

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**TAM SAYILI DOĞRUSAL PROGRAMLAMA İLE
ARAÇ ROTALAMA PROBLEMİ ÇÖZÜMÜ VE BİR
SERVİS AĞINDA UYGULAMASI**

YÜKSEK LİSANS TEZİ

Semiha ERDOĞAN

Enstitü Anabilim Dalı : ENDÜSTRİ MÜHENDİSLİĞİ
Tez Danışmanı : Dr. Öğr. Üyesi
Tuba CANVAR KAHVECİ

Mayıs 2019

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**TAM SAYILI DOĞRULSA PROGRAMLAMA İLE
ARAÇ ROTALAMA PROBLEMİ ÇÖZÜMÜ VE BİR
SERVİS AĞINDA UYGULAMASI**

YÜKSEK LİSANS TEZİ

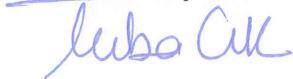
Semiha ERDOĞAN

Enstitü Anabilim Dalı

: ENDÜSTRİ MÜHENDİSLİĞİ

Bu tez ~~14/04~~2019 tarihinde aşağıdaki jüri tarafından oybirliği / oyçokluğu ile kabul edilmiştir.


**Dr. Öğr. Üyesi
Tuba CANVAR KAHVECİ
Jüri Başkanı**



**Doç. Dr.
Özer UYGUN
Üye**



**Dr. Öğr. Üyesi
Fuat ŞİMŞİR
Üye**



BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.



Semiha ERDOĞAN

08.05.2019

TEŐEKKÜR

Tez süreci boyunca deęerli bilgi ve deneyimlerini paylaŐan, desteęini ve yardımlarını esirgemeyen deęerli danıŐman hocam Dr. Öęr. Üyesi Tuba CANVAR KAHVECİ'ye teŐekkürlerimi sunarım.

Birlikte alıŐmaktan mutlu olduęum, her daim anlayıŐ, destek ve iyi dileklerini gördüęüm yöneticim Sayın Muhammed Emin FURKAN'a, bir ekipten daha ok bir aile olduęumuz, tez hazırlama süresince desteklerini esirgemeyen ekip arkadaşlarıma da teŐekkürü bir bor bilirim.

Dostluklarıyla beni bu dünyada ender insanlardan kılan, iyi günümde kötü günümde her daim yanımda olan, ok kıymetli arkadaşlarım Őirin ÖZLEM ve Fatma ARSLAN'a teŐekkür ederim.

Son olarak, en kıymetlilerime, anneme ve babama herŐey için teŐekkür ederim.

İÇİNDEKİLER

TEŞEKKÜR.....	i
İÇİNDEKİLER.....	ii
SİMGELER VE KISALTMALAR LİSTESİ.....	iv
TABLolar LİSTESİ.....	vi
ÖZET.....	vii
SUMMARY.....	viii
BÖLÜM 1.	
GİRİŞ	1
BÖLÜM 2.	
ARAÇ ROTALAMA PROBLEMİ	2
2.1. Araç Rotalama Problemi Tanımı.....	2
2.2. ARP Kısıtları	3
2.3. ARP Uygulama Alanı.....	4
2.4. Problemin Matematiksel Modeli.....	6
2.5. Araç Rotalama Probleminin Bileşenleri	8
BÖLÜM 3.	
ARP ÇEŞİTLERİ	10
3.1. Kısıtlarına göre ARP	11
3.2. Yolların Durumuna göre ARP	13
3.3. Rotalama Durumuna göre ARP.....	13
3.4. Çevre Durumuna göre ARP	13

BÖLÜM 4.	
ARP ÇÖZÜM YÖNTEMLERİ.....	15
4.1. Kesin Yöntemler	16
4.2. Sezgisel Yöntemler	18
4.3. Tam Sayılı Doğrusal Programlama	23
BÖLÜM 5.	
LİTERATÜR TARAMASI	28
BÖLÜM 6.	
UYGULAMA.....	33
6.1. Problemin Tanımı.....	33
6.2. Problemin Varsayımları.....	37
6.3. Matematiksel Model.....	37
BÖLÜM 7.	
ARAŞTIRMA SONUÇLARI.....	41
BÖLÜM 8.	
TARTIŞMA VE SONUÇ	54
KAYNAKLAR	56
EKLER.....	60
ÖZGEÇMİŞ.....	63

SİMGELER VE KISALTMALAR LİSTESİ

ARP	: Araç Rotalama Problemi
BBAOA	: Bakteriyel Besin Arama Optimizasyonu Algoritması
ÇDARP	: Çok Depolu Araç Rotalama Problemi
ÇKZPARP	: Çok Kullanımlı ve Zaman Pencerele Araç Rotalama Problemi
DP	: Doğrusal Programlama
GIS	: Geographical Information System
GPS	: Global Positioning System
GSP	: Gezgin Satıcı Problemi
KKARP	: Kapasite Kısıtlı Araç Rotalama Problemi
KKO	: Karınca Kolonisi Algoritması
MILP	: Mixed Integer Lineer Programming
OSRP	: Okul Servisi Rotalama Problemi
PARP	: Periyodik Araç Rotalama Problemi
SARP	: Stokastik Araç Rotalama Problemi
SZPARP	: Sıkı Zaman Pencerele Araç Rotalama Problemi
TB	: Tavlama Benzetimi
TSP	: Travelling Salesman Problem
ZPARP	: Zaman Pencerele Araç Rotalama Problemi

ŞEKİLLER LİSTESİ

Şekil 2.1. Araç rotalama problemi.....	3
Şekil 3.1. ARP çeşitleri.....	10
Şekil 4.1. ARP çözüm yöntemleri.....	15
Şekil 4.2. Tabu arama Algoritması (Akca, 2015)	21
Şekil 4.3. Genetik algoritmanın işlem basamakları (Çalışkan, Yüksel, & Dayık, 2016)	22
Şekil 4.4. Karınca davranışları.....	23
Şekil 6.1. Durakların harita görünümü.....	35
Şekil 7.1. LINGO 18.0 çözüm sonucu	42
Şekil 7.2. Araç rotalarının haritada görünümü.....	43
Şekil 7.3. LINGO 18.0 çözüm sonucu	44
Şekil 7.4. Araç rotalarının haritada görünümü.....	45
Şekil 7.5. LINGO 18.0 çözüm sonucu	46
Şekil 7.6. Araç rotalarının haritada görünümü.....	47
Şekil 7.7. LINGO 18.0 çözüm sonucu	48
Şekil 7.8. Araç rotalarının haritada görünümü.....	49
Şekil 7.9. Lingo Çözümü	50
Şekil 7.10. Harita Görünümü	51

TABLULAR LİSTESİ

Tablo 6.1. Probleme ait veriler.....	34
Tablo 6.2. Duraklar ve personel sayıları	35
Tablo 6.3. Uzaklık matrisi.....	36
Tablo 8.1. Toplam mesafe ve toplam maliyet.....	52
Tablo 8.2. Personelin servise bindikten sonra gittiği mesafeler (km).....	53
Tablo 8.3. Rota bazlı personelin gittiği mesafeler (km).....	53

ÖZET

Anahtar Kelimeler: Araç Rotalama Problemi, Optimizasyon, Tamsayılı Doğrusal Programlama, LINGO

Günümüzde şirketlerin üretim/hizmet faaliyetlerini gerçekleştirirken göz önüne aldığı, katlanmak durumunda olduğu ya da problem olarak karşılaştığı süreçlerden biri de lojistik yönetimidir. Literatürde çok çeşidine rastlanılan ve problem türüne göre değişiklik arz eden lojistik yönetimde Araç Rotalama Problemi en yaygın problem türlerindedir.

Bu tez çalışmasında Araç Rotalama Problemi, literatürde yer alan uygulamalar ve ARP çözüm yöntemlerinden genel olarak bahsedilmiş ve Ankara'da bulunan bir firmanın personeline sağladığı servis hizmeti ele alınmıştır. Firma üç ana semtte bulunan 37 personeline üç araç ile belirli duraklardan almakta ve mesai sonrasında duraklara bırakmaktadır.

Araçların firmadan hareket edip tekrar firmaya dönmesinden dolayı Kapalı Uçlu Araç Rotalama Problemi olarak ele alınan bu çalışmada Tamsayılı Doğrusal Programlama yöntemiyle LINGO 18.0 program kullanılarak çözüme gidilmiştir. Araç kapasitesinin değişmesine göre oluşan rotaların toplam mesafeleri ile rota uzunluğunun değiştirilmesiyle elde edilen toplam mesafe hesaplanmıştır. Personelin serviste geçirdiği sürelerin ortalaması hesaplanarak çözüm analiz edilmiştir.

VEHICLE ROUTING PROBLEM SOLUTION WITH INTEGER LINEAR PROGRAMMING AND AN APPLICATION IN SERVICE NETWORK

SUMMARY

Keywords: Vehicle Routing Problem, Optimization, Integer Linear Programming, LINGO

Nowadays, logistic management is one of the processes that companies take into account when they perform their production / service activities, or where they face problems. Vehicle Routing Problem is one of the most common problem types in logistics management which has a wide variety in the literature and which varies according to the problem type.

In this thesis, the Vehicle Routing Problem, its applications in the literature and VRP solution methods are mentioned in general and an application for a company in Ankara which provides shuttle service for the staff is discussed. The company has 37 employees in three main districts with three vehicles at certain stops and leaves them at the stops after work.

In this study, which is considered as Closed End Vehicle Routing Problem due to moving of the vehicles from the company and returning to the company, LINGO 18.0 program has been solved by using Integer Linear Programming method. The total distance calculated by changing the total distance of the routes formed according to the change in vehicle capacity and the total length of the route were calculated. The average time spent in the service of the personnel was calculated and the solution was analyzed.

BÖLÜM 1. GİRİŞ

Hizmet/üretim sektörünün önemli süreçlerinden biri olan lojistik yönetimi, tedarikten üretim hatlarına, ürün dağıtımından personel servislerine kadar çok geniş yer tutmakta ve maliyetlerde etkin bir rolü bulunmaktadır.

İşletmelerin öncelikli amaçlarından olan maliyetlerin azaltılması için lojistik yönetimiyle ilgili pek çok çalışma yapılmakta, uygulanan problem türleri gün geçtikçe yaygınlaşmaktadır. Bu duruma paralel olarak çözüm yöntemleri de gelişmektedir.

Personele hizmet veren servis araçları, ürün dağıtım ya da ürün toplama, taşıma/depolama, müşteri talepleri gibi lojistik faaliyetler içerisinde yer alan süreçler toplam maliyetin içerisindeki önemli bir yeri vardır.

Yöneylem araştırması alanında çeşitli varyasyonları olan ve çok fazla çalışmanın günümüze kadar yapıldığı araç rotalama problemi, teknolojik gelişmeler, tüketim alışkanlıklarının evrilmesi, ekonomik etkenler vb. durumların etkisiyle günden güne üzerindeki ilgiyi artırarak devam ettirmektedir. Uluslararası ekonomik ilişkilerin gittikçe daha çok gelişmesi ARP nin boyutunu arttırdıkça, çeşitliliğini de artırmaktadır.

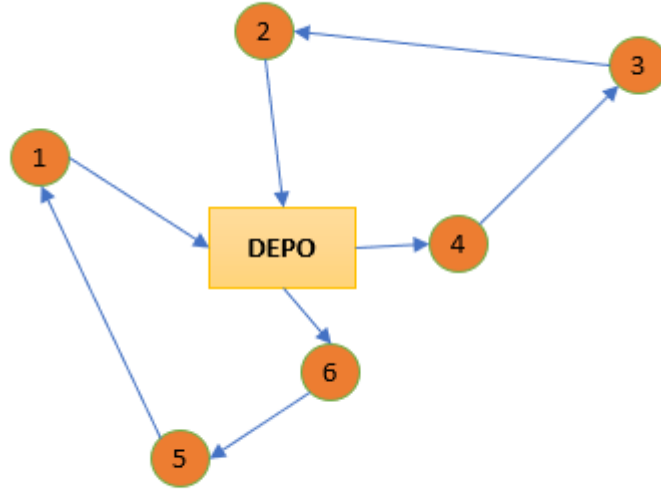
Bu tez çalışmasında öncelikle Araç Rotalama Probleminin tanımı ve çeşitleri hakkında bilgiler yer verilmiştir. Kesin ve Sezgisel çözüm yöntemlerden bahsedilmiş, Ankara'da bir firmanın servis araçlarının rotalanması üzerinde uygulama Tam Sayılı Doğrusal Programlama yöntemiyle yapılmıştır. Çalışmada 18 kişi, 22 kişi ve 24 kişi kapasiteli araçlar için ve rota uzunluğunun maksimum 50 km olduğu durum için problem ele alınmıştır. LINGO 18.0 programında elde edilen çözümler değerlendirilmiştir.

BÖLÜM 2. ARAÇ ROTALAMA PROBLEMİ

Araç rotalama, belirli hedef noktalarına belirli araçların en kısa yol, maksimum kapasite ya da minimum maliyet amaçlanarak rotaların belirlenmesidir. Araç rotalama problemi, elli yılı aşkın süredir çözümler üretilen giderek daha yaygın ve çeşitleri artan problem türüdür. Literatürde deterministik çözümlerin yanı sıra problem büyüklüğünün ve kompleksliğinin etkisiyle sezgisel yöntemlerle de çözüm aranmış, yeni yöntemler eklenmiştir.

2.1. Araç Rotalama Problemi Tanımı

Genel ifade ile Araç Rotalama Problemi (ARP), araç rotalarının araç sayısına göre oluşturulması ile ilgilidir. Araçlar bir depodan başlar ve tüm müşterileri belirli bir sıra ile ziyaret eder, sonrasında başladığı noktaya döner. Müşteriler rotalardan birinde muhakkak yer alır ve araç kapasitesi araçlar müşterilere atanırken göz önünde bulundurulur. Toplam kapasite geçmeyecek şekilde rotalama yapılır (Düzakın & Demircioğlu, 2009). Örnek bir araç rotalama Şekil 2.1.'de gösterilmiştir.



Şekil 2.1. Araç rotalama problemi

ARP nin en bilinen amaçları şu şekildedir;

- Taşıma maliyetlerinin minimizasyonu: Araçların toplam kat ettiği mesafeye ve araç maliyetlerini içeren toplam taşıma maliyetlerini minimize etmek.
- Toplam araç sayısının minimizasyonu
- Rotalarda dengenin olması,
- Müşterileri taleplerinin birden fazla araçla temin edilmesinden kaynaklanan cezaların minimizasyonu (Atmaca, 2012).

Bunun yanı sıra yan amaç müşteri memnuniyetinin maksimize edilmesidir.

2.2. ARP Kısıtları

Araç rotalama problemi, NP-zor problem sınıfında yer alan, hesaplamının problemin boyutuyla birlikte eksponansiyel olarak arttığı en yaygın bir tam sayılı programlama problemidir (Yılmaz, 2008).

ARP, çok iyi bilinen iki problem olan Gezgin Satıcı Problemi (TSP) ile Sırt Çantası Problemi (BBP) birleşimi gibi görülebilir (Machado, 2002). ARP’lerde geçerli olan kısıtları şu şekildedir;

- Kapasite kısıtı: Araçlar kapasite, yük ve yolcu vb. açıdan kısıtlıdır.
- Mesafe Kısıtı: Noktalar arası mesafeler belirlidir. Özellikle servis araçları gibi her noktaya uğranması gereken problemlerde araçların aldıkları toplam mesafe kısıtıyla birlikte bu kısıt ta değerlendirilir.
- Zaman: belirlenmiş bir zaman aralığında noktalardan geçilmesi gerekmektedir. Sürücülerin çalışma saatleri belirli ve sınırlıdır.
- Müşteri/Taleple ilgili kısıtlar: Her noktada bir ürünün talep edilmesi, her duraktan en az bir müşterinin alınacak olması vb. kısıtlar bu tür kısıtlardır.

Araç Rotalama Probleminde sağlanması gereken koşullar şunlardır;

- Tüm talepler karşılanmalıdır.
- Her müşteri muhakkak bir araç rotasında olmak zorundadır.
- Rotada yer alan müşterilerin toplam talebi, rotaya atanmış aracın kapasitesinden düşük olmak zorundadır.
- Başlangıç ve bitiş noktaları her rota için belirlenmelidir.
- Rotanın maksimum uzunluğu daha önce belirlenmişse, rotaların uzunlukları bu şartı sağlamalıdır.
- Araç sayısının sabit olduğu ARP olduğu gibi, değişken olduğu ARP de bulunmaktadır (Düzakın & Demircioğlu, 2009).

2.3. Araç Rotalama Problemlerinin Uygulama Alanı

ARP’nin günümüzde uygulama alanları günden güne genişlemektedir. Teknolojik gelişmeler sonucu hava sahasının da kullanım alanlarını genişlettiği/genişleteceği göz önüne alındığında ARP’nin tahminden öte çeşitleneceği düşünülebilir. Uygulama alanlarının artmasının yanısıra yazılımsal çözümler sunan firmalar da artmaktadır. İnternet servis sağlayıcı gibi teknoloji şirketlerinin müşterilerine sundukları kurulum,

arıza gibi alt hizmetler için bahsedilen çözümlerden destek alabilmektedirler. ARP uygulamalarının başarılı olması açısından dikkate alınması önerilen sekiz prensip şunlardır (Golden, Ragwahan, & Wasil, 2005);

- Birbirine en yakın olanlar noktalar seçilmelidir. Böylece toplam gidilen yol kısalmır.
- Benzer noktalardaki dağıtımlar farklı günler için birleştirilerek, yakın zaman aralığında tekrar ziyaret edilmesi engellenmelidir.
- Rotanın başlangıç noktası olarak mümkünse en uzak mesafedeki müşteri seçilmelidir.
- Gözyaşı şeklinde bir rotalama ile uzak noktalara ulaşımında kazanç elde edilebilecektir.
- Maliyetin azaltılmasının yollarından biri de mümkün olan en yüksek kapasiteli araçlar kullanılmalıdır.
- Mümkünse aynı araç kullanarak dağıtım ve toplama yapılmalıdır.
- Kapasitesi daha az araçlar rota dışında kalan noktalara ulaşımında kullanılabilir.
- İhtiyaç duyuluyorsa dağıtımların ve toplamaların zamanları kıyaslanarak zaman minimizasyonu sağlanmalıdır.

