

Süleyman Demirel Üniversitesi
İktisadi ve İdari Bilimler Fakültesi
Y.2001, C.6, S.1 s.139-155.

ARAÇ ROTALAMA PROBLEMİNE AİT BİR UYGULAMA

Mehmet ERYAVUZ*
Cevriye GENCER**

ÖZET

Araç Rotalama Problemi (ARP), merkezi bir depodan coğrafi olarak dağılmış çeşitli talep noktalarına dağıtım veya toplama rotalarının, araç filosunun kat ettiği toplam mesafeyi minimize edilecek şekilde bulunmasıdır. Çalışmada, ARP çözüm yöntemleri kullanılarak, Balıkesir Ordudonatım Okulu personel servis araçlarının toplam güzergah mesafesi minimize edilmeye çalışılmıştır. Problem çözümünde, sezgisel çözüm metodlarından tasarruf (savings) ve rassal tasarruf (randomized savings) algoritmaları ve VRP 328 yazılımı kullanılmıştır. Bulunan çözümler tur geliştirme algoritmalarıyla (2-opt ve Or-opt) daha iyi hale getirilmeye çalışılmış; elde edilen sonuçlar kendi aralarında ve mevcut durum ile karşılaştırılmıştır.

Araç Rotalama Problemi, Sezgisel Algoritmalar, Tasarruflar Algoritması

GİRİŞ

ARP, coğrafi olarak dağıtık müşterilere bir veya birden fazla depodan hizmet vermek üzere görevlendirilen araçların optimum dağıtım/toplama rotalarının tasarlanması problemi [1]. ARP dağıtım yönetiminin kalbidir. ARP'nin en basit şekline genel ARP denir. Genel ARP 'nde, birinci şehir depo olmak üzere n adet şehir ve m adet araç vardır. Her bir aracın kapasitesi Q ve i düğümünden j düğümüne olan mesafe c_{ij} olarak tanımlanır. ARP ile m adet aracın rotası belirlenirken ;

1. Her bir şehir yalnız bir defa ziyaret edilir.
2. Her bir araç rotasına aynı depoda başlar ve sonlandırır.

* Balıkesir Ordudonatım Okulu ve Eğitim Merkez Komutanlığı, Misafirhane Müdürlüğü, Çayırhisar/Balıkesir

** Gazi Üniversitesi, Mühendislik-Mimarlık Fakültesi, Endüstri Mühendisliği Bölümü, 06570 Maltepe/Ankara

3. Rota sayısı ve konfigürasyonu ile ilgili kısıtlar vardır.

Bu açıklanan temel kısıtlar haricindeki kısıtlar problemin özelliğine bağlı olarak değişmektedir.

ARP'nin temel bileşenlerini; talep yapısı, taşınacak malzemenin tipi, dağıtım/ toplama noktaları ve araç filosu oluşturur.

1)*Talep yapısı*: ARP 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[2].

2)*Malzeme Tipi*: Araçlarla çok çeşitli malzemeler taşınır. Tehlikeli maddeler, gıda maddeleri, gazete dağıtımı, çöp toplama bütün bunlar basit paketler olarak adlandırılır ve probleme ilave bir karmaşıklık getirmezler [3]. 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.

3)*Dağıtım/Toplama noktaları*: Birçok ARP'in de, 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 ARP'ne dönüşür. Araç bir depodan çıkıp, başka bir depoda yükleme/boşaltma yapabilir. Bu durumda problem bir bütün olarak ele alınmalıdır.

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 ARP'lerin de 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.

4)*Araç Filosu*: Bütün ARP'lerin de 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 belirlenmesi ilave bir kararı gerektirir. Araçların diğer

özellikleri arasında hız, yakıt tüketimi, taşınacak malzemeye uygunluğu sayılabilir. Bu özelliklerin rotalama kararlarına doğrudan etkisi yoktur.

ARP literatüründe çok çeşitli optimizasyon kriterleri mevcuttur. Bunlardan en yaygın olanları, rota sayısı, toplam rota uzunluğu, rota süresi, müşteri memnuniyeti, yük dengeleme şeklinde sıralanabilir. Bu kriterlerden rota sayısı ve toplam rota uzunluğu ARP amaç fonksiyonunda en yaygın olarak kullanılmaktadır. ARPnin çözümünde hem sezgisel hem de optimum metotlar kullanılabilir. Laporte ve Nobert [1], ARP için çeşitli optimum çözüm algoritmaları geliştirmişlerdir. Optimum çözüm algoritmaları; doğrudan ağaç arama (direct tree search), dinamik programlama ve tamsayılı doğrusal programlama olmak üzere üç kategoriye ayrılabilir. Magnanti [4], 1981 yılında yaptığı çalışmasında tamsayılı doğrusal programlama formülasyonunu; küme bölme formülasyonu, araç akış formülasyonu ve malzeme/eşya akış formülasyonu şeklinde üç alt kategoriye ayırarak incelemiştir.

