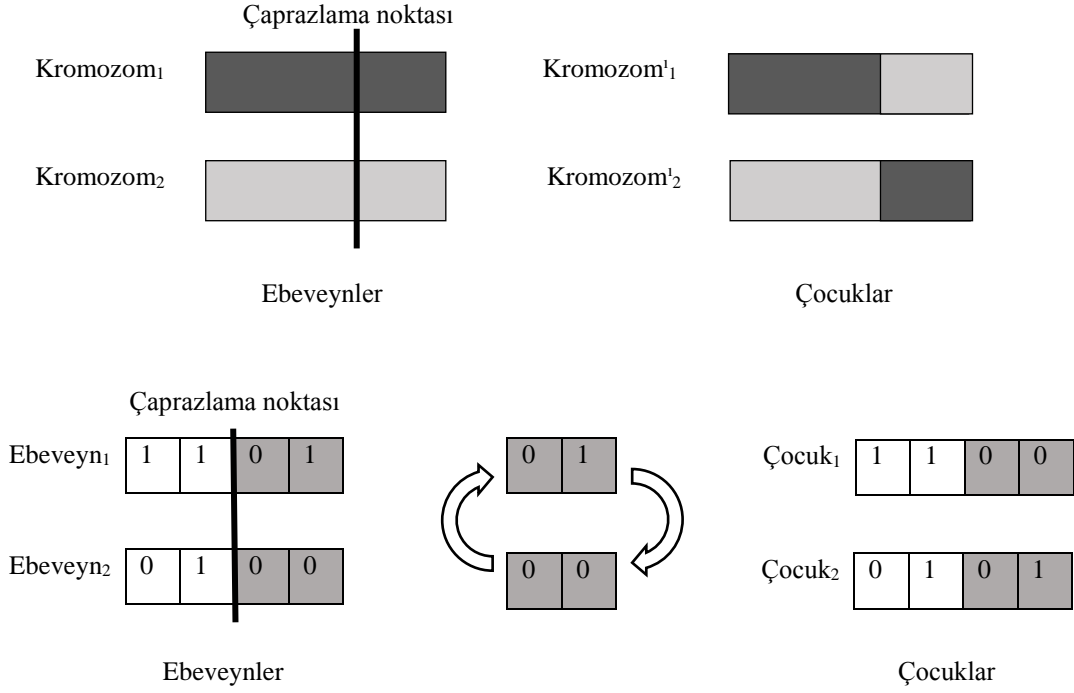
Çaprazlama operatöründe temel parametre çaprazlama olasılığı (P_c) dır. P_c : Çaprazlamanın ne sıklıkla yapılacağını tanımlayan bir parametredir (Javidi, Fard and Jampour, 2015). Her yinelemede P_c (Çaprazlama olasılığı) x N (Kromozom sayısı) kadar kromozoma çaprazlama uygulanır (Grefenstette, 1986). Çaprazlama genellikle çiftleşme için seçilen bütün kromozomlara uygulanmaz. Çaprazlama olasılığının genel olarak 0,6 ile 1,0 arasında olduğu rastgele bir seçim yapılır. Eğer çaprazlama yapılmazsa, çocuklar ebeveynlerin kopyası olarak oluşturulur. Çaprazlamanın belirli bir oranda yapılması, her kromozoma çaprazlama ile bozulmadan genlerini direkt olarak sonraki nesile taşıma şansı verir (Beasley vd.,1993). Çaprazlama olasılığı çözümün kalitesi üzerinde önemli bir etkiye sahiptir ve problemin yapısına göre karar verici tarafından belirlenir. Bu nedenle çaprazlama olasılık değerini seçerken dikkatli olunmalıdır.

Belirlenen çaprazlama olasılığına göre yapılacak çaprazlama işleminde diğer bir önemli nokta ise, hangi çaprazlama tekniğinin kullanılacağıdır. Her problem için farklı bir çaprazlama tekniği kullanılması gerektiğinden çok çeşitli çaprazlama teknikleri geliştirilmiştir. Aşağıda bu tekniklerden bazılarının detaylarına yer verilecektir.

Tek noktalı çaprazlama tekniği: En temel çaprazlama tekniğidir. Rastgele bir çaprazlama noktası seçilir ve iki ebeveynin o noktadan sonra gelen kısımları karşılıklı değiş tokuş edilir.

Tek noktalı çaprazlama özellikle ikili kodlama ile oluşturulmuş kromozom dizilimlerinde kullanılmaktadır, farklı şekillerde kodlanmış problemlerde farklı çaprazlama operatörlerinin kullanılması yararlı ve çoğunlukla gerekli olacaktır (Günay, 2013).

Şekil 2.9.'da görüldüğü gibi tek noktalı çaprazlama için öncelikle ikişerli olarak eşleştirilen kromozomlar üzerinde çaprazlanması için rastgele olarak bir çaprazlama noktası belirlenir. Bu aşamadan sonra eşleşen kromozomların çaprazlama noktasından sonraki ikinci kısımları yer değiştirerek yeni çözüm kromozomları elde edilir.

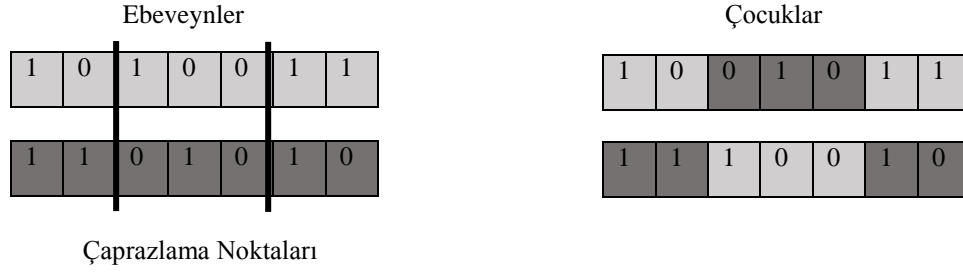


Şekil 2.9. Tek noktalı çaprazlama

Çift noktalı çaprazlama tekniği: Tek noktalı çaprazlama tekniğine benzer şekilde çaprazlama noktalarının kromozom boyunca iki yerde yapılması ile aynen tekrarlanır (Şen, 2004). Rastgele iki nokta belirlenir ve bu noktalar arasında kalan bölümlerin karşılıklı olarak yer değiştirilmesiyle yeni kromozomlar elde edilir.

Tek noktalı çaprazlama yapılabilmesi için her iki kromozomun da aynı gen uzunluğunda olması gerekmektedir. Çift noktalı çaprazlamada ise kromozom iki noktadan kesilir ve karşılıklı olarak pozisyonlar yer değiştirilmektedir (Fırlalı ve Engin, 2002).

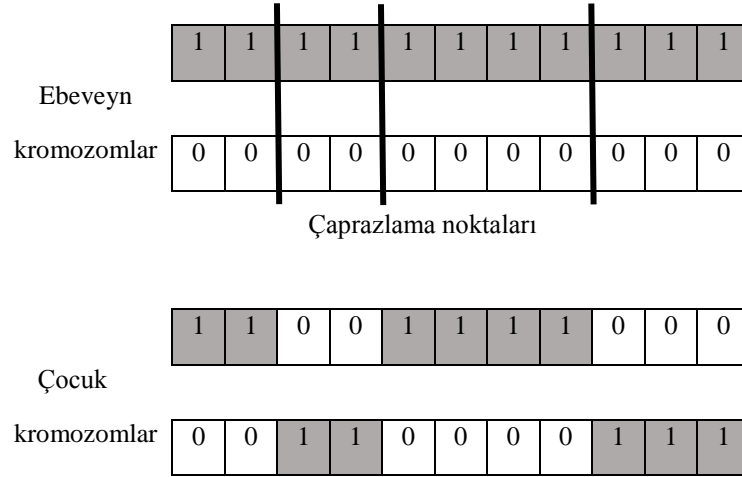
Şekil 2.10.'da görüldüğü gibi çift noktalı çaprazlama için öncelikle eşleştirilen kromozomlar üzerinde çaprazlanması için rastgele olarak iki çaprazlama noktası belirlenir. Sonrasında, bu noktalar arasında kalan genler karşılıklı olarak yer değiştirilir ve yeni çözüm kromozomları elde edilir.



Şekil 2.10. İki noktalı çaprazlama

Çok noktalı çaprazlama: Çift noktalı çaprazlama tekniğinin gelişmiş halidir. Kromozomlar ikiden daha fazla parçalara ayrılır ve bir atlanarak elde edilen çiftler arasında değiştirilerek yeni kromozomlar elde edilir (Bolat, Erol, İmrak, 2002).

Şekil 2.11.'de çok noktalı çaprazlama işleminin ikili kodlamalı kromozomlar üzerinde uygulamasının örneği verilmiştir.



Şekil 2.11. Çok noktalı çaprazlama (Yücel,2016)

Düzenli çaprazlama tekniği: Rastgele olarak genlerin değişimine dayalı bir tekniktir. Bu teknikde, kromozom büyüklüğü ile aynı sayıda geni içeren ve 'çaprazlama maskesi' olarak adlandırılan rastgele bir kromozom oluşturulur. Birinci çocuk oluşturulurken çaprazlama maskesinde 1 değeri, aynı konumda bulunan genin ilk ebeveynden kopyalanacağını, 0 değeri ise aynı konumda bulunan o genin ikinci ebeveynden kopyalanacağını ifade eder (Reeves, 2006). İkinci çocuk oluşturulurken de maskenin tersi kullanılmaktadır. Şekil 2.12.'de düzenli çaprazlama tekniğine örnek verilmiştir.

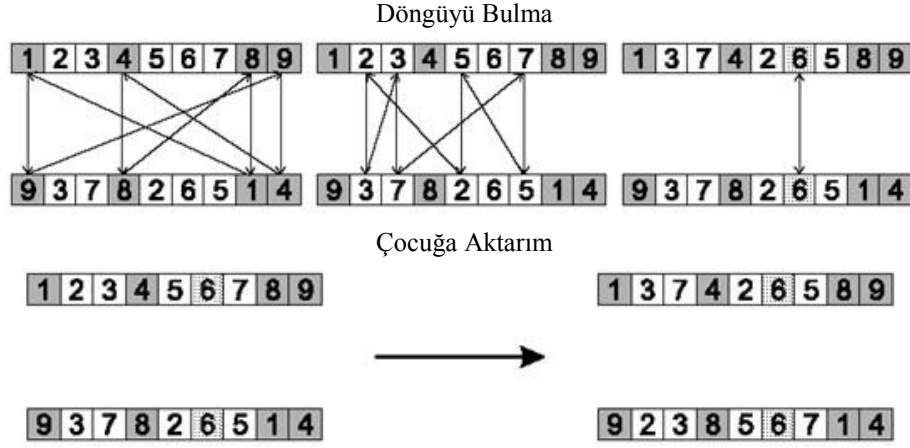
Ebeveyn	0	0	1	0	0	1	1	0	1	0
kromozomlar	1	1	1	0	0	1	0	0	0	1
Çaprazlama maskesi	1	0	0	1	0	1	1	1	0	0
1.Çocuk	0	1	1	0	0	1	1	0	0	1

Şekil 2.12. Düzenli Çaprazlama (Ağalday, 2018)

Dairesel çaprazlama (CX-Cycle Crossover): Oliver ve arkadaşları (1987) tarafından, permutasyon çaprazlama operatörleri kullanılan gezgin satıcı problemleri için önerilmiştir. Bu çaprazlama tekniğinde öncelikle birinci ebeveynin ilk geni alınarak birinci çocuğun ilk geni oluşturulur. Bu gen değerinin ikinci ebeveyndeki pozisyonu belirlenir, birinci ebeveynin aynı pozisyonunda olan gen değeri alınır. Alınan bu genin birinci ebeveyndeki bulunduğu pozisyona karşılık gelen gen değeri yine ikinci kromozomdan alınır. Bu şekilde ilk alınan gen değeriyle yeniden karşılaşmaya kadar işleme devam edilir. Bu bir döngü oluşturmaktadır (Günay,2013). Tekniğin adımları şu şekildedir (Cheng, Gen and Tusujiura, 1999):

- i. Ebeveynlerin genleri arasındaki ilişki bulunarak, döngü kurulmaktadır.
- ii. Döngü içindeki genler birinci ebeveynden alınarak çocuktaki aynı noktaya aktarılmaktadır.
- iii. İkinci ebeveyn den döngüde halen döngüde bulunan ilgili genler silinmektedir.
- iv. Kalan genler çocuğa yerleştirilmektedir.

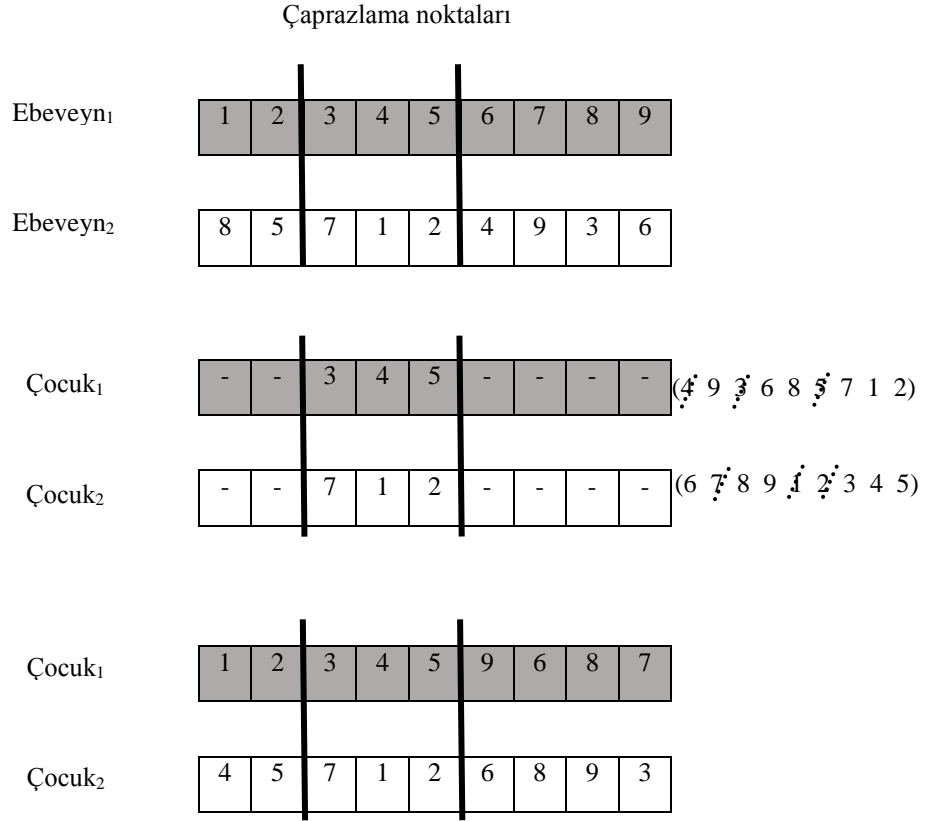
Şekil 2.13.'te dairesel çaprazlamaya örnek verilmiştir.



Şekil 2.13. Dairesel çaprazlama (Yücel, 2016)

Sıralı çaprazlama (OX- Order Crossover): Davis (1985) tarafından, gezgin satıcı problemleri için önerilmiş, daha sonra da Oliver ve arkadaşları tarafından geliştirilmiştir. Bu teknikte yapılmak istenen, eşleştirilen iki ebeveynden bir tanesinde izlenen sırayı koruyup diğer ebeveynden bir alt dizi seçerek iki çocuk oluşturmaktır (Taşkın ve Emel, 2009).