Araç rotalama problemlerinin çeşitli uygulama alanları vardır. Temel prensip, belirli noktalar arasındaki araç hareketleri ve müşteri taleplerinin/problemin amacının gerçekleştirilmesi olduğundan günümüzde geniş bir alanda uygulamaları bulunmaktadır. Belli başlı bazı uygulama alanları şunlardır (Düzakın & Demircioğlu, 2009);

- Üretim planlama,
- Stok planlaması ve ürünlerin satış yerlerine sevkiyatı,
- Gıda sektörü için Dağıtım,
- DVD film kiralama hizmeti,
- Para dağıtımı,
- Benzin ve mazot dağıtımı,
- Ana depodan mağazalara ürün dağıtılması

- Süt dağıtımı ve toplanması,
- Çöp toplama/taşıma
- Müşterilere ürün/hizmet dağılımı (bir veya daha fazla depo)
- İnternet alışverişi
- Posta hizmetleri
- Havayolu şirketleri ile yolcu ve ürün taşınması

2.4. Problemin Matematiksel Modeli

ARP'ler, bir veya birden fazla depodan coğrafi olarak dağınık merkezlere hizmet vermek için atanan araçların optimum rotalarının (dağıtım/rotalama) planlanması problemleridir (Yılmaz, 2008).

Klasik bir ARP'de araç sayısı kadar rota sayısı oluşur. Her müşteriye bir araç muhakkak uğrar ve tüm araçlar, belirlenen bitiş noktasıyla rotayı tamamlar. ARP modeli şu şekildedir;

Ziyaret edilecek noktaları tanımlayan $V = \{v_0, v_1, \dots, v_n\}$ düğüm kümesi ve v_i den v_j ye hareketi gösteren, tüm noktalar arası hareketleri içeren $E = \{(v_i, v_j) : v_i, v_j \in V, i \neq j\}$ bir kenar kümesi ise $G(V, E)$ grafi ARP'nin çözüm bölgesidir. Kombinasyonel olan problemin çözüm bölgesini ise grafin kenarları sınırlamaktadır.

V kümesinde v_0 ana depoyu, müşterileri ise n sayısı ifade etmektedir. Her müşteri bir q_i talebine sahiptir. Araçların her biri C kapasitelidir. Literatürde bu kapasite kısıtının yer almadığı problemlere Çoklu Gezgin Satıcı Problemi adı verilmektedir. Problemin temel amacı, hizmeti tüm müşterilerin almasını sağlamak için her aracın güzergâhını belirlemek yani araç sayısı kadar rota belirlemektir (Yılmaz, 2008).

Aşağıdaki formülasyonda tek bir depoya sahip klasik bir ARP'nin doğrusal modeli yer almaktadır.

M: Araç Sayısı

N: Müşteri Sayısı

d_{ij} : i. ve j. nokta arasındaki mesafe

q_i : i. Müşterinin talep miktarı

C: Araç Kapasitesi

Değişken;

$$X_{ijk} : \begin{cases} 1, k \text{ nolu araç } i \text{ noktasından } j \text{ noktasına hareket ederse} \\ 0, \text{ aksi takdirde} \end{cases}$$

Amaç fonksiyonu:

$$\text{Min } Z = \sum_{i=0}^N \sum_{j=0, j \neq i}^N \sum_{k=1}^M d_{ij} X_{ijk} \quad (5.1)$$

Şu kısıtlara göre;

$$i=0 \text{ için; } \sum_{k=1}^M \sum_{j=1}^N X_{ijk} = M \quad (5.2)$$

$$i \in \{1, \dots, N\} \text{ için; } \sum_{k=1}^M \sum_{j=0}^N X_{ijk} = 1 \quad (5.3)$$

$$j \in \{1, \dots, N\} \text{ için; } \sum_{k=1}^M \sum_{i=0, i \neq j}^N X_{ijk} = 1 \quad (5.4)$$

$$k \in \{1, \dots, M\} \text{ için; } \sum_{i=1}^M X_{i0k} \leq 1 \quad (5.5)$$

$$k \in \{1, \dots, M\} \text{ için; } \sum_{i=1}^N q_i \sum_{j=0, j \neq i}^N X_{ijk} \leq C \quad (5.6)$$

Amaç fonksiyonu (5.1) toplam kat edilecek mesafeyi en aza indirerek maliyetinin minimize edilmesidir. (5.2) nolu kısıt denklemi M adet aracın işletmeden çıkmasını

sağlar. (5.3) nolu kısıt denklemi i. Müşteriye sadece bir aracın mutlaka gelmesini sağlayan denklemdir. (5.4) nolu kısıt denklemi (5.3) nolu kısıda benzer şekilde sadece bir aracın müşteriden ayrılmasını sağlar. (5.5) nolu kısıt denklemi bir aracın işletmeden bir defa çıkarak bir rotada yer almasını sağlar. (5.6) nolu kısıt denklemi ise araçların uğradıkları müşterilerin toplam talebinin araç kapasitesini (C) yi geçmemesini sağlar. Bu modelde araç sayısı rota sayısına eşittir ancak bazı ARP türlerinde rota sayısı en fazla (rota sayısı $\leq M$) araç sayısı kadar olabilir.

Bir ARP problemine genel olarak şu bilgilere ihtiyaç vardır (Erol, 2006):

- Müşteriler arası ulaşım süresi veya aralarındaki mesafe
- Depo/firma ile müşteriler arası mesafe ya da trafik süresi
- Müşteri talep miktarları
- Araç sayısı ve araç kapasite değeri
- Araç fonksiyonu

2.5. Araç Rotalama Probleminin Bileşenleri

ARP'nin temel bileşenleri için talep, malzeme tipi, dağıtım/toplama noktaları ve araç oluşturur (Aydemir, 2006).

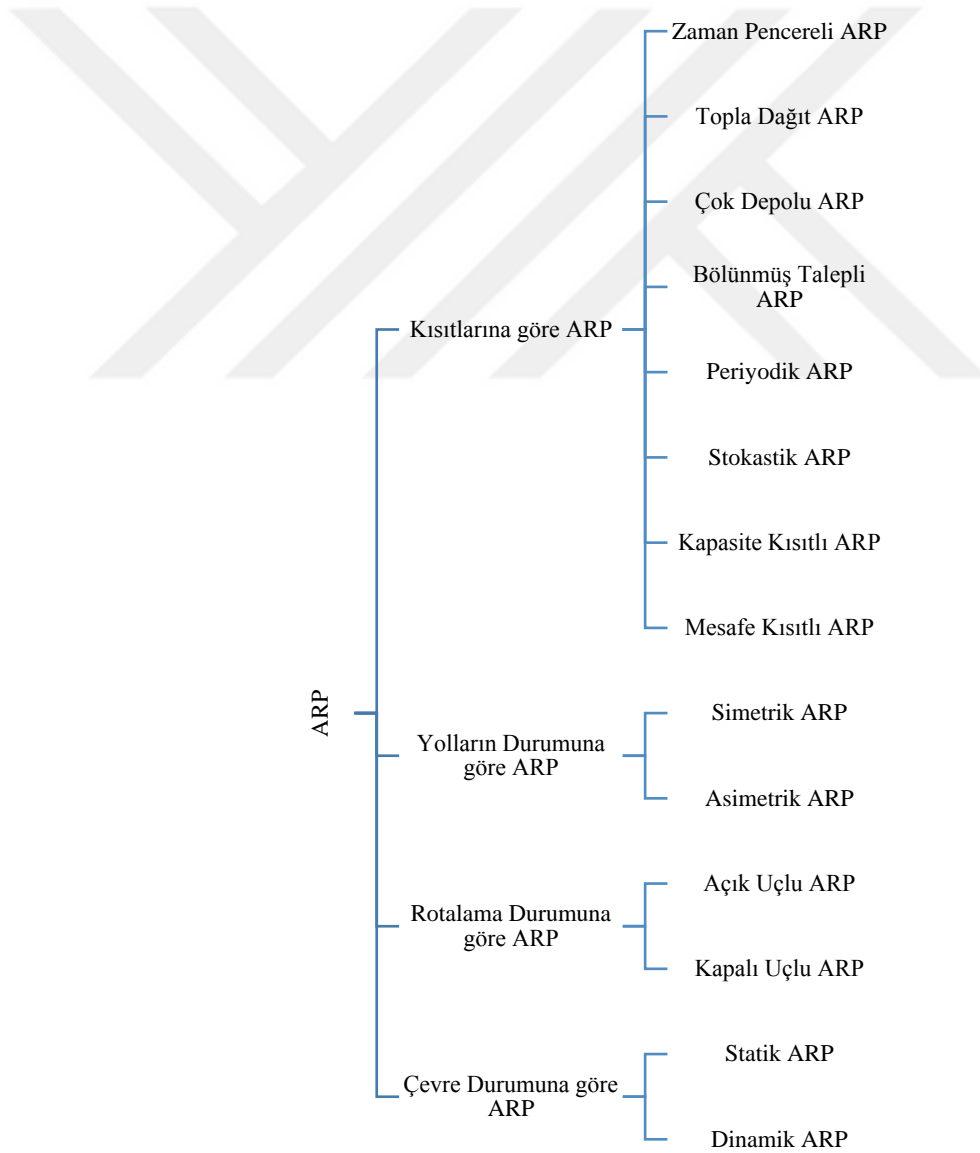
- Talep: Taleplerin önceden bilindiği, statik talep olarak da adlandırılan talepler olabileceği gibi, bazı müşterilerin talepleri önceden bilinmeden araç rotada ilerlerken belli olabilir.
- Malzeme Tipi: Çeşitli malzemelerin taşındığı araçlarda malzeme tipine göre önemliliği değişen durumlar vardır. Örneğin gıda maddeleri, gazete dağıtımı, çöp toplama basit paketler olarak adlandırılır ve probleme karmaşıklık eklemesler. Diğer yandan öğrenci ya da personel servisleri güvenlik, etkinlik, eşitlik gibi ilave ihtiyaçlardan dolayı daha karmaşık bir yapıya sahiptir. Tehlikeli malzemelerin taşındığı araçlar için ise coğrafi özellikler büyük önem kazanır.

- Dağıtım/Toplama Noktaları: Genel olarak ARP’de, müşteriler dağıtım noktaları, firma ya da depo ise toplama noktasıdır. Bu duruma en iyi örneklerden biri fabrikada üretilen malların toptancılara dağıtılmasıdır. Araç genellikle depodan hareket eder ve depoya geri döner. Tek depolu ve çok depolu olan araç rotalama problemlerinde her deponun kendine ait araçları varsa problem yine tek depolu araç rotalama problemleri olarak ele alınır. Başlangıç deposu ve bitiş deposu farklı ise bir bütün olarak ele alınır.
- Araç: Araç kapasitesi tüm ARP’ler için belirlidir ve genellikle araçların aynı kapasitede olduğu kabul edilir. Araçların farklı kapasitelere sahip olması probleme yeni bir kısıt/karar verici olarak eklenir (Aydemir, 2006).



BÖLÜM 3. ARAÇ ROTALAMA PROBLEM ÇEŞİTLERİ

Kısıtlara, yol durumuna, problem verilerinin dinamik ya da statik olmasına ve rota başlangıç-bitiş noktasının aynı olup olmamasına göre ARP çeşitleri vardır. Şekil 3.1.'de ARP nin çeşitleri genel olarak gösterilmiştir.



Şekil 3.1. ARP çeşitleri

3.1. Kısıtlarına göre ARP

ARP bileşenleri olan kapasite, mesafe, zaman ve müşteri talepleri göz önüne alınan araç rotalama problemleridir. Depo sayısının birden fazla olduğu ARP, toplama ve dağıtım ARP ler yine bu kategoride yer almaktadır.

Kapasite kısıtlı araç rotalama problemi (KKARP), bir işletmenin yükleme kapasiteleri kısıtlı araçlarla talepleri belli n adet müşterisine ulaşabilmesi için rota planlaması problemidir (Yılmaz, 2008). KKARP şu kısıtlara sahiptir;

- Her rota depodan başlar ve depoda biter.
- Her müşteri bir kez ziyaret edilir. Müşterinin talebi bölünemez.
- Her rotadan karşılanan taleplerin toplamı araç kapasitesinden fazla olamaz.

Mesafe kısıtlı araç rotalama probleminde araçların kat edebileceği belirli bir toplam mesafe bulunmaktadır. Araç sürüsünün çalışma süresi kısıtlı olduğundan ya da taşınan ürünün belli bir zamana kadar taşınması gerekliliğinden bu tür kısıt olabilir.

Zaman pencereci araç rotalama probleminin diğerlerinden ayıran özellik, belli bir zaman aralığında müşterinin ziyaret edilmesidir. Zaman Pencereci Araç Rotalama Problemi (ZPARP), bahsi geçen zaman aralığına göre ikiye ayrılır. Belirlenen zaman aralığından önce gidildiğinde beklenildiği, sonra gidildiğinde ise teslimatın/işlemin yapılamadığı Sıkı Zaman Pencereci ARP ve belirlenen zaman aralığında gidildiğinde teslimatın/işlemin gerçekleştirildiği ancak ceza maliyetinin olduğu Esnek Zaman Pencereci ARP'dir (Dursun, 2009).

Bölünmüş talepli ARP tipinde müşteriye birden fazla araç uğrayarak müşteri talebi bölünebilir. Amaçlardan biri de maliyetinin minimize edilmesi olduğundan, maliyetin azalması söz konusu ise ARP nin bu şekilde ele alınması söz konusu olabilir.

Çok Depolu Araç Rotalama Problemi (ÇDARP), araçların birden fazla depodan dağıtımını gerçekleştirdiği araç rotalama problemi türüdür. Bu problemde her araç,

atanmış olduğu depoya tur sonunda dönmelidir. ÇDARP, PARP probleminin özel bir durumudur. PARP modelinin formülasyonunda yer alan gün terimleri, depoları ifade edecek şekilde revize edilerek ÇDARP formülasyonu elde edilir (Tokaylı, 2005).

Topla-dağıt ARP'de rotaların başlangıç ve bitiş noktasının depo olduğu ve her müşterinin bir araçla talebinin sağlandığı, bu gerçekleştirilirken araç kapasitesinin rotadaki toplam talebin aşmadığı dağıtım toplamalı araç rotalama problemleridir (Atmaca, 2012).

Dağıtım ve toplama işleminin gerçekleştirilmesine göre üçe ayrılır (Atmaca, 2012);

- Önce dağıtım sonra toplama: Depodan tüm malzeme müşterilere dağıtılmasının ardından depoya gönderilecek malzemeler müşterilerden toplanır.
- Karışık dağıtım toplama: Toplama ve dağıtım işi karışık yapılır.
- Eş zamanlı dağıtım ve toplama: Dağıtım ve toplama müşteride aynı anda yapılır.

Bu şekilde müşteriler bir kez ziyaret edilmiş olur.

Periyodik Araç Rotalama Problemi (PARP), planlama periyodunun birden fazla gün için yapıldığı ARP'dir. Belirlenen periyotta her müşteriye en az bir kere uğranılması gerekmektedir. Klasik PARP'nde araçlar aynı kapasiteye sahiptir ve müşteri ziyaretleri tamamlanınca tekrar başlangıç deposuna geri gitmek durumundadır. Müşteriler bir kez ziyaret edilir ve araç kapasite aşılmayacak şekilde ziyaret gerçekleşir. Planlama periyodu belirlenen gün kadardır (Tokaylı, 2005).

Stokastik Araç Rotalama Problemi (SARP), problem kısıtlarının rassal olduğu klasik araç rotalama problemidir. Üç farklı türde SARP vardır (Yılmaz, 2008);

- Stokastik müşteriler: Her i müşterisinin varlığı için p_i olasılığı vardır, $1-p_i$ olasılığıyla müşteri yoktur.

- Stokastik talepler: Her müşterinin talebi q_i , rassal bir değişkendir.
- Stokastik zamanlar: Servis zamanları s_i ve dolaşım zamanları t_{ij} rassal değişkenlerdir

3.2. Yolların Durumuna göre ARP

Düğümler arası mesafelerin dikkate alındığı bu ARP türü gidiş ve dönüş mesafesinin eşit olup olmamasına göre ikiye ayrılmaktadır.

Simetrik araç rotalama probleminde i . ve j . Düğümler için; i .den j .ye gidilen mesafe j .den i .ye gidilen mesafe ile aynı ise simetrik ARP'dir. Diğer bir ifadeyle, grafa ait tüm düğümler için $d_{ij} = d_{ji}$ ise problem simetriktir.

Asimetrik araç rotalama probleminde i . ve j . Düğümler için; i .den j .ye gidilen mesafe j .den i .ye gidilen mesafe ile aynı değilse asimetrik ARP'dir.

3.3. Rotalama Durumuna göre ARP

Araç rotalama problemleri, rotanın başlangıç ve bitiş noktasının aynı nokta olması veya farklı nokta olmasına göre açık ve kapalı uçlu olmak üzere iki farklı şekilde incelenir (Erol, 2006).

Açık uçlu ARP, rotanın depodan başlatıp depoda bitmediği, farklı noktalarda bittiği ARP türüdür.

Kapalı uçlu ARP, rotanın depodan başlayıp depoda bittiği ARP türüdür. Araçlar depodan çıkar, müşteri talepleri karşılandıktan sonra depoya geri döner.

