

**YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**STOKASTİK ARAÇ ROTALAMA ALGORİTMALARININ
KARŞILAŞTIRMALI İNCELENMESİ**

Elektrik Mühendisi Ulaş DARCAN

**F. B. E. Endüstri Mühendisliği Anabilim Dalı Sistem Mühendisliği Programında
Hazırlanan**

YÜKSEK LİSANS TEZİ

Tez Danışmanı : Yrd. Doç. Dr. Tufan DEMİREL

İSTANBUL, 2007

İÇİNDEKİLER

	Sayfa
SİMGE LİSTESİ	iv
KISALTMA LİSTESİ	v
ŞEKİL LİSTESİ	vi
ÇİZELGE LİSTESİ	vii
1. GİRİŞ	1
2. STOKASTİK ARAÇ ROTALAMA PROBLEMİ	3
2.1 Postacı ve İlgili Yay Rotalama Problemleri	3
2.1.1 Giriş ve Örnekler	3
2.1.1.1 Çinli Postacı Problemi	3
2.1.1.2 Kapasiteli Çinli Postacı Problemi	4
2.1.1.3 Kapasiteli Yay Rotalama Problemleri	4
2.1.2 Euler Turları	4
2.1.2.1 Euler Turlarının Oluşturulması	6
2.1.3 Yönlendirilmemiş Grafikler İçin Postacı Problemi	9
2.1.4 Yönlendirilmiş Grafikler İçin Postacı Yöntemi	14
2.1.5 Karma Grafikler İçin Postacı Problemi	25
2.1.5.1 Karma Postacı Algoritması	28
2.2 Gezgin Satıcı ve İlgili Nokta Rotalama Problemleri	34
2.2.1 Giriş ve Örnekler	34
2.2.1.1 Toplama ve Dağıtım Problemleri	35
2.2.1.2 Bilgisayar Kablolama (Lenstra ve Rinnooy Kan, 1975)	35
2.2.1.3 Devre Levhası Delimi (Magirou, 1986)	35
2.2.1.4 Bir Ambarda Sipariş Toplama (Ratliff ve Rosenthal, 1983)	36
2.2.1.5 İş Atölyelerinde Sıralama	37
2.2.2 Gezgin Satıcı Probleminin Temel Nitelikleri	38
2.2.2.1 Bir Hamilton Döngüsünün Varlığı	40
2.2.3 GSP İçin Sezgisel Algoritmalar	47
2.2.3.1 Tur Yapım Sezgiselleri	47
2.2.3.2 Christofides'in Sezgiselleri	49
2.2.3.3 Tur Geliştirme Sezgiselleri	53
2.3 Araç Rotalama Problemi	55
2.3.1 Temel Problem Tanımları	62
2.3.1.1 Kapasiteli ve Mesafe Kısıtlı ARP	63
2.3.1.2 Zaman Kısıtlı ARP	64
2.3.1.3 Geri Dönüslü ARP	64
2.3.1.4 Toplama ve Dağıtım ile ARP	65
2.3.2 Araç Rotalama Problemi İçin Tur Oluşturma ve İyileştirme Sezgiselleri	66
2.3.2.1 Tur Yapım Algoritmaları	66
2.3.2.2 Çok Rota İyileştirmeli Sezgisel Algoritmalar	68
2.4 Stokastik Araç Rotalama Problemi	70
2.4.1 Giriş	70

2.4.2	Stokastik Araç Rotalama	70
2.4.2.1	Stokastik Müşteri ve Talepli SARP	71
2.4.2.2	Stokastik Dolaşım ve Hizmet Süreli SARP	72
3.	SARP İÇİN MODELLER VE ÇÖZÜM YÖNTEMLERİ	75
3.1	Literatür Araştırması.....	75
3.2	Modeller.....	83
3.2.1	Gerçek Zamanlı Bilgiler Işığında Dinamik Araç Rotalama Problemi İçin İskelet Yapı ve Çözüm Yaklaşımı	84
3.2.2	Stokastik Talepli Araç Rotalama Problemi İçin Dağıtım Bölgelerinin Tasarımı İçin Model Formülasyonu	87
3.2.2.1	Beklenen Maliyetin Tahmini	87
3.2.2.2	Beklenen Başvuru Maliyetinin Tahmini.....	87
3.2.3	Bir Stokastik Araç Rotalama Problemine Sezgisel Bir Çözüm Yöntemi İçin Matematiksel Formülasyon	88
3.2.4	Stokastik Talepli ve Yumuşak Zaman Çerçevesi Araç Rotalama Problemine Bir Genetik Algoritma Çözümü İçin Model	92
3.3	Çözüm Yöntemleri	95
3.3.1	Araç Rotalama Problemleri İçin Yapay Tavlama Algoritması.....	95
3.3.2	Stokastik Talepli Araç Rotalama Problemi İçin Metasezgiseller	96
3.3.2.1	Yapay Tavlama (SA)	97
3.3.2.2	Tabu Araştırma (TS).....	97
3.3.2.3	Tekrarlanmış Lokal Araştırma (ILS)	98
3.3.2.4	Karınca Koloni Optimizasyonu (ACO).....	98
3.3.2.5	Evrimsel Algoritma (EA)	99
4.	STOKASTİK TALEP İÇİN SEZGİSEL BİR YÖNTEMİN GELİŞTİRİLMESİ VE UYGULANMASI	100
4.1	Uygulama.....	101
5.	SONUÇ.....	114

SİMGE LİSTESİ

C_0	Aracın başlangıç kapasitesi
C_i	i. müşteriye hizmet götürdükten sonra aracın kalan kapasite miktarı
d_i	i. müşteri (talep) noktasının olasılığa bağlı talep değeri
$D(i, j)$	i ve j talep noktaları arası mesafe
m	Araç sayısı
$s(i, j)$	i ve j talep noktalarının rota oluşturması ile sağlanan tasarruf miktarı
v_0	Merkez depo
v_i	i. müşteri (talep) noktası

KISALTMA LİSTESİ

AKARP	Asimetrik Kapasiteli Araç Rotalama Problemi
ARP	Araç Rotalama Problemi
ÇPP	Çinli Postacı Problemi
GSP	Gezgin Satıcı Problemi
KARP	Kapasiteli Araç Rotalama Problemi
KÇPP	Kapasiteli Çinli Postacı Problemi
KYRP	Kapasiteli Yay Rotalama Problemi
SARP	Stokastik Araç Rotalama Problemi
SDHSARP	Stokastik Dolaşım ve Hizmet Süreli Araç Rotalama Problemi
SDSARP	Stokastik Dolaşım Süreli Araç Rotalama Problemi
SDSGSP	Stokastik Hizmet Süreli Gezgin Satıcı Problemi
SHSARP	Stokastik Hizmet Süreli Araç Rotalama Problemi
SKARP	Simetrik Kapasiteli Araç Rotalama Problemi
SMARP	Stokastik Müşterili Araç Rotalama Problemi
SMTARP	Stokastik Müşteri ve Talepli Araç Rotalama Problemi
STARP	Stokastik Talepli Araç Rotalama Problemi

ŞEKİL LİSTESİ

Şekil 2.1	Euler turuna sahip bir grafik	5
Şekil 2.2	Euler turuna sahip olmayan bir grafik	6
Şekil 2.3	Euler turuna sahip olmayan bir grafik	7
Şekil 2.4	Yönlendirilmemiş bir grafik ve en kısa yol uzunluğu matrisi	11
Şekil 2.5	En kısa yol uzunluk matrisi.....	12
Şekil 2.6	G' grafiği	12
Şekil 2.7	G^* grafiği	13
Şekil 2.8	Postacı rotası mevcut değil	14
Şekil 2.9	Yönlendirilmiş bir grafikte postacı problemi.....	16
Şekil 2.10	Yönlendirilmiş ÇPP için ağ akış problemi.....	17
Şekil 2.11	G^* grafiği	17
Şekil 2.12	Örnek.3 için nakliye ağı.....	19
Şekil 2.13	Euler turuna sahip bir karma grafik.	27
Şekil 2.14	Euler turuna sahip olmayan bir karma grafik	27
Şekil 2.15	G_D grafiği	32
Şekil 2.16	G' grafiği	33
Şekil 2.17	Ambar koridor konfigürasyonu.....	36
Şekil 2.18	Sipariş toplanmasının grafik sunumu.....	37
Şekil 2.19	Optimum satıcı rotası.....	39
Şekil 2.20	Grafik ve onun son derece bağlantılı elemanları	41
Şekil 2.21	5 Noktalı Bir GSP'nin 2'li değişimleri	53
Şekil 2.22	Clark-Wright algoritması için başlangıç çözümü	59
Şekil 2.23	i ve j müşterilerinin birleştirilmesi.....	60
Şekil 2.24	ARP'nin temel problemleri ve ara bağlantıları.....	63
Şekil 2.25	Süpürme Algoritması Adımları.....	68
Şekil 2.26	Yol Değişimi Yöntemi	69
Şekil 2.27	Nokta Değişimi Yöntemi	69
Şekil 2.28	Nokta Atama Yöntemi	70
Şekil 2.29	Stokastik bir kurulumda deterministik bir atama.....	73
Şekil 3.1	Değerlendirme Yapısının Konsept Yapısı. (Hu vd., 2002).....	86

ÇİZELGE LİSTESİ

Çizelge 2.1	Değişken başlangıç durumları.....	21
Çizelge 2.2	Yeni değişken değerleri	22
Çizelge 2.3	Güncellenen değişken değerleri.....	23
Çizelge 2.4	$r(i, j)$ değerleri	24
Çizelge 2.5	2-li değişimler ve maliyetleri	54
Çizelge 2.6	Yeni çözüm için 2-li değişimler.....	55
Çizelge 2.7	Müşteriler ve talep değerleri	60
Çizelge 2.8	Müşteriler ve depo arası dolaşım süreleri	61
Çizelge 4.1	Müşteriler ve merkez depo arası mesafeler	101
Çizelge 4.2	Her bir müşteri için olasılığa bağlı talep değerleri.....	101
Çizelge 4.3	Her bir ikili karşılaştırma sonucu oluşan tasarruf değerleri.....	104
Çizelge 4.4	Her bir müşteri için olasılığa bağlı talep değerleri.....	106
Çizelge 4.5	Müşteriler ve merkez depo arası mesafeler	107
Çizelge 4.6	Her bir ikili karşılaştırma sonucu oluşan tasarruf değerleri.....	112

ÖNSÖZ

Günümüzün küresel dünyasında yeni teknolojik gelişmeler ve yapılan çalışmalar göstermektedir ki üretim hızı ve kalitesine bağlı olarak ürünlerin kullanıcıya ulaşması ve bu yapılırken oluşacak maliyet ve kullanılan zamanın en aza indirgenmesi de önem kazanmıştır.

Müşteri memnuniyetini üst seviyede tutmak ve aynı zamanda taşıma maliyetini en aza indirmek için yıllardır yapılan çalışmalar Araç Rotalama Problemi (ARP) adı altında toplanmaktadır. ARP ile, müşterinin varlığı, kapasite, zaman, mesafe gibi bir takım kısıtlar çerçevesinde araçların müşterilere atanması gerçekleştirilerek lojistik maliyetlerini önemli ölçüde azaltmak ve buna bağlı olarak da firmalara rekabet üstünlüğü getirmek amaçlanmaktadır.

Bu çalışmada, araçların minimum maliyetle rotalanması için yapılan çalışmalar incelenmiş ve Stokastik Araç Rotalama kavramı incelenerek bununla ilgili bir sezgisel çözüm algoritması ve örnek model sunulmuştur.

Tezimi hazırlamamda bana yardımcı olan ve desteklerini esirgemeyen danışman hocam Yrd. Doç. Dr. Tufan Demirel'e, Yrd. Doç. Dr. Semih Önüt'e, kaynak bulmam konusundaki yardımları dolayısıyla değerli arkadaşım End. Müh. Kaya Balta'ya ve her türlü destekleri için aileme teşekkür ederim.

İstanbul, 2007

Ulaş Darcan

ÖZET

Bu çalışmada, öncelikle Araç Rotalama Probleminin (ARP) çözümüne öncülük edebilecek, taşıma ağları ve grafikler için optimizasyon algoritmaları ve buna bağlı olarak Postacı Problemi ve Gezgin Satıcı Problemi üzerinde durulmuştur. Daha sonra Araç Rotalama Problemi için temel bilgiler verilerek başta kapasite, mesafe ve zaman kısıtı olmak üzere problem türleri üzerinde durulmuştur. ARP için tur yapım ve tur iyileştirme sezgiselleri verildikten sonra ARP'nin, bazı değişkenlerin önceden bilinemez olduğu durumlarda ortaya çıkan bir alt türü olan Stokastik Araç Rotalama Problemi (SARP) üzerinde durulmuş ve çeşitli bilinmezlik durumları için SARP türleri incelenmiştir. Daha sonra bir literatür araştırması ile SARP üzerine yapılan çalışmalar verilmiş ve bazı çalışmalardan örnek modeller ve çözüm yöntemleri üzerinde durulmuştur. Son olarak da sezgisel bir çözüm algoritması oluşturularak kurulan bir model uygulanmış ve sonuçlar karşılaştırılmıştır.

Anahtar Kelimeler: Postacı Problemi, Gezgin Satıcı Problemi, Araç Rotalama Problemi, Stokastik Araç Rotalama Problemi, Sezgisel Algoritma.

ABSTRACT

In this study, first of all optimization algorithms for transportation networks and graphs, which are Postman Problem and Traveling Salesman Problem for this study, which can lead us to solve the VRP are studied. Then basic knowledge for VRPs is given and problem types, such as capacity, distance and time restricted VRP are studied. Also tour construction and tour improvement heuristics for VRP are given and Stochastic Vehicle Routing Problem (SVRP), which means some of the variables of the VRP are uncertain, is considered and some sort of uncertainty types for the SVRP are studied. Furthermore a literature research and studies on SVRP are given and some models and solutions methods from these studies are considered. Lastly an heuristic solution algorithm is proposed and performed on a model and findings are compared.

Keywords: Postman Problem, Traveling Salesman Problem, Vehicle Routing Problem, Stochastic Vehicle Routing Problem, Heuristic Algorithm.

1. GİRİŞ

Global dünyada gelişen üretim teknolojileri sonucu artan ürün kalitesi ve üretim hızına bağlı olarak firmalar; pazarda kendilerine yer edinebilmek ve rekabet edebilmek için hızlı ve kaliteli üretime ayak uydurabilecek şekilde, ürün veya hizmetlerin talep noktalarına ulaştırılmasını en hızlı şekilde ve en az maliyet ile gerçekleştirmek zorundadır. Günümüzün küreselleşen dünyasında artık sınır tanımayan dağıtım kanallarını en uygun şekilde oluşturmak ve hem müşteri memnuniyetini üst seviyelerde tutmak hem de en hızlı ve ucuz şekilde ürün ve hizmet dağıtımını gerçekleştirmek, pazarda lider konuma gelebilmek için gerekli şartlardan biridir.

Müşteri, mesafe, zaman, kapasite gibi kısıtlar da göz önüne alınarak oluşturulacak etkin rota planları, organizasyonların işlemesi ve müşteri memnuniyeti açısından son derece önemlidir. Değişen koşullara (talepteki değişiklikler, trafik koşulları, kapasite aşılması gibi) ayak uydurabilen ve hızlı bir şekilde optimize edilebilen rota planlarının oluşturulması işletme açısından önem teşkil etmektedir.

Bir araç rotalama probleminin (ARP) çözümü, tüm müşteri gereksinimlerini yerine getirecek, tüm operasyonel kısıtları sağlayacak ve toplam taşıma maliyetini minimize edecek şekilde, her biri işlemine depodan başlayıp depoda bitirecek araçlar tarafından gerçekleştirilecek rota kümelerinin belirlenmesini gerektirir. Ürünlerin taşınması için kullanılan yol ağı, genellikle çizgileri yol bölümlerini ifade eden ve köşe noktaları da yol eklemlerine, depoya ve müşteri noktalarına karşılık gelen bir grafik üzerinden tanımlanır. Çizgiler (ve sonuçta ona karşılık gelen grafikler), sırasıyla tek bir yönde gitmeleri (örneğin tek yönlü sokaklar ya da çevre yolu gibi) ya da her iki yönde de gitmelerine bağlı olarak yönlendirilmiş veya yönlendirilmemiş olabilirler. Her bir çizgi, araç tipine ya da dolaşım periyoduna bağlı olarak genellikle onun uzunluğunu ve bir dolaşım süresini gösteren bir maliyetle ilişkilidir.

Araç Rotalama Problemi, $V = \{v_1, v_2, \dots, v_n\}$ noktalar kümesi ve $A \subseteq \{(v_i, v_j) : i \neq j, v_i, v_j \in V\}$ yay kümesi olmak üzere verilen bir $G = (V, A)$ grafiği ile tanımlanır. Yayların depoda başlayıp biten döngüsel bir bağlantısından oluşan optimal bir rota kümesi noktalarda verilen bir müşteri kümesine hizmet vermek için seçilir. Bu problem ilk olarak 1959'da Dantzig ve Ramser tarafından istasyonlara benzinin dağıtımı ile ilgili bir gerçek uygulamayı çözmek için ortaya atılmıştır.

ARP için çözüm yöntemleri iki ana başlıkta incelenebilir: Kesin çözüm yöntemleri ve

yaklaşık çözüm yöntemleri. Kesin çözüm yöntemleri hesaplama açısından zor ancak iyi sonuç veren yöntemlerdir. Bununla beraber problem boyutu büyüdükçe kesin çözüm yöntemlerinin uygulanması zorlaşır. Büyük boyutlu problemlerde daha az işlem yapılarak ve daha kısa sürede optimale yakın sonuç veren yaklaşık yöntemler kullanılması tercih edilir. Bunlar sezgisel yöntemler olarak adlandırılır.

Klasik Araç Rotalama Problemi (ARP), önceden tanımlanan bir grafik üzerindeki müşteri kümesine hizmet vermek için bir araç filosunun kullandığı optimal rota kümesini saptar; hedefi toplam dolaşım maliyetini (dolaşım süreleri veya mesafe ile orantılı olarak) ve operasyon maliyetini (kullanılan araç sayısı ile orantılı olarak) minimize etmektir. Stokastik Araç Rotalama Problemi (SARP), ARP'nin bazı parametreleri rasgele olduğu zaman ortaya çıkar. Yaygın örnekler belirsiz müşteriler ve talepler ile stokastik dolaşım veya hizmet süreleridir.

Stokastik Araç Rotalama Problemi, problemdeki belirsizliğin ne olduğuna göre (bu müşteri, talep, dolaşım süresi (mesafe) veya hizmet süresi olabilir), modelleme yöntemine göre ve çözüm yöntemlerine göre (kesin ve yaklaşık) sınıflandırılabilir.

Bu çalışmada ikinci bölümde ilk olarak, Araç Rotalama Probleminin (ARP) çözümüne öncülük edebilecek, taşıma ağları ve grafikler için optimizasyon algoritmaları ve buna bağlı olarak Postacı Problemi ve Gezgin Satıcı Problemi üzerinde durulmuştur. Daha sonra Araç Rotalama Problemi ana hatlarıyla tanımlanmış ve temel problem türlerinden bahsedilmiştir. ARP için tır oluşturma ve tur iyileştirme sezgiselleri verildikten sonra Stokastik Araç Rotalama Problemine geçilmiş ve tanımı ve problem türleri üzerinde durulmuştur.

Üçüncü bölümde Stokastik Araç Rotalama Problemi üzerine yapılan çalışmalardan bahsedilmiş ve daha sonra bu çalışmalar içerisinde örnek modeller ve çözüm yöntemleri incelenmiştir.

Dördüncü bölümde stokastik talep için sezgisel bir algoritma önerilmiş ve problemin genel bir tanımı yapılmıştır.

Beşinci bölümde iki adet farklı boyuttaki uygulama üzerinde önerilen sezgisel algoritma uygulanarak optimale en yakın sonuçlar bulunmaya çalışılmıştır.

Son bölümde ise yapılan çalışma özetlenerek sonuç ve değerlendirmelere yer verilmiştir.

2. STOKASTİK ARAÇ ROTALAMA PROBLEMİ

Bu bölümde öncelikli olarak Araç Rotalama Probleminin çözümünde kullanılan, taşıma ağları ve grafikler için optimizasyon algoritmaları ve buna bağlı olarak Postacı Problemi ve Gezgin Satıcı Problemi üzerinde durulacaktır. Daha sonra Araç Rotalama Probleminin ve çözüm algoritmalarının temelleri işlenecek ve son olarak da Stokastik Araç Rotalama Problemi üzerinde durulacaktır.

2.1 Postacı ve İlgili Yay Rotalama Problemleri

Öncelikle postacı ve ilgili yay rotalama problemleri üzerinde duralım. Bu bölümde Postacı Probleminin tanımı, çeşitleri ve çözüm yöntemlerinden bahsedilmektedir.

2.1.1 Giriş ve Örnekler

Pek çok rotalama problemi, bir grafikteki bir grup yay boyunca kat eden yollar veya döngüler bulmayı içerir. Genelde, bu tür problemlere yay rotalama problemleri denir. Yay rotalama problemlerinin pek çok pratik uygulaması vardır. Aşağıdaki örnekler yay rotalama problemlerinin ana sınıflarını tanımlamaktadır.

2.1.1.1 Çinli Postacı Problemi

Bir posta taşıyıcısı, rotasına başlamadan önce, postaneden mektubu almalı, onu rotasındaki rota boyunca her bloğa dağıtmalı ve sonunda tekrar postaneye dönmelidir. İş daha kolay ve daha verimli yapmak için her postacı, rotasını mümkün olan en az yürümeyle bitirmek isteyecektir. Bu nedenle, problem, en kısa toplam sürede atanmış tüm sokakların nasıl dolaşılacağını ve başlangıç noktasına nasıl dönüleceğini kararlaştırmaktır. Eğer, her bir kenarın mektubun dağıtılması gerektiği sokağı ve her bir noktanın da kavşağı (kesişme noktası) belirten bir $G = (X, E)$ grafiği oluşturulursa, bu problem G 'deki, minimum toplam sürede her bir noktadan en azından bir kez geçen bir döngüyü bulmakla eşdeğerdir.

Bu problem ilk kez bir Çinli matematikçi, Kwan Mei-Ko tarafından bulunmuştur ve Çinli Postacı Problemi (ÇPP) olarak bilinir. Pek çok problem ÇPP olarak modellenebilir. Örneğin, sokağı süpürenlerin, kar temizleme araçlarının, eyaletler arası çim biçicilerin, devriye gezen ekip otolarının, elektrik hatlarının kontrolcülerinin veya bir fabrikadaki otomatik kılavuzlu araçların rotalanması, grafiğin kenarlarını kapsayan minimum maliyetli yolların bulunmasını içerir. Klasik Çinli Postacı Problemi grafikteki tüm kenarların yönlendirilmemiş olduğunu

varsayar. Bazı problemler, tüm kenarlarının (yönlendirilmiş postacı problemi) veya sadece bazı kenarlarının (karışık postacı problemi) yönlendirilmiş olduğu grafikler içerir (Evans ve Minieka, 1992).

2.1.1.2 Kapasiteli Çinli Postacı Problemi

Kapasiteli Çinli Postacı Problemi (KÇPP), ilgili her bir yayın pozitif bir talebinin olduğu ve rotalanan araçların sonlu bir kapasiteye sahip olduğu durumda ortaya çıkar. Örnek olarak, yol tuzlamayı içeren uygulamalarda, kamyonlar yalnızca belli bir maksimum değerde tuz taşıyabilirler. Örneğin 10 ton olsun. Tuz yayma oranı genellikle sabittir, o da 1,5 km başına 250 kg olsun. Bu nedenle, her bir yol dilimindeki talep, yol diliminin mesafesinin tuz yayma oranı kadar katı olarak hesaplanabilir. Bir kamyon limitli kapasitesinden ötürü bir bölgedeki tüm yollara hizmet veremeyebilir. KÇPP tek bir depodan rotadaki tüm noktalara minimum maliyetle, her bir rotada toplam talebin araç kapasitesini aşmama kısıtına da bağlı kalarak rota kümeleri bulur.

2.1.1.3 Kapasiteli Yay Rotalama Problemleri

KÇPP ‘nin en pratik genellemeleri kapasiteli yay rotalama problemleridir (KYRP). Bir KYRP’de, bazı talepler sıfır olabilir. Bu bazı yol tuzlama uygulamalarında görülen bir durumdur. Bu yüzden, problemi tanımlayan grafik çok fazla sayıda kenar içeriyorsa onların yalnızca bir alt kümesi hizmet görmelidir. Diğer kenarlar hizmet gerektiren kenarlar geçiş için boş geçilen yol olarak kullanılabilir. KÇPP ve KYRP, NP-zor problemlerdir; bu nedenle bir çözüm elde etmek için genelde sezgisel prosedürlere başvurulur.

2.1.2 Euler Turları

Matematikçi Leonhard Euler tarafından çözülen Königsberg Köprüsü problemi hatırlandığında, bu problemin amacı, bir grafikteki her bir kenarın (her bir köprünün) yalnızca bir kez dolaşılıp başlangıç noktasına dönülebilip dönülemeyeceğini bulmaktır. Königsberg Köprüsü problemi için Euler bunun imkansız olduğunu göstermiştir. Bir grafikteki herhangi bir kenarı yalnızca bir kez geçen döngüye bir Euler turu denir. Eğer bir postacı probleminde bir Euler turu mevcutsa, postacının bir noktaya varış sayısının, o noktadan ayrılış sayısına eşit olması gereklidir. Eğer postacı bir noktaya düşen herhangi bir kenarı tekrarlamazsa, o zaman bu noktaya gelen kenar sayısı çift sayı olmalı veya bu nokta çift dereceli olmalıdır. Bu sebeple Euler aşağıdaki sonucu ispatlamıştır:

Teorem 1. Yönlendirilmemiş bir grafik sadece tüm noktaları çift dereceli ise Euler ‘dir. Königsberg köprüsü problemi incelenirse grafiğin Euler olmadığı ve bunun sonucu olarak Euler turu içermediği görülmektedir. Şekil 2.1 bir Euler grafiği örneğini göstermektedir (Evans ve Minieka, 1992). S noktasından başlayan grafikte pek çok farklı Euler turu vardır. Örneğin, şu dört rotanın her biri bir Euler turudur:

Rota 1: (s, 1), (1, 2), (2, 3), (3, 4), (4, 2), (2, s)

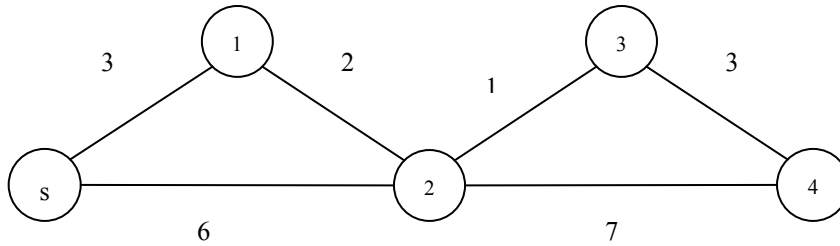
Rota 2: (s, 1), (1, 2), (2, 4), (4, 3), (3, 2), (2, s)

Rota 3: (s, 2), (2, 3), (3, 4), (4, 2), (2, 1), (1, s)

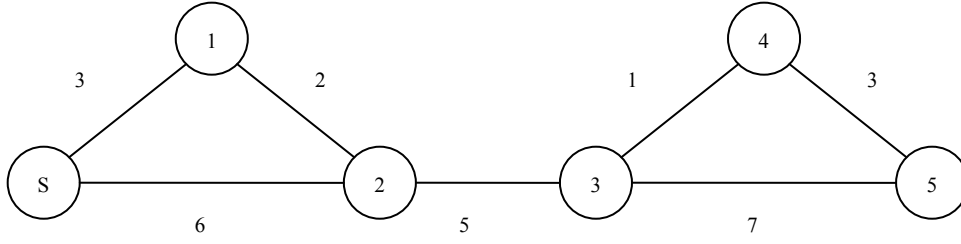
Rota 4: (s, 2), (2, 4), (4, 3), (3, 2), (2, 1), (1, s)

Bu dört rotanın her biri, her bir kenarı yalnızca bir kez geçmiştir; böylece her bir rotanın toplam uzunluğu $3 + 2 + 1 + 3 + 3 + 7 + 6 = 22$ olur. Postacı bundan daha iyisini yapamaz.

Eğer bir grafik, bir Euler turuna sahipse Çinli postacı problemi çözülür; yalnızca bazı Euler turları bulmak gereklidir. Eğer bir grafik hiç Euler turuna sahip değilse, o zaman en azından bir kenar birden fazla kez geçilmelidir. Şekil 2.2’de örneğin, postacının (2, 3) kenarını bir kez geçmesinin yolu yoktur. Euler turu mevcut değildir. Her bir kenarı en azından bir kez geçen en kısa tur (s, 1), (1, 2), (2, 3), (3, 4), (4, 5), (5, 3), (3, 2), (2, s) olur. Başka bir optimal çözüm yoktur. Turun toplam uzunluğu, tüm kenarların uzunluğu ($3 + 2 + 5 + 1 + 3 + 7 + 6$) ve buna ilaveten (5) kenarının boşa geçilmesi yani 32 ‘dir. Tüm kenar uzunlukları sabit olduğu için, Çinli postacı problemi grafikteki gerekli olan boşa geçmeleri minimize etmek olarak yorumlanabilir.



Şekil 2.1 Euler turuna sahip bir grafik.(Evans ve Minieka, 1992)



Şekil 2.2 Euler turuna sahip olmayan bir grafik. (Evans ve Minieka, 1992)

2.1.2.1 Euler Turlarının Oluşturulması

$G_1 = (X, E_1)$ bir Euler grafiği olsun. Bir Euler turu bulunması için basit bir algoritma aşağıdaki gibi gelişir:

Adım 1. Herhangi bir s noktasından başlanır ve bir C döngüsü oluşturulur. Bu, s noktasına bağlı herhangi bir (s, x) kenarı dolaşarak ve bu kenar “kullanıldı” olarak işaretlenerek yapılabilir. Daha sonra x noktasına bağlı herhangi bir kullanılmamış kenar dolaşılır. Bu kullanılmamış kenarların dolaşılması işlemi s noktasına geri dönülünceye kadar tekrarlanır. (Bu süreç s noktasına dönmelidir çünkü her nokta çift derecelidir ve bir noktaya her ziyaret bu noktaya bağlı çift sayıda kullanılmamış kenar bırakmaktadır. Bu nedenle, bir noktaya her girildiğinde bu noktadan ayrılacak bir kullanılmamış kenar mevcuttur.)

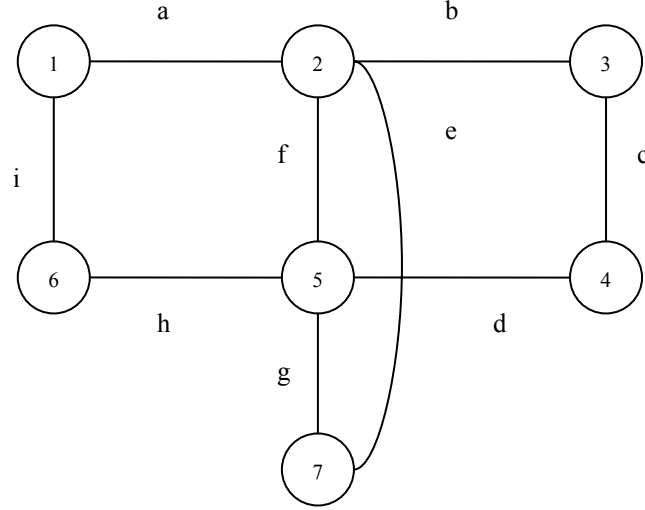
Adım 2. Eğer C , G ‘nin tüm kenarlarını içeriyorsa durulur. İçermiyorsa, C ‘nin tüm kenarlarının çıkartıldığı G' alt grafiği, C ‘nin her bir noktası çift sayıda kendisine gelen kenar içerdiği için Euler turu olmalıdır. G bağlantılı olduğundan C ile paylaşılan en az bir v noktası olmalıdır.

Adım 3. v noktasından başlanarak, G' içinde bir döngü oluşturulur, buna C' denir.

Adım 4. C ve C' döngüleri birbirine eklenir, birleşik döngü C olarak adlandırılır. Adım 2’ye geri dönülür (Evans ve Minieka, 1992).

Bu metod, Şekil 2.3’teki grafik ile gösterilmiştir. 1 noktasından başlanırsa, a, f, h ve i kenarlarını kapsayan C döngüsü oluşturulabilir. $v, 2$ noktası olsun. Grafiğin kalanı, b, c, d, g ve e kenarlarını içeren C' döngüsüne sahiptir. Döngüler bir araya şu şekilde getirilir. C ‘nin başlangıç noktasından başlanır ve ortak nokta v ‘ye ulaşıncaya kadar döngü etrafında ilerlenir. Bu noktada, v noktasına geri dönülünceye kadar C' döngüsü dolaşılır. Sonra v ‘den başlangıç noktasına C ‘de kalan yol boyunca devam edilir. Yukarıdaki örnekteki birleşik döngü a, b, c, d, g, e, f, h ve i kenarlarını kapsar ve bu G ‘de bir Euler turudur. Bu

algoritmadaki temel fikirler Edmonds ve Johnson (1973) tarafından şekillendirilmiştir. Onların uygulamaları, x noktasına k . kere ulaşıldığında bir sonraki noktayı ziyaret edecek bir $L(x_1), L(x_k), \dots, L(x_k)$ noktalar listesini sağlar (Evans ve Minieka, 1992).



Şekil 2.3 Euler turu örneği. (Evans ve Minieka, 1992)

Adım 1. Herhangi bir v_0 noktası seçilir. Tüm x noktaları için $v = v_0$ ve $k = k_0$ olsun. Tüm kenarlar “kullanılmamış” olarak etiketlenir.

Adım 2. v noktasına gelen rasgele bir kullanılmamış kenar seçilir. Bu kenar “kullanılmış” olarak işaretlenir. y , kenarın diğer ucundaki nokta olsun. $k_y = k_y + 1$ ve $L_y(k_y) = v$ durumuna getirilsin. Eğer y noktasına gelen, kullanılmamış kenar varsa Adım 3’e gidilir. Aksi halde, y noktası v_0 olmalıdır. Bu durumda Adım 4’e gidilir.

Adım 3. $v = y$ durumuna getirilir ve Adım 2’ye geri dönlür.

Adım 4. v_0 , kendisine bağlı en azından bir kullanılmış bir de kullanılmamış kenarı olan herhangi bir nokta olsun. $v = v_0$ durumuna getirilir ev Adım 2’ye dönlür. Eğer böyle bir nokta yoksa Adım 5’e gidilir.

Adım 5. Turu oluşturmak için, orijinal v_0 noktasından başlanır. x noktasına ilk kez ulaşıldığında, oradan $L_x(k_x)$ noktasına gidilir. $k_x = k_x - 1$ durumuna getirilir ve devam edilir. Her seferinde x noktasından $L_x(k_x)$ noktasına gidilir (Evans ve Minieka, 1992).

Örnek 1:

Bu algoritmayı Şekil 2.3’ü kullanarak gösterelim.

Adım 1. $v_0 = v = v$. $x = 1, \dots, 7$ için $k_x = 0$.

Adım 2. a kenarını seçelim; $y = 2$; $k_2 = 1$; $L_2(1) = 1$. a kenarını “kullanılmış” olarak işaretleyelim.

Adım 3. $v = 2$.

Adım 2. b kenarını seçelim; $y = 3$; $k_3 = 1$; $L_3(1) = 2$. b kenarını “kullanılmış” olarak işaretleyelim.

Adım 3. $v = 3$.

Adım 2. c kenarını seçelim; $y = 4$; $k_4 = 1$; $L_4(1) = 3$. c kenarını “kullanılmış” olarak işaretleyelim.

Adım 3. $v = 4$.

Adım 2. d kenarını seçelim; $y = 5$; $k_5 = 1$; $L_5(1) = 4$. d kenarını “kullanılmış” olarak işaretleyelim.

Adım 3. $v = 5$.

Adım 2. h kenarını seçelim; $y = 6$; $k_6 = 1$; $L_6(1) = 5$. h kenarını “kullanılmış” olarak işaretleyelim.

Adım 3. $v = 1$.

Adım 2. i kenarını seçelim; $y = 1$; $k_1 = 1$; $L_1(1) = 6$. i kenarını “kullanılmış” olarak işaretleyelim. $y = v_0$. Adım 4’e gidilir.

Adım 4. $v_0 = 2 = v$ durumuna getirilir. Adım 2’ye dönülür.

Adım 2. e kenarı seçilir; $y = 7$; $k_7 = 1$; $L_7(1) = 2$. e kenarı “kullanılmış” olarak işaretlenir.

Adım 3. $v = 7$.

Adım 2. g kenarı seçilir; $y = 5$; $k_5 = 2$; $L_5(2) = 7$. g kenarı “kullanılmış” olarak işaretlenir.

Adım 3. $v = 5$.

Adım 2. f kenarı seçilir; $y = 2$; $k_2 = 2$; $L_2(2) = 5$. f kenarı “kullanılmış” olarak işaretlenir.

Adım 4’e gidilir ve geriye nokta kalmadığı belirlenir. Adım 5’e gidilir.

Adım 5. $v_0 = 1$ noktasında başlanır. $L_2(2) = 5$ 'e gidilir; $k_2 = 1$ durumuna getirilir. Daha sonra, $L_5(2) = 7$ 'ye gidilir; $k_5 = 1$ durumuna getirilir. Ziyaret edilmek için kalan noktalar sırasıyla 2, 1, 6, 5, 4, 3, 2'dir.

2.1.3 Yönlendirilmemiş Grafikler İçin Postacı Problemi

Bu bölümde, herhangi bir yönlendirilmemiş $G = (X, E)$ grafiği için postacı probleminin nasıl çözüleceği anlatılmaktadır. Eğer G bir Euler grafiği ise herhangi bir Euler turu problemi çözer ve hiç boş geçmeye gerek kalmaz. Eğer G bir Euler grafiği değilse, gereken boş geçme miktarı azaltılmaya çalışılır. $a(i, j)$, G 'de (i, j) kenarının uzunluğu olsun.

Herhangi bir postacı rotasında, postacının bir noktaya giriş sayısı, postacının o noktadan ayrılış sayısına eşittir. Sonuç olarak, eğer x noktası çift dereceli değilse, x 'e bağlı en azından bir kenar postacı tarafından tekrar geçilmelidir.

$f(i, j)$, (i, j) kenarının postacı tarafından kaç kez tekrar edildiğini gösterebilir. (i, j) kenarı, postacı tarafından $f(i, j) + 1$ kez dolaşılır. Tabi ki, $f(i, j)$ negatif olmayan bir tamsayı olmalıdır. $f(i, j)$ 'nin, (i, j) kenarı boyunca dolaşım yönü hakkında bir bilgi içermediğine dikkat edilmelidir.

G grafiğindeki her bir (i, j) kenarının $f(i, j) + 1$ kadar kopyasını içeren yeni bir $G^* = (X, E^*)$ grafiği oluşturulduğunda görülmektedir ki, G^* grafiğinin bir Euler turu, G grafiğinde bir postacı rotasına karşılık gelmektedir.

Postacı aşağıdaki koşulları sağlamak için $f(i, j)$ değişkenleri için değerler seçer:

(a) G^* grafiği, bir çift grafiktir.

(b) $\sum a(i, j)f(i, j)$, tekrar edilen kenarların toplam uzunluğu, minimize edilir (Evans ve Minieka, 1992).

Eğer x noktası G grafiğinde bir tek dereceli nokta ise, x noktasına bağlı tek sayıda kenar postacı tarafından tekrar edilmelidir, böylece G^* grafiğinde x noktası çift dereceli olur. Benzer şekilde, eğer x noktası G grafiğinde çift dereceli bir noktaysa, x noktasına bağlı çift sayıda kenar (sıfır da çift sayıdır) postacı tarafından tekrarlanmalıdır. Böylece G^* grafiğinde x noktası çift derece alır.

Tek dereceli bir noktadan başlayan tekrar edilmiş kenarlardan oluşan bir yol mümkün olduğunca uzağa izlenirse, bu yolun başka bir tek dereceli noktada son bulması gereklidir. Bu

nedenle, yollarda tekrar edilen kenarların başlangıç ve bitiş noktaları tek dereceli noktalardır. Tabi ki, böyle bir yolun ara noktalarından biri çift dereceli nokta olabilir. Sonuç olarak, postacı, (a) hangi tek dereceli noktalar tekrar edilen kenarlardan oluşan bir yol tarafından birleştirileceğine ve (b) kesin olarak bu şekildeki her bir yolun düzenine karar vermek zorundadır.

Problemi çözmek için bir yöntem, tekrar edilen kenarlardan oluşan yollarla tek dereceli noktaları keyfi olarak birleştirmek ve Mei-Ko tarafından ispatlanan aşağıdaki teoremi kullanmaktır:

Teorem 2. Eğer ve sadece,

- (i) herhangi bir orijinal kenara birden daha fazla benzer (kopya, eş) kenar eklenmezse ve
- (ii) herhangi bir döngüye eklenen kenarların uzunluğu döngünün yarı uzunluğunu aşmazsa

postacı problemine uygun bir çözüm optimaldir (Evans ve Minieka, 1992).

İspat. Uygun bir çözümün (i) şartını bozduğunu varsayalım. Gereksiz olan iki eklenmiş kenar kaldırılabilir ve daha kısa bir toplam uzunluk ile yeni bir uygun çözüm elde edilebilir. Daha sonra, uygun bir çözümün (ii) şartını bozduğunu varsayalım. Eklenmiş kenarlar olmadan döngünün uzunluğu, döngünün uzunluğunun yarısından daha azdır. Döngüde eklenmiş bir kenar olan her bir yerden bir eklenmiş kenar çıkarılır ve olmayan her bir yere bir tane eklenir. Açık bir şekilde, döngüdeki eklenmiş kenarların toplam uzunluğu azalmıştır ve tüm noktalar çift dereceli olarak kalmıştır. Yeterlilik, aşağıdaki ispatı olmayan yardımcı önermeler sonucu ortaya çıkar:

Yardımcı Önerme 1. Eğer iki uygun çözüm (i) ve (ii) şartlarını sağlıyorsa, eklenen kenarlarının uzunlukları eşittir.

Yardımcı Önerme 2. Her zaman optimal çözümler vardır.

Teorem 2.'nin yapıcı ispatı, optimal bir çözüm bulunana kadar uygun çözümleri sırasıyla geliştiren bir yöntem sağlar. Tek sorun, (ii) koşulunda kontrol edilmesi gereken döngülerin sayısı grafiğin boyutunda üstel olarak artar. Bundan dolayı, bu algoritma polinomsal zamanda gerçekleştirilemez. Bununla beraber, diğer algoritmaları kullanan bir polinomsal algoritma probleme mükemmel bir çözüm sağlayabilir (Evans ve Minieka, 1992).

