

**T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**HETEROJEN FİLOLU ARAÇ ROTALAMA PROBLEMİNİN
META SEZGİSEL HİBRİT MODEL İLE ÇÖZÜMÜ**

SELİM EROĞLU

**YÜKSEK LİSANS TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI
ENDÜSTRİ MÜHENDİSLİĞİ PROGRAMI**

**DANIŞMAN
DOÇ. DR. ALEV TAŞKIN GÜMÜŞ**

İSTANBUL, 2015

T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**HETEROJEN FİLOLU ARAÇ ROTALAMA PROBLEMİNİN
META SEZGİSEL HİBRİT MODEL İLE ÇÖZÜMÜ**

Selim EROĞLU tarafından hazırlanan tez çalışması 03.07.2015 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Endüstri Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Tez Danışmanı

Doç. Dr. Alev TAŞKIN GÜMÜŞ
Yıldız Teknik Üniversitesi

Jüri Üyeleri

Doç. Dr. Alev TAŞKIN GÜMÜŞ
Yıldız Teknik Üniversitesi

Yrd. Doç. Dr. Nezir AYDIN
Yıldız Teknik Üniversitesi

Yrd. Doç. Dr. Berk AYVAZ
İstanbul Ticaret Üniversitesi

ÖNSÖZ

Tez çalışmamda yürütmüş olduğum proje, uzun yıllar lojistik sektöründe çalışan biri olarak sektörde gözlemlediğim ana sorunlardan birisi olan araç rotalama planlaması yapan firmaların karşılarına çıkan verimsiz işler, kişiye bağlı çalışma, zaman kaybı ve gereksiz maliyetler üzerine odaklanmamı sağlamıştır. Ana temel kısıtlar ortaya çıkarılarak kullanıcılara en azından bir öneri mekanizması ortamı oluşturulmuştur.

Çalışmamda her zaman yanımda olan, yönlendiren, kolaylık sağlayan değerli hocam Doç.Dr. Alev TAŞKIN GÜMÜŞ'e, üst yöneticim Sn. İbrahim BEKTAŞ'A, kıymetli eşim Kübra EROĞLU'ya, her zaman bana destek veren aileme, bilgi sağlamamda yardımcı olan iş ve okul arkadaşlarıma teşekkürlerimi sunarım.

Saygılarımla.

Selim EROĞLU

Mayıs, 2015

İÇİNDEKİLER

	Sayfa
SİMGE LİSTESİ.....	vii
KISALTMA LİSTESİ.....	viii
ŞEKİL LİSTESİ.....	ix
ÇİZELGE LİSTESİ	xi
ÖZET	xii
ABSTRACT.....	xiii
BÖLÜM 1	
GİRİŞ	1
1.1 Literatür Özeti	1
1.2 Tezin Amacı	3
1.3 Hipotez/Öneri	4
1.3.1 Çalışmanın Aşamaları.....	5
BÖLÜM 2	
LOJİSTİK VE LOJİSTİK YÖNETİMİ	6
2.1 Lojistik Kavramı ve Temel Unsurları.....	6
2.2 Lojistiğin Tanımı ve Gelişim Süreci.....	8
2.3 Lojistik Sistemindeki Problemler.....	12
2.3.1 Dağıtım Ağı Konfigürasyon Problemi	12
2.3.2 Üretim Planlaması Problemi	13
2.3.3 Stok Kontrol	13
2.3.4 Stok ve Dağıtımın Entegrasyonu	13
2.3.5 Araç Filosu Yönetimi	14
2.3.6 Araç Rotalama.....	14
2.3.7 Paketleme Problemi.....	14

2.3.8	Zaman Penceresiyle Dağıtım.....	15
2.3.9	Toplamalı Dağıtım Sistemleri	15
2.4	Lojistik Problemlerini Modelleme	15
BÖLÜM 3		
ARAÇ ROTALAMA PROBLEMLERİ		17
3.1	Araç Rotalama Problemi Tanımı	17
3.2	Araç Rotalama Problemi Temel Bileşenleri	18
3.3	Araç Rotalama Probleminin Prensipleri.....	19
3.4	Araç Rotalama Probleminin Amaçları	20
3.5	Araç Rotalama Problemi Türleri.....	20
3.6	Araç Rotalama Problemleri Çözüm Yöntemleri	22
3.6.1	Kesin Yöntemler	22
3.6.1.1	Dal Sınır Algoritması	22
3.6.1.2	Dal ve Kesme Algoritması	22
3.6.1.3	Küme Kapsama Algoritması	23
3.6.2	Klasik Sezgisel Yöntemler	23
3.6.2.1	Tasarruf Modeli	23
3.6.2.2	Süpürme Yöntemi	24
3.6.2.3	En Kısa Yol Yöntemi	25
3.6.2.4	İki Aşamalı Yöntem	26
3.6.3	Meta sezgisel Yöntemler.....	27
3.6.3.1	Genetik Algoritmalar	28
3.6.3.2	Bulanık Mantık.....	29
3.6.3.3	Uzman Sistemler.....	30
3.6.3.4	Yapay Sinir Ağları	31
3.6.3.5	Karınca Kolonisi	32
3.6.3.6	Tabu Arama	34
BÖLÜM 4		
UYGULANACAK YÖNTEMİN BELİRLENMESİ		35
4.1	Araç Rotalama Problemi İçin Uygun Bir Sezgisel Algoritmanın Özellikleri	35
4.2	Meta sezgisel Yöntemlerin Kıyaslanması	38
4.2.1	Bulanık Mantık Algoritması Seçilmeme Nedenleri	38
4.2.2	Uzman Sistemler Seçilmeme Nedenleri	40
4.2.3	Yapay Sinir Ağları Seçilmeme Nedenleri	44
4.2.4	Karınca Kolonisi Algoritması Seçilmeme Nedenleri	48
4.2.5	Tabu Arama Algoritması Seçilmeme Nedenleri	50
4.3	Uygulamada Kullanılacak Yöntemler	52

4.3.1	Clarke and Wright Tasarruf Algoritması	52
4.3.1.1	Tasarruf Yönteminin Aşamaları	55
4.3.2	Genetik Algoritma	56
4.3.2.1	Genetik Algoritmaya Giriş.....	56
4.3.2.2	Genetik Algoritma Terminolojisi ve Akış Algoritması	58
4.3.2.3	Basit Genetik Algoritma	60
4.3.2.4	Genetik Algoritma Parametreleri	67
BÖLÜM 5		
LOJİSTİK FİRMA UYGULAMASI		70
5.1	Mevcut Durum	70
5.1.1	Problemin Verileri	71
5.2	Problemin Tanımı	74
5.3	Problemin çözümünde kullanılan ek ana veriler	76
5.3.1	Süt toplama noktalarının koordinatlarının ve mesafelerinin belirlenmesi.	76
5.3.2	Farklı Tipteki Araçların Tanımlanması.....	76
5.3.3	Tırın Girip Giremeyeceği Noktaların Belirlenmesi	76
5.4	Maliyet Hesaplaması.....	77
5.5	Mevcut Operasyonun Rotalama Sonuçları	78
5.6	Problemin Matematiksel Modeli ve GAMS ile Çözümü.....	80
5.2.1	Problemin Matematiksel Modeli	80
5.6.1	Matematiksel Modelin GAMS’de Gösterimi.....	82
5.6.2	Gams sonuçları.....	84
5.7	Problemin Hibrit Model ile Çözümü, Uygun Rotaların Belirlenmesi	86
5.7.1	Hibrit Modelin İş Akış Diyagramı	87
5.7.2	Hibrit Programlama	89
5.7.3	Hibrit Programlama Sonuçları.....	90
5.7.4	Hibrit Modelin Diğer Sistemler ile Karşılaştırılması	92
BÖLÜM 6		
SONUÇ VE ÖNERİLER		93
KAYNAKLAR		95
EK-A		
NOKTALARARASI MESAFE MATRİS TABLOSU		99
EK-B		
HİBRİT MODELİN ROTA GÖRÜNTÜLERİ		99
ÖZGEÇMİŞ		107

SİMGE LİSTESİ

i	Çıkış noktaları indeksi (süt toplama merkezi veya süt alım noktası)
j	Variş noktaları indeksi(süt toplama merkezi veya süt alım noktası)
k	Araç tipi indeksi(Tır, Kamyon, Kırkayak)
d_{ij}	i noktasından j noktasına olan km mesafe değeri
c_k	k aracının km başına taşıma yakıt litre oranı
f_k	Sefere çıkan k aracının sabit maliyeti
y_{TL}	Yakıt litre birim maliyeti
mk_{km}	Aracın yapabileceği maksimum km sınır değeri
u_j	j noktasından alınması gereken süt miktarı
Q_k	k aracının süt alım kapasitesi
M	Toplam araç sayısı
N	Toplam süt alım noktası sayısı

KISALTMA LİSTESİ

ARP	Araç Rotalama Problemi
BM	Bulanık Mantık
DRP	Dağıtım Kaynakları Planlaması
GA	Genetik Algoritma
GSP	Gezgin Satıcı Problemi
JIT	Tam Zamanlı Üretim
KKA	Karınca Kolonisi Algoritması
KKARP	Kapasite Kısıtlı Araç Rotalama Problemi
Km	Kilometre
LT	Litre
MRP	Malzeme İhtiyaç Planlama
QAP	Kuadratik Atama
SCOOP	Optimum Programların Bilimsel Hesaplanması
TA	Tasarruf Algoritması
TAA	Tabu Arama Algoritması
TL	Türk Lirası
US	Uzman Sistemler
YSA	Yapay Sinir Ağları

ŞEKİL LİSTESİ

	Sayfa
Şekil 2. 1 Lojistiğin Temel Unsurları [5].....	7
Şekil 3. 1 Araç Rotalama Probleminin Şekilsel Gösterimi [11].....	17
Şekil 3. 2 Tasarruf Yöntemindeki Müşteri Birleştirilmesi.....	24
Şekil 3. 3 Süpürge Algoritması Adımları.....	25
Şekil 3. 4 Gerçek karıncaların en kısa yolu bulması	33
Şekil 4. 1 Tasarruf Algoritmasına ait örnek.....	54
Şekil 4. 2 Tasarruf Yönteminin Aşamaları[61].....	55
Şekil 4. 3 Genetik algoritmada popülasyon yapısı	59
Şekil 4. 4 Genetik Algoritma Akış Diyagramı.....	60
Şekil 4. 5 Kromozomların uygunluk değerine göre sıralanması.....	61
Şekil 4. 6 Kromozomların Uyumluluk değerine göre ağırlıkları	62
Şekil 4. 7 Rulet çarkına göre seçilme.....	62
Şekil 4. 8 Tek kesimli çaprazlama.....	64
Şekil 4. 9 Çift kesimli çaprazlama	65
Şekil 4. 10 Çok kesimli çaprazlama	65
Şekil 4. 11 Tekdüze çaprazlama	66
Şekil 4. 12 Mutasyon Örneği.....	67
Şekil 5. 1 Süt çiftliklerinin görsel olarak dağılımı.....	74
Şekil 5. 2 Her rota için maliyet hesaplama.....	77
Şekil 5. 3 Her rota için değişken maliyet hesaplama	77
Şekil 5. 4 Manuel rotalama sonucu oluşan örnek rota gösterimi 1.....	79
Şekil 5. 5 Manuel rotalama sonucu oluşan örnek rota gösterimi 2.....	79
Şekil 5. 6 Matematiksel modelin GAMS kodlama görüntüsü 1	83
Şekil 5. 7 Matematiksel modelin GAMS kodlama görüntüsü 2	83

Şekil 5. 8 GAMS rotalama sonucu oluşan örnek rota gösterimi 1	85
Şekil 5. 9 GAMS rotalama sonucu oluşan örnek rota gösterimi 2	85
Şekil 5. 10 Programın iş akış şeması Tasarruf Model kısmı	87
Şekil 5. 11 Programın iş akış şeması Genetik Algoritma kısmı.....	88
Şekil 5. 12 Hibrit programın makro kod görüntüsü 1	89
Şekil 5. 13 Hibrit programın makro kod görüntüsü 2	89
Şekil 5. 14 Hibrit programın makro kod görüntüsü 3	90
Şekil 5. 15 Metasezgisel hibrit rotalama sonucu oluşan örnek rota gösterimi 1	91
Şekil 5. 16 Metasezgisel hibrit rotalama sonucu oluşan örnek rota gösterimi 2	91

ÇİZELGE LİSTESİ

	Sayfa
Çizelge 4. 1 Klasik mantık ile bulanık mantık arasındaki temel farklılıklar	39
Çizelge 4. 2 Noktalar arası mesafe ve sij değerleri	53
Çizelge 5. 1 Araç tipleri hakkındaki veriler	71
Çizelge 5. 2 26.02.2014 Tarihli Süt Siparişleri	72
Çizelge 5. 3 26.02.2014 Tarihli Süt Siparişleri Devamı	73
Çizelge 5. 4 Lojistik Firmasının Karakteristik Özellikleri	75
Çizelge 5. 5 Tırın Girip Giremeyeceği Noktalar	77
Çizelge 5. 6 Mevcut Operasyon Rotalama Sonuçları	78
Çizelge 5. 7 Matematiksel Model Rotalama Sonuçları	84
Çizelge 5. 8 Hibrit Model Rotalama Sonuçları	90
Çizelge 5. 9 Hibrit Modelin diğer sonuçlar ile karşılaştırılması	92

HETEROJEN FİLOLU ARAÇ ROTALAMA PROBLEMİNİN META SEZGİSEL HİBRİT MODEL İLE ÇÖZÜMÜ

Selim EROĞLU

Endüstri Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

Tez Danışmanı: Doç. Dr. Alev TAŞKIN GÜMÜŞ

Bu çalışmada, belirli noktalardan süt toplayan heterojen araç filosuna sahip bir lojistik firmasının sabit ve değişken maliyetlerinin minimuma indirilmesi, farklı tiplerdeki araçların hangi noktaya atanacağını ve buna bağlı olarak da en uygun rotanın belirlenmesi amaçlanmıştır. Problemin çözümünde, öncelikle mevcut durum incelenerek, araç tipleri, kapasitesi ve sayısı, firmanın süt toplayacağı noktaların konumları ve bu noktaların birbirlerine olan uzaklıkları, noktalardan toplanacak sütün miktarı ve işletmenin toplam maliyeti gibi veriler elde edilmiştir. Elde edilen bu veriler belirli kısıtlar altında sezgisel bir modele dökülerek, sonrasında yazılıma geçilmiş, Excel Makro programlamada optimum çözümler elde edilmiştir. Elde edilen sonuçlarla firmanın mevcut durumu karşılaştırılmış ve iyileştirme yapıldığı ortaya konmuştur.

Anahtar Kelimeler: Süt Toplama, Heterojen Filolu Araç Rotalama Problemi, Meta Sezgisel Yöntem ile Çözüm

ABSTRACT

AN APPLICATION WITH METAHEURISTIC HYBRID MODELS FOR HETEROGENEOUS FLEET VEHICLE ROUTING PROBLEM

Selim EROĞLU

Department of Industrial Engineering

MSc. Thesis

Adviser: Assoc.Prof.Dr. Alev TAŞKIN GÜMÜŞ

In the present study, it is aimed to specify the lowering of the fixed and variable cost, the assigning points of different of vehicles and determining the most appropriate route of a logistics company with a heterogeneous fleet which is collecting milk from specific point. Primarily the current situation was researched and the data's like vehicle types, capacity and number of vehicles, the location of the point where the company collects the milk, the distance of the points from each other, the amount of milk which will be collected and the total cost of these operations have been archived to solve this problem. These archived data's were poured into a intuitive model under certain constraints and we obtained an optimal solution in Excel macro programming which guarantees the exact solutions. The obtained results were compared with the current status of the company and the improving was obtained.

Key Words: Collecting milk, routing problem of heterogeneous fleet vehicle, meta heuristic method and solutions

YILDIZ TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

1.1 Literatür Özeti

Hızla büyüyen ekonomide artık rekabet, firmalar arasında olmaktan çıkıp tedarik zincirleri arasında olmaya doğru yönelmiştir. Dolayısıyla tedarik zincirinin genel performansı ile işletmelerin kârlılıkları arasındaki bağlantı ön plana çıkmaktadır. Bu yönelim firmaların tedarik zinciri yönetimine verdiği önemi gün geçtikçe arttırmakta ve bu alanda birçok çalışma yapılmasına neden olmaktadır[1].

Lojistik genel olarak ürünün ya da hizmetin bir tedarik noktasından çeşitli talep noktalarına taşınması olarak tanımlanmaktadır. Tam bir lojistik sistemi hammaddelerin satıcılardan ya da tedarikçilerden alınarak taşınması, bu hammaddelerin üretim için imalat fabrikalarına dağıtılması, üretilen bu ürünlerin depolara ya da dağıtım merkezlerine aktarılması ve son olarak müşterilere dağıtılmasını kapsamaktadır. Hem dağıtım hem de tedarik prosedürleri etkili bir taşıma yönetimine ihtiyaç duymaktadır. İyi bir toplama/dağıtım yönetimi işletmenin toplam lojistik maliyetlerinde önemli ölçüde bir azalma sağlayabilmektedir [1].

İşletmelerin toplam lojistik maliyetlerinin $1/3$ - $2/3$ 'ü dağıtım maliyetlerinden kaynaklandığından, dağıtım ekipmanının ve personelinin etkili ve verimli bir şekilde kullanılması işletme yöneticileri açısından önemli bir ilgi alanı haline gelmiştir. Dağıtım maliyetlerini azaltmak ve müşterilere sunulan servisin kalitesini arttırmak için en kısa zamanı ya da mesafeyi verecek olan, bir aracın şebeke içerisinde izleyeceği en uygun rotayı bulmak günümüzde en çok tartışılan bir konu haline gelmiştir [2].

Dağıtım sistemleri, müşterilerin sürekli artan beklentileri ve sürekli değişen ürün nitelikleri ile son derece karmaşık bir hale gelmiştir. Bu da bu alanda hizmet veren şirketlerin daha etkili dağıtım planları yapmasını gerektirmektedir. Dağıtım ağı içerisinde bulunan müşterilerin taleplerinin en az maliyet ile karşılanması bu dağıtım planlarının merkezinde yer almaktadır.

Günümüz küresel piyasasında yoğun rekabet, kısa yaşam eğrisine sahip ürünler ve müşterilerin artan beklentileri, üreticileri dağıtım sistemlerine yatırım yapmalarına ve gereken önemi vermelerine zorlamıştır. Bu durum, iletişim ve ulaşım teknolojilerindeki değişimle birlikte, örneğin mobil iletişim ve gıda dağıtım gibi, lojistik yönetiminin sürekli gelişimine neden olmuştur. Lojistik sisteminin en önemli parçası ve problemi ise Araç Rotalama Problemidir [3]. Günlük hayatta ürün dağıtım ve toplanması, banka para transfer araçları, öğrenci servisleri, yakıt dağıtım ve daha birçok dağıtım işlemi araç rotalama probleminin örnekleri olarak gösterilebilir.

Bu tarz karmaşık problemlerin çözümünde genel olarak kesin ve sezgisel olmak üzere iki tarz çözüm geliştirilmiştir. Kesin algoritmalar en iyi çözümü bulmak ve bunun en iyi çözüm olduğunu ispatlamak üzerine konumlandırılmıştır. Ancak bu algoritmalar kombinasyonel problemlerde düşük performans göstermiştir. Ayrıca çözüme ulaşmadaki zaman faktörü kesin yöntemlerinde karşılaşılan problemlerdendir. Sezgisel algoritmalar ise kesin algoritmaların aksine kısa zaman dilimlerinde yüksek kalitede çözümler ortaya koymuştur.

Günümüzde, biyolojik yapılardan esinlenerek oluşturulan sistemler önem kazanmakta ve yapay zeka araştırmacılarının ilgisini çekmektedir. Doğadaki bazı sosyal sistemler, sınırlı yetenekli basit bireyler tarafından oluşturulmalarına rağmen kolektif zeka davranışları sergilerler. Problemlere uyarlanan zeki çözümler, bu bireylerin kendi içerisindeki organizasyonları ve dolaylı iletişimlerinden ortaya çıkar. Bu sistemler, dağıtık yapay zeka sistemlerinin geliştirilmesinde önemli tekniklerin kaynağı olmaktadır [4].

Bu algoritmalarından birisi Genetik Algoritmadır ve araç rotalama, gezgin satıcı problemi gibi birçok kombinasyonel problemin çözümünde başarılı sonuçlar ortaya koymuştur.

1.2 Tezin Amacı

Dağıtım/toplama yapan firmalar araç rotalama problemi ile karşı karşıya kalmaktadır. Bu problem firmalara bir dağıtım/toplama maliyeti oluşturmaktadır. Bu da ürün maliyetlerinin yaklaşık %15-20'sine den gelmektedir[5]. Genellikle firmalar bu probleme herhangi bir matematiksel model kullanmadan, geçmiş tecrübelerinden veya kendi geliştirdikleri algoritmaları kullanarak çözüm aramaktadırlar. Firmaların bir matematiksel model kullanmadan yaptıkları bu araç rotaları, yüksek dağıtım maliyetlerinin ortaya çıkmasına neden olmaktadır. Etkin olmayan dağıtım rotaları, firmalara ekstra bir maliyet yaratmaktadır. Sonuç olarak etkin bir dağıtım rotası oluşturmak firmaya büyük bir maliyet tasarrufu oluşturmakta ve günümüzdeki rekabetçi ortamda önemli bir avantaj sağlamaktadır.

Bu çalışmada bir ARP için iki sezgisel yöntemden oluşan hibrit model incelenecektir. Hibrit model kullanılan araç rotalama probleminin amacı, müşteri tarafından belirlenen belirli bir zaman aralığında, dağıtım rotalarının optimal bir şekilde oluşmasını sağlamaktır.

Araç Rotalama Problemlerinde kullanılan kesin çözüm algoritmaları, orta büyüklükteki problemleri çözmek için çok büyük bilgisayar sürelerine ihtiyaç duyduğu için pek tatmin edici değildir. Meta sezgisel yöntemler ise kısa sürede optimuma yakın sonuçlar vermektedir. Günlük hayatta karşılaşılan problemler çoğunlukla çok büyük boyutta ve karmaşık olduğundan, bu tür problemleri çözmek için sezgisel yöntemler daha uygundur. Literatürde de ARP için az sayıda kesin çözüm yöntemi kullanılarak yapılmış çalışmalar vardır. Genellikle araştırmacılar meta sezgisel çözüm yöntemi üzerinde çalışmalarını yoğunlaştırmışlardır.

Bu çalışmada bir ARP için iki sezgisel yöntemden oluşan hibrit model incelenecektir. Hibrit model kullanılan araç rotalama probleminin amacı, müşteri tarafından belirlenen belirli bir zaman aralığında, toplama/dağıtım rotalarının optimal bir şekilde oluşmasını sağlamaktır.

Araç rotalama problemi, fiziksel dağıtım ve lojistik alanında önemli bir yönetim problemidir. Tipik bir araç rotalama problemi, bir dağıtım noktasından şehir, mağaza, depo, okul, müşteri gibi coğrafik olarak dağılmış noktalara, en düşük maliyetli rotaları

tasarlama problemidir. Bir rota, her noktanın bir kez ve bir araç tarafından ziyaret edildiği, tüm rotaların dağıtım noktasında başlayıp bittiği ve belirli bir rotadaki tüm noktaların toplam talebinin, bu rotayı yönetmek için tahsis edilen araç kapasitesini aşmadığı şekilde tasarlanmalıdır [3].

Uygulamalarda araç rotalama problemi sezgisel çözümü, yöneylem araştırması alanında önemli faydalar sağlamıştır. Öyle ki araç rotalama problemlerinin uygulanmasıyla milyonlarca dolar tasarruf sağlanmıştır. Örneğin 1991 yılında Amerika’da ulaşım ve kamu sektörüne yaklaşık 506 milyar dolar yatırılmıştır, araç rotalama yönteminde ufak bir iyileştirme bile önemli tasarruflar sağlayabilmektedir [3].

Araç rotalama probleminin sağladığı bu faydalar doğrultusunda bu çalışmada, araç rotalama probleminde sezgisel çözüm teknikleri kullanılarak, optimal dağıtım rotaları belirlenecek ve bu dağıtım rotasının, dağıtım süreleri ve maliyetler üzerindeki etkisi saptanacaktır. Bu çerçevede yapılacak uygulama ile aşağıdaki araştırma sorularına da cevap aranacaktır;

- Firmanın mevcut dağıtım rotası belirleme tekniği nedir?
- Firmada optimal dağıtım rotası belirlemede hangi bilimsel metot uygundur?
- Firmada belirlenen bilimsel metot kullanılarak optimal dağıtım rotaları belirlendiğinde dağıtım süreleri azalmakta mıdır?
- Firmada belirlenen bilimsel metot kullanılarak optimal dağıtım rotaları belirlendiğinde dağıtım maliyetleri azalmakta mıdır?

1.3 Hipotez/Öneri

Bir lojistik firmasında uygulama yapılarak mevcut dağıtım/toplama sistemi ile ilgili gerekli veriler toplandıktan sonra, araç rotalama problemi için geliştirilen meta sezgisel hibrit yaklaşım, mevcut sisteme uygulanarak yeni bir toplama rotasının belirlenmesine çalışılacaktır. En uygun toplama rotası bulunurken, toplama sistemini oluşturan maliyetlerin azaltılması amaçlanmıştır.

Uygun toplama rotası bulunabilmesi için araç rotalama problemlerinde kullanılan bir sezgisel yöntem olan, tasarruf yöntemi kullanılacaktır. Daha önceden modellenecek

olan araç rotalama problemi, tasarruf yöntemi kullanılarak toplama rotaları belirlenecek, sonrasında bu rotalar Genetik Algoritma ile iyileştirilecektir.

Bu çalışmayı benzer çalışmalardan farklı kılan özellik ise araçların heterojen filolu olması ve de çözümünde meta sezgisel hibrit model kullanılmasıdır.

1.3.1 Çalışmanın Aşamaları

Tez çalışması altı bölümden oluşmaktadır. Birinci bölüm giriş bölümü olup, problemin tanımı yapılmıştır. Daha sonra ise çalışmanın amacı, yöntemi, önemi ve çalışma planı hakkında bilgiler yer almaktadır.

İkinci bölümde, lojistik kavramı ve temel unsurları, lojistiğin tarihsel gelişimi ve lojistik problemlerin modellenmesi ele alınmıştır. Ayrıca çalışmamızdaki problemlerden biri olan araç rotalama problemini de içeren lojistik sistemindeki problemlere yer verilmiştir. Bu bölümde, lojistik yönetimi kavramı, tarihsel gelişimi, faydaları, önemi ve amacı ele alınmıştır.

Üçüncü bölümde, araç rotalama probleminin tanımı, temel prensipleri, amaçları ve türleri açıklanmıştır. Daha sonra ise araç rotalama problemlerini çözmek için kullanılan kesin ve sezgisel çözüm yöntemlerine detaylı olarak yer verilmiştir.

Dördüncü bölümde, verilerin toplanması ve toplanan verilerin çözümlenmesinde kullanılan yöntemler üzerinde durulmuştur. Ayrıca sezgisel yöntemler arasından seçilen yöntemin neden seçildiğini açıklayan kıyaslama bölümüne yer verilmiştir. Toplama rotasındaki verilerin analizi yapılmıştır. Firmaya ait mevcut durumdaki toplama rotaları incelenmiş ve çalışmadaki önerilen yöntem sonucunda çıkan dağıtım rotaları ile karşılaştırılmıştır.

Beşinci bölüm bir lojistik firmasında uygulama yapılmış ve buna dair veriler ve program çıktılarını içermektedir.

Altıncı bölüm, sonuç bölümüdür. Bu bölümde araştırma bulguları ve değerlendirmeler ışığında elde edilen sonuçlar özetlenmiştir ve daha sonraki çalışmalar için önerilerde bulunulmuştur.

LOJİSTİK VE LOJİSTİK YÖNETİMİ

2.1 Lojistik Kavramı ve Temel Unsurları

Lojistik ve ilgili eşdeğer kelimelerin en uygun tanımı daima önemli bir soru olmuştur. Genel olarak lojistik, zaman ve yer faydası yaratmada, arz ve talebin koordinasyonu ve hareketini kolaylaştıran tüm faaliyetlerin yönetimidir.

Kitaplarda ve internette ayrıca birçok tanımlaması vardır. Seçilen birkaç tanesi;

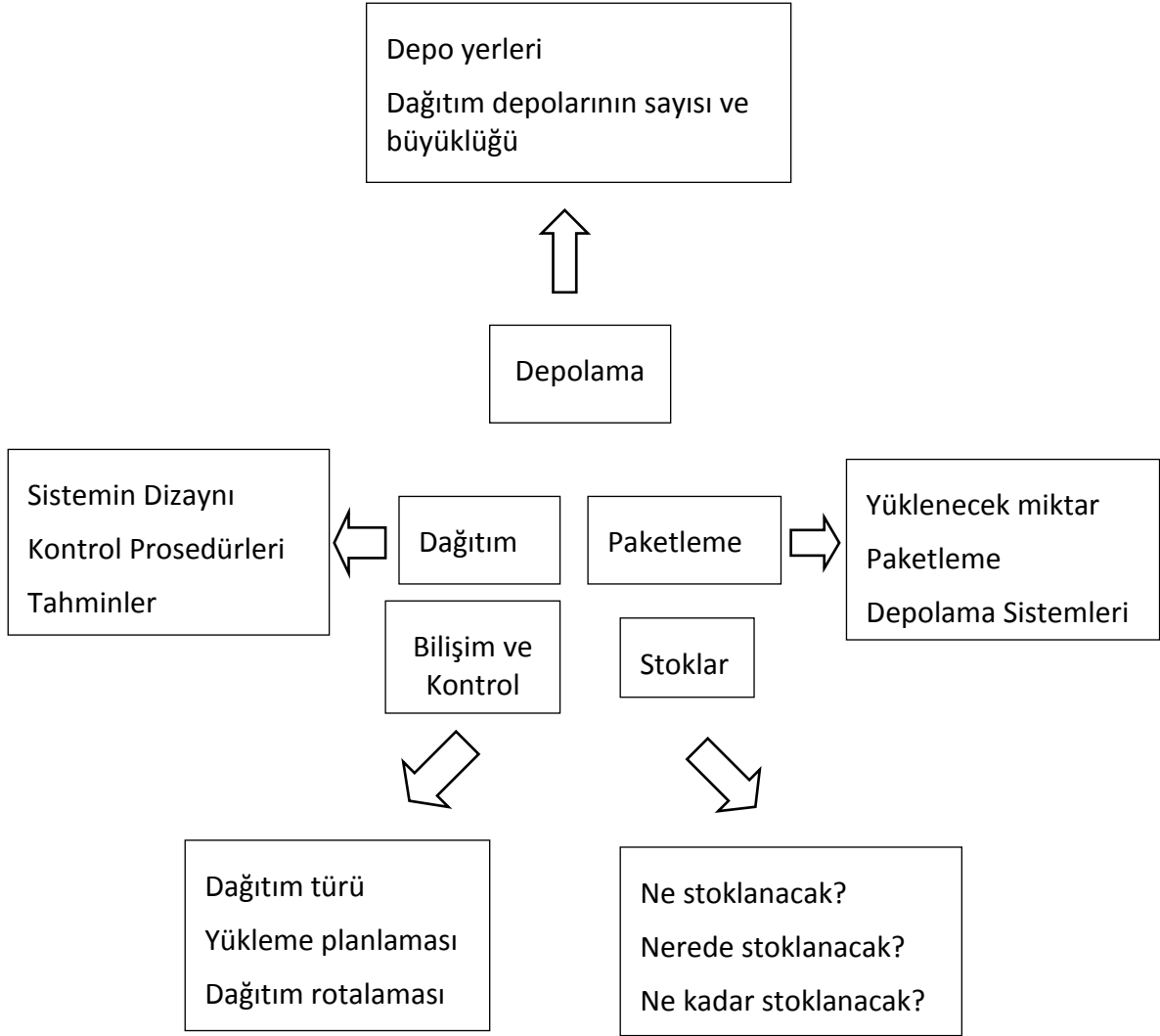
Wikipedia’da “Lojistik, ürün, enerji, bilgi ve diğer kaynakların akışını yönetme ve kontrol etme sanat ve bilimidir” şeklinde tanımlanmıştır.

CSCMP(Tedarik Zinciri Profesyoneller Heyeti) tarafından, “Lojistik yönetimi, müşteri isteklerini karşılamak için, başlangıç ve tüketim noktası arasında mal, hizmet ve ilgili bilgilerin etkin, etkili gönderimi ve tersine akışını ve depolanmasını planlamak, uygulamak ve kontrol etmektir” olarak tanımlanmıştır.

Lojistik ve Ulaşım Enstitüsü’ne göre “Lojistik, kaynağın doğru zamanda, doğru yerde, doğru maliyette, doğru kalitede konumlandırılmasıdır”.

Birçok sektöre uygulanan uygun modern bir tanım, “Lojistik, müşteriye kabul edilebilir hizmet sunmakla birlikte, maliyet etkin bir yolla, arz kaynağından üretim yeri aracılığıyla tüketim noktasına, ürünlerin etkin bir şekilde transfer edilmesiyle ilgilidir.” şeklinde yapılmaktadır [5].

Çoğu firma için dağıtım ve lojistiğin temel unsurlarını temsil eden anahtar alanların uygun listesini hazırlamak mümkündür. Bunlar dağıtım, depolama, stok, paketleme ve bilişim ve kontrolü içermektedir. Şekil 2.1’de gösterilen bu liste, farklı unsurlar içerisinde detaylı yönlere ulaşmak için tekrar incelenebilir [5].



Şekil 2. 1 Lojistiğin Temel Unsurları [5]

Üretim ve stok sistemlerine bakış açısına göre, lojistik kararları tipik olarak aşağıdaki gibi sınıflandırılabilir [3];

- **Stratejik seviye**, firma üzerinde uzun vadeli etki yaratan kararlardır. Bunlar üretim ortamlarının ve depoların kapasitesi, yerleşimi ve sayısı veya lojistik ağı aracılığıyla malzemenin akışı dikkate alınarak alınan kararları içermektedir.

- **Taktik seviye**, her çeyrekte bir ve her yılda bir yenilenen kararları içermektedir. Bunlar ziyaret edilen müşteri sıklığını içeren satın alma ve üretim kararları, stok politikaları ve ulaşım stratejilerini içermektedir.
- **Operasyonel seviye**, çizelgeleme, rotalama ve kamyonları yükleme gibi günlük kararları içermektedir.

2.2 Lojistiğin Tanımı ve Gelişim Süreci

Tarihte lojistik kavramı askerlik ve endüstri yönetiminde ortaya çıkmıştır. Lojistik askeri açıdan, ilk olarak sistem/ürün desteği ile başlamıştır. Bakım planlaması, insan gücü ve personel, ikmal desteği, destek donatımı, teknik veri, eğitim ve eğitim malzemesi, bilgisayar kaynak desteği, tesisler, ambalajlama, kullanma, depolama, taşıma, gibi lojistik faktörlerin güvenilirlik ve elde edilebilirliğinin etkileşimidir [6].

Dağıtım ve lojistik yönetiminin elemanları, her zaman üretim, depolama ve ürünlerin hareketlerine temel oluşturmuştur. Dağıtım safhasının ilk zamanlarında mamullerin değişik noktalarda depolanması popüler hale gelmiştir. Fakat bu da yeterli olmamıştır çünkü yüksek maliyetlerle acil taşıma gereksinimleri ortaya çıkmaya başlamıştır. Bu nedenle, bazı firmalar kendi dağıtım şirketlerini kurmaya başlamışlardır [6].

1940 yılının sonlarında Dantzing tarafından, ABD (Amerika Birleşik Devletleri) Hava Kuvvetlerinde gerçekleştirilen SCOOP (Optimum Programların Bilimsel Hesaplaması) projesinde doğrusal programlamanın ilk çalışmaları yapılmış ve Berlin hava köprüsü ile ilgili lojistik problemin çözümünde önceden belirlenmiş bir amaç fonksiyonunun kaynak kısıtları altında optimizasyonu için yöntemler geliştirilmiştir. Bu proje Hava Kuvvetlerinde yedek parça sisteminin envanter kontrolü için önce kart sistemini daha sonra bilgisayarları devreye sokmuştur [6].

SCOOP projesi ayrıca, üretim imkan ve kabiliyetinin, materyal ve insan gücü ihtiyaçlarının hesaplanması, 1950'lerdeki benzer seferberlik şartlarındaki endüstriyel koşulların belirlenmesi gibi problemlerin çözümüne yeni boyutlar kazandırmıştır. Doğrusal programlama, lojistik problemlerine öncelikle ulaştırma, ürünlerin dağıtımı ve kaynak kullanımı gibi konularda gerçek şartların formülizasyonu ile büyük ölçüde yararlı olmuştur. Doğrusal programlamanın yanı sıra diğer matematiksel programlama

yöntemlerinde uygulama alanlarında büyük gelişme göstermiştir. 1950'li ve 1960'lı yıllarda lojistik faaliyetlerinde uygulama alanı bulan diğer bir yöntem ise benzetimdir. Benzetim, envanter tahsisi, işgücü dağıtımı ve üretim programlarının hazırlanmasında yaygın olarak kullanılmıştır. Başta envanter yönetimi için olmak üzere bilgisayar ilk kez lojistik ihtiyaçlarını karşılamasında yönetim tasarımının bir parçası olmuştur [6].

1960'ların başında lojistik yönetiminin ufku genişlemiştir. Bu periyot boyunca yönetim, müşteri hizmet anlayışına önem vermeye başlamıştır. Bu dönemde birçok farklı lojistik anlaşmaları, üretim ve/veya pazarlamayı desteklemek için kullanılmıştır. O zamanki en önemli stratejik amaç; lojistik yönetimini geliştirmek ve tamamlamak için mümkün olan en önemli maliyetle, bunu sağlayabilecek özelleştirilmiş müşteri hizmetini kurabilmek olmuştur [6].

1960 sonrası çoğu firma parçalanmış yönetim anlayışından, derece derece bütünleştirilmiş yönetim anlayışına geçiş yapmıştır. Parçalanmış yönetim anlayışının satın alma ve depolama gibi bireysel süreçleri, malzeme yönetimi ve fiziksel dağıtım adı altında iki başlıkta toplanmıştır. Malzeme yönetimi; satın alma, hammaddeler, yarı mamullerin stok kontrolü, işletme içi ulaşım, artıklar ve üretim programlaması gibi fonksiyonları içermektedir. Fiziksel dağıtım ise navlun (nakliye ücreti), depolama, malzeme taşıma, ambalajlama, sipariş süreci, talep tahmini, envanter kontrolü ve müşteri hizmetini içermektedir [6].

1970-1979 arası yılların, lojistik kavramının gelişimi ve uygulanmasındaki etkisi önemlidir. Bu dönemde hemen hemen her girişimcilik alanında tedirginlik ve belirsizlik hâkim olmuştur. 70'li yılların ilk dönemlerinde ortaya çıkan grevler ve akaryakıtı gelen ambargo, dünyayı yeni bir krizin eşiğine getirmiş, II. Dünya Savaşı'ndan bu yana geçen ilk zaman periyodu için düşük maliyetli enerjinin bulunabilirliği kritik bir kavram haline gelmiştir. Enerji açıklarına, akaryakıt ve petrole bağlı malzemelerin fiyatının yükselmesinin eklenmesi ile birçok temel malzemedeki ve üretilen mamullerdeki açıklar en yüksek noktaya çıkmıştır.