ARP hizmet sunulan müşteri sayısı ve coğrafi alan olarak büyük ölçekli bir problemdir. Müşteri sayısı arttıkça alternatif rota sayısı artmakta ve hesaplama zorlaşmaktadır. Bu nedenle ARP çözümünde sezgisel algoritmalar ön plana çıkmaktadır.

ARP sezgisel algoritmaları genel olarak üç ana sınıfa ayrılır [5]:

1. Tur kurucu sezgiseller,
2. Tur geliştirici sezgiseller,
3. İki aşamalı metotlar,

Tur kurucu sezgiseller mümkün olmayan (infeasible) atamalarla çözüme başlar, her defasında iki düğüm arasına bir dal ekleyerek mümkün çözüme ulaşırlar. Dal eklenirken araç kapasite kısıdına uyulup uyulmadığı kontrol edilir. Eklenecek dal, bazı maliyet tasarruflarına göre seçilir. Tur kurucu metotlar arasında en çok tercih edileni, Clarke ve Wright [6]'ın Dantzig ve Ramser'in [7] çalışmasından esinlenerek geliştirdikleri *Tasarruf (Savings) Algoritması* dır. Tasarruf algoritmasının temel konsepti değiştirilmeden sayısız algoritmalar geliştirilmiştir [8-11].

Tur geliştirici sezgiseller bir mümkün çözümü başlangıç çözümü olarak alır ve o çözümü geliştirirler. Her bir iterasyonda dal kombinasyonları değiştirilir ve değişimin mümkün çözüme ulaştırıp ulaştırmadığı, maliyeti düşürüp düşürmediği kontrol edilir. Tur geliştirme sezgiselleri Lin [12] ve Lenstra ve Rinnooykan [13]'un Seyyar Satıcı Problemi (SSP) sezgiseline dayanır.

İki aşamalı metotların birinci aşamasında, düğümler araçlara kapasiteyi aşmayacak şekilde atanır. İkinci aşamada ise her bir araç için SSP sezgiselleri kullanılarak rota oluşturulur. Önce gruplama-sonra rotalama tipindeki algoritmalar, iki aşamalı metotlara örnek olarak verilebilir. Gillet ve Miller'in [14] 1974 yılında geliştirdikleri Süpürme (Sweep) algoritması, Fisher ve Jaikumar'ın [5] 1981 geliştirdikleri algoritma, Christofides, Mingozzi ve Toth'un [15] 1980 yılında geliştirdikleri algoritma iki aşamalı algoritmalara örnek olarak verilebilir.

ARP'nin genel matematiksel modeli aşağıda verilmektedir [5]

$$\min Z = \sum_{i,j} c_{ij} \sum_k x_{ijk} \quad (1)$$

Kısıtlar

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

$$\sum_j x_{ijk} = y_{ik} \quad i = 1, \dots, n \quad k = 1, \dots, m \quad (3)$$

$$\sum_{i,j \in S} x_{ijk} \leq |S| - 1 \quad \text{tüm } S \subseteq \{2, \dots, n\} \quad k = 1, \dots, m \quad (4)$$

$$y_{ik} \in \{0, 1\} \quad i = 1, \dots, n \quad k = 1, \dots, m \quad (5a)$$

$$x_{ijk} \in \{0, 1\} \quad i, j = 1, \dots, n \quad k = 1, \dots, m \quad (5b)$$

Burada;

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

$$y_{ik} = \begin{cases} 1, & k \text{ aracı } i \text{ düğümünü ziyaret ederse} \\ 0, & \text{diğer durumda} \end{cases}$$

m = araç sayısı,

n = düğüm sayısı,

S = düğümler setinin alt seti,

|S| = S alt setindeki düğüm sayısı,

Modelde (2) kısıt, her bir müşterinin (depo haricinde) bir araca atanmasını; (3) kısıt müşteriye hizmet için gelen aracın tekrar gitmesini sağlayan ve (4) kısıt ise SSP alt tur eleme kısıdır.

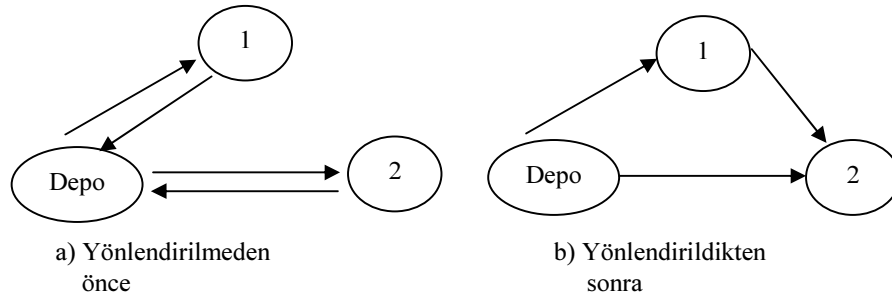
Yukarıdaki modele özel durumlar ilave edilerek oluşturulmuş çeşitli matematiksel modeller literatürde mevcuttur [15,16].

