

**T.C.**  
**İSTANBUL KÜLTÜR ÜNİVERSİTESİ**  
**LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

**ÇOKLU ÖĞRENCİ SERVİSLERİNİN ROTA OPTİMİZASYONU**

**YÜKSEK LİSANS TEZİ**

**Hasan ŞAHİN**

**1800005526**

**Anabilim Dalı: Bilgisayar Mühendisliği**  
**Programı: Bilgisayar Mühendisliği**

**Tez Danışmanı: Prof. Dr. Özgür Koray ŞAHİNGÖZ**

**ŞUBAT 2021**

**T.C.**  
**İSTANBUL KÜLTÜR ÜNİVERSİTESİ**  
**LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

**ÇOKLU ÖĞRENCİ SERVİSLERİNİN ROTA OPTİMİZASYONU**

**YÜKSEK LİSANS TEZİ**

**Hasan ŞAHİN**

**1800005526**

**Anabilim Dalı: Bilgisayar Mühendisliği**

**Programı: Bilgisayar Mühendisliği**

**Tez Danışmanı: Prof. Dr. Özgür Koray ŞAHİNGÖZ**

**Jüri Üyeleri: Doç. Dr. Akhan AKBULUT**

**Dr. Öğretim Üyesi Hakan AYDIN**

**ŞUBAT 2021**

## ÖNSÖZ

“ÇOKLU ÖĞRENCİ SERVİSLERİNİN ROTA OPTİMİZASYONU” adlı yüksek lisans tez çalışmam süresince emeklerini esirgemeyen süreç boyunca daima destek olan akademik yayınlar hazırlama konusunda bilgi ve tecrübesi ile yol gösteren saygıdeğer danışman Hocam Prof. Dr. Özgür Koray ŞAHİNGÖZ’e, çalışmam boyunca yardımlarını esirgemeyen arkadaşım Mehmet ÖZMEN’e teşekkürlerimi sunarım.



# İÇİNDEKİLER

ÖNSÖZ.....	iii
İÇİNDEKİLER.....	iv
ŞEKİL LİSTESİ.....	vi
TABLolar LİSTESİ.....	viii
KISALTMALAR .....	ix
ÖZET .....	x
ABSTRACT.....	xii
<b>1. GİRİŞ.....</b>	<b>1</b>
1.1. Problemin Tanımı.....	2
1.2. Literatüre Katkıları .....	3
1.3. Tezin Aşamaları.....	4
<b>2. LİTERATÜR ARAŞTIRMASI.....</b>	<b>5</b>
2.1. Araç Rotalama Problemi .....	5
2.2. Okul Servisi Rotalama Problemi.....	6
2.3. Gezgin Satıcı Problemi .....	7
2.4. Çoklu Gezgin Satıcı Problemi .....	9
<b>3. ROTALAMA PROBLEM YÖNTEMLERİ .....</b>	<b>10</b>
3.1. Araç Rotalama Problem Kısıtları .....	10
3.1.1. Araç Rotalama Problem Çeşitleri.....	11
3.1.1.1. Kısıtlı Araç Rotalama Problemi .....	11
3.1.1.2. Yol Durumlu Araç Rotalama Problemi .....	13
3.1.1.3. Dinamik / Statik Araç Rotalama Problemi .....	13
3.1.2. Araç Rotalama Problemi Çözüm Yöntemleri .....	13
3.1.2.1. Kesin Çözüm Yöntemleri.....	14
3.1.2.2. Sezgisel Yöntemler.....	14
3.1.2.3. Metasezgisel Yöntemler .....	14

3.2.	Okul Servisi Rotalama Problemi.....	18
3.2.1.	Okul Servisi Rotalama Problem Etkenleri .....	18
3.3.	Gezgin Satıcı Problemi .....	19
3.4.	Çoklu Gezgin Satıcı Problemi .....	20
<b>4.</b>	<b>ÖNERİLEN MODEL.....</b>	<b>21</b>
4.1.	Amaç .....	21
4.2.	Kromozom .....	24
4.3.	Fitness Fonksiyonu.....	24
4.4.	Başlatma .....	25
4.5.	Seçim.....	25
4.6.	Çaprazlama .....	25
4.7.	Mutasyon.....	26
4.8.	2-Opt Optimizasyon.....	27
4.9.	Fesih.....	27
4.10.	Goggle Maps Api.....	28
<b>5.</b>	<b>TEST SONUÇLARI VE DEĞERLENDİRMELER.....</b>	<b>30</b>
<b>6.</b>	<b>SONUÇLAR.....</b>	<b>36</b>
	<b>KAYNAKÇA.....</b>	<b>37</b>

## ŞEKİL LİSTESİ

Şekil 1.1- Akıllı Şehirlerde Ulaşım .....	1
Şekil 1.2- Gezgin Satıcı Problemi Rota Optimizasyonu Çözümü .....	4
Şekil 2.1- Araç Rotalama Problemi .....	5
Şekil 2.2- Tasarruf ve Süpürme Algoritmaları Sonucu Rota.....	7
Şekil 2.3- Genetik Algoritmalar Sonucu Oluşan Rotalar .....	8
Şekil 2.4- Optimizasyonun Sonucu Oluşan Rota Uzunluğu.....	9
Şekil 3.1- Üç Aracın Araç Rotalama Problem Optimizasyonu .....	10
Şekil 3.2- Kapasite Kısıtlı Araç Rotalama Problemi Çözümü.....	11
Şekil 3.3- Zaman Pencere Araç Rotalama Probleminin Çözümü .....	12
Şekil 3.4- Genetik Algoritmanın Akış Şeması .....	16
Şekil 3.5- Gezgin Satıcı Problemi .....	19
Şekil 3.6- İki Araçlı Gezgin Satıcı Problemi.....	20
Şekil 4.1- Uygulama MTSP Genetik Algoritma Akış Şeması.....	22
Şekil 4.2- Akış Diyagramı Sembollerinin Anlamları .....	23
Şekil 4.3- Kesme Noktası Kümelerine Dayalı Üç Parçalı Kromozom.....	24
Şekil 4.4- Çaprazlama İşlemi Sonucu Oluşan Kromozom Yapısı.....	26
Şekil 4.5- Mutasyon İşlemi Sonucu Oluşan Kromozom Yapısı.....	26
Şekil 4.6- 2-Opt Algoritması Konsepti.....	27
Şekil 4.7- Çalışmadaki Noktaların Uzaklık Matrisi Çıktısı.....	28
Şekil 5.1- Bir Aracın Toplam Mesafe İterasyon Grafiği .....	31
Şekil 5.2- Bir Araca Ait Noktaların Enlem ve Boylam Konumları.....	31
Şekil 5.3- Bir Araç İçin Sonuçların Google Maps Uygulaması İle Gösterilmesi.....	32
Şekil 5.4- İki Aracın Toplam Mesafe İterasyon Grafiği.....	32
Şekil 5.5- İki Araca Ait Noktaların Enlem ve Boylam Konumları .....	33
Şekil 5.6- İki Araç İçin Sonuçların Google Maps Uygulaması İle Gösterilmesi.....	33
Şekil 5.7- Üç Aracın Toplam Mesafe İterasyon Grafiği.....	34
Şekil 5.8- Üç Araca Ait Noktaların Enlem ve Boylam Konumları .....	35
Şekil 5.9- Üç Araç İçin Sonuçların Google Maps Uygulaması İle Gösterilmesi .....	35



## TABLÖLAR LİSTESİ

Tablo 2.1- Çalışma Sonucu Mevcut ve İyileştirilmiş Durumun Karşılaştırılması.....	6
Tablo 5.1- Test Platformu Özellikleri.....	30
Tablo 5.2- Genetik Algoritma Parametreleri.....	30



## KISALTMALAR

cij	: i Şehrinden j Şehrine Rota Maliyeti
P	: Permütasyon
C	: Maliyet Matrisi
ARP	: Araç Rotalama Problemi
AARP	: Asimetrik ARP
ÇDARP	: Çok Depolu ARP
DARP	: Dinamik ARP
DOSRP	: Dinamik Okul Servis Rotalama Problemi
DTARP	: Dağıtım - Toplamalı ARP
ERX	: Kenar Yeniden Birleştirme Operatörü
GA	: Genetik Algoritma
GDARP	: Geri Dönüşümlü ARP
GPS	: Küresel Konumlama Sistemi
KKO	: Karınca Kolonisi Optimizasyonu
KKARP	: Kapasite Kısıtlı ARP
MKARP	: Mesafe Kısıtlı ARP
OSRP	: Okul Servisi Rotalama Problemi
OX1	: Sipariş Çaprazlama Operatörü
PSO	: Parçacık Sürü Optimizasyonu
SA	: Simüle Tavlama
SARP	: Statik ARP
SYARP	: Simetrik Yollu ARP

TA : Tabu Arama  
TB : Tavlama Benzetimi  
TSP : Gezgin Satıcı Problemi  
MTSP : Çoklu Gezgin Satıcı Problemi  
YSA : Yapay Sinir Ağları  
ZPARP : Zaman Pencereli ARP



<b>Üniversite</b>	:	<b>T.C. İstanbul Kültür Üniversitesi</b>
<b>Enstitüsü</b>	:	<b>Lisansüstü Eğitim Enstitüsü</b>
<b>Anabilim Dalı</b>	:	<b>Bilgisayar Mühendisliği</b>
<b>Program</b>	:	<b>Bilgisayar Mühendisliği</b>
<b>Tez Danışmanı</b>	:	<b>Prof. Dr. Özgür Koray ŞAHİNGÖZ</b>
<b>Tez Türü ve Tarihi</b>	:	<b>Yüksek Lisans – Şubat 2021</b>

## ÖZET

### ÇOKLU ÖĞRENCİ SERVİSLERİNİN ROTA OPTİMİZASYONU

Büyük şehirlerin karmaşık yapısından kaynaklı çok sayıda sorun bulunmaktadır. Şehirlerdeki sorunların çözülmeye çalışılması sonucunda ortaya çıkan akıllı şehir kavramı, şehirlerin çeşitli alanlardaki sorunlarını çözebilir.

Şehirlerin çözülmeyi bekleyen sorunları arasında ulaşım sorunu ön plana çıkmaktadır. Ulaşım sorunları içerisinde temel bir sorun okul servisi sorunudur. Okul servisi, öğrencilerin ikamet ettikleri yerden alınarak eğitim gördükleri okula en uygun şekilde ulaşmalarını ve evlerine geri bırakılmalarını sağlayacak optimizasyon yolunun belirlenmesi sorunudur. Araç Rotalama Probleminin (ARP) alt türü olan Okul Servis Rotalama Problemi (OSRP) durakların araçlara dağıtımı ve araçların izleyecekleri yolun belirlenmesi aşamalarından oluşur. OSRP, bir servis aracının en uygun sürede, en kısa rotayı izleyerek duraklardan öğrencileri toplayıp okula götürmesini ve okulda toplanan öğrencileri evlerine bırakmasını amaçlar.

Bu projede OSRP'nin optimizasyonu gerçekleştirilmeye çalışılmıştır. Bir okula hizmet veren bir firmanın birden fazla servisi için servis rotası sorunu genetik algoritmalar kullanılarak çözülmeye çalışılmıştır. Şirketin servis rotasını ve servis sayısını optimize ederek elde edilen kazancı maksimize etmek ve maliyetleri düşürmek için yönlendirme yöntemleri kullanılmıştır.

**Anahtar Kelimeler:** Akıllı Şehirler, Gerçek Zamanlı Okul Servisi Rotalama Problemi, Optimizasyon, Yapay Zeka Teknikleri.

**University** : T.C. İstanbul Kültür University  
**Institute** : Institute of Graduate Studies  
**Department** : Computer Engineering  
**Program** : Computer Engineering  
**Thesis Advisor** : Prof. Dr. Özgür Koray ŞAHİNGÖZ  
**Degree Awarded And Date** : MA – February 2021

## **ABSTRACT**

### **ROUTE OPTIMIZATION OF MULTIPLE STUDENT SERVICES**

There are many problems arising from the complex structure of large cities. The smart city concept, which emerged as a result of trying to solve the problems in cities, can solve the problems of cities in various areas.

The transportation problem stands out among the problems of cities waiting to be solved. One of the main problems in transportation is the school bus problem. School bus is the problem of determining the optimization way that will enable students to reach the school where they study in the most appropriate way and leave them back to their homes. The School Shuttle Routing Problem (OSRP), which is a sub-type of Vehicle Routing Problem (ARP), consists of the stages of distribution of stops to vehicles and determining the path the vehicles will follow. OSRP aims that a shuttle bus takes the shortest route, in the most appropriate time, to pick up students from the stops and take them to the school and leave the students gathered at the school to their homes.

In this project, OSRP has been optimized. For more than one service of a company serving a school, the service route problem has been tried to be solved by using genetic algorithms. Routing methods have been used to maximize earnings and reduce costs by optimizing the company's service route and number of services.

**Keywords:** Smart Cities, Real Time School Bus Routing Problem, Optimization, Artificial Intelligence Techniques.

# 1. GİRİŞ

Akıllı şehir kavramı ile yeni teknolojiler kullanılarak hayata geçirilen daha sürdürülebilir ve yaşanabilir şehirler, paydaşlar arası işbirliği ve yenilikçi yaklaşımlar, uzmanlık ve verilere dayalı, gelecekteki sorunları ve ihtiyaçları öngörerek hayata değer katan çözümler düşünülür. Güvenlik, iletişim teknolojileri, yönetim, ekonomi, alan yönetimi, binalar, sağlık, enerji, bilgi teknolojileri, bilgi güvenliği, altyapı, coğrafi bilgi sistemleri, afet ve acil durum yönetimi, çevre ve ulaşım sistemlerini akıllı hale getirilir.

Akıllı bir şehirde ulaşım, belediyelerin vatandaşlara daha iyi hizmet sunmasına, güvenliği ve emniyeti daha iyi yönetmesine ve maliyetleri düşürmesine katkıda bulunur. Yolcuların sıkıcı ve kalabalık kentsel alanlarda seyahat etme biçiminde büyük bir fark yaratabilir ve bu sorunların üstesinden gelmeye yardımcı olabilir. Akıllı şehir ulaşımı, büyük bir şehirde iki nokta arasında daha güvenli, daha akıllı ve daha hızlı seyahate erişim sağlamaya yardımcı olur. Şehir yönetimi için daha zengin bilgi ve trafik akışları üzerinde daha fazla kontrol sağlar. Elektronik cihazlar kablosuz ve internet gibi teknolojileri kullanır. Akıllı şehirlerde ilk olarak akıllı ulaşım altyapısı önerilmektedir.



Şekil 1.1- Akıllı Şehirlerde Ulaşım

Günümüz küresel piyasa ortamındaki rekabet, kısa yaşam süresine sahip ürünler ve müşterilerin artan beklentileri, üreticileri dağıtım sistemlerine önem vermeye ve yatırım yapmaya zorlamıştır. İletişimdeki ve ulaşım teknolojilerindeki değişim lojistik yönetiminin gelişimine neden olmuştur. Teknolojik alanda yaşanan gelişmelere de bağlı olarak yolcu taşımacılığı zaman, yakıt ve mesafe

etkenleri göz önünde bulundurularak alternatif güzergâh seçenekleri ile yapılabilir duruma gelmiştir. Nüfus artışındaki önlenemez gidişata bağlı olarak tüketim ihtiyaçlarının artması ile ulaşım, taşımacılık ve dağıtım sistemleri daha fazla kullanılarak önemini artırmaktadır. Taşımacılık ve ulaşım sistemlerindeki bu yoğunluk trafik sıkışıklığı, çevre kirliliği, zamanı etkin kullanamama gibi sorunlara yol açmaktadır. Gelineen noktada firmalarının başarısı lojistik sisteminde elde ettiği başarıya bağlı hale gelmiştir. Özellikle müşteri beklentilerindeki etkenler firmaları dağıtım sistemlerine yatırım yapmaya zorlamıştır. İletişim ve ulaşım teknolojilerindeki değişim lojistik yönetiminin devamlı gelişimine katkıda bulunmuştur. Ulaşımında kaybedilen sürelerin artmasına birden fazla faktör etki etmektedir. Trafik sıkışıklığı, yol durumunun uygun olmaması, hava koşullarında yaşanan olumsuzluklar, plansız rota tercihleri bu faktörlerden bazılarıdır. Ulaşımında harcanan zaman yakıt israfına neden olduğu gibi hava kirliliğine etki yaparak ciddi sağlık sorunlarına da neden olmaktadır. Teknoloji gelişimi ile ulaşımında yaşanan olumsuzlukları giderebilmek amacıyla yeni kavramlar ortaya çıkmıştır. Bu kavramlardan ve lojistik sistemi temel problemlerinden biri araç rotalama problemi (ARP)'dir [1].

ARP şirket depolarında bulunan ürünlerin müşterilere ulaşım araçları ile en uygun şekilde dağıtılması ve ürünlerin geri toplanması problemidir. Çeşitli sayıda depoya sahip müşterileri veya şehirleri olan firmanın, ürün dağıtım süresi boyunca seyahat mesafesini azaltmayı hedefleyen bir problem çeşididir. Problemin yol uzunluğu, araç kapasitesi, belirli zaman bölümünde servis yapılması gibi kısıtları bulunmaktadır. Sabit bir dağıtım noktasından çeşitli dağılmış noktalara, en az maliyetli rotaları hesaplama problemidir. Rotalar, her noktanın bir araç tarafından bir kez ziyaret edildiği, tüm rotaların dağıtım yerinden başlayarak dağıtım yerinde bittiği ve rotadaki tüm noktaların rotayı yönetmek için belirlenen araç kapasitesini geçmediği şekilde hesaplanmalıdır [2].

## **1.1. Problemin Tanımı**

ARP alt türü olan okul servisi rotalama problemi (OSRP) öğrencilerin ikametgâhta buldukları yerler ile eğitim gördükleri okul arasındaki mesafelerin yaşadığımız dönemde artması sonucu oluşan sorunları çözümlenmeyi hedefler. Bu problem okul servislerinin kullanım ihtiyacının artmasına neden olmaktadır. Dolayısıyla öğrenci servisleri öğrencilerin okullara ulaşım için velilerin tercihleri arasında ilk sıralarda yer almaktadır. Öğrenci taşımacılığı hizmeti sağlayan servis firmaları mevcut servis araçları ile daha fazla okula hizmet sağlamayı amaçlamaktadırlar. Servis hizmeti sunan firmalar araç dolaşım mesafesini en aza indirerek maliyetleri ve öğrencilerin servis içerisinde geçirdiği süreyi azaltmayı hedeflemektedirler. OSRP öğrencilerin ikametgâhlarından alınarak eğitim gördükleri okula, eğitim gördükleri okuldan alınarak ikametgâhlarına bırakılması sorununa çözüm üretmeye çalışır.

