

**T.C.  
BALIKESİR ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI**

**TOPLAMA VE DAĞITIM ZAMAN PENCERELİ ARAÇ ROTALAMA  
PROBLEMİ İÇİN KESİN ÇÖZÜM YAKLAŞIMI VE ÖRNEK  
UYGULAMALAR**

**YÜKSEK LİSANS TEZİ**

**Tuba TEZER**

**Balıkesir, Temmuz-2009**

T.C.  
BALIKESİR ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI


TOPLAMA VE DAĞITIM ZAMAN PENCERELİ ARAÇ ROTALAMA  
PROBLEMİ İÇİN KESİN ÇÖZÜM YAKLAŞIMI VE ÖRNEK  
UYGULAMALAR


YÜKSEK LİSANS TEZİ


Tuba TEZER

Tez Danışmanı: Prof. Dr. Ramazan YAMAN

Sınav Tarihi: 29.07.2009

Jüri Üyeleri: Prof. Dr. Ramazan YAMAN (Danışman-BAÜ) 

Yrd. Doç. Dr. Yakup KARA (Selçuk Üniversitesi) 

Yrd. Doç. Dr. Necati ÖZDEMİR (BAÜ) 

Balıkesir, Temmuz-2009

## ÖZET

### TOPLAMA VE DAĞITIM ZAMAN PENCERELİ ARAÇ ROTALAMA PROBLEMİ İÇİN KESİN ÇÖZÜM YAKLAŞIMI VE ÖRNEK UYGULAMALAR

**Tuba TEZER**

**Balıkesir Üniversitesi, Fen Bilimleri Enstitüsü  
Endüstri Mühendisliği Anabilim Dalı**

**(Yüksek Lisans Tezi / Tez Danışmanı: Prof. Dr. Ramazan YAMAN)**

**Balıkesir, 2009**

Bu çalışmada, birçok toplama noktalarının ve kargo şubesi ya da depo olarak ifade edilen yalnızca tek bir genel dağıtım noktasının bulunduğu, statik-deterministik toplama ve dağıtım zaman pencereli araç rotalama problemi ele alınmıştır. Bu problem ile zaman pencereli araç rotalama problemi arasındaki farklar ve benzerlikler incelenmiştir. Problemin çözümüne yönelik iki aşamalı bir kesin çözüm algoritması geliştirilmiştir. Bu algoritmanın, birinci aşamasında dal ve sınır ile çözülen derinlik öncelikli arama prosedürüyle oluşabilecek baskın olmayan tüm uygun rotalar elde edilmiştir. İkinci aşamada ise, modeli kabul edilebilir zamanda çözebilmek için, küme bölümlene formülasyonunun yalnızca kolonların alt kümelerini içeren kısıtlanmış versiyonu kolon üretimi tekniği uygulanarak çözülmüştür. Geliştirilen kesin çözüm yaklaşımı ile Lin [24] ve Solomon [44]' da bulunan bazı test problemleri için sonuçlar elde edilmiş ve yorumlanmıştır.

**ANAHTAR SÖZCÜKLER:** Araç Rotalama, Toplama ve Dağıtım Zaman Pencereleri, Kesin Çözüm, Derinlik Öncelikli Arama, Küme Bölümlene, Kolon Üretimi

## **ABSTRACT**

### **AN EXACT APPROACH FOR A VEHICLE ROUTING PROBLEM WITH PICKUP AND DELIVERY TIME WINDOWS AND SAMPLE APPLICATIONS**

**Tuba TEZER**

**Balikesir University, Institute of Science, Department of Industrial Engineering**

**(M.Sc. Thesis / Supervisor: Prof. Dr. Ramazan YAMAN)**

**Balikesir - Turkey, 2009**

In this thesis, a static-deterministic vehicle routing problem with pickup and delivery time windows, which include many pickup points and only one delivery point that called cargo agent or depot, is studied and is compared with the vehicle routing problem with time windows. Two phase exact solution algorithm is applied for solving current problem. In first phase, recessive feasible routes which would be formed depth first search solved by branch and bound was obtained. In second phase, restricted version that is included only subset of column of set partitioning formulation was solved by implemented column generation. Improved exact solution approach was applied on some of the test problems from Lin (2008) and Solomon's benchmark problems.

**KEY WORDS:** Vehicle Routing, Pickup and Delivery Time Windows, Exact Solution, Depth First Search, Set Partitioning, Column Generation.

## İÇİNDEKİLER

ÖZET, ANAHTAR SÖZCÜKLER.....	ii
ABSTRACT, KEY WORDS.....	iii
İÇİNDEKİLER.....	iv
KISALTMALAR LİSTESİ .....	vi
ŞEKİL LİSTESİ .....	vii
ÇİZELGE LİSTESİ .....	viii
ÖNSÖZ.....	ix
1. GİRİŞ .....	1
2. OPTİMİZASYON KAVRAMI.....	5
2.1 Optimizasyonun Tanımı.....	5
2.2 Optimizasyon Problemlerinin ve Çözüm Yaklaşımlarının Sınıflandırılması.....	7
2.3 Kombinatoryal Optimizasyon Kavramı .....	9
2.4 Karmaşıklık Teorisi.....	10
2.4.1 Polinom-Zaman Çözülebilirliği.....	10
2.4.2 P.....	10
2.4.3 NP.....	10
2.4.4 NP- Tam Problemler .....	11
2.5 Kombinatoryal Optimizasyon Problemi Tanımı .....	13
2.5.1 Kombinatoryal Optimizasyon Problemlerinin Karmaşıklığı .....	14
2.5.2 Kombinatoryal Optimizasyon Problemlerinin Çözüm Karmaşıklığına Göre Sınıflandırılması .....	14
2.5.2.1 P Sınıfındaki Kombinatoryal Optimizasyon Problemleri .....	15
2.5.2.2 NP Sınıfındaki Kombinatoryal Optimizasyon Problemleri .....	16
3. ARAÇ ROTALAMA PROBLEMİ.....	17
3.1 Araç Rotalama Probleminin Tanımı .....	17
3.2 Araç Rotalama Problemi için Çözüm Yöntemleri .....	21
3.2.1 Kesin Algoritmalar .....	22
3.2.1.1 Ağaç Arama Yaklaşımları.....	23
3.2.1.2 Dinamik Programlama .....	25
3.2.2 Sezgisel Algoritmalar .....	25

3.2.2.1	Klasik Sezgisel Algoritmalar .....	26
3.2.2.2	İleri Sezgisel Algoritmalar .....	27
3.3	Araç Rotalama Problemi Türleri .....	28
3.3.1	Kapasite Kısıtlı Araç Rotalama Problemi .....	31
3.3.2	Dağıtım ve Toplama Taleplerinin Farklı Müşterilere Ait Olduğu Araç Rotalama Problemi .....	34
3.3.3	Toplama ve Dağıtım ile Araç Rotalama Problemi (TDARP) .....	35
3.3.4	Periyodik Araç Rotalama Problemi.....	37
3.3.5	Çok Depolu Araç Rotalama Problemi.....	37
3.3.6	Karışık Filolu Araç Rotalama Problemi.....	38
3.3.7	Yer Bağımlı Araç Rotalama Problemi .....	38
3.3.8	Zaman Pencere Araç Rotalama Problemi .....	38
3.3.8.1	Zaman Pencere Araç Rotalama Problemi Formülasyonu .....	39
4.	<b>KARGO SERVİSİ İÇİN TOPLAMA VE DAĞITIM ZAMAN PENCERELİ ARAÇ ROTALAMA PROBLEMİ VE KESİN ÇÖZÜM YAKLAŞIMI</b> .....	42
4.1	Problem Tanımı.....	42
4.2	Toplama ve Dağıtım Zaman Pencere Araç Rotalama Problemi için Model .....	45
4.3	Mevcut Problemin Zaman Pencere Araç Rotalama Problemi ile Karşılaştırılması .....	46
4.4	Zaman Pencere Araç Rotalama Problemi için Literatür Araştırması ...	47
4.5	Toplama ve Dağıtım Zaman Pencere Araç Rotalama Problemi için Uygulanan Çözüm Yaklaşımı Adımları .....	51
4.5.1	Toplama ve Dağıtım Zaman Pencere Araç Rotalama Problemi için Model Kabulleri .....	51
4.5.2	Kesin Rotaların Oluşturulması için Derinlik Öncelikli Arama .....	53
4.5.3	Küme Bölümleme ve Kolon Üretimi .....	56
4.6	Örnek Problem Uygulamaları .....	59
4.6.1	Test Problemlerinin Karakteristikleri.....	63
4.6.2	Tasarlanan Modelin Çözümü .....	63
5.	<b>SONUÇ ve GELECEK ARAŞTIRMA</b> .....	67
	<b>KAYNAKLAR</b> .....	69

## KISALTMALAR LİSTESİ

<u>Adı</u>	<u>Tanımı</u>
KOP	Kombinatoryal Optimizasyon Problemleri
P	Polinomial Time
NP	Nondeterministically Polynomial-Time
NPO	KOP için NP' nin karşılığı
NP-tam	Nondeterministically Polynomial-Time-complete (tam)
KAP	Karesel Atama Problemi
GSP	Gezgin Satıcı Problemi
ÇP	Çizelgeleme Problemi
YSP	Yer Seçimi Problemi
KTYSP	Kapasite Kısıtlı Tesis Yer Secimi Problemi
KPP	Kutu Paketleme Problemi
DP	Doğrusal Programlama
ARP	Araç Rotalama Problemi
KKARP	kapasite kısıtlı araç rotalama problemi
MKARP	Mesafe Kısıtlı Araç Rotalama Problemi
ZPARP	Zaman Pencereci Araç Rotalama Problemi
BARP	Dağıtım ve Toplama Taleplerinin Farklı Müşterilere Ait Olduğu Araç Rotalama Problemi
ABARP	Asimetrik Dağıtım ve Toplama Taleplerinin Farklı Müşterilere Ait Olduğu Araç Rotalama Problemi
BZPARP	Zaman Pencereci Dağıtım ve Toplama Taleplerinin Farklı Müşterilere Ait Olduğu Araç Rotalama Problemi
TDARP	Toplama ve Dağıtım ile Araç Rotalama Problemi
ETDARP	Eş Zamanlı Toplama ve Dağıtım ile Araç Rotalama Problemi
ZPTDARP	Zaman Pencereci ve Toplama ve Dağıtım ile Araç Rotalama Problemi
PARP	Periyodik Araç Rotalama Problemi
ÇDARP	Çok Depolu Araç Rotalama Problemi
KFARP	Karışık Filolu Araç Rotalama Problemi
m-	Çoklu (Multi)
GRASP	Greedy Rastgele Uyarlamalı Arama Prosedürü
YEZPARP	Yarı Esnek Zaman Pencereci Araç Rotalama Problemi

## ŞEKİL LİSTESİ

Şekil 1.1 Kargo taşımacılığında servis ağı.....	2
Şekil 2.1 Problemlerin <i>NP-tam</i> olduğunu gösteren indirgeme şeması.....	12
Şekil 3.1 Araç Rotalama Probleminin Biçimsel Gösterimi (Graf).....	18
Şekil 3.2 ARP sınıfının temel problemleri ve birbirleriyle bağlantıları.....	29
Şekil 4.1 Lin (2008) örnek problemi için optimum çözüm.....	61



## ÇİZELGE LİSTESİ

Tablo 3.1 Karakteristiklerine göre ARP türleri.....	30
Tablo 4.1 Toplama ve dağıtım zaman pencereli araç rotalama probleminin sınıflandırması.....	44
Tablo 4.2 Lin (2008) örnek problemine ait müşterilerin karakteristikleri.....	60
Tablo 4.3 Lin (2008) örnek problemine ait müşteriler arası seyahat süreleri.....	60
Tablo 4.4 Lin (2008) örnek problemi için derinlik öncelikli arama prosedürüyle elde edilen rotalar.....	61
Tablo 4.5 Tasarlanan modelin Lin (2008) örnek problemi için optimum çözümü....	62
Tablo 4.6 Tasarlanan algoritmanın Lin(2008) test problemleri için sonuçları.....	64
Tablo 4.7 Tasarlanan algoritmanın Solomon test problemleri için sonuçları.....	66

## ÖNSÖZ

Bir gün kamyon şoförlerinin manevi lideri olacağım kimin aklına gelirdi ki... İlginç bir mizah anlayışına sahip olan çok sevgili kardeşim Halil' in, her çalışmanın olduğu gibi yorucu, ciddi emek gerektiren ve bir o kadar da stresli tez çalışmamın sonunda bana layık gördüğü unvan bu. Her ne kadar ciddi bir bilimsel çalışma olduğunu kabul ettiremesem de araç rotalama problemi üzerinde çalıştığım için asla pişman değilim ve bu konunun önemi ve ciddiyetini kabul ettirmekte hala kararlıyım.

Bu çalışma neşeli bir şekilde okunmaya başlasın istedim. Şimdi gerçeklere dönme ve teşekkür etme zamanı...

Her şeyden önce bana hayata karşı farklı ve geniş ufuklu bir bakış açısı kazandıran, mücadele etmenin gereği ve önemini her zaman hatırlatan, tüm bilgi ve tecrübelerini hiçbir zaman esirgemeyen değerli hocam Prof. Dr. Ramazan Yaman' a bana kazandırdığı her şey için sonsuz teşekkürlerimi sunarım.

Yapmış olduğu çalışma ve yardımlarıyla bu tezin gelişmesinde büyük katkıları olan Hong Kong City Üniversitesi öğretim üyesi değerli Yrd. Doç. Dr. Ka Yuk Carrie Lin' e teşekkür ve şükranlarımı sunmayı bir borç bilirim.

Aras Kargo Balıkesir Çarşı Şubesi Sorumlusu Bekir Uçmuş' a değerli katkılarından dolayı teşekkürler...

Sevgili babam, annem, ablam, Mustafa abim, canım kardeşim Halil, biricik kuzenlerim Emre ve Merve sizlere olan sevgimi, bağlılığımı ve hayatta bana verdiğiniz sevgi ve değer için şükranlarımı anlatacak sözler var mı bilmiyorum.

Beni kızları olarak bağrılarına basan, kocaman bir aile olmamızı sağlayan, yardım ve desteklerini esirgemeyen eşimin çok değerli ailesine de sonsuz sevgi ve şükranlarımı sunarım.

Ve benim kocaman yürekli, sabırlı ve anlayışlı eşim Emre' ye bu çalışma boyunca gösterdiği üstün sabır ve anlayıştan dolayı *yılın eşi* seçildiğini söylemekten gurur duyarım. Sen bir tanesin.

Balıkesir, 2009

Tuba TEZER

## 1. GİRİŞ

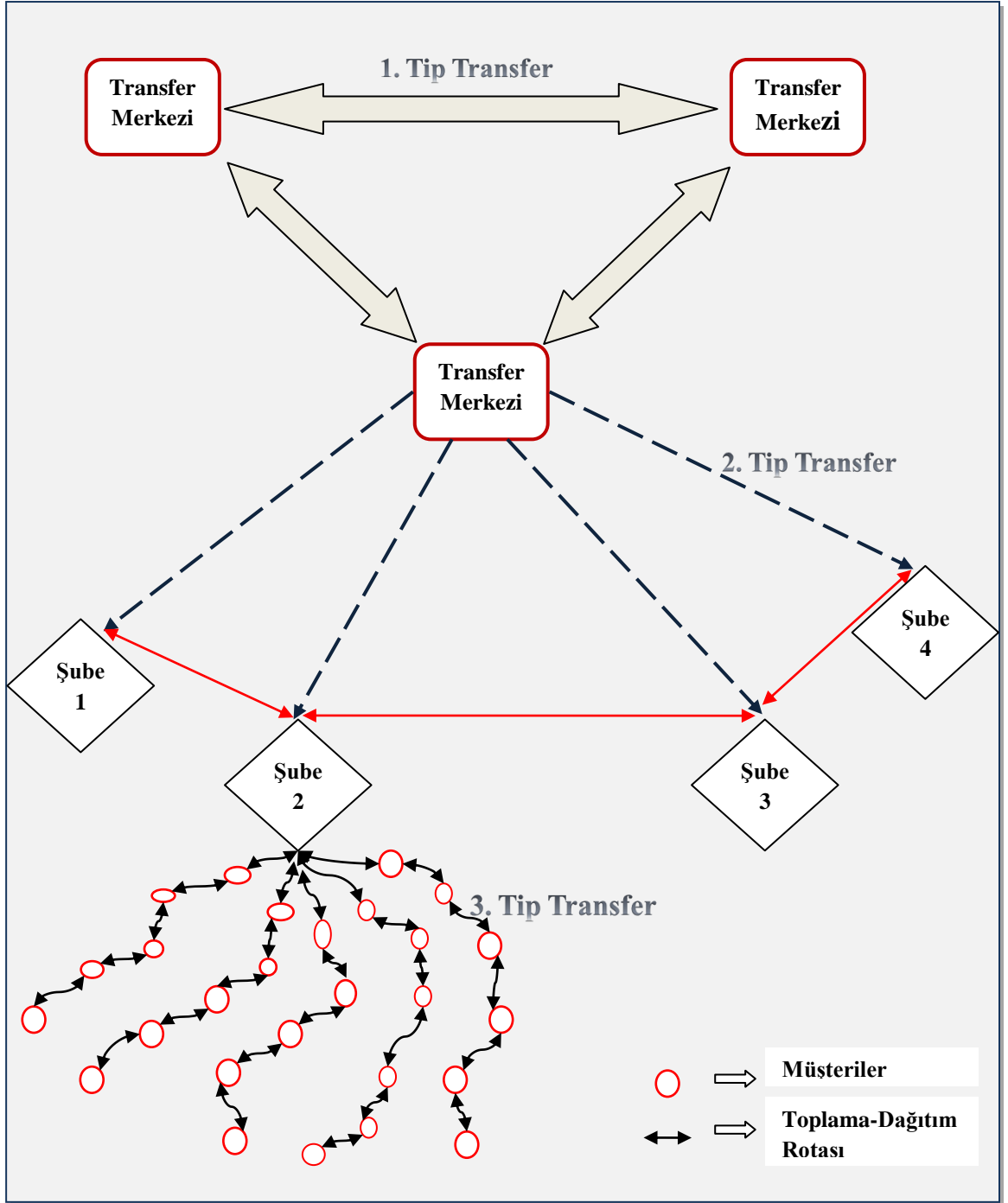
Taşımacılık faaliyetlerinin yoğunluğu ve kalitesi toplumların gelişmişlik düzeyini ve ekonomisindeki canlılığını gösteren en önemli göstergelerden biridir. Bu anlamda ülkemiz her geçen gün konumu ve dünya ekonomisinde hızla yükselen önemi nedeniyle ulusal ve uluslararası taşımacılık sektöründe Orta Doğu' nun lojistik merkezi haline gelmektedir. Lojistik faaliyetlerin önemli ve büyük bir parçası da malların stoklama gereği duyulmadan, ihtiyaç duyulduğu zamanda ve ihtiyaç duyulduğu miktarlarda alınarak talep noktalarına iletilmesi esasına dayanan kargo taşımacılığıdır[1].

Günümüzde, taşımacılık faaliyetlerinde yalnızca maliyet unsurunun değil müşteri memnuniyeti ve servis hızını arttırmanın da firmaların başlıca hedefi haline gelmesinden dolayı bu çalışmada kargo taşımacılığı uygulamaları model alınmıştır.

Ülkemizde kargo taşımacılık ağı, hem yurtdışından gelen hem de yurtiçinde farklı noktalardan çıkan gönderilerin en kısa sürede Türkiye' nin her noktasına ulaştırılmasını sağlayacak şekilde yapılandırılmıştır. Bu yapının bölgesel ayağı kargo taşımacılık firmalarının şube ya da irtibat bürosu adı verilen birimleriyle sağlanmaktadır. Kargo taşımacılığında bu birimler, taşıma servisinin başlangıcında ve sonunda adresten alım ve adrese teslim prensibiyle müşterilere direkt olarak servisi sağlamaları nedeniyle servis ağının en önemli parçalarıdır.

Kargo taşımacılığında servis ağları transfer merkezleri, şube veya acente olarak ifade edilen irtibat büroları ve araçlardan oluşan kompleks bir yapıya sahiptir. Şekil 1.1' de gösterildiği gibi, bu yapı işleyiş açısından üçe ayrılarak değerlendirilebilir. Birinci kısım, kargoların bölgeler arası yolculuğunu yani başlangıç noktasına en yakın olan transfer merkezine gönderilip oradan varış noktasına en yakın olan transfer merkezine iletildiği, bir transfer merkezinden diğerine yapmış olduğu seyahati içermektedir. İkinci kısım, kargonun varış

noktasına en yakın transfer merkezinden ilgili şubeye gönderildiği seyahati, üçüncü kısım ise kargonun varış noktasına en yakın olan şubeden varış adresine iletiildiği nispeten daha kısa mesafeli ancak rotalanması açısından çok daha karmaşık olan seyahati içermektedir.



Şekil 1.1 Kargo taşımacılığında servis ağı

Kargo taşımacılığı şirketlerinin servis ağlarında tüm irtibat büroları yani şube ve acenteler ait oldukları bölgeye en yakın olan transfer merkezine atanmışlardır. Bir şube yalnız bir transfer merkezine atanmıştır ve şubeler arası direkt kargo gönderimi yoktur. Şubeler arasında kargonun taşınması transfer merkezleri vasıtasıyla sağlanmaktadır ve bu transfer yapısı transfer merkezleri arasında da bağlantıların olması anlamına gelmektedir. Ancak kargonun servis ağındaki seyahati boyunca yalnızca başlangıç ve varış noktalarının ait olduğu bölgelere atanmış olan transfer merkezlerine gönderilmeleri dolayısıyla tüm transfer merkezleri arasında direkt (hub-to-hub) bağlantı yoktur. Kargonun en kısa yoldan en düşük maliyetle seyahatini sağlayacak şekilde transfer merkezleri konumlandırılmıştır.