3.4. Çevre Durumuna göre ARP

Çevre durumuna göre ARP'de problem verilerinin statik ya da dinamik olmasına göre çeşitlenir.

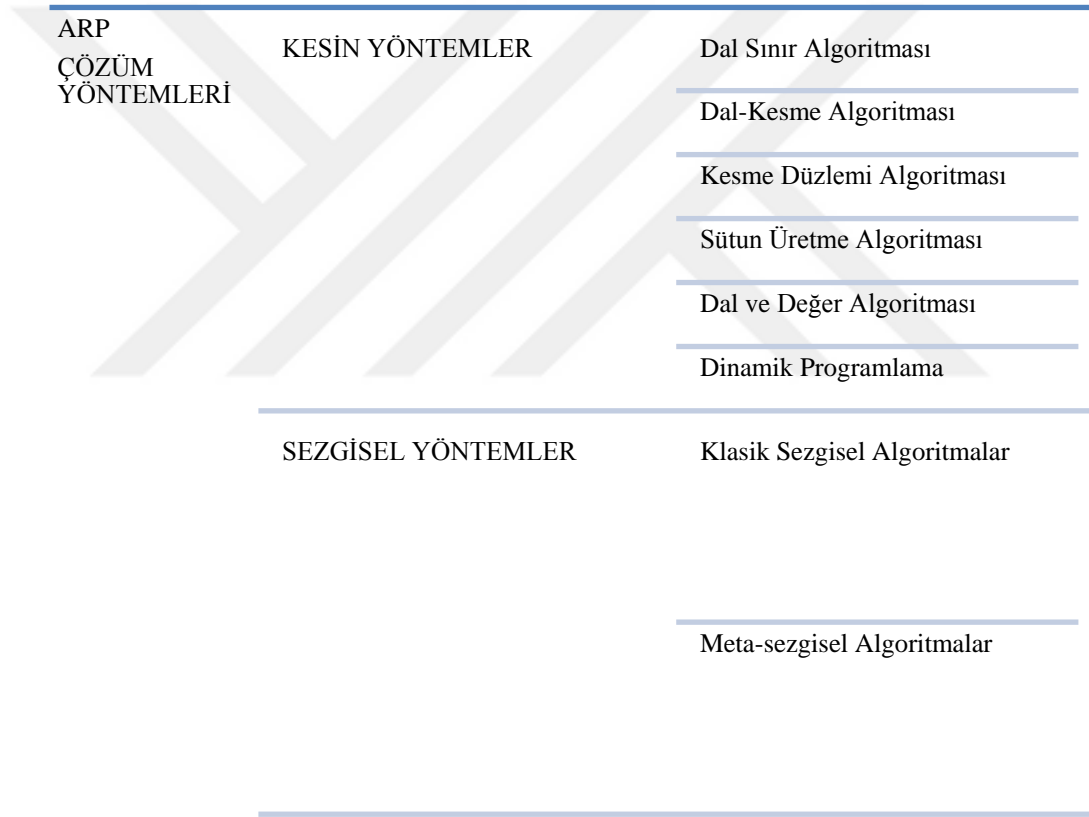
Statik ARP, çevrenin statik olmasına göre olan araç rotalama problemlerinde, çözümden önce talepler, maliyetler, bir veya birden fazla kısıtlar gibi değişkenler bilinmekle birlikte bahsi geçen bu bilgiler sabitliklerini problemin çözüm aşamasında da korurlar. Statik Araç Rotalama Problemi (SARP) üzerinde genellikle çalışmalar yapılmıştır ve deterministik ARP olarak da karşımıza çıkmaktadır.

Gerçek hayattan örneklerinde ise miktarı ve zamanı önceden bilinebilen talepler için rotalar oluşturulmakta ve servis sistemlerini genel değerlendirmede kullanılmaktadır (Akca, 2015).

Dinamik araç rotalama probleminde; çözüm gerekli olan ve rotaları belirleyen bilgiler dinamik bir yapıya sahiptir. Örneğin; müşteri siparişleri. Dinamik araç rota probleminde strateji, girdi güncellemesinin olduğu her defada yeni bir problem tanımlı statik problem serisinin çözümlenerek dinamik çevrenin ele alınmasıdır (Bianchi, 2000).

BÖLÜM 4. ARP ÇÖZÜM YÖNTEMLERİ

Probleminin çözüm yaklaşımları, çözümün optimum olup olmamasına göre Kesin ve Sezgisel olmak üzere iki ana gruba ayrılır. Şekil 4.1.'de başlıca ARP çözümlerine gösterilmiştir.



Şekil 4.1. ARP çözüm yöntemleri

4.1. Kesin Yöntemler

Kesin yöntemler, optimum sonuca götüren yöntemler olup literatürde yaygın başlıca yöntemlerden bazıları aşağıda kısaca açıklanmıştır.

Dal-sınır algoritmaları; Çözüm uzayını alt problemlere ayırarak böl ve ele geçir stratejisiyle bunu gerçekleştirir. Sonrasında her alt problemi ayrı olarak optimize etmeyi amaçlar. Bu şekilde çözüm uzayında yer alan tüm sonuçlar değerlendirilir (Yılmaz, 2008).

Dal-sınır yönteminin temelinde, tüm fizibil çözümleri belirlemeyi sağlayan bir tekniktir. Fakat optimal çözüme gitmeyen çözüm seçeneklerinin önceden elimine edilmesini de sağlar. Bu yüzden gereke değerlendirme sayısı, genel olarak çözüm alanını küçük alt setlere böler. Bu alt setler " dallandırma noktaları" olarak adlandırılır (www.ktu.edu.tr, 08.04.2019).

Her alt set, araştırma için daha fazla ilerleme gerekip gerekmediği belirlenmek üzere değerlendirilir. Amaç fonksiyonuna ait değerleri mevcut sınırlarla karşılaştırarak değerlendirme gerçekleştirilir. Maliyetin minimize edildiği problemlerde alt setin olurlu çözümü için amaç fonksiyon değerlerine bir alt sınır bulunmuş olunur.

Kesme düzlemi algoritması, modelin en iyi çözümü elde edildikten sonra çözümün tamsayı olması sağlanana kadar, her iterasyonda, tamsayı değerine sahip değişkenlerden yola çıkarak yeni bir kısıt modele eklenerek işleme devamı şeklindeki algoritmadır.

İlk önce G. Dantzig, D. Fulkerson ve S. Johnson 1954' de, daha sonra Markowitz ve A. Manne 1957' de kesme düzlemine ilişkin yaklaşımlar geliştirmişler, belirli sayıda ardıştırmadan sonra en iyi çözümü veren yapılandırılmış algoritmayı 1958' de R. Gomory geliştirmiştir. İzleyen yıllarda önceki yaklaşımlara ekler ve özel durumlar için uygulanabilir yeni kesme düzlemi teknikleri geliştirilmiştir.

Gomory' nin yaklaşımı, konu üzerinde ilk genellemeleri Gomory' nin yapmış olması nedeniyle, kaynaklarda “Gomory' nin Kesme Düzlemi Tekniği” olarak da yer almaktadır.

İki değişkenli problemlerde Gomory kısıtlamasının özellikleri şunlardır;

- Elde edilen bu kısıtlamalar bir önceki uygun alandan genellikle konveks bir alan keserler.
- Kesme düzeyi uygun olmasa bile en az bir kafes noktasından geçer.
- Her kesim, bütün uygun kafes noktalarını kapsayacak daha küçük bir alana yaklaşır.

Optimal çözüm tablosuna yeni bir sıra olarak eklenen eşitlik sınırlamasına ait katsayılar tüm tamsayının elde edilmesini sağlayacak formdadır. Doğrusal Programlama çözüm yöntemleri bu yöntemle uygulanarak optimal çözüm bulunur.

Dal-kesme yöntemi; tam sayılı programlama problemleri için etkili bir yöntem olup kesme düzlemi algoritması ve dal-sınır yöntemlerinin bir birleşimi ile elde edilmiştir. Dal-kesme yöntemi de diğer tam sayılı programlama algoritmalarına benzer şekilde tam sayılı programlama probleminin, doğrusal programlama ile yapılacak çözümü ile başlar.

Yalnızca kesme düzlemi yaklaşımı ile tamsayılı programlama problemini verimli olarak çözebilmek mümkün değildir, bu yüzden alternatif optimum çözümleri bulmak için dallandırma yapmak da ek olarak gerekmektedir. Dal-sınır yaklaşımı, kesme düzlemi algoritmasının uygulanması ile oldukça hız kazanabilir. Dallandırma yapılmadan kesme eklenebileceği gibi ağacın her düğümünün çözüm aşamasında da kesmeler kullanılabilir (Başkaya & Avcı Öztürk, 2005).

Dinamik programlama, parçalarına bölünen problem ya da problemin bir kısmına ait çözümler üreterek, üretilen bu çözümleri depolayarak çözüm yaklaşımı sunmaktadır. Bu çözümler, ihtiyaç duyulduğunda, yeniden çözmek yerine, yeniden canlandırılmak

suretiyle problemin genel çözümüne eklenerek, nihai çözüme ulaşılmaktadır. Dinamik programlama, çok aşamalı karar verme problemlerinde optimal bir silsileye karar vermede kullanılabilir.

Problem, alt problemlere ayrılır ve ayrılan her bir alt problem için optimal bir çözüm bulunur, karar verme aşamalarının n sayıda olduğu düşünülen bir problem, n sayıda olan ve her birinin tek bir karar değişkenine sahip olduğu problemlere bölünür (Yılmaz, 2008).

Dinamik programlama yaklaşımında, (orijinal olana “benzer”) bir alt-problemler ailesi tanımlanır ve verilen bir aşamada durumlar, tekrar edilerek, bir önceki aşamadaki durumlar üzerine dinamik programlama uygulanarak hesaplanır. Optimum çözüm, en son aşamada en iyi durumuna karşılık gelmektedir (Tezer, 2009).

4.2. Sezgisel Yöntemler

Geçmişten günümüze bilimsel çalışmalar göz önüne alındığında, gerçek hayatta karşılaşılan problemlerin kompleksliği ve boyutu büyüdükçe çözüm kümesi değişkenlik göstermiştir. Kimi çözüm yöntemleri problemin optimum çözümü için yeterli olurken, bazı zaman ve koşullarda optimum çözüme gitmek zorlaşmış hatta imkânsız hale gelmiştir.

Klasik Sezgisel yaklaşımlar belirli bir sürede olurlu çözümlere ulaşabilirken, Meta Sezgisel yaklaşımlar optimum sonuca en yakın çözümler de sunabilmektedir (Dursun, 2009).

Klasik sezgisel algoritmalar, Rotaların (turların) oluşturulduğu ya da iyileştirildiği algoritmalara göre ayrılan klasik sezgisel yöntemler; tur kurucu, tur geliştirici ve iki aşamalı yöntemler olarak 3 gruba ayrılmaktadır.

Yapısal Sezgisel Algoritmada önce maliyetin minimizasyonuna göre müşteriler seçilir, daha sonra kapasite ve zaman kısıtları dikkate alınarak rotalama belirlenir (Dursun,

2009). Tasarruf Algoritması, Yerleştirme Algoritması, En Yakın Komşu, En Kısa Yol Yöntemi bilinen başlıca yapısal sezgisel algoritmalarıdır.

İyileştirme (tur geliştirici) sezgisel algoritmalar, bir rota üzerinde iyileştirme söz konusu ise bu algoritma kullanılarak çözüme gidilebilir. Tek Rota İyileştirmeli Sezgisel Algoritmalar, Çok Rota İyileştirmeli Algoritmalar, Van Breedam Sezgiseli, Thomson ve Psaraftis Sezgiseli ve Kinderwater ve Savelsbergh Sezgiseli bu tür algoritmalara örnektir.

İki aşamalı sezgisel yöntemler, önce kümeleme sonra rotalama ya da önce rotalama sonra kümeleme gibi yöntemlerdir. Önce kümelemenin yapıldığı ve iki-aşamalı yöntemlerin en yaygın olanı Süpürge Algoritmasıdır. Kapasiteli ARP problemini, ilk aşamada müşterileri kümelere ayırarak m-GSP Problemine çevirir. İki kriter altında kümeleme gerçekleştirilir. Müşteriler ve depo, depo orijin noktasında konumlanacak şekilde, polar koordinatlara taşınır. İlk kriter müşterilerin birbirlerine olan açısıdır ve en az açığa göre kümeleme gerçekleştirilir. İkinci kriter ise kümelenen müşterilerin taleplerinin araç kapasitesini geçmemesidir. İkinci aşamada ise her küme GSP problemi gibi çözülür (Dursun, 2009). Süpürme Algoritması, Fisher ve Jaikumar Algoritması, Bramle ve Simichi Levi Algoritması, Önce Rotala Sonra Grupla Yöntemi, Taç Yaprağı Algoritması, Taillard Algoritması ve Budanmış Dal sınır Algoritması bu tür yaklaşımlardandır.

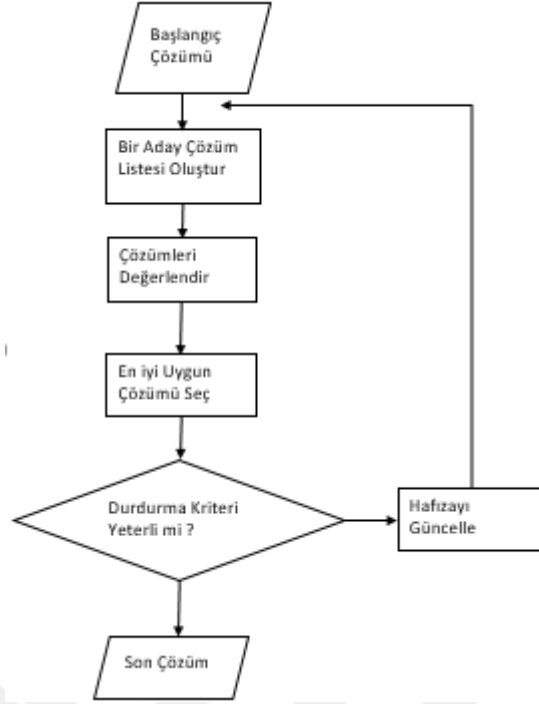
Metasezgisel yaklaşım, çözüm uzayında olasılık temelli ancak bilinçli bir mantıkla arama gerçekleştiren yöntemleri içermektedir. Bu yöntemler her adımda oluşturulan çözüm kümesinden yola çıkarak yeni çözümler üretmektedirler. Böylece arama uzayının en uygununa yakın olan noktalarında aramalar yapılarak, yerel en iyi nokta seçiminden de kurtularak en uygun çözüme ulaşmaya çalışılır (Pekdemir, 2012).

Tavlama Benzetimi, stokastik bir arama algoritmadır. Belirli bir başlangıç sıcaklığından başlayarak yavaş yavaş soğutulan katıların tavlama sürecinin benzetimidir.

TB algoritmasının amacı olarak, mümkün olan tüm çözüm noktalarının bir alt kümesinde (S) tanımlanmış bir $f(x)$ fonksiyonunu eniyileyecek bir x çözümü bulmaktır. Rassal olan başlangıç çözümüyle aramaya başlayan algoritma uygun bir mekanizma ile bu çözüme komşu olan bir çözümü seçer ve $f(x)$ 'de meydana gelen değişiklik hesaplanır. Eğer değişiklik istenilen yönde ise komşu çözüm mevcut çözüm olarak ele alınır. Eğer arzu edilen yönde bir değişiklik elde edilmemişse algoritma bu çözümü "Metropolis Kriteri" ile sağlanan olasılık değeri ile kabul eder. Amaç fonksiyonunda ters yönde bir değişiklik olmasına sebep olan bir çözümün belli olasılık değeri ile kabulü, söz konusu algoritmanın yerel en iyi noktalardan kurtulmasını sağlar. Yukarıdaki olasılık değerine göre T değeri yüksek olmasıyla amaç fonksiyonunda meydana gelen artışların büyük bir kısmı kabul edilecektir. T değeri azaldıkça kabul edilme oranı da azalacaktır (Güden, Vakvak , Özkan, Altıparmak, & Dengiz, 2005).

Tabu Arama, Glover (1986) tarafından önerilen ve kendisi tarafından yazılmış bir terimdir. Tabu arama yöntemi problemin ihtiyaçlarına göre geliştirilebilmesi özelliğinden dolayı literatürde farklı uygulamalar bulunmaktadır. Çözüm gelişmiyor olsa bile çözüm uzayında hareket edebilmesi ve yerel en iyi çözümlerin tekrarlanmasını engelleyebilmesi iki tamamlayıcı özelliğidir (Sarıcıoğlu, 2014).

Tabu Arama algoritmasının temel prensibi her iterasyonda değerlendirme fonksiyonuna ait en yüksek değerlendirme değerinin yer aldığı hareketin bir sonraki çözümü oluşturmak amacıyla seçilmesidir. Oluşturulan tabu listesinin asıl amacı ise hareket tekrarını engellemekten ziyade tersine ilerlenmesini engellemektir. Tabu listesi kronolojik bir yapıya sahiptir ve esnek bir hafıza yapısı kullanır. Tabu arama algoritması her ne kadar istenmeyen noktaların işaretlenmesi olarak açıklanmış olsa da daha cazip noktaların işaretlenmesi olarak ta kullanılır (Akca, 2015). Şekil 4.2.'de algoritma adımları yer almaktadır.