Çalışmada tur kurucu sezgisellerden tasarruf algoritması kullanılmıştır.

TASARRUF ALGORİTMASI

İşlem bir noktanın başlangıç noktası (Depo) olarak alınmasıyla başlar. Başlangıçta bütün durakların depodan ziyaret edildiği varsayılır. Sonra iki alt turun birleştirilmesiyle elde edilen tasarruflar hesaplanır.

En büyük tasarruf değerinden başlayarak rotalar birleştirilir. Birleştirilen müşteriler bir tek büyük müşteri olarak kabul edilir. Algoritmanın temel konsepti Şekil-1'de gösterilmiştir.



Şekil-1 Tasarruf (Savings) Algoritmasının konsepti

Önce bir durak çifti (i;j) yönlendirilir ve yönlendirilmiş olarak kalır. Algoritma bütün duraklar atanıncaya kadar devam eder. Sağlanan tasarruf

$$S_{ij} = c_{1i} + c_{1j} - c_{ij}$$

şeklinde tasarruf formülüyle hesaplanır.

Algoritmanın temel adımları şu şekilde sıralanabilir:

1.Adım: Tüm i ve j çiftleri için tasarrufları hesapla.

$$(S_{ij} = c_{1i} + c_{1j} - c_{ij})$$

S_{ij} tasarrufu, (i,j) dalı Şekil-1' deki gibi $(1,i,1)$ ve $(1,j,1)$ gibi iki alt tur yerine, $(1,i,j,1)$ turu kurulduğunda sağlanan tasarruftur.

2.Adım: Tasarrufları (S_{ij}) büyükten küçüğe sırala.

3.Adım: Listenin başından başlayarak versiyon çeşidine göre aşağıdaki adımları uygula.

Paralel Versiyonda

4.Adım: Listenin başından aldığımız dal, ARP kısıtlarına uyararak bir mümkün çözüm üretiyorsa, bu dalı tura ilave et, aksi durumda çıkar.

5.Adım: Listedeki sıradaki dalı al ve 4. adımı listede dal kalmayana kadar tekrarla.

Sıralı (Sequential) Versiyonda

4.Adım: Mevcut rotanın iki ucundan birini geliştirecek mümkün dalı listeden bul.

5.Adım: Rota daha fazla geliştirilemiyorsa, rotayı sonlandır. İlk mümkün dalı, yeni bir rotaya başlamak üzere seç.

6.Adım: Seçilecek dal kalmayınca kadar 4.Adımı ve 5.Adımı tekrarla:

Rotalar arka arkaya sıralı (sequential) veya aynı anda (paralel) oluşturulabilir. Araç kapasitesi dolana kadar talep noktaları (düğüm) mevcut rotaya eklenirse, rotalar arka arkaya sıralı şekillenmiş olur. Her bir araç için aynı anda kısmi rotalar oluşturulursa, rotalar paralel şekillenmiş olur.

Tasarruf algoritmasını modife etmek için yapılan çalışmalardan birisi tasarruf denkleminin değiştirilmesidir [11]. Yellow [11] çalışmasında, rota biçimi parametresi olarak θ 'yi $(0,1)$ aralığında tanımlayarak aşağıdaki denklemi önermiştir.

$$S_{ij} = c_{1i} + c_{1j} - \theta c_{ij}$$

θ parametresi değiştirilerek, i ve j duraklarının maliyetine (onların merkez depo ile ilgili maliyeti göz önüne alınarak) verilen önem değiştirilmektedir. $\theta = 1$ alındığında model tasarruflar algoritmasına indirgenmiş olur. Yellow [11] algoritması her zaman en azından tasarruflar algoritması kadar iyi sonuçlar ortaya koymuştur.

UYGULAMA

Bu çalışmada, Balıkesir Ordudonatım Okulu ve Eğitim Merkezi Komutanlığı personel servis otobüsleri için güzergah belirlemede tasarruf (savings) ve rassal tasarruf (randomized savings) algoritması yardımıyla iyileştirilme yapılmaya çalışılmıştır.

Personelin, servis otobüsleriyle belirli duraklardan alınıp iş yerine götürülmesi ve mesai bitiminde tekrar aynı noktalara dağıtılmasını kapsayan problemde kapasite dışında herhangi bir kısıt bulunmamaktadır.

Problem, kapasite kısıtlı araç rotalama problemi olarak adlandırılabilir, çünkü toplama ve dağıtım zamanlarında herhangi bir kısıtlama söz konusu değildir.