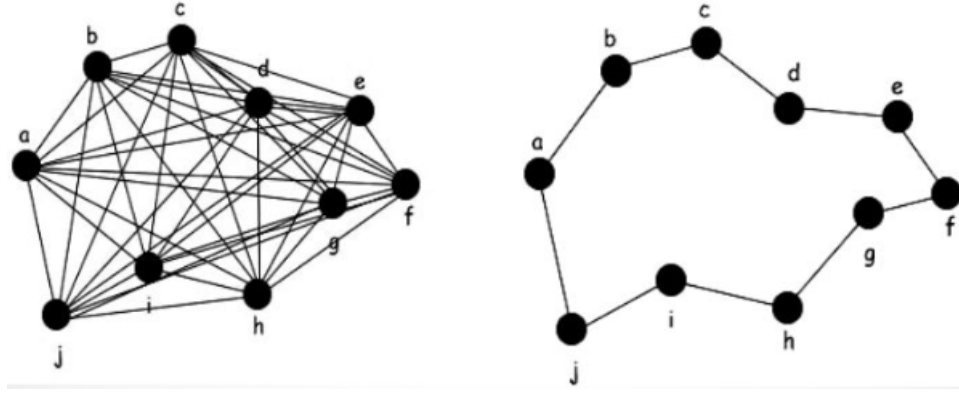
## 1.2. Literatüre Katkıları

Yapılan literatür incelemelerinde öğrencilerin ikametgahlarına yürüme mesafelerindeki duraklardan alındığı, kapasite ve zaman ölçütü olan, bir serviste farklı okullardan öğrencilere hizmet sağlanan, serviste öğrencilerin bulunma süresinin hesaplandığı çalışma ve uygulamalar ile karşılaşılmıştır. Araç rotalama problemlerinin çözümünde en kısa ve en uygun rotanın bulunması için karınca kolonisi optimizasyonu, tabu arama, tavla benzetimi, genetik algoritma, parçacık sürü optimizasyonu, tasarruf algoritması gibi yapay zekâ teknikleri kullanılarak metasezgisel yaklaşımlar ile daha başarılı sonuçlara ulaşılmıştır. Bu yaklaşımların esenlik kaynağı doğal ortamda bulunan hayvanların ihtiyaçlarını içgüdüsel özelliklerle en kısa ve en uygun yolları kullanarak bulabilme kabiliyetleridir. Bu canlıların sezgisel yetenekleri sayesinde metasezgisel yöntemler kullanarak insanlar en kısa yol sorununu çözmeye çalışmışlardır.

Gezgin satıcı problem (TSP) tahminen en çok bilinen ilişkisel en uygun şekle sokma problemidir. Gezgin satıcının kendi şehirden başlayıp, tanımlanmış diğer şehirleri ziyaret edip, tekrar kendi şehrine dönmesi için gereken en kısa yolu bulma problemidir. Alınan yol, şehirlere uğrama sırasına göre değişmektedir. Böylece gezgin satıcı problemi en uygun şehir sıralamasını bulma problemi şeklinde de tanımlanabilir. Gezgin satıcı problemi amacı, satış elemanının verilen yerleri sadece bir kez ziyaret etmesi ve sonunda başlangıç konumuna geri dönmesi şartıyla, satış elemanının minimum maliyetle rotasıdır. Çözümü için en uygun yaklaşımın matematiksel programlama temelli olan problemlerden biridir.

TSP çoklu araç ve tek depo için geliştirilmiş hali çoklu gezgin satıcı problemi (MTSP) olarak nitelendirilir. MTSP bir şirketin farklı şehirlerde yaşayan birden fazla satış elemanı olduğunu ve şirketin şehirleri belirli bir miktardan daha az olmamak üzere ziyaret etmeyi planladığı durumlarda problem çözüm yöntemi olarak kullanılmaktadır. Birden çok gezici satış elemanı sorunu yani çoklu gezgin satıcı problemi tek satış görevlisinin durumunu birkaç satış görevlisini içeren bir soruna dönüştürüp gereksinimleri karşılamak için bazı ek koşullar ekler. MTSP de amaç bir dizi şehir verildiğinde, bir depo m satış görevlileri bulunur ve bir maliyet ölçütü ile her bir satış elemanı için, toplam tur maliyetini en aza indirecek ve her şehri yalnızca bir satıcı tarafından tam olarak bir kez ziyaret edilecek şekilde bir tur belirlemeyi hedefler.

TSP, bir satıcının kendi şehirden başlayarak tüm şehirleri  $((n-1)$  şehir) dolaşarak, tekrar kendi şehrine dönüş mesafesini en aza indirmeyi hedefler. ARP bilinen yerlerde  $n$  müşterinin taleplerini, bir depoda bulunan  $k$  araç ile bir dağıtım rotası kullanılarak çözülmesi problemidir. Dağıtım yolu belirli bir noktadan başlamalı ve müşterilerin dolaşarak depoya geri dönülmesinden oluşur. ARP'nin amacı, her bir araç için dağıtım rotalarının toplam mesafeni en aza indirme hedeflenmektedir [3].



Şekil 1.2- Gezgin Satıcı Problemi Rota Optimizasyonu Çözümü

Çalışmada gezgin satıcı problemi, araç rotalama problemi ve araç rotalama problemi alt türü olarak nitelendirilen okul servisi rotalama problemi açıklanmıştır. Yapılan tez çalışması ile dinamik okul servis rotalama problemi (DOSRP) metasezgisel yöntemler kullanılarak gerçek zamanlı optimize edebilmek amaçlanmıştır. Servis hizmeti sağlayan firmanın araç kapasiteleri, hizmetin verildiği okul ve okuldaki öğrenci bilgileri kullanılarak servis araçları için en uygun rotaların gerçek zamanlı dinamik olarak üretilmesi sağlanmıştır. Öğrencilerin harita konumlarına göre öğrenciler servis araç kapasiteleri değerlendirilerek en uygun şekilde araçlara atanmıştır. Yapay zeka ve metasezgisel yöntemler ile algoritmalar geliştirilerek en uygun rotaların tespiti amaçlanmıştır. Firmanın taşıt bilgileriyle hizmet sağladığı okula ait öğrenci bilgileri değerlendirilerek servis rotalarının optimizasyonu yapılmaya çalışılmıştır. Problemin çözümü için servis firmasında her öğrencinin evden alınması ve okul saatlerine müdahale edilemediği varsayılmıştır. Küresel konumlama sistemi (GPS) konumları kullanılarak farklı öğrenciler gruplandırılmış, gruplar yoluyla servis sayısını ve toplam mesafeyi azaltmak amaçlanmıştır. Metasezgisel ve yapay zeka yöntemleri sayesinde servis araçları için en uygun rotanın dinamik olarak optimize edilmeye çalışılmıştır.

### 1.3. Tez Aşamaları

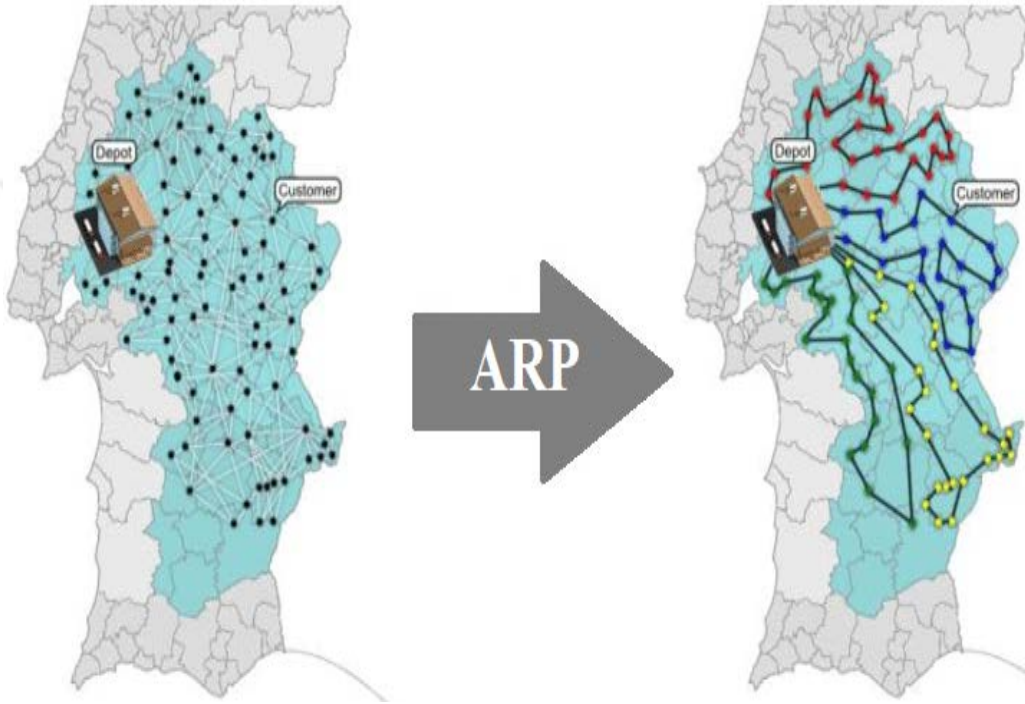
Bu tez çalışması 5 kısımdan oluşmaktadır.

- İlk kısımda, problemin tanımı aktarılmıştır. Sonuçlandırılan çalışmanın amacı ve önemi anlatılmıştır. Çalışmanın akademik anlamda getirileri vurgulanmıştır.
- İkinci kısımda, araç rotalama problemi, okul servisi rotalama problemi, gezgin satıcı problemi ve gezgin satıcılar problemleri ile alakalı akademik araştırmalar aktarılmıştır.
- Üçüncü kısımda, rotalama işlemini en iyileme işleminde kullanılan yöntemler, bu yöntemlerin alt başlıkları ve bunların nasıl uygulandığından söz edilmiştir.
- Dördüncü kısımda, yapılan çalışma önerilen yöntemle ilgili detaylardan, bileşenlerden, yöntemin akış diyagramından detaylı bir şekilde bahsedilmiştir.
- Beşinci ve son kısımda testler sonucu edilen sonuçlar analiz edilerek aktarılmıştır.

## 2. LİTERATÜR ARAŞTIRMASI

Optimizasyon, yapay zekâ teknikleri, metasezgisel ve sezgisel yöntemler, araç rotalama problemi, öğrenci servis rotalama problemi, dinamik öğrenci servis rotalama problemi, gezgin satıcı problemi ve çoklu gezgin satıcılar problemleri konuları ile alakalı yapılan önceki çalışmalar sonucu elde edilen verilere dayanılarak edinilen bilgiler aşağıda özetlenmiştir.

### 2.1. Araç Rotalama Problemi



Şekil 2.1- Araç Rotalama Problemi

ARP, ürünleri veya yolcuları almak ya da teslim etmek için tekraren duraklama yapması gereken araç filolarının etkin kullanılması problemidir [4]. Rota hesaplamalarında minimizasyon işlemleri için izlenen bir yöntemdir. ARP ilk olarak Dantzig ve Ramser [5] tarafından 1959 yılında çalışılmıştır. Clarke ve Wright [6] tarafından 1964 yılında yöntem geliştirmiş ve klasik tasarruf metodu önerilmiştir. ARP'nin değişik çeşitlerine çözüm bulmak için farklı model ve algoritmalar önerilmiştir. Uygulama alanının çokluğu ve problemin farklı olması nedeniyle ARP fazla sayıda araştırmacının ilgisini çekmiştir. Eryavuz ve Gencer [7], Balıkesir Ordudonatım Okulu personel servis araçlarının taşımacılık hizmeti yaparken izledikleri rotalara tasarruf algoritmalarını uygulayarak mevcut rota durumlarına göre %28'lik bir iyileştirme sağlandığı sonucuna varmışlardır. Bu durum sonucuna göre araç sayıları ve hizmet maliyetlerinin ciddi oranda azaltılabileceğini saptanmıştır. Elde edilen sonuçlar aşağıda gösterilmiştir.

Tablo 2.1- Çalışma Sonucu Mevcut ve İyileştirilmiş Durumun Karşılaştırılması

	Rota No	Rota Adı	Rota Uzunluğu	Durak Sayısı	Yolcu Sayısı
Mevcut Durum	1	O	96	9	51
	2	M	182	16	60
	3	C	150	12	42
	4	E	166	20	67
	TOPLAM			594	57
İyileştirilmiş Durum	1	R1	153	18	67
	2	R2	83	14	66
	3	R3	149	21	67
	4	R4	40	4	20
	TOPLAM			425	57

Çetin [8], bir akaryakıt dağıtım sistemi için ARP'yi etkin şekilde kullanarak dağıtım maliyetlerinin optimizasyonunu sağlamaya çalışmıştır. Problemin çözümü için başlangıç rotalarını tasarruf algoritması kullanarak oluşturduktan sonra bu rotaların optimizasyonu için genetik algoritma (GA) temelli tabu arama (TA) algoritması geliştirmiştir. Geliştirilen algoritmayı test verileri üzerinde uyguladığında yöntemin makul seviyede kabul edilebilir sonuçlar verdiği sonucuna ulaşmıştır. Güvez ve çalışma arkadaşları [9], Kırıkkale ili tıbbi atık toplama işlemlerinde kullanılan rotalar üzerinde ARP'yi kullanarak rota optimizasyonu elde etmişlerdir. Ulaşılan durum anlık rotalara göre yıllık bazda %20,63 iyileştirilebileceği sonucu ortaya çıkmıştır. Çalışkan [10], sezgisel yöntemler ve kümeleme teknikleri ile birlikte ARP'ne çözüm sağlamayı amaçlamıştır. Birden fazla deposu bulunan bir dağıtım firmasının depolarının müşterilere atanması için k ortalama kümeleme tekniğinden yararlanır.

## 2.2. Okul Servisi Rotalama Problemi

OSRP toplam maliyeti minimize ederek, öğrencilerin belirli toplama noktalarından alınarak okula bırakılması veya öğrencilerin okuldan alınarak belirli dağıtım noktalarına bırakılması problemidir [11]. Yapılan araştırmalara bakıldığında OSRP çözümüne yönelik çalışmaların geniş oranda rota mesafesi iyileştirmesine yönelik olduğu saptanmıştır. İlk olarak 1969 yılında Newton ve Thomas [12], tarafından önerilmiştir. Park ve Kim [13], okul servis aracının farklı noktalardan öğrencileri aldığı, servis aracı kapasitesi ve öğrencilerin servis içerisindeki zamanı değerlendirerek okul zaman aralığı faktörlerini de göz önünde bulundurup okullara öğrencileri ulaştırmayı hedeflemişlerdir. Sghaier ve çalışma arkadaşları [14], OSRP'nin çözümü için sağlamış oldukları GA ile okul servis hizmetlerinde %17,87 oranında bir maliyet kazancı elde edilebileceği sonucuna

ulaşmışlardır. Belirli noktalardan alınan öğrencilerin konumları saptanırken ani değişimler varsayım dışı tutulmuştur. Li ve Fu [15], Hong Kong'daki bir anaokulunda servis rotalarını iyileştirmek için zaman kısıtı ve araç kapasitelerini de değerlendirerek sezgisel bir yöntemle optimizasyon geliştirmiştir. Optimizasyon sonucu oluşan yeni rota ile eski rota planlamasına göre %29'luk bir optimizasyon başarısı gerçekleştirilmiştir. Martinez ve Viegas [16], Lizbon kentinde okul servisi hizmeti konusunda OSRP'nin çözümüne iki basamaklı bir çözüm yöntemi ile yaklaşmışlardır. İlk basamakta öğrencilerin koordinat konumlarını kullanarak, en uygun yöntem ile servis hizmeti sağlayan araçlara yerleşimleri sağlanmıştır, bir sonraki basamakta ise yerleşim sonucuna göre oluşan servisler için en iyi rota planlamasını yapılmıştır. Demiral [17], OSRP üzerine yapmış olduğu çalışma ile Isparta ilinde bir okulun servis hizmetine ait rotaların iyileştirilmesi maksadıyla rotaların optimizasyonunu gerçekleştirmiştir. Yapmış olduğu çözüm sayesinde servis araçları için orta noktalar belirlenmiş ve bu noktalara servis araçları ataması yapılarak orta noktada öğrencilerin birleştirilmesi sağlanmıştır. Orta noktaya gelme imkânı olmayan öğrenciler için servis güzergâhı kısa mesafede sürücü tarafından belirlenmek üzere sürücü alternatifine bırakılmıştır. Öğrencilerin servis araçlarına yerleştirilme durumları, bulunmuş oldukları duraklar göz önünde bulundurularak problemin çözümü için rota optimizasyonu yapılmıştır. Konuyla ilgili çalışma sonucu ortaya çıkan rota aşağıda belirtilmiştir.

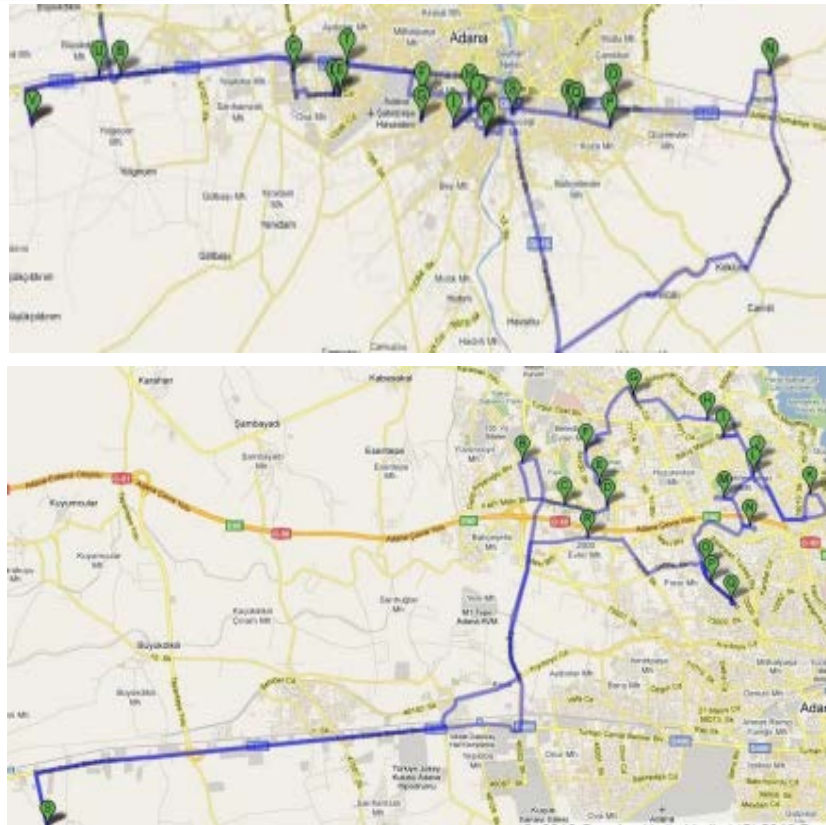


Şekil 2.2- Tasarruf ve Süpürme Algoritmaları Sonucu Rota

### 2.3. Gezgin Satıcı Problemi

Gezgin satıcı probleminin (TSP) kolay formülasyonu, zor çözümü ve çok sayıda uygulama alanına hitap ediyor olması literatürde gezgin satıcı probleminin çok sayıda incelemesinin oluşmasına yol açmıştır. Euler tarafından Königsberg köprüleri üzerinde tasarlanan bir problemin çözümü için TSP tanımlanmıştır. Königsberg köprüleri problemi, eski doğu Prusya topraklarında kalan Königsberg kentinin halkı tarafından, Pregel nehri üzerindeki yedi köprüden geçiş yapmaya dair bir oyundur [18]. Karl Menger çözüme giden tüm yolları deneyerek, optimum sonuca ulaşma

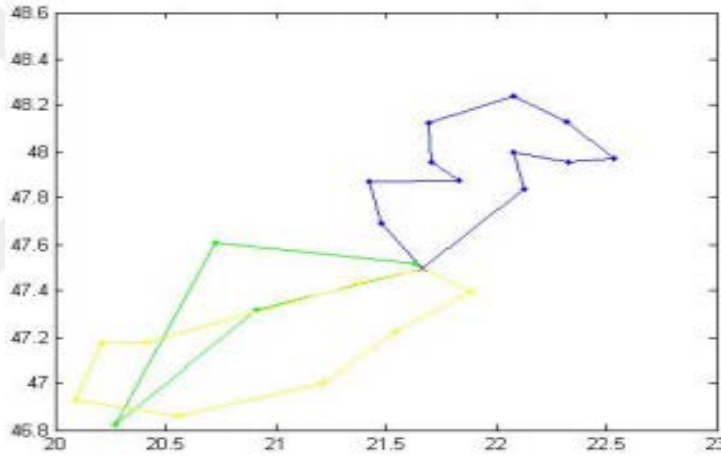
stratejisini ve en yakın komşu algoritmasının optimum sonucu garanti etmediğini belirtmiştir. Bazen en yakın komşuyu seçme stratejisinin en kötü sonuca getireceğini savunmuştur [19]. Dantzig ve arkadaşları doğrusal programlamayı kullanarak kesme düzlemi metodunu geliştirmişler ve bu yöntemle 49 şehirli bir TSP'yi çözmüşlerdir [5]. Christofides TSP'nin en iyi çözümü için önerilen minimum örten ağaç ve atama problemine dayalı alt sınır hesaplamalarına ek olarak, yeni bir alt sınır algoritması önermiştir. 14 test problemi için bu yeni alt sınır algoritması uygulanmış ve simetrik durumda en iyi değer %4,7 asimetrik durumda ise en iyi değer sadece %3,8 altında alt sınır değerleri elde edilmiştir [20]. Karp tarafından bölümlenme algoritması sunulmuştur. Bu algoritmada şehirler küçük gruplara bölünür, her grup için en iyi tur bulunur ve turlar birleştirilir [21]. Crowder ve Padberg çalışmalarında tanımladıkları kesme düzlemi algoritmasını geliştirerek 318 şehirli bir problemi çözmüşler daha sonra dal-kesme algoritması ile 532 şehirli bir problemi çözmüşlerdir [22]. Alaykırın ve Engin tarafından karınca kolonileri metasezgiseli kullanılarak TSP üzerine bir uygulama gerçekleştirmişlerdir. Karınca kolonileri metasezgiseli için uygun parameter değerleri belirlenip TSP çözümü gerçekleştirilmiştir [23]. Çolak genetik algoritmalarından faydalanılarak Adana ili gıda sektöründe faaliyet gösteren bir firma için rota planlamasına yönelik bir uygulama yazılımı gerçekleştirilmiştir. Yazılımda optimum çözüm bulmak için genetik ve karınca kolonisi algoritmalarından faydalanılmıştır [24]. Edilen rota sonuçları aşağıdaki gibidir.