1970'lerin başlarında, lojistik sistemlerin tasarımı ve kontrolü için büyük kapasiteli bilgisayarların araştırılması yapılmış ve önemli gelişmeler sağlanmıştır. Lojistik modeller, alternatif lojistik stratejilerin gerçeğe dönüşmesine yardımcı olmuşlardır.

1970'lerin sonlarında birçok yönetici malzeme yönetimine ve fiziksel dağıtım faaliyetlerine organik bir bütün olarak bakmaya başlamıştır [7].

1970'lerin sonlarında bilgisayar teknolojisinde yaşanan gelişmeler meyvelerini 1980'lerde vermeye başlamıştır. Büyük, hantal bilgisayarların yerini küçük ve kapasitesi geniş bilgisayarlar almıştır. Bilgisayar donanımları ucuzlamış, dördüncü ve beşinci jenerasyon bilgisayar dilleri gelişmiş bu sayede de iş görme kapasiteleri, karar verme süreçleri gelişmiştir. Bununla birlikte lojistik kaynak planlamasında, birbirleriyle ilişkili bölümler arasında veri alışverişinin sağlanması ile lojistik verimliliğinde benzeri görülmemiş düzeyde başarı sağlanmıştır. ABD'de uydu sistemleri aracılığı ile yöneticiler taşıma araçları ile haberleşmeye başlamışlar, bunun ardından araç telefonlarının ve diğer iletişim teknolojilerini kullanmışlardır. Böylelikle lojistik operasyonların koordinasyonu ve kontrolü kolaylaşmış ve maliyetler önemli düzeyde azalmıştır [7].

1980'lerin başlarında ise bir önceki döneme göre çok keskin değişiklikler olmuştur. En önemli değişiklikler şöyledir:

- Taşımacılıktaki değişiklikler
- Bilgisayar teknolojisine giriş
- İletişimdeki değişim

Taşımacılıkta verimliliği arttırmak üzerine çalışmalar yapılmıştır. Bu çalışmaların odağını, kullanılan ekipmanların faydasını arttırmaya yönelik çalışmalar oluşturmuştur. Bununla birlikte geleneksel fiziksel dağıtım faaliyetleri, üretimi destekleme ve tedarik üzerine de birçok gelişme yaşanmıştır. Bunun sonucunda taşıma giderlerinde %50'lere yaklaşan bir azalma yaşanmış ve bu da firmaların kar rakamlarını etkilemiştir.

Ayrıca bilgi teknolojilerindeki gelişmeler, malların ve malzemelerin siparişlerinin verilmesi, taşınması ve depolanması gibi şirketlerin yoğun faaliyetlerini daha iyi takip edebilmelerini sağlamıştır. Bilgisayarla bütünleşik nicel modellerle birlikte bu bilgi, envanter seviyesini ve hareketini optimize etme yeteneğini arttırmıştır. Malzeme ihtiyaç planlaması (MRP), dağıtım kaynakları planlaması (DRP) ve tam zamanında üretim (JIT) gibi sistemler, şirketlerin sipariş yönetiminden envanter yönetimine, tedarikçiye sipariş vermeye, satış tahminine ve üretim planlamaya kadar bir çok malzeme yönetim faaliyetlerini birbirine bağlamayı sağlamıştır [7].

Lojistiğin gelişimi temelde üç aşamada toplanabilir;

a) Parçalanma (1960 – 1980) : Bu dönemde lojistiği oluşturan faaliyetlerin ayrı ayrı yapıldığı görülmektedir. Aşağıdaki operasyonların bir kısmı işletme içinde yapılırken, kısmen de dışarıdan hizmet alma şeklinde gerçekleştiği görülmektedir.

- Talep Öngörüsü
- Satın Alma
- İhtiyaç Planlama
- Üretim Planlama
- Fabrika Stokları (girdi düzeyindeki stoklar)
- Depolama
- Malzeme İşlemleri
- Paketleme
- Mamul Stokları
- Sipariş Süreci
- Taşıma
- Müşteri Hizmetleri
- Dağıtım Planlama

b) Birleşme (1980 – 2000) : Bu dönemdeki lojistik faaliyetler iki kavram altında toplanmıştır:

- Madde ve Malzeme Yönetimi
- Fiziksel Dağıtım

c) Toplam Bütünleşme (2000 – Günümüze) : Halen de devam etmekte olan bu süreç parçalanma ve birleşme kısmında verilen faaliyetlerin bir çatı altında toplanmasını gündeme getirmiştir. Dünya ekonomisinde yaşanan küreselleşme, liberalleşme ve buna paralel olarak firmaları zorlayan uyum çabaları, lojistik faaliyetlerin önemini arttırırken günümüzdeki entegre lojistik kavramını ortaya çıkarmıştır [8].

Sonuç olarak, eskiden var olan salt nakliyenin yerine, günümüzün gerektirdiği hizmetleri sağlayabilmek için faaliyet, kapsam ve konu alanları geliştirilmiş bir şekilde yeni hizmet anlayışı ortaya çıkmıştır. Bu kavram çeşitli şekillerde, tanımlanmaya çalışılmıştır. Lojistik; doğru şeyin, doğru yerde, doğru zamanda olmasını sağlamak, olarak karşımıza çıkmaktadır. Ancak, buna bir dördüncü parametreyi daha ekleyebiliriz. Böylelikle de, uluslararası rekabet şartları çerçevesinde istenilen şeyi makul ve kabul edilebilir masrafla yani rekabet edebilir bir fiyatla sağlamayı yukarıda saydığımız üç kriterli tanıma dahil edebiliriz. Bununla beraber bu kavramın çeşitli farklı tanımlamaları da yapılmıştır.

Örneğin, günümüz iş dünyasında kullanılmasına gittikçe daha fazla ihtiyaç duyulan enformasyon, iletişim ve kontrol sistemlerinin içinde yer aldığı mal, hizmet, bilgi ve sermaye akımının iş planlama çerçevesidir [8]. Lojistik, kaynakların zamana bağlı olarak, tedarik zincirinde konumlandırılmasını, tedarikçiden nihai müşteriye kadar yönetimini içerir. Değişimlere hızlı ve ekonomik tepki için, tedarik zinciri içindeki tüm fonksiyonların, süreçlerin ve kaynakların entegre bir şekilde yönetimi gerekir[8].

2.3 Lojistik Sistemindeki Problemler

Lojistik sistemi, dağıtım ağı konfigürasyonundan başlayan ve paketleme işlemine kadar süren çözülmesi gereken birçok farklı problemi içermektedir. Bu problemlerden, içerisinde bu çalışmada sonucu aranan Araç Rotalama problemini ve zaman penceresi ile dağıtımı da içine alan on problem aşağıda belirtilmiştir.

2.3.1 Dağıtım Ağı Konfigürasyon Problemi

Coğrafi olarak dağınık perakendecilere ürün sunan birkaç üretim yeri olduğu düşünülün. Mevcut depolar yeterli görülmemekte ve yönetim, dağıtım ağını yeniden tasarlamak ve organize etmek istemektedir. Bu durum, örneğin değişen talepler veya birkaç mevcut depo için finansal kiralama sözleşmelerinin sona ermesi gibi nedenlerden kaynaklanabilir. Buna ek olarak, değişen talepler fabrika üretim seviyesinde değişikliğe, yeni tedarikçi seçimine ve genel olarak dağıtım ağı boyunca ürünlerin yeni akış yoluna neden olabilir. Amaç, toplam üretim, stok ve ulaşım maliyetlerini minimize ederek ve çeşitli hizmet seviyesi isteklerini karşılayarak, depo

yeri ve kapasitesini seçmek, her bir üretim ortamında her bir ürün için üretim seviyelerini belirlemek, merkezler arasında (üretim ortamından depoya veya depodan tedarikçiye) ulaşım akışını düzenlemektir [3].

2.3.2 Üretim Planlaması Problemi

Bir üretim merkezi sabit sonlu dönemde, ürün için talebi karşılamak üzere üretim yapmalıdır. Birçok gerçek örnekte talebin dönem boyunca bilindiğini varsaymak uygundur. Bu, örneğin siparişler daha sonraki dönem için verilmişse veya sözleşmeler gelecek birkaç ay için dağıtım yapmak üzere imzalanmışsa mümkündür. Üretim maliyeti, makine hazırlık maliyetleri gibi sabit kısımdan ve bir birimi üretmek için katlanılan maliyetler gibi değişken kısımdan oluşmaktadır. Elde bulundurma maliyetleri stoktaki her bir birim için katlanılan maliyetlerdir. Planlamacının amacı, her dönemde ürünler için talebi karşılamak ve sabit dönemde toplam üretim ve stok maliyetlerini minimize etmektir. Bu problem üretilen ürün sayısı arttıkça zorlaşmaktadır [3].

2.3.3 Stok Kontrol

Belirli bir ürünün stokunu tutan bir perakendeci düşünölsün. Müşteri talepleri belirsiz olduğundan, perakendeci talebin belirsiz dağılım bilgisine sahiptir. Perakendecinin amacı, hangi noktada yeni parti siparişı ve ne miktarda sipariş vereceğine karar vermektir. Genellikle sipariş maliyetleri iki bölümden oluşmaktadır; sabit kısım siparişin büyüklüğünden bağımsızdır, örneğin depodan perakendeciye araç gönderme maliyeti gibi, değişken kısım ise sipariş edilen ürün sayısına bağlıdır. Doğrusal stok tutma maliyeti birim zamanda birim ürün için sabit bir oran olarak ortaya çıkmaktadır. Perakendeci beklenen sipariş maliyetlerini ve stok tutma maliyetlerini minimize etmek için optimal stok politikası belirlemelidir. Bu problem sipariş edilen ürün sayısı arttıkça zorlaşmaktadır ve sipariş maliyetleri sipariş edilen malzemeye bağlıdır [3].

2.3.4 Stok ve Dağıtımın Entegrasyonu

Bir depo çeşitli ürünlerle perakendecilere hizmet sunmaktadır. Faaliyet maliyetlerini azaltmak için yönetim, stoklar ve dağıtım maliyetleri arasında uygun bir denge sağlamalıdır. Depo ve perakendeci arasındaki sık turlar, her bir dağıtımın küçük olduğu,

stok maliyetlerinin düşük ve ulaşım maliyetlerinin yüksek olduğu anlamını taşımaktadır. Nadir turlar büyük dağıtımlara, yüksek stok maliyetlerine ve düşük ulaşım maliyetlerine sebep olmaktadır. Varsayın ki, her bir perakendeci ürün için sabit taleple karşılaşmaktadır. Amaç, araç rotalarını ve çizelgelerini ve ziyaret edilen tedarikçi sıklığını belirlemede bir stok politikası ve ulaşım stratejisi oluşturmak, dolayısıyla sistem genelinde stok ve ulaşım maliyetlerini minimize etmektir [3].

2.3.5 Araç Filosu Yönetimi

Bir depo sınırlı kapasiteli araç filosu kullanarak perakendecilere ürünlerini sunmaktadır. Bir gönderici, araçlara yük dağıtma ve araç rotalarını belirleme görevindedir. Birinci olarak gönderici, bir araç tarafından uygun olarak sunulan, yükü bir araca uyan gruplara perakendecinin nasıl katılacağına karar vermelidir. İkinci olarak gönderici, maliyeti minimize etmek için hangi sırayı kullanacağına karar vermelidir. Doğal olarak, iki maliyet fonksiyonundan biri mümkündür. Birinci olarak amaç, kullanılan araç sayısını minimize etmek; ikinci olarak odak noktası, seyahat edilen toplam uzaklığı azaltmaktır. Sonrası, sınırlı kapasiteli araç filosu tarafından müşterilere hizmet verilmesi gereken tek depolu Kapasiteli Araç Rotalama Problemi örneğidir. Araçlar başlangıçta depoda yer almaktadır ve amaç minimum toplam uzunluğu olan araç rotaları bulmaktır [3].

2.3.6 Araç Rotalama

Perakendecilere ürün dağıtmak için depodan ayrılan bir araç olduğu varsayalım. Perakendecilerin verdiği sipariş miktarı, dağıtımın ne kadar uzun olacağını ve hangi sürede aracın depoya dönebileceğini belirleyecektir. Dolayısıyla aracın etkin bir rota izlemesi önemlidir. Bir depodan perakendecilere minimum uzunluktaki rotayı bulma problemi (hem zaman, hem uzaklık açısından) Gezgin Satıcı Problemi örneğidir ve araç rotalama filo yönetiminin alt problemidir [3].

2.3.7 Paketleme Problemi

Çoğu lojistik uygulamalarında, parçalar kutulara veya sınırlı ölçüde araçlara paketlenmelidir. Amaç kullanılan kutu sayısının mümkün olduğunca küçük olduğu parçaları paketlemektir. Bu problem Sırt Çantası Problemi olarak tanımlanır. Örneğin,

amaç ürünleri dağıtırken kullanılan araç sayısını minimize etmek olduğu zaman Kapasiteli Araç Rotalama Probleminin özel bir durumu olarak ortaya çıkmaktadır. Aynı zamanda diğer birleşik problemlerde alt problem olarak ortaya çıkmaktadır [3].

2.3.8 Zaman Penceresiyle Dağıtım

Birçok durumda ürünleri müşterilere ve perakendecilere özel zaman pencereleri boyunca dağıtmak gerekir. Bu durumda örneğin belirli bir perakendeci veya müşteri saat 9-11 arası dağıtım isteyebilir. Her bir perakendeci bir zaman penceresi belirlerse, kapasite kısıtlarını ve zaman penceresi kısıtlarını karşılayan araç rotalarını bulma problemi daha zor olmaktadır [3].

2.3.9 Toplamalı Dağıtım Sistemleri

Bazı dağıtım sistemlerinde her bir müşteri, alım yeri ve bir dağıtım veya varış yeri olabilmektedir. Gönderici, ürünlerin alım ve dağıtımını koordine etmeye ihtiyaç duyar, dolayısıyla her bir müşteri alım/dağıtım çifti tek bir kamyon tarafından karşılanır ve toplam seyahat edilen mesafenin mümkün olduğunca küçük olması amaçlanır. Bir kamyon rotası, araç kapasite kısıdını, her bir alım ve dağıtım için zaman penceresi gerekliliklerini karşılamak ve alımın dağıtımdan önce gerçekleştirileceğini garanti etmektedir [3].

2.4 Lojistik Problemlerini Modelleme

Önceki bölümde sunulan, matematiksel olarak tanımlanabilen problemler, gerekli veriler mevcut olmadığı sürece anlamsızdır. Verileri bulmak, doğrulamak ve tablo haline getirmek oldukça güçtür. Gerçekte stok tutma maliyetleri, üretim maliyetleri, ekstra araç maliyetleri ve depo kapasitelerini saptamak genellikle zordur. Bunun yanında, belirli lojistik problemi ile ilgili verileri tanımlamak, veri toplama problemine başka bir karmaşıklık getirmektedir. Veriler olsa bile, karmaşık gerçek hayat problemlerini modelleme ile ilgili başka güçlükler vardır. Analizlerde seyahat zamanlarında sapmalar, üretimde yer alan değişkenler, stok azaltma, tahmin, işgücü çizelgeleme gibi unsurlar genellikle dikkate alınmaz. Bu unsurlar, lojistik uygulamaları daha da karmaşıklştırmaktadır [3].

Firmalar lojistik problemlerini çözmek için çeşitli yaklaşımlar kullanmaktadır. Birinci olarak, insanlar geçmişte iyi olan şeyleri tekrar etmeye eğilimlidir. Eğer geçmiş yılın emniyet stok seviyesi yığın talepten kaçınmak için yeterli ise, benzer seviye bu yıl da kullanılabilir. Eğer geçmiş yılın dağıtım rotası başarılıysa, tüm perakendeciler zamanında teslim alırlar ve dolayısıyla değıştirmezler. İkinci olarak, sıklıkla kullanılan “öncelik kuralı” çok etkin olabilir. Örneğin birçok lojistik yöneticisi sık sık “20/80 kuralını” kullanır, buna göre ürünlerin yaklaşık %20’si, toplam maliyetin yaklaşık %80’ini yaratmaktadır ve dolayısıyla bu kritik ürünlere yoğunlaşmak yeterlidir [3].

Lojistik ağı tasarımı çeşitli tecrübelerden de faydalanılarak yapılır. Bunun bir örneği, eğer firma ABD’de hizmet sunuyorsa ve sadece bir depoya ihtiyaç varsa, bu deponun Chicago’da yer alması gerekirken iki depoya ihtiyaç var ise birinin Los Angeles’ta diğlerinin Atlanta’da kurulması gerekir. Sonuç olarak bazı firmalar lojistik uzmanları sezgisine ve deneyimine başvurmayla çalışırlar ve rakipleri için iyi olan şeyin kendileri için de iyi olması gerektiği fikrine sahiptirler [3].

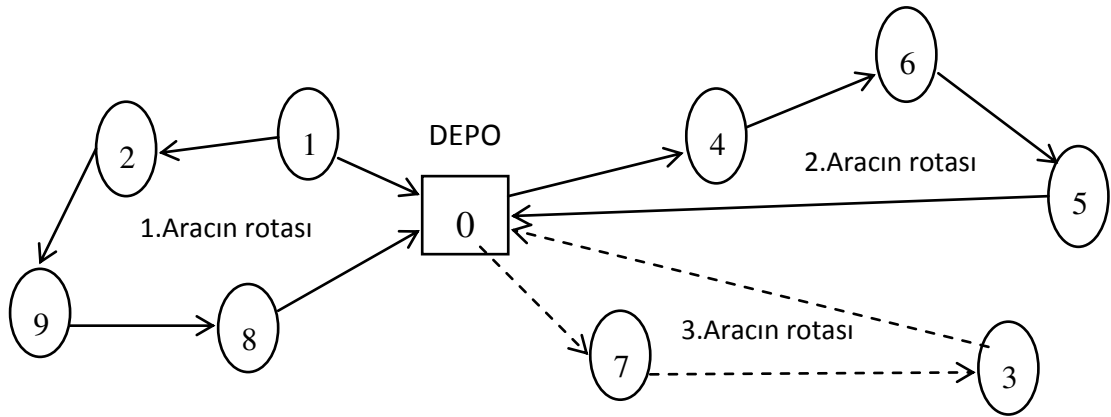
Tüm bu yaklaşımlar cazip ve anlam yaratan lojistik stratejilerle sonuçlanırken, belirli olaylar için “en iyi” stratejiye odaklanmamak yüzünden ne kadar kaybedildiği açık değildir. Yakın dönemde, hızla artan bilgisayar çözüm süreleri ile, sadece büyük firmalar için değil birçok firma için, lojistik stratejilerini optimize etmek amacıyla sofistike karar destek sistemlerini almak ve kullanmak şart olmuştur. Bu sistemlerde veriler girilmekte, gözden geçirilmekte ve çeşitli algoritmalar yönetilerek önerilen sonuç sade bir şekilde sunulmaktadır. Sistem uygun problemi çözmektedir ve bu karar destek sistemleri, sistem genelindeki maliyetleri önemli ölçüde azaltmaktadır. Her ne kadar tam anlamda kesin olarak “optimizasyon” olarak düşünülme de, sistem kullanıcıları için faydalı bir araç olarak hizmet vermektedir. Çoğu olayda bu sistemler yöneylem araştırması, yönetim bilimleri ve bilgisayar bilimleri tarafından geliştirilen teknikleri uygulamaktadır [3].

Yukarıdaki problemlerin çoğu, NP-zor problemler olarak nitelendirilen, zor birleşik problemlere sahiptir. Bu durum göstermektedir ki, her zaman optimal çözümü veya en iyi olası kararı bulacak bir algoritmayı yaratabilmek çok zordur. Dolayısıyla birçok durumda kesin çözüm yöntemleri kullanılarak optimal çözüme ulaşılamamaktadır ve böylece sezgisel yöntemler kullanılarak problemlere çözümler aranmaktadır [3].

ARAÇ ROTALAMA PROBLEMLERİ

3.1 Araç Rotalama Problemi Tanımı

Araç Rotalama Problemi, coğrafi olarak dağınık müşterilere bir/birden fazla depodan hizmet için görevlendirilen araçların en ideal şekilde dağıtım ve toplama rotalarının tasarlanması problemine denir [9]. ARP ilk olarak 1959 yılında Dantzig ve Ramser tarafından bulunan ve üzerinde en fazla optimizasyon geliştirilen yöntemdir [10]. Araç Rotalama Probleminin görsel anlatımı şekil 3.1 de aşağıdaki belirtilmektedir.



Şekil 3. 1 Araç Rotalama Probleminin Şekilsel Gösterimi [11]

Tipik bir Araç Rotalama Probleminde, depodan başlayarak belirli sayıdaki müşterileri gezen ve ardından tekrar depoya dönen belirli koşullar altında toplam yolculuk maliyetini minimize eden rotalar kümesinin bulunması amaçtır [12]. Belirli kapasitelere sahip bir dizi araç bir merkez depoda konumlandırılır. Bunlar bir dizi müşteri siparişine

hizmet vermek için mevcuttur. Her bir müşteri siparişinin özel bir yeri ve büyüklüğü vardır. Yerler arasında ulaşım maliyetleri mevcuttur. Amaç araçlar için en az maliyetli rota gruplarını bulmaktır. Öyle ki tüm müşteriler bir kez ziyaret edilir ve araç kapasitelerine bağlı kalınır [13].

Araç Rotalama yapılırken aşağıdaki unsurlara dikkat edilmelidir:

- Şebeke içerisinde bulunan müşterilerin talepleri tamamıyla karşılanmalıdır.
- Şebekede bulunan her varış noktası tek bir araç tarafından sadece bir kez ziyaret edilmelidir.
- Rota depodan başlamalı ve tekrar depoda sonlanmalıdır.
- Rota üzerinde bulunan müşterilerin toplam talep miktarı aracın toplam kapasitesinden fazla olmamalıdır.
- Her bir araç sadece bir rota üzerinde faaliyet göstermelidir.
- Araç rotalamanın temel amacı araçların kat edecekleri toplam mesafenin minimize edilmesi olmalıdır [14].

3.2 Araç Rotalama Problemi Temel Bileşenleri

Araç Rotalama problemlerinin temel bileşenlerini; talep yapısı, taşınacak malzemenin tipi, dağıtım/ toplama noktaları ve araç filosu oluşturur.

1-Talep Yapısı: Araç Rotalama problemlerinde talep statik veya dinamik olabilir. Statik talep durumunda talep önceden bilinir. Dinamik durumda ise bazı düğümlerdeki talep bilinmekte bazıları ise araç rotasında devam ederken belirli olmaktadır. Bu çalışmada statik talep yapısı kullanılmıştır.

2-Malzeme Tipi: Araçlarla çok çeşitli malzemeler taşınır. Tehlikeli maddeler, gıda maddeleri, gazete dağıtımı, çöp toplama bütün bunlar basit paketler olarak adlandırılır ve problem ilave bir karmaşıklık getirmezler. Diğer taraftan öğrenci servisleri; güvenlik, etkinlik, eşitlik gibi ilave bazı amaçlardan ötürü daha karmaşık bir yapıya sahiptir. Tehlikeli maddeleri taşıyan araçların rotalarının belirlenmesinde ise coğrafi özellikler büyük önem kazanır [10].

3-Dağıtım/Toplama noktaları: Birçok Araç Rotalama probleminde, dağıtım noktaları müşterilerin bulunduğu yer, toplama noktaları ise depodur. Tüketim mallarının fabrikalardan toptancılara dağıtımına buna iyi bir örnektir.

Depo genellikle aracın rotasına başladığı ve geri döndüğü noktadır. Depo sayısına göre problem, tek depolu ve çok depolu diye adlandırılabilir. Çok depolu problemlerde, depoların her biri kendi araçlarıyla işlerini yürütebilir, bu durumda problem birkaç bağımsız tek depolu Araç Rotalama problemine dönüşür. Araç bir depodan çıkıp, başka bir depoda yükleme/boşaltma yapabilir. Bu durumda problem bir bütün olarak ele alınmalıdır [15].

Dağıtım noktaları sabit ve önceden biliniyorsa hangi noktalara, hangi araçların hizmet vereceği belirlenmelidir. Diğer durumda dağıtım noktaları potansiyel yerler arasından seçileceği için ilave bir yerleştirme kararı gerekir.

Bazı Araç Rotalama problemlerinde dağıtım ve toplama noktaları aynıdır. Örneğin öğrenci servislerinde okul, gidişte dağıtım noktası, duraklar toplama noktası; öğrenciler evlerine dönerken ise okul depo, duraklar ise dağıtım noktalarıdır [15].

4- Araç Filosu: Bütün Araç Rotalama problemlerinde araçların kapasitesinin bilindiği ve çoğunlukla araçların homojen (aynı kapasitede) olduğu varsayılır. Filo heterojen ise filodaki araçların taşıma kapasiteleri farklıdır. Bu durum hangi araç tipinin, hangi rotaya hizmet vereceğinin belirlenmesini, yani ilave bir kararı gerektirir. Araçların diğer özellikleri arasında hız, yakıt tüketimi, taşınacak malzemeye uygunluğu sayabiliriz. Bu özelliklerin rotalama kararlarına doğrudan etkisi yoktur [16].

3.3 Araç Rotalama Probleminin Prensipleri

Literatürde çok sayıda Araç Rotalama Problemleri ve bu problemlerin her biri için geliştirilmiş çok sayıda algoritma bulunmaktadır. Ancak bu algoritmaların hiç biri günlük yaşamda işletmeler için en optimum çözümü verememektedir. Bundan dolayı araştırmacılar bugün hala Araç Rotalama Problemlerinde en etkin ve iyi sonuçları verebilecek algoritmalar yaratmak için çalışmaktadırlar [16].

Araştırmacılar ve bu konunun uygulayıcıları daha başarılı bir rotalama için, aşağıdaki sekiz prensibin dikkate alınmasını önermişlerdir [17].

- Noktalar içerisinde birbirine en yakın olanlar seçilmelidir. Bunun aracılığıyla toplam gidilen yol kısalmır.
- Farklı günlerdeki dağıtımlar birleştirilmelidir. Benzer noktalardaki dağıtımlar birleştirilerek, aynı rotaların yakın tarihlerde tekrar gidilmesi engellenir.
- Rotalara, başlarken mümkün olan en uzak nokta seçilmelidir.
- Yapılacak olan rotalamaların şekli, gözyaşı şeklinde olmalıdır. Bu şekilde uzak noktalara ulaşımında kazanç elde edilebilecektir.
- Mümkün olan en yüksek kapasiteli araçlar kullanılmalıdır. Bu sayede toplam maliyetler azalacak ve avantaj elde edilecektir.
- Eğer yapılabilir ise dağıtım ve tedarik aynı araçlarla yapılmalıdır. Bu şekilde toplam maliyet ve bunun yanı sıra gereken zaman da azalacaktır.
- Rota dışındaki noktalara ulaşımında, küçük araçlar kullanılmalıdır.
- Eğer gerekiyor ise dağıtımların ve tedariklerin zamanları tekrardan kararlaştırılarak zaman tasarrufu sağlanmalıdır.

3.4 Araç Rotalama Probleminin Amaçları

Araç rotalama problemleri için genellikle zıt olan birkaç amacın varlığı kabul edilebilir. Tipik amaçları şunlardır [10]:

- Araçların toplam gideceği mesafeyi veya rota süresini azaltmaya çalışarak global taşıma maliyetlerinin minimize edilmesi,
- Tüm müşterilerin taleplerini karşılamak koşulu ile gerekli olan araç sayısının minimizasyonu,
- Araç rota süreleri ve araç yükleri açısından tüm rotaların dengelenmesi,
- Müşteri hizmetleri ile ilgili sorunların minimizasyonudur.

3.5 Araç Rotalama Problemi Türleri

Araç rotalama problemleri gerçek hayattaki bazı özel durumlardan kaynaklanan bazı kısıtlar nedeniyle çeşitli dallara ayrılır [18]. Bunlar:

1-Karma Kapasiteli Araç Rotalama Problemi: Araç rotalama probleminde yer alan dağıtım yapan araçların belirli bir kapasitesinin olması durumudur. Karma kapasiteli araç rotalama probleminde her bir aracın birbirinden farklı bir kapasitesi olabilir.

2-Çoklu Depoya Sahip Araç Rotalama Problemi: Dağıtım firmasının müşterilere hizmet vermek için birden fazla deposunun olması durumudur. Eğer müşteriler depoların etrafında kümelenmiş ise, dağıtım problemi ayrı birer ARP olarak modellenenebilir. Ama müşteriler ve depoların yerleri birbirlerine karışmış ise, çoklu depoya sahip araç rotalama probleminin çözülmesi gerekmektedir. Bu problemde araçlar depolara atanır ve her bir araç ait olduğu depodan çıkarak müşteriye hizmet verir ve yine aynı depoya geri döner.

3-Bölünmüş Talebe Sahip Araç Rotalama Problemi: Bölünmüş talebe sahip araç rotalama problemi, aynı müşteriye birden fazla aracın servis yapabilmesine olanak veren araç rotalama problemidir.

4-Belirsiz Talebe Sahip Araç Rotalama Problemi: Bu tür bir problem, talebin belirsiz olduğu araç rotalama problemidir. Dağıtım aracı müşteriye vardığı zaman o müşterinin talebinin ne olacağı belli olur.

5-Geri Toplaması Olan Araç Rotalama Problemi: Geri toplaması olan araç rotalama problemi, müşterilerin depozito, ambalaj ve palet gibi, ürünlere ait bazı parçaları iade etme durumu olabilen araç rotalama problemidir. Bu durumda müşterilerden geri verilecek olan parçalar hesaba katılarak araç kapasiteleri hesaplanmalıdır.

6-Zaman Pencereci Araç Rotalama Problemi: Zaman pencereci araç rotalama problemi, her bir müşteriye ait bir zaman aralığı kısıdı olan araç rotalama problemidir. Bu problemde dağıtım aracı, her bir müşteriye belirli bir zaman aralığında hizmet vermek zorundadır.

7-Asimetrik Araç Rotalama Problemi: Dağıtım aracının depodan müşteriye gidiş mesafesi ile aynı müşteriden depoya olan uzaklığın farklı olduğu araç rotalama problemine, asimetrik araç rotalama problemi denir. Bu durumda maliyet (mesafe) matrisi simetrik değildir.

3.6 Araç Rotalama Problemleri Çözüm Yöntemleri

Araç Rotalama problemlerinin çözümü değişen çevre faktörleri, artan rekabet koşulları ile birlikte gelen kısıtlarla giderek zorlaşmaktadır. Zaman aralıkları, farklı kapasitelere sahip çok sayıda araç, rota üzerinde müsaade edilen seyahat zamanı, araç sürücülerinin dinlenme zamanları Araç Rotalama problemlerinde karşımıza çıkan kısıtlar içinde yer almaktadır [19].

Bir şebeke için Araç Rotalama problemi o probleme özel tasarlanan bir yöntem ile çözülebilir. En sık kullanılan çözüm yöntemlerinden birisi tasarruf metodudur. Ayrıca sıkça kullanılan diğer yöntemler süpürge metodu, gezgin satıcı yöntemi ve en kısa yol yöntemidir. Bu yöntemlerin her biri kendine özgü amaçlar ve kısıtlar taşımaktadır.

3.6.1 Kesin Yöntemler

3.6.1.1 Dal Sınır Algoritması

Dal ve sınır algoritması çözüm uzayını alt problemlere ayırmak ve daha sonra her bir alt problemi ayrı ayrı optimize etmek için böl ve ele geçir stratejisini kullanır [20].

Dal ve sınır algoritması ile çözülecek problem bir minimizasyon problemi olarak ortaya atılır. Dal ve sınır algoritmaları, bir ağaç olarak canlandırılan durum uzayında en düşük maliyetli çözüm düğümünü arayan bir araştırma içine girer [21].

Dal ve sınır algoritmasını, iki temel fikre bağlarız. Bunlardan birincisi, arama alanı olarak bir arama ağacı formu düzenlemektir. Fikirleri düzeltmek için, ne kullanacağımızı gösteren bir ikili arama ağacına karar veririz. Ağacın her düğümü problemin bir değişkenine tekabül eder ve düğümün dışındaki iki dal değişkenin reddedilmesi veya kabul edilmesine tekabül eder. Dal sınır algoritmasındaki “dal” arama ağacındaki dalları keşfetme süreci anlamına gelir [22].

3.6.1.2 Dal ve Kesme Algoritması

Dal ve kesme algoritması tamsayılı doğrusal programların çözümü için kombinatoriyal optimizasyon yöntemidir. Bu algoritma dal ve sınır algoritmasının bir türevidir [23].

Bu metot tam sayı kısıdı olmayan doğrusal programı, düzenli simpleks algoritma kullanarak çözer. Optimum bir çözüm elde edildiğinde ve bu çözüm tam sayı varsayılan bir değişken için tam sayı olamayan değer içerdiğinde, tüm uygun tamsayı noktalar tarafından tatmin edilen ama mevcut kesirli çözüm tarafından bozulan daha ilerdeki lineer kısıtları bulmak için kesme düzlemi algoritması kullanılır. Eğer böyle bir eşitsizlik bulunursa, buna lineer programa eklenir ve böylece “daha az kesirli” farklı bir sonuç bulunabilir. Bu proses tam sayılı çözüm bulunana veya artık kesme düzlemi bulunmayana kadar tekrar eder [23].

3.6.1.3 Küme Kapsama Algoritması

Küme kapsama problemi, literatürde genellikle örtüleme problemi “covering problem”, “le probleme de recouvrement” gibi isimlerle yer almaktadır [24]. Küme örtüleme problemleri 0-1 tam sayılı programlama modelinin özel bir hâlidir. Bu problemler, gerçek dünya problemleri olarak oldukça fazla uygulama alanı bulmaktadır [24].

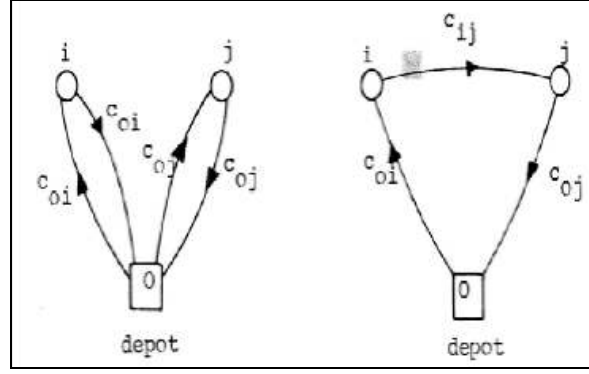
Dağıtım ve rotalama problemleri, plânlama problemleri ve yerleştirme problemleri sık sık küme örtüleme yapısında karşımıza çıkmaktadır. Bu yapı sayesinde yerleşim, araç ya da insanlar tarafından her bir müşterinin ihtiyacını karşılayabilir [25].

3.6.2 Klasik Sezgisel Yöntemler

Bu kısımda kısaca ARP için üretilen klasik sezgisel yöntemlerden bahsedilecektir.

3.6.2.1 Tasarruf Modeli

ARP çözmek için geliştirilen yöntemlerden birisi, Clarke ve Wright tarafından 1964 yılında geliştirilen ve belki de bilinen en iyi tur oluşturma sezgiseli olan Tasarruf yöntemidir. Bu yöntem, her bir müşteri ikilisi arasındaki maliyet tasarrufunu hesaplayarak başlar. Maliyet tasarrufları hesaplanarak iki müşteri arasına bir müşteri eklenir. Şekil 3.2’de görüldüğü gibi i ve j . müşteri ayrı turlardadır, i . müşteriden sonra j . müşteri eklenerek turlar birleştirilir.



Şekil 3. 2 Tasarruf Yöntemindeki Müşteri Birleştirilmesi

$$s_{ij} = c_{0i} + c_{0j} + c_{ij} - c_{0i} - c_{0j} - c_{ij} \quad (3.1)$$

$$s_{ij} = c_{0i} + c_{0j} - c_{ij} \quad (3.2)$$

Denklem (3.2)'deki tasarruf miktarı (s_{ij}) müşteri i ve j . müşterinin ayrı turlarda değil aynı turda hizmet almasından kaynaklanan bir maliyet tasarrufudur. Bu maliyet tasarrufu iki bağımsız turun birleştirilmesi ile ortaya çıkmaktadır. Her zaman tasarruf yönteminde, en büyük tasarrufu sağlayan (i, j) ikilisi, müşteri talebi ve araç kapasitesi kısıtları dikkate alınarak seçilir. Bütün müşterilerin araçlara atanmasına kadar bu işlem tekrarlanır [26].

Clarke-Wright'ın tasarruf yöntemi, kolay anlaşılabilirliği ve diğer ARP yöntemlerine göre esnek olması sayesinde geniş bir kullanım alanına sahiptir. Bu yöntem Gaskell (1967) ve Yellow (1970) gibi pek çok araştırmacı tarafından günümüze kadar uzanan zaman diliminde geliştirilmiştir [27].

3.6.2.2 Süpürme Yöntemi

Bu algoritma ARP'nin düzlemsel örneklerine uygulanır. Uygun kümeler ilk olarak depoyu merkez alan ışının ekseni üzerinde dönerik şekillenir. Bir araç rotası her küme için bir GSP çözülerek elde edilir. Bu algoritma ilk olarak 1971 yılında Wren'in kitabına geçmektedir fakat bu konuyu Gillett ve Miller (1974) popüler hale getirmiştir. Bu metodun basitleştirilmiş yürütme sistemi şöyledir:

Her i tepe noktası polar koordinat sisteminde (Q_i, r_i), ile gösterilir. Burada Q_i açığı ve r_i ışın uzunluğunu göstermektedir. Keyfi olarak seçilen i^* noktasına $Q_{i^*} = 0$ değeri atanır

ve merkezi $(0,i^*)$ ilk ışıktan 0 da konuşlanan geri kalan açığı hesaplar. Kenarları ve açıları Q_i 'nin artacağı şekilde sıralanır. Algoritmanın adımları şu şekildedir [10]:

Adım 1: Rota Sıfırlama

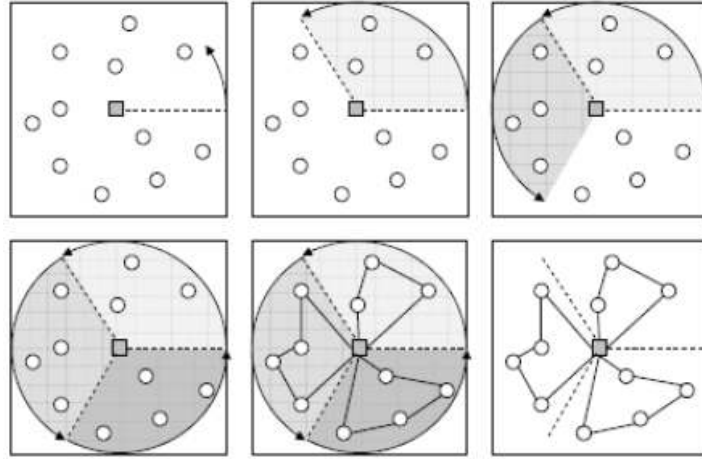
Kullanılmamış k aracı seçilir.

Adım 2: Rota yapma

En küçük açığa sahip rotalanmamış kenardan başlayarak araç kapasitesini veya maksimum rota uzunluğunu aşmayacak şekilde noktalar k aracına atanır. Eğer rotalanmamış noktalar kaldıysa 1. Adıma dönülür.

Adım 3: Rota Optimizasyonu

GSP ile çözülerek ayrı ayrı her aracın rotası optimize edilir. Şekil 3. 3 de bu aşamalar gösterilmektedir.