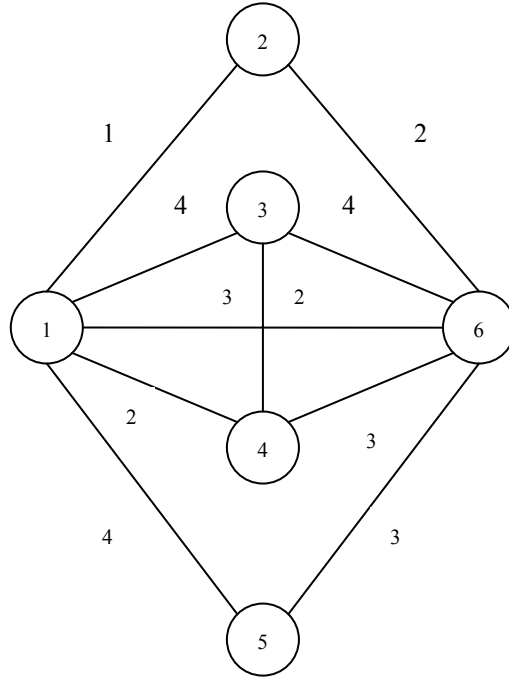
Floyd veya Dantzig en kısa yol algoritmalarını kullanarak postacı G grafiğindeki her bir tek dereceli nokta arasında en kısa yolu belirleyebilir (Evans ve Minieka, 1992).

Postacı tekrar edilen kenarlardan oluşan yollar tarafından hangi tek dereceli nokta çiftlerinin birleştirileceğini şu şekilde belirleyebilir: Nokta kümesi tamamen G grafiğindeki tek dereceli noktalardan oluşan ve kenar kümesi her bir nokta çiftini bağlayan bir kenarı içeren bir $G' = (X', E')$ grafiği oluşturulur.

Daha sonra postacı, maksimum ağırlık eşleşme algoritmasını kullanarak G' grafiği için bir maksimum ağırlık eşleşmesi bulmalıdır (Evans ve Minieka, 1992). G' grafiği çift sayıda noktalara sahip olduğundan ve G' grafiğindeki her bir nokta çifti bir kenar tarafından birleştirildiğinden, maksimum ağırlık eşleşmesi her bir noktayı kesinlikle bir kez kapsayacaktır. Bu eşleşme G grafiğindeki tek dereceli noktaları birlikte eşler. Bir eşleşmiş tek dereceli noktalar çiftine katılan en kısa yoldaki kenarlar postacı tarafından tekrar edilmelidir. Bu eşleşme maksimum toplam ağırlığa ulaştığında, son postacı rotası minimum toplam uzunlukta olmalıdır.

Böylece, yönlendirilmemiş bir grafik için postacı problemi Floyd veya Dantzig algoritmaları ve maksimum ağırlık eşleşmesi kullanılarak çözülebilir. Yeni bir algoritmaya ihtiyaç yoktur.

Örnek 2: Şekil 2.4.'teki yönlendirilmemiş grafik için optimal postacı rotasını bulalım. 1, 3, 4 ve 6 noktalarının tek dereceli olduğuna dikkat edilmelidir. Tüm tek dereceli nokta çiftleri arasında en kısa yolların uzunluğu Şekil 2.5'te gösterilmektedir (Evans ve Minieka, 1992).



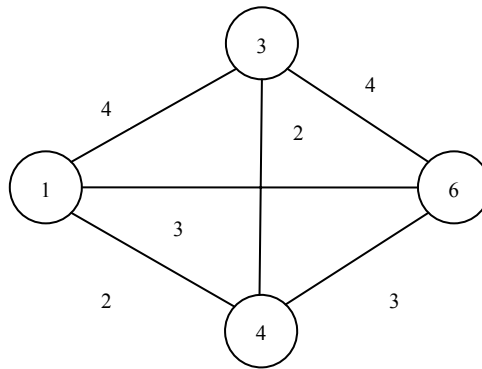
Şekil 2.4 Yönlendirilmemiş bir grafik ve en kısa yol uzunluğu matrisi (Evans ve Minieka, 1992)

	1	2	3	4	5	6
1	0	1	4	2	4	3
2	1	0	5	3	5	2
3	4	5	0	2	7	4
4	2	3	2	0	6	3
5	4	5	7	6	0	3
6	3	2	4	3	3	0

Şekil 2.5 En kısa yol uzunluk matrisi (Evans ve Minieka, 1992)

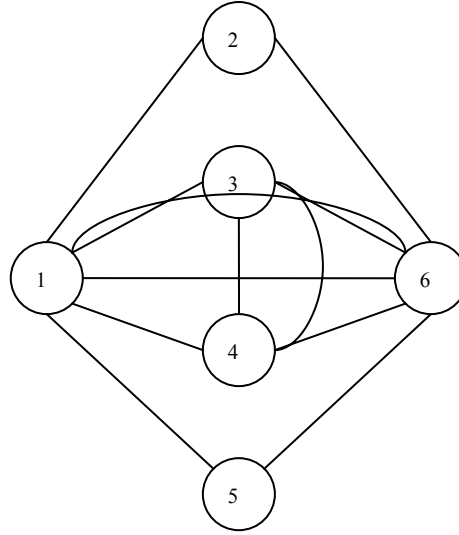
Şekil 2.6'da gösterilen G' grafiği oluşturulur. G' grafiğinin noktaları, G grafiğinin tek dereceli 1, 3, 4 ve 6 noktalarıdır. Mümkün olan tüm kenarlar G' grafiğinde mevcuttur. G' fazla noktaya sahip olmadığı için, G' grafiğindeki tüm kenarların minimum ağırlık eşleşmesi maksimum ağırlık eşleme algoritmasını kullanmak yerine numaralandırma yoluyla bulunabilir. Üç eşleşme mümkündür:

Eşleşme	Ağırlık
(1, 3), (4, 6)	$4 + 3 = 7$
(1, 4), (3, 6)	$2 + 4 = 6$
(1, 6), (3, 4)	$3 + 2 = 5$



Şekil 2.6 G' grafiği (Evans ve Minieka, 1992)

Sonuç olarak, minimum ağırlık eşleşmesi (1, 6), (3, 4) olur. Bu nedenle, postacı (1, 6) kenarını oluşturan 1'den 6'ya en kısa yolu ve (3, 4) kenarını oluşturan 3'ten 4'e en kısa yolu tekrarlamalıdır. Şekil 2.7 (1, 6) ve (3, 4) kenarlarının birer kez kopyalandığı G^* grafiğini göstermektedir. G^* grafiğindeki tüm noktalar çift derecelidir ve Şekil 2.4'teki orijinal grafik için optimal postacı rotası şekil 2.7'deki G^* grafiğinin bir Euler turuna karşılık gelmektedir. A durumunda çift dereceli grafikler için tanımlanan teknik G^* grafiğine uygulanabilir. Optimal rota, G^* grafiğindeki her bir kenar yalnızca bir kez ve G grafiğindeki her bir kenar en az bir kez dolaşmak üzere (1, 2), (2, 6), (6, 5), (5, 1), (1, 3), (3, 6), (6, 4), (4, 3), (3, 4), (4, 1), (1, 6), (6, 1) olur. G grafiğinde yalnızca (1, 6) ve (3, 4) kenarları tekrar edilir. Bu rotanın toplam uzunluğu 34 birimdir ve bu kenar uzunluklarının toplamından 5 birim fazladır.



Şekil 2.7 G^* grafiği (Evans ve Minieka, 1992)

Eğer maksimum ağırlık eşleme algoritması kullanılmış olsaydı, G^* grafiğindeki her bir kenar ağırlığı büyük bir sayının, M olsun, G 'de karşılık gelen iki bitiş noktası arasındaki bir en kısa yol uzunluğundan farkına eşitlenirdi. Böylece, eşleşme (1, 3), (4, 6), $(M-4) + (M-3) = 2M-7$ toplam ağırlığına sahip olurdu. (1, 4), (3, 6) eşleşmesi, $(M-2) + (M-4) = 2M-6$ toplam ağırlığına eşit olurdu. (1, 6), (3, 4) eşleşmesi, $(M-3) + (M-2) = 2M-5$ toplam ağırlığına sahip olurdu ve maksimum ağırlık eşleşmesi olarak seçilirdi.

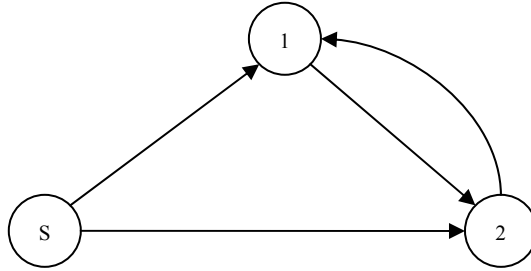
Büyük M değeri, tüm noktaları kapsayan bir minimum ağırlık eşleşmesi bulma problemini bir maksimum ağırlık eşleşmesi problemine dönüştüren bir araç olarak görülebilir (Evans ve Minieka, 1992).

2.1.4 Yönlendirilmiş Grafikler İçin Postacı Yöntemi

Bu bölümde yönlendirilmiş bir $G = (X, A)$ grafiği için postacı problemi incelenecektir. Yönlendirilmiş bir grafik, tüm sokakların tek yön olduğu bir fiziksel duruma karşılık gelir. Bir yayın yönü ilgili yolun hangi yönde geçileceğini belirtir.

Yönlendirilmiş bir grafikte x noktasına doğru yönlendirilmiş yayların sayısı x noktasının giriş derecesi olarak, x noktasından dışa doğru yönlendirilmiş yayların sayısı da x noktasının çıkış derecesi olarak tanımlanır (Evans ve Minieka, 1992). x noktasının giriş ve çıkış dereceleri sırasıyla $d^-(x)$ ve $d^+(x)$ olarak gösterilir. Eğer G grafiğindeki tüm x noktaları için giriş ve çıkış dereceleri eşitse G grafiğine simetrik denir.

Yönlendirilmemiş grafikler için postacı probleminden farklı olarak, postacı probleminin yönlendirilmiş grafik için hiçbir çözümü bulunmayabilir. Örneğin, Şekil 2.8'deki grafiği göz önüne alalım. Postacı 1 veya 2 noktasına vardığında s noktasına dönemez çünkü $\{1, 2\}$ kümesinden, bu kümede olmayan bir nokta için ayrılan bir yay yoktur. Genelde, kendi içinde bulunan bir noktadan kendinde bulunmayan bir noktaya hiçbir yay gitmeyen bir S noktalar kümesi varsa postacı probleminin çözümü yoktur. Eğer böyle bir S kümesi yoksa postacı için rotasını tamamlaması, ne kadar uzun sürerse sürsün, hiçbir yerde takılamayacağı için her zaman mümkündür.



Şekil 2.8 Postacı rotası mevcut değil (Evans ve Minieka, 1992)

Daha önce de olduğu gibi postacının bir noktaya giriş sayısı o noktadan ayrılma sayısına eşit olmalıdır. Sonuç olarak, eğer x noktasına, kendisinden çıkandan daha fazla yay giriyorsa [yani $d^-(x) > d^+(x)$] postacı x noktasından çıkan bazı yayları tekrar etmelidir. Benzer şekilde, eğer x noktasından kendisine girenden daha fazla yay çıkıyorsa [yani $d^+(x) > d^-(x)$] postacı x noktasına giren bazı yayları tekrar etmelidir. Bu nedenle, eğer bazı x noktaları için, $d^+(x) \neq d^-(x)$ ise hiçbir Euler turu mümkün değildir.

İki durum ayrı ayrı göz önünde bulundurulmalıdır:

Durum A: G grafiği simetriktir [tüm x 'ler için, $d^+(x) = d^-(x)$]

Durum B: G grafiği simetrik değildir.

Durum A. Eğer G grafiği simetrik ise, postacının rotasını hiçbir yayı tekrar etmeden gerçekleştirmesi mümkündür; şöyle ki, postacı problemi için optimal çözüm bir Euler turudur.

$G = (X, A)$ yönlendirilmiş grafiğindeki bir Euler turu, yönlendirilmemiş çift dereceli bir grafikte Euler turu bulunmasıyla benzer teknikler kullanılarak bulunabilir. Tek fark bir noktadan ayrılırken seçilen yaylar noktadan dışarı doğru yönlendirilmiş olmalıdır.

Durum B. Daha önce olduğu gibi, $f(i, j)$ postacının (i, j) yayını tekrarlama sayısını gösterebilir. Postacı $f(i, j)$ değişkenler için negatif olmayan tam sayılar seçmek ister ki,

$$\sum_j a(i, j) f(i, j) \quad (2.1)$$

tekrar edilen yayların toplam uzunluğu değerini minimize edebilsin. Her bir x noktasına aynı sayıda girip aynı sayıda çıkar yani,

$$d^-(i) + \sum_j f(j, i) = d^+(i) + \sum_j f(i, j) \quad (2.2)$$

(2.2) eşitliği tekrar yazılırsa şu sonucu verir:

$$\sum_j [f(i, j) - f(j, i)] = d^-(i) - d^+(i) \equiv D(i) \quad (2.3)$$

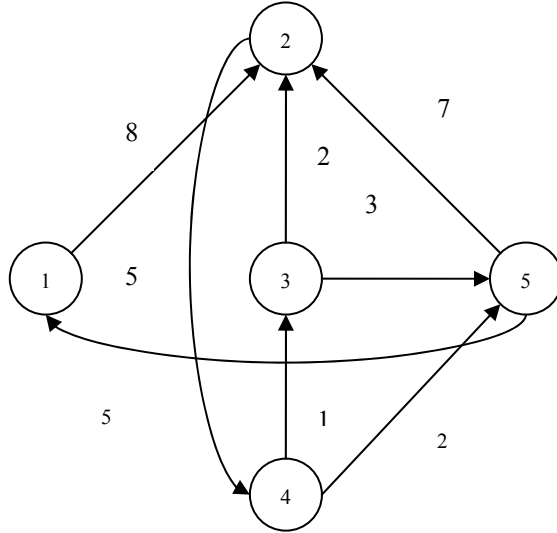
Bu nedenle, postacı (2.1) ifadesini, (2.3) eşitliği G grafiğindeki tüm i noktaları için sağlansın diye minimize etmek ister.

Bu minimizasyon problemi gerçekte yalnızca bir minimum maliyet akış problemidir. $D(i) < 0$ olan noktalar, talebi $-D(i)$ 'ye eşit olan hedeflerdir. $D(i) > 0$ olan noktalar, stok miktarı $D(i)$ 'ye eşit olan kaynaklardır. $D(i) = 0$ olan noktalar aktarma noktalarıdır. Tüm yay kapasiteleri sonsuzdur. Bu minimum maliyet akış problemi herhangi bir minimum maliyet akış algoritması kullanılarak çözülebilir.

(2.3) eşitliğindeki tüm sağ taraf değerleri tam sayı olduğu için minimum maliyet akış algoritması $f(i, j)$ için negatif olmayan tam sayı olan değerler üretecektir.

$f(i, j)$ değişkenleri için optimal tamsayı değerleri bulunduktan sonra, tüm $(i, j) \in A$ 'lar için

(i, j) yayının $f(i, j) + 1$ kopyası ile bir G^* grafiği yaratılır. (2.3) eşitliğine göre G grafiği simetriktir. A durumu için tanımlanan teknik şimdi G^* grafiğinin bir Euler turunu bulmak için uygulanabilir. G^* grafiğinin bir Euler turu, G grafiğindeki her bir yayı $(i, j)f(i, j) + 1$ kez dolaşan bir postacı rotasına karşılık gelir. $F(i, j)$ 'nin optimal değerleri (2.1) ifadesini minimize ettiğinden G^* grafiğindeki bu Euler turu G grafiğindeki optimal bir postacı turuna karşılık gelmelidir (Evans ve Minieka, 1992).



Şekil 2.9 Yönlendirilmiş bir grafikte postacı problemi (Evans ve Minieka, 1992)

Örnek 3.

Şekil 2.9'daki yönlendirilmiş grafik için optimal bir postacı rotası bulalım. Giriş ve çıkış dereceleri incelendiğinde şu sonuçlara ulaşılır:

$d^-(1) = 1 = d^+(1)$; 1 noktası bir aktarma noktasıdır.

$d^-(2) = 3 > d^+(2) = 1$; 2 noktası $3 - 1 = 2$ birim stok malzemeli kaynaktır.

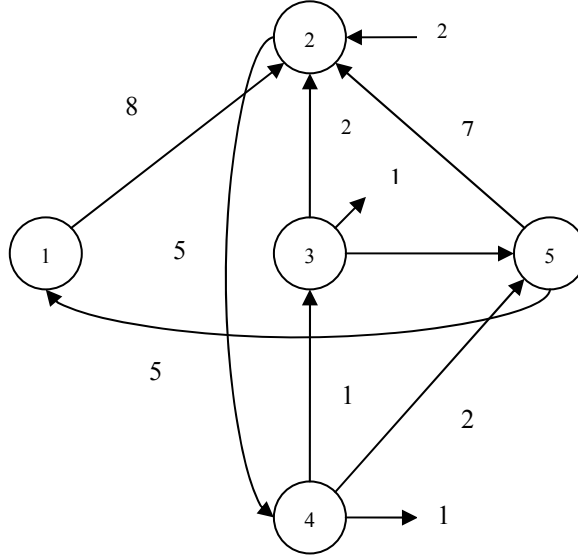
$d^-(3) = 1 < d^+(3) = 2$; 3 noktası $2 - 1 = 1$ bir talepli bir hedef noktasıdır.

$d^-(4) = 1 < d^+(4) = 2$; 4 noktası $2 - 1 = 1$ birim talepli bir hedef noktasıdır.

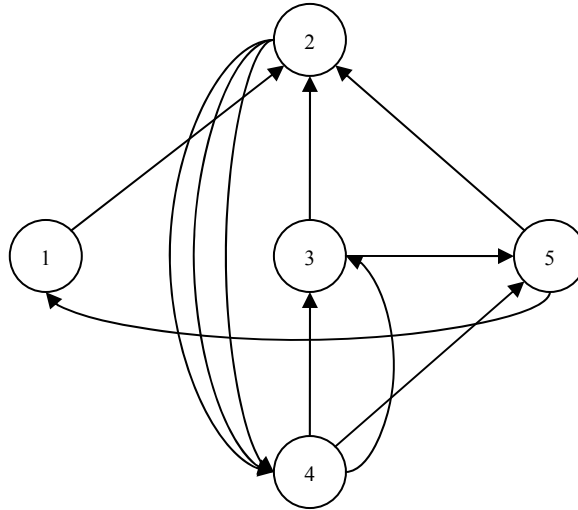
$d^-(5) = 2 = d^+(5)$; 5 noktası bir aktarma noktasıdır.

Şekil 2.10 sonuçta çıkan minimum maliyet akış problemini göstermektedir.

Böylece, bir simetrik grafiği yaratmak için 4 noktasından 3 noktasına bir yay ve 2 noktasından 4 noktasına 2 yay eklenir. Bu Şekil 2.11’de gösterilmektedir.



Şekil 2.10 Yönlendirilmiş ÇPP için ağ akış problemi. (Evans ve Minieka, 1992)



Şekil 2.11 G^* grafiği. (Evans ve Minieka, 1992)

Durum A'dan bilindiği üzere, G^* bir Euler turuna sahiptir. Örneğin, G^* grafiğinin bir Euler turu ve G grafiğinin bir optimal postacı rotası (1, 2), (2, 4), (4, 3), (3, 2), (2, 4), (4, 3), (3, 5), (5, 2), (2, 4), (4, 5), (5, 1)'dir. Bu turun toplam uzunluğu, 33 birim (grafikteki tüm yayların toplam uzunluğu) artı 11 birim (tekrar edilen tüm yayların toplam uzunluğu) olmak üzere 44

birimdir.

G^* grafiğini belirlemenin alternatif bir yolu Beltrami ve Bodin'e (1974) dayanır (Evans ve Minieka, 1992). Orijinal grafik üzerinde bir ağ akış problemi yaratmak yerine, problem bir ulaştırma problemine dönüştürülebilir. İlk olarak, giriş derecelerinin çıkış derecelerinden daha büyük olduğu i noktalarının yerleri belirlenir. Fark, $s(i)$, i noktasından dışa doğru yönlendirilmesi gereken ek yayların sayısını gösterir. Ayrıca, çıkış dereceleri giriş derecelerinden büyük olan tüm j noktalarının yerleri belirlenir. Bu farka $d(j)$ denir. Bu da j noktasına doğru yönlendirilmesi gereken ek yayların sayısını belirtir. Her bir i noktasından her bir j noktasına en kısa yollar bulunur ve bu mesafeye $c(i, j)$ denir. Aşağıdaki ulaştırma problemi çözülür:

$$\min \sum_i \sum_j c(i, j) g(i, j) \quad (2.4)$$

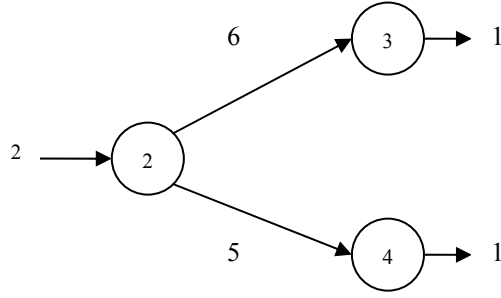
$$\sum_j g(i, j) = s(i) \text{ tüm } i' \text{ ler için} \quad (2.5)$$

$$\sum_i g(i, j) = d(j) \text{ tüm } j' \text{ ler için} \quad (2.6)$$

$$g(i, j) \geq 0 \text{ tüm } i, j' \text{ ler için} \quad (2.7)$$

Burada, $g(i, j)$ i ve j arasındaki en kısa yolun grafiğe kaç kez eklenmesi gerektiğini göstermektedir.

Örnek 3'te, 2 noktasının, $s(2) = 2$ olacak şekilde giriş derecesinin çıkış derecesinden daha büyük olduğu görülmektedir. 3 ve 4 noktalarının da $d(3) = d(4) = 1$ olacak şekilde çıkış dereceleri giriş derecelerinden daha büyüktür. 2 noktası ve 3 noktası arasındaki en kısa yol (2, 4) ve (4, 3) yaylarından oluşur ve mesafe $c(2, 3) = 6$ 'dır; 2 noktası ve 4 noktası arasındaki en kısa yol (2, 4) yayından oluşur ve mesafe $c(2, 4) = 5$ 'dir. Ulaştırma ağı Şekil 2.12'de gösterilmiştir ve $g(2, 3) = g(2, 4) = 1$ sonucunu verir. Bu nedenle G^* grafiğini yaratmak için (2, 4) yayını iki kez, (4, 3) yayını bir kez eklemek gerekir.



Şekil 2.12 Örnek.3 için nakliye ağı (Evans ve Minieka, 1992)

Bir dönüştürülmüş minimum maliyet akış probleminin ya da ulaştırma probleminin çözümünü gerektirmeden direkt olarak ağ üzerinde çalışan bir algoritma Lin ve Zhao (1988) tarafından önerilmiştir. Böyle bir algoritma eğer daha büyük bir algoritmanın ya da karar destek sisteminin içine yerleştirilirse avantajlıdır. Bu yaklaşım biraz farklı bir lineer programlama formülasyonuna dayanır (Evans ve Minieka, 1992).

$h(i, j)$ Euler turunda (i, j) yayının kaç kez dolaşıldığını gösterebilir. (önceki formülasyonda $f(i, j)$ (i, j) yayının Euler turunda kaç kez tekrar edildiğini göstermekteydi).

$$\min \sum a(i, j) h(i, j) \quad (2.8)$$

$$-\sum h(i, j) + \sum h(i, j) = 0 \quad \text{tüm } i' \text{ ler için} \quad (2.9)$$

$$h(i, j) \geq 1 \text{ 'dir ve tam sayıdır, tüm } (i, j) \text{ 'ler için} \quad (2.10)$$

İkili değişkenler $p(i)$, her b, r (2.9) kısıtıyla ve $r(i, j)$ her bir (2.10) kısıtıyla ilişkilendirilir. İkili lineer program:

$$\max \sum r(i, j) \quad (2.11)$$

$$-p(i) + p(j) + r(i, j) \leq a(i, j) \quad (2.12)$$

$$r(i, j) \geq 0 \quad (2.13)$$

$$p(i) \text{ kısıtlanmamış} \quad (2.14)$$

Şimdi de tamamlayıcı serbestlik durumları göz önüne alınır.

$$\left[-p(i) + p(j) + r(i, j) - a(i, j) \right] h(i, j) = 0 \quad (2.15)$$

$$\left[h(i, j) - 1 \right] r(i, j) = 0 \quad (2.16)$$

Ana probleme herhangi bir fizibil çözüm için $h(i, j) \geq 1$ olduğundan (2.15)'e göre şu sonucu elde etmeliyiz:

$$r(i, j) = a(i, j) + p(i) - p(j) \quad (2.17)$$

[(2.12)'nin otomatik olarak sağlandığına dikkat edilmelidir.]Ayrıca,

$$\text{Eğer } h(i, j) = 1 \text{ ise, } r(i, j) \geq 0$$

$$\text{ve} \quad (2.18)$$

$$\text{Eğer } h(i, j) \geq 2 \text{ ise, } r(i, j) = 0$$

olur (Evans ve Minieka, 1992).

Bu yüzden, ana fizibilite, ikili fizibilite ve tamamlayıcı serbestlik optimallik durumları (2.9), (2.10), (2.17) ve (2.18) ile özetlenmektedir.

Algoritma, (2.9), (2.17) ve (2.18)'i sağlayan fakat (2.10)'u sağlamayan bir başlangıç çözümü ile başlar ve diğer durumları bozmadan (2.10)'u sağlayıncaya kadar primal ve dual çözümleri ayarlamaya devam eder. En kısa yol problemleri algoritmanın gidişatı boyunca düzenli olarak çözülür. En kısa yol hesaplamaları için yaylarla ilişkili ağırlıkları gösteren bir $w(i, j)$ yardımcı değişkeni kullanılır. Algoritma şu şekildedir:

Adım 1. Tüm (i, j) yayları için; $h(i, j) = 0, r(i, j) = a(i, j), w(i, j) = 0$ ve $w(i, j) = \infty$ yapılır. Tüm i noktaları için $p(i) = 0$ yapılır.

Adım 2. Eğer her bir yay en azından bir kez dolaşılıyorsa (yani, $h(i, j) \geq 1$), durulur. Aksi halde, $h(t, s) = 0$ yapan bir (t, s) yayı seçilir.

Adım 3. $w(i, j)$ ağırlıkları kullanılarak s 'ten ağdaki her bir i noktasına bir en kısa yol bulunur. Bu yola $P(s, i)$ denir ve s 'ten i 'ye en kısa mesafe $d(i)$ olarak gösterilir. Döngünün, $P(s, t)$ 'nin ve (t, s) yayının birleşiminden oluştuğu göz önüne alınır. Döngüdeki, (t, s) ile aynı yönde olan her bir yay için $h(i, j) = h(i, j) + 1$ yapılır; (t, s) 'nin aksi yönündeki her bir yay içinse $h(i, j) = h(i, j) - 1$ yapılır.

Adım 4. Tüm (i, j) yayları için, $r(i, j) = \min\{d(i), d(t)\} - \min\{d(j), d(t)\} + r(i, j)$ yapılır.

Eğer $h(i, j) = 0$ ise $w(i, j) = 0$ ve $w(j, i) = \infty$ yapılır. Eğer $h(i, j) = 1$ ise, $w(i, j) = r(i, j)$ ve $w(j, i) = \infty$ yapılır. Son olarak eğer $h(i, j) \geq 2$ ise $w(i, j) = w(j, i) = 0$ yapılır. Adım 2'ye geri dönülür (Evans ve Minieka, 1992).

Örnek 4:

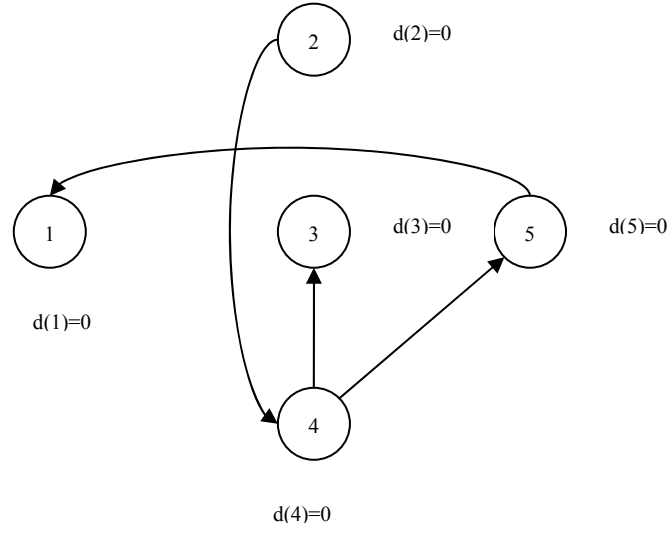
Şekil 2.9'daki problemi göz önüne alalım.

Adım 1. Tüm değişkenler başlangıç durumuna getirilir. (2.10) hariç tüm tamamlayıcı serbestlik durumlarına sahip olduğu gözlemlenmelidir.

Çizelge 2.1 Değişken başlangıç durumları (Evans ve Minieka, 1992)

Yay	$r(i, j)$	$h(i, j)$	$w(i, j)$	$w(j, i)$
1, 2	8	0	0	∞
2, 4	5	0	0	∞
3, 2	2	0	0	∞
3, 5	3	0	0	∞
4, 3	1	0	0	∞
4, 5	2	0	0	∞
5, 1	5	0	0	∞
5, 2	7	0	0	∞

Adım 2. $(t, s) = (1, 2)$ seçilir. $w(i, j)$ ağırlıkları kullanılarak 2 noktasından diğer tüm noktalara en kısa yollar bulunur. En kısa yol ağaç yapısı ve mesafe etiketleri aşağıda görülmektedir:



(1, 2) yayı yönünde 1-2-4-5-1 döngüsü dolaşarak,

$h(1, 2) = 1$, $h(2, 4) = 1$, $h(4, 5) = 1$ ve $h(5, 1) = 1$ yapılır.

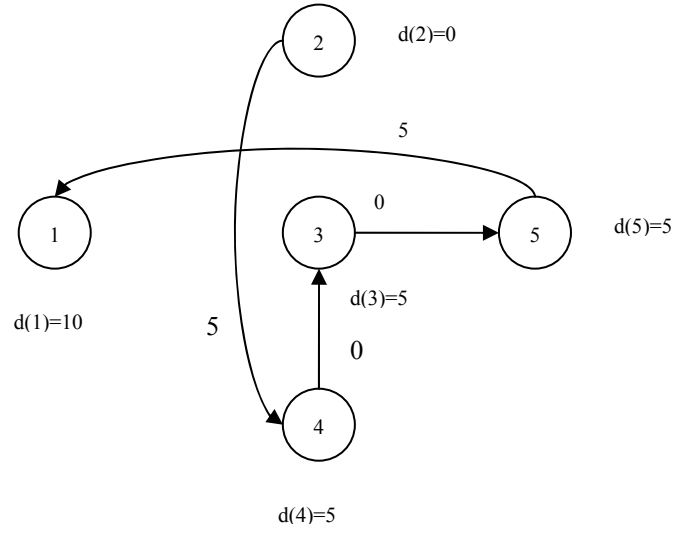
Adım 4. $r(i, j)$ ve $w(i, j)$ değerleri güncellenir. Aşağıda özetlenmiştir:

Çizelge 2.2 Yeni değişken değerleri

Yay	$r(i, j)$	$h(i, j)$	$w(i, j)$	$w(j, i)$
1, 2	8	1	8	∞
2, 4	5	1	5	∞
3, 2	2	0	0	∞
3, 5	3	0	0	∞
4, 3	1	0	0	∞
4, 5	2	1	2	∞
5, 1	5	1	5	∞
5, 2	7	0	0	∞

Adım 2. Yay $(3, 2) = (t, s)$ seçilir.

Adım 3. 2 noktasından diğer tüm noktalara en kısa yollar bulunur. En kısa yol ağaç yapısı ve mesafe etiketleri aşağıda gösterilmiştir:



3-2-4-3 döngüsü dolaşarak $h(3, 2) = 1$, $h(2, 4) = 2$ ve $h(4, 3) = 1$ yapılır.

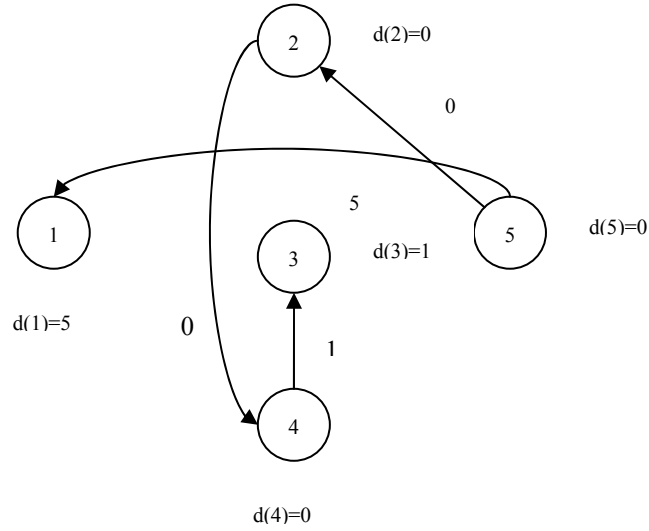
Adım 4. $r(i, j)$ ve $w(i, j)$ güncellenir.

Çizelge 2.3 Güncellenen değişken değerleri

Yay	$r(i, j)$	$h(i, j)$	$w(i, j)$	$w(j, i)$
1, 2	13	1	13	∞
2, 4	0	2	0	0
3, 2	7	1	7	∞
3, 5	3	0	0	∞
4, 3	1	1	1	∞
4, 5	2	1	2	∞
5, 1	5	1	5	∞
5, 2	12	0	0	∞

Adım 2. Yay $(3, 5) = (t, s)$ seçilir.

Adım 3. 5 noktasından diğer tüm noktalara en kısa yollar belirlenir. En kısa yol ağaç yapısı ve mesafe etiketleri aşağıdaki gibidir:



3-5-2-4-3 döngüsü dolaşarak $h(3, 5) = 1$, $h(5, 2) = 1$, $h(2, 4) = 3$ ve $h(4, 3) = 2$ yapılır. Bu noktada, tüm $h(i, j) \geq 1$ olur ve durulur. Çözümün Şekil 2.11'deki G^* 'a karşılık geldiği görülmektedir.

$r(i, j)$ 'nin Adım 4'teki son değerleri hesaplanırsa aşağıdaki sonuçlar elde edilir:

Çizelge 2.4 $r(i, j)$ değerleri

Yay	$r(i, j)$
1, 2	14
2, 4	0
3, 2	8
3, 5	4
4, 3	0
4, 5	2
5, 1	4
5, 2	12

İkili hedef fonksiyonu $\sum r(i, j)$ olduğundan, sonuçta $\sum r(i, j) = 44$ olduğu ve bunun da optimal birincil sonuç değeri olduğu görülür (Evans ve Minieka, 1992).

Algoritmanın ispatı: Bu algoritmanın yönlendirilmiş ÇPP'yi doğru bir şekilde çözdüğünü

göstermek için $r(i, j)$, $h(i, j)$ ve $p(i)$ 'nin optimallik koşulları (2.9), (2.10), (2.17) ve (2.18)'i sağladığını göstermek gerekmektedir. $r(i, j)$ 'nin algoritmanın gidişatı süresince tüm (i, j) 'ler için ikili kısıtları sağladığını ispatlamak yeterlidir. İlk olarak, $r(i, j) \geq 0$ olduğu gösterilir. Bu tümevarım yoluyla yapılır. k iterasyon sayacı olsun. Açık bir şekilde $k = 0$ iken $r^0(i, j) = a(i, j) \geq 0$ olur. $r^{k-1} \geq 0$ olduğunu varsayalım. Algoritmanın 4. adımında, $d^k(j) \leq d^k(i) + w^k(i, j)$ ve $w^k(i, j) \leq r^k(i, j)$ elde edilir. Bu nedenle, $d^k(i) + r^{k-1}(i, j) - d^k(j) \geq 0$ sonucu çıkar. Bu sonuçlar birleştirildiğinde $r^k(i, j) \geq 0$ sonucu elde edilir.

Daha sonra, $r(i, j)$ 'nin (2.12)'yi sağladığı gösterilir.

$$p(i) = p^k(i) = p^{k-1}(i) + \delta p^k(i) \text{ olsun.}$$

Burada $\delta p^k(i) = \min\{d^k(i), d^k(t)\}$ 'dir. Algoritmadan, tüm i 'ler için $p^0(i) = d^0(i) = 0$ sonucu bulunur. Bu nedenle, tüm i 'ler için, $p^1(i) = 0$ ve $\delta p^1(i) = 0$ olur. Böylece, $k = 1$ için, $r^k(i, j) = \delta p^k(i) + r^{k-1}(i, j) - \delta p^k(j) = p^k(i) + a(i, j) - p^k(j)$ olur. Bunun $k - 1$ için doğru olduğu varsayılır. Daha sonra,

$$\begin{aligned} r^k(i, j) &= \delta p^k(i) + r^{k-1}(i, j) - \delta p^k(j) \\ &= \delta p^k(i) + \left[p^{k-1}(i) + a(i, j) - p^{k-1}(j) \right] - \delta p^k(j) \\ &= p^k(i) + a(i, j) - p^k(j) \end{aligned}$$

olur. Böylece (2.12) sağlanır ve ispat tamamlanır (Evans ve Minieka, 1992).

2.1.5 Karma Grafikler İçin Postacı Problemi

Bu bölümde, bazı yayların yönlendirilmiş ve bazılarının yönlendirilmemiş olduğu (karma grafik) bir G grafiğinde postacı problemi göz önüne alınacaktır. Eğer bir yay yönlendirilmişse, postacı bu yayı yalnızca yönü boyunca geçmelidir (tek yönlü yol). Eğer bir yay yönlendirilmemişse, postacı bu yayı her iki yönde de geçebilir (çift yönlü yol). Yönlendirilmemiş bir yay, noktaların giriş ve çıkış dereceleri hesaplanırken göz önüne alınmaz.

Postacı karma bir grafikte her zaman bir rota bulamaz. Bir S noktalar kümesine sahip bir grafikte, S 'te olan bir noktayı S 'te olmayan bir noktaya birleştiren tüm yaylar S 'teki noktalara doğru yönlendirilmiş olabilir. Bu durumda, postacı bir kez S 'teki bir noktaya ulaştığında artık S dışındaki bir noktaya gidemez. Tuzağa düşmüştür ve postacı problemi için bir çözüm yoktur. Eğer bu özellikte bir S kümesi yoksa postacı, ne kadar uzun sürerse sürsün tüm yolları dolaşıp başlangıç noktasına dönünceye kadar yolculuğunu sürdürebilir.

Bir karma grafiğin Euler turu içermesi için üç koşul sağlanmalıdır. İlk olarak, G bağlantılı olmalıdır. İkinci olarak, G grafiğinin her noktası çift dereceli olmalıdır (yönlere bakılmaksızın). Üçüncüsü, her S noktalar alt kümesi için S 'ten tamamlayıcısına yönelmiş yayların sayısı ve tamamlayıcısından S 'e yönelmiş yayların sayısı arasındaki fark, S 'e ve tamamlayıcısına gelen yönlendirilmemiş yayların sayısından daha az ya da eşit olmalıdır. Üçüncü koşulun sağlanması zordur. Yönlendirilmemiş grafikler için ikinci koşulun üçüncü koşulu kapsadığına; yönlendirilmiş grafikler içinse üçüncü koşulun grafiğin simetrik olmasına karşılık geldiğine dikkat edilmelidir.

Karma bir $G = (X, A)$ grafiği için, üç durum ayrı ayrı ele alınmalıdır:

Durum A: G grafiği çift dereceli ve simetriktir.

Durum B: G grafiği çift derecelidir fakat simetrik değildir.

Durum C: G grafiği ne çift ne de simetriktir.

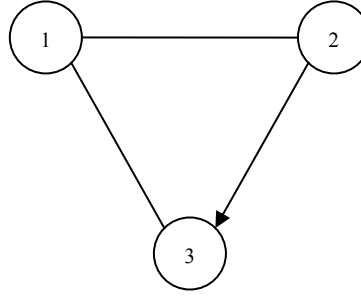
Durum A. Bu analiz etmesi en basit durumdur, çünkü bu durum için çözüm tekniği daha önce sözü edilen çift dereceli yönlendirilmiş ve çift dereceli yönlendirilmemiş grafikler için çözüm tekniklerinin bir bileşkesidir.

G grafiğindeki herhangi bir yönlendirilmiş yaydan başlanarak, yönlendirilmiş yaylardan, daha önce yönlendirilmiş çift dereceli grafikler için olduğu gibi bir döngü oluşturulur. (G simetrik olduğundan bu mümkündür.) Tüm yönlendirilmiş yaylar kullanılıncaya kadar bu işlem sürdürülür. (Kullanılmamış yaylar her zaman çift dereceli simetrik bir grafik oluşturacağından bu mümkündür.)

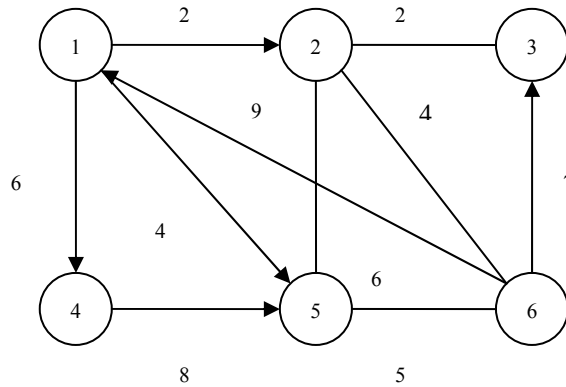
Daha sonra, bu işlem G grafiğindeki yalnızca yönlendirilmemiş yaylar kullanılarak tekrarlanır. (Yine bu her zaman mümkündür çünkü kullanılmamış yaylar bir çift dereceli grafik oluşturur) G grafiğindeki tüm yaylar kullanıldıktan sonra, yukarıda oluşturulan tüm döngüler tek bir C döngüsü oluşturacak şekilde birleştirilir. C döngüsü G grafiğinin bir Euler turunu oluşturur ve Durum A için postacı problemine optimal bir çözümdür.

Durum B. G grafiği çift dereceli değil fakat simetriktir. Bu durumda, önceden postacının herhangi bir yayı tekrar etmesinin gerekip gerekmediğini bilmek kolay değildir. Örneğin, 1'den 2'ye dolaşılan yönlendirilmemiş $(1, 2)$ yayı ve 3'ten 1'e dolaşılan yönlendirilmemiş $(3, 1)$ yayına sahip Şekil 2.13'teki simetrik olmayan çift dereceli grafik için Euler turu bir optimal çözümdür. Diğer yandan, Şekil 2.14'teki çift dereceli simetrik olmayan grafik için hiçbir Euler turu optimal çözüm olamaz, çünkü $(6, 1)$ yayının, postacının $(1, 2)$, $(1, 4)$ ve $(1, 5)$ yayları boyunca 1 noktasından ayrılabilmesi için iki kez tekrar edilmesi gerekmektedir.

Karma postacı algoritması, G grafiğindeki her bir yönlendirilmemiş yay için keyfi olarak bir yön seçer. Bu, G grafiğini G_D çift dereceli yönlendirilmiş grafiğine dönüştürür ve daha önce çift dereceli yönlendirilmiş grafikler için bahsedilen çözüm tekniği G_D grafiğine uygulanabilir. Bununla beraber, yay yönlerinin keyfi seçiminden dolayı bazı yay yönlerini düzeltmek için bazı modifikasyonlar gereklidir (Evans ve Minieka, 1992).



Şekil 2.13 Euler turuna sahip bir karma grafik. (Evans ve Minieka, 1992)



Şekil 2.14 Euler turuna sahip olmayan bir karma grafik (Evans ve Minieka, 1992)

2.1.5.1 Karma Postacı Algoritması

$G = (X, A)$ herhangi bir çift dereceli karma grafik olsun. Optimal bir postacı rotası (eğer varsa) aşağıdaki şekilde bulunabilir.

U , G grafiğindeki tüm yönlendirilmemiş yayların kümesini, V de G grafiğindeki tüm yönlendirilmiş yayların kümesini gösterebilir. Geçici olarak U 'daki her bir yay için bir yön seçilir. Sonuç grafiğine G_D diyelim. G_D 'deki her bir i noktası için,

$$D(i) = d^-(i) - d^+(i) \quad (2.19)$$

hesaplanır. Eğer $D(i) < 0$ ise i noktası $-D(i)$ talepli bir hedeftir. Eğer $D(i) > 0$ ise i noktası $D(i)$ stok miktarlı bir kaynaktır. Eğer $D(i) = 0$ ise i noktası bir aktarma noktasıdır.

Eğer G_D 'deki tüm noktalar aktarma noktası ise G_D grafiği bir çift dereceli, simetrik yönlendirilmiş grafikdir ve önceden bahsedilen çözüm tekniği G_D grafiğine uygulanabilir. Bu teknik G_D grafiğinde, G grafiğinin bir optimal postacı rotasına karşılık gelen bir Euler turu sağlar.

Aksi halde, aşağıdaki gibi bir $G' = (X, A')$ grafiği oluşturulur.

- (a) Her bir $(i, j) \in V$ yayı için, A' 'ne (i, j) 'nin uzunluğuna eşit maliyette ve sonsuz kapasitede bir (i, j) yayı yerleştirilir.
- (b) Her bir $(i, j) \in U$ yayı için, A' 'nde yönlendirilmiş (i, j) ve (j, i) yayları oluşturulur. Bu yayların her biri (i, j) 'nin uzunluğuna eşit maliyette ve sonsuz kapasitede olsun.
- (c) Her bir $(i, j) \in U$ yayı için, A' 'nde, yönü G_D 'de bu yayya atananın tersi olan bir yönlendirilmiş (j, i) yayı oluşturulur. Bu yaylara yapay yaylar denir. Her bir yapay yayya sıfır maliyet ve iki kapasite atanır.

Yukarıda G_D grafiği için tanımlanan kaynak stok miktarı ve hedef talepleri kullanılarak; G' grafiğinde tüm hedef taleplerini karşılayacak bir minimum maliyet akışı bulmak için minimum maliyet akış algoritması uygulanır.

Eğer böyle bir akış mevcut değilse postacı rotası da yoktur. Aksi halde, $f(i, j)$, minimum maliyet akış algoritması tarafından üretilen minimum maliyet akışında, G' 'nde (i, j) yayı boyunca gönderilen akış birimlerinin sayısını gösterir. Her bir optimal akış değeri negatif olmayan bir tamsayıdır. Bu algoritmanın ispatında, her bir yapay yayın sıfır ya da iki akış

birimi taşıdığı gösterilecektir.

Aşağıdaki gibi bir G^* grafiği oluşturulur:

- (a) G' 'ndeki her bir yapay olmayan (i, j) yayı için, G^* grafiğinde (i, j) yayının $f(i, j) + 1$ kopyası yerleştirilir.
- (b) Eğer bir yapay yayda akış iki birimse, G^* grafiğine bu yayın bir kopyası yerleştirilir.
- (c) Eğer bir yapay yayda akış sıfır ise, bu yayın yönü tersine çevrilir ve G^* grafiğine bunun bir kopyası yerleştirilir. (Bu nedenle, eğer bir yapay yayı geçen birim olmazsa G_D 'de bu yay için atanan geçici yön tutulur; eğer bir yapay yayı iki birim geçiyorsa, bu yay G_D 'de atanan yön tersine çevrilir.)

G^* grafiği, çift dereceli, simetrik, yönlendirilmiş bir grafiktir. Çift dereceli, simetrik ve yönlendirilmiş grafikler için önceden tanımlanan çözüm tekniği şimdi bir Euler turu bulmak için G^* grafiğine uygulanabilir. G^* grafiğinin bu Euler turu orijinal G grafiğinin bir optimal postacı rotasına karşılık gelmektedir.

İspat. G grafiği muhakkak simetrik olacak diye bir şey olmadığından, bazı noktalar gereğinden fazla giren yaya sahipken diğer noktalar da gereğinden fazla çıkan yaya sahip olabilir. İdeal olarak, sonuçta çıkacak olan yönlendirilmiş grafiğin simetrik olması için, G grafiğindeki tüm yönlendirilmemiş yaylara bir yön atanmak istenir. O zaman çift dereceli, simetrik ve yönlendirilmiş grafikler için önceden tanımlana çözüm tekniği G grafiğinin bir Euler turunu bulmak için uygulanabilir.

Bununla beraber, sonuç grafiğinin simetrik olması için yönlendirilmemiş grafiklere yön verilmesi mümkün olmayan durumlarda olabilir. Bu durumda, bazı yaylar (yönlendirilmiş veya yönlendirilmemiş) postacı tarafından tekrar edilmelidir. Tabi ki, postacı toplam uzunluklarını mümkün olduğunca kısa yayları tekrar etmek isteyecektir.

Algoritma G grafiğindeki her bir yönlendirilmemiş yay için bir geçici yön seçer. Sonuçta ortaya çıkan yönlendirilmiş grafiğe G_D grafiği denir. Çift dereceli yönlendirilmiş grafikler için önceden bahsedilen çözüm tekniği G_D grafiğine uygulanabilirdi. Fakat, bu teknikle oluşturulan çözüm geçici yönler dayanmaktadır ve G_D grafiğindeki geçici yönlerin postacı probleminde optimal olmayan bir çözüme götürmesi her zaman mümkündür.

Algoritma bir G' grafiği oluşturur ve G_D grafiğindekiyle aynı kayna stok miktarları ve hedef taleplerini kullanarak, tüm hedef taleplerini karşılayan bir minimum maliyet akışı bulur.

Bunlar G' grafiğindeki üç çeşit yaydır:

- (a) G grafiğindeki yönlendirilmiş yaylara karşılık gelen yaylar (bu yaylar sıfırdan farklı bir maliyete ve sonsuz kapasiteye sahiptir).
- (b) G grafiğindeki yönlendirilmemiş yaylara karşılık gelen yapay olmayan yaylar (bu yaylar da sıfırdan farklı bir maliyete ve sonsuz kapasiteye sahiptir).
- (c) G grafiğindeki yönlendirilmemiş yaylara karşılık gelen yapay yaylar (bu yaylar sıfır maliyete ve iki kapasiteye sahiptir).

Minimum maliyet akışında (a) veya (b) tipi bir yay tarafından taşınan $f(i, j)$ sayısı, karşılık gelen yönlendirilmiş yayların postacı tarafından tekrarlanma sayısına eşittir. Bu nedenle, postacı her bir $(i, j) \in V$ yayını $f(i, j) + 1$ kez dolaşacak ve her bir $(i, j) \in U$ yayını da toplam $f(i, j) + f(j, i) + 1$ kez dolaşacaktır.

Her bir yapay yay (c tipi) sıfır veya iki akış birimi taşır. Eğer bir $(j, i)_1$ yapay yayı minimum maliyet akışında iki birim taşıyorsa, bu yay j 'de stok miktarını iki birim düşürür ve i 'de stok miktarını iki birim artırır. Aynı etki G_D grafiğinde bu yay için ters geçici yön seçilerek de elde edilebilirdi. Bu nedenle, algoritma bu yayın geçici yönünü terinse çevirir.

Eğer bir $(j, i)_1$ yapay yayı minimum maliyet akışında hiç akış birimi taşımıyorsa, bu yayın i ve j noktalarındaki stok miktarlarına etkisi yoktur. Bu, G_D grafiğinde bu yaya verilen geçici yönün tutulmasına eşittir. Bu nedenle, algoritma bu yayın geçici yönünü tersine çevirir.

G' grafiğindeki yaylar için $f(i, j)$ minimum maliyet akış değerlerinden, algoritma bir G^* yönlendirilmiş grafiği oluşturur. Buna göre;

- (a) G^* grafiği çift dereceli ve simetriktir.
- (b) G^* grafiğinin bir Euler turu G grafiğinin bir optimal postacı rotasına karşılık gelir.
- (c) Eğer minimum maliyet akış algoritması G' grafiğindeki tüm sink taleplerini karşılayacak herhangi bir akış bulamasa, G grafiği için postacı rotası mevcut değildir.

İspat (a). G_D grafiği bir çift dereceli grafik olduğu için, $d^+(i) + d^-(i)$ G_D 'deki tüm i noktaları için çift sayıdır. Bu nedenle, $d^+(i)$ ve $d^-(i)$ her ikisi de ya tek ya da çifttir. Her iki durumda da $D(i)$ çift olmalıdır. Sonuç olarak, G' grafiğindeki tüm stok miktarları ve talepler çift sayıdır (sıfır da çift sayıdır). Ayrıca, tüm yay kapasiteleri çift sayıdır. Bu nedenle, bir minimum maliyet akış algoritması tüm akış değerlerinin çift sayı olduğu bir optimal akış üretecektir. Bu nedenle, minimum maliyet akışındaki akış birimleri çiftler halinde

dolaşacaktır.

G^* 'daki $d(i)$ 'nin değeri G 'deki $d(i)$ 'nin değeri artı yapay olmayan yaylar boyunca i noktasına giren veya ayrılan akış birimlerinin sayısına eşittir. Tüm akış birimleri çiftler halinde dolaştığından G^* grafiğinde $d(i)$ 'nin çift olduğu sonucu çıkar. Bu nedenle G^* grafiği çifttir.

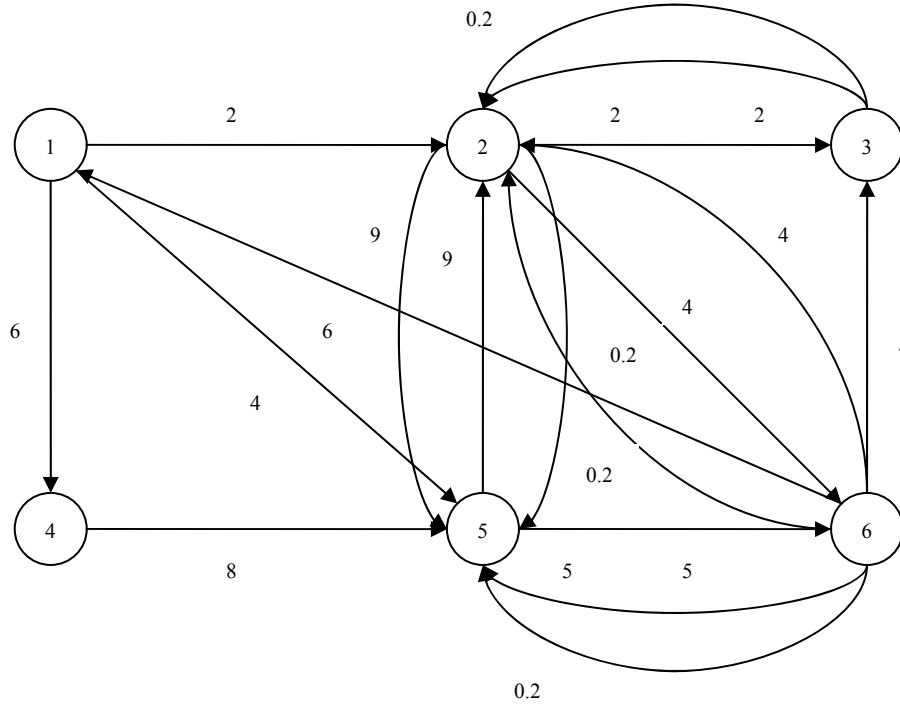
i noktasına yapay olmayan bir yay yoluyla ulaşan bir akış birimi çifti G^* grafiğinde $d^-(i)$ 'nin değerini iki birim artırır. i noktasını yapay olmayan bir yay yoluyla terk eden bir akış birimi çifti $d^+(i)$ 'yi iki birim yükseltir. i noktasına yapay bir yay yoluyla ulaşan bir akış birimi çifti bu yayın geçici yönünün ters dönmesine sebep olur, bu da $d^-(i)$ 'nin iki birim artmasına yol açar. i noktasını yapay bir yay boyunca terk eden bir akış birimi çifti bu yayın geçici yönünün ters dönmesine sebep olur ve bu da $d^+(i)$ 'nin iki birim artmasına yol açar. Bu nedenle, i noktasına varan her bir akış birimi $d^-(i)$ 'yi bir birim artırır ve i noktasını terk eden her bir akış birimi $d^+(i)$ 'yi bir birim artırır. Minimum maliyet akışı G grafiğindeki tüm noktalar için (3) eşitliğini sağladığından, G^* grafiğinin simetrik olduğu sonucu çıkar.

İspat (b). Karma postacı algoritması ile üretilen postacı rotasının optimal olmadığını varsayalım. O halde, kopya yayları daha da küçük bir toplam uzunluğa sahip başka bir postacı rotası vardır. Bu rota, G' grafiğindeki, minimum maliyet akış algoritmasıyla oluşturulan akıştan daha düşük maliyetli bir akışa karşılık gelmelidir ki bu bir çelişkidir.

İspat (c). Herhangi bir G_D grafiğinde $\sum_i d^-(i) = \sum_i d^+(i)$ olduğundan, G_D grafiğindeki toplam kaynak stok miktarı yine G_D grafiğindeki toplam hedef talep miktarına eşit olmalıdır. Bu nedenle, tüm kaynak stoku, tüm hedef talebini karşılamak için sırayla yollanmalıdır.

Eğer G' grafiği i 'den j 'ye bir yay içeriyorsa bu yay sonsuz kapasiteli olmalıdır. Minimum maliyet akış algoritmasının kullanıldığını ve tüm hedef talepleri karşılanmadan sonlandırıldığını varsayalım. S , minimum maliyet akış algoritmasının son tekrarından sonra etiketlenen noktalar kümesini gösterebilir. Minimum maliyet akış algoritmasının bir alt rutini olan akış büyütme algoritmasından, S 'teki bir noktadan S 'te bulunmayan bir noktaya tüm yayların tam kapasite akış taşınması gerektiği bilinmektedir. Bazı yayların kapasiteleri sonsuz olduğundan bu mümkün değildir. Bu nedenle, bu tarz yaylar olamaz ve bir etiketlenmiş bitiş noktası ve bir etiketlenmemiş bitiş noktası olan tüm yaylar S kümesine doğru yönlendirilmiştir. Sonuç olarak, postacı S kümesine bir kez ulaştığında buradan ayrılamaz ve hiçbir postacı rotası mevcut değildir (Evans ve Minieka, 1992).

$f(3, 2)_1 = f(2, 6) = f(6, 1) = 2$ olur ve tüm diğer akış değerleri sıfıra eşit olur. $f(3, 2)_1 = 2$ olduğundan bu yayın başlangıç yönünü 3'ten 2'ye yönlenmesi için tersine çevirmek gereklidir. Tüm diğer keyfi yönler tutulur. Bundan başka, (2, 6) ve (6, 1) yayları $f(2, 6) = f(6, 1) = 2$ olduğundan iki kez tekrar edilmelidir.



Şekil 2.16 G' grafiği. (Evans ve Minieka, 1992)

G^* grafiği Şekil 2.17'de gösterilmektedir. G^* grafiğinin çift dereceli, simetrik ve yönlendirilmiş bir grafik olduğuna dikkat edilmelidir. Önceden bahsedilen çift dereceli, simetrik ve yönlendirilmiş grafikler için çözüm yöntemi, G^* grafiğinin bir Euler turunu bulmak için uygulanabilir. Bu rota G grafiğindeki bir optimal postacı rotasına karşılık gelmektedir. G' grafiğindeki minimum maliyet akışı 20 birim olduğundan, G grafiğindeki optimal postacı rotası yayları 20 birim toplam uzunluğunda tekrar edecektir.

Durum C. G grafiği ne çift dereceli ne de simetriktir.

Yazarların bilgisi dahilinde bu durum için etkili bir optimal çözüm tekniği mevcut değildir. Bu probleme iki aşamalı bir problem gibi yaklaşılabilir: İlk olarak grafik optimal bir şekilde çift dereceli yapılır; daha sonra çift dereceli grafik Durum B'de verilen optimal prosedür kullanılarak simetrik yapılır. Ancak her bir aşamanın optimal çözümünün bütün problem için optimum bir çözüm vereceğinin bir garantisi yoktur (Evans ve Minieka, 1992).

2.2 Gezgin Satıcı ve İlgili Nokta Rotalama Problemleri

Bu bölümde ise Araç Rotalama Probleminin çözümüne öncülük edecek olan Gezgin Satıcı Problemi ve çözüm algoritmaları üzerinde durulacaktır.

2.2.1 Giriş ve Örnekler

Önceki bölümde, bir grafiğin kenarları boyunca rotalama problemleri çalışılmıştır. Bu bölümde, bir grafiğin noktaları üzerinde rotalama problemleri çalışılacaktır. Bunlar, ağ ve grafik optimizasyonunda önemli ve uygulamalı (elverişli) problem sınıflarıdır. Muhtemelen en ünlü rotalama problemi, gezgin satıcı problemidir ve şu şekilde tanımlanabilir. Bir gezgin satıcı, evine dönmeden önce bölgesindeki her bir şehre (kasabaya) uğramalıdır. Satıcı, mümkün olduğunca az dolaşmak için ziyaretlerini programlamak isteyebilir. Böylece satıcı, bölgedeki tüm kentleri ziyaret etmek için ihtiyaç duyulan toplam mesafeyi (veya süre veya maliyet) minimize edecek rotayı bulma problemini karşılayabilir.

Gezgin satıcı problemi bir grafik ile ifade edilebilir. Noktaları satıcının bölgesindeki kentlere ve kenarları da iki kenti birleştiren yollara karşılık gelecek bir $G = (X, A)$ grafiği oluşturalım. Her bir (x, y) kenarının $d(x, y) \geq 0$ uzunluğu, (x, y) kenarı boyunca yapılan yolculuğa karşılık gelen mesafeye (veya süre veya maliyet) eşit olsun. G 'deki her bir noktayı en azından bir kez içeren bir döngüye *satıcı döngüsü* denir. G 'deki her bir noktayı yalnızca bir kez içeren döngüye, bu problemleri 1859 yılında ilk kez çalışan İrlandalı matematikçi Sir William Rowan Hamilton nedeniyle *Hamilton Döngüsü* denir. Hamilton döngüleri grafikte genellikle gezgin satıcı turları olarak adlandırılır. Genel satıcı problemi, mümkün olan en küçük toplam uzunluk ile bir satıcı döngüsü bulma problemidir. (Gezgin) Satıcı Problemi ya da GSP, mümkün olan en küçük toplam uzunluk ile bir Hamilton döngüsü bulma problemidir.

Satıcı problemleri hem yönlendirilmiş hem de yönlendirilmemiş grafiklerle çözülebilir. Yönlendirilmemiş bir grafikte noktalar boyunca bir döngü aranır; yönlendirilmiş bir grafikte ise yönlendirilmiş bir döngü aranır. Yönlendirilmemiş grafiklerdeki problemler gene olarak simetrik gezgin satıcı problemleri olarak adlandırılır. Yönlendirilmiş grafiklerde olanlar ise asimetrik gezgin satıcı problemi olarak adlandırılır. Bir yönlendirilmemiş kenar, her biri yerini aldıkları yönlendirilmemiş kenarla aynı uzunlukta olan iki tane karşılıklı yönlendirilmiş yay ile değiştirilebilir. Böylece, yalnızca yönlendirilmiş durumda çözüm yapan algoritmayla ilgilenmek gerekir. Bununla beraber, yönlendirilmemiş durum bazı önemli algoritmik sadeleştirmelere imkan vermektedir. Bu nedenle, bu bölümde her iki problem tipi de göz önünde bulundurulacaktır.

Uygulamadaki pek çok problem satıcı problemleri olarak formüle edilebilir (Evans ve Minieka, 1992).

2.2.1.1 Toplama ve Dağıtım Problemleri

Farklı bölgelerden oluşan bir kümeden periyodik toplama ve dağıtım içeren birçok örnek gezgin satıcıyla doğal bir benzerliğe sahiptir. Örneğin, Federal Express, United Paket Servisi ve U.S. Posta Servisi gibi şirketler her iş günü 10:30'a kadar gece mektup ve paketlerini dağıtmak zorundadırlar. Her bir kamyon her bir müşteriyi yalnızca bir kez ziyaret etmelidir. Her bir rota boyunca dolaşılan toplam süreyi minimize etmek, sürücülerin verimliliğini maksimize eder. Diğer bir örnek de alkolsüz içki satıcısının periyodik olarak makineleri yeniden doldurması ve parayı toplamasıdır. Buna benzer birçok durum bulunabilir.

Yol ağı pek çok tek yönlü yol içermedikçe x noktasından y noktasına gitme süresi genellikle y noktasında x noktasına gitme süresine eşittir. Bu nedenle, grafik yönlendirilmemiştir ve bu bir simetrik gezgin satıcı problemidir (Evans ve Minieka, 1992).

2.2.1.2 Bilgisayar Kablolama (Lenstra ve Rinnooy Kan, 1975)

Bu örnek, Amsterdam'daki Nükleer Fizik Araştırma Enstitüsü'nde ortaya çıkmıştır. Bir bilgisayar arayüzü pek çok çeşitli modüller içermektedir. Her bir modüle pek çok pin yerleştirilmiştir. Her bir modülün pozisyonu önceden kararlaştırılır. Pinlerin verilen bir alt kümesi birbirlerine kablolarla bağlı olmalıdır. Gelecekte mümkün olan değişiklik ve düzeltmeler ve pinlerin küçük boyutları göz önüne alınarak, herhangi bir pine en fazla iki kablo bağlanmalıdır. Sinyal parazitlenmesini önlemek ve kablolanmanın kolaylık ve düzenini artırmak için toplam kablo uzunluğu minimize edilmelidir.

Bu problemin bir satıcı problemi olduğu, her bir pinin bir grafiğin bir noktasına ve $d(x, y)$ 'nin iki pin arasındaki mesafeye karşılık geldiği göz önüne alındığında kolayca görülmektedir. Eğer bir pine en fazla iki kablo bağlanabilir kısıtlaması ortadan kaldırılırsa, problem bir minimum mesafe ağacı bulunarak çözülebilir (Evans ve Minieka, 1992).

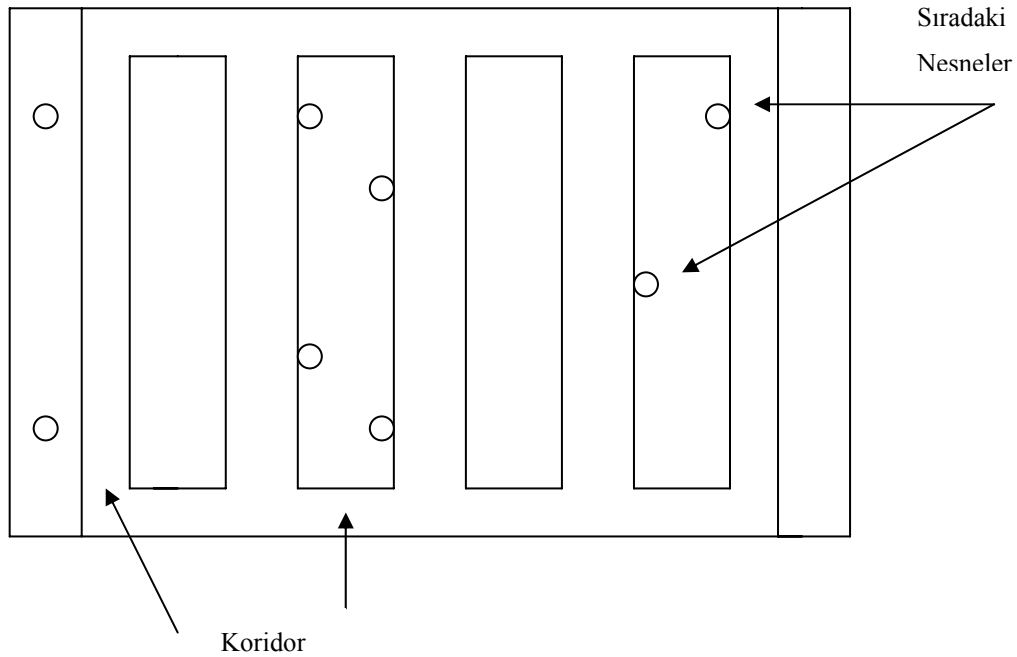
2.2.1.3 Devre Levhası Delimi (Magirou, 1986)

Metelco S.A. bir Yunan basılı devre levhası (PCB) üreticisidir. Delikler, PCB'de, daha sonra elektrik elemanlarının levhaya lehimleneceği pinlere karşılık gelecek yerlere açılmalıdır. Tipik bir PCB 500 pin yerleşimine sahiptir. Delme işleminin büyük kısmı, programlanabilir

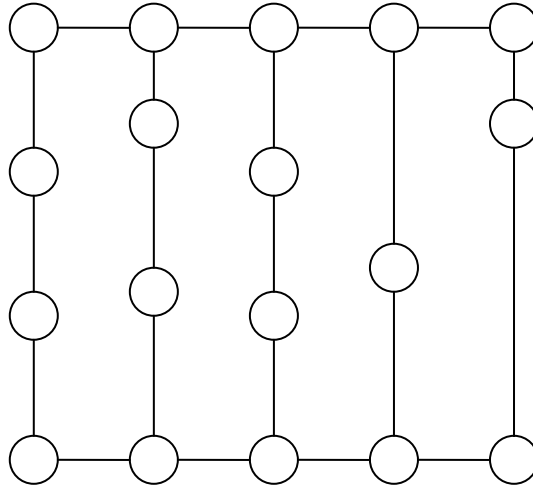
bir delme makineleriyle gerçekleştirilir. Optimum delme sırasını bulmak, bir gezgin satıcı problemi olarak ifade edilebilir. Metelco, bir PC’de PCB’ler için satıcı problemlerini çözerek levhaları delme için gereken süreyi %30 azaltmıştır (Evans ve Minieka, 1992).

2.2.1.4 Bir Ambarda Sipariş Toplama (Ratliff ve Rosenthal, 1983)

Bir ambarda depolanmış bir nesneler alt kümesini içeren sipariştir. Sipariş alırken, ambar siparişteki nesneleri almak için bir araç gönderir ve paketleme ve nakil bölgesine ulaştırır. Şekil 2.17 bir ambardaki koridor konfigürasyonunu göstermektedir. Kolayca görülmektedir ki, siparişin en kısa sürede alınması problemi, her bir noktanın bir nesnenin yerine veya her bir koridorun sonuna karşılık geldiği bir grafik tanımlandığında, bir satıcı problemidir. (Şekil.2.18) Bir sipariş alma turu, grafikteki nesne yerlerine karşılık gelen her bir noktayı en azından bir kez içeren bir dögüdür (Evans ve Minieka, 1992).



Şekil 2.17 Ambar koridor konfigürasyonu. (Evans ve Minieka, 1992)



Şekil 2.18 Sipariş toplanmasının grafik sunumu. (Evans ve Minieka, 1992)

2.2.1.5 İş Atölyelerinde Sıralama

Bir üretim tesisinde, tek bir makinede pek çok farklı iş görülmektedir. x işi için kurulum süresi, ondan hemen önce hangi işin yapıldığına bağlıdır. Bu, bir önceki iş için yapılan kurulumun yeni iş için de kullanılabileceği ve böylece daha fazla zaman kazanılacağı sonucunu vermektedir. Diğer durumlarda, daha uzun bir kurulum süresiyle sonuçlanan tamamen yeni bir kurulum gerçekleştirilmelidir. Bu problem, her bir iş grafiğinin bir noktasına ve $d(x, y)$ de y işinden hemen önce x işinin düzenlenmesi ile ilgili kurulum maliyetine karşılık gelecek şekilde bir satıcı problemi olarak formüle edilebilir. Kurulum maliyetini minimize etmek ve her bir işin gerçekleştirildiğini garanti etmek için grafikte bir minimum maliyetli satıcı turu aranır. Bu durumda, genellikle $d(x, y) \neq d(y, x)$ eşitsizliği doğru olacaktır. Bu nedenle, grafik her bir nokta çifti arasında farklı uzunlukta karşılıklı yönlendirilmiş yaylara sahip olacaktır. Bu bir asimetrik gezgin satıcı problemidir.

Araç rotalama durumlarında, gezgin satıcı formülasyonu, tek bir aracın rota boyunca tüm duraklara hizmet verebileceğini varsaymaktadır. Pratikte araçlar kapasite ya da zamanla kısıtlıdır. Örneğin, okul servisleri kapasite kısıtlıdır ve bir okul bölgesine hizmet vermek için pek çoğu gerekebilir. Aynı şekilde, 10:30 bitiş süresini karşılamak için pek çok paket dağıtım aracına gerek duyulabilir. Bu tür problemler araç rotalama ve programlama problemleri adı verilen daha genel bir problem sınıfına aittir (Evans ve Minieka, 1992).

2.2.2 Gezgin Satıcı Probleminin Temel Nitelikleri

Gezgin satıcı problemi kolay karakterize edilmesine rağmen çözümü çok zordur. GSP, NP-zor problem sınıfına girmektedir. Bu nedenle çözmek için yeterli bir algoritmanın geliştirilmesi pek mümkün değildir. Bununla beraber sadeliğinden (basitliğinden) ötürü GSP, bu sınıfta üzerinde en çok çalışılan problemlerden biridir.

En az toplam uzunluktaki bir satıcı döngüsü, optimum satıcı döngüsü olarak adlandırılır ve genel satıcı problemi için optimum çözümdür. En az toplam uzunluktaki bir Hamilton döngüsü de optimum Hamilton döngüsü olarak adlandırılır ve satıcı problemine optimum çözümdür. Tabi ki, yönlendirilmiş bir grafikte bir satıcı veya Hamilton döngüsündeki tüm yaylar aynı yöne yönelmiş olmalıdır. Yani, bir yönlendirilmiş döngü olmalıdır.

Bir optimum satıcı döngüsü, optimum Hamilton döngüsü olmak zorunda değildir. Örneğin, Şekil.2.19'teki grafiği göz önüne alalım. Bu grafikteki tek Hamilton döngüsü (1, 2), (2, 3), (3, 1) 'dır ve toplam uzunluğu $1 + 20 + 1 = 22$ birimdir. (Optimum) Satıcı döngüsü ise 1 noktasından iki kez geçmek üzere (1, 2), (2, 1), (1, 3), (3, 1) 'dır ve toplam uzunluğu $1 + 1 + 1 + 1 = 4$ birimdir. Buna göre, bir optimum satıcı döngüsü, optimum Hamilton döngüsü olmak zorunda değildir.

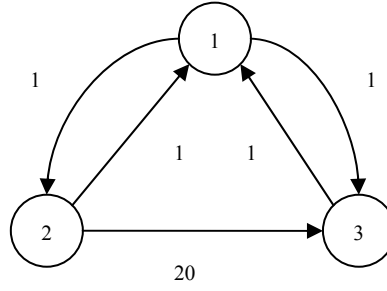
Aşağıdaki teorem ne zaman bir Hamilton döngüsü genel bir satıcı problemine çözüm olur sorusuna cevap vermektedir:

Teorem 1. Eğer G grafiğindeki her bir çift x, y noktası için,

$$d(x, y) \leq d(x, z) + d(z, y) \quad (\text{tüm } z \neq x, z \neq y \text{ 'ler için}) \quad (2.20)$$

ise, G grafiği için (eğer bir çözüm varsa) genel satıcı problemine optimum çözüm bir Hamilton döngüsüdür.

Koşul (2.20) sadece, x 'ten y 'ye direk mesafenin hiçbir zaman diğer herhangi bir z noktası yoluyla mesafeden daha fazla olmayacağını söylemektedir. Buna *üçgensel eşitsizlik* denir.



Şekil 2.19 Optimum satıcı rotası (Evans ve Minieka, 1992)

İspat. Genel satıcı problemine hiçbir optimum çözümün Hamilton döngüsü olmadığını varsayalım. C herhangi bir optimum satıcı döngüsü olsun. C bir Hamilton döngüsü olmadığından, herhangi bir nokta, örneğin z noktası, C döngüsünde en azından iki kez görünür. Saticının z noktasına ilk kez vardığında x noktasından geldiğini ve y noktasına ayrılacağını varsayalım. C döngüsünü satıcı x noktasından direkt olarak y noktasına geçecek ve z 'yi pas geçecek şekilde değiştirelim. Sonuç rotası C' her noktayı en azından bir kez dolaştığından ayrıca bir döngüdür. Üstelik (2.20)'ye göre, C' 'nün toplam uzunluğu C 'nin uzunluğunu aşmamaktadır. C ve C' 'nü yer değiştirerek ve bu argümanı tekrarlayarak başka bir satıcı döngüsü C'' 'nü oluşturabiliriz. En sonunda, bu süreç bizi, her ardışık döngü bir öncekinden bir az yaya sahip olacağından Hamilton olan bir optimum döngüye götürür. Teorem 1.e göre, eğer G grafiği üçgensel eşitsizliği sağlıyorsa, o zaman G grafiği için satıcı probleminin çözümleri aynı zamanda G grafiği için genel satıcı probleminin de optimum çözümleridir.

Bir tane genel satıcı problemi ve bir tane de satıcı problemi için olmak üzere iki tane çözüm tekniği geliştirmenin gereksizliğinden kurtulmanın basit bir yolu vardır. Eğer G grafiği üçgensel eşitsizliği sağlamıyorsa, üçgensel eşitsizliği bozan her bir (x, y) yay uzunluğu x 'ten y 'ye bir en kısa yol uzunluğu ile değiştirilir. x 'ten y 'ye yayın artık x 'ten y 'ye direk bir yol değil, x 'ten y 'ye en kısa yol boyunca bir yolculuğu ifade ettiğine dikkat edilmelidir. Artık, $d(x, y)$ üçgensel eşitsizliği sağlamaktadır.

Eğer G grafiği için satıcı problemine bir optimum çözüm, uzunluğu yukarıda belirtildiği gibi kısaltılan bir (x, y) yayı içeriyorsa optimum çözümde (x, y) yayı x 'ten y 'ye bir en kısa yol ile değiştirilir. Böylece, yalnızca satıcı problemi için çözüm tekniklerine ihtiyaç duyulur. Mesafeleri üçgensel eşitsizliği sağlayan her probleme GSP denebilir.

Örneğin, Şekil 2.19'da $d(2, 3) = 20 > d(2, 1) + d(1, 3) = 1 + 1$ 'dir. Böylece (2.20), (2, 3) yayı

için bozular. Eğer (2, 3) yayının uzunluğu 2 'ye, 2 'den 3 'e bir en kısa yol uzunluğuna düşürülürse sonuç grafiğinde tek Hamilton döngüsü uzunluğu $1 + 2 + 1 = 4$ olan (1, 2), (2, 3), (3, 1) olur. (2, 3) 'ü (2, 1), (1, 3) ile değiştirmek (1, 2), (2, 1), (1, 3), (3, 1) döngüsünü verir ki bu orijinal grafik için bir optimum satıcı döngüsüdür.

En küçük toplam uzunluktaki Hamilton döngüsünü bulmak yerine, en büyük toplam uzunluktaki Hamilton döngüsünü bulmak istediğimizi varsayalım. Örneğin, banka kuryesi, taşıdığı ek mektupların sayısını minimize etmek yerine maksimize etmek isteyebilir. Bu problem de bir satıcı problemi olarak çözülebilir mi?

M, G grafiğindeki en büyük yayı gösterebilir ve

$$d'(x, y) = M - d(x, y) \geq 0 \quad (\text{tüm } (x, y) \text{ 'ler için}) \quad (2.21)$$

olsun. $G = (X, A)$ grafiğindeki her Hamilton döngüsü tam olarak $|X|$ kadar yay içerecektir. İlk yay uzunluklarını baz alan bir optimum (en kısa) Hamilton döngüsü C,

$$\begin{aligned} \sum_{(x,y) \in C} d'(x, y) &= \sum_{(x,y) \in C} [M - d(x, y)] \\ &= |X| M - \sum_{(x,y) \in C} d(x, y) \end{aligned}$$

toplam uzunluğuna eşit olur. Böylece, sonuç olarak ilk yay uzunluklarına bağlı bir minimum uzunlukta Hamilton döngüsü, orijinal ilk olmayan yay uzunluklarına bağlı bir maksimum uzunlukta Hamilton döngüsüne karşılık gelir.

Bu nedenle, maksimum uzunlukta bir Hamilton döngüsü bulmak için sadece (2.21) 'deki gibi dönüştürülen yay uzunluklarını kullanarak satıcı problemini çözmek gerekir (Evans ve Minieka, 1992).

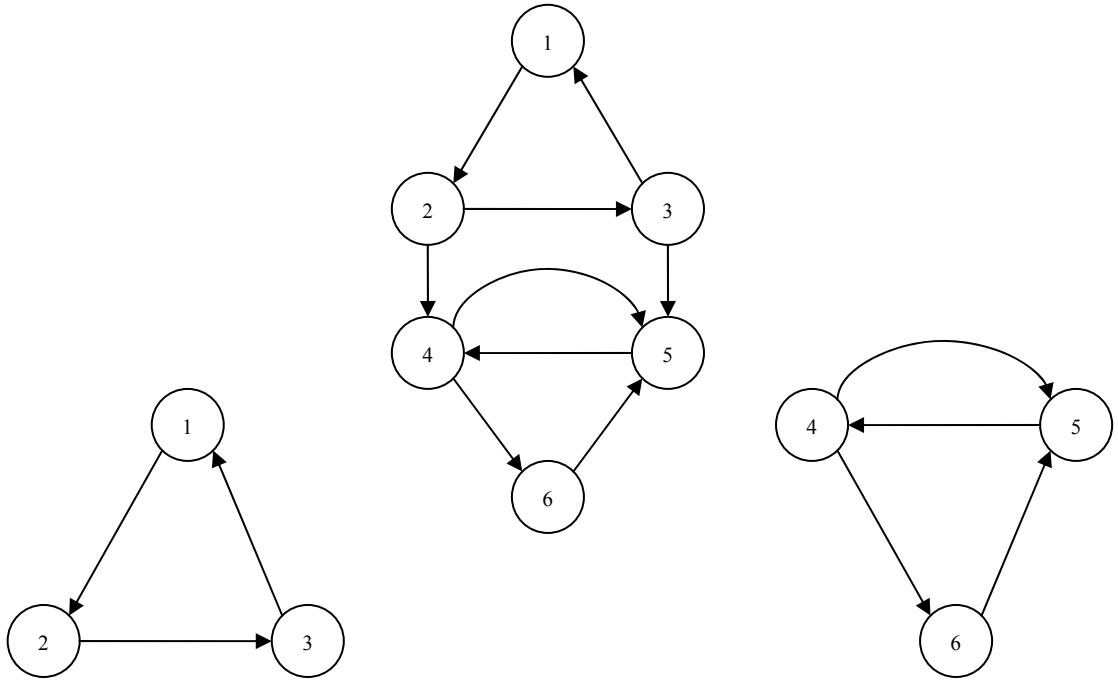
2.2.2.1 Bir Hamilton Döngüsünün Varlığı

Görüldüğü gibi, satıcı problemi optimum bir Hamilton döngüsü bulunarak çözülmüştür. Ne yazık ki, tüm grafikler bir Hamilton döngüsü içermez. Sonuç olarak, bir optimum Hamilton döngüsü aramaya başlamadan önce, en azından grafiğin herhangi bir Hamilton döngüsüne sahip olup olmadığı saptanmaya çalışılmalıdır. Bu bölüm bir grafiğin Hamilton döngüsüne sahip olduğu pek çok durumu tanımlamaktadır.

Grafikteki herhangi iki x ve y noktası için, eğer bu noktalar arasında direk bir yol mevcut ise

bu grafik *son derece bağlantılı* olarak adlandırılır. Eğer herhangi iki $x \in X_i$ ve $y \in X_i$ noktası için x 'ten y 'ye grafikte direkt bir yol varsa ve X_i aynı özellikteki başka kümede içerilmiyorsa noktaların X_i alt kümesi *son derece bağlantılı nokta alt kümesi* olarak adlandırılır. Bir son derece bağlantılı nokta alt kümesi tarafından oluşturulan alt grafik orijinal grafiğin son derece bağlantılı bir elemanı olarak tanımlanır.

Örneğin, 2.19'daki grafik her noktadan diğer her noktaya direkt yol olduğu için son derece bağlantılıdır. Şekil 2.20'deki grafiği göz önüne alalım. Bu grafik 4 noktasından 2 noktasına direkt bir yol olmadığı için son derece bağlantılı değildir. $\{1, 2, 3\}$ noktaları, her birinden bir diğerine yönlendirilmiş bir yol olduğu için bir son derece bağlantılı nokta alt kümesi oluşturmaktadır. Bununla birlikte, bu özellik yitirilmeden başka hiçbir nokta bu kümeye eklenemez. Örneğin, 4 noktası kümeye eklenemez çünkü 4 noktasından 1 noktasına yönlendirilmiş bir yol yoktur.



Şekil 2.20 Grafik ve onun son derece bağlantılı elemanları (Evans ve Minieka, 1992)

$\{1, 2, 3\}$ ile oluşturulan alt grafik Şekil.2.20'de gösterilmektedir. Bu alt grafik orijinal grafiğin son derece bağlantılı bir elemanıdır.

4'ten 5'e ve 5'den 4'e yönlendirilmiş bir yol mevcuttur. Ancak, $\{4, 5\}$ bir son derece bağlantılı nokta alt kümesi değildir çünkü 6 noktası son derece bağlantılı olma özelliği yitirilmeden bu kümeye eklenebilir. Diğer hiçbir nokta bu özellik kaybedilmeden eklenemez.

Bu yüzden, $\{4, 5, 6\}$ bir son derece bağlantılı nokta alt kümesidir. $\{4, 5, 6\}$ ile oluşturulan son derece bağlantılı nokta alt kümesi de Şekil 2.20’de gösterilmektedir.

Eğer G grafiği son derece bağlantılı değilse, bir Hamilton döngüsüne sahip değildir. Bir Hamilton döngüsü grafikteki tüm nokta çiftleri arasında yönlendirilmiş bir yol içerdiğinden bu sonuca varılır. Bundan dolayı, bir Hamilton döngüsünün varlığı için gerekli şart G grafiğinin son derece bağlantılı olmasıdır.

Bir çevrim iki ucu aynı noktada olan yaydır. Örneğin, (x, x) şeklindeki bir yay. Hiçbir Hamilton döngüsü çevrim içermez ve sonuç olarak bir Hamilton döngüsünün varlığı G grafiğinde çevrim eklenip silinmesiyle etkilenmez.

Eğer iki nokta, x ve y olsun, aynı yöndeki birden fazla yayla $(x, y)_1, (x, y)_2, \dots$, şeklinde bağlantılı ise, en kısa olan hariç, tüm bu yayların silinmesi grafikteki Hamilton döngüsünün varlığını veya optimum bir Hamilton döngüsünün uzunluğunu (eğer varsa) etkilemez.

Bu sebeplerden dolayı, bundan böyle G grafiğinin hiç çevrim içermediği ve her x ve y için x ’den y ’ye birden fazla yay içermediği varsayılacaktır.

$D^-(x)$, $G = (X, A)$ grafiğindeki $(y, x) \in A$ olmak üzere x ‘e gelen tüm noktalar olan tüm y noktalarını gösterebilir. $D^+(x)$ ise $(x, y) \in A$ olmak üzere x ‘ten çıkan tüm y noktalarını gösterebilir. $D^-(x)$ ve $D^+(x)$ sırasıyla x ‘in öncülleri ve ardılları olarak adlandırılır. $D(x) = D^-(x) \cup D^+(x)$ olsun. Önceden olduğu gibi, $d^-(x) = |D^-(x)|$, $d^+(x) = |D^+(x)|$ ve $d(x) = |D(x)|$ olsun. Son olarak, n , G grafiğindeki nokta sayısını gösterebilir (Evans ve Minieka, 1992).

Evans ve Minieka’ya (1992) göre bu tanımlarla Ghouila-Houri (1960)’nin bulduğu aşağıdaki çok genel teoremi açıklayabiliriz:

Teorem 2. Eğer $G = (X, A)$ grafiği aşağıdaki koşulları sağlıyorsa o zaman G grafiği bir Hamilton döngüsüne sahiptir:

- (I) G grafiği son derece bağlantılıdır
- (II) $d(x) \geq n$, tüm $x \in X$ için.

İspat. İspat, G grafiğindeki nokta sayısı olan n ‘e tümevarım ile elde edilmiştir. Önemsiz bir şekilde teorem $n = 2$ ve $n = 3$ için doğrudur. Teoremin n ($n > 3$)’den daha az sayıda noktalı tüm grafikler için doğru olduğu varsayılacaktır ve bu varsayım ayrıca n noktalı grafikler için

de teoremin doğru olduğunu göstermek için kullanılacaktır.

G , n noktalı ve (I) ve (II) şartlarını sağlayan herhangi bir grafik olsun. C , G içindeki mümkün olan en fazla sayıdaki yaya sahip basit bir döngü olsun. $x_1, x_2, \dots, x_m$ C 'nin G 'de ziyaret ettiği noktalar olsun. Eğer $m = n$ ise teorem doğrudur. Aksi halde, $m < n$ 'dir ve C bir Hamilton döngüsü değildir. $X_0 = \{x_1, x_2, \dots, x_m\}$ olsun. $X_1, X_2, \dots, X_p$, $X - X_0$ tarafından oluşturulan alt grafiğin son derece bağlantılı elemanları olsun.

İddia (a). Her bir son derece bağlantılı eleman $X_1, X_2, \dots, X_p$ bir Hamilton döngüsüne sahiptir.

İddia (a)'yı ispatlamak için yalnızca, tüm $i = 1, 2, \dots, p$ için her $x \in X_i$ noktasının X_i 'deki derecesinin en az $|X_i|$ olduğunu göstermek gerekmektedir. Daha sonra X_i elemanı koşul (I) ve koşul (II)'yi sağlayacaktır ve tümevarım hipotezine göre bir Hamilton döngüsü içerir.

$d(x) \geq n$ olduğunu biliyoruz. $j \neq i$ olmak üzere her bir $y \in X_j$ noktasını göz önüne alalım. (x, y) ve (y, x) yaylarının ikisi de A 'da mevcut bulunamaz. Aksi halde X_i ve X_j bir son derece bağlantılı eleman oluştururlardı. $k = 1, 2, \dots, m$ için (x_k, x) ve (x, x_k) mevcut olamaz, aksi halde C döngüsü x noktasını içermek için genişletilebilirdi. Sonuç olarak, X_i 'de bulunmayan bir x noktasına gelen yayların sayısı $|X| - |X_i|$ değerini aşamaz. Bu nedenle, X_i 'deki x 'in derecesi X_i 'den az değildir ve İddia (a) doğrudur.

İddia (b). $i = 1, 2, \dots, p$ için

$$|X_i| \leq |X_0| \text{ olur.}$$

Aksi halde, X_i 'de C 'den daha fazla yaya sahip bir Hamilton döngüsü olurdu.

İddia (c). X_0 'daki $x \in X_i$ olan her noktaya en azından $2 + |X_0| - |X_i| \geq 2$ tane yay gelmelidir.

İddia (a)'da bahsedildiği gibi X_j 'deki noktalarda tüm $j \neq i$ ve $j \neq 0$ lar için x noktasına $|X_j|$ 'den daha fazla yay gelemez. Ayrıca, X_i 'deki x noktasına $2(|X_i| - 1)$ yaydan fazlası gelemez. $d(x) \geq n$ olduğu için, bunun sonucu olarak X_0 'daki x noktasına en azından $2 + |X_0| - |X_i|$ yay gelir ve (c) doğrulanır.

İddia (d). X_i 'deki bir noktadan X_0 'daki bir noktaya ve X_0 'daki bir noktadan X_i 'deki bir noktaya bir yay olacak şekilde bir X_i elemanı mevcuttur.

$p = 1$ için İddia (d) doğru olmalıdır çünkü G son derece bağlantılıdır.

$p > 1$ için, X_1 ve X_0 'a gelen tüm yayların X_1 'den X_0 'a yönlendirilmiş olduklarını varsayalım. $X_1 \cup X_0$ 'la oluşan alt grafiği göz önüne alalım. X_1 'deki bir nokta, $X_1 \cup X_0$ 'da olmayan her bir noktaya en fazla bir kez geldiği için; X_1 'deki noktalar bu alt grafikte (II) koşulunu sağlar. İddia (a)dan anlaşılmaktadır ki X_0 'daki noktalar ayrıca bu grafikte (II) koşulunu sağlamaktadır.

G son derece bağlantılı olduğundan, $x \in X_0$ noktasından, $y \in X_1$ noktasına, bu yolun ara noktalarının hiç biri $X_1 \cup X_0$ üzerinde olmayacak şekilde bir yönlendirilmiş P yolu mevcuttur. P yolu en azından iki yay içermelidir.

Eğer (x, y) yayı $X_1 \cup X_0$ tarafından oluşturulmuş alt grafiğe eklenseydi bu alt grafik son derece bağlantılı olurdu. Sonuç olarak, (x, y) yayıyla birlikte bu alt grafik tümevarım hipotezini sağlardı ve (x, y) yayını içeren bu alt grafiğin bir Hamilton döngüsü olurdu. (x, y) yayını P yoluyla değiştirelim. Sonuç döngüsü C döngüsünden daha fazla yay içerir ve bu mümkün değildir. Bu nedenle, her $X_1, X_2, \dots, X_p$ elemanı iki yönlü yaylarla X_0 'a gelmelidir.

İddia (e). X_1 , İddia(d)'yi sağlayan bir eleman olsun. Her bir $y \in X_1$ noktası için, y noktasından X_0 'daki bir noktaya ve X_0 'daki bir noktadan y noktasına bir yay vardır.

C_1 , X_1 'in bir Hamilton döngüsü olsun. $y_1, y_2, \dots, y_q$ C_1 'in, X_1 'de ziyaret ettiği noktaların sırası olsun. [C_1 döngüsü, İddia (a)'dan mevcuttur.] X_0 'daki herhangi bir noktadan y_j noktasına bir yay olmadığını varsayalım. C_1 döngüsü, $y_{j+1}, y_{j+2}, \dots$ boyunca y_{j+t} noktasına kadar izlendiğinde, X_0 'da meydana gelen bir (x_k, y_{j+t}) yayıyla karşılaşılır. y_{j+t-1} noktasını göz önüne alalım. y_{j+t-1} 'den $x_{k+1}, x_{k+2}, \dots, x_{k+q}$ 'ya hiçbir yay yönlendirilemez. Aksi halde, C döngüsü y_{j+t} 'den y_{j+t-1} 'e $q - 1$ kadar yayla, mümkün olmayacak şekilde m 'den daha fazla yayla birleşebilirdi. Sonuç olarak, X_0 'dan y_{j+t-1} 'e hiç yay yönlendirilmemiştir ve y_{j+t-1} 'den X_0 'daki noktalara en fazla $m - q$ kadar yay yönlendirilmiştir. $m = |X_0|$ ve $q = |X_1|$ olduğundan bu İddia (c) ile çelişmektedir ve İddia (e) geçerli olmalıdır.

İddia (f) (Sonuç). Daha önce olduğu gibi, $x_1, x_2, \dots, x_m$ C tarafından ziyaret edilen noktaların sırasını ve $y_1, y_2, \dots, y_q$ C_1 tarafından ziyaret edilen noktaların sırasını gösterebilir.

$$a(i, j) = \begin{cases} 1 & (x_i, y_j) \in A \\ 0 & \text{aksi halde} \end{cases}$$

$$b(i, j) = \begin{cases} 1 & (y_{j-1}, x_{i+1}) \in A \\ 0 & \text{aksi halde} \end{cases}$$

olsun. İddia (c) 'ye göre X_0 ve X_1 'e gelen en az $q(m - q + 2)$ kadar yay olduğuna göre sonuç olarak;

$$\sum_{ij} a(i, j) + b(i, j) \geq q(m - q + 2)$$

olur. Bazı j_0 için,

$$\sum_i a(i, j_0) + b(i, j_0) \geq m - q + 2 \quad (2.22)$$

olur. İddia (e) 'ye göre X_0 'dan y_{j_0} noktasına yönlendirilmiş en azından bir (x_{i_0}, y_{j_0}) yayı vardır. Sonuç olarak, m tane yaydan daha fazlasına sahip bir döngü bulunmadığı için; $b(i_0, j_0) = 0, b(i_0 + 1, j_0) = 0, \dots, b(i_0 + q - 1, j_0) = 0$ sonucu çıkar. Bu nedenle, y_{j_0-1} 'den X_0 'daki noktalara en fazla $m - q + 1$ tane yay vardır. Eğer, $a(i_0 + 1, j_0) = 1$ ise $b(i_0 + q, j_0) = 0$ olur. Eğer $a(i_0 + 2, j_0) = 1$ ise $b(i_0 + q + 1, j_0) = 0$ olur. Bu nedenle, X_0 'dan y_{j_0} 'a yönlendirilmiş her bir ek yayın varlığı, y_{j_0} 'dan X_0 'a en azından bir tane daha yayın varlığını engeller.

Sonuç olarak,

$$\sum_i a(i, j_0) + b(i, j_0) \leq m - q + 1$$

olur ki bu da eşitsizlik (3) ile çelişir. Bu nedenle, C döngüsü yalnızca $m < n$ tane yay içeremez, bu bir çelişkidir.

Bir grafiğin Teorem 2’de verilen koşul (I) ve koşul (II)’yi sağladığını doğrulamak kolaydır. Koşul (I), grafikteki her bir nokta çiftine gelen sonlu uzunlukta bir yok olup olmadığını ortaya çıkarmak için Floyd veya Dantzig en kısa yol algoritmaları uygulanarak doğrulanır. Koşul (II) grafikteki her bir noktaya gelen yaylar sayılarak doğrulanır.

Eğer G grafiği yönlendirilmemiş bir grafikse, her bir yayda belli bir yön belirtilmemiştir ve satıcı yay üzerinde her iki yönde de yol alabilir. Chvatal (1972)’in ortaya çıkardığı aşağıdaki sonuç, yönlendirilmemiş bir grafiğin bir Hamilton döngüsüne sahip olması için yeterli koşulu tanımlamaktadır (Evans ve Minieka, 1992).

Yine n , grafikteki noktaların sayısını belirtsin. Noktaların isimleri, $x_1, x_2, \dots, x_n$ olsun ve $d(x_1) \leq d(x_2) \leq \dots \leq d(x_n)$ olsun.

Teorem 3. Eğer $n \geq 3$ ise ve eğer

$$d(x_k) \leq k < \frac{1}{2}n \Rightarrow d(x_{n-k}) \geq n - k \quad \text{ise} \quad (2.23)$$

grafik $G = (X, E)$ bir Hamilton döngüsü içerir.

İspat. Teoremin yanlış olduğunu varsayalım; örneğin, G grafiği (2.23) koşuluna sağlasın fakat Hamilton döngüsü içermesin. G ‘de olmayan herhangi bir (x_i, x_j) kenarını göz önüne alalım.

Eğer bu kenarın eklenmesi bir Hamilton döngüsü yaratmıyorsa, bu kenarı G grafiğine ekleyelim. G grafiğine daha fazla kenar eklenemeyecek duruma gelene kadar bu işlemi tekrarlayalım. G grafiğine her bir yeni kenar eklenmesinden sonra da, eklenen kenar herhangi bir noktanın derecesini düşürmediği için Koşul (2.23)’ün hala sağlandığına dikkat edelim.

Son grafiğe $G^* = (X, E^*)$ diyelim. Şimdi bir çelişki elde etmek için G^* grafiği kullanılmalıdır. u ve v , $d(u) + d(v)$ mümkün olduğunca büyük olacak şekilde birbirine komşu olmayan iki nokta olsun. Genelliği kaybetmeden, $d(u) \leq d(v)$ olduğunu varsayalım. C , u ‘dan v ‘ye en uzun basit yol olsun. Bir Hamilton döngüsü yaratılmadan G^* grafiğine başka kenar eklenemediği için, C ‘nin X ‘teki tüm noktaları içerdiği sonucuna ulaşılır. $\bar{u} = u_1, u_2, \dots, u_{n-1}, u_n$, C ‘nin X ‘teki noktaları ziyaret etme sırasını gösterebilir.

S , u_{i+1} u ‘ya komşu olacak şekilde tüm u_i noktaları kümesini gösterebilir. T , v ‘ye komşu olan tüm noktaları gösterebilir. Hiçbir u_j noktası hem S ‘nin hem de T ‘nin bir üyesi olamaz; çünkü eğer olsaydı, $u_j, u_{j-1}, \dots, u_1, u_{j+1}, u_{j+2}, \dots, u_n, u_j$ döngüsü bir Hamilton döngüsü olurdu. Ayrıca, $S \cup T = \{u_1, u_2, \dots, u_{n-1}\}$ ‘dir. Bu nedenle, $d(u) + d(v) = |S| + |T| \leq n$ olur. Sonuç

olarak, $d(u) \leq \frac{1}{2}n$ olur.

Hiçbir u_j noktası hem S 'nin hem de T 'nin bir üyesi olmadığı için, eğer $u_j \in S$ ise u_j, v 'ye komşu değildir. $d(u) + d(v)$ 'nin maksimalliğinden $d(u_j) \leq d(u)$ sonucu çıkar. Bu nedenle, derecesi u 'nun derecesini aşmayan en az $|S|$ nokta vardır. $k = d(u)$ olsun. Böylece, $d(x_k) \leq k$ olur. Koşul (2.23)'e göre $d(x_{n-k}) \geq n - k$ sonucu çıkar. Bu nedenle, dereceleri en az $n - k$ kadar olan en az $k + 1$ tane nokta olmalıdır. $d(u) = k$ olduğu için u noktası bu $k + 1$ noktalarının hiç biriyle komşu değildir. Bu nedenle, u 'ya komşu olmayan bir w noktası vardır ve $d(w) \geq n - k$ 'dır. Sonuç olarak, $d(u) + d(w) > d(u) + d(v)$ olur ve bu bir çelişkidir. Sonuç olarak, G^* grafiği bir Hamilton döngüsü içermelidir.

Koşul (2.23)'ü doğrulamak basittir. Sadece yükselen derecelerine göre noktalar sıralanır ve ilk $\frac{1}{2}n$ nokta için Koşul(2.23) 'ün sağlandığı kontrol edilir (Evans ve Minieka, 1992).

2.2.3 GSP İçin Sezgisel Algoritmalar

Gezgin satıcı probleminin optimal bir şekilde çözümü çok zor olduğu için, birçok sezgisel geliştirilmiştir. Bu sezgisel prosedürler genellikle optimum çözümün yüzde birkaçı dahilinde çözümler sunar ve hızlıdır. Bu nedenle, gerçekçi boyutlardaki problemler için, sezgiseller pratik bir çözüm yaklaşımı sunar. Önerilen bir çok farklı sezgisel tanımlayabiliriz. Bunların birçoğu basit modifikasyonlarla asimetrik problemlere uygulanabilirler; bu nedenle incelememizi simetrik durumla sınırlamalıyız.

Gezgin satıcı problemi için sezgiseller belli başlı iki grup içinde sınıflandırılabilir: tur yapım sezgiselleri ve tur geliştirme sezgiselleri. Tur yapım sezgiselleri bir fizibil tur oluştururlar. Tur geliştirme sezgiselleri fizibil bir turla başlayıp onu geliştirmeye çalışırlar. Pratikte, önce bir başlangıç turu bulmak için tur yapım sezgiseli kullanılır, daha sonra buna bir tur geliştirme sezgiseli uygulanır.

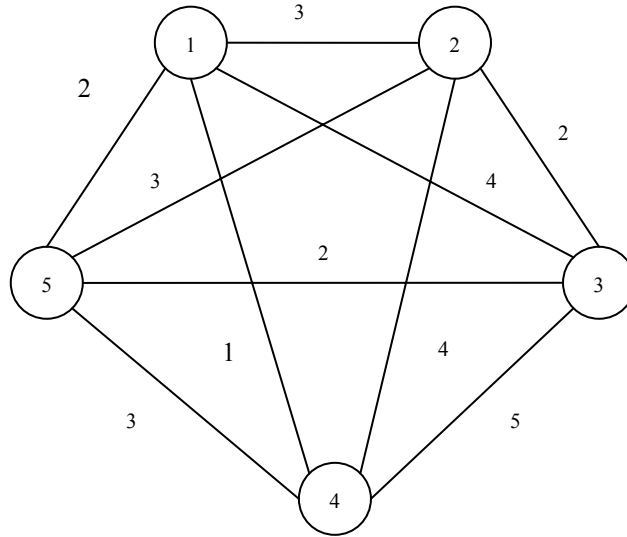
2.2.3.1 Tur Yapım Sezgiselleri

Bir tur oluşturmak için basit bir sezgisel, en yakın komşu sezgiselidir. Herhangi bir x noktası ile başlanır ve tüm y noktaları içinde $d(x, y)$ 'yi en kısa yapacak y noktası bulunur. Daha sonra, y 'ye henüz tura dahil olmayan en yakın nokta bulunur. z olsun. (z, y) kenarı tura eklenir. Bu işlem son nokta ekleninceye kadar sürdürülür ve sonra da ilk ve son noktalar

aralarındaki tek kenar ile birleştirilir.

Örnek:

En yakın komşu sezgiselini göstermek için aşağıdaki örneği kullanalım (Evans ve Minieka, 1992).



1 noktasıyla başlarsak sırasıyla, 4, 5, 3 ve son olarak da 2 noktalarını ekleriz. Son tur 1-4-5-3-2 olur. Sadece raslantı olarak bu optimal çözüm olur. Genellikle algoritma ilerledikçe seçilen kenarlar yetersiz olmaya başlar.

İkinci tür bir tur yapım sezgiseli *ekleme prosedürü*dür ve Rosenkrantz ve diğerleri tarafından 1974'de önerilmiştir. *En yakın eklemeyi* tanımlayalım. Başlangıç için bir nokta seçelim, i noktası olsun. En yakın noktayı seçelim, j olsun ve i - j - i alt turunu oluşturalım. Her tekrarda, alt turda olmayan, alt turdaki herhangi bir noktaya en yakın konumda olan bir k noktası bulalım. Alt turdaki, $d(i, k) + d(k, j) - d(i, j)$ değerini minimum yapan (i, j) kenarını bulalım. k noktasını i ve j arasına yerleştirelim. Bir tur oluşana kadar bu işlemi sürdürelim. Tekrarlama adımında, (i, j) kenarının kaldırılıp (i, k) ve (k, j) kenarlarının eklenmesiyle mevcut alt tura en az miktardaki mesafenin eklenmeye çalışıldığına dikkat edilmelidir.

Örnek:

Bir önceki örnekteki grafiği kullanarak en yakın ekleme prosedürünü gösterelim. 1 noktasıyla başladığımızı varsayalım ve 1-4-1 alt turunu oluşturalım. 1 veya 4'ün her ikisine de en yakın

olan nokta 5 noktasıdır. 5 noktasını ekleyerek, 1-4-5-1 alt turunu oluşturalım. Bu alt turdaki herhangi bir noktaya en yakın nokta 3 noktasıdır. 3 noktasını, 1 ve 4, 4 ve 5 veya 5 ve 1 noktaları arasına yerleştirmeyi göz önüne alalım. 3 noktasını 1 ve 4 noktaları arasına yerleştirmenin artan maliyeti, $d(1, 3) + d(3, 4) - d(1, 4) = 5 + 4 - 1 = 8$ olur. Benzer şekilde, 3 noktasını 4 ve 5 arasına yerleştirmenin artan maliyeti 4; 5 ve 1 arasına yerleştirmenin artan maliyeti ise yine 4 ‘tür. Sonuç eşitliği olduğu için keyfi olarak 4 ve 5 arasına yerleştirelim ve 1-4-3-5-1 alt turunu oluşturalım. Son olarak 2 noktasını göz önüne alalım. Hesaplamaların özeti aşağıdaki gibidir:

Yay	Artan Maliyet
(1, 4)	$3 + 4 - 1 = 6$
(4, 3)	$4 + 2 - 5 = 1$
(3, 5)	$3 + 2 - 2 = 3$
(5, 1)	$3 + 3 - 2 = 4$

2 noktasını 4 ve 3 noktalarının arasına yerleştirelim. Son tur 1-4-2-3-5 olur. Toplam dolaşılan mesafe 12’dir.

Ekleme prosedürünün diğer varyasyonları en uzak ekleme, en ucuz ekleme (burada artan maliyet yalnızca en yakın olan için değil tura dahil olmayan tüm noktalar için hesaplanır) ve keyfi eklemedir.

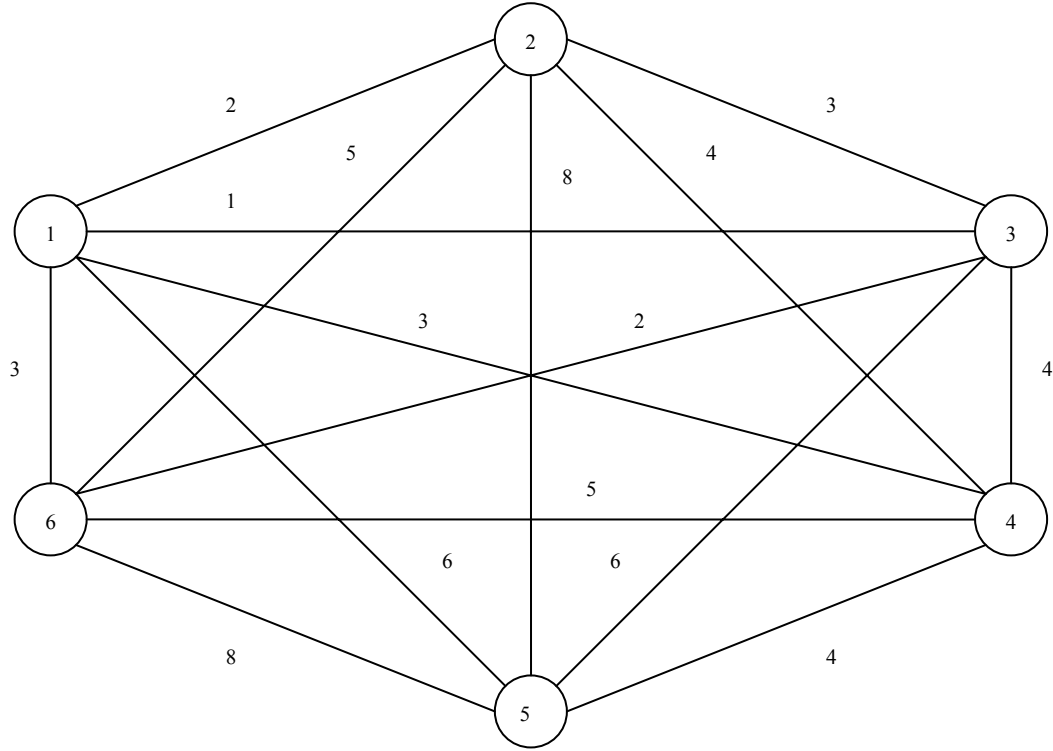
2.2.3.2 Christofides’in Sezgiselleri

Basit ve bir o kadar da akılcı bir sezgisel, Christofides tarafından 1976 yılında tanımlanmıştır (Evans ve Minieka, 1992). Bu yöntem üçgensel eşitsizliğin sağlanmasını gerektirir ve aşağıdaki şekilde devam eder:

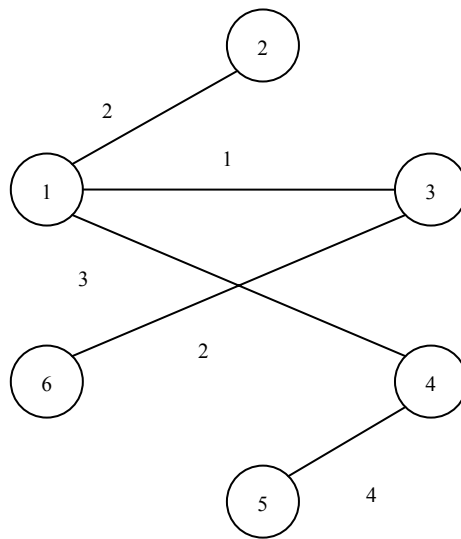
- Grafiğin minimum mesafe ağacı oluşturulur.
- Mesafe ağacındaki tek dereceli noktaların minimum maliyetli eşleşmeleri bulunur. Kenarlar, bir Euler Grafiği oluşturmak için optimal eşleşmeden ağaca eklenir.
- Bu grafikte bir Euler turu bulunur. Euler turu Hamilton döngüsüne dönüştürülür. Bu şu şekilde yapılır : $x_1, x_2, \dots, x_k$ Euler turunda ziyaret edilen noktaların sırası olsun. x_i ‘nin tekrar edilecek ilk nokta olduğunu varsayalım. x_j, x_i ‘den sonra Euler turunda tekrar edilmeyen ilk nokta olsun. $x_{i-1}, x_i, \dots, x_j, (x_{i-1}, x_j)$ tek kenarıyla yer değiştirilsin.

Örnek:

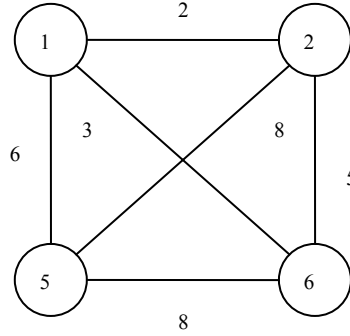
Aşağıda gösterilen GSP ‘yi göz önüne alalım.



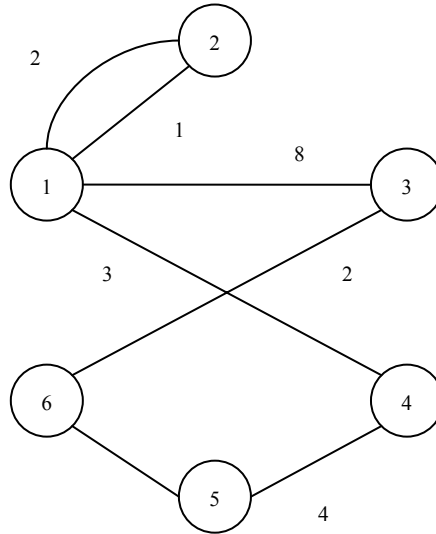
Minimal mesafe ağacı;



Tek dereceli noktalar 1, 2, 5 ve 6'dır. Bu noktalarındaki eşleşme problemi aşağıdaki gibidir.



Optimal eşleşme (1, 2) ve (5, 6) 'dır. Bu kenarların minimal mesafe ağacına eklenmesi bir Euler grafiği yaratır.



Bu grafikteki bir Euler turu 1-2-1-4-5-6-3-1 olur. 1 noktası tekrar edildiğinde, 2-1-4 yolu (2, 4) kenarı ile değiştirilir ve 1-2-4-5-6-3-1 Hamilton döngüsü oluşturulur.

Christofides'in sezgiseli 1,5 en kötü durum sınırını garanti eder. Bunun anlamı optimal çözümün değeri, sezgisel çözümün değerinde asla 1,5 kat daha fazla olamaz. Bunu göstermek için öncelikle aşağıdakini ispatlamalıyız.

Yardımcı Önerme 1. n noktalı bir GSP için, n çift sayı olmak üzere, minimum maliyetli mükemmel eşleşmenin değeri, $C(M^*)$, minimum maliyetli Hamilton döngüsünün, $C(H^*)$, yarısından fazla değildir; yani, $C(M^*) \leq 0.5C(H^*)$.

İspat. Minimum maliyetli Hamilton döngüsü H^* 'ın, $x_1, x_2, \dots, x_n$ nokta sıralamasıyla verildiğini varsayalım. Bu döngüyü, x_1 'den başlayarak kat edelim ve kenarları sırasıyla M_1 ve M_2 olmak üzere iki kümeye ayıralım. Böylece,

$$M_1 = \{(x_1, x_2), (x_3, x_4), \dots, (x_{n-1}, x_n)\}$$

$$M_2 = \{(x_2, x_3), (x_4, x_5), \dots, (x_n, x_1)\}$$

olur. Açık olarak, M_1 ve M_2 'nin birleşimi turdur ve M_1 ve M_2 'nin maliyetlerinin toplamı turun maliyetine eşittir. Daha sonra, iki eşleşme maliyetinden küçük olanı, tur maliyetinin yarısına eşit ya da küçük olmalıdır. Yani; $C(M^*) \leq \min \{C(M_1), C(M_2)\} \leq 0.5C(H^*)$.

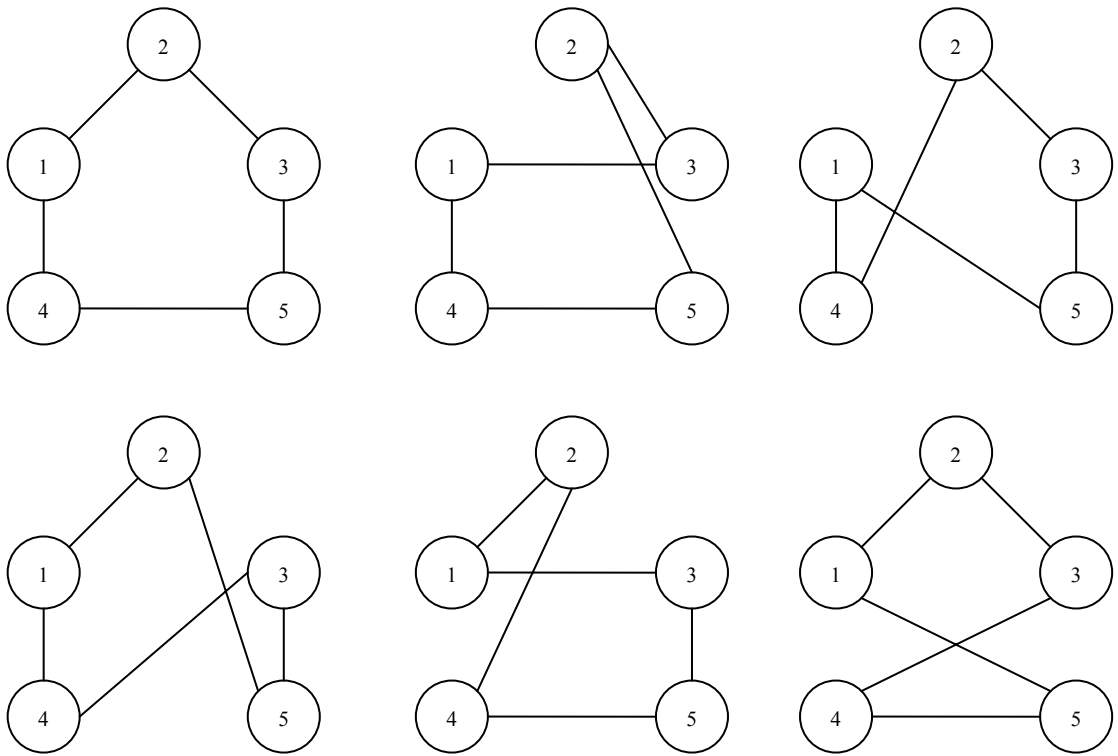
Ana sonucu ispatlamak için, herhangi bir Hamilton yolunun (grafığın tüm noktalarından geçen bir yol) bir mesafe ağacı olduğuna dikkat edilir. Açık olarak, minimum mesafe ağacının maliyeti $C(T^*)$, $C(H^*)$ 'den kesinlikle daha az olan $C(H_p)$ 'den az ya da ona eşit olmalıdır. M_T , T^* 'ın tüm tek dereceli noktalarının minimum maliyetli eşleşmeleri olsun. $M_T \cup T^*$, bir E Euler turu olduğu için, Euler turunun maliyeti $C(T^*)$ ve $C(M_T)$ 'nin maliyetlerinin toplamıdır. Yardımcı Önerme 1'den, H_T 'nin minimum mesafe ağacındaki tek dereceli noktalar kümesi tarafından oluşturulan alt grafikteki optimal tur olduğu $C(M_T) \leq 0.5C(H_T)$ ifadesine sahibiz. $C(H_T) \leq C(H^*)$ olduğu için $C(M_T) \leq 0.5C(H^*)$ 'dür. Bu sonuçları birleştirirsek, Euler turunun maliyetinin kesinlikle $1.5C(H^*)$ 'dan az olması gerektiği sonucuna varırız. Üçgensel eşitsizliğe göre kolayca görülmektedir ki, Euler turu tarafından yaratılan Hamilton döngüsünün maliyeti Euler turunun maliyetinden büyük olamaz. Bu nedenle, sezgisel tarafından oluşturulan Hamilton döngüsünün maliyeti, optimal turun maliyetinden 1.5 kez daha küçüktür.

Corneujols ve Nemhauser (1978), bu sınırı biraz daraltmışlardır ve m $0.5n$ 'den daha büyük olmayan en büyük tam sayı olmak üzere, sezgisel çözümün maliyetinin optimal turun maliyetinin $(3m - 1)/2m$ katından daha büyük olmadığını ispatlamışlardır (Evans ve Minieka,

1992).

2.2.3.3 Tur Geliştirme Sezgiselleri

En popüler tur geliştirme sezgiselleri 1965'te Lin tarafından bulunan ve 1973'te Lin ve Kernighan tarafından genişletilen k-opt sezgiselleridir (Evans ve Minieka, 1992). Bir turun bir k-değişikliği yeni bir tur oluşturmak için k kadar kenarın silinmesi ve onların diğer k kadar kenarla değiştirilmesini kapsar. 5 noktalı bir GSP'nin tüm muhtemel 2li değişiklikleri Şekil. 2.21'deki bir örnekle gösterilmiştir.

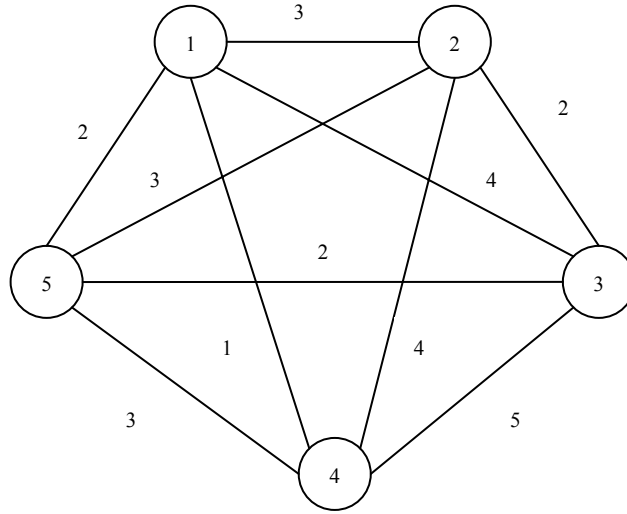


Şekil 2.21 5 Noktalı Bir GSP'nin 2'li değişiklikleri. (Evans ve Minieka, 1992)

Bu sezgisel prosedür herhangi bir fizibil turla başlar. Bu turdan tüm muhtemel k-değişiklikleri incelenir. Mevcut çözümden daha düşük maliyetli bir tur bulunursa, o yeni çözüm olur. Daha fazla k-değişikliği daha iyi sonuç vermeyinceye kadar işlem tekrarlanır. Algoritma durduğunda bir *yerel optimal sonucumuz* vardır. Tabi ki, sonuç çözümünün global olarak optimum çözüm olduğunun garantisi yoktur.

Örnek:

Daha önce kullanılan problemi tekrar kullanarak 2-opt sezgiseli gösterelim.



Keyfi olarak 1-2-3-4-5-1 turuyla başlayalım. Tüm 2li değişiklikler ve maliyetleri aşağıda listelenmiştir:

Çizelge 2.5 2-li değişimler ve maliyetleri (Evans ve Minieka, 1992)

Tur	Maliyet
1-3-2-4-5-1	15
1-4-3-2-5-1	13
1-2-4-3-5-1	16
1-2-5-4-3-1	18
1-2-3-5-4-1	11

2li değişikliklerin minimum maliyeti (11) mevcut çözümün maliyetinden (15) düşük olduğu için 1-2-3-5-4-1 turu yeni çözüm olarak seçilir. Bu yeni çözümün tüm 2li değişiklikleri hesaplanır:

Çizelge 2.6 Yeni çözüm için 2-li değişimler (Evans ve Minieka, 1992)

Tur	Maliyet
1-3-2-5-4-1	13
1-5-3-2-4-1	11
1-2-5-3-4-1	14
1-2-4-5-3-1	16
1-2-3-4-5-1	15

Hiçbir 2li değişiklik mevcut çözümden daha düşük maliyetli değildir, bu yüzden algoritma durur.

Genelde, 3lü değişikliklerin sayısı 2li değişikliklerin sayısını aşar; bu nedenle, 3lü değişikliğin 2li değişiklikten daha iyi bir sonuç bulma ihtimali oldukça yüksektir. Fakat, tüm 3lü değişiklikleri numaralandırmanın hesaplama maliyeti 2li değişiklikleri numaralandırmanın maliyetinden fazladır. Daha iyi bir sonuç bulmanın değeri, yüksek hesaplama eforuna karşı dengelenmelidir.

Tur yapım ve tur geliştirme algoritmaları çoğunlukla birlikte kullanılır. Örneğin, herhangi bir tur yapım prosedürü kullanılarak bir başlangıç turuyla başlanıp daha sonra buna 2-opt geliştirme sezgiseli uygulanabilir. Bir yerel optimum değere erişildiğinde, çözümü daha ileri götürmek için 3-opt sezgiseli uygulanır. Daha fazla geliştirme yapılamayacak duruma gelinceye kadar 2-opt ve 3-opt değiştirilerek kullanılır. Böyle bir prosedür oldukça etkilidir ve genellikle optimumun yalnızca birkaç yüzdelik dilimi içerisinde çözümler bulur.

2.3 Araç Rotalama Problemi

Malların dağıtımı; bir grup müşteriye, verilen bir zaman aralığında, bir ya da daha fazla depoda konumlanmış bir grup sürücü tarafından işletilen ve uygun bir yol ağını kullanarak hareketlerini gerçekleştiren bir grup araç tarafından hizmet sağlanması ile ilgilidir. Özellikle bir araç rotalama probleminin (ARP) çözümü, tüm müşteri gereksinimlerini yerine getirecek, tüm operasyonel kısıtları sağlayacak ve toplam taşıma maliyetini minimize edecek şekilde, her biri işlemine depodan başlayıp depoda bitirecek tek bir araç tarafından gerçekleştirilecek rota kümelerinin belirlenmesini gerektirir. Bu bölümde, ana komponentleri (yol ağı, müşteriler, depolar, araçlar ve sürücüler), rotaların yapımına empoze edilebilecek farklı

operasyonel kısıtlar ve optimizasyon sürecinde elde edilmesi mümkün olan hedefler üzerinde durularak rotalama ve çizelgeleme problemlerinin tipik karakteristikleri tanımlanmıştır.

Malların taşınması için kullanılan yol ağı, genellikle çizgileri yol bölümlerini ifade eden ve köşe noktaları da yol eklemlerine, depoya ve müşteri noktalarına karşılık gelen bir grafik üzerinden tanımlanır. Çizgiler (ve sonuçta ona karşılık gelen grafikler), sırasıyla tek bir yönde gitmeleri (örneğin tek yönlü sokaklar ya da çevreyolu gibi) ya da her iki yönde de gitmelerine bağlı olarak yönlendirilmiş veya yönlendirilmemiş olabilirler. Her bir çizgi, araç tipine ya da dolaşılma periyoduna bağlı olarak genellikle onun uzunluğunu ve bir dolaşım süresini gösteren bir maliyetle ilişkilidir.

Müşterilerin tipik karakteristikleri şu şekildedir;

- Müşterinin konumlandırıldığı yol grafiği üzerindeki noktalar;
- Müşterilere dağıtılan ya da müşterilerden toplanan, muhtemelen farklı tiplerde malların miktarı (talep);
- Müşteriye hizmet sağlanabilecek gün periyodu (zaman kısıtı), (örneğin, trafik sınırlamalarına bağlı olarak, müşterinin açık olduğu ya da mekanın erişilebilir olduğu spesifik periyotlar);
- Muhtemelen araç tipine bağlı olarak, müşteri konumunda malların dağıtılması ya da toplanması (sırasıyla yükleme ve boşaltma süreleri) için gereken süre ve
- Müşterilere hizmet verebilecek olan mevcut araçların alt kümesi (örneğin, muhtemel erişim sınırlamalarından ya da yükleme ve boşaltma gereksinimlerinden ötürü).

Bazen her bir müşterinin talebini tam olarak karşılamak mümkün değildir. Bu durumlarda, dağıtılacak ya da toplanacak miktar azaltılabilir veya bir müşteri alt kümesine hizmet verilmeyebilir. Bu durumların üstesinden gelmek için toplam ya da parçalı hizmet eksiklikleriyle ilişkili olarak müşterilere farklı öncelikler veya cezalar atanabilir.

Müşterilere hizmet vermek için gerçekleştirilen rotalar, yol grafiğinin noktaları üzerinde bulunan bir veya daha fazla depoda başlar ve biter. Her bir depo, onunla ilişkili olan araç sayısı ve tipiyle ve karşılayabileceği toplam mal miktarı ile tanımlanır. Bazı gerçek uygulamalarda, müşteriler önsel olarak depolar arasında bölüştürülmüştür ve araçlar her bir rotanın sonunda kendi depolarına dönmek zorundadırlar. Bu durumlarda, tüm ARP problemi, her biri farklı bir depoya alakalı olacak şekilde pek çok bağımsız probleme ayrıştırılabilir.

Malların taşınması, düzenleme ve boyutları sabit veya müşteri gereksinimlerine göre tanımlanabilecek bir araç filosu tarafından gerçekleştirilir. Araçların tipik karakteristikleri şu

şekildedir;

- Aracın *ana deposu* ve hizmeti ana depo harici bir depoda tamamlama olasılığı;
- Aracın yüklenebileceği toplam ağırlık, hacim veya palet sayısı açısından ifade edilen araç *kapasitesi*;
- Aracın, her biri kendi kapasitesi ve taşıyabileceği mal tipiyle tanımlanan muhtemel alt bölümlere (kompartımanlara) ayrılması;
- Yükleme ve boşaltma işlemleri için kullanılabilecek aygıtlar;
- Yol grafiğinin araç tarafından dolaşılacak alt yolları ve
- Aracın kullanımıyla ilişkili maliyet (birim mesafe başına, birim zaman başına rota başına vs.).

Araçları işleten sürücüler, sendika kontratları ve şirket düzenlemeleri ile ortaya çıkan çok sayıda kısıtı sağlamalıdır (örneğin, gün boyunca çalışma periyotları, hizmet boyunca verilen araların sayı ve süresi, dürtüş periyodunun maksimum uzunluğu, zamanaşımı).

Rotalar; taşınan malların doğasına, hizmet seviyesinin kalitesine ve müşteri ve araçların karakteristiklerine bağlı olan çok sayıda operasyonel kısıtları sağlamalıdır. Bazı tipik operasyonel kısıtlar şunlardır: her bir rota boyunca, ilgili aracın mevcut yükü araç kapasitesini aşamaz; bir rotada hizmet verilen müşteriler yalnızca malların toplanmasını ya da dağıtılmasını veya her ikisini birden talep edebilir; ve müşteriler yalnızca kendi zaman kısıtları içerisinde ve onları ziyaret eden araçlarla ilişkili sürücülerin çalışma periyotları içerisinde hizmet görebilirler. Öncelik kısıtları, ziyaret edilen bir rotadaki hizmet verilen müşteri sırasına empoze edilebilir. Öncelik kısıtının bir türü; aynı rotada verilen bir hizmet götürülmüş müşterinin diğer müşterilerin verilen bir alt kümesine hizmet etmesini ve müşterinin ilgili alt kümeye ait müşterilerden önce ya da sonra ziyaret edilmesini gerektirir. Bu *toplama ve dağıtma problemi* olarak anılan, rotaların hem toplama hem de dağıtım görevlerini gerçekleştirdiği bir durumdur ve toplanacak müşterilerden toplanan malların karşılık gelen dağıtılacak müşterilere dağıtımı yine aynı araçla sağlanmalıdır. Diğer bir öncelik kısıtı türü de; eğer aynı rota üzerinde farklı tipte müşterilere hizmet veriliyorsa, müşterilerin ziyaret edilme sırası sabittir. Bu durum, *Geri Dönüşlü ARP* adı verilen durumdur ve yine rotalar hem malların toplanması hem de dağıtımını gerçekleştirebilir, ancak yükleme ve boşaltma operasyonları ile ilgili kısıtlar ve rota boyunca araç yükünün yeniden düzenlenmesindeki zorluk tüm dağıtımın toplamadan önce yapılması gerektiğini göstermektedir.

Rotaların toplam maliyetinin hesaplanması ve onlara empoze edilen operasyonel kısıtların

kontrolü, müşteri çiftleri ve depolar ile müşteriler arasındaki dolaşım maliyeti ve dolaşım süreleri bilgisini gerektirir. Sıklıkla çok dağınık olan orijinal yol grafiği genel olarak, tepe noktaları yol grafiğinin tepe noktaları olan ve müşteriler ve depolara karşılık gelen toplu bir grafiğe dönüştürülür. Toplam grafiğin her bir nokta çifti i ve j için, maliyeti yol grafiğinde i noktasından başlayıp j noktasına varan *en kısa yolun* maliyeti ile verilen c_{ij} olan bir (i, j) yolu tanımlanır. Toplam grafiğin (i, j) yoluyla ilişkili t_{ij} dolaşım süresi, yol grafiğinde i 'den j 'ye en kısa yola ait yolların dolaşım sürelerinin toplamı olarak hesaplanır. Burada, orijinal yol grafiği yerine, karşılık gelen maliyet ve dolaşım süresi matrislerinin niteliklerinin asimetric ve simetric olmasına bağlı olarak düzleştirilebilen veya dolaylı hale getirilebilen toplam grafik göz önüne alınmıştır.

Araç rotalama problemi için birçok ve genellikle çelişen hedefler göz önünde bulundurulabilir. Tipik hedefler;

- Dolaşılan toplam mesafeye (veya toplam dolaşım süresine) ve kullanılan araçlarla (ve ilgili sürücülerle) ilişkili sabit maliyetlere bağlı olarak toplam taşıma maliyetinin minimizasyonu;
- Tüm müşterilere hizmet vermek için gerek duyulan araç (veya sürücü) sayısının minimizasyonu;
- Dolaşım süresi ve araç yükü için rotaların dengelenmesi;
- Müşterilerin kısmi hizmet görmesi ile ilgili cezaların minimizasyonu veya bu hedeflerin çeşitli ağırlıkta kombinasyonlarıdır.

Bazı uygulamalarda, her bir araç düşünülen zaman aralığında birden fazla rotada iş yapabilir veya rotalar 1 günden daha fazla sürebilir (Toth ve Vigo, 2000).

Gezgin satıcı problemi, araç rotalama problemleri sınıfının uygulamada önemi olan pek çok örneğinden yalnızca biridir. Daha genel bir durum şu şekilde gibi tanımlanabilir. Bilinen dağıtım gereksinimleri ve yerleri ile bir grup müşteri verilsin. Sınırlı kapasiteli bir kamyon filosu mevcuttur. Müşteriler, toplam zamanı ya da dolaşılan mesafeyi minimize etmek için farklı rotalara nasıl atanmalıdır? Örneğin, benzin taşıyan bir kamyon filosunun benzin istasyonlarına rotalanması ile ilgilenebiliriz. Her bir istasyon tankları doldurmak için sabit bir miktar benzine ihtiyaç duyar.

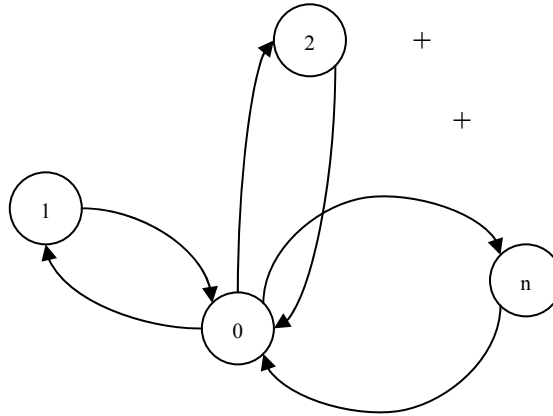
Her bir aracın sabit bir W kapasitesine sahip olduğunu varsayalım. $d(i)$, i noktasındaki talep ve $a(i, j)$ i noktasından j noktasına gitmek ile ilgili süre veya maliyet olsun. Ayrıca tüm araçların 0 noktası olarak tanımlanan merkezi bir bölgeden (depo) gönderildiğini varsayalım.

Bu problem için efektif bir sezgisel Clarke ve Wright (1963) tarafından geliştirilmiştir ve tasarruf yaklaşımı olarak da anılır (Evans ve Minieka, 1992). Prosedür, Şekil.2.22'deki gibi her müşterinin bağımsız olarak depodan hizmet gördüğü başlangıç çözümü ile başlar. Böyle bir çözüm, araç her müşteriye hizmet verdikten sonra depoya dönmek zorunda olduğundan oldukça etkisizdir. Gereksiz geri dönüş yolculuklarını yok etmek için bir yöntem alternatif rotaları birleştirmektir.

i ve j müşterilerine olan rotaların şekil 2.23'deki gibi birleştirildiğini varsayalım. Bu rotaların birleştirilmesiyle oluşan tasarruf;

$$s(i, j) = a(0, i) + a(0, j) - a(i, j) \text{ olur.}$$

Eğer tasarruf pozitif ise bu rotaları birleştirmeye değer demektir. Bu fikrin, GSP için ekleme sezgiseliyle aynı olduğuna dikkat edilebilir. Clarke – Wright algoritmasının adımları şu şekildedir:



Şekil 2.22 Clark – Wright algoritması için başlangıç çözümü. (Evans ve Minieka, 1992)

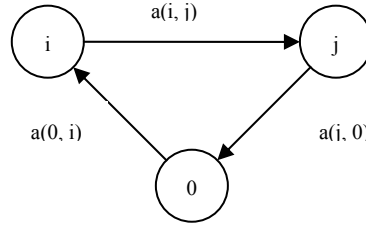
Adım 1. Her müşteri çifti için $s(i, j)$ tasarrufları hesaplanır.

Adım 2. En fazla tasarrufa sahip müşteri çiftleri seçilir ve birleştirilmelerinin fizibil olup olmadığı incelenir. Eğer öyleyse, bunlar birleştirilerek yeni bir rota oluşturulur. Eğer değilse, bu olasılık göz ardı edilir ve bir sonraki en yüksek tasarrufa sahip müşteri çifti seçilir.

Adım 3. Tasarruflar pozitif olduğu sürece Adım 2'ye devam edilir. Tüm pozitif tasarruflar göz önüne alındığında durulur (Evans ve Minieka, 1992).

Örnek :

Bir bira dağıtımıcısının bir sonraki gün dağıtım için yedi müşteriden sipariş aldığını varsayalım. Her bir müşteri tarafından gereksinim duyulan kasa sayısı ve her bir müşteri çifti arasındaki dolaşım süreleri tablolarda gösterilmiştir.



Şekil 2.23 i ve j müşterilerinin birleştirilmesi. (Evans ve Minieka, 1992)

Çizelge 2.7 Müşteriler ve talep değerleri (Evans ve Minieka, 1992)

Müşteri	Talep
1	46
2	55
3	33
4	30
5	24
6	75
7	30

Çizelge 2.8 Müşteriler ve depo arası dolaşım süreleri (Evans ve Minieka, 1992)

Dolaşım Süreleri								
	0	1	2	3	4	5	6	7
0	-							
1	20	-						
2	57	51	-					
3	51	10	50	-				
4	50	55	20	50	-			
5	10	25	30	11	50	-		
6	15	30	10	60	60	20	-	
7	90	53	47	38	10	90	12	-

Dağıtım aracı 80 kasalık kapasiteye sahiptir.

Yedi rotayı içeren başlangıç çözümü ile başlarız.

0-1-0

0-2-0

0-3-0

0-4-0

0-5-0

0-6-0

0-7-0

Eğer, örneğin, 1 ve 2 müşterilerini bağlarsak tasarruf aşağıdaki gibi hesaplanır:

$$s(1, 2) = a(0, 1) + a(0, 2) - a(1, 2) = 20 + 57 - 51 = 26$$

Her bir müşteri ile ilgili tasarruflar aşağıda listelenmiştir.

	1	2	3	4	5	6	7
1	-						
2	26	-					
3	61	58	-				
4	15	87	51	-			
5	5	37	50	10	-		
6	5	62	6	5	5	-	
7	57	100	103	130	10	93	-

İlk önce, en yüksek tasarruf seçilir – müşteri 4 ve 7. Müşteri 4 ve 7’yi birleştirmek fizibil bir rotayla sonuçlanır çünkü talep $30 + 30 = 60$ olur, bu da araç kapasitesinden daha azdır. Yeni rota kümesi şu şekildedir:

0-1-0

0-2-0

0-3-0

0-4-7-0

0-5-0

0-6-0

Daha sonra, bir sonraki en yüksek tasarrufa sahip olduklarından 3 ve 7 numaralı müşterileri birleştirilmeye çalışılır. Fakat bu rotadaki talep araç kapasitesini aşar, bu yüzden bu olasılık göz ardı edilir.

Tasarruf listesinde aşağılara doğru gidildiğinde, 2 ve 7, 7 ve 6, 2 ve 4 ve 2 ve 6 müşterilerinin birleştirilmesinin kapasite kısıtını bozduğu görülür. Bir sonraki en yüksek tasarruf olan 61, 3 ve 1 müşterilerinin birleştirilmesine karşılık gelmektedir ve fizibil bir rota verir. Devam edildiğinde, son rota kümesi şu şekilde elde edilir:

0-4-7-0

0-3-1-0

0-2-5-0

0-6-0

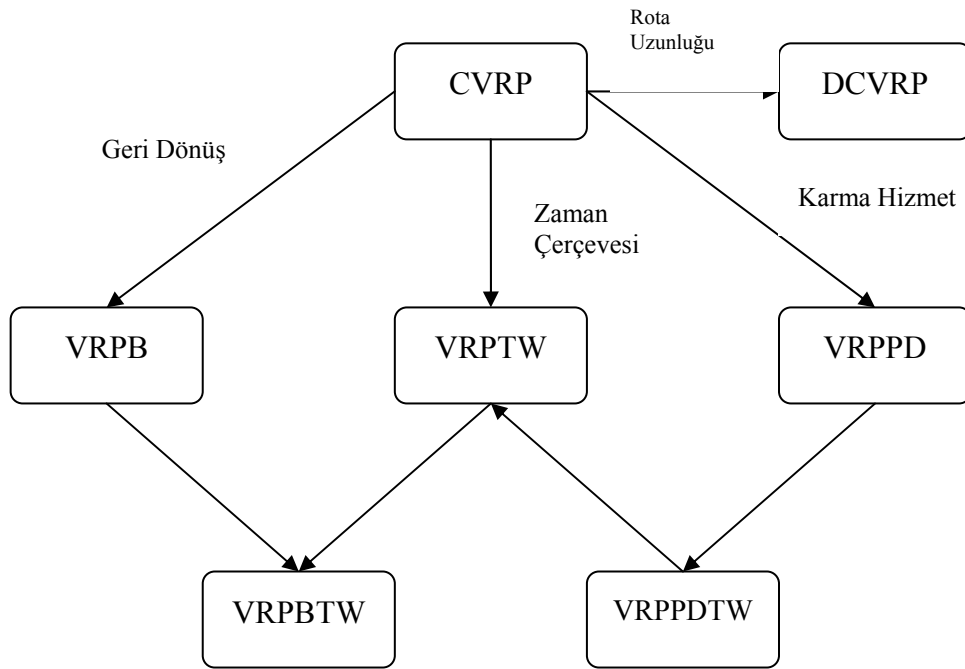
2.3.1 Temel Problem Tanımları

Bu bölümde araç rotalama sınıfının temel problemlerinin grafik teorik modeller olarak

tanımları verilmiştir. Bunlar sırasıyla *Kapasiteli ARP*, *Mesafe Kısıtlı ARP*, *Zaman Kısıtlı ARP* ve *Geri Dönüslü ARP* 'dir (Toth ve Vigo, 2000).

2.3.1.1 Kapasiteli ve Mesafe Kısıtlı ARP

KARP'de dağıtımlara karşılık gelen tüm müşteriler ve talepleri deterministiktir, önceden bilinmektedir ve bölünemez. Araçlar özdeştir ve tek bir merkezi depoda konumlanmışlardır ve araçlar için yalnızca kapasite kısıtlamaları empoze edilmiştir. Burada hedef tüm müşterilere hizmet vermek için toplam maliyetin (örneğin, rota sayılarının ve uzunluklarının veya dolaşım sürelerinin bir ağırlıklandırılmış fonksiyonu) minimize edilmesidir.



Şekil 2.24 ARP'nin temel problemleri ve ara bağlantıları (Toth ve Vigo, 2000)

KARP aşağıdaki gibi bir grafiksel teorik problem olarak tanımlanabilir.

$V = (0, \dots, n)$ nokta kümesi ve A da yol kümesi olmak üzere $G = (V, A)$ bir toplam grafik olsun. $i = 1, \dots, n$ noktaları müşterilere karşılık gelir, 0 noktası depoyu ifade eder. Bazen depo $n + 1$ noktası ile ifade edilir.

Negatif olmayan maliyet c_{ij} , her bir $(i, j) \in A$ yolu ile ilişkilidir ve i noktasından j noktasına dolaşım maliyetini verir. Genel olara (i, i) şeklinde yolların kullanılmasına izin verilmez ve her $i \in V$ için $c_{ii} = +\infty$ olarak tanımlanır. Eğer G dorulaştırılmış bir grafik ise, maliyet matrisi c asimetriktir ve problem asimetrik KARP (AKARP) adını alır. Aksi takdirde, tüm $(i, j) \in A$

için $c_{ij} = c_{ji}$ olur ve problem simetrik KARP (SKARP) adını alır ve yol kümesi A genel olarak bir dolaylı kenarlar kümesi, E ile yer değiştirir. Bir $e \in E$ kenarı verilsin ve $\alpha(e)$ ve $\beta(e)$ onun bitiş noktalarını gösterebilir. Burada kenarlar bitiş noktaları (i, j) , $i, j \in V$ 'nin ortalaması olarak gösterildiğinde dolaylandırılmış G grafiğinin kenar kümesi A ile ve kenarlar tek bir indis, e üzerinden gösterildiğinde E ile belirtilir (Toth ve Vigo, 2000).

2.3.1.2 Zaman Kısıtlı ARP

Zaman kısıtlı ARP, KARP'nin, kapasite kısıtları empoze edilmiş ve her bir i müşterisinin bir *zaman kısıtı* adı verilen bir $[a_i, b_i]$ zaman aralığıyla ilişkili olduğu genişletilmiş halidir. Araçların depodan ayrılma anı, her bir $(i, j) \in A$ (veya $t_e, e \in E$ için) için t_{ij} dolaşım süresi ve her bir i müşterisi için s_i servis süresi ayrıca verilir. Her bir müşterinin hizmeti ilgili zaman kısıtının içinde başlamalıdır ve araç müşteri konumunda s_i süresi için kalmalıdır. Bundan başka, i müşterisine erken varması halinde, aracın genellikle servisin başlayabileceği a_i anına kadar beklemesine izin verilir.

Normal olarak, maliyet ve dolaşım süresi matrisleri birbiriyle uyur ve zaman kısıtı tüm araçların 0 zaman anında depodan ayrıldığı varsayılarak tanımlanır. Ayrıca orijinal matrisin simetrik olması durumunda bile zaman kısıtı gereksinimlerinin her bir rotayla tam bir uyum sağladığı gözlemlenir. Bu nedenle *Zaman Kısıtlı ARP* normalde asimetrik bir problem olarak modellenir.

Zaman Kısıtlı ARP, minimum maliyetle tam olarak K basit çevrimin toplamının bulunmasından oluşur ve;

- Her çevrim depo noktasını ziyaret eder;
- Her müşteri noktası kesinlikle bir çevrim tarafından ziyaret edilir;
- Bir çevrim tarafından ziyaret edilen noktaların taleplerinin toplamı araç kapasitesi C 'yi aşmamalıdır;
- Her i müşterisi için hizmet $[a_i, b_i]$ zaman kısıtı içinde başlar ve araç müşteri noktasında s_i süresi kadar durur (Toth ve Vigo, 2000).

2.3.1.3 Geri Dönüslü ARP

Geri Dönüslü ARP, KARP'nin, $V \setminus \{0\}$ müşteri kümesinin iki alt kümeye ayrıldığı genişletilmiş şeklidir. İlk alt küme L , bir miktar ürünün dağıtılması istenen n tane *merkezden hedefe müşteri*'den oluşur. İkinci alt küme B , bir miktar gelen ürünün toplanması gereken m

tane *hedeften merkeze müşteri*'den oluşur. Müşteriler $L = \{1, \dots, n\}$ ve $B = \{n + 1, \dots, n + m\}$ şeklinde numaralandırılmıştır.

Geri Dönüşlü ARP'de merkezden hedefe ve hedeften merkeze müşteriler arasında bir öncelik kısıtı mevcuttur: eğer bir rota her iki tip müşteriye de hizmet veriyorsa, tüm merkezden hedefe müşteriler hedeften merkeze müşterilerden önce hizmet almalıdır. Türüne göre toplanacak ya da dağıtılacak bir negatif olmayan d_i talebi her bir i müşterisiyle ve depo da bir hayali $d_0 = 0$ talebiyle ilişkilendirilmiştir. Maliyet matrisi asimetrik olduğunda problem *Asimetrik Geri Dönüşlü ARP* adını alır. *Geri Dönüşlü ARP* (ve aynı zamanda *Asimetrik Geri Dönüşlü ARP*) minimum maliyetle tam olarak K basit çevrimin toplamının bulunmasından oluşur ve:

- Her çevrim depo noktasını ziyaret eder;
- Her müşteri noktası kesinlikle bir çevrim tarafından ziyaret edilmelidir;
- Bir çevrim tarafından ziyaret edilen merkezden hedefe ve hedeften merkeze müşterilerin toplam talepleri, ayrı ayrı olarak, C araç kapasitesini aşmamalıdır;
- Her bir çevrimde, eğer varsa merkezden hedefe müşteriler hedeften merkeze müşterilerden önce gelmelidir.

Yalnızca hedeften merkeze müşterilerden oluşan çevrimlere genellikle izin verilmez. Bundan başka, öncelik kısıtının “karışık” araç rotalarının (örneğin hem merkezden hedefe hem de hedeften merkeze müşterileri ziyaret eden rotalar) tam bir uyumunu ortaya çıkardığı gözlemlenir.

K_L ve K_B sırasıyla tüm merkezden hedefe ve hedeften merkeze müşterilere hizmet verebilmek için ihtiyaç duyulan minimum araç sayılarını gösterebilir. Bu değerler karşılık gelen müşteri alt kümeleriyle ilişkili BPP örnekleri çözülerek elde edilebilir. Fizibiliteden emin olmak için K 'nın tüm müşterilere hizmet verebilmek için gerekli olan minimum araç sayısından daha küçük olmadığı varsayılır, $K \geq \max (K_L, K_B)$ (Toth ve Vigo, 2000).

2.3.1.4 Toplama ve Dağıtma ile ARP

Toplama ve Dağıtma ile ARP'nin temel versiyonunda her bir i müşterisi, sırasıyla müşteri noktasında dağıtılacak ve toplanılacak homojen malların talebini temsil eden d_i ve p_i miktarlarıyla ilişkilidir. Bazen dağıtma ve toplama talepleri arasındaki net farkı gösteren (muhtemelen negatif olacaktır) tek bir talep miktarı $d_i = d_i - p_i$ kullanılır. Her bir i müşterisi için O_i , dağıtım talebinin merkezi olan noktayı ve D_i de toplama talebinin hedefi olan noktayı gösterir.

Her müşteri noktasında dağıtımın toplamadan önce gerçekleştirildiği varsayılmıştır. Bu nedenle bir aracın verilen bir noktaya varmadan önceki mevcut yükü, başlangıç yükünden tüm dağıtım ve toplama taleplerinin toplamının çıkartılmasıyla tanımlanır.

Toplama ve Dağıtım ile ARP minimum maliyetle tam olarak K basit çevrimin toplamının bulunmasından oluşur ve

- Her çevrim depo noktasını ziyaret eder;
- Her müşteri noktası kesinlikle bir çevrim tarafından ziyaret edilmelidir;
- Çevrim boyunca aracın mevcut yükü negatif olmamalı ve araç kapasitesi C 'yi aşmamalıdır;
- Her i müşterisi için, O_i müşterisi, depodan farklı noktalar için, i müşterisiyle aynı çevrimde ve i müşterisinden önce hizmet görmelidir;
- Her i müşterisi için, D_i müşterisi, depodan farklı noktalar için, i müşterisiyle aynı çevrimde ve i müşterisinden sonra hizmet görmelidir.

Çoğunlukla taleplerin merkezi veya hedefi ortaktır (örneğin KARP'ta ya da *Geri Dönüşlü ARP*'de olduğu gibi depoya ilişkilidir) ve bu nedenle onları açıkça belirtmeye gerek yoktur. Bu problem *Eşzamanlı Toplama ve Dağıtımli ARP* olarak adlandırılır (Toth ve Vigo, 2000).

2.3.2 Araç Rotalama Problemi İçin Tur Oluşturma ve İyileştirme Sezgiselleri

Araç Rotalama Probleminin çözümü için kullanılan sezgiseller tur oluşturma ve tur iyileştirme sezgiselleri olarak iki gruba ayrılır.

2.3.2.1 Tur Yapım Algoritmaları

Tur yapım algoritmaları Araç Rotalama Probleminin ilk çözümü için kullanılan sezgisel algoritmalarlardır. Bu bölümde En Kısa Yol Yöntemi ve Süpürme Algoritması incelenecektir.

En Kısa Yol Yöntemi

GSP için olduğu gibi tek işletme biriminin bulunduğu simetrik KARP'ne uygulanabilen bu yöntemin temel mantığı müşteri noktaları ve merkez depo arasındaki uzaklıklara ve aracın kapasitesine göre araçlara müşterilerin atanmasıdır. En kısa yol yönteminde süreç aşağıdaki gibi işlemektedir(Breedam, 2001):

- Merkez depodan rotalamaya başlanır. İlk araca depoya en yakın olan müşteri atanır.
- Rotaya atanan müşteriye en yakın olan ve daha önce rotaya eklenmemiş noktalar incelenir.

Eğer müşteriye en yakın iki nokta varsa her biri için süreç oluşturulur ve ayrı çözüm dalları yaratılır.

- Eğer müşteri direkt olarak merkezi birim ile bağlantılı değil ise (yani oluşturulan rotada müşteri merkezi birimden sonra gelmiyorsa) ve müşteriye en yakın başka müşteri ile merkezi depo aynı mesafede ise süreç yine ikiye ayrılarak yeni çözümler oluşturulur. Rotalamada ilk önce müşteri başka müşteriye araç kapasitesi aşılmıyorsa başlanır. Daha sonra çözüm ağacında yeni bir dal oluşturularak müşteri direkt olarak merkez depoya bağlanır.
- Çözümler ayrı ayrı hesaplanarak en uygun çözüm seçilir.

Süpürme Algoritması

1974 yılında Gillet ve Miller tarafından geliştirilen Süpürme Algoritmasında rotalarda yer alacak müşteriler, depo merkezli bir doğrunun döndürülmesi ile elde edilmektedir. Döndürme esnasında doğrunun üzerinden geçtiği müşteriler bir gruba ayrılır ve kapasite veya mesafe kısıtı aşıldığı zaman grup kapatılarak yeni bir grup ile devam edilir. Oluşturulan nokta gruplarına merkez depo da eklenip, genel olarak GSP gibi çözülerek rotalar belirlenir (Cordeau vd., 2002). Bu yöntemde müşterilerin koordinatları öklidyen formatta (x, y) değil, θ açı ve ρ doğru uzunluğunu göstermek üzere kutupsal formatta (θ_i, ρ_i) bulunur. Böylece talep noktaları θ açısına göre küçükten büyüğe sıralanıp, kapasite ve mesafe kısıtları dikkate alınarak gruplanır. Genel olarak süpürme algoritmasının çözüm adımları aşağıdaki gibidir (Laporte vd., 2000):

Adım 1. Bir harita üzerinde depo ve müşteri noktalarının yeri tespit edilir ve koordinatlar kutupsal formata (θ_i, ρ_i) çevrilir. Rotaya atanmamış herhangi bir araç belirlenir.

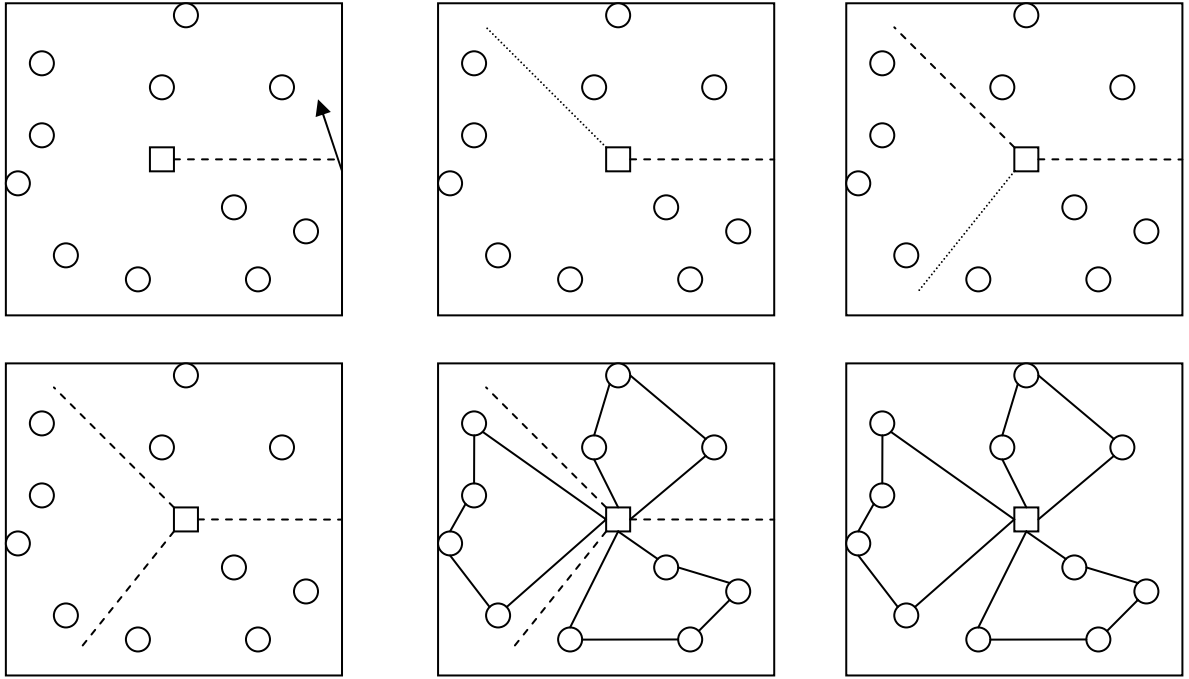
Adım 2. Depodan yatayla 0° açı ile başlanarak saat yönünün tersine doğru taramaya başlanır. Eğer bir müşteri ile karşılaşılırsa ve eğer müşterinin talep miktarı aracın kapasitesini geçmiyorsa müşteri araca atanır. Aksi takdirde saat yönünün tersi yönde hareket edilir. Eğer araca her iki yönde de müşteri atanamıyorsa, diğer araca geçilir.

Adım 3. Önceki aracın kaldığı yerden taramaya devam edilir. Eğer daha önce rotalanmayan müşteri ile karşılaşılırsa ve müşterinin araca atanması halinde kapasite kısıtı aşılmıyorsa, müşteri araca atanır. Bu süreç açıkta bir talep noktası kalmayıncaya kadar (tüm noktalar rotalanıncaya kadar) sürer.

Adım 4. Tüm noktalar araçlara atandıktan sonra gruplar uygun şekilde optimize edilerek rotalar belirlenir.

Süpürme algoritmasının adımları aşağıdaki 11 müşterili KARP üzerinde Şekil 2.25'te gösterilmektedir. Şekilde 0° açısından başlanarak saat yönünün tersi yönünde dönülmektedir. Kapasite veya mesafe kısıtı ihlal edildiğinde ise diğer gruba geçilmektedir. Daha sonra noktalar birleştirilerek rotalar oluşturulmaktadır.

Süpürme yöntemi tasarruf algoritmasına nazaran uygulanması daha basit olmasına rağmen, çoğu ARP için kesinlik ve hız bakımından Tasarruf algoritmasından geridedir. Bunun sebebi gruplama yapılırken noktalar arasındaki uzaklıkların dikkate alınmaması ve rotaların belirli bir alandan başlayarak oluşturulmasıdır. Ayrıca bu tekniğin yeni kısıtlarla entegre olması zor olmakla beraber iki boyutlu Öklidyen sistemde çözümler üretmesi uygulanabilirliğini kısıtlamaktadır (Cordeau vd., 2002).



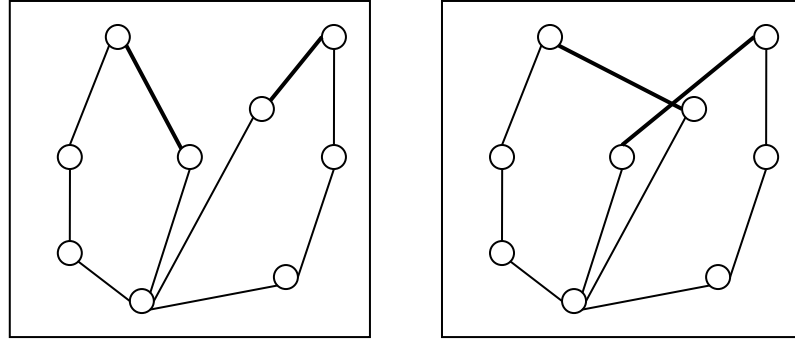
Şekil 2.25 Süpürme Algoritması Adımları

2.3.2.2 Çok Rota İyileştirmeli Sezgisel Algoritmalar

Çok Rota İyileştirmeli Sezgisel Algoritmelerde GSP'ye benzer şekilde bir ARP içinde yer alan rotalar arasında nokta alış verişi yapılmaktadır. Thompson ve Psarafitis'in 1993 yılında geliştirdikleri b-devirsel k-transfer yönteminde dairesel permutasyon ile sıralanan b rota ele

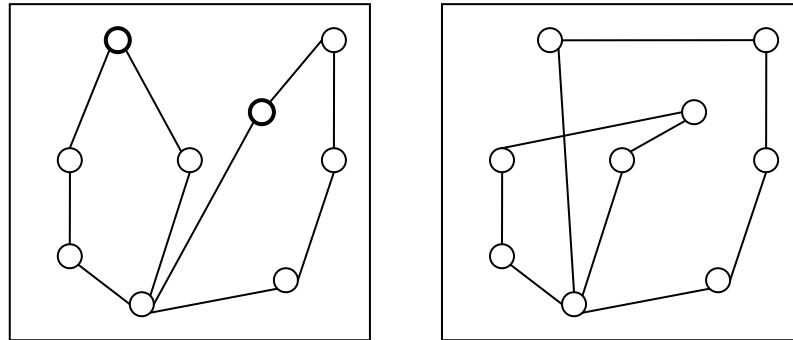
alınıp her rotadan k adet müşteri diğer rotaya atanarak yeni çözümler elde edilmektedir. Literatürde bu konuda yapılan çalışmaları Breedam toplayarak 2-devirsel yer değiştirme yöntemleri için aşağıdaki sınıflandırmayı yapmıştır (Van Breedam, 2001):

a) Yol Değişimi Yöntemi: Bu yöntemde iki rota arasında, rotalarda yer alan ardışık iki noktayı bağlayan bir yol (kenar) yer değiştirilerek yeni rotalar oluşturulur. Şekil 2.26'da bu yöntem ile ilgili bir örnek gösterilmektedir.



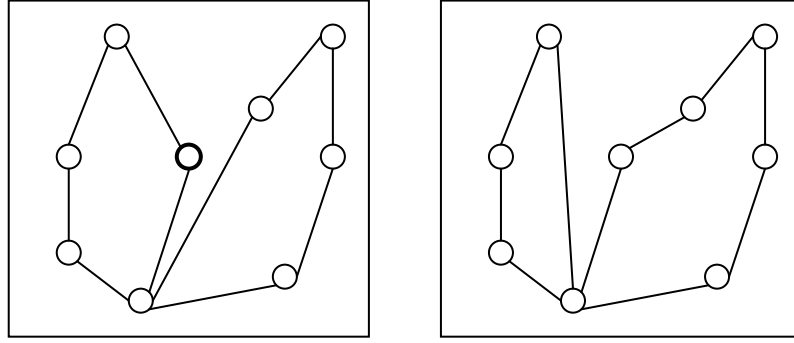
Şekil 2.26 Yol Değişimi Yöntemi

b) Nokta Değişimi Yöntemi: Mevcut iki rota arasında birer nokta yer değiştirilip, rota içinde uygun bir sıraya eklenerek yeni rotalar oluşturulmaktadır. Şekil 2.27'de yer değişimi yapılacak noktalar kalın olarak gösterilmektedir.



Şekil 2.27 Nokta Değişimi Yöntemi

c) Nokta Değişimi Yöntemi: Bu teknikte bir rota içersinden bir nokta seçilip diğer rotada en az maliyet artışı sağlayacak yere atanmaktadır. Bu yöntemle ilgili bir örnek Şekil 2.28'de gösterilmektedir.



Şekil 2.28 Nokta Atama Yöntemi

2.4 Stokastik Araç Rotalama Problemi

Bu bölümde Araç Rotalama Probleminin alt türü olan ve ARP'deki bazı parametrelerin rasgele olduğu durumlarda ortaya çıkan Stokastik Araç Rotalama Problemi (SARP) üzerinde durulacaktır.

2.4.1 Giriş

Klasik Araç Rotalama Problemi (ARP), önceden tanımlanan bir grafik üzerindeki müşteri kümesine hizmet vermek için bir araç filosunun kullandığı optimal rota kümesini saptar; hedefi toplam dolaşım maliyetini (dolaşım süreleri veya mesafe ile orantılı olarak) ve operasyon maliyetini (kullanılan araç sayısı ile orantılı olarak) minimize etmektir. Stokastik Araç Rotalama Problemi (SARP), ARP'nin bazı parametreleri rasgele olduğu zaman ortaya çıkar. Yaygın örnekler belirsiz müşteriler ve talepler ile stokastik dolaşım veya hizmet süreleridir. Bu bölümde, genel SARP kısaca tanımlanacak ve belirsizlik tipi, modelleme yöntemi ve çözüm tekniklerine göre sınıflandırılacaktır.

2.4.2 Stokastik Araç Rotalama

Araç Rotalama Problemi, $V = \{v_1, v_2, \dots, v_n\}$ noktalar kümesi ve $A \subseteq \{(v_i, v_j) : i \neq j, v_i, v_j \in V\}$ yay kümesi olmak üzere verilen bir $G = (V, A)$ grafiği ile tanımlanır. Yayların depoda başlayıp biten döngüsel bir bağlantısından oluşan optimal bir rota kümesi noktalarda verilen bir müşteri kümesine hizmet vermek için seçilir. Bu problem ilk olarak 1959'da Dantzig ve Ramser tarafından istasyonlara benzinin dağıtımı ile ilgili bir gerçek uygulamayı çözmek için ortaya atılmıştır. Stokastik Araç Rotalama (SARP)

sorgulanan sistem içerisindeki parametrelerde bazı belirsizlik öğeleri ortaya atar. SARP, deterministik karşılığında şu şekilde ayrılır: deterministik VRP'nin birçok temel özelliği stokastik durumda dikkate alınmaz ve çözüm yöntemleri bir hayli daha karmaşıktır. Stokastik Stokastik Araç Rotalama Problemi genel olarak aşağıdaki kriterlere göre sınıflandırılabilir:

Problemdeki Belirsizlik: Belirsizlik Araç Rotalama Probleminin farklı bölümlerinde ortaya çıkabilir. Stokastik müşteri ARP (SMARP), stokastik talepli ARP (STARP), stokastik dolaşım süreli ARP (SDSARP) ve stokastik hizmet süreli ARP (SHSARP) olarak ayrılabilir.

Modelleme Yöntemi: Modelleme yöntemi de Stokastik Araç Rotalama Problemlerini sınıflandırmada kriter olabilir. Stokastik ARP'ler daha sonra şans kısıtlı programa ve başvurmali stokastik programa ayrılmak üzere bir stokastik programlama yapısına dönüştürülebilir. SARP'yi modellemek için bir başka yaklaşım da onu bir Markov karar süreci olarak görmektir. Bazı daha güncel modelleme yaklaşımları, nörodinamik programlama/ takviye öğrenme metodolojilerini içerir.

Çözüm Teknikleri: Farklı çözüm teknikleri, farklı modelleme tekniklerinin ve modelin doğasının direkt sonucudur. Genel olarak iki kategoriye ayrılırlar: Kesin yöntemler ve sezgisel yöntemler. Kesin çözüm yöntemleri; dal ve sınır, dal ve dilim, tamsayı L-şekli yöntemi ve geliştirilmiş dinamik programlama içeren SARP'lere başarıyla uygulanmıştır. Bazı ılımlı varsayımlar altında, şans kısıtlı SARP'lerin pek çok sınıfı eşdeğer deterministik ARP'lere dönüştürülebilir. SARP'yi çözmek için, tasarruf algoritması, tabu araştırma vb. sayısız sezgiseller kullanılmıştır. Sezgisel çözüm teknikleri genellikle yapıcı sezgiseller, geliştirme sezgiselleri ve metasezgiseller olarak ayrılabilir (Shen vd., 2006).

2.4.2.1 Stokastik Müşteri ve Talepli SARP

Bireysel dağıtım (toplama) noktalarındaki talepler rasgele değişkenler olarak davranırlarsa ARP stokastik talepli (STARP) olur. STARP için ilk algoritma Tillman (1969) tarafından, Clark ve Wright'ın tasarruf algoritması (1964) baz alınarak önerilmiştir. STARP üzerine diğer bir erken büyük katkı ise problemi modellemek için şans kısıtlı programlama ve başvurma yöntemlerini uygulayan Stewart ve Golden'dan (1983) gelmiştir (Shen vd., 2006). Daha sonraları, Dror (1986), tasarlanan bir rotanın yönünün beklenen maliyete etkisini göstermiştir (Shen vd., 2006). STARP konusundaki çalışmalara çok büyük bir katkı Bertsimas'tan (1988, 1992) gelmiştir (Shen vd., 2006). Bu çalışma STARP'yi çözmek için farklı başvuru politikalarıyla önsel tur yöntemini göstermiş ve çeşitli sınırlar, asimptotik sonuçlar ve diğer

teorik özellikleri türetmiştir. Bertsimas ve Simchi-Levi (1996) edinilen anlayışlar ve önerilen algoritmalar üzerine vurgu yaparak STARP'deki gelişmeyi incelemişlerdir (Shen vd., 2006). Geleneksel stokastik programlama yapısının yanında, tek aşamalı ve çok aşamalı stokastik modeller için bir Markov karar süreci STARP'yi incelemek için Dror ve diğerleri (1989, 1993) tarafından ortaya atılmıştır. Daha güncel olarak STARP için bir yeniden optimizasyon tipi rotalama politikası 2001 yılında Secomandi tarafından ortaya atılmıştır (Shen vd., 2006).

Müşterilerin deterministik taleplerinin ve bir p_i mevcut olma olasılığının bulunduğu Stokastik Müşterili ARP (SMARP) ve SMARP ve STARP'yi birleştiren stokastik müşterili ve talepli ARP (SMTARP) literatürde ilk kez 1985, 1987 ve 1988 yıllarında Jaillet, Jaillet ve Odoni ve Jezequel tarafından ortaya atılmıştır (Shen vd., 2006). Bertsimas (1988) daha sistematik bir analiz vermiş ve birçok özellikler, sınırlar ve sezgiseller sunmuştur (Shen vd., 2006). Gendreau ve diğerleri (1995, 1996) SMTARP için ilk kesin çözümü, L-Şekli yöntemini ve bir metasezgisel ile tabu araştırmayı sunmuştur (Shen vd., 2006).

Stokastik talepli veya müşterili ARP'nin bir başka araştırma yönü taleplerin zamanla çeşitlilik gösterdiği dinamik olarak değişen bir çevreyi olaya dahil etmiştir. Bertsimas ve Van Ryzin (1991, 1993) bu çalışmaya öncülük etmişler ve Dinamik Gezgin Tamirci Problemi olarak isimlendirmişlerdir (Shen vd., 2006). Çözmek için kuyruk oluşturma teorisini uygulamışlar ve hem hafif hem de ağır trafikte pek çok sevk politikasını analiz etmişlerdir. Onların sonuçlarını temel alarak Papastavrou ve diğerleri (1996, 1999) dallandırma sürecini kullanarak yeni bir rotalama politikası tanımlamış ve analiz etmişlerdir (Shen vd., 2006). Ayrıca bu problem için güncel olarak Çoklu Senaryo Yaklaşımı (Bent, Hentanryck; 2004) ve bekleme stratejileri (Branke, Middendorf, Noeth ve Dessouky; 2005) ortaya atılmıştır (Shen vd., 2006).

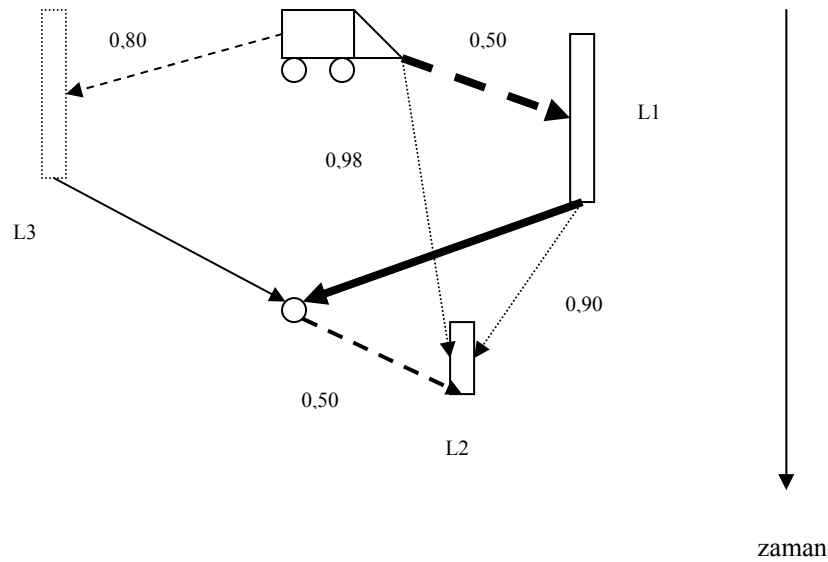
2.4.2.2 Stokastik Dolaşım ve Hizmet Süreli SARP

Stokastik müşteri ve taleplerle karşılaştırıldığında, ARP ve GSP için stokastik dolaşım ve hizmet süresi problemi daha az işlenmiştir. Stokastik dolaşım süreli ARP (SDSARP) yol trafik durumunun belirsiz ortamını tanımlar. İlk olarak Kao (1978) stokastik dolaşım süreli GSP için (SDSGSP) dinamik programlamaya ve kesin numaralandırmaya dayanan sezgiselleri önermiştir (Shen vd., 2006). Carraway ve diğerleri (1989) SDSGSP'yi çözmek için genelleştirilmiş bir dinamik programlama metodolojisini kullanmışlardır (Shen vd., 2006). Laporte, Louveaux ve Mercure (1992) stokastik dolaşım ve hizmet süreli ARP (SDHSARP) üzerine sistematik araştırmayı uygulamışlardır (Shen vd., 2006). SDHSARP için üç model önermişlerdir: şans kısıtlı model, 3 indisli başvuru modeli, 2 indisli başvuru modeli.

Her 3 model için de genel bir dal ve dilim algoritması sunmuşlardır. SDHSARP modeli bir bankacılık durumuna uygulanmıştır ve tasarruf algoritmalarının bir uyarlaması 1993 yılında Lambert, Louveaux ve Laporte tarafından kullanılmıştır (Shen vd., 2006). Julia ve diğerleri (2005) sıkı zaman çerçevesinin varlığında noktalara varış zamanlarını tahmin etmek için bir prosedür geliştirmişlerdir (Shen vd., 2006). Ek olarak, optimal rotaları elde etmek için bu tahminleri bir dinamik programlama algoritmasının içine yerleşik olarak kullanmışlardır. Hadjiconstantinou ve Roberts (2002) müşterilerden çağrı (hata raporu) alan ve teknisyenleri müşteri noktalarına gönderen bir şirket modellemek için stokastik hizmet süreli bir ARP formüle etmişlerdir. Bunu çözmek için iki aşamalı başvuru modeli ve eşlenmiş ağaç araştırma algoritması kullanmışlardır (Shen vd., 2006).

Stokastik Hizmet Sürelerinin Etkisi

Deterministik varsayımlar altında oluşturulan bir atamanın, stokastik ortamda alt-optimal olabileceği bir örnek gösterilebilir. Tüm atamaların deterministik önsel çözümler olduğu ve hedefin, hizmet maliyeti minimize edilerek mümkün olduğunca çok sayıda yüke hizmet vermek olduğu varsayılmıştır.



Şekil 2.29 Stokastik bir kurulumda deterministik bir atama. (Wang ve Regan, 2000)

Şekil 2.29’da gösterilen örnekte, kesikli çizgiler boş hareketleri gösterirken koyu çizgiler ise yüklü hareketleri göstermektedir. Çizgilerin yanındaki sayılar, yükün zaman penceresi sınır noktaları ile verilen son toplanma anından önce aracın varma olasılığını göstermektedir.

Diyagramda araçtan 3 yüküne olan mesafe, araçtan 1 yüküne olan mesafeden çok az bir değer daha uzundur. Aracın bu yüklere başarılı bir şekilde hizmet götürme olasılığı şekilde verilmiştir ve ağdaki dolaşım süresinin stokastikliğine dayanır. 3 ve 1 yükünün diğer tüm karakteristikleri aynıdır. Aracın 2 yükünden sonra 1 yüküne veya 2 yükünden sonra 3 yüküne hizmet verebileceği ancak üç yüke birden hizmet veremeyeceği varsayılır. Şekildeki tüm bağlantıları içeren deterministik model $L_1 L_2$ optimal atamasına götürür; bu da aracın 2 yükünden sonra 1 yüküne hizmet verdiği anlamına gelmektedir. Şimdi bu atama bilgileri altında sınır sürelerinden önce hizmet verilecek yüklerin beklenen sayısı hesaplanır.

$A = 1$ yüküne hizmet verilmesi durumu

$B = 2$ yüküne hizmet verilmesi durumu olsun.

Hizmet verilen yüklerin beklenen sayısının hesaplanması aşağıdaki gibidir:

$$\begin{aligned} E[L] &= P(A) + P(B) \\ &= P(A) + P(B \setminus A)P(A) + P(B \setminus \overline{A})P(\overline{A}) \\ &= 0.5 + 0.5 \times 0.5 + 0.5 \times 0.90 = 1.2 \end{aligned}$$

Beklenen değer, deterministik atamanın hedef değerinden önemli derecede farklı olduğu görülebilir (1.2'ye karşılık 2.0). Yukarıdakine göre daha aşağı bir deterministik ortamda alternatif bir atama, $L_3 L_2$; olduğunu göz önüne alalım. $L_3 L_2$ ataması, deterministik optimal atamadan daha yüksek olan 1.44 'lük bir beklenen değere sahiptir.

Örnekten de anlaşıldığı üzere, stokastik faktörlerin deterministik atamaların optimallliğini etkilediği sonucu çıkarılır (Wang ve Regan, 2000).

3. SARP İÇİN MODELLER VE ÇÖZÜM YÖNTEMLERİ

Literatürde SARP üzerine pek çok çalışma bulunmaktadır. Bu bölümde öncelikle bir literatür araştırması sunulup daha sonra da SARP üzerine yapılan çalışmalardan bazı detaylı örnekler üzerinde durulacaktır.

3.1 Literatür Araştırması

Bianchi ve arkadaşlarının 2005 yılında yaptığı “Stokastik Talepli Araç Rotalama Problemleri İçin Melez Metasezgiseller” (Hybrid Metaheuristics for the Vehicle Routing Problem with Stochastic Demands) çalışması, metasezgisellerin stokastik talep içeren araç rotalama problemi üzerindeki performansını analiz etmektedir. Burada, yakın ilişkili bir problemin (Gezgin Satıcı Problemi, Traveling Salesman Problem) hedef fonksiyonu araya dahil edilerek metasezgisel araştırma sürecinin melezleşmesi incelenmektedir. Test edilen örneklerin sonuçları incelendiğinde görülmektedir ki pek çok metasezgiseller – gezgin satıcı – hedef fonksiyonu ile melezlendiğinde performansları artmaktadır (Bianchi vd., 2005).

Yine Bianchi ve arkadaşlarının 2005 yılı çalışmaları “Stokastik Talepli Araç Rotalama Problemi İçin Metasezgiseller” (Metaheuristics for the Vehicle Routing Problem with Stochastic Demands) adlı makalelerine göre: Stokastik talepli araç rotalama problemlerinde hedef, araçla dolaşılması beklenen mesafeyi minimize edecek bir müşteri permutasyonu bulmaktır (a öncelik turu). Hedef fonksiyonu hesaplama açısından çaba gerektirse de efektif yaklaşımlar algoritmanın performansını artırabilir. Makale sonucunda görülmektedir ki; burada bulunan metasezgiseller, bilinen efektif bir sezgisel ve problemi iki ilişkili deterministik problem olarak çözme ile birlikte daha iyi sonuçlar elde etmiştir (Bianchi vd., 2005).

“Karmaşık İşletme Yönetimi Problemleri İçin İleri Araç Rotalama Algoritmaları” (Advanced Vehicle Routing Algorithms for Complex Operations Management Problems) adlı Tarantilis ve arkadaşlarının 2004 yılı çalışmaları, araç rotalama probleminin standart ve çok geniş çalışılmış versiyonu *Kapasiteli Araç Rotalama Problemi* için metasezgisel çözüm metodolojileri üzerine güncel araştırma çabalarını dikkatle incelemiştir. Her bir metasezgiselin hesaplamalı performansı Christafides ve arkadaşlarının 14 benchmark örnek durumu için gösterilmiştir (Tarantilis vd., 2004).

Cheung ve arkadaşlarının 2003 yılı makaleleri “Rasgele Hizmet Süreli Sürücü Görev Ataması için Bir Zaman Kısıtı Kaydırma Prosedürü” (A Time Window Sliding Procedure for Driver

Task Assignment with Random Service Times), konteynerlerin taşınması görevine sürücülerin atanması problemine işaret etmektedir. Görevler kesin zamanlarda başlamalıdır ve hizmet süreleri kararlar zaman içinde verildiğinden kesin değildir. Bu kararlar yeni bir bilgi edinildiğinde değişebilir. Problem, beklenen gelecek maliyetlerin ve mevcut sürücü görev atama maliyetlerinin minimize edilmesi hedefiyle bir stokastik optimizasyon yapısında şekillendirilmiştir. Bir zaman-kısıtı kaydırma çözüm prosedürü, minimum maliyet akış problemleri tekrarlamalı olarak çözülerek beklenen gelecek maliyetleri tahmin etmek için geliştirilmiştir (Cheung vd., 2002).

Hvattum, Lokketangen ve Laporte'nin ortak çalışması olan “Bir Stokastik Araç Rotalama Problemine Sezgisel Bir Çözüm Metodu” (A Heuristic Solution Method to a Stochastic Vehicle Routing Problem) adlı makalede, müşterilerin siparişleri günlük operasyonlar sırasında verdiği ve hem müşteri yerinin hem de talebin önceden bilinemeyebileceği göz önünde tutulmuştur. Bir sezgisel yaklaşım, bilinmeyen parametrelerin, bilinen bir yaklaşık dağılımı olması durumunda son uygulamada beklene araç sayısını ve dolaşım sürelerini minimize etmeye çalışmaktadır (Hvattum vd., 2003).

“Dinamik Dolaşım Süreli Araç Rotalama ve Çizelgeleme” (Vehicle Routing and Scheduling with Dynamic Travel Times) adlı 2004 yılında yayımlanan Potvin, Xu ve Benyahia çalışmasında, gerçek zamanlı müşteri istekleri ve dinamik dolaşım sürelerinin göz önünde bulundurulduğu zaman-kısıtlı bir dinamik araç rotalama ve çizelgeleme problemi tanımlanmıştır. Farklı tepkisel sevkiyat stratejileri tanımlanmış ve tek bir tolerans parametresinin ayarlanmasıyla karşılaştırılmıştır (Potvin vd., 2004).

Gupta'nın 2003 yılı çalışması olan “Araç Rotalama Problemlerinin Çözümü İçin Yeni Bir Algoritma” (A New Algorithm to Solve Vehicle Routing Problems) makalesine göre etkili kaynak yönetiminin iş performansı ve karlılık üzerinde direk bir etkisi vardır. Bu makaledeki araç rotalama problemleri, hedefin sınırlı sayıda kaynağın büyük sayıda iş için kullanılması ve böylece maksimum sayıda işin minimum maliyetle tamamlanması olduğu kaynak yönetimi problemleridir. Hedef, tamamlanan işler bazında ölçülmüş çalışma miktarının maksimize edilmesi ve tüm mühendisler tarafından dolaşılacak mesafenin minimize edilmesidir. Hill-Climbing (HC), Tabu Araştırma (TS), Genetik Algoritmalar (GA) ve Simüle Edilmiş Tavlama (SA) bu problemlerin etkili çözümü için uygun bulunmuştur. Bu makaledeki araç rotalama problemlerini çözmek için yeni bir algoritma geliştirilmiştir. Önerilen algoritma belli bir sayıdaki rasgele oluşturulmuş araç rotalama problemi üzerinde deneysel olarak test edilmiştir. Pek çok araç rotalama probleminde önerilen algoritma çok sayıda işi SA'ya göre

minimum maliyetle tamamlamıştır (Gupta, 2003).

Kim, Lewis ve White tarafından 2005 yılında ortaklaşa oluşturulan “Gerçek Zamanlı Trafik Bilgileri İle Optimal Araç Rotalama” (Optimal Vehicle Routing with Real-Time Traffic Information) çalışması, sabit olmayan stokastik bir ağda optimal araç rotalama için gerçek zamanlı trafik bilgisinin değerini araştırmaktadır. Gerçek zamanlı bilgi teknolojisi ile bütünleştirilmiş nakliyat sistemlerinin uygulamasında yardım için sistematik bir yaklaşım sunulmuştur. Bir Markov karar süreç formülasyonuna dayalı zaman değişkenli trafik akışı durumunda optimal sürücü hizmet süresi, optimal ayrılma süreleri, optimal rotalama hizmetlerine karar vermek için karar verme prosedürleri geliştirilmiştir (Kim vd., 2005).

“Bir Stokastik Zaman Değişkenli Ortamda Araç Rotalama için Dağınık Algoritmalar” (Decentralized Algorithms for Vehicle Routing in a Stochastic Time-Varying Environment) adlı Frazzoli ve Bullo’nun 2004 yılı çalışmasında; hedefi, stokastik olarak oluşan hedeflere hizmet için beklenen bekleme süresinin minimize edilmesi olan, bir grup özerk aracın hareket koordinasyonu için dağınık algoritmalar sunulmuştur (Frazzoli ve Bullo, 2004).

Mak ve Guo’nun 2004 yılı çalışması “Stokastik Talepli ve Esnek Zaman Kısıtlı Araç Rotalama Algoritmaları için Bir Genetik Algoritma” (A Genetic Algorithm For Vehicle Routing Problems With Stochastic Demand And Soft Time Windows), esnek zaman kısıtlı stokastik araç rotalama problemi üzerinde çalışmaktadır. Sınırlı kapasiteli araçlar, merkez bir depodan coğrafi olarak dağınık müşterilere, bilinmeyen talep, önceden tanımlanan yerinde bulunma olasılığı ve zaman kısıtlarıyla rotalanmıştır. Müşteri noktasına geç varmaya, hedef değerine bir ceza eklemek suretiyle izin verilmektedir. Bu tarz bir dağıtım sisteminin davranışını tanımlamak için bir matematiksel model geliştirilmiştir. Bir yeni çağ (novel age) tabanlı genetik çizelgeleme algoritması toplam maliyeti minimize ederek bu kontrol edilmesi güç araç rotalama problemini çözmek için bir optimizasyon aracı olarak önerilmiştir. Önerilen çizelgeleme algoritmasının efektifliği rasgele oluşturulmuş sayısal örnekler grubu kullanılarak gösterilmiştir. Sonuçlar göstermektedir ki önerilen genetik yaklaşım bu problemlerin çözümü için basit ama efektiftir (Mak ve Guo, 2004).

Swihart ve Papastavrou’nun 2004 yılı çalışmaları “Tek araçlı Toplama Ve Dağıtma Problemi İçin Stokastik ve Dinamik Bir Model” (A Stochastic And Dynamic Model For The Single-Vehicle Pick Up And Delivery Problem) de Toplama ve Dağıtma Problemi için stokastik ve dinamik bir model geliştirilip analiz edilmiştir. Burada hedef sistemdeki talepler için beklenen süreyi minimize etmektir. Analizler hem analitik hem de simülasyon metodları kullanılarak

yapılmıştır (Swihart ve Papastavrou, 2004).

“Kesin Olmayan Görev Süreleri İçeren Dinamik Sürücü Görev Ataması İçin Bir Etiketleme Metodu” (A Labeling Method For Dynamic Driver-Task Assignment With Uncertain Task Durations), Cheung, Hang ve Shi’nin 2004 yılı çalışmalarıdır. Bu makaleye göre, servis süresi açısından ve kesin olmayan görev süreleri içeren, sürücüleri görevlere atama problemi bir dinamik stokastik karar modeli olarak şekillendirilmiştir. Çeşitli pratik kısıtları ve iş kurallarını dahil eden bir uyarlanabilir etiketleme çözüm prosedürü geliştirilmiştir (Cheung vd., 2004).

Haughton’un 1998 yılında yayımlanan “Stokastik Araç Rotalamada Rota Modifikasyonu ve Talep Dengelenmesi Stratejilerinin Performansı” (The Performance Of Route Modification And Demand Stabilization Strategies In Stochastic Vehicle Routing) çalışmasına göre, stokastik talepler altında araç rotalama operasyonlarının yönetimindeki anahtar endişe, dolaşım süresinin temelinde, rota modifikasyonunun sabit rotalara göre maddi olarak daha büyük lojistik verim sağlayıp sağlamadığıdır. Bu makale, lojistik verimdeki bu farklılığın tahmin edilmesi için sağlam ve kolay kontrol edilebilir bir model geliştirebilmek için istatistiksel kalibrasyonu birincil teknik olarak kullanmaktadır (Haughton, 1998)

Yine Haughton’un 2002 yılındaki çalışması “Rota Yeniden Optimizasyonunun Dağıtım Verimliliği Üzerine Etkisi” (Route Reoptimization’s Impact On Delivery Efficiency)’nin birincil katkıları, sürücülerin birden fazla rota atamalarını öğrenmeleri sonucu oluşacak dengesizliklerin sonuç gereksinimlerinin değerlendirilmesi için sağlam bir ölçü sistemi geliştirmesi, ölçmesi ve modellemesidir. Araştırmanın anahtar sonucu bu ölçü sisteminin davranışını modellemek için regresyon analizinin kullanılabilecek olmasıdır (Haughton, 2002).

2002 yılına ait “Hizmet Seviyesi Göz Önünde Bulundurularak Tedarik Zinciri Lojistik Sisteminin Tasarımı” (Design Of Supply-Chain Logistics System Considering Service Level) adlı Hwang çalışması, hem ambar ve dağıtım merkezlerinin sayısında hem de araç rotalama çizelgesinde gerekli hizmet seviyesine lojistik sisteminin performansını optimize edecek lojistik sistemi tasarımı ile ilgilidir. Bu makalede lojistik sistemi fabrika, W/D (ambar/depo) ve müşterileri içermektedir. Bu problem ilk olarak stokastik *set-covering* problemi kullanılarak şekillendirilmiştir. Daha sonra gelişmiş bir genetik algoritma kullanılarak bir araç rotalama problemi oluşturulmuştur (Hwang, 2002).

Fu’ya ait 2002 yılı çalışması “Zaman Değişkenliği Altında *Dial-a-Ride Paratransit*

Çizelgeleme, Stokastik Tıkanıklık” (Scheduling Dial-a-Ride Paratransit Under Time-Varying, Stochastic Congestion), dar hizmet süresi kısıtları ve zaman değişkenliği, stokastik trafik tıkanıklığına özne olan *paratransit* hizmet sistemlerinde ortaya çıkan *dial-a-ride paratransit* çizelgeleme problemleri üzerinde bir çalışmayı tartışmaktadır. Var olan metodolojilerden farklı olarak bir zaman bağımlı, stokastik dolaşım süresi modeli problem formülasyonuna dahil edilmiştir (Fu, 2002).

“Stokastik Rotalama Uygulamalarıyla Sıralama Problemlerine Bir *Rollout* Yaklaşımının Analizi” (Analysis Of A Rollout Approach To Sequencing Problems With Stochastic Routing Applications) adlı Secomandi’ye ait 2003 yılı çalışması, gezgin satıcı probleminin sıralama problemlerinin doğal temsilcisi olduğunu göz önüne almaktadır. Ana temel algoritma olarak bir döngüsel sezgisel kullanan bir *rollout* yaklaşımı üzerinde çalışılmıştır. Makale göstermektedir ki dinamik ve stokastik ortamlarda gömülü bulunan uygulamalarda çok uygundur (Secomandi, 2003).

Cheung, Xu ve Guan’ın ortak çalışması olan 2004 yılına ait “Stokastik Talepli Çok Dolaşımli Dağıtım Problemi” (Multiple-Trip Delivery Problem Witj Stochastic Demands) makalesinde, araçların ürün dağıtımı için günde birden fazla kez sevk edildiği bir araç rotalama problemi üzerinde çalışılmaktadır. Bu problemde bazı müşteri siparişleri önceden bilinmekle beraber diğerleri kesin değildir, ancak gün içinde aşama aşama öğrenilmektedir. Anahtar kararlar, hangi bilinen siparişlerin ilk dolaşım sırasında ve hangilerinin daha sonraki bir dolaşımda dağıtılacağına karar verme ve müşteri siparişleri için rotaların ve çizelgelerin bulunmasını içermektedir. Bu problem beklenen toplam maliyeti minimize etme hedefiyle bir iki kademeli stokastik programlama olarak formüle edilmiştir. Bir en kötü durum analizi stokastik yaklaşımın deterministik bir yaklaşıma karşı potansiyel karını değerlendirmek için gerçekleştirilmiştir. Daha sonra bir sezgisel önerilmiş ve onun varyasyonları geliştirilmiştir (Cheung vd., 2004).

Jula, Dessouky ve Ioannau ortak çalışması olan “Müşteri Noktalarında Zaman Kısıtlı Sabit Olmayan Stokastik Ağlarda Kamyon Rota Planlaması” (Truck Route Planning In Non-Stationary Stochastic Networks With Time-Windows At Customer Locations) adlı makalede, yollar boyunca dolaşım sürelerinin ve müşteri noktasındaki hizmet sürelerinin stokastik süreçler olduğu Zaman Kısıtlı Stokastik Gezgin Satıcı Problemi adı verilen bir kamyon rota planlaması problemi incelenmektedir. Her bir müşteriye kamyonun varış zamanını tahmin eden bir metodoloji geliştirilmiştir. Tahmin edilen varış zamanları kullanılarak dinamik programlamaya dayanan bir yaklaşık çözüm metodu önerilmiştir Jula vd., 2006).

Pisinger ve Ropke'nin 2005 yılı çalışmaları olan "Araç Rotalama Algoritmaları İçin Genel Bir Sezgisel" (A General Heuristic For Vehicle Routing Problems) adlı makalede, araç rotalama probleminin beş farklı çeşidini çözeabilen birleştirilmiş bir sezgisel sunulmaktadır (Zaman Kısıtlı Araç Rotalama Problemi, Kapasiteli Araç Rotalama Problemi, Çok Depolu Araç Rotalama Problemi, Yer-Bağımlı Araç Rotalama Problemi ve Açık Araç Rotalama Problemi) (Pisinger ve Ropke, 2005).

"Stokastik Araç Rotalama İçin Dinamik Bir Plan" (A Dynamic Scheme For Stochastic Vehicle Routing) adlı Erera ve Daganzo'ya ait 2003 yılı çalışması, bir homojen araç filosunun bir merkez depodan çevredeki müşterilerin taleplerini karşılamak için gönderildiği stokastik yük kısıtlı araç rotalama sistemini göz önüne almaktadır. Müşteri yerleri ve yük boyutları yalnızca operasyonlardan önce dağıtımda bilinmektedir. Bu araştırma, maliyeti önemli bir şekilde düşürmek için gerçek zamanlı yeniden optimizasyondan yararlanan *Başlangıç Global Paylaşım* gösteren daha esnek bir plan önermektedir (Erera ve Daganzo, 2003).

Chepuri ve Homem-De-Mello'nun 2004 yılında geliştirdikleri "Stokastik Talepli Araç Rotalama Probleminin Çapraz-Entropi Kullanılarak Çözümü" (Solving The Vehicle Routing Problem With Stochastic Demands Using The Cross-Entropy Method) çalışmasında, çapraz-entropi metoduna dayalı yeni bir sezgisel yöntem önerilmiştir. Alandaki her bir noktada hedef fonksiyonunu daha iyi tahmin edebilmek için yöntem Monte Carlo örnekleme dahil edilmiştir. Ayrıca çeşitli durumlar altında problemin alt sınırlarını daraltmak ve kesin sonuçlar elde etmek için bir yapı geliştirilmiştir (Chepuri ve Homem-De-Mello, 2004).

Secomandi'nin 2000 yılı çalışması olan "Nöro-Dinamik Programlama Algoritmalarının Stokastik Talepli Araç Rotalama Problemi İçin Karşılaştırılması" (Comparing Neuro-Dynamic Programming Algorithms For The Vehicle Routing Problem With Stochastic Demands), müşteri taleplerinin kesin olmadığı bir araç rotalama problemini göz önüne almaktadır. Burada hedef her bir müşterinin talebine karşılık vermek için beklenen dolaşım mesafesini minimize etmektir. Bu çalışma, bu zor stokastik kombinasyonel optimizasyon problemine yaklaşık çözümler sağlamada Nöro-Dinamik Programlama (NDP)'nin meydana çıkan sahasının uygunluğu üzerinde durulmaktadır (Secomandi, 2000).

"Ulaştırma Sistemlerinin Planlanması ve Kontrolü : Rasgele Ağırlıklı Stokastik Sırt Çantası Problemi (SSÇP)" (Planning and Control of Transportation Systems : The Stochastic Knapsack Problem with Random Weights) adlı Barnhart'a ait 2000 yılı çalışmasında, SSÇP

için bir miktar motivasyon ve formülasyon sağlanmıştır. Ayrıca çözümde kullanılabilecek çeşitli özellikler kurulmuştur. Daha sonra bu özelliklerden yararlanmak için algoritmik yaklaşımlar önerilmiştir (Barnhart, 2000).

Haughton ve Stenger'in 1999 yılına ait çalışmaları “ Stokastik Talep Ayarlamalarında Dağıtım Eksikliklerini Göstermek İçin Stratejilerin Karşılaştırılması” (Comparing Strategies For Addressing Delivery Shortages in Stochastic Demand Settings) adlı makaleye göre, bir tedarik noktası (depo) ve coğrafik olarak dağınık çok sayıda talep noktası (perakende mağazaları) içeren lojistik ağlarında eğer dağıtım rotalarının tasarımı, müşteri taleplerindeki rasgele periyottan periyoda dalgalanmaları görmezden gelirse dağıtım eksiklikleri ortaya çıkacaktır. Dağıtım eksiklikleri, deponun onları önlemek için stratejiler aramasını gerektirecek kadar maliyetli olabilir. Stratejilerin rasyonel karşılaştırmaları için bir gereksinim, onların toplam tedarik zinciri maliyetleri üzerindeki etkisini sayısallaştırmaktır. Ulaştırma maliyet elementi için bu ön koşulu sağlamaya yardımcı olacak kesin mesafe tahmin modelleri geliştirilmiştir. Bu modeller, bu stratejilerin envanter ve taşımayı nasıl etkileyeceği bazında stratejilerin bir karşılaştırmasına entegre edilmiştir (Haughton ve Stenger, 1999).

“Tek Araçlı Bir Stokastik Rotalama Problemi İçin Yeni Optimallik Şekilleri” (New Optimality Cuts For A Single-Vehicle Stochastic Routing Problem) adlı Hjoring ve Holt’a ait makalede, talepleri araç rotalarının oluşturulmasından önce bilinmediği ancak bilinen bir olasılık dağılımını izlediği bir uygulama göz önüne alınmıştır. Müşteri taleplerindeki çeşitlilikten dolayı, güncel müşteri talebi o hizmete atanmış aracın kapasitesini aşabilir. Bu durumda bir rota başarısızlığı oluşur. Bu makalede, toplam dolaşılacak mesafeye artı rota başarısızlığından dolayı oluşacak beklenen ilave maliyetlerin de minimize edilmesi hedeflenmiştir. Optimal rotaların bulunması için tamsayı L-şekli yöntemi uygulanmıştır (Hjoring ve Holt, 1999).

Hu, Liao ve Lu'nun 2002 yılı çalışmaları olan “Gerçek Zamanlı Bilgiler Altında Dinamik Araç Rotalama Problemleri İçin Çözüm Yaklaşımın Üzerine Bir Çalışma” (A Study On The Solution Approach For Dynamic Vehicle Routing Problems Under Real-Time Information) araştırmasında, stokastik araç rotalama problemleri göz önüne alınmış ve dinamik araç rotalama problemleri için gerçek zamanlı bilgileri de dahil etmek için genişletilmiştir. SARP bir şans-kısıtlı model ile formüle edilmiştir ve dal-ve-sınır teknikleri ile CPLEX tarafından çözülmüştür (Hu vd., 2002).

Haugland, Ho ve Laporte'nin 2005 yılında yaptıkları “Stokastik Talepli Araç Rotalama

Problemi İçin Dağıtım Bölgelerinin Tasarımı” (Designing Delivery Districts For The Vehicle Routing Problem With Stochastic Demands) çalışması, stokastik talepli araç rotalama problemleri için bölgelerin tasarımını göz önüne almaktadır. Özellikle taleplerin bölgeler belirlendiğinde kesin olmadığı varsayılmaktadır ve bunlar yalnızca bölgeleme kararları verildikten sonra ortaya çıkmaktadır. Bu stokastik bölgeleme problemi için Tabu Search ve Multi Start sezgiselleri geliştirilmiş ve karşılaştırılmıştır (Haugland vd., 2005).

Wang ve Regan’ın 2000 yılı ortak çalışması olan “Hizmet ve Dolaşım Sürelerinin Stokastik Olduğu Zaman Kısıtlı Yerel Filo Atamaları İçin Alternatif Atama Modelleri” (Alternative Assignment Models For Time Constrained Local Fleet Assignment In Which Service And Travel Times Are Stochastic) makalesine göre, lokal yüklü kamyon taşımacılığı yük taşıma endüstrisinde önemli bir rol oynamaktadır. Dolaşım süreleri, müşteri noktalarındaki ve taşıma araçları arasındaki bekleme süreleri ile yükleme ve boşaltma sürelerinin stokastik olduğu operasyonlarda, deterministik atama yöntemleri zayıf kalabilir. Etkili stokastik atama modelleri önemli karlar sağlayabilir. Bu makalede, atamaları geliştirmek için problemin stokastik elemanlarını açıkça dahil eden teknikler araştırılmıştır. Bu makale hem zaman kısıtlarını hem de stokastik hizmet sürelerini hesaba katacak pek çok alternatif atam modelleri önermektedir (Wang ve Regan, 2000).

“Minimum Karşılanmayan Talep İçin Stokastik Araç Rotalama Problemi” adını taşıyan Shen, Ordenez ve Dessouky’e ait 2006 yılı çalışmasında yazarlar, belirsizlik altında karşılanamayan talebi minimize etmek için araç rotalama ile ilgilenmişlerdir. Böyle bir problem, tüm talep noktalarını karşılamamanın zor veya imkansız olduğu büyük talepli veya sıkı kısıtları olan durumlarda ortaya çıkar. Önemli bir uygulama alanı, doğal afetler ya da terörist saldırıları gibi geniş ölçekli acil durumlara cevap vermek için gerekli sağlık malzemelerinin dağıtımıdır. Yazarlar hem talebin hem de dolaşım süresinin belirsiz olduğu bir rotalama problemi göz önüne almışlardır. Stokastik parametrelerin dağılımına ılımlı tahminler altında modifiye edilmiş talep ve dolaşım süresi parametrelerine sahip bir deterministik probleme eşdeğer olan bir şans kısıtlı formülasyon sunulmuştur. Bu MIP’yi çözmek için bir tabu sezgiseli önerilmiştir ve hem deterministik hem de şans kısıtlı formülasyonlardan oluşturulan rotaların kalitesini ölçmek için simülasyonlar yönetilmiştir. Çalışma sonunda, şans kısıtlı rotaların, karşılanmayan talebi kısmen sıkı sınırlar ve toplam tedarik kısıtları için %2-%6 civarında düşürdüğü gözlemlenmiştir (Shen vd., 2006).

Bertsimas, “Stokastik Talepli Bir Araç Rotalama Problemi” isimli 1991 yılına ait makalesinde klasik araç rotalama probleminin taleplerin stokastik olduğu doğal bir olasılığa bağlı

varyasyonunu göz önüne almıştır. Talebin yalnızca olasılığa bağlı bir tanımı verildiğinde VRP için rotaların tasarlanması gerekmektedir. Stratejik planlama ve dağıtım sistemlerindeki uygulamalardan hareketle, talepler bilinir hale geldikten sonra problemi çözmektense, minimum beklenen toplam mesafenin tüm müşterileri arasında bir öncelik dizisi oluşturmak önerilmiştir. Farklı teorik yaklaşımlar kullanılarak problem analiz edilmiştir. Genel olasılığa bağlı yaklaşımlar altında bir öncelik dizisinin beklenen uzunluğunu hesaplamak için kapalı form ifadeleri ve algoritmalar bulunmuştur. Bu ifadelere dayanarak, her örnekte optimal rotanın bulunduğu olasılığa bağlı ARP ve ARP yeniden optimizasyon stratejisi için üst ve alt sınırlar bulunmuştur. Sezgiseller önerilmiş ve olasılık analizinden teknikler kullanılarak en kötü durum performansları ve ortalama davranışları analiz edilmiştir (Bertsimas, 1991).

Topaloğlu'nun 2005 yılına ait "Rasgele Dolaşım Süreli Bir Paralleştirilebilir Dinamik Filo Yönetimi" adlı makalesinde rasgele dolaşım süreli dinamik filo yönetimi problemi için bir stokastik model sunulmuştur. Bu yaklaşım, problemi dinamik bir program gibi zaman evreli alt problemlere ayırtırmakta ve değer fonksiyonunun tahminlerini kullanmaktadır. Rasgele dolaşım süreleri ile başa çıkabilmek için dinamik programın durum değişkeni, geçmişin ilgili bir kısmındaki bireysel kararları baştan başa içerir. Bu yeni yüksek boyutlu durum değişkeni altında, değer fonksiyonunun kolay izlenebilir bir tarzda nasıl tahmin edileceği gösterilmiştir. Tahmin planının altında, her bir zaman periyodu için alt problem, modeli yüksek ölçekli uygulamalara cazip hale getirmek için sırasıyla mevkilere ayrıştırılır. Sayısal çalışmalar göstermektedir ki önerilen yaklaşım yüksek kaliteli sonuçlar sağlamakta ve standart benchmark yöntemlerinde daha iyi performans sergilemektedir (Topaloğlu, 2005).

Chan, Carter ve Burnes'ün 1999 yılına ait "Stokastik Talepli Çok Depolu Çok Araçlı Yer Rotalama Problemi" makalesinde yazarlara göre stokastik talepli, çok depolu, çok araçlı yer rotalama probleminin stokastik versiyonu iki adımda işlemektedir: bölüştürme ve yer rotalama. Bölüştürme adımı, yükleri her bir depoya atar ve rotalama adımı talep noktalarına hizmet verebilmek için her bir bölgesel depodan çoklu araç turları oluşturur. İlki rasgele örnekleme adımı iken ikincisi geleneksel matematiksel programlama adımıdır. Beraber stokastik çok depolu, çok araçlı yer rotalama problemini çözerler (Chan vd., 1999).

3.2 Modeller

Literatürde bulunan ve Stokastik Araç Rotalama Probleminin çözümünde kullanılan bazı modeller şunlardır:

3.2.1 Gerçek Zamanlı Bilgiler Işığında Dinamik Araç Rotalama Problemi İçin İskelet Yapı ve Çözüm Yaklaşımı

Hu vd. (2003)'ne göre buradaki araştırma dinamik Araç Rotalama Problemi için Stokastik Araç Rotalama Problemine dayanan bir çözüm yaklaşımı geliştirerek, çözüm stratejisinin daha sonra gerçekçi bir trafik simülasyonu içinde değerlendirilmesine işaret etmektedir. Tüm araştırma yapısı Şekil 3.1'de gösterilmiştir. Bu yapıda üç basit eleman vardır. Bunlar; simülasyon-atama modeli, araç rotalama işlemi ve ileri trafik bilgi sistemidir. Böylece, nakil ve rotalama işlemleri, mevcut trafik şartlarında olduğu kadar araç mevkileri ve talepler üzerinde de gerçek zamanlı bilgi ile tam avantaj elde eder.

Önerilen değerlendirme yapısı gerçek zamanlı rotalama stratejilerinin etkinliğini görmek için kullanılır. Bu yapının temel girdileri; ilgili niteliklerdeki araçlar, trafik ağı tanımı ve trafik kontrolü gibi diğer verileri içerir. Bu yapı içerisinde araçlar, ticari ya da diğer özel araçlar da dahil olmak üzere, spesifik talepleri karşılamak için nakliyeciler tarafından gönderilebilir. Simülasyon-atama modeli – DYNASMART, zamana bağlı akış örnekleri simule etmek için kullanılır ve gerçek zamanlı bilgi dinamik nakil stratejileri tasarlamak için işler. Sonuçlar tekrar ticari araçlara geri atanır.

Dinamik Araç Rotalama Problemi için çözüm yaklaşımları iki aşamada uygulanır: rota oluşturulması ve rota geliştirilmesi. Orijinal rota serbest akış koşulları altında dolaşım süresine göre bir matematiksel formülasyon yoluyla oluşturulur. Dolaşım süresi mevcut trafik durumundan etkilendiği için, dolaşım süresi belirsizliğini ortalama ve sapma bakımından ele alan bir şans kısıtlı model geliştirilir ve çözülür. Şans kısıtlı matematiksel model dinamik dolaşım süresi bilgisi ve ilgili sapmalara göre uygulanır. Daha sonra, dinamik rota bilgisi gerçek zamanlı rota gelişmesi elde etmek için ticari araçlara sağlanır. Modelin karmaşıklığından dolayı şans kısıtlı model bir matematiksel programlama yazılımı olan CPLEX ile çözülür.

Aşağıdaki Laporte vd (1992) tarafından önerilen Stokastik Araç Rotalama Problemi formülasyonu bu çalışmada trafik şartları ve dolaşım süresi özellikleri göz önüne alınarak uygulanmıştır. Şans kısıtlı model, bir stokastik problemi öncelikle, B'den daha büyük bir dolaşım süresi olasılığının bir α eşik değerinden daha düşük olduğu ek bir kısıt ile çözer. Formülasyon aşağıdaki gibi gösterilmiştir (Hu vd., 2002):

$$\text{Min } fm + \sum c_{ij} x_{ij} \quad (3.1)$$

$$\sum_{i=1}^n x_{oj} = 2m \quad (3.2)$$

$$\sum_{i=1}^n x_{ij} = 1 \quad (3.3)$$

$$\sum_{j=1}^n x_{ij} = 1 \quad (3.4)$$

$$\text{Geçersiz rota eliminasyon kısıtları;} \quad (3.5)$$

$$x_{ij} \in \{0, 1\} \quad (3.6)$$

$$X = (x_{ij}) \in S \quad (3.7)$$

$$m \geq 1 \text{ ve } m \text{ tamsayı} \quad (3.8)$$

$$P\left(\sum_{k=0}^u [t_{i_k i_{k+1}} + \tau_{i_k}] x_{ij} > B\right) < \alpha \quad (3.9)$$

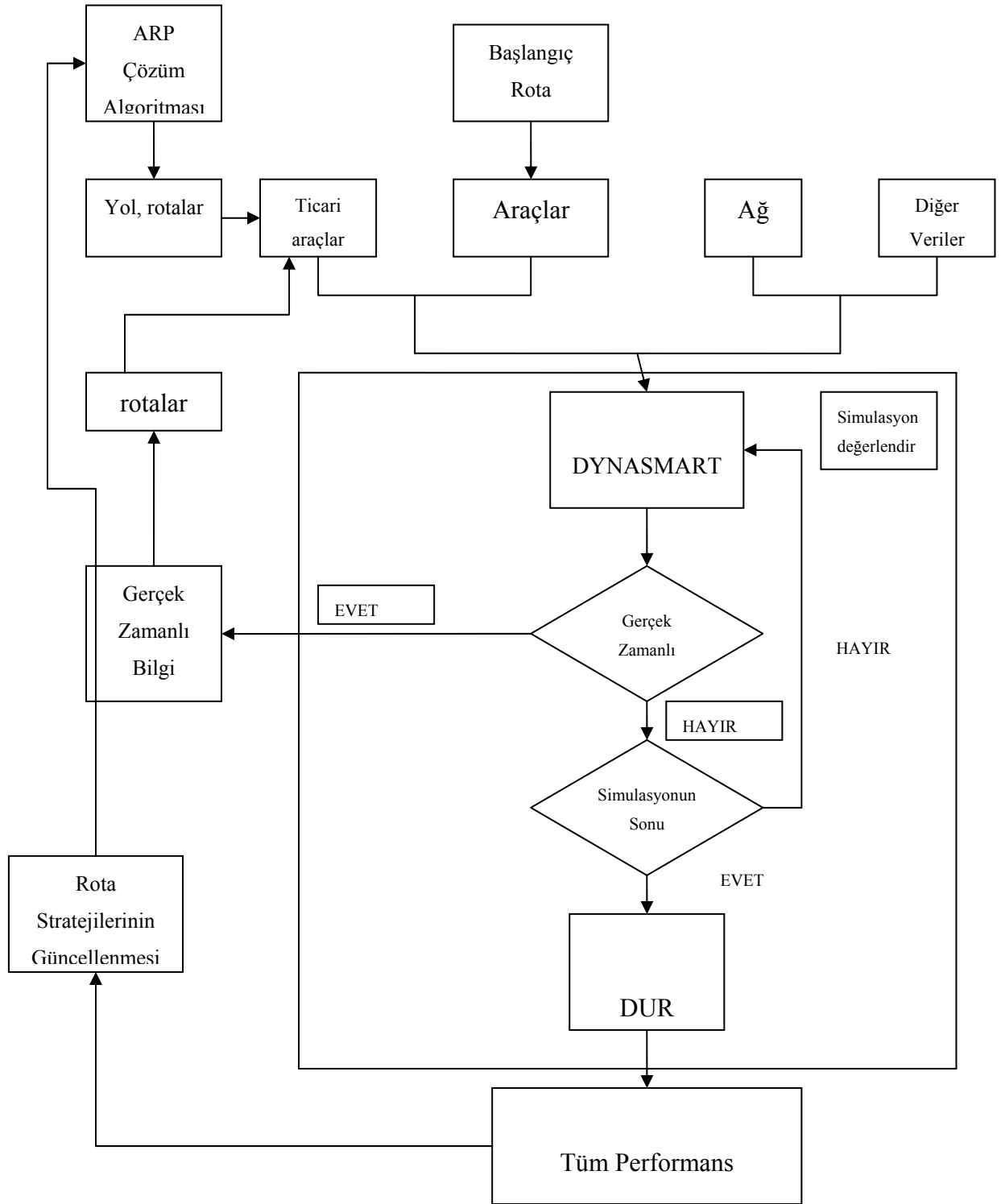
x_{ij} : rota sırası için karar değişkeni; örneğin $x_{ij} = 1$, rota (i, j) bağlantısını içeriyor demektir; 0, rota (i, j) bağlantısını içermiyor demektir.

f : m araç için sabit maliyet

m : araç sayısı

Hedef fonksiyonu araç dolaşım maliyetinin ve araç maliyetlerinin kombinasyonunu belirtmektedir. Geçersiz rota eliminasyon kısıtları farklı yollarla formüle edilebilirdi, kullanılan formülasyon aşağıdaki gibidir:

$$\sum_{i \in R} \sum_{j \in R} x_{ij} \leq |R| - 1 \quad R : \text{Nokta kümesi } \{2, 3, \dots, n\} \text{ 'in bir alt kümesi} \quad (3.10)$$



Şekil 3.1 Değerlendirme Yapısının Konsept Yapısı. (Hu vd., 2002)

3.2.2 Stokastik Talepli Araç Rotalama Problemi İçin Dağıtım Bölgelerinin Tasarımı İçin Model Formülasyonu

Haugland vd (2005).’ne göre problem, iki aşamalı başvuru bir stokastik program olarak modellenenebilir. İlk aşamada, bölgeleme kararları alınır, ikinci aşamada ise her bir bölge için bir ARP çözülür. Bir bölgenin düzenine ilişkin kısıt, dolaşım maliyetinin verilen bir sayıdan daha büyük olmamasıdır.

$D_1, \dots, D_n$ taleplerinin bağımsız stokastik değişkenler olması sağlanır. $S \subseteq V \setminus \{0\}$ için; $f(S, D)$, $D = (D_1, \dots, D_n)$ talep vektörü ile ilgili stokastik rotalama maliyet fonksiyonunu, Ω , D ’nin tüm gerçek değerlerinin kümesi olsun, ve $d \in \Omega$ için $f(S, d)$, d gerçek değerini veren rotalama maliyetini gösterebilir. Ayrıca, $E[X]$ rasgele bir X değişkeninin beklenen değeri ve $P(\varepsilon)$ da bir ε olayının olasılığı olsun. Her bir bölgedeki dolaşım maliyetine bir t üst sınırı verilerek, $V \setminus \{0\}$ ‘ın, beklenen en düşük maliyet bölgelerine $\{S_1, \dots, S_m\}$ bölünmesi aranır, yani aşağıdaki problem çözülür (Haugland vd., 2005);

$$\min_{\{S_1, \dots, S_m\}} \sum_{j=1}^m E[f(S_j, D)] \quad (3.11)$$

$$f(S_j, d) \leq t \quad j = 1, \dots, m, d \in \Omega \quad (3.12)$$

3.2.2.1 Beklenen Maliyetin Tahmini

Verilen bir S bölgesinin beklenen maliyetinin hesaplanması belli bir sayıda rotalama probleminin çözümünü gerektirdiği için, bu bir başvuru talep operasyonudur. Alt problemlerin bir yaklaşık sonucuna güvenilerek, sezgisel, maliyetin kendisi yerine üst sınırlar ile yönlendirilebilir. Bu sınırlar şu şekilde hesaplanır:

3.2.2.2 Beklenen Başvuru Maliyetinin Tahmini

(3.13) kısıtını sağlayan bir $S \subseteq V \setminus \{0\}$ bölgesi göz önüne alınır ve $\pi_S = (v_1, \dots, v_{|S|})$ müşterilerin bir permütasyonu olur. Verilen bir $d \in \pi$ için

$$\sum_{v \in S} E[D_v] \leq kq \quad (3.13)$$

$$g^*(\pi_S, d) = \min_{r, i_0, \dots, i_r} g(\pi_S, i_1, \dots, i_{r-1}) \quad (3.14)$$

$$0 = i_0 < i_1 < \dots < i_r = |S| \quad (3.15)$$

$$r \in \{1, \dots, l\} \quad (3.16)$$

$$\sum_{j=i_{h-1}+1}^{i_h} d_{v_j} \leq q \quad \forall h = 1, \dots, r \quad (3.17)$$

tanımlanır ve bu $(|S|^{l+1})$ süresinde hesaplanabilir.

$\left\{ (0, v_{i_{h-1}+1}, \dots, v_{i_h}, 0) \right\}_{h=1}^r$ rotalar kümesi $VRP(S, d)$ 'ye bir fizibil çözüm teşkil ettiği için, $f(S, d) \leq g^*(\pi_S, d)$ sonucu çıkar. (k, l) – bütünlüğü ile ayrıca $f(S, d) \leq t$ elde edilir ve bu nedenle üst sınır,

$$E[f(S, D)] \leq B(\pi_S) = \sum_{d \in \Omega} P(D=d) \min \left\{ t, g^*(\pi_S, d) \right\}$$

olur.

Benzer şekilde, $u \in S$ ve $v \in V \setminus \{0\} \setminus S$ için, sırasıyla $S \setminus \{u\}$ ve $S \cup \{v\}$ 'nin permütasyonları olarak $\pi_S - u$ ve $\pi_S + v$ tanımlanır. Burada, π_S 'teki sıra korunmuştur ve $\pi_S + v$, v , π_S 'teki bir pozisyona $B(\pi_S + v)$ minimize edilecek şekilde eklenerek oluşturulmuştur (Haugland vd., 2005).

3.2.3 Bir Stokastik Araç Rotalama Problemine Sezgisel Bir Çözüm Yöntemi İçin Matematiksel Formülasyon

Hvattum vd (2003). 'nin çalışmasına göre zaman çerçeveli ve homojen bir araç filosuna sahip deterministik, kapasiteli araç rotalama problemi bir karma tamsayı programı (MIP) olarak formüle edilebilir. Müşterilerin kümesi $C = \{1, 2, \dots, n\}$ ile gösterilsin ve 0 depo olsun. Böylelikle tüm yerlerin kümesi $N = C \cup \{0\}$ olsun. Dolaşım süreleri, c_{ij} , her bir yer çifti $i, j \in N$ arasında verilir ve muhtemelen hizmet süresini de içerir. Her bir i müşterisinin bir d_i talebi ve hizmet için bir $[a_i, b_i]$ zaman çerçevesi vardır. Depo için zaman çerçevesi $[a_0, b_0]$ ile verilir ve müşteriler için olan zaman çerçevesi gibi zor (hard) olduğu göz önüne alınır. Yani, hizmet a_i 'den önce başlamalı ve b_i 'den önce bitmelidir. (Hizmet süresi c_{ij} 'ye dahil olduğundan karşılık gelen zaman çerçevesinin başlangıç ve bitiminden çıkarılmalıdır.) Her biri bir q kapasitesine sahip bir araç kümesi, $V = \{1, 2, \dots, k\}$, a_0 'da depodan başlamalı ve

tüm müşterilere hizmet verdikten sonra b_0 'da geri dönmüş olmalıdır.

Karar değişkenleri x_{ijk} 'dir. Eğer k aracı i 'den j 'ye gidiyorsa $x_{ijk} = 1$, aksi halde $x_{ijk} = 0$ olur. Ek olarak, s_{ik} , k aracının i müşterisindeki hizmeti tamamladığı zamanı gösterir. s_{ik} değişkenlerinin, atanmış alt tur eliminasyon kısıtlarını gereksiz hale getirdiğine dikkat edilmelidir. Daha sonra, Ho ve Haugland (2002) ve Kallehaugei Larsen Ve Madsen (2001) tarafından kullanılan formülasyonlara dayanarak aşağıdaki MIP yazılabilir (Hvattum vd., 2003):

$$\min z(x, s) = \sum_{k \in V} \sum_{i \in N} \sum_{j \in N} c_{ij} x_{ijk} + K \sum_{k \in V} (1 - x_{00k}) \quad (3.18)$$

$$\sum_{k \in V} \sum_{i \in N} x_{ijk} = 1 \quad \forall j \in C \quad (3.19)$$

$$\sum_{j \in N} x_{0jk} = 1 \quad \forall k \in V \quad (3.20)$$

$$\sum_{i \in N} x_{ihk} - \sum_{j \in N} x_{hjk} = 0 \quad \forall h \in C, \forall k \in V \quad (3.21)$$

$$\sum_{i \in N} x_{i0k} = 1 \quad \forall k \in V \quad (3.22)$$

$$\sum_{i \in C} d_i \sum_{j \in N} x_{ijk} \leq q \quad \forall k \in V \quad (3.23)$$

$$s_{ik} + c_{ij} - M_{ij}(1 - x_{ijk}) \leq s_{jk} \quad \forall j \in C, \forall i \in N, \forall k \in V \quad (3.24)$$

$$a_i \leq s_{ik} \leq b_i \quad \forall i \in N, \forall k \in V \quad (3.25)$$

$$s_{ik} + c_{i0} - M_{i0}(1 - x_{i0k}) \leq b_0 \quad \forall i \in C, \forall k \in V \quad (3.26)$$

$$s_{0k} = a_0 \quad \forall k \in V \quad (3.27)$$

$$x_{iik} = 0 \quad \forall i \in C, \forall k \in V \quad (3.28)$$

$$x_{ijk} \in \{0, 1\} \quad \forall i, j \in N, \forall k \in V \quad (3.29)$$

Burada (3.18) hedefi, kullanılan toplam araç sayısı artı dolaşılan toplam süreyi minimize etmektir. K sabiti ekstra bir araç çıkarma maliyetidir. (3.20), (3.21) ve (3.22) sabitleri, her bir aracın depodan bir kez ayrılması gerektiğini, ulaştığı her bir müşteriden ayrılıp depoya bir kez dönmesi gerektiğini belirtmektedir. (3.23) eşitsizliği araç kapasitesinin aşılmamasını

sağlamaktadır. (3.24) kısıtı, k aracının j müşterisine hizmet vermeden önce i ve j arasında doğru miktarda sürüş zamanı harcamasını sağlarken, (3.25) eşitsizlikleri her bir müşteri için zaman çerçevelerine karşılık gelmektedir. Ek olarak (3.26), her bir aracı zaman çerçevesinin sonundan önce depoya varmaya zorlamaktadır. M_{ij} sabiti yeterince büyük olacak şekilde tanımlanmıştır. Örneğin; $M_{ij} = b_i + c_{ij} - a_j$.

İki aşamalı stokastik karşılığında, zamanda önceden bilinmeyen bilgilerin ortaya çıktığı bazı tek bilinen $t \in [a_0, b_0]$ noktalarının olduğu varsayılır. Bu da, tüm stokastik müşterilerin, yerlerinin, taleplerinin ve zaman çerçevelerinin bilinir hale geldiği anlamına gelir. Bu, başvuru hareketlerinin karşılanması için yeni değişkenlerin ortaya konacağı iki aşamalı bir stokastik program olarak modellenebilir.

$C' = \{n+1, n+2, \dots, n+n'\}$ ortaya çıkan müşterilerin kümesi olsun. d_i ve $[a_i, b_i]$, $i \in C'$ olmak üzere, müşteriler için talep ve zaman çerçeveleri olsun. Tüm yerlerin kümesi, $C' = C \cup C'$ olmak üzere, $N' = N \cup C'$ olsun. i ve j arasındaki dolaşım süresi, $i, j \in N'$ olmak üzere, c_{ij} olsun. $\xi = (C', D', A', B', c')$, $\tilde{\xi} = (\widetilde{C'}, \widetilde{D'}, \widetilde{A'}, \widetilde{B'}, \widetilde{c'})$ rasgele değişkenlerinin belirli bir gerçekliği olsun.

İki aşamalı stokastik programlama problemi aşağıdaki gibi formüle edilebilir:

$$\min E \tilde{\xi} \left[z(x, s) + Q(x, s, \tilde{\xi}) \right] \quad (3.30)$$

$$Q(x, s, \tilde{\xi}) = \sum_{i \in N'} \sum_{j \in N'} \sum_{k \in V} c_{ij} (x_{ijk}^{t+} - x_{ijk}^{t-}) + K \sum_{k \in V} (x_{00k}^{t-} - x_{00k}^{t+}) \quad (3.31)$$

$$\sum_{k \in V} \sum_{i \in N'} (x_{ijk} + x_{ijk}^{t+} - x_{ijk}^{t-}) = 1 \quad \forall j \in C' \quad (3.32)$$

$$\sum_{j \in N'} (x_{ojk} + x_{ojk}^{t+} - x_{ojk}^{t-}) = 1 \quad \forall k \in V \quad (3.33)$$

$$\sum_{i \in N'} (x_{ihk} + x_{ihk}^{t+} - x_{ihk}^{t-}) - \sum_{j \in N'} (x_{hjk} + x_{hjk}^{t+} - x_{hjk}^{t-}) = 0 \quad \forall h \in C', \forall k \in V \quad (3.34)$$

$$\sum_{i \in N'} (x_{i0k} + x_{i0k}^{t+} - x_{i0k}^{t-}) = 1 \quad \forall k \in V \quad (3.35)$$

$$\sum_{i \in C} d_i \sum_{j \in N} (x_{ijk} + x_{ijk}^{t+} - x_{ijk}^{t-}) \leq q \quad \forall k \in V \quad (3.36)$$

$$x_{ijk}^{t-} \leq x_{ijk} \quad \forall i, j \in N, \forall k \in V \quad (3.37)$$

$$x_{ijk}^{t-} = 0 \quad \forall i, j \in C', \forall k \in V \quad (3.38)$$

$$tx_{ijk}^{t-} \leq s_{ik} \quad \forall i, j \in N, \forall k \in V \quad (3.39)$$

$$s_{ik}^t + c_{ij} - M_{ij} (1 - x_{ijk} - x_{ijk}^{t+} + x_{ijk}^{t-}) \leq s_{jk}^t \quad \forall j \in C', \forall i \in N', \forall k \in V \quad (3.40)$$

$$a_i \leq s_{ik}^t \leq b_i \quad \forall i \in N', \forall k \in V \quad (3.41)$$

$$s_{ik}^t + c_{i0} - M_{i0} (1 - x_{i0k} - x_{i0k}^{t+} + x_{i0k}^{t-}) \leq b_0 \quad \forall i \in C', \forall k \in V \quad (3.42)$$

$$s_{0k}^t = a_0 + (t - a_0) x_{00k} \quad \forall k \in V \quad (3.43)$$

$$x_{iik}^{t+} = 0 \quad \forall i \in C', \forall k \in V \quad (3.44)$$

$$x_{ijk}^{t+} \in \{0, 1\} \quad \forall i, j \in N', \forall k \in V \quad (3.45)$$

$$x_{ijk}^{t-} \in \{0, 1\} \quad \forall i, j \in N', \forall k \in V \quad (3.46)$$

Burada, x_{ijk}^{t+} , x_{ijk}^{t-} ve s_{ik}^t , $\widetilde{\xi}$ rasgele vektörüne bağlıdır ve bu yüzden onlar da rasgele değişkenlerdir. $E_{\widetilde{\xi}}$, $\widetilde{\xi}$ dağılımına göre beklenen değere karşılık gelir. $Q(x, s, \widetilde{\xi})$ başvuru maliyetidir ve yeni araçlar çıkartmanın ve dolaşım süresindeki değişimin maliyetlerini toplamıdır. Eğer k aracı, i'den j'ye ikinci aşama başvurusunda gidip ilk aşama çözümünde gitmediyse $x_{ijk}^{t+} = 1$ olur. Eğer k aracı, i'den j'ye ilk aşama çözümünde gidip başvurudan sonra gitmediyse $x_{ijk}^{t-} = 1$ olur. s_{ik}^t , k aracının ikinci aşama başvuru hareketine göre i müşterisinde hizmeti tamamladığı süredir.

(3.32)-(3.35) kısıtları, (3.19)-(3.22) ile aynıdır ve tüm müşterilere hizmet verildiğini, herhangi bir aracın depodan bir kez ayrılmasını, ziyaret edilen tüm müşterilerden ayrılıp depoya bir kez varmasını sağlar. (3.36), ikinci aşama çözümünde araç kapasitelerinin aşılmamasını sağlar. (3.37)-(3.39)'de ilk aşama çözümünde i müşterisinden j müşterisine bir yolculuğun, ikinci aşama başvurusunda eğer aracın ayrıldığı yerdeki hizmet t zamanından önce tamamlanırsa atlanamayacağı gösterilmiştir. Depodan ilk aşama çözümünde ($x_{00k} = 1$) ayrılmayan bir k aracı, (3.43) da belirtildiği gibi depoyu bir ikinci aşama kararındaki t zamanından önce terk edemez. (3.40)-(3.42) kısıtları son, ikinci aşama çözümünde hizmetin tamamlandığı zamanı

izlemek için kullanılır.

Daha önce belirtildiği gibi, Her bir stokastik müşteri için çağırma zamanı belirsizdir. Bu yüzden iki aşamalı modelimiz dayandığı gerçek dünya durumu dinamik unsurunu yeterince özelliğini yitirmeden koruyamaz. Olası bir çare, problemdeki aşamaların sayılarını stokastik bir değişken olarak (çağırma başına bir aşama) ele almak olabilirdi. Bu çalışmada, zaman ufkunun belirli sayıda aralıklara (veya aşamalara), $I_1, \dots, I_n$ şeklinde, bölündüğü daha basit bir yaklaşım yeğlenmiştir. Amaç, mevcut bilinen müşteriler kullanılarak günün geri kalanı için plan yapmaktır. Bu her zaman aralığı için dinamik olarak tekrar edilir. Her seferinde bir önceki aralıkta elde edilen bilgiyle problem modifiye edilir (Hvattum vd., 2003).

3.2.4 Stokastik Talepli ve Yumuşak Zaman Çerçevesi Araç Rotalama Problemine Bir Genetik Algoritma Çözümü İçin Model

Mak ve Guo (2004)'ya göre talep ve müşteri varlığının belirsizliğinden dolayı bir araç hizmetini her zaman programlandığı gibi yürütemeyebilir. SVRPSTW, “önsel optimizasyon problemi” adı verilen ve bir stokastik programlama yapısı ile modellenip iki aşamada çözölen bir problem sınıfına aittir. İlk aşama çözümü m nakliye rotasından daha fazlasını içermeyiz. Bu rotalar: 1) her rota merkez depoda başlar ve biter; 2) her müşteri bir araç tarafından kesinlikle bir kez ziyaret edilir; şartlarını sağlar. Bir müşterinin var olup olmadığı, bir önceki müşteriden ayrılmadan önce bilinir. Böylece bu rotadaki bir önceki mevcut müşteri anında belirlenebilir. Öncelikli rota tasarlandığında her bir müşterinin gerçek talebi bilinmediğinden, araç kapasitesi Q'nun hizmet sırasında aşılması muhtemeldir. Bu durumda, rota başarısızlığı meydana gelir. Bu istisna durum için bir başvuru hareketi (recourse action) ikinci aşamada gerçekleştirilir. Araç yükünü boşaltmak için depoya döner ve hizmetine kapasitenin aşıldığı anda ziyaret ettiğı müşteriden veya eğer kapasiteye tam olarak erişildiyse rotadaki bir sonraki mevcut müşteriden devam eder. Ayrıca, araç ikinci aşamada tüm mevcut durumda olmayan müşterileri atlar. Rota başarısızlığı için başvuru hareketi ek bir maliyet oluşturur ve atlama hareketi de üçgensel eşitsizliğe göre potansiyel bir maliyet tasarrufu sağlar. Müşterinin var olma olasılığı ve stokastik talebi çözüme değişik etkiler yapar.

Burada herhangi bir müşterideki hizmet süresinin sıfır olduğu varsayılmıştır. Ayrıca tüm rotaların için aynı zaman sınırlamaları olduğu ve tüm araçların sıfır anında başladığı varsayılmıştır.

Amaç ikinci aşama çözümünün beklenen maliyetini minimize etmek için bir ilk aşama çözümü tasarlamaktır. İkinci aşama maliyeti; birinci aşama maliyetini, başvuru hareketiyle

ilgili beklenen maliyeti ve zaman aşımı ile ilgili beklenen cezayı içerir. Burada tüm mesafeler öklit mesafeleri ile belirtilmiştir ve tüm araçların hızlarının 1 birim olduğu varsayılmıştır. Böylece, sayısal olarak, dolaşım maliyeti c_{ij} , dolaşım süresi t_{ij} ve noktalar arası öklit mesafesi birbirine eşittir.

Fizibil çözümde m adet araç olduğu varsayılmıştır. $r = \{r_1, r_2, \dots, r_m\}$, r_k 'nin k. araca atanmış alt tur olduğu belirli bir çözüm olsun. $r_k = \{v_1, v_i, \dots, v_j, v_1\}$, dolaşım sırasının, k aracı tarafından hizmet verilmek üzere $v_1 \rightarrow v_i \rightarrow \dots \rightarrow v_j \rightarrow v_1$ olduğu özel bir alt tur olsun. k. alt turda yalnızca k_n ($k_n \leq n$) kadar nokta olduğu varsayılarak her v_{kn+1}, v_1 olur. Noktalar yeniden etiketlenir ve k. alt rota $\{v_1, \dots, v_{kn}, v_{kn+1} = v_1\}$ olur. $z_k(S)$, ikinci aşamada k. alt rotanın beklenen maliyeti olsun. Böylece tüm beklenen maliyet $z(S)$, tüm alt turların toplamı olur. Aşağıda bu modelde kullanılan değişkenler tanımlanmıştır:

m: araç sayısı, tüm araçlar eşdeğerdir;

f : bir aracı görevlendirmenin maliyeti;

c_{v_i, v_j} : (v_i, v_j) kenarında dolaşım maliyeti;

p_{v_i} : v_i noktasının varolma olasılığı;

$p_i^k(g)$: k rotasında i. müşteride, i. müşteriye varıştaki kalan g kapasitesi ile meydana gelen rota başarısızlığından dolayı maruz kalınan beklenen ek maliyet;

Q: araç kapasitesi;

λ : tüm araçlar için sabit zaman cezası katsayısı;

h: müşteri taleplerine (sıfır talep de dahil) karşılık gelen stokastik değerler vektörü. h'nin her bir gerçekleşmesi "talep durumu" olarak adlandırılır.

H: h'nin tüm aralığı, H bu modelde sınırlıdır;

$y_i^k(h)$: k rotasında, h talep durumunda araç i. müşteriye ulaştığındaki gecikme süresi;

d_{v_i} : v_i müşterisinin talep değeri;

q_{v_i}, d : v_i müşterisinin, değeri d'ye eşit olan talep olasılığı;

p_h^k : k rotasında h talep durumunun olasılığı;

t_{ih}^k : k. rotada, i. müşteride, h talep durumunda hizmet başlangıç zamanı.

Stokastik başvuru modeli:

$$\min z(S) = \sum_{i=1}^{kn} z_k(S) + mf \quad (3.47)$$

k. rotanın beklenen maliyeti:

$$z_k(S) = \sum_{i=1}^{kn} \sum_{j=i+1}^{kn+1} c_{v_i, v_j} \bar{p}_{v_i, v_j} + P_2^k(Q) + \lambda E_h \left(\sum_{i=1}^{kn+1} y_i^k(h) \right) \quad 1 \leq \underline{m} \leq m \leq \bar{m} \quad (3.48)$$

$$\bar{p}_{v_i, v_j} = p_{v_i} p_{v_j} \prod_{k=i+1}^{j-1} (1 - p_{v_k}) \quad (3.49)$$

$$P_{v_{kn}}^k(g) = 2p_{v_{kn}, v_1} c_{v_{kn}, v_1} \sum_{d \setminus (d_{v_{kn}} > g)} q_{v_{kn}, d} \quad (3.50)$$

$$P_{v_1}^k = (1 - p_{v_i}) P_{v_{i+1}}^k(g) + p_{v_i} G1_{v_i}^k(g) + q_{v_i, d \setminus (d_{v_i, d} = g)} G2_{v_i}^k(g) \quad (3.51)$$

$$G1_{v_i}^k(g) = \sum_{d \setminus (d_{v_i, d} > g)} q_{v_i, d} \left(P_{v_{i+1}}^k(Q - d_{v_i, d}) + 2c_{v_1, v_i} + \sum_{d \setminus (d_{v_i, d} < g)} q_{v_i, d} P_{v_{i+1}}^k(g - d_{v_i, d}) \right) \quad (3.52)$$

$$G2_{v_i}^k(g) = p_{v_i} P_{v_{i+1}}^k(Q) + \sum_{j=i+1}^{kn} \bar{p}_{v_i, v_j} (c_{v_1, v_i} + c_{v_1, v_j} - c_{v_i, v_j}) \quad (3.53)$$

$$y_i^k(h) = p_h^k (t_{ih}^k - b_i)^+ \quad (3.54)$$

$$(t_{ih}^k - b_i)^+ = \max \left\{ 0, (t_{ih}^k - b_i) \right\} \quad (3.55)$$

Yukarıdaki modelde, (3.47) eşitliği problemin amaç fonksiyonudur. (3.48) eşitliği tek bir rotanın beklenen maliyetini hesaplar. (3.48) eşitliğinin ilk bölümü beklenen ilk aşama maliyeti ile mevcut durumda olmayan müşterilerin atlanmasıyla sağlanan maliyet tasarrufunun farkını birleştirir. (3.49) kısıtı, alt ve üst sınırları araç sayısına uygular. Eğer $\underline{m} = \bar{m}$ ise araç sayısı sabittir. Tekrarlamalı (3.51) fonksiyonu (3.50), (3.52) ve (3.53) eşitlikleri ile birlikte, v_i müşterisine varıldığında kalan kapasitenin g'ye eşit olması ile rota başarısızlığı oluşması durumunda, k. rotanın beklenen negatif olmayan ek mesafesini belirtir. Ayrıca, eğer müşterinin talebi ayrı ayrı ulaştırılabiliyorsa, (3.52) eşitliğinde $Q + g - d_{r_{ki}, d}$ ile

$Q - d_{r_{ki},d}$ yer değiştirilir. (3.54) ve (3.55) eşitlikleri i. müşterinin, k. rota boyunca, h talep durumunda zaman cezasını hesaplar (Mak ve Guo, 2004).

3.3 Çözüm Yöntemleri

Literatürde bulunan, SARP için çözüm yöntemlerinin bazıları üzerinde duralım.

3.3.1 Araç Rotalama Problemleri İçin Yapay Tavlama Algoritması

Gupta (2002)'ya göre Yapay Tavlama (SA), Hill-Climbing (HC) algoritmasının lokal aşırılıklardan kaçınan geliştirilmiş bir versiyonudur. SA fikri fizikten gelmektedir. Sıcaklık önceden karar verilen bir başlangıç değerine getirilir ve soğutma planına göre 0 dereceye kadar kademeli olarak düşürülür. İkinci derece çözümlere kabul edilme şansları verilir. Derece ne kadar yüksekse, ikinci derece çözümlerin kabul edilme şansı da o kadar yüksektir. SA rasgele veya bazı sezgisel araştırmalarla oluşturulan bir başlangıç çözümü ile başlar. Her bir tekrarda bir komşulama çözümü incelenir. Eğer mevcut olandan daha iyi ise bu yeni çözüm mevcut çözüm olur. Aksi halde, komşulama çözümü sıcaklıkla alakalı bir olasılık altında kabul edilir. Eğer reddedilirse, başka bir incelenmemiş komşulama çözümü, eğer varsa, aynı tarzda incelenir (Gupta, 2002).

SA Algoritması

1. Input

- (a) Bir başlangıç çözümü S
- (b) Maliyet fonksiyonu
- (c) Düzensizlik fonksiyonu, Başlangıç derecesi (T)
- (d) Soğutma programı (R), Son Derece (T^*)
- (e) Tekrarların sayısı (N)

2. Output

- (a) Mümkün olan en iyi çözüm
- (b) Çözümün maliyeti

3. f(S)'i değerlendir

4. Değişiklik = Doğru

5. While (Değişiklik) Do

(a) Değişiklik = Yanlış

(b) N kez tekrar et

i. Düzensizlik fonksiyonunu kullanarak S 'in S^* komşulama çözümünü oluştur

ii. $\Delta = f(S^*) - f(S)$

iii. Eğer ($\Delta < 0$) ise goto Kabul

iv. $P(\Delta) = e^{\left(-\frac{\Delta}{T}\right)}$

v. Rasgele x üret.

vi. Eğer ($x < P(\Delta)$) ise goto Kabul değilse Çıkış

vii. Kabul

A. $S = S^*$

B. $f(S) = f(S) + \Delta$

C. Eğer ($\Delta \neq 0$) ise Değişiklik = Doğru

(c) $T = T \times R$

(d) Eğer ($T < T^*$) ise Çıkış

End SA Algoritması.

3.3.2 Stokastik Talepli Araç Rotalama Problemi İçin Metasezgiseller

Bianchi vd. (2006)'nin bu çalışmasında stokastik talepli araç rotalama problemi için iyi bilinen beş farklı metasezgiselin performansının yansız bir karşılaştırması hedef alınmıştır. Sonuçların adil ve anlamlı bir analizini elde etmek için metasezgisel yaklaşımlar OrOpt ortak lokal araştırmasının kullanımı ile kısıtlanmıştır. Aşağıda her bir metasezgiselin ana prensipleri kısaca ve şematik olarak tanımlanmış ve Stokastik Talepli Araç Rotalama Problemi için uygulama detayları verilmiştir (Bianchi vd., 2006).

3.3.2.1 Yapay Tavlama (SA)

Başlangıç aday çözümü s belirlenir.

Tavlama programına göre başlangıç sıcaklığı T ayarlanır.

Sonuç koşulu sağlanmazsa:

Olasılıklı olarak s 'in bir s' komşusu seçilir.

Eğer s' olasılıklı kabul kriterini sağlıyorsa (T 'ye bağlı)

$s := s'$

T tavlama programına göre güncellenir.

Başlangıç sıcaklığı T_i , μ tarafından çoğaltılan başlangıç turunun 100 çözümünün bir örneğinin maliyeti ile verilir; her $\psi \cdot n$ tekrarda sıcaklık $T \leftarrow \alpha \times T$ olarak güncellenir (standart geometrik soğutma); geliştirme olmadan $\rho \cdot \psi \cdot n$ tekrar sonra, sıcaklık mevcut değere T_i eklenmek suretiyle yükselir; geliştirmeleri kontrol etmek için göz önüne alınan çözüm, son yeniden ısıtmadan beri en iyi çözümdür.

3.3.2.2 Tabu Araştırma (TS)

Başlangıç aday çözümü s belirlenir.

Sonuç kriteri sağlanmıyorsa:

Olasılıklı olarak s 'in bir s' komşusu seçilir.

Eğer s' olasılıklı kabul kriterini sağlıyorsa (tabu durumuna bağlı olarak):

$s := s'$

Tabu özellikleri s' baz alınarak güncellenir.

Tabu araştırma bir olasılıklı kabul kriterini izler. Bir çözümün tüm komşuları incelenir ve değerlendirilmiş en iyi komşu seçilir. Komşuların değerlendirilmesi şu şekilde yapılır: eğer bir komşu tabu değilse, onun maliyeti bir p_m olasılığına göre değerlendirilir; aksi halde maliyeti bir p_t olasılığı ile değerlendirilir ve yalnızca eğer o ana kadar bulunan en iyi çözüme götürüyorsa yeni s' çözümü olarak ayarlanabilir. Bu yöntemde, eğer bir çözümün hiçbir komşusu tabu değilse, komşular arasında en iyisi o ana kadarki en iyi sonuçla karşılaştırma

yapılmadan seçildiğinden yeni s' çözümü daha kötüleşmiş bir çözüm olabilir.

3.3.2.3 Tekrarlanmış Lokal Araştırma (ILS)

Başlangıç aday çözümü s belirlenir.

s üzerinde lokal araştırma gerçekleştirilir.

Sonuç kriteri sağlanmıyorsa:

$r := s'$

s üzerinde perturbation (bozulma, sapma, düzensizlik) uygulanır.

Kabul kriterine dayanarak, s tutulur veya $s := r$ olarak eski haline getirilir.

Bozulma, n komşunun örneklemesinden oluşur. Her bir yeni çözüm kesin maliyet fonksiyonu ile değerlendirilir ve o ana kadar bulunan en iyi çözüm artı ε 'dan daha düşük bir maliyetli bir çözüm bulunursa, örnekleme biter; aksi halde, örnekleme sırasında bulunan en iyi çözüme geri dönlür; kabul kriteri eğer o ana kadar bulunan en iyi çözüm ise s 'i tutar.

3.3.2.4 Karınca Koloni Optimizasyonu (ACO)

Ağırlıklar başlangıç durumuna getirilir (pheromone izleri).

Sonuç kriteri sağlanmıyorsa:

Rasgele duruma getirilmiş bir yapıcı sezgisel ile çözümlerin bir sp popülasyonu üretilir.

Sp üzerinde lokal araştırma gerçekleştirilir.

Ağırlıklar sp 'ye dayanarak uyarlanır.

Pheromone izleri τ_0 başlangıç durumuna getirilir; sp çözümleri yapıcı bir sezgisel ile üretilir ve lokal araştırma ile temizlenir; bir global güncelleme kuralı r kez uygulanır; daha sonra sp çözümleri, pheromone matrisinde saklanan bilgi kullanılarak oluşturulur; her bir yapım adımından sonra seçilmiş müşteri çiftine karşılık gelen $\tau_{i,j}$ elemanına bir lokal güncelleme kuralı uygulanır: $\tau_{i,j} = (1 - \psi) \cdot \tau_{i,j} + \psi \cdot \tau_0$, $\psi \in [0,1]$ için; lokal araştırmadan sonra ağırlıklar global güncelleme kuralı tarafından tekrar güncellenir (sezgisel bilginin yalnızca başlangıç durumuna getirme aşamasında kullanıldığına dikkat edilmelidir).

3.3.2.5 Evrimsel Algoritma (EA)

Başlangıç popülasyonu sp belirlenir.

Sonuç kriteri sağlanmıyorsa:

Yeniden birleşim ile çözümlerin bir spr kümesi üretilir.

Mutasyon yolu ile spr 'den çözümlerin bir spm kümesi üretilir.

spm üzerine lokal araştırma uygulanır.

sp , spr ve spm 'deki çözümlerden sp popülasyonu seçilir.

Her tekrarda Kenar Yeniden Birleşimi aracılığıyla yeni spr çözümü üretmek için en iyilerin içinden iki çözüm seçilir (ne zaman mümkün olursa, diğer iki turun her ikisinde de kenarlar kullanılarak bir tur üretilir); mutasyon komşu müşterileri (depoyu göz önüne almadan) p_m olasılığı ile değiştirir; sonuçta, lokal araştırma ile geliştirilen çözüm popülasyondaki en kötü çözümün yerini alır.

Tüm metasezgiseller için başlangıç çözümü (s) Farthest Insertion (En Uzak Ekleme) yapıcı sezgiseli ile elde edilir; bir sonraki müşteriyi mevcut olandan en uzakta olmak suretiyle henüz ziyaret edilmemiş müşteri olarak seçerek bir tur oluşturur. Burada, bu sezgiselin (RFI), ilk müşteriyi rasgele seçen ve tur tamamlandıktan sonra başlangıç müşterisini depo ile değiştiren, rasgele hale getirilmiş bir versiyonu göz önüne alınır.

Bir metasezgiseller arası karşılaştırma referans algoritması kullanmak için ayrıca RFI sezgiselleri ile beraber lokal araştırmayı da kullanan ve sonuç koşuluna erişilinceye kadar her lokal optimum bulunduğu yeniden başlayan basit bir rasgele başlangıç algoritması uygulanmıştır. Tüm yeniden başlangıçlar arasından bulunan en iyi çözüm sonuç çözümü olarak alınır; Bu algoritmaya RR denir.

4. STOKASTİK TALEP İÇİN SEZGİSEL BİR ALGORİTMANIN GELİŞTİRİLMESİ VE UYGULANMASI

Bu bölümde örnek bir model üzerine sezgisel rota oluşturma yöntemlerinden en kısa yol prensibi kullanılarak araçlar rotalara atanacak ve daha sonra bulunan ilk sonuçlar sezgisel rota iyileştirme algoritmaları kullanılarak iyileştirilmeye çalışılacaktır. Son olarak da aynı modele Clark-Wright'ın tasarruf algoritması uygulanacak ve sonuçlar karşılaştırılacaktır.

4.1 Stokastik Talepli Araç Rotalama Problemi İçin Sezgisel Bir Algoritma

Algoritmanın işleyişi aşağıdaki gibidir:

1. Araç; merkez depodan (v_0), tam kapasite (C_0) ile yüklenmiş olarak, i adet müşteriye ($v_1, v_2, \dots, v_i$) hizmet verecek şekilde öncelikle depo ve müşteriler arasında en kısa yol prensibine göre belirlenen rotaya atanır.
2. Depoya göre en yakın nokta olan müşterinin (v_1) talebi (d_1) karşılanır. Araç, kapasitesi aşılmadıysa ($d_1 < C_0$) rota sıralamasındaki bir sonraki noktaya (v_2) devam eder. Bu noktadaki talep aracın v_1 noktasındaki talebi karşıladıktan sonraki kalan kapasitesinden (C_1) küçük ise ($d_2 < C_1$) buradaki talep de karşılanıp bir sonraki noktaya (v_i), nokta talebi (d_i) aracın hizmet verilen son noktadan sonra kalan kapasitesinden (C_{i-1}) fazla oluncaya kadar ($d_i > C_{i-1}$) devam edilir. Talep araç kapasitesini aştığı zaman merkez depoya geri dönlür.
3. Kapasitesi aşılp merkez depoya dönen aracın yerine ikinci bir araç tam kapasite ile yüklenmiş olarak, rota hatası oluşmaması halinde birinci aracın talebini karşıladığı müşteriler dışarıda bırakılarak, depoya en yakın müşteriye atanır ve 2. adımdaki işlemler tekrarlanır. Rota hatası oluşmuş ise, hatanın olduğu nokta daha sonraki bir rotaya yine en kısa yol prensibine göre eklenerek yeni rota oluşturulur. Tüm nokta talepleri karşılanıncaya kadar bu işlemler sürdürülür.
4. Araçlar rotalara atanıp tüm müşteri talepleri karşılandıktan sonra sezgisel rota iyileştirme yöntemleri (nokta değişimi, nokta atama, rota değişimi) kullanılarak oluşturulan başlangıç rotası toplam dolaşım süresi kısaltılarak optimize edilir.
5. Son olarak aynı talep değerleri ve mesafeler için modele Clark-Wright tasarruf algoritması uygulanarak optimal bir çözüm elde edilir.

4.2 Uygulama

Öncelikle $i = 12$ adet müşteri içeren, talep değerleri olasılığa bağlı olan, her biri $C_0 = 250$ birim kapasiteye sahip $m = 3$ araçtan meydana gelen bir dağıtım ortamını göz önüne alalım. Müşteriler ve merkez depo arası mesafeler aşağıdaki gibi olsun;

Çizelge 4.1 Müşteriler ve merkez depo arası mesafeler

	0	1	2	3	4	5	6	7	8	9	10	11	12
0	0	66,5	53	40,5	29,5	38,5	58,5	75,5	66	45,5	43	44,5	41
1		0	24,5	29,5	74	70,5	95,5	118	104,5	90	77	72,5	44,5
2			0	30,5	47,5	45	76	105,5	100,5	80	72,5	71	39,5
3				0	55	61	89,5	110,5	94,5	74,5	49	54	24,5
4					0	23,5	34	55	49,5	35,5	54,5	69,5	63,5
5						0	46	69	68	50,5	69,5	80,5	75,5
6							0	35,5	44	35,5	61,5	79,5	74,5
7								0	34,5	39,5	66	84,5	94
8									0	23,5	51	71	81
9										0	50,5	66	67,5
10											0	24,5	30
11												0	29,5
12													0

Her bir müşteri için olasılığa bağlı talepler ise şu şekildedir:

Çizelge 4.2 Her bir müşteri için olasılığa bağlı talep değerleri

Müşteri	Talep
1	50
2	70
3	60
4	50
5	80
6	60
7	60
8	80
9	70
10	20
11	60
12	40

Buna göre en kısa yol prensibine göre oluşturulan rotalar aşağıdaki gibidir:

1. araç için;

0 $\Rightarrow$ 4 $\Rightarrow$ 5 $\Rightarrow$ 2 $\Rightarrow$ 1 $\Rightarrow$ 0

Burada toplam kullanılan kapasite 250 birim, toplam dolaşılan mesafe ise 187 birimdir.

2. araç için;

0 $\Rightarrow$ 3 $\Rightarrow$ 12 $\Rightarrow$ 11 $\Rightarrow$ 10 $\Rightarrow$ 9 $\Rightarrow$ 0

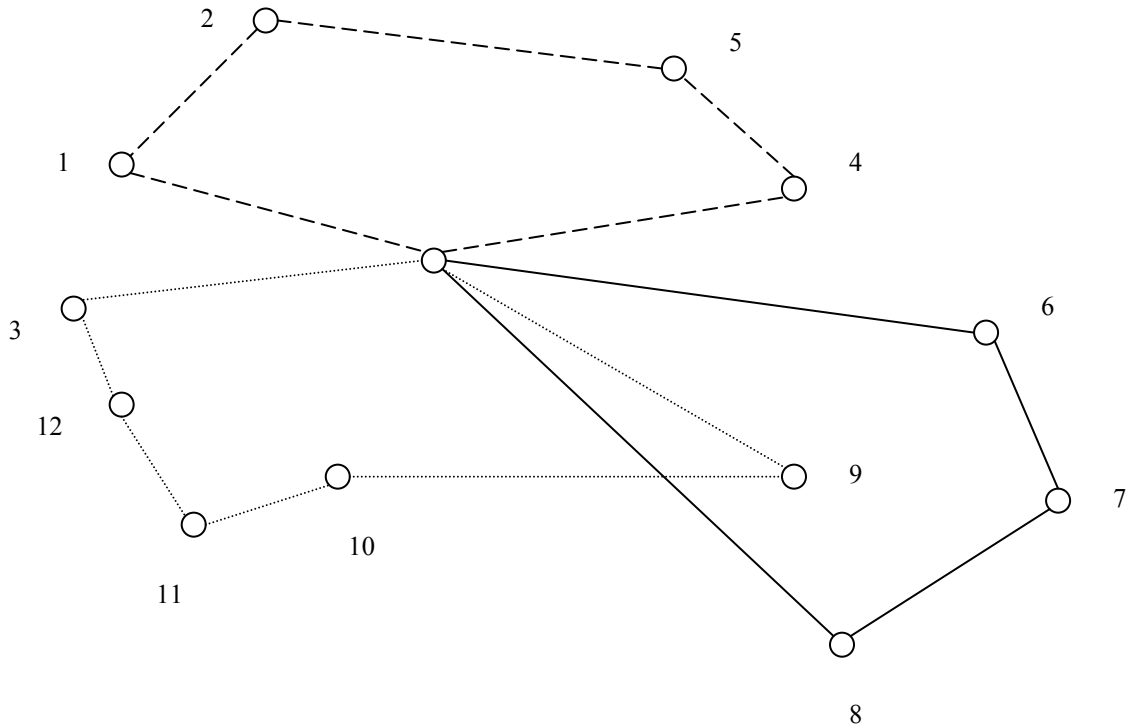
Burada toplam kullanılan kapasite 250 birim, toplam dolaşılan mesafe ise 215 birimdir.

Son olarak 3. araç için;

0 $\Rightarrow$ 6 $\Rightarrow$ 7 $\Rightarrow$ 8 $\Rightarrow$ 0

Bu rotada ise toplam kullanılan kapasite 200 birim, toplam dolaşılan mesafe ise 194,5 birimdir.

Buna göre üç rota için toplam dolaşılan yol 596,5 birim olmaktadır. Oluşturulan rotaları şekil üzerinde gösterecek olursak;



Rotalar üzerindeki çeşitli denemelerden sonra nokta atama yöntemi kullanılarak bu dağıtım sistemi için daha uygun bir çözüm bulunmuştur. Buna göre 4 numaralı müşteri, 6 – 7 – 8 müşterilerini içeren rotaya atandığında toplam dolaşılan mesafe 596,5 birimden 587 birime düşmektedir:

1. araç için;

0 $\Rightarrow$ 5 $\Rightarrow$ 2 $\Rightarrow$ 1 $\Rightarrow$ 0

Burada toplam kullanılan kapasite 200 birim, toplam dolaşılan mesafe ise 172,5 birimdir.

2. araç için;

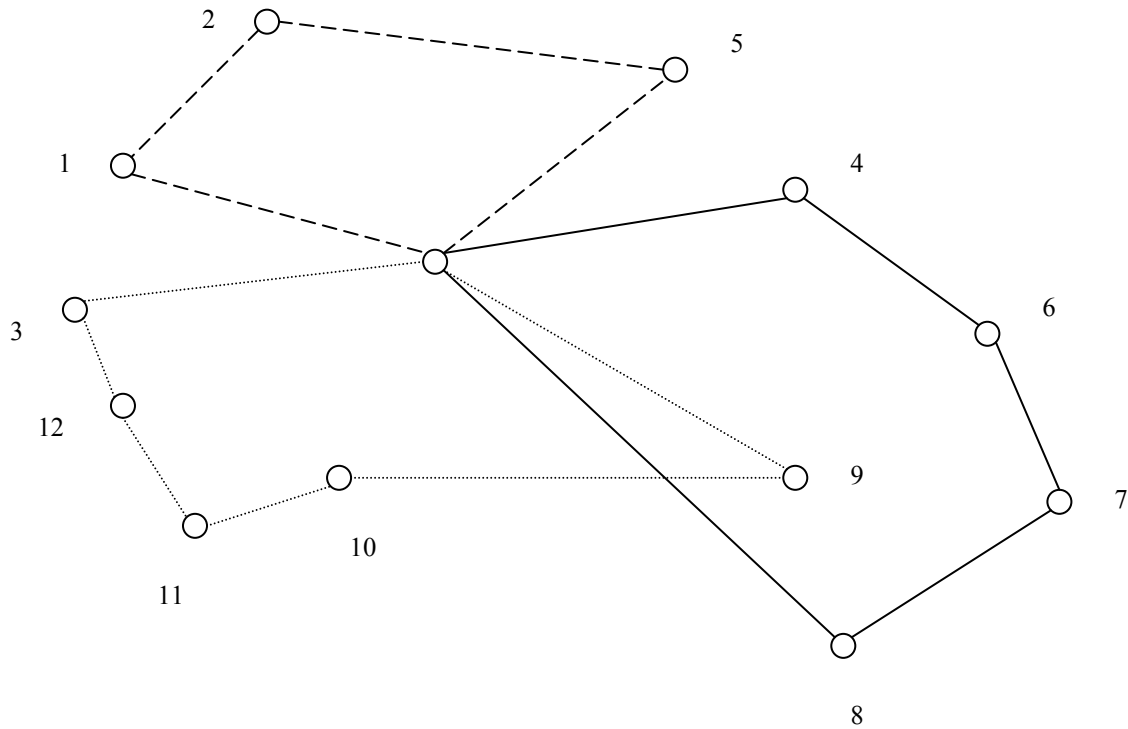
0 $\Rightarrow$ 3 $\Rightarrow$ 12 $\Rightarrow$ 11 $\Rightarrow$ 10 $\Rightarrow$ 9 $\Rightarrow$ 0

Burada toplam kullanılan kapasite 250 birim, toplam dolaşılan mesafe ise 215 birimdir.

Son olarak 3. araç için;

0 $\Rightarrow$ 4 $\Rightarrow$ 6 $\Rightarrow$ 7 $\Rightarrow$ 8 $\Rightarrow$ 0

Bu rotada ise toplam kullanılan kapasite 250 birim, toplam dolaşılan mesafe ise 199,5 birimdir. Rotanın yeni hali aşağıdaki gibidir:



Şimdi de aynı modele Clark-Wright tasarruf algoritmasını uygulayalım. Bunun için her bir ikili müşteri grubu arasında aşağıdaki eşitliğe göre sağlanan tasarruflar hesaplanır:

$$s(i, j) = D(0, i) + D(0, j) - D(i, j)$$

Buna göre modelimiz için oluşan durum aşağıdaki çizelgede gösterilmektedir:

Çizelge 4.3 Her bir ikili karşılaştırma sonucu oluşan tasarruf değerleri

	1	2	3	4	5	6	7	8	9	10	11	12
1	-											
2	97	-										
3	77,5	63	-									
4	26	35	15	-								
5	34,5	46,5	18	44,5	-							
6	29,5	42	9,5	54	51	-						
7	24	10,5	5,5	50	45	98,5	-					
8	28	18,5	12	49	36,5	80,5	107	-				
9	22	18,5	11,5	39,5	33,5	68,5	81,5	88	-			
10	32,5	23,5	34,5	18	12	40	52,5	58	38	-		
11	38,5	26,5	31	4,5	2,5	23,5	35,5	39,5	24	63	-	
12	63	54,5	57	7	4	25	22,5	35	19	54	56	-

Bu tasarruf tablosuna göre, en büyük tasarrufu sağlayan ikiliden başlanarak, kapasite kısıtına da dikkat edilerek eşleştirmeler yapılır. Buna göre rotalar aşağıdaki şekilde oluşturulur:

0 – 4 – 6 – 7 – 8 – 0

Bu rotanın toplam kullandığı kapasite 250 birim, toplam dolaşılan mesafe ise 199,5 birimdir.

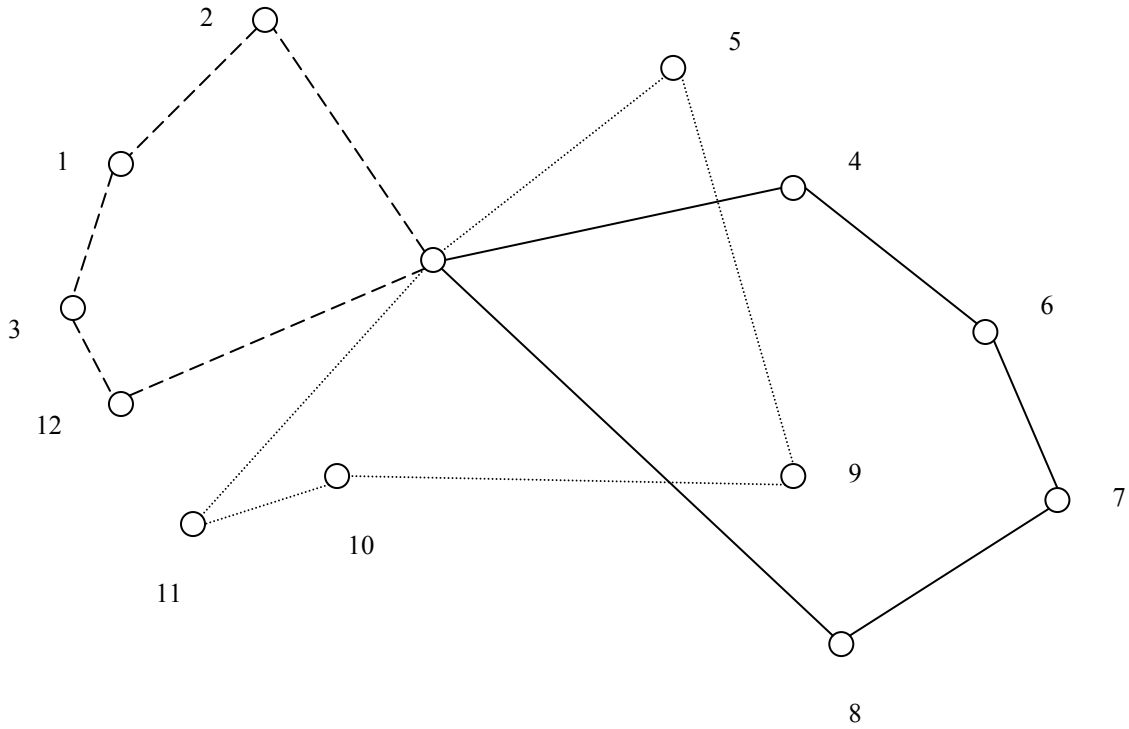
0 – 2 – 1 – 3 – 12 – 0

Bu rotada toplam kullanılan kapasite 220 birim ve toplam dolaşılan mesafe de 170,5 birimdir.

0 – 5 – 9 – 10 – 11 – 0

Son olarak bu rota için toplam kullanılan kapasite 230 birim ve dolaşılan toplam mesafe ise 208,5 birimdir.

Tüm bu verilere göre toplam dolaşılan mesafe ise 578,5 birim çıkmaktadır. Oluşturulan bu rotayı şekil üzerinde gösterecek olursak:



Bu rotalar üzerindeki çeşitli denemelerden sonra, nokta değişimi ve rota değişimi yöntemleri birlikte kullanılarak aşağıdaki rotalar elde edilmiştir:

0 – 2 – 1 – 3 – 12 – 0

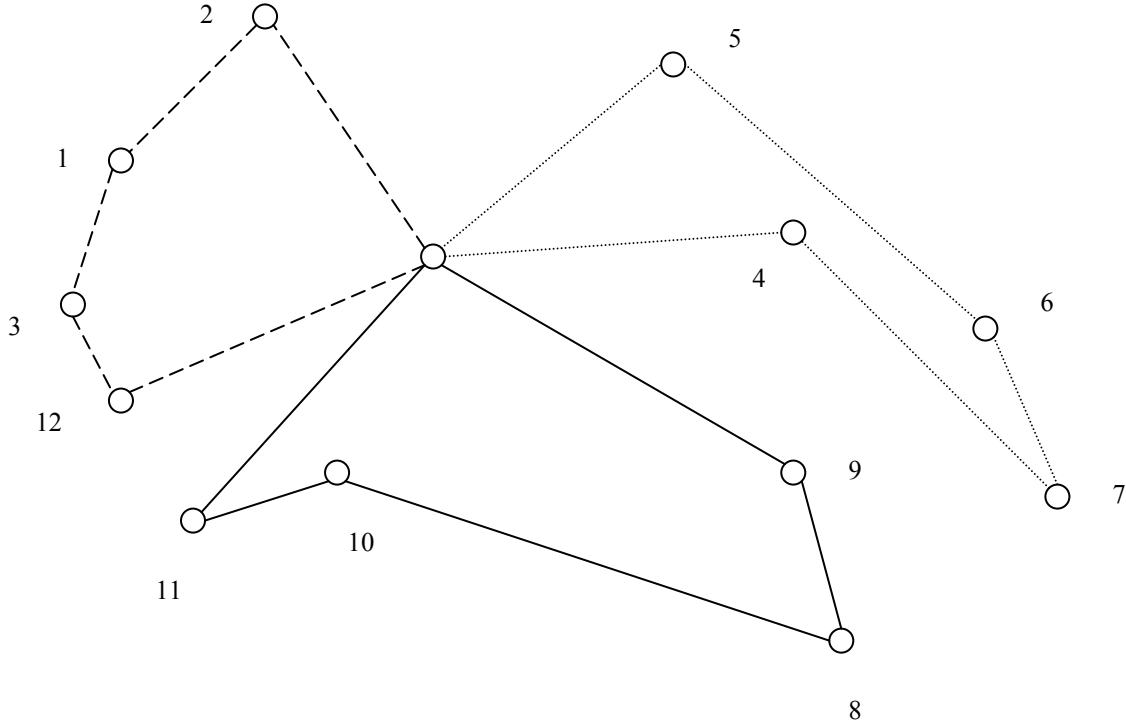
Bu rotada toplam kullanılan kapasite 250 birim ve toplam dolaşılan mesafe 170,5 birimdir.

0 – 9 – 8 – 10 – 11 – 0

Bu rota için toplam kullanılan kapasite ise 230 birim ve toplam dolaşılan mesafe ise 189 birimdir.

0 – 5 – 6 – 7 – 4 – 0

Bu rotada toplam kullanılan kapasite 250 birim ve toplam dolaşılan mesafe de 204,5 birimdir. Buna göre; 5 ve 8 noktaları karşılıklı olarak yer değiştirmişlerdir. Buna göre sonuç olarak toplam dolaşılan mesafe 564 birimdir ve bu şu ana kadar bulunan en düşük maliyettir. Rotanın son hali şu şekildedir:



Şimdi aynı yöntemleri biraz daha geniş ölçekli bir model üzerinde uygulamaya çalışalım. Yeni modelimiz $i = 20$ adet müşteri noktasından oluşmaktadır ve araç kapasitesi $C_0 = 300$ olacaktır. Her bir müşteri için olasılığa bağlı talep değerleri şu şekildedir:

Çizelge 4.4 Her bir müşteri için olasılığa bağlı talep değerleri.

Müşteri (i)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Talep $d(i)$	50	70	50	60	70	40	70	70	80	80	30	80	30	60	60	10	20	10	20	30

Müşteriler ve merkez depo arası mesafeler ise şu şekildedir:

Çizelge 4.5 Müşteriler ve merkez depo arası mesafeler.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0	0	52	35	48,5	76	38,5	29	52	91,5	75	61,5	66,5	53	93,5	59	81,5	68,5	41,5	91	84,5	65,5
1		0	61	55	48	68	34,5	81	87	31	41	62	63,5	85,5	80	76,5	55,5	57	85,5	66,5	70,5
2			0	14,5	44	29,5	47	70,5	34,5	38,5	52	77,5	66,5	53	98	92	83	55,5	53	56	66
3				0	66,5	30	43,5	60,5	68	70	45,5	69	60	47	69,5	69	73,5	65,5	37	52	64
4					0	52	61	50	70,5	44,5	42	59	62,5	30	57	56,5	49,5	79	70,5	62,5	66
5						0	56	54,5	49,5	33,5	68	50,5	39	73,5	64,5	68,5	102	69	71,5	77	51,5
6							0	36,5	30	57,5	60,5	61,5	32	77	56	32	64,5	47	58	84	49,5
7								0	30,5	34	46,5	74	48,5	67	85	33	64	47,5	83	78	67
8									0	32	49,5	67	28,5	84	100,5	42	76,5	56,5	79,5	61,5	60,5
9										0	36,5	67	34	60,5	58,5	40	64	54	30,5	53,5	64
10											0	80	29,5	53	62	49,5	64	59,5	34,5	61	74,5
11												0	53	76	61,5	80	59,5	49	33,5	23	89
12													0	56	56,5	73	54	19,5	55	102	88,5
13														0	75,5	99,5	55,5	35	48	32,5	60,5
14															0	26,5	44	64	37	71	50
15																0	84,5	61,5	53,5	41	44
16																	0	64	48,5	86	56,5
17																		0	28,5	70	65
18																			0	78,75	35,5
19																				0	20,5
20																					0

Bu verilere göre önceki modelde olduğu gibi yine ilk olarak en kısa yol prensibine göre oluşturulan rotalar aşağıdaki gibidir:

Birinci rota;

0 – 6 – 8 – 12 – 17 – 18 – 9 – 0

Bu rota için toplam kullanılan kapasite miktarı 300 birim ve toplam dolaşılan yol ise 241 birimdir.

İkinci rota;

0 – 2 – 3 – 5 – 11 – 19 – 20 – 15 – 0 – 15 – 0

Bu rotada 15 numaralı müşteri noktasında rota hatası oluşmuş, araç kapasitesi aşılmıştır. Bunu sonucu olarak araç merkez depoya dönüp yeniden 15 numaralı müşteriye giderek hizmetini tamamlamıştır. Sonuç olarak bu rotada kullanılan toplam kapasite 300 (+20) birimdir ve

Birinci rota;

0 – 6 – 8 – 12 – 17 – 18 – 9 – 0

Bu rota için kullanılan toplam kapasite 300 birim ve dolaşılan toplam mesafe de 241 birimdir.

İkinci rota;

0 – 2 – 3 – 5 – 11 – 19 – 20 – 15 – 0

Bu rotada kullanılan toplam kapasite 300 birim ve dolaşılan toplam mesafe ise 299 birimdir.

Burada 15 numaralı müşteri talebinin yalnızca 40 birimlik kısmı karşılanmıştır.

Üçüncü rota;

0 – 1 – 10 – 4 – 13 – 16 – 14 – 15 – 0

Bu rotada kullanılan toplam kapasite 300 birim, toplam dolaşılan mesafe ise 372,5 birimdir.

Bu rotada da yine 15 numaralı müşterinin kalan talebinin 10 birimlik kısmı karşılanmıştır.

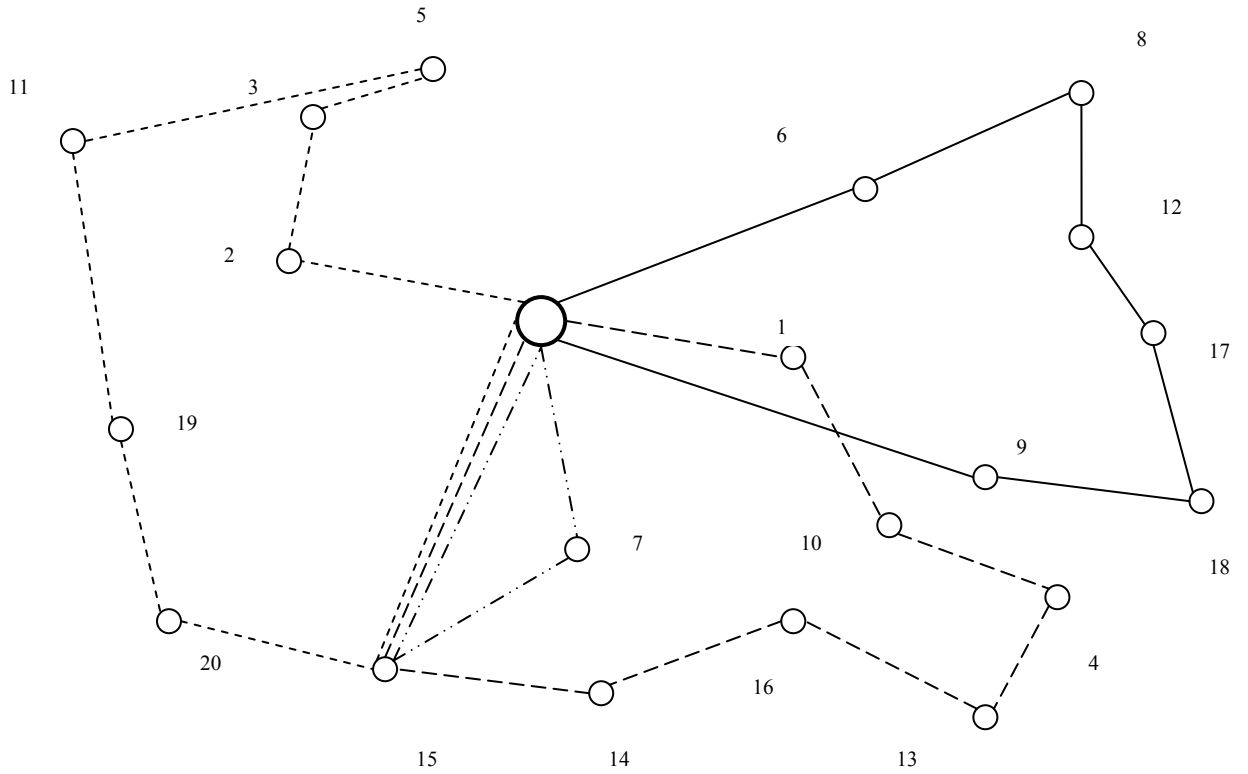
Dördüncü rota;

0 – 7 – 15 – 0

Burada kullanılan toplam kapasite 80 birim ve dolaşılan toplam mesafe 166,5 birimdir. Bu rotada 15 numaralı müşterinin kalan 10 birimlik talebi karşılanmıştır.

Bu verilere göre bu rotada dolaşılan toplam mesafe 1079 birim olmuştur. 15 numaralı müşterinin talebinin rotalara dağıtılması kağıt üzerinde daha iyi bir sonuç verse de gerçek uygulamalarda müşteri memnuniyetsizliği gibi sıkıntılar doğurabilir.

Bu rota sembolik olarak şu şekilde gösterilebilir:



Aynı müşteri noktasının üç farklı araç tarafından da ziyaret edilmesinin getireceği sıkıntı ve doğabilecek ek maliyetleri önlemek için rotada yeniden bir iyileştirilmeye gidilmiştir. Buna göre 15 numaralı müşterinin tüm talebi dördüncü rota dahilinde karşılanmaktadır. Buna göre oluşan rotalar şu şekildedir;

Birinci rota;

0 – 6 – 8 – 12 – 17 – 18 – 9 – 0

Bu rota için kullanılan toplam kapasite 300 birim ve dolaşılan toplam mesafe de 241 birimdir.

İkinci rota;

0 – 2 – 3 – 5 – 11 – 19 – 20 – 0

Bu rota için toplam kapasite kullanımı 260 birim ve toplam dolaşılan yol ise 239 birimdir.

Üçüncü rota;

0 – 1 – 10 – 4 – 13 – 16 – 14 – 0

Burada kullanılan toplam kapasite 290 birim ve dolaşılan toplam mesafe 323,5 birimdir.

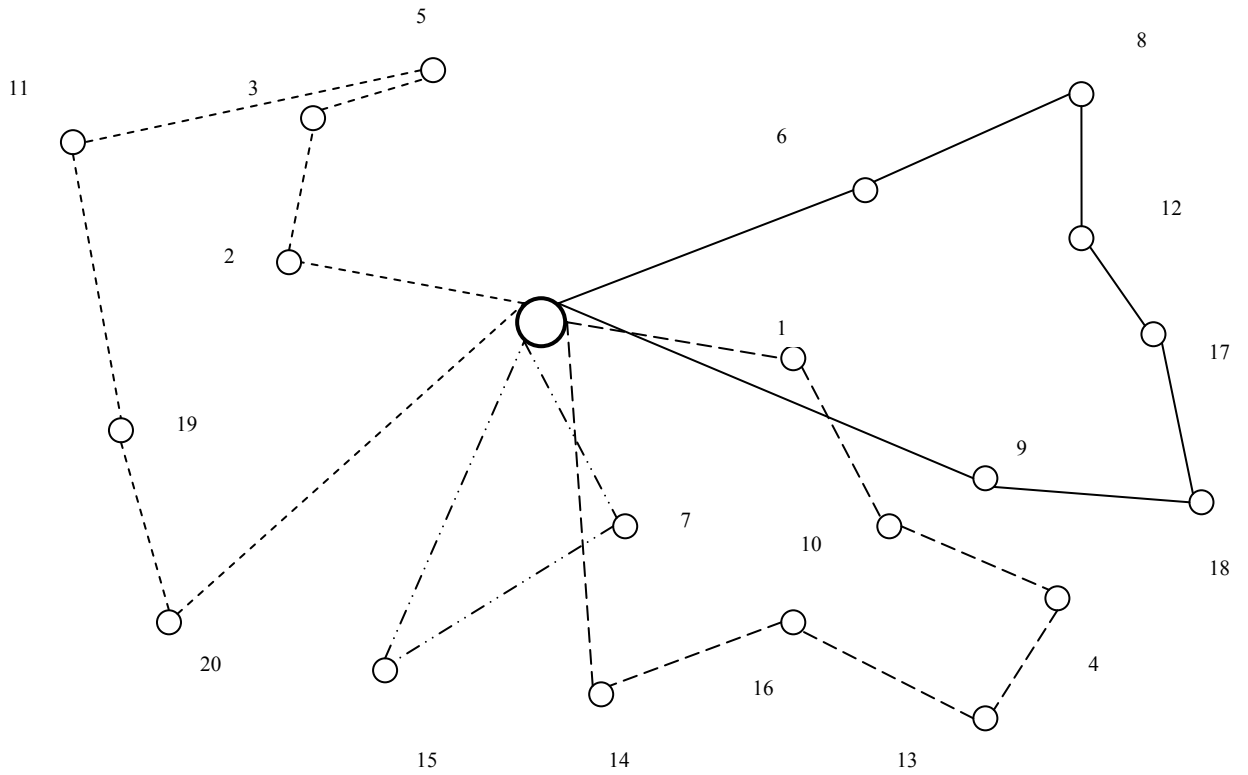
Dördüncü rota;

0 – 7 – 15 – 0

Burada kullanılan toplam kapasite 130 birim ve dolaşılan toplam mesafe de 166,5 birimdir.

Buna göre bu rotalama durumunda dolaşılan toplam mesafe 970 birim olmaktadır. Bu şu ana kadar bulunan en düşük dolaşım miktarıdır.

Son rota şu şekildedir:



Aynı modele bu kez Clark-Wright tasarruf algoritmasını uygulayalım. Bunun için yine önceki modelde olduğu gibi her bir ikili müşteri grubu arasında aşağıdaki eşitliğe göre sağlanan tasarruflar hesaplanır:

$$s(i, j) = D(0, i) + D(0, j) - D(i, j)$$

Buna göre modelimiz için oluşan durum aşağıdaki çizelgede gösterilmektedir:

Çizelge 4.6 Her bir ikili karşılaştırma sonucu oluşan tasarruf değerleri.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	26	45,5	80	22,5	46,5	23	56,5	96	72,5	56,5	41,5	60	31	57	65	36,5	57,5	70	47
2		69	67	44	17	16,5	92	71,5	44,5	24	21,5	75,5	-4	24,5	20,5	21	73	63,5	34,5
3			58	57	34	40	72	53,5	64,5	46	41,5	95	38	61	43,5	24,5	102,5	81	50
4				62,5	44	78	97	106,5	95,5	83,5	66,5	139,5	78	101	95	38,5	96,5	98	75,5
5					11,5	36	80,5	80	32	54,5	52,5	58,5	33	51,5	5	11	58	46	52,5
6						44,5	90,5	46,5	30	34	50	45,5	32	78,5	33	23,5	62	29,5	45
7							113	93	67	44,5	56,5	78,5	26	100,5	56,5	46	60	58,5	50,5
8								134,5	103,5	91	116	101	50	131	83,5	76,5	103	114,5	96,5
9									100	74,5	94	108	75,5	116,5	79,5	62,5	135,5	106	76,5
10										48	85	102	58,5	93,5	66	43,5	118	85	52,5
11											66,5	84	64	68	75,5	59	124	128	43
12												90,5	55,5	61,5	67,5	75	89	35,5	30
13													77	75,5	106,5	100	136,5	145,5	98,5
14														114	83,5	36,5	113	72,5	74,5
15															65,5	61,5	119	125	103
16																46	111	67	77,5
17																	104	56	42
18																		96,75	121
19																			129,5
20																			

Bu tasarruf tablosuna göre, daha önceki modele uygulandığı gibi en büyük tasarrufu sağlayan ikiliden başlanarak, kapasite kısıtına da dikkat edilerek eşleştirmeler yapılır. Buna göre rotalar aşağıdaki şekilde oluşturulur:

Birinci rota;

0 – 20 – 19 – 13 – 4 – 9 – 18 – 8 – 0

Bu rota için kullanılan toplam kapasite miktarı 280 birim ve dolaşılan toplam mesafe 394,5 birimdir.

İkinci rota;

0 – 7 – 15 – 14 – 16 – 10 – 0

Bu rotada kullanılan toplam araç kapasitesi 280 birim ve dolaşılan toplam yol 281 birimdir.

Üçüncü rota;

0 – 6 – 12 – 17 – 11 – 1 – 0

Burada kullanılan toplam kapasite 190 birim ve dolaşılan toplam yol ise 243,5 birimdir.

Dördüncü rota;

0 – 2 – 3 – 5 – 0

Bu rotada kullanılan toplam kapasite miktarı 190 birim ve dolaşılan toplam mesafe 115 birimdir.

Buna göre rota için toplam dolaşılan mesafe 1034 birim olmaktadır. Yukarıdaki rota düzenlerine bakıldığında birinci rotadaki 8 ve 18 numaralı müşteri noktaların rotaya yük bindirdiği anlaşılmaktadır. Buna göre yeni düzenlemelerle 18 numaralı müşteri noktası ikinci rotaya ve 8 numaralı müşteri noktası da üçüncü rotaya kapasite kısıtına da dikkat edilerek eklendiğinde oluşan yeni rotalar şu şekildedir:

Birinci rota;

0 – 20 – 19 – 13 – 4 – 9 – 0

Burada kullanılan toplam kapasite 200 birim ve dolaşılan toplam yol 268 birimdir.

İkinci rota;

0 – 7 – 15 – 14 – 18 – 10 – 16 – 0

Bu rotada kullanılan toplam araç kapasitesi 290 birim ve dolaşılan toplam yol 315,5 birimdir.

Üçüncü rota;

0 – 6 – 8 – 12 – 17 – 11 – 1 – 0

Buradaki kullanılan toplam kapasite miktarı 260 birim ve dolaşılan mesafe 270 birimdir.

Son olarak dördüncü rota;

0 – 2 – 3 – 5 – 0

Bu rotada ise kullanılan toplam araç kapasitesi 190 birim ve dolaşılan toplam yol 115 birimdir.

Bu verilere göre tüm rotalar için dolaşılan toplam mesafe 968,5 birim olmaktadır ve bu şu ana kadar bulunan en düşük miktardır.

5. SONUÇ

Firmaların, global dünyanın rekabet koşullarına ayak uydurabilmesi için, gelişen üretim ve hizmet teknolojilerine uyumlu bir şekilde, üretilen hizmet ve ürünleri talep noktalarına ulaştırırken aynı zamanda hem müşteri memnuniyetini en üst seviyede tutması hem de taşıma maliyetlerini ve kullanılan zamanı minimuma indirmesi gereklidir. Bunun için pazarda lider konumuna ulaşmak isteyen işletmeler, uygun lojistik planları geliştirmek zorundadırlar.

Araç Rotalama Problemi; müşteri, mesafe, zaman, kapasite gibi kısıtlar da göz önüne alınarak ve değişen koşullara (talepteki değişiklikler, trafik koşulları, kapasite aşılması gibi) ayak uydurabilen ve hızlı bir şekilde optimize edilebilen rota planlarının oluşturulması ile çözülerek işletmelere pazarda söz sahibi olma imkanı vermektedir.

Hizmet verilirken oluşacak maliyetlerin en aza indirgenmesi ve zamanın etkili bir şekilde kullanılması optimizasyon işleminin etkin bir şekilde gerçekleştirilmesi ile mümkündür. Araç Rotalama Problemleri çözümü zor optimizasyon problemleridir. Bir çok farklı kısıtı içinde barındırabilen ARP için iki türlü çözüm yöntemi mevcuttur: Kesin ve Yaklaşık çözüm yöntemleri. Kesin çözüm yöntemleri küçük boyutlu problem çözümlerinde etkili sonuçlar verirken daha büyük boyutlu problemlerde hesaplama zorlukları ve işlem uzunlukları nedeniyle zorluklara neden olabilir. Yaklaşık yöntemler ise büyük boyutlu problemler için hızlı ve etkili çözümler sunarak optimale yakın sonuçlar vermekte ve bu yüzden büyük boyutlu problem çözümlerinde tercih edilmektedir. Yaklaşık çözüm yöntemleri klasik sezgisel ve metasezgisel olarak iki gruba ayrılabilir. Klasik Sezgisel yöntemler, metasezgisel yöntemlere göre daha hızlı çözüm üreten ancak orta kalitede sonuç bulan yöntemlerdir. Klasik sezgisel yöntemlerin en bilinen örnekleri Clark-Wright tasarruf algoritması, süpürme algoritması ve taç yaprağı yöntemleri olarak sayılabilir. Metasezgisel yöntemler, klasik sezgisel yöntemlere göre daha uzun işlem süreleri gerektirse de daha uygun çözümler üretebilmektedir.

Bu çalışmada, ARP'nin bazı parametreleri rasgele olduğu zaman ortaya çıkan Stokastik Araç Rotalama problemi için çözüm teknikleri üzerinde durulmuştur. Stokastik Araç Rotalama Problemi, problemdeki belirsizliğin ne olduğuna göre (bu müşteri, talep, dolaşım süresi (mesafe) veya hizmet süresi olabilir), modelleme yöntemine göre ve çözüm yöntemlerine göre (kesin ve yaklaşık) sınıflandırılabilir. Bu çalışmada kullanılan SARP talebin kesin değerinin önceden bilinmediği ancak olasılık dağılımının bilindiği (Stokastik Talepli Araç Rotalama Problemi (STARP) kullanılmıştır.

İki adet farklı boyutlardaki model üzerine önerilen sezgisel algoritma uygulanmıştır. İlk olarak 12 adet talep noktasına sahip bir dağıtım ortamında, talep noktaları araçlara kapasite kısıtları da göz önünde bulundurularak en kısa yol prensibine göre atanmış ve başlangıç rotaları oluşturulmuştur. Daha sonra bu rotalar, rota iyileştirme sezgiselleri (nokta atama, nokta değişimi ve rota değişimi gibi) kullanılarak optimize edilmiş ve ortaya çıkan yeni maliyet değerinin (dolaşılacak toplam mesafe) başlangıç rotasına göre daha iyi bir sonuç verdiği gözlenmiştir.

Bir sonraki adımda ise aynı modele bu kez Clark-Wright tasarruf algoritması uygulanmış ve bunun sonucunda ilk çözüme göre daha iyi bir sonuca ulaşılmıştır. Daha sonra tasarruf algoritması ile bulunan sonuç yine rota iyileştirme sezgiselleri kullanılarak optimize edilmiş ve bu model için bulunan en iyi sonuç ortaya çıkmıştır.

İkinci modelde ise 20 adet talep noktası içeren daha geniş ölçekli bir dağıtım ortamı oluşturulmuş ve rota oluşturma sezgiseli yardımıyla (en kısa yol prensibi) başlangıç rotası oluşturulmuştur. Bu esnada iki kez rota hatası ortaya çıkmış ve araç iki ayrı rotada aynı müşteriye iki kez uğrayarak hizmetini tamamlamıştır. Bunun sonucunda oluşan artı maliyetleri gidermek için rota üzerinde sezgisel iyileştirme yöntemleri uygulanmış ve yeni bir rota oluşturulmuştur. Bu kez daha iyi bir sonuç elde edilmesine karşın aynı müşterinin üç farklı rotada da bulunmasının doğuracağı sıkıntıları (ek maliyetler, müşteri memnuniyetsizliği gibi) giderebilmek için rotada yeniden iyileştirilmeye gidilmiştir. Rotanın son hali en düşük maliyetle en iyi sonucu vermektedir.

Aynı modele bu kez yine Clark-Wright tasarruf algoritması uygulanarak sonuçlar karşılaştırılmıştır. Tasarruf algoritmasının verdiği ilk sonuç, en kısa yol prensibinin verdiği ilk sonuca göre daha büyük bir maliyet gösterse de daha sonraki aşamalarda yine rota iyileştirme sezgiselleri kullanılarak en iyi sonuca ulaşılmıştır.

Sezgisel yöntemler gerek hızlı çözüm olanağı sağlaması gerek de diğer yöntemlere göre daha basit işlemler içermesi açısından büyük ölçekli ve dinamik olarak değişen parametrelere (talep, trafik durumu vb.) sahip problemlerde yaygın olarak kullanılmaktadır.

KAYNAKLAR

- Barnhart, C. (2000), "Planning and Control of Transportation Systems: The Stochastic Knapsack Problem with Random Weights", New England University Transportation Center, USA.
- Bertsimas, D. J. (1992), "A Vehicle Routing Problem with Stochastic Demand", *Operations Research* 40(3):574-585.
- Bianchi, L., Briattar, M., Chiarandini, M., Manfrin, M., Mastrolilli, M., Paquete, L., Rossi-Doria, O. ve Schiavinotto, T. (2006), "Hybrid Metaheuristics for the Vehicle Routing Problem with Stochastic Demands", *Journal of Mathematical Modelling and Algorithms* 5(1):91-110.
- Chan, Y., Carter, W.B. ve Burnes, M. D. (2001), "A Multiple-Depot, Multiple-Vehicle, Location-Routing with Stochastically Processed Demands", *Computers and Operations Research* 28:803-826.
- Chepuri, K. ve Homem-De-Mello, T. (2005), "Solving the Vehicle Routing Problem with Stochastic Demands Using the Cross-Entropy Method", *Annals of Operations Research*, 134:153-181.
- Cheung, R. K. ve Hang, D. D. (2003), "A Time –Window Sliding Procedure for Driver-Task Assignment with Random Service Times", *IIE Transactions* 35:433-444.
- Cheung, R. K., Hang, D. D. ve Shi, N. (2005), "A Labeling Method for Dynamic Driver-Task Assignment with Uncertain Task Durations", *Operations Research Letters*, 33:411-420.
- Cheung, R. K., Xu, D. ve Guan, Y. (2004), "Multiple-Trip Delivery Problem with Stochastic Demands", Grant HKUST6217/02E of the Research Grant Council of Hong Kong.
- Cordeau, J-F, Gendreau, M., Laporte, G., Potuin, J-Y. ve Semet, F. (2002), "A Guide to Vehicle Routing Heuristics", *Journal of the Operational Research Society*, 53:512-522.
- Demirel, T., Karahan, A. ve Demirel, N. Ç. (2005), "The Vehicle Routing Problem with Stochastic Demands", *Proceedings of the 35th International Conference on Computers and Industrial Engineering*, 1063-1068.
- Erera, A. L. ve Daganzo, C. F. (2003), "A Dynamic Scheme for Stochastic Vehicle Routing", *Transportation Research Part B*, 2006.
- Evans, J. R. ve Minieka, E. (1992), *Optimization Algorithms for Networks and Graphs*, Marcel Dekker Inc., New York.
- Fleury, G. Philippe, L. ve Prins, C. (2005), "Stochastic Capacitated Arc Routing Problem", *Research Report LIMOS/RR-05-12*.
- Frazzoli, E. ve Bullo, F. (2004), "Decentralized Algorithms for Vehicle Routing in a Stochastic Time-Varying Environment", *43rd IEEE Conference on Decision and Control* December 14-17, 2004:3357-3363.
- Fu, L., (2002), "Scheduling Dial-a-Ride Paratransit Under Time-Varying, Stochastic Congestion", *Transportation Research Part B* 36, 485-506.
- Gendreau, M., Pelleu, J., Seguin, R. ve Soriano, P. (2001), "Improving Robustness of a Tabu Search Algorithm for Stochastic Vehicle Routing Problems via Diversification", *4th Metaheuristics International Conference*, Porto, Portugal, July 16-20, 2001:435-438.

- Gupta, D. K. (2003), "A New Algorithm to Solve Vehicle Routing Problems (VRPs)", *International Journal of Computer Mathematics*, 80(3):267-274.
- Haughton, M. A. (2002), "Route Reoptimization's Impact on Delivery Efficiency", *Transportation Research Part E*, 38:53-63.
- Haughton, M. A. (1998). "The Performance of Route Modification and Demand Stabilization Strategies in Stochastic Vehicle Routing", *Transportation Research*, 32(8):551-566.
- Haughton, M. A. ve Stenger, A. J. (1999), "Comparing Strategies for Addressing Delivery Shortages in Stochastic Demand Settings", *Transportation Research Part E*:25-41.
- Haugland, D., Ho, S. C. ve Laporte, G. (2005), "Designing Delivery Districts for the Vehicle Routing Problem with Stochastic Demands", *European Journal of Operational Research*, 180(3):997-1010.
- Hjorring, C. ve Holt, J. (1999), "New Optimality Cuts for a Single-Vehicle Stochastic Routing Problem", *Annals of Operations Research*, 86:569-584.
- Hu, T-Y., Liao, T-Y. ve Lu, Y-C. (2003), "A Study on the Solution Approach for Dynamic Vehicle Routing Problems Under Real-Time Information", *Transportation Research Record* 1857:102-108.
- Hvattum, L. M., Lokketangen, A. ve Laporte, G. (2003), "A Heuristic Solution Method to a Stochastic Vehicle Routing Problem", *Proceedings NIK 2003 – Norwegian Informatics Conference*.
- Jula, H., Dessouky, M. ve Ioannou, P. (2006), "Truck Route Planning in Non-Stationary Stochastic Networks with Time-Windows at Customer Locations", *IEEE Transactions on Intelligent Transportation Systems*, 37:51-63.
- Kim, S., Lewis, M. E. ve White, C. C. (2005), "Optimal Vehicle Routing With Real-Time Traffic Information", *IEEE Transactions on Intelligent Transportation Systems*, 4(2):178-188.
- Mak, K. L. ve Guo, Z. G. (2004), "A Genetic Algorithm for Vehicle Routing Problems with Stochastic Demand and Soft Time Windows", *Proceedings of the 2004 IEEE Systems and Information Engineering Design Symposium, Virginia, USA*.
- Pisinger, D. ve Ropke, S. (2007), "A General Heuristic for Vehicle Routing Problems", *Computers and Operations Research*, 34(8):2403-2435.
- Potvin, J-Y., Xu, Y. ve Benyahia, I. (2004), "Vehicle Routing And Scheduling With Dynamic Travel Times", *Computers and Operations Research*, 33 (2006) 1129–1137.
- Secomandi, N. (2003), "Analysis of a Rollout Approach to Sequencing Problems with Stochastic Routing Applications", *Journal of Heuristics*, 9:321-352.
- Secomandi, N. (2000), "Comparing Neuro-Dynamic Programming Algorithms for the Vehicle Routing Problem with Stochastic Demands", *Computers and Operational Research*, 27:1201-1225.
- Shen, Z., Ordenez, F. ve Dessouky, M. M. (2007), "The Stochastic Vehicle Routing Problem for Minimum Unmet Demand", *Springer Series on Optimization and its Applications*.
- Swihart, M. R. ve Papastavrou, J. D. (1999), "A Stochastic and Dynamic Model for the Single-Vehicle Pick-Up and Delivery Problem", *European Journal Operational Research* 114:447-464.

Tarantilis, C. D., Ioannou, G. ve Probst, G. (2005), “Advanced Vehicle Routing Algorithms for Complex Operations Management Problems”, *Journal of Food Engineering*, 70:455-471.

Topaloglu, H. (2006), “A Parallelizable Dynamic Fleet Management Model with Random Travel Times”, *European Journal of Operational Research*, 175:782-805.

Toth, P. ve Vigo, D. (2000), *The Vehicle Routing Problem*, Universita degli Studi di Bologna, Bologna, Italy.

Van Breedam, A. (2001), “Comparing Descent Heuristics and Metaheuristics for the Vehicle Routing Problem”, *Computers and Operations Research*, 28:289-315.

Wang, X. ve Regan, A. C. (2000), “Alternative Assignment Models for Time Constrained Local Fleet Assignment in which Service and Travel Times are Stochastic”, *Institute of Transportation Studies and Department of Civil and Environmental Engineering, University of California, Irvine, CA 92697-3600, USA.*

ÖZGEÇMİŞ

Doğum Tarihi 09.11.1982

Doğum Yeri İstanbul

Lise 1996-2000 Haydarpaşa Lisesi

Lisans 2000-2004 Yıldız Teknik Üniversitesi Elektrik-Elektronik Fak.
Elektrik Mühendisliği Bölümü

Yüksek Lisans 2004-2007 Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü
Endüstri Mühendisliği Anabilim Dalı
Sistem Mühendisliği Programı