
GAZİ ÜNİVERSİTESİ
Gazi
1926

YEŞİL EŞZAMANLI TOPLA DAĞIT ARAÇ ROTALAMA PROBLEMİ İÇİN BİR HİPER SEZGİSEL ALGORİTMA

Büşra OLGUN

YÜKSEK LİSANS TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ ANA BİLİM DALI

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

MAYIS 2019

Büşra OLGUN tarafından hazırlanan "YEŞİL EŞZAMANLI TOPLA DAĞIT ARAÇ ROTALAMA PROBLEMİ İÇİN BİR HİPER SEZGİSEL ALGORİTMA" adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Endüstri Mühendisliği Ana Bilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman: Prof. Dr. Fulya ALTIPARMAK

Endüstri Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum. ...……

Başkan: Prof. Dr. Hadi GÖKÇEN

Endüstri Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum. ……...

Üye: Dr. Öğr. Üyesi Çağrı KOÇ

İşletme Ana Bilim Dalı, Ankara Sosyal Bilimler Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum. ……...

Tez Savunma Tarihi: 03/05/2019

Jüri tarafından kabul edilen bu tezin Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

……..……

Prof. Dr. Sena YAŞYERLİ

Fen Bilimleri Enstitüsü Müdürü

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Büşra OLGUN

03/05/2019

YEŞİL EŞZAMANLI TOPLA DAĞIT ARAÇ ROTALAMA PROBLEMİ İÇİN BİR HİPER SEZGİSEL ALGORİTMA

(Yüksek Lisans Tezi)

Büşra OLGUN

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Mayıs 2019

ÖZET

İşletmeler günümüzün rekabetçi koşullarında ayakta kalabilmek için maliyetlerinin önemli bir parçası olan lojistik maliyetlerini azaltmaya çalışmaktadırlar. Bu durum, işletmeleri lojistik faaliyetlerini daha etkin bir şekilde sürdürmenin yollarını aramaya zorlamaktadır. Araç rotalama problemi (ARP), bir grup müşterinin taleplerini karşılamak için sınırlı sayıda aracın bir veya birden fazla amacı eniyileyecek şekilde rotalanması olarak tanımlanır. Lojistik faaliyetlerde, tesislerden müşterilere dağıtım yapmanın yanısıra müşterilerden tesislere ürün toplamak da söz konusu olabilmektedir. Bu tezde, müşterilerin toplama ve dağıtım taleplerinin eşzamanlı karşılandığı ARP dikkate alınmıştır. Bu problem literatürde eşzamanlı topla-dağıt ARP (ETDARP) olarak bilinir. Son zamanlarda çevresel problemlere olan farkındalığın artması ve lojistik faaliyetlerde kullanılan araçların zararlı gaz salınımları, ARP'de çevresel etkilerin göz önüne alınmasını gerekli kılmıştır. Bu tezde ele alınan ETDARP için yakıt tüketimi maliyetinin enazlanması amaçlanmış ve problem yeşil ETDARP (Y-ETDARP) olarak adlandırılmıştır. Y-ETDARP'ın çözümü için öncelikle literatürden bir matematiksel model uyarlanmıştır. Problem, NP-zor problemler sınıfında yer almaktadır. Bu nedenle, orta ve büyük boyutlu Y-ETDARP'ın çözümü için iteratif yerel arama ve değişken komşu iniş tabanlı bir hipersezgisel algoritma (İYA_HS) geliştirilmiştir. İYA_HS ile farklı boyuttaki Y-ETDARP çözülmüş ve toplam yakıt tüketimi maliyetinde %3,16 azalma sağlanmıştır. Ayrıca, literatürde ETDARP için geliştirilmiş olan sezgisel algoritmalar ile İYA_HS çözüm kalitesi ve çözüm zamanı açısından karşılaştırılmış ve bu algoritmalar ile benzer performansa sahip olduğu görülmüştür.

Bilim Kodu	:	90615
Anahtar Kelimeler	:	Eşzamanlı topla dağıt araç rotalama problemi, hipersezgisel, yeşil lojistik, iteratif yerel arama
Sayfa Adedi	:	71
Danışman	:	Prof. Dr. Fulya ALTIPARMAK

A HYPER HEURISTIC ALGORITHM FOR GREEN SIMULTANEOUS PICKUP AND DELIVERY PROBLEM

(M. Sc. Thesis)

Büşra OLGUN

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

May 2019

ABSTRACT

Companies are looking for ways to reduce their logistics costs which is an important part of operating costs in order to survive in today's competitive conditions. This has forced firms to look for ways to maintain their logistics activities more effectively. Vehicle routing problem (ARP) is defined as routing the limited number of vehicles to meet the demands of a group of customers to optimize one or more specific objectives. In logistics activities, it is possible to collect products from customers to facilities as well as distribution from facilities to customers. In this thesis, we studied ARP where collection and distribution requests of customers were met simultaneously. This problem is known as simultaneous pickup and delivery VRP (SPDVRP) in the literature. Recently, increased awareness of environmental issues and the harmful gas emissions of the vehicles used in logistics activities have necessitated consideration of environmental impacts in vehicle routing problems. The aim of this study is to minimize the cost of fuel consumption for SPDVRP and the problem was named as green SPDVRP (G-SPDVRP). Firstly a mathematical model was adapted from the literature to solve the G-SPDVRP. The problem is in the class of NP-hard problems. Therefore, an iterative local search and variable neighborhood descent based hyper heuristic (ILS_HH) has been developed to obtain solutions for medium and large scale G-SPDVRPs. The developed ILS_HH algorithm provides a reduction of the total cost of consumption for G-SPDVRP by 3.16%. In addition, ILS_HH and the heuristic algorithms developed for SPDVRP in the literature are compared in terms of solution quality and solution time, and similar performance is observed with these algorithms.

Science Code	:	90615
Key Words	:	Simultaneous pickup and delivery problem, hyper heuristic, green logistics, iterated local search
Page Number	:	71
Supervisor	:	Prof. Dr. Fulya ALTIPARMAK

TEŞEKKÜR

Çalışmalarım boyunca çok değerli yardım ve katkılarını hiçbir zaman esirgemeyerek beni yönlendiren tez danışmanım Sayın Prof. Dr. Fulya ALTIPARMAK'a sonsuz teşekkürlerimi sunarım. Ayrıca birikim ve tecrübelerinden yararlandığım Sayın Doç. Dr. İsmail KARAOĞLAN'a teşekkürü borç bilirim.

Tez çalışmalarım boyunca her zaman desteğini esirgemeyen eşim Mehmet Onur OLGUN'a, kardeşime, anneme ve babama sonsuz teşekkürlerimi sunarım.