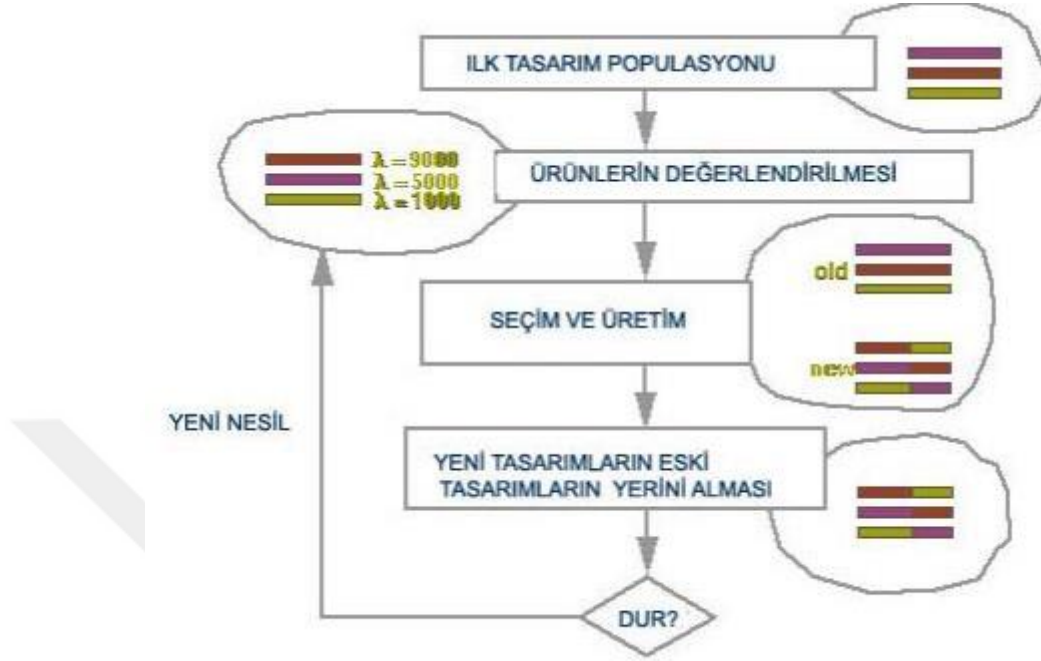


Şekil 4.2. Tabu arama Algoritması (Akca, 2015)

Bir başlangıç çözüm ile aramaya başlayan algoritma her iterasyonda tabu olmayan bir hareket ile mevcut çözümde yer alan komşular içerisinde bir tanesini seçer ve değerlendirme yapar. Eğer amaç fonksiyonunun değerinde bir iyileştirme sağlanıyorsa komşu çözüm, mevcut çözüm olarak ele alınır. Seçilen bir hareket tabu olmasına rağmen tabu yıkma kriterlerini sağlıyorsa, mevcut çözümü oluşturmak için uygulanabilir. Bazı hareketler tabu listesine kaydedilerek tekrar yapılması belirli bir süre için yasaklanır bu şekilde geriye dönüşler engellenmiş olur. Algoritmanın çalışmasının sonlanması ise belirlenen bir durma koşuluna bağlıdır (Akca, 2015).

Genetik algoritma, sezgisel teknik olan bu algoritmanın temelini Holland'ın 1960'lı yılların sonlarına doğru yaptığı çalışmalar oluşturur. Holland öğrenebilen makineler üzerinde yaptığı çalışmalarda öğrenmenin tek bir organizmanın yanı sıra türlerin nesiller boyunca evrimsel uyumu ile gerçekleştiği dikkatini çeker. Genetik algoritmaların gelişim süreci bu duruma dayanmaktadır. Bu algoritmalar, hayatta kalabilen ve özelliklerini yeni nesillere aktarabilen organizmaların davranışlarını taklit etmektedirler.

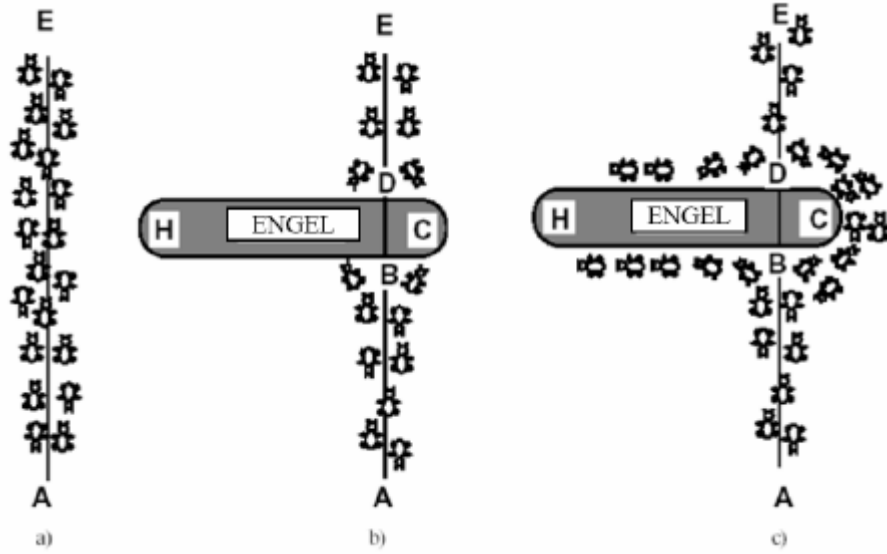
Bu sürecin taklit edilmesinin sebebi, iyi çözümleri çoğaltarak bireylerin güçlü özelliklerinin alınmasıyla daha iyi sonuçlar elde etmektir (Akca, 2015).



Şekil 4.3. Genetik algoritmanın işlem basamakları (Çalışkan, Yüksel, & Dayık, 2016)

Karınca Kolonisi Algoritması (KKA) optimizasyon problemine çözümler bulmak için yapay karınca kullanan metasezgisel bir tekniktir (Yılmaz, 2008). Karıncalar, yuvadan yiyecek kaynağına ya da yiyecek kaynağından yuvaya en kısa yolu bulma doğasına sahiptirler.

En kısa yolu bulmak ve aralarında haberleşmek için karıncalar yola “bir miktar feromon” maddesi bırakırlar. Şekil 4.4a.’da A noktasından E noktasına giden karıncaların engelsiz yolda devam ettikleri görülmektedir. Şekil 4.4b.’de ise yerleştiren engele bağlı olarak karıncalar tercihen H ya da C noktasından E noktasına gitmektedir. Şekil 4.4c.’de ise feromon miktarının çok olduğu C noktasını arkadan gelen karıncalar tarafından yüksek ihtimalle seçilecektir (Tokaylı, 2005).



Şekil 4.4. Karınca davranışları

- a- A-E arasındaki yol karıncaların izledikleri yoldur.
- b- Engel konulan yola geldiklerinde karıncalar hangi yönü seçeceklerine karar verirler.
- c- Daha fazla feromon olan yol daha kısa olan yoldur. (Alaykiran & Engin, 2005)

4.3. Tam Sayılı Doğrusal Programlama

Çözüm yöntemlerinden kesin yöntemlere giren bu yöntem aynı zamanda bu tezin çalışmasını oluşturmaktadır.

Yöneylem araştırması uygulamalarında Doğrusal Programlama problemleri bir doğrusal amaç fonksiyonunun doğrusal eşitlikler ve eşitsizlikler kısıtlamaları ile optimizasyon yapılmasıdır. Model, sürekli değişkenlere ve tek bir doğrusal amaç fonksiyonuna sahipse ve kısıtları doğrusal eşitlik veya eşitsizliklerden oluşturuluyorsa doğrusal (lineer) program olarak adlandırılır (Özçiloğlu, 2009).

Amaç fonksiyonu, kısıtlayıcı fonksiyonlar ve pozitif olma kısıtı olmak üzere DP'nin üç bileşeni bulunmaktadır.

Amaç Fonksiyonu: Doğrusal olan amaç fonksiyonu, kârı en büyükleme ya da maliyeti en küçükleme şeklindedir. Amaç fonksiyonu Z , kontrol edilebilir değişkenler x_j ($j=1,2,$

..., n) ve sabit katsayılar (birim başına kâr ya da birim başına maliyet katsayıları) $c(j)$ ($j=1,2, \dots, n$) olmak üzere;

$$Z = \sum_{j=1}^n c_j x_j \quad (4.1)$$

biçiminde formüle edilir. Bu amaç fonksiyonunun açık yazılmış hali ise şöyledir:

$$Z = c_1x_1 + c_2x_2 + \dots + c_nx_n \quad (4.2)$$

Kısıtlayıcı Fonksiyonlar: İşletme faaliyetleri birtakım kısıtlamalar altındadır. Finansman, iş gücü, zaman, makine vb. gibi şartlar bu kısıtlamalara birer örnektir. Kısıtlar, teknoloji matrisi $a(i,j)$, ihtiyaç vektörü $b(i)$ olmak üzere standart maksimizasyon probleminde;

$$\sum_{j=1}^n a_{ij} x_j \leq b_i \quad (i = 1,2,\dots,m) \quad (4.3)$$

Standart minimizasyon probleminde ise;

$$\sum_{j=1}^n a_{ij} x_j \geq b_i \quad (i = 1,2,\dots,m) \quad (4.4)$$

biçiminde yazılabilirler. Standart DP problemlerinde “ \geq ” ya da “ \leq ” kullanılabilirdiği gibi tam kapasiteyle çalışma durumunda “ $=$ ” kullanılır. Standart olmayan DP problemlerinde söz konusu kısıtların “ \geq ”, “ \leq ” ya da “ $=$ ” işaretleri karışık olarak ta kullanılabilir.

Pozitif Kısıtlama: Negatif üretim ya da negatif maliyet olmayacağından aynı şekilde müşteri talepleri ve karar verme durumları negatif olamayacağından tüm karar değişkenler pozitifdir.

Bu açıklamalardan yola çıkarak bir doğrusal programlama probleminin genel yapısı;

Kâr maksimizasyonunda;

Amaç fonksiyonu;(maksimum)

$$Z = \sum c_j x_j \quad (j = 1, 2, \dots, n) \quad (4.5)$$

Kısıtlayıcılar;

$$\sum_{j=1}^n a_{ij} x_j \leq b_i \quad (i = 1, 2, \dots, m; j = 1, 2, \dots, n) \quad (4.6)$$

Pozitif kısıtlama;

$$x_j \geq 0 \quad (j = 1, 2, \dots, n) \quad (4.7)$$

Maliyet Minimizasyonu;

Amaç fonksiyonu;(minimum)

$$Z = \sum c_j x_j \quad (j = 1, 2, \dots, n) \quad (4.8)$$

Kısıtlayıcılar;

$$\sum_{j=1}^n a_{ij} x_j \geq b_i \quad (i = 1, 2, \dots, m; j = 1, 2, \dots, n) \quad (4.9)$$

Pozitif kısıtlama;

$$x_j \geq 0 \quad (j = 1, 2, \dots, n) \quad (4.10)$$

biçiminde verilir.

Yukarıda genel matematiksel modeli verilen doğrusal programlama modeli daha açık biçimde aşağıdaki gibi yazılabilir.

Amaç fonksiyonu:

$$Z = c_1x_1 + c_2x_2 + \dots + c_nx_n \quad (4.11)$$

Kısıtlayıcılar: (4.12)

$$\begin{aligned} a_{11}x_{11} + a_{12}x_{12} + \dots + a_{1n}x_{1n} &\leq b_1 \\ a_{21}x_{21} + a_{22}x_{22} + \dots + a_{2n}x_{2n} &= b_2 \\ \dots & \\ a_{m1}x_{m1} + a_{m2}x_{m2} + \dots + a_{mn}x_{mn} &\geq b_m \end{aligned}$$

Pozitif Kısıtlama: (4.13)

$$x_j \geq 0 \quad (j = 1, 2, \dots, n)$$

Kâr maksimizasyonu olan bu modelde kısıtlayıcı eşitsizliklerin sağ tarafındaki “ \leq ” işareti yerine “ \geq ” işareti yazılırsa maliyet minimizasyonunun matematiksel modelini elde etmek için elde edilmiş olur.

Problemlerin türlerine göre değişik yöntemler bulunmaktadır. Doğrusal Programlara (DP)’da kısıtlar çerçevesinde amaç fonksiyonu çalışır. Doğrusal Programlamanın tercih edilmesinin önde gelen sebeplerinden biri, analiz aşamasındaki formülasyonudur. Diğer bir tercih sebebi, amaçların ve kısıtların en az zorlukta tanımlanmasıdır (Luenberger & Ye, 1984).

DP modelinin uygulanmasında, çözümün – örneğin araç sayısı, karar verme gibi-tamsayı olması gereken durumlar vardır. Bu tür problemlerin çözümü için kullanılan model Tam Sayılı Doğrusal Programlama (TDP) modelidir. TDP, değişkenlerin bazılarının ya da hepsinin pozitif tamsayı değeri almasını hedefler.

Amaç fonksiyonu ve sınırlar olmak üzere iki bileşenden oluşan doğrusal programlamada, zaman, maliyet ya da tezgâhta geçen boş sürenin minimize edilmesi,

aynı şekilde problemin türüne bağlı olarak da trafik süresinin minimize edilmesi gibi amaç fonksiyonları bulunurken, kapasite kullanımının ya da toplam karın maksimize edilmesi gibi amaç fonksiyonları da bulunmaktadır. Kısıtlarda ise makine kapasitesi, malzeme miktarı, depo, rota uzunluğu, araç kapasitesi, müşteri talebi, sermaye gibi kısıtlı kaynaklar bulunmaktadır.

Bu kısıtlarda önemli olan nokta şudur ki, makine gibi varlıklar tamsayı ile ifade edilir. Örneğin; 3 makine, 5 makine vb. Yine aynı şekilde kullanılan araç sayısı 2,75 olamaz. Benzer olarak da taleplerin bütünlüğü bölünemez, araç rotalama problemleri olarak düşünüldüğünde 35 kişi bulunan bir duraktan 25,75 kişi alınamaz. Bu tür durumlar tamsayılı doğrusal programlamaya örnektir. Tamsayılı doğrusal programlama modelleri pozitiflik durumuna göre dört sınıfa ayrılmaktadır. Tüm karar değişkenlerinin tamsayılı pozitif olduğu Saf pozitif, bazılarının tamsayılı değişken bazılarının pozitif olduğu karma pozitif, tüm değişkenlerin karar verici olduğu saf sıfır-bir ya da karışık olan karma sıfır-bir tamsayılı doğrusal programlama bulunmaktadır (Şahin, 2015).

BÖLÜM 5. LİTERATÜR TARAMASI

1959 yılında ilk kez Dantzig ve Ramser tarafından önerilen ARP için günümüzde pek çok sayıda çalışma değişik yöntem ve uygulamalarla karşımıza çıkmaktadır. Yakın zamanda yapılmış bazı çalışmalardan bu bölümde bahsedilmiştir.

Bowerman v.d. (1995) okul servis araçlarının çok amaçlı optimizasyonu üzerine çalışmışlardır. Çalışmada önce grupla sonra rotala prensibiyle problem ele alınmıştır (Bowerman, Hall, & Calamai, 1995).

Kara ve Bektaş (2006) yan koşul olarak gezicinin minimum gitmesi gereken düğüm sayısını ekledikleri çalışmada Tam Sayılı Doğrusal Programlama ile çözüme gitmişlerdir (Kara & Bektaş, 2006).

Miori (2010) yükleme aracı rotalama problemi üzerinde çalışmıştır. Çalışmada deterministik çok amaçlı formülasyon kullanılmıştır. Hedef Programlama ve Tabu Arama yöntemleriyle rotalama çözümü geliştirilmiştir.

Maden v.d. (2010), toplam mesafeyi minimize eden sezgisel bir algoritma üzerinde çalışmışlardır. Uygulamada araç rotalarının uzunluğu trafiğin yoğunluğuna bağlı olarak ele alınmıştır (Maden, Eglese, & Black, 2010).

Desaulniers (2010) çalışmasında bölünmüş dağıtım Zaman Pencere Araç Rotalama Problemini ele almıştır. Araç rotalarının maliyetinin minimizasyonu amaçlanmıştır. Yeni bir Dal-Değer -ve Kesme metodu ile çözüme gidilmiştir (Desaulniers, 2010).

Çetin ve Gencer (2011), Heterojen Araç Filolu Zaman Pencere Eş Zamanlı Dağıtım – Toplamalı Araç Rotalama Problemi için çalışma yapmış. Bilinen amaç

fonksiyonlarından farklı olarak bekleme süresini en azlayan bir amaç fonksiyonu için çözüme gitmişlerdir (Çetin & Gencer, 2011).

Archetti v.d. (2011) Dal ve Değer ve Kesme Algoritması ile bölünmüş dağıtım Zaman Pencereli ARP çözümü için çalışma yapmışlardır. Burada müşterilere birden fazla araç uğrayabilmektedir (Archetti, Bouchard, & Desaulniers, 2011).

Güvez v.d. (2012) Kırıkkale’de faaliyet gösteren atık toplayıcı bir işletmenin tıbbi atık toplama araçları için rotalama çalışması yapmıştır. Tam sayılı doğrusal programlama modeli kullanılmıştır (Güvez, Dege, & Eren, 2012).

Atmaca (2012), bir kargo şirketi için eş zamanlı dağıtım toplamalı araç rotalama problemi üzerine çalışma yapmıştır. Problem GAMS ile çözülmüştür (Atmaca, 2012).

Koç ve Karaoğlan (2012), Çok Kullanımlı ve Zaman Pencereli Araç Rotalama Problemi (ÇKZPARP) için karma Tam Sayılı doğrusal programlama modelini ele almışlardır (Koç, 2012).

Zachariadis v.d. (2012), palet yükleme arp diye adlandırılan yeni bir ulaştırma problemi üzerine çalışma yapmışlardır. Yükleme kısıtlarının da yer aldığı, yüklemenin araca direkt yapılmayıp paletlerle yapıldığı bu problemin optimal çözümü çok zor olduğundan yasak arama sezgisel yöntemi ile çözüme gidilmiştir (Zachariadis, Tarantilis, & Kiranoudis, 2012).