Şekil 2.3- Genetik Algoritmalar Sonucu Oluşan Rotalar

## 2.4. Çoklu Gezgin Satıcı Problemi

MTSP,  $n$  şehre  $m$  satış görevlisinin atanmasını içerir ve her şehir, minimum bir toplam maliyet gerektirirken bir satıcı tarafından ziyaret edilmelidir. NP-zor bir problem olarak, MTSP araştırmacılar tarafından değerlendirilmiştir. Çoklu seyahat eden satıcı problemi yani çoklu gezgin satıcı problemi, her şehri alt turlar olmadan tam olarak bir kez ziyaret eden gezgin satıcı probleminin bir uzantısıdır [25]. Zhou ve Li mTSP problemi değiştirilmiş bir GA ile çözmüşlerdir. İlk popülasyonu oluşturmak için açgözlü bir strateji uygulanır ve mutasyon operatörü, kaliteli komşu çözümleri hızlı bir şekilde belirlemeye izin veren ve algoritmanın yakınsamasını hızlandıran yerel arama stratejisi 2 – Opt ile birleştirilir [26]. Chen ve arkadaşları mTSP'nin çözümünü temsil eden iki parçalı kromozom adı verilen bir kodlama tekniği kapsamında yeni mutasyon ve rekombinasyon operatörlerinin önerildiği bir genetik algoritma önermektedir. Bu kodlama, seyahat eden her satıcı tarafından ziyaret edilmesi gereken şehirlerin sırasını ve sayısını dikkate alır [27].



Şekil 2.4- Optimizasyonun Sonucu Oluşan Rota Uzunluğu

İki aşamalı hibrit algoritma Yousefikhoshbakh ve Seidighpour tarafından sunulmuştur. İlk aşamada mTSP bulunan çözümü iyileştirmek için yerel arama stratejisi 3-Opt ile birlikte tarama algoritmasıyla uygulanmış ve ikinci aşamada karınca kolonisi optimizasyonu ile çözülmüştür [28]. Al-Khateeb, parçacık sürü optimizasyonu (PSO) ile çoklu seyahat eden satıcı problemini çözümlendi. Çalışmasında seçilen birçok değer bu faktörler için test edildiği en iyi ivme faktörünü bulmaya odaklandı. Elde edilen sonuçlar, faktörlerin probleme bağlı olduğunu göstermiştir [29]. Xu ve arkadaşları çoklu seyahat eden satıcı problemini çözmek için hibrit bir genetik algoritma (GA) ve simüle tavlama (SA) kullandı. Genel genetik algoritmadan daha iyi bir global optimal çözüme ulaşmak için yerel arama yeteneğini geliştirdi [30]. Arthur E. ve arkadaşları iki parçalı kromozomu kullanarak MTSP'yi modellemişlerdir. İki parçalı kromozomun daha küçük çözüm alanı ile daha iyi uygunluk değerlerine sahip çözümler ürettiği belirtilmiştir. Mesafe en aza indirilerek, satış eleman sayısı arttığında iki parçalı kromozom üstünlüğü ortaya çıkmıştır [31].

### 3. ROTALAMA PROBLEM YÖNTEMLERİ

#### 3.1. Araç Rotalama Problem Kısıtları

Düzakın ve Demircioğlu [32] ARP kısıtları üç başlıkta incelenebilir:

1) Araçlarla

- Araç kapasite sınırlaması
- Toplam zaman sınırlaması
- Sürücü çalışma saatleri yasal sınırlaması

2) Müşterilerle

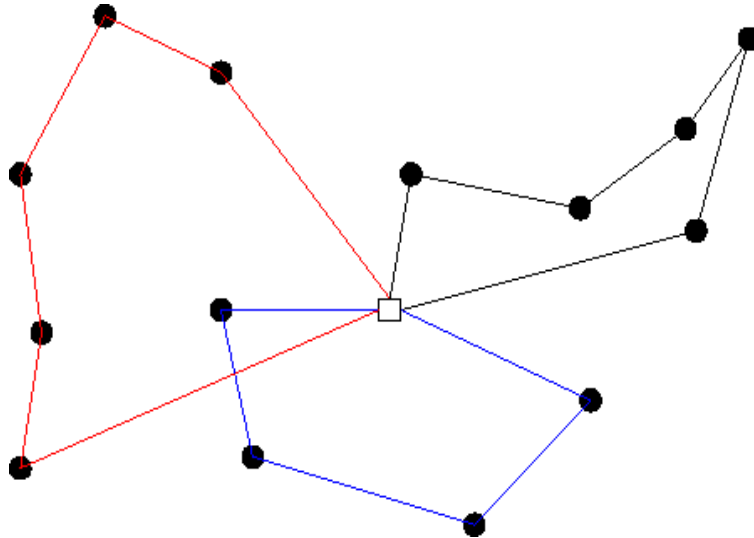
- Müşterinin bir tür ürün talep etmesi veya belirli çeşitte ürün dağıtılması;
- Dağıtımın belirli zaman diliminde yapılacak olması

3) Diğer

- Aynı aracın tekrarlı olarak aynı günde depoya dönerek birden fazla tur yapması
- Bir turun bir günden daha fazla süre alması
- Çok sayıda depo olması

ARP dağıtım rotalarında ise aşağıda belirtilen özellikler bulunmalıdır:

- Müşteri talebini karşılamalıdır.
- Müşteriler bir rotada olmalıdır.
- Bir rotadaki müşteri talepleri, aracın kapasitesinden düşük olmalıdır.
- Her rota, depodan başlayıp depoda bitmelidir.
- Bir rotadaki alınan mesafe, daha önceden belirlenen uzaklığı aşmamalıdır.
- Bazı ARP çeşitlerinde m araç sayısı sabit, bazılarında değişkenlik gösterir.



Şekil 3.1- Üç Aracın Araç Rotalama Problem Optimizasyonu

### 3.1.1. Araç Rotalama Problemi Çeşitleri

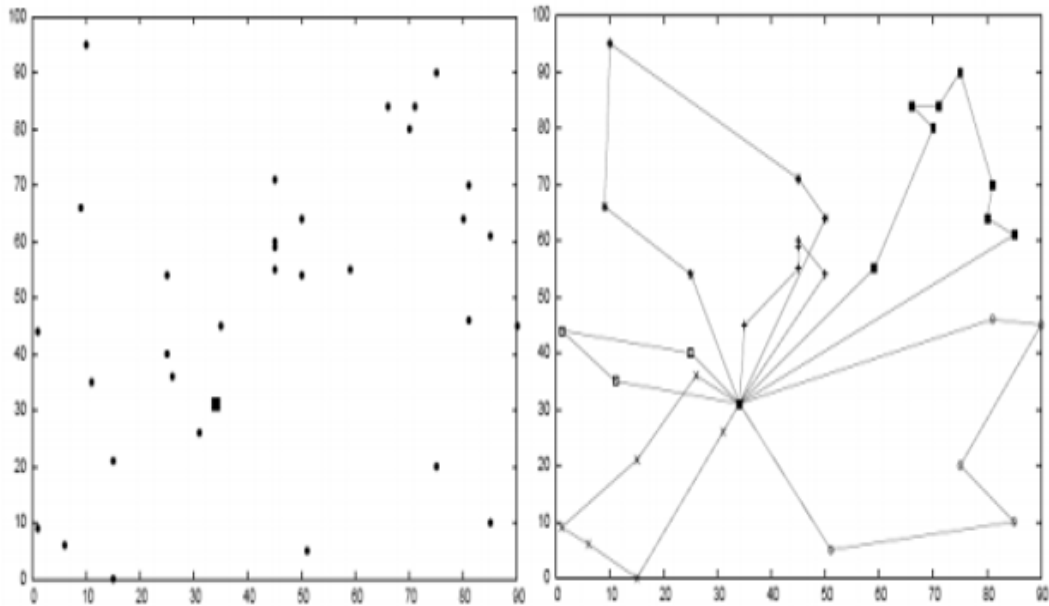
#### 3.1.1.1. Kısıtlı Araç Rotalama Problemi

##### Dağıtım - Toplamalı Araç Rotalama Problemi

Dağıtım ve toplamalı araç rotalama probleminde (DTARP) bazı müşterilere ürünler ulaştırılırken, bazılarında da toplanmaktadır. Araçlar dağıtıma depodan başlayıp dağıtımı depoda sonlandırırlar. Araçlar ilk olarak müşterilere ürünleri dağıtır, daha sonra ürünleri toplayacağı müşterilere uğrayarak ürünleri müşterilerden alarak depoya dönerler. Hava yolu ve deniz yolu taşımacılığı örnek olarak verilebilir. Burada amaç araç toplam dolaşım süresini en aza indirerek, araçların müşterilere malları dağıtmak ve müşterilerden toplayarak depoya getirmek için yeterli kapasiteye sahip olmasına dikkat etmektir.

##### Kapasite Kısıtlı Araç Rotalama Problemi

Bir veya daha fazla sayıda deposu bulunan işletmelerin talepleri belli sayıda müşterisine ulaşabilmesi için yükleme kapasiteleri kısıtlı araçların rota planlaması problemidir [33]. Kapasite kısıtlı araç rotalama probleminde (KKARP) araç kapasitesi sabit, araçlardaki ürünlerin bitmesi durumunda depoya dönülerek ürün tedariki yapılır. Müşteri talepleri bölünme olmamasının nedeni müşteriye tek araç uğramasıdır. Rota üzerindeki müşteri istekleri araç kapasitesi altında olmalıdır. Su dağıtım işleminde izlenebilen yol örnek olarak verilebilir.



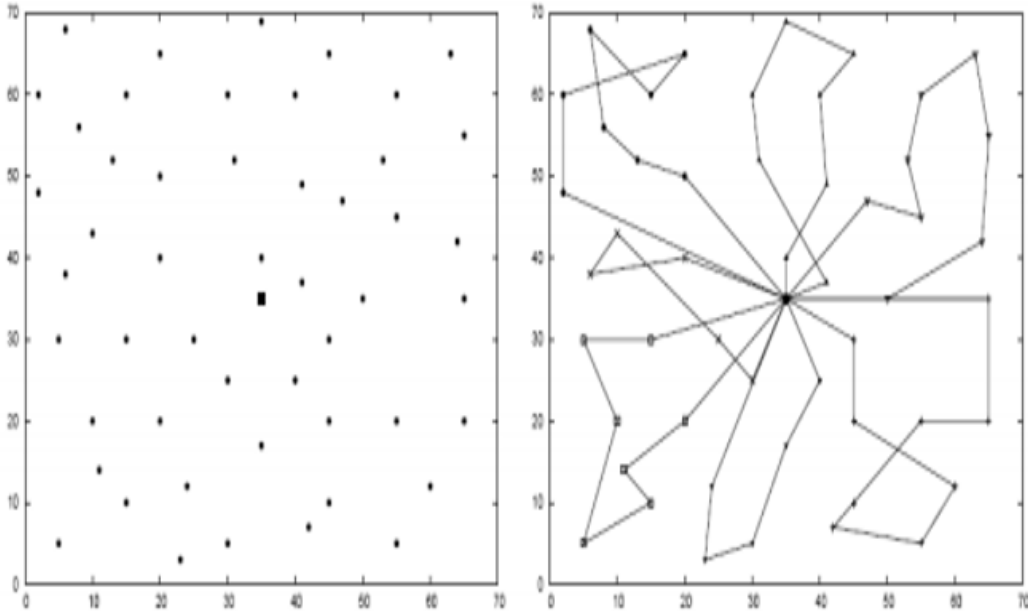
Şekil 3.2- Kapasite Kısıtlı Araç Rotalama Problemi Çözümü

## Çok Depolu Araç Rotalama Problemi

Çok depolu araç rotalama probleminde (ÇDARP) çok sayıda deposu olan dağıtım işletmesi her bir deposunda kendi müşteri grubunu belirleyecek şekilde dağıtım hizmetini sunar. Müşterilere en uygun depodan çıkan araç ile hizmet götürülür. Dağıtım bittiğinde araç depoya geri döner. Talepler araç kapasitesini geçmemelidir. Depolardan her bir tanesine bir araç filosu tahsis edilir. Her araç bir depodan dolaşmaya başlayarak, o depoya atanan müşterileri dolaşır ve başlangıç deposuna geri döner.

## Zaman Pencereci Araç Rotalama Problemi

Araç filosunun maliyetleri en alt düzeye çekerek, müşterilere hizmet sunmak için çeşitli yerlerde durması ve araçları etkin olarak kullanılmasını amaçlamaktadır. Zaman pencereci araç rotalama problemi (ZPARP) müşterilere bir zaman aralığında dağıtım yapılır. Araçların kapasitesi ve müşterilerin belirli bir zaman diliminde karşılanması gereken talepleri vardır. Zaman penceresi müşterilerin ziyaret edilmek istendiği zaman dilimidir. Dağıtım zaman aralığında önce ulaşırsa aracın dağıtım yapılan süre boyunca beklemesi gerekmektedir. Randevulu kargo teslimat işlemleri örnektir. Amaç kullanılacak araç sayısını ve toplam seyahat mesafesini minimize etmektir. Yaklaşımların bazılarında zaman penceresinde uyulması gereken kısıtlar bulunurken bazılarında ise uyulmamasına izin verilir ancak maliyet söz konusudur [34].



Şekil 3.3- Zaman Pencereci Araç Rotalama Probleminin Çözümü

## **Geri DönüŖlü Araç Rotalama Problemi**

Geri dönüŖlü araç rotalama problemi (GDARP) müŖterilerin ürünleri geri göndermesine dayanır. Araçların arkadan yüklemeli olması nedeniyle toplama işleminin yapılabilmesi için ürünlerin dağıtım işlemleri yapılmalıdır. Dağıtılacak ve toplanacak miktarlar sabittir ve önceden bilinir. Dağıtım işinin toplama işlemlerinden ve araç kapasitesinin rotaya atanan geri dönüŖ noktalarıyla deęiŖmedięi rotalardan oluşur.

### **3.1.1.2. Yol Durumlu Araç Rotalama Problemi**

#### **Simetrik yollu Araç Rotalama Problemi**

ARP güzergâhındaki  $i$  ve  $j$  noktaları arasındaki gidiŖ ( $d_{ij}$ ) ve dönüŖ ( $d_{ji}$ ) mesafelerinin eŖit ve mesafeler matrisinin simetrik olduęu ARP, simetrik yollu araç rotalama problemi (SYARP) olarak tanımlanır. GidiŖ dönüŖ mesafesi birbirine eŖittir ( $d_{ij} = d_{ji}$ ) [33].

#### **Asimetrik Araç Rotalama Problemi**

Dağıtımın depodan müŖteriye gidiŖ mesafesi ile müŖteriden depoya olan uzaklıęın farklı olduęu araç rotalama problemidir. ARP güzergâhındaki  $i$  ve  $j$  noktaları arasındaki gidiŖ ( $d_{ij}$ ) ve dönüŖ ( $d_{ji}$ ) mesafelerinin eŖit ve mesafeler matrisinin simetrik olmadıęı ARP, asimetrik araç rotalama problemi (AARP) olarak tanımlanır. GidiŖ ve dönüŖ rotaları ayrı hesaplanmalıdır. Problemden  $i$  ve  $j$  noktaları için  $i$  noktasından  $j$  noktasına gitmek için gerekli olan mesafe  $j$ 'den  $i$  noktasına olan mesafeye eŖit olamayabilir ( $d_{ij} \neq d_{ji}$ ) [33].

### **3.1.1.3. Statik / Dinamik Araç Rotalama Problemi**

#### **Statik Araç Rotalama Problemi**

Statik araç rotalama problemi (SARP), rotalama süreci baŖlamadan rotalarla ilgili tüm bilgilerin bilinmesi ve rotalamaya iliŖkin bilgilerin deęiŖmemesi gerekmektedir [35]. Statik araç rotalama probleminde (SARP) zaman, müŖteri talep ve konumları, araç kapasiteleri, rota durumları sabittir ve çözüm boyunca deęiŖmemektedir. SARP rotalama işleminin yapılmasından önce, durumların saptanarak rotalama bitiminde ilk durumların deęiŖmemesi olarak deęerlendirilmektedir.

#### **Dinamik Araç Rotalama Problemi**

Pillac [36], dinamik araç rotalama problemlerinde (DARP) çözüm sürecinde araç rotalarının yeniden tanımlanabileceęini belirtmiŖtir. Optimizasyonunda zaman tasarrufu, hedeflenen maliyet gibi etkenler yenilenir. Okul servislerinde öğrencilerin devamsızlıklarına göre rotanın yeniden hesaplanması örnek olarak verilebilir. Optimizasyon problemini dinamik yapan iki etkin

unsur problemle ilgili bilgilerin zamana bağımlı ve çözümlerin zaman geçtikçe yeni gelen bilgilerle aynı zamanlarda bulunmalıdır. Bu durum problemde ilk çözümün bulunamayacağı anlama gelmektedir [37].

### **3.1.2. Araç Rotalama Problemi Çözüm Yöntemleri**

#### **3.1.2.1. Kesin Çözüm Yöntemleri**

Yaygın olarak kullanılan kesin çözüm yöntemleri kesme, dal/sınır ve fiyat algoritmalarıdır. Bu algoritmalar yoluyla problem tam sayılı doğrusal programlama modeli şeklinde ele alınıp optimizasyon işlemlerinde uygun olan sonucu belirlemek hedeflenmiştir. Hadjar ve arkadaşları [38], Fügenschuh [39], Kumar ve Jain [40], konuyla ilgili çalışmalar yapmışlardır.

#### **3.1.2.2. Sezgisel Yöntemler**

Sezgisel yöntemler, hedefe ulaşmanın kesin yolunu göstermektedir. Bir eksiklik konusu olduğunda bu eksiklik sezgisel yöntemlerden ziyade problemin kendisi ile ilgilidir. Algoritma problem çözümüne bir takım verilerden yola çıkarak ve sonlu sayıda aşamalardan geçerek kesin komutlar yoluyla olanak sağlar. Algoritma çalışma esnasında her türlü belirsizlikten arınmış durumdadır. Algoritmanın çalışması, her biri komutla belirlenen bir dizi etkiyi oluşturur ve bu her bir dizi etki önceki komutun sona ermesiyle birlikte yürütülür. Algoritmanın temel yapısı bu sıra düzen ilkesine dayanır [41].

#### **3.1.2.3. Metazgisel Yöntemler**

##### **Tavlama Benzetimi**

Tavlama benzetimi (TB) [42], metallerin tavlama sıcaklığı süresince ısıtılıp sonra yavaşça soğutulmasına benzediğinden bu ismi almaktadır. Yapay ısıtım işlem algoritması olarak da bilinir. Genetik algoritmanın (GA) evrimsel benzetim yöntemi ve katıların belirli bir başlangıç sıcaklığından yavaşça soğutulduğu tavlamanın benzetimi olan arama yöntemi şeklinde tanımlanmaktadır. Rastgele çözüm kullanır. Komşu çözümler yoluyla amaç fonksiyonundaki değişimi hesaplayarak iyileşme olmama durumunda aramaya devam etmektedir. Yerel en iyi sıcaklık artışına neden olan komşu hareketleri kontrol ederek sonuca ulaşabilmektedir. Spada ve arkadaşları [43], benzetimli tavlama algoritması kullanarak sabit sayıda okul servis aracı ile okula sunulan hizmet seviyesini en üst düzeye çıkarmayı hedeflemişlerdir. Dana ve arkadaşları [44], okul servis aracının bir öğrenciden diğerine hareketi anında yolun durumuna bağlı olarak zamanın belirsiz oluşunu da değerlendirilerek tavlama benzetimine temelli bir metod geliştirmişlerdir.

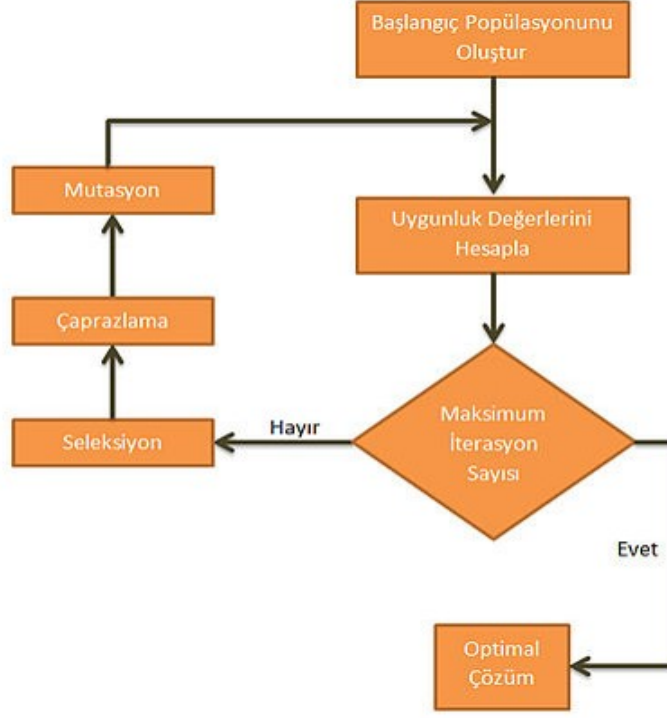
## Tabu Arama

Tabu Arama (TA) arama süresince en iyi komşu tasarımların elde edilmesini sağlayan hareketler yapay kısa dönemli bir hafızaya kaydedilir. Bu hafıza sayesinde aramada önceki tasarımlardaki hareketlerin tekrarının yasaklanmasına tabu denir. Yasaklanan bu hareketler yapay hafıza görevi gören tek boyutlu bir listede tutulur. Tabu listesine kaydedilen her hareket bir tasarım değişkenini ifade etmektedir. Her tasarım değişkeni için tanımlanan komşu tasarımlar içerisinde en iyi olan tasarım tabu listesinde kontrol edilerek mevcut tasarımın yerine iyileştirme yapılır. Arama işlemine hazır durumda bulunan tasarımın yerine kullanılan en iyi komşu tasarım ile devam edilir. Aynı işlem diğer tasarım değişkenleri için de tekrarlanarak bir iterasyon tamamlanır. Son aşama faktörü sağlanıncaya kadar her iterasyon için bu işlemler yenilenir [45].

Glover ve Laguana [46], tabu arama (TA) yöntemini problemin çözümü olabilecek kümenin oluşturulması ve bu kümede bulunan öğelerden birinin doğru kabul edilmesine kadar iterasyonlara devam edilmesi şeklinde tanımlamıştır. Elde edilen sonuçlar arasında tekrara düşülmesini engellemek amacıyla veriler tabu listesi şeklinde gösterilir. Oluşturulan tabu listesi dinamik olarak yeni elemanların listeye eklenmesi ve listede en çok kalan elemanın liste dışına çıkarılması şeklinde güncellenir. Endüstriyel üretim ve ağ optimizasyonu için kullanılmaktadır.

## Genetik Algoritma

Genetik algoritma (GA), bir problemi çözmek için seçim, mutasyon, kalıtım ve rekombinasyon gibi evrimsel biyoloji tekniklerini kullanır. Genetik algoritmalarda kullanılan en yaygın ve popüler yöntem, belirli bir popülasyon içinden rastgele bir grup birey oluşturmaktır. Bu şekilde oluşturulan bireyler, tasarımcı veya programcı tarafından sağlanan değerlendirme fonksiyonu yardımıyla değerlendirilir. Bireyler, verilen duruma uygunluğunu dolaylı olarak vurgulayan bir puanla sağlanan bir sonraki aşamadır. En iyi iki birey, bir veya daha fazla yavru oluşturmak için kullanılan ikinci aşamadır, bu aşamadan sonra yavru üzerinde rastgele mutasyonlar yapılır. Uygulamanın ihtiyaçlarına göre, kabul edilebilir bir çözüm bulunana kadar veya belirli sayıda nesil geçene kadar süreç devam eder. Genetik algoritma, yapay zeka ve hesaplamada kullanılan oldukça yaygın bir arama yöntemidir. Evrimsel biyoloji ve doğal seleksiyon teorisine dayanan problemleri aramak için basitleştirilmiş çözümler bulmak için kullanılır. Genetik algoritmalar, çok büyük ve karmaşık veri kümeleri ve veri kümeleri arasında arama yapmak için olağanüstüdür. Genetik algoritmalar, sınırlandırılmamış ve kısıtlı optimizasyon sorunlarını çözmek konusunda oldukça yetenekli olduklarından, karmaşık sorunlara mantıklı ve faydalı çözümler bulabilecekleri düşünülmektedir [47]. İlgili makalede yer alan problem çözümü için geliştirilen genetik algoritma akış diyagramı aşağıda belirtilmiştir.



Şekil 3.4- Genetik Algoritmanın Akış Şeması

Holland [48], doğada oluşan olayları bilgisayar sistemlerinde ifade edebilmek amacıyla ilk olarak genetik algoritma kavramını açıklamıştır. En iyi olanın hayatta kalma mantığına dayanır. 0 veya 1 değeriyle çözüm kümelerini için rastgele kromozomlar oluşturmuştur. Kromozomların her biri için uygunluk fonksiyonu hesaplanmaktadır [49]. En uygun olanlardan yeni nesiller oluşturabilmek için anne ve babalar seçilmektedir. Seleksiyon sonrası eş bireylerin oluşmasını önlemek için anne ve babalar çaprazlanarak yeni yavrular oluştur ve bunlar anne ve babaların yerine geçmektedir.

Genetik algoritmada olası çözümlerin kodlandığı kromozom adı verilen bir çözüm grubu oluşturulur. Her kromozomun uygunluk fonksiyonu bulunur. Bu kromozomlar eşlenerek yeniden kopyalama ve değiştirme operatörleri uygulanır ve yeni bir toplum oluşturulur. Eski kromozomlar ortadan kaldırılarak yeni kromozomlara yer açılır. Kromozomların uygunlukları tekrardan hesaplanır. Jenerasyon süresi dolmamışsa tekrar eşlenerek yeniden kopyalama ve değiştirme operatörleri uygulanır. Bu zaman süresince bulunmuş en iyi kromozom problem sonucunu oluşturur.

## Yapay Sinir Ağları

Yapay sinir ağları (YSA) insan beyninde öğrenme işlevini yapan bilgisayar sistemleridir. Öğrenmeyi örneklerle yapar. Birbirine bağlı bir ağırlık değeri olan süreç adı verilen yapay sinir hücrelerinden oluşur. Sahip olduğu bilgi bu ağırlık değerlerinde saklı halde ağa yayılmış

durumdadır. Bir yapay sinir ağı, ağırlıklı bağlantılarla bağlanmış birimlerden oluşan sayısal bir modeldir, bir birimden başka bir birime bir bağlantı boyunca sinyal gönderilir ve ilgili ağırlıklar değiştirilir [50]. Ortama uyum sağlayan, belirsizlikler altında karar verebilen, eksik bilgi ile çalışabilen, hatalara karşı toleranslı olan bir hesaplama yöntemine sahiptir. Ağ yapısı belirlenmesi ve parametrelere seçiminde belirli bir standardın olmaması, problemlerin nümerik bilgilerle ifadesi, eğitimin sonucunun bilinmemesi, ağın davranışlarını açıklayamaması gibi olumsuzluklara sahiptir. Sınıflandırma, örüntü tanıma, sinyal filtreleme, veri sıkıştırma ve optimizasyon çalışmalarında oldukça etkilidir. Veri madenciliği, parmak izi tanıma, optimum rota belirleme, malzeme analizi, kalite kontrol ve tıbbi analiz alanlarında kullanılmaktadır [51].

### **Parçacık Sürü Optimizasyonu**

Parçacık sürü optimizasyonu (PSO) yöntemi ile Kennedy ve Eberhart [19], kuş sürüleri davranışına dayanan hesaplamalar yapmışlardır. Sürede bulunan her parçacık, deneyimlerinden yararlanarak sürünün en iyi pozisyonuna geçmeye çalışır. Her parçacık pozisyonun sürüdeki en iyi pozisyonundaki üyeye göre ayarlar. Parçacığın geldiği yeni konum önceki konuma göre iyi olmakta ve bu konuma yaklaşma hızı rasgele oluşmaktadır. Hedefe ulaşıldığında çözüm son bulmaktadır. Cura [52], PSO üzerinden geliştirdiği algoritma ile bilinen ve bilinmeyen küme sayılarındaki kümeleme problemlerine çözüm üretmiştir.

### **Karınca Kolonisi Optimizasyonu**

Karınca kolonisi optimizasyonu (KKO) karıncaların besin kaynağına doğru hareketlerinden ortaya çıkan bu algortmada karıncalar yiyeceğe ulaşmak için kısa yolu tercih etmektedir. Gidiş gelişlerinde yola bıraktıkları feromonları sonrasında takip ediyor oluşlarının bunu sağladığı bilinmektedir. Karıncaların hafızasız olması ve uzun yollarda feromon etkisinin azalması koloninin takip durumunu olumsuz etkilemektedir. Mazzeo, Loiseau [53], Gajpal, Abad [54], optimizasyonla ilgili çalışmalar yapmışlardır.

Karınca kolonileri, besin odaklarından yuva istikametlerine ve ya tam zıttı yönde hareket ederken toprak üzerine, yol boyunca iz oluşturmak amacıyla feromon denilen maddeler bırakırlar. Bu feromonlar karıncalar tarafından koklanarak karıncaların yollarını seçim yaparak bulmaları gerektiği durumlarda karıncalara katkı sağlarlar. Feromon izi karıncaların yolunu bulmalarına olanak tanır. Aynı zamanda yuvadaki karıncalar arasında istikamet tayin etmede kolaylık sağlar [55]. Karıncaların yol takibi esnasında yola bırakmış oldukları feromon miktarı problem çözümünün kalitesiyle doğru orantılıdır. Karıncaların birden fazla yol arasından geniş miktar feromon bırakılan yolun güzergâh olarak seçilme olasılığı bulunmaktadır [56].

## **3.2. Okul Servisi Rotalama Problemi**

ARP alt türü OSRP aynı zamanda gezgin satıcı (TSP) problemidir. OSRP, bir servis aracının en makul süre ve en kısa rotada listesindeki duraklardan öğrencileri toplayıp okula götürmesini ve okulda toplanan öğrencileri evlerine bırakmasıdır. Amaç servis hizmetinde ortaya çıkan zaman, yol maliyetini en aza indirmektir [10].

### **3.2.1. Okul Servisi Rotalama Problemi Etkenleri**

#### **Verilerin Elde Edilmesi**

Servis araçları, öğrenci, okul ve mesafeler matrisi saptanır. Servis araçlarının başlangıç konumu, öğrencilerin durumuna göre özellikleri belirlenir. Öğrencilerin okula olan mesafeleri, adresleri ve şayet var ise öğrenci engel durumları belirlenir. Okulun konumu, başlangıç ve bitiş zamanı ve mevcut öğrenci sayıları belirlenir. GPS konumlarından yararlanılarak mesafeler matrisinde en kısa düğümler arasındaki mesafe belirlenir.

#### **Servis Rotasının Belirlenmesi**

Servis rotaları için yararlanılan algoritmalar, önce rota belirle sonra sınıfla ve önce sınıfla sonra rota belirle olarak isimlendirilir. Önce rota belirle sonra sınıfla yönteminde GSP metodu gibi tüm duraklardan oluşan büyük bir rota belirlenir, problemin verilerine göre alt rotalara bölünür. Önce sınıfla sonra rota belirle metodunda öğrenciler kısıtlara göre gruplara ayrılır ve gruplar için rotalar oluşturulur. Sezgisel yöntemlerle rota optimizasyonu gerçekleştirilir.

#### **Durakların Belirlenmesi**

Servis hizmeti alınan bölge şehir merkezi konumunda ise öğrenciler evlerinden duraklara gitmelidir, kırsal alanlarda öğrenciler evlerinden alınarak servis hizmetten yararlanmalıdırlar. Öğrenci güzergâhı üzerinde yer alan durakların yerleri belirlenmelidir.

#### **Rota Planının Düzenlenmesi**

Okul ya da okullara servis hizmeti sağlayan firmaların servis hizmetini sunduğu güzergâhtaki okullar için rota belirleme ve düzenleme yapması gerekir. Rota optimizasyonu belirlenen bu güzergâhlara göre değerlendirilerek yapılmaktadır.

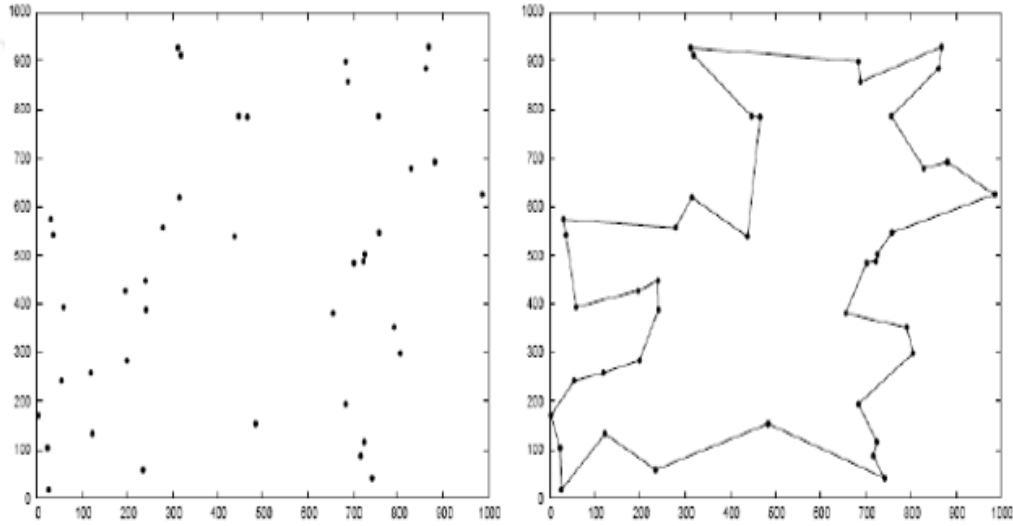
#### **Okul Başlangıç ve Bitiş Zamanı Belirlenmesi**

Çok sayıda okul için servis hizmeti sunulan problemlerin çözümünde hizmetin sunulduğu okullarda okul ders başlangıç ve bitiş saati zamanlarının önceden belirlenmesi ve düzenlenmesi gerekmektedir.

### 3.3. Gezgin Satıcı Problemi

#### Problemin Tanımı ve Önemi

Gezgin satıcı problemi (TSP) bir satıcı tarafından başladığı şehre geri dönmesi şartıyla ziyaret edilmesi gereken şehir grubundan meydana gelir. Problem çözümünde toplam mesafeyi minimize edecek rotalar bulunmaya çalışılır. Her satıcı bir depo ile başlar ve kendi dolaşımı bittiğinde bu depoya dönmek zorundadır. Arasındaki uzaklıkları bilinen noktaların her birisinden sadece bir kez geçen en az maliyetli ve en kısa turun bulunmasını hedefleyen problem türüdür. Probleminin önemi, seyahatini minimize etmek isteyen satıcı sorunundan kaynaklanmaz, çoğu seyahat rotalarıyla ilgili bilinmeyen çok sayıda farklı sorundan kaynaklanır.



Şekil 3.5- Gezgin Satıcı Problemi Rota Optimizasyonu

#### Problemin Matematiksel İfadesi

Gezgin satıcı probleminin matematiksel ifadesini Helsgaun, n adet şehir için,  $i, j = 1, \dots, n$  olmak üzere  $c_{ij}$ , i şehirden j şehrine gitmenin rota maliyeti,  $c_{ij}$ 'lerden oluşan  $C = (c_{ij})$  maliyet matrisi olarak belirtir ve gezgin satıcı problemini matematiksel olarak:  $\text{Min: } c_{i_1 i_2} + c_{i_2 i_3} + \dots + c_{i_n i_1}$  formülü ile 1'den n'e kadar P:  $(i_1, i_2, i_3, \dots, i_n)$  permütasyonunun en küçük değerini bulmak şeklinde tanımlar [57].