İÇİNDEKİLER

Sayfa

ÖZET iv

ABSTRACT v

TEŞEKKÜR vi

İÇİNDEKİLER vii

ÇİZELGELERİN LİSTESİ ix

ŞEKİLLERİN LİSTESİ x

SİMGELER VE KISALTMALAR xi

1. GİRİŞ 1

2. YEŞİL EŞZAMANLI TOPLA DAĞIT ARAÇ ROTALAMA PROBLEMİ 3

2.1. Yeşil Lojistik 3

2.2. Araç Rotalama Problemi (ARP) 5

2.3. Eşzamanlı Topla Dağıt Araç Rotalama Problemi 8

2.4. Yeşil Araç Rotalama Problemi 9

3. LİTERATÜR ARAŞTIRMASI 13

3.1. ETDARP için Literatür Araştırması 13

3.2. Y-ARP için Literatür Araştırması 14

4. Y-ETDARP İÇİN MATEMATİKSEL MODEL VE SEZGİSEL YAKLAŞIM 19

4.1. Y-ETDARP için Matematiksel Model 19

4.2. Sezgisel Yaklaşım 23

4.2.1. Hipersezgisel 23

4.2.2. İteratif yerel arama algoritması 25

4.2.3. Değişken komşu arama algoritması 27

Sayfa

5. Y-ETDARP İÇİN İYA_HS ALGORİTMASI 29
5.1. Çözüm Gösterimi 29
5.2. Başlangıç Çözümünün Oluşturulması 30
5.3. Komşuluk Yapıları 30
5.4. Komşuluk Yapısı Seçim Fonksiyonu 35
5.5. İYA_HS algoritmasının adımları 36
6. DENEYSEL ANALİZ 41
6.1. Test Problemleri 41
6.2. İYA_HS Parametreleri 42
6.3. Y-ETDARP için Performans Analizi 43
6.4. ETDARP için İYA_HS Algoritmasının Performans Analizi 47
6.5. İYA_HS için Duyarlılık Analizi 52
6.5.1. Komşuluk yapısı havuzu için analiz 52
6.5.2. Komşuluk seçim fonksiyonu için analiz 54
6.5.3. Yerel arama algoritması için analiz 56
7. SONUÇ VE DEĞERLENDİRME 59
KAYNAKLAR 63
ÖZGEÇMİŞ 71

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

Çizelge 3.1. Y-ARP ile ilgili çalışmalar..... 17

Çizelge 4.1. ARP için hipersezgisel kullanan çalışmalar..... 24

Çizelge 6.1. Y-ETDARP ve ETDARP için İYA_HS ile elde edilen sonuçlar 44

Çizelge 6.2. ETDARP için Dethloff (2001) test problemleri sonuçları 48

Çizelge 6.3. ETDARP için Salhi ve Nagy (1999) test problemleri sonuçları..... 51

Çizelge 6.4. Dethloff (2001) test problemleri için İYA_HS ve İYA_HS-1 sonuçları.... 53

Çizelge 6.5. Dethloff (2001) test problemleri için İYA_HS ve İYA_HS-2 sonuçları.... 55

Çizelge 6.6. Dethloff (2001) test problemleri için İYA_HS ve İYA_HS-3 sonuçları.... 58

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Şekil 2.1. ARP'nin temel tipleri ve bunların bağlantıları 6

Şekil 4.1. İteratif yerel arama algoritmasının adımları 25

Şekil 4.2. Değişken komşu iniş algoritmasının adımları 28

Şekil 5.1. İYA_HS'de çözüm gösterimi 29

Şekil 5.2. En yakın komşu sezgiselinin adımları 30

Şekil 5.3. İYA_HS algoritmasının adımları 37

Şekil 5.4. YAA1_HS adımları 38

Şekil 5.5. YAA2_HS adımları 39

Şekil 5.6. İYA_HS için karıştırma yaklaşımı adımları 40

Şekil 6.1. YAA-3 adımları 57

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	Açıklamalar
c_0	birim yakıt maliyeti
c_{ij}	i ve j müşterileri arasındaki mesafe
d_i	i müşterisinin dağıtım talebi
p_i	i müşterisinin dağıtım talebi
M	araç sayısı
N	müşteri sayısı
N_c	müşteriler kümesi
p_i	i müşterisinin toplama talebi
Q	herbir aracın kapasite miktarı

Kısaltmalar	Açıklamalar
ARP	Araç rotalama problemleri
AUARP	Açık uçlu araç rotalama problemi
BTARP	Bölünmüş talepli araç rotalama problemi
CF	Seçim fonksiyonu
ÇDARP	Çok depolu araç rotalama problemi
ÇS	Çözüm süresi
DKA	Değişken komşu arama
DKİ	Değişken komşu iniş algoritması
EİÇ	Literatürde bilinen en iyi çözüm
ETDARP	Eşzamanlı topla dağıt araç rotalama problemi
EYK	En yakın komşu
GDKA	Genel değişken komşu arama algoritması
HARP	Heterojen araç rotalama problemi
İDKA	İndirgenmiş değişken komşu arama algoritması
İYA	İteratif yerel arama algoritması

Kısaltmalar	Açıklamalar
İYA_HS	İteratif yerel arama tabanlı hipersezgisel algoritma
KKARP	Kapasite kısıtlı araç rotalama problemi
KTDARP	Karma topla-dağıt araç rotalama problemi
MKARP	Mesafe kısıtlı araç rotalama problemi
ÖDSTARP	Önce dağıt sonra topla araç rotalama problemi
PARP	Periyodik araç rotalama problemi
TARP	Topla araç rotalama problemi
TDARP	Topla-dağıt araç rotalama problemi
TDKA	Temel değişken komşu arama algoritması
YAA	Yerel arama algoritması
Y-ARP	Yeşil araç rotalama problemi
Y-ETDARP	Yeşil eşzamanlı topla dağıt araç rotalama problemi
ZBARP	Zaman bağımlı araç rotalama problemi
ZPARP	Zaman pencereli araç rotalama problemi

1. GİRİŞ

Araç rotalama problemleri (ARP), lojistik ve dağıtım yönetiminde geniş uygulama alanına sahiptir. İşletmeler rekabetin iyice arttığı günümüz şartlarında maliyetlerini düşürmek, daha az maliyetle daha etkin şekilde çalışmak zorundadırlar. Bu durum işletmeleri hem işletme içi çalışma alanlarında hem de tedarik zincirinde çok önemli bir yer tutan dağıtım faaliyetlerinde daha verimli olmanın yollarını aramaya itmiştir. Müşteri siparişlerinin gönderilebilmesi için eniyi rotalar kümesini elde etmeyi amaçlayan ARP, dağıtım maliyetlerinin düşürülmesinde büyük ölçüde etkili olmaktadır. İlk olarak Dantzig ve Ramser (1959) tarafından tanımlanan ARP, işletmelerin dağıtım problemlerinin yanısıra, çöp toplama, okul servisleri rotalama, afet durumlarında yardımların ulaştırılması gibi araçların rotalanmasının gerektiren birçok uygulama alanına sahiptir. Literatürde, müşterilerin ve kullanılan araçların özelliklerine, çevresel etmenlere bağlı birçok ARP tipi tanımlanmıştır. Bu tez kapsamında, tesislerden müşterilere dağıtımın yanısıra müşterilerden tesislere toplama yapılması durumunun da söz konusu olduğu, eşzamanlı topla dağıt ARP (ETDARP) ele alınmıştır.