Şekil 3. 3 Süpürge Algoritması Adımları

3.6.2.3 En Kısa Yol Yöntemi

En kısa yol yönteminin temel amacı müşteri noktaları arasındaki yakınlık ve uzaklık durumu ve araç yükleme kapasitelerine göre araçlara müşteri atanmasıdır. En kısa yol yöntemi süreç aşamaları aşağıdaki gibidir:

Adım 1: Merkezi birimden rootalamaya başlanır. İlk araca merkezi noktaya en yakın müşteri atanır.

Adım 2: Rotaya atanan müşteriye en yakın, daha önce rotaya eklenmemiş noktalar incelenir. Eğer müşteriye en yakın iki nokta varsa her biri için süreç oluşturularak ayrı çözüm dalları yaratılır.

Adım 3: Eğer müşteri direkt olarak merkezi birim ile bağlantılı değilse (Oluşturulan rotada müşteri merkezi birimden sonra gelmiyorsa) ve müşteriye en yakın başka müşteri ile merkezi birim aynı mesafede ise süreç yine ikiye ayrılarak yeni çözümler oluşturulur.

Rotalamada ilk önce müşteri başka müşteriye araç kapasite kısıtı sağlıyorsa bağlanır. Ardından çözüm ağacında yeni bir dal oluşturularak müşteri direkt olarak merkezi birime bağlanır.

Adım 4: Çözümler ayrı ayrı hesaplanarak en uygun çözüm seçilir.

Ancak bir orijin noktasından çok sayıda varış noktasına ya da çok sayıda orijin noktasından çok sayıda varış noktasına en kısa mesafenin bulunması durumu söz konusu olduğu zaman yapılan işlemler daha da karmaşık hale gelmektedir. Bu gibi durumlarda doğrusal programlama yardımıyla çözüm yolu bulunmaktadır [28].

3.6.2.4 İki Aşamalı Yöntem

İki Aşamalı Yöntem, KKARP problemlerini çözmek için geliştirilmiştir. Yöntem aşağıda açıklandığı gibi iki aşamadan oluşmaktadır [29]:

Aşama 1

Adım 1: $k=1$ olarak atanır.

Adım 2: Herhangi bir tura dâhil olmayan müşteriler (s) seçilerek R_k turu oluşturulur. Bütün tura dâhil olmayan müşteriler ($i \neq s$) için aşağıdaki gibi δ_i hesaplanır.

$$\delta_i = c_{0i} + \lambda c_{is}, \quad \lambda \geq 1$$

Adım 3: $\delta_i^* = \min[\delta_i]$ olan ve olurlu olan, k rotasına (R_k) i^* seçilerek eklenir. r - Opt kullanılarak R_k optimize edilir ve bu adım, R_k turuna başka bir müşteri eklenemeyinceye kadar devam eder.

Adım 4: $k = k + 1$ olarak atanır ve 2.adım ve 3.adım bütün müşteriler turlara Alınana kadar tekrar edilir.

Aşama 2

Adım 1: $h, \bar{R}_r = (0, i_r, 0)$ h olan 1. aşamadan elde edilen tur sayısı ve R_r 'den

Seçilen müşteri i_r olsun. $K = (\bar{R}_1, \dots, \bar{R}_h)$ olarak atanır.

Adım 2: Her $\bar{R}_k \in K$ ve her tura alınmayan j noktası için, $\mu \geq 1$ olan $\epsilon_{rj}A = c_{0j_r} + \mu c_{ji_r} - c_{0i_r}$ ve $\epsilon_{r^*j} = \min[\epsilon_{rj}]$ hesaplanır.

Adım 3: $\bar{R}_r \in K$ seçilir ve $K = K \setminus \bar{R}_r$ olarak atanır. Her bir j için olan $\epsilon_{r'j} = \min_{R_r \in K} [\epsilon_{rj}]$ olan $\bar{\delta}_j = \pi r^2 \epsilon_{r'j} - \epsilon_{rj}$ hesaplanır.

Adım 4: $\delta_j^* = \max[\delta_i]$ olan j^* seçilir ve $\bar{R}_r$ 'ye eklenir. r-Opt kullanılarak $\bar{R}_r$ optimize edilir ve başka bir kenar kalmayana kadar 3. adım tekrar edilir.

Adım 5: Eğer $K \neq \emptyset$ ise, 2. adıma git. Aksi halde eğer bütün kenarlar tura dâhil edilmiş ise dur, aksi halde 1. Aşamadaki 2. adıma git.

3.6.3 Meta sezgisel Yöntemler

Çözüm uzayını etkin bir şekilde aramayı sağlayacak temel sezgisel yöntemleri birleştirmeye çabalayan yeni yaklaşık yöntemlerin geliştirilmesidir[34].

Meta sezgisel, arama uzayında araştırma ve işletme için farklı kavramları zeki bir şekilde birleştirerek alt seviye sezgisellere rehberlik eden iteratif üretim sürecidir[34].

Meta sezgisel, yüksek kaliteli çözümleri etkin bir şekilde üretmek için altseviye sezgisellere rehberlik eden bir iteratif üst seviye procestir. Her iterasyonda, bir çözümü ya da çözümlerin bir topluluğunu kullanır. Alt seviye sezgiseller, basit yerel arama algoritması ya da çözüm kurucu bir yöntem olabilir[34].

Meta sezgisel, arama uzayının yüksek kaliteli çözümlerini kapsayan bölgelerinde aramayı gerçekleştirmek için probleme özgü sezgisellere rehberlik etmek amacıyla tasarlanan genel amaçlı sezgisel yöntemdir[27].

Meta sezgisel yöntemlerden bazıları, sınıflandırmaya tabi tutmadan, şunlardır: Genetik Algoritmalar, Karınca Kolonisi, Yapay Sinir Ağları, Tabu Arama, Bulanık Mantık.

3.6.3.1 Genetik Algoritmalar

Genetik Algoritmalar evrimsel hesaplamanın bir parçasıdır. Bu alan Yapay Zekânın hızla gelişen bir dalıdır. Genetik algoritmalar Darwin' in evrim teorisinden etkilenecek geliştirilmiştir. Basitçe açıklayacak olursak problemler evrimsel bir süreç kullanılarak bu süreç sonunda en iyi sonucu veren çözüme erişmeye çalışmaktadır. Başka bir ifadeyle çözüm evrimleşmektedir. Evrimsel hesaplama 1960'larda I.Rechenberg'in Evrim Stratejileri (Evolutionstrategie) adlı çalışmasında tanıtılmıştır. Daha sonra fikri diğer araştırmacılar tarafından geliştirilmiştir. Genetik algoritmalar John Holland tarafından icat edilmiş ve öğrencileri ve iş arkadaşları tarafından geliştirilmiştir. 1992 yılında John Koza, genetik algoritmaları kullanarak programları evrimleştirerek belli işleri yapmakta kullandı. Bu yönteme "genetik programlama" adını verdi. LISP dilinde programlar Ayırıştırma Ağaçları" şeklinde ifade edildiği için LISP diliyle geliştirilmiştir. Ayırıştırma Ağaçları genetik algoritmaların çalıştığı temel nesnedir [30].

Genetik algoritmalar Darwin'in Evrim teorisinden esinlenilerek üretilmiştir. Bir problemin çözümü evrimsel süreç kullanılarak çözülmektedir. Algoritma toplum adı verilen ve kromozomlarla temsil edilen bir çözüm kümesi ile başlamaktadır. Bir toplumdaki çözümler yeni toplumların üretilmesinde kullanılmaktadır. Bu işlem, yeni toplumun eskisinden daha iyi olacağı umuduyla yapılmaktadır. Yeni çözümler (yavru) üretmek için alınan çözümler uygunluklarına göre seçilmektedir. Daha uygun olan tekrar üretim için daha fazla şansa sahiptir. Bu süreç belli bir durum (örneğin belli sayıda toplum veya en iyi çözümün gelişmesi) karşılanana kadar tekrar edilmektedir [30].

3.6.3.2 Bulanık Mantık

İngilizce fuzzy kelimesinin sözlük anlamı bulanık, hayal mayal'dir. İlk defa Prof. Zadeh tarafından kullanılan bu terim temelde mantık, olasılık kuramı, yapay zeka ve yapay sinir ağları alanları üzerine oturtulmuş, olayların oluşum olasılığından çok oluşum derecesiyle ilgilenen bir kavramı tanımlar. Olasılık ve bulanıklık kavramları arasındaki en önemli farklılık bulanıklığın bir deterministik belirsizlik olmasıdır.

Bulanık denetim kuramı temelde insan düşünüş tarzını örnek alır. Oldukça kapsamlı ve ayrıntılı bir matematiksel temeli varsa da, ana özellikleri aşağıdaki şekilde açıklanabilir [31].

Geleneksel mantıkta bir kümeyi oluşturan elemanlar kesin elemanlar olup bir eleman kümenin ya elemanıdır ya da değildir (var veya yok, 0 veya 1). Bu tür kümelere kesin kümeler denilir. Bir örnek olarak orta yaşlı kavramı ele alındığında, eğer 40 yaş orta yaş olarak kabul edilecek olunursa, geleneksel kümelendirmede 30 yaşın altındaki kişiler "genç", 30-50 arası "orta yaşlı", 50 yaşın üstünde "yaşlı" kümelerine sokulabilir. Dolayısıyla 29.5 yaşındaki birisi "genç" iken, 30.5 yaşındaki diğer bir kişi "orta yaşlı" olarak anılacaktır. Bir endüstriyel denetleyici için bu durum ele alınacak olursa, eğer bu denetleyicide fiziksel büyüklüklerin dahil olduğu kümeler birbirlerinden böyle kesin çizgilerle ayrılmışlarsa denetim çıktısının ani değişiklikler göstermesi kaçınılmaz olacaktır. Örneğin soğuk /sıcak sınırının 25° olduğu bir sayısal açık/kapalı denetleyicide 24.5° soğuk olarak algılanacak, buna karşın 25.5° sıcak olarak ele alınarak denetim çıktısı ani olarak değiştirilebilecek, örneğin buhar vanası ani olarak kapatılabilecektir [31].

Yukarıda açıklananlara karşıt olarak bulanık mantık, keskin mantığı açık /kapalı, soğuk/sıcak, hızlı/yavaş gibi ikili denetim değişkenlerinden oluşan keskin dünyayı, az açık/az kapalı, serin/ılık, biraz hızlı/biraz yavaş gibi gevşek denetleyicilerle yumuşatarak gerçek dünyamıza benzetir. Otuz beş yaşındaki bir insana pek orta yaşlı denemeyeceği gibi o kişi pek gençte sayılmaz, duruma göre belki genç tanımı, belki de orta yaşlı tanımı daha uygun düşer. İşte bulanık kümeler böyle esnek bir düşünüşe olanak tanır. Kümelerin birbirinden keskin çizgilerle ayrılmamış olması, aralarında belirli bir örtüşüm olması, otuz beş yaşın bir oranda hem orta yaşlı, hem genç; ısı denetleyicisi örneğinde

ise 20° sıcaklığın hem biraz serin hem de biraz sıcak olarak düşünülmesine olanak tanır [31].

3.6.3.3 Uzman Sistemler

Uzman sistemler olarak anılan bilgi tabanlı yazılım sistemleri, belirli bir alanın önde gelen uzmanlarından derlenmiş bilgilere dayanarak düzenlenmiş çok sayıdaki "eğer... ise ... dir" biçimindeki kuraldan oluşmaktadır [32]. Bir uzman sistem genellikle bilgi kazanma, bilgi tabanı, çıkarım mekanizması, çalışma alanı, kullanıcı arabirimi, açıklama, çıkarım mekanizmasını iyileştirme gibi bileşenlerden oluşur.

Bilgi kazanma: Bazı bilgi kaynaklarından bir bilgisayar programına problem çözümü için bilgi aktarma ve dönüştürme işlemleri yapılır. Potansiyel bilgi kaynakları; uzman insanlar, kitaplar, veri tabanları, özel araştırma raporları ve kullanıcının kendi deneyimleri olabilir [32].

Bilgi tabanı: Bilgi tabanı problemlerin anlaşılması, formülasyonu ve çözümü için gerekli olan tüm bilgileri içerir. Örneğin olaylar ve durumlar hakkında bilgi ve bunlar arasındaki mantıksal ilişki yapılarını ihtiva eder. Ayrıca standart çözüm ve karar alma modellerini de içerir [32].

Çıkarım mekanizması: Uzman sistemin beynidir. Bilgi tabanı ve çalışma alanında bulunan bilgiler üzerine düşünmek için bir metodoloji sunan ve sonuçları biçimlendiren bir bilgisayar programıdır. Bir başka deyişle problemlere çözümler üreten bir mekanizmadır. Burada sistem bilgisinin nasıl kullanılacağı hakkında karar alınır [32].

Çalışma alanı: Giriş verileri tarafından belirlenmiş problem tanımları için hafızanın bir köşesinde bulunan çalışma alanıdır. Bu alan işlemlerin ara seviyelerindeki sonuçları kaydetmek için de kullanılır [32].

Kullanıcı arabirimi: Uzman sistemler, kullanıcı ile bilgisayar arasında probleme yönelik iletişimin sağlanması için bir dil işleyici içerir. Bu iletişim, en sağlıklı doğal dil ile yapılır. Kısaca kullanıcı ara birimi kullanıcı ile bilgisayar arasında bir çevirmen rolünü üstlenmiştir [32].

Açıklama: Uzman sistemleri diğer sistemlerden farklı yapan bir özelliği de açıklama modülünün olmasıdır. Açıklama modülünden kasıt, kullanıcıya çeşitli yardımların

verilmesi ve soruların açıklanması olduğu kadar, uzman sistemin çıkardığı sonucu nasıl ve neden çıkardığını açıklayabilmesidir. Burada uzman sistem karşılıklı soru cevap şeklinde davranışlarını açıklar [32].

Çıkarım mekanizmasını iyileştirme: Bir uzman insan kendi performansını analiz edebilir, öğrenebilir ve gelecekteki kullanım için onu iyileştirebilir. Sistemlerin de bu tip davranışlar göstermeye ihtiyacı vardır. Sistemin kendini iyileştirmesi öğrenme ile ilgili bir konudur. Sistemlerin bir uzman insan gibi öğrenebilmelerine yönelik çalışmalar sinirsel ağlar üzerinde sürdürülen araştırmalarla devam etmektedir. Amaç bir insan beyni gibi çalışan yapay zekâyı geliştirebilmektir [32].

3.6.3.4 Yapay Sinir Ağları

Yapay sinir ağları, insan beyninin özelliklerinden olan öğrenme yolu ile yeni bilgiler türetebilme, yeni bilgiler oluşturabilme ve keşfedebilme gibi yetenekleri herhangi bir yardım almadan otomatik olarak gerçekleştirmek amacı ile geliştirilen bilgisayar sistemleridir. Bu yetenekleri geleneksel programlama yöntemleri ile gerçekleştirmek oldukça zordur veya mümkün değildir. O nedenle, yapay sinir ağlarının programlanması çok zor veya mümkün olmayan olaylar için geliştirilmiş adaptif bilgi işleme ile ilgilenen bir bilim dalı olduğu söylenebilir.

Yapay sinir ağları; olayların örneklerine bakmakta, onlardan ilgili olay hakkında genellemeler yapmakta, bilgiler toplamakta ve daha sonra hiç görmediği örnekler ile karşılaşınca öğrendiği bilgileri kullanarak o örnekler hakkında karar verebilmektedir.

1990'lı yıllardan beri bilgisayarların öğrenmesini sağlayan Yapay Sinir Ağları teknolojisinde oldukça hızlı bir gelişme görüldü. Yapay sinir ağları, insan beyninin özelliklerinden olan öğrenme yolu ile yeni bilgiler türetebilme, yeni bilgiler oluşturabilme ve keşfedebilme gibi yetenekleri herhangi bir yardım almadan otomatik olarak gerçekleştirmek amacı ile geliştirilen bilgisayar sistemleri olduklarından hem yeni gelişmelere neden oluyor hem de nasıl çalıştığı bilinmeyen insan beyni hakkında yapılan araştırmalara da önemli katkılar sağlıyordu [33].

Yapay sinir ağları başlıca; Sınıflandırma, Modelleme ve Tahmin uygulamaları olmak üzere pek çok alanda kullanılmaktadır. Başarılı uygulamalar incelendiğinde, YSA'ların

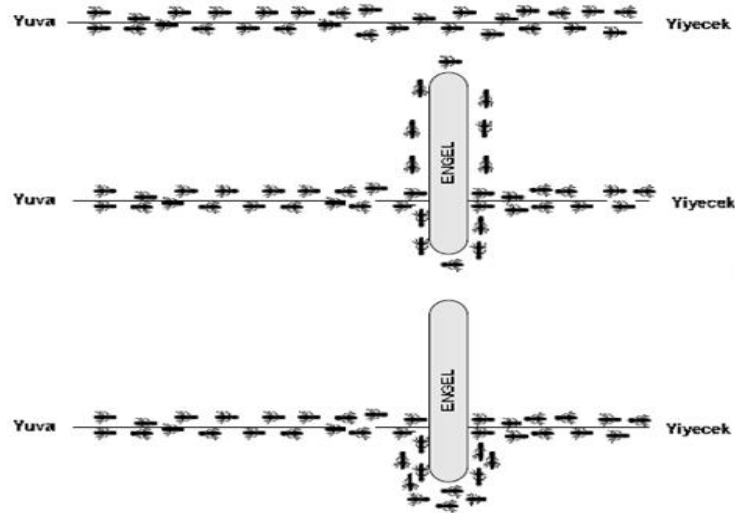
çok boyutlu, gürültülü, karmaşık, kesin olmayan, eksik, kusurlu, hata olasılığı yüksek sensor verilerinin bulunduğu ve problemi çözmek için matematiksel modelin ve algoritmaların bulunmadığı, sadece örneklerin var olduğu durumlarda yaygın olarak kullanıldıkları görülmektedir.

3.6.3.5 Karınca Kolonisi

Karınca algoritmaları ilk olarak Dorigo ve meslektaşları tarafından; gezgin satıcı problemi (GSP) ve kuadratik atama (QAP) gibi zor optimizasyon problemlerinin çözümü için geliştirilmiştir. Optimizasyon problemlerinin çözümü amacıyla karınca algoritmaları üzerine birçok çalışma devam etmektedir. Bu çalışma alanlarından bazıları, iletişim ağlarının belirlenmesi, grafik renklendirme, iş çizelgeleme vs.dir [34].

Karıncalar, yiyecek kaynaklarından yuvalarına en kısa yolu görme duyularını kullanmadan bulma yeteneğine sahiptirler. Aynı zamanda, çevredeki değişime adapte olma yetenekleri vardır. Dış etkenler sonucu takip ettikleri mevcut yol artık en kısa yol değilse, yeni en kısa yolu bulabilmektedirler [35] [36].

Şekil 3.4’de de görüldüğü gibi karıncalar, başlangıçta düz bir hattı takip etmekte ve bu esnada feromon olarak adlandırılan bir maddeyi yol güzergâhına bırakarak kendilerinden sonra gelen karıncaların yollarını bulmalarını kolaylaştırmaktadırlar. Önlerine bir engel konulduğunda feromonları takip edemediklerinden, karıncalar gidebilecekleri iki yoldan birini öncelikle rastsal olarak seçmektedirler. Kısa olan yoldan birim zamandaki geçiş daha fazla olacağından bırakılan feromon miktarı da daha fazla olur. Buna bağlı olarak, zaman içerisinde kısa olan yolu tercih eden karıncaların sayısında artış olur. Belli bir süre sonra tüm karıncalar kısa yolu tercih ederler.



Şekil 3. 4 Gerçek karıncaların en kısa yolu bulması

Başta rastsal hareket eden karıncaların izleri kontrol ederek yüksek olasılıkla izlerin yoğun olduğu yönü takip etmesi otokatalitik bir davranış şeklidir ve karıncaların karşılıklı etkileşiminde sinerjik bir etki vardır. Algoritma, karınca kolonilerinden esinlenerek geliştirildiğinden sisteme, karınca sistemi (KS), algoritma ise karınca kolonileri algoritması (KKA) olarak adlandırılır. Karınca kolonileri optimizasyon problemlerinde kullanılır. Karınca sistemindeki karıncalar doğal karıncalardan farklıdır. Hafızaya sahiptirler, tamamen kör değildirler ve zamanın kesikli olduğu bir çevrede yaşarlar [37].

Yapay karıncalardan oluşan Karınca Kolonisi Algoritması, yapay feromon izlerinin güncelleştirilmesiyle tekrarlanan bir yapıya sahiptir. Algoritmanın çalışma sürecinde, karıncalar tarafından güncellenen feromon izleriyle iyi bir çözümün bulunması için bilgi oluşturulmakta ve her iterasyonda bu bilgiler güncellenmektedir. Karınca kolonisi algoritmasına ait döngüdeki gibidir [34].

Adım 1: Başlangıç feromon değerleri belirlenir.

Adım 2: Karıncalar her düğüme rastsal yerleştirilir.

Adım 3: Her karınca, sonraki şehri denklemde verilen lokal arama olasılığına bağlı olarak seçmek suretiyle turunu tamamlar.

Adım 4: Her karınca tarafından kat edilen yolların uzunluğu hesaplanır ve lokal feromon güncellemesi yapılır.

Adım 5: En iyi çözüm hesaplanır ve global feromon yenilemesinde kullanılır.

Adım 6: Maksimum iterasyon sayısı ya da yeterlilik kriteri sağlanana kadar adım 2 ye gidilir.

Algoritmanın çalışma sürecinde temel işlemler, yapay karıncaların turları sonunda geçmiş oldukları yolların feromon miktarlarının arttırılması, belirli bir oranda feromon buharlaşmasının gerçekleştirilmesi, en iyi çözümün bulunması, buna bağlı olarak global feromon güncellemesinin yapılması ve karıncaların yenilenen bu feromon miktarlarına bağlı olarak yeni turlarını gerçekleştirmeleridir. Tüm bu işlemlere ait hesaplama yöntemleri ayrıntılarıyla alt başlıklarda verilmiştir.

3.6.3.6 Tabu Arama

GSP için ilk Tabu Arama yöntemi Glover tarafından 1986 yılında önerilmiştir. Bu yöntem 2-kenar değişimi stratejisinden uyarlanmıştır. Önerilen tabu liste boyutu ve esinlenme ölçütleri farklı olacak şekilde Tabu Arama yönteminin farklı türleri bulunmaktadır.

GSP için Tabu Arama ile 2-kenar değiştirme yöntemi, mevcut turdan 2 yakın olmayan kenarın silinmesi ve olurlu bir tur oluşturacak şekilde 2 farklı kenarın eklenmesi şeklinde ifade edilir. Tabu Arama yöntemi aşağıdaki gibidir [38]:

Adım 1: Rastgele veya bazı başlangıç tur oluşturma yöntemleri kullanılarak başlangıç turu oluşturulur.

Adım 2: 2-kenar değiştirme seti oluşturulur ve en uygun aday belirlenir.

Adım 3: En uygun aday uygulanır.

Adım 4: Tabu listesi, esinlenme ölçütleri, diğer değişkenler ve şu ana kadarki bulunan en iyi tur güncellenir. Eğer durma ölçütlerine ulaşılmış ise 5. Adıma geçilir. Aksi halde 2. Adıma gidilir.

Adım 5: Eğer bulunan son tur şu ana kadarki en iyi tur ise, global en iyi tur güncellenir.

Adım 6: Eğer daha önceden belirlenen arama sayısına ulaşılmış ise çıktı en iyi tur olur ve işlem durdurulur. Aksi halde 1. Adıma gidilir.

UYGULANACAK YÖNTEMİN BELİRLENMESİ

4.1 Araç Rotalama Problemi İçin Uygun Bir Sezgisel Algoritmanın Özellikleri

Cordeau ve arkadaşları ARP için iyi sonuçlar veren bir sezgisel algoritmanın dört temel özelliğe sahip olması gerektiğini vurgulamışlardır. Bu özellikler ve açıklamaları aşağıda yer almaktadır [39]:

1. Kesinlik: Literatürde kesinlik kavramı, bir algoritmanın ürettiği değerin problemin optimum çözümüne oranı olarak bilinir. ARP literatüründe bazı büyük boyutlu problemler için arama uzayı tam olarak taranmadığı ve dolayısıyla optimum çözüm kesin olarak bilinmediği için çoğu karşılaştırmalar bilinen en iyi çözüm ile yapılmaktadır. Bundan dolayı karşılaştırma için internette [40] araştırmacıların ortak kullandığı soru kütüphaneleri bulunmaktadır. Böylece yapılan çalışmalarda önerilen yöntemin bu problemler için verdiği çözüm değerleri verilmekte ve bu da araştırmacılara iyi bir kıyaslama imkânı sunmaktadır. Yazarlar kesinlik kavramının ARP için çok önemli olduğunu belirtmekte, iki nokta arasındaki mesafe değerlerindeki yuvarlamalara dikkat çekmektedirler. Örneğin, bazı büyük boyutlu problemlerde mesafe değerinin alt veya üst tamsayıya yuvarlanması amaç fonksiyonu değerinde %3'lük bir değişime sebebiyet verebilmektedir. Bunun yanında KKARP kütüphanesinde yer alan sorular değişik özelliklerde olduğu için, geliştirilen sezgisel yöntem sadece bazı sorularda iyi sonuç vermemeli, tüm sorularda optimuma yakın sonuçlar verebilmelidir. Son olarak yazarlar iyi bir sezgisel algoritma işlerken, zaman geçtikçe sürekli daha iyi sonuçlar üretmesi gerektiğini vurgulamışlar ve iyi bir sezgiselin genellikle işleyiş sürecinin erken

zamanlarında iyiye yakın bir çözüm bulunduğunu ve eğer optimum sonuç bulunamamışsa algoritmanın sürekli aynı çözümde takılıp kalmadığını, sürekli bulduğu çözümleri geliştirdiğini belirtmişlerdir [39].

2. Hız: Bilgisayar hesaplama hızının günümüzde son derece artmış olması, ARP gibi kombinatoryal problemlerin iteratif süreçlerle çözümlerinin kısa zamanda gerçekleşmesini mümkün kılmıştır. Özellikle gerçek sistemlerde yapılan uygulamalarda çözüm süresi önemli bir faktör teşkil etmekte, uygulanabilir bir çözüm bulmak için problemin iyi bir şekilde analiz edilip bellek-işlemci kaynak kullanımını doğru planlanarak yazılım diline dökülmesi gerekmektedir. Bir lojistik firmasının günlük araç rotalarının çıkartılması gibi büyük boyutlu problemlerde hesaplama süresinin 10 - 20 dakikayı bulması, toplam maliyetin azaltılması açısından mantıklı gözükse de, özellikle gerçek zamanlı interaktif uygulamalarda örneğin saha satış planlarının yapılmasında anlık talep değişiklikleri olabileceği için, son derece hızlı sonuçlar veren algoritmaların geliştirilmesi daha uygundur. Bu tip algoritmalarda kayıp, üretilen sonucun kesinliğinin düşük olması başka bir deyişle optimum sonuca uzak çözümlerin oluşturulmasıdır. Bunların yanında yazarlar, paralel programlamanın doğru bir şekilde kullanılabilirse ARP için sezgisel yöntemlerle hızlı çözümler üretilebileceğinin üzerinde durmuşlardır [39].

3. Basitlik: Çoğu sezgisel yöntemin gerçek zamanlı sistemlere uygulanmayışının sebeplerinden biri anlaşılmasının ve kodlanmasının zor olmasıdır. Algoritma geliştiriciler yayınladıkları çalışmalarında algoritmanın her adımını detaylı bir şekilde açıklamaları gerekmektedir. Bilhassa problem bazında geliştirilmiş ve çok fazla unsuru bir arada barındıran bir yöntemin başka tip bir probleme uygulanmasında zorluklar yaşanabilmektedir. Bunun yanında çok fazla sayıda parametre değerinin belirlenmesi gereken algoritmaların kullanımı da ayrı bir zorluk teşkil etmekte, hangi durumlarda parametrelerin hangi değerleri alacağı muallakta kalmaktadır. Bu sorun özellikle son 10 yıldır geliştirilmekte olan meta sezgisel yöntemlerde sıkça görülmektedir. Yazarlar buna çözüm olarak geliştiricilere, işleyiş süresince sabit tutulan veya parametrelerine kendi kendine değerler atayabilen mekanizmalar kurulmasını önermişlerdir [39].

4. Esneklik: Yazarlara göre iyi bir ARP sezgisel algoritması çeşitli kısıtlar ile entegre olarak gerçek sistemlere uygulanabilir esneklikte olmalıdır. Genellikle ARP literatüründe kapasite ve mesafe kısıtlı problemler yer alıyorsa da yeni bir yan kısıt eklendiğinde algoritmada ne tür bir değişiklik yapılması gerektiğinden çoğunlukla bahsedilmemektedir. Modele bu tür değişikliğin yapılması genellikle algoritma performansında büyük ölçüde düşüşe sebebiyet vermektedir. Yazarlar bir ARP sezgiseline yeni kısıtlar eklemek için en iyi yolun, amaç fonksiyonuna her yeni kısıtla ilgili olarak ağırlıklandırılmış ceza terimlerinin eklenmesi olduğunu belirtmişlerdir. Eğer $Q(x)$ ve $D(x)$ fonksiyonları sıra ile kapasite ve süre kısıtları ile ilgili ceza terimlerini ifade ederse yeni amaç fonksiyonu $F'(x) = F(x) + \alpha Q(x) + \beta D(x)$ halini almaktadır. Bu formülasyonda $F(x)$, toplam maliyetin (toplam gidilen mesafe) minimum hale getirilmesi ile ilgili fonksiyon olurken α ve β ise algoritmanın kendi içinde değerlerini ayarlayabildiği ceza ağırlıklardır. Yazarlara göre ilk olarak 1 değeri atanan bu ağırlıkların değerleri, iterasyonlarda elde edilen çözümlerin uygun olup olmamasına göre periyodik olarak artırılabilir veya azaltılabilir. Bu sayede arama uzayı daha etkili bir şekilde taranırken, algoritmanın lokal minimum noktalarına takılma olasılığı da azaltılmış olur. Bu şekilde oluşturulan bir modelin diğer bir avantajı da iterasyonlarda yeni çözümlerin, örneğin noktaların rotalar arasında yer değiştirmesi gibi basit bir şekilde üretilmesidir. Böylece algoritmik esneklik, tasarımın basitleştirilmesi ile elde edilebilmektedir. Sezgisel yöntemler genel olarak Yapısal, Çok Aşamalı ve İyileştirmeli olmak üzere üç kategoriye ayrılmaktadır. Yapısal sezgisel algoritmalarda amaç fonksiyonu göz önüne alınarak adım adım uygun bir çözüm oluşturmaya çalışılır, fakat çoğunlukla bulunan bu çözüm üzerinde iyileştirilme yapılmaz. Çok Aşamalı sezgisellerde problem birden fazla parçaya bölünerek ve her parça için ayrı süreçler izlenerek çözüm bulunmaya çalışılır. İyileştirmeli sezgisel algoritmalarda ise uygun bir çözüm ele alınır, bu çözümün komşu çözümleri taranarak daha iyi bir çözüm elde edilmeye çalışılır. Yapısal ve iyileştirmeli sezgisel metotlar birleştirilerek yeni yöntemler oluşturulabilir [39].

4.2 Meta sezgisel Yöntemlerin Kıyaslanması

Literatürde rotalama problemini çözen birçok algoritma yer almaktadır. Bu bölümde öncelikle temel algoritmalarından kullanılmayacak olanların nedenleri üzerinde durulmuştur.

4.2.1 Bulanık Mantık Algoritması Seçilmeme Nedenleri

Bulanık mantığın temeli bulanık küme ve alt kümelerine dayanır. Klasik yaklaşımda bir varlık ya kümenin elemanıdır ya da değildir. Matematiksel olarak ifade edildiğinde varlık küme ile olan üyelik ilişkisi bakımından kümenin elemanı olduğunda "1", kümenin elemanı olmadığı zaman "0" değerini alır. Bulanık mantık klasik küme gösteriminin genişletilmesidir. Bulanık varlık kümesinde her bir varlığın üyelik derecesi vardır. Varlıkların üyelik derecesi, $(0, 1)$ aralığında herhangi bir değer olabilmektedir [41].

Örnek olarak normal oda sıcaklığını 23 derece olarak kabul edersek klasik küme kuramına göre 23 derecenin üzerindeki sıcaklık derecelerini sıcak olarak kabul ederiz ve bu derecelerin sıcak kümesindeki üyelik dereceleri "1" olur. 23 altındaki sıcaklık dereceleri ise soğuktur ve sıcak kümesindeki üyelik dereceleri "0" olur. Soğuk kümesini temel aldığımızda bu değerler tersine döner. Bulanık küme yaklaşımında üyelik değerleri $[0,1]$ aralığında değerler almaktadır. Örneğin 14 derecelik sıcaklık için üyelik derecesi "0", 23 sıcaklık derecesi için üyelik değeri "0,25" olabilir [41].

Klasik kümelerin aksine bulanık kümelerde elemanların üyelik dereceleri $[0, 1]$ aralığında sonsuz sayıda değişebilir. Bunlar üyeliğin derecelerinin devamlı ve aralıksız bütünüyle bir kümedir. Keskin kümelerdeki soğuk-sıcak, hızlı-yavaş, aydınlık-karanlık gibi ikili değişkenler, bulanık mantıkta biraz soğuk, biraz sıcak, biraz karanlık gibi esnek niteleyicilerle yumuşatılarak gerçek dünyaya benzetilir. En önemli fark, böyle bir çatıda bilginin kaynağındaki küme üyeliğinin kesin tanımlanmış önkoşullarının olmayışı ve daha çok problemlerle rastgele değişkenlerin hazır bulunmasındadır.

Çizelge4. 1 de klasik mantık ve bulanık mantık arası farklar özetlenmiştir.

Çizelge 4. 1 Klasik mantık ile bulanık mantık arasındaki temel farklılıklar

Klasik Mantık	Bulanık Mantık
A veya A Değil	A ve A Değil
Kesin	Kısmi
Hepsi veya Hiçbiri	Belirli Derecelerde
0 veya 1	0 ve 1 Arasında Süreklilik
İkili Birimler	Bulanık Birimler

İkili birimler, iki ayrık değer alabilen değişkenleri ve mantıksal anlam taşıyan işlemleri ele alır. Değişkenlerin alabileceği iki değer farklı şekillerde adlandırılabilir (örneğin doğru ve yanlış, evet ve hayır, vs.) burada her değişken ancak ve ancak olası iki ayrı değerden birini alabilir: 1 ve 0. [42]. Bulanıklık, çoklu değerlilik (multi – valued) demektir. İkili mantığın 0-1 önermelerine karşın bulanıklık, üç veya daha fazla, belki de sonsuz sayıda önermeler yapar. Yani bu mantıkta küme üyeleri derecelendirilebilir. Başka bir deyişle siyah ile beyaz arasında yer alan sonsuz sayıda gri tonlarını içermektedir. Örneğin uzaklıkla ilgili bir problemde mesafenin yalnızca yakın ya da uzak olduğunu belirtmekle kalmayıp ne kadar yakın ya da ne kadar uzak olduğunu da belirtir [42].

Normal bir programın yapısı: Temel girdiler → Program → Sabit bir sonuç şeklindedir.

Oysaki bir bulanık mantık uygulaması: Sayısı belli olmayan veri yığını → Program → Girdilere ve varsayıma göre değişken bir veya birden fazla sonuç şeklindedir.

Bir bulanık mantık uygulamasındaki sonuç, aynı girdiler olsa bile değişik bir sonuç döndürebilir ve bir öbek halinde veriyi alabilir. Bulanık mantıktaki özellik bunun haricinde verilen verilerin örnekleme mantığı ile alınması ve tümü simgelediği varsayımı yapılması ve buna göre bir olasılık değerinin elde edilmesidir [42].

Bulanık Mantığın avantajlarını şöyle sıralayabiliriz:

- İnsan düşünme tarzına yakındır.
- Matematiksel modele ihtiyaç duyulmaz.
- Yazılımı basit ve ucuzdur.
- Bulanık mantıkta sonuç hayır ve evetten ibaret değildir. Bunların ara değerlerini de içerir. Hem denetim kalitesi arttırılmış olur hem de enerji tasarrufu sağlanmış olur.

- İnsana özgü tecrübe ile öğrenme olayının kolay modellenenebilmesi ve belirsiz kavramların bile matematiksel olarak ifade edilmesine imkân tanınmasıdır.
- Hız bakımından problemin karmaşıklığı önemli değildir.

Bulanık Mantığın dezavantajlarını şöyle sıralayabiliriz:

- Kararlılık analizinin yapılışının zorluğu
- Üyelik fonksiyonlarının deneme yanılma yoluyla bulunmasından dolayı uzun zaman alabilmesi
- Uygulamada kullanılan kuralların oluşturulmasının bulanıklığa bağlılığı gibi dezavantajları vardır.

Sonuçta yapılan değerlendirme şu şekildedir:

Terimler ya da ölçüler kesin olarak tanımlanıp ölçülemediğinden dolayı insanlar çoğu zaman belirsiz (kesin olmayan) ifadeler kullanırlar. İşte bulanık mantık bazı sorulara basitçe evet-hayır cevabı verilemeyen durumları kapsar. Bulanıklığın ve bulanık mantığın temeli de budur. Uygulamanın kurallarının oluşturulmasında bulanıklığa bağlılık dezavantajı olarak gösterilebilir [42].

Bulanık mantık bir matematiksel modele ihtiyaç duymaz fakat ARP çözümü için matematiksel model kullanımı şarttır.

Bizim problemimizin verileri sayısal ifadelerden oluşmaktadır ve de sonuç olarak kesin bir değere ulaşılması hedeflenmektedir. Seçilecek yöntem ARP mantığına en iyi entegre olmalı, en optimal sonucu vermelidir. Problemin mantığına uyuşmayan bu yöntemi seçmek olanak dışıdır.

4.2.2 Uzman Sistemler Seçilmeme Nedenleri

Uzman sistemler, belirli bir uzmanlık alanında, gerçek kişilerden derlenen bilgileri temel alarak, zamanla kendisini geliştirebilme yeteneği de olan yazılımlardır. Uzman davranışını taklit etmek üzere çok farklı metotlar kullanılmakla birlikte, en yaygın kullanılanlar 1) bilgi temsili yöntemleri ve 2) bilgi mühendisliği yöntemleridir. Her uzman sistemde öğrenme davranışı olmayabilir fakat her uzman sistem geliştirildikten sonra gerçek problemler karşısında insan uzmanla aynı sonuca varmalıdır [43].

1970'lerde yapay zekâ alanındaki araştırmacılar tarafından geliştirilmiş ve ticari olarak 1980'lerde uygulanmaya başlanmıştır. Bu programlar, belirli bir problem hakkındaki bilgiyi çözümleyen, problemlere çözümler sağlayan, tasarımına bağlı olarak, düzeltmeleri yapmak için bir iş dizisi öneren programlardır.

Özellikle tıp ve danışmanlık gibi hizmet sektörlerinde, uzman eksikliğini giderme veya maliyetleri düşürmek amacıyla kullanılırlar. Uzman sistemler konusu; sistem teorisi, yöneylemsel araştırma, iş süreçleri, uygulamalı matematik ve bilim yönetimi konularıyla ilişkilidir [43].

Uzman Sistemlerin avantajlarını şu şekilde sıralayabiliriz [43]:

- Üretim Artışı: Uzman bir sistem Uzman bir insandan daha hızlı hesaplar yapıp kararlar verebileceğinden zamandan tasarruf edilir ve bu sayede örneğin üretim artışı sağlanabilir.
- Süreklilik: İnsani hatalar ve yıpranmalar söz konusu olmadığı için zamanla oluşabilecek veri ve bilgi kaybı yoktur. Yeterince iyi bir Dinamik yapı oluşturabilirseniz kendi kendini günceller, bilgi kazanabilir.
- Maliyet Tasarrufu: Zor bulunan ve maliyetli uzmanların yaptığı işi sürekli yaparak tasarruf ve dolaylı yoldan üretim artışı sağlayabilir.
- Kalite Düzeyinin Yükselmesi: Düşünerek/Hesaplayarak ve hata payı olmadan sonuca varması nedeniyle kaliteyi yüksek tutabilir.
- Uzmanlığın Yaygınlaştırılması: Birçok alanda uzman yetiştirmek sıkıntılı bir iştir. Bu işi yapan yaşlanmayan, unutmayan bir sistemin bilgilerini ve kurallarını aktarması sorunsuzdur. Dolayısıyla yeni uzman yetiştirmeye gerçek katkı sağlayabilir.
- Eğitim Verme: Konu hakkında uzman olmayanlara, nedenlerini de açıklayarak öğretebilir.
- Sağlıklı Öneri Üretimi: İnsani kusurlar olan acele etme, çabuk karar verme ve telaşa kapılma gibi nedenlerden dolayı sağlıklı karar verme ve öneri üretimi kısıtlamaktadır. Uzman sistemlerde ise yeterli düşünme, analiz etme sonuç üretmeyi çok kısa bir sürede yapabilir.

- Güvenilirlik: İyi tasarlanmış Uzman sistemler hiçbir detayı kaçırmadan son ayrıntısına kadar taradıktan sonra sonuca ulaşma yöntemini kullanır. Hiçbir belirsizlik yoktur.
- Tam ve Kesin Olmayan Bilgi ile çalışabilme: Bir insandaki tüm özellikleri taşıyarak “emin değilim”, “bilmiyorum” gibi kesinlik belirtmeyen ifadelere de yer verilmesi, değerlendirme ve kıstaslarını buna göre ayarlamasıdır.
- Ara Sonuçları Gösterebilme: Uzman sistemlerin kullanıldığı bazı alanlarda ara sonuçları, raporları görmek ilerisine yönelik plan ve tasarı yapmak daha uygundur. Böyle durumlarda istenildiği anda müdahale edilebilir.
- Sonuçlardan Yararlanma: Dinamik bit yapı oluşturulduğu takdirde öğrenebilme kabiliyetine sahip olduğu andan itibaren her sonucu birbiri ile kıyaslama yorum yürütme ve gerçek doğruyu bularak bilgilerde tutarlılık sağlayabilir.
- Geleceğin Fabrikalarının Gelişmesine Katkı: Gelecekte kurulacak olan insansız fabrika yapılarak uzman sistemler ile daha da gelişmesi sağlanabilir. Her türlü kazanç sağlanarak büyük bir atılım ile gelişme beklenmektedir.

Uzman Sistemlerin Dezavantajlarını şu şekilde sıralayabiliriz [43]:

Burada belirtmek istediğimiz bir uzman sistemin dezavantajlarından daha çok tasarım aşamasında veya kullanım aşamasında karşılaşılan güçlükler, ileriki zamanlarda yaşanabilecek problemleri içermektedir.

- Uzmanlık Bilgisinin Olmayışı: Günümüz koşullarında her alanda yeterli miktarda uzman bulamamak ve bulunsa dahi zaman ayıramamaktadır.
- Uzmanlardan Bilgi Edilmesindeki Güçlükler: Kimi uzmanlar bildiklerini aktaramamaktadır. Bir uzman sistem tasarlamamanın başlıca problemlerindendir. Bazı uzmanlar ise yeni teknolojiye ayak uydurmak istemeyip bilgi paylaşımından kaçınmaktadır.
- Aynı Konuya Uzmanların Farklı Bakış Açıları: Uygulanacak alana göre değişiklik göstermekle birlikte her uzmanın kendi benimsediği ve doğru olduğu bilgilerde tutarsızlık meydana gelebilir. Kimi durumlarda daha ekonomik veya daha doğru bir bilgi sunabilir. Oluşturulmuş olan uzman sistemde dinamik bir alt yapı oluşturulursa bu sorunun üzerinden gelinebilme imkânı olabilir.

- Dar Alana Yönelik Olması: Uzman sistemlerin çok dar bir alana hitap etmesi ve bunun sonucunda dışarı çıkılma gerçekleştiği zaman sistemin çalışmasında problemler meydana gelmesi anlamındadır. Örneğin her konuda bir uzman olduğu, bir uzmanın birden çok uzmanlık dalının bulunmadığı bulunsa da çok az sayıda olmasından dolayı alanları genişletilmeye başlatılırsa sistemde karışıklıklar başlayacaktır.
- Üst Düzey Yöneticilerin Tutuculuğu: Yeni teknolojiye şüpheci yaklaşım ve maliyet pahalılığı tüm yöneticileri düşündürmektedir. Aynı işi yapabilecek bir insan varken böyle bir bilgisayar programına maliyet ayıramamaktadırlar.
- Uzmanların Objektif ve Bağımsız Denetim Mekanizmaların Olmaması: Uzmanların birçoğu kendi geliştirdikleri, kişisel yöntemlerini kullanmaktadırlar. Bunları açıklayamamaları ve ister istemez objektif bir anlatıma sahip olamamalarından gelmektedir.
- Ortak Terminoloji Olmaması: Çeşitli kullanılan kavramların kişiden kişiye değişerek farklı anlamlar kazanması ve sözlük anlamı ile gerçek anlamının birbirinden farklı oluşu yüzünden doğan karışıklıklardır.
- Maliyet Yüksekliği ve Zaman Fazlalığı: Uzman sistem tasarlanmasının yüksek maliyetli olduğu düşünülmelidir. Bilgi Mühendisi, Uzman ve Programcı ile ortalama 3-5 yıl arasında değişen sistemlerin maliyetleri göz önüne alınmalıdır.
- Yaratıcılık: Normal durumları dışında alışlagelmemiş bir olay karşısında uzman insanın yaratıcılığı ile yeni bir çözüm üretebilirken uzman sistem bunu yapamamaktadır.
- Öğrenme: Uzman bir insanın öğrenme yeteneği çok kolay iken, dinamik yani öğrenebilen bir uzman sistem tasarlamak başlı başına bir problem teşkil eder. [44].

Sonuçta yapılan değerlendirme şu şekildedir:

Uzman sistemler kesin ve net algoritmalar yerine deneyime dayalı çıkarım yöntemleri kullanırlar bu nedenden ötürü uzman sistemlerin tasarımı karışık ve zaman alan bir işlemdir.

Ortak terminoloji olmayışı, çeşitli kullanılan kavramların kişiden kişiye değişerek farklı anlamlar kazanması ve sözlük anlamı ile gerçek anlamının birbirinden farklı oluşu yüzünden doğan karışıklıklardır.

Maliyet yüksekliği ve zaman fazlalığı bir sorun olarak tezahür eder, bunun nedeni ise uzman sistem tasarlamanın yüksek maliyeti ve buna ek olarak, bilgi mühendisi, uzman ve programcı ile 3-5 yıl arasında değişen sistemlerin maliyetleridir.

Diğer yapay zekâ teknolojileri herhangi bir insanın çözdüğü problemi çözmek için tasarlanırken, Uzman Sistemler bir uzman insanın çözdüğü problemi çözmek için tasarlanırlar.

Yukarıda belirtilen sebepler yüzünden bizim problemimizi çözmek için Uzman Sistemler seçilemez.

4.2.3 Yapay Sinir Ağları Seçilmeme Nedenleri

Yapay sinir ağları (YSA), insan beyninin bilgi işleme tekniğinden esinlenerek geliştirilmiş bir bilgi işlem teknolojisidir. YSA ile basit sinir sisteminin çalışma şekli simüle edilir (benzetilir). Simüle edilen sinir hücreleri nöronlar içerirler ve bu nöronlar çeşitli şekillerde birbirlerine bağlanarak ağı oluştururlar. Bu ağlar öğrenme, hafızaya alma ve veriler arasındaki ilişkiyi ortaya çıkarma kapasitesine sahiptirler. Diğer bir ifadeyle YSA'lar, normalde bir insanın düşünme ve gözlemlemeye yönelik doğal yeteneklerini gerektiren problemlere çözüm üretmektedir. Bir insanın, düşünme ve gözlemeleme yeteneklerini gerektiren problemlere yönelik çözümler üretebilmesinin temel sebebi ise insan beyninin ve dolayısıyla insanın sahip olduğu yaşayarak veya deneyerek öğrenme yeteneğidir [45].

Biyolojik sistemlerde öğrenme, nöronlar arasındaki sinaptik bağlantıların ayarlanması ile olur. Yani, insanlar doğumlarından itibaren bir yaşayarak öğrenme süreci içerisine girerler. Bu süreç içinde beyin sürekli bir gelişme göstermektedir. Yaşayıp tecrübe ettikçe sinaptik bağlantılar ayarlanır ve hatta yeni bağlantılar oluşur. Bu sayede öğrenme gerçekleşir. Bu durum YSA için de geçerlidir. Öğrenme, eğitime yoluyla örnekler kullanarak olur; başka bir deyişle, gerçekleşme girdi/çıkı verilerinin işlenmesiyle, yani eğitime algoritmasının bu verileri kullanarak bağlantı ağırlıklarını

(*weights of the synapses*) bir yakınsama sağlanana kadar, tekrar tekrar ayarlamasıyla olur. YSA'lar, ağırlıklandırılmış şekilde birbirlerine bağlanmış birçok işlem biriminden (nöronlar) oluşan matematiksel sistemlerdir. Bir işlem birimi, aslında sık sık transfer fonksiyonu olarak anılan bir denklemdir. Bu işlem birimi, diğer nöronlardan sinyalleri alır; bunları birleştirir, dönüştürür ve sayısal bir sonuç ortaya çıkartır. Genelde, işlem birimleri kabaca gerçek nöronlara karşılık gelirler ve bir ağ içinde birbirlerine bağlanırlar; bu yapı da sinir ağlarını oluşturmaktadır [45].

Sinirsel (*neural*) hesaplamanın merkezinde dağıtılmış, adaptif ve doğrusal olmayan işlem kavramları vardır. YSA'lar, geleneksel işlemcilerden farklı şekilde işlem yapmaktadırlar. Geleneksel işlemcilerde, tek bir merkezi işlem birimi her hareketi sırasıyla gerçekleştirir. YSA'lar ise her biri büyük bir problemin bir parçası ile ilgilenen, çok sayıda basit işlem birimlerinden oluşmaktadır. En basit şekilde, bir işlem birimi, bir girdiyi bir ağırlık kümesi ile ağırlıklandırır, doğrusal olmayan bir şekilde dönüşümünü sağlar ve bir çıktı değeri oluşturur. İlk bakışta, işlem birimlerinin çalışma şekli yanıltıcı şekilde basittir. Sinirsel hesaplamanın gücü, toplam işlem yükünü paylaşan işlem birimlerinin birbirleri arasındaki yoğun bağlantı yapısından gelmektedir. Bu sistemlerde geri yayılım metoduyla daha sağlıklı öğrenme sağlanmaktadır [45].

Çoğu YSA'da, benzer karakteristiğe sahip nöronlar tabakalar halinde yapılandırılırlar ve transfer fonksiyonları eş zamanlı olarak çalıştırılırlar. Hemen hemen tüm ağlar, veri alan nöronlara ve çıktı üreten nöronlara sahiptirler.

YSA'nın ana ögesi olan matematiksel fonksiyon, ağın mimarisi tarafından şekillendirilir. Daha açık bir şekilde ifade etmek gerekirse, fonksiyonun temel yapısını ağırlıkların büyüklüğü ve işlem elemanlarının işlem şekli belirler. YSA'ların davranışları, yani girdi veriyi çıktı veriyeye nasıl ilişkilendirdikleri, ilk olarak nöronların transfer fonksiyonlarından, nasıl birbirlerine bağlandıklarından ve bu bağlantıların ağırlıklarından etkilenir [45].

Yapay sinir ağlarının üstünlüklerinin yanı sıra bazı sakıncaları da vardır. Bu sakıncalar şu şekilde listelenebilir [45]:

- Sistem içerisinde ne olduğu bilinemez.
- Bazı ağlar hariç kararlılık analizleri yapılamaz.

- Farklı sistemlere uygulanması zor olabilir.

Yapay sinir ağlarının diğer yöntemlere nazaran güçlü yanlarını aşağıdaki gibi sıralamak mümkündür [45].

- Verilerin veya kuralların tamamının bilinmediği, doğru olmayan ve eksik verileri tolere etme yeteneği,
- Kendi kendini organize etme ve ağ bağlantılarını değiştirerek öğrenme yeteneği,
- Deneyimler tarafından eğitime ve çevredeki dinamiklere uyum sağlama yeteneği,
- Belirli örneklerden genelleme yapabilme yeteneği,
- Girdi değişkenleri arasında karmaşık ilişkiler bulma ve kurma yeteneği,
- Paralel olarak verileri işleyebilme yeteneği,
- Verilerin dağılımı ya da faktörler arası ilişki şekilleri ile ilgili varsayımlar gerektirmiyor olması [46].

Yapay Sinir Ağlarının dezavantajlarını şu şekilde sıralayabiliriz:

Yapay sinir ağlarının yararlı olduğu çoğu çalışmada görülse de, bu yöntemin bazı dezavantajları bulunmaktadır. Bu dezavantajlar literatür kaynaklarından derlendiğinde, aşağıdaki şekilde listelenebilir [47],[48],[49],[50].

Yapay sinir ağları üzerlerinde çalışacağı donanıma aşırı derecede bağımlıdır. Özellikle paralel işlem yapmalarından dolayı çok hızlı çalışan paralel işlemcilerle ihtiyaç duyarlar.

- YSA' lar her türlü problemin çözülmesi için hemen kullanılabilecek bir yöntem değildir. Probleme göre ağ yapısında, verilerin girilmesinde değişikliğe gitmek gerekebilir. Bu yapılmadan kullanıldığında ilgisiz ve kabul edilemez sonuçlar üretebilirler.
- Problemin özelliklerine uygun bir yapay sinir ağı modeli oluşturmak kolay değildir. Çünkü farklı ağ mimarileri, öğrenme yöntemleri ve parametreler bulunmaktadır. Probleme uygun ağ yapısının belirlenmesi genellikle deneme yanılma yöntemi ile yapılmaktadır. Dolayısı ile doğru ağ modeli kullanılmazsa performansı düşük sonuçlar elde edilebilir. Bu nedenle, uygun yapay sinir ağı yapısının tanımlanması için çok sayıda test yapılması gerekmektedir.

- İstatistiksel çözümlerle fazla yoruma ve açıklamaya ihtiyaç bırakmayan sonuçlar elde edilebilmesine rağmen YSA' nın ürettiği sonuçlardan ağırlıkların yorumlanması o kadar kolay değildir. Dolayısı ile YSA ile ulaşılan sonuçlarda model kapalı bir kutu gibi düşünülebilir. Bu kapsamda, eğitilmiş ağlardan bilgi çıkarabilmek için geliştirilmiş algoritmalar bulunmaktadır. Söz konusu algoritmalar aracılığıyla çıkarılan bilgiler, "if (koşul), then (sonuç)" formunda sembolik kurallar olarak açıklanabilmektedir. Fakat çoğu problem için eğitilmiş ağlardan elde edilen kuralların sayısı çok fazla olduğundan, kullanıcı için pratik bulunmamaktadır.
- Ağın davranışlarını açıklamak kolay değildir. Bu durum ağa olan güveni azaltmaktadır. Özellikle hayati önem taşıyan konularda ürettiği sonucun nedeninin açıklanamaması kullanım alanını sınırlamaktadır.
- Yapay sinir ağları spesifik eğitim örneklerine aşırı derecede duyarlı olduklarından, sonuçları güvenilmez olabilmektedir. Ağın eğitimi için kullanılan süreye bağlı olarak ya da diğer bir ifadeyle kullanılan iterasyon sayısına göre, aynı eğitim seti üzerindeki sonuçlar bile birbirinden çok farklı olabilmektedir. Eğitimin ne zaman durdurulması gerektiği konusunda çeşitli yöntemler bulunmakla birlikte, söz konusu yöntemlerin her zaman en iyi performansı sağlayacağı garanti değildir. Ağ eğitmek önemli bir problemidir. YSA' lar sadece sayısal bilgiler ile çalışabilirler. Dolayısı ile tüm veriler ağa girilmeden önce sayısallaştırılmalı ve ölçeklendirilmelidirler. Sayısallaştırma işlemi ise kullanıcının tecrübesiyle ve başarısıyla doğru orantılıdır.
- Ağın ne kadar eğitileceğinin bilinmemesi de ayrı bir dezavantajdır. Genellikle hatanın kabul edilebilir değerler içerisinde kalmasıyla eğitim tamamlanmış olarak kabul edilebilir. Fakat bunun doğru bir çözüm olacağı garanti edilemez. Bazı zamanlar ağın takılması ya da ezberlemesi de bir dezavantaj yaratmaktadır.
- Bazı durumlarda, yapay sinir ağının karmaşık ilişkileri öğrenebilmesi için çok sayıda veriye gereksinim duyulmaktadır. Bu da, örnek sayısının az olduğu alanlarda yapay sinir ağı kullanımını zorlaştırmaktadır. Ayrıca, verilerin çok olması ve öğrenilecek ilişkinin fazla karmaşık olması durumunda yapay sinir ağlarının eğitilmesi uzun zaman alabilmektedir.

Sonuçta yapılan değerlendirme şu şekildedir:

Farklı sistemlere, örneğin araç rotalama problemi içeren sistemlere, uygulanması zor olabilir.

Yapay sinir ağları, olayların örneklerine bakmakta, onlardan ilgili olay hakkında genellemeler yapmakta, bilgiler toplamakta ve daha sonra hiç görmediği örnekler ile karşılaşınca öğrendiği bilgileri kullanarak o örnekler hakkında karar verebilmektedir. Gereğinden fazla olması sistemin öğrenmesini zorlaştırmakta, Gereğinden az olması YSA sonuçlarındaki hata oranında artmasına sebep olmakta.

Burada karşılaşılan güçlükler ise, uygun bir YSA yapısının, fonksiyon tipinin ve öğrenme algoritmasının belirlenmesi, simülatörün birbiri ardınca eğitime tabi tutulmasında eğitim performansının düşmesi, aynı YSA yapısının eğitime tabi tutulmasında bile başlangıç ağırlık değerlerinin rasgele atanmasından kaynaklanan benzer sonuçların üretilmemesi olarak sıralanabilir.

Bu algoritma hesaplanması kolay olmasına ve sinir ağları topluluğunda yaygın biçimde kullanılmasına rağmen; uzun sürede çözüme ulaşması ve hata yüzeyinde kimi zaman yerel minimum değerine ulaşması (yakalanması) nedeniyle bazı dezavantajlara sahiptir. Bununla birlikte, bir gradyen arama tekniği olarak, büyük veri grupları üzerinde çalışması iyi sonuç verirken, sınırlı büyüklükteki veri üzerinde "en iyi çözüm" olduğu söylenemez. Benzer şekilde gereğinden daha basit bir yapıya sahip olan bir sinir ağı, deneme seti içinde yer almayan veriler için bile iyi yaklaşım değerleri üretemez; gereğinden daha karmaşık (kompleks) bir sinir ağı ise deneme setine aşırı uyar (overfit); yani veri seti içindeki modele iyi uyarken bilinmeyen örüntüler için kötü yaklaşımlar ve sonuçlar verir [51],[52].

4.2.4 Karınca Kolonisi Algoritması Seçilmeme Nedenleri

Karınca kolonisi algoritması "Sürü zekâsı" kavramına dayanan bir algoritmadır. Sürü zekâsı, özerk yapıdaki basit bireyler grubunun kolektif bir zekâ geliştirmesidir [53]. Sürü zekâsı, sürüdeki bireylerin kendi başlarına yaptıkları eylemler sonucu ortaya çıkan etkiler aracılığıyla iletişim kurmaları, kendi aralarında hareketlerini düzenleme durumu ve kendiliğinden düzen (self organization) denen iki ana mekanizmadan oluşur. Bu iletişim, bireylerin yaptıkları işlerin ortamda meydana getirdiği değişiklikler sonucu

sağlanır. Bu sayede, önceden yapılmış ve merkezi bir plan olmadan kendiliğinden düzen sayesinde sonuç üretebilmeleri ve merkezi yönetim birimi olmadan esnek ve sağlam bir şekilde yapılanmaları sağlanır. Sürülerin bu özellikleri ve gerçekleştirdikleri eylemler, kolektif olarak çözdükleri problemler (Yiyecek bulma, taşımada yardımlaşma, kolektif kümelenme gibi) klasik yapay zekâ yöntemlerine yeni bir soluk getirmiştir. Klasik yapay zekâ yöntemlerinde bulunan insan zekâsını modellemeye odaklı, karmaşık, merkezî, planlı yaklaşımların aksine, sürü zekâsı basit yapılı, özerk, önceden planlama yapmayan dağınık ajanların karmaşık problemlerin çözümünde başarılı olduklarını göstermiştir [54].

Doğanın taklidiyle oluşturulan bir meta-sezgisel algoritma laboratuvar ortamında incelenen karıncaların, yiyeceğe ulaşmak için zamanla en kısa mesafeyi bulduğu ve bu bilgiyi diğer karıncalara feromon vasıtasıyla aktardığı tespit edilmiştir. Sonunda bu davranışın yapay karıncalara da aktarılabilceği öngörülmüştür. Yapay karıncaların gerçek karıncalara göre gidecekleri alternatif yolların uzaklığını önceden bilmeleri gibi bir avantajları vardır. Karınca bir yere ulaşmak için feromon miktarı ile mesafe arasında (belirlenen bir olasılığa) göre bir seçim yapar. Mesela bir noktada olduğunuzu ve önünüzde üç yol olduğunu varsayalım. Bu yollarda bırakılmış feromon miktarı (gerçek karınca bu yolu direkt seçer) ve kafanızda önünüzdeki yolun uzunluğu (daha doğrusu 1/uzunluk) bilgisi vardır. Bu iki seçenek arasında belirlenen katsayılara göre seçme olasılığı hesaplanır karınca buna göre yolu seçer. Zamanla tercih edilen yollarda feromon birikir, diğerlerinde feromon kaybolur gider. Çözüm kalitesi yönünden çok başarılı olmasına rağmen karınca sayısı, feromon oranı, feromon uçuş oranı, sabit sayılar vs. gibi çok fazla parametre vardır. Bu yüzden çözüm hızı biraz düşüktür (tabu aramaya göre). Ayrıca bu parametrelerin belirlenmesi için ayrı ayrı çalışma yapmak gerekir [54].

Algoritmanın temel dezavantajı, yön tayin ederken sadece koku miktarını kullanmasıdır. Bu sebeple bazı optimizasyon problemlerinde erken yakınsama ve bölgesel minimuma takılma problemiyle karşı karşıya kalınmaktadır. Bu dezavantajı gidermek amacıyla, alt yolların koku miktarının rasgele değişimine dayanan bir strateji geliştirilmiştir [55].

Uygulamada bulunan problem açısından düşünürsek tek depolu bir sistem için tasarım yapılacak olması ve KKA'nın çok depolu algoritmalar için daha iyi sonuç veriyor olması gibi sınırlamalar bu algoritmanın seçilmemesine sebep olmuştur.

Literatürde sınırlı sayıda çalışma yapılmış olmasından dolayı dezavantaj ve avantajlarına kabaca değinilmiştir.

4.2.5 Tabu Arama Algoritması Seçilmeme Nedenleri

Tabu Arama'da temel yaklaşım, son çözüme götüren adımın, dairesel hareketler yaratmasını engellemek için bir sonraki döngüde tekrarın yasaklanması veya cezalandırılmasıdır. Tabu yöntemi kısmen, benzer olaylarda rastlantılı işlemler gerçekleştirirken insan davranışının tutarsızlığa eğilimli olmasından etkilenmiştir. Söz konusu işlem yeni rotaların rastlantılı olarak seçilmemesi dışında bu şekilde işler. Başka bir ifadeyle tabu arama zaten incelenmiş bir yol olmadığı sürece her çözümü araştırabilen bir süreçtir. Böylece yeni çözüm uzayının incelenmesi suretiyle yerel minimumdan kaçınılarak istenilen çözüme ulaşabilmektedir. Algoritma yerel minimuma doğru hareket ederek başlar. Daha önce yapılmış hareketlere tekrar dönüş yapmayı engellemek için yöntem bir veya daha fazla tabu listesi tutar. Listenin orijinal amacı, önceden yapılmış bir hareketin tekrarından çok tersine dönmesini önlemektir. Tabu listesi kronolojik bir yapıya sahiptir ve tabu arama belleğini biçimlendirir. Belleğin rolü algoritma ilerledikçe değişebilir. Başlangıçta hedef, çözüm uzayında kaba araştırma yapmaktır. Aday konumlar belirlendikçe arama yerel optimum çözümü üretmeye daha fazla odaklanır [56].

Tabu araştırma algoritmasında, tek bir çözümün komşuları (çözüm adayları) çeşitli fonksiyonlarla belirlenir. Eğer bu komşulardan daha önce denenmişler varsa (tabu listesinde iseler), bunların yerine başka komşular üretilerek araştırmanın hızlandırılması sağlanır ve yerel optimum noktalarda ortaya çıkan kısır döngüye girilmesi engellenir. Bu, tabu araştırma algoritmasının en önemli avantajıdır [56].

Sıklığa dayalı hafıza tabu aramanın önemli bir özelliğidir: Bu, algoritma tarafından yapılan en sık hareketler esnasında değişen çözüm özelliklerini aklında tutan bir çeşit kısa dönem hafızadır. Tabu-aktif olarak, son zamanlarda ziyaret edilmiş çözümlerin seçilmiş özellikleri etiketlenir. Sıklık ölçümü genellikle bir elemanın değişmeye uğradığı

iterasyon sayısı ya da bir elemanın çözümün bir parçası olarak yer aldığı iterasyon sayısı olarak belirlenir. Bir elemanın değişime uğradığı veya bir elemanın çözümün bir parçası olduğu iterasyon sayısını tutan sıklık ölçümleri birbirleriyle alakalı bilgi içermelerine karşı farklı manalar taşımaktadır. Bir elemanın çok sayıda çözümde var olması, kaliteli çözümlere ulaşılmışsa o elemanın değerli bir eleman olduğunu gösterir. Bir elemanın değişime uğradığı iterasyon sayısı çok fazlaysa ve bir süredir araştırmada ilerleme gözlenememişse bu elemanın, etkisi düşük bir eleman olduğu anlaşılır. Böyle bir elemanın hareket etmesi engellenerek, araştırmanın seyrinin değişmesi sağlanabilir. Yüksek frekans değeri olan elemanlar bir tabu aktivasyon kuralını tetikleyebilirler ve tabu listesine alınarak tekrarlanmaları engellenebilir. Literatürde belirtilen pratik uygulamaların çoğu direk olarak bu temel TAA'sını uygulamaktadır. Bu, en temel TAA algoritması çeşididir, yasaklanan özellikler listesine ve amaç kriterine dayanır. TAA listesinin temel amacı döngüyü engellemek ve komşuluğu azaltmaktır (tabu olarak etiketlenen durumları çıkararak). TAA listesi kullanmanın asıl dezavantajı, yasaklanan özelliklerin incelenmemiş etkili komşulukların bir parçası olabileceğidir. Bu problemin üstesinden gelmek için bir tabu durumu ile ilgili maliyet o ana kadarki en düşük maliyetten daha düşükse tabu durumu ortadan kalkar ve değişime izin verilir [57].

Tabu aramanın çok genel bir arama metodu olması ve bu metodun işleyişi ile ilgili kesin prensiplerin olmayışı algoritmayı geliştiren kişiye büyük esneklik sağlamaktadır.

Ancak arama metodunun başarılı olması algoritmasının iyi tasarlanmış olmasına bağlıdır. Bu nedenle bu metodun esnek olması avantajı iyi kullanılmalı ve çözülmesi istenen probleme uygun bir tasarım yapılmalıdır.

Diğer meta-sezgisellere göre nispeten hem programlanması, hem de çözüm hızı açısından daha basittir. Fakat çözüm kalitesi başlangıç çözümünün etkisi çok fazla olduğu için pek iyi değildir.

Algoritmanın dezavantajı problem boyutu ve amaç sayısı arttıkça çözüm zamanının da uzayacak olmasıdır. Özellikle amaç sayısı artırıldıkça çözüm zamanı da artacaktır [58].

Yukarıda bahsedilen dezavantaj ve avantajlar göz önünde bulundurularak ve literatürde GA'nın yaygın olarak kullanılmasının da etkisiyle TAA çözüm yöntemi olarak

seçilmemiştir. Bunun yanı sıra GA'ya çok yakın sonuçlar veren bir algoritma olarak görülmesi, TAA'nın da seçilebileceğine işaret eder.

4.3 Uygulamada Kullanılacak Yöntemler

Teze konu olan problemin çözümünde geliştirilen hibrit modelde kullanılacak yöntemler olarak Tasarruf ve Genetik Algoritmaları seçilmiştir.

4.3.1 Clarke and Wright Tasarruf Algoritması

Çalışmada Kullanılacak yöntemlerden biri Clarke ve Wright Tasarruf Algoritmasıdır. Bir sezgisel algoritmanın uygulanabilirliğinin fazla olması için gereken karakteristiklerinden biri de üretilen çözüm kalitesinin başlangıç koşullarına çok fazla bağlı olmamasıdır. Yazarlara göre Clarke ve Wright Algoritmasının çok popüler olmasının sebeplerinden en önemlisi anlaşılabilirliğinin kolay olması, kısa sürede kodlanabilmesi ve bir problem için aynı sonucu üretebilmesidir. Yöntemin basit olması uygulanabilirliğini artırırken, bilgisayar kaynaklarının daha verimli bir şekilde kullanımını da sağlayacağından ARP'ye hızlı bir şekilde çözüm üretmek de mümkün olmaktadır [10].

Orijinal Clarke and Wright tasarruf Algoritmasının dezavantajı, iyi rotalar üretmesine rağmen bazı çevresel yollarda dâhil olmak üzere rotaların sonlarına doğru bunu koruyamamasıdır. Bunu düzeltebilmek amacıyla Gaskell (1967) ve Yellow (1970) tarafından tasarruf hesaplama fonksiyonu $s_{ij} = c_{i0} + c_{0j} - \lambda c_{ij}$ olarak değiştirilmiş ve bu sayede köşeler arasındaki mesafe λ terimi ile ağırlıklandırılarak daha etkin bir tasarruf değeri elde edilmiştir [10].

ARP için en çok bilinen yöntemlerden biri Clarke ve Wright'ın 1964'de geliştirdikleri Tasarruf Algoritmasıdır. Algoritma araç sayısının değişken olarak ele alındığı problemlerde uygulanmaktadır. Bu metot ile rotaların hazırlanmasının yanı sıra, kaç adet aracın da gerekeceğini bulunmaktadır. Tasarruf algoritmasının Paralel ve Sıralı olmak üzere iki versiyonu bulunmaktadır. Bu yöntemin anlatımı aşağıda belirtilmiştir [59]:

Önce her müşteri çifti için tasarruflar $s_{ij} = d_{i0} + d_{0j} - d_{ij}$ formülü ile hesaplanmaktadır. S_{ij} değerleri büyükten küçüğe sıralanır. Sonra Paralel Tasarruf algoritmasında s_{ij} değerlerine göre büyükten küçüğe sıralanan müşteri ikilileri (i-j) şu kurala göre

birleştirilir: Eğer sıralamada xy ve yz müşteri çiftleri ardı ardına geliyorsa araç kapasite kısıtı göz önüne alınarak 0-x-y-z-0 güzergâhında bir rota oluşturulur. Aksi takdirde xy ve tz formatında müşteri çifti art arda geliyorsa, bu müşteri çiftleri için ayrı ayrı rota oluşturulur. Sıralamada ele alınan müşteri çiftleri daha önce oluşturulmuş rotalar ile mukayese edilir, eğer uygunsa rotaya dâhil edilir. Burada birden fazla rota oluşturulup bu rotaları paralel isleyerek müşteri çiftlerini bu rotalardan birine atamak mümkündür. Tasarruf algoritmasının Sıralı versiyonunda ise ele alınan xy ve tz formatındaki müşteri çiftleri kapasite kısıtı dikkate alınıp birleştirilerek 0-x-y-t-z-0 güzergâhında rota oluşturur. Aynı rota için, sıralamada ele alınan müşteri çiftleri mukayese edilir, eğer kapasite kısıtına uygunsa rotaya dâhil edilir. Bu yöntemde bir rota tamamıyla oluşuncaya kadar (kapasite kısıtı dikkate alınarak) müşteri çiftleri taranmaktadır. Her iki yöntemde de müşteri çiftleri rotalara toplam maliyet minimum tutulacak şekilde yerleştirilir.

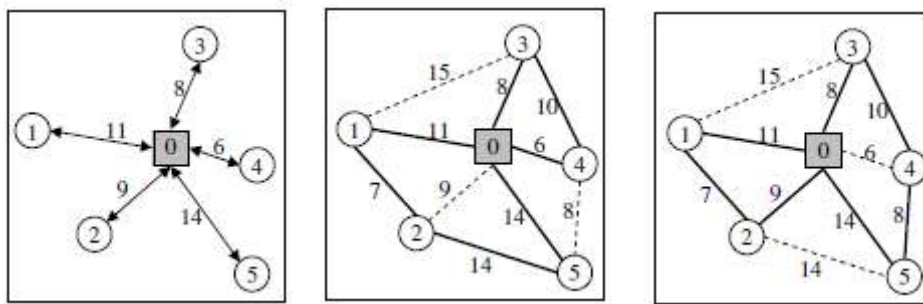
Tasarruf Algoritması'nda Paralel yöntem, Sıralı yönteme göre daha iyi sonuçlar vermektedir. Bunun sebebi Sıralı Tasarruf algoritmasında sadece bir rota ele alınıp bu rotayı oluşturmak için sij tasarruf sıralamasının dikkate alınması, Paralel versiyonda ise sij sıralamasına bakılarak müşteri çiftinin birden fazla rota ile uyumluluğu kontrol edilmesi ve bu sayede daha fazla alternatifin gözden geçirilebilmesidir [60]. Çizelge 4.2'de mesafe ve sij tasarruf değerleri verilen ve 5 talep noktasından oluşan bir ARP'nin Tasarruf algoritması ile çözüm süreci yer almaktadır. Problemde "0" merkez depoyu göstermektedir.

Çizelge 4. 2 Noktalar arası mesafe ve sij değerleri

v_i	v_j	d_{ij}	s_{ij}	v_i	v_j	d_{ij}	s_{ij}	v_i	v_j	d_{ij}	s_{ij}
0	1	11	-	1	2	7	13	2	4	13	2
0	2	9	-	1	3	15	4	2	5	14	9
0	3	8	-	1	4	17	0	3	4	10	4
0	4	6	-	1	5	19	6	3	5	16	8
0	5	14	-	2	3	16	1	4	5	8	12

Problemde araç kapasitesi 10 birim olarak, her müşteri noktası için de talep miktarı 3 birim olarak verilmiştir. Çizelge 4.2'ye göre sij değerleri büyükten küçüğe sıralanırsa, su müşteri çiftleri elde edilir: 1-2, 4-5, 2-5, 3-5, 1-5, 1-3, 3-4, 2-4, 2-3 ve 1-4.

Sıralı Tasarruf Algoritmasına göre ilk olarak 0-1-2-0 başlangıç rotası oluşturulur ve bu rotaya uygun olan diğer müşteri çiftlerine sıra ile bakılır. 0-1-2-5-4-0 rotası kapasite kısıtı olan 10 birimi aştığı için 4-5 nolu müşteri çifti atlanır ve sıradaki 2-5 müşteri çifti rotaya eklenerek 0-1-2-5-0 rotası kontrol edilir. Bu rota kapasite kısıtına uymaktadır, buna göre 5 müşterisi rotaya atanmıştır. Bu rotaya başka müşteri atanamayacağı için diğer rotaya geçilir. Bu rota 4-5 nolu müşteri çifti ile kurulamaz çünkü 5 müşterisi daha önceki rotada yer almaktadır. Diğer rota mecburen 0-3-4-0 güzergâhında olacaktır. Böylece Sıralı Tasarruf yöntemi ile 0-1-2-5-0 ve 0-3-4-0 rotası oluşturulmuş olup toplam gidilen mesafe yani amaç fonksiyonu değeri $(11+7+14+14) + (8+10+6) = 70$ olarak bulunmuştur. Paralel Tasarruf algoritmasına göre problem çözülürse ilk olarak 1-2 müşteri çifti ele alınarak 0-1-2-0 rotası oluşturulur. Sıradaki müşteri çifti 4-5 için ise 0-4-5-0 güzergâhında yeni rota oluşturulmak durumundadır, çünkü 0-1-2-0 rotasında araya atanabilecek müşteri içermemektedir. 2-5 nolu müşteri çifti geçilir, çünkü hem 2 ve hem de 5 nolu müşteri de daha önce rotalarda kullanılmıştır. Sıradaki 3-5 nolu müşteri çifti ise 0-4-5-0 rotasına atanarak 0-3-4-5-0 rotası elde edilir. Bu rota kapasite kısıtına uymaktadır. Tüm müşteriler rotalara atandığı için algoritma sonlandırılır. Bu teknik ile çözüm olarak iki araç için 0-1-2-0 ve 0-3-4-5-0 rotaları bulunmuş, amaç fonksiyonu değeri de $(11+7+9) + (8+10+8+14) = 67$ olarak hesaplanmıştır. Bu örneğe göre Paralel Tasarruf Algoritması, Sıralı Tasarruf algoritmasından daha az maliyetli rotalar hazırlamıştır. Şekil 4. 1’de bu probleme ait çözümler yer almaktadır.



Başlangıç Rotalar Sıralı Tasarruf Çözümü Paralel Tasarruf Çözümü

Şekil 4. 1 Tasarruf Algoritmasına ait örnek

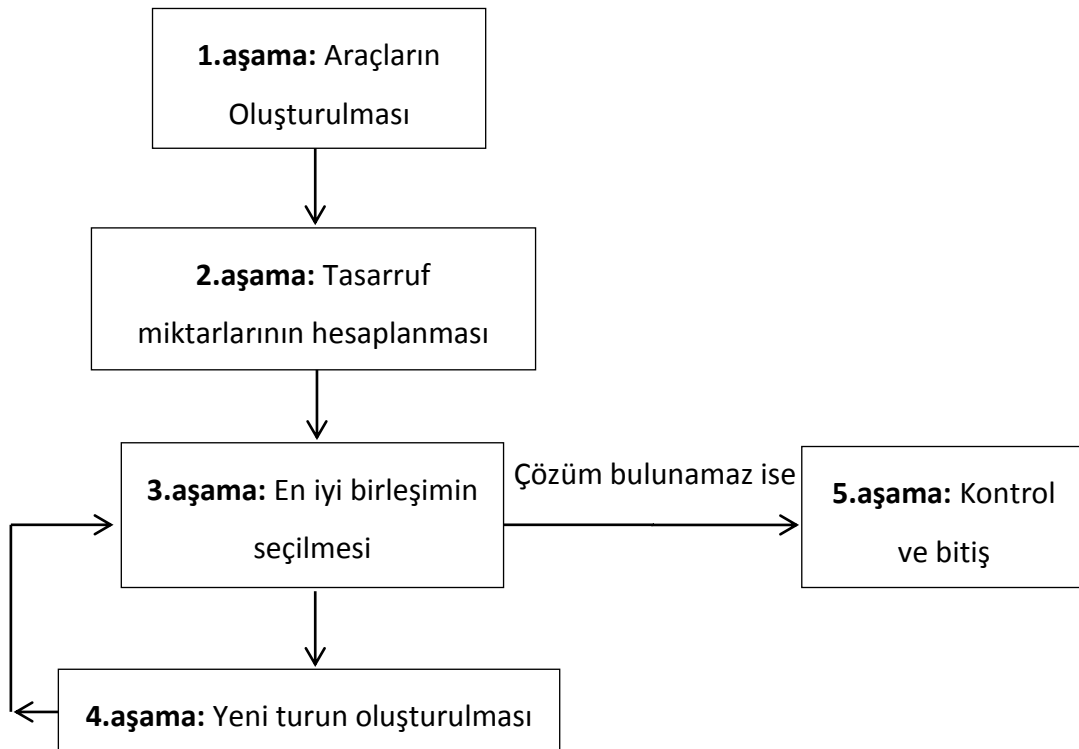
Tasarruf algoritmasının mantığında, ilk olarak her müşteri için bir rota hazırlanılıp bu rotaların bir sıraya göre uygun bir şekilde (kısıtlar dikkate alınarak) birleştirilmesi yatmaktadır. Bu sıra ise her müşteri çifti için tasarruf miktarı hesaplanarak

bulunmaktadır. Clarke ve Wright'ın Tasarruf Algoritmaları asimetrik KKARP'ye de uygulanabilmektedir.

Tasarruf algoritması ile çok hızlı bir şekilde çözümler alınabilmesine rağmen, çözüm kalitesi çoğu araştırmacı tarafından vasat kabul edilmektedir. Ayrıca bu yöntemin esnekliği düşük olup yeni kısıtların probleme eklenmesi halinde çözümler oldukça kötüleşmektedir. Bunun sebebi daha önce oluşturulmuş rotalar üzerinde değişikliklerin yapılamamasıdır [61].

4.3.1.1 Tasarruf Yönteminin Aşamaları

Tasarruf yöntemi beş aşamadan oluşmaktadır. Bu aşamalar Şekil 4.2'de gösterilmektedir[61].



Şekil 4. 2 Tasarruf Yönteminin Aşamaları[61].

1. Aşama: Araçların oluşturulması. Her bir müşteriye bir araç atanır ve n araçlı bir rotalama yapılmış olur.

2. Aşama: Tasarruf miktarların hesaplanması. Aşağıdaki durumları sağlayacak şekilde tasarruflar hesaplanır.

a. Tur büyüklüğü araç kapasitesini aşmaması gerekir.

b. Atama yapabilmek için araç kısıtlarına uyan en az bir aracın bulunması gerekir.

c. Tur içindeki sabit pozisyonlar bozulamaz.

d. Toplam tur zamanı araç için bulunan maksimum zamanı aşamaz.

3. Aşama: *En iyi birleşimin seçilmesi.* Bütün kısıtları sağlayan olurlu en büyük tasarrufa ait olan birleşim seçilir. Eğer bütün kısıtları sağlayan pozitif bir tasarruf miktarı bulunamaz ise 5. aşamaya geçilir.

4. Aşama: *Yeni turun oluşturulması.* En iyi birleşimdeki müşteriler aynı tur içerisine alınarak yeni bir tur oluşturulur. 3. aşamaya geri dönülür.

5. Aşama: *Kontrol ve bitiş.* Bütün müşterilerin araçlara atandığı kontrol edilir ve tasarruf yöntemi sonlandırılır. Müşteri taleplerinin araç kapasiteleri toplamından fazla olduğu durumda, tasarruf yönteminin sonunda hizmet alamamış müşteriler kalacaktır.

4.3.2 Genetik Algoritma

Kurulacak modelin 2. Aşaması olarak Genetik Algoritma ele alınmıştır.

4.3.2.1 Genetik Algoritmaya Giriş

Genetik algoritmaların temeli doğal seleksiyon ve genetik bilimine dayanmaktadır. Bu algoritmalarda, belirlenen bir problemin potansiyel çözümleri, ikili (binary) ya da ikili olmayan sistemlere dayalı veri yapısında basit diziler olarak şifrelenmekte ve kritik bilgileri saklamak için bu dizilere bir takım işlemler uygulanmaktadır. Genetik algoritma yöntemi, evrim teorisi esaslarına göre çalışarak verilen bir sorun için en iyi çözüm veya çözümleri arayarak bulmaya yaramaktadır. Bu esaslar ortama en fazla uyum sağlayan canlıların hayata devam etmesi ve uyum sağlayamayanların da elenmesi olarak algılanmalıdır. Genetik algoritmalar bu iki kuralı bir arada kullanarak en iyiyi aramayı hedef edinen bir en iyileme yöntemidir [62],[63].

Genetik algoritma John Holland ve öğrencileri tarafından geliştirilmiştir. Bu algoritmanın gelişmesinde öne çıkan iki adet hedef bulunmaktadır. Bunlardan ilki doğal sistemlerin uyarlanabilirliklerini teorik ve kesin olarak ortaya koymak, ikincisi ise doğal sistemlerin mekanizmasının önemli özelliklerini kullanan bir yapay sistem yazılımı

geliştirmektedir. Bu iki yaklaşım doğal ve yapay sistemlerin ortaya çıkmasına öncülük yapmıştır [63].

Genetik algoritmaların karmaşık uzaylarda iyi sonuçlar elde edebildiği teorik ve görsel olarak kanıtlanmıştır [63]. Bu konuda ilk eseri John Holland vermiştir. Başlangıçta pratik bir yararı olmadığı düşünülen genetik algoritmalara olan ilgi, Holland'ın öğrencisi olan Goldberg'in yaptığı doktora teziyle National Science Foundation tarafından verilen genç araştırmacı ödülünü alması ve dört yıl sonrada klasik eserini [63] yayınlamasıyla çoğalmıştır [62].

Genetik algoritmanın hızlı bir şekilde yaygınlaşmasının nedeni hesaplama olarak basit ve en iyiyi arama mekanizması konusunda güçlü bir yapıya sahip olmasıdır. Arama uzayı sadece amaç fonksiyonunu etkileyen kısıtlarla sınırlıdır. Bunların yanı sıra ticaret, bilim ve mühendislikte yaygın kullanım alanları bulunmaktadır (Goldberg, 1989).

Genetik algoritma, evrim teorisinden geliştiğine göre canlı kalıtımıyla ilgili yanları oldukça fazladır. Şifrelenmiş sayı dizilerini DNA'lara benzetirsek, nasıl ki canlılar arasında iki bireyden bir araya gelen farklı DNA özellikleri yeni bir birey oluşturuyorsa genetik algoritmada da bu böyledir. Ancak genetik algoritmada tek bir birey yerine, bu aileden oluşan bir çift yeni birey bulunmaktadır [63].

Yeni bireyler, aile fertlerinin özelliklerinin çaprazlanması ile oluşurlar. Daha sonra kendini geliştiren bu bireyler, bazı özellikleri değişerek, bir sonraki nesil için yeni aile bireylerine dönüşürler. Tıpkı gerçek hayattaki gibi sağlıklı bireylerin yaşama olasılıkları daha yüksek olmaktadır. Genetik algoritmalarda bireylerin yaşayabilirliğini sınavabilmek için bir uygunluk (fitness) değeri kullanılmaktadır. Sağlıklı bireyler uzun süreler yaşayabilirken, sağlıksız bireyler kendi neslinden bireylerle birleşerek yine sağlıklı bireyler oluşturabilmektedirler.

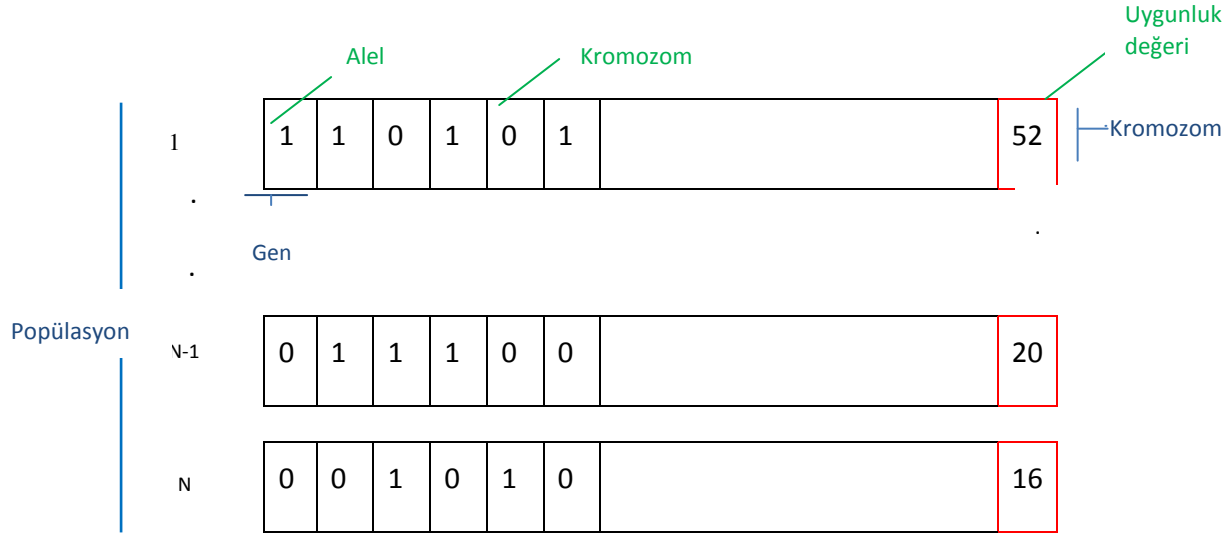
Bireyler arasında geçen bu alışveriş sonucunda sürekli daha güçlü ve sağlıklı bireylere doğru bir ilerleme olmaktadır. Kötü bireyler yerlerini daha iyi bireylere bırakmaktadırlar ve yok olmaktadır. Bu da en iyileme açısından sonuca daha hızlı bir şekilde yaklaşmak demektir [62].

4.3.2.2 Genetik Algoritma Terminolojisi ve Akış Algoritması

Genetik algoritmalar, genetik bilimi ile benzerlik gösterdiğinden birbirine yakın terminolojileri bulunmaktadır. Bir bireyin DNA'sını tanımlamak için kullanılan terimlerden; kromozom, gen, alel (allele), konum (locus), fenotip (phenotype) ve genotip (genotype) terimleri aynı zamanda genetik algoritmadaki sayı dizilerini ve özelliklerini tanımlamak için de kullanılmaktadır [63],[64].

Genetik algortmada bireylerin oluşturduğu bir topluluk bulunmaktadır. Bu topluluğa popülasyon adı verilmektedir. Popülasyon içerisinde, her bir bireyin kalıtsal özelliklerinin taşındığı kromozomlar bulunmaktadır. Kromozomlar yapay sistemlerde dizi (string) olarak gösterilmektedir. Bu kromozomlar aynı zamanda bireyin kalıtsal özelliklerinin bilgilerini şifrelenmiş bir şekilde taşıdığı için bireyin genotipi olarak adlandırılmaktadır. Doğal sistemlerde genotip tek bir birey için birden fazla kromozomun birleşmesinden oluşabilir. Ancak yapay sistemlerde bir birey için tek bir kromozom bulunmaktadır ve bu sistemlerin genotipi kromozomun içinde taşıdığı özelliklerdir. Bunlar da yapı (structure) olarak gösterilmektedir. Doğal yaşamda genotipi oluşturan şifrelenmiş bilgilerin göz rengi, ten rengi gibi özelliklere dönüşmesiyle oluşan genetik yapıya ise fenotip adı verilmektedir. Yapay sistemlerde fenotip, parametre kümeleri olarak gösterilmektedir. Dolayısıyla genotip yapısı deşifrelendiğinde fenotipi oluşturacaktır [63],[64].

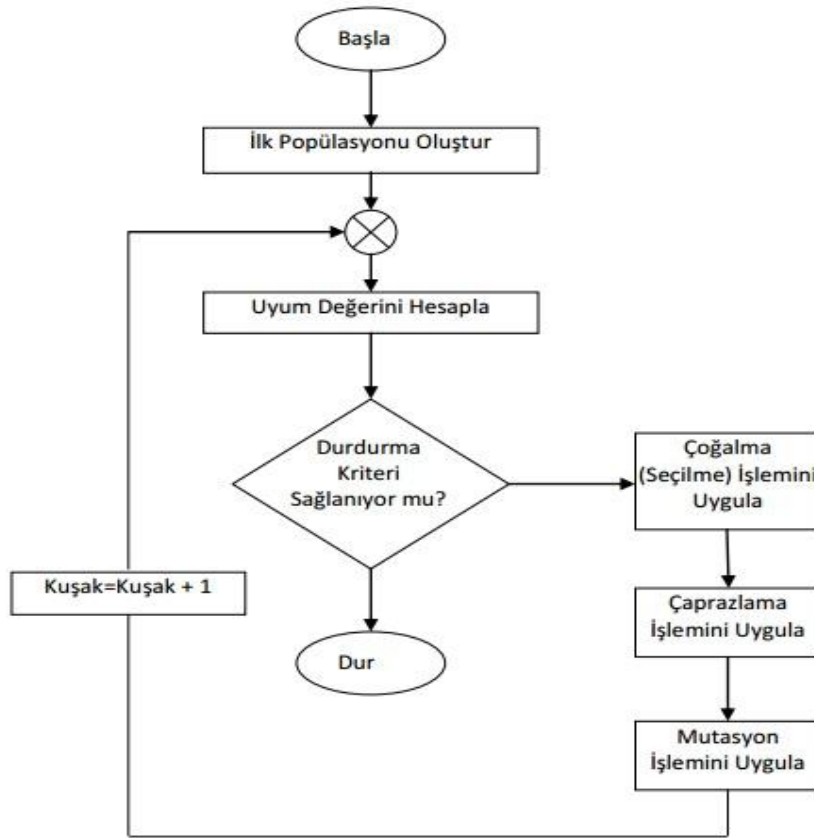
Kromozomlar, genlerin birleşiminden meydana gelmiştir. Bu genler yapay sistemlerde kromozom üzerinde bilgilerin tutulduğu karakterler veya özellikler olarak gösterilmektedir. Genlerin taşıdıkları değere alel ve her alelin bilgisinin tutulduğu yere konum adı verilmektedir. Yapay sistemlerde alel ilgili özelliğin değeri ve konum ise bu özelliğin dizi üzerindeki yerine karşılık gelmektedir. Popülasyon yapısına örnek Şekil 4.3'te gösterilmiştir.



Şekil 4. 3 Genetik algorithmada popülasyon yapısı

Basit bir GA (Simple Genetic Algorithm), en iyileme için son derece uygun ve güçlü bir yöntemdir. Genetik algoritma, en uygun (optimal) çözümü bulmak için birden çok noktadan aramaya başladığından bu arama noktalarının başlangıç değerlerini oluşturmak önemlidir. Başlangıç popülasyonu genellikle rastlantısal olarak oluşturulur [18], [65].

Genetik algoritma sürecinin başlaması için öncelikle, başlangıç popülasyonundaki bireylerin her birinin uygunluk değerleri hesaplanması gerekmektedir. Daha sonra seçim yöntemleri kullanılarak bu bireyler içinde yeni nesle aktarılacak olanlar seçilecektir. Seçilen bireyler arasında evrimsel işlemler uygulanmaktadır. Önce çaprazlamaya (crossing-over) maruz kalan bireyler daha sonra mutasyon (mutation) geçirmektedirler. Oluşan yeni neslin uygunluk fonksiyonları yeniden hesaplanmakta ve bir sonraki nesle aktarılacak bireyler uygunluk değerlerine göre tekrar seçim işlemine tabi tutulmaktadır [18], [65]. Bu işlem süreci problemin niteliğine ve beklentilerine göre en uygun sonuç elde edilinceye kadar sürer. Genetik algoritmanın akış diyagramı Şekil 4. 4'te gösterilmiştir.



Şekil 4. 4 Genetik Algoritma Akış Diyagramı

Algoritma, en uygun sonucun bulunması ile süreci durduracaktır. Ancak beklenen kesin bir sonuç yoksa ve daha iyi bir sonuca yaklaşmak isteniyorsa, o zaman süreç aynı sonucun belli bir süre tekrarından sonra da durdurulabilmektedir [18].

4.3.2.3 Basit Genetik Algoritma

Genetik algoritmanın akış şemasında görüldüğü üzere yeni bireyleri oluşturmak için bir dizi işlem gerekmektedir. Bu işlemler kopyalama, çaprazlama ve mutasyon olarak adlandırılmakta ve sürekli iyileşme için büyük önem taşımaktadırlar. Belirlenen problem ister en küçükleme ister en büyükleme olsun yapılan işlemlerin tekrarı yeni ve daha sağlıklı sonuçlar veren nesiller oluşmasına yardımcı olacaktır.

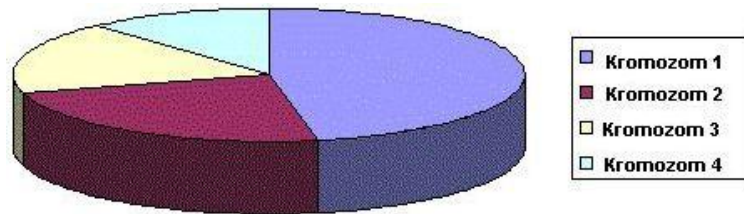
Kopyalama

Kopyalama işlemi, daha önceden uygunluk değerleri hesaplanmış bireyler arasından sağlıklı nesillerin yeni nesillere kopyalanması işlemidir. Ayrıca yeni kromozomlardan oluşan daha güçlü bir popülasyonun oluşmasına yardımcı olur.

Evrım teorisine göre, varlıklar arasında en iyilerin yaşama şansı her zaman daha yüksektir. Klasik yöntemlerin tümünde her an yegane diyebileceğimiz bir tane en iyi çözüm olmasına karşılık, genetik algoritmalarda çözüm olmaya aday bir nüfus vardır. Bunlar arasında sadece bir tanesi en iyidir. Diğerleri de bulundukları karar değişkeni bölgesinde en iyi olmaya adaydır [66].

En iyi olan bireylerin yeni nesilde kaybolmaması ve ağırlığını gösterebilmesi için geliştirilen ve bir sonraki nesle aktarılmasına olanak veren bir takım seçim yöntemleri bulunmaktadır. Bunlardan en yaygın kullanılanları rulet çarkı seçimi ve rank seçimi yöntemleridir.

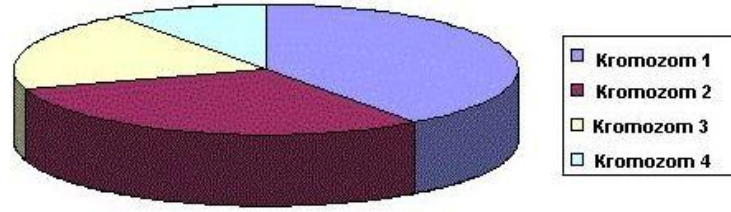
Rulet çarkı seçimi bireylerin uygunluk fonksiyonlarının çark üzerinde kapladığı alanla ilişkilidir. Böylece sağlıklı olmayan bireylere de seçilme şansı doğmaktadır. Her bir bireyin uygunluk değeri, kendisi de dâhil olmak üzere tüm bireylerin uygunluk değerleri toplamına bölünerek seçilme yüzdeleri hesaplanır. Seçilme yüzdeleri aynı zamanda rulet tekerleği üzerinde kaplayacağı alanı da temsil etmektedir. Kromozomların uygunluk değerleri ne kadar iyiye seçilme şansı o kadar fazla olmaktadır.



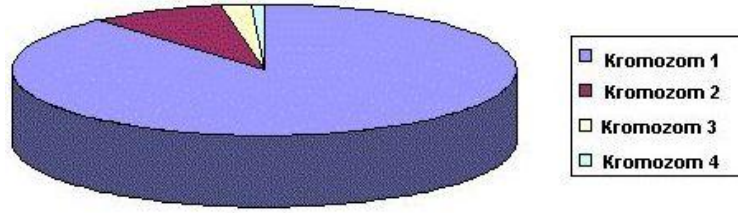
Şekil 4. 5 Kromozomların uygunluk değerine göre sıralanması

Seçim işlemi gerçekleştirilirken, seçilecek kromozomu belirlemek için sıfır ile yüz arasında rastgele bir sayı üretilmektedir. Bireylerin sahip olduğu seçilme yüzdeleri birikimli olarak toplanmakta ve sıfırdan yüze kadar bireylerin sınıf aralıkları oluşturulmaktadır. Üretilen sayı hangi sınıfa denk geliyorsa o aralığa denk gelen değere ait birey bir sonraki nesle kopyalanır [62] ,[63],[65].

Rank seçimi yöntemi, popülasyonu oluşturan bireylerin uygunluk değerleri arasında çok büyük farklar varsa kullanılmaktadır. Bir uygunluk değerinin rulet çarkı üzerinde kapladığı alan fazla ise diğer kromozomların seçilme şansı düşecektir. Rank seçiminde bu ölçeklendirme problemi ortadan kalkmaktadır (Şekil 2.6 ve 2.7).



Şekil 4. 6 Kromozomların Uyumluluk değerine göre ağırlıkları



Şekil 4. 7 Rulet çarkına göre seçilme

Rank seçimi yönteminde, rulet çarkı yönteminde olduğu gibi, popülasyonu oluşturan bireylerin uygunluk değerleri hesaplanmaktadır. Daha sonra uygunluk değerleri amaç fonksiyonunun beklentisine (minimum, maksimum) göre sıralanmaktadır. Uygunluk değerleri sıralandıktan sonra, en kötü olan uygunluk değerine 1, en iyiye ise N (popülasyondaki birey sayısı) seçilme değeri verilmektedir. Seçilme yüzdeleri hesaplanırken 1'den N'ye kadar olan tüm değerler toplanır ve her bir bireyin sahip olduğu değer, bu toplam değere bölünerek seçilme şansı yüzdesine ulaşılmaktadır. Bu işlemden sonra rulet çarkında olduğu gibi rassal sayılar yardımı ile yeni nesle eklenecek bireyler seçilecektir [65],[67].

Seçim yöntemleri haricinde kopyalama yapan bir takım yöntemlerde bulunmaktadır. Yaygın olarak kullanılan kopyalama teknikleri bütünüyle yer değiştirme, jenerasyon aralığı ve kararlı durum yöntemleridir.

Bütünüyle yer değiştirme yönteminde her hangi bir seçim işlemi olmaksızın bütün bireyler yeni nesle aktarılmaktadır. Sadece genetik operatörler sayesinde bir takım değişikliğe uğrayan bireyler yeni birer birey haline gelmektedir [66].

Jenerasyon aralığı yönteminde yine bir seçim işlemi olmaksızın bireylerin bir kısmı belirlenen bir yüzde ile yeni nesle aktarılmaktadır. Aktarılabacak bireyler, rastgele

seçileceği gibi belli bir sıralama sonrasında da seçilebilmektedirler. Popülasyon birey sayısını tamamlamak içinse yeniden bireyler üretilir. Böylelikle popülasyona girememiş bireylere de şans verilmiş olmaktadır. Eğer bu değer %100 olarak verilirse bütünüyle yer değiştirme yapılmış olacaktır. Küçük yüzdelerde çeşitlilik de fazlaşacağından yeni neslin daha iyi sonuçlar verebileceğini garanti etmemektedir [18],[66].

Kararlı durum yöntemi ise kararlı durum genetik algorithmada kullanılmaktadır. Kararlı durumda bireylerin sadece 1 tanesi silinmekte ve sadece 1 tane yeni dizi üretilmektedir. Böylece geri kalan tüm bireyler yeni nesle kopyalanarak, her nesilde sadece 1 adet yeni birey oluşturulmaktadır [18],[66].

Elitizm ise kopyalamanın vazgeçilmezlerindendir. Elitizm ile yeni nesle bir veya birkaç en iyi kromozom doğrudan kopyalanmakta ve kopyalanan bu bireylere hiçbir operatör uygulanmamaktadır. Eğer yeni nesilde daha iyi bireyler varsa seçilmiş bireylerin yerini almakta ve süreç bu şekilde devam etmektedir. Elitizm kararlı durum, bütünüyle yer değiştirme ve jenerasyon aralığı yöntemleriyle beraber kullanılabilir. Böylelikle kopyalanan bireyler içerisinde en iyi bireyler kaybolmadan bir sonraki nesle aktarılmış olacaktır [18],[67].

Çaprazlama

Çaprazlama, popülasyonu oluşturan bireyler arasındaki gen alışverişi olarak tanımlanabilmektedir. Bu operatörün görevi, bir önceki nesilden kopyalanan bireylerin, kromozom yapılarını değiştirerek popülasyona yeni bireyler oluşturmaktır. Böylece problemin çözüm kümesi içerisinde, uygun olabilecek diğer çözümlere erişebilmeye olanak sağlanmaktadır. Ancak bu durum yeni oluşturulan bireyin daha iyi olmasını garanti etmemektedir [65].

Çaprazlama, iki adet bireyin popülasyon içerisinde rassal olarak seçilerek, yine rassal olarak belirlenen bir noktadan itibaren genlerinin yer değiştirmesi işlemidir. Doğal yöntemlerde çaprazlama sonucu tek yeni birey oluşurken, yapay çaprazlama sonucunda iki adet yeni birey oluşmaktadır. Bu operatörün uygulanması ile ilgili yöntemler tek kesimli, çift kesimli, çok kesimli ve tekdüze kesimli olarak gösterilebilmektedir [62],[65].

Öncelikle çaprazlama yapılacak kromozom sayısını belirlemek için bir çaprazlama oranı belirlenmektedir. Popülasyondaki birey sayısı ile bu oranı çarpılması sonucunda çaprazlama yapılacak kromozom sayısı elde edilmektedir. Çaprazlama oranı genelde yüksek seçilmektedir. Böylelikle değişik bireylerin oluşması ve uygunluklarının değerlendirilmesi hızlanabilecektir.

Çaprazlanacak kromozom sayısı belirlendikten sonra, bu sayı kadar popülasyon içinden rastgele kromozomlar seçilmektedir. Daha sonra belirlenen çaprazlama yöntemi kullanılarak işleme devam edilmektedir.

Çaprazlama yöntemlerinden ilki tek kesimli çaprazlamadır. Bu yöntem ile tek bir noktadan çaprazlama gerçekleştirilir. Bu nokta rassal seçilmektedir ve konumu hangi geni gösteriyorsa o genden itibaren iki birey arasında genler yer değiştirmektedir [62],[65]. Seçilecek nokta, kromozomun uzunluğu k olması durumunda, 1 ile $k-1$ arasında bir sayı olmalıdır. Kesim noktası m olan bir çaprazlamada $m+1$ 'den k 'ya, k . gen de dâhil olmak üzere genler yer değiştirecektir. Kromozom uzunlukları $k=7$ olan iki adet bireyin oluşturduğu rastgele üretimin sonucunda bir $m=3$ çaprazlama noktası değeri belirlenmiş ve çaprazlama noktası “|” ile gösterilen bir çaprazlama Şekil 4. 8'deki gibi gerçekleşmektedir [62],[65].

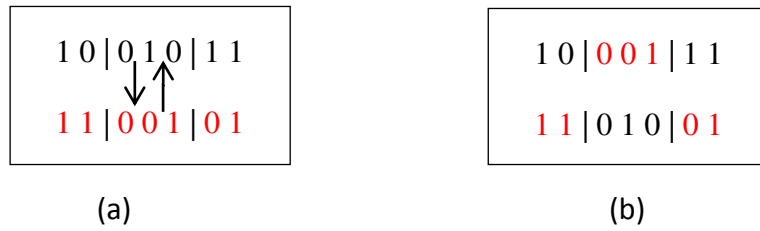


Şekil 4. 8 Tek kesimli çaprazlama (a) çaprazlama öncesi bireyler
(b) çaprazlama sonrası bireyler

Şekil 4.8 a'da görüldüğü gibi çaprazlama ok yönünde gerçekleşmiş ve Şekil 4.8 b'deki bireyler meydana gelmiştir.

Çift kesimli çaprazlamada, iki adet kesim noktası bulunmaktadır. Gerçekleşme mantığı tek kesimli çaprazlama ile benzerlik göstermektedir. Tek farkı yer değiştirecek olan genlerin bu iki konum arasında kalan bölüm olmasıdır [62],[65].

Şekil 4.8 a'daki bireylere, kesim noktaları $m_1=2$ ve $m_2=5$ olan çift kesimli çaprazlama uygulaması Şekil 4-9'da gösterilmiştir.



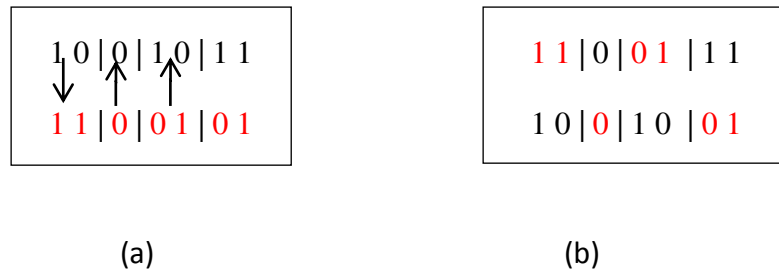
Şekil 4. 9 Çift kesimli çaprazlama (a) çaprazlama öncesi bireyler

(b) çaprazlama sonrası bireyler

Şekil 4.9'da görüldüğü gibi bireyler üç parçaya bölünmüş ve ortadaki parça üzerindeki genler diğer bireyin genleri ile yer değiştirerek iki adet yeni birey oluşturmuşlardır.

Çok kesimli çaprazlamada n kadar kesim noktası bulunmaktadır. Bu sayı problemin gelişine veya karar vericiye göre değişebilir [62],[65].

Şekil 4.8 a'daki bireylere, kesim noktaları $m_1=2$, $m_2=3$ ve $m_3=5$ olan $n=3$ kesimli çaprazlama uygulaması Şekil 4.10'da gösterilmiştir.



Şekil 4. 10 Çok kesimli çaprazlama (a) çaprazlama öncesi bireyler

(b) çaprazlama sonrası bireyler

Şekil 4.10'da görüldüğü gibi bireyler $n+1$ adet parçaya bölünmüş ve bölünen parçalardan, istenilen parçalar bireyler arasında yer değiştirmiştir. Yer değiştirmelerin sırayla yapılması için bir koşul bulunmamaktadır. Tekdüze çaprazlama, rastgele hanelerin iki dizi arasında yer değiştirmesi ile gerçekleşmektedir. Bu rastgelelik belirlenmeden önce kromozom içinde çaprazlanacak gen sayısının olasılığı belirlenir. Kromozom uzunluğu $k=10$ olan bir birey için bu olasılık değeri $p=0,5$ olarak seçildiği takdirde, pxk işleminin sonucunda 5 adet genin yer değiştireceği belirlenmiş

olmaktadır. Daha sonra bu 5 adet gen, kromozom içerisinde rassal olarak seçilmektedir (Şekil 4.11).



Şekil 4. 11 Tekdüze çaprazlama (a) çaprazlama öncesi bireyler

(b) çaprazlama sonrası bireyler

Şekil 4.11’de ikinci, üçüncü, beşinci, sekizinci ve onuncu olmak üzere $p_{xk}=5$ adet gen ile çaprazlama gerçekleştirilmiştir [65],[66].

Mutasyon

Çaprazlama neticesinde oluşan ve eski nesillerin gen özelliklerini taşıyan bireylere yeni özellikler kazandırmak için kullanılan yöntem mutasyon adı verilmektedir. Mutasyon, araştırma uzayında daha önceden denenmemiş ve uygun çözüm olmaya aday noktaları incelemeyi sağlamaktadır. Böylece arama yönünü değiştirerek, yeni ve uygun bir çözüm elde edebilmeye olanak vermektedir. Ancak bulunan yeni bireyin daha uygun bir birey olması mutlak bir durum değildir. Uygun olmayan bir çözüm ile karşılaşılsa bile, diğer genetik operatörler sayesinde bir sonraki nesilde bu uygun olmayan birey elenebilecek veya değiştirilebilecektir [62],[18].

Mutasyon operatöründe de aynen çaprazlamada olduğu gibi bir oran bulunmaktadır. Bu oran popülasyon içerisindeki kaç adet genin değişeceğinin belirlenmesi için kullanılmaktadır. Değişecek gen adedi mutasyon oranı ile gen sayısının (kromozom uzunluğu ve popülasyondaki birey sayısının çarpımı) çarpımı sonucu bulunmaktadır. Bu adet belirlendikten sonra popülasyon içerisindeki genler rastgele seçilerek değiştirilmektedir. Mutasyon operatörü, çaprazlama operatörüne göre daha kolay gerçekleştirilmektedir.

Mutasyon operatöründe ikili (binary) yapıda kromozomlardan oluşan bir küme tercih edildiği durumda, kromozom üzerinde seçilen genin değeri 0 ise 1, 1 ise 0 yapılmaktadır. Mutasyon oranı $p_m=0,05$ ve kromozom uzunluğu $k=25$ olan bir adet

kromozom üzerinde $pxk=1,25$ adet genin mutasyon geçireceği belirlenmiş olmaktadır. Bu sayı bir üst değere yuvarlanarak 2 adet gen üzerinde değişiklik yapılmıştır. Mutasyon örneği Şekil 4.12’de görülmektedir. Değiştirilmek üzere seçilen genler rastgele belirlenmiştir.

1 1 0 1 1 1 1 0 0 1 0 0 1 1 0 1 0 0 0 1 1 0 1 0 1	Bireyin ilk hali
1 1 0 1 1 1 1 0 0 0 0 0 1 1 0 1 0 0 0 1 1 1 0 1	Değişmiş birey

Şekil 4. 12 Mutasyon Örneği

Şekil 4.12’de de görüldüğü üzere onuncu ve yirmi ikinci gen mutasyon işlemi için seçilmiş ve değiştirilmiştir.

4.3.2.4 Genetik Algoritma Parametreleri

En iyileme yöntemlerinde parametreler problemin çözüme ulaşmasını sağlayan en etkili elemanlardır. Bir en iyileme yöntemi olan genetik algoritmanı da parametreleri bulunmaktadır. Genetik algoritma parametreleri tüm problemlerde benzer olmasına rağmen alacağı değerler problemin çözüm aralığına göre değişiklik göstermektedir. Bu parametreler; popülasyon büyüklüğü, çaprazlama oranı, mutasyon oranı, seçim stratejisi, dizinin kodlama stili, çaprazlama stili ve mutasyon stili olarak sayılabilmektedir [67].

Popülasyon büyüklüğü, genetik algoritma kullanıcısı tarafından verilen en önemli kararlardan birisidir. Bu değer çok küçük olduğunda genetik algoritma yerel bir en iyiye takılabilmektedir. Popülasyonun çok büyük olması ise çözüme ulaşma zamanını arttırmaktadır [18]. Bu konuda Goldberg [68], yalnızca kromozom uzunluğuna bağlı bir popülasyon büyüklüğü hesaplama yöntemi önermiştir. En iyi popülasyon büyüklüğü eldeki sorunun davranış biçimine göre belirlenmelidir [69].

Başlangıç popülasyonu belirlendikten sonra, kromozomu oluşturan gen dizilerinin kodlama stili belirlenmelidir. Algoritmanın çalışması esnasında uygulanacak genetik operatörler bu parametrenin seçimi ile şekillenmektedir. Kodlama çözüm için izlenen yola göre değişmektedir. Aynı problem için farklı kodlama stilleri kullanılabilmektedir. Bu stillerden ikili kodlama kullanım kolaylığı açısından en çok tercih edilen yöntemdir.

İkili kodlama dışında permütasyon kodlama, değer kodlama ve ağaç kodlamada bulunmaktadır. İkili kodlamada 0 ve 1, permütasyon kodlamada onluk düzende sayılar, değer kodlamada reel sayılar, karmaşık sayılar veya harfler kullanılabilir. Ağaç kodlama ise genellikle genetik programlamada kullanılmaktadır [67].

Çaprazlama oranı, çaprazlama operatörüne tabi bırakılacak kromozom adedini belirlemek için bir ölçüt olarak kullanılmaktadır. Çaprazlama ile mevcut bireylerin genetik özelliklerini birbirleriyle değiştirerek daha iyi bireyler oluşturmak amaçlanmıştır. Ancak çaprazlama oranını yüksek tutmak iyi bireylerin bozulma olasılığını ve işlem süresini arttırmaktadır [70]. Bu oranı düşük tutmak ise, eski nesildeki birçok bireyin yeni nesle doğrudan kopyalanmasına ve böylelikle yeni neslin sürekli yerinde saymasına neden olmaktadır.

Mutasyon oranı, değişim geçirecek genlerin belirlenmesi için kullanılmaktadır. Amaç popülasyondaki genetik çeşitliliği korumaktır. Eğer mutasyon oranı artarsa, genetik arama rassal bir aramaya dönüşecektir. Fakat bu aynı zamanda kayıp genetik bireyleri tekrar bulmaya da yardımcı olmaktadır. Bu oran genelde popülasyondaki birey sayısı arttıkça artış gösterir [67].

Çaprazlama olasılığı, genellikle 0.25 ila 1 arasında, mutasyon olasılığı 0.01 ila 0.001 arasında seçilmektedir. Özellikle küçük popülasyonlarda sistemin performansını çaprazlama olasılığından çok mutasyon olasılığı belirler [66].

Seçim stratejisi, eski nesilden yeni nesle bireylerin aktarılması için kullanılacak yöntemin belirlenmesidir. Seçim operatörü, uygunluk değerlerinin analizi ile belirlenebilmektedir.

Tek kesimli çaprazlama en güçlü çaprazlama yöntemidir. Bunun nedeni eski nesildeki bireyin bir kısmının tamamen yeni nesildeki bireye kopyalanmasıdır. Böylelikle iki adet iyi bireyin birleşmesinden daha iyi bir birey meydana gelebilmektedir. Bu konuda en zayıf yöntem ise tekdüze çaprazlamadır [65].

Mutasyon ise kromozom içinde rastgele haneler seçilerek bu hanelerin değiştirilmesi ile gerçekleşmektedir. Mutasyon ve çaprazlama ile ilgili parametrelerin kullanım amacı her yeni nesilde daha iyi bireyler oluşturmaktır. Ancak seçim, çaprazlama veya mutasyon sırasında kötü bireylerde oluşabilmektedir ve uygunluğu yüksek olan bir

bireyin bozulmasına neden olabilmektedir [65]. Bu nedenle özellikle mutasyon oranı düşük tutulmaktadır. Böylelikle bir kromozom içinde birden fazla genin deęiřmesi ile kromozomun tamamen farklılařması engellenmiř olacaktır.

Parametreleri oluřturmak iin yapılan seimler zme ulařırken izleyeceėimiz yntemleri belirlemektedir. Bu parametrelerin yanı sıra ama fonksiyonun belirlenmesi de byk nem tařımaktadır. Amacı belli olmayan bir genetik algoritmanın uygun zm bulabilmek iin gerekli parametreleri oluřturması beklenemez.

BÖLÜM 5

LOJİSTİK FİRMA UYGULAMASI

Bu bölümde önce mevcut durum, tezin amacı, kapsamı ve araştırma yöntemine ilişkin bilgilere yer verilmiş, sonrasında verilerin analizi, elde edilen bulgular ve bulgular ışığında değerlendirmeler yapılmıştır.

5.1 Mevcut Durum

Uygulama yapılan lojistik firması birçok sektördeki şirkete depolama ve nakliye çözümleri başta olmak üzere farklı lojistik hizmetlerini entegre bir şekilde tedarik zinciri yönetimi başlığı altında sunmaktadır. Tedarik zinciri süreçlerinin geliştirilmesi için yaratıcı çözümler sunarken, üreticiden/tedarikçiden perakendeciye, nihai tüketicilere kadar her aşamada operasyonel mükemmeliyetin yakalanmasını sağlamaya çalışmaktadır. Lojistik işletmesi müşterileri tarafından tercih edilen ‘lider lojistik sağlayıcısı ve tedarik zinciri yöneticisi’ konumunda olmak istemektedir. Firmanın hizmet verdiği alanlardan biri de, belirli noktalardan heterojen araç filosu kullanılarak süt toplamasıdır. Bu bölümde Süt Tedarikinin lojistik sektörünün önde gelen bir işletmesinde uygulaması yapılmıştır. Bu çalışmada kullanılan bilgiler karşılıklı görüşme yoluyla işletme çalışanları, yöneticileri ve müşterilerden alınmıştır. Firmanın hizmet verdiği fabrikalardan biri Kahramanmaraş’ta süt ve süt ürünleri üreten bir firmadır. Çalışmada bu firmanın tedarik ettiği sütlerin çiftliklerden toplanmasıyla ilgili araç rotalama problemi ele alınmıştır. Süt siparişleri Excel ortamında alınarak, yine Excel ortamında manuel

seferler oluşturulmaktadır. 26.02.2014 tarihli süt siparişleri ele alınarak bu çalışma yapılmıştır.

Program çalıştırıldıktan sonra, elde edilecek sonuçlar, mevcut sistem ve matematiksel model ile karşılaştırılacaktır. Matematiksel modelde GAMS programı, hibrit modelde ise Microsoft excel ortamında makro programlama kullanılmıştır.

Süt toplama araçları kırkayak, kamyon ve tır olmak üzere 3 tiptir. Bu araçlar Türkiye'nin farklı bölgelerindeki süt çiftliklerinden alınması gereken çiğ sütleri toplayarak ihtiyaç olan fabrikalara nakil etmektedir.

5.1.1 Problemin Verileri

- Araçlar süt işleme merkezlerinden çıkıp tekrar süt işleme merkezlerine döner.
- Her çiftliğe bir araç uğrar.
- Rotadaki toplanacak süt miktarı araç kapasitesini geçemez. Tüm noktaların süt toplaması yapılır.
- Çiftliklerden alınacak süt miktarları günlüktür. Haftanın 6 günü bu çiftliklerden süt alınmaktadır.
- Bazı noktalara araç tiplerinden Tır giremez kuralı söz konusudur.
- Kat edilecek yol, araç sayısı minimize edilmeye çalışılır.
- Pilot çalışma Kahramanmaraş fabrikası ve bağlı olduğu çiftlikler üzerinde yapılmıştır.

Araç tipleri hakkındaki veriler aşağıdaki çizelge 5.1 de gösterilmektedir.

Çizelge 5. 1 Araç tipleri hakkındaki veriler

Araç Tipi	Araç Sayısı	Araç Kapasitesi	YAKIT LT /KM	Mak Km Sınırı	TL/LT	Sabit Maliyet(TL)/Sefer
Kamyon	5	15000	0,3	300	4,38	125,00
Kırkayak	9	20000	0,32	300	4,38	160,71
Tır	3	25000	0,35	300	4,38	232,14

Çizelge 5.1' de görüldüğü gibi, kapasitesi 15000 LT olan 5 adet, 20000 LT olan 9 adet ve 25000 LT olan 3 adet araç bulunmaktadır. Rotaların belirlenmesinde yapılan km kısıdı

göz önüne alınmıştır. Uygulamada firma için günlük olarak en uygun dağıtım rotası bulunacağı için, dağıtım firmasının karşılaştığı günlük maliyet, o günde oluşacak yakıt ve sabit maliyet toplamıdır.

Sütlerin taşınacağı nokta sayısı 60'tır. İşletmedeki farklı tipteki araçların bu noktalardan süt alıp ihtiyaç olan adreslere bırakırken kat edeceği mesafenin kilometre cinsinden değerleri Ekler bölümünde Ek A'da gösterilmektedir.

60 farklı noktadan gelen 26.02.2014 gününe ait siparişler (toplanacak süt miktarı) aşağıdaki çizelge 5.2 ve 5.3 de listelenmiştir.

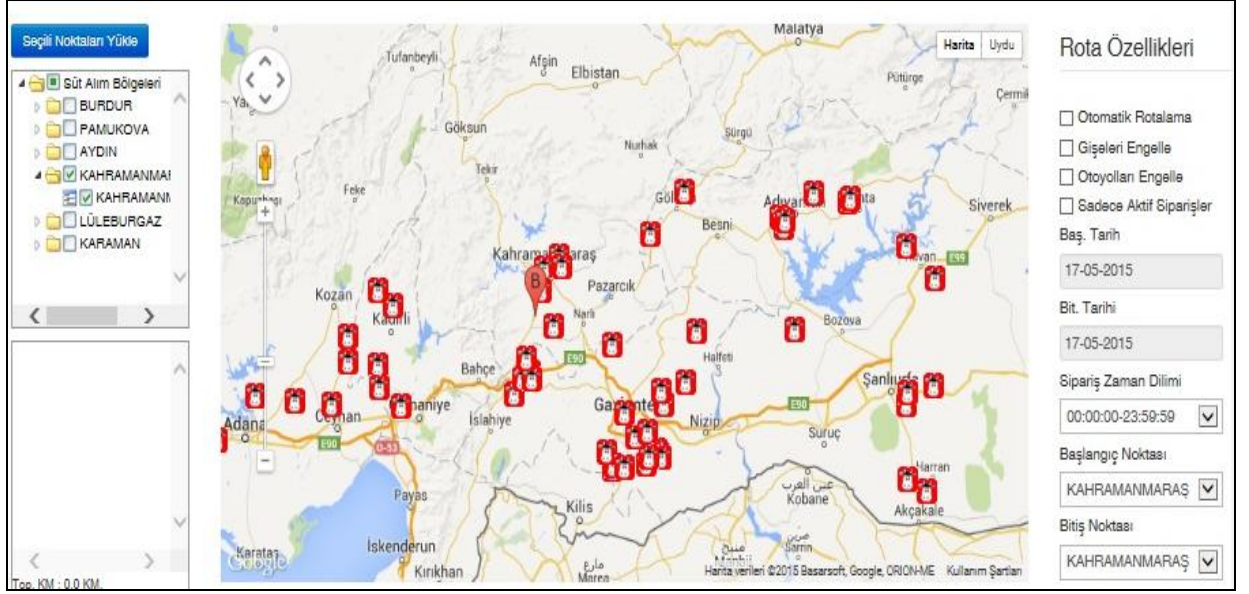
Çizelge 5. 2 26.02.2014 Tarihli Süt Siparişleri

Nokta	Nokta Adı	İl	İlçe	Günlük Süt Miktarı (LT)
1	KAHRAMANMARAŞ(FABRİKA)	KAHRAMANMARAŞ	Merkez	-
2	ADEM IŞIK	GAZİANTEP	SEHİTKAMİL	4000
3	ADNAN TÜKETİM	GAZİANTEP	NURDAGI	2800
4	AHMET BAYRAKTAR	GAZİANTEP	OGUZELİ	3750
5	AHMET CAHİT	OSMANİYE	TOPRAKKALE	6000
6	AKAY DOĞU	SANLIURFA	AKÇAKALE	1500
7	ALİ ŞAHAN	GAZİANTEP	NİZİP	2200
8	ALTUNLAR	SANLIURFA	MERKEZ	8055
9	ANAZ KÖYÜ	SANLIURFA	HARRAN	7801
10	AYHAN TASDOĞAN	GAZİANTEP	SAHİNBEY	9832
11	BALKESENLER	GAZİANTEP	NURDAGI	7521
12	BESLER	GAZİANTEP	OGUZELİ	3730
13	BİLİM	KAHRAMANMARAŞ	MERKEZ	2200
14	BİZİM SÜT	GAZİANTEP	NURDAGI	8109
15	CEYLAN VETERİNERLİK	GAZİANTEP	SEHİTKAMİL	7855
16	CEYLANPINAR	SANLIURFA	CEYLANPINAR	9882
17	CUMALİ BUCAK	SANLIURFA	AKÇAKALE	7602
18	ÇALKAYA	KAHRAMANMARAŞ	MERKEZ	3750
19	ÇAPALI KÖYÜ	GAZİANTEP	SAHİNBEY	10393
20	DERVİŞ BEYAZILDIZ	ADİYAMAN	GÖLBASI	16069
21	DOĞAN AKDOĞAN	GAZİANTEP	NİZİP	4004
22	FEHMİ TURANLI	ADİYAMAN	KAHTA	2358
23	GİNES	ADANA	İMAMOĞLU	7580
24	GÖKPA TARIM	KAHRAMANMARAŞ	MERKEZ	6010
25	GÖLBASI (ADİYAMAN)	ADİYAMAN	GÖLBASI	2230
26	GÜRBÜZ(YAVUZELİ)	GAZİANTEP	YAVUZELİ	2851
27	H.G.B İNŞAAT	ADİYAMAN	MERKEZ	1503
28	HACI YUSUF REÇBER	ADİYAMAN	KAHTA	2612
29	HACİBEYLİ	ADANA	KOZAN	4045

Çizelge 5. 3 26.02.2014 Tarihli Süt Siparişleri Devamı

Nokta Kodu	Nokta Adı	İl	İlçe	Günlük Süt Miktarı (LT)
30	HASAN TURAN	ADIYAMAN	KAHTA	4495
31	HÜNKAR TARIM	KAHRAMANMARAS	MERKEZ	4430
32	IRMAK TARIM	MARDIN	DERİK	2877
33	İBRAHİM DAMAR	GAZİANTEP	SAHİNBEY	3584
34	İDO BAY	GAZİANTEP	NURDAGI	6705
35	İNTERVET	GAZİANTEP	YAVUZELİ	8600
36	KALENDER HAYVANCILIK	SANLIURFA	HILVAN	8984
37	KARABİBER	GAZİANTEP	OGUZELİ	7345
38	KARAKÜTÜK	OSMANİYE	KADIRLI	4412
39	KENAN PAMUKÇU	OSMANİYE	MERKEZ	2588
40	KEPİRHİSAR	SANLIURFA	HILVAN	565
41	KÖMÜRCÜLER	GAZİANTEP	NURDAGI	1288
42	KÖYTEPE	GAZİANTEP	OGUZELİ	1103
43	LARA TARIM	KILIS	ELBEYLİ	1043
44	MAHMUT AKGÜN	SANLIURFA	BOZOVA	1687
45	MARKOÇ HAY	KAHRAMANMARAS	MERKEZ	2164
46	MASER	GAZİANTEP	OGUZELİ	2262
47	MELİH KOLDANCA	ADANA	CEYHAN	1852
48	NURSÜT	GAZİANTEP	NURDAGI	1637
49	SAF ET	ADIYAMAN	KAHTA	2607
50	SALMAN POLAT	GAZİANTEP	NURDAGI	3878
51	SARIT	GAZİANTEP	SAHİNBEY	4456
52	SULTANTEPE KÖYÜ	SANLIURFA	HARRAN	4138
53	SUMBAS	OSMANİYE	SUMBAS	2979
54	SÜTKA	GAZİANTEP	SAHİNBEY	1647
55	TAÇ SÜT	ADANA	SEYHAN	673
56	TATARLI TARIM	ADANA	CEYHAN	144
57	UZER TARIM	GAZİANTEP	NURDAGI	1076
58	VEYSEL AYBAR	SANLIURFA	MERKEZ	1644
59	YENİ KEŞİF	GAZİANTEP	OGUZELİ	2112
60	YEŞİLDAM	ADANA	CEYHAN	2178
61	ZEKİ SELÇUK	ADIYAMAN	MERKEZ	1804

Kahramanmaraş bölgesine ait bu çiftliklerden alınacak sütlerin dağılımı görsel olarak aşağıdaki şekil 5.1' de harita üzerinden listelenmiştir.



Şekil 5. 1 Süt çiftliklerinin görsel olarak dağılımı

5.2 Problemin Tanımı

Çalışmanın temel amacı, en iyi çözümü verecek süt toplama noktaları rotasının ve her rotada kullanılacak araç tipinin belirlenmesidir. Bu çözüm, hedeflenen hizmeti veren, hizmetin kalitesini ve personel memnuniyetini sağlayan en düşük maliyetli çözüm olmalıdır. Fabrika ve özel taşımacılık şirketleri arasındaki mevcut sözleşmelere bakıldığında maliyetteki temel etkenlerin; araç tipi, sayısı ve sütlerin toplanacağı merkezlerin mesafesi olduğu göze çarpmaktadır. Kâğıt haritalar, krokiler gibi alışlagelmiş araçlarla bu miktarda detaylı bilginin toplanması ve kullanılması oldukça zor olmaktadır.

Araç tipleri, kapasitesi ve sayısı, firmanın süt toplayacağı noktaların konumları ve bu noktaların birbirlerine olan uzaklıkları, noktalardan toplanacak sütün miktarı gibi veriler toplanmış ve sorunlar gözlemlenmiştir.

Süt toplama noktalarının konumları ve birbirlerine km cinsinden olan uzaklıkları özel amaçlı oluşturulan bir bilgisayar programından temin edilmiştir. Araç tipleri, kapasitesi ve sayısı firmadan temin edilmiştir. Firmanın rotalama maliyetleri; araç kiralama sonucu sefer bazlı oluşan sabit ve yol alınan kilometreye göre yakıttan kaynaklı değişken giderlerden oluşmaktadır. Amaç bu giderlerin en aza indirgenmesidir.

Uygulama yaptığımız lojistik firmasının çözüm bulmamızı istediği kısıtlar aşağıda gösterilmiştir.

- Üç farklı tipteki aracın (kırkayak, kamyon, tır) maliyetlerin minimize edilmesi göz önünde bulundurularak uygun rotalara atanamaması,
- Fiziksel koşullardan dolayı tırın süt toplama için her noktaya atanamaması.

ARP'nin değişmeyen iki tane elemanı (araçlar, çiftlikler) vardır. Bu elemanların şu temel özelliklere sahip olduğu varsayılmaktadır.

Araçlar;

- Farklı özelliklere (kapasite ve hız) sahiptir.
- Sınırlı kapasiteleri vardır.
- Bakım giderleri yoktur. Kiralama yöntemi ile çalışmaktadırlar.

Çiftlikler;

- Her çiftliğin önceden bilinen günlük sabit miktarlarda süt arzı vardır.
- Servis süresi talep miktarına bağlıdır.
- Haftanın 6 günü süt toplama operasyonu yapılmaktadır.
- Sefere çıkan her aracın maksimum yol katedeceği km yolu kısıtı bulunmaktadır.

Uygulama yapılan lojistik firmasının karakteristik özelliklerini çizelge 5.4'de özetleyebiliriz.

Çizelge 5. 4 Lojistik Firmasının Karakteristik Özellikleri

No	Özellikler	Durum
1	Mevcut Araç Sayısı	17 araç
2	Mevcut Filo Tipi	Heterojen (3 tip)
3	Araçların Yerleşimi	Tek süt işleme merkezinden
4	Araç Kapasite Kısıtları	Belirli (Değişik araç kapasiteleri)
5	Maksimum Rota Km Sınırı	300 km
6	Maliyet	*Değişken veya rotalama maliyeti *Sabit işletme veya araç maliyeti
7	Amaç	*Toplam rotalama maliyetini minimize etmek *Gerekli araç sayısını minimize etmek

5.3 Problemin çözümünde kullanılan ek ana veriler

Proje’de kullanılacak bazı anaveriler ile ilgili düzenlemelere bu başlık altında yer verilmiştir.

5.3.1 Süt toplama noktalarının koordinatlarının ve mesafelerinin belirlenmesi

Süt çiftliklerinin adresleri lojistik firmasından temin edilmiştir. Bu verilere göre firmanın Kahramanmaraş fabrika dışında uğraması gereken 60 tane nokta vardır. Önce adres verileriyle firmanın bilgilerini almış olduğu özel bir programla mevcut koordinatlar belirlenmiştir. Koordinatların mevcut adres ile birbirine uyumluluğu kontrol edilmiştir. Koordinatların doğruluğu tespit edildikten sonra, bu noktaların birbirine olan uzaklıkları kilometre cinsinden hesaplanmıştır. Bu veriler ele alınarak 60x60’lık bir uzaklık matrisi oluşturulmuştur. EK-A’da bu matris yer almaktadır.

5.3.2 Farklı Tipteki Araçların Tanımlanması

Firmanın ana problemlerinden birisi, 3 farklı tipteki aracın (kamyon, kırkayak, tır) maliyet minimizasyonu hedef alınarak uygun rotalara atanmasıdır. Mevcut araçların sayısı, kapasitesi ve yakıt gideri bilgileri firmadan temin edilmiştir. Bu verilere çizelge 5.1’de yer verilmiştir. Listedeki bilgiler, diğer anaveriler ile birlikte ileriki bölümlerde geçen uygulamalardan hem GAMS hem de Makro programlamada kullanılacaktır.

5.3.3 Tırın Girip Giremeyeceği Noktaların Belirlenmesi

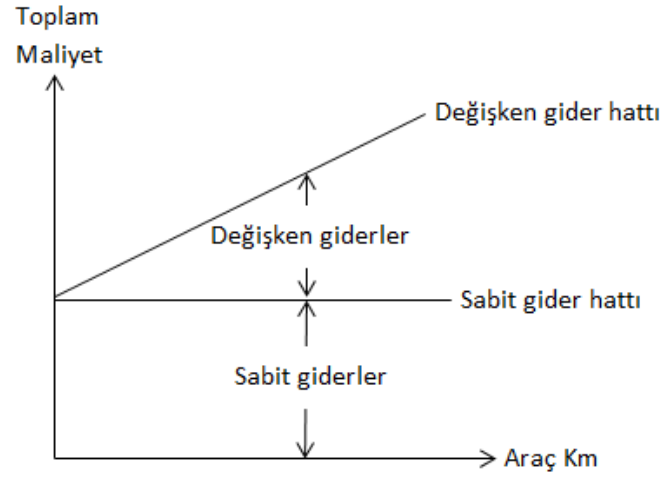
Firmanın ana problemlerinden bir diğeri de tırın her noktaya girememe kısıtıdır. Süt toplama farklı coğrafi yapılara sahip olan çiftliklerden temin edilmektedir. Tırın büyüklüğünden dolayı bu farklı yapıya sahip çiftliklerin hepsine girilememektedir. Bu kısıt göz önüne alınarak, koordinatları belirlenmiş 60 noktadan, tırın gireceği noktalar 1, giremeyeceği noktalar ise 0 olarak tanımlanıp bir matris oluşturulmuştur. Bu matris yine Gams ve Makro programlamada kullanılacaktır. Tırın girip giremeyeceği noktalara ait matris çizelge 5.5’de listelenmiştir.

Çizelge 5. 5 Tırın Girip Giremeyeceği Noktalar

	Noktalar																			
Nokta	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
TIR ?	0	1	0	0	0	0	0	0	0	1	1	0	1	1	1	1	0	0	1	0
Nokta	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41
TIR ?	1	0	0	0	1	1	0	0	0	0	1	0	1	1	0	0	0	0	0	1
Nokta	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61
TIR ?	1	1	0	0	1	0	0	0	1	0	0	0	1	0	1	1	1	1	0	0

5.4 Maliyet Hesaplaması

Maliyet hesaplamasında değişken ve sabit maliyet toplanarak toplam maliyet baz alınır. Her bir rota için bu maliyetler tek tek hesaplanır. Şekilde 5.2’de bunun grafik sunumu yer almaktadır.



Şekil 5. 2 Her rota için maliyet hesaplama

Burada bahsedilen sabit maliyet; araç kiralama, bakım giderleri, işçi çalıştırma vs gibi sabit nitelikli giderleri içine alır. Sabit maliyetler çizelge 5.1’de listelenmiştir.

Değişken maliyet ise her bir rota için bulunan km, km başına yakıt (lt olarak) ve lt başına TL değeri ile çarpılarak bulunur. Şekil 5.3’de değişken maliyetin görsel hali yer almaktadır.

$$\text{KM} \times \text{LT/KM} \times \text{TL/LT} = \text{Değişken Maliyet}$$

Şekil 5. 3 Her rota için değişken maliyet hesaplama

5.5 Mevcut Operasyonun Rotalama Sonuçları

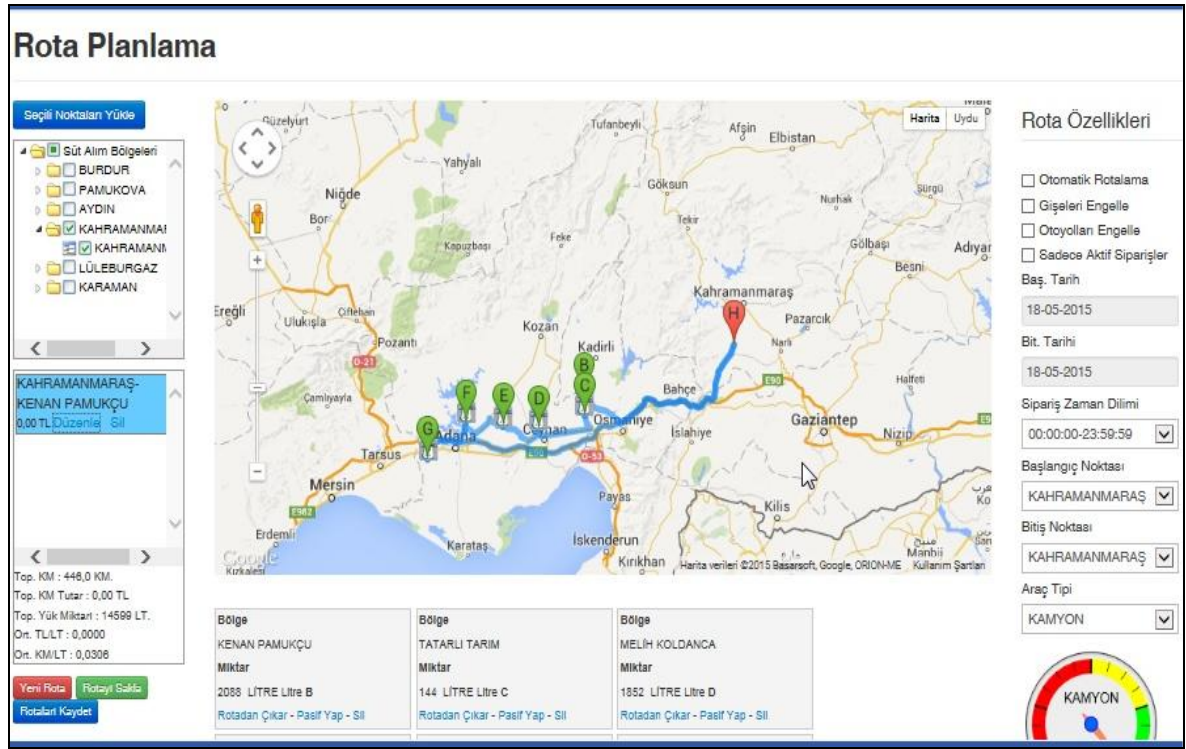
Planlama operasyon ekibi, mevcut sistemlerinde 26.02.2014 tarihine ait toplanması gereken süt miktarları ve diğer kısıtları(araç sayısı, nokta-tır kontrolü, maks km sınırı, araç doluluk) göz önünde bulundurup manuel olarak planlama yapmışlardır. Bu çerçevede elde edilen sonuçlar şu şekildedir:

Mevcut durumun toplam maliyeti 10942 TL oluşmuştur. Yapılan toplam yol mesafesi 6132 km'dir. Toplam 14 araçlık bir kullanım söz konusudur ve toplam araç doluluk oranı %92 dir. Çizelge 5.6'da bu değerler ve rotalar hakkında diğer bilgiler detaylı olarak yer almaktadır.

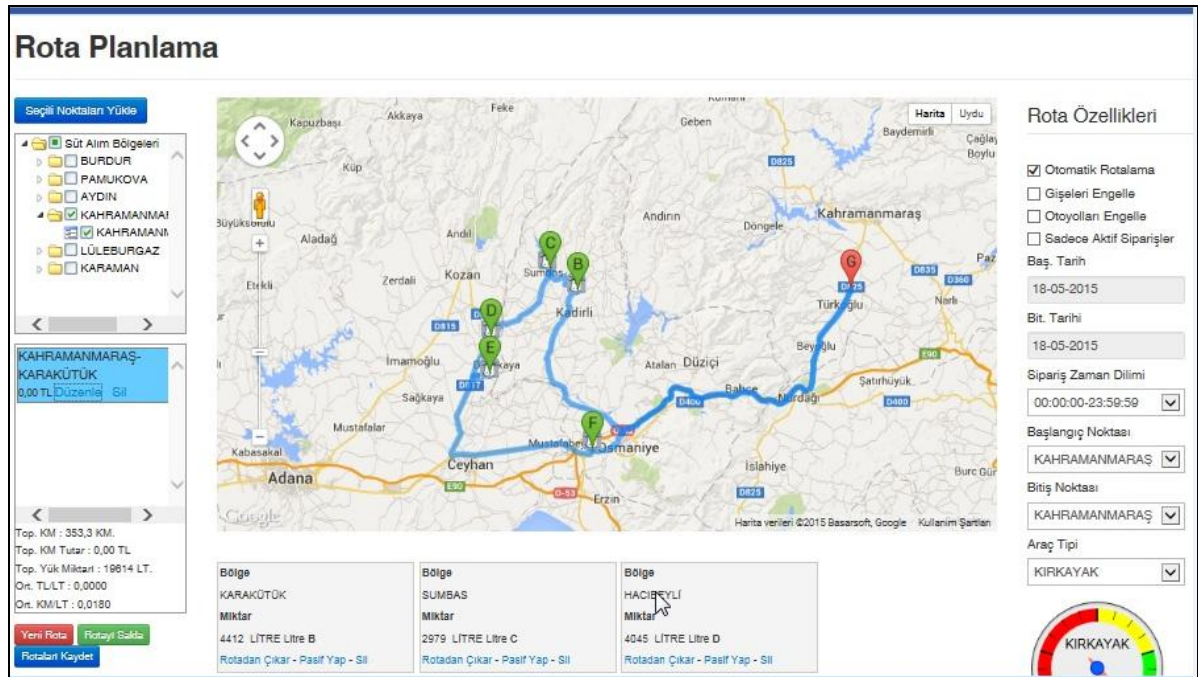
Çizelge 5. 6 Mevcut Operasyon Rotalama Sonuçları

Araç Tipi	Rota Noktaları								TL Rota Maliyet	Rota KM	LT Araç Doluluk	LT Araç Kapasite	Doluluk %
Kırkayak	1	56	55	46	23	47	5	1	953,20 TL	565,21	18511	20000	93%
Kırkayak	1	39	60	29	53	38	50	1	726,49 TL	403,46	19580	20000	98%
Kırkayak	1	34	48	3	14	1			299,46 TL	98,79	19251	20000	96%
Kırkayak	1	31	18	13	45	24	1		314,62 TL	109,6	18554	20000	93%
Kırkayak	1	4	6	9	33	7	1		1.379,75 TL	869,54	18835	20000	94%
Kırkayak	1	19	10	1					473,42 TL	222,9	20225	20000	101%
Kırkayak	1	51	40	36	21	44	1		1.298,33 TL	811,45	19696	20000	98%
Kırkayak	1	20	25	27	1				617,87 TL	325,96	19802	20000	99%
Kırkayak	1	52	28	30	22	61	49	1	908,81 TL	533,54	18014	20000	90%
Tır	1	32	16	58	17	57	41	1	1.617,53 TL	903,8	24369	25000	97%
Tır	1	35	15	11	1				604,37 TL	242,9	23976	25000	96%
Kamyon	1	2	8	1					783,69 TL	501,288	12055	15000	80%
Kamyon	1	54	43	59	42	37	1		455,69 TL	251,67	13250	15000	88%
Kamyon	1	12	26	1					509,74 TL	292,8	6581	15000	44%

Bu çalışma sonucu oluşan bazı rotaların harita sonuçları şekil 5.4 ve 5.5’de gösterilmektedir. Manuel rotalamaya ait tüm görseller proje ekinde verilmiştir.



Şekil 5. 4 Manuel rotalama sonucu oluşan örnek rota gösterimi 1



Şekil 5. 5 Manuel rotalama sonucu oluşan örnek rota gösterimi 2

5.6 Problemin Matematiksel Modeli ve GAMS ile Çözümü

Bu bölümde, ele alınan problemin kısıtları ve beklenileni göz önünde bulundurularak oluşturulan matematiksel model ve bu modele uygun şekilde problemin GAMS’de kodlanarak çözümü hakkında bilgiler yer almaktadır.

5.6.1 Problemin Matematiksel Modeli

Öncelikle problemin çözümünde kurulan modelin açıklanmasında kullanılacak olan değişkenler ve diğer yardımcı parametreler aşağıda listelenmiştir.

Modelde Kullanılan İndisler, Parametreler, Değişkenler

İndis seti

i	Çıkış noktaları indeksi (süt toplama merkezi veya süt alım noktası)
j	Varış noktaları indeksi (süt alım noktası veya süt toplama merkezi)
k	Araç tipi indeksi (Tır, Kırkayak, Kamyon)

Parametreler

d_{ij}	i noktasından j noktasına olan mesafe
c_k	k aracının km başına taşıma yakıt litre oranı
f_k	sefere çıkan k aracının sabit maliyeti
y_{TL}	yakıt litre birim maliyeti
u_j	j noktasından alınması gereken süt miktarı
Q_k	k aracının süt alım kapasitesi
M	Toplam araç sayısı
N	Toplam süt alım noktası sayısı
mk_{km}	aracın yapabileceği maksimum km sınır değeri

$$s_{kj} = \begin{cases} 1, & k \text{ aracı } j \text{ noktasına giriyorsa} \\ 0, & \text{Aksi halde} \end{cases}$$

Değişkenler;

$$x_{ijk} = \begin{cases} 1, & i \text{ noktasından } j \text{ noktasına } k \text{ aracı ile gidiliyor ise} \\ 0, & \text{Aksi halde} \end{cases}$$

Modelin Amaç Fonksiyonu:

$$\min z = (ytl * \sum_{i=0}^N \sum_{j=0}^N \sum_{k=1}^M c_k * d_{ij} * x_{ijk}) + (\sum_{j=0}^N \sum_{k=0}^M f_k * x_{1jk})$$

Modelin Kısıtları

$$\sum_{j=1}^N \sum_{k=1}^M x_{ijk} = 1 \quad i=2,...,N, \quad \forall j \neq i \quad (1)$$

$$\sum_{i=1}^N \sum_{k=1}^M x_{ijk} = 1 \quad j=2,...,N, \quad \forall i \neq j \quad (2)$$

$$\sum_{j=2}^N \sum_{k=1}^M x_{1jk} \leq M \quad \forall j \in N, \quad \forall k \in M \quad (3)$$

$$\sum_{i=2}^N \sum_{k=1}^M x_{i1k} \leq M \quad \forall i \in N, \quad \forall k \in M \quad (4)$$

$$\sum_{i=1}^N \sum_{k=1}^M x_{iik} = 0 \quad \forall i \in N, \quad \forall k \in M \quad (5)$$

$$\sum_{i=1}^N x_{ijk} = \sum_{i=1}^N x_{jik} \quad j=2,...,N, \quad k=1,...,M \quad (6)$$

$$\sum_{k=1}^M x_{ijk} + \sum_{k=1}^M x_{jik} \leq 1 \quad i=2,\dots,N, \quad j=2,\dots,N \quad (7)$$

$$\sum_{i=1}^N \sum_{j=2}^N u_j * x_{ijk} \leq Q_k \quad k=1,\dots,M \quad (8)$$

$$\sum_{i=1}^N x_{ijk} \leq s_{kj} \quad j=2,\dots,N, \quad k=1,\dots,M \quad (9)$$

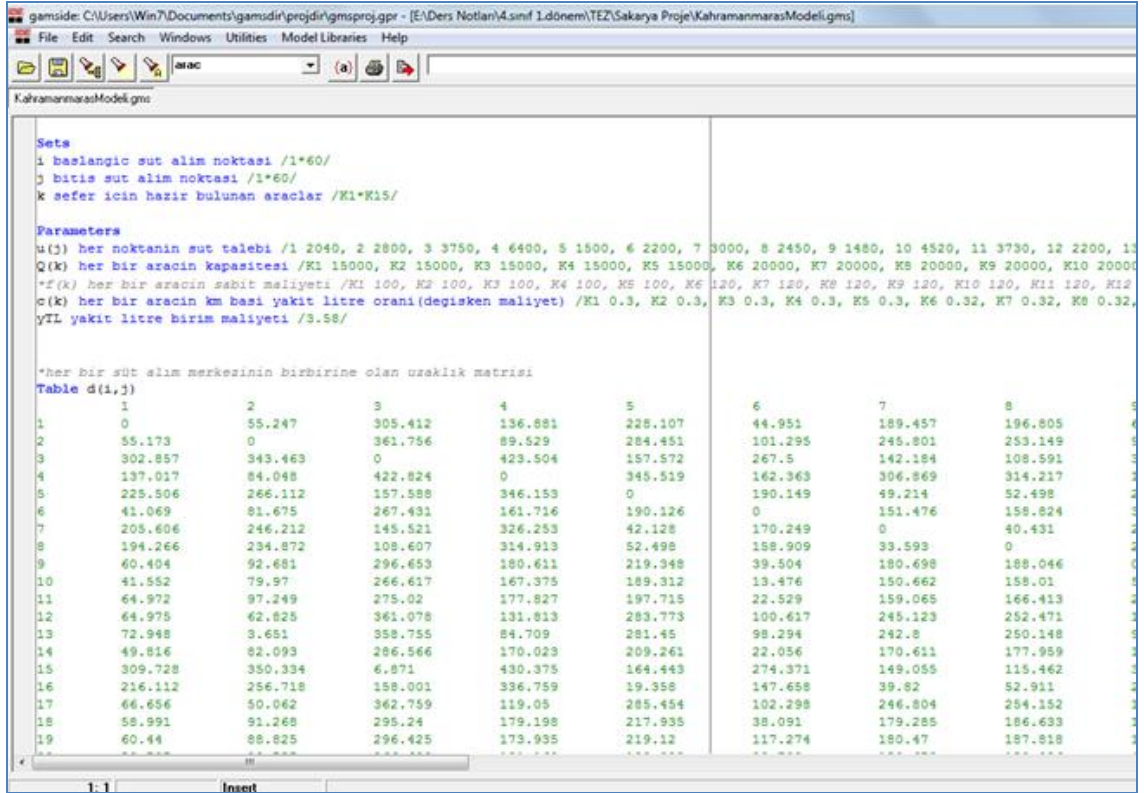
$$\sum_{i=2}^N \sum_{j=2}^N d_{ij} * x_{ijk} \leq m_{kkm} \quad k=1,\dots,M \quad (10)$$

Min z amaç fonksiyonu, yola çıkan her bir aracın değişken ve sabit maliyetlerinin toplamının minimize edilmesidir.

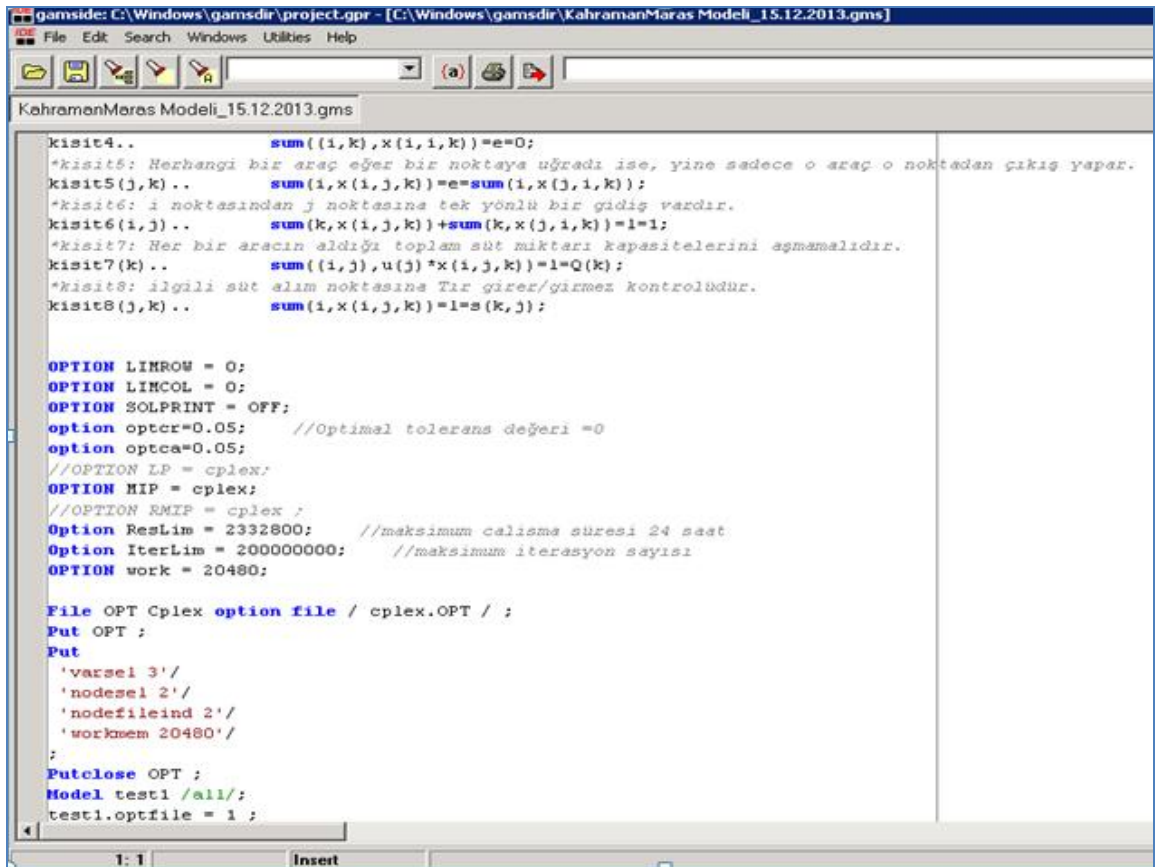
- (1): Her bir süt alım noktasından sadece tek araç çıkış yapar.
- (2): Her bir süt alım noktasına sadece tek araç uğrar.
- (3): Yola çıkan tüm araçların ilk çıkış noktası fabrikadır.
- (4): Seferi tamamlayan tüm araçların dönüş noktası yine fabrikadır.
- (5): Aracın başlangıç ve bitiş noktası aynı nokta olmaması kontrolüdür.
- (6): Herhangi bir araç eğer bir noktaya uğradı ise, yine sadece o araç o noktadan çıkış yapar.
- (7): i noktasından j noktasına tek yönlü bir gidiş vardır.
- (8): Her bir aracın aldığı toplam süt miktarı kapasitelerini aşmamalıdır.
- (9): İlgili süt alım noktasına Tır girer/girmez kontrolüdür.
- (10): Her bir rota kümesi içerisinde, noktalar arası yapılan toplam km, maksimum km parametresini aşmamalıdır.

5.6.2 Matematiksel Modelin GAMS’de Gösterimi

Elde edilen veriler ışığında oluşturulan matematiksel model GAMS programında çözümlenerek maliyeti minimize eden kesin sonuç elde edilmiştir. GAMS kodlama ile ilgili bazı ekran görüntüleri şekil 5.6 ve 5.7’de yer almaktadır.



Şekil 5. 6 Matematiksel modelin GAMS kodlama görüntüsü 1



Şekil 5. 7 Matematiksel modelin GAMS kodlama görüntüsü 2

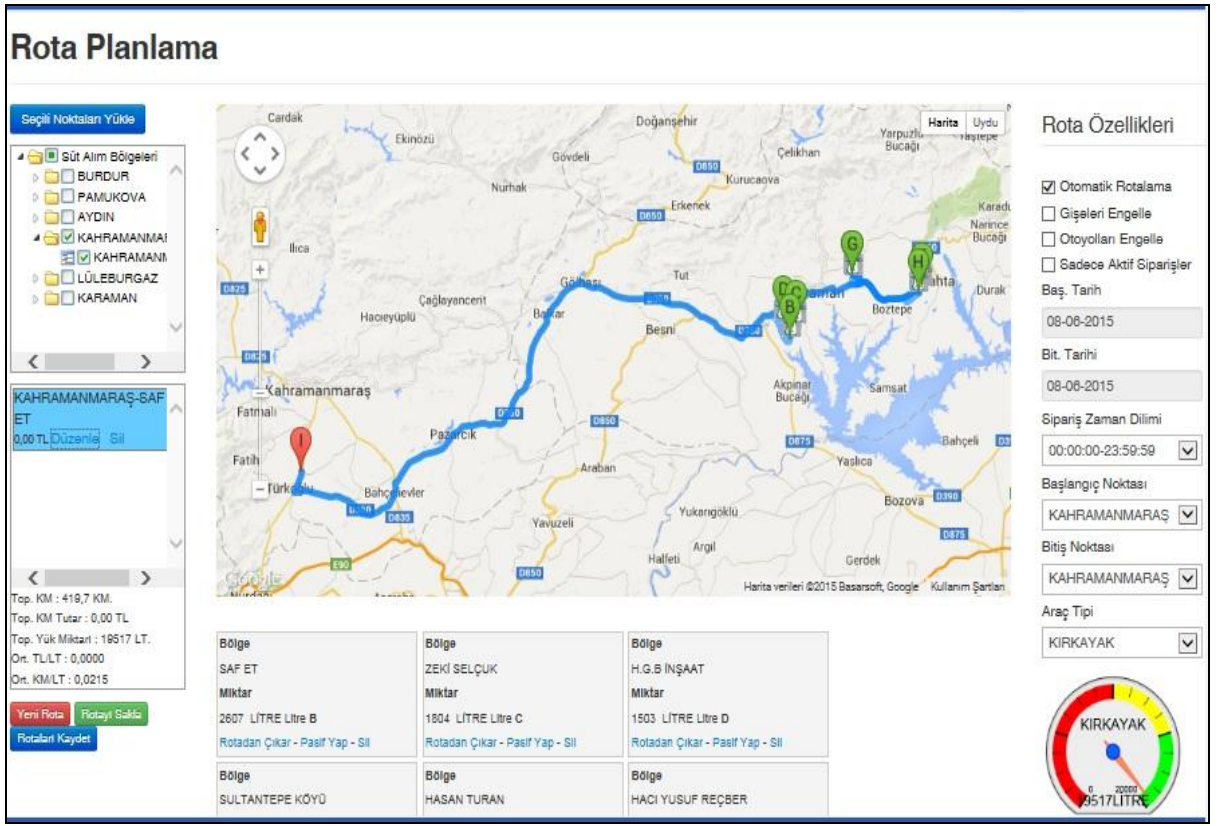
5.6.3 Gams sonuçları

Gams uygulaması AMD Opteron Processor 64 bit- 4 işlemcili, 2.30 Ghz hızında 16 GB Ram kapasiteli Windows Server 2008 R2 işletim sistemli bir server üzerinde farklı günlerde tekrar tekrar çalıştırılmıştır. Toplam çalışma süresi 4 gün ve üzerindedir. Farklı zaman dilimlerinde yapılan gözlemler sonucu rota iyileştirmesi daha fazla ilerlemediği için son çalıştırma da 4. günün sonunda program sonlandırılmış, elde edilen değerler dikkate alınmıştır. Buna göre oluşan toplam maliyet optimum olarak 9293 TL ve yapılan yol mesafesi 5193 km'dir. 14 araçlık bir kullanım söz konusudur ve toplam araç doluluk oranı %95 dir. Çizelge 5.7'de bu değerler ve rotalar hakkında diğer bilgiler detaylı olarak yer almaktadır.

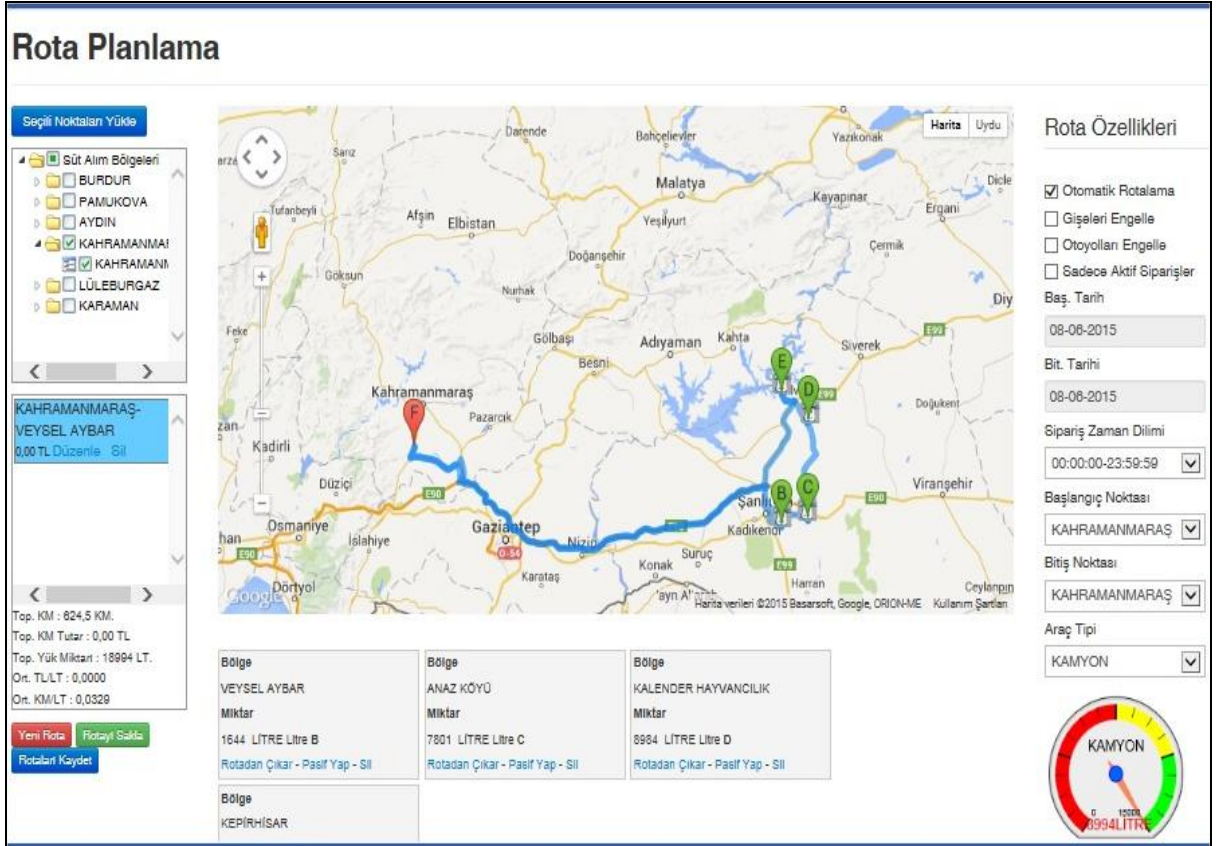
Çizelge 5. 7 Matematiksel Model Rotalama Sonuçları

Araç Tipi	Rota Noktaları									TL Rota Maliyet	Rota KM	LT Araç Doluluk	LT Araç Kapasite	Doluluk %
Kırkayak	1	5	56	47	39	60	29	53	1	792,32 TL	450,428	19286	20000	96%
Kırkayak	1	24	35	33	48	1				474,67 TL	223,793	19831	20000	99%
Kırkayak	1	37	12	43	2	59	1			647,72 TL	347,258	18230	20000	91%
Kırkayak	1	25	20	1						459,82 TL	213,2	18299	20000	91%
Kırkayak	1	7	4	32	16	1				1.289 TL	805,023	18709	20000	94%
Kırkayak	1	15	54	10	1					490,88 TL	235,357	19334	20000	97%
Kırkayak	1	49	61	27	52	30	28	22	1	861 TL	499,479	19517	20000	98%
Kırkayak	1	42	17	6	8	44	1			995,17 TL	595,154	19947	20000	100%
Tır	1	57	14	50	3	41	34	1		437,31 TL	133,929	23856	25000	95%
Kamyon	1	11	26	21	1					509,34 TL	292,493	14376	15000	96%
Kamyon	1	31	18	45	13	1				238,25 TL	86,186	12544	15000	84%
Kamyon	1	38	23	46	55	1				779,47 TL	498,075	14927	15000	100%
Kamyon	1	51	19	1						414,44 TL	220,276	14849	15000	99%
Kamyon	1	58	9	36	40	1				903,42 TL	592,402	18994	20000	95%

GAMS ile problemin matematiksel çözümü sonucu oluşan bazı rotaların harita sonuçları şekil 5.8 ve 5.9'da gösterilmektedir. Matematiksel modelin diğer tüm görselleri proje ekinde verilmiştir.



Şekil 5. 8 GAMS rotalama sonucu oluşan örnek rota gösterimi 1



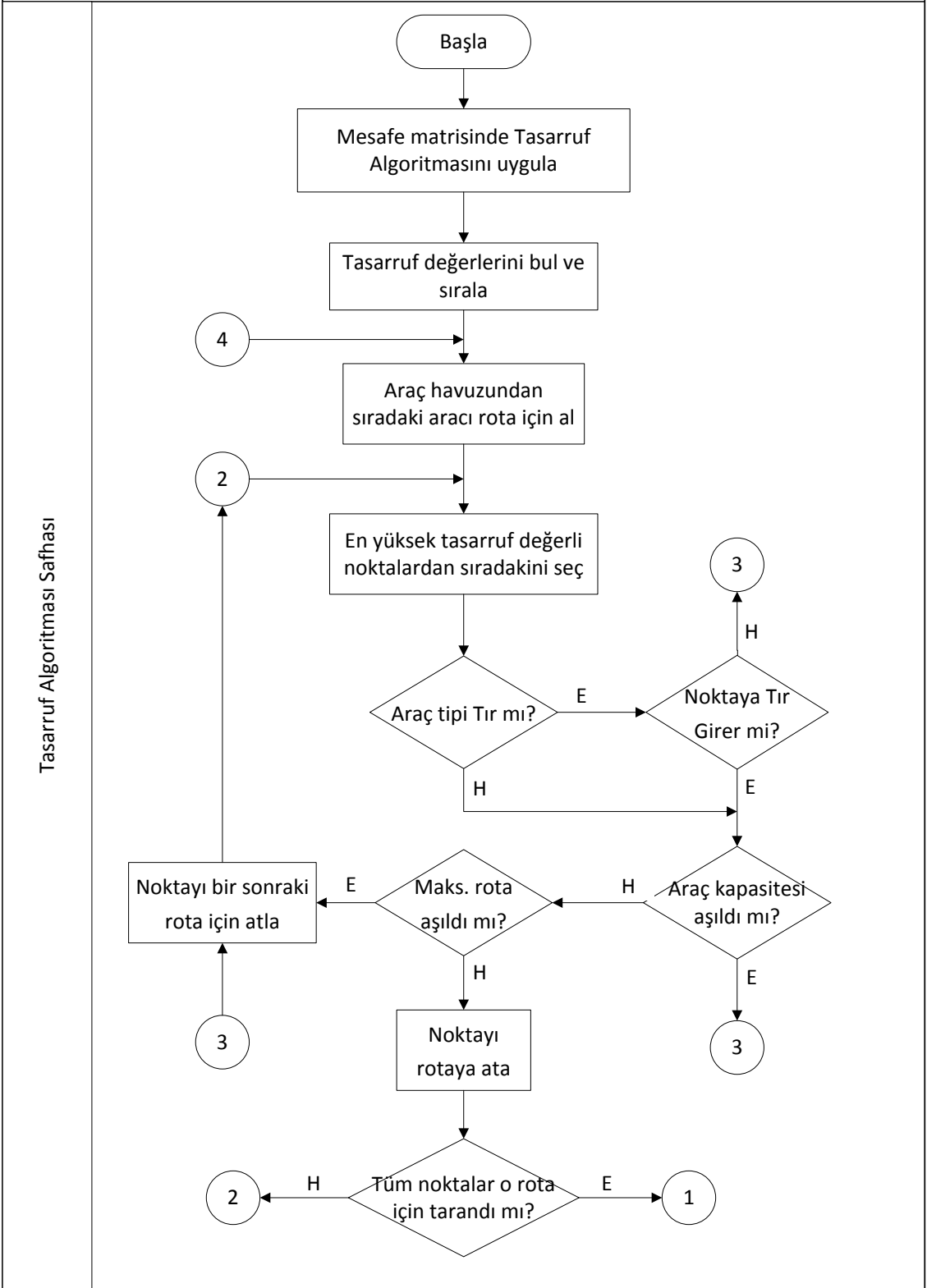
Şekil 5. 9 GAMS rotalama sonucu oluşan örnek rota gösterimi 2

5.7 Problemin Hibrit Model ile Çözümü, Uygun Rotaların Belirlenmesi

Kısıtlar dikkate alınarak ve mevcut durumdaki veriler kullanılarak geliştirilen programda Tasarruf Algoritması ile rotalar oluşturulmuş, sonrasında Genetik Algoritma ile bu rotalar iyileştirilmiştir. Hibrit modelin yapısını özetleyen iş akış diyagramı şekil 5.10 ve 5.11’de yer almaktadır.

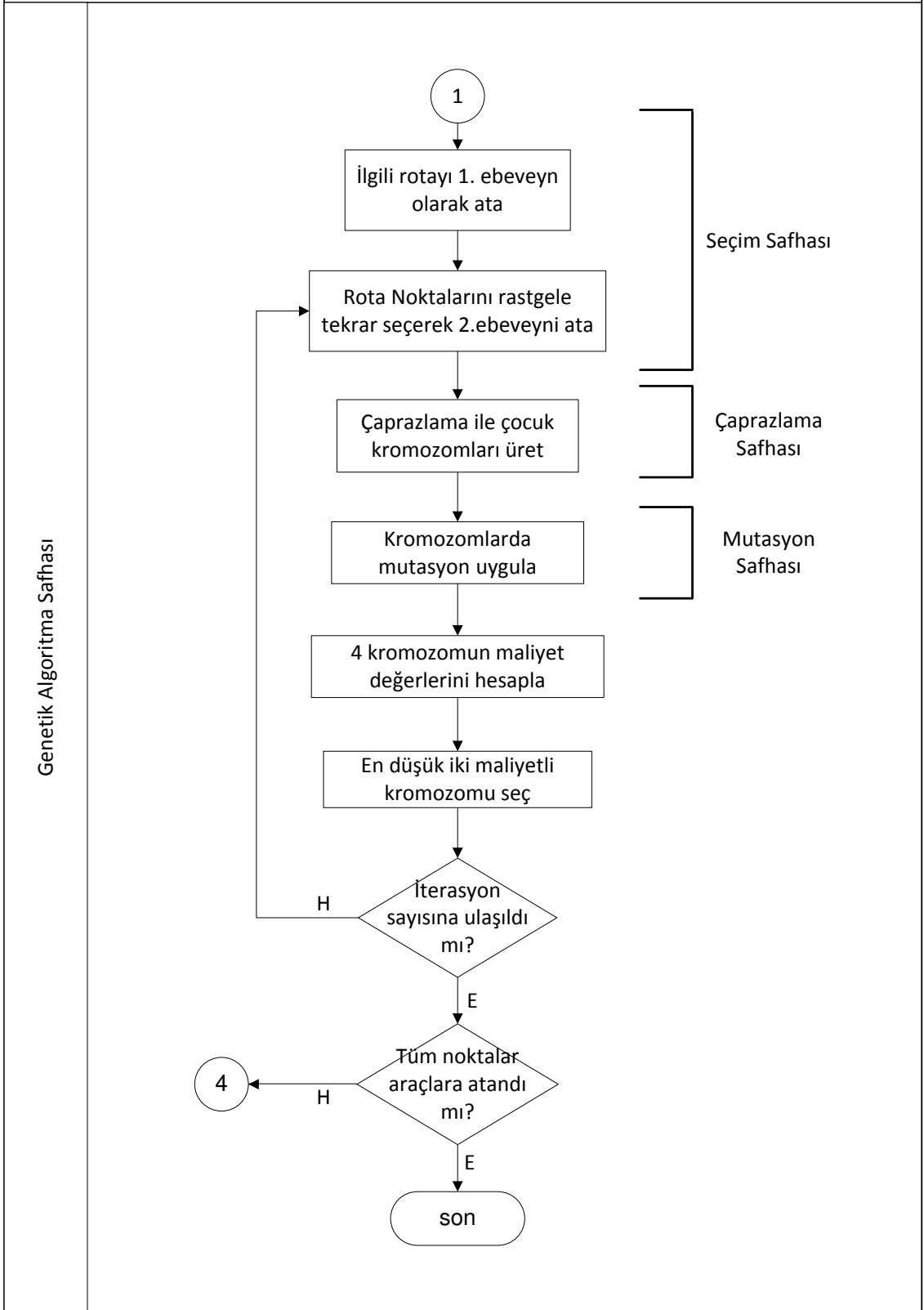
5.7.1 Hibrit Modelin İş Akış Diyagramı

Şekil 5. 10' da hibrit modelin ilk safhası olan Tasarruf Algoritmasının akışı gösterilmektedir.



Şekil 5. 10 Programın iş akış şeması Tasarruf Model kısmı

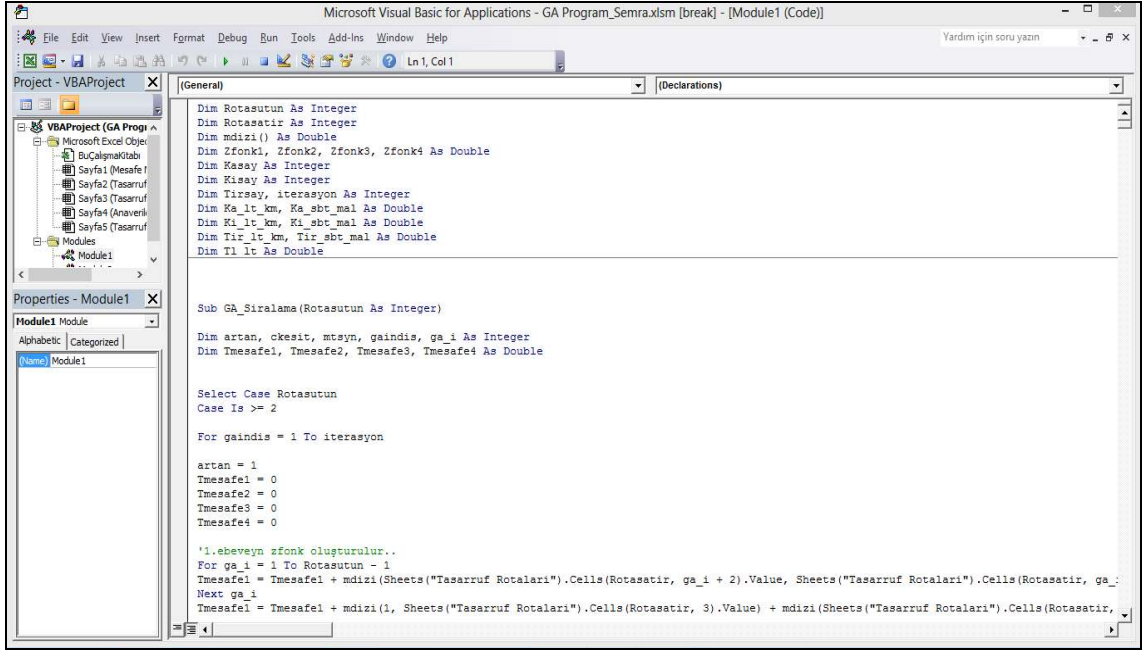
Şekil 5. 11' de ise hibrit modelin ikinci safhası olan Genetik Algoritmanın akışı gösterilmektedir.



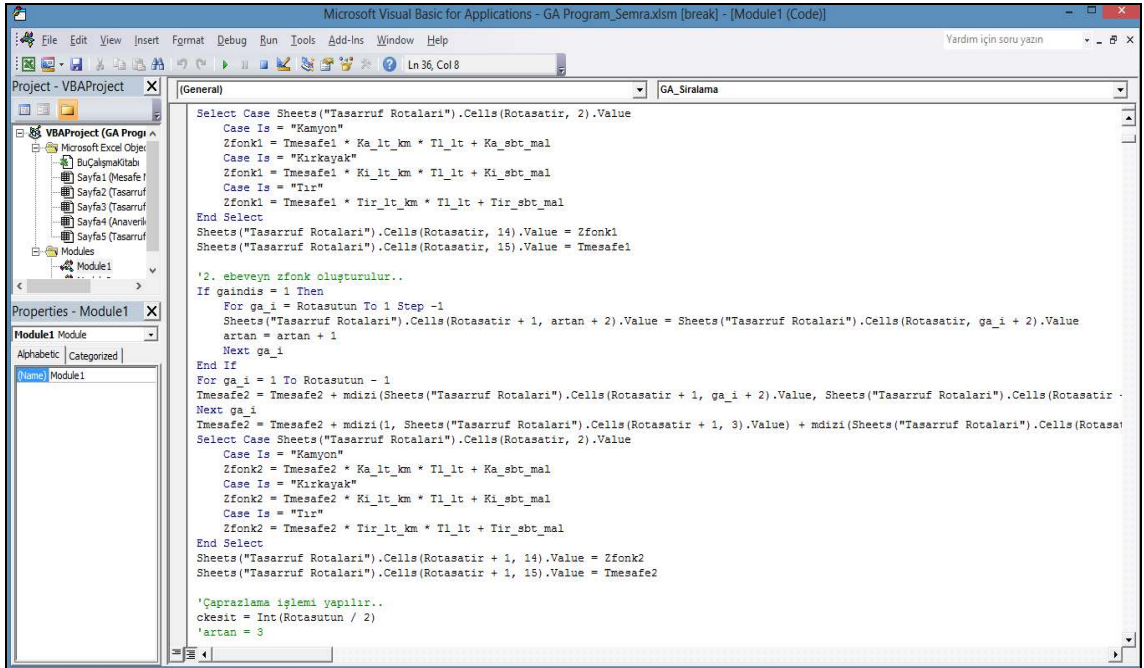
Şekil 5. 11 Programın iş akış şeması Genetik Algoritma kısmı

5.7.2 Hibrit Programlama

GA ve TA'dan oluşan hibrit model ile çözümlenerek maliyeti minimize eden sonuç elde edilmiştir. Bu sonucun elde edilmesinde kullanılan Makro yazılımın örnek ekran görüntülerine aşağıda şekil 5.12, 5.13 ve 5.14'de yer verilmiştir. Programın dosya hali proje ektindedir.



Şekil 5. 12 Hibrit programın makro kod görüntüsü 1



Şekil 5. 13 Hibrit programın makro kod görüntüsü 2

F1		f2		Rota Bilgi											
A	B	C	D	E	F	G	H	I	J	K	L				
Tasarruf Değeri	Konum i	Konum j	Miktar i	Miktar j	Rota Bilgi	Araç Kısıt i	Araç Kısıt j								
2	709,66	4	16	3750	2880 Rota 1	2	3								
3	691,408	16	32	2880	2877 Rota 1	3	3								
4	677,666	4	32	3750	2877 YANLIŞ	2	3								
5	526,206	6	17	1500	1600 Rota 1	2	3								
6	499,407	40	36	2565	2984 Rota 1	2	2								
7	492,93	6	8	1500	3000 Rota 1	2	2								
8	492,93	17	8	1600	3000 YANLIŞ	3	2								
9	492,478	9	4	2450	3750 Rota 1	2	2								
10	492,478	9	16	2450	2880 YANLIŞ	2	3								
11	492,478	9	32	2450	2877 YANLIŞ	2	3								
12	482,551	36	16	2984	2880 Rota 1	2	3								
13	482,551	36	32	2984	2877 YANLIŞ	2	3								
14	482,551	36	4	2984	3750 YANLIŞ	2	2								
15	474,737	6	4	1500	3750 Rota 1	2	2								
16	474,737	6	16	1500	2880 YANLIŞ	2	3								
17	474,737	6	32	1500	2877 YANLIŞ	2	3								
18	474,459	8	58	3000	1644 YANLIŞ	2	3								
19	474,372	6	58	1500	1644 YANLIŞ	2	3								
20	473,394	40	4	2565	3750 YANLIŞ	2	2								
21	473,394	40	16	2565	2880 YANLIŞ	2	3								
22	473,394	40	32	2565	2877 YANLIŞ	2	3								
23	471,22	6	9	1500	2450 YANLIŞ	2	2								
24	470,625	58	4	1644	3750 YANLIŞ	3	2								
25	470,625	58	9	1644	2450 YANLIŞ	3	2								

Şekil 5. 14 Hibrit programın makro kod görüntüsü 3

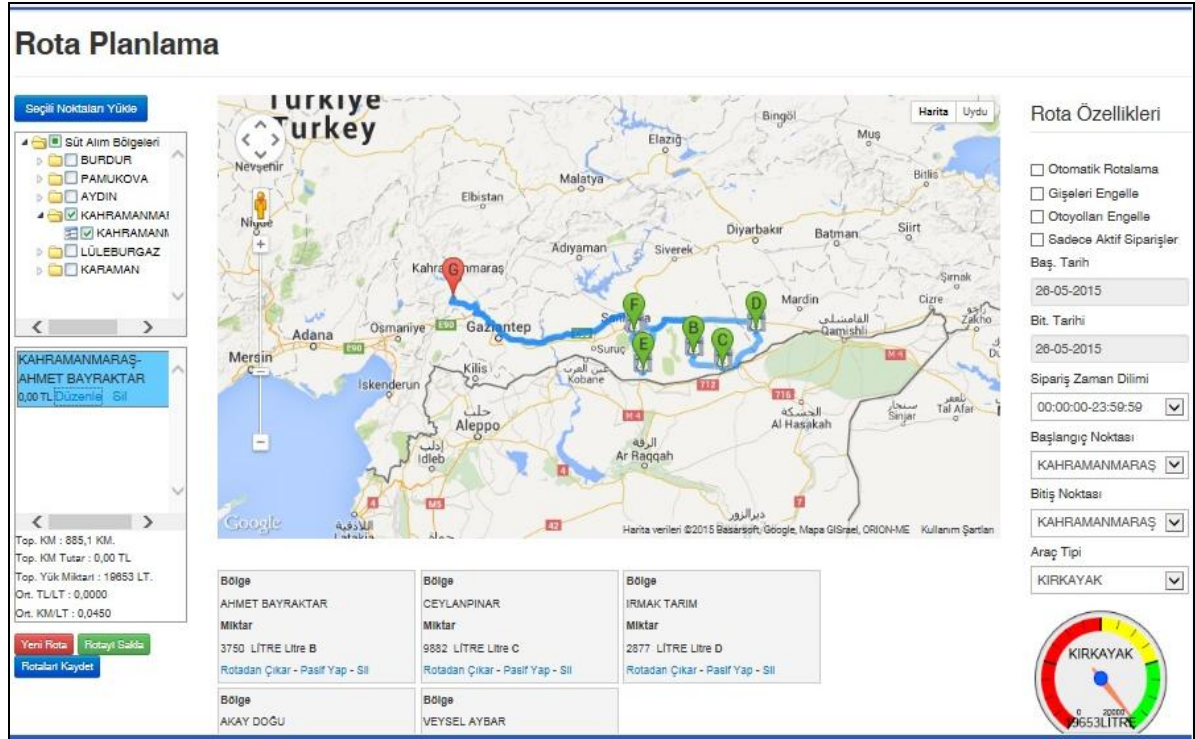
5.7.3 Hibrit Programlama Sonuçları

Meta sezgisel hibrit model ile optimum rotaların belirlendiği ve bu rotalara uygun araç tipinin atandığı çizelge aşağıda gösterilmektedir. Toplam çalışma süresi 5 dakika sürmektedir. Her bir rota için GA iterasyon sayısı 100 olarak alınmıştır. Bu bağlamda çıkan sonuçlara göre toplam maliyet 9698 TL olarak hesaplanmıştır. Yapılan yol mesafesi 5268 km'dir. 14 araçlık bir kullanım söz konusudur ve toplam araç doluluk oranı %94'dür. Çizelge 5.8'de bu değerler ve rotalar hakkında diğer bilgiler detaylı olarak yer almaktadır.

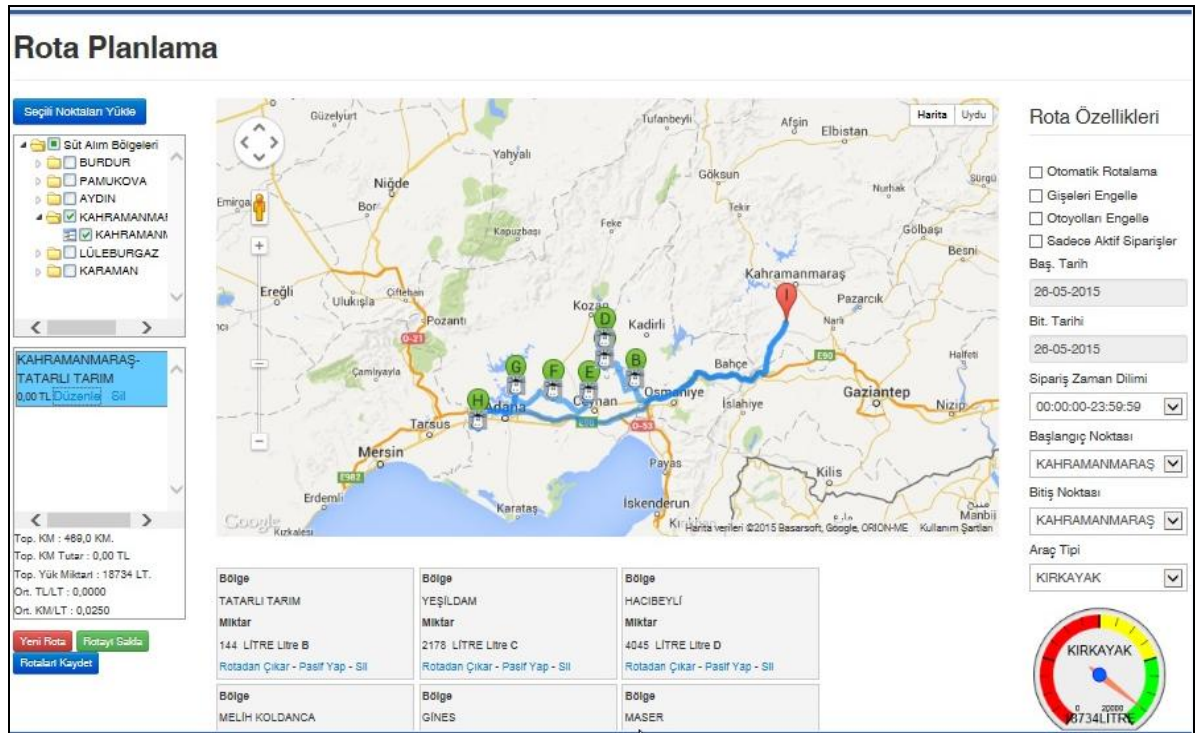
Çizelge 5. 8 Hibrit Model Rotalama Sonuçları

Araç Tipi	Rota Noktaları								TL Rota Maliyet	Rota KM	LT Araç Doluluk	LT Araç Kapasite	Doluluk %
Kırkayak	1	4	16	32	6	58	1		1.358,89 TL	854,863	19653	20000	98%
Kırkayak	1	9	36	40	44	1			988,34 TL	590,485	19037	20000	95%
Kırkayak	1	17	8	22	61	1			1.083,59 TL	658,446	19819	20000	99%
Kırkayak	1	7	25	49	27	30	28	52	945,16 TL	559,676	19785	20000	99%
Kırkayak	1	56	60	29	47	23	55	46	914 TL	537,422	18734	20000	94%
Kırkayak	1	38	53	39	5	48	41	57	749,73 TL	420,245	19480	20000	97%
Kırkayak	1	26	54	42	37	12	59	43	654 TL	352,059	19831	20000	99%
Kırkayak	1	33	51	10	1				479,29 TL	227,295	17872	20000	89%
Kırkayak	1	21	11	15	1				527,35 TL	261,581	19380	20000	97%
Tır	1	35	3	14	50	1			560,73 TL	214,343	23387	25000	94%
Tır	1	20	34	1					596,49 TL	237,669	22774	25000	91%
Kamyon	1	2	19	1					422,85 TL	226,673	14393	15000	96%
Kamyon	1	31	18	13	45	1			241,23 TL	88,455	12544	15000	84%
Kamyon	1	24	1						176,73 TL	39,37	6010	15000	40%

Geliştirilen metasezgisel hibrit model ile problemin çözümü sonucu oluşan bazı rotaların harita sonuçları şekil 5.15 ve 5.16’da gösterilmektedir. Hibrit modelin diğer tüm görselleri EK-B’de yer almaktadır.



Şekil 5. 15 Metasezgisel hibrit rotalama sonucu oluşan örnek rota gösterimi 1



Şekil 5. 16 Metasezgisel hibrit rotalama sonucu oluşan örnek rota gösterimi 2

5.7.4 Hibrit Modelin Diğer Sistemler ile Karşılaştırılması

Aşağıdaki Çizelge 5.9'da görüldüğü üzere maliyet anlamında en düşük, doluluk anlamında en uygun çözüm matematiksel model çözümüdür. Fakat çözümlemesi Gams uygulaması ile 4 gün ve üzeri sürmektedir. Operasyonun ihtiyacına göre süt toplamak için araçlar günlük olarak yola çıkmaktadır. Hali ile rota planlamalarının gün içerisinde kısa bir sürede tamamlanması gerekmektedir. Bu bakımdan matematiksel model, problemin çözümü açısından uygun değildir.

Mevcut sistemde kullanıcılar kendi imkanları ile bilgisayar başında aynı işlemleri yaptıklarında rotalama süresi 1 saati bulmaktadır ve matematiksel model sonuçlarına göre çok daha verimsiz, maliyeti yüksek rotalamalar ortaya çıkmaktadır. Araç doluluk olarak yakın değerler elde edilse de, yapılan toplam kilometrenin paralelinde maliyeti yüksek iş çıkıyor durumdadır. Aşağıdaki listeye göre manuel rotalama sonucu oluşan toplam navlun değeri 10942 TL'dir ve bu rakamda matematiksel modele göre %18 daha fazla durumdadır.

Geliştirmiş olduğumuz Hibrit modelin rotalama süresi 5 dakika civarındadır. Buna karşın oluşan navlun değeri 9698 TL, alınan mesafe ise 5268 km'dir. Bu rakamlara göre manuel sistem karşısında fiyat olarak %13, mesafe olarak ise %16'lık bir iyileştirme elde edilmiştir. Matematiksel modele göre ise fiyat olarak %4, mesafe olarak ise %1'lık bir fark ile değerler minimize edilmiştir.

Yola çıkan araç sayıları her 3 modelde aynı olmasına karşın zaman, araç dolulukları ve oluşan maliyete göre modelimiz, diğerlerine göre avantajlı durumdadır.

Çizelge 5. 9 Hibrit Modelin diğer sonuçlar ile karşılaştırılması

	TL Toplam Maliyet	Toplam KM	Araç Doluluk	Araç Sayısı	İşlem Süresi
HİBRİT SİSTEM	9.698 TL	5268	92%	14	5 dak
MATEMATİKSEL MODEL	9.293 TL	5193	95%	14	4 gün üzeri
MANUEL SİSTEM	10.942 TL	6132	92%	14	1 saat

SONUÇ VE ÖNERİLER

Bu çalışmada heterojen araç filosuna sahip bir lojistik firmasının, süt ve süt ürünleri üreten bir firma için belirli çiftliklerden gelen siparişler doğrultusunda süt toplaması ile ilgili sorunlar ele alındı. Bu sorunlar doğrultusunda maliyeti optimum seviyeye düşüren uygun rotaların belirlenmesi sağlanmış oldu. Problemin çözümünde öncelikle mevcut operasyon incelenmiş, farklı araç tipleri, kapasiteleri, noktalara araçların girip girmemesi gibi kısıtlar ile firmanın süt toplayacağı noktaların konumları, bu noktaların birbirlerine olan uzaklıkları, noktalardan toplanacak sütün miktarı gibi veriler göz önünde bulundurulmuştur. Firmanın karşılaştığı iki temel sorun olan uygun rotalama ve maliyet faktörlerinin üzerine gidilmiştir ve bir öneri sistemi geliştirilmiştir. Bu çerçevede geliştirilen metasezgisel hibrit modelin ilk aşaması olan Tasarruf Algoritması ile rotalar oluşturulmuş, ikinci aşama olan Genetik Algoritma sayesinde ise bu rotaların noktaları arasındaki sıralamada iyileştirme yapılmıştır.

Elde edilen çözüm sonuçları, zaman, kaynak, maliyet açısından mevcut durum ile karşılaştırılmış, firma üst yönetimine sunum yapılmıştır. Uygulamanın pilot çalışmadan çıkıp genele hitap edecek şekilde geliştirilmesi beklenilmektedir. İlk etapta, firma içerisinde özel olarak geliştirilen, kullanıcıların siparişleri harita üzerinden gördüğü ve anlık planladığı dinamik rotalama uygulamasına, kullanıcı için bir alternatif öneri sistemi olarak eklenilmesi istenilmiştir. Çalışmalar bu çerçevede yürütülecektir.

Araç rotalama problemleri, dünyada ve paralelde lojistiğin gelişmekte olduğu Ülkemiz’de odaklanılması gereken en önemli maliyet kalemlerinden birisi olduğunu düşünüyorum. Rekabet şartlarında varlıklarını devam ettiren firmalar bir yandan yeni iş

fırsatları ararken, diğ er yandan i  dinamiklerinde olu an gereksiz giderleri tespit edip d    rmek zorundalar. B  ylelikle firmada karlılı la birlikte m    terilerine kar  ı maliyet, zaman avantajı ve kaliteli hizmet ortamı saėlanmı  olur.  zellikle Daėıtım/Toplama yapan firmalarda ara  rotalama problemi bu  er evede ele alınmı , konu hakkında akademik anlamda pek  ok   z m  retilmi tir. Tezimizde bu   z mlerden ikisi alınıp hibrit bir alternatif model olu turulmu tur. Firmaların ihtiya larına g re bu tarz  alı maların geni letilerek devam etmesi  irketlere, dolayısı ile   lke ekonomisine  ok b  y k bir katkı saėlayacaktır.

Geli tirilen hibrit model ile, mevcut sistem ve matematiksel model sonu ları kar  ıla tırılmı tır. Gelecekte akademik anlamda diğ er alternatif   z mler tekil veya hibrit olarak adapte edilip elde edilecek optimum   z mler daha geni   er evede kıyaslanarak deėerlendirilebilir. B  ylelikle bu problem i in daha uygun   z mler aranmı  olacaktır.

KAYNAKLAR

- [1] Tan, K., (2000). "A Framework of Supply Chain Management Literature".
- [2] Ballou, R., (1999). "Business Logistics Management: Planning, Organizing and Controlling The Supply Chain, McGraw-Hill".
- [3] Simchi Levi, D., ve Brammel, J., (1997). "Logic of Logistics: Theory, Algorithms & Applications for Logistics Management", New York: Springer- Verlag.
- [4] Nabiyev, V., (2004). "Yapay Zeka - İnsan Bilgisayar Etkileşimi", Ankara: Seçkin.
- [5] Rushton, A., Croucher, P., ve Baker, P., (2006). "Handbook of Logistics and Distribution Management 3rd Edition", Kogan Page Limited.
- [6] Ünel, Ö. H., (2003). "Toplam Kalite Anlayışında Lojistik Yönetimi", Marmara Sosyal Bilimler Enstitüsü, İstanbul, 32-60.
- [7] Işık, F., ve Güner, E., (2003) "Lojistik Sistemlerde Yer Alan Ulaştırma Hizmetinde Karşılaşılan Problemler ve Bir Model İncelemesi", 5:8-35, Gazi Üniversitesi Fen Bilimler Enstitüsü, Ankara.
- [8] Lojistik, <http://www.igeme.org.tr/tur/pratik/lojistik.pdf> , 11 Aralık 2013
- [9] Laporte, G., Nobert, Y., ve Taillefer, S., (1987). "Ecole des Haltes Etudes Commerciales de Montreal".
- [10] Toth, P., ve Vigo, D., (2002). "The Vehicle Routing Problem", Siam: Bologna.
- [11] Tüfekçiler, H., (2008). "İki Amaçlı Açık Araç Rotalama Problemi İçin Bir Çözüm Yaklaşımı", Osmangazi Üniversitesi, Eskişehir.
- [12] Yu, B., Yang, Z., ve Yao, B., (2009). "An Improved Ant Colony Optimization for Vehicle Routing Problem", European Journal of Operational Research, 196:171–176.
- [13] Hasle, G., Lie, K., ve Quak, E., (2007). "Geometric Modelling, Numerical Simulation and Optimization: Applied Mathematics at SINTEF", Oslo Tallin.
- [14] Tan, K., (2000). "A Framework of Supply Chain Management Literature", European Journal of Purchasing Supply Management.

- [15] Eryavuz, M., (2001). "Araç Rotalama Problemi ve Örnek Bir Uygulama", Gazi Üniversitesi, Fen Bilimleri Enstitüsü Endüstri Mühendisliği Ana Bilim Dalı, Ankara.
- [16] Şeker, Ş., (2007). "Araç Rotalama Problemleri ve Zaman Pencereli Stokastik Araç Rotalama Problemine Genetik Algoritma Yaklaşımı", Yıldız Teknik Üniversitesi.
- [17] Golden, B. L., Raghavan S., ve Wasil E. A., (2005). "The Next Wave in Computing, Optimization, and Decision Technologies".
- [18] Reeves, C. R. J., (2002). "Genetic Algorithms: Principles and Perspectives: A Guide to GA Theory", Norwell: Kluwer Academic Publishers, 56-63.
- [19] Erel, R., (1995). "Taşıt Rotalaması ve Çizelgelemesi: Otobüsle Kentler arası Yolcu Taşımacılığı İçin Bir Model", Yıldız Teknik Üniversitesi, İnşaat Mühendisliği Bölümü, İstanbul.
- [20] Dal Sınır Algoritması, <http://neo.lcc.uma.es/radi-aeb/WebVRP/>, 15 Aralık 2013
- [21] Temel, C., (1989). "Dal-Sınır Algoritması ile Tesis Yerleşim Düzenlenmesi: Bilgisayar Destekli Bir Uygulama", Gazi Üniversitesi.
- [22] Hachtel, G. D., ve Somenzi, F., (1996). "Logic Synthesis and Verification Algorithms", Kluwer Academic Publishers, Boston.
- [23] Dal ve Kesme Algoritması, http://en.wikipedia.org/wiki/Branch_and_cut, 15 Aralık 2013
- [24] Korkmaz, M., (2004). "Küme Örtüleme Modeli Kullanılarak Optimum Yangın Gözetleme Noktalarının Belirlenmesi", 37-49.
- [25] Bali Ö., ve Gencer, C., (2002). " K. K. Lojistik Komutanlığı 3ncü, 4ncü, 5nci Kademe Depolarında Küme Örtüleme Yaklaşımı İle Bir İyileştirme Çalışması", 14-17.
- [26] Demircioğlu, M., (2009). "Araç Rotalama Probleminin sezgisel bir yaklaşım ile çözümlenmesi üzerine bir uygulama", Çukurova Üniversitesi, İşletme Anabilim Dalı, Adana.
- [27] Laporte, G., (1992). "The Vehicle Routing Problem: An overview of exact and approximate algorithms", European Journal of Operational Research, 59:345-358.
- [28] Hill, M., (1999). "Business Logistics Management: Planning, Organizing and Controlling The Supply Chain".
- [29] Christofides, N., Mingozzi, A., ve Toth, P., (1979). "The Vehicle Routing Problem In Combinatorial Optimization", Wiley Chichester.
- [30] Genetik Algoritmalar, <http://www.yapayzeka.org/modules/wiwiwimod/index.php?page=GA&back=WiwiHoe>, 18 Aralık 2013
- [31] Kaynak, O., ve Armağan, G., (2003). "Süreç Denetiminde Yeni bir Yaklaşım: Bulanık Mantık", 74-82.

- [32] Uzman Sistemler, <http://www.yapayzeka.org/modules/wiwiwimod/index.php?page=Expert+Systems&back=WiwiHome>, 18 Aralık 2013
- [33] Öztemel, E., (2003). Yapay Sinir Ağları, Papatya Yayınevi, İstanbul.
- [34] Dorigo, M., Di Caro, G., ve Gambardella, L., (1999). "Ant algorithms for discrete optimization", 5:137-172.
- [35] M. Dorigo ve L. Gambardella, "Ant colonies for the traveling salesman problem", BioSystems, 1997, 43:73-81.
- [36] Dorigo, M., ve Gambardella, L., (1997). "Ant colony system: A cooperative learning approach to the traveling salesman problem".
- [37] Dorigo, M., Maniezzo, V., ve Colorni, A., (1991). "The Ant System: An Autocatalytic Optimizing Process", Technical Report No. 91-016 Revised, Politecnico di Milano, Italy.
- [38] Knox, J., (1994). "Tabu Search Performance on the Symmetric Traveling Salesman Problem", 867-876.
- [39] Cordeau, J. F., Gendreau, M., Laporte, G., Potvin J., ve Semet, F., (2002). "A Guide To Vehicle Routing Heuristics", 53:512-522.
- [40] Araç rotalama problemi uygun yöntemin belirlenmesi, <http://www.neo.lcc.uma.es/radi-aeb/webvrp/index>, 20 Aralık 2014
- [41] Bulanık mantık, http://tr.wikipedia.org/wiki/Bulanik_mantik, 26 Nisan 2014.
- [42] Mano, M., (1997). Sayısal Tasarım, MEB Yayınları, 26.
- [43] Uzman sistemler, http://tr.wikipedia.org/wiki/Uzman_sistemler, 27 Nisan 2014.
- [44] Uzman sistemlerin avantajları, <http://irkilata.wordpress.com/2013/07/20/uzman-sistemlerin-avantajlaridezavantajlari>, 27 Nisan 2014.
- [45] Yapay sinir ağları, http://tr.wikipedia.org/wiki/Yapay_sinir_aglari, 27 Nisan 2014.
- [46] Gritta, R. D., ve Davalos, S., (2002). "Forecasting Small Air Carrier Bankruptcies Using A Neural Network Approach: A Preliminary Analysis", 13:44-49.
- [47] Swinger, K., (1994). "Some Pointers, Pitfalls, and Common Errors, Center for Cognitive and Computational Neuroscience".
- [48] Öztemel, E., (2003). Yapay Sinir Ağları, Papatya Yayınevi, İstanbul.
- [49] Yıldız, B., (2001). "Finansal Başarısızlığın Öngörülmesinde YSA Kullanımı ve Halka Açık Şirketlerde Ampirik Bir Uygulama", 17:51-68.
- [50] Krishaswamy, C. R., Gilbert, E. W., ve Pashley, M. M., (2000). "Neural Network Applications in Finance: A Practical Introduction", 4:78-79.
- [51] Şen, Z., (2004). Yapay sinir ağları ilkeleri, Su Vakfı Yayınları, İstanbul.

- [52] Arslan, O., (2001). "Görüntü Sınıflandırması için Yapay Sinir Ağlarının Analiz ve Optimizasyonu", İTÜ Fen Bilimleri Enstitüsü, İstanbul.
- [53] Bonabeau, E. , ve Theraulaz, G., (2000). "Swarm Smarts",72-79.
- [54] Uğur, A. ve Aydın, D., (2006). "Ant system algoritmasının java ile görselleştirilmesi", Denizli.
- [55] Kalınlı, A., Karaboğa, N., ve Karaboğa, D., (2001). "A modified touring ant colony optimisation algorithm for continuous problems", 437-444.
- [56] Cura, T., (2008). "Doğrusal Olmayan Küresel Optimizasyon Problemleri için Tabu Arama Algoritmasının Kullanılması".
- [57] Monticelli, A. J., Romero, R., ve Asada, E. N., "Fundamentals of Tabu Search,Modern Heuristic Optimization Techniques: Theory and Applications to Power Systems", Wiley-IEEE Press,101-123.
- [58] Temiz, İ., (2010). "Çok Kriterli Permütasyon Akış Tipi Çizelgeleme Problemi için Bir Tavlama Benzetimi Yaklaşımı", Çankaya Üniversitesi Journal of Science and Engineering , 2:141-153.
- [59] Laporte, G., (2000). "Classical And ModernHeuristics For The Vehicle Routing Problem", 7:285-300.
- [60] Breedam, A. V., (2002). "A Parametric Analysis of Heuristics for the Vehicle Routing Problem",137:348-370.
- [61] Cordeau, J. , (2002). "A Guide To Vehicle Routing Heuristics", 53:512-522.
- [62] Şen, Z., (2004). Genetik Algoritmalar ve En İyileme Yöntemleri, Su Vakfı Yayınları,İstanbul.
- [63] Goldberg, D., (1989). "Genetic Algorithms in Search Optimization, and machine Learning, Addison-Wesley".
- [64] Michalewicz, Z., (1996). "Genetic Algorithms + Data Structures = Evaluation Programs", New York: Springer-Verlag.
- [65] Deb, K., (2001). "Multi-objective Optimization Using Evolutionary Algorithms".
- [66] Işık, Y., (2006). "Genetik Algoritma Tabanlı Bulanık Kontrolün Uçuş Kontrol Sistem Tasarımına Uygulanması", Anadolu Üniversitesi, Fen Bilimleri Enstitüsü, Eskişehir.
- [67] Obitko, M., (1998). "Genetic Algortihms", Czech Technical University,Department of Cybernetics, Faculty of Electrical Engineering, Prague.
- [68] Goldberg, D., (1985). "Optimal Initial Population Size for Binary-Coded Genetic Algorithms", University of Alabama, Tuscaloosa.
- [69] Bektaş, T., (2004). "Okul Araç Rotalama Probleminin Tamsayılı Programlama ile Çözümü".
- [70] Gen, M., ve Cheng, R., (1997). "Genetic Algorithms and Engineering Design", John Wiley & Sons, Japan

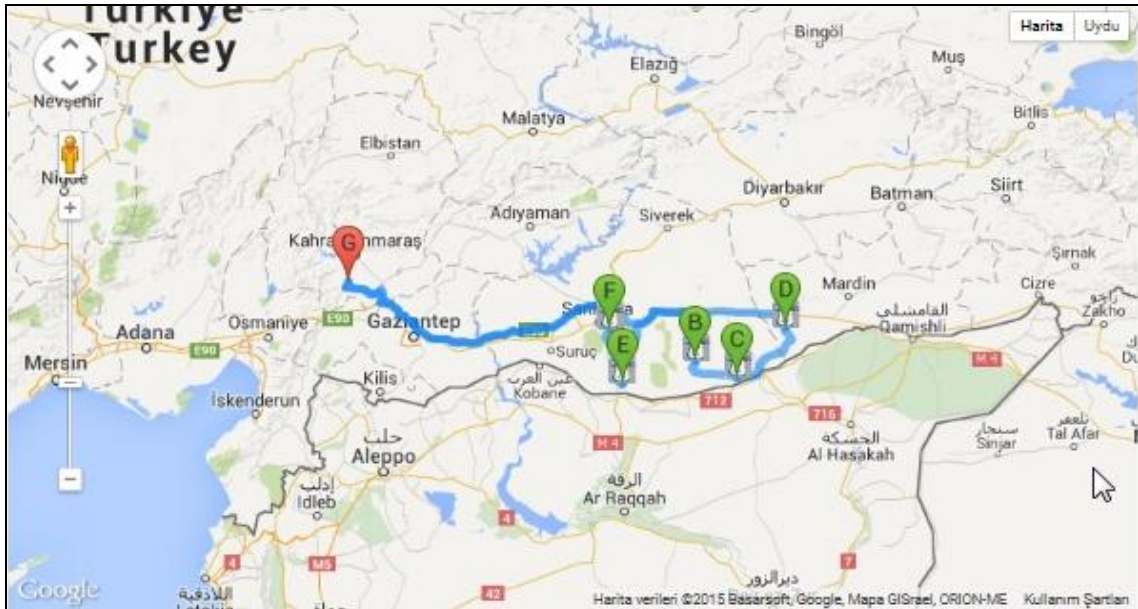
NOKTALAR ARASI MESAFE MASTRİS TABLOSU

Tablo görüntüsü aşağıdaki gibidir. Tüm verileri ihtiva eden düzenlenebilir hali, excel şablonunda proje ekinde yer almaktadır.

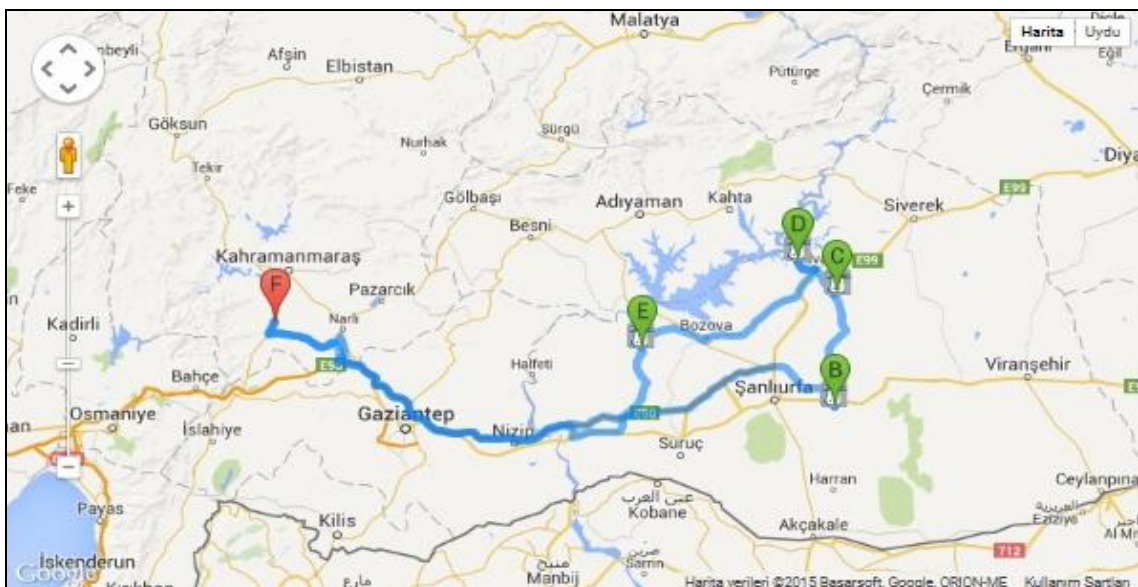
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	1
1	0	58,132	35,759	354,83	101,049	277,479	93,042	257,579	246,239	111,937	98,701	109,153	27,186	32,758	101,349	361,701	268,085	14,423	110,52
2	58,108	0	55,173	302,857	137,017	225,506	41,069	205,606	194,266	60,404	41,552	64,972	64,975	72,948	49,816	309,728	216,112	66,656	58,99
3	35,757	55,247	0	343,463	84,048	266,112	81,675	246,212	234,872	92,681	79,97	97,249	62,825	3,651	82,093	350,334	256,718	50,062	91,26
4	353,505	305,412	361,756	0	422,824	157,588	267,431	145,521	108,607	296,653	266,617	275,02	361,078	358,755	286,566	6,871	158,001	362,759	295,2
5	104,745	136,881	89,529	423,504	0	346,153	161,716	326,253	314,913	180,611	167,375	177,827	131,813	84,709	170,023	430,375	336,759	119,05	179,19
6	276,2	228,107	284,451	157,572	345,519	0	190,126	42,128	52,498	219,348	189,312	197,715	283,773	281,45	209,261	164,443	19,358	285,454	217,93
7	93,044	44,951	101,295	267,5	162,363	190,149	0	170,249	158,909	39,504	13,476	22,529	100,617	98,294	22,056	274,371	147,658	102,298	38,09
8	237,55	189,457	245,801	142,184	306,869	49,214	151,476	0	33,593	180,698	150,662	159,065	245,123	242,8	170,611	149,055	39,82	246,804	179,28
9	244,898	196,805	253,149	108,591	314,217	52,498	158,824	40,431	0	188,046	158,01	166,413	252,471	250,148	177,959	115,462	52,911	254,152	186,63
10	110,963	61,535	93,937	313,047	178,972	235,696	39,392	215,796	204,456	0	52,699	28,686	117,83	97,588	17,964	319,918	226,302	119,511	1,41
11	98,851	50,758	79,741	266,715	168,17	189,364	13,485	169,464	158,124	52,512	0	41,408	106,424	104,101	35,064	273,586	179,97	108,105	51,09
12	109,217	61,124	98,113	275,089	178,536	197,738	22,411	177,838	166,498	28,472	41,446	0	1116,79	114,467	15,228	281,96	155,247	118,471	27,05
13	27,255	66,859	62,932	364,263	128,222	286,912	102,475	267,012	255,672	121,37	108,134	118,586	0	59,931	110,782	371,134	277,518	16,465	119,95
14	32,756	73,554	3,651	360,177	81,047	282,826	98,389	262,926	251,586	96,332	104,048	100,9	59,824	0	85,744	367,048	273,432	47,061	94,91
15	100,251	50,823	83,225	295,61	168,26	218,259	21,955	198,359	187,019	17,175	35,262	16,452	107,118	86,876	0	302,481	208,865	108,799	15,76
16	360,376	312,283	368,627	6,871	429,695	164,459	274,302	152,392	115,478	303,524	273,488	281,891	367,949	365,626	293,437	0	164,872	369,63	302,11
17	266,806	218,713	275,057	157,985	336,125	19,358	147,657	32,734	52,911	176,879	179,918	155,246	274,379	272,056	166,792	164,856	0	276,06	175,46
18	20,7	62,569	56,377	359,973	121,667	282,622	98,185	262,722	251,382	117,08	103,844	114,296	12,763	53,376	106,492	366,844	273,228	0	115,66
19	109,55	60,122	92,524	311,634	177,559	234,283	37,979	214,383	203,043	1,413	51,286	27,273	116,417	96,175	16,551	318,505	224,889	118,098	
20	82,075	60,674	88,816	295,779	172,818	218,428	117,239	198,528	187,188	136,134	122,898	133,35	83,08	108,749	125,546	302,65	209,034	84,761	134,72
21	112,629	64,536	93,519	266,722	181,948	189,371	20,708	169,471	158,131	66,29	13,778	58,038	120,202	117,879	48,842	273,593	179,977	121,883	64,87
22	191,967	170,566	198,708	224,433	282,71	191,276	195,877	171,376	160,036	246,026	162,115	203,466	192,972	218,641	235,438	231,304	181,882	194,653	244,61
23	172,866	205,000	157,65	491,625	81,324	414,774	229,837	394,374	383,034	248,732	235,496	245,948	199,934	152,83	238,144	498,496	404,88	187,171	247,31
Anavertiler Mesafe Masteri Tasarruf Matrisi Tasarruf Sıralama Tasarruf Rotaları																			

HİBRİT MODELİN ROTA GÖRÜNTÜLERİ

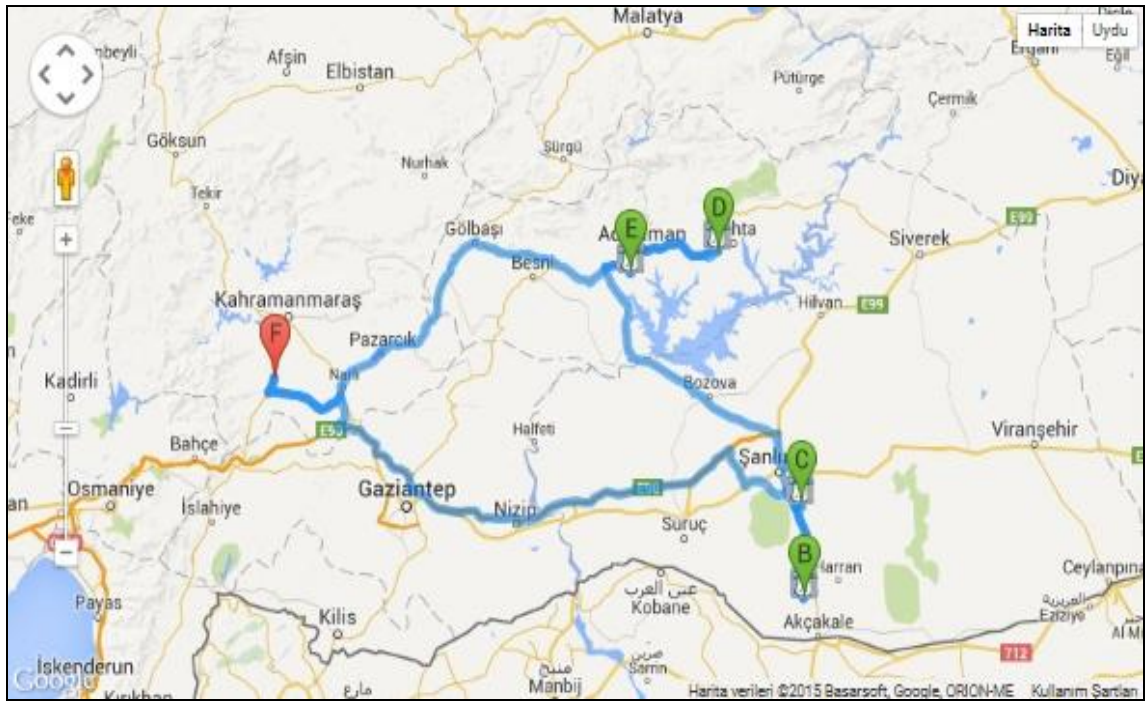
Rota 1:



Rota 2:



Rota 3:



Rota 4:



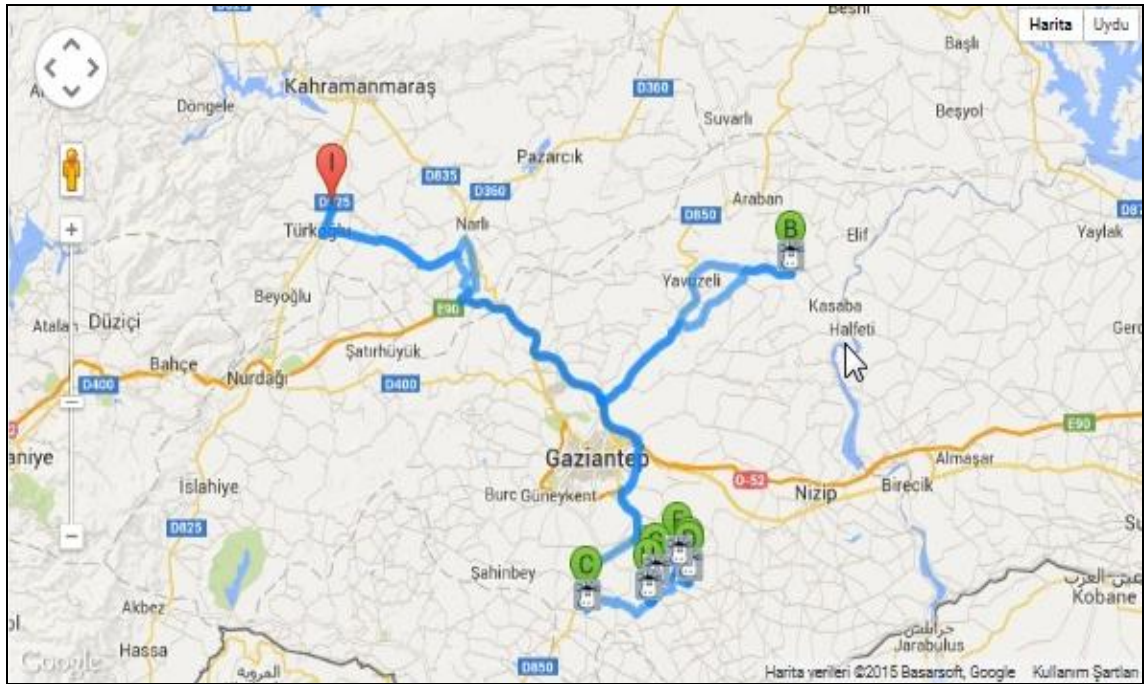
Rota 5:



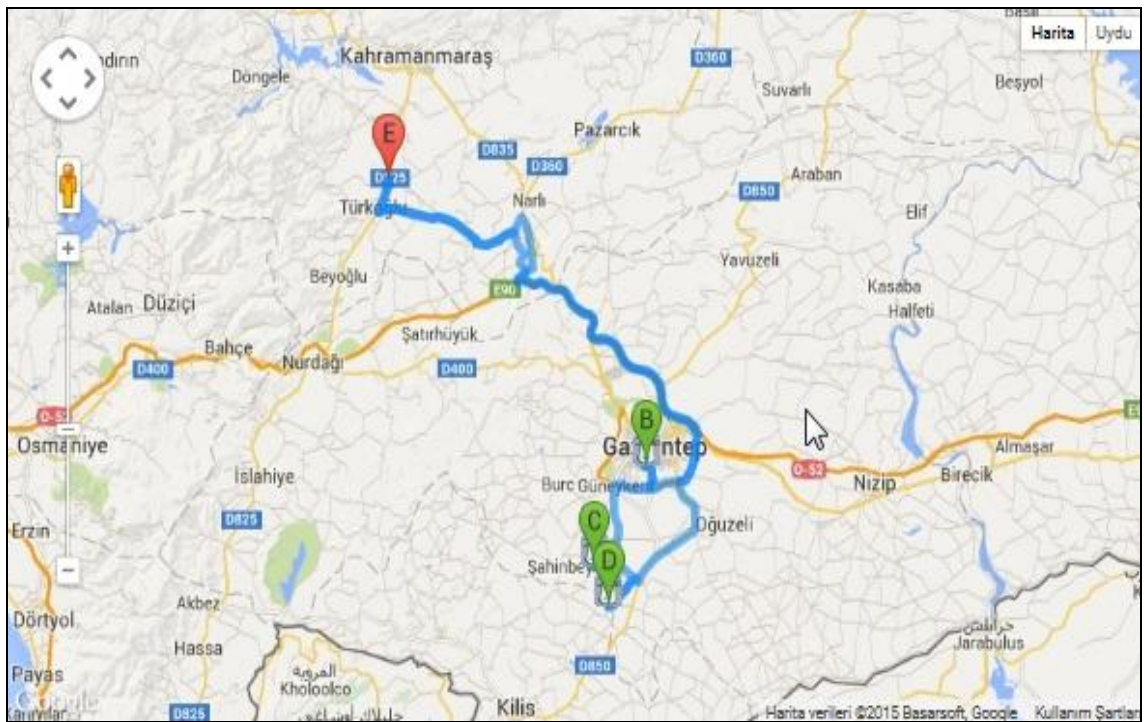
Rota 6:



Rota 7:



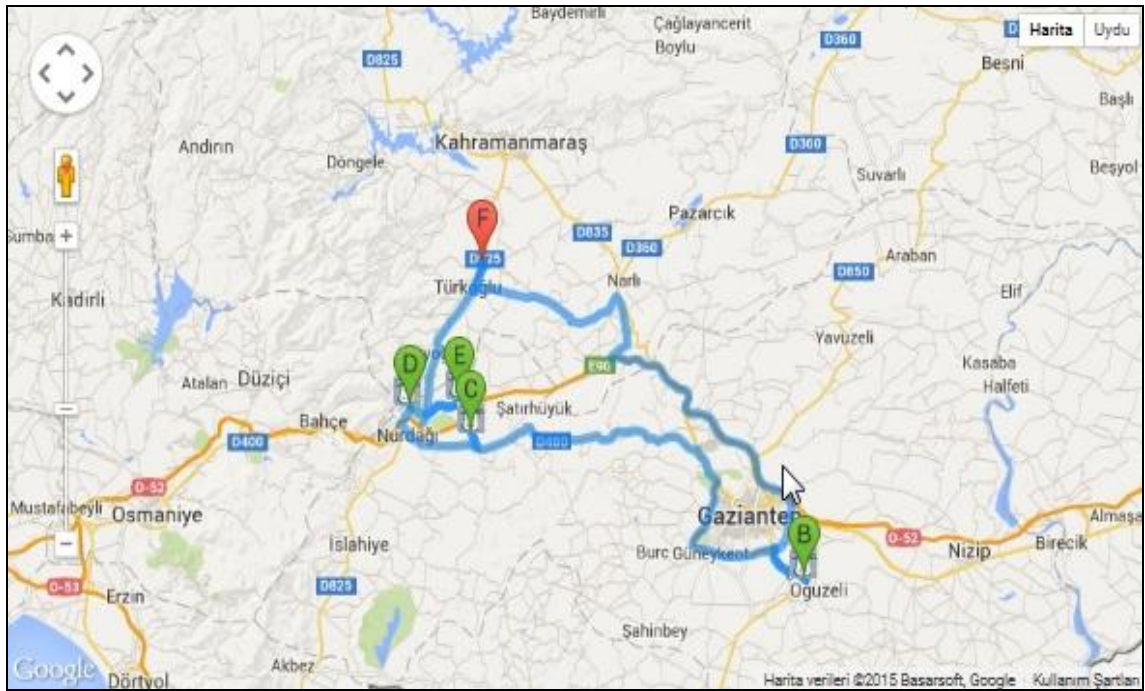
Rota 8:



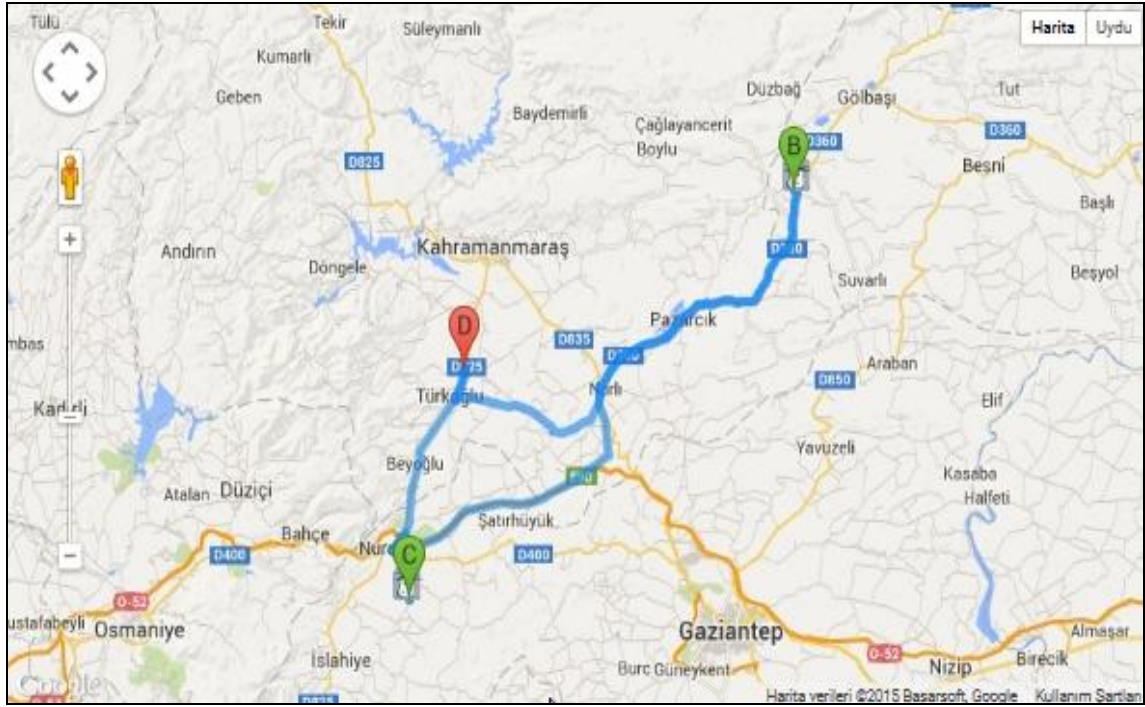
Rota 9:



Rota 10:



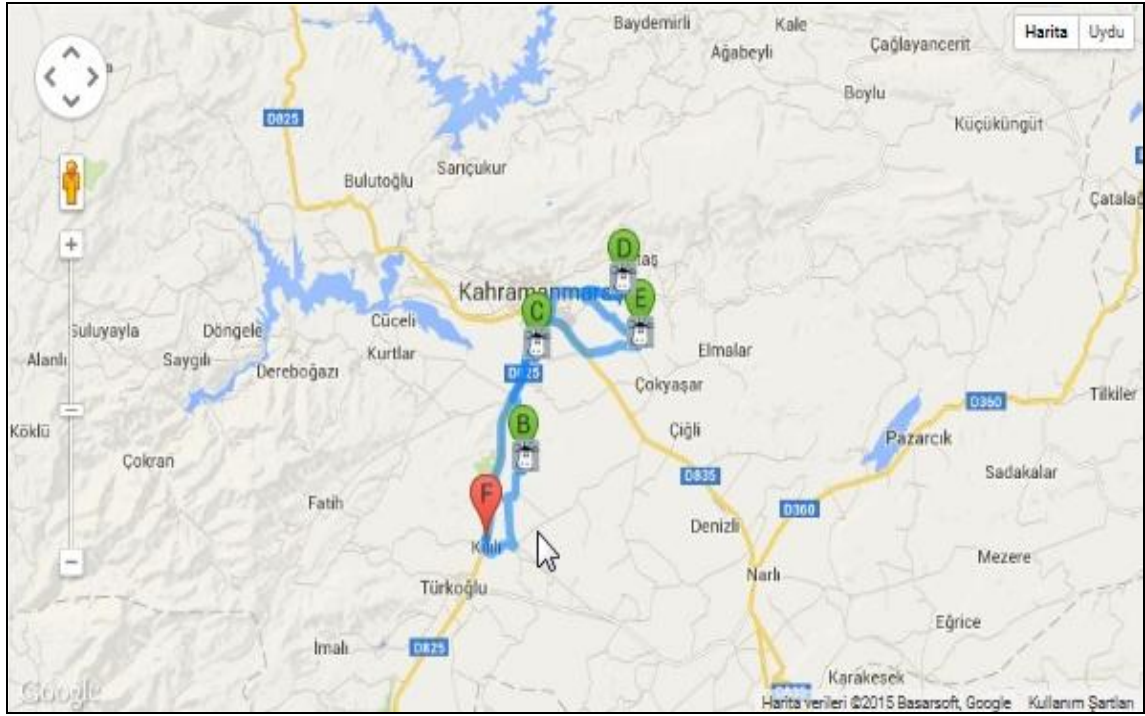
Rota 11:



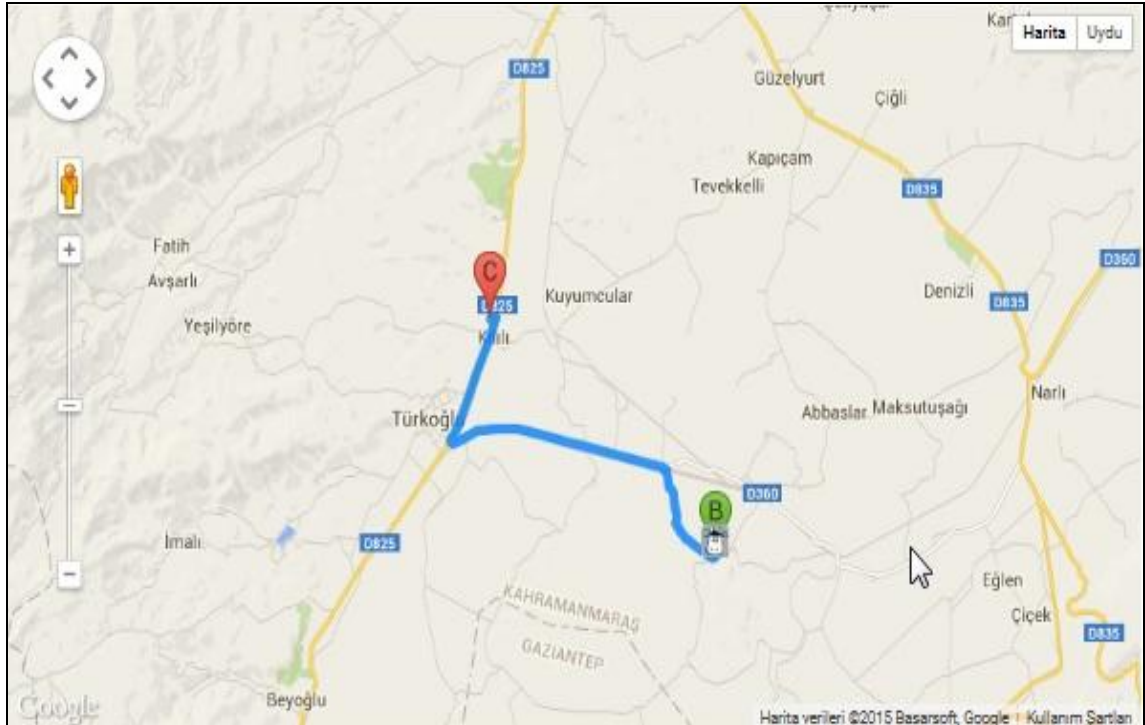
Rota 12:



Rota 13:



Rota 14:



ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Selim EROĞLU
Doğum Tarihi ve Yeri : 25.02.1984 GÜMÜŞHANE
Yabancı Dili : İngilizce
E-posta : eroglu_selim@hotmail.com

ÖĞRENİM DURUMU

Derece	Alan	Okul/Üniversite	Mezuniyet Yılı
Y. Lisans	Endüstri Müh.	Yıldız Teknik Üniv	--
Lisans	Bilgisayar Müh.	Sakarya Üniv	2007
Lisans	Endüstri Müh.	Sakarya Üniv	2006
Lise	Fen Bilimleri	Gümüşhane Lisesi	2001

İŞ TECRÜBESİ

Yıl	Firma/Kurum	Görevi
2006	Netlog Loj. Hiz. A.Ş.	Sistem Geliştirme Müh.
2009	Netlog Loj. Hiz. A.Ş.	Sistem Geliştirme Destek Şefi
2010	Netlog Loj. Hiz. A.Ş.	İş ve Sistem Geliştirme Md. Yrd.
2012	Netlog Loj. Hiz. A.Ş.	İş ve Sistem Geliştirme Md.