#### Probleminin Çözüm Yöntemleri

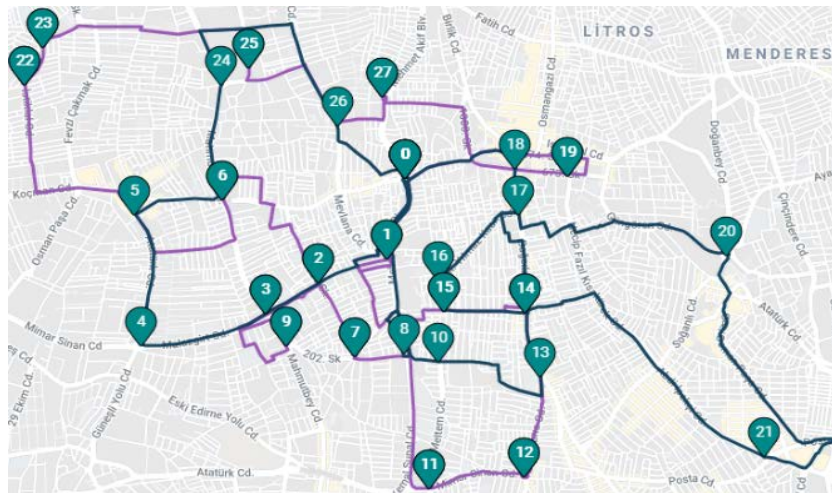
Problem çözümde kullanılan algoritmalar kesin ve yaklaşık algoritmalar olarak iki ana kısma ayrılabilir. Kesin algoritmalarda optimum sonucu bulmak kesindir. Ancak kesin algoritmalardan bir tanesi olan kesme düzlemi yöntemiyle yaklaşık 7400 şehirlik bir problemi çözmek, geniş bir bilgisayar ağı yardımıyla ortalama 3-4 yıl gibi uzun bir süre alacaktır. Artan

şehir sayısı ile kesin algoritmaların verimliliği ortadan kalkar. Yaklaşık algoritmalarda ise optimum sonuç kesin algoritmalara göre daha net verilemese bile kısa bir süre içerisinde makul sonuçlar elde edilebilmektedir [57].

### 3.4. Çoklu Gezgin Satıcı Problemi

Çoklu gezgin satıcı problemi (MTSP) çoklu sayıda gündelik problemi modellemek amacıyla kullanılabilir karmaşık yapıya sahip bir problemdir.  $N$  adet şehir için alt tur dolaşımını gözardı eden, her şehri sadece bir kez dolaşarak gezen, en iyiye yakın turu bulmayı amaçlayan ve yine karmaşık bir problem olan gezgin satıcı problem çeşidi olarak bilinmektedir [58]. Çoklu gezgin satıcı problemi  $m$  adet satıcının rotada yer alan şehirleri gezdiği ve her şehrin yalnızca bir satıcı tarafından dolaşıldığı problemdir türüdür. Bir şirketin farklı şehirlerde yaşayan birden fazla satış elemanı olduğu ve şirketin çeşitli şehirlerde yer alan depolarını belirli bir miktardan daha az olmamak üzere bu satış elemanlarının ziyaret etmesi gerektiği durumlarda kullanılır. Tek satış görevlisinin durumunu birkaç satış görevlisini içeren bir soruna dönüştürür. Birden fazla gezici satıcı sorunu, birden fazla satış görevlisine izin verilen gezici satış elemanı probleminin bir genellemesidir. Bir dizi şehir verildiğinde, bir depo  $m$  satış görevlileri bulunur ve bir maliyet ölçütü her bir satış elemanı için, toplam tur maliyetini en aza indirecek ve her şehri yalnızca bir satıcı tarafından tam olarak bir kez ziyaret edilecek şekilde bir tur belirlemektir.

MTSP problemlerini çözmek için iki yaklaşım vardır, kesin yaklaşım ve sezgisel yaklaşım. Kesin algoritmaların katı matematiksel temelleri olmasına rağmen, problem çözme yetenekleri tamamen problemin boyutuna bağlıdır. Boyut artırıldığında, kabul edilebilir bir süre içinde çözülemeyebilir, hatta çözülemeyebilir. Problem ölçeği genişledikten sonra kesin algoritmaların hesaplama zorluklarının üstesinden gelmek için, sezgisel algoritmaların araştırması giderek ana akım yön haline gelmiş ve gezgin satıcı probleminin için sezgisel bir arama tekniği olmuştur.



Şekil 3.6- İki Araçlı Gezgin Satıcı Problemi Rota Optimizasyonu

## 4. ÖNERİLEN MODEL

Çoklu seyahat eden satış elemanı problemi, tutarlı bir grafiğin bazı düğümlerinden başlayan, diğer tüm düğümleri tam olarak bir kez ziyaret eden ve başlangıç düğümüne geri dönen en kısa rotayı bulmaya yönelik iyi bilinen seyahat eden satıcı probleminin bir genellemesidir. TSP'nin MTSP'ye genelleme yöntemi, düğümlerde, aynı depo düğümünden başlayıp biten tüm m satış elemanlarıyla birlikte m satış elemanlarından biri tarafından bir kez ziyaret edilmesi gerektiğidir. M satıcının bulunduğu bir depo, bir dizi şehir ve bir maliyet ölçütü göz önüne alındığında, çoklu seyahat eden satış elemanı probleminin amacı, her bir satış görevlisinin her bir şehri yalnızca bir kez ziyaret etmesi ve turun maliyet toplamını en aza indirmesi için bir tur ayarlamaktır.

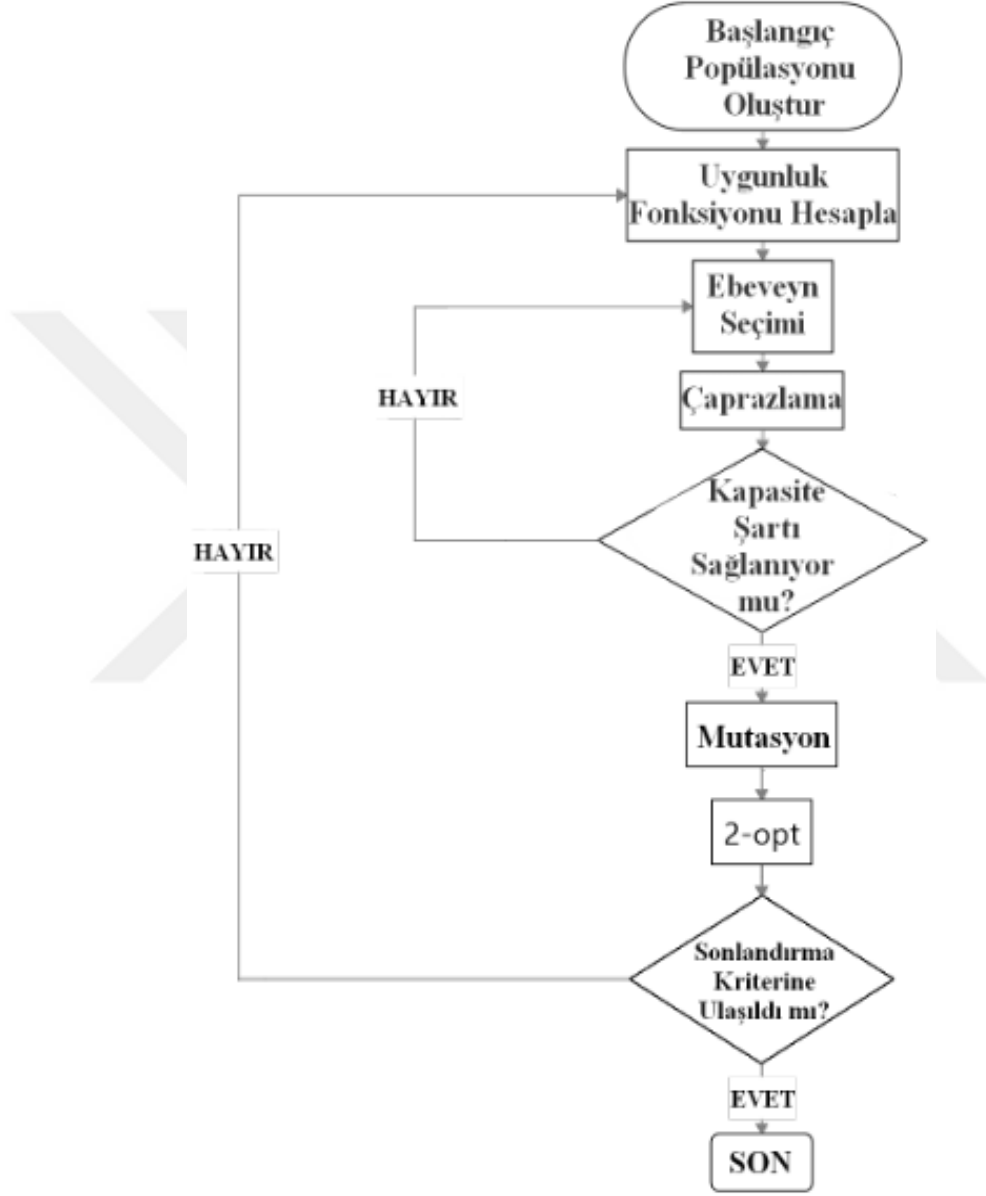
Çalışmada kullanılan genetik algoritma, bir problemi çözmek için seçim, mutasyon, kalıtım ve rekombinasyon gibi evrimsel biyoloji tekniklerini kullanır. Genetik algoritmalarda kullanılan en yaygın ve popüler yöntem, belirli bir popülasyondan rastgele bir grup birey oluşturmaktır. Bu şekilde oluşturulan bireyler, tasarımcı veya programcı tarafından sağlanan değerlendirme fonksiyonu yardımıyla değerlendirilir. Bireyler, verilen duruma uygunluğunu dolaylı olarak vurgulanan bir puanla sağlanan bir sonraki aşamadır. İkinci aşama en iyi iki birey, bir veya daha fazla yavru oluşturmak için kullanılır. Bu aşamadan sonra yavru üzerinde rastgele mutasyonlar yapılır ve makul bir çözüm bulunana kadar veya belirli sayıda nesil geçene kadar devam eder.

Çalışmada Google Maps Distance Matrix API uygulamasından yararlanılarak noktalar arası mesafeler anlık olacak şekilde alınmıştır. Rota çizdirmek maksadıyla Google Maps Directions API uygulamasından yararlanılmıştır. Bu uygulamalar Google Maps API kütüphanesinde yer alan harita işlemleri için kullanılan uygulamalardır. Google Maps Distance Matrix API uygulamasından trafik yoğunluğunu da dikkate alarak yaya, bisiklet, toplu taşıma ve araba olacak şekilde mesafe ölçümlerinden dört farklı şekilde yararlanılabilmektedir. Çalışmamızda göz önünde bulundurulmuş husus trafik yoğunluğunun anlık olarak dikkate alınıp rotanın bu doğrultuda çizdirilmesidir. Bu yöntemle edinilen veriler genetik algoritmada değerlendirilip rota oluşturulur ve rotaların Google Maps Directions API uygulaması ile gerçek güzergâh olarak çizdirilmesi sağlanmıştır. Google Maps Directions API uygulamasını çalışmamızda kullanabilmek için C# ile yazılmış açık kaynak kodlu `vivet/GoogleApi` kütüphanesi kullanılmıştır [59].

### 4.1. Amaç






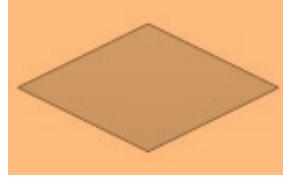



Bu proje, bir bölgedeki bir okula hizmet veren bir firmanın servis rotasını optimize edebilmek amacı için hazırlanmıştır. Okulda ders alan öğrencilerin ders saati başlamadan önce evden alınıp okula bırakılmaları ve daha sonra son dersin bitimi ile birlikte öğrencilerin okuldan alınıp evlerine bırakılmalarındaki sürelerin iyileştirilmesinin sağlanması amaçlanmaktadır. Programda genetik

algoritma yöntemi kullanılarak en kısa rota ve optimize edilmiş bir sonuç hedeflenir. Öğrenci servisinin rotası, öğrencinin okula devam etme durumuna göre ya da önceki gün okula devamına ilişkin elde edilen verilere göre dinamik olarak oluşturulur. Çalışmada, akıllı şehirlerin öğrenci servis hizmeti probleminin yapay zeka teknikleri ve genetik algoritmalar kullanılarak optimize edilmesi için gayret edilmiştir.



Şekil 4.1- Uygulama MTSP Genetik Algoritma Akış Şeması

Yukarıda gösterilen akış şeması veya akış diyagramı olarak bilinen algoritma yararlanmış olduğum genetik algoritmanın problemdeki çözümünü ifade etmektedir. Algoritmada ifade edilen şekiller bilgisayar programlama kodlarının tasarımcı tarafından anlaşılmasına yarar sağlamaktadır. Şekil 4.2’de akış şemasında belirtilen sembollerin ne anlama geldiği ile ilgili bilgiler gösterilmiştir.

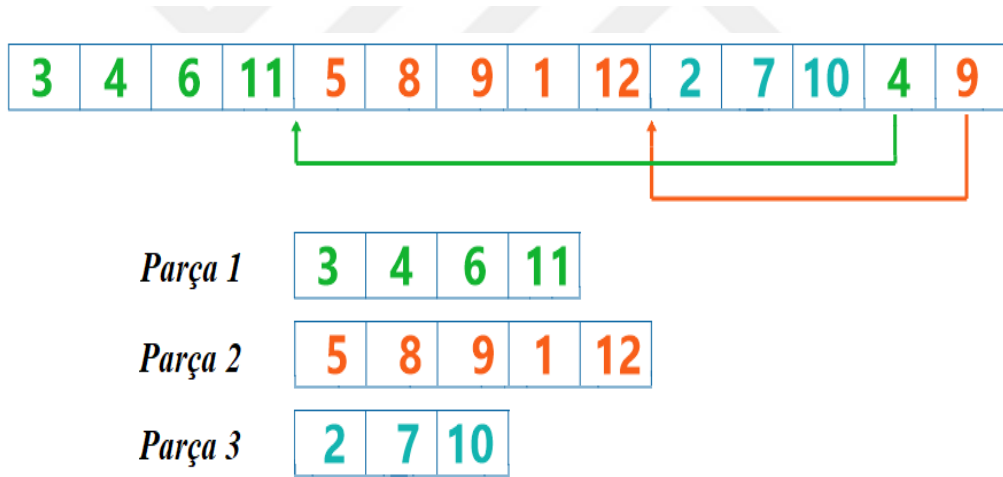
<p><b>Bağlantı Simgesi</b> Sembollerin bağlantı yönünü belirtir.</p>	
<p><b>Başlangıç / Bitiş Simgesi</b> İşlemin başlangıcını ve bitişini belirtir.</p>	
<p><b>Eylem Simgesi</b> Bir eylemi belirtir.</p>	
<p><b>Belge Simgesi</b> Bir rapor veya belgeyi belirtir.</p>	
<p><b>Birden Fazla Belge Simgesi</b> Birden fazla raporu veya belgeyi belirtir.</p>	
<p><b>Karar Simgesi</b> Bir karar alma olayını belirtir.</p>	
<p><b>Giriş / Çıkış Simgesi</b> Program girdi veya çıktılarını belirtir.</p>	
<p><b>Manuel Giriş Simgesi</b> Programa manuel bilgi girilmesi gerektiğini belirtir.</p>	
<p><b>Hazırlık Simgesi</b> Bir işlemin devam ettiğini belirtir.</p>	

Şekil 4.2- Akış Diyagramı Sembollerinin Anlamları

## 4.2. Kromozom

Kromozom yapısını doğru bir şekilde oluşturmak, genetik algoritmanın temeli olduğu için çok önemli bir adımdır. Bu çalışmadaki problem, kromozom koordinat numaralarını ve bu numaraların kırılma noktası indeksine işaret eden ve koordinat numarası dizisinin sonuna eklenen sayıları gösteren bir dizi sayıdan ibarettir. Kesme noktaları arasındaki ve dışındaki sayı dizileri, her biri bir aracın takip edeceği rotayı belirten rotanın bölümlerini oluşturacaktır.

Çalışmamızda, önce koordinat numaraları daha sonra bir önceki takımın koordinatının bittiğini gösteren kesme noktası noktaları tek bir diziye yazılmıştır. Tüm rotalar tek bir başlangıç noktasından başlayacak ve hedef nokta olarak bu noktaya dönecektir. Koordinatlar numaralandırılırken 1'den başlayarak toplam koordinat sayısına kadar numaralandırılır. 0 numaralı koordinat, başlangıç / hedef noktasıdır. Koordinat dizisinin başına ve sonuna ayırıcı koymak gerekli olmadığından, kesme noktası sayısı 1 eksi parça numarasıdır. Şekil 4.3'te örnek kromozom yapısındaki 12 noktanın koordinatları 3 yol parçasına bölünmüş ve parçaların kırılma noktası dizinin sonuna eklenmiştir.



Şekil 4.3- Kesme Noktası Kümelerine Dayalı Üç Parçalı Kromozom

## 4.3. Fitness Fonksiyonu

Uygunluk fonksiyonu, bir genetik algoritma probleminin çözümü için bireyin yeterli olup olmadığını belirleyen hedefdir. Fitness değeri, gelecek nesil için seçilecek bireyleri doğrudan etkiler. Bu problemde, uygunluk işlevi aşağıdaki gibi açıklanmıştır:

1. Kromozomlardaki maksimum parça değeri hesaplanmıştır.
2. Popülasyondaki minimum kromozom değerleri (ilk adımda hesaplanır) alınmıştır.

Toplam mesafeyi hesaplamak için aşağıdaki adımlar atılır:

1. Tüm mesafeleri tutmak için statik bir mesafe matrisi  $N \times N$  ( $N$  düğüm sayısıdır) oluşturulmuştur.

2. Google Distance API kullanarak tüm nokta çiftlerinin mesafesi alınmıştır ve mesafe matrisine yazılmıştır.

3. Kromozomdaki her bir ardışık nokta arasındaki mesafe değerinin toplamı alınarak rota bölümlerinin toplam mesafesi hesaplanmıştır.

4. Çevrimleri tamamlamak için parça dizisinin başına ve sonuna hedef nokta eklenmiştir.

#### **4.4. Başlatma**

Popülasyonda kromozom yapısı belirlendikten ve ilk kromozom oluşturulduktan sonra, genetik algoritmanın uygulanabilmesi için ilk kromozom kullanılmalıdır. Bir popülasyon oluşturmak için popülasyonun yöntemini ve büyüklüğünü belirlemek gerekir. Optimal popülasyon büyüklüğüne deneme ile karar verilir. Çok fazla popülasyon, uygulamanın yavaş çalışmasına neden olabilirken, çiftleşme havuzundaki çeşitlilik, popülasyondaki bireylerin sayısının yetersiz olması durumunda olumsuz etkilenir. Çalışmamızda ilk popülasyon, ilk kromozomun koordinatlarının rastgele karıştırılması ve yeni bireylerin elde edilmesi ile oluşturulmuştur.

#### **4.5. Seçim**

Seçim süreci, bir sonraki nesle yeni çocuk yaratmak için geçiş sürecini başlatmak amacıyla popülasyondan anne ve babaları seçme adımıdır. Anne baba seçimi çok önemli bir süreçtir çünkü çeşitliliğin dengesini korumak ve daha iyi bir uyum sağlamak, ele alınması gereken önemli bir konudur. Bu çalışmada, turnuva seçim (tournament selection) yöntemi aşağıda tanımlandığı şekilde kullanılmıştır:

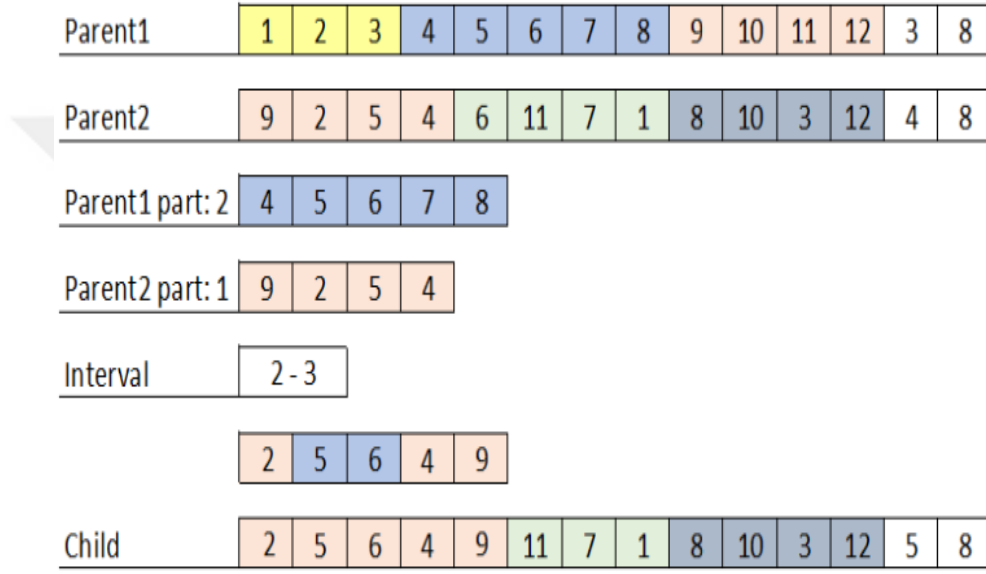
1. Popülasyondan 3 birey seçilmiştir.
2. Anne ve babalardan biri olarak en iyi performansa sahip birey seçilmiştir.
3. Başka bir anne ve baba seçmek için ilk iki adım tekrar uygulanmıştır.

#### **4.6. Çaprazlama**

Çaprazlama operatörü, seçme adımından üst öğeleri seçtikten sonraki ilk adımdır. Çok sayıda çapraz geçiş operatörü vardır ve bu projede sipariş çaprazlama operatörü (Order Crossover Operator) (OX1) kullanılmıştır. Projede kenar yeniden birleştirme operatörü (Edge Recombination Crossover) (ERX) de denenmiş ve daha kısa uygulama süreli OX1 tercih edilmiştir.

Sipariş çaprazlama (order based crossover) süreci aşağıdaki şekilde uygulanmıştır: Parent1 ve Parent2'den rastgele bir parça seçilmiştir (Parent1'den seçilen kısım P1Part, Parent2'den seçilen kısım P2Part olarak adlandırılmıştır). 1 ile parça uzunluğu arasında rastgele

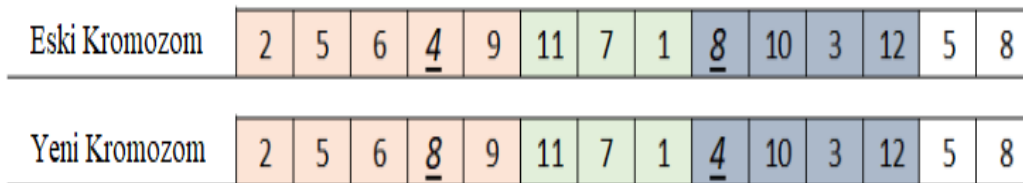
aralık seçilmiştir (Araç kapasite niteliklerini karşılamama olasılığını azaltmak için kısa parçanın uzunluğu üst sınır olarak belirlenir). P1Part'tan en üstteki rastgele seçilen aralıktaki sayı dizisini alınmıştır. Konumunuza göre P2Part'a yerleştirilmiştir. P2Part'ta kalan diğer öğeler sondan başlayıp başa dönecek ve öğeleri P1Part'tan hariç tutacaktır. P2Part'tan elde edilen elemanlar baştan başlayarak dizinin içine yerleştirilmiştir. P2Part Parent1'de ait olduğu konuma yerleştirilmiş ve dizinin geri kalanı için P2Part'ta yinelenen bir numara varsa diziden çıkarılmıştır. Çaprazlama işleminden sonra oluşturulan yeni çocuk, en kısa ve en uzun araç kapasitesi istekleri karşılıyorsa alınır, değilse seçim adımına geri dönlür.



Şekil 4.4- Çaprazlama İşlemi Sonucu Oluşan Kromozom Yapısı

## 4.7. Mutasyon

Bu aşamada, rastgele iki farklı numara yaratan ve bunları değiştiren değiş tokuş işlemini (mutasyon) gerçekleştiren metot kullanılır. Mutasyon oranı genetik algorithmada genel olarak önerildiği gibi, mutasyona uğrayıp uğramamaya karar vermek için kullanılan çok düşük olasılıklı bir yöntemdir. İki seçenek, seyahat eden satıcı problemini çözmek için bilinen ve basitçe uygulanabilir bir yerel arama algoritmasıdır. Tesisat parçalarında çapraz kenarı kaldırmak için bir takas işlemi gerçekleştirir.

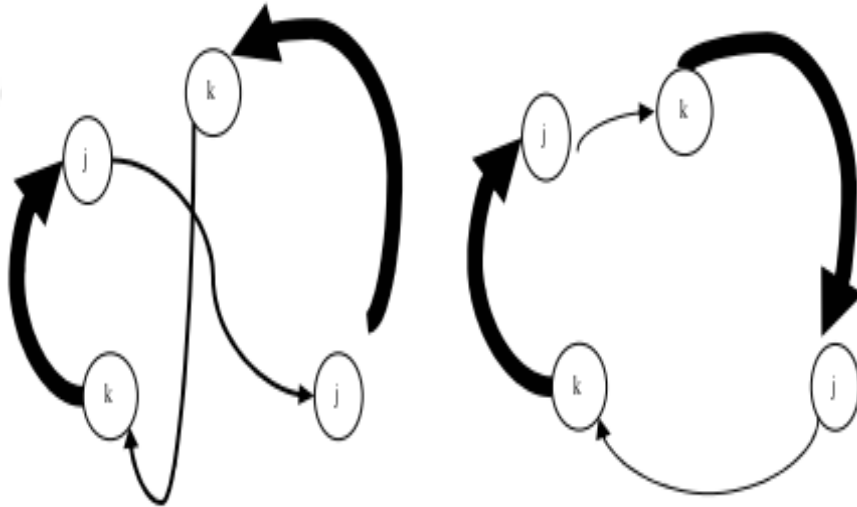


Şekil 4.5- Mutasyon İşlemi Sonucu Oluşan Kromozom Yapısı

## 4.8. 2-opt Optimizasyon

Temel çalışma mantığı olarak, tur içerisinde yer alan her iki kenarın silinmesi ve bu yöntem ile iki parçaya ayrılan turun maliyetleri düşürecek biçimde, farklı olarak bağlanması olarak nitelendirilir. Bu işlem sonucunda ikili değişikliklerin çözümü iyileştiremediği son noktaya kadar devam ettirilir [19]. 2-opt algoritmasının gezgin satıcı probleminde çalışma yöntemi: Rassal olarak turdaki nokta çiftlerini belirler, tur bozulmayacak şekilde nokta çiftlerinin yerini değiştirir, yeni oluşan tur önceki tura göre bir gelişme sağlamış ise nokta çiftlerini yeni yerlerinde bırakır aksi halde eski yerine iade eder. 2-Opt algoritmasında her bir düğüm eşi için aşağıda ifade edilen basamaklar izlenir.

- 1) Düğüm eşleri tur işlemi dışında tutulur.
- 2) Tur çalışması aksamayacak biçimde bir tur meydana gelmiş olan diğer bir düğüm eşi ile yer değiştirilir.
- 3) Yeni meydana gelen tur önceki tura göre bir gelişme sağlamış ise düğümler yeni yerlerine aktarılır, şayet bir gelişme sağlanmamış olması durumunda ise düğümler eski yerlerine gönderilir [7].



Şekil 4.6- 2-Opt Algoritması Konsepti

## 4.9. Fesih

Genetik algoritma kullanarak gerçekleştirmiş olduğumuz optimizasyon işlemi sonlandırmak için iki kriter kullanıyoruz:

- Yineleme belirtilen sayıya ulaşmışsa çalışma sonlandırılacaktır.
- Mevcut yineleme ile son en iyi yineleme arasındaki yüzde, toplam yineleme sınırının yarısına ulaşırsa işlem sonlandırılacaktır.

#### 4.10. Google Maps API

Google Maps API çalışmamızda yararlandığımız geniş ölçekli bilgiler ve gerçek zamanlı trafik verilerinin kullanıcıların bir noktadan başka bir noktaya giderken en iyi yolu bulmasına yardımcı olan bir Google uygulamasıdır. API, bir yazılımın diğer yazılımlar içerisinde belirli işlevlerini kullanabilmesi adına oluşturulmuş bir arayüzdür. Google Maps API uygulamasında, harita tabanlı Google Maps Developer Console arayüzünde bulunan hazır kodlardan faydalanılmaktadır. Google Maps API internette açık kaynak kodlu olarak yer almaktadır. Bir kullanıcı tarafından sisteme eklenebilir. 200 dolar aylık ücret ile geliştiricilerine deneyebilme olanağı sağlamaktadır. Google Maps API uygulaması kullanıcıların X noktasından Y noktasına giderken izleyecekleri yol ile rotalama işlemi sonucunda zaman ve maliyetten kazanç sağlamalarına yardımcı olabilir. Birden çok durağın mesafe hesaplama matrisi kullanarak birbirlerine uzaklıklarına göre bir sıralama yapabilir ve optimizasyon işlemi gerçekleştirilebilir.

Google Maps Api mesafe hesaplamayı Google Maps Distance Matrix API uygulaması ile gerçekleştirilir. Bu hizmet başlangıç ve varış noktalarından oluşan bir matris için güzergah mesafesi ve süresi sağlayan bir hizmettir. API, Google Maps API tarafından hesaplandığı şekliyle başlangıç ve bitiş noktaları arasındaki rotaya dayalı olarak bilgi verir. Zaman ve mesafeleri hesaplarken hangi ulaşım yönteminin kullanılacağını belirler. Bunlar: Bisiklet, bisiklet yolları ve tercih edilen sokaklar üzerinden bisiklet için yol tarifi alır. Araç varsayılandır ve yol ağını kullanan rotaları belirtir. Toplu taşıma, toplu taşıma rotalarını alır. Yürüme, yaya yolları ve kaldırım rotalarını alır [60].

0	459	903	1467	5056	5155	5699	4682	6480	6720	6256	11604	3982	4229	2474	2978	2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612
459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597	4207	7164	9610	10183	11148	11952	13795	14042	4808
903	482	0	595	3514	3612	4156	5248	7046	7286	6822	12170	2797	3044	1564	2068	1504	2460	2971	3581	7534	9979	10552	11518	12321	14164	14411	5178
1467	1047	595	0	2973	3590	4134	5733	7531	7771	9397	12655	2359	2606	593	1382	1481	2438	2533	3559	8019	10465	11038	12003	12807	14650	14896	7753
5056	4139	3514	2973	0	511	1212	11504	13303	13543	10301	15807	1015	498	2283	2160	1999	1352	484	131	13790	16236	16809	17774	18578	20421	16457	8656
5155	4238	3612	3590	511	0	817	12079	13878	14118	11233	16382	1590	1073	2858	2735	2639	1662	1059	471	14365	15767	11870	18349	19153	20996	17032	9589
5699	4782	4156	4134	1212	817	0	10041	11839	12079	11536	17608	2324	1807	3533	3520	2942	1965	1598	1205	12327	15417	15990	16956	17115	18958	19196	9892
4682	4878	5248	4134	11504	12079	10041	0	2991	4271	6971	8186	8746	11650	7809	7784	8444	8781	8920	9494	3479	5995	6568	7533	8337	10180	10427	5428
6480	6676	7046	7286	13303	13543	10301	2991	0	2067	2504	3478	8864	11769	7927	7903	8563	8899	9038	9612	1856	2717	3120	6763	7567	9410	9657	2987
6720	6917	6452	6917	10301	15807	1015	498	2283	0	1298	2272	10112	13016	9175	9150	9811	10147	10286	10860	838	1128	1914	2758	5068	6910	7157	2350
6256	11604	3982	4229	2474	2978	2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917
3982	4229	2474	2978	2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801
4229	2474	2978	2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422
2474	2978	2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670
2978	2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014
2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518
4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129
4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086
5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597
6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597	4207
9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597	4207	7164
9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597	4207	7164	9610
10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597	4207	7164	9610	10183
11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597	4207	7164	9610	10183	11148
13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597	4207	7164	9610	10183	11148	11952
13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597	4207	7164	9610	10183	11148	11952	13795
4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597	4207	7164	9610	10183	11148	11952	13795	14042
459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597	4207	7164	9610	10183	11148	11952	13795	14042	4808
903	482	0	595	3514	3612	4156	5248	7046	7286	6822	12170	2797	3044	1564	2068	1504	2460	2971	3581	7534	9979	10552	11518	12321	14164	14411	5178
1467	1047	595	0	2973	3590	4134	5733	7531	7771	9397	12655	2359	2606	593	1382	1481	2438	2533	3559	8019	10465	11038	12003	12807	14650	14896	7753
5056	4139	3514	2973	0	511	1212	11504	13303	13543	10301	15807	1015	498	2283	2160	1999	1352	484	131	13790	16236	16809	17774	18578	20421	16457	8656
5155	4238	3612	3590	511	0	817	12079	13878	14118	11233	16382	1590	1073	2858	2735	2639	1662	1059	471	14365	15767	11870	18349	19153	20996	17032	9589
5699	4782	4156	4134	1212	817	0	10041	11839	12079	11536	17608	2324	1807	3533	3520	2942	1965	1598	1205	12327	15417	15990	16956	17115	18958	19196	9892
4682	4878	5248	4134	11504	12079	10041	0	2991	4271	6971	8186	8746	11650	7809	7784	8444	8781	8920	9494	3479	5995	6568	7533	8337	10180	10427	5428
6480	6676	7046	7286	13303	13543	10301	2991	0	2067	2504	3478	8864	11769	7927	7903	8563	8899	9038	9612	1856	2717	3120	6763	7567	9410	9657	2987
6720	6917	6452	6917	10301	15807	1015	498	2283	0	1298	2272	10112	13016	9175	9150	9811	10147	10286	10860	838	1128	1914	2758	5068	6910	7157	2350
6256	11604	3982	4229	2474	2978	2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917
3982	4229	2474	2978	2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801
4229	2474	2978	2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422
2474	2978	2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670
2978	2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014
2644	4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518
4003	4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129
4641	5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086
5124	6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597
6968	9414	9987	10952	11756	13599	13845	4612	459	0	482	1047	4139	4238	4782	4878	6676	6917	6452	11801	3422	3670	2014	2518	2129	3086	3597	4

Yukarıda gösterilen Şekil 4.7- görüntüsü projemizde ifade edilen 27 noktaya ait X ve Y koordinatlarının belirlenerek Google Maps Distance Matrix işlemi sonucu oluşan tablosunun görüntüsüdür. Burada Google Maps Distance Matrix işlemi sonucu oluşan bu görüntüde verilerin simetrik olmamasının iki sebebi vardır. Bu sebeplerden birincisi bazı yolların tek yön olmasıdır bu durum gidilen yoldan geri dönülme olasılığının olmamasından kaynaklanmaktadır. İkinci sebep ise geliş ve gidiş esnasında trafik yoğunluğu farklı olması sebebiyle geliş ve gidiş rotalarında farklı yolların seçilmiş olmasıdır.

Google Maps Directions API, web sayfalarında ve mobil cihazlarda kendi içeriğiniz ve görüntülerinizle özelleştirmenize olanak tanıyarak harita görüntüleme işlemi gerçekleştirir. Haritalar, katmanları ve stilleri, kontrolleri ve çeşitli hizmetleri dört temel harita türü olan yol haritası, uydu haritası, karma harita ve arazi haritasını olarak içerir. Google Maps API ile adres bulabilmek için varlıkların, uydu görüntüsü Google Maps API for Directions üzerine işlenebilir ve araştırmalar gerçekleştirilebilir. Geocoding API uygulaması ile X ve Y koordinatından açık adresi il, ilçe, mahalle, cadde, sokak ve kapı numarasına kadar belirleme özelliği vardır. Bunların yanında açık adresten de X ve Y koordinatını elde etme yeteneğine de sahiptir. Çalışmada Google Maps Directions API uygulaması kullanılarak, Google Maps Distance Matrix sonucu elde edilen verilerin genetik algoritmada değerlendirilerek bu verilerin gerçek güzergâh olarak çizdirilmesi sağlanmıştır. Elde edilen rota çizimleri tek, iki ve üç araç için kapasite sorunu faktörleri de göz önünde bulundurularak test edilmiştir.

Google Maps API Key alma işlemi için <https://console.cloud.google.com> adresine girilip oturum açılmalıdır. API anahtarı, kullanım ve faturalandırma amacıyla çalışma ile ilişkili istekleri doğrulayan anahtardır. Çalışmanız da ilişkilendirilmiş en az bir API anahtarına sahip olmanız gerekmektedir. API Anahtarı oluşturmak için: Cloud Console'da, proje seçici sayfasında, API Anahtarı eklemek istediğiniz bir Google Maps projesi seçmeli veya oluşturmalısınız. Proje seçimi sayfasında: API'ler ve Hizmetler - Kimlik Bilgileri sayfasına gidilmelidir. Kimlik Bilgileri sayfasında, Kimlik bilgileri oluştur - API anahtarı seçenekleri belirlenmelidir. API anahtarı oluşturulan iletişim kutusu, yeni oluşturulan API anahtarınızı görüntüler. Yeni API anahtarı, API anahtarları altındaki Kimlik Bilgileri sayfasında listelenir. Bu esnada API anahtarını kısıtlanmalıdır. API Anahtarı ve faturalandırma hataları ise belirli koşullar altında, karartılmış bir harita veya yalnızca geliştirme amaçlı metni ile filigranlı negatif Sokak Görünümü resmi görüntülenebilir. Bu durum API anahtarı veya faturalandırma ile ilgili sorunları gösterir. Google Maps Platformu erişimi için, hesabınızda faturalandırmanın etkinleştirilmesi ve tüm isteklerde geçerli bir API anahtarı bulunması gerekir [60].

## 5. TEST SONUÇLARI VE DEĞERLENDİRMELER

Bu bölümde önerilen uygulama aşağıda özellikleri belirtilen bir bilgisayar sistemi üzerinde test edilmiştir. Sonuçların olumlu şekilde analiz edilebilmesi için kullanılan bilgisayar sisteminin özelliklerinin iyi bilinmesi gerekmektedir. Uygulamanın test edildiği bilgisayar sistemi özellikleri tablo 5.1'de gösterilmektedir.

Tablo 5.1- Test Platformu Özellikleri

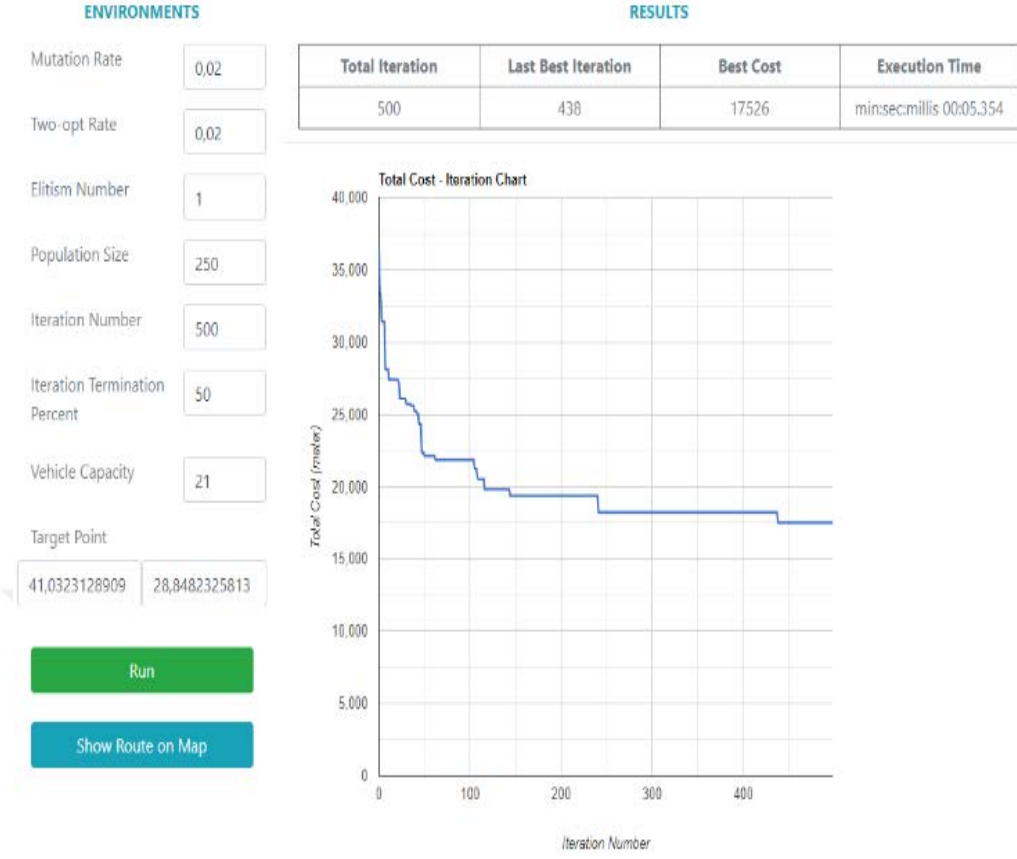
Bilgisayar	Toshiba Portege Z930-14L
İşletim Sistemi	Microsoft Windows 10 Pro
Merkezi İşlem Birimi	Intel® Core™ i7-3687U Central Processing Unit @ 2.1 GHz
Erişim Belleği	8 GB
Geliştirme Ortamı	mSATA 6Gbps Solid State

Önerilen sistemin performansı, doğrudan genetik algoritmanın parametrelerine bağlı olarak değişir ve sonuçlar, birden çok veri yolu için yararlı değerler üretir. Uygulamamızı farklı değişken parametre değerleri ile testlere tabi tuttuk. Elde edilen sonuçların ve grafiklerin tablo 5.2'de ifade edilen değerlerle doğrudan ilişkili olduğunu görüyoruz. Belirtilen değerlerdeki her bir değişiklik için sonuç değerlerinde belirgin farklılıkların olduğunda test sonuçlarında saptanmıştır.

Tablo 5.2- Genetik Algoritma Parametreleri

Parametre	Değer
Birincil Sonlandırma Kuralı (Yineleme Numarası)	500
İkincil Sonlandırma Kuralı (Yineleme Yüzdesi)	50
Anne ve Baba Seçimi	Turnuva Seçimi
Çaprazlama Tipi	Sipariş çaprazlama (OX1)
Mutasyon Oranı	0,02
2-Opt Oranı	0,02
Elitizm Sayısı	1
Popülasyon Boyutu	250

Uygulamanın bilgisayar ortamında web ara yüzünde kapasitesi 21 kişi alabilen bir araç ile çalıştırılması sonucunda elde edilen sonuçlardan biri aşağıda ifade edilmiştir. Bu uygulamadan elde edilen sonuçlar 21 farklı nokta, tek bir araç 500 iterasyonlu bir popülasyonda aşağıdaki gibi en iyi çözüm şeklinde gösterilmiştir. Sonuçların ilk bakışta en iyi çözüm olmadığı anlaşılır.

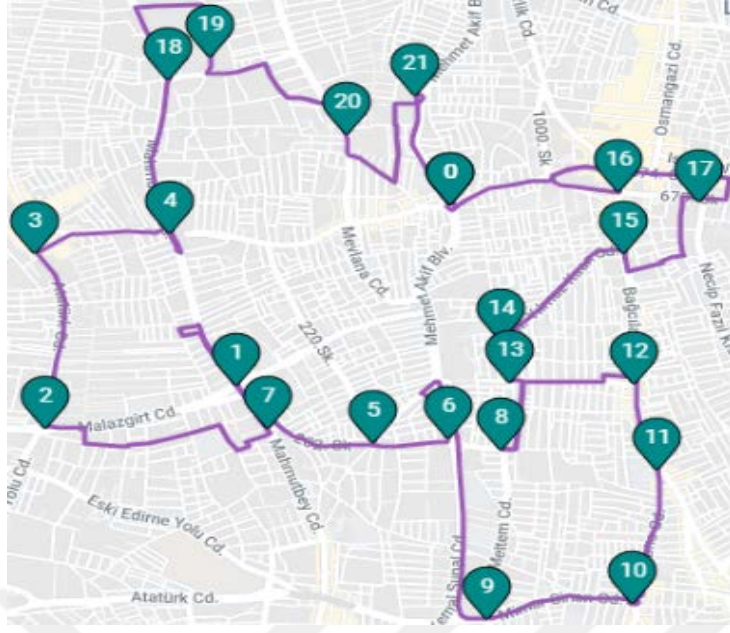


Şekil 5.1- Tek Araç İçin Toplam Mesafe İterasyon Grafiği

**Cost: 17526**

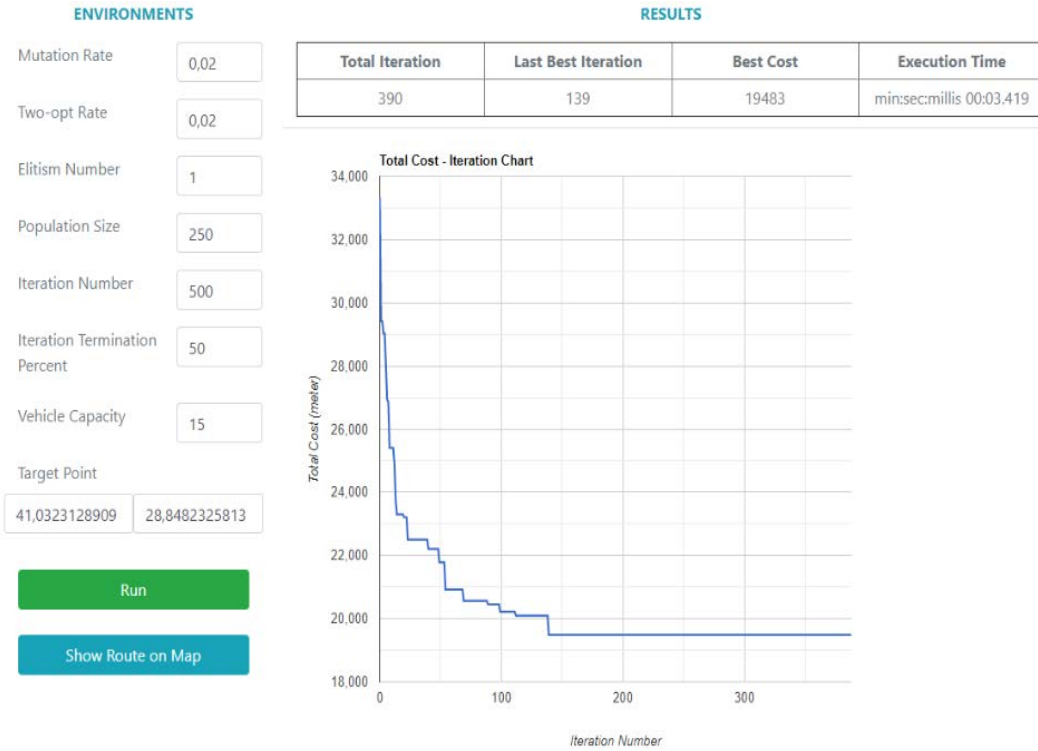
No	Latitude	Longitude
0	41,0323128909	28,8482325813
16	41,0328592378	28,8564358199
17	41,0324494706	28,8604168506
15	41,0300836588	28,8566768185
14	41,026170757	28,8507054574
13	41,0242597308	28,8511276269
8	41,0212113545	28,8507054369
12	41,0242140332	28,8572190502
11	41,0203464029	28,8583042005
10	41,0142948345	28,857097467
9	41,0135670175	28,8499817869
6	41,0217573162	28,8481122585
5	41,0215751778	28,8444335708
1	41,0241681671	28,8377394723
7	41,0221662593	28,8392470844
2	41,022164758	28,8283922809
3	41,0300813306	28,8279071525
4	41,0309923814	28,8344811913
18	41,0379538752	28,8344206092
19	41,0389549157	28,8364712635
20	41,0354974953	28,8431062202
21	41,0372262833	28,8464838785
0	41,0323128909	28,8482325813

Şekil 5.2- Tek Araca Ait Noktaların Enlem ve Boylam Konumları



Şekil 5.3- Tek Araca Ait Sonuçların Google Maps Uygulaması İle Gösterilmesi

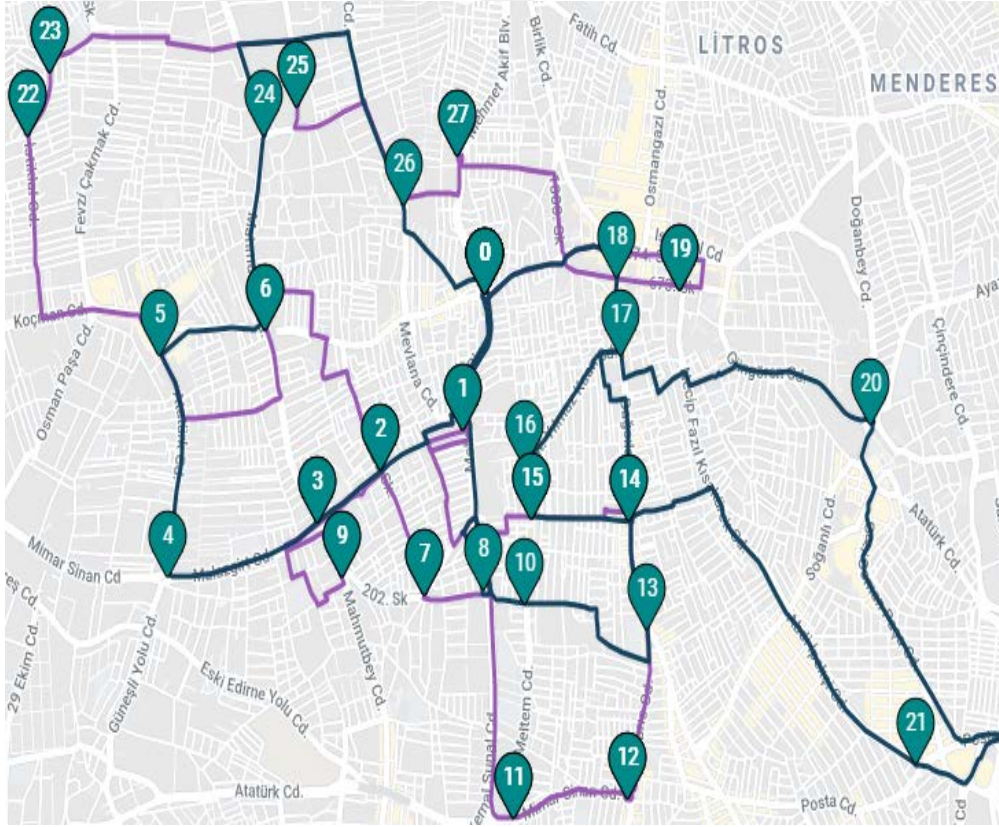
Uygulamanın bilgisayar ortamında web ara yüzünde kapasitesi 15 kişi alabilen iki araç ile çalıştırılması sonucunda elde edilen sonuçlardan biri aşağıda ifade edilmiştir. Bu uygulamadan elde edilen sonuçlar 27 farklı nokta, iki farklı araç 500 iterasyonlu bir popülasyonda aşağıdaki gibi en iyi çözüm şeklinde gösterilmiştir. Buradan, tek araç ile üretilen optimizasyon sonuçlarının iki araç ile üretilen optimizasyon sonuçlarına göre daha iyi olduğu gözlenmektedir.



Şekil 5.4- İki Araç İçin Toplam Mesafe İterasyon Grafiği

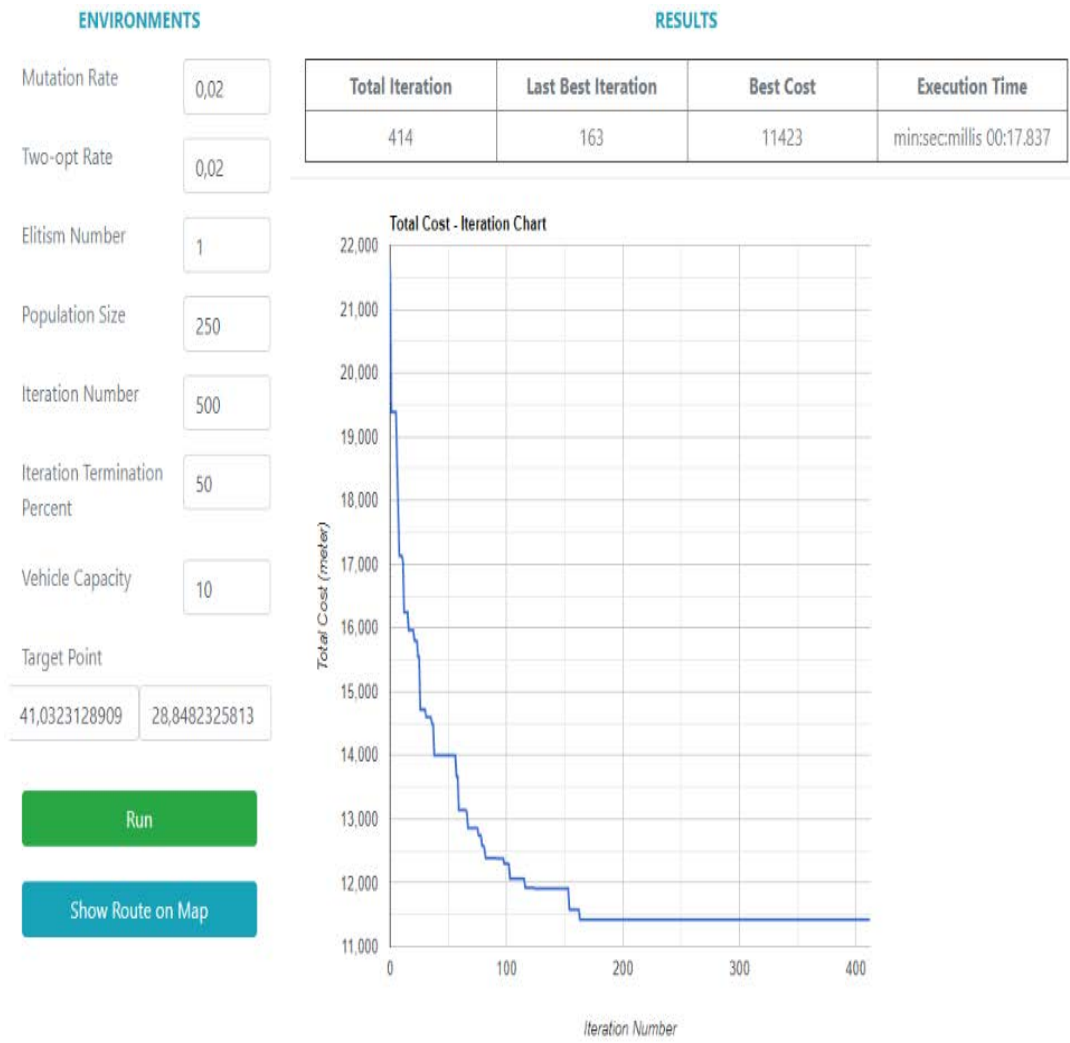
Cost: 19477			Cost: 19483		
No	Latitude	Longitude	No	Latitude	Longitude
0	41,0323128909	28,8482325813	0	41,0323128909	28,8482325813
19	41,0324494706	28,8604168506	4	41,022164758	28,8283922809
27	41,0372262833	28,8464838785	5	41,0300813306	28,8279071525
25	41,0389549157	28,8364712635	24	41,0379538752	28,8344206092
23	41,040226907	28,820968822	26	41,0354974953	28,8431062202
22	41,0379518361	28,8196426159	18	41,0328592378	28,8564358199
6	41,0309923814	28,8344811913	16	41,026170757	28,8507054574
2	41,0259884516	28,8416592066	21	41,0155207331	28,8751891353
3	41,0241681671	28,8377394723	20	41,0276247597	28,8723586427
9	41,0221662593	28,8392470844	17	41,0300836588	28,8566768185
7	41,0215751778	28,8444335708	13	41,0203464029	28,8583042005
15	41,0242597308	28,8511276269	10	41,0212113545	28,8507054369
14	41,0242140332	28,8572190502	8	41,0217573162	28,8481122585
12	41,0142948345	28,857097467	0	41,0323128909	28,8482325813
11	41,0135670175	28,8499817869			
1	41,0274446581	28,846845645			
0	41,0323128909	28,8482325813			

Şekil 5.5- İki Araca Ait Noktaların Enlem ve Boylam Konumları



Şekil 5.6- İki Araca Ait Sonuçların Google Maps Uygulaması İle Gösterilmesi

Uygulamanın bilgisayar ortamında web ara yüzünde kapasitesi 10 kişi alabilen üç araç ile çalıştırılması sonucunda elde edilen sonuçlardan biri aşağıda ifade edilmiştir. Bu uygulamadan elde edilen sonuçlar 27 farklı nokta, üç farklı araç 500 iterasyonlu bir popülasyonda aşağıdaki gibi en iyi çözüm şeklinde gösterilmiştir. Buradan üç araç ile üretilen optimizasyon sonuçlarının iki araç ile üretilen optimizasyon sonuçlarına göre daha iyi olduğu gözlenmektedir. Bir, iki ve üç araç ile yapılan çalışmalar neticesinde en iyi sonucun üç araç ile gerçekleştiği gösterilmiştir.



Şekil 5.7- Üç Araç İçin Toplam Mesafe İterasyon Grafiği

**Cost: 11423**

No	Latitude	Longitude
0	41,0323128909	28,8482325813
1	41,0274446581	28,846845645
3	41,0241681671	28,8377394723
9	41,0221662593	28,8392470844
4	41,022164758	28,8283922809
5	41,0300813306	28,8279071525
22	41,0379518361	28,8196426159
23	41,040226907	28,820968822
25	41,0389549157	28,8364712635
6	41,0309923814	28,8344811913
0	41,0323128909	28,8482325813

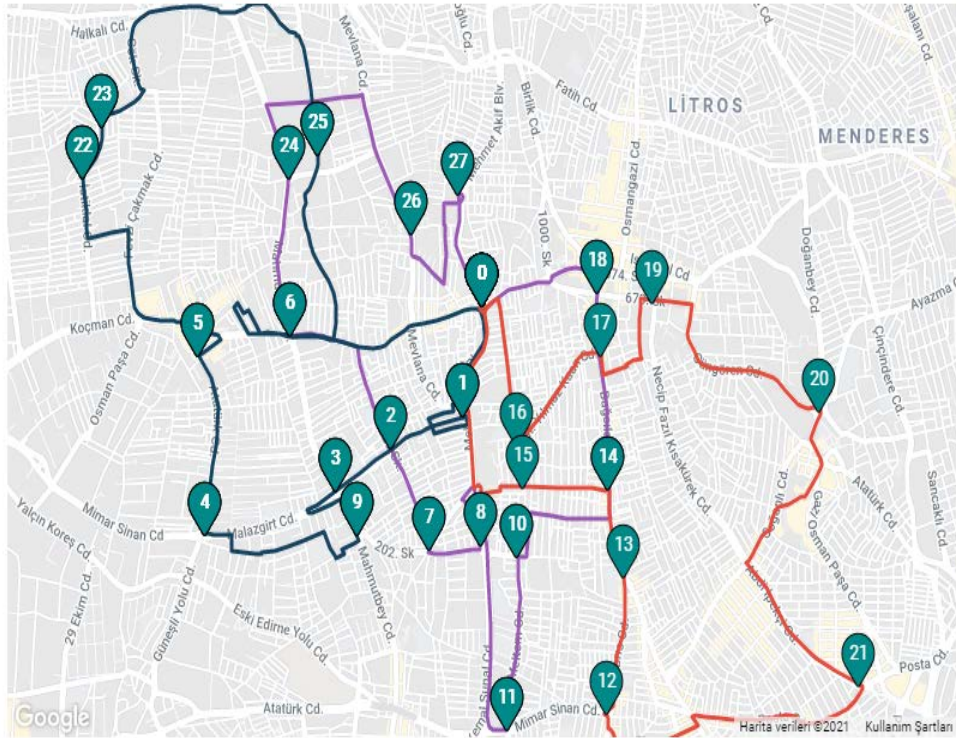
**Cost: 11271**

No	Latitude	Longitude
0	41,0323128909	28,8482325813
18	41,0328592378	28,8564358199
14	41,0242140332	28,8572190502
10	41,0212113545	28,8507054369
11	41,0135670175	28,8499817869
8	41,0217573162	28,8481122585
7	41,0215751778	28,8444335708
2	41,0259884516	28,8416592066
24	41,0379538752	28,8344206092
26	41,0354974953	28,8431062202
27	41,0372262833	28,8464838785
0	41,0323128909	28,8482325813

**Cost: 10796**

No	Latitude	Longitude
0	41,0323128909	28,8482325813
16	41,026170757	28,8507054574
13	41,0203464029	28,8583042005
12	41,0142948345	28,857097467
21	41,0155207331	28,8751891353
20	41,0276247597	28,8723586427
19	41,0324494706	28,8604168506
17	41,0300836588	28,8566768185
15	41,0242597308	28,8511276269
0	41,0323128909	28,8482325813

Şekil 5.8- Üç Araca Ait Noktaların Enlem ve Boylam Konumları



Şekil 5.9- Üç Araca Ait Sonuçların Google Maps Uygulaması İle Gösterilmesi

Yapılan arařtırmalar sonucunda evrimsel bir yaklařım olan genetik algoritmanın rota optimizasyon srelerinde sunduėu sonu zm, her tekrarlandığında daha iyi sonular vermektedir. Bu uygulamada 3 farklı veri yolu mevcuttur ve 500 iterasyondan sonra en iyi zm yukarıdaki Őekilde gsterilmiřtir. Buradan da anlařılacaėı gibi, retilen sonular ilk bakıřta en iyi zm olarak grlemez. Ancak sistemin yrtlmesinde gerek zamanlı trafik deėerlerinin kullanılması hedeflenmektedir. Bu nedenle, trafik sıklıřıklıėına baėlı olarak nerilen algoritmalar farklı yollar nerebilmektedir. Bazı yollar tek yn olarak kullanılmaktadır ve bu da en iyi grnen yolların doėrudan deėiřtirilmesine yol amaktadır. Grafiklerde daha iyi alternatiflerin olduėu grlmesine raėmen elde edilen sonular sadece trafik sıklıřıklıėına baėlı deėildir. Rota optimizasyon iřlemlerinde yolların ynne de optimizasyon bařarısı iin nemlidir. Problemden optimizasyonu saėlayabilmek amacıyla yapay zeka tekniėi olarak nitelendirilen metasezgisel yntemlerden biri olan genetik algoritmalar kullanılmıřtır. Google Maps Distance Matrix API uygulamasından yararlanılarak noktalar arası mesafeler anlık olacak Őekilde alınmıřtır. Rota izdirmek maksadıyla Google Maps Directions API uygulamasından yararlanılmıřtır. C# dilinde .Net ortamında geliřtirilen uygulama ile rota optimizasyonu saėlanmıřtır.

## 6. SONUÇLAR

Büyük şehirlerin karmaşık yapısından kaynaklı çok sayıda sorun bulunmaktadır. Şehirlerdeki sorunların çözülmeye çalışılması sonucunda ortaya çıkan akıllı şehir kavramı, şehirlerin çeşitli alanlardaki sorunlarını çözebilir. Değişen ve gelişen teknoloji ile birlikte ulaşım olanaklarına bağlı olarak yol çeşitliliği ve ulaşım vasıta araçları da değişmiş buna bağlı olarak yol optimizasyonu sağlanabilmesi adına rota problemleri kavramı ortaya çıkmıştır. Seyahat öncesi ve esnasında en doğru rota seçimi zaman kayıplarının ve maliyetlerinin düşürülmesi konusunda ciddi yaralar sağlamaktadır.

Son yıllarda artan yaşam kalitesi nedeniyle kentsel alanlara, özellikle akıllı şehirlere göç etme eğilimi artmaktadır. Bu şehirlerde pek çok gerçek dünya sorunu çözülmüş olsa da, hala bazı aktif araştırma alanları bulunmaktadır. Bu çalışmada, genetik algoritma olarak evrimsel bir algoritma kullanılarak okul otobüslerinin rota planlamasının optimize edilmesi amaçlanmıştır. Bu kullanımda Google Maps navigasyon programı kullanılarak gerçek trafik değerlerinin kullanılması hedeflenmekte olup, üretilen sonuç dinamik rota planlaması için sürücüye aktarılacaktır.

Çalışma sonuçları, önerilen modelin okul hizmetleri için birden fazla otobüs için kabul edilebilir sonuçlar ürettiğini belirlemiştir. İleride yapılacak çalışmalarda daha çok ziyaret noktası eklenerek ve daha fazla otobüs kullanılarak sistemin verimliliğinin artırılması hedeflenmektedir.

## KAYNAKÇA

- [1] J. Bramel, J. Braca, D. Simchi-Levi and B. Posner, A Computerized Approach to the New York City School Bus Routing Problem, IIE Trans. (Institute Ind. Eng., pp. 693-702, 1997.
- [2] J. Kelly and J. Xu, A Network Flow-Based Tabu Search Heuristic for the Vehicle Routing Problem, Transportation Science, [Online].Erişilebilirlik: <https://doi.org/10.1287/trsc.30.4.379/>, [Erişim Tarihi 01 Kasım 1996].
- [3] R. Krishnamurti and A. Gupta, Parallel Algorithms For Vehicle Routing Problems, Parallel Processing Letters, pp. 673-687, 2003.
- [4] M. L. Fisher, Optimal Solution of Vehicle Routing Problems using Minimum k-trees, Operations Research, pp. 626-642, 1994.
- [5] G. Dantzig, S. Johnson and R. Fulkerson, Solution of a Large-Scale Travelling Salesman Problem, Journal of Operations Research Society, pp. 393-410, 1954.
- [6] G. Clarke, and J. W. Wright, Scheduling of Vehicles From a Central Depot to a Number of Delivery Points, Operations Research, pp. 568-581, 1964.
- [7] C. Gencer ve M. Eryavuz, Araç Rotalama Problemine Ait Bir Uygulama, Süleyman Demirel Üniversitesi, İktisadi ve İdari Bilimler Fakültesi, sf. 139-155, 2001.
- [8] O. Çetin, Akaryakıt Dağıtımında Araç Rotalama Problemi, İstanbul Üniversitesi, Sosyal Bilimler Enstitüsü, Doktora Tezi, İstanbul, 2013.
- [9] K. Çalışkan, Karınca Kolonisi Optimizasyonu İle Araç Rotalama Probleminin Maliyetlerinin Kümeleme Tekniği İle İyileştirilmesi, TOBB Ekonomi ve Teknoloji Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Ankara, 2011.
- [10] T., Eren, H. Güvez ve M. Dege, Kırıkkale'de Araç Rotalama Problemi İle Tıbbi Atıkların Toplanması, International Journal of Engineering Research and Development, sf. 41-45, 2012.
- [11] E. Demir, Okul Taşıtları Rotalama Problemi için Tamsayı Karar Modelleri, Yüksek Lisans Tezi, Başkent Üniversitesi, 2008.

- [12] W. H. Thomas and R. M. Newton Design of School Bus Routes by Computer, Socioecon, Plann, Sci., pp. 75-85, 1969.
- [13] B. I. Kim and J. Park, The School Bus Routing Problem: A Review, Eur. J. Oper. Res., pp. 311-319, 2009.
- [14] N. Ben Guedria, S. Ben Sghaier and R. Mraïhi, Solving School Bus Routing Problem with Genetic Algorithm, in 2013 International Conference on Advanced Logistics and Transport, ICALT 2013, pp. 7-12, 2013.
- [15] Z. Fu and L.Y.O. Li, The School Bus Routing Problem: A Case Study. Journal of the Operational Research Society, pp. 552-558, 2002.
- [16] J. M. Viegas and L. M. Martinez, Design and Deployment of an Innovative School Bus Service in Lisbon, Predictors, Procedia - Social and Behavioral Sciences, pp. 120-130, 2011.
- [17] M. F. Demiral, Servis Araçlarının Rotalamasında Optimizasyon ve Bir Uygulama, Süleyman Demirel Üniversitesi, Sosyal Bilimleri Enstitüsü, Yüksek Lisans Tezi, Isparta, 2008.
- [18] S. Gönüloğlu, Gezgin Satıcı Problemi için Veri Madenciliği Tabanlı Sezgisel Bir Yaklaşım, Yüksek Lisans Tezi, Kocaeli Üniversitesi Fen Bilimleri Enstitüsü, Endüstri Mühendisliği Anabilim Dalı, Kocaeli, 2009.
- [19] A. P. Punnen and G. Gutin, The Travelling Salesman Problem: Applications, Formulations and Variations, Kluwer Academic Publishers, 2002.
- [20] N. Christofides, Technical Note-Bounds for the Travelling-Salesman Problem, Operations Research, pp. 1044-1056, 1972.
- [21] R. M. Karp and M. Held, The Traveling Salesman Problem and Minimum Spanning Trees, Operations Research, pp. 1138-1162, 1970.
- [22] G. Rinaldi and M. Padberg, Optimization of a 532-city Symmetric Traveling Salesman Problem by Branch and Cut, Operations Research Letters, pp. 1-7, 1987.
- [23] K. Alaykırın, ve O. Engin, Karınca Kolonileri Metasezgiseli ve Gezgin Satıcı Problemleri Üzerinde Bir Uygulaması, Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, sf. 69-76, 2005.

- [24] S. Çolak, Genetik Algoritmalar Yardımı ile Gezgin Satıcı Probleminin Çözümü Üzerine Bir Uygulama, Ç.Ü. Sosyal Bilimler Enstitüsü Dergisi, sf. 423-438, 2010.
- [25] R. Gerhard, The Traveling Salesman: Computational Solutions for TSP Applications, [Online]. Erişilebilirlik: <https://dl.acm.org/doi/book/10.5555/1744275/>, [Erişim Tarihi Ocak 1994].
- [26] C. B. Li and K. Z. Zhou, How Strategic Orientations Influence the Building of Dynamic Capability in Emerging Economies, Journal of Business Research, pp. 224-231, 2010.
- [27] M. C. Chen, and S. H. Chen, Operators of the Two-Part Encoding Genetic Algorithm in Solving the MTSP, [Online]. Erişilebilirlik: <https://www.researchgate.net/publication/254018201/>, [Erişim Tarihi Kasım 2011].
- [28] D. N. Mahmoodi, M. Sedighpour and M. Yousefikhoshbakht, An Effective Genetic Algorithm for Solving the MTSP, [Online]. Erişilebilirlik: <https://www.researchgate.net/publication/287817168/>, [Erişim Tarihi Mart 2011].
- [29] B. Al-Khateeb and M. Yousif, A Novel Metaheuristic Algorithm for Multiple Traveling Salesman Problem, [Online]. Erişilebilirlik: <https://www.researchgate.net/publication/331345138/>, [Erişim Tarihi Ocak 2018].
- [30] M. Xu, J. Guo, and S., Li, Optimization of Multiple Traveling Salesman Problem Based on Simulated Annealing Genetic Algorithm, MATEC Web of Conferences, pp. 100, 2017.
- [31] E. C. Arthur, and T. R. Cliff, A New Approach to Solving the Multiple Traveling Salesperson Problem Using Genetic Algorithms, European Journal of Operational Research, pp. 246-257, 16 Ocak 2006,
- [32] M. Demircioğlu, Araç Rotalama Probleminin Sezgisel bir Yaklaşım ile Çözümlemesi Üzerine Bir Uygulama, Doktora Tezi, TC Çukurova Üniversitesi Sosyal Bilimler Enstitüsü, İşletme Anabilim Dalı, Adana, 2009.
- [33] V. Erol, Araç Rotalama Problemleri için Popülasyon ve Komsuluk Tabanlı Metasezgisel Bir Algoritmanın Tasarımı ve Uygulaması, Yıldız Teknik Üniversitesi, Sistem Mühendisliği, Yüksek Lisans Tezi, Sabancı Kütüphanesi, İstanbul, 2006.

- [34] M. Solomon, J. Cordeu, G. Desrosiers and F. Sourmis, The VRP with Time Windows, Society for Industrial and Applied Mathematics, Philadelphia, Society for Industrial and Applied Mathematics, Philadelphia, 2002.
- [35] A. Larsen, The Dynamic Vehicle Routing Problem, Ph.D. thesis, Department of Mathematical Modelling, Technical University of Denmark, 2001.
- [36] M. Gendreau, V. Pillac, C. Gueret, and A. L. Medaglia, A Review of Dynamic Vehicle Routing Problems, *European Journal of Operational Research*, pp. 1–11, 2013.
- [37] L. Bianchi, Notes on Dynamic Vehicle Routing The State of the Art Technical Report, IDSIA, 2000.
- [38] O. Marcotte, A. Hadjar and F. Soumis, A Branch-and-Cut Algorithm for the Multiple Depot Vehicle Scheduling Problem, *Oper. Res.*, pp. 130-149, 2006.
- [39] A. Fügenschuh, Solving a School Bus Scheduling Problem with Integer Programming, *Eur. J. Oper. Res.*, pp. 867-884, 2009.
- [40] S. Jain and Y. Kumar, School Bus Routing Based on Branch and Bound Approach, in *IEEE International Conference on Computer Communication and Control, IC4*, 2006.
- [41] V. V. Nabyev, *Yapay Zeka: Problemler, Yöntemler, Algoritma*, Seçkin Yayınevi, Ankara, 2003.
- [42] C. D. Gelatt, S. Kirkpatrick and M. P. Vecchi, Optimization by Simulated Annealing, *Science*, pp. 671-680, 1983.
- [43] M. Bierlaire, M. Spada and T. M. Liebling, Decision-Aiding Methodology for the School Bus Routing and Scheduling Problem, *Transp. Sci.*, pp. 477-490, 2005.
- [44] M. M. Paydar, S. S. Dana and J. Jouzdani, A simulated annealing solution method for robust school bus routing, *Int. J. Oper. Res.*, pp. 307, 2017.
- [45] M. S. Hayalioglu, S. Ö. Degertekin ve M. Ulker, Uzay Çelik Çerçevelerin Tabu Arama ve Genetik Algoritma Yöntemleriyle Optimum Tasarımı, *İMO Teknik Dergi*, Yazı 259, sf. 3917-3934, 2006.
- [46] M. Laguna and F. Glover, *Modern Heuristics Technics For Combinatorial Problems; Tabu Search*. Blackwell Scientific Publications, pp. 364s, 1993.

- [47] M. Özmen, H. Şahin and O. K. Şahingöz, Genetic Algorithm Based Optimized Waste Collection in Smart Cities, 2020 International Conference on Electrical Engineering (ICEE), [Online].Erişilebilirlik: 10.1109/ICEE49691.2020.9249837/, [Erişim Tarihi 25-27 Eylül 2020].
- [48] J. H. Holland, Adaptation in Natural and Artificial Systems, University of Michigan, 1975.
- [49] C. H. Wang and J. Z. Lu, A Hybrid Genetic Algorithm that Optimizes Capacitated Vehicle Routing Problems, Expert Syst. Appl., pp. 2921-2936, 2009.
- [50] B. Gendreau, G. Laporte, J. Potvin and F. Semet, Classical and Modern Heuristics for the Vehicle Routing Problem, International Transactions, in Operational Research, pp. 285-300, 2000.
- [51] Kennedy, J. and Eberhart, R.C. Particle swarm Optimization, Proceedings of the IEEE International Conference on Neural Networks, pp. 1942- 1948, ISBN: 0780327683, Perth, Western Australia, 1995.
- [52] T. Cura, A particle swarm optimization approach to clustering, Expert Syst. Appl., pp. 1582-1588, 2012.
- [53] I. Loiseau, and S. Mazzeo, An Ant Colony Algorithm for the Capacitated Vehicle Routing, Electronic Notes in Discrete Mathematics, pp. 181–186, 2004.
- [54] P. Abad and Y. Gajpal, An ant colony system (ACS) for vehicle routing problem Problem with Simultaneous Delivery and Pickup, 2009.
- [55] M. Dorigo, and L. M. Gambardella, An Ant Colony System Hybridized with a New Local Search for the Sequential Ordering Problem, INFORMS Journal on Computing, pp. 237-255, 2000.
- [56] I. Anagnostopoulos, N. V. Karadimas, V. Loumos and G. Kouzas, Urban Solid Waste Collection and Routing: the Ant Colony Strategic Approach, I.J. of Simulation, pp. 45-53, 2006.
- [57] K. Helsgaun, An Effective Implementation of the Lin-Kernighan Traveling Salesman Heuristic, European Journal of Operational Research, pp. 106-130, 2000.

[58] A. Carter, Design and Application of Genetic Algorithms for the Multiple Traveling Salesperson Assignment Problem, Doctoral Dissertation, Virginia Polytechnic Institute and State Uni., Virginia, 2003.

[59] Michael Vivet, Vivet/GoogleApi, 2018.

[60] Başarsoft, [Online].Erişilebilirlik: <https://www.basarsoft.com.tr/google-maps-platform-api-nedir/>, 2020.