Son zamanlarda çevre için hem bireysel hem de hükümetler bazında hızla artan hassasiyet söz konusudur. Bu nedenle, işletmeler faaliyetlerini gerçekleştirirken çevresel etkilerini de yasal kısıtlamalardan dolayı dikkate almak zorundadırlar. Lojistik faaliyetlerinde ise bu durum yakıt tüketim oranlarının ve zararlı gaz salınımlarının fazla olması sebebiyle ekstra önem kazanmaktadır. Dolayısıyla, son yıllarda literatürde çalışılan ARP'de çevreye olan zararlı etkilerin azaltılması da bir amaç olarak dikkate alınmaya başlanmıştır. Bu tez kapsamında da, ETDARP için taşınan yük miktarına bağlı olarak hesaplanan toplam yakıt tüketimi maliyetinin enazlanması amaçlanmış ve problem yeşil ETDARP (Y-ETDARP) olarak adlandırılmıştır. Bilindiği kadarıyla bu amaç fonksiyonunu ile Y-ETDARP ilk kez bu tez kapsamında ele alınmıştır.

Y-ETDARP'ın amaç fonksiyonu katedilen mesafenin enazlanması olarak değiştirildiğinde problem ETDARP'a dönüşmektedir. ETDARP, NP-zor problemler sınıfında yer aldığından dolayı Y-ETDARP'da bu sınıfta yer almaktadır. Bu nedenle orta ve büyük boyutlu problemler için kısa sürede kaliteli çözümlerin elde edilmesi için iteratif yerel arama tabanlı bir hipersezgisel algoritma (İYA_HS) geliştirilmiştir.

Çalışmanın ikinci bölümünde yeşil lojistik, ARP, ETDARP ve Y-ETDARP ile genel bilgi verilmiştir. Üçüncü bölümde ETDARP ve Y-ARP için literatür araştırması, dördüncü bölümde Y-ETDARP için matematiksel model ve geliştirilen İYA_HS algoritmasında kullanılan sezgisel yaklaşımlar verilmiştir. Beşinci bölümde Y-ETDARP için geliştirilen İYA_HS algoritmasının işleyişi ve özellikleri anlatılmış, altıncı bölümde İYA_HS algoritmasına ilişkin deneysel analizlere yer verilmiştir. Son bölümde ise çalışmadan elde edilen sonuçlar özetlenmiş ve gelecekte yapılabilecek çalışmalardan bahsedilmiştir.

2. YEŞİL EŞZAMANLI TOPLA DAĞIT ARAÇ ROTALAMA PROBLEMİ

İşletmelerin dağıtım ağlarının çok geniş olması ve lojistik faaliyetlerinin yüksek maliyetlerle gerçekleşmesi dağıtım yapan araçların nasıl yönlendirileceği problemi çok karmaşık ve önemli bir lojistik problemi haline gelmiştir. Çevresel kaygılarla birlikte problem daha da karmaşık bir hal almıştır. Bu bölümün ilk kısmında yeşil lojistik hakkında bilgi verilmiştir. İkinci kısımda, işletmelerin dağıtım/toplama faaliyetlerini en küçük maliyetlerle yapmasına imkan tanıyan ARP'nin tanımı ve çeşitleri verilmiştir. Üçüncü kısımda, dağıtım ve toplama faaliyetlerini aynı araçlarla yaparak maliyetten tasarrufun yanısıra çevresel zararların da azalmasını sağlayan topla-dağıt ARP'den (TDARP) bahsedilmiştir. Son olarak, günümüzde çevre bilincinin artmasıyla oldukça önem kazanan ve çevresel zararların probleme doğrudan kısıt veya amaç fonksiyonu olarak eklendiği yeşil ARP'den (Y-ARP) bahsedilmiştir.

2.1. Yeşil Lojistik

20. yüzyıldan itibaren sanayi devriminin bir sonucu olarak hızla gelişen sanayi ve teknoloji, son dönemlerde dünya çapında endişe yaratacak düzeyde çevre kirliliğine ve küresel ısınmaya sebep olmuştur. Ayrıca, dünyada hızla artan nüfus ile birlikte yüksek miktarlarda üretilen ürünlerin büyük dağıtım ağları ile müşterilere ulaştırılmasını zorunlu kılmıştır. İşletmelerin zorlaşan rekabetçi piyasa şartlarında ayakta kalabilmesi, düşük maliyetlerle üretim ve dağıtım faaliyetlerini gerçekleştirebilmesine bağlıdır. Günümüzde buna ek olarak doğal kaynakların azalması, çevre kirliliği ve küresel ısınma konusundaki hassasiyet ile işletmeler, tüm tedarik zincirleri boyunca çevreye karşı duyarlı faaliyetleri gerçekleştirme çabası içine girmiştir. Dolayısıyla, işletmelerde yeşil tedarik zinciri yönetimi kavramı önem kazanmıştır. Yeşil tedarik zinciri, tüm ürün ve hizmetlerin tedarikçiden başlayıp en son aşamada müşteriye ulaşana kadar geçen süreçte faaliyetlerin çevreye verilen zararı enaza indirme çabasıyla gerçekleştirilmesi olarak ifade edilebilir.