Şekil 2.14.'te sıralı çaprazlamanın nasıl uygulandığı gösterilmiştir. Öncelikle ebeveynler üzerinde alt tur oluşturacak şekilde kesim noktaları rastgele belirlenir. Birinci ebeveynin kesim noktaları arasında kalan genler ilk çocuğa, ikinci ebeveyndeki genler ise oluşacak diğer çocuğa aynen kopyalanır. Birinci çocuk oluşturulurken, ikinci ebeveynde bulunan genler ikinci kesim noktasının sağından başlayarak zaten mevcut olan genleri atlayarak aynı sırada yazılır. Dizinin sonuna ulaşıldığında ilk pozisyondan devam edilir. Bu işlemin tersi ikinci çocuk içinde uygulanır (Deep and Mebrahtu, 2011).



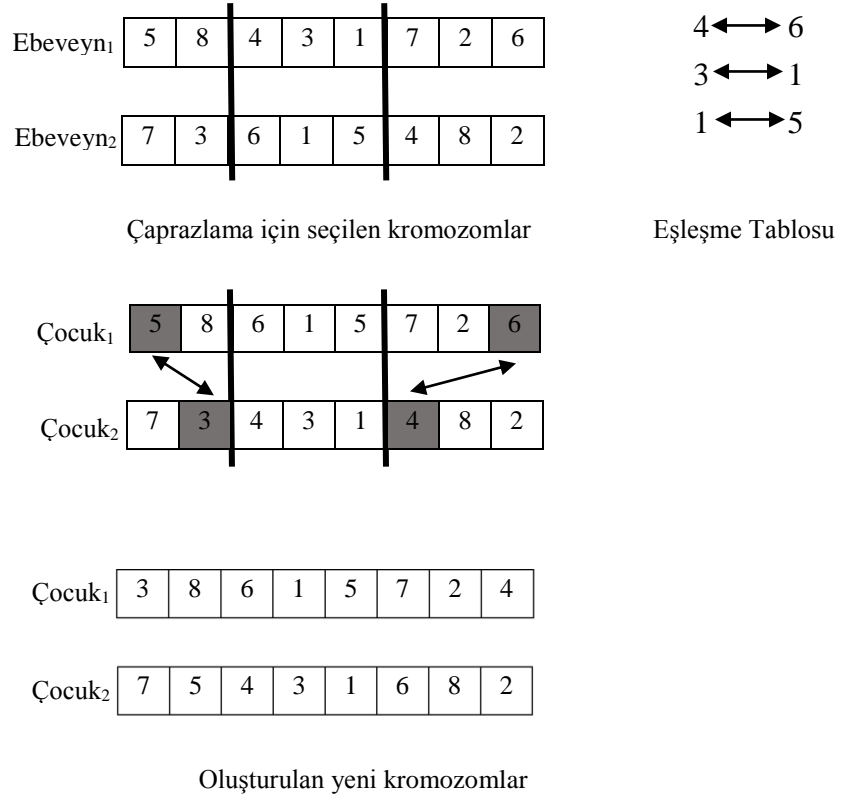
Şekil 2.14. Sıralı Çaprazlama (Deep and Mebrahtu, 2011)

Kısmi eşleştirmeli çaprazlama (PMX- Partially Matched Crossover): Golberg ve Lingle (1985) tarafından önerilen bu teknik gezgin satıcı problemleri ve araç rotalama problemlerinde yaygın olarak kullanılmaktadır (Taşkın ve Emel, 2009). Bu tekniğin adımları şu şekildedir (Cheng, Gen and Tusujimura, 1999):

- i. Kromozomlar üzerinde rastgele olarak iki çaprazlama noktası belirlenir. Bu noktalar ile bölünen parçalara 'eşleştirme parçası' denmektedir.
- ii. Ortada kalan eşleştirme parçaları karşılıklı olarak yer değiştirilir.
- iii. Değiştirilmeyen parçalar arasındaki ilişki belirlenir.
- iv. Değiştirilmeyen parçalara da eşleştirme ilişkisi kapsamında değişimler uygulanır.

Öncelikle, ebeveynlerden rastgele iki kesim noktası seçilir. Bu noktalar arasında kalan alt diziler eşleşme parçasıdır. Ebeveynlerin eşleşme parçaları karşılıklı olarak eşleştirilir ve bu eşleşmeye uyan genler, karşılık gelen genler ile değiştirilir. Birinci ebeveynin eşleştirme parçası ikinci çocuğa, ikinci ebeveynin eşleştirme parçası birinci

çocuğa kopyalanır. Daha sonra kalan pozisyonlar için tekrar eden genler, eşleştirilen genlere göre değiştirilerek düzeltilir. Bu şekilde iki yeni çocuk kromozom oluşturulmuş olur. Şekil 2.15.'te kısmi eşleştirmeli çaprazlamanın nasıl uygulanacağını gösteren bir örnek verilmiştir.



Şekil 2.15. Kısmi eşleştirmeli çaprazlama

2.3.6. Mutasyon operatörü

Mutasyon, çaprazlama işlemi gerçekleştirilip yeni kromozom popülasyona katıldıktan sonra, oluşan yeni kromozomun uygunluk değerini arttırmak amacıyla genlerinde değişiklik yapılması işlemidir. Oluşan yeni popülasyondaki kromozomların öncekilere benzemesini önlemek ve daha çabuk mükemmel kromozoma ulaşabilmek amacıyla gerçekleştirilir (Kabadayı, 2007). Ancak bulunan yeni kromozomun daha uygun bir kromozom olması kesin bir durum değildir. Uygun olmayan bir çözümle karşılaşılabilir, diğer genetik operatörler sayesinde bir sonraki nesilde bu uygun olmayan çözümler elenebilecek veya değiştirilebilecektir (Tabak, 2008).

Mutasyon operatörü, popülasyonda genetik çeşitliliği sağlamak ve korumak amacıyla kullanılır (Çolak, 2010). Mutasyon, başka türlü bulunamayacak olası çözümlere ulaşmanın rasgele bir yoludur (Karaova, Smarkov and Penev, 2005).

Mutasyon işlemi, önceki işlemlerde kaybedilen yararlı genlerin kurtarılması için tasarlanmıştır. Mutasyon işleminde, mutasyon olasılığına bağlı olarak kromozomdaki değer değiştirilir (Chen and Chen, 1997). Mutasyon işlemi, tüm kromozomlara uygulanmayıp sadece üretilen çocukların belli bir yüzdesine uygulanır. Bu yüzdeye mutasyon olasılığı veya mutasyon oranı denir. Mutasyon olasılığı, problemin başında belirlenen ve genellikle (0,01 gibi) çok düşük bir oranla gerçekleştirilir. Eğer mutasyon olasılığı çok büyük olursa, bu kromozomların gereğinden fazla değişikliğe uğraması anlamına gelir ve çözümün optimumuna ulaşması zorlaşabilir (Kırda, 2013).

Problemin yapısına bağlı olarak aşağıdaki mutasyon operatörlerinden biri seçilebilir (Bayata,2012):

- Ters çevirme
- Yer değişikliği
- Ekleme
- Karşılıklı değişim

Şekil 2.16’da bu operatörlerin nasıl uygulandığı gösterilmiştir.

Mutasyon öncesi	1	2	3	4	5	6	7	8
Mutasyon sonrası	1	2	5	4	3	6	7	8

Ters çevirme işlemi

Mutasyon öncesi	1	2	3	4	5	6	7	8
Mutasyon sonrası	1	2	6	7	3	4	5	8

Yer değiştirme işlemi

Mutasyon öncesi	1	2	3	4	5	6	7	8
Mutasyon sonrası	1	2	7	3	4	5	6	8

Ekleme işlemi

Mutasyon öncesi	1	2	3	4	5	6	7	8
Mutasyon sonrası	1	6	5	4	3	2	7	8

Karşılıklı değişim işlemi

Şekil 2.16. Mutasyon İşlemi

2.3.7. Sonlandırma kriteri ve algoritmanın sona ermesi

Genetik algoritma, doğadakinine benzer şekilde hiç bitmeyen bir süreçtir. Bununla birlikte, algoritmanın bir yerde sona ermesi ve sonucunun elde edilmesi için karar verici tarafından belirlenen bir kriterin sağlanmış olması gereklidir (Koza, 1998). Genellikle yineleme sayısının artmasıyla birlikte çözümün kalitesi de artacaktır, fakat gerçek çözüme ulaşabilmek için yineleme sayısının ne olması gerektiğiyle ilgili kesin bir kural bulunmamaktadır. Burada kullanıcı tercihleri devreye girer ve çoğunlukla zaman kısıtı ön plandadır (Ünal, 2018). Sonlandırma kriterini sağladığında, programın çalışması durdurulur ve bulunan en iyi kromozom, algoritma sonucunda problemin optimum çözümü olacaktır.

Çeşitli sonlandırma koşulları aşağıdaki gibi listelenmiştir (Sivanandam ve Deepa, 2008):

- Maksimum yineleme sayısı: Karar verici tarafından belirlenen bir yineleme sayısına ulaştığında algoritma sona erer.
- Geçen süre: Süreç, önceden tanımlanan bir süreyi doldurduğunda sona erer.
- Uygunluk değerinin değişmemesi: Önceden belirlenmiş bir uygunluk kriteri aşıldığında algoritma sona erer.
- Yakınsama: Belirli bir yineleme sayısı sonrasında bulunan en iyi uygunluk değerinin değişmemesi durumunda algoritma sonlandırılabilir.

Genetik algoritma sürecinde en yaygın olarak kullanılan sonlandırma kriteri, belirli bir yineleme sayısına ulaşılmasıdır. İterasyon sayısı, problemin yapısı ve arama uzayının büyüklüğüne dikkate alınarak karar verici tarafından belirlenmelidir (Yücel, 2016).

2.4. Genetik Algoritmanın Özellikleri

Genetik algoritma, pek çok problem türü için uygun parametrelerle çalıştırıldığında optimuma yakın çözümler vermektedir. Optimizasyondaki temel amaç optimal bir noktaya ulaşabilmek, daha doğrusu mümkün olduğunca yaklaşabilmektir. Bunu sağlamak için birçok klasik teknik vardır. Bu tekniklerin başarısı optimal noktaya ulaşp ulaşamadıkları veya ne kadar ulaşabildikleriyle ölçülür. Genetik algoritma, klasik optimizasyon algoritmalarından aşağıda belirtilen özellikleriyle dört temel noktada ayrılır (İşçi ve Korukoğlu, 2003):

- Genetik algoritma, parametrelerin kendileriyle değil, parametrelerin kodlanmış şekilleriyle uğraşır.
- Genetik algoritma aramaya tek bir noktada değil, noktalar kümesinden başlar. Dolayısıyla yerel optimuma takılmadan çalışabilirler.
- Genetik algoritma amaç fonksiyonunun türevlerini ve bir takım yardımcı bilgileri değil, doğrudan amaç fonksiyonunun kendisini kullanır.
- Genetik algoritmada deterministik değil rassal geçiş kuralları kullanılır.

2.4.1.Genetik algoritmanın avantajları

Genetik algoritmanın kullanılmasının başlıca avantajları şu şekilde belirtilmektedir (Haupt and Haupt, 2004):

- Sürekli ya da kesikli değişkenlerden oluşan problemlerde optimum çözümler verebilmektedir.
- Ek bilgilere ihtiyaç duymadan çalışabilmektedir.
- Çok sayıda değişkenle çalışabilmektedir.
- Geniş çözüm uzayında eş zamanlı tarama yapabilmektedir.
- Değişkenler üzerinde kodlama yapılmasına izin vermektedir.

2.4.2.Genetik algoritmanın dezavantajları

Genetik algoritma, sağladığı birçok avantajının yanında çeşitli dezavantajlara da sahiptir. Bu dezavantajlar, problemin çözümünü zorlaştırabileceği gibi doğru kullanıldığında problemin çözümünü kolaylaştırabilmektedir. Genetik algoritmanın dezavantajları aşağıdaki gibi sıralanabilmektedir (Sivanandam ve Deepa, 2008):

- Uygunluk fonksiyonunun belirlenmesi zorluk çıkarabilmektedir.
- Erken yakınsama oluşabilir.
- Popülasyon büyüklüğü, mutasyon olasılığı, çaprazlama olasılığı, seçim tekniği gibi çeşitli parametrelerin seçilmesi problemi vardır.
- Özel bilgiler probleme kolayca dahil edilemez.
- Yerel optimumun belirlenmesinde iyi değildir.
- Kesin bir sonlandırma kriteri bulunmamaktadır.
- Düzgün tek modlu fonksiyonlar için etkili değildir.

- Bir yerel arama tekniđi ile birlikte kullanılması gerekir.
- Tam olarak global optimumu bulmak zordur.
- Çok sayıda uygunluk fonksiyonu deđerlendirmesi gerektirir.
- Yapılandırma basit deđildir.

2.4.3.Genetik algoritmanın kullanım nedenleri

Genetik algoritma, karmaşık problemlere en uygun veya en yakın çözümlü bulmak için etkili ve sağlam optimizasyon tekniklerinden biridir. Genel olarak genetik algoritma:

- Oldukça geniş bir çözümlü uzayına sahip olan,
- Birden çok optimumun varlığı söz konusu olan,
- Amaç fonksiyonunun türevi alınamayan,
- Amaç fonksiyonu süreksiz durumlar içeren,
- Verilerin doğrusal olmadığı,
- Verilerde anlamlı hataların olduğu,
- Verilerin durađan olmadığı problemlerde kullanılabilir (Pereira, 2000).

Çeşitli yapılarıdaki birçok problemin çözümlüde genetik algoritma kullanılabilir. Problemlerde genetik algoritma kullanmanın en temel avantajları şu şekilde açıklanabilir (Chen and Chen, 1997):

- En iyi sonuca erişildikten sonra genelde hesaplaması zor olan ve zaman alan duyarlılık analizlerine gerek duyulmaması,
- Farklı yapılarıda deđerkenlere sahip problemlerin çözümlüde kolaylık sağlanması,
- İterasyonlar sonucunda farklılaşan kromozomlar sayesinde daha geniş arama uzayına yayılarak, global optimumu bulma olasılığının yüksek olması.

2.5.Genetik Algoritmanın Uygulama Alanları

Genetik algoritmanın en uygun olduğu problemler, klasik tekniklerle çözümlü mümkün olmayan ya da çözümlü süresi problemin büyüklüğüne göre üstel olarak artan durumlarda kullanılmaktadır.

Genel uygulama alanları ařağıdaki gibi verilebilir (Emel ve Tařkın, 2002):

- Optimizasyon
- Otomatik programlama ve bilgi sistemleri
- Mekanik öğrenme
- Ekonomik ve sosyal sistem modelleri
- İřletmelerdeki uygulama alanları

Optimizasyon: Çözüm uzayında bir araştırma tekniğı olan genetik algoritma optimizasyon problemlerinin çözümünde kullanılmaktadır. Genetik algoritmanın uygulandığı optimizasyon problemleri, fonksiyon optimizasyon problemleri ve kesikli optimizasyon problemleri altında toplanabilir.