Araç rotalama probleminde hareket merkezi ya da depo olarak tanımlanan, araçların servislerine başladıkları ve servislerini tamamladıktan sonra geri döndükleri birimler, kargo taşımacılığında transfer merkezi veya şube olarak tanımlanmıştır. Transfer merkezleri tüm gelen ve giden kargoların toplandığı ve varış adreslerine göre ayrılarak ilgili transfer merkezine ya da şubeye gönderildikleri noktaldır. Transfer merkezleri arasındaki taşıma nispeten daha uzun mesafe dolayısıyla daha uzun seyahat zamanı ve de daha fazla yük miktarını içermektedir.

Kargo taşımacılığında şube ve acente ise müşterilerin gönderi taleplerini alan ve hizmet bölgelerindeki adreslere gönderilmiş olan gönderilerin ilgili adreslere iletilmesini sağlayan irtibat bürolarıdır. Ulusal ya da uluslararası kargo taşımacılığında direkt olarak gönderen ve alıcılara (yani müşterilere) toplama ve dağıtım hizmetini gerçekleştiren yerel birimler şubelerdir. Bu anlamda şubeler yine ARP' ndeki depoların karşılığı olarak düşünülebilir. Benzer şekilde gönderen ve alıcılarda ARP' ndeki müşterilere karşılık gelmektedir.

Kargo servis ağında transfer merkezleri arasındaki taşıma bu merkezlerin konumları belirli ve buna bağlı olarak araçların transfer merkezleri arasında kat edecekleri mesafe ve seyahat süresi belirli olduğundan esas karmaşık ve dar boğaza sebep olan, direkt olarak müşteri konumlarından gönderilerin toplanmasıdır. Çünkü bu, transfer taleplerinin iletildiği, yerleşim yerlerinde konumlandırılmış şubelerin, kendi servis bölgelerinde bulunan çok sayıda ve farklı koordinatlardaki müşteri

konumlarına en hızlı ve en düşük maliyetli bir biçimde servis etmelerini ve de toplanan gönderilerin alıcı adreslerine tam zamanında ulaştırılabilmeleri için belirlenen zamanda şubeye getirilip, transfer merkezlerine iletilmelerini gerektirmektedir. Bu, çok sayıda müşteri konumlarından toplama ve şubeden dağıtım problemi, bu tezin konusu olan *toplama ve dağıtım zaman pencereci araç rotalama problemi*dir.

Bu çalışmadaki örnek uygulamalarda kargo dağıtım şirketlerinin rotalaması daha karmaşık ve güç olan bölgesel ayağı model alınarak şube adı verilen birimler ARP' ndeki genel dağıtım noktası olan depolar, müşteri konumları da toplama noktaları olan düğümler olarak değerlendirilmiştir. Her bir müşterinin gönderilerinin hazır olduğu ve alınmasını talep ettikleri süreler, sırasıyla en erken ve en geç toplama zamanı penceresini, müşterilerden toplanan gönderilerin varacakları son adreslere vaktinde teslimatı için depoya (şubeye) ulaştırılmaları gereken süreler de dağıtım zamanı penceresini meydana getirmektedir.

Bu çalışmanın amacı, kargo taşımacılığında gecikmelerden kaynaklanan maliyetleri ve müşteri memnuniyetsizliklerini önlemek amacıyla araçların, tüm müşterilere servis için gerçekleştirecekleri rotaları planlayarak seyahat süresi ve maliyetini optimize etmektir.

Çalışmanın ikinci bölümünde genel optimizasyon kavramı incelenip çözüm yaklaşımları değerlendirilmiş ve bu problemin ait olduğu kombinatoriyal optimizasyon problemleri incelenmiştir. Üçüncü bölümde, araç rotalama problemleri ve çözüm yöntemlerine genel bir bakış sunularak ilgili sınıflandırma verilmiştir. Dördüncü bölümde, mevcut problemin tanımı, zaman pencereci araç rotalama problemi ile kıyaslanması, literatür araştırması verildikten sonra toplama ve dağıtım zaman pencereci araç rotalama probleminin çözümü için geliştirilen model ayrıntılarıyla sunulmuştur. Örnek problemlerin çözümleri hazırlanan algoritmaya dayalı olarak sonuçlarıyla birlikte verilmiştir. Beşinci ve son bölümde ise sonuçlar değerlendirilerek, geliştirilen kesin çözüm yaklaşımının sonraki çalışmalar için olası katkılarına değinilmiştir.

## 2. OPTİMİZASYON KAVRAMI

### 2.1 Optimizasyonun Tanımı

Matematiksel modelleme tekniği öncelikle doğrusal ve az sayıda değişkenlerin kullanılmasıyla başlamıştır. Bir süre sonra doğrusallık varsayımının her problem için geçerli olmadığı anlaşılmıştır. Bu durumda doğrusal olmayan modellemeye gidilmiştir. Ancak doğrusal olmayan modellerin kendine özgü çözümleri uygulamada birçok sorunu beraberinde getirmiştir. Zamanla geliştirilen bazı yöntemlerle doğrusal olmayan modellerin hızla çözümlenmesi sağlanmış ve bu *optimizasyon* teorisini geliştirmiştir[2].

Optimizasyon teorisi, verilen tanım kümesinde amaç fonksiyonunun minimum ya da maksimum değerlerini arayan problemlerin matematiksel çözümünü araştıran çalışmadır. Bu teori hem çözümlerin varlığı, yapısal özellikleri hem de algoritma yönü üzerine çalışmaları içerir[3].

Optimizasyon gerçek hayat şartlarında birçok olası sonuç içerisinde en iyisini seçmekle ilgilenir[4].

Optimizasyon problemleri üç temel bileşenden oluşmaktadır. Bunlar;

❶ **Amaç fonksiyonu:** Minimize ya da maksimize etmek istediğimiz fonksiyondur. Örneğin, bir üretim prosesinde karı maksimize etme ya da maliyeti minimize etme hedeflenir.

❷ **Bilinmeyenler veya Değişkenler:** Objektif fonksiyonun değerini etkileyen sistem karakteristikleridir. Örneğin, bir üretim probleminde değişkenler, kullanılan değişik kaynakların miktarını ya da her bir iş için harcanan zamanı

içerebildiği gibi panel dizaynı probleminde ise değişkenler, panelin şeklini ve boyutlarını belirlemek için kullanılır.

③ **Kısıtlar:** Değişkenlerin belirli bir aralığın dışında kalan değerleri almasına müsaade etmeyen parametrelerdir. Örneğin, üretim probleminde herhangi bir iş için harcanan zamanın negatif olması mantıksal olarak mümkün olamayacağından tüm zaman değişkenleri pozitif değerler alacak şekilde kısıtlanır.

Tüm bu bileşenlerin doğrultusunda optimizasyon problemi, “*kısıtları sağlayarak amaç fonksiyonunu minimize ya da maksimize eden değişkenlerin değerini bulmaktır*”[5].

Genel optimizasyon problemi aşağıdaki standart formda değerlendirilir:

$$\text{Min } f(c, x) \quad (2.1)$$

Kısıtlar;

$$g_i(a, x) \leq 0, \quad (2.2)$$

$$x \in X, \quad i = 1, \dots, n \quad (2.3)$$

Burada;

- $x$ , değişkenlerin vektörü,
- $X$ , uygun çözümlerin kümesi,
- $f$  ve  $g_i$ , konveks fonksiyonu,
- $c$  ve  $a$ , problem parametreleri (belirsiz) olarak ifade edilmiştir[6].



## 2.2 Optimizasyon Problemlerinin ve Çözüm Yaklaşımlarının Sınıflandırılması

Optimizasyon problemleri farklı kaynaklarda değişik biçimlerde sınıflandırılmıştır. Jongen vd.[3] optimizasyonu sürekli ve kesikli optimizasyon olmak üzere iki katagoride irdelemişlerdir.

Sürekli optimizasyon sınıfında: kısıtlanmamış optimizasyon ve kısıtlanmış problemler için Lagrange teorisi; yarı-sonsuz problemler; doğrusal programlama için üç temel çözüm algoritması (simplex metodu, elipsoid metodu ve orijinal *Karmarkar* iç nokta algoritması); doğrusal olmayan optimizasyon problemleri ve çözüm yöntemleri *steepest descent*, Newton metodu, the Lagrange-Newton ve birleşik gradyan metodu, penaltı, bariyer (engel) ve çarpan (multiplier) metotları ele alınmıştır[3].

Temel (taban) kümelerin sonlu olduğu kesikli optimizasyon sınıfında ise: genel doğrusal programlama algoritmaları ile başarıyla çözülebilen değişkenler için tamsayılı doğrusal programlama ve *NP* sınıfındaki *kombinatorial optimizasyon problemleri* (KOP) irdelenmiştir[3].

Optimizasyon modelleri ve çözümleri için kullanılan temel algoritmalar için NEOS optimizasyon sunucusu tarafından diğer bir sınıflandırma yaklaşımı da şu şekilde verilmiştir[7]:

- ***Kesikli ve Tamsayılı Optimizasyon***
  - Tamsayılı Programlama
    - Karışık Tamsayılı Programlama → Karışık Tamsayılı Doğrusal Olmayan Programlama
    - Stokastik Tamsayılı Programlama
    - Özel Problemler

- **Sürekli Optimizasyon**
  - Kısıtlanmamış Optimizasyon
    - Bağlı Kısıtlı Optimizasyon
  - Kısıtlanmış Optimizasyon
    - Doğrusal olmayan kısıtlı optimizasyon → Doğrusal olmayan eşitlikler → Doğrusal olmayan en küçük kareler
    - Kuadratik programlama → Doğrusal programlama → Network programlama
    - Yarı sonlu programlama
    - Stokastik programlama
    - Global optimizasyon
    - Diferansiyellenemeyen optimizasyon
  - Bütünlük Kısıtları
    - Doğrusal olmayan tümeleme problemleri → Doğrusal tümeleme problemleri
    - Bütünlük kısıtları ile matematiksel programlama
  - Kısmi diferansiyel denklem kısıtlı optimizasyon
  - Yarı Sonsuz Programlama

Optimizasyon problemleri için karar değişkenleri ve kısıtların durumuna göre yapılan sınıflandırmalar doğrultusunda bu çalışmanın konusu olan araç rotalama problemi modeli, kesikli optimizasyon sınıfında yer alan *NP* sınıfına ait kombinatoriyal optimizasyon problemidir. İzleyen kısımlarda, kombinatoriyal optimizasyona ait kavramlar ve bazı detaylar sunulmaktadır.

### 2.3 Kombinatoryal Optimizasyon Kavramı

Kombinatoryal optimizasyon, nesnelerin sonlu toplamında bir optimum nesneyi arar. Genel anlamda toplam, nesnelerin sayısı çok büyük olduğu halde, *graf* gibi özlü bir gösterime sahiptir. Daha açık olarak, nesneler üstel olarak arttığı için tüm nesnelerin birer birer taranıp en iyinin seçilmesi etkin bir metotla gerçekleştirilmelidir.

1960' lı yıllarda Edmonds, bir metodun etkili olduğunu söylemek için, çalışma zamanının temsil büyüklüğünde bir polinom ile sınırlandırılmış olması gerektiği fikrini savunmuştur. O zamandan beri, bu kıstas geniş kabul görmüştür. Çünkü, Edmonds aynı zamanda (eşleştirme problemi gibi) birçok önemli optimizasyon problemi için polinom-zamanlı algoritmaları bulmuştur. Polinom-zamanlı çözülebilir problemler sınıfı  $P$  ile ifade edilmektedir.

1970' li yıllarda, Cook ve Karp' ın (gezgin satıcı problemi gibi) birçok önde gelen kombinatoryal optimizasyon problemini bulmalarıyla, en zor problem sınıfı olan  $NP$  (*nondeterministically polynomial-time*) ortaya çıkmıştır.  $NP$  sınıfı birçok kombinatoryal optimizasyon problemini içermektedir.

Hemen hemen her kombinatoryal optimizasyon problemi polinom-zaman çözülebilirliğinde ya da  $NP$ -tam olarak gösterilmiştir yani bu problemlerden hiçbiri her iki sınıfa birden ait olarak gösterilmemiştir. Bu özelliklerin ayrık ( $P \neq NP$ ) mı yoksa birbiriyle örtüşüyor ( $P = NP$ ) mu olduğu büyük bir muamma olarak dikkat çekmektedir[8, s.1].

## 2.4 Karmaşıklık Teorisi

### 2.4.1 Polinom-Zaman Çözülebilirliği

Bir polinom-zamanlı algoritma, girdi boyutunda bir polinom tarafından sınırlandırılan adım sayısından sonra sonlanan algoritmadır.

Bu tanımlamada girdi boyutu, girdiyi tanımlayan bit sayısı olan girdinin büyüklüğüdür. Eğer bir problem polinom-zamanlı algoritma ile çözülebiliyorsa bu problem *polinom-zamanlı çözülebilir* ya da *polinom-zamanda çözülebilir* denir[8, s.39-43].

### 2.4.2 P

$P$  ve  $NP$  karar problemleri sınıfındadır. Karar problemleri, ‘evet’ ya da ‘hayır’ şeklinde cevaplandırılabilen problemlerdir. Örneğin; verilen bir grafin tamamen eşlenip eşlenemediği ya da graftaki tüm düğümlerden yalnızca bir kez geçen döngü anlamına gelen *Hamilton çevrimi*’ ne sahip olup olmadığı gibi. Bir optimizasyon problemi karar problemi değildir. Fakat sıklıkla, belirlenmiş anlamda karar problemine indirgenebilir.

Tüm polinom-zamanda çözülebilir problemler  $P$  ile ifade edilir[8, s.39-43].

### 2.4.3 NP

$NP$ , pozitif cevaplı her bir girdi için, cevabının doğruluğu polinom zamanda kontrol edilebilir sertifikasına sahip tüm karar problemlerinin toplamı olarak tanımlanır.

Sertifikayı polinom zamanda kontrol etme, orijinal girdi büyüklüğünde bir polinom ile sınırlandırılmış zamanda kontrol etme anlamına gelmektedir.

Diğer bir deyişle  $NP$ , deterministik olmayan polinom-zamanda çözülebilir problem sınıfı anlamına gelmektedir[8, s.39-43].

Deterministik problemler, belirli giriş değerleri kümesiyle çağırıldığında ve aynı veri tabanı konumu verildiğinde daima aynı sonuçları geri döndüren fonksiyonlarla ifade edilen problemlerdir. Deterministik olmayan problemler ise, belirli giriş değerleri kümesiyle çağırıldıklarında eriştikleri veri tabanı konumu aynı olsa bile her safesinde farklı değerler verebilen fonksiyonlarla ifade edilen problemlerdir[9].

#### 2.4.4 NP - Tam Problemler

$NP$ -tam problemler  $NP$  sınıfındaki en zor problemlerdir ve  $NP$  sınıfındaki her problem bunlara indirgenebilir.

$\Sigma^*$  bir sonlu küme olmak üzere, eğer herhangi bir  $\omega \in \Sigma^*$  girdisi için  $\chi \in \Sigma^*$  çıktısını,

$$\omega \in \Pi \Leftrightarrow \chi \in \Lambda$$

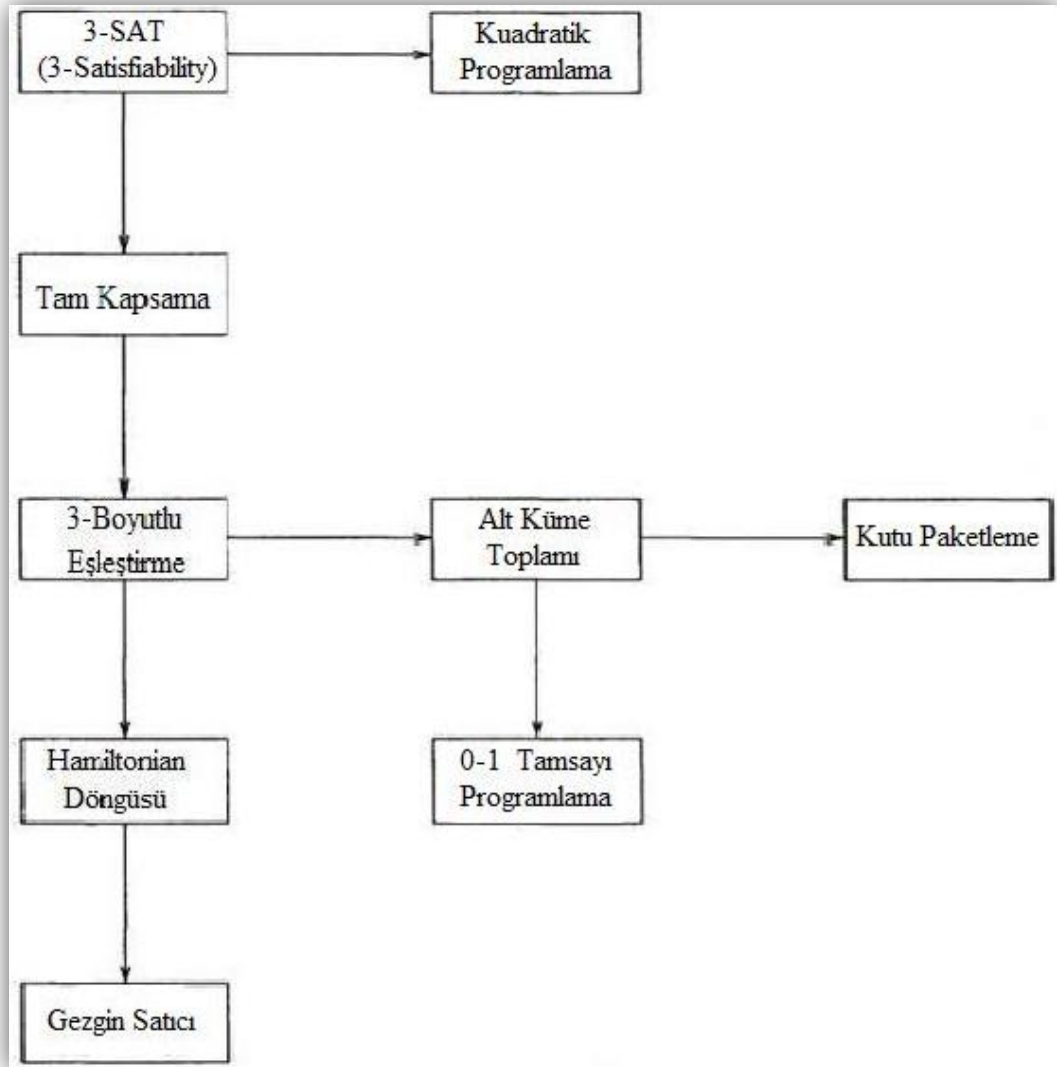
özelliğiyle döndüren bir polinom-zamanlı algoritma varsa,  $\Pi \subseteq \Sigma^*$  problemi  $\Lambda \subseteq \Sigma^*$  problemine indirgenebilir (reducible) denir.

Bu, eğer  $\Pi \in \Lambda'$  ye indirgenebilirse ve  $\Lambda \in P'$  ye aitse,  $\Pi$  de  $P'$  ye aittir anlamına gelir. Benzer olarak, eğer  $\Pi \in \Lambda'$  ye indirgenebilirse ve  $\Lambda \in NP'$  ye aitse,  $\Pi$  ninde  $NP'$  ye ait olacağı anlamına gelir.

Eğer  $NP'$  deki her bir problem,  $\Pi'$  ye indirgenebiliyorsa,  $\Pi$  problemi “ $NP$ -tam”dir denir. Bundan dolayı, eğer herhangi bir  $NP$ -tam problemi  $P$  sınıfındaysa,  $P = NP$  demektir.

Gezgin satıcı problemi, maksimum (*clique*) problemi, maksimum (*cut*) problemi gibi birçok bilinen kombinatoryal optimizasyon problemi *NP-tam* sınıfındadır[8, s.39-43].

Bir problemin diğer problemlerden birine indirgeme yönü Şekil 2.1' de gösterilmiştir. Bir kutudan diğerine çıkan ok, okun çıktığı kutudaki problemin, okun gösterdiği yöndeki probleme polinomsal olarak indirgeneceği anlamına gelmektedir[3, s.313].



Şekil 2.1 Problemlerin *NP-tam* olduğunu gösteren indirgeme şeması

[3, s.314]

## 2.5 Kombinatoryal Optimizasyon Problemi Tanımı

Bir kombinatoryal optimizasyon problemi  $\Pi := (\mathbf{I}, \{Sol(I)\}_{I \in \mathbf{I}}, m)$  minimizasyon ya da maksimizasyon problemidir ve üç kısımdan meydana gelir:

- i. Probleme dair örneklerin kümesi  $\mathbf{I}$ ;
- ii. Her  $I \in \mathbf{I}$  örneği için olası çözümlerin sonlu kümesi  $Sol(I)$ . Bu küme  $I$  örneklerinin uygun çözümlerinin kümesi olarak adlandırılır;
- iii.  $m: \{(I, \sigma) | I \in \mathbf{I}, \sigma \in Sol(I)\} \rightarrow Q_+$  fonksiyonu. Burada  $Q_+$  pozitif rasyonel sayıların kümesidir.  $m(I, \sigma)$  değeri uygun çözüm  $\sigma$ 'nin değeri olarak adlandırılır.

Uygun çözüm  $\sigma^*$ ,  $(\mathbf{I}, \{Sol(I)\}_{I \in \mathbf{I}}, m)$  probleminin bir  $I$  örneği için optimum çözüm olarak adlandırılır; sahip olunan tüm uygun  $\sigma \in Sol(I)$  çözümleri için,

- $m(I, \sigma^*) \leq m(I, \sigma)$  olduğu durumda minimal çözüm değeri ile ilgilenilir ve problem kombinatoryal minimizasyon problemi olarak adlandırılır.
- $m(I, \sigma^*) \geq m(I, \sigma)$  olduğu durumda maksimal çözüm değeri ile ilgilenilir ve problem kombinatoryal maksimizasyon problemi olarak adlandırılır.