Uygulamadaki problem “Arabacı firması otobüs filosu tarafından servis yapılacak olan duraklar gurubunun oluşturduğu rotaların belirlenmesi” şeklinde tanımlanabilir. Zaman ve durakların dolaşılma sırası konularında herhangi bir sınırlama yoktur. Temel problem, her bir araç için en az maliyetli mümkün bir rotanın bulunmasıdır. Rota bir otobüsün servis hizmeti verirken uğraması gereken durakların birleşiminden oluşmaktadır. Güzergah ve rota kelimeleri aynı anlamda kullanılmaktadır. Sonuç olarak; bu çalışma da tek depolu, çok araç ve duraklı rotalama (güzergah belirleme) problemi çözüm yöntemleri kullanılarak Balıkesir Ordudonatım Okulu ve Eğitim Merkez Komutanlığına hizmet veren Arabacı firması servis otobüslerinin güzergahları belirlenip ve araçlar tarafından kat edilen mesafe minimize edilmeye çalışılmıştır.

Mevcut Durum

Balıkesir Ordudonatım Okulu ve Eğitim Merkezi Komutanlığında çalışan personelden yaklaşık 320 kişi servis hizmetlerinden yararlanmaktadır. Balıkesir de çeşitli semtlerde ikamet eden söz konusu personel, her gün mesai başlamadan önce yerleri önceden belirlenmiş, evlerine yakın duraklardan servis

araçlarıyla alınıp Ordudonatım Okuluna getirilmektedir. Servis hizmetini sunan firmaya ait araçlar homojen (kapasiteleri ve tipleri aynı) olup araçlara ait bilgiler Tablo-1 de görülmektedir.

Tablo-1. Ulaştırma hizmetlerinde kullanılan araçların özellikleri

Kategori	Araç Tipi	Araç Miktarı	Araç Kapasitesi
Otobüs	Mercedes-403	5	67

Halen 5 adet otobüs ile 5 farklı güzergah (rota) kullanılarak ulaştırma hizmeti yürütülmektedir.

Rotaların belirlenmesinde zaman kısıtı göz önüne alınmamıştır. Araçlar sabah belirli bir saatte okulda olacak şekilde ilk duraklarından hareket etmektedirler. Lojman bölgesinde tek durakta bir aracı dolduracak kadar yolcu olduğu için bu araç problemin çözümüne dahil edilmemiştir, bunun dışında kalan 57 durak ve 220 personel için problem çözülmüştür.

Halen kullanılmakta olan güzergahların karakteristik özellikleri Tablo-2’de yer almaktadır.

Tablo-2 Mevcut güzergah bilgileri

Rota No	Rota Adı	Rota Uzunluğu (Birim)	Durak Sayısı	Yolcu Sayısı
1	O	96	9	51
2	M	182	16	60
3	C	150	12	42
4	E	166	20	67
TOPLAM		594	57	220

Not : Güvenlik gereği rota adları kodlanmıştır.

Problemin Çözümü

Problemin çözüm aşamasında öncelikle durak yerleri şehir haritası (1/7000 ölçekli) üzerinde işaretlenmiş, durakların koordinatları X-Y koordinat düzlemine göre belirlenmiştir. Uygulama probleminde depo hariç 57 adet durak olup, bu duraklarda toplam 220 yolcu vardır. Durakların koordinatları ve her bir duraktaki yolcu sayısı Tablo-3 ‘de gösterilmiştir.

Tablo-3 Durak koordinatları ve yolcu sayıları

Sıra No	X	Y	Yolcu Sayısı	Durak İsimleri	Sıra No	X	Y	Yolcu Sayısı	Durak İsimleri
0	1.5	-36.5	0	Depo	31	21.8	-12.4	4	C-31
1	-8.2	-21.5	6	O-1	32	23.9	-13.6	5	C-32
2	-7.3	-16.6	6	O-2	33	25.3	-11.4	2	C-33
3	-6.8	-15.3	5	O-3	34	25.3	-10.1	3	C-34
4	-7	-14.3	6	O-4	35	23.8	-6	1	C-35
5	-4.7	-9.1	7	O-5	36	20	-3.5	2	C-36
6	0	0	8	O-6	37	14.6	-2.2	4	E-37
7	-7.5	-2.7	4	O-7	38	0.8	-33.6	3	E-38
8	-6.9	-4.5	4	O-8	39	0	-29.5	3	E-39
9	-6.5	-7.3	5	O-9	40	-1.7	-24.8	2	E-40
10	-3.9	-10.6	6	M-10	41	-2.1	-21.3	4	E-41
11	5.9	4.9	3	M-11	42	-0.6	-18.9	2	E-42
12	4.5	8.1	4	M-12	43	-0.6	-15	2	E-43
13	4.5	10.4	2	M-13	44	-0.5	-13.2	5	E-44
14	5.4	15.1	4	M-14	45	-0.3	-10.1	4	E-45
15	8.7	22.7	1	M-15	46	-1.6	-5.5	2	E-46
16	4.7	24.6	3	M-16	47	3.4	-4.7	3	E-47
17	2.4	26.6	4	M-17	48	9.1	0	2	E-48
18	3.8	33.1	6	M-18	49	11.1	8.5	2	E-49
19	-1.3	26.8	3	M-29	50	13.4	9.7	4	E-50
20	-5.3	24.6	2	M-20	51	16.3	11.3	5	E-51
21	-5.9	21.8	5	M-21	52	15.4	15.1	6	E-52
22	-6.3	19.1	2	M-22	53	12.7	18.4	2	E-53
23	-6.5	15.5	3	M-23	54	13.6	21.7	2	E-54
24	-3.1	12.1	6	M-24	55	15	21.7	4	E-55
25	1.2	8.4	6	M-25	56	17.2	20.1	3	E-56
26	-5	-19.5	7	C-26	57	19.6	17.9	3	E-57
27	4.1	-15.1	8	C-27					
28	10.6	-12.3	2	C-28					
29	13	-10.7	6	C-29					
30	16	-10	2	C-30					