Otomatik programlama ve bilgi sistemleri: Genetik algoritmanın yaygın olarak kullanıldığı alanlardan biridir. Belirli ve özel görevler için gerekli olan bilgisayar programlarını geliřtirmedir. Ayrıca, diđer hesaplama gerektiren yapıların tasarımı için de kullanılmaktadır. Bunlara örnek olarak, bilgisayar çipleri tasarımı, ders programlarının hazırlanması ve ağıların çizelgelenmesi verilebilir.

Mekanik öğrenme: Gözlenmiş bir veri takımını anlamak ve yorumlamak, görülmemiş objelerin özelliklerini tahmin etmek gibi amaçlar için model kurmaya çalışan mekanik öğrenme yapılarında da genetik algoritma kullanılmaktadır.

Ekonomik ve sosyal sistem modelleri: Ekonomide, bir sistemi ölçerek deneylere dayalı olarak belirlenmiş değıřkenlerin birbirleriyle olan matematiksel ilişkilerini keřfetmeye çalışan problemler için genetik algoritma kullanılabilir.

Genetik algoritma yenilik oluřturma süreçlerini, fiyat verme stratejilerinin gelişim süreçlerini ve kazanç getiren pazarların ortaya çıkıř süreçlerini modelleme alanlarında da kullanımları oldukça yaygındır. Genetik algoritma sosyal sistemlerin evrimsel yönlerini anlamak amacıyla kullanılmaktadır. Bunlara örnek olarak iş birliğıinin evrimi, iletişimin evrimi ve karıncalardaki iz takibi davranıřının evrimi verilmektedir.

İřletmelerdeki uygulama alanları: Genetik algoritma işletmelerde finans, pazarlama, üretim/iřlemler, montaj hattı dengeleme problemi, çizelgeleme problemi, tesis yerleşim problemi, atama problemi, hüresel üretim problemi, sistem güvenilirliğı problemi, taşıma problemi, gezgin satıcı problemi, araç rotalama problemi, minimum yayılan ağaç problemi gibi birçok farklı problemde kullanılabilir.

Finansal modelleme problemlerinde genetik algoritma amaç fonksiyonu odaklı olup genellikle amaç fonksiyonlarını tahmin etmeye gücü veya doğru kıyaslamalara bağlı getirilerle ilgili olduğundan oldukça uygundur. Özellikle hisse senedi fiyatlarındaki değişimleri tahmin etmede, kaynak tahsisi ve uluslararası sermaye tahsisi stratejilerini belirlemede kullanılabilirler.

Pazarlamanın en önemli fonksiyonlarından biri, tüketicilere ait verileri analiz etmek, çeşitli tüketici profillerini çıkartmak ve bunlara bağlı olarak pazarlama stratejileri uygulamaktır. Bunları yapabilmeyen ilk adımı ise çok büyük veri tabanları oluşturmaya ve bu veri tabanlarını işletmenin amaçlarına yönelik etkin bir biçimde kullanmaya bağlıdır. Bu amaçlarla kullanılan veri madenciliği tekniklerinin uygulanabilmesi için genetik algoritma verimli şekilde çalışabilmektedir.

İşletmeler, nihai sonuçlar elde edebilmelerini sağlayan üretim ve bunun için gerçekleştirdikleri işlem süreçlerinde pek çok farklı problemle karşılaşmaktadırlar. Bu problemlerin çözümü için genetik algoritma oldukça etkin şekilde kullanılan yaklaşımlardan biridir. Bu problemlerden en sık görülenleri şunlardır:

- Montaj hattı dengeleme
- Çizelgeleme problemi
- Tesis yerleşim problemi
- Atama problemi
- Hücresele üretim problemi
- Sistem güvenilirliği problemi
- Taşıma problemi
- Gezgin satıcı problemi
- Araç rotalama problemi
- Minimum yayılan ağaç problemi

Bahsedilen problem türlerinin çözümünde başka arama teknikleri de kullanılabilir. Fakat genetik algoritma, genel çözümleri belirleyebilmesi ve alternatiflerin hızlı bir şekilde değerlendirmesi gibi geleneksel tekniklere göre sağladığı üstünlükler sebebiyle daha etkin çözüm tekniği olarak kabul görmektedir. Anlatıldığı gibi genetik algoritma pek çok alanda sıklıkla kullanılmaktadır.

ÜÇÜNCÜ BÖLÜM

3. GENETİK ALGORTİMA KULLANILARAK ESKİŞEHİR HALK EKMEK İŞLETMESİ İÇİN ARAÇ ROTALAMA PROBLEMİNİN ÇÖZÜMÜ

Araç rotalama probleminin, genetik algoritma tekniği ile çözümü bir uygulama ile gösterilmeye çalışılacaktır. Uygulama, Eskişehir’de bulunan ve günlük ekmek üretip dağıtan Eskişehir Halk Ekmek A.Ş.’de yapılmıştır. Uygulama amacı; müşteri taleplerinin tümü karşılanırken, en kısa rotayı oluşturabilmek için işletmeye ait hangi aracın hangi siparişleri nasıl bir sırayla taşınması gerektiği belirlenmeye çalışılmasıdır.

3.1. Eskişehir Halk Ekmek İşletmesi ve Problem Tanımı

Eskişehir Halk Ekmek İşletmesi, son teknolojiyle donatılmış modern halk ekmek üretim tesisleri, Eskişehir Büyükşehir Belediyesi’nin artan hayat pahalılığı karşısında eski küçük halk ekmek fırınının artan talebi karşılayamaması nedeniyle, 2002 yılında sosyal hizmet projelerinden biri olarak kurulmuştur. Ekmekler, tünel tipi fırınların yer aldığı tesislerde el değmeden üretilmekte ve poşetlenmektedir.

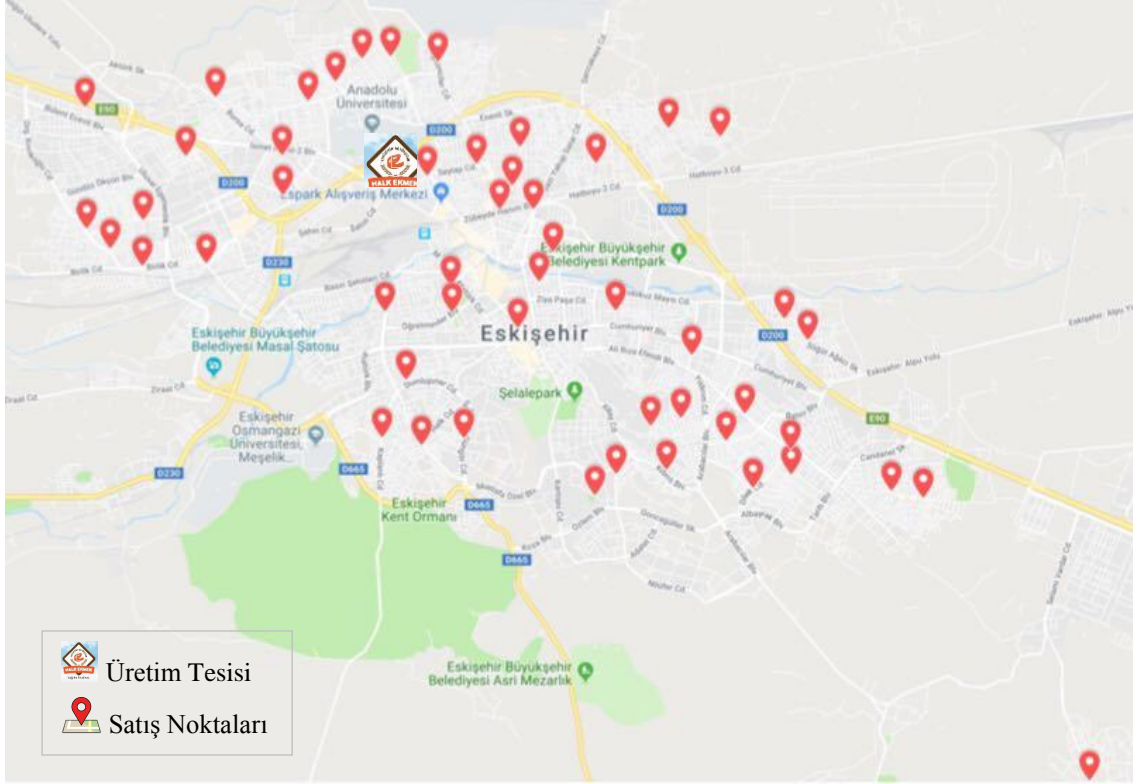
Eskişehir Halk Ekmek İşletmesi üretim tesisi 6750 adet/saat beyaz ekmek, 10000 adet/saat roll ekmek üretim kapasitesine sahiptir. Tesiste beyaz ekmek dışında ayrıca kepekli ekmek, tam buğday ekmek ve roll ekmek çeşitleri olmak üzere dört çeşit ekmek üretilmektedir. Ekmek çeşitlerine ait görseller Şekil 3.1.’de verilmektedir.



Şekil 3.1. Eskişehir Halk Ekmek İşletmesi'ne ait ürünler (<http://www.eskisehirhalkekmek.com/>, 28 Haziran 2019)

Ekmekler şu an mevcut olan tek bir üretim tesisinde üretilerek 51 adet satış büfesi aracılığıyla halka ulaştırılmaktadır. Ayrıca satış büfeleri haricinde, Eskişehir'de bulunan okul, otel, fabrika gibi birçok kuruma da hizmet verilmektedir. Bu çalışma kapsamında, tesisin satış büfelerine ekmek dağıtımını ele alınmaktadır. Üretim tesisinin ve dağıtım noktalarının adresleri bilindiği için Eskişehir Halk Ekmek İşletmesi'nin mevcut üretim tesisi ve satış büfelerinin, Google Maps (<https://www.google.com/maps>) üzerindeki

konumları Şekil 3.2.'de görsel olarak gösterilmektedir. Bu şekilde ilgili 51 noktanın yerlerinin toplu olarak görülmesi sağlanmıştır.



Şekil 3.2. Eskişehir Halk Ekmek İşletmesi üretim tesisi ve satış noktalarının konumları
(<https://drive.google.com/open?id=1QSWTXqzZLq-1cQNo3z13D58Pdml7twya&usp=sharing>, 28 Haziran 2019)

Eskişehir Halk Ekmek İşletmesi, satış büfelerine dağıtım sürecinde homojen yapıda olan 3 araca sahiptir. Araçlar, Isuzu marka ve NPR Long 2016 model olup özel olarak üretilmiş kasalardan toplam 150 adet kasa ekmek taşıma kapasitesine sahiptir. Her bir kasada 20 adet beyaz ekmek/kepekli ekmek/tam buğday türünde ekmek ayrıca 80 adet roll ekmek yer alabilmektedir.

Siparişler önceden bellidir ve günlük olarak oluşturulmaktadır. Araçlar üretim tesisinden (depo) dağıtıma çıkmaktadır ve dağıtım sonunda tekrar üretim tesisine dönmektedir.

Birden fazla araç olduğu için ve belirli bir kapasite kısıtı olduğu için problem araç rotalama problemi kapsamında yer almaktadır. Ele alınan problemde amaç, satış büfelerinin siparişlerinin tümünü karşılayacak olan araç rotalarını belirleyerek, toplam en kısa mesafeyi elde etmektir.

Bu amaçla problemin varsayımları:

- Bir noktaya sadece bir aracın gitme zorunluluğu vardır ve bir noktaya sadece bir kez uğranılmalıdır.
- Her müşteri yalnızca bir araca atanmalıdır.
- Her araç sadece bir rotada kullanılacaktır.
- Her noktaya en az bir defa uğranılmalıdır.
- Varılan her nokta aynı araçla terkedilmelidir.
- Her araç dağıtım rotasına depoda başlayıp depoda sonlandırmaktadır.
- Her aracın kapasitesi 3000 adet beyaz ekmek, tam buğday ve kepekli ekmek kadardır. Eğer araç kapasitesi, rotasındaki noktaların sipariş miktarını (talebini) karşılayamayacaksa (kapasite aşımı), kapasitesi kadar ürünün dağıtımını rotasındaki belirli noktalara dağıtıp tekrar depoya geri dönmeli ve aracını doldurmalıdır. Rotasındaki tüm noktaların siparişleri karşılanana kadar bu döngüyü devam ettirmelidir.
- Dağıtım noktaları ve üretim tesisi (depo) noktası arasındaki mesafeler belirlidir ve herhangi bir durumdan (yol çalışması, kaza, hava şartları vb.) etkilenmediği varsayılmıştır.
- Herhangi bir zaman süresi ve servis süresi kısıtı bulunmamaktadır.

3.2.Uygulama Problemi Verileri

Hesaplamanın yapılabilmesi için ihtiyaç duyulan veriler; depo ve satış büfelerinin isimleri ve numaralandırılması, her noktanın önceden belirli günlük sipariş miktarı ve noktalar arasındaki mesafe matrisi gibi bilgilerden oluşmaktadır. Uygulamada Eskişehir Halk Ekmek İşletmesi'nin bir güne ait sipariş miktarlarından yararlanılmıştır ve yaklaşık olarak diğer günlerde de aynı miktarlarda sipariş olacağı öngörülmüştür.

Eskişehir Halk Ekmek İşletmesi'nin üretim tesisinde dört farklı ürün çeşitliliği bulunduğu için kapasite hesaplaması sırasında zorluk yaşanmaktadır. Bu karmaşıklığı gidermek amacıyla ekmeklere kapasite bazında eşitleme getirecek bir düzenleme yapılmıştır. Bir kasada 20 adet beyaz ekmek/kepekli ekmek/tam buğday ekmek yer alabilirken 80 adet roll ekmek yer almaktadır. Bu sebeple sipariş miktarı verilerinde her dört roll ekmek, bir beyaz ekmek/kepekli ekmek/tam buğday ekmek olarak düşünülmektedir. Eskişehir Halk Ekmek İşletmesi'ne ait bir günlük sipariş miktarı

tablosu Tablo 3.1.'de verilmektedir. Üretim tesisinin numaralandırılması bir (1) olarak belirlendiği için satış büfelerinin numaralandırılması ikiden (2) başlatılmıştır.