Lojistik, ürünleri tedarik zinciri boyunca taşımak için gereken tüm faaliyetlerin toplamını ifade etmektedir ve tedarik zinciri yönetiminin tüm temel aşamalarında yer almaktadır. Lojistik yönetimi, ürünlerin üretim kaynaklarından son tüketiciye ulaştırılması sürecinde her aşamada yer alması sebebiyle ekonomik hayatın çok önemli bir unsuru olarak görülmektedir. Aynı zamanda, lojistik faaliyetleri tedarik zinciri içerisinde yüksek maliyetli ve çevreye

verilen zararın fazla olduğu faaliyetler içerisinde büyük yere sahiptir (Dişkaya ve Dinçer, 2018). Lojistik faaliyetlerinin ekonomik ve sürdürülebilir bir şekilde gerçekleştirilebilmesi, yeşil tedarik zinciri süreci için hayati önem taşımaktadır. Sürdürülebilirlik, günlük ihtiyaçların doğal kaynaklara zarar vermeden temin edilerek kalkınmanın sağlanabilmesi anlamını taşımaktadır. Artan nüfusun taleplerini karşılamaya yönelik olarak yüksek miktarlarda üretim yapılması, doğal kaynakların gereğinden fazla harcanmasını ve üretim faaliyetleri sırasında doğaya verilen zararın artmasını beraberinde getirmiştir. Bu sebeple, özellikle son dönemde sürdürülebilirlik kavramının işletmelerin tüm faaliyetlerinde bir amaç olarak benimsenmesi gerekmektedir.

Sürdürülebilir lojistik, ürünlerin tedarikçilerden nihai müşteriye doğru hareketinin doğaya verilen zararı enazlayacak şekilde gerçekleştirilmesini hedefler. Bunun yanısıra sürdürülebilir lojistik, atıkların ve kullanılmış malzemelerin geri dönüşümünü sağlayarak gereksiz kaynak kullanımının önüne geçmeye yardımcı olan tersine lojistik faaliyetlerini de içermektedir. Ürün ve malzemelerin tekrar kullanımına olanak sağlayan tersine lojistik süreci, hammadde kaynaklarının azalması nedeniyle tedarik zinciri içerisinde anahtar süreçlerden biri haline gelmiştir.

Hem lojistik faaliyetleri hem de tersine lojistik faaliyetleri işletmeler için yüksek maliyet gerektiren ve çevreye zararlı etkileri bulunabilen faaliyetlerdir. Harcanan yüksek yakıt miktarları ve buna bağlı olarak gerçekleşen yüksek miktarlardaki zararlı gaz salınımları sebebiyle lojistik faaliyetleri içerisinde özellikle karayolu taşımacılığı hem önemli bir maliyet kalemi ve hem de çevresel bir tehdittir. (Dişkaya ve Dinçer, 2018). Avrupa Çevre Ajansı'nın verilerine göre karayolu taşımacılığı, Avrupa'da genel sera gazı emisyonlarının %17.5'inden sorumludur. Dünya genelinde sera gazı salınımına en fazla etkisi olan sektörlerin başında gelen karayolu taşımacılığı sektöründe faaliyet gösteren işletmeler, çevreye duyarlı politikalar izlemek durumdadır. Türkiye'nin de 2009 yılında dahil olduğu, Birleşmiş Milletler İklim Değişikliği Çerçeve Sözleşmesi içinde küresel ısınma ve iklim değişikliği konusunda uluslararası mücadeleyi sağlamaya yönelik imzalanan Kyoto Protokolü, dahil olan ülkelerin sera gazı emilim oranlarını belirli bir seviyeye getirmesini zorunlu kılmaktadır (Elbasan, 2015). Bu gibi yasal zorunluluklar işletmeler için faaliyetlerini çevreye duyarlı bir şekilde sürdürmeyi bir tercih olmaktan çıkarmış ve zorunlu hale getirmiştir.

2.2. Araç Rotalama Problemi (ARP)

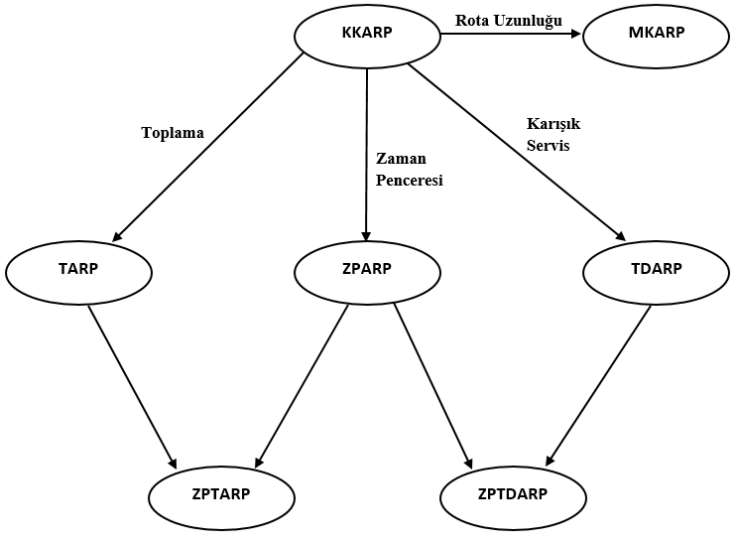
Araç rotalama problemi (ARP), ilk olarak Dantzig ve Ramser (1959) tarafından benzin istasyonlarına yağ taleplerinin dağıtımı için kamyon filosunun minimum seyahat mesafesine sahip olacak şekilde rotalanması problemi olarak çalışılmıştır. Problem bu çalışmada "Kamyon Sevkiyatı Problemi" olarak adlandırılmıştır. Clarke ve Wrigt (1964), bu problemi bir depoya sahip heterojen araçlardan oluşan bir araç filosunun farklı konumlardaki müşteri taleplerini karşılamak için nasıl hizmet verebileceğine ilişkin bir doğrusal optimizasyon problemi olarak tanımlamıştır.

ARP, farklı konumlarda bulunan bir grup müşterinin taleplerinin heterojen veya homojen araçlardan oluşan, bir veya birden fazla depodan yola çıkan bir araç filosu ile belirli kısıtlaraltında karşılanabilmesi için araçların toplam seyahat mesafesini (maliyetini / süresini) eniyileyecek şekilde rotalanması olarak tanımlanır.

Problem tipine göre farklılık göstermesine rağmen ARP'de dikkate alınan temel kısıtlar:

- Müşteriler sadece bir kez ziyaret edilmelidir.
- Tüm araçların rotası depoda başlamalı ve depoda sonlanmalıdır.
- Rotalar üzerindeki araç yükleri araç kapasitesini aşmayacak şekilde olmalıdır.
- Araçların kapasitesi müşteri taleplerindenbüyük olmalıdır.

Literatürde, farklı uygulama alanları için çevresel özellikleri ve yeni kısıtları dikkate alan farklı ARP türleri üzerinde çalışılmıştır. Toth ve Vigo (2002: 5-6), ARP'nin temel çeşitlerin kapasite kısıtlı ARP, mesafe kısıtlı ARP, zaman pencereli ARP, topla dağıt ARP, topla ARP, topla-dağıt ARP ve bunların beraber ele alındığı problemler olarak belirlemişlerdir. Şekil 2.1'de temel ARP tipleri ve bağlantıları gösterilmektedir.