Optimum değeri şu şekilde gösterilir;

$$OPT(I) := m(I, \sigma^*).$$

Eğer uygun çözümlerin kümesi  $Sol(I)$  boşsa, optimum değerler,

$$OPT(I) := 1$$

şeklinde tanımlanır[3, s.354].

### 2.5.1 Kombinatoryal Optimizasyon Problemlerinin Karmaşıklığı

$\Pi := (\mathbf{I}, \{Sol(I)\}_{I \in \mathbf{I}}, m)$  kombinatoryal optimizasyon problemi için:

a) Problem  $\Pi$ ' nin *NPO* sınıfına ait olması için gerekli ve yeterli şartlar aşağıda verilmiştir:

- Herhangi bir  $I \in \mathbf{I}$  örneği için,  $Sol(I)$  daki tüm uygun çözümler  $p(size(I))$  tarafından sınırlandırılan bir boyuta sahip olacak şekilde, bir  $p$  polinomu vardır.
- Herhangi bir  $I \in \mathbf{I}$  girdisi ve en çok  $p(size(I))$  boyutunda olan rastgele  $w$  değeri (arbitrary string) için  $w \in Sol(I)$  olup olmadığı polinom zamanda karar verilebilir.
- Ölçü fonksiyonu (measure function)  $m$ , polinom zamanda ( $I$  örneği boyutu ve  $\sigma \in Sol(I)$  uygun çözümünde) hesaplanabilir.

b) Eğer ölçü fonksiyonunun tüm değerlerini üstten sınırlandıran bir  $q$  polinomu varsa, yani;

$$m(I, \sigma) \leq q(size(I)) \forall I \in \mathbf{I}, \sigma \in Sol(I)$$

ise bir  $\Pi \in NPO$  problemi polinomsal olarak sınırlandırılmıştır.

c) Her  $I \in \Pi$  örneği için optimum değer  $OPT(I)$ ' yi hesaplayan bir polinom zamanlı algoritma varsa ve bu  $P = NP$  olmasını gerektiriyorsa (anlamına geliyorsa),  $\Pi \in NPO$  problemi *NP-zor* sınıfındadır[3, s.356-357].

### 2.5.2 Kombinatoryal Optimizasyon Problemlerinin Çözüm Karmaşıklığına Göre Sınıflandırılması

Kombinatoryal optimizasyon problemlerinde sıklıkla iki çeşit problemle ilgilenilir:



- (1) Kolay problemler; bunlar polinomsal olarak sınırlanmış algoritmaların var olduğu problemlerdir;
- (2) Zor problemler; bunlar polinomsal olarak sınırlanmış algoritmaların henüz bulunmadığı problemlerdir

Polinomsal olarak sınırlanmış algoritma, gerektirdiği sayısal işlemlerin sayısının, çözülen problemin büyüklüğünün polinom fonksiyonu olduğu algoritma anlamına gelmektedir. Örneğin;  $n$  boyutlu bir problem için, algoritma  $n^2$  işlem gerektiriyorsa polinomsal olarak sınırlanmış, algoritma  $2^n$  işlem gerektiriyorsa polinomsal olarak sınırlanmamıştır.

*NP-tam* problemleri kümesindeki tek bir problem için, polinomsal olarak sınırlanmış bir algoritma bulunuyorsa, kümedeki tüm problemler için polinomsal olarak sınırlanmış algoritmalar vardır. Bu anlamda *NP-tam* problemleri birbirlerine denk olma özelliğine sahiptirler[10].

### 2.5.2.1 P Sınıfındaki Kombinatoryal Optimizasyon Problemleri

*P* sınıfındaki bir problem; çözüm zamanı problem boyutunun polinom fonksiyonu olarak artan bir algoritma ile çözülebilir. Örneğin,  $n$  düğüme sahip en kısa yol problemi Dijkstra's algoritma ile çözülebilir.

Dijkstra's algoritmasının çözüm zamanı  $O(n^2)$  ile sınırlıdır. Parantez içerisindeki fonksiyon polinom olduğundan dolayı, Dijkstra's algoritması polinom zamanlı bir algoritma ve dolayısıyla *En kısa yol problemi*, *P* sınıfında yer alan bir kombinatoryal optimizasyon problemi (KOP) dir.

*P* sınıfında yer alan diğer KOP' lar:

- *Atama problemi*
- *Minimum yayılan ağaç*
- *Şebeke akış problemleri*[11].

### 2.5.2.2 NP Sınıfındaki Kombinatorial Optimizasyon Problemleri

*NP-zor* sınıfındaki problemlerin çözümü için polinom zamanlı bir algoritma yoktur. 0 ya da 1 değerini alan  $n$  değişkene sahip bir problem için tüm çözümlerin sıralama (enumeration) zamanı  $O(2^n)$ ' dir. Küçük boyutlu problemler sıralama yöntemi ile çözülebilmeye rağmen, büyük boyutlu problemler için bu yöntem ile çözüme ulaşmak mümkün değildir.

*NP-zor* sınıfındaki problem için dal-sınır ya da kesme-düzlemi gibi etkin yöntemlerin de başarısız olmasının nedeni, bu yöntemlerin de üstel sınırlara sahip olmasıdır.

*NP-zor sınıfında yer alan kombinatorial optimizasyon problemlerine örnek olarak aşağıdaki problemler gösterilebilir:*

- Karesel atama problemi (KAP)
- Gezgin satıcı problemi (GSP)
- Araç rotalama problemi (ARP)
- Çizelgeleme problemi (ÇP)
- Yer secimi problemi (YSP)
- Kapasite kısıtlı Tesis Yer Secimi Problemi (KTYSP)
- Kutu Paketleme Problemi (KPP)[11].

Kombinatorial optimizasyon problemlerinin *NP-zor* sınıfında tanımlanan, gerçek hayat problemlerinde de yaygın olarak karşımıza çıkan araç rotalama problemleri üzerinde birçok çalışma yapılmıştır ve halen de bu çalışmalar problemin birçok değişik türü için çözüm yaklaşımları geliştirmek amacıyla yoğun olarak devam etmektedir. Araç rotalama problemleriyle ilgili temel kavramlar, çözüm yaklaşımları ve sınıflandırmaları izleyen bölümde daha fazla ayrıntılarıyla incelenecektir.

### 3. ARAÇ ROTALAMA PROBLEMİ

#### 3.1 Araç Rotalama Probleminin Tanımı

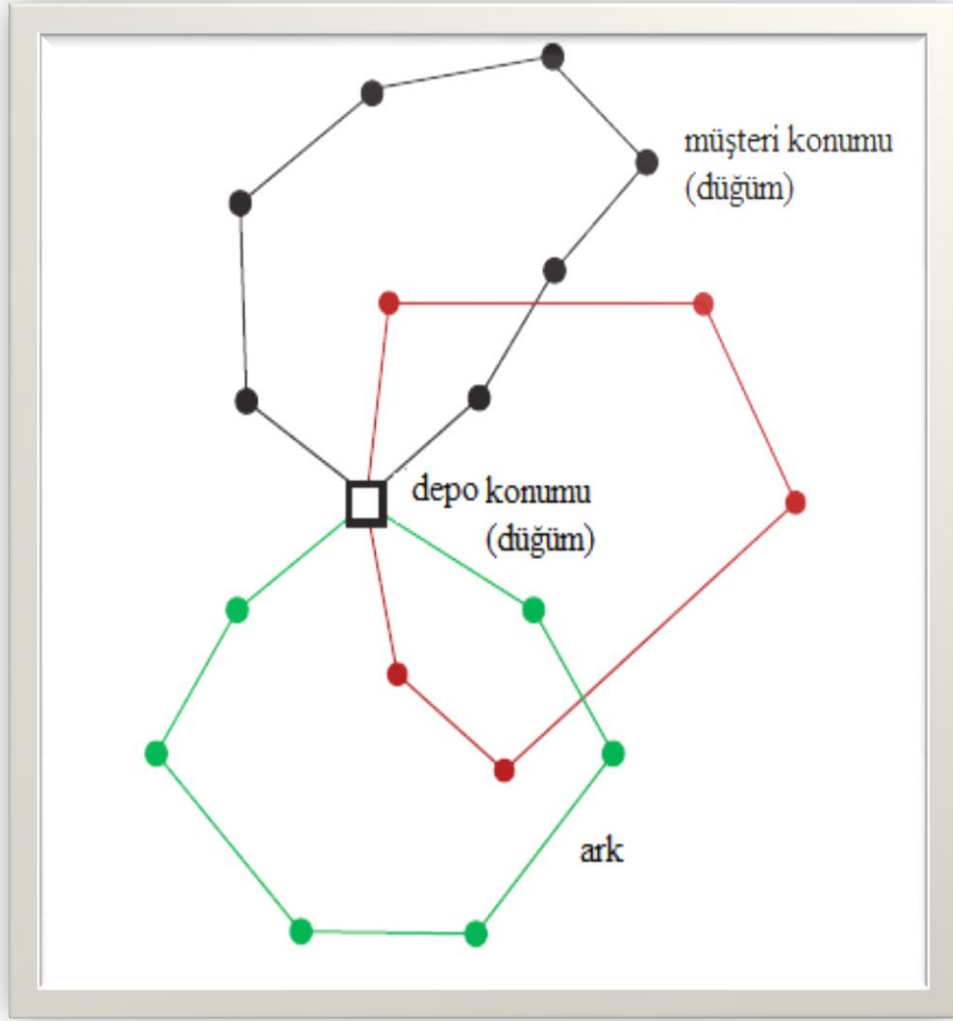
Genel olarak, malların depo ve son kullanıcılar (müşteriler) arasında dağıtımıyla ilgilenen problemler, araç rotalama problemi (ARP) ya da araç programlama problemi olarak bilinir[12, s.1].

Diğer bir ifadeyle, araç rotalama problemleri (ARP), coğrafi olarak dağıtım merkezlerine bir veya birden fazla depodan servis etmek üzere görevlendirilen araçların optimum dağıtım ve toplama rotalarının planlanması problemleridir[13].

Araç rotalama probleminin çözümü, her biri kendi deposundan hareket eden ve yine kendi deposuna dönen bir araç tarafından gerçekleştirilen, öyle ki müşterilerin tüm gereksinimlerinin yerine getirildiği, tüm operasyonel kısıtların karşılandığı ve global taşıma maliyetinin minimize edildiği, rotaların kümesinin belirlenmesini gerektirir.

Rotalama problemlerinin ana bileşenleri yol ağı (network), müşteriler, depolar, araçlar ve sürücülerdir.

Gönderilerin taşınması için kullanılan *yol ağı* genellikle, “yay (*ark*)”ları yol kısımlarını temsil eden ve “köşe”leri yol bağlantı noktalarına, depo ve müşteri konumlarına karşılık gelen **graf** olarak ifade edilir. Yaylar ve dolayısıyla bunlara ilişkin graflar, sadece bir yönlü mü yoksa her iki yönlü mü geçilebildiklerine göre sırasıyla, yönlü ya da yönsüz olabilirler. Her bir yay genel olarak uzunluğunu ifade eden maliyet ve araç tipine veya yayın kat edilme zamanına bağlı olabilen seyahat mesafesi ile ilişkilendirilmiştir. Şekil 3.1’de ARP için örnek bir graf gösterilmiştir.



Şekil 3.1 Araç rotalama probleminin biçimsel gösterimi (Graf)

Araç rotalama problemlerinde müşterilerin tipik karakteristikleri şunlardır:

- Müşterilerin konumlandığı yol ağının noktaları,
- Dağıtılması ya da müşterilerden toplanması gereken farklı türlerde olabilen malların (taleplerin) miktarı,
- Müşterilerin servis edilebildiği günlük periyot (zaman penceresi) (örneğin; müşterilerin açık bulunma zamanları ya da trafik sınırlamalarına bağlı olarak müşteri mevkilerine ulaşma zamanı gibi belirli zamanlar nedeniyle),

- Araç tipine bağlı olabilen, gönderileri müşteri konumlarına dağıtmak ya da müşteri konumlarından toplamak için gerekli olan zaman (boşaltma ve yükleme zamanları),
- Müşterilere servis için kullanılacak uygun araçların alt kümesi (örneğin, muhtemel kullanım sınırlamaları ya da yükleme veya boşaltma gereksinimleri nedeniyle).

Bazen, her bir müşterinin talebini tamamen karşılamak mümkün değildir. Böyle durumlarda dağıtılacak ya da toplanacak miktarlar düşürülebilir ya da bazı müşterilere servis yapılamayabilir. Bu durumla ilgili olarak, servisin kısmi ya da toplam eksikliği ile ilgili farklı öncelikler ya da yaptırımlar müşterilere atanabilir.

Müşterilere servis yapmak için gerçekleştirilen rota/yollar, yol ağının köşelerinde konumlanmış bir ya da daha çok depoda başlar ve sonlanır. Her bir depo kendisine ait araçların sayısı ve cinsiyle ya da dağıtabildiği malların global miktarıyla nitelendirilmiştir. Bazı gerçek hayat uygulamalarında müşteriler önceden depolara göre gruplandırılmışlardır ve araçlar rotalarının sonunda, ait oldukları depoya dönmek zorundadırlar. Bu durumda, geniş kapsamlı ARP, her biri farklı bir depoya ilgili olan birkaç bağımsız probleme ayrıştırılabilir.

Gönderilerin taşınması, bileşenleri ve büyüklüğü sabit olabilen veya müşterilerin gereksinimlerine bağlı olarak belirlenebilen bir araç filosu kullanılarak gerçekleştirilir. Kullanılan araçların tipik karakteristikleri ise şunlardır:

- Aracın ait olduğu depo ve servisin aracın ait olduğu depodan başka bir depoda sonlanma ihtimali,
- Aracın yüklenebileceği maksimum ağırlık, değer ya da paletlerin sayısı olarak ifade edilen araç kapasitesi,
- Aracı, her biri kapasitesiyle ve taşıyabileceği malların cinsleriyle tanımlanan bölmelere ayırma imkanı,
- Yükleme ve boşaltma operasyonları için kullanılabilir gereçler,
- Araç tarafından geçilebilecek yolların alt kümesi,

- Aracın kullanımına ait maliyet (birim mesafe başına, birim zaman başına, birim rota/yol başına, vs.).

Araç rotalama problemlerinde rotalar taşınan mallara, servisin kalitesine ve müşterilerin ve araçların karakteristiklerine bağlı olarak birçok operasyonel kısıtları sağlamalıdır. Bu kısıtlardan bazıları şunlardır:

- Her bir rota boyunca aracın mevcut yükü, aracın kapasitesini geçmemelidir,
- Bir rota boyunca servis edilen müşteriler yalnızca malların dağıtımını ya da toplanmasını talep edebilirler ya da her iki talep birden de söz konusu olabilir.
- Müşteriler yalnızca kendilerine ait zaman pencerelerinde servis edilebilirler ve sürücülerin çalışma süreci onları ziyaret eden araçlara bağlıdır.
- Rota boyunca servis edilen müşterilerin ziyaret sırası için öncelik kısıdı şart koşulabilir. Öncelik kısıtlarının bir türü, bir rotada servis edileceği belirlenen müşterilerin, aynı rotada servis edilecek diğer müşteri grubundan önce (ya da sonra) servis edilmesini gerektirir. Bu duruma örnek olarak; toplama ve dağıtımın her ikisinin de uygulandığı ve toplama müşterilerinden toplanan malların aynı araçla ilgili dağıtım müşterilerine taşınmasının gerektiği *toplama ve dağıtım problemleri* olarak adlandırılan problemler verilebilir. Öncelik kısıdının diğer bir türü ise, değişik tipteki müşterilerin aynı rotada servis edilmesinin gerekliliğini şart koşar ve müşterilerin ziyaret sırası karmadır. Bu durum, örnek olarak, malların dağıtım ve toplanmasının her ikisinin de aynı rotada gerçekleştiği ancak, yükleme ve boşaltma operasyonlarıyla ilgili kısıtların ve rota boyunca aracın yükünü yeniden düzenlemenin zorluğunun yani, tüm dağıtımların toplama işlemlerinden önce gerçekleştirilmesi gerekliliğinin olduğu, “*Dağıtım ve toplama taleplerinin farklı müşterilere ait olduğu araç rotalama problemi*” olarak bilinen problemde görülür.

Araç rotalama problemleri için dikkate alınan çok sayıda ve çoğunlukla birbirinden farklı amaçlar olabilmektedir. Bu amaçları şu şekilde sıralayabiliriz:

- Seyahat edilen global mesafeye ya da seyahat zamanına ve kullanılan araçların ve araçların sürücülerinin sabit maliyetine bağlı olan global taşıma maliyetinin minimizasyonu,
- Tüm müşterilere servis yapmak için ihtiyaç duyulan araçların (ya da sürücülerin) sayılarının minimizasyonu,
- Seyahat zamanı ve araç yükü için rotaların dengelenmesi,

Araç rotalama problemleri için bu amaçlardan biri ya da bunların ağırlıklı kombinasyonu dikkate alınabilir[12, s.1-4].

Araç rotalama problemi ilk olarak Dantzig and Ramser (Management Science, 1959) tarafından ortaya konmuştur.[ 12, s.1-4, 14].

### **3.2 Araç Rotalama Problemi için Çözüm Yöntemleri**

1959' da ARP kavramının ilk ortaya çıkışından beri farklı problem şartlarına karşılık gelen ARP' ne dair matematik modeller ortaya çıkmaktadır. Yapılan analizler ışığında şu sonuçlara varılabilir:

i - ARP' nin bir matematik modeli sadece karmaşık değil aynı zamanda geniş kapsamlı bir yapıdır. ARP' nde sadece 10 görev ve 3 araç olsa bile söz edilen bu problem modelinde 393 değişken ve 100' den fazla sınırlı denklik vardır. Buna ek olarak, problemin boyutunun genişlemesiyle birlikte modelin boyutu daha da fazla genişlemektedir. Fakat çok boyutlu ARP lojistikte tek boyutludan daha yaygındır. Bilgisayar için matematik sembollerin anlaşılması ve modelin gerçek zamanlı üretiminin ve çözümünün zor olmasından dolayı bir bilgisayar için bu çeşit yüzlerce değişkenli ve kısıtlı eşitliklere sahip modellerin üretimi ve çözümü karmaşık ve de verimsizdir.

ii – ARP' nin hangi modeli kabul edilirse edilsin bunlar tamamen deęişkenlerin düzenlenmesi, hedef fonksiyon ve beş çeşit kısıttan ibarettir. Bu kısıtlar, araç kapasite kısıtı, zamanlama kısıdı, görev kısıdı, zaman sınırlaması kısıdı ve dięer kısıtlardır.

ARP modeli için algoritmalar, kesin algoritmalar ve sezgiye dayalı algoritmalar olmak üzere iki sınıfa ayrılabilir. Kesin algoritmaların kapasiteleri sınırlıdır. Bu sınırlama nedeniyle araştırmacılar problem çözme prosesine sezgiye dayalı çözüm bilgisini ortaya koymuşlar ve sezgiye dayalı algoritmalarda çok iyi başarılar elde etmişlerdir[15].

Araç rotalama problemi için çözüm yaklaşımları, kesin algoritmalar ve sezgiseller olmak üzere ikiye ayrılır. Uygulamada daha çok sezgiseller kullanılmasına karşın, kesin algoritmalar ise nispeten daha küçük örnekler için kullanılmaktadır[14].

### 3.2.1 Kesin Algoritmalar

Laporte tarafından yapılan araştırmaya göre, kesin algoritmalar sınıfına giren çözüm yöntemleri ve literatür aşağıdaki gibi verilmiştir (n müşteri sayısını ifade etmektedir).

- Konum uzayı gevşemesi ile dinamik programlama
  - Christofides, Mingozi, Toth, 1981 ( $10 \leq n \leq 25$ ).
- Dal-ve-Sınır Algoritması (k-shortest spanning trees, q-paths)
  - Christofides, Mingozi, Toth, 1981 ( $10 \leq n \leq 25$ ).
- Dal-ve-Kesme Algoritması
  - Laporte, 1985, Nobert, Desrochers, Operations Research. ( $n \leq 60$ )
  - Fisher, 1994 (ARP' nin kısıtlanmış bir versiyonu için) ( $n \leq 135$ ).
  - Ralphs vd., 1994 ( $n \leq 101$ ).



- Augerat vd., 1994 ( $n \leq 135$ ).
- Blasum and Hochstättler, 2000 ( $n \leq 76$ ).
- Naddef and Rinaldi, 2002.
- Wenger, 2003 (University of Heidelberg).
- Dal-ve-kesme-ve-değer (Branch-and-cut-and-price) Algoritması
  - Fukasawa et al.,2003, Relatorios de Pesquisa en Engenharia de Produção
- İki-emptia (Two-commodity) ağ akış formülasyonu
  - Baldacci, Hadjiconstantinou, Mingozzi ( $n \leq 135$ ) [14].

Laporte ve Nobert (1987)' ye göre, ARP için kesin çözüm algoritmaları üç genel kategoride sınıflandırılabilir:

- i-* Doğrudan ağaç arama metodları; dal-ve-sınır, dal-ve-kesme, dal-kesme-değer,
- ii-* Dinamik programlama,
- iii-* Tamsayılı doğrusal programlama; küme ayrıştırma (bölümleme) ve kolon üretimi (Soumis and Desrochers, 1984), üç indeksli araç akış formülasyonu, iki indeksli araç akış formülasyonu, iki-ürün ağ akış formülasyonu[16].

### 3.2.1.1 Ağaç Arama Yaklaşımları

Ağaç arama yaklaşımında, orijinal problemden başlayarak, mevcut problem daha basit alt-problemlere bölünür, ve her biri bağımsızca çözülür; örneğin karar ağacı uygun bir dallandırma şeması kullanılarak üretilir. Her bir karar ağacı düğümünde, mümkün çözümlerin tam sıralamasından kaçınmak için, mevcut problemin relaksasyonunun en uygun çözüm değeri dikkate alınarak bir sınır hesaplanır. Eğer sınır, o ana dek bulunan optimum çözümün değerinden daha iyi

değilse, düğümün temeline inilir (dallanma). En iyi çözüm, daha iyi bir uygun çözüm bulunduğunda, güncellenir. Ağaç arama yaklaşımları, her bir düğümündeki sınırı hesaplamak için kullanılan relaksasyon tipine göre değişiklik göstermektedirler[17].

### ***Dal ve Sınır Algoritmaları***

Dal ve sınır algoritmaları, Lagrangian, vekil (surrogate), Lagrangian analizi (decomposition), durum-uzayı, (additive), ...v.s. gibi bir ya da daha fazla “iyi-yapılanmış” relakse problemlerin tespit edilmesini gerektiren (genellikle polinomik olarak çözülebilen ve “kısa” hesaplama zamanı gerektiren durumlarda) orijinal problemin relaksasyonlarını kullanır. Sezgisel yöntemler, genellikle çarpan ekleme tekniği ya da subgradient optimizasyonu vasıtasıyla, çarpanları, vekili (surrogate) ve “iyi” Lagrangianı hızlı bir şekilde tayin etmek için kullanılır[17].

### ***Dal ve Kes Algoritmaları***

Dal ve kes algoritmalarında, sadece kısıtların bir alt kümesi, karar değişkenleri üzerindeki integralite kısıtlarını kaldırarak, orijinal problemde elde edilen problemin doğrusal programlama (DP) relaksasyonu içine başlangıçta yüklenir. DP relaksasyonu, mevcut DP relaksasyonunun optimum çözümüyle ihlal edilen geçerli eşitsizliklerin eklenmesiyle, tekrarlanarak güçlendirilir. İhlal edilmiş eşitsizlikler, kesin veya sezgisel ayırma prosedürleri kullanılarak, sözde ayırma probleminin çözümüyle tanımlanır. Dal ve kes yaklaşımı, etkili DP çözümlerinin uygun olmasını gerektirir; mesela bilgisayar paket programları geniş DP örneklerini kısa sürede çözebilir.

Şu kayda değerdir ki, eklenen eşitsizlikler, orijinal problem için fazlalık olabilirler; fakat mevcut DP relaksasyonunun optimum çözümüyle ihlal edilirler (*kesme düzlemi tekniği*) [17].

### ***Dal-ve-Değer Algoritmaları***

Dal-ve-değer algoritmalarında, orijinal problemin modeli genellikle çok sayıda karar değişkeni (sütunu) ihtiva eder. Bu yaklaşımda, sütunların sadece küçük bir alt kümesi, problemin DP relaksasyonunda başlangıç itibariyle düşünülür. Orijinal DP relaksasyonunun çözümünün optimumluğu, mevcut kaynağa girmeleri için tekrarlanarak tanımlanan yeni sütunlardan elde edilir (*sütun üretme tekniği*). Yeni sütunlar, fiyatlandırma (*pricing*) problemi olarak bilinen probleminin çözümüyle tanımlanır (mesela; Dual DP'nin ayırma problemi). Dal-ve-değer yaklaşımı, etkili DP çözümlerini de gerektirir[17].

#### **3.2.1.2 Dinamik Programlama**

Dinamik programlama (DP) yaklaşımında, (orijinal olana “benzer”) bir alt-problemler ailesi tanımlanır ve verilen bir evrede (ya da mertebede) durumlar, tekrarlanarak, bir önceki fazdaki durumlar üzerine dinamik programlama rekursiyonu uygulanarak hesaplanır. Optimum çözüm, en son evrenin en iyi durumuna karşılık gelir [17].

#### **3.2.2 Sezgisel Algoritmalar**

Problem çözme prosesinde, sezgiye dayalı algoritma şu karakterleri gösterir; bir ilk çözümden başlanır daha iyi bir çözüm buluncaya kadar daha önce tekrarları azaltan ve çözüm prosesini en iyileyen sezgisel modele ait önceki modelden daha uygulanabilir yeni bir çözüm bulmak için tekrar tekrar deneme yapılır ve bu şekilde en iyi çözüm elde edilir. Bu proses dört birimle alakalıdır; çözümleri başlatma birimi, tekrarlanan çözümler üretme birimi, yeni çözümler saptama birimi, muhakeme ve kontrol birimi[15].

### 3.2.2.1 Klasik Sezgisel Algoritmalar

Sezgisel yöntemler, arama uzayının oldukça sınırlı aramasını gerçekleştiren ve kabul edilebilir hesaplama zamanı içerisinde uygun kaliteli çözümler üreten yöntemlerdir[18].

Laporte tarafından yapılan araştırmaya göre, klasik sezgisel algoritma sınıfına giren çözüm yöntemleri ve literatür aşağıdaki gibi verilmiştir.

- Kurtarımlar (Savings) Algoritması
  - Clarke, Wright, 1965.
- Süpürme Algoritması
  - Gillett, Miller, 1974.
- Önce Gruplama Sonra Rotalama(cluster first, route second)
  - Fisher, Jaikumas, 1981.
- Rota içerisinde İyileştirme (İntra-route Improvement) Metotları (GSP Sezgisel Yöntemi).
- Rotalar arasında İyileştirme (İnter-route Improvement) Metotları
  - $\lambda$ -iç değişiklikleri, Osman, 1993;
  - Tekrarlı değişiklikler, Thompson and Psaraffis, 1993;
  - Sınır değişim durumları, Kindervater and Savelsbergh, 1997;
  - Çıkarma Zincirleri (Ejection chains) (Xu and Kelly, 1996;
  - Rego and Roucairol, 1996; Rego, 1998;
  - Çok büyük komşuluk arama (Ergun v.d., 2003).
- SERR Algoritması
  - De Franceschi, Fischetti, Toth, 2004[14].

### 3.2.2.2 İleri Sezgisel Algoritmalar

İleri sezgisel yöntemler, çözüm alanının en elverişli bölgelerinin derin aramasını gerçekleştirmeye dayalı yöntemlerdir. İleri sezgisel yöntemlerle üretilen çözümlerin kalitesi, klasik sezgisellerle elde edilenlere göre daha yüksektir[18].

İleri sezgiselleri çözüm yaklaşımları açısından temel olarak üç ana grupta toplayabiliriz; yerel arama (örn; tavlama benzetim, deterministik tavlama, tabu arama), popülasyon arama (örn; uyarlamalı hafıza metotları, genetik arama), öğrenme mekanizmaları (yapay iletişim ağları, karınca kolonisi sistemleri). Laporte tarafından yapılan araştırmaya göre, ileri sezgisel algoritma sınıfına giren çözüm yöntemleri ve literatür aşağıdaki gibi verilmiştir.

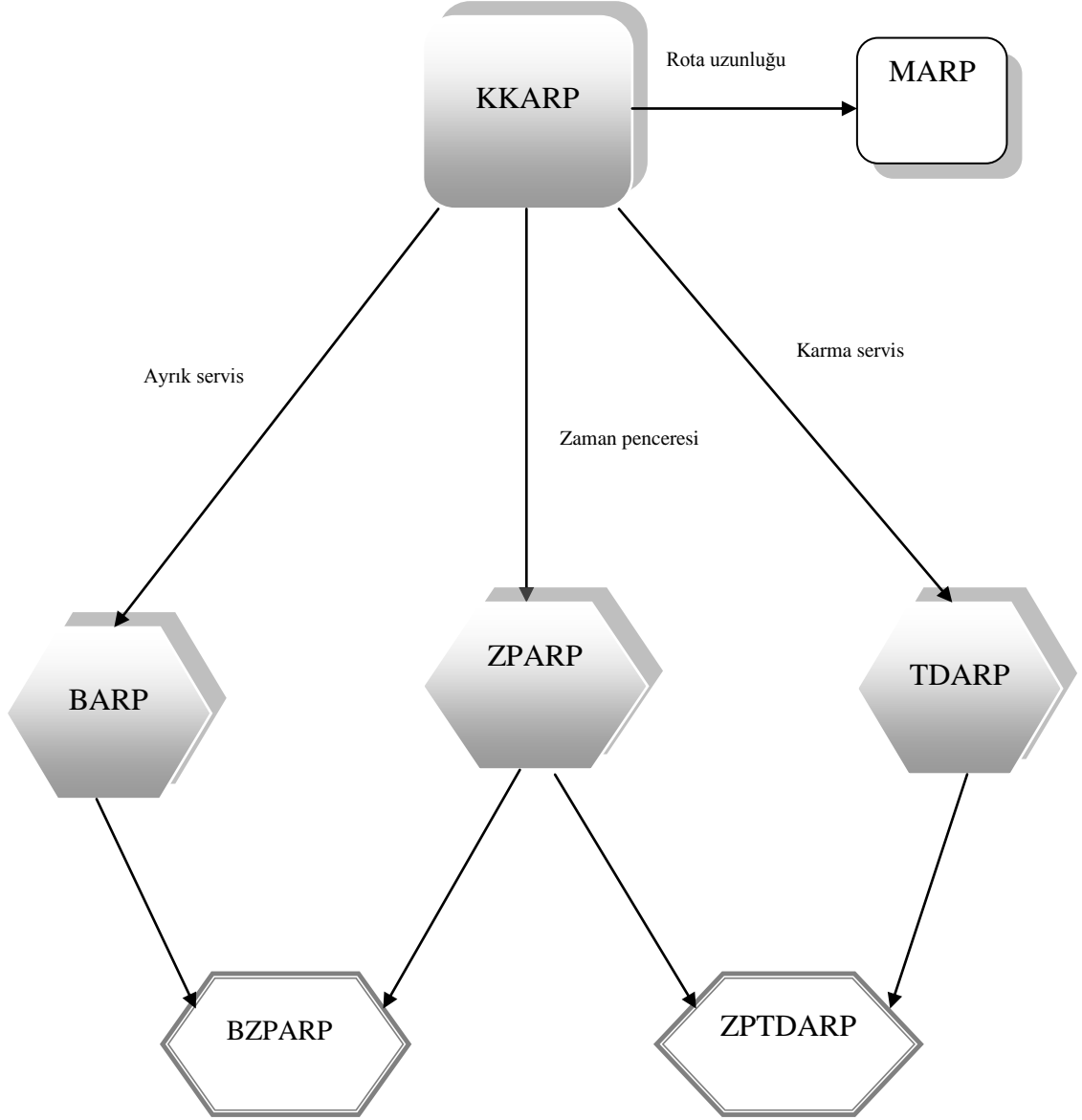
- İlk tabu arama uygulaması (Willard, 1989)
- Taburoute' un ilk versiyonu (Gendreau, Hertz, Laporte, 1991)
- Tabu arama (Taillard, 1993)
- Tavlama benzetimi (Simulated Annealing) ve tabu arama (Osman, 1993)
- Taburoute (Gendreau, Hertz, Laporte, 1994)
- Uyarlamalı bellek (Adaptive memory) (Rochat, Taillard, 1995)
- Çıkarma zincirleri (Ejection chains) (Rego, Roucairolx, 1996)
- Birleştirilmiş tabu arama (Cordeau, Laporte, Mercier, 2001)
- Uyarlamalı bellek (Adaptive memory) (Tarantilis, Kiranoudis, 2002)
- Tanecikli (Granular) tabu arama (Toth, Vigo, 2003)
- Çok büyük komşuluk arama (Ergun, Orlin, Steele-Feldman, 2003)
- Belirleyici tavlama (Deterministic annealing) (Li, Golden, Wasi, 2004)
- Popülasyon arama (Prins; Mester and Bräysy, 2004)
- Karınca sistemi optimizasyonu (Reinmann, Doerner, Hartl, 2004) [14].

### 3.3 Araç Rotalama Problemi Türleri

Araç rotalama sınıfının temel problemlerinin biçimsel tanımlaması graf teorisi modeli olarak ya da yol ağı şeklinde verilebilir.

Araç rotalama problemlerinin literatürde ele alınmış birçok farklı türü olmakla birlikte, temel olarak dört sınıftan bahsetmek mümkündür. Bunlar; bu sınıfın en basit ve üzerinde en çok çalışılan üyesi kapasite kısıtlı araç rotalama problemi (KKARP), KKARP' nin bir versiyonu olarak mesafe kısıtlı araç rotalama problemi (MKARP), zaman pencereci araç rotalama problemi (ZPARP), dağıtım ve toplama taleplerinin farklı müşterilere ait olduğu araç rotalama problemi (BARP), toplama ve dağıtım ile araç rotalama problemi (TDARP) dir. Bu temel problemlerin, ilave kısıtlar tanımlanmasıyla birçok alt türleri oluşturulmuştur. Şekil 3.2' de araç rotalama problemi için yukarıda bahsedilen dört temel sınıfa bağlı olarak oluşturulan sınıflandırma verilmiştir[12, s.1-10].

Tablo 3.1' de ise araç rotalama problemlerinin temel karakteristikleri dikkate alınarak yapılan ayrıntılı sınıflandırma verilmiştir.



Şekil 3.2 ARP sınıfının temel problemleri ve birbirleriyle bağlantıları[12, s.1-10]

Tablo 3.1 Karakteristiklerine göre ARP türleri

ARP TÜRÜ	DEPO SAYISI	ARAC HİLOSU BÜYÜKLÜĞÜ	ARAC FİLOSU TİPİ	ARAC KAPASİTESİ	Talep Türü	TALEP TÜRÜNE GÖRE MÜŞTERİLER	TRANSFER TÜRÜ	PLANLAMA PERİYODU	DİĞER KISITLAR	AMAÇ FONKSİYONU	ÖNEMLİ REFERANSLAR
KKARP	tek	çoklu	özdeş/farklı	kısıtlı	statik-deterministik	Dağıtım Talebine Sahip Müşteriler	Depodan Müşteri Konumlarına dağıtım	Tek Gün	-	tüm müşterilere servis toplam maliyetini minimize etmek	G. Laporte (1992)
Mesafe Kısıtlı ARP	tek	çoklu	özdeş/farklı	-	statik-deterministik	Dağıtım Talebine Sahip Müşteriler	Depodan Müşteri Konumlarına dağıtım	Tek Gün	Her bir Rota için Mesafe ya da Süre Kısıdı	tüm müşterilere servis toplam maliyetini minimize etmek	G. Laporte, M. Desrochers, Y. Nohert (1984)
Dağıtım ve Toplama Taleplerinin Farklı Müşterilere Ait Olan ARP (VRP With Backhauls)	tek	çoklu	özdeş/farklı	kısıtlı	statik-deterministik	1. tip: Dağıtım Talebine sahip müşteriler 2. tip: Toplama Talebine Sahip müşteriler	Depodan Müşteri konumlarına Dağıtım ve Müşteri Konumlarından Toplama	Tek Gün	Müşteriler arasında öncelik kısıdı	tüm müşterilere servis toplam maliyetini minimize etmek	P. Toth ve D. Vigo (1997)
Toplama ve Dağıtım ile Araç Rotalama Problemi (TDARP)	tek	çoklu	özdeş/farklı	kısıtlı	statik-deterministik	Dağıtım ve Toplama Taleplerinin her ikisine birden sahip müşteriler	Müşteri Konumlarından Toplama ve İlgili Müşteri Konumlarına Dağıtım	Tek Gün	herbir i müşterisi için dağıtım toplamadan önce olması kısıdı	tüm müşterilere servis toplam maliyetini minimize etmek	M. Gendreau, G.Laporte, D. Vigo (1999)
Zaman Pencereli ARP	tek	çoklu	özdeş/farklı	kısıtlı	statik-deterministik	Dağıtım Talebine Sahip Müşteriler	Depodan Müşteri Konumlarına dağıtım	Tek Gün	Zaman Penceresi Kısıdı	tüm müşterilere servis toplam maliyetini minimize etmek	P.Badeau, F. Guertin, M. Gendreau, J-Y. Potvin (1997)
Periyodik ARP	tek	çoklu	özdeş/farklı	kısıtlı	statik-deterministik	Dağıtım Talebine Sahip Müşteriler	Depodan Müşteri Konumlarına dağıtım	t gün	-	tüm müşterilere servis toplam maliyetini minimize etmek	J-F.Cordeau, M. Gendreau, G. Laporte (1997)
Çok Depolu ARP	çoklu	çoklu	özdeş/farklı	kısıtlı	statik-deterministik	Dağıtım Talebine Sahip Müşteriler	Depodan Müşteri Konumlarına dağıtım	Tek Gün	-	tüm müşterilere servis toplam maliyetini minimize etmek	E. Hadjiiconstantinou, R. Baldacci (1998) ve J-F.Cordeau, M. Gendreau, G. Laporte (1997)



### 3.3.1 Kapasite Kısıtlı Araç Rotalama Problemi

Kapasite kısıtlı araç rotalama problemi (KKARP), ARP' nin temel versiyonudur. KKARP'nde, tüm müşteriler dağıtımların karşılığıdır, talepler belirleyicidir (deterministic); önceden bilinir ve ayrık olmaları mümkün değildir. Araçlar, tek bir merkez depodan hareket ederler, özdeşirler ve araçlar için yalnızca kapasite kısıtlı söz konusudur. KKARP' nde amaç, tüm müşterilere servisin toplam maliyetini minimize etmektir.

KKARP, yol ağı üzerinde tarif edilebilir. Şöyle ki;  $V = \{0, \dots, n\}$  köşe (düğüm) kümesi ve  $A$ , ark kümesi olduğu durumda  $G = (V, A)$  bir yol ağını ifade etsin. Bu yol ağında,  $i = 1, \dots, n$  düğümleri müşterilere, 0 düğümü ise depoya karşılık gelmektedir. Bazen depo  $n + 1$  düğümü ile gösterilebilmektedir.

Negatif olmayan  $c_{ij}$  maliyeti her bir  $(i, j) \in A$  arkıyla ilişkilendirilmiştir ve  $i$  düğümünden  $j$  düğümüne gitmek için harcanan seyahat maliyetini ifade etmektedir.

Eğer  $G$  yol ağında gidiş ve dönüş için kat edilen mesafenin maliyeti farklı değerler alıyorsa maliyet ya da mesafe matrisi asimetric olur ve problem *Asimetrik KKARP* olarak tanımlanır. Tam tersi,  $c_{ij} = c_{ji}$  ( $\forall (i, j) \in A$ ) olduğunda ise problem *Simetrik KKARP* olarak ifade edilir ve  $A$  yay kümesi, yol ağında gidiş ve dönüş seyahatleri için aynı değerlere sahip kenarları ifade eder.

Birçok uygulamalı problemde, maliyet matrisi aşağıda verilen üçgen eşitsizliğine uymaktadır;

$$c_{ik} + c_{kj} \geq c_{ij}, \quad \forall i, j, k \in V.$$

Her bir  $i$  ( $i = 1, \dots, n$ ) müşterisi negatif olmayan belirli  $d_i$  talebine sahiptir ve deponun talebi ise  $d_0 = 0$  ile belirtilir. Verilen  $S \subseteq V$  düğüm kümesi için  $d(S) = \sum_{i \in S} d_i$  kümenin toplam talebini ifade eder.

Her biri  $C$  kapasiteli  $K$  özdeş araç depoda mevcuttur. Her bir araç en fazla bir rotada çalışabilmelidir. Tüm müşterilere servis için gerekli olan araç sayısı  $K_{min}$  olduğunda,  $K$ ,  $K_{min}$  'den küçük olmamalıdır. Buna göre, KKARP' nin optimum çözümü için hesaplanan araç sayısının rota sayısı ile eşit olduğu açıktır.

KKARP, çevrimlere ait olan arkların maliyetlerinin toplamı olarak tanımlanan, her biri bir aracın rotasıyla eşdeğer minimum maliyetli  $K$  çevrimlerinin tamamının toplamının bulunmasından ibarettir. Şöyle ki;

- i- her bir çevrim (rota) depo düğümünü ziyaret eder,
- ii- her bir müşteri düğümü sadece bir çevrim tarafından ziyaret edilir ve
- iii- çevrim tarafından ziyaret edilen düğümlerin taleplerinin toplamı araç kapasitesi  $C$  'yi geçmemelidir.

Toth ve Vigo, KKARP' nin ilk varyantı olarak, her bir rota için kapasite kısıdının maksimum uzunluk ya da maksimum süreyle yer değiştirdiği *mesafe kısıtlı araç rotalama problemini (MKARP)* göstermiştir. MKARP' de, negatif olmayan uzunluk  $t_{ij}$  her bir  $(i, j) \in A$  arkıyla ilişkilendirilmiştir ve her bir rotanın arklarının toplam uzunluğu maksimum rota uzunluğu  $T$  'yi geçmemelidir. Eğer araçlar farklıysa, bu durumda maksimum rota uzunlukları  $T_k$  ( $k = 1, \dots, K$ ) olur[12, s.1-10].

Bu problemin çözümü için başlangıçta basit bir tamsayı programlama modelinden yararlanılmaktadır. *Klasik ARP' nin çözümü için model şu şekilde tanımlanır*[19].

***Parametreler:***

$Q_k$ : araç kapasitesi,

$N$  : müşteri veya durak sayısı,

$q_i$ :  $i$  ( $i > 0$ )müşterisinin talebi,

$d_{ij}$  :  $i$  ve  $j$  müşterileri arasındaki uzaklık,

**Değişkenler:**

$x_{ij}$  ,  $i \neq j$  : araç  $i$  ' den  $j$  ' ye gidiyorsa "1", yoksa "0" olmak üzere,  
(  $i, j \in \{0, \dots, n\}$  ve "0" başlangıç deposu iken)

**Amaç Fonksiyonu:**

$$\text{Min} \sum_{i=0}^n \sum_{\substack{j=0, \\ i \neq j}}^n d_{ij} x_{ij} \quad (3.1)$$

**Kısıtlar:**

$$\sum_{\substack{i=1 \\ i \neq j}}^n x_{ij} = 1 \quad \forall i, j \in \{1, \dots, n\} \quad (3.2)$$

$$\sum_{\substack{j=1 \\ i \neq j}}^n x_{ij} = 1 \quad \forall i, j \in \{1, \dots, n\} \quad (3.3)$$

$$\sum_{i=0}^n \sum_{\substack{j=0 \\ j \in S}}^n x_{ij} \leq |S| - 1 \quad (3.4)$$

$$\sum_{i=0}^n \sum_{\substack{j=0 \\ j \in T}}^n x_{ij} \leq |T| - n \quad (3.5)$$

Burada (3.2) ve (3.3) kısıtları ziyaret edilen müşteriden ayrılma kısıtlarıdır. (3.4) kısıdı depodan başlamayan ve depoda bitmeyen turları elemekte kullanılır. Son olarak, (3.5) kısıdı araçlardaki yük durumunu kontrol etmektedir. Bu kısıt, depo dahil olmak üzere her  $T$  müşteri kümesine eklenir. Bu kümelerin her biri,  $\sum_{i \in T} q_i <$

$Q_k$  şartını sağlamaktadır. Ayrıca  $n$  ise aşırı yüklemeyi engellemek için  $T$  kümesinden çıkarılması gereken minimum sayıda müşteri sayısıdır[19].

### 3.3.2 Dağıtım ve Toplama Taleplerinin Farklı Müşterilere Ait Olduğu Araç Rotalama Problemi

Dağıtım ve toplama taleplerinin farklı müşterilere ait olduğu araç rotalama problemi (BARP), KKARP' nin,  $V \setminus \{0\}$  müşteri kümesinin iki altkümeye ayrıldığı, bir uzantısıdır. Birinci altküme  $L$ , her biri belirli miktarda ürünün kendilerine dağıtılmasını talep eden  $n$  adet (*linehaul*) müşteriden meydana gelir. İkinci altküme  $B$  ise, belirli miktardaki depoya gidecek ürünün kendilerinden toplanması gereken  $m$  adet (*Backhaul*) müşteriyi kapsar. Buna göre müşteriler,  $L = \{1, \dots, n\}$  ve  $B = \{1, \dots, m\}$  şeklinde numaralanır. Müşterilerin bu şekilde gruplandığı gerçek hayat uygulamaları oldukça fazladır. Bunlardan en yaygın olanı, süper marketlerin dağıtım (*Linehaul*) müşterileri ve tedarikçilerin de toplama (*Backhaul*) müşterileri olduğu gıda endüstrisidir.

Toplama ( $B$ ) ve dağıtım( $L$ ) müşterileri arasında öncelik kısıdı söz konusudur; rota her iki tip müşteriye de servis ediyorsa, tüm  $L$  müşterileri  $B$  müşterilerinden önce servis edilmelidir. Tipine bağlı olarak dağıtılan ya da toplanan ve negatif olmayan belirli  $d_i$  talebi her bir  $i$  müşterisi ile ilişkilendirilmiştir ve deponun talebi  $d_0 = 0$  ile belirtilir. Maliyet matrisi asimetric olduğunda problem, *Asimetrik BARP* olarak isimlendirilir. Dağıtım ve toplama taleplerinin farklı müşterilere ait olduğu araç rotalama problemi, minimum maliyetli  $K$  çevrimlerinin tamamının toplamının bulunmasını içerir. Şöyle ki;

- i-* her bir çevrim depo düğümünü ziyaret eder,
- ii-* her bir müşteri düğümü sadece bir çevrim tarafından ziyaret edilir,
- iii-* bir çevrim tarafından ziyaret edilen dağıtım (*Linehaul*) ve toplama (*Backhaul*) müşterilerinin talepleri tek başına araç kapasitesi  $C$ ' yi geçmemelidir ve

iv- her bir çevrimde dağıtım (*Linehaul*) müşterileri, eğer varsa, toplama (*Backhaul*) müşterilerinden önce gelir.

Bu problem türünde, genellikle yalnızca toplama (*Backhaul*) müşterilerine izin verilmez. Ayrıca, öncelik kısıdı (iv), dolaylı olarak “*karma*” araç rotalarına yönelmeyi gösterir; başka bir ifadeyle, rotaların her iki tip müşteriyi de ziyaret ettiğini gösterir.

$K_L$  ve  $K_B$  sırasıyla tüm dağıtım (*Linehaul*) ve toplama (*Backhaul*) müşterilerine servis için gerekli olan minimum araç sayısı olduğu durumda bu değerler, ilgili müşteri altkümeleriyle benzerliği olan kutu paketleme problemi (KPP/Bin Packing Problem) örneklerinin çözümüyle elde edilebilir. Uygunluktan emin olmak için  $K$ 'nin tüm müşterilere servis için gerekli olan minimum araç sayısından küçük olmadığını ( $K \geq \max\{K_L, K_B\}$ ) farz edilir.

Dağıtım ve toplama taleplerinin farklı müşterilere ait olduğu araç rotalama problemi ve *Asimetrik* dağıtım ve toplama taleplerinin farklı müşterilere ait olduğu araç rotalama problemi (ABARP),  $B = \emptyset$  olduğunda, sırasıyla SKKARP ve AKKARP' nin temel örneklerinin daha geniş kapsamlı örnekleri olduğu için *NP-zor* sınıfındadırlar. Dağıtım ve toplama taleplerinin farklı müşterilere ait olduğu araç rotalama problemi' nin literatürde zaman pencereleriyle ifade edilen örnekleri zaman pencereli dağıtım ve toplama taleplerinin farklı müşterilere ait olduğu araç rotalama problemi (BZPARP) olarak adlandırılmaktadır[12, s.1-10].

### 3.3.3 Toplama ve Dağıtım ile Araç Rotalama Problemi (TDARP)

TDARP' nde,  $i$  müşterisindeki dağıtım talebinin miktarını gösteren  $d_i$  ve  $i$  müşterisindeki toplama talebinin miktarını gösteren  $p_i$  ile her bir  $i$  müşterisi ilişkilendirilmiştir.  $d_i$  ve  $p_i$  aynı türden mallardır. Bazen, her bir  $i$  müşterisi için, dağıtım ve toplama talepleri arasındaki net farkı gösteren bir  $d_i = d_i - p_i$  talep miktarı kullanılır. Her bir  $i$  müşterisi için  $O_i$  dağıtım talebinin başlangıcı olan düğümü ve  $D_i$  toplama talebinin hedefi (varış noktası) olan düğümü ifade eder.

Her bir müşteri konumunda dağıtımın toplamadan önce gerçekleştiği varsayılır. Bun nedenle, aracın belirli bir konuma varmadan önceki mevcut yükü, başlangıç yükü eksi o ana kadar dağıtılan tüm taleplerin toplamı artı o ana kadar toplanan tüm taleplerin toplamı ile tanımlanır.

TDARP, minimum maliyetli  $K$  çevrimlerinin tamamının toplamının bulunmasını içerir. Şöyle ki;

- i-* her bir çevrim depo düğümünü ziyaret eder,
- ii-* her bir müşteri düğümü sadece bir çevrim tarafından ziyaret edilir,
- iii-* çevrim boyunca aracın mevcut yükü pozitif olmalıdır ve araç kapasitesi  $C'$  yi geçmemelidir,
- iv-* her bir  $i$  müşterisi için,  $O_i$  müşterisi, depodan farklı olduğunda, aynı çevrimde ve  $i$  müşterisinden önce ziyaret edilmelidir ve
- v-* her bir  $i$  müşterisi için,  $D_i$  müşterisi, depodan farklı olduğunda, aynı çevrimde ve  $i$  müşterisinden sonra ziyaret edilmelidir.

Sıklıkla taleplerin başlangıç ve hedefleri ortaktır (örneğin KKARP ve dağıtım ve toplama taleplerinin farklı müşterilere ait olduğu araç rotalama problemindeki gibi depoya bağlıdır) ve bu nedenle ayrıca belirtmeye gerek yoktur. Bu şekildeki problemler eş zamanlı toplama ve dağıtım ile araç rotalama problemi (ETDARP) olarak bilinmektedir.

TDARP ve ETDARP *NP-zor* sınıfındadır çünkü her iki problemde de  $\forall i \in V$  için  $O_i = D_i = 0$  ve  $p_i = 0$  olduğunda problem, kapasite kısıtlı araç rotalama problemini tanımlamaktadır.

TDARP' nin literatürde üzerinde çalışılmış olan diğer bir versiyonu da zaman pencerelerinin mevcut olduğu, zaman pencereleri ve toplama ve dağıtım ile araç rotalama problemi (ZPTDARP) olarak adlandırılmaktadır[12, s.1-10].

### 3.3.4 Periyodik Araç Rotalama Problemi

Periyodik araç rotalama problemi (PARP)  $t$  günlük planlama periyodunda çalışır ve her bir  $i$  müşterisi servis frekansı  $e_i$ ' yi belirler ve  $C_i$  kümesi ziyaret günlerinin izin verilen kombinasyonudur. Örneğin;  $e_i = 2$  ve  $C_i = \{\{1, 3\}, \{2, 4\}, \{3, 5\}\}$  olduğunda, bu,  $i$  müşterisinin iki kez ziyaret edilmesini ve bu ziyaretlerin 1. ve 3., 2. ve 4. ya da 3. ve 5. günlerde gerçekleşebileceği anlamına gelir. Problem bundan sonra, ARP kuralları doğrultusunda, her müşteri için ziyaret kombinasyonunun eş zamanlı seçiminden ve planlama evresinin her bir günü için araç rotalarını oluşturmaktan ibarettir. PARP, market malları dağıtımı ve atık toplama gibi alanlarda ortaya çıkmaktadır[20].

### 3.3.5 Çok Depolu Araç Rotalama Problemi

Çok depolu araç rotalama problemi (ÇDARP) tek günlük planlama için tanımlanmıştır. Ancak araçlar tek bir depo yerine birçok deponun birinden hareket ederler ve her bir aracın rotası hareket ettiği bu depoda başlar ve sonlanır. Bunların haricinde, problemin tanımı ARP' nin tanımıyla aynıdır.

Başka bir ifadeyle, ÇDARP PARP' nin özel bir durumu olarak da ifade edilebilir. PARP' ndeki günleri depolar olarak düşünülürse  $t$ , gün değil de depoyu ifade edecektir. Böylelikle  $C_i = \{\{1\}, \{2\}, \dots, \{k\}\}$  olacaktır ve problem ARP kuralları çerçevesinde her bir depo için ayrı ayrı araç rotalarını oluşturma şeklinde ifade edilecektir[20].

ÇDARP' nin hedefi, toplam seyahat mesafesini minimize ederken kısıtların altında yani her bir rota için servis edilen müşterilerin talepleri araçların kapasitesini aşmayacak şekilde tüm müşterilere servis sunmaktır[21].

### 3.3.6 Karışık Filolu Araç Rotalama Problemi

Karışık Filolu Araç Rotalama Problemi (KFARP), değişik kapasiteli heterojen araç filolarının bulunduğu taşıma sistemleri için geliştirilmiş bir ARP tipidir [19].

### 3.3.7 Yer Bağımlı Araç Rotalama Problemi

Yer Bağımlı Araç Rotalama Probleminde (YBARP), heterojen araç filosu müşterilere servis yapar. Tipik olarak birkaç araç tipi bulunmaktadır. Bu problemde araç tipi ile müşteriler arasında uyumluluk vardır. YBARP' de bir ana depo,  $N$  tane müşteri ve her müşteriye uyumlu araç tiplerinin kümesi vardır. Bu problemde amaç tüm bu kısıtlar altında araçların seyahat ettiği toplam mesafeyi minimize etmektir[19].

### 3.3.8 Zaman Pencereci Araç Rotalama Problemi

Zaman pencereci araç rotalama problemi (ZPARP) birçok dağıtım sisteminde görülen önemli bir problemidir. Banka dağıtımı, posta dağıtımı, kargo taşımacılık hizmetleri, endüstriyel atık toplama, güvenlik devriye hizmetleri ve tam zamanında üretim (*just in time manufacturing*) ZPARP sınıfına giren gerçek hayat problemlerinden bazılarıdır[22].

ZPARP, bir depodan coğrafi olarak dağıtılmış noktalara en düşük maliyetli rotaların planlanması problemi olarak tanımlanabilir. Rotalar şu şekilde planlanmalıdır:

- Her bir nokta verilen zaman aralığında bir araç tarafından ve yalnızca bir defa ziyaret edilir,
- Tüm rotalar depoda başlar ve depoda biter,



- Bir rota üzerindeki her noktanın toplam talebi o rotaya atanmış aracın kapasitesini geçmemelidir.

ZPARP, amacın yalnızca ihtiyaç duyulan araç sayının değil, aynı zamanda tüm araçların toplam seyahat zamanı ve toplam seyahat mesafesinin de minimize edilmiş olması sebebiyle çok amaçlıdır[23].

### 3.3.8.1 Zaman Pencereci Araç Rotalama Problemi Formülasyonu

ZPARP  $(N, A)$  yol ağı üzerinde tanımlanır. Düğüm kümesi  $N$ ,  $C$  ile gösterilen müşteri kümesi ile  $0$  ve  $n + 1$  düğümlerinden oluşur. Müşteri sayısı  $|C|$ ,  $n$  ile ve müşteriler de  $1, 2, \dots, n$  ile gösterilecektir. Düğümler arasındaki mümkün bağlantılar bir  $A$  ark (yay) setine karşılık gelmektedir. Hiçbir yay  $0$  düğümünde sonlanamaz ve yine hiçbir yay  $n + 1$  düğümünden çıkamaz. Tüm rotalar  $0$  düğümünde başlar ve  $n + 1$  düğümünde sonlanır.  $c_{ij}$  maliyeti ve  $t_{ij}$  seyahat süresi ağıdaki her bir  $(i, j) \in A$  yayıyla ilişkilendirilmiştir. Seyahat süresi  $t_{ij}$ ,  $i$  müşterisindeki servis zamanını da içerir.  $V$ , özdeş araçların kümesini ifade eder. Her bir araç  $q$  kapasitesine ve her bir müşteri  $d_i$  ( $i \in C$ ) talebine sahiptir. Her bir müşteride servisin başlangıcı zaman penceresi olarak ifade edilen verilen zaman aralığında  $[a_i, b_i]$ , ( $i \in C$ ) olmalıdır. Aynı zamanda araçlar depodan  $[a_0, b_0]$  zaman penceresinde ayrılmalı ve  $[a_{n+1}, b_{n+1}]$  zaman penceresi zarfında depoya dönmelidirler. Aracın zaman penceresi başlangıcından önce varmasına ve servisin en erken başlama süresine kadar beklemesine ek bir maliyet gerektirmeden müsaade edilirken, en geç servis zamanından sonra varmasına müsaade edilmemektedir. Bekleme zamanının bir ilave maliyeti olmamasından dolayı,  $a_0 = b_0 = 0$  olduğu kabul edilir; bu tüm rotaların  $0$ . dakikada başladığı anlamına gelir.

Model iki çeşit karar değişkenini içermektedir. Eğer  $k$  aracı,  $i$  düğümünden  $j$  düğümüne gidiyorsa,  $X_{ij}^k$  ( $\forall (i, j) \in A, \forall k \in V$ ) karar değişkeni  $1$ ' e, aksi takdirde  $0$ ' a eşittir.  $S_i^k$  ( $\forall i \in N, \forall k \in V$ ) karar değişkeni ise  $k$  ( $k \in V$ ) aracının  $i$  ( $i \in C$ ) müşterisinde servise başladığı zamanı ifade eder. Eğer  $k$  aracı  $i$

müşterisine servis yapmıyorsa,  $S_i^k$  tanımsızdır.  $S_0^k = 0$  olduğu ve  $S_{n+1}^k$ 'nin da  $k$  aracının depoya dönüş zamanı olduğu varsayılır. Amaç, her biri bir aracı ifade eden ve tüm müşterilerin yalnızca bir kere servis edildiği minimum maliyetli rotayı planlamaktır. Bu nedenle parçalı dağıtıma (split deliveries) izin verilmemektedir. Rotalar araçların kapasitesine ve servis edilen müşterilerin zaman pencerelerine uygun olmalıdır. Tüm bu kabullere göre zaman pencereci araç rotalama problemi aşağıdaki şekilde formüle edilebilir[23]:

$$\text{Min} \sum_{k \in V} \sum_{(i,j) \in A} c_{ij} X_{ij}^k \quad (3.6)$$

Kısıtlar:

$$\sum_{k \in V} \sum_{j \in N} X_{ij}^k = 1, \quad \forall i \in C \quad (3.7)$$

$$\sum_{i \in C} d_i \sum_{j \in N} X_{ij}^k \leq q, \quad \forall k \in V \quad (3.8)$$

$$\sum_{j \in N} X_{0j}^k = 1, \quad \forall k \in V \quad (3.9)$$

$$\sum_{i \in N} X_{ih}^k - \sum_{j \in N} X_{hj}^k = 0, \quad \forall h \in C, \forall k \in V \quad (3.10)$$

$$\sum_{i \in N} X_{i,n+1}^k = 1, \quad \forall k \in V \quad (3.11)$$

$$X_{ij}^k (S_i^k + t_{ij} - S_j^k) \leq 0, \quad \forall (i,j) \in A, \forall k \in V \quad (3.12)$$

$$a_i \leq S_i^k \leq b_i, \quad \forall i \in N, \forall k \in V \quad (3.13)$$

$$X_{ij}^k \in \{0, 1\}, \quad \forall (i,j) \in A, \forall k \in V \quad (3.14)$$

Amaç fonksiyonu (3.6) maliyetin minimize edilmesini ifade/tayin eder. Kısıt kümesi (3.7) her bir müşterinin yalnızca bir araca atanmış olmasını ve (3.8) de hiçbir aracın kapasitesinin müsaade ettiğiinden fazla müşteriye servis yapamamasını tayin eder. (3.9), (3.10) ve (3.11) kısıt kümeleri, her bir  $k$  aracının 0 (depo) düğümünden

yalnızca bir kere ayrılıp, ancak ve ancak  $h$  ( $h \in C$ ) düğümüne giriş yaptıysa bu düğümünden ayrılmasını ve sonunda  $n + 1$  (depo) düğümüne dönmesini gerektiren akış kısıtlarıdır. Burada kısıt (3.11)'in gereğinden fazla olduğuna dikkat edilmelidir, ancak bu kısıt ağ yapısını vurgulamak için modelde korunmuştur.  $(0, n + 1)$  yayı boş turlara müsaade etmek için ağ içerisine dahil edilmiştir. Kesin olarak, araçların sayısında sınırlama getirilmemiştir ancak kullanılan her bir araca  $c_v$  maliyeti eklenmektedir. Bu  $c_{0,n+1} = -c_v$  saptaması ile yapılmaktadır.  $c_v$  değeri birinci olarak araçların sayısını, ikinci olarak seyahat maliyetini minimize etmek için yeterince büyüktür. Doğrusal olmayan kısıt kümesi (3.12) eğer  $k$  aracı  $i$  düğümünden  $j$  düğümüne seyahat ediyorsa,  $k$  aracının  $j$  düğümüne  $S_i^k + t_{ij}$  süreden önce varamayacağını tayin eder. Kısıt kümesi (3.13) tüm zaman pencerelerine uyulmasını garanti altına alır ve (3.14) bütünlük (integrality) kısıtları kümesidir[23].

Zaman pencereli araç rotalama probleminin kargo taşımacılığı uygulamalarını da kapsayan özel bir durumu olan toplama ve dağıtım zaman pencereli araç rotalama problemi izleyen bölümde ayrıntılı olarak açıklanacaktır. Bu çalışmada, günlük hayatta oldukça geniş bir uygulama sahasına sahip olması, hızla önem kazanan teknolojik gelişmelerin ve buna bağlı olarak artan rekabet sonucunda giderek önem kazanan müşteri memnuniyeti, zaman ve maliyet unsurlarının taşımacılık sektöründe de ivme kazanması nedeniyle kargo taşımacılığı uygulamaları model alınmıştır. Söz konusu problemin çözümü için geliştirilen model ve örnek problem çözümleri yine izleyen bölümde sunulacaktır.

#### **4. KARGO SERVİSİ İÇİN TOPLAMA VE DAĞITIM ZAMAN PENCERELİ ARAÇ ROTALAMA PROBLEMİ VE KESİN ÇÖZÜM YAKLAŞIMI**

Bu bölümde kargo taşımacılığı uygulamalarında, çok sayıda ve farklı koordinatlardaki müşteri konumlarından gönderilerin toplanması ve toplanan gönderilerin alıcı adreslerine tam zamanında ulaştırılmak üzere belirlenen zamanda şubeye getirilip transfer merkezlerine iletilmeleri için araçların rotalarının planlanması problemi ele alınmaktadır. TDARP ve ZPARP' nin zaman penceresi kısıtlı ve kapasite kısıdı olmayan özel bir durumu olan mevcut problem, Lin (2008) tarafından toplama ve dağıtım zaman pencereci araç rotalama problemi olarak ifade edilmiştir. Problemin çözümünde, kullanılan herhangi bir sezgisel yöntemden kaynaklanacak tahmin hatalarından kaçınmak için kesin çözüm metodu uygulanmıştır. Ayrıca, kesin çözüm metotlarıyla elde edilen sonuçlar, daha sonra daha büyük boyutlu problemleri çözmek için uygulanacak sezgisel yöntemlerin geliştirilmesi ve sonuçlarının değerlendirilmesi için de temel oluşturabilmektedir.

##### **4.1 Problem Tanımı**

Bu çalışmanın konusu olan kargo taşımacılığına uygun olarak, Lin (2008) tarafından ele alınan, posta ya da koli dağıtım uygulamalarını kapsayan araç rotalama probleminin özel bir durumu temel alınmıştır. Birçok toplama noktalarının ve kargo şubesi ya da depo olarak ifade edilen yalnızca tek bir genel dağıtım noktasının bulunduğu, tüm müşterilerin kendilerine ait zaman penceresinde aynı araçla servis edilmeleri için toplama ve dağıtım talebi ikilileriyle tanımlandığı mevcut problem, statik-deterministik toplama ve dağıtım zaman pencereci araç rotalama problemi olarak tanımlanabilir.

Bu problemde müşteriler, müşteri konumlarında dokümanların toplanması ve şubeden ilgili noktalara dağıtımını gerçekleştirmek amacıyla belirlenmiş hareket zamanını karşılamak üzere şubeye varış için olmak üzere iki adet zaman penceresi talebine sahiptirler. Posta merkezinde, toplama dağıtımdan önce meydana gelmektedir. Araçların taşıdığı mallar genellikle mektup, küçük paket gibi ağırlık ve hacim açısından araç kapasitesini etkilemeyen büyüklükte olduğundan, örnek problemlerde kapasite kısıdı dikkate alınmamıştır. Zaman penceresi kısıtlı ancak kapasite kısıdı olmayan, tek bir noktadan (depodan) dağıtım ve çok noktadan (müşteri konumlarından) toplama problemi olarak ifade edilen örnek problem, zaman pencereleri ve toplama ve dağıtım ile araç rotalama problemi (ZPTDARP)' nin özel bir durumudur[24].

Araç filosu, tümü depodan hareket eden ve tüm dağıtımların depoda görüldüğü  $m$  özdeş araçtan oluşmuştur. Bundan dolayı problem, toplama ve dağıtım zaman pencereli  $m$  -ARP olarak da kabul edilebilir.

Ele alınan problemin amacı, zaman penceresi kısıtlarını karşılayarak tüm müşterilere servisin minimum maliyetli çözümünü bulmaktır. Bu, gerekli olan araç sayısına ve müşterilere servis eden araç rotalarının kümesine karar vermeyi içerir.

Önceki bölümlerde ortaya konulduğu üzere, araç rotalama problemleri  $NP$ -zor sınıftadır ve dolayısıyla ele alınan problem de  $NP$ -zor sınıfına aittir.

Toplama ve dağıtım ile araç rotalama problemleri için çalışma kısıtlarına bağlı olarak çeşitli sınıflandırmalar yapılabilir. Toplama ve dağıtım zaman pencereli ARP için problem yapısı ve çözüm yaklaşımı bakımından sınıflandırma Tablo.1' de verilmiştir. Bu sınıflandırmaya göre, ele alınan problem, problemin fiili dağıtım zamanından önce çözüldüğü *statik-deterministik talepli* problem sınıfına aittir. *Çok sayıda özdeş ve kapasite kısıtsız aracın bulunduğu yalnızca bir merkez depo* bulunur. Tüm müşteriler için, *toplama dağıtımdan önce meydana gelir* ve amaç, aracın sabit maliyeti ve seyahat maliyetinden oluşan *toplam maliyeti minimize etmek* amacıyla zaman penceresi kısıdı altında tüm müşterilere servis eden *bağımsız* araç rotalarının kümesini bulmaktır[24].

Tablo 4.1 Toplama ve Dağıtım Zaman Pencereci Araç Rotalama Probleminin Sınıflandırması[24]

<b>A. Problem Yapısına Göre</b>	
<b>I. Transfer gereksiniminin varlığı</b> <ul style="list-style-type: none"> <li>• Statik-deterministik</li> <li>• Dinamik</li> <li>• Stokastik</li> </ul>	<b>II. Depo Sayısı</b> <ul style="list-style-type: none"> <li>• Tek</li> <li>• Birden fazla</li> </ul>
<b>III. Araç Filosu Büyüklüğü</b> <ul style="list-style-type: none"> <li>• Tek</li> <li>• Çoklu (özdeş/farklı)</li> </ul>	<b>IV. Araç Kapasitesi</b> <ul style="list-style-type: none"> <li>• Kapasite kısıtlı</li> <li>• Kapasite kısıtsız</li> </ul>
<b>V. Toplama ve dağıtımlar arasındaki ilişki</b> <ul style="list-style-type: none"> <li>• Önce toplama sonra dağıtım</li> <li>• Önce dağıtım sonra toplama</li> <li>• Eş zamanlı Toplama ve dağıtım</li> <li>• Karma toplama ve dağıtım</li> </ul>	<b>VI. Amaç Fonksiyonu</b> <ul style="list-style-type: none"> <li>• Seyahat zamanını minimize etmek</li> <li>• Rota uzunluğunu minimize etmek</li> <li>• Müşteri memnuniyetsizliğini minimize etmek</li> <li>• Araç sayısını minimize etmek</li> <li>• Maliyeti minimize (ya da karı maksimize) etmek</li> </ul>
<b>VII. Bekleme izni (nakliye sırasında)</b> <ul style="list-style-type: none"> <li>• evet</li> <li>• hayır</li> </ul>	<b>VIII. Araç operasyonları</b> <ul style="list-style-type: none"> <li>• Bağımsız</li> <li>• kooperatif</li> </ul>
<b>B. Çözüm Yaklaşımına Göre</b>	
<b>I. Kesin Algoritmalar</b> <ul style="list-style-type: none"> <li>• Tamsayı Programlama</li> <li>• Dinamik Programlama</li> <li>• Dal-Ve-Sınır (Etkili Alt Sınır ile)</li> <li>• Lagrange Relaksasyonu</li> <li>• Küme Bölümleme (ya da Küme Kaplama) Modelinin Relaksasyonu Ve Kolon Üretimi</li> </ul>	<b>II. Segisel algoritmalar</b> <ul style="list-style-type: none"> <li>• Yapım ve ekleme</li> <li>• Ayrıştırma</li> <li>• Lagrange relaksasyonu</li> <li>• melez</li> </ul>
<b>III. İleri sezgiseller (Metaheuristics)</b>	
<ul style="list-style-type: none"> <li>• Tabu arama</li> </ul>	<ul style="list-style-type: none"> <li>• Genetik algoritma</li> </ul>

## 4.2 Toplama ve Dağıtım Zaman Pencereci Araç Rotalama Problemi için Model

Zaman pencereci araç rotalama probleminin, kapasite kısıdının dikkate alınmadığı ve araçların depoya dönüşünü zorlayan kısıt olarak her bir müşteri için dağıtım zaman pencerelerinin tanımlandığı, bir varyantı olan toplama ve dağıtım zaman pencereci araç rotalama problemi için model kabulleri, ZPARP için bölüm 3.3.8.1' de sunulan kabullerle aynı olmak kaydıyla, söz konusu farklılıkların ZPARP formülasyonuna uyarlanmasıyla elde edilen model aşağıda sunulmuştur.

$$\text{Min} \sum_{k \in V} \sum_{(i,j) \in A} c_{ij} X_{ij}^k \quad (4.1)$$

Kısıtlar:

$$\sum_{k \in V} \sum_{j \in N} X_{ij}^k = 1, \quad \forall i \in C \quad (4.2)$$

$$\sum_{j \in N} X_{0j}^k = 1, \quad \forall k \in V \quad (4.3)$$

$$\sum_{i \in N} X_{ih}^k - \sum_{j \in N} X_{hj}^k = 0, \quad \forall h \in C, \forall k \in V \quad (4.4)$$

$$\sum_{i \in N} X_{i,n+1}^k = 1, \quad \forall k \in V \quad (4.5)$$

$$X_{ij}^k (S_i^k + t_{ij} - S_j^k) \leq 0, \quad \forall (i,j) \in A, \forall k \in V \quad (4.6)$$

$$a_i \leq S_i^k \leq b_i, \quad \forall i \in N, \forall k \in V \quad (4.7)$$

$$\sum_{i=0}^N \sum_{j=1}^N (\text{Max}\{t_j - \delta_p, S_i^k + t_{ij}\} + t_{j0}) X_{ij}^k \leq \text{Min}\{\tau_{i_1}, \tau_{i_2}, \dots, \tau_{i_k}, \tau_j\} + \delta_D \quad (4.8)$$

$$X_{ij}^k \in \{0, 1\}, \quad \forall (i,j) \in A, \forall k \in V \quad (4.9)$$

Bu modelde ZPARP formülasyonundan farklı olarak, kapasite kısıdını ifade eden kısıt (3.8) kaldırılmış ve dağıtım zaman penceresi kısıdını ifade eden kısıt (4.8) ilave edilmiştir. Kısıt (4.7) ve (4.8), bölüm 4.5.2’ de derinlik öncelikli arama prosedürü madde (6)’ da açıklanan toplama ve dağıtım zaman pencereleri kısıtlarını ifade etmektedir.

### **4.3 Mevcut Problemin Zaman Pencereci Araç Rotalama Problemi ile Karşılaştırılması**

Toplama ve dağıtım zaman pencereci araç rotalama problemi olarak ifade ettiğimiz problemin zaman pencereci araç rotalama problemi (ZPARP) ile arasındaki temel iki fark, araç kapasitesi kısıdı ve araçların turlarını sonlandırarak depoya dönmelerini zorlayan dağıtım zamanı kısıdındır. Bu iki fark aşağıda daha ayrıntılı olarak ifade edilmiştir.

- (i) ZPARP’ nde dikkate alınan araç kapasitesi kısıdı, mevcut problemde gönderilerin genellikle kapasiteyi zorlamayacak ağırlık ve hacimde paket ya da dosya şeklinde olmasından ve de modelde dikkate alınan kargo dağıtımının yalnızca şubeler ve müşteriler arasındaki yerel ağında, müşteri ve dolayısıyla gönderi sayısının nispeten daha az olmasından dolayı dikkate alınmamıştır.
- (ii) Mevcut problemde müşteriler,  $[t_i - \delta_p, t_i]$  toplama zamanı penceresiyle ve  $[0, \tau_i + \delta_D]$  dağıtım zamanı penceresiyle karakterize edilmiştir. ZPARP’ nde ise müşteriler gönderilerinin toplanması için hazır oldukları en erken toplama zamanı ve en geç toplama zamanını ifade eden, yalnızca tek bir zaman penceresi  $[a_i, b_i]$  ile karakterize edilmişlerdir. Depodan dağıtım için zaman kısıdı, deponun zaman penceresi  $[a_{n+1}, b_{n+1}]$  ile ifade edilmektedir ( $i = 0$  ve  $n + 1$  depo düğümünü ifade etmektedir). Depodan dağıtım için en geç zaman zamanı ifade eden  $b_{n+1}$ , araçları turlarını bitirerek depoya dönmeye zorlamaktadır. Mevcut problemde ise araçları depoya dönmeye



zorlayan ziyaret ettikleri müşteri konumlarına ait en küçük dağıtım zamanı penceresidir.

İki problem arasındaki farkı ortaya koymak amacıyla mevcut problem için Lin (2008) test problemleri üzerinde uygulanan dal ve sınır ile çözülen derinlik öncelikli arama prosedürü, yukarıda ifade edilen farklar dikkate alınarak yeniden düzenlenmiş ve Solomon test problemlerinin bazıları için de uygulanmıştır. Elde edilen rotalar üzerinde Lin (2008) örnek problemleri için uygulanan küme bölümlene formülasyonunda kolon üretimi uygulanarak tüm müşterileri ziyaret eden tüm uygun çözümler üretilerek en düşük maliyetli çözüm elde edilmiştir.

#### **4.4 Zaman Pencereci Araç Rotalama Problemi için Literatür Araştırması**

Zaman pencereci araç rotalama problemi (ZPARP) ve toplama ve dağıtım ile araç rotalama problemi (TDARP) üzerine literatürde birçok çalışma olmasına karşın, mevcut problemle ilgili çok az çalışma olduğu söylenebilir. Bu nedenle, zaman pencereci araç rotalama problemi ve kesin çözüm yaklaşımları üzerine ulaşılabilen literatürde yer alan çalışmalar sunulmuştur.

Desrosiers v.d. (1984), m-gezgin satıcı problemi olarak da ifade edilen zaman pencereci ve kapasite kısıdı olmayan ARP probleminin çözümü için, simpleks ve dal-sınır ile çözülen küme bölümlene probleminde, düğümler üzerinde zaman pencereci en kısa yol algoritması ile kolon üretimini kullanmışlardır[25].

Desrochers v.d. (1992), tamsayı küme bölümlene formülasyonunu çözmek için dal ve sınır algoritmasında kullanılan doğrusal programlama relaksasyonunu kolon üretimi ile çözerek elde edilen uygun kolonları, zaman pencereci ve kapasite kısıtlı en kısa yol probleminin dinamik programlama ile çözümünde kullanmışlardır[26].

Forbes v.d. çok depolu otobüs programlama probleminin çözümü için problemin doğrusal programlama relaksasyonlarını kullanan kesin algoritma sunmuşlardır. Birinci olarak, ikinci relaksasyon için ikili uygun çözümü elde etmek amacıyla kullanılan basit ağ akış problemi, ikinci olarak ise, ikili simpleks kullanılarak çözülen çoklu-mal ağ akış problemi relaksasyonlarını uygulamışlar ve daha sonra optimum tamsayılı çözümü elde etmek için dal-ve-sınır algoritmasını kullanmışlardır. üç depo ve altı yüzden fazla seyahati içeren problemleri çözmeye başarılı olmuşlardır[27].

Hadjiconstantinou v.d. (1995), KKARP için, q-yol ve k-en kısa yol hesaplamalarına dayanan orijinal problemin iki relaksasyonunun kombinasyonundan elde edilen alt sınırları kullanan kesin algoritma sunmuşlardır. Problemin boyutunu indirgemek ve alt sınırların kalitesini arttırmak için, bu sınırların hesaplanmasından elde ettikleri bir dizi indirgeme kriterini uygulamışlar ve sonuçta oluşan alt sınırları arama ağacı prosedürüne girerek problemin optimum çözümünü elde etmişlerdir[28].

Soumis v.d. (1998), zaman pencereli ve çok depolu araç programlama problemini, dal-ve-değer sistemi içine yerleştirilmiş kolon üretimi yaklaşımını kullanarak, zaman değişkenleri ile tamsayılı doğrusal olmayan çok-mal ağ akış modeli olarak formüle etmişlerdir. Birbirini izleyen iki görev arasında minimum bekleme zamanı yerine maliyeti dikkate almışlardır. Geliştirdikleri bu yeni ve daha gerçekçi maliyet yapısı modelde doğrusal olmayan amaç fonksiyonunun oluşmasına neden olmuştur[29].

Bard v.d. (2002), zaman pencereli araç rotalama problemine dal-ve-sınır metoduna dayanan kesin çözüm yaklaşımı sunmuşlardır. Hesaplamalarda, optimum çözüm üzerinde sürekli artan alt sınırlar, yeni bulunan, geçerli eşitsizlikleri içine alan bir dizi gevşetilmiş problemlerin çözülmesi ile elde edilmiştir. Uygun çözümler ya da üst sınırlar ise greedy rastgele uyarlamalı arama prosedürünün (GRASP) yardımı ile elde edilmiştir[30].

Irnich v.d. (2005), dal-ve-değer algoritması içerisinde alt problem çözücüsü olarak tüm uygun rotaların üretilmesi için kullanılan ve *Labeling* algoritması olarak

tanımlanan k-çevrim eliminasyonu ile kaynak kısıtlı en kısa yol problemi algoritması geliştirmişlerdir[31].

Gendreau v.d. (2006), aynı aracın zaman penceresi ile bir dizi müşteriye servis yapmak için birkaç rotayı gerçekleştirdiği problemi çözmek amacıyla bir kesin algoritma tanımlamışlardır. Sundukları metot, kaynak kısıtlı basit en kısa yol algoritmasına dayanan iki aşamalı bir algoritmadır: birinci aşamada, tüm baskın olmayan uygun rotalar üretilmiş; ikinci aşamada ise bazı rotalar seçilmiş ve aracın günlük rotasında sıralanmıştır[32].

Chabrier (2006), kolon üretimine dayalı basit en kısa yol (elementary shortest path) ile araç rotalama problemine kesin çözüm yaklaşımı sunmuştur. Arama ağacını budama ile daha iyi bir alt sınır elde ederek Solomon (1987)' un 17 örnek problemi için kesin çözümü elde etmiştir[33].

Mourgaya ve Vanderbeck (2007), periyodik araç rotalama probleminin çözümü için indirgenmiş kolon üretimi prosedürü ardından yuvarlama sezgiseli kullanmışlardır. Bu matematiksel programlamaya dayalı prosedür ile, literatürde ileri sezgisellerle çözümü en çok ele alınan beşten fazla çalışma günü ve 50-80 müşterili problemleri çözmüşlerdir[34].

Oukil v.d. (2007), çok depolu araç rotalama probleminin doğrusal relaksasyonunu çözmek için kolon üretimi, değişkenleri önceden belirleme ve dengelemenin bir kombinasyonu olan çözüm yaklaşımı geliştirmişlerdir. Bu dengelenmiş kolon üretimi yaklaşımı, penaltı ve kredi (trust) bölgesi kavramlarını birleştiren yakınsak tipte bir algoritmadır. Doğrusal penaltı fonksiyonunu, dengelenmiş problemlerin kolon üretimi tarafından çözümünden önce mevcut ikili denge merkezini içeren kısımda, ikili değişkenleri kısıtlamak için kullanmışlardır[35].

Moccia vd. (2008), konveks olmayan parçalı doğrusal maliyetler, zaman pencereleri ve diğer kısıtlar ile başlangıç-hedef tamsayı çoklu-mal akış formülasyonu uygulayarak alt sınırları hesaplamak için kolon üretimi algoritması

tasarlamışlardır. Uygun tamsayı çözümlerini bulmayı amaçlayan sezgisel içine, bu kolon üretimi algoritması yerleştirilmiştir[36].

Qureshi v.d. (2009), yarı esnek zaman pencereci araç rotalama ve programlama problemi (YEZPARP) için kolon üretimine dayanan kesin çözüm yaklaşımı sunmuşlardır. Dantzig–Wolfe ayrıştırma metoduyla oluşan geç varış cezalı ve kaynak kısıtlı başlangıç en kısa yol problemi, alt problem olarak çözülmüştür. Kesin çözümler YGZPARP ve katı zaman pencereci versiyonu Solomon’ un test problemleri ve Tokyo yol ağına ait örnekler üzerinde kıyaslamışlar ve YGZPARP çözümünün az sayıda rota için toplam maliyeti düşürdüğünü ve geç varış cezalarının toplam maliyete yalnızca küçük miktarda artış getirdiğini göstermişlerdir[37].

Azi v.d. (2009), her bir aracın kesin zaman sınırlarına bağlı olarak işgünü boyunca birçok rotayı gerçekleştirdiği, rotaların kazanç, talep ve zaman pencereci müşteriler üzerinden tanımlandığı zaman pencereci araç rotalama problemi için, kolon üretimini kullanarak, alt sınırların küme kapsama formülasyonunun doğrusal programlama relaksasyonunun çözümü ile hesaplandığı dal-ve-değer yaklaşımı sunmuşlardır[38].

Brønmo v.d. (2009), esnek kargo boyutlu gemi programlama problemi için Dantzig–Wolfe analizi prosedürüne dayalı kolon üretimi tekniği sunmuşlardır. Ele aldıkları bu problem, sabit değerler yerine belirli sınırlar içerisinde değişen değerlerde olabilen kargo büyüklükleri ve buna bağlı olarak yükleme ve boşaltma zamanları ile tanımlanan zaman pencereleri ile toplama ve dağıtım problemidir. Algoritmalarında problemi, asıl problem ve birkaç alt problem şeklinde ele alarak formüle etmişlerdir. Alt problemlerin çözümüyle dinamik olarak kolonların üretildiği orjinal problemin küme bölümlenme formülasyonunu kullanmışlardır. Elde ettikleri sonuçlar küçük ya da dar sınırlı kısıtlara sahip problemlerde optimum sonucu verirken, daha büyük ya da geniş sınırlı kısıtlara sahip problemlerde optimumdan küçük bir sapmayla yaklaşık sonuçlar elde etmişlerdir[39].

## **4.5 Toplama ve Dağıtım Zaman Pencereci Araç Rotalama Problemi için Uygulanan Çözüm Yaklaşımı Adımları**

Ele alınan problem genel olarak, bir kargo şirketinin bir günlük çalışma süresinde, her bir müşteri için zaman penceresi sınırları içerisinde, müşterilerin gönderilerinin toplanması ve şubeye (depo) ulaştırılarak şubenin bağlı olduğu transfer merkezine dağıtılması şeklindedir. Müşteri sayısı  $N$  arttığında kesin çözüm yaklaşımıyla oluşturulabilecek tüm uygun rotaların bulunmasından sonra, bu rotalar içerisinde tüm müşterilere servis yapabilecek en düşük maliyetli çözümün bulunması uzun zaman gerektirmektedir. Bu nedenle, problemi kabul edilebilir zaman içerisinde çözmek için, elde edilen tüm rotalar bağımsız birer çözüm grubu olarak ele alınmıştır.

### **4.5.1 Toplama ve Dağıtım Zaman Pencereci Araç Rotalama Problemi için Model Kabulleri**

Modeldeki kabuller, Lin (2008)' in modelindeki kabulleri esas alınarak düzenlenmiştir. Modelde, kargo dağıtım şubeleriyle müşteri konumları arasındaki bölgesel servis, çok ağır olmayan ve araçların servis devresinde tüm dokümanlarını toplayabildikleri sayıca çok fazla ve büyük boyutlu olmayan dokümanları kapsadığından araçlar için kapasite kısıdı dikkate alınmamıştır. Bu kabul, Lin (2008) [24], Shang ve Cuff (1996)[40], Langevin and Soumis (1989)[41], Mitrovic' - Minic' v.d. (2004)[42] ve Mitrovic'-Minic' ve Laporte (2004)[43]' de koli ya da tıbbi belgelerin toplama ve dağıtımını için kabul edilmiştir.

Kargo dağıtım ağında, gönderilerin toplanarak teslim adreslerine zamanında ulaştırılmak üzere şubede toplanıp transfer merkezlerine iletilmesi ve şubeye teslim adreslerine ulaştırılmak üzere gelen gönderilerin ilgili adreslere dağıtılması şeklinde iki yönlü transfer söz konusudur. Şubeye teslim adreslerine ulaştırılmak üzere gelen gönderilerin dağıtılmasında rotalama ve termin açısından güçlük söz konusu değildir. Şubeye gelen gönderiler bir günlük servis devresinin ilk yarısında ilgili adreslere dağıtılır. Modelde esas alınan, müşteri konumlarından toplama zamanı penceresinde

gönderilerin toplanarak, teslim adreslerine zamanında ulaştırılmaları için belirlenen dağıtım zamanı penceresinde şubeye dönmelerini sağlayan en kısa mesafeli dolayısıyla en düşük maliyetli rotaların planlanmasıdır.

Bir servis devresi içinde verilen  $N$  müşteri kümesi için,  $i (= 1, \dots, N)$  müşterisi için  $t_i$  olarak belirtilen toplama zamanı ile toplama zaman penceresi, verilen daha erken kabul toleransı  $\delta_p (> 0)$  tüm müşteriler için geçerli olduğunda,  $[t_i - \delta_p, t_i]$ ' dir. Yerinde servis zamanı  $\varphi$ , tüm müşteri mevkilerinde söz konusudur. Tüm müşteriler için dağıtım konumu kolilerin teslimat dağıtımından (outbound delivery) önce işlendiği şubelerdir.  $i$  müşterisine dağıtım zamanı için  $\tau_i$  ile belirtilen zaman penceresi, verilen tolerans  $\delta_D (> 0)$  olduğunda  $[0, \tau_i + \delta_D]$ ' dir. Özet olarak, bu çalışmada modelin temelini teşkil eden Lin (2008)' in ortaya koyduğu kabuller aşağıdakileri içermektedir[24]:

- (i) Servis periyodu bir günlük çalışma süresidir.
- (ii) Kargo taşımacılığında, müşteri adreslerinden toplanan dokümanların boyutlarının küçük olmasından dolayı araç kapasitesi kısıtlanmamıştır.
- (iii) Her bir müşterideki toplama talebi ana dağıtım yerindeki (şube) dağıtım talebinden önce meydana gelir.
- (iv) İki konum arasındaki seyahat zamanı simetrik ya da asimetrik olabilir.
- (v) Sadece iki konum arasındaki uzaklıklar mevcut olduğunda araç hızının  $V$  km/h' ın, ortalama trafik hızı olduğu varsayılır. Bu, toplam seyahat mesafesini tahmini seyahat süresine dönüştürür.
- (vi) Müşteri adresinden toplama, belirlenen toplama zamanından  $\delta_p$  dakika önceden daha erken olmamalıdır.
- (vii) Eğer araç müşteri adresine en erken toplama zamanından önce varırsa bekleme zamanına müsaade edilir.
- (viii) Yerinde servis süresinin  $\varphi$  dakika olduğu varsayılır.
- (ix) Posta merkezinde dağıtım zamanı, belirlenen dağıtım zamanından sonra  $\delta_D$  dakikadan daha geç olmamalıdır.

#### 4.5.2 Kesin Rotaların Oluşturulması için Derinlik Öncelikli Arama

Derinlik öncelikli arama (depth-first search), bir ağaç yapısı ya da yol ağını araştıran ya da derinlemesine inceleyen bir algoritmadır. Bu algoritmanın işleyişi, başlangıç (kök) olarak seçilen bir düğümden başlayarak gidebileceği en uzak noktaya kadar tüm dalları taramak şeklindedir. Başka bir ifadeyle, derinlik öncelikli arama prosedürü ağaçtaki en son uygun düğümden dallanmaktır [10].

Lin (2008)'in modeli temel alınarak Zaman Pencereci Araç Rotalama Problemi'nde, zaman penceresi kısıtlarını sağlayan müşterilerin uygun sıralanmasından oluşabilecek bütün rotalar dal ve sınır ile çözülen derinlik öncelikli arama (depth-first search) algoritmasına dayanan rota oluşturma prosedürü ile tespit edilmiştir. Derinlik öncelikli arama prosedürünün adımları şu şekildedir:

(1) Öncelikle tüm müşteriler toplama zamanı  $t_i (i = 1, \dots, N)$ 'ye göre küçükten büyüğe sıralanır.

(2) Oluşan sıralı müşteri listesindeki tüm müşteriler için sırayla uygunluk kontrolü yapılır. Şöyle ki:

- Toplama zamanı kısıdı: seyahat zamanı ( $depo \rightarrow i$ )  $\leq t_i$
- Dağıtım zamanı kısıdı:  $\{Max\{t_i - \delta_p, seyahat zamanı (depo \rightarrow i)\} + \varphi + seyahat zamanı (i \rightarrow depo) \leq \tau_i + \delta_D$

Yukarıdakilerin ikisinden birini sağlayamayan müşteri düğümleri çıkarılır. Bu şartları sağlayan uygun müşteri düğümlerinin sayısı için  $N$  değeri yeniden gözden geçirilip düzeltilir.  $t_{L_1} \leq t_{L_2} \leq \dots \leq t_{L_N}$  olduğu durumda, uygun müşteri düğümlerinin sıralı listesi,  $L = \{L_1, L_2, \dots, L_N\}$  olarak oluşturulmuş olur.

(3) Arama ağacındaki seviyelerin (ya da rotada ki düğümlerin) maksimum sayısı, başlangıç ve bitişin depo olduğu ve her bir ara seviyenin müşteri düğümünden oluştuğu durumda  $N + 2$ 'dir.

(4) Arama ağacındaki bir  $(i, j)$  yayı, müşteri düğümü  $i$ ' den (toplamadan sonra)  $j$  düğümüne seyahate tekabül eder, eğer aşağıdaki Adım (6)' ya göre böyle bir hareket mümkünse.

(5) Dallanma stratejisi: Düğümlerin sıralı kümesinden  $(depo, i_1, i_2, \dots, i_k)$  kısmi ağaç  $\mathcal{T}'$  nin meydana geldiğini farz edelim.  $\mathcal{T}'$  nin servis bitiş zamanını  $t(\mathcal{T})$  ile gösterelim.  $L$  listesindeki  $\mathcal{T}'$  de henüz oluşmamış en erken düğüm yani  $t_{i_1} \leq t_{i_2} \leq \dots \leq t_{i_k} \leq t_j$  şartını sağlayan düğüm seçilir ve buna  $j$  denilir.

(6) Her bir seviyede uygunluk kontrolü: Kısmi ağaç  $\mathcal{T} (depo, i_1, i_2, \dots, i_k)$ ' ya bir  $j$  düğümünü eklemeye karar vereceğimiz zaman,  $i_k$ ' daki servis  $t(\mathcal{T})$  süresinde bitiyorsa aşağıdaki iki şartı kontrol et.

- Toplama zamanı kısıdı:  $t(\mathcal{T}) + \text{seyahat zamanı } (i_k \rightarrow j) \leq t_j$
- Dağıtım zamanı kısıdı:  $Max\{t_j - \delta_p, t(\mathcal{T}) + \text{seyahat zamanı } (i_k \rightarrow j)\} + \varphi + \text{seyahat zamanı } (j \rightarrow depo) \leq Min\{\tau_{i_1}, \tau_{i_2}, \dots, \tau_{i_k}, \tau_j\} + \delta_D$

Eğer her iki kısıt da sağlanıyorsa, kısmi ağaç  $\mathcal{T}'$  yi  $\mathcal{T} \rightarrow (\mathcal{T}, j)$  ve servis bitiş zamanı  $t(\mathcal{T})$ ' yi de  $t(\mathcal{T}) \rightarrow Max\{t_j - \delta_p, t(\mathcal{T}) + \text{seyahat zamanı } (i_k \rightarrow j)\} + \varphi$  olarak güncelle.

(7) Geri dönüş(*Backtracking*) stratejisi: Adım (5) ve (6) yoluyla kısmi ağaç  $\mathcal{T} (depo, i_1, i_2, \dots, i_k)$ ' ya eklenebilecek  $L$  listesinde hiçbir düğüm kalmadığında bu  $\mathcal{T} \rightarrow depo$ ' nun tamamlanmış kesin bir rota oluşturduğu anlamına gelir. Bundan sonra  $\mathcal{T}'$  deki bir önceki seviyeye yani  $k - 1$ ' e geri dön ve  $L$  listesinde  $i_k$ ' dan sonra (yani onun yerine) yeni bir düğüm dalandır.

(8) Son bulma şartı: Bu, algoritma  $L_N$  düğümünü içeren kısmi ağaç  $\mathcal{T}'$  nin ilk seviyesine geri döndüğünde meydana gelir. Bunun sonucu olarak bulunan en son rota  $(depo \rightarrow L_N \rightarrow depo)$  olmaktadır[24].



Öncelikle müşteri düğümleri toplama zamanlarına ( $t_i$ ) göre küçükten büyüğe sıralandıktan (1) sonra her müşteri için uygunluk kontrolü yapılır (2). En erken toplama zamanına sahip olan düğümden başlamak üzere sırasıyla, her bir müşteri konumu kaynak düğümü olarak seçilir ve sıralı müşteri listesinde kendisinden hemen sonra gelen düğümden başlayarak, toplama ve dağıtım zaman penceresi kısıtlarını sağlayan düğümlere giden yollar kaynak düğüme eklenerek rotalar oluşturulur (5)-(6). Sıralı müşteri listesinde,  $i$  kaynak düğüme eklenecek hiçbir uygun müşteri düğümü kalmadığında son düğümden bir önceki düğüme dönülerek,  $L$  listesinde rotayı oluşturan son düğümden sonra gelen müşteri düğüme dallanılarak adım (5)'ten itibaren adımlar tekrar edilir.  $L$  listesindeki son düğüm için de aynı adımlar uygulandığında elde edilen son rota, depodan sıralı müşteri listesindeki son müşteri düğüme giden yol olacaktır.

Gerçek hayat verilerinde müşteriler, kentsel bölgelerin karakteristik özellikleri olan yoğun olarak yerleşilen bölgelere konumlanmıştır.  $\delta_p$ , seyahat zamanı ile kıyaslandığında küçüktür, uygun rotalardaki müşterilerin en erken toplama zamanı sıralamasında servis edildiği farz edilebilir. Bu nedenle müşteriler derinlik öncelikli arama prosedürü tarafından mümkün rota grupları halinde düzenlenmeden, ilk olarak *toplama zamanının*  $\{t_i, i = 1, \dots, N\}$  artan sırasında sıralanır[24].

Derinlik öncelikli arama prosedürü ile oluşturulan her bir rota, şubeye gelen ve giden gönderilerin adresi olan uygun müşteri gruplarını diğer bir ifadeyle ziyaret edilen müşterilerin sırasını temsil etmektedir. Amaç fonksiyonu doğrultusunda çözüm kümesi, tüm *uygun* müşterileri ziyaret eden en düşük toplam maliyete sahip rota grubudur. Kesin çözümün elde edilebilmesi için öncelikle tüm *uygun* müşterileri ziyaret eden rota gruplarının elde edilip daha sonra içlerinden en düşük maliyetli olanın seçilmesi gerekmektedir. Problem boyutunu küçültmek ve kabul edilebilir sürede çözümünü sağlamak için, derinlik öncelikli arama prosedürüne dayanan dal-ve-sınır ile çözülen küme bölümlene formülasyonu içerisinde kolon üretimi uygulanmıştır.

### 4.5.3 Küme Bölümleme ve Kolon Üretimi

Küme bölümleme (set-partitioning) modeli ilk olarak Balinski ve Quandt (1964) tarafından sunulmuştur. Küme bölümleme modeli, her biri farklı  $(i, j)$  yolları ve  $n$  düğümlerini kapsayan farklı çevrimler (başka bir deyişle rotalar) ile bağlantılı ikili değişkenlerin üstel değerlerini kullanır[12, s.21].

#### Küme bölümleme Formülasyonu:

Parametreler:

- $N$  Bir servis devresindeki müşterilerin sayısı  
 $n$  Prosedür  $\mathcal{Y}$  veya Prosedür  $\mu$  ile oluşturulan tekli ya da çoklu rota gruplarının sayısı  
 $C_g$  Servis grubu  $g$ ' nin maliyeti (=araçların sabit maliyeti ve seyahat maliyetinin toplamı)  
 $S_i$   $i$  müşterilerine servis edebilen grupların kümesi,  $i= 1, \dots, N$

$$\text{Kararlar: } X_g \begin{cases} 1, \text{ eğer } g \text{ grubu seçildiyse (bir araca atanmışsa)} \\ 0, \text{ diğer durumda} \end{cases} \quad g= 1, \dots, n$$

$$\text{Min } Z = \sum_{g=1}^n C_g X_g \quad (4.10)$$

Kısıtlar:

$$\sum_{g \in S_i} X_g = 1, \quad i = 1, \dots, N \quad (4.11)$$

$$X_g = 0, 1, \quad g = 1, \dots, n \quad (4.12)$$

Amaç, minimum maliyetle tüm müşterilere servis edecek grupların kümesini seçmektir (4.10). S modelindeki kısıtların tekli kümesi, bazı uygun gruplarda her bir müşterinin en azından bir kere ziyaret edilmiş olmasını gerektirir. Bu aynı zamanda her bir araç için yalnızca bir rotanın kullanılabilceği anlamına gelmektedir (4.11)[24].

Yukarıdaki küme bölümlenme formülasyonunda uygulama karmaşıklığı açıkça gösterilmemiştir. Bunun yerine tüm bu komplikasyonlar derinlik öncelikli arama prosedürüne dayanan dal ve sınır formülasyonunda kolonlar içine yerleştirilmiştir. Başka bir ifadeyle rotaların üretilmesi sırasında modele ait tanımlanan tüm kısıtlar uygulanmıştır.

Küme bölümlenme modeli genellikle, doğrudan çözülemeyecek kadar büyük sayıda, araçların turlarına (rotalarına) karşılık gelen kolonlar yani rotalar üretir. Modeli kabul edilebilir zamanda çözebilmek için, derinlik öncelikli arama prosedürüyle üretilen baskın olmayan tüm uygun kolonların (rotaların) çözümde dikkate alınması yerine, küme bölümlenme formülasyonunun yalnızca kolonların alt kümelerini içeren kısıtlanmış versiyonu çözülmüştür.

Bu çözüm yaklaşımı araç rotalamanın çalışma alanı olarak, Rao ve Zions (1968), Foster ve Ryan (1976), Orloff (1976), Desrosiers, Soumis ve Desrochers (1984), Agarwal, Mathur ve Salkin (1989), Desrochers, Desrosiers ve Solomon (1990) tarafından tanımlanmış olan *kolon üretimi (column generation)* tekniğidir. Kolon üretiminde, tüm mümkün kolonların yalnızca sınırlanmış alt kümesini içeren indirgenmiş problem tekrarlanarak çözülür[16].

Modelin çözümü için uygulanan, derinlik öncelikli arama prosedürüne dayanan dal-ve-sınır ile çözülen küme bölümlenme formülasyonu içerisinde kolon üretiminin adımları şu şekildedir:

- Derinlik öncelikli arama prosedüründe, sırasıyla en geç toplama zamanlarına göre sıralanmış olan her bir müşteriden başlayan ve depoda sonlanan baskın olmayan tüm uygun rotalar üretilir.

- Üretilen rotalar depodan sonra gelen başlangıç düğümlerine göre gruplandırılır.
- Oluşan her bir rota grubu, problemin bir alt kümesi olarak değerlendirilip bunlar üzerinde küme bölümlene formülasyonu uygulanır. İlk gruptan başlayarak tüm uygun müşterilere yalnızca bir defa servis edecek uygun rota kombinasyonları yani uygun çözümler üretilir. Bu kolon üretimi adımları şu şekilde gerçekleştirilmiştir:
  - (i) Grubun ilk elemanı olan rotadan başlayarak sırasıyla tüm rotalar için diğer rota gruplarından çözümü oluşturacak rotalar küme bölümlene formülasyonuna göre seçilerek başlangıç rotasıyla oluşabilecek tüm uygun çözüm grupları oluşturulur.
  - (ii) Oluşan çözüm grubu içerisinde en düşük maliyetli olanı seçilerek başlangıç çözümü olarak atanır. Diğer çözümler ise elenir.
  - (iii) Rota grubunun tüm rotaları için adım (i) tekrarlanır.
  - (iv) Oluşan çözümlerin her birinin maliyeti, başlangıç çözümünün maliyetiyle kıyaslanarak daha düşük maliyetli bir çözüm elde edildiyse en iyi çözüm olarak atanır, aksi halde bir önceki çözüm en iyi çözüm olarak tutulur, diğerleri elenir.
  - (v) Daha sonra sırasıyla tüm rota grupları için adım (i), (ii), (iii) ve (iv) tekrarlanır.
  - (vi) Son rota grubunun son elemanı için de yukarıdaki adımlar tekrarlandığında program sonlanır, elde edilen son çözüm optimum çözümdür.

#### 4.6 Örnek Problem Uygulamaları

Toplama ve dağıtım zaman pencereli araç rotalama problemini optimum olarak çözmek amacıyla tasarlanan iki aşamalı çözüm yaklaşımını daha iyi anlatabilmek için Lin (2008)' de yer alan altı müşterili örnek probleme ait veriler Tablo 4.2 ve Tablo 4.3' te, çözümün ilk aşaması olan dal ve sınır ile çözülen derinlik öncelikli arama algoritmasıyla elde edilen rotalar Tablo 4.4' te ve çözümün ikinci aşaması olan, küme bölümlene formülasyonunda kolon üretimi uygulanmasıyla elde edilen optimum çözümün sonuçları Tablo 4.5' te verilmiştir. Optimum çözümün şekilsel gösterimi ise Şekil 4.1' de verilmiştir.

Lin (2008)' de, mevcut problemde kesin çözüm elde etmek için derinlik öncelikli arama prosedürüyle tüm baskın olmayan rotalar tekli ve çoklu rota grupları şeklinde elde edildikten sonra, en düşük maliyetli çözümün bulunması için küme bölümlene formülasyonu uygulanmış ve bu çözüm bağımsız strateji olarak adlandırılmıştır. Esas olarak, Lin' in çalışmasının amacı ise, araçların işbirliğine dayanan, transfer noktalarında araçları diğer müşterileri ziyaret etmek için serbest bırakarak uygun rotaları birleştirip, maliyeti ve araç sayısını düşürmek gibi ilave artırımlar sağlamaktır. İşbirliğine dayanan bu strateji Lin (2008) tarafından *kooperatif strateji* olarak tanımlanmıştır. Kooperatif stratejinin kazanımlarını ölçmek ve değerlendirmek için bağımsız stratejiye başvurulmuştur.

Bu çalışmada, model aldığımız kargo taşımacılığı uygulamalarında böyle bir işbirliği stratejisinin uygun olmaması nedeniyle, Lin (2008)' in örnek problemleri üzerinde uygulanan çözüm yaklaşımı, yalnızca Lin (2008)' in bağımsız strateji sonuçları ile karşılaştırılmıştır.

Tablo 4.2 Lin (2008) örnek problemine ait müşterilerin karakteristikleri

Müşteri numarası	Depo	1	2	3	4	5	6
<b>Toplama zamanı penceresi</b>	-	[117, 122]	[102, 107]	[137, 142]	[191, 196]	[228, 233]	[305, 310]
<b>Dağıtım zamanı penceresi</b>	-	[0, 281]	[0, 280]	[0, 318]	[0, 360]	[0, 379]	[0, 393]

Tablo 4.3 Lin (2008) örnek problemine ait müşteriler arası seyahat süreleri

Müşteri Numarası	Depo	1	2	3	4	5	6
Depo	-	72	102	89	59	40	45
1		-	31	26	38	35	35
2			-	26	61	65	63
3				-	38	49	45
4					-	24	16
5						-	8
6							-

Tabloda, seyahat zamanları simetrik olduğundan tek yönlü olarak gösterilmiştir.

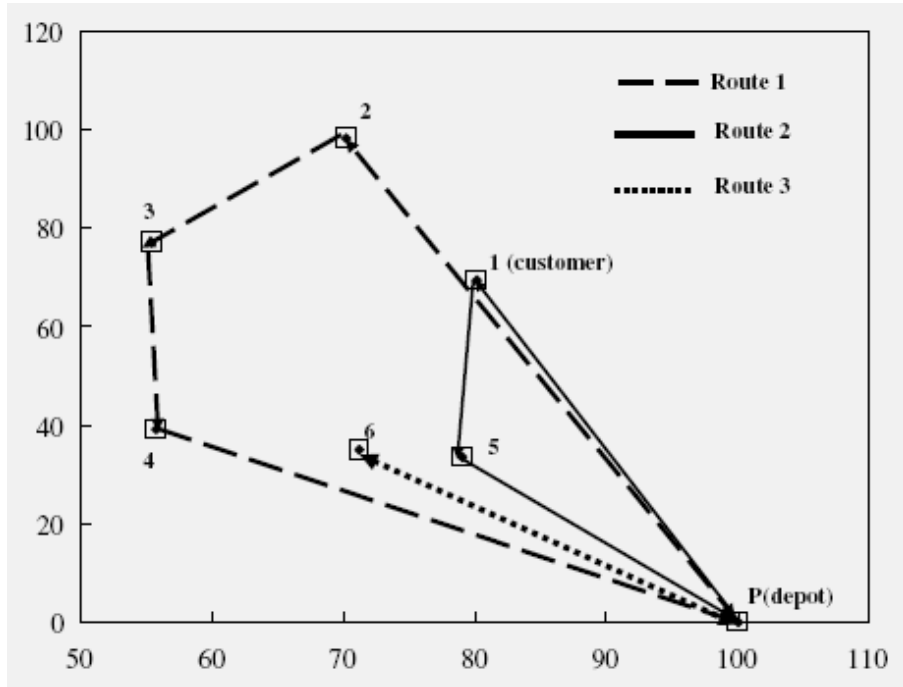
Yerinde servis zamanı ( $\varphi$ ) = 5 dk.

Araçların sabit maliyetleri = \$7000.

Birim seyahat maliyeti = saat başına \$70 = dakika başına  $\$1\frac{1}{6}$ .

Tablo 4.4 Lin (2008) örnek problemi için derinlik öncelikli arama prosedürüyle elde edilen rotalar

Rota Grubu	Rota Numarası	Oluşan Rotalar	Depoya Dönüş Zamanı	Rota Grubu	Rota Numarası	Oluşan Rotalar	Depoya Dönüş Zamanı
<b>1. Rota Grubu</b>	1. Rota	<b>2</b>	209	<b>3. Rota Grubu</b>	13. Rota	<b>3</b>	231
	2. Rota	<b>2-3</b>	231		14. Rota	<b>3-4</b>	255
	3. Rota	<b>2-3-4</b>	255		15. Rota	<b>3-4-5</b>	273
	4. Rota	<b>2-3-4-5</b>	273		16. Rota	<b>3-5</b>	273
	5. Rota	<b>2-3-5</b>	273	<b>4. Rota Grubu</b>	17. Rota	<b>4</b>	255
	6. Rota	<b>2-4</b>	255		18. Rota	<b>4-5</b>	273
	7. Rota	<b>2-4-5</b>	273		19. Rota	<b>4-5-6</b>	355
	8. Rota	<b>2-5</b>	273		20. Rota	<b>4-6</b>	355
<b>2. Rota Grubu</b>	9. Rota	<b>1</b>	194	<b>5. Rota Grubu</b>	21. Rota	<b>5</b>	273
	10. Rota	<b>1-4</b>	255		22. Rota	<b>5-6</b>	355
	11. Rota	<b>1-4-5</b>	273	<b>6. Rota Grubu</b>	23. Rota	<b>6</b>	355
	12. Rota	<b>1-5</b>	273				



Şekil 4.1 Lin (2008) örnek problemi için optimum çözüm [24].

Tablo 4.5 Tasarlanan modelin Lin (2008) örnek problemi için optimum çözümü

Ziyaret Edilen Müşteriler					
<b>1. rota</b>	<b>Depo</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>Depo</b>
Variş Zamanı ve Toplama Zaman Penceresi (dk)	-	102 ∈ [102, 107]	133 < 137	180 < 191	255
Bekleme Zamanı (dk)	-	-	4	11	-
Ayrılış Zamanı (dk)	-	107	142	196	-
Dağıtım Zamanı Penceresi	-	[0, 280]	[0, 318]	[0, 360]	-
Rota için En Geç Dağıtım Zamanı Sınırı	-	280	-	255 < 280	-
<b>2. Rota</b>	<b>Depo</b>	<b>1</b>	<b>5</b>	<b>Depo</b>	
Variş Zamanı ve Toplama Zaman Penceresi (dk)	-	72 < 117	157 < 228	273	
Bekleme Zamanı (dk)	-	-	71		
Ayrılış Zamanı (dk)	-	-	233		
Dağıtım Zamanı Penceresi	-	[0, 281]	[0, 379]		
Rota için En Geç Dağıtım Zamanı Sınırı	-	281		273 < 281	
<b>3. Rota</b>	<b>Depo</b>	<b>6</b>	<b>Depo</b>		
Variş Zamanı ve Toplama Zaman Penceresi (dk)	-	45 < 305	355		
Bekleme Zamanı (dk)	-	-			
Ayrılış Zamanı (dk)	-	310			
Dağıtım Zamanı Penceresi	-	[0, 393]			
Rota için En Geç Dağıtım Zamanı Sınırı	-	393		355 < 393	
<b>Çözüm Maliyeti (\$)</b>	<b>21539</b>	<b>Toplam Bekleme Zamanı</b>		<b>86 dk</b>	

Toplam Seyahat Zamanı = (102+26+38+59) + (72+35+40) + (45+45) = 462 dk

Toplam Maliyet = \$7000 × 3(aracı) + \$1 $\frac{1}{6}$  × 462 = \$21539

Toplam Bekleme Zamanı = (137 – 133) + (191 – 180) + (228 – 157) = 86 dk

(Aracın depodan ilk müşterinin en erken toplama zamanını karşılayacak şekilde hareket ettiği varsayılarak ilk müşteride bekleme süresi “0” kabul edilir).



#### 4.6.1 Test Problemlerinin Karakteristikleri

Lin (2008)'in toplama ve dağıtım zaman pencereli araç rotalama problemi için kullandığı ve bu çalışmada da kullanılan gerçek hayat verilerinin birinci kümesinin parametreleri aşağıdaki gibi hesaplanır:

- (i) 1-10 örneklerdeki konumlama verisi bir yerel dağıtım servisinden edinilmiştir. Her bir örnek bir tek depo ve 27-30 müşteriyi içermektedir. 1-4 örneklerinde seyahat uzunlukları sürücüler (Lin ve diğerleri, 2002) ve coğrafi bilgi sistemi tarafından, 5-10 örneklerinde Ağ Analist yazılımı (Lin ve Kwok, 2006) tarafından karşılanır.
- (ii) Maliyet bileşenleri, personel ve aracın sabit maliyetleri (= HK\$7000) kurye servisinden ve araçların değişken işletme maliyetleri (= HK\$70/sa) dağıtım servisinden edinilmiştir.
- (iii) Toplama ve dağıtım zaman pencerelerinin çeşitleri ( $\delta_p = 5 dk.$ ,  $\delta_D = 10 dk.$ ) kurye servisinden edinilmiştir. Her bir örnek için toplama zamanı ( $t_i$ ) ve dağıtım zamanı ( $\tau_i$ ), uygun depoya dönüş seyahat zamanı  $[0, T]$  sınırları içinde hesaplanarak benzetilmiştir. Ortalama hız  $V = 20.8 km/sa$  olarak kabul edilmiştir[24].

#### 4.6.2 Tasarlanan Modelin Çözümü

Tasarlanan model için geliştirilen iki aşamalı çözüm yöntemi, Windows Vista Home Premium 32 bit işletim sistemli, core-2-duo 2 GHz CPU ve 2 GB RAM' e sahip bilgisayarda Dev-C++ 4.9.9.2 programında kodlanarak uygulanmıştır.

Lin (2008) yerel örnekleri üzerinde, sunulan çözüm modelinin uygulanmasıyla elde edilen sonuçlar Tablo 4.6' da verilmiştir.

Tablo 4.6 Tasarlanan algoritmanın Lin(2008) test problemleri için sonuçları

Örnek Problem	Müşteri Sayısı/Uygun Müşteri Sayısı	Elde Edilen Rota Sayısı	Araç Sayısı	Çözüm Maliyeti	Çözümü Oluşturan Rotalar	Bağımsız strateji Maliyeti (araç sayısı)
1	27/20	66	12	86179.33	10-5 13-3 19-1 17-15 8-4 21-22 7-11 2 26 24-27 20 6	100716.00 (14)
2	27/21	66	14	100104.66	15-17 6 1 23-22 9-21 19 12-14 26-13 18 10-5 7 3-20 11 2	114640.17(16)
3	27/25	94	15	107615.00	9-20-7 22-1-13 15-2 23-16 11 21-25 12-19 6 17 14-24 26-10 4 27 18 3	107675.17 (15)
5	29/23	280	9	64152.67	4-26-17-2 22-13-12-10-14-21 3-28 1 19-23 6 7-9-29 11-20 27-16	64229.67 (9)

Tablo 4.6' nın devamı

6	29/17	75	9	64387.16	4-26-17 24 13-12-10-14 19-23 28-6 7-9 27 2 29	78887.67 (11)
9	30/21	173	9	64765.16	27-26-5-10 19 9-21-23-24 15-1 28-18-8-13 22-20 6-4 30 2	64824.67 (9)
10	30/26	287	12	86164.16	16-5-12-22 18-30-28-14-2 7-3-13-21 11 15-1 25 23-24 6-17 4-19 20 10 27	79034.67 (11)

Lin (2008) test problemleri üzerinde uygulanan dal ve sınır ile çözülen derinlik öncelikli arama prosedürü zaman pencereli araç rotalama problemi için yeniden düzenlenerek Solomon test problemlerinin bazıları için de uygulanmıştır. Elde edilen rotalar üzerinde Lin (2008) örnek problemleri için kullanılan küme bölümlene formülasyonunda kolon üretimi uygulanmasıyla tüm müşterileri ziyaret eden uygun çözümler üretilerek en düşük maliyetli çözüm elde edilmiştir. Elde edilen çözümlerin ayrıntılı sonuçları ve Solomon internet sitesinde[44] yayınlanan optimum çözümler Tablo 4.7' de verilmiştir. Kabul edilebilir çözüm zamanını ve bilgisayar hafızasını aşmasından dolayı, kolon üretimi aşamasında kolon sayısı için bir üst sınır belirleyerek çözüm zamanı kısaltıldığından optimuma yakın sonuçlar elde edilmiştir.

Tablo 4.7 Tasarlanan çözüm modelinin Solomon test problemleri için sonuçları

Örnek Problem	Müşteri Sayısı	Elde Edilen Rota Sayısı	Araç Sayısı	Çözüm Maliyeti	Çözümü Oluşturan Rotalar	Optimum Çözüm Maliyeti (Araç Sayısı)
R101	25	918	8	646	14-15-22-24 5-18-6 2-21-3-24-25 12-9-20-1 11-19-10 23-13 16-17 7-8	617.1(8)
R102	25	832	7	658	2-23-22-4-13 3-1-12-9-20-24 7-11-18-10-17 5-21-25 14-16 16-6 19-8	547.1(7)

## 5. SONUÇ ve GELECEK ARAŞTIRMA

Dağıtım sistemlerinde malların tedarik ve servisinin etkili bir biçimde planlanması ve yönetilmesi konusu üretim sektöründe olduğu kadar hizmet sektörü kapsamındaki taşımacılık faaliyetlerinde de büyük önem taşımaktadır. Bu nedenle, özellikle taşıma faaliyetlerinin planlaması kapsamındaki araç rotalama problemlerinin çözümü için, yöneylem araştırması ve matematiksel programlamaya dayalı optimizasyon yazılımları üzerine yapılan çalışmalar her geçen gün gelişerek artmaktadır.

Bu yazılımların etkinliği, problemi en doğru şekilde ifade eden modelin kurulması ve modelin çözümü için tasarlanan algoritmanın hatasız ve hızlı sonuç veren yapıda olmasına bağlıdır. Problem girdilerinin ve kısıtlarının artması modeli kurmakta olduğu kadar programın çalıştırılmasında da zorluğa ve hesaplama zamanının artmasına sebep olmaktadır. Gerçek hayat uygulamalarında karşılaşılan büyük boyutlu problemlerde bu sorunu aşmak için çözüm uzayının yalnızca sınırlı bir kısmıyla ilgilenen sezgisel ya da ileri sezgisel yöntemler geliştirilebildiği gibi kesin çözümler elde etmek için de problemi alt problemlere ayıran ve kabul edilebilir zamanda çözümünü sağlayan çeşitli matematiksel modeller uygulanabilmektedir.

Bu tezin konusu olan toplama ve dağıtım zaman pencereli araç rotalama problemi için kesin çözümün bulunmasında, problem boyutu arttıkça bilgisayar kapasitesi ve zaman gibi kısıtlamalardan dolayı etkin bir program tasarlanmasının oldukça güç olduğu görülmüştür. Bunun için problem birbirinden bağımsız alt problemlere ayırarak, her bir alt problem ayrı bir problem gibi optimize edilmiştir. Lin (2008) de bağımsız strateji olarak sunulan model temel alınarak geliştirilen, derinlik öncelikli arama prosedürüne dayanan dal-ve-sınır ile çözülen küme bölümlene formülasyonu içerisinde kolon üretimi uygulanmıştır.

Örnek alınan kargo taşımacılık şirketinin bir yerel şubesi ile müşteriler arasındaki servis ağında, her şubenin bulunduğu şehrin yalnızca belirli bir bölgesine servisten sorumlu olması dolayısıyla, müşteri sayısının fazlalığından ziyade müşteri konumlarının rastgele koordinatlarda olmasının rotalamayı güçleştirmesi söz konusudur. Bu nedenle müşteri sayısı 25 ila 30 arasında değişen örnek problemler çözülmüş ve sonuca ulaşılmıştır.

Aynı zamanda toplama ve dağıtım zaman pencereli araç rotalama probleminin zaman pencereli araç rotalama problemi ile benzerlik ve farklılıkları ortaya konulmuş ve program algoritması zaman pencereli araç rotalama problemine göre düzenlenerek Solomon test problemlerinden modele uygun olan iki örnek problem çözülmüştür.

Örnek uygulamalar göstermiştir ki, problemi alt problemlere ayırarak iki aşamalı olarak çözmek kesin çözümün bulunmasında etkili olmuştur.

Yapılan bu çalışmada kullanılan yöntem, daha sonra daha büyük boyutlu ve daha çok sayıda kısıtlara sahip gerçek hayat problemlerinin çözümü için sezgisel ve ileri sezgisel yöntemlerin geliştirilmesi ve sonuçlarının kıyaslanması açısından faydalı olacaktır.

## KAYNAKLAR

- [1] [http://www.arascargo.com/if\\_sorular.html#1](http://www.arascargo.com/if_sorular.html#1).
- [2] [http://www.megabilim.com/index2.php?option=com\\_content&do\\_pdf=1&id=337](http://www.megabilim.com/index2.php?option=com_content&do_pdf=1&id=337)
- [3] Jongen, H. Th., Meer, K., Triesch, E., Optimization Theory, Springer Science + Business Media, Inc., Kluwer Academic Publishers, Boston, (2004), s.313-357.
- [4] Sun, W., Yuan, Y-X., Optimization Theory And Methods Nonlinear Programming, Springer Science+Business Media, LLC, USA, (2006).
- [5] <http://wiki.mcs.anl.gov/NEOS/index.php/Optimization>.
- [6] Nikulin Y., Robustness in Combinatorial Optimization and Scheduling Theory: An Extended Annotated Bibliography, working paper, (2006).
- [7] [http://wiki.mcs.anl.gov/NEOS/index.php/Optimization\\_Tree](http://wiki.mcs.anl.gov/NEOS/index.php/Optimization_Tree).
- [8] Schrijver, A., Combinatorial Optimization: Polyhedra and Efficiency, Volume A, Springer, Germany, (2003), S.1-43.
- [9] <http://msdn.microsoft.com/en-us/library/ms178091.aspx>.
- [10] Beasley, J.E., Integer Programming Solution Methods, [http://people.brunel.ac.uk/~mastjjb/jeb/natcor\\_ip\\_rest.pdf](http://people.brunel.ac.uk/~mastjjb/jeb/natcor_ip_rest.pdf).
- [11] <http://www.baskent.edu.tr/~hsyngdn/END407/END407%20Ders1.pdf>.
- [12] Toth, P., Vigo, D., The Vehicle Routing Problem, SIAM, (2001), s.1-21.
- [13] Güngör İ., Ergülen A., “Bulanık Araç Rotalama Problemlerine Bir Model Önerisi ve Bir Uygulama”, *Yönetim ve Ekonomi*, Celal Bayar Üniversitesi İ.İ.B.F. Manisa, 13, 1, (2006).
- [14] Gilbert Laporte, Metaheuristics for the Vehicle Routing Problem: Fifteen Years of Research, <http://neumann.hec.ca/chairedistributique/metaheuristics.pdf>.

- [15] Zheng, W., Xiang-pei H., Yang C., Wen-li L., “Real-time Modeling Method for Vehicle Routing Problems”, International Conference on, Harbin, P.R.China, ICMSE, (2007).
- [16] Laporte, G., “The Vehicle Routing Problem: An overview of exact and approximate algorithms”, *European Journal of Operational Research*, 59, (1992), 345-358.
- [17] Toth, P., “Optimization engineering techniques for the exact solution of NP-hard combinatorial optimization problems”, *European Journal of Operational Research*, 125, (2000), 222-238.
- [18] Solution Techniques for VRP, by Wolfgang Garn, Last update: 20.07.2007, <http://osiris.tuwien.ac.at/~wgarn/VehicleRouting/neo/algorithms/Algorithms.html>.
- [19] M.A., Tokaylı, Zaman Pencereli Araç Rotalama Problemi İçin Bir Karar Destek Sistemi, Yüksek Lisans Tezi, Endüstri Mühendisliği, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara, (2005).
- [20] Cordeau, J.F., Gendreau, M., Laporte, G., A Tabu Search Heuristic for Periodic and Multi-Depot Vehicle Routing Problems, *Networks*, 30, (1997), 105-119.
- [21] Lim, A., Wang, F., “Multi-Depot Vehicle Routing Problem: A One-Stage Approach”, *Automation Science and engineering, IEEE Transactions on*, (2005).
- [22] Braysy, O., Gendreau, M., “Vehicle Routing Problem with Time Windows, Part I: Route Construction and Local Search Algorithms”, *Transportation Science*, 39, 1, (2005), 104-118.
- [23] Braysy, O., Gendreau, M., “Route Construction and Local Search Algorithms for the Vehicle Routing Problem with Time Windows”, Sintef Report, STF42 A01024, (2001).
- [24] Lin,C.K.Y., “A cooperative strategy for a vehicle routing problem with pickup and delivery time Windows”, *Computers & Industrial Engineering*, 55, (2008), 766–782.
- [25] Desrosiers, J., Soumis, F., Desrochers, M., “Routing with time windows by column generation”, *Networks*, Vol. 14, No. 4. (1984), 545-565.



- [26] Desrochers, M., Desrosiers, J., Solomon, M., “A New Optimization Algorithm for the Vehicle Routing Problem with Time Windows”, *Operations Research*, 40, 2, (1992), 342-354.
- [27] Forbes, M. A., Holt, J. N., Watts, A. M., “An exact algorithm for multiple depot bus scheduling”, *European Journal of Operational Research*, 72, 1, ( 1994), 115-124.
- [28] Hadjiconstantinou, E., Christofides, N., Mingozzi, A., “A new exact algorithm for the vehicle muting problem based on q-paths and k-shortest paths relaxations” , *Annals of Operations Research*, 61, (1995), 21-43.
- [29] Desaulniers, G., Lavigne, J., Soumis, F., “Multi-depot vehicle scheduling problems with time windows and waiting costs”, *European Journal of Operational Research*, 111, 3, (1998), 479-49.
- [30] Bard, J. F., Kontoravdis, G., Yu, G., “A Branch-and-Cut Procedure for the Vehicle Routing Problem with Time Windows”, *Transportation Science*, 36, 2, (2002), 250-269.
- [31] Irnich S., Villeneuve, D., “The Shortest-Path Problem with Resource Constraints and  $k$ -Cycle Elimination for  $k \geq 3$ ”, *INFORMS, Journal on Computing*, 18, 3, (2005), 391-406.
- [32] Azi, N., Gendreau, M., Potvin, J.Y., “An exact algorithm for a single-vehicle routing problem with time windows and multiple routes[J]” *European Journal of Operational Research*, 178, 3, (2007), 755-766.
- [33] Chabrier, A., “Vehicle routing problem with elementary shortest path based column generation”, *Computers And Operations Research*, 33, 10, (2006), 2972-2990.
- [34] Mourgaya, M., Vanderbeck, F., “Column generation based heuristic for tactical planning in multi-period vehicle routing”, *European Journal of Operational Research*, 183, (2007), 1028–1041.
- [35] Oukil, A., Ben Amor, H., Desrosiers, J., El Gueddari, H., “Stabilized column generation for highly degenerate multiple-depot vehicle scheduling problems”, *Computers & Operations Research*, 34, 3, (2007), 817-834.
- [36] Moccia, L., Cordeau, J.-F., Laporte, G., Ropke, S., Valentini, M. P., “Modeling and solving a multimodal routing problem with timetables and time Windows”, *Networks*, (2008).

- [37] Qureshi, A.G., Taniguchi, E., Yamada, T., “An exact solution approach for vehicle routing and scheduling problems with soft time windows”, *Transportation Research Part E: Logistics and Transportation Review*, Article in Press, Corrected Proof, (2009).
- [38] Azi, N., Gendreau, M., Potvin, J.-Y., “An Exact Algorithm for a Vehicle Routing Problem with Time Windows and Multiple Use of Vehicles”, *European Journal of Operational Research*, In Press, Accepted Manuscript, (2009).
- [39] Brønmo, G., Nygreen, B., Lysgaard, J., “Column generation approaches to ship scheduling with flexible cargo sizes”, *European Journal of Operational Research*, In Press, Corrected Proof, (2009).
- [40] Shang, J. S., & Cuff, C. K., “Multicriteria pickup and delivery problem with transfer opportunity”, *Computers & Industrial Engineering*, 30(4), (1996), 631–645.
- [41] Langevin, A., & Soumis, F., “Design of multiple-vehicle delivery tours satisfying time constraints.”, *Transportation Research Part B*, 23(2), (1989), 123–138.
- [42] Mitrovic-Minic, S., Krishnamurti, R., & Laporte, G., “Double-horizon based heuristics for the dynamic pickup and delivery problem with time windows”, *Transportation Research Part B*, 38(8), (2004), 669–685.
- [43] Mitrovic-Minic, S., & Laporte, G., “Waiting strategies for the dynamic pickup and delivery problem with time Windows”, *Transportation Research Part B*, 38(7), (2004), 635–655.
- [44] <http://w.cba.neu.edu/~msolomon/r1r2solu.htm>.