Tablo 3.1. *Eskişehir Halk Ekmek İşletmesi bir günlük sipariş miktarları*

Satış Büfelerinin Numaraları	Satış Büfelerinin İsimleri	Sipariş Miktarları (adet)				Toplam Satış Miktarları (adet)
		Beyaz Ekmek	Kepekli Ekmek	Tam Buğday Ekmek	Roll Ekmek (Eşitlendi)	
2	71 Evler-1	1000	40	40	-	1080
3	71 Evler-2	200	-	20	-	220
4	Bağlar	160	40	40	30	270
5	Batıkent	160	40	40	15	255
6	Batıkent-2	50	20	30	13	113
7	Büyükdere	740	60	100	18	918
8	Çamlıca-1	340	40	60	8	448
9	Çamlıca-2	580	40	40	24	684
10	Çamlıca-3	600	20	40	4	664
11	Çamlıca-4	520	20	40	7	587
12	Çankaya	160	-	-	-	160
13	Derman	500	40	60	8	608
14	Elmalı	400	20	20	9	449
15	Emek-2	800	40	40	-	880
16	Emek-3	400	-	20	5	425
17	Emek-4	1100	40	40	8	1188
18	Emeksiz	460	20	40	10	530
19	Erenköy-1	500	20	20	38	578
20	Ertaş	800	40	60	13	913
21	Esentepe	400	20	20	3	443
22	Fevziçakmak-1	360	-	-	-	360
23	Fevziçakmak-2	400	-	10	-	410
24	Gökmeşdan-2	500	40	110	30	680
25	Göztepe	200	20	40	13	273
26	Gültepe	700	20	80	88	888
27	Hava Hastanesi	760	120	220	85	1185
28	Kırım	260	40	40	8	348

Tablo 3.1. (Devam) *Eskişehir Halk Ekmek İşletmesi bir günlük sipariş miktarları*

Satış Büfelerinin Numaraları	Satış Büfelerinin İsimleri	Sipariş Miktarları (adet)				Toplam Satış Miktarları (adet)
		Beyaz Ekmek	Kepekli Ekmek	Tam Buğday Ekmek	Roll Ekmek (Eşitlendi)	
29	Kurtuluş	400	60	100	-	560
30	Odunpazarı	640	80	160	-	880
31	Opera	280	40	80	-	400
32	Ömerağa	340	40	60	-	440
33	Savaş	280	40	140	30	490
34	Sihhiye	260	40	60	18	378
35	Sultandere-1	280	20	40	25	365
36	Şahintepesi	360	40	40	13	453
37	Şarhöyük	220	-	-	-	220
38	Şirintepe	440	40	60	18	558
39	Takkalı-1	800	40	40	122	1002
40	Takkalı-2	400	20	20	-	440
41	Tepebaşı	700	80	140	23	943
42	Toki-3	300	20	80	17	417
43	Toki-5	400	20	20	-	440
44	Uluönder	300	20	40	12	372
45	Vişnelik	340	40	200	48	628
46	Yenidoğan-1	600	60	80	38	778
47	Yenikent	560	60	160	29	809
48	Yeşiltepe-1	400	20	20	7	447
49	Yeşiltepe-2	400	-	-	-	400
50	Yimpaş	280	20	40	12	352
51	Yunustepe	700	60	60	8	828
52	Zincirlikuyu	480	40	40	7	567

Üretim tesisi ve satış büfeleri arasındaki mesafeler Google Maps (<https://www.google.com/maps>) aracılığı ile bulunmuştur. Mesafe değerlerinin belirlenmesi aşamasında koordinat veya kuş uçuşu mesafe değerleri yerine gerçek yol güzergahının km değerlerinin kullanılması sayesinde uygulamanın daha gerçekçi sonuçlar vermesi amaçlanmaktadır. Araçlar dağıtımına sabaha doğru saat 03:00'da başladığı için trafik yoğunluğunun yaşanması durumu göz ardı edilmiştir. Mesafe matrisi simetrik bir matris olarak tasarlanmıştır ve herhangi iki noktanın gidiş ve geliş güzergahı arasındaki yolların kullanım şekli nedeniyle karşılaşılabilen düşük değerli kilometre (km) farkı yok sayılmıştır. Bu açıklamalar doğrultusunda Microsoft Excel programı üzerinde oluşturulan mesafe matrisi Şekil 3.3.'de gösterilmektedir.

Eseritepe	Fevziçakmak-1	Fevziçakmak-2	Gökmeşen-2	Göztepe	Gültepe	Hava Haskeseni	Kırm	Kurtuluş	Odunazari	Opera	Ömerga	Savaş	Sihhiye	Sultandere-1	Şahinçepesi	Şarhöyük	Şirinepe	Takka-1
2,6	5,1	6,5	8,6	9,5	6,1	2,8	2,1	3,6	4,1	2,3	2	3,5	1,2	17,1	15,5	4	3,1	10
10,7	7,7	8,9	6	5,8	8,8	8,7	9,7	7,7	8,1	9,3	9,8	8,7	10,8	7,1	8,8	8,8	12,9	3,8
10,8	7,8	9	6,1	9,6	8,9	8,8	9,8	7,8	8,2	9,4	9,9	8,8	10,9	7,1	5,1	8,9	13,1	3,9
2,1	4,6	5,4	4,6	4,6	5,2	2,3	1,6	2,5	3,6	2	1,5	3	0,7	16,6	12,7	3,2	2,6	9,4
5,7	8,2	9	11,7	11,7	8,3	5,8	6,4	7,9	7,1	7,1	6,4	4,9	20,1	20,1	8,1	2,3	12,9	12,9
8,7	11,2	12,6	14,7	11,7	11,3	8,8	8,7	9,7	10,1	9,2	9	9,4	7,7	23,1	15,3	11,1	5	15,9
6,9	8	8,1	4,6	1	0,7	3,3	4,7	4,2	3,3	5,2	5,1	2,7	5,5	16,5	4,8	6,3	6,8	8,5
6,3	8,8	9,6	7	4,9	5,5	4,1	7	5,8	5,4	7,7	6,9	4,7	6	20,7	10,4	8,7	3,5	11
5,9	9,3	10,2	10,3	6,8	6,9	5,9	7,6	9,1	7,2	7,4	7,5	6,5	6,6	21,3	11	9,3	4,3	14,1
7,1	10,6	12,1	11,8	7,6	7,7	7,6	8,2	10,4	8,9	8,7	8,8	8,3	7,9	22,6	11,8	10,6	4,9	15,4
6,1	8,5	10	12	6,5	6,6	5,5	6,1	8,3	6,8	6,5	5,8	6,1	5,8	20,5	10,7	8,5	3,5	13,3
9,9	6,9	8	3,6	5,5	4,8	4,7	5,7	4,3	4,1	5,3	5,8	4,7	6,8	9,4	1,6	6,4	12,1	4,3
2	2,9	3,2	4,1	4,9	5,9	2,6	1,2	3	3,9	1,3	0,7	3,2	0,9	14,8	13,4	1,3	3,6	7,6
6,8	9,3	10,1	12,8	6,3	6,4	5,4	6,8	9	6,7	7,3	7,4	6	6,6	21,2	10,5	9,3	4,3	14
9,6	6,7	7,8	4,6	8	7,3	7,2	8,6	6,6	6,6	8,2	8,7	7,2	9,7	9,2	3,5	7,7	11,9	3,3
10,3	7,3	7,8	3,9	7,3	6,5	6,5	7,5	5,4	5,9	7,1	7,6	6,5	8,6	9	2,8	7,7	12,5	2,7
9,4	6,5	7,6	3,4	7,9	7,1	7	7,5	6,2	4,9	6,4	6,9	5,7	9,5	9,7	4,4	7,5	11,7	2,5
8,5	5,6	6,7	2,5	5,5	5,5	4,7	5,7	4	4,1	5,3	5,7	4,7	8,6	11	2,5	6,6	10,8	3,7
9,5	6,5	7,7	4,8	8,1	7,4	7,3	7,6	6,3	6,5	7,2	7,7	7,3	9,6	9	3,6	7,6	11,8	2,6
.	4,2	5	5,3	6,6	6,9	3,6	2,4	3,9	4,7	3,1	2,3	4,3	1,7	16,2	11,8	3,6	2,2	9
4,2	.	0,9	6	7,4	7,7	4,8	3,4	4,6	5,4	3	3,5	5,5	4,2	14,5	8,9	2,3	5,8	7,2
5	0,9	.	6,1	7,6	7,8	5	3,6	4,7	5,6	3,2	3,6	5,6	4,7	14,6	9	2,6	7,1	7,4
5,3	6	.	.	3,7	4	2,7	3,2	1,4	2,1	2,9	3,3	2,9	4,3	11,8	4,3	3,9	10	4,6
6,6	7,4	7,6	3,7	.	1,2	2,2	4,1	3,4	2,5	4,4	4,4	1,6	4,8	14,9	5,4	5,4	6,2	7,7
6,9	7,7	7,8	4	1,2	3,2	3,2	4,9	3,6	2,7	4,6	5,1	2,7	5,6	33,9	4,1	5,7	7	7,9
3,6	4,8	5	2,7	2,2	.	.	2	1,7	1,3	2,5	2,3	0,7	2,7	33,8	5,3	3,6	4,1	6,9
2,4	3,4	3,6	3,2	4,1	4,9	2	.	1,8	2,6	1	0,6	2,2	3	15,5	6,1	2,2	4	8,3
3,9	4,6	4,7	1,4	3,4	3,6	1,7	1,8	1,2	1,5	2,4	2,9	1,7	3	13	4,8	2,6	5,9	5,8
4,7	5,4	5,6	2,1	2,5	2,7	1,3	2,6	1,2	2,4	2,4	2,9	1,3	3,9	35,8	4,1	3,5	5,4	5,8
3,1	3	3,2	2,9	4,4	4,6	2,5	1	1,5	2,4	.	1	2,5	2	14,9	5,9	1,6	4	7,9
2,3	3,5	3,6	3,3	4,4	5,1	2,3	0,6	2	2,9	1	.	2,5	0,9	15,1	6,6	4	7,9	7,9
4,3	5,5	5,6	2,9	1,6	2,7	0,7	2,2	1,7	1,3	2,4	2,5	3,8	3,8	38,4	5,7	4,6	5,1	7,4
1,7	4,2	4,7	4,3	4,8	5,6	2,7	1	3	3,9	2	0,9	3,8	.	16,4	16,4	16,4	16,4	16,4
16,2	14,5	14,6	11,8	14,9	33,9	33,8	15,5	13	35,8	14,9	15,1	38,4	.	.	10,2	14,3	18,5	9,3
11,8	8,9	9	4,3	5,4	4,1	5,3	6,1	4,8	4,1	5,9	6,6	5,7	16,4	10,2	.	6,9	13,1	5,9
3,6	2,3	2,6	3,9	5,4	5,7	3,6	2,2	2,6	3,5	1,6	1,5	4,6	16,4	14,3	6,9	5,9	5,9	5,9
2,2	5,8	7,1	10	6,2	7	4,1	4	5,9	5,4	5,8	4	5,1	16,4	18,5	13,1	5,9	.	11,9
9	7,2	7,4	4,6	7,7	7,9	6,9	8,3	5,8	5,8	7,7	7,9	7,4	16,4	10,9	5,9	7,4	13,3	0,6
10,4	8,7	8,8	6,1	9,3	9,5	8,5	9,7	7,2	7,3	9,1	9,3	9	16,4	10,3	7,5	5,9	11,9	.
3,6	6,1	6,9	6,9	5,9	6,7	3,8	3,8	5,6	5,2	5,5	4,1	4,8	16,4	18,2	12	5,6	1,3	9,6
7,4	8,9	8,2	3,5	4,8	3,5	4,5	5,3	4	3,3	5,1	5,8	5	16,4	11,2	2,1	5,6	9	5,9
8,7	7	7,1	3,1	4,9	5,5	4,1	4,9	3,6	3	4,7	5,4	4,6	16,4	10,3	0,6	5,6	11,6	4,1
4,5	6,9	7,7	7,8	7,1	7,2	5	4,7	6,6	9,1	5,6	5,2	6,9	16,4	19	11,2	6,5	2,2	10,5
5,3	7,1	6,7	5,1	1,3	3,1	1,9	3,7	3,6	3,7	3,7	4	2,3	16,4	37,6	8,1	5,4	5,4	8,2
7,3	5,8	5,9	1,8	4,8	5	4	4,6	3	2,8	4,4	4,8	4,5	16,4	10,8	4,2	4,5	10,9	3,3
6,6	7,4	7,5	3,7	1,6	0,8	2,9	4,4	3,3	2,3	4,4	3,3	3,4	16,4	33,6	3,6	5,7	7,5	7,9
1	5	5,8	8,6	7,9	8,7	5,1	4,4	5,4	6,3	4,4	3,8	6,8	16,4	17,1	12,3	4,6	1,4	8,5
1,5	5,5	6,9	7,2	7	7,8	4,9	3,9	5,9	6,2	4,9	3,8	5,9	16,4	17,6	12,9	5,1	1,1	9,1
4,1	4,2	4,3	1,6	4	4,3	2,5	1,9	0,8	2,1	1,4	1,9	3	16,4	13,3	5,2	2,5	6,1	4,7
1,8	5,3	6,1	7	6,5	7,3	4,4	3,7	5,7	5,7	4,7	3,6	5,4	16,4	17,4	12,7	4,9	0,6	8,9
4,1	7,3	8,6	12,6	9,2	9,3	5	5,6	8,7	6,3	7,7	5,5	8,7	16,4	21,2	13,3	6,8	1,5	12,6

Şekil 3.3. (Devam) Eskişehir Halk Ekmek İşletmesi üretim tesisi ve satış büfeleri arasındaki mesafe matrisi (km)

Takhi-2	Tepelasi	Tokli-3	Tokli-5	Uluener	Viyapasi	Yenidogan-1	Yenilener	Yesiltepe-1	Yesiltepe-2	Yimyas	Yonustepe	Zinciflikuyu
11,3	2,8	7,3	7	3,8	3,7	8,4	5,8	3,4	3,9	4	3,7	5,9
5,3	12,7	4,9	5,2	13,5	11,7	3,7	7,9	11,6	12,1	7,8	11,9	15,6
5,5	12,8	5	5,4	13,6	11,9	3,8	8,1	11,7	12,2	7,9	12	15,7
10,8	2,3	6,8	9,1	3,3	3,4	7,9	5,2	2,9	3,4	3,1	2,9	5,4
14,4	1,8	11,1	13	2,4	6,7	11,6	10	4,3	4	12,5	2,9	1,5
17,4	4,6	12,7	15,6	4,6	7,5	14,5	13	7	6,2	11,2	5,7	4,2
10,1	6,6	4,2	5,7	7	2	5,6	1,5	7,8	7,6	4,9	7,1	7,7
12,6	2,6	9,6	8,2	2,3	3,9	8,2	6,4	6,9	6,3	5,9	4,2	3,5
15,6	3,4	10,5	11,5	3,1	5,5	11,4	8,7	7,2	8,2	8,5	5	4,9
16,9	4,4	11,2	12,8	4,5	7,3	14	9,4	7,9	7,8	10,7	7,3	5,8
14,8	2,6	10,2	11,7	2,3	5,1	11,8	8,4	6,5	4,7	8,4	4,2	3,4
5,9	11,8	1,1	1,1	11,6	7,7	2,2	4	10,8	11,3	4,6	11,1	13,7
9,1	4	9,3	7,3	4,8	4,9	6,2	5,7	2,9	3,4	2,4	3,2	5,1
15,5	3,4	10	11,5	3,1	4,7	12,6	8,2	7,6	6,8	8,4	6,3	4,8
4,3	11,6	3,4	2,7	12,4	10,9	2,6	6,4	10,5	11	6,7	10,8	14,6
4,9	12,2	2,7	2,2	13,1	8,6	3,4	5,7	11,2	11,7	5,5	11,5	15,2
4,4	11,7	2,6	1,5	12,5	10	1,7	5,5	10,6	11,1	4,8	10,9	14,6
2,5	11,4	3,4	2	12,3	9,1	1,8	6,3	10,3	10,9	6,5	10,7	14,4
5,2	10,5	1,7	0,5	11,4	6,9	1	4,6	9,4	9,9	4,1	9,8	13,5
4,2	11,5	3,5	2,6	12,3	9,4	2,5	6,5	10,4	10,9	6,3	10,7	14,4
10,4	3,6	7,4	8,7	4,5	5,3	7,3	6,6	1	1,5	4,1	1,8	4,1
8,7	6,1	8,9	7	6,9	7,1	5,8	7,4	5	5,5	4,2	5,3	7,3
8,8	6,9	8,2	7,1	7,7	6,7	5,9	7,5	5,8	6,9	4,3	6,1	8,6
6,1	6,9	3,5	3,1	7,8	3,1	1,8	3,7	8,6	7,2	1,6	7	12,6
9,3	5,9	4,8	4,9	7,1	1,3	4,8	1,6	7,9	7	4	6,5	9,2
9,5	6,7	3,5	5,5	7,2	3,1	5	0,8	8,7	7,8	4,3	7,3	9,3
8,5	3,8	4,5	4,1	5	1,9	4	2,9	5,1	4,9	2,5	4,4	5
9,7	3,8	5,3	4,9	4,7	3,7	4,6	4,4	3,4	3,9	1,9	3,7	5,6
7,2	5,6	4	3,6	6,6	3,6	3	3,3	5,4	5,9	0,8	5,7	8,7
7,3	5,2	3,3	3	9,1	3,7	2,8	2,3	6,3	6,2	2,1	5,7	6,3
9,1	5,5	5,1	4,7	5,6	3,7	4,4	4,4	4,4	4,9	1,4	4,7	7,7
9,3	4,1	5,8	5,4	5,2	4	4,8	4,8	3,3	3,8	1,9	3,6	5,5
9	4,8	5	4,6	6,9	2,3	4,5	3,4	6,8	5,9	3	5,4	8,7
16,4	16,4	16,4	16,4	16,4	16,4	16,4	16,4	16,4	16,4	16,4	16,4	16,4
10,9	18,2	10,3	11,2	19	37,6	10,8	33,6	17,1	17,6	13,3	17,4	21,2
7,5	12	0,6	2,1	11,2	8,1	4,2	3,6	12,3	12,9	5,2	12,7	13,3
7,4	5,6	7,6	5,6	6,5	5,4	4,5	5,7	4,6	5,1	2,5	4,9	6,8
13,3	1,3	9	11,6	2,2	5,4	10,9	7,5	1,4	1,1	6,1	0,6	1,5
0,6	9,6	5,9	4,1	10,5	8,2	3,3	7,9	8,5	9,1	4,7	8,9	12,6
-	9,1	6,3	4,3	9,9	7,6	3,2	7,4	8	8,5	4,2	8,3	12
9,1	-	11,8	12,6	0,8	6,7	11,5	10	3,3	2,5	5,8	2,1	1,5
6,3	11,8	-	1,4	11,4	6,7	3,5	3,8	11,7	12,2	4,5	12	13,5
4,3	12,6	1,4	-	11,7	6,4	1,3	4,7	9,8	10,3	5,2	10,1	13,8
9,9	0,8	11,4	11,7	-	5,2	11,7	10,2	5,2	3,2	5,9	2,7	2,4
7,6	6	6,7	6,4	5,2	-	5,1	2,7	7,1	6,2	3,7	5,7	8,3
3,2	11,5	3,5	1,3	11,7	5,1	-	4,8	8,4	8,9	3,1	8,7	12,4
7,4	10	3,8	4,7	10,2	2,7	4,8	-	8,7	7,8	4	7,4	11
8	3,3	11,7	9,8	5,2	7,1	8,4	8,7	-	0,5	5	1	3,1
8,5	2,5	12,2	10,3	3,2	6,2	8,9	7,8	0,5	-	5,6	0,6	2,8
4,2	5,8	4,5	4,2	5,9	3,7	3,1	4	5	-	5,4	-	8,3
8,3	2,1	12	10,1	2,7	5,7	8,7	7,4	1	0,6	5,4	-	2,2
12	1,5	13,5	13,8	2,4	8,3	12,4	11	3,1	2,8	8,3	2,2	-

Şekil 3.3. (Devam) Eskişehir Halk Ekmek İşletmesi üretim tesisi ve satış büfeleri arasındaki mesafe matrisi (km)

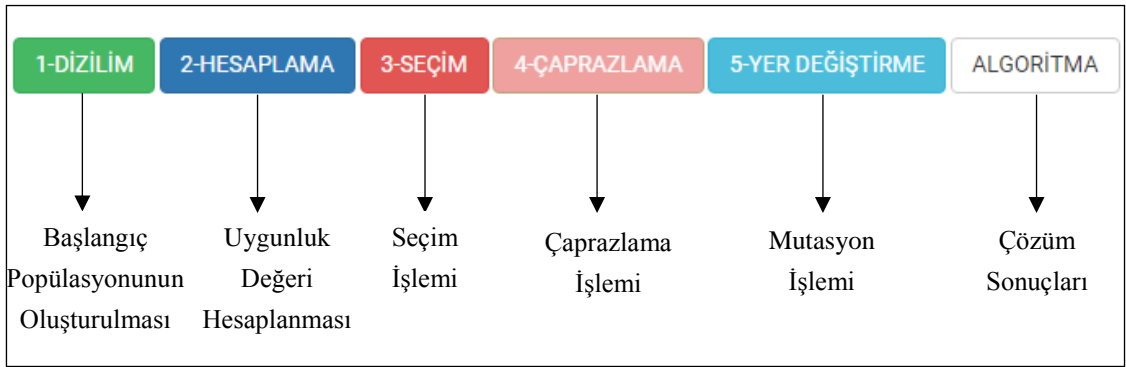
3.3.Uygulama Probleminin Genetik Algoritma ile Çözümü

Eskişehir Halk Ekmek İşletmesi satış büfelerine ekmek dağıtım sürecinde araç rotasyonunun belirlenmesi probleminde genetik algoritmanın kullanılabilmesi için hesaplamayı sağlayacak bir yazılıma gereksinim duyulmaktadır. Hesaplamalar çok uzun süreceği için elle hesaplama yapmak mümkün değildir. Bu sebeple, bilgisayarların hızlı işlem yapma özelliğinden yararlanmak amacıyla bir yazılım geliştirilmiştir.

Problemin genetik algoritma ile çözümünde, Java programlama dili kullanılarak Java web uygulaması şeklinde geliştirilmiştir. Ara yüz teknolojisi olarak AngularJs, JavaScript, HTML, CSS ve Bootstrap teknolojileri kullanılmıştır. Veri tabanı olarak ise

MySQL kullanılmıştır. Geliştirilen yazılım, Windows 10 işletim sisteminde çalışan, SSD sabit disk, 8 GB RAM ve 3.20 GHz 4 çekirdekli AMD işlemciye sahip bir bilgisayar üzerinde denenmiştir. Geliştirilen yazılım, daha sonra internet ortamında erişilebilir hale getirilmiştir.

Öncelikli olarak, uygulama verileri program veri tabanına eklenmelidir. Sonrasında genetik algoritmanın çalışma düzenine göre çalıştırılmaktadır. Yazılımın arayüzü, genetik algoritmanın çalışma prensibini temel alan 6 kısımdan oluşmaktadır ve Şekil 3.4.'te gösterilmektedir. Başlangıç popülasyonunun oluşturulmasından başlayarak hesaplama, seçim, çaprazlama, mutasyon ve son olarak genetik algoritma çözüm sonuçlarının gösterildiği algoritma kısmı şeklinde devam etmektedir.



Şekil 3.4. Yazılımın arayüzündeki kısımların görseli

3.3.1.Kromozom yapısı ve başlangıç popülasyonu

Eskişehir Halk Ekmek İşletmesi ekmek dağıtım probleminde, sıralamanın önemli olması sebebiyle permutasyon kodlama tercih edilmiştir. Kromozomlar, permutasyon şeklinde kodlanan rotasyonlardan oluşmaktadır. Dizilimdeki her öge sıralamada gidilecek bir sonraki noktayı temsil etmektedir. Bu durumda dizilimde eksik eleman olmayacak ve her bir noktanın bir kere bulunmasını sağlayacak şekilde sıralama yapılmalıdır.

Genetik algoritmanın temel kavramları, Eskişehir Halk Ekmek İşletmesi ekmek dağıtım probleminde:

- Gen: Dağıtım rotasındaki üretim tesisi ve 51 satış büfesinden oluşan noktaların her biri,
- Kromozom: Problemin olası çözümünü temsil eden araç rota güzergâhlarının her biri,

- Başlangıç Popülasyonu: Rastgele oluşturulan araç rota güzergâhlarının kümesi şekilde ifade edilmektedir.

Genetik algoritma da her nesilde iyileşme işleminin yapılabilmesi için bir popülasyona ihtiyaç vardır. Popülasyon kromozomlardan oluşmaktadır, kromozomlar ise noktalardan oluşmaktadır. Başlangıçta rastgele oluşturulan kromozomlar, her yinelemede çaprazlama ve mutasyon işlemleri yardımıyla iyileştirilir.

Eskişehir Halk Ekmek İşletmesi'nin dağıtım sürecinde kapasite kısıtlaması olması sebebiyle rastgele oluşturulan başlangıç popülasyonuna, sipariş miktarlarından yararlanılarak kapasite kısıtlamasını karşılayacak şekilde dizilimde dağıtım noktalarının arasına üretim tesisi noktası eklenmelidir. Bu sayede, dizilimde araçların rotasyonunda gitmesi gereken bir sonraki noktanın bir satış büfesi mi yoksa üretim tesisi mi olması gerektiği belirlenmesi sağlanmaktadır.

Başlangıç popülasyon büyüklüğü parametresi 100 olarak belirlenmiştir ve bu sayı nesiller boyunca aynı kalmaktadır. Bunun anlamı, popülasyon 100 tane kromozomdan oluşmaktadır. Popülasyon büyüklüğü, üretilecek çözüm sayısını ifade etmektedir. Problemin çözümünde en önemli işlem basamağı uygun kromozom yapısının oluşturularak, devamında genetik işlemlerin uygulanmasına elverişli hale getirmektir. Kromozom yapısı oluşturulurken üç araç için toplam en kısa mesafeyi bulmayı amaçlayan bir yapı tasarlanmıştır, çözüm sonuçları sonrasında bulunan rota üç araç arasında dengeli bir şekilde dağıtılmaya çalışılacaktır. Problemin kromozom yapısı ve başlangıç popülasyonu örneği Tablo 3.2'de verilmektedir.

Tablo 3.2. Probleme ait kromozom yapısı ve başlangıç popülasyon örneği

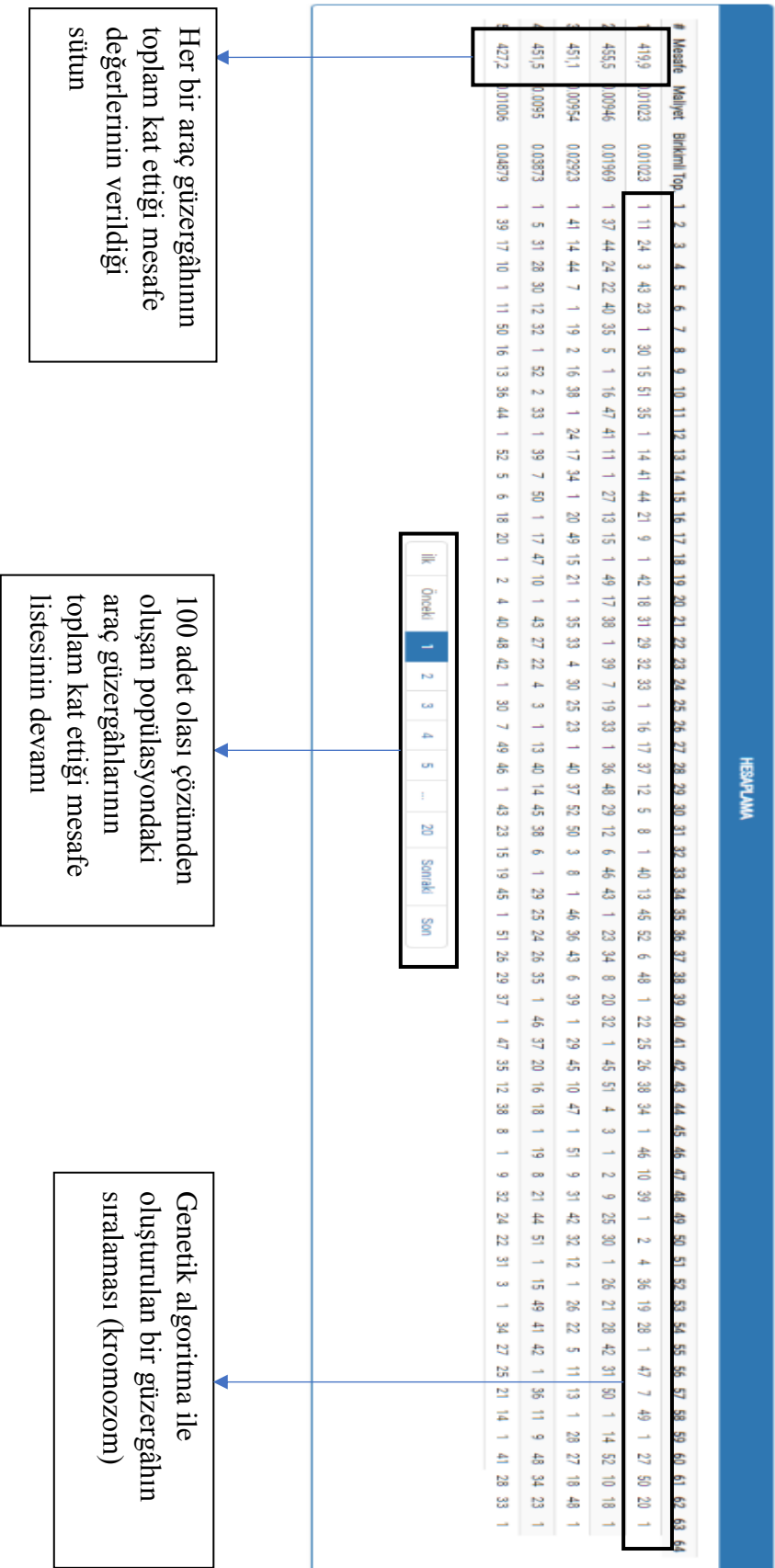
1. Dizilim	1	11	24	3	43	23	1	...	27	50	20	1
2. Dizilim	1	37	44	24	22	40	35	...	52	10	18	1
3. Dizilim	1	41	14	44	7	1	19	...	27	18	48	1
100. Dizilim	1	18	14	15	3	43	32	...	4	1	40	1

İkinci bölümde belirtildiği üzere genetik algoritmalarda kromozomlar çözüme ait tüm bilgileri içinde bulundurlar. Tabloda bir (1) ile gösterilen gen üretim tesisini, diğer sayılar ise dağıtım noktalarını ifade etmektedir. Problemin yapısı gereği dağıtım, üretim

tesisinden başlayarak üretim tesisinde sonlanması gerektiğinden kromozomun ilk ve son geninde üretim tesisi konumlandırılmıştır. Aynı zamanda, kapasite kısıtına uygun olacak şekilde dağıtım noktalarının arasına üretim tesisini ifade eden bir (1) sayısı tekrar kullanılmakta ve dizilimler bu kurala uygun olacak bir şekilde oluşturulmaktadır.

3.3.2.Uygunluk fonksiyonu

Kromozomlar oluşturulduktan sonra uygunluk değerinin hesaplanması gerekir. Bütün değerlendirmeler uygunluk değeri ile yapılır. Uygunluk değeri, kromozoma ait sıralamadaki ardışık her iki nokta arasındaki mesafenin toplanması sonucu elde edilmektedir. Problemin amacı en kısa mesafeyi elde etmek olduğu için uygunluk değeri ne kadar küçükse bulunan çözüm o kadar iyidir. Uygunluk fonksiyonuna ait geliştirilen yazılımın ara yüzü Şekil 3.5.'te verilmektedir.



Şekil 3. 5.Uygunluk fonksiyonunun geliştirilen yazılımdaki ara yüz görseli

3.3.3.Genetik operatörler

Genetik operatörlerin uygulanması aşamasında kromozom yapısına kapasite kısıtlaması nedeniyle eklenen üretim tesisi geni olmadan yani popülasyonun ham haliyle işlem yapılmaktadır. Bunun sebebi, işlemler sırasında oluşabilecek karmaşıklığı engellemektir. Dizilimlerin ham hallerinin geliştirilen yazılımın ara yüzündeki görseli Şekil 3.6.'te gösterilmektedir.

#	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51
1	11	24	3	43	23	30	15	51	35	14	41	44	21	9	42	18	31	29	32	33	16	17	37	12	5	8	40	13	45	52	6	48	22	25	26	38	34	46	10	39	2	4	36	19	28	47	7	49	27	50	20
2	37	44	24	22	40	35	5	16	47	41	11	27	13	15	49	17	38	39	7	19	33	36	48	29	12	6	46	43	23	34	8	20	32	45	51	4	3	2	9	25	30	26	21	28	42	31	50	14	52	10	18
3	41	14	44	7	19	2	16	38	24	17	34	20	49	15	21	35	33	4	30	25	23	40	37	52	50	3	8	46	36	43	6	39	29	45	10	47	51	9	31	42	32	12	26	22	5	11	13	28	27	18	48
4	5	31	28	30	12	32	52	2	33	39	7	50	17	47	10	43	27	22	4	3	13	40	14	45	38	6	29	25	24	26	35	46	37	20	16	18	19	8	21	44	51	15	49	41	42	36	11	9	48	34	23
5	39	17	10	11	50	16	13	36	44	52	5	6	18	20	2	4	40	48	42	30	7	49	46	43	23	15	19	45	51	26	29	37	47	35	12	38	8	9	32	24	22	31	3	34	27	25	21	14	41	28	33

Şekil 3.6.Başlangıç popülasyonunun geliştirilen yazılımdaki ara yüz görseli

Seçim işlemi: Başlangıç popülasyonunun oluşturulması ve popülasyondaki her kromozomun uygunluk değerinin hesaplanmasının ardından sıra seçim aşamasına gelir. İkinci bölümde belirtildiği üzere genetik algorithmada seçim için çeşitli seçim stratejileri mevcuttur. Bu çalışmada, seçim işleminde rulet tekerleği seçimi kullanılmaktadır. Seçim işlemiyle popülasyonda en güçlü adayların yani uygunluk değeri daha iyi olan adayların bir sonraki popülasyona aktarılma ihtimali artmaktadır.

Rulet tekerleği seçim tekniğinde; popülasyonda bulunan her güzergâhın (her kromozomun) uygunluk değeri öncelikle bire bölünmelidir. Problemin amacı en kısa mesafenin elde edilmesi olduğundan çözüm kümesi içinde en kısa mesafenin en büyük oranı vermesini sağlamak için bu işlem gerçekleştirilmektedir. Sonrasında bu oranların birikimli toplamları hesaplanır. İyi değere sahip kromozomların oranının daha yüksek olması sebebiyle bir sonraki popülasyona aktarılma ihtimalinin yüksek olması beklenir. Sonrasında popülasyon büyüklüğü kadar [0-1] arasında rasgele değerler üretilir, bu değer hangi dizilimin birikimli ihtimalinden küçükse o dizilim seçim işlemine tabi tutulur. Rulet tekerleği seçim tekniğinin yazılım ara yüzü üzerindeki görseli Şekil 3.7.'de verilmektedir.

SEÇİM		
#	Birikimli Toplam	Rastgele Oran
5	0.04879	0.04877
85	0.84786	0.84429
52	0.51735	0.50812
20	0.19754	0.1883
21	0.2076	0.20641

ilk Önceki 1 2 3 4 5 ... 20 Sonraki Son

Şekil 3.7. Seçim işleminin geliştirilen yazılımdaki ara yüz görseli

Elit seçim: Seçim stratejisi her ne kadar iyi adayları bulmayı amaçlasada, işlemler rasgele gerçekleşeceği için bazı güçlü adaylar seçim işleminde elenebilir. Bu durumun engellenmesi ve popülasyondaki en kısa mesafeyi veren araç güzergâhlarının mutlak suretle bir sonraki nesile aktarılabilmesi için elit seçim işlemi uygulanır. Bu çalışmada elitlik oranı 0,05 olarak belirlenmiştir. Elit seçim için, nesildeki güzergahlar (kromozomlar) her bir güzergâh için bulunan mesafe değerlerine (uygunluk değeri) göre sıralanır. Daha sonra elitlik oranı x popülasyon büyüklüğü kadar en kısa mesafeye sahip güzergahtan başlamak üzere bir sonraki nesile aktarılır.

Çaprazlama: Genetik algoritmada üreme çaprazlama işlemi ile gerçekleştirilir. Çaprazlamada amaç, popülasyondaki iki adayın kromozom özelliklerinden bir kısmının alınarak yeni bir kromozom oluşturulmasıdır. Çaprazlama için birçok strateji mevcuttur. Problemin yapısına uygun olacak şekilde bir çaprazlama tekniği seçilmelidir.

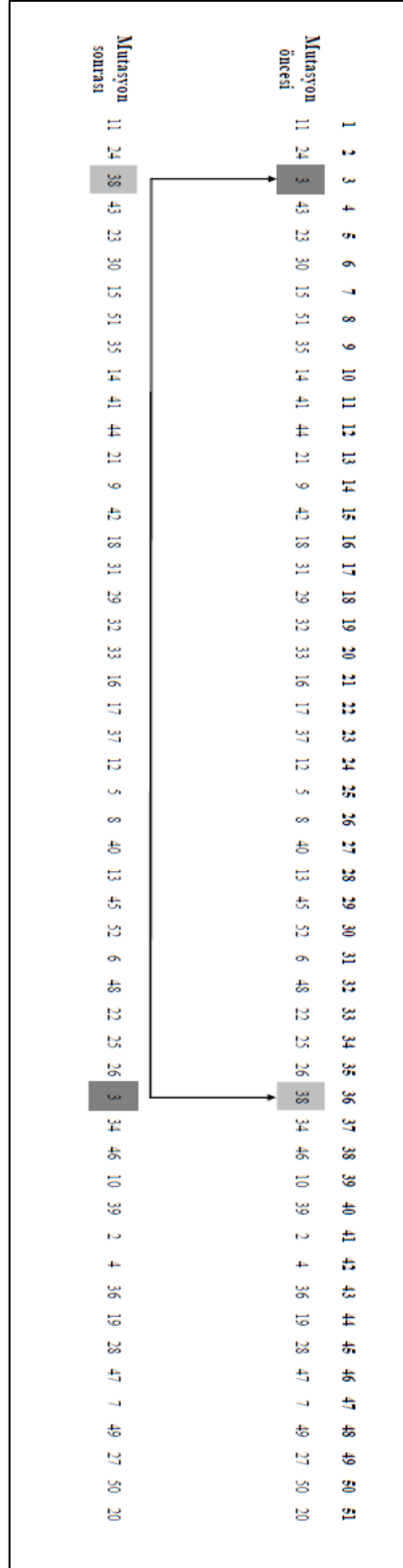
Seçilen çaprazlama tekniği, dizilimde eksik eleman olmaması ve her bir noktanın bir kere bulunması kurallarını sağlayacak şekilde oluşturulan kromozom yapısını bozmamalıdır. Bu çalışmada sıralı çaprazlama tekniği tercih edilmektedir. Bunun sebebi, araç rotalama problemlerine uygun bir teknik olmasıdır.

Çaprazlama işleminde diğer bir önemli konu da çaprazlama oranıdır. Literatürde bu konuda kesin bir kısıtlama olmamasına karşın çoğunlukla kullanılan oran %60'tır (Tüfekçier, 2008).

Eskişehir Halk Ekmek İşletmesi için olası 100 adet çözüm içinden belli bir oranda rastgele olarak seçilen araç güzergâhları ikili olarak eşleştirilir. Seçilen çaprazlama tekniğinin uygulanmasıyla eşleşen araç güzergâhları ile yeni araç güzergâhları oluşturulur. Sonrasında bulunan yeni araç güzergâhları çözüm kümesine eklenerek mesafe değerleri hesaplanır. Problemin amaç fonksiyonuyla çelişen yüksek mesafe değerine sahip araç güzergâhları çözüm kümesinden atılarak yine 100 adet çözüm değeriyle işlemlere devam edilmektedir.

Mutasyon: Genetik algoritmada mutasyon işlemi, çözüm sırasında yerel optimuma takılmamak için yapılan etkili bir işlemdir. Problem kimi zaman bir çözüm üzerinde yoğunlaşır ve sabitlenir. Bu durumda, bu kısır döngüden çıkmanın en iyi yolu mutasyon operatörünün uygulanmasıdır. Mutasyon için bugüne kadar çok çeşitli teknikler kullanılmıştır. Sıklıkla başvurulan teknik rasgele bir genin değiştirilmesi veya bir başka gen ile yer değiştirme yapılmasıdır (Ünal, 2018).

Bu çalışmada, mutasyon işlemi sırasında yer deęiřtirme teknięi kullanılmaktadır. Rasgele olarak seilen herhangi iki noktanın (genin) birbiri ile yer deęiřtirilmesiyle uygulanır. Problem yapısı gereęi güzergahlar oluřturulurken sıralamanın önemli olması sebebiyle herhangi bir noktanın deęiřtirilmesiyle mutasyon işleminin uygulanması mümkün olmamaktadır. Herhangi bir noktanın deęiřtirilmesi rotasyonda tüm noktaların var olmasını engelleyecek ve aynı noktanın iki kez tekrarlanmasına sebep olacaktır. Eskiřehir Halk Ekmek İşletmesi ekmek daęıtım probleminde mutasyon işleminin nasıl uygulandıęını gösteren örnek Şekil 3.8.'de verilmektedir. Şekil 3.8.'de verilen Eskiřehir Halk Ekmek İşletmesi ekmek daęıtım probleminde mutasyon işlemi örneęinde, rastgele olarak seilen 3. ve 36. noktadaki satış büfeleri yerlerinin karřılıklı olarak yer deęiřtirilmesiyle mutasyon işlemi gerekleřtirilmektedir ve yeni bir güzergâh oluřturulmaktadır.



Şekil 3.8. Eskişehir Halk Ekmek İşletmesi ekmek dağıtım probleminde mutasyon işlemi örneği

Genetik operatörlerin uygulanmasında kullanılan oranlar problemin çözüm kalitesi üzerinde oldukça önemli olmaktadır. Genetik algoritma için De Jong (1975), Grefenstette (1986) ve Schaffer vd. (1989) tarafından önerilen parametre değerleri aşağıdaki Tablo 3.2.'de verilmektedir.

Tablo 3.2. Genetik algoritmalar için önerilen parametre değerleri (Cevre, 2008)

Kontrol Parametreleri	De Jong	Grefenstette	Schaffer
Popülasyon Büyüklüğü	50- 100	30	20 – 30
Çaprazlama Oranı	0,60	0,95	0,75 – 0,95
Mutasyon Oranı	0,001	0,01	0,005 – 0,01

Literatürde yapılan çalışmalardan yararlanılarak bu çalışma kapsamında, genel kabul edilen oranlar tercih edilerek popülasyon büyüklüğü 100, çaprazlama oranı 0,60 ve mutasyon oranı 0,01 olarak kullanılmaktadır ve bu değerler tüm uygulama boyunca sabit tutulmaktadır.

Çaprazlama oranı x popülasyon büyüklüğü kadar kromozom çaprazlama işlemi sonucunda oluşturulur. Mutasyon işlemi için de benzer şekildedir. Mutasyon oranı x popülasyon büyüklüğü kadar kromozom mutasyona uğratılmaktadır.

3.3.4.Sonlandırma kriteri

Sonlandırma kriteri kullanıcının tercihlerine göre değişiklik gösterebilir. Çalışma kapsamında, sonlandırma kriteri olarak maksimum yineleme sayısı belirlenmiştir ve işleyiş sürecinde durdurma işlemi yapılmadığı takdirde genetik algoritma bu sayıya ulaşıncaya kadar devam eder. Ele alınan problemde maksimum nesil sayısı 2000 olarak belirlenmiştir. Bu değer, genetik algoritma adımlarının kaç kez tekrarlanacağını ifade etmektedir.

Belirlenen maksimum yineleme sayısı kadar tekrarlanan nesil oluşturma işlemlerinden sonra bulunan minimum mesafeye sahip kromozom, optimizasyon probleminin çözümüdür.

3.4. Analiz Sonuçları

Ele alınan problem için genetik algoritma 2000 kez çalıştırılmıştır ve yaklaşık olarak 25 saniye (sn) gibi kısa bir sürede sonuca ulaşılmıştır. Araç rotalama problemi kapsamında yer alan bu problemde çözümün hızlı ve optimuma yakın olması amacıyla genetik algoritma tercih edilmektedir. Genetik algoritma ile belirlenen araç rotalama işlemi sonucunda Eskişehir Halk Ekmek İşletmesi'nin mevcut olarak kullandığı rotasyon iyileştirilerek daha iyi bir sonuç elde edilmesi sağlanmıştır.

Genetik algoritma optimum sonucu garanti etmez fakat optimuma yakın sonuçları garanti eder ve uygulamada da görüldüğü gibi pek çok çözüm seçeneği sunar. Çalışma öncesinde, Eskişehir Halk Ekmek İşletmesi rotalarının oluşturulmasında 17 yıllık tecrübesinden yararlanarak 3 aracın toplam kat ettiği mesafeyi 236,54 km olarak belirlediği görülmektedir. Genetik algoritma ile hesaplama sonucunda ise 3 aracın toplam kat ettiği mesafe 210,24 km olarak hesaplanmıştır. Bu durumda, günlük olarak hesaplanan değerde 26,3 km'lik bir iyileşme sağlanmaktadır.

Bir günlük siparişlere göre belirlenen rotalama işleminin ayın diğer günleri içinde aynı olacağı düşünüldüğünde, Eskişehir Halk Ekmek İşletmesi'nin mevcut kullanmakta olduğu rotasyona göre aylık olarak yaklaşık 790 km'lik bir farkla iyileşme görülmektedir. Bu durumda, işletmenin hem maliyet hem zaman olarak kar etmesini sağlamaktadır. Maliyet bazında bakıldığında, kullanılan araçlar şehir içinde gidilen her km'de ortalama 1,6 TL/lt yakıt harcamaktadır ve aylık olarak düşünüldüğünde işletme için 1.264 TL'lik bir yakıt tasarrufu ile maliyette azalma sağlanmaktadır.

Eskişehir Halk Ekmek İşletmesi'nin kullanmakta olduğu mevcut rota ve genetik algoritma ile çözüm sonucu oluşturulan rota arasındaki karşılaştırılmanın yapılabilmesi amacıyla Tablo 3.3. ve Tablo 3.4. ile araçların rotasyonları gösterilmektedir. Tablo 3.3.'te işletmenin üretim tesisinden çıkan 1. Araç ilk olarak 39 numaralı noktaya uğramaktadır, daha sonra 40 numaralı noktaya geçmektedir. Bu şekilde tüm satış büfelerine uğrayarak rotasında bulunan tüm satış büfelerinin talebini karşıladıktan sonra tekrar üretim tesisine dönmektedir. 1. Araç için üretim tesisi ve satış büfeleri arasındaki mesafe tablonun ikinci sütununda verilmektedir. Üretim tesisi ile 39 numaralı satış büfesinin arasındaki mesafe 10 km, 39 numaralı ve 40 numaralı satış büfesi arasındaki mesafe 0,6 km şeklindedir. En alt satırda ise tüm araçların toplam rota mesafesi 236,54 km olarak görülmektedir. Tablo 3.4.'te benzer şekilde bilgiler içermektedir.

Tablo 3. 3. Eskişehir Halk Ekmek İşletmesi'nin kullandığı mevcut rota

1. Araç		2. Araç		3. Araç	
Rota	Mesafe (km)	Rota	Mesafe (km)	Rota	Mesafe (km)
1 (Üretim Tesisi)		1 (Üretim Tesisi)		1 (Üretim Tesisi)	
39 (Takkalı-1)	10	27 (Hava Has.)	2,8	13 (Derman)	2,1
40 (Takkalı-29)	0,6	33 (Savaş)	0,7	22 (Fevziçakmak-1)	2,9
2 (71 Evler-1)	5,3	45 (Vişnelik)	2,3	23 (Fevziçakmak-2)	0,9
3 (71 Evler-2)	0,04	25 (Göztepe)	1,3	21 (Esentepe)	5
1 (Üretim Tesisi)	11,3	1 (Üretim Tesisi)	5,8	48 (Yeşiltepe-1)	1
35 (Sultandere-1)	17,1	7 (Büyükdere)	8,6	49 (Yeşiltepe-2)	0,5
20 (Ertaş)	9	26 (Gültepe)	0,7	1 (Üretim Tesisi)	3,9
17 (Emek-4)	1,1	47 (Yenikent)	0,8	51 (Yunustepe)	3,7
16 (Emek-3)	1	1 (Üretim Tesisi)	5,8	38 (Şirintepe)	0,6
1 (Üretim Tesisi)	11,3	36 (Şahintepesi)	15,5	52 (Zincirlikuyu)	1,5
15 (Emek-2)	10,5	42 (Toki-3)	0,6	41 (Tepebaşı)	1,5
43 (Toki-5)	2,7	12 (Çankaya)	1,1	1 (Üretim Tesisi)	2,8
19 (Erenköy-1)	0,5	30 (Odunpazarı)	4,1	44 (Uluönder)	3,8
46 (Yenidoğan-1)	1	29 (Kurtuluş)	1,2	8 (Çamlıca-1)	2,3
1 (Üretim Tesisi)	8,4	50 (Yimpaş)	0,8	14 (Elmalı)	0,9
24 (Gökmeydan-2)	8,6	1 (Üretim Tesisi)	4	9 (Çamlıca-2)	0,7
18 (Emeksiz)	3,4	31 (Opera)	2,8	10 (Çamlıca-3)	0,5
37 Şarhöyük)	7,5	28 (Kırım)	1	1 (Üretim Tesisi)	6,5
1 (Üretim Tesisi)	4	4 (Bağlar)	1,6	11 (Çamlıca-4)	5,7
		32 (Ömerağa)	1,5	6 (Batıkent-2)	3,4
		1 (Üretim Tesisi)	2	5 (Batıkent)	1,9
				34 (Sıhhiye)	4,9
				1 (Üretim Tesisi)	1,2
Toplam Mesafe	113,34	Toplam Mesafe	65	Toplam Mesafe	58,2
3 Aracın Toplam Mesafesi = 236,54					

Kullanılan mevcut rotada araçların kat ettiği mesafelerde iyi bir dağılım olmadığı görülmektedir. Taşıma kapasitesi sık sık yeniden doldurulduğu için yakın veya uzak fark etmeksizin her seferinde üretim tesisine dönüş yapılmaktadır. Genetik algoritma ile bulunan çözümde araçların kat ettiği mesafelerde her araca km bazında uygun bir dağılım yapılması sağlanmaktadır.

Tablo 3. 4. Genetik algoritma ile çözüm sonucu oluşturulan optimum rota

1. Araç		2. Araç		3. Araç	
Rota	Mesafe (km)	Rota	Mesafe (km)	Rota	Mesafe (km)
1 (Üretim Tesisi)		1 (Üretim Tesisi)		1 (Üretim Tesisi)	
41 (Tepebaşı)	2,8	28 (Kırım)	2,1	27 (Hava Has.)	2,8
44 (Uluönder)	0,8	24 (Gökmeydan-2)	3,2	47 (Yenikent)	2,9
8 (Çamlıca-1)	2,3	36 (Şahintepesi)	4,3	26 (Gültepe)	0,8
14 (Elmalı)	0,9	16 (Emek-3)	2,8	1 (Üretim Tesisi)	6,1
1 (Üretim Tesisi)	6,1	12 (Çankaya)	1,8	34 (Sıhhiye)	1,2
51 (Yusuntepe)	3,7	42 (Toki-3)	1,1	33 (Savaş)	3,8
38 (Şirintepe)	0,6	43 (Toki-5)	1,4	25 (Göztepe)	1,6
49 (Yeşiltepe-2)	1,1	1 (Üretim Tesisi)	7	7 (Büyükdere)	1
48 (Yeşiltepe-1)	0,5	30 (Odunpazarı)	4,1	45 (Vişnelik)	2
1 (Üretim Tesisi)	3,4	46 (Yenidoğan-1)	2,8	1 (Üretim Tesisi)	3,7
39 (Takkalı-1)	10	50 (Yimpaş)	3,1	17 (Emek-4)	10,7
3 (71 Evler-2)	3,9	29 (Kurtuluş)	0,8	15 (Emek-2)	1,3
2 (71 Evler-1)	0,04	1 (Üretim Tesisi)	3,6	18 (Emeksiz)	1,3
35 (Sultandere-1)	7,1	32 (Ömerağa)	2	1 (Üretim Tesisi)	9,1
1 (Üretim Tesisi)	17,1	40 (Takkalı-2)	9,3	52 (Zincirlikuyu)	5,9
21 (Esentepe)	2,6	20 (Ertaş)	4,2	5 (Batıkent)	1,5
22 (Fevziçakmak-1)	4,2	19 (Eranköy-1)	2	11 (Çamlıca-4)	1,8
23 (Fevziçakmak-2)	0,9	1 (Üretim Tesisi)	9,1	9 (Çamlıca-2)	0,9
31 (Opera)	3,2			10 (Çamlıca-3)	0,5
37 (Şarhöyük)	1,6			6 (Batıkent-2)	3,4
13 (Derman)	1,3			1 (Üretim Tesisi)	7
4 (Bağlar)	1,6				
1 (Üretim Tesisi)	0,5				
Toplam Mesafe	76,24	Toplam Mesafe	64,7	Toplam Mesafe	69,3
3 Aracın Toplam Mesafesi = 210,24					

İşletmenin kullandığı mevcut rota ve genetik algoritma ile bulunan rota karşılaştırıldığında bir günlük verilerden yararlanılarak oluşturulan rota planlamasında sağlanan iyileşme %11 gibi oldukça önemli bir değerdir. Aylık olarak düşünüldüğünde bu oran ile elde edilebilecek iyileşme işletme maliyetleri açısından oldukça önemli olmaktadır.

DÖRDÜNCÜ BÖLÜM

4.SONUÇ VE ÖNERİLER

Eskişehir Halk Ekmek İşletmesi ekmek dağıtımı için, araçların büfelere en kısa güzergâhtan rotalanması üzerine genetik algoritma ile çözümü gerçekleştirecek web ortamında çalışacak bir uygulama yazılmıştır ve mevcut araç rota planlaması iyileştirilmeye çalışılmıştır. Sonrasında mevcut ve önerilen sistemlerde araç rotalarının karşılaştırılması yapılmıştır.

Geliştirilen yazılımda genetik algoritma 2000 kez çalıştırılmıştır ve yaklaşık olarak 25 saniye (sn) gibi kısa bir sürede sonuca ulaşılmıştır. Yineleme sayısı için herhangi bir kısıtlama bulunmaması ve karar vericinin tercihinde olması sebebiyle mevcut sistemin araç rotasyonundan daha iyi bir sonuca ulaşacak bir yineleme sayısı belirlenmeye çalışılmıştır. Program daha fazla yineleme sayısı ile çalıştırıldığında, çözüm sonuçları hep aynı değer üzerinde yoğunlaşıp yeni bir sonuç elde edilemediği için ve çözüm süresi uzadığı için 2000 yinelemede program durdurulmaktadır.

İşletmenin bir günlük sipariş miktarı kullanılarak yapılan çalışmada, toplam kat edilen mesafeler ve araç rotalarının mesafe değerleri dağılımı karşılaştırılmıştır. Tablo 3.3. ve Tablo 3.4.'te görüldüğü üzere; mevcut sistemde 3 aracın toplam kat ettiği mesafe 236,54 km olarak hesaplanırken, genetik algoritma ile çözüm sonucunda bu mesafe 210,24 km olarak hesaplanmaktadır. Bu durumda, günlük olarak hesaplanan değerde 26,3 km'lik bir iyileşme sağlanmaktadır. Aylık olarak düşünüldüğünde, bir ay için yaklaşık olarak 790 km'lik bir iyileşme sağlanmıştır. Maliyet bazında bakıldığında, kullanılan araçlar şehir içinde gidilen her km'de ortalama 1,6 TL/lt yakıt harcamaktadır ve aylık olarak düşünüldüğünde işletme için 1.264 TL'lik bir yakıt tasarrufu ile maliyette azalma sağlanmaktadır. Bu çalışma sonucunda işletmeye ait 51 adet satış büfesi için tüm talepleri karşılayacak bir rotasyon oluşturularak araçlarının hem zamandan tasarruf etmesini hem de maliyetin düşürülmesini sağlayan bir çözüm sunulmuştur.

Araçların rotasyon dağılımına bakıldığında, mevcut sistemde 1. Aracın rotası 113,34 km, 2. Aracın rotası 65 km ve 3. Aracın rotası 58,2 km'dir. Araçlar arasında km bazında dengeli bir dağılım sağlanamamıştır. Diğer yandan genetik algoritma ile elde edilen çözümde 1. Aracın rotası 76,24 km, 2. Aracın rotası 64,7 km ve 3. Aracın rotası 69,3 km olarak hesaplanmaktadır. Genetik algoritma ile elde edilen çözümün araç rotasyonlarında km bazında kısmen dengeli bir dağılım sağlanmaktadır.

Çalışmada yer verilen işletme Eskişehir Halk Ekmek A.Ş.'dir ve ucuz, sağlıklı ve yüksek kalitede ekmek ve ekmek ürünleri üretimi ve dağıtımını yapmak üzere Eskişehir Büyükşehir Belediyesi bünyesinde kurulan bir organizasyondur. Hem her işletme gibi kâr amacı gütmemesinin yanında hem de Eskişehir halkının her kesimine uygun bir ürün fiyatlandırmasıyla sosyal hizmet projesi olarak kurulması nedeniyle her türlü maliyet kalemi işletme için önem taşımaktadır. İşletmenin iş tanımı genel olarak ekmek üretimi ve dağıtımından oluşmaktadır. Bugüne kadar Eskişehir Halk Ekmek İşletmesi'nde araç rotalarının oluşturulmasıyla ilgili herhangi bilimsel bir çalışma yapılmamıştır. Daha öncesinde ilgili çalışanların tecrübelerinden yararlanılarak bir rota planlaması yapılmıştır. Ekmeklerin dağıtımında araç rotalarının planlanması, işlemin önemli maliyet kalemlerinden biri olması sebebiyle çalışma bu alanda önem kazanmaktadır.

Çalışma kapsamında, genetik algoritma parametreleri önceden belirlenen değerlerle uygulanmıştır fakat parametre değerleri için kesin bir kısıtlama olmaması sebebiyle farklı değerler ile genetik algoritma çalıştırıldığında daha iyi sonuçlar elde etmek mümkündür.

Gelecekte işletmenin satış büfelerinin sayısının azalacağı veya artacağı düşünüldüğünde yazılım, satış büfesi eklenmesine ve çıkarılmasına olanak sağlayacak şekilde geliştirilebilir. Aynı zamanda, bir günlük sipariş miktarı verilerinden yararlanılarak oluşturulan rotalamanın değişen sipariş miktarları içinde uygulanabilir olması adına sipariş miktarlarının girileceği bir ara yüz tasarlanması kullanıcı yararına olacaktır.

Çalışmada ele alınan işletme, satış büfeleri dışında kurumlara da hizmet vermektedir. Bu nedenle geliştirilen bilgisayar sisteminin daha geniş boyutta kullanımı da mümkündür. Bunların yanında ve daha da önemlisi, geliştirilen sistemin benzer problemlerle karşılaşılan başka işletmeler tarafından da kullanılabilir esneklikte olduğu düşünülmektedir.

KAYNAKÇA

- Acar, E. (2014). Yatırım Problemlerinin Değerlendirilmesinde Evrimsel Algoritma Kullanımı Yap İşlet Devret Modeli Altyapı Projesi Üzerine Bir Uygulama. Doktora Tezi. Eskişehir: Anadolu Üniversitesi Sosyal Bilimler Enstitüsü.
- Ağalday, M.F., (2018). Genetik Algoritma ile Merkezi Sınavlarda Tek ve Çok Boyutlu Yakınlığa Göre En İyi Oturum Planının Oluşturulması. Yüksek Lisans Tezi. İstanbul: Fatih Sultan Mehmet Vakıf Üniversitesi Mühendislik ve Fen Bilimleri Enstitüsü.
- Aksaraylı, M., Pala, O., Cenger, A., Özlü, Y. ve Aksoy, M. A., (2017). Turizm Sektöründe Araç Rotalama Problemi ve Karar Destek Sistemi Uygulaması. Adnan Menderes Üniversitesi, Sosyal Bilimler Enstitüsü Dergisi, Cilt: 4, Sayı: 4, s. 203-213.
- Atlas, M., (2008). Çok Amaçlı Programlama Çözüm Tekniklerinin Sınıflandırılması. Anadolu Üniversitesi Sosyal Bilimler dergisi, Cilt:8, Sayı:1, s.47-68
- Bagley J. D., (1967). The Behavior of Adaptive Systems Which Emplly Genetic and Correlation Algorithms. PhD Thesis, University of Michigan
- Bayata, Ö., (2012). Genetik Algoritmaların Ders Çizelgeleme Probleminde Kullanımı ve Eğitim Kurumlarında Uygulanması. Yüksek Lisans Tezi. Ankara: Gazi Üniversitesi Fen Bilimleri Enstitüsü.
- Beasley, D., Bull, D.R., Martin, R.R., (1993). An Overview of Genetic Algorithms: Part 1, Fundamentals. University Computing, 15(2), 58-69.
- Bolat, B., Erol, K.O., İmrak, C.E., (2004). Genetic Algorithms in Engineering Applications and The Function of Operators. Sigma Journal of Engineering and Natural Sciences Mühendislik ve Fen Bilimleri Dergisi, 2004/4, 264-271.
- Cevre, U. (2008). Çoklu Gezgin Satıcı Probleminin Çözümü İçin Bir Eniyileme Kütüphanesinin Tasarımı ve Görsel Yazılım Geliştirme Ortamı ile Birlikte Gerçekleştirimi. Yüksek Lisans Tezi. İzmir: Ege Üniversitesi Fen Bilimleri Enstitüsü.
- Chen, T. and Chen, C., (1997). Improvements of Simple Genetic Algorithm in StructuralL Desing. Internationall Journal for Numerical Methods in Engineering, Vol. 40, 1323–1334.

- Cheng R., Gen, M. and Tusujimura, Y., (1999). A Tutorial Survey of Job-Shop Scheduling Problems Using Genetic Algorithms, Part II: Hybrid Genetic Search Strategies. *Computers & Industrial Engineering*, C.36, No:2.
- Christofides, N., (1976). The Vehicle Routing Problem. *Recherche Operationnelle*, 10(2), p.55-70.
- Çalışkan, F., Yüksel, H. Ve Dayık, M., (2016). Genetik Algoritmaların Tasarım Sürecinde Kullanılması. *SDU Teknik Bilimler Dergisi*, Cilt 6, Sayı:2, s. 21-27.
- Çeyrekoğlu, S., (2017). Araç Rotalama Problemine Genetik Algoritma Yaklaşımı ve Örnek Bir Uygulama. Yüksek Lisans Tezi. İstanbul: İstanbul Üniversitesi Fen Bilimleri Enstitüsü.
- Çolak, S., (2010). Genetik Algoritmalar Yardımı ile Gezgin Satıcı Probleminin Çözümü Üzerine Bir Uygulama. *Çukurova Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, Cilt 19, Sayı 3, s. 423-438.
- Dasgupta, D. ve Michalewicz, Z., (1997). *Evolutionary Algorithms In Engineering Applications*, Germany: Springer.
- Deep, K. and Mebrahtu, H., (2011). New Variations of Order Crossover for Travelling Salesman Problem. *International Journal of Combinatorial Optimization Problems and Informatics*, Vol. 2, No. 1, Jan-April, pp.2-13.
- Demircioğlu, M., (2009). Araç Rotalama Probleminin Sezgisel Bir Yaklaşım ile Çözümlemesi Üzerine Bir Uygulama. Doktora Tezi. Adana: Çukurova Üniversitesi Sosyal Bilimler Enstitüsü.
- Dreo, J., Siarry, P., Petrowski, A., Taillard, E., (2006). *Metaheuristics for Hard Optimization*. Germany: Springer-Verlag, 978-3-540-23022-9.
- Duman, C., (2007). Genetik Algoritma ile Tesis Yerleşimi Tasarımı ve Bir Uygulama. Yüksek Lisans Tezi. İstanbul: Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü.
- Düzakın, E. ve Demircioğlu, M., (2009). Araç Rotalama Problemleri ve Çözüm Yöntemleri. *Çukurova Üniversitesi İİBF Dergisi*, Cilt:13. Sayı:1, ss.68-87.
- Elmas, Ç., (2007). *Yapay Zeka Uygulamaları*. Ankara: Seçkin Yayıncılık
- Emel, G.G., ve Taşkın, Ç., (2002). Genetik Algoritmalar ve Uygulama Alanları. Bursa: Uludağ Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, Cilt XXI, Sayı 1, s. 129-152.
- Er, H., Çetin, M. ve İpekçi Çetin, E. (2005). Finanstada Evrimsel Algoritmik Yaklaşımlar: Genetik Algoritma Uygulamaları. *Akdeniz İ.İ.B.F. Dergisi* (10), 73-94.

- Er. H.R. (2013). Gezgın Satıcı Probleminin Hadoop Üzerinde Çalışan Paralel Genetik Algoritma ile Çözümü. Yüksek Lisans Tezi. İstanbul: İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü.
- Erol, V., (2006). Araç Rotalama Problemleri İçin Populasyon ve Komşuluk Tabanlı Metasezgisel Bir Algoritmanın Tasarımı Ve Uygulaması.Yüksek Lisans Tezi. İstanbul: Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü.
- Fıđlalı, A. Ve Engin, O. (2002). Genetik Algoritmalarla Akış Tipi Çizelgelemede Üreme Yöntemi Optimizasyonu. İTÜ Dergisi, s.1-6.
- Fisher, M. L. ve Jaikumar, R., (1981). A Generalized Assignment Heuristic for Vehicle Routing, Networks, 11, 109-124.
- Goldberg, D.E., (1989). Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading.
- Gomez, P. A. and Hougen, D. F., (2007). Initial Population for Genetic Algorithms: A Metric Approach. International Conference on Genetic and Evolutionary Methods. Las Vegas, Nevada, USA, 25-28 Haziran.
- Grefenstette, J.J., (1986). Optimization of Control Parameters for Genetic Algorithm. IEEE Transactions on Systems and Cybernetics, 16:1. 122-128.
- Gupta, S. M. ve Imtavanich, P. (2010). Evolutionary Computational Approach for Disassembly Sequencing in a Multiproduct Environment. Biomedical Soft Computing and Human Sciences. 15(1):71-76
- Güden, H., Vakvak, B., Özkan, B.E., Altıparmak F. Ve Dengiz, B., (2005). Genel Amaçlı Arama Algoritmaları İle Benzetim Eniyilemesi: En İyi Kanban Sayısının Bulunması. TMMOB Endüstri Mühendisliği Dergisi, C:16, S:1, s.2-15.
- Günay, N.S. (2013). Genetik Algoritma ile Araç Rotalama Problemlerinin Çözümü İçin Görsel Rotalama Yazılımı Geliştirme. Yüksek Lisans Tezi. İstanbul: İstanbul Üniversitesi Fen Bilimleri Enstitüsü.
- Haupt, R. L. and Haupt, S.E. (2004). Practical Genetic Algorithms. John Wiley & Sons Inc., Publication.
- İşçi, Ö. ve ;Korukođlu, S., (2003). Genetik Algoritma Yaklaşımı ve Yöneylem Araştırmasında Bir Uygulama. Manisa: Celal Bayar Üniversitesi Yönetim ve Ekonomi Dergisi, 10(2), s.191-208.

- Javidi M. M., Fard R. H. and Jampour M., (2015). Research in Random Parameters of Genetic Algorithm and Its Application on TSP and Optimization Problems. *Walailak J Sci & Tech* 2015; 12(1): 27-34.
- Kabadayı, N., (2007). Seri Tedarik Zincirinde Temel-Stok Seviyelerinin Simülasyon Temelli Genetik Algoritma ile Çözülmesi. Yüksek Lisans Tezi. İstanbul: İstanbul Üniversitesi Sosyal Bilimler Enstitüsü.
- Kahvecioğlu, A., (2003). Onarılabilir Elemanlara Önleyici Bakım Etkisi ve Optimizasyonu. Denizli: Bakım Teknolojileri Kongresi ve Sergisi, TMMOB Mak. Müh. Odası, s.343-354.
- Karaova, M., Smarkov, V. and Penev, S. (2005). Genetic Operators Crossover and Mutation in Solving The TSP Problem. *International Conference on Computer Systems and Technologies – CompSysTech*.
- Kaya, İ. (2012). Genetik Algoritmaları Optimal Güzergah Belirlenmesine Uygulanması. Yüksek Lisans Tezi. İstanbul: Haliç Üniversitesi Fen Bilimleri Enstitüsü.
- Keçeci, B., (2008). Önce Dağıt Sonra Topla Araç Rotalama problemi İçin Tamsayılı Karar Modelleri. Yüksek Lisans Tezi. Ankara: Başkent Üniversitesi Fen Bilimleri Enstitüsü.
- Kek, A.G.H., Cheu, R.L. ve Meng, Q., (2008). Distance-Constrained Capacitated Vehicle Routing Problems With Flexible Assignment of Start and End Depots. *Mathematical and Computer Modelling*, 47, 140-152.
- Keskintürk, T., Topuk, N. ve Özyeşil, O. (2015). Araç Rotalama Problemleri ile Çözüm Yöntemlerinin Sınıflandırılması ve Bir Uygulama. *İşletme Bilimi Dergisi*, Cilt:3, Sayı:2, s.77-107.
- Kırda, K., (2013). Evsel İlaç Atıklarının Toplanması Projesindeki Tersine Lojistik Sürecinin Modellenmesi İçin Genetik Algoritmaların Kullanılması. İzmir: Dokuz Eylül Üniversitesi Sosyal Bilimler Enstitüsü.
- Kiremitçi, B., Kiremitçi, S. Ve Kesnkintürk, T., (2014). Zaman Pencereli Çok Araçlı Dağıtım Toplamalı Rotalama Problemi İçin Gerçek Değerli Genetik Algoritma Yaklaşımı. *İstanbul Üniversitesi İşletme Fakültesi Dergisi*, Cilt:43, Sayı:2, s.391-403.
- Koza, J. R., (1998). *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. London: The MIT Press,0-262-11170-5.

- Kurt, M. ve Semetay, C. (2001). Genetik Algoritma ve Uygulama Alanları. Makine Mühendisleri Odası Dergisi, Ekim.
- Michalewicz, Z., (1992). Genetic Algorithms + Data Structures = Evolution Programs. Berlin: Springer-Verlag.
- Mitchell, M. and Taylor, C.E., (1999). Evolutionary Computation: An Overview. Annual Review of Ecological Systems. 30:593–616.
- Osman, İ.H. ve Kelly, J.P., (1996). Meta-Heuristics: Theory and Applications, Boston: Kluwer Academic Publishers.
- Önsel, Ş., (Tarihsiz). TP Sorunlarının Çözümü. (Çevrimiçi) <https://web.itu.edu.tr/kabak/dersler/EM302/pdf/TSPcozum.pdf>, 11 Haziran 2019.
- Özkan, B., (2008). Dinamik Gezgin Satıcı Probleminin Çözümü İçin Bir Eniyileme Kütüphanesinin Tasarımı ve Görsel Yazılım Geliştirme Ortamı ile Birlikte Gerçekleştirimi. Yüksek Lisans Tezi. İzmir: Ege Üniversitesi Fen Bilimleri Enstitüsü.
- Öztürk, A. (2011). Yöneylem Araştırması. Bursa: Ekin Yayınevi.
- Pan, L., (2015). Cutting Plane Method. The Chinese University of Hong Kong, Operations Research and Logistics Jan. 20.
- Paolucci, M., (2005). Vehicle Routing Problems. ICCS.
- Pereira, R., (2000). Genetic Algorithm Optimisation for Finance and Investment. (Çevrimiçi) <https://mpa.ub.uni-muenchen.de/8610/>, 26 Mayıs 2019.
- Pulat, M. ve Deveci Kocakoç, İ. (2016). Gezgin Satıcı Probleminin Genetik Algoritmalar Kullanarak Çözümünde Çaprazlama Operatörlerinin Örnek Olaylar Bazlı İncelenmesi. İzmir: Yöneylem Araştırması Endüstri Mühendisliği 36. Ulusal Kongresi.
- Pulat, M. ve Deveci Kocakoç, İ., (2017). Gezgin Satıcı Probleminin Genetik Algoritmalarla Çözümünde Başlangıç Popülasyonun Belirlenmesi, JOEEP Journal of Emerging Economies and Policy Vol.2(1), s. 95-123, July
- Razali, N. M. and Geraghty J. (2011). Genetic Algorithm Performance with Different Selection Strategies in Solving TSP. London, U.K.: Proceedings of the World Congress on Engineering 2011 Vol II WCE 2011, July 6 – 8.
- Reeves, C.R., (2006). “Genetic Algorithms”, Handbook of Metaheuristics, Ed. Fred W. Glover, Gary A. Kochenberger, 57. Bs., Springer Science & Business Media.

- Rothlauf, F., (2006). Representations for Genetic and Evolutionary Algorithms. Springer-Verlag, Netherlands,978-3-540-25059-3.
- Sawaka, M. (2001). Genetic Algorithms and Fuzzy Multiobjective Optimization, London: Kluwer Academic Publishers.
- Sivanandam, S.N. and Deepa, S.N., (2008). Introduction to Genetic Algorithms. Berlin: Springer Science& Business Media.
- Şahin, Y. ve Erođlu, A., (2014). Kapasite Kısıtlı Araç Rotalama Problemi İçin Metasezgisel Yöntemler: Bilimsel Yayın Taraması. Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, C.19, S.4, s.337-355.
- Şen, T., Cömert Ercan, S. ve Yazgan, H.R., (2015). Kapasite Kısıtlı Araç Rotalama Probleminin Çözümü İçin Yeni Bir Algoritma Geliştirilmesi: Bir Süpermarket Zincirinde Uygulanması. Sakarya Üniversitesi Fen Bilimleri Enstitüsü Dergisi, Cilt 19, Sayı:1, s. 83-88.
- Şen, Z., (2004). Genetik Algoritmalar ve Eniyileme Yöntemleri. İstanbul: Su Vakfı Yayınları.
- Tabak, Ö., (2008). Genetik Algoritma ile Kapasiteli Servis Güzergahı Belirlenmesi ve Bir Uygulama. Yüksek Lisans Tezi. Eskişehir: Anadolu Üniversitesi Fen Bilimleri Enstitüsü.
- Taşkın, Ç., ve Emel, G.G., (2009). Sayısal Yöntemlerde Genetik Algoritmalar, Bursa: Alfa Aktüel Yayınları.
- Toth, P. ve Vigo, D., (2002a). Discrete Applied Mathematics 123. Elsevier Science B.V., p. 487 – 512.
- Toth, P. ve Vigo, D., (2002b). The Vehicle Routing Problem. Society for Industrial and Applied Mathematics (SIAM).
- Tüfekçier, H., (2008). İki Amaçlı Açık Araç Rotalama Problemi İçin Bir Çözüm Yaklaşımı. Yüksek Lisans Tezi. Eskişehir: Osmangazi Üniversitesi Fen Bilimleri Enstitüsü.
- Ünal, H. T., (2018). Awacs Uçaklarında Ekip Çizelgeleme Probleminin Genetik Algoritmalar Yöntemiyle Çözümü. Yüksek Lisans Tezi. Konya: Selçuk Üniversitesi Fen Bilimleri Enstitüsü.
- Yücel, A., (2016). Mesafe Kısıtlı Çok Yönlü Kümelenmiş Açık Araç Rotalama Probleminin Genetik Algoritma ile Çözümü. Doktora Tezi. İstanbul: İstanbul Üniversitesi Sosyal bilimler Enstitüsü.

http-1: <http://www.eskisehirhalkekmek.com/>, 28 Haziran 2019

http-2: [https://drive.google.com/open?id=1QSWTXqzZLq-](https://drive.google.com/open?id=1QSWTXqzZLq-1cQNo3z13D58Pdml7twya&usp=sharing)

[1cQNo3z13D58Pdml7twya&usp=sharing](https://drive.google.com/open?id=1QSWTXqzZLq-1cQNo3z13D58Pdml7twya&usp=sharing), 28 Haziran 2019