Depo, araçların rotalarına başladıkları ve yolculara hizmet verdikten sonra (dağıtım/toplama) rotalarını bitirdikleri noktadır. Servis hizmetini yürüten otobüsler 47 kişilik oturarak ve 20 kişi de ayakta olmak üzere toplam 67 personel taşıma kapasitesine sahiptir. Çözüm algoritmasında mevcut durum ile

karşılaştırma yapabilmek için araç kapasitesi 67 olarak alınıp buna göre çözüm bulunmuştur. Halihazırda araçlarda boş kapasite bulunmamaktadır. Bu durumda problem çözümünde amaç rota sayısını, dolayısıyla kullanılan araç sayısını minimize etmek olmayacaktır. Problem çözümünde amaç, belirlenen kapasite kısıdını aşmadan, araçlar tarafından kat edilen toplam mesafenin minimize edilmesidir. Problemin çözümünde Northwestern üniversitesinden Mark S. Daskin tarafından geliştirilen VRP328 yazılımı kullanılmıştır [17]. VRP 328 yazılımı windows tabanlı bir uygulamadır. Dört adet veri dosyası, bir adet harita dosyası ve uygulama programından oluşmaktadır. Yazılımda, problem tek aşamada tasarruf ve rassal tasarruf algoritmalarıyla çözülmekte ve çözümü geliştirmek için 2-opt ve Or-opt algoritmaları kullanılmaktadır.

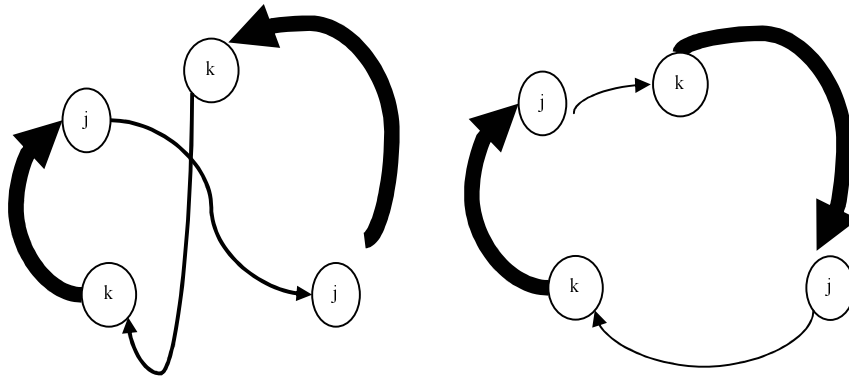
2-Opt Algoritması

Tasarruf algoritmasının uygulanmasını müteakip çözümü geliştirmek için tur geliştirici algoritmalarından 2-opt algoritması uygulanmış olup, bu algoritmanın adımları aşağıda açıklanmıştır. Algoritmanın konsepti Şekil-2 de yer almaktadır.

2-opt algoritmasında her bir dal çifti için aşağıdaki işlemler uygulanır.

1. *İşlem:* Dal çifti turdan çıkarılır.
2. *İşlem:* Tur bozulmayacak şekilde, bir tur oluşturmuş olan başka bir dal çifti ile yer değiştirilir.
3. *İşlem:* Yeni oluşan tur önceki tura göre bir gelişme sağlamış ise dallar yeni yerlerinde kalır, gelişme sağlanmamış ise eski yerine iade edilir.

Bütün dal çiftleri için yukarıdaki işlemler tekrarlanır.



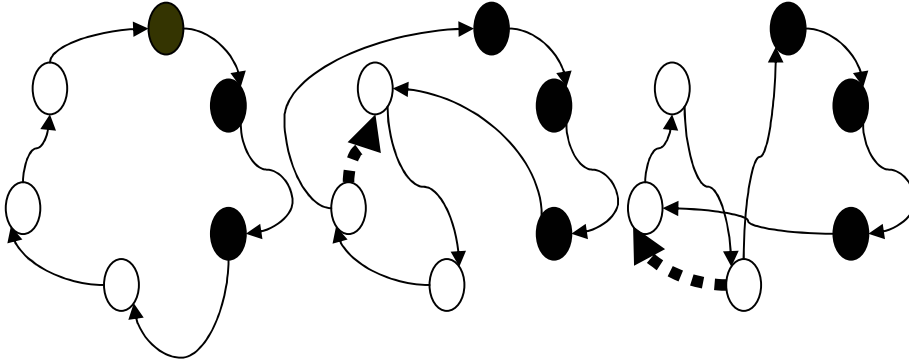
Şekil-2 2-opt algoritmasının konsepti

Or-Opt Algoritması

Problemin çözümünde kullanılan diğer bir tur geliştirici algoritma *Or-opt* algoritmasıdır. Or-opt algoritması Northwestern Üniversitesinden Ilan Or İse tarafından geliştirilmiş olup, algoritmanın adımları aşağıda açıklanmıştır. Ayrıca Şekil-3’de algoritmanın konsepti sunulmuştur.

Or-opt algoritmasında $k=3$ ‘ten 1’ e kadar aşağıdaki işlemler yürütülür.

- 1.İşlem: Turda arka arkaya gelen k sayıda düğüm alınır;
- 2.İşlem: Düğümler mevcut yerlerinden kaldırılıp, deneme yoluyla turun kalan kısmında en uygun yerlere yerleştirilir.
- 3.İşlem: Yeni yerleşim sonucu turda bir gelişme sağlanır ise kaydırma işlemi kabul edilir. Aksi durumda düğümler orijinal yerlerine iade edilir. k sayısı düşürülmeden önce mümkün bütün k sayıda arka arkaya gelen düğüm kümesi için aynı işlem yürütülür.



Şekil-3. Or-opt algoritmasının konsepti

Rassal Tasarruf Algoritması

Yine problem çözümünde kullanılan algoritmalarından birisi olan *Rassal Tasarruf Algoritması*, tasarruf algoritmasını temel alır. Ancak bu algoritma da turların birleştirilmesi esnasında sıradaki en büyük tasarruf yerine tasarruf listesindeki $D+1$ tasarruf arasından tesadüfi olarak seçim yapılır. ‘D’ derinlik (depth) olarak adlandırılır. Her bir derinlik ‘D’ değeri için iterasyon sayısı kadar tekrar yapılır. Algoritma adımları aşağıda sunulmuştur.

1. *Adım*: Derinlik (depth) ve iterasyon (iteration) sayısını girin.

2. *Adım*: Standart tasarruf algoritmasını çalıştır.

3. *Adım*: $D=1$ 'den girilen derinlik değerine kadar; Tasarruf algoritmasını (OPT algoritmalarıyla birlikte) çalıştır, ancak işleme girecek sıradaki tasarrufu listedeki $D+1$ tasarruf arasından rassal olarak seç. Tasarruf algoritmasını bu D değeri için iterasyon sayısı kadar çalıştır.

4. *Adım*: Tasarruf algoritmasının en iyi sonuçlarını kaydet.

Rassal tasarruf algoritmasıyla elde edilen sonuçlar da 2-opt ve Or-opt algoritmalarıyla iyileştirilmiştir. VRP328 yazılımı yardımıyla problemin çözümü sonucunda elde edilen sonuçlar; rota sayısı, kat edilen toplam mesafe, en iyi sonucun elde edildiği iterasyon ve derinlik sayıları, rota sayısı gibi bilgiler Tablo-4'te verilmiştir.

Tablo-4. Algoritma ve rota özet bilgileri

Maksimum Yolcu Sayısı : 67			
Maksimum Derinlik Sayısı : 20		En iyi derinlik : 6	
Maksimum İterasyon Sayısı : 20		En iyi iterasyon : 14	
Durum	Rota Sayısı	Mesafe	En İyi Çözümünden Sapma (%)
Tasarruf Alg.	4	446	4,941
2-Opt. Alg.	4	446	4,941
2-Opt.+ Or.-Opt.Alg.	4	444	4,471
Rassal Tas.Alg.	4	444	4,471
2-Opt. Alg.	4	426	0,235
2-Opt.+ Or.-Opt. Alg.	4	425	0

Tablo-4 ten görüleceği gibi yazılım ile 6. Derinlik (depth) değerinin 14. İterasyonun da en iyi sonuca ulaşılmıştır. Tasarruf ve rassal tasarruf algoritmalarıyla elde edilen rotaların toplam uzunlukları sırasıyla 446 ve 444 birimdir. Tasarruf algoritmasından sonra çözüm geliştirmek için 2-opt algoritması kullanılmış toplam rota uzunluğunda bir iyileşme kaydedilmemiştir. Tasarruf algoritmasından sonra 2-opt ile birlikte Or –opt algoritması kullanılınca toplam mesafede 2 birimlik bir iyileştirme sağlanmış, kat edilen toplam mesafe 444 birime inmiştir. Rassal tasarruf algoritmasından sonra 2-opt tur geliştirme algoritmasının kullanılmasıyla toplam mesafede 18 birimlik bir iyileşme sağlanmış olup, toplam mesafe 426 birime inmiş; 2-opt ve Or-opt algoritmalarının birlikte kullanılması ile en iyi sonuca ulaşılmış, toplam mesafe 425 birime düşmüştür. Tablo-4’ ün en sağ sütununda ise bulunan diğer sonuçların en iyi çözümden % olarak ne kadar saptıkları yer almaktadır.

VRP328 yazılımının çalıştırılması sonucunda elde edilen rota özet bilgileri ve rotalardaki durakların listesi ile ilgili bilgiler birleştirilerek Tablo-5’ de sunulmuştur.

Tablo-5 Rota özet bilgileri ve durak sırası

Rota No	Rota Uzunluğu	Yolcu Sayısı	Durak Sırası
1	153	67	47-11-12-13-14-15-16-17-18-19-20-21-22-23-24-25-6-46
2	83	66	42-27-43-44-45-7-8-9-5-10-4-3-2-40
3	149	67	29-30-31-32-33-34-35-36-37-50-51-57-56-55-54-53-52-49-48-28-38
4	40	20	39-41-26-1
Toplam	425	220	

Tablo-5’ de her bir rotanın uzunluğu, toplam rota uzunluğu, her bir rotada taşınan yolcu sayısı ve her bir rotada ziyaret edilen duraklarla ilgili bilgileri görülmektedir. Bir ve üçüncü rotalardaki araçlar sırasıyla 153 ve 149 birim ile en uzun mesafeyi kat etmişler ve araçlarda boş koltuk kalmamıştır. Dördüncü araç ile 4 adet durak ziyaret edilerek, 20 yolcu taşınmış ve 40 birim mesafe kat edilmiştir. Toplam 268 koltuk kapasitesinden 220 si kullanılmış, 48 kişilik daha boş koltuk mevcuttur. Tablodan da görüleceği gibi 4 araç toplam 425 birim mesafe kat ederek 220 yolcuya hizmet sunmuştur.

SONUÇ ve DEĞERLENDİRME

Çalışmada, Balıkesir Ordudonatım Okulu personel servis araçlarının güzergahları ARP sezgisel çözüm algoritmalarından tasarruf ve rassal tasarruf algoritmaları kullanılarak bulunmuştur. Her iki algoritmayla bulunan sonuçlar, tur geliştirici sezgisel algoritmalarından 2-opt ve Or-opt algoritmaları ile geliştirilmeye çalışılmıştır. Tablo-4 ten görüleceği gibi problem çözümü sonucunda tasarruf algoritmasıyla toplam rota uzunluğu 446 birim, rassal tasarruf algoritması ile bulunan uzunluk ise 444 birimdir (Haritaya göre 1 birim 70 metredir). İki algoritmanın sonuçları arasında çok küçük bir fark görülmektedir. Ancak tasarruf algoritmasından sonra tur geliştirme algoritmalarının uygulanması ile sonuç 444 olmuş, rassal tasarruf algoritmasından sonra tur geliştirici işlemlerin kullanılmasıyla toplam rota uzunluğu 425 birime inmiştir. Problem tek aşamada Mark S. Daskin tarafından geliştirilen VRP328 yazılımı kullanılarak çözülmüştür.

Mevcut durum ile iyileştirilmiş durumun karşılaştırması Tablo-6'da verilmektedir.

Tablo-6 Mevcut durum ile iyileştirilmiş durumun karşılaştırması

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

Tablo-6'dan görüleceği gibi mevcut durumda toplam rota uzunluğu 594 birim, iyileştirilmiş durumda ise 425 birimdir. Toplam rota uzunluğunda 169 birimlik yani %28'lik iyileştirme sağlanmıştır. Bu, günlük sabah ve akşam olmak üzere iki defa servis çıktığı için $2 \times 169 = 338$ birim (23660m) tasarruf anlamına gelmektedir. Toplam mesafenin azalması araçların yıpranması, kaza riskinin

azalması ve zaman bakımından da önemlidir. Uzun bir zaman diliminde ise önemli bir tasarruftur.

Elde mevcut 268 koltuk kapasitesine karşılık 220 yolcuya hizmet sunulmaktadır. Aradaki 48 kişilik fark bir aracın kapasitesi olan 67 den düşük olduğu için araç tasarrufu sağlanamamıştır. Ancak 4. rota için büyük kapasiteli bir araç yerine küçük kapasiteli bir başka araç görevlendirilebilir.