Şekil 2.1. ARP'nin temel tipleri ve bunların bağlantıları (Toth ve Vigo, 2002)

ARP'nin literatürde ele alınan en temel tipi kapasite kısıtının dikkate alındığı kapasite kısıtlı ARP (KKARP)'dir. KKARP' de her araç belirli bir kapasiteye sahiptir ve araç rotaları araç kapasiteleri aşılmayacak şekilde belirlenmektedir.

Mesafe kısıtlı ARP (MKARP), araçların katedeceği seyahat mesafe için bir üst sınırın olduğu ARP tipidir. Mesafe kısıtı yerine araçların seyahat süresi için zaman kısıtı söz konusu olabilir. Araçlar için mesafe (zaman) kısıtı genelde taşınan ürünlerin bozulabilir ürünler olduğu problemlerde dikkate alınmaktadır.

Her müşteriye özgü bir zaman dilimi [ai, bi] içerisinde talebini karşılamak için uğranıldığı ARP, zaman pencereli ARP (ZPARP) olarak adlandırılmaktadır. Zaman pencereleri katı zaman pencereleri ve esnek zaman pencereleri olmak üzere iki şekilde ele alınabilmektedir. Katı zaman pencereli ARP'de, her müşteriye belirlenen zaman dilimi içerisinde servis yapılmalıdır ve araç müşteriye en erken servis zamanından (ai) önce ulaşırsa, servis yapmak için beklemek zorundadır. Esnek zaman pencerelerinde ise belli bir cezaya katlanılarak zaman penceresi dışında müşteri servisleri gerçekleştirilebilmektedir.

Topla ARP (TARP), tersine lojistik faaliyetinin gerçekleştirildiği, müşterilerden tesislere toplama faaliyetlerini ele alan problem tipidir. Müşterilerin sadece dağıtım değil aynı zamanda toplama da talep edebildikleri problem türü ise topla dağıt ARP (TDARP) olarak adlandırılmaktadır.

Bir dizi müşteriye farklı günlerde teslimat yapılması gereken durumlar periyodik ARP (PARP) olarak tanımlanmıştır. Tekrar eden operasyonların eniyilenmesi maliyet tasarrufuna önemli katkılar sağlayabilmektedir. Bu nedenle PARP, ARP çeşitleri arasında önemli bir yere sahiptir. Periyodik teslimatlar kurye hizmetleri, çöp/atık toplama problemleri gibi birçok uygulamada ortaya çıkmaktadır.

ARP'de genellikle müşteriler arası ulaşım sürelerinin sabit olduğu varsayılmaktadır. Fakat gerçek hayat problemlerine bakıldığında iki müşteri arası ulaşım süresi trafik sıkışıklıklarına bağlı olarak gün içerisinde değişiklik göstermektedir. Zaman bağımlı ARP (ZBARP), bu durumun dikkate alındığı günün farklı bölümlerinde müşteriler arası ulaşım sürelerinin değişiklik gösterdiği ARP'dir. Özellikle şehir içi taşımacılığın ele alındığı rotalama problemlerinde ulaşım sürelerinin zamana bağlı olarak değişimi daha büyük önem kazanmaktadır.

Çok depolu ARP'de (ÇDARP), müşterilere tek bir depodan değil birden çok depodan hizmet verilmesi durumunu söz konusudur. Problemde müşteri ve depo yerleri önceden bilinmektedir. ÇDARP'de genellikle her araç bir depoya ait olup o depodan çıkıp tekrar aynı depoya dönmek zorundadır.

Müşterilere hizmet verecek araç filosuna dahil olan araçların birbirlerinden farklı kapasitelere ve özelliklere sahip olduğu durumda heterojen ARP (HARP) söz konusudur.

Bir depodan yola çıkan araçların depoya dönme durumu söz konusu olmadığında problem açık uçlu ARP (AUARP) olarak tanımlanır. AUARP'nin uygulama alanları genellikle üçüncü parti lojistik olarak adlandırılan araç filosu sahibinin dağıtıcı dışında başka bir işletme olduğu problemlerdir.

ARP ve tipleri için literatürde Ekşioğlu, Vural & Reisman (2009), Breakers, Ramaekers & Nieuwenhuyse (2016) tarafından ayrıntılı literatür araştırması çalışmaları yapılmıştır.

2.3. Eşzamanlı Topla Dağıt Araç Rotalama Problemi

Günümüzde artan nüfus ve gelişen teknoloji ile birlikte çevreye verilen zararın giderek artması ve kaynakların hızla tüketilmesi, insanların geri dönüşüm ve çevre konusunda bilinçlenmesine ve çevreye karşı olan hassasiyetin artmasına sebep olmuştur. Lojistik faaliyetlerinin önemli kısmını da, müşterilerden kullanılmış ürünlerin veya atıkların geri dönüştürülmek üzere alınarak tesislere taşınması faaliyetleri oluşturmaktadır. Bu faaliyetler literatürde tersine lojistik olarak adlanıdırılır. ARP'nin türlerinden birisi olan TDARP, tesislerden müşterilere dağıtım ve müşterilerden tesislere toplama faaliyetlerinin aynı araçlarla yapıldığı ARP türüdür. TDARP'de lojistik ve tersine lojistik faaliyetlerinin aynı araçlarla yapılmasıyla hem maliyet tasarrufu sağlanmakta hem de araçların çevreye olan zararlı etkileri azaltılmaktadır.

TDARP'de müşteriler dağıtım ve toplama müşterileri olarak ayrı ayrı sınıflandırılabildiği gibi müşterilerin toplama ve dağıtımı aynı anda talep ettikleri durumlarda söz konusudur. Bu durumda TDARP, önce dağıt sonra topla ARP (ÖDSTARP), karma topla dağıt ARP (KTDARP) ve eşzamanlı topla dağıt ARP (ETDARP) olarak üçe ayrılmaktadır.

ÖDSTARP'de müşteriler toplama ve dağıtım noktaları olarak iki gruba ayrılmaktadır. Araçlar, tek bir tur üzerinde toplama veya dağıtım yapacakları talep noktalarına sahiptir. Bu durumda, araçların kapasite kısıtlarını sağlamada kolaylık sağlaması nedeniyle, önce dağıtım yapılacak müşterilere, daha sonra ise toplama müşterilerine uğramaktadır. KTDARP'de müşteriler yine dağıtım ve toplama müşterileri olarak ikiye ayrılmakta, ancak müşterilere uğramada herhangi bir öncelik yoktur. ETDARP ise, bir müşterinin aynı anda hem toplama talebi hem de dağıtım talebinin olabildiği durumları inceleyen TDARP türüdür. Bu nedenle, ETDARP için müşteriler arasında herhangi bir öncelik söz konusu değildir ve her noktada kapasite kısıtının sağlanması daha zor bir hale gelmektedir.