Lee v.d. (2012), araç kapasitesi ve müşteri talep teslim süreleri dikkat alınarak toplam mesafeyi minimize etmeyi amaçlayan bir çalışma yapmışlardır. Bu problemlerde müşteri talep ve iki müşteri arasındaki mesafe belirsiz olarak ele alınmıştır. Belirsiz veri seti için Dantzig-Wolfe Ayrışım yaklaşımı, alt problem çözümü içinse dinamik programlama önerilmiştir.

Vansteenwegen v.d. (2012) çalışmalarında otel seçimli GSP için sezgisel yöntemle çözüme gitmişlerdir. İki yapıcı başlangıç prosedürü ve bir geliştirme prosedürü ile

birlikte Komşu Arama yöntemi kullanılmıştır CPLEX (10.0) ile problem çözümü gerçekleştirilmiştir.

Wang v.d. (2014) çok seferli zaman pencereli araç rotalama problemi için rota havuzuna dayalı bir sezgisel yöntem geliştirmişlerdir. Söz konusu çalışmada araçlar belirlenmiş sürede müşteriyi ziyaret edecek şekilde birden fazla sefer yapabilirler. Çok katmanlı bir çözüm yapısına sahip problemin çözümünde, öncelikle araçların gidebileceği rotalar havuza atılıyor ve oradan bazı rotalar seçilerek ve birleştirilerek araç çalışma çizelgesi oluşturulmuştur.

Hezer ve Kara (2013), çalışmalarında Eşzamanlı Dağıtım ve Toplamalı Araç Rotalama Problemini ele almışlardır. Müşterilerin dağıtım ve toplama talepleri eşzamanlı olarak karşılanan bu problem için, Bakteriyel Besin Arama Optimizasyonu Algoritması (BBAOA) tabanlı yöntem geliştirmişlerdir, sezgisel bir çözüm olan yöntemin performansı değerlendirilmiştir (Hezer & Kara, 2013).

Martinez ve Amaya (2012) çalışmalarında çok seferli zaman pencereli ARP için iki aşamalı bir modelle çözüme gitmişlerdir. İlk aşamada Sıralı Ekleme Algoritması'nın geliştirilmiş bir modeli ve ikinci aşamada Tabu Arama yöntemini kullanmışlardır (Martinez & Amaya, 2013).

Bozyer v.d. (2014) çalışmalarında önce grupta sonra rotalara prensibine sahip sezgisel bir yöntemi kapasite kısıtlı araç rotalama problemi (KKARP) için önermişlerdir. Gruplamada, bulanık c-ortalama kümeleme yöntemi ile taleplerin yer aldığı noktaların olası tüm rotalara 0-1 arasında üyelik dereceleri hesaplanmıştır. Rotalamada ise Tabu Arama prensiplerine dayanan bir arama algoritması ile rotalar iyileştirilmeye çalışılmıştır (Bozyer, Alkan, & Fırlı, 2014).

Chu v. d. (2015) çalışmalarında, stokastik seyahat sürelerine sahip günlük envanter ikmal için esnek zaman pencereli çok seferli bölünmüş dağıtım araç rota problemini ele almışlardır. Çalışmada ceza ve ödül verilerek olası araç varış ve ürün teslimleri

kategorize edilmiştir. İki aşamalı sezgisel çözüm önerilmiştir (Chu, Yan, & Huang, 2017).

Serap Ercan Cömert v. d. (2017) çalışmasında bir süpermarket zincirinin belirli zaman aralıklarında servis gören müşterilerinin taleplerinin karşılanmasında ortaya çıkan Sıkı Zaman Pencereli Araç Rotalama Problemi (SZPARP)'nin çözümü yapılmıştır. Önce kümele sonra rotala yaklaşımına dayanan iki aşamalı hiyerarşik bir yöntem önerilmiştir. İlk aşamada müşteriler K-medoids ve DBSCAN kümeleme algoritmaları kullanılarak araçlara atanmıştır. İkinci aşamada ise rotalama problemi MILP ile çözülmüştür. Çalışmanın en önemli katkısı olarak önerilen yöntem büyük boyutlu gerçek problemler ele alınırken kesin çözüm yöntemlerini kullanmamıza olanak sağlaması olduğunu belirtmişlerdir. İki algoritmanın sonuçları ve firmadan alınan gerçek sonuçlar ANOVA ile karşılaştırılmıştır. Test sonucuna göre DBSCAN'ın daha iyi sonuç verdiği görülmüştür (Ercan Cömert, Yazgan , Sertvuran, & Şengül, 2017).

Çelikkanat Filiz ve Eroğlu (2017) çalışmasında bulanık hedef programlama yaklaşımı ele almışlardır. Kapasite kısıtlı araç rotalama probleminde uygulama yapılmıştır. Öneri olarak bulanık hedef programlama yaklaşımı verilmiştir. Örnek sayısal verilerle önerilen modelde uygulama yapılmıştır. GAMS programı ile elde edilen sonuçlara göre her hafta en az 6 araçla gönderim yapan şirket maliyetinin, önerilen modelle düşürülebileceği belirtilmiştir (Çelikkanat Filiz & Eroğlu, 2017).

Atan ve Şimşek (2017) araç atama problemini Doğrusal Programlama ile çözmüşlerdir. Toplam mesafe ve süreyi en küçükleyerek araçların depoya geri dönmesi amaçlanmıştır (Atan & Şimşek, 2017).

Bektaş ve Elmastaş (2017) okul servislerinin rotalamasında Doğrusal Programlama kullanarak çözüme gitmişlerdir. Kapasite ve mesafe kısıtlı olan bu çalışmada ARP türü aynı zamanda açık uçlu ARP dir (Bektaş & Elmastaş, 2017).

Ünsal ve Yiğit (2018), Okul Servisi Rotalama problemini ele almışlardır. Kümeleme teknikleri ve yapay zekâ yöntemleri kullanılarak, OSRP'nin optimizasyonu için GIS,

GPS araçları ve mobil uygulama desteğiyle bir yazılım geliştirilmiştir. Söz konusu yazılım Ankara'da bulunan servis firmalarından toplanan rota verileri üzerinde uygulanmıştır. Elde edilen deneysel sonuçlarda mesafe, zaman ve rakım değişimi parametreleri açısından rotaları başarılı bir şekilde iyileştirilebileceğini belirtmişlerdir (Ünsal & Yiğit, 2018).

Yazgan ve Büyükyılmaz (2018), bir firmanın depo ve müşterileri arasında hizmet sağlan araçların rotalama problemini ele almıştır. Amaç, minimum sayıda araç ile gidilen mesafeyi minimize etmektir. Eş zamanlı topla dağıt araç rotalama problemi olan çalışmada karışık Tam Sayılı matematiksel model kullanılmış ve sezgisel bir algoritma geliştirilmiştir. Farklı büyüklükteki veri setlerine algoritma uygulanmış elde edilen çözümler regresyon analizi ile değerlendirilmiştir (Yazgan & Gökçen Büyükyılmaz, 2018).

Pala ve Aksaraylı (2018), ulaştırma sektöründe bir firmanın tur sürelerinin toplamını ve bir yolcu için ulaşımda geçen ortalama süresini minimize eden bir çalışma yapmışlardır. Çalışmayı, araçların yolcu taşıma kısıtı nedeniyle Çok Amaçlı Kapasite Kısıtlı Araç Rotalama Problemi olarak ele almışlardır. Karınca Kolonisi Algoritması ile çözüme gidilen çalışmada her iki parametre için de (toplam tur süresi ve bir yolcunun ulaşımda geçirdiği ortalama süre) önemli iyileşmeler sağlandığını belirtmişlerdir (Pala & Aksaraylı, 2018).

BÖLÜM 6. UYGULAMA

Bu bölümde problemin tanımı, mevcut durum, probleme ait verilerden bahsedildikten sonra problemin matematiksel modeli, LINGO 18.0 programından elde edilen çözümler ve sonuçlar yer almaktadır.

6.1. Problemin Tanımı

Tez uygulaması kapsamında Ankara'da bir firmanın servis araçlarının toplamda minimum mesafe amaçlanarak rotalanması üzerinde çalışılmıştır. Bu çalışma ile amaçlanan; Tamsayılı Doğrusal Programlama yöntemi kullanılarak uygulama yapılan problem başlangıç ve bitiş noktasının aynı olması, duraklar arası mesafenin simetrik olması ve kapasite kısıtı bulunması özelliklerinden dolayı ARP içerisindeki Kapalı Uçlu Kapasite Kısıtlı Simetrik türde ARP'dir.

Firmada vardiyalı çalışanlar olduğu gibi, mesai saatlerinde çalışanlar da bulunmaktadır. Ankara ili içerisindeki semtlere dağılım yapan servis araçları için, vardiyalı çalışanların kullandığı servisler çalışmaya dahil edilmemiştir. Servis yoğunluğu mevsime göre değişkenlik gösterse de servis kullanabilecek personel sayısı belli olduğundan mevsimsel durum göz önüne alınmamıştır.

Mevcut uygulamada servis araçları firmadan hareket eder, duraklara uğrayarak personeli alır ve tekrar firmaya döner. Mesai bitiminde yine aynı şekilde servis araçları personeli duraklara bırakır ve firmaya geri döner.

Servis güzergâhına 800 metreden uzak olan ya da servisin uğramadığı mahallede/sokakta olan personel servis kullanamadığı için yol ücreti almaktadır.

Servis araçları 18 kişilik kapasiteye sahiptir. Sefer mesafesi 60 km'yi aşmaması gerekmektedir. Bu uygulamada Yeni Mahalle, Batıkent ve Keçiören semtlerine ait servis araçları ele alınmıştır. Adı geçen semtlerde oturan personel sayısı 37 dir. Araç maliyetleri minibüs için 225,00 TL/Gün ve küçük otobüs için 281,25 TL/Gün'dür. Aylık maliyet, 20 gün üzerinden hesaplanmıştır.

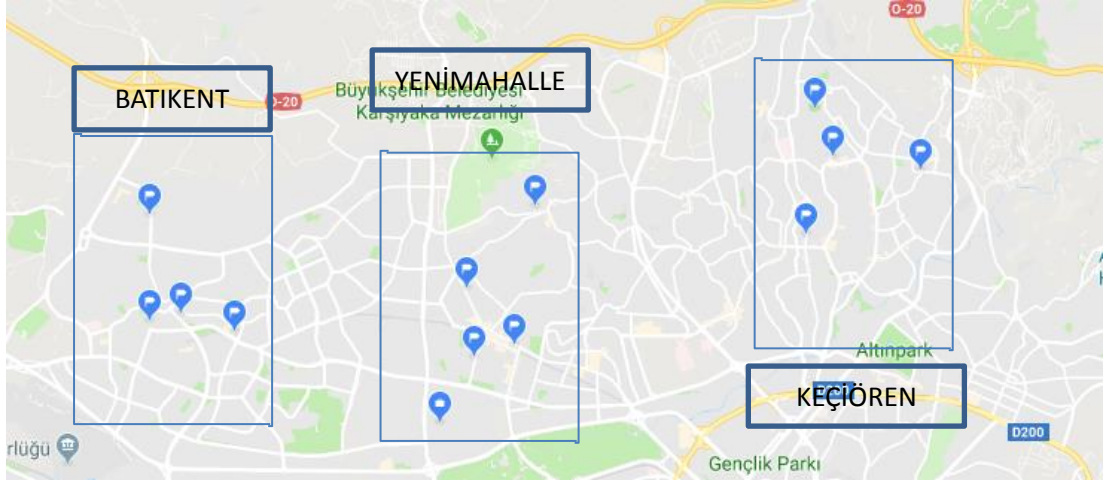
Problem verilerine ait bilgiler genel olarak Tablo 6.1.'de yer almaktadır.

Tablo 6.1. Probleme ait veriler

Bölge sayısı	3
Personel Sayısı	37 kişi
Sefer uzunluğu	60 km
Araç Kapasitesi	18 kişi
Araç Maliyeti Minibüs	225,00 TL
Araç Maliyeti- Minibüs	281,25 TL

Araçların uğradığı duraklar Google Maps'te işaretlenmiş ve Şekil 6.1.'de gösterilmiştir. Ek 1'de daha ayrıntılı harita bulunmaktadır. Duraklar arası mesafeler Tablo 6.3.'te yer almaktadır.

Duraklar arası mesafenin ölçümü için Google Maps kullanılmıştır. Google Maps, mesafe hesaplarken sadece en kısa yol algoritması kullanmaz aynı zamanda trafik koşulları, trafik kuralları, yol ağı hiyerarşisi gibi unsurları da dikkate almaktadır. A*, Dijkstra gibi algoritmaların kullanılarak en kısa yolu tespit eder. (www.quora.com, 14.04.2019)



Şekil 6.1. Durakların harita görünümü

Mevcut uygulamada servislerin uğradıkları durak ve yolcu sayıları Tablo 6.2.'de verilmiştir. Duraklar arası Uzaklık Matrisi metre olarak Tablo 6.3.'te yer almaktadır.

Tablo 6.2. Duraklar ve personel sayıları

KEÇİÖREN	Durak 2	Süleymaniye Camisi	5 kişi
	Durak 3	Meslek Hastanesi	1 kişi
	Durak 4	Gazino Durağı	2 kişi
	Durak 5	Tepebaşı	3 kişi
	Durak 6	Şentepe	3 kişi
YENİ MAHALLE	Durak 7	İvedik Caddesi	4 kişi
	Durak 8	Demetevler 1. Cad.	2 kişi
	Durak 9	Demetevler 12. Cad.	3 kişi
	Durak 10	Çakırlar	3 kişi
BATIKENT	Durak 11	Atlantis Önü	5 kişi
	Durak 12	Gimsa	3 kişi
	Durak 13	Mesa Postanesi	3 kişi

Tablo 6.3. Uzaklık matrisi

	FRM	DURAK 1	DURAK 2	DURAK 3	DURAK 4	DURAK 5	DURAK 6	DURAK 7	DURAK 8	DURAK 9	DURAK 10	DURAK 11	DURAK 12
FRM	0	21300	22900	21800	19600	20000	18800	17600	17700	23800	21500	20700	22700
DURAK 1	21300	0	2900	2100	3300	10200	18200	11300	12700	25400	19000	17800	21800
DURAK 2	22900	2900	0	1400	3000	7600	12700	9700	14400	17300	20800	17800	22500
DURAK 3	21800	2100	1400	0	1900	7000	10500	11000	12400	16900	19800	16400	21500
DURAK 4	19600	3300	3000	1900	0	7300	8900	9200	10500	18000	15800	14600	16400
DURAK 5	20000	10200	7600	7000	7300	0	2600	4200	4300	8600	8500	7400	9200
DURAK 6	18800	18200	12700	10500	8900	2600	0	1600	1500	7200	6100	4900	6800
DURAK 7	17600	11300	9700	11000	9200	4200	1600	0	1400	8800	7600	6500	9200
DURAK 8	17700	12700	14400	12400	10500	4300	1500	1400	0	8700	6400	5300	7100
DURAK 9	23800	25400	17300	16900	18000	8600	7200	8800	8700	0	2900	3800	2400
DURAK 10	21500	19000	20800	19800	15800	8500	6100	7600	6400	2900	0	2300	1000
DURAK 11	20700	17800	17800	16400	14600	7400	4900	6500	5300	3800	2300	0	1800
DURAK 12	22700	21800	22500	21500	16400	9200	6800	9200	7100	2400	1000	1800	0

6.2. Problemin Varsayımları

Mevcut durumda, firma servis hizmetini verecek yeterli araca sahiptir. Servislerin hareket yeri merkez yani firmadır. Servis kullanan personel belli olduğu için duraklar da bellidir. Kapalı Uçlu Kapasite Kısıtlı Simetrik ARP olarak ele alınan problemin uygulanmasına dair şu şekildedir;

- Servis hizmeti, merkezi yerden, firmadan sağlanmaktadır.
- Servise binecek duraklar ve personel sayısı bellidir.
- Araçlar aynı tip ve özdeş kapasitelidir.
- Yeterli sayıda servis aracı bulunmaktadır.
- Trafik durumu, yol koşulları, araç arızası, sürücüden kaynaklanabilecek olumsuz durumların olmadığı varsayılmaktadır.
- Servise binme, inme vb. zamanla ilgili kısıtlar yer almamaktadır.
- Duraklar arası uzaklıklar simetriktir.

Buna göre yukarıdaki varsayımlar dikkate alınarak aşağıdaki şartlar yerine getirilmelidir:

- Her aracın rotası firmada başlayıp firmada sonlanmalıdır.
- Her rota üzerinde sadece bir araç bulunmalıdır.
- Bir rotada bir müşteri sadece bir defa ziyaret edilmelidir.
- Her rotada aracı kullanan toplam personel sayısı araç kapasitesini aşmamalıdır.
- Tüm duraklardaki personel, araçları kullanabilmelidir.

6.3. Matematiksel Model

Tamsayılı Doğrusal Programlama yönteminin kullanıldığı bu problem için parametreler ve değişkenler belirlenerek yöntemin matematiksel modeli öncelikle kurulmalıdır. Problemin amaç fonksiyonu; servis araçlarının toplam mesafesinin minimize edilmesidir. Problemin kısıtları; her durağa sadece bir aracın gelmesi, duraktaki tüm yolcunun araca binmesi ve araç kapasitesidir.

Modele ait parametreler ve karar deęişkenleri ařaęıda verilmiřtir.

Parametreler:

M : Durak sayısı olup, $M = 12$ olarak alınmıřtır.

Q : Her bir aracın (ortak) kapasitesi (kiři) olup, farklı senaryolar için $Q = 18, 22$ ve 24 kiři alınmıřtır.

T : Her bir aracın kat edebileceęi en büyük uzaklık (km) olup, farklı senaryolar için $T = 60$ km ve 50 km alınmıřtır.

k : Kullanılan araç sayısı (adet) olup, farklı senaryolar için $k = 2$ veya 3 alınmıřtır.

$d(ijk)$: i noktasından j noktasına olan en kısa uzaklık (m) ($i \neq j$)

Karar Deęişkenleri:

$$x(ijk) : \begin{cases} 1, k. \text{ servis aracı } i. \text{ düęümünden } j. \text{ düęüme giderse;} \\ 0, \text{ dięer durumda} \end{cases}$$

$$y(ik) : \begin{cases} 1, i. \text{ durak } k. \text{ araç tarafından ziyaret edilirse;} \\ 0, \text{ dięer durumda} \end{cases}$$

řeklinde tanımlanmaktadır.

Ařaęıda modelde tüm araçlar tarafından kat edilecek toplam mesafeyi gösteren amaç fonksiyonu ve kısıtlar verilmiřtir.

Amaç fonksiyonu

$$\text{Min } Z = \sum_{i=0}^N \sum_{j=0, j \neq i}^N \sum_{k=1}^M d_{ij} X_{ijk} \quad (6.1)$$

Kısıtlar;

$$\sum_{k=1}^M \sum_{j=1}^N X_{ijk} = M \quad i=0 \quad (6.2)$$

$$\sum_{k=1}^M \sum_{x=1}^N X_{ijk} = M \quad j=0 \quad (6.3)$$

$$\sum_{k=1}^M \sum_{j=0}^N X_{ijk} = 1 \quad i \in \{1, \dots, N\} \quad (6.4)$$

$$\sum_{k=1}^M \sum_{i=0, i \neq j}^N X_{ijk} = 1 \quad j \in \{1, \dots, N\} \quad (6.5)$$

$$u_i - u_j + Qx_{ij} + (Q - q_i - q_j)x_{ji} \leq Q - q_j \quad i \neq j \in \{1, \dots, N\} \quad (6.6)$$

$$u_i \geq q_i \quad i \in \{1, \dots, N\} \quad (6.7)$$

$$u_i - q_i x_{0i} + Qx_{0i} \leq Q \quad i \in \{1, \dots, N\} \quad (6.8)$$

$$\begin{aligned} v_i - v_j + (T - d_{id} - d_{0j} + d_{ij})x_{ij} + (T - d_{id} - d_{0j} + d_{ij})x_{ij} \\ \leq T - d_{id} - d_{0j} \end{aligned} \quad (6.9)$$

$$v_i - d_{0i}x_{0i} \geq 0 \quad (6.10)$$

$$v_i - d_{0i}x_{0i} + Tx_{0i} \leq T \quad (6.11)$$

$$x_{ij} \in \{0,1\} \quad (6.12)$$

Denklem 6.1.'deki denklem, yukarıda da bahsedildiği gibi kat edilen toplam mesafeyi gösteren amaç fonksiyonudur. Denklem 6.2. ve Denklem 6.3 araçların firmadan hareket edip firmaya dönmesini sağlar. Denklem 6.4 ve Denklem 6.5, her durağa bir kez uğranılmasını sağlar. Denklem 6.6, Denklem 6.7 ve Denklem 6.8 kapasiteyle ilgili kısıtlardır. Araç kapasitesini aşmayan duraklarla rota çizilmesini sağlar. Aynı zamanda alt tur engelleyeci kısıtlardır. Denklem 6.9, Denklem 6.10 ve Denklem 6.11 ise mesafeyle ilgili kısıtlardır. Maksimum rota uzunluğunun aşılmasını sağlar. Denklem 6.12 ise karar değişkeni x_{ij} sıfır veya bir değerini almasını sağlar.

BÖLÜM 7. ARAŞTIRMA SONUÇLARI

Problemin çözümü ele alınırken araç kapasitesi, süre gibi bazı hususlar ele alınarak çözümler incelenmiştir. Kısıtlardan biri olan araç kapasitesini 22 kişi ve 24 kişi kapasiteli olarak da hesaplama yapılmış, bir diğer senaryoda süreyi değerlendirme olarak rota uzunluğu ele alınmıştır. Son olarak karma bir modele gidilerek araç kapasitesi 18 ve 24 kişi olan araçlarla da hesaplama yapılmıştır.

Araç kapasitesinin arttırılması araç sayısına etki edeceğinden maliyeti düşürebilir ancak rota uzunluğunun belli bir mesafeye kadar olması bu durumu sınırlayacaktır. Aynı şekilde araç sayısının azalması rotanın uzunluğunu, dolayısıyla personelin serviste geçireceği süreyi etkileyecektir.

Model LINGO 18.0 da çözülmüştür. Ayrıntılı model Ek 1’de yer almaktadır. Çözüme ait sonuçlar ilgili başlıkta yer verilmiştir.

- Araç kapasitesinin 18 kişi olduğu durum:

Araç kapasitesinin 18 kişi olduğu durumda oluşan rotaların çözümü Şekil 7.1.’de gösterilmiştir. 3 rota oluşmuştur. Toplam gidilen mesafe 139,5 km’dir.

```

Global optimal solution found.
Objective value:                139500.0
Objective bound:                139500.0
Infeasibilities:                0.000000
Extended solver steps:         1895
Total solver iterations:       32648
Elapsed runtime seconds:       3.48

```

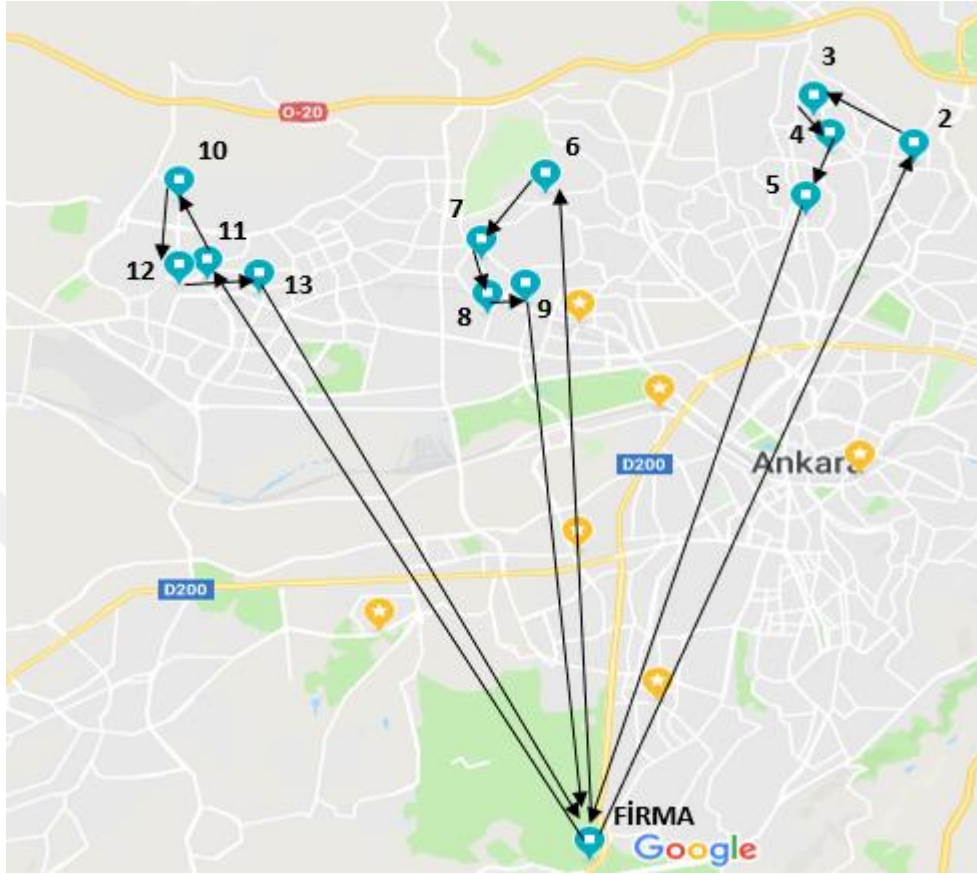
Variable	Value	Reduced Cost
X(FRM, 2)	1.000000	21300.00
X(FRM, 6)	1.000000	20000.00
X(FRM, 11)	1.000000	21500.00
X(2, 3)	1.000000	2900.000
X(3, 4)	1.000000	1400.000
X(4, 5)	1.000000	1900.000
X(5, FRM)	1.000000	19600.00
X(6, 7)	1.000000	2600.000
X(7, 8)	1.000000	1500.000
X(9, FRM)	1.000000	17600.00
X(8, 9)	1.000000	1400.000
X(10, 12)	1.000000	2400.000
X(11, 10)	1.000000	2900.000
X(13, FRM)	1.000000	20700.00
X(12, 13)	1.000000	1800.000

Şekil 7.1. LINGO 18.0 çözüm sonucu

Şekil 7.1.'de gösterilen sonuçlara göre;

- Birinci rota durak sırası; Süleymaniye Camisi, Meslek Hastanesi, Gazino, Tepebaşı olup toplam mesafe 47,1 km,
- İkinci rota durak sırası; Şentepe, İvedik Caddesi, Demetevler 1. Cadde, Demetevler 12. Cadde olup toplam mesafe 43,1 km,
- Üçüncü rota durak sırası; Atlantis Önü, Çakırlar Gimsa, Mesa Postanesi olup toplam mesafe 49,3 km olarak hesaplanmıştır.

Rotaların haritadaki görünümü Şekil 7.2.'de yer almaktadır.



Şekil 7.2. Araç rotalarının haritada görünümü

- Araç kapasitesinin 22 kişi olduğu durumda;

Araç kapasitesinin 22 kişi olduğu durumda araç sayısı ikiye düşmüştür. Rotalama Batıkent ve Keçiören servisi için aynı kalırken Yeni Mahalle de yer alan duraklar adı geçen servisler arasında dağıtılmış, duraklar daha çok Keçiören'e dahil olmuştur. Toplam mesafe 107,400 km dir. LINGO 18.0 çözüm sonuçları Şekil 7.3.'de yer almaktadır.

Buna göre rotalar;

- Birinci rota durak sırası; Tepebaşı, Süleymaniye Camisi, Gazino Durağı, Meslek Hastanesi, Şentepe, İvedik, Demetevler 12. Cadde olup toplam mesafe 55,8 km,
- İkinci rota durak sırası; Atlantis, Çakırlar, Gimsa, Mesa Postanesi ve Demetevler 1. Cadde olup toplam mesafe 51,6 km olarak hesaplanmıştır.

```

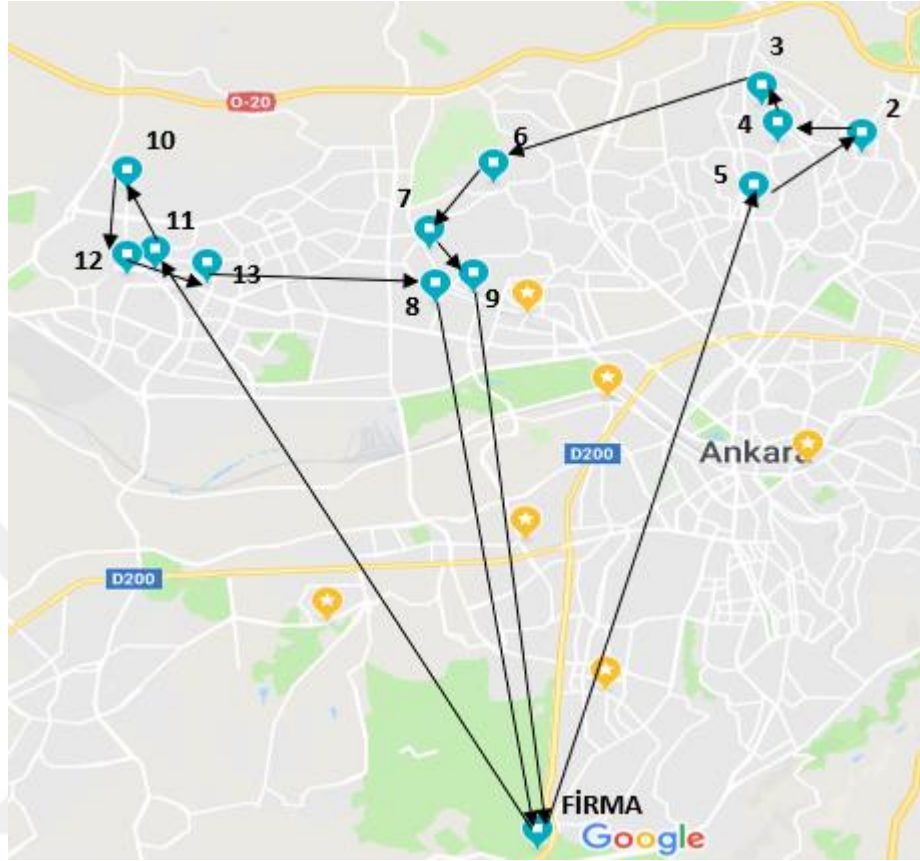
Global optimal solution found.
Objective value:                107400.0
Objective bound:                107400.0
Infeasibilities:                0.000000
Extended solver steps:         4020
Total solver iterations:       67104
Elapsed runtime seconds:       6.11

```

Variable	Value	Reduced Cost
X(FRM, 5)	1.000000	19600.00
X(FRM, 11)	1.000000	21500.00
X(2, 4)	1.000000	2100.000
X(3, 6)	1.000000	7600.000
X(4, 3)	1.000000	1400.000
X(5, 2)	1.000000	3300.000
X(6, 7)	1.000000	2600.000
X(7, 9)	1.000000	1600.000
X(9, FRM)	1.000000	17600.00
X(8, FRM)	1.000000	17700.00
X(10, 12)	1.000000	2400.000
X(11, 10)	1.000000	2900.000
X(13, 8)	1.000000	5300.000
X(12, 13)	1.000000	1800.000

Şekil 7.3. LINGO 18.0 çözüm sonucu

Rotaların haritadaki görünümüne Şekil 7.4.'te yer verilmiştir.



Şekil 7.4. Araç rotalarının haritada görünümü

- Araç kapasitesinin 24 kişi olduğu durumda;

Bu kapasitede araç sayısı 2 olmakla birlikte rotalamada farklılık olmuştur. Batıkent rotalaması yine aynı kalırken, Yeni Mahalle Keçiören durakları diğer rotayı oluşturmuştur. Toplam mesafe 106,4 km'dir. Lingo 18.0 çözüm sonucu Şekil 7.5.'de gösterilmiştir.

Buna göre rotalar;

- Birinci rota durak sırası; Tepebaşı, Süleymaniye Camisi, Gazino Durağı, Meslek Hastanesi, Şentepe, İvedik Cad, Demetevler 1. cad, Demetevler 12. Cadde olup toplam mesafe 57,1 km,

- İkinci rota durak sırası; Atlantis önü, Çakırlar, Gimsa, Mesa Postanesi olup toplam mesafe 49,3 km olarak hesaplanmıştır.

```

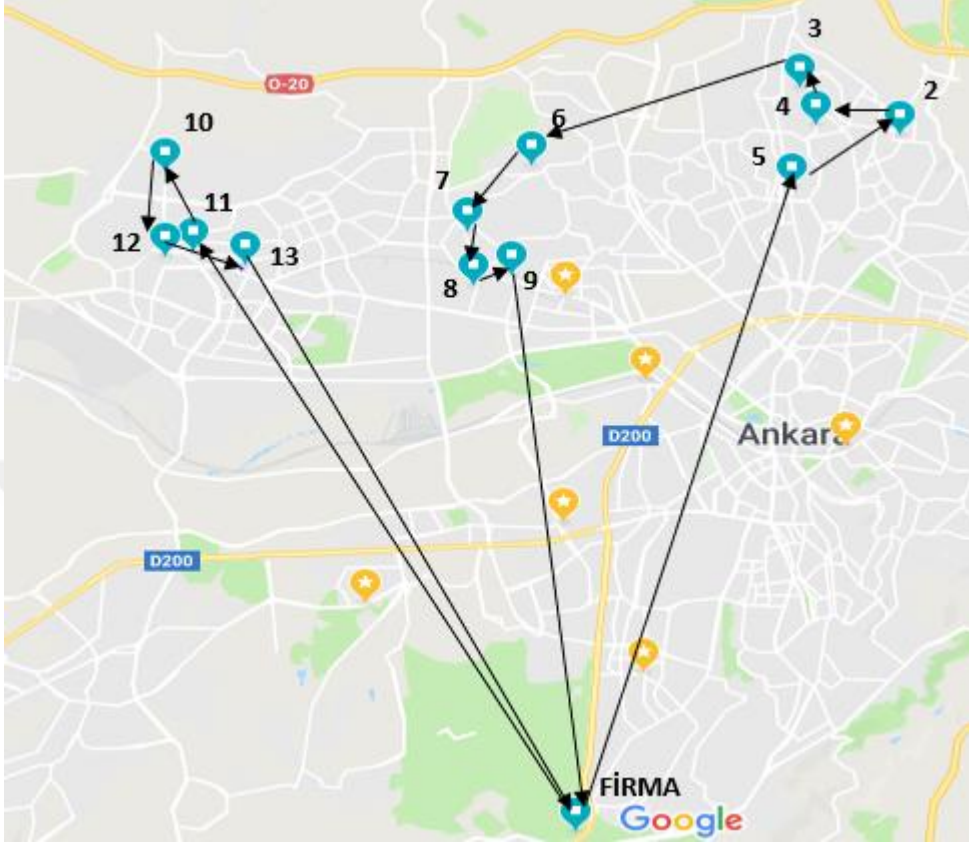
Global optimal solution found.
Objective value:                106400.0
Objective bound:                106400.0
Infeasibilities:                0.000000
Extended solver steps:         1302
Total solver iterations:        37167
Elapsed runtime seconds:        3.90

```

Variable	Value	Reduced Cost
X(FRM, 5)	1.000000	19600.00
X(FRM, 11)	1.000000	21500.00
X(2, 4)	1.000000	2100.000
X(3, 6)	1.000000	7600.000
X(4, 3)	1.000000	1400.000
X(5, 2)	1.000000	3300.000
X(6, 7)	1.000000	2600.000
X(7, 8)	1.000000	1500.000
X(9, FRM)	1.000000	17600.00
X(8, 9)	1.000000	1400.000
X(10, 12)	1.000000	2400.000
X(11, 10)	1.000000	2900.000
X(13, FRM)	1.000000	20700.00
X(12, 13)	1.000000	1800.000

Şekil 7.5. LINGO 18.0 çözüm sonucu

Rotaların haritadaki görünümüne Şekil 7.6.'da yer verilmiştir.



Şekil 7.6. Araç rotalarının haritada görünümü

- Rota uzunluğunun maksimum 50 km olması durumunda;

Sefer süresiyle ilgili bir değerlendirmede bulunmak için bu uygulamada toplam mesafe kısıtı üzerinden çalışma yapılabilir. Şöyle ki bu problemin varsayımında trafik koşulları ve araç performansının dikkate alınmadığı belirtilmişti. Dolayısıyla toplam mesafenin azaltılması ya da artırılması sefer süresini etkileyecektir. Mevcut durumda rota uzunluğu maksimum 60 km'dir. Bu senaryoda 50 km olduğu varsayılarak model LINGO'da çalıştırılmıştır. Şekil 7.7.'de sonuçlar yer almaktadır.

```

Global optimal solution found.
Objective value:                139500.0
Objective bound:                139500.0
Infeasibilities:                0.000000
Extended solver steps:          903
Total solver iterations:        25916
Elapsed runtime seconds:        2.95

```

Variable	Value	Reduced Cost
X(FRM, 2)	1.000000	21300.00
X(FRM, 9)	1.000000	17600.00
X(FRM, 11)	1.000000	21500.00
X(2, 3)	1.000000	2900.000
X(3, 4)	1.000000	1400.000
X(4, 5)	1.000000	1900.000
X(5, FRM)	1.000000	19600.00
X(6, FRM)	1.000000	20000.00
X(7, 6)	1.000000	2600.000
X(9, 8)	1.000000	1400.000
X(8, 7)	1.000000	1500.000
X(10, 12)	1.000000	2400.000
X(11, 10)	1.000000	2900.000
X(13, FRM)	1.000000	20700.00
X(12, 13)	1.000000	1800.000

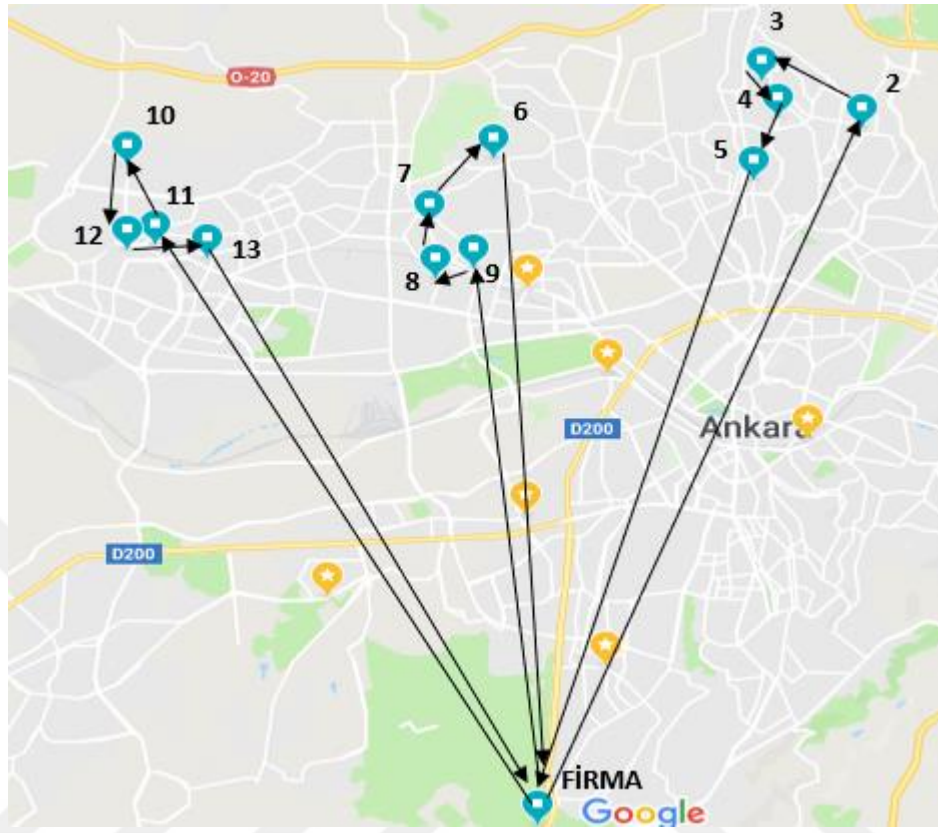
Şekil 7.7. LINGO 18.0 çözüm sonucu

Rotalar şu şekildedir;

- Birinci rota durak sırası: Süleymaniye Camisi, Meslek Hastanesi, Gazino Durağı, Tepebaşı olup toplam mesafe 47,1 km,
- İkinci rota durak sırası; Demetevler 12. Caddesi, Demetevler 1. Caddesi, İvedik Caddesi ve Şentepe olup toplam mesafe 43,1 km olarak hesaplanmıştır.
- Üçüncü rota durak sırası;. Rota: Atlantis Önü, Çakırlar, Gimsa, Mesa Postanesi olup toplam mesafe 49,3 km olarak hesaplanmıştır.

Dikkat edilirse, mevcut duruma göre çözüm aslında aynıdır. 2. Rotadaki (9-8-7-6.durakların yer aldığı rota) sıralama 18 kişi kapasiteli aracın yer alan çözümdeki sıralamanın tersi yani simetriğidir. Duraklar arası mesafeler simetrik kabul edildiğinden sonuç değişmemektedir.

Rotaların haritadaki görünümüne Şekil 7.8.'de yer verilmiştir.



Şekil 7.8. Araç rotalarının haritada görünümü

Rota uzunluğunun kısıtı için başka varyasyonlar düşünülebilir. Ancak burada firmaya en yakın durağın (Durak 7) 17,6 km uzaklıkta olduğu ve duraklar arası mesafelerin uzun olduğu – örneğin Durak 2 ve Durak 6 arasındaki mesafenin 12,7 km olması- düşünüldüğünde rota uzunluğu kısıtının 45 km indirilmesi problemin fizibil çözüm alanında tutulmasını zorlayan bir etkidir.

- Araç kapasitesinin 18 ve 24 kişi olduğu durum:

Bu senaryoda araçların özdeş olduğu varsayımı gözardı edilerek 18 ve 24 kişi kapasiteli araçlar modelde kullanılmıştır.

Bu durumda oluşan rotalar şöyledir;

- Birinci rota durak sırası; Rota: Atlantis Önu, Çakırlar, Gimsa, Mesa Postanesi ve Demetevler 1. Cadde olup toplam gidilen mesafe 57,1 km,
- İkinci rota durak sırası; Tepebaşı, Süleymaniye Camisi, Gazino Durağı, Meslek Hastanesi, Şentepe, İvedik Caddesi, Demetevler 12. Cadde olup toplam gidilen mesafe 49,3 km olarak hesaplanmıştır.

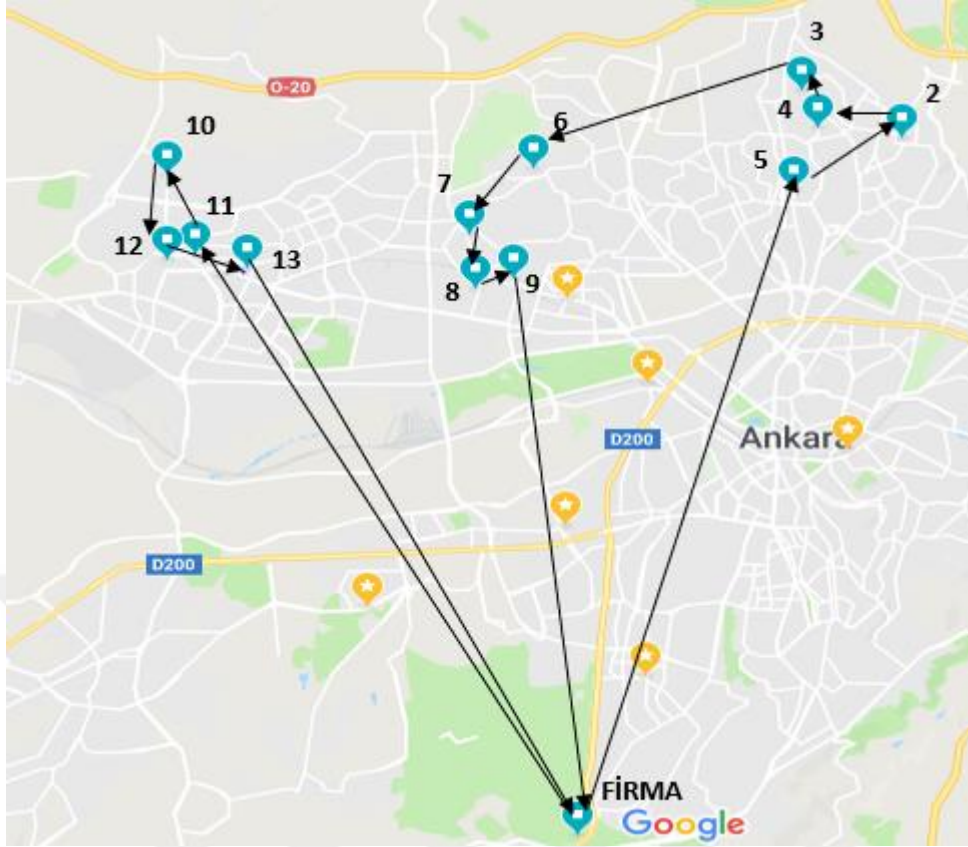
LINGO 18.0 da çalıştırılan modelin çözümü Şekil 7.9.'da gösterilmektedir. Rotaların harita görünümü Şekil 7.10.'da yer almaktadır.

```

Global optimal solution found.
Objective value:                107400.0
Objective bound:                107400.0
Infeasibilities:                0.000000
Extended solver steps:          4020
Total solver iterations:        67104
Elapsed runtime seconds:        5.93

Variable           Value           Reduced Cost
X( FRM, 5)         1.000000        19600.00
X( FRM, 11)        1.000000        21500.00
X( 2, 4)           1.000000        2100.000
X( 3, 6)           1.000000        7600.000
X( 4, 3)           1.000000        1400.000
X( 5, 2)           1.000000        3300.000
X( 6, 7)           1.000000        2600.000
X( 7, 9)           1.000000        1600.000
X( 9, FRM)         1.000000        17600.00
X( 8, FRM)         1.000000        17700.00
X( 10, 12)         1.000000        2400.000
X( 11, 10)         1.000000        2900.000
X( 13, 8)          1.000000        5300.000
X( 12, 13)         1.000000        1800.000
  
```

Şekil 7.9. Lingo Çözümü



Şekil 7.10. Harita Görünümü

Uygulama sonuçlarının toplam mesafeler ve maliyet hesaplamalarıyla ilgili genel değerlendirmesi aşağıda yer almaktadır.

18 kişi kapasiteli araç için gidilen çözümde 3 rota oluşmuştur. Ortalama rota uzunluğu 46,5 km'dir. Toplam mesafe uzunluğu 139,5 km'dir.

22 kişi kapasiteli araç için gidilen çözümde 2 rota oluşmuştur. Ortalama rota uzunluğu 53,7 km'dir. Toplam mesafe uzunluğu 107,4 km'dir.

24 kişi kapasiteli araç için gidilen çözümde 2 rota oluşmuştur. Ortalama rota uzunluğu 53,2 km'dir. Toplam mesafe uzunluğu 106,4 km'dir.

Maksimum rota uzunluğunun 50 km olduğu durum için gidilen çözümde 3 rota oluşmuştur. Ortalama rota uzunluğu 46,5 km'dir. Toplam mesafe uzunluğu 139,5 km'dir.

18 ve 24 kişi kapasiteli araçlarla gidilen çözümde 2 rota oluşmuştur. Ortalama rota uzunluğu 53,2 km'dir. Toplam mesafe uzunluğu 106,4 km'dir.

Tüm çözümlerin yer aldığı Tablo 7.1.'de araçların maliyeti de yer almaktadır. Aylık çalışma günü 20 gün kabul edilerek aylık maliyetler hesaplanmıştır.

Tablo 7.1. Toplam mesafe ve toplam maliyet

	Araç kapasitesi (18 kişi)	Araç kapasitesi (22 kişi)	Araç Kapasitesi (24 kişi)	Rota uzunluğu maks. 50 km	Araç Kapasitesi (18 ve 24 kişi)
Araç/Rota Sayısı	3	2	2	3	2
Toplam Mesafe	139,5 km	107,4 km	106,4 km	139,5 km	106,4 km
Ortalama Rota uzunluğu	46,5 km	53,7 km	53,2 km	46,5 km	53,2 km
Toplam Maliyet	13.500,00	9.000,00	11.250,00	13.500,00	10.125,00
	TL/ay	TL/ay	TL/ay	TL/ay	TL/ay

Burada dikkat edilmelidir ki 18 ve 22 kişilik araçların günlük maliyeti aynı olmakla birlikte 24 kişilik aracın maliyeti farklıdır. 24 kişi kapasiteli araç otobüs iken, diğerleri minibüstür. Son senaryoda her iki tip aracın maliyetleri, toplam maliyeti oluşturur. Diğer bir dikkat edilecek husus rota uzunluğunun artmasının personelin serviste geçirdiği süreyi de arttırmasıdır.

Personelin serviste geçirdiği süre, servise bindikten sonra gittiği mesafe üzerinden her durak ve her senaryo için hesaplanmıştır. Tüm hesaplamalar Tablo 7.2.'de gösterilmiştir. Rota sayısının düştüğü yani rota uzunluğunun arttığı durumlarda, ilk durakta servise binen personelin gideceği mesafe artarken, aynı şekilde diğer duraklardaki personelin gideceği mesafe de bu durumdan etkilenmektedir.

Tablo 7.2. Personelin servise bindikten sonra gittiği mesafeler (km)

	Araç kapasitesi (18 kişi)	Araç kapasitesi (22 kişi)	Araç Kapasitesi (24 kişi)	Rota uzunluğu maks. 50 km	Araç Kapasitesi (18 ve 24 kişi)
Durak 2	25,8	32,9	34,2	25,8	34,2
Durak 3	22,9	29,4	30,7	22,9	30,7
Durak 4	21,5	30,8	32,1	21,5	32,1
Durak 5	19,6	36,2	37,5	19,6	37,5
Durak 6	23,1	21,8	23,1	20	23,1
Durak 7	20,5	19,2	20,5	22,6	20,5
Durak 8	19	17,7	19	24,1	19
Durak 9	17,6	17,6	17,6	25,5	17,6
Durak 10	24,9	27,2	24,9	24,9	24,9
Durak 11	27,8	30,1	27,8	27,8	27,8
Durak 12	22,5	24,8	22,5	22,5	22,5
Durak 13	20,7	23	20,7	20,7	20,7
Genel Ort. mesafe	22,16	25,89	25,88	23,16	25,88

Her senaryoda her rota için ayrıca ortalama serviste gidilen mesafelere ait hesaplama Tablo 7.3.'te gösterilmiştir.

Tablo 7.3. Rota bazlı personelin gittiği mesafeler (km)

	Araç kapasitesi (18 kişi)	Araç kapasitesi (22 kişi)	Araç Kapasitesi (24 kişi)	Rota uzunluğu maks. 50 km	Araç Kapasitesi (18 ve 24 kişi)
Rota 1	22,450	26,843	26,838	22,450	26,838
Rota 2	20,050	24,560	23,975	23,050	23,975
Rota 3	23,975			23,975	

BÖLÜM 8. TARTIŞMA VE SONUÇ

Toplam mesafenin minimizasyonu amaçlanan bu uygulamada yapılacak değerlendirmede araç maliyetleri kararı etkileyen ikinci husustur. Toplam mesafenin 106,4 km olduğu 24 kişi kapasiteli 2 aracın yer aldığı çözüm ve kapasiteleri farklı olan (18 ve 24 kişi kapasiteli araçlar) araçların yer aldığı çözüm seçenekleri arasında en iyi çözümlerdir. Maliyetlere bakıldığında ise, araç tipi değiştiğinden maliyetler de değişmektedir. Minibüs olan 22 kişilik araç için toplam maliyet 9.000,00 TL iken, 24 kişilik otobüs için toplam maliyet 11.250,00 TL'dir. Her iki tip aracın da kullanıldığı son senaryoda ise toplam maliyet 10.125,00 TL'dir. Bu durumda son senaryonun sunduğu çözüm tercih edilebilir.

Personelin toplam gittiği mesafe göz önüne alındığında öncelikle rota uzunluğunun 60 km ve 50 km olduğu durumlarda toplam gidilen mesafe ve rotalarda yer alan duraklar aynı olmasına rağmen personelin ortalama gittiği mesafe değişkenlik gösterdiği fark edilmektedir. Bir rota için söz konusu iki seçeneğin birbirinin simetriği olduğunu daha önce belirtilmişti. Rotanın başlangıç durağı değiştiği için personelin ortalama gittiği mesafenin aynı olmaması olayın doğası gereğidir.

22 kişi ve 24 kişi kapasiteli araçlarda personelin yol aldığı ortalama mesafeler bir rota için 1 km'den az iken, diğer rota için dikkate alınmayacak bir fark (5 metre) bulunmaktadır. Durak bazlı bakıldığında ise fark 2 ila 3 km arasında değiştiği duraklar vardır. Maliyet ve toplam mesafe ile personelin gittiği mesafe birlikte değerlendirildiğinde personelin toplam gittiği mesafedeki artış kabullenilerek tercih 22 kişi kapasiteli araç yönünde yapılabilir.

Toplam gidilen mesafeyi maliyetle ilişkilendirerek çözümlerinin değerlendirildiği bu uygulama farklı olarak da ele alınabilir. Personelin serviste geçirdiği sürenin

minimizasyonu amaçlanabilir. Günümüz iş dünyasında çalışanın daha iyi koşullarda çalışabilmesi ve memnuniyetinin sağlanması açısından bu yönde bir analiz etkili olacaktır.

Bu uygulamanın özelinde yukarıda örnek verilen analizden ayrı olarak, servis kullanmayan personel de probleme dahil edilebilir. Mevcutta servis kullanamayan personel için yol ücreti ödenmektedir. Personele ödenen yol ücreti ceza maliyeti gibi kabul edilerek model de yer alabilir. Bu durumda araç maliyeti de dahil edilerek toplam maliyetin minimizasyonu hedeflenebilir.

Yine aynı şekilde bölünmüş dağıtımli araç problemi olarak ele alınarak daha esnek kullanılan araç kapasitesinin toplam maliyet ya da toplam mesafeye etkisi değerlendirilebilir.

KAYNAKLAR

- Akca, K. (2015). Hammadde tedarik aktivitesi için kesin zaman pencereli araç rotalama problemi. Bursa: Uludağ Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği, Yüksek Lisans Tezi.
- Alaykırın, K., & Engin, O. (2005). Gazi Üniversitesi Müh. Mi, 20(1), 49-76.
- Archetti, C., Bouchard, M., & Desaulniers, G. (2011). Enhanced branch and price and cut for vehicle routing with split deliveries and time windows. *Transportation Science*, 45(3), 285-298.
- Atan, M., & Şimşek, P. (2017). Doğrusal programlama ile araç rotalama probleminin çözümlenmesi. *Gazi Üniversitesi Sosyal Bilimler Dergisi*, 4(11), 339-359.
- Atmaca, E. (2012). Bir kargo şirketinde araç rotalama problemi ve uygulaması. *TÜBAV Bilim Dergisi*, 5(2), 12-27.
- Aydemir, E. (2006). Esnek zaman pencereli araç rotalama problemi ve Bir uygulama. Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği. Ankara: Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi.
- Başkaya, Z., & AVCI ÖZTÜRK, B. (2005). Tamsayılı programlamada dal kesme yöntemi ve bir ekmek fabrikasında oluşturulan araç rotalama problemine uygulanması. *Uludağ Üniversitesi İBBF Dergisi*, 14(1), 101-114.
- Bektaş, T., & Elmastaş, S. (2017). Solving school bus routing problems through integer programming. *Journal of the Operational Research Society*, 58(12), 1599-1604.
- Bianchi, L. (2000). Notes on Dynamic Vehicle Routing. Technical Report - IDSIA-.
- Bowerman, R., Hall, B., & Calamai, P. (1995). A multi-objective optimization approach to urban school bus routing: formulation and solution method. *Transpn. Res-A*, 29A(2), 107-123.
- Bozyer, Z., Alkan, A., & Fırlalı, A. (2014). Kapasite kısıtlı araç rotalama probleminin çözümü için önce grupla sonra rotala merkezli sezgisel algoritma önerisi. *Bilişim Teknolojileri Dergisi*, 7(2), 29-38.
- Can Atasagun, G. (2015). Zaman bağımlı eş zamanlı topla dağıt araç rotalama problemi. Konya: Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği, Yüksek Lisans Tezi.
- Chu, J., Yan, S., & Huang, H.-J. (2017). A multi-trip split-delivery vehicle routing problem with time windows for inventory replenishment under stochastic travel times. *Netw Spat Econ*, 17, 41-68.

- Çalışkan, F., Yüksel, H., & Dayık, M. (2016). Genetik algoritmaların tasarım sürecinde kullanılması. *SDU Teknik Bilimler Dergisi*, 6(2), 21-27.
- Çelikkanat Filiz, S., & Eroğlu, E. (2017). Fuzzy goal programming approach in vehicle routing problem. *Journal of Transportation and Logistics*, 2(2), 49-64.
- Çetin, S., & Gencer, C. (2011). Heterojen araç filolu zaman pencereli eş zamanlı dağıtım-toplamalı araç rotalama problemleri: matematiksel model. *International Journal of Research and Development*, 3(1), 19-27.
- Desaulniers, G. (2010). Branch-and-price-and-cut for the split-delivery vehicle routing problem with time windows. *Operations Research*, 58(1), 179-192.
- Dursun, P. (2009). Zaman pencereli araç rotalama problemi'nin genetik algoritma ile modellenmesi. İstanbul: İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği, Yüksek Lisans Tezi.
- Düzakın, E., & Demircioğlu, M. (2009, Haziran). Araç rotalama problemleri ve çözüm yöntemleri. *Çukurova Üniversitesi İİBF Dergisi*, pp. 68-87.
- Ekmekçi, N. (2015). Sanayi işletmelerinde üretim planlaması ve doğrusal programlama ile bir sanayi işletmesinde optimizasyon uygulaması. Konya: Selçuk Üniversitesi, Sosyal Bilimleri Enstitüsü, İşletme Bölümü, Yüksek Lisans Tezi.
- Ercan Cömert, S., Yazgan, H., Sertvuran, İ., & Şengül, H. (2017). Sıkı Zaman Pencereli Araç Rotalama Probleminin Çözümü için Yeni Bir Yöntem Önerisi. *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi*.
- Ercan, C., & Gencer, C. (2013). Dinamik İnsansız Hava Sistemleri Rota Planlaması Literatür Araştırması Ve İnsansız Hava Sistemleri Çalışma Alanları. *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 19(2), 104-111.
- Erol, V. (2006). Araç rotalama problemleri için populasyon ve komşuluk tabanlı metasezgisel bir algoritmanın tasarımı ve uygulaması. İstanbul: Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi.
- Golden, B., Ragwahan, S., & Wasil, E. A. (2005). *The Next Wave in Computing, Optimization and Decision Technologies*. Newyork: Springer Science Business Media.
- Güden, H., Vakvak, B., Özkan, B., Altıparmak, F., & Dengiz, B. (2005). Genel amaçlı arama algoritmaları ile benzetim eniyilemesi: en iyi kanban sayısının bulunması. *MMO Endüstri Mühendisleri Dergisi*, 16(1), 2-15.
- Güvez, H., Dege, M., & Eren, T. (2012). Kırıkkale'de araç rotalama problemi ile tıbbi atıkların toplanması. *International Journal of Engineering Research and Development*, 4(1), 41-46.
- Hezer, S., & Kara, Y. (2013). Eşzamanlı dağıtım ve toplamalı araç rotalama problemlerinin çözümü için bakteriyel besin arama Optimizasyonu Tabanlı Bir Algoritma. *Gazi Üniv. Müh. Mim. Fak. Der.*, 28(2), 373-382.
- Kara, İ., & Bektaş, T. (2006). Integer linear programming formulations of multiple salesman and its variations. *European Journal of Operational Research*, 174, 1449-1458.

- Keskintürk, T., Topuk , N., & Özyeşil, O. (2015). Araç Rotalama Problemleri ile Çözüm Yöntemlerinin Sınıflandırılması ve Bir Uygulama. *İşletme Bilimi Dergisi*, 3(2).
- Koç, Ç. (2012). Çok kullanımlı ve zaman pencereci araç rotalama problemi için bir matematiksel model. *Gazi Üniv. Müh. Mim. Fak. Der*, 27(3), 569-576.
- Lee, C., Lee, K., & Park, S. (2012). Robust vehicle routing problem with deadlines and travel time/demand uncertainty. *The Journal of the Operational Research Society*, 63(9), 1294-1306.
- Luenberger, D., & Ye, Y. (1984). *Linear and Nonlinear Programming* (Third Edition ed.). London: Springer.
- Machado, P. T. (2002). *Vehicle Routing Problem: Doing it the Evolutionary Way*. GECCO. New York.
- Maden, W., Eglese, R., & Black, D. (2010). Vehicle routing and scheduling with time-varying data: a case study. *The Journal of the Operational Research Society*, 61(3), 1524-1532.
- Martinez, L., & Amaya, C. (2013). A vehicle routing problem with multi-trips and time windows for circular items. *Journal of the Operational Research Society* (, 64, 1630-1643.
- Özçiloğlu, M. M. (2009). Algılayıcı Ağlarda Gözlemlenememe Olgusunun Doğrusal Programlama ile İncelenmesi. Ankara: TOBB Ekonomi ve Teknoloji Üniversitesi, Yüksek Lisans Tezi.
- Pala, O., & Aksaraylı, M. (2018). An ant colony optimization algorithm approach for solving multi-objective capacitated vehicle routing problem. *Alphanumeric Journal*, 6(1), 37-48.
- Pekdemir G. (2012). Çoklu imge eşikleme problemlerinde metasezgisel algoritmaların performans analizi. Konya, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği, Yüksek Lisans Tezi.
- Sarıcıoğlu, A. Y. (2014). A Heuristic Framework for Solving Time Dependent Vehicle Routing Problem with Time Windows. İstanbul: Bahçeşehir Üniversitesi.
- Şahin, Y. (2015). Yonca Levha Fabrikasında Tamsayılı Doğrusal Programlama Metodu ile Üretim Planlaması. Kahramanmaraş: Sütçü İmam Üniversitesi, Yüksek Lisans Tezi.
- Tansini, L., Urquhart, M., & Viera, O. (2000). Comparing assignment algorithms for the Multi-Depot VRP. Montevideo, URUGUAY: UDELAR Latin-American Conference on Operations Research and Systems(CLAIO).
- Tezer, T. (2009). Tamsayılı programlamada dal kesme yöntemi ve bir ekmek fabrikasında oluşturulan araç rotalama problemine uygulanması. Balıkesir: Balıkesir Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği, Yüksek Lisans Tezi.
- Tokaylı, M. A. (2005). Zaman Pencereci Araç Rotalama Problemi İçin Bir Karar Destek Sistemi. Ankara: Gazi Üniversitesi.

- Ünsal, Ö., & Yiğit, T. (2018). Yapay zeka ve kümeleme teknikleri kullanılarak geliştirilen yöntem ile okul servisi rotalama probleminin optimizasyonu. *Mühendislik Bilimleri ve Tasarım Dergisi*, 6(1), 7-20.
- Vansteenwegen, P., Souffriau, W., & Sörensen, K. (2012). The travelling salesperson problem with hotel selection. *Journal of the Operational Research Society*, 63, 207-217.
- Wang, Z., Liang, W., & Hu, X. (2014). A metaheuristic based on a pool of routes for the vehicle routing problem with multiple trips and time windows. *Journal of the Operational Research Society*, 65, 37-48.
- www.ktu.edu.tr/dosyalar/ormanamenajmani_090db.pdf, Erişim Tarihi: 08.04.2019
- www.quora.com/What-path-finding-algorithm-does-Google-Maps-use-to-find-the-shortest-path-and-alternate-routes?no_redirect=1, Erişim Tarihi: 14.04.2019
- Yazgan, H., & Gökçen Büyükyılmaz, R. (2018). Eş zamanlı topla dağıt araç rotalama problemine sezgisel bir çözüm yaklaşımı. *Sakarya Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 22(2), 436-449.
- Yılmaz, Ş. (2008). Çok depolu araç rotalama probleminin karınca kolonisi optimizasyonu ile modellenmesi ve bir çözüm önerisi. İstanbul: Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Endüstri Mühendisliği, Yüksek Lisans Tezi.
- Zachariadis, E., Tarantilis, C., & Kiranoudis, C. (2012). The pallet-packing vehicle routing. *Transportation Science*, 46(3), 341-558.

EKLER

EK 1: LINGO 18.0 PROGRAMI MODELİ

SETS:

```
CITY: Q, U, TD, TME, TML, TMV, TMA;  
CXC( CITY, CITY): DIST, X;
```

ENDSETS

DATA:

```
TMAX = 99999;
```

```
CITY = FRM 2 3 4 5 6 7 9 8 10 11 13 12 ;
```

```
! Amount to be delivered to each customer;
```

```
Q= 0 5 1 2 3 3 4 2 3 3 5 3 3 ;
```

```
! city 1 represents the common depot, i.e. Q( 1) = 0;
```

```
! city 1 represents the common depot, i.e. Q( 1) = 0;
```

```
! Distance from city I to city J is same (but need not be) from J to I,
```

```
distance from city I to the depot maybe 0 (but need not be),
```

```
if vehicle need not return to the depot ;
```

DIST=

```
0      21300 22900 21800 19600 20000 18800 17600 17700 23800 21500  
      20700 22700  
21300 0      2900  2100  3300  10200 18200 11300 12700 25400 19000  
      17800 21800  
22900 2900 0      1400  3000  7600  12700 9700  14400 17300 20800  
      17800 22500  
21800 2100 1400 0      1900  7000  10500 11000 12400 16900 19800  
      16400 21500  
19600 3300 3000 1900 0      7300  8900  9200  10500 18000 15800  
      14600 16400  
20000 10200 7600 7000 7300 0      2600  4200  4300  8600  8500  
      7400  9200  
18800 18200 12700 10500 8900 2600 0      1600  1500  7200  6100  
      4900  6800  
17600 11300 9700  11000 9200 4200 1600 0      1400  8800  7600  
      6500  9200
```

```

17700 12700 14400 12400 10500 4300 1500 1400 0 8700 6400
      5300 7100
23800 25400 17300 16900 18000 8600 7200 8800 8700 0 2900
      3800 2400
21500 19000 20800 19800 15800 8500 6100 7600 6400 2900 0
      2300 1000
20700 17800 17800 16400 14600 7400 4900 6500 5300 3800 2300 0
      1800
22700 21800 22500 21500 16400 9200 6800 9200 7100 2400 1000
      1800 0;

```

```

VCAP = 24;      ! arac kapasitesi;
DMAX = 60000;  ! maksimum mesafe;
MXTRK = 9999; ! Max vehicles allowed;
ENDDATA

! Amaç fonksiyonu toplam mesafenin minimum olması;
MIN = TDIST;
      TDIST = @SUM( CXC: DIST * X);

@FOR( CITY(k):
      ! a vehicle does not travel inside itself,...;
      X( k, k) = 0;
      );

! For each city, except depot....;
@FOR( CITY( k) | k #GT# 1:

      ! a vehicle must enter city K from some city I,... ;
      [NTR] @SUM( CITY( i) | i #NE# k #AND# ( i #EQ# 1 #OR#
      Q( i) + Q( k) #LE# VCAP): X( i, k)) = 1;

      ! a vehicle must leave K after service to some city J;
      [XIT] @SUM( CITY( j) | j #NE# k #AND# ( j #EQ# 1 #OR#
      Q( j) + Q( k) #LE# VCAP): X( k, j)) = 1;

      ! U( k) is at least amount needed at K, but can't
      exceed vehicle capacity;
      @BND( Q( k), U( k), VCAP);

      ! If K follows I, then can bound U( k) - U( i);
      @FOR( CITY( i) | i #NE# k #AND# i #NE# 1:
      [UL] U( k) >= U( I) + Q( k) - VCAP + VCAP *
      ( X( k, i) + X( i, k)) - ( Q( k) + Q( i))
      * X( k, i);
      );

      ! If K is 1st stop, then U( k) = Q( k);
      U( k) <= VCAP - ( VCAP - Q( k)) * X( 1, k);

      ! If K is not 1st stop...;

```



```

    U( k) >= Q( k) + @SUM( CITY( i) |
      I #GT# 1: Q( i) * X( i, k));
  );
@FOR (

  ! Compute the total distance traveled by the vehicle through J;
  @FOR( CITY( j):
    [R_TD_1] TD( j) >= DIST( 1, j) * X( 1, j);
    @FOR( CITY( i) | i #GT# 1:
      [R_TD] TD( j) >= TD( i) + DIST( i, j) * X( i, j) - DMAX * ( 1 -
X( i, j))
    );
  );

  ! Longest trip cannot exceed max trip length;
  TD( 1) <= DMAX;

  ! Make the X's binary;
  @FOR( CXC: @BIN( X));

  ! Minimum no. vehicles required, fractional, e.g., 3.7222;
  VEHCLF = @SUM( CITY( I) | I #GT# 1: Q( I)) / VCAP;
  ! and rounded up, e.g. 4.0;
  VEHCLR = @FLOOR( VEHCLF + 0.999);
  ! Must send enough vehicles out of depot;
  @SUM( CITY( j) | j #GT# 1: X( 1, j)) >= VEHCLR;

  ! Max vehicles\trucks constraint;
  @SUM( CITY( j) | j #GT# 1: X( 1, j)) <= MXTRK;
END

```

ÖZGEÇMİŞ

Semiha Erdoğan 1984 yılında Nazilli- Aydın' da doğmuştur. İlköğrenimini 1989-1992 yılları arasında Nazilli Turan İlköğretim Okulu'nda, 1992-1993 yılında Mustafakemalpaşa Lalaşahin İlköğretim Okulu'nda tamamlamıştır. Ortaokulu Mustafakemalpaşa Sedat Karan Anadolu Lisesi'nde okuduktan sonra lise eğitimini Erzurum İbrahim Hakkı Fen Lisesi'nde almıştır. Lisans eğitimini Uludağ Üniversitesi Mühendislik Mimarlık Fakültesi Endüstri Mühendisliği Bölümü'nde tamamlamıştır. Halen bir firmada çalışmaktadır.