Mevcut durumda güzergahlar yılda bir defa tayin döneminde belirlenmektedir. Ara dönemlerde personel durumunda değişiklik olsa bile manuel olarak rotaların belirlenmesi zaman alıcı bir faaliyet olduğu için yeniden rotalama yapılmamaktadır.

Mevcut durum 4 rotadan oluşmakta ve bir rotada dönem başında belirlenen bir araca düşen yolcu sayısına göre yolcuların 47'si oturarak, 20'si ayakta olacak şekilde taşıma işlemi yapılmaktadır. Çalışmada mevcut durum iyileştirilmeye çalışıldığı için yolcu sayısı 67 olarak alınıp hesap yapılmıştır. Ancak müşteri memnuniyeti de düşünülerek, bütün yolcuların oturarak taşınması istenirse araç kapasitesinin 47 olarak alınması gerekir. Bu durumda rota sayısı yani araç sayısı 5'e çıkmakta ve mesafe 512 birim olarak bulunmaktadır. Araç sayısında artış olmasına rağmen mesafede 82 birim yani % 14'lük iyileştirme kaydedilmiştir.

Çalışmada karşılaşılan ve elde edilen sonuçlara göre gelecekte yapılacak çalışmalarda aşağıdaki durumlar göz önüne alınabilir:

1. Durak yerleri bir modelle belirlenebilir.
2. Uygulama probleminde bazı durakların birbirine çok yakın olduğu görülmüştür. Bazı duraklar birleştirilerek durakların sayısı azaltılabilir. Durak sayısının fazla olması hem personele hem de araçlara zaman kaybettirmektedir.
3. Elde edilen rotalar ayrı ayrı SSP olarak alınıp, SSP optimum çözüm algoritmalarıyla daha iyileştirilebilir.
4. Araç kapasiteleri toplam yolcu sayısından büyük olarak girilerek büyük bir SSP turu oluşturulabilir ve oluşan rota küme bölme yaklaşımlarıyla alt turlara bölünüp SSP olarak çözülebilir.
5. Tasarruf algoritması hızlı çözüm verdiği için askeri alanda uçar birlik hareketi planlamasında kullanılabilir.

KAYNAKLAR

1. LAPORTE, G., NOBERT, Y. and TAILLEFER, S., "Solving a Family of Multi-Depot Vehicle Routing and Location Routing Problems", Chaier du Gerad G-87-10, Ecole des Haltes Etudes Commercialse de Montreal, 1987.
2. SAVELSBERGH, M. W. P. and SOL, M., "The General Pick-up and Delivery Problem", *Transportation Science* 29, 17-29, 1995.
3. BOWERMAN, R. ; HALL, B. and CALOMAI, P. " A multi-objective Optimization Approach to Urban School Bus Routing: Formulation and Solution Method", *Transportation Research Part A: Policy and Practice*, 29(2), 107-123, 1995.
4. MAGNANTI, T. L., "Combinatorial Opt. And Vehicle Fleet Planning: Perspectives and Prospects", *Networks* 11, 179-214, 1981.
5. FISHER, M. L. and JAIKUMAR, R., "A Generalised Assignment Heuristic for Vehicle Routing", *Networks* 11, 109-124, 1981.
6. CLARKE, G. and WRIGHT, J., "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points", *Operations Research* 12(4), 568-581, 1964.
7. DANTZIG, G. and RANSER, J., "The Truck Dispatching Problem", *Management Science* 4, 407- 430, 1957.
8. GASKELL, T., "Bases for Vehicle Fleet Scheduling", *Operations Research Quarterly* 26, 717-734, 1975.
9. TILLMAN, F. and CAIN, T., "An Upper bounding Algorithm for the Single and Multiple Terminal Problem", *Management Science* 11, 664-682, 1972.
10. WREN, A. and HOLLIDAY, A., "Computer Scheduling of Vehicles from one or more Depots to a Number of Delivery Points", *Operations Research Quarterly* 3, 333-344, 1972.
11. YELLOW, P. , "A Computational Modification to the Savings Method of Vehicle Scheduling", *Operations Research Quarterly* 3, 281, 1970.
12. LIN, S., "Computer Solutions of Travelling Salesman Problem", *Bell System Technical Journal* 44, 2245-2269, 1965.

13. LENSTRA, J. K. and RINNOOY KAN, A. H. G. , “Some Simple Applications of the Travelling Salesman Problem”, Operational Research Quarterly 26, 717-734, 1975.
14. GILLET, B. and MILLER, L., “A Heuristic Algorithm for the Vehicle Dispatching Problem”, Operations Research, 22, 340-349, 1974.
15. CHRISTOFIDES, N., MINGOZZI A. And TOTH, P., “Exact Algorithms for the TSP with Additional Constraints” , Euro-4 Cong. Cambridge, England, 1980.
16. GOLDEN, B., “A Statistical Approach to the TSP”, Networks 7, 209-225, 1977.
17. www.users.iems.nwu.edu/~msdaskin/