Müşterilere yeni ürün teslimi yapılırken kullanılan eski ürünlerin geri dönüştürülmek için toplanması, müşterilerin memnun olmadığı ürünlerin toplanarak yerlerine yeni ürünlerin teslim edilmesi veya ürün teslimatı yapılırken atıkların geri toplanması gibi durumlarda ETDARP büyük önem kazanmaktadır. Bu sebeple ETDARP lojistik faaliyetleri içerisinde çok önemli bir problemdir. Literatürde ETDARP ilgili yapılan çalışmalarda genellikle

aşağıdaki kısıtlar altında toplam seyahat mesafesi veya toplam seyahat maliyeti enazlanmaya çalışılır:

- Bir müşteri sadece bir araç ile bir kez ziyaret edilebilir, müşterinin dağıtım ve toplama talepleri aynı araç tarafından karşılanmalıdır.
- Araçtaki toplam dağıtım ve toplama talebi miktarı rotanın hiçbir noktasında araç kapasitesini aşmamalıdır.
- Belirli bir depodan yola çıkan araçlar aynı depoya geri dönmelidir.
- Müşterilerin tüm talepleri eksiksiz karşılanmalıdır.

İşletmelerin lojistik faaliyetleri için ayırdığı büyük bütçeler ve büyük dağıtım ağları boyunca çevreye salınan zararlı gaz miktarı dikkate alınınca, toplama ve dağıtım faaliyetlerinin eşzamanlı olarak daha az maliyetle ve çevreye daha az zarar vererek gerçekleştirilmesini sağlayan ETDARP, işletmeler ve çevre için oldukça önemli bir ARP türüdür (Belgin, 2018).

Son yıllarda literatürde TDARP üzerinde çok çalışma yapıldığı görülmektedir. Bu konuda yapılan çalışmalara Berbeglia, Cordeau, Gribkovskaia & Laporte (2007), Parragh, Doerner & Hartl (2008) ve Koç & Laporte (2018) çalışmalarından erişilebilir.

2.4. Yeşil Araç Rotalama Problemi

Dünyada çevre için hızla artan farkındalık lojistik endüstrisinde de çevresel kaygının artmasına sebep olmuştur. Motorlu araç sayısının oldukça artması ve bu araçların kirli gaz salınımları nedeniyle hava kirliliğindeki payının oldukça fazla olması lojistik işletmelerinin operasyonlarının çevresel etkilerini göz ardı etmelerini imkansız kılmıştır. Bunun yanında araçların yakıt tüketiminin artmasının gaz salınımlarıyla doğru orantılı olması nedeniyle, araçların yakıt tüketimlerinin düşürülmesi hem çevre hem de lojistik işletmelerinin ekonomisi için önemlidir. Birçok işletme için, zararlı gaz salınımının azaltımı halihazırda ana hedeflerden biridir ve bu eğilim artış göstermektedir. Çevreye daha duyarlı olmanın ardındaki motivasyon sadece yasal kısıtlamalarla ilgili değil, aynı zamanda çevreye duyarlı bir iş yapmanın para tasarrufu sağlaması ve sürdürülebilirliği desteklemek isteyen müşterilere hitap etmesi ile de ilgilidir (Çimen & Soysal, 2017). Bu durum son yıllarda, önemli bir lojistik karar verme problemi olan ARP'nin özel bir tipi olan yeşil ARP'yi (Y-ARP) gündeme getirmiştir.

Tedarik zinciri içerisinde çevreye zararlı gaz salınımlarının büyük bir kısmı lojistik faaliyetlerinden kaynaklanmaktadır. Bu sebeple, günümüzde işletmeler lojistik faaliyetlerini gerçekleştirirken, bu faaliyetlerin işletmeye olan maliyeti kadar çevreye olan etkilerini ölçme ve enerji tüketimini azaltma gibi konularla ilgilenmektedirler. Y-ARP çevresel etkilerin ve lojistik maliyetlerinin dengeli bir şekilde optimize edilmesine olanak sağlayan bir ARP türü olması sebebiyle büyük önem taşımaktadır.

Klasik ARP'de amaç fonksiyonu olarak ele alınan toplam seyahat maliyeti hesaplanırken araçların yakıt tüketim maliyetleri dikkate alınmamaktadır. Taşıma maliyetinin büyük bir kısmını yakıt tüketim maliyetleri oluşturmaktadır. Bu sebeple toplam seyahat maliyetinin sadece seyahat uzunluğuna bağlı olarak hesaplanması önemli bir maliyet kaynağının gözardı edilmesine sebep olmakta ve amaç fonksiyonunu gerçeklikten uzaklaştırmaktadır (Belbağ, 2017).

Literatürde yapılan çalışmalarda yeşil özellik problemlere amaç fonksiyonu veya kısıt olarak yansıtılabilmektedir. Yapılan çalışmaların çoğunda yakıt tüketimi veya CO2 salınımı miktarları amaç fonksiyonu olarak alınmakta ve problem yakıt tüketimi / CO2 salınımı miktarının enazlanmasına yönelik olarak çözülmektedir. Araçların yakıt tüketimi miktarlarına etki eden faktörler genel olarak, aracın tipi, taşıdığı yük miktarı, hızı, ivmesi, araç motorunun özellikleri, kullanılan yakıt tipi, yolun eğimi, seyahat mesafesi olarak sayılabilir. Aşağıda, yakıt tüketimi / CO2 salınımı enazlanması için farklı faktörleri dikkate alan amaç fonksiyonlarına örnekler verilmiştir.

Araç Yükü: Aracın taşıdığı yük ile harcanan yakıt miktarı ve dolayısı ile CO2 salınımı doğru orantılıdır. Zhang, Lee, Choy, Ho & Lp (2014), araç yüküne ve seyahat mesafesine bağlı olarak CO2 salınımı miktarını hesaplayan bir amaç fonksiyonu kullanmışlardır. Amaç fonksiyonu Eşitlik 2.1.'de verilmektedir.

$$Min \sum_{i}\sum_{j}\sum_{k} CER\left(p_0 + \frac{p^* - p_0}{Q}\right) d_{ij} X_{ijk} \qquad (2.1)$$

CER : CO_2 emilimi oranı (yakıt türüne göre değişen sabit bir değerdir)

p_0 : Aracın yüksüzken yakıt tüketim oranı

p^* : Araç tamamen doluyken yakıt tüketim oranı

d_{ij} : i, j düğümleri arasındaki mesafe

q_{ijk}: k aracının i, j düğümleri arasında taşıdığı yük miktarı

X_{ijk}: $\begin{cases} 1, & k \text{ aracı } i \text{ müşterisinden } j \text{ müşterisine gidiyorsa} \\ 0, & \text{dd} \end{cases}$

Araç Hızı ve Araç İvmesi: Bir aracın sabit bir hızla yol alması ile değişken bir hızla yol alması yakıt tüketim miktarını etkilemektedir (Belbağ, 2017). Aracın ivmesinin ve hızının yüksek olması durumunda yakıt tüketimide artış gösterecektir. Bektaş & Laporte (2011) çalışmalarında hızın sabit ve değişken kabul edildiği iki çeşit amaç fonksiyonu tanımlamış ve aracın hızının sabit olduğu durumda yakıt tüketimi ve zararlı gaz salınımının daha düşük olduğunu göstermişlerdir. Eşitlik 2.2. çalışmada hızın sabit olduğu durum için tanımlanan amaç fonksiyonunu, Eşitlik 2.3. ise hızın değişken olduğu durum için tanımlanan amaç fonksiyonunu göstermektedir.

$$Min \sum_{i,j \in A} \alpha_{ij} d_{ij} W X_{ij} + \sum_{i,j \in A} \alpha_{ij} f_{ij} d_{ij} \tag{2.2}$$

$$Min \sum_{i,j \in A} \alpha_{ij} d_{ij} W X_{ij} + \sum_{i,j \in A} \alpha_{ij} f_{ij} d_{ij} + \sum_{i,j \in A} d_{ij} \beta \left(\sum_{r \in R} (-v^r)^2 Z_{ij}^r \right) \tag{2.3}$$

d_{ij} : i, j düğümleri arasındaki mesafe

α_{ij} : Hat spesifik sabit

β : Araç spesifik sabit

W : Boş aracın ağırlığı

f_{ij} : i, j düğümleri arasında taşınan yük miktarı

$-v^r$: Ortalama hız $(R = 1,2, \dots, r, \dots)$

$-v^r = (l^r + u^r)/2$

$[l^r + u^r]$: r hız seviyesinin hız aralığı, $l^1 = l, u^R = u$

X_{ij}: $\begin{cases} 1, & \text{bir araç } i \text{ müşterisinden } j \text{ müşterisine gidiyorsa} \\ 0, & \text{dd} \end{cases}$

Z_{ij}^r: $\begin{cases} 1, & (i, j) \text{ hattında } r \text{ hızıyla seyahat ediliyorsa} \\ 0, & \text{dd} \end{cases}$

Yolun Eğimi: Araçların eğimli bir yolda yukarı veya aşağı yönlü hareketi harcanan yakıt miktarını farklı şekillerde etkilemektedir. Aracın eğimli bir yolda yukarı doğru yol alması durumunda yakıt tüketimi miktarı artarken tam tersi durumda yakıt tüketimi miktarı azalmaktadır. Elbasan (2015) çalışmasında CO2 salınımı miktarının hesaplanması için

müşteriler arası uzaklıkları yol eğimini dikkate alarak Eşitlik 2.4'te verildiği gibi revize ederek kullanmıştır.

$$c'_{ij}: \sum_{h=1} d^h_{ij}(sin\theta^h_{ij} + C_r cos\theta^h_{ij}) \tag{2.4}$$

c'_{ij} : i ve j müşterileri arası yolun h parçası için revize edilmiş uzaklık

d^h_{ij} : i ve j müşterileri arası yolun h parçasının uzunluğu

θ^h_{ij} : i ve j müşterileri arası yolun h parçası için eğim açısı

C_r : Yuvarlanma sürtünme katsayısı

Bu tez kapsamında yeşil ETDARP için yakıt tüketimi maliyetini aracın yük miktarına ve seyahat mesafesine bağlı olarak hesaplayan bir amaç fonksiyonu kullanılmıştır. Kullanılan amaç fonksiyonu Bölüm 4.1'de verilmiştir.

3. LİTERATÜR ARAŞTIRMASI

Yakıt tüketim maliyetine dayalı olarak tanımlanan Y-ETDARP ilk kez bu tez kapsamında ele alındığı için literatürde bu konuda bir çalışmaya rastlanmamıştır. Bunun yanısıra, Y-ETDARP'da amaç fonksiyonu toplam seyahat mesafesinin (maliyetinin) enazlanması olarak değiştirildiğinde problem ETDARP'a, Y_ETDARP'da sadece dağıtım/toplama talepleri dikkate alındığında problem Y-ARP'ye indirgenir. Bu nedenle bu bölümde Y-ETDARP'ın alt problemleri olan ETDARP ve Y-ARP için literatür araştırmasına yer verilmiştir.

3.1. ETDARP için Literatür Araştırması

ARP, literatüre ilk olarak sadece dağıtım ya da sadece toplama problemi olarak girmiştir. Daha sonra müşterilerin bazılarının sadece toplama ya da sadece dağıtım talep edebildiği ARP ele alınmaya başlanmıştır. Min (1989), ilk olarak bir noktada aynı anda hem toplama hemde dağıtım yapılabilmesi durumunu ele alarak eşzamanlı topla dağıt ARP'yi literatüre kazandırmıştır. Min (1989) çalışmasında, örnek olay olarak bir halk kütüphanesinin dağıtım sistemini ele alarak problemin gerçek hayata uygulanabilirliğini göstermiştir.

Salhi ve Nagy (1999), TDARP tipleri için ekleme sezgiseli olarak adlandırdıkları sezgisel bir algoritma geliştirmişlerdir. Algoritmayı tek depolu ve çok depolu TDARP için kullanmışlardır. Bu çalışmayla beraber ETDARP literatürde ilk kez çok depolu olarak ele alınmıştır.

Dethloff (2001) ve Nagy ve Salhi (2005), büyük boyuttaki ETDARP'nin çözümü için sezgiseller geliştirmişlerdir. Dethloff (2001), artık kapasiteye ve radyal ek mesafe ücretlerini kriter olarak alan bir ekleme sezgiseli geliştirmiş ve bu sezgiseli aynı zamanda kurucu sezgisel olarak önermiştir.

ETDARP, NP-zor problem olduğundan dolayı, literatürde problemin çözümü için çeşitli sezgisel algoritmalar geliştirilmiştir. 2005 yılından itibaren ise problemin çözümünde metasezgisel algoritmaların kullanıldığı görülmektedir. Tabu arama algoritmasının ETDARP için oldukça iyi sonuçlar verdiği görülmüştür. Bu nedenle tabu arama algoritması çok kez problemin çözümünde kullanılmıştır. Crispim ve Brandao (2005), tabu arama ve

değişken komşu iniş metasezgisellerinden oluşan karma bir algoritma geliştirmiştir, Montane ve Galvao (2006), problemi dört farklı komşuluk yapısı kullanan tabu arama algoritması ile çözmüştür, Chen ve Wu (2006), iyi bir başlangıç çözüm elde etmek için ekleme tabanlı bir çözüm kurucu algoritma ve problemin çözümü için tabu listesine ve rota iyileştirme prosedürlerine dayalı bir sezgisel algoritma geliştirmiştir. Wassan, Hassan ve Nagy (2008), önerilen hareketlerin uygunluk kontrolünü hızlı bir şekilde yapabilen ve tekrar edilen hareketlere tepki verebilen bir metasezgisel geliştirmiştir.

Subramanian, Ochi ve Cabral (2008), Subramanian ve Cabral (2008), Subramanian, Drummond, Bentes, Ochi ve Farias (2010), Subramanian, Uchoa ve Ochi (2013a) çalışmalarında önceki çalışmalardan farklı olarak problem için iteratif yerel arama algoritmasını kullanmışlardır. Bunun dışında Ai ve Kachitvichyanukul (2009), Wei, Zhang ve Tang (2010) ve Gan, Wang, Le ve Niu (2012) parçacık sürüsü optimazasyonu ile Gajpal ve Abad (2009) ve Çatay (2010) karınca kolonisi algoritması ile, Tasan ve Gen (2012) genetik algoritma ile problem için çözüm algoritmaları geliştirmişlerdir.

Son yıllarda, ETDARP'ın çözümü için daha iyi performans gösteren karma metasezgisel algoritmaların geliştirildiği görülmektedir. Zachariadis, Tarantilis ve Kiranoudis (2009) tabu arama ve rehberli yerel arama algoritmaları ile, Goksal, Karaoglan ve Altiparmak (2013) parçacık sürüsü optimizasyonu ve değişken komşu iniş algoritmaları ve Avci ve Topaloglu (2015) tavlama benzetimi ve değişken komşu iniş algoritmalarına dayalı karma algoritmalar geliştirmişlerdir.

ETDARP için çözüm elde etmek amacıyla az sayıda da olsa kesin çözüm algoritmalarının geliştirildiği görülmektedir. Bu algoritmalar Dell'Amico, Righini ve Salani (2006), Rieck ve Zimmermann (2009), Subramanian, Uchoa, Pessoa ve Ochi (2011), Subramanian, Uchoa, Pessoa ve Ochi (2013b) tarafından geliştirilmiştir.

3.2. Y-ARP için Literatür Araştırması

Literatürde özellikle son yıllarda çevresel etkilerin göz önüne alındığı ARP çok fazla ilgi görmeye başlamış ve ARP'nin özel bir tipi haline gelmiştir. Araçların çevresel zararlarının enaza indirilmeye çalışıldığı ARP genel olarak yeşil ARP olarak adlandırılmak üzere, farklı varyasyonları için farklı tanımlamalar da gündeme gelmiştir. Y-ARP kavramı ilk kez

Erdoğan & Miller-Hooks (2012) tarafından alternatif yakıtlı araçların yakıt ikmallerinin de dikkate alınarak rotalanması problemi için kullanılmıştır. Daha sonraki yıllarda Y-ARP terimi genel olarak CO2 salınımı veya yakıt tüketimi gibi çevresel zarara sebep olabilecek etmenlerin dikkate alındığı ARP için kullanılmaya başlanmıştır (Tiwari & Chang, 2015). Bektaş & Laporte (2011) ise yakıt tüketimine ilişkin araç hızını dikkate alan enerji tabanlı bir model kurmuş ve bu problemi kirlilik rotalama problemi olarak adlandırmışlardır.

Y-ARP'nin terim olarak ilk kez Erdoğan & Miller-Hooks (2012) tarafından kullanılmasına karşın bu çalışmadan önce Figliozzi (2010), Kuo (2010), Bektaş & Laporte (2011), Ubeda, Arcelus & Faulin (2011) ve Suzuki (2011) Y-ARP literatürüne yön veren çalışmalar yapmışlardır.

Literatürde yakıt tüketiminin ve yakıt tüketimi ile doğrusal ilişki içerisinde olan zararlı gaz salınımının hesaplanabilmesi için birçok farklı model tanımlanmıştır. Çalışmaların büyük bir bölümünde yakıt tüketimi / CO2 salınımı miktarları, katedilen mesafeye ve aracın taşıdığı yüke bağlı olarak hesaplanmıştır. Ubeda, Arcelus & Faulin (2011), çalışmalarında bir yiyecek dağıtım şirketinin dağıtım ve toplama faaliyetlerinde kullanılacak araçları beş sınıfa ayırmış ve her bir sınıf için bir emisyon faktörü belirleyerek taşınan yük miktarına dayalı CO2 salınımlarını dikkate alan bir model oluşturmuşlardır. Lin, Choy, Ho & Ng (2014) ETDARP'yi ele almış ve genetik algoritmaya dayalı bir sezgisel algoritma ile yük miktarına dayalı CO2 salınımını en azlamaya çalışmıştır. Benzer şekilde Xiao, Zhao, Kaku & Xu (2012), Zhang, Lee, Choy, Ho & Lp (2014), Suzuki (2016) ve Tiwari & Chang (2015) ARP için; Kopfer, Schönberger & Kopfer (2014) HARP için; Küçükoğlu, Ene, Aksoy & Öztürk (2015) ZPARP için; Cinar, Gakis & Pardalos (2015) çok rotalı ARP için; Lang, Yang & Zhang (2015) zaman pencereli-zaman bağımlı ARP için; Xiao & Konak (2016) heterojen-zaman bağımlı araç rotalama ve çizelgeleme problemi için yük miktarına dayalı CO2 salınımı ve yakıt tüketimi miktarlarını dikkate almışlardır.

Yük miktarı, araçların yakıt tüketimi / CO2 salınımı miktarlarının belirlenmesinde dikkate alınan tek faktörn değildir. Bunun yanında çalışmalarda aracın hızı, yük miktarı ile beraber çoğunlukla dikkate alınan diğer bir faktördür. Bektaş & Laporte (2011) çalışmalarında, hız seviyeleri belirlemiş, her bir hız seviyesi için ortalama hızlar hesaplamış ve bu hızları ve araçların taşıdıkları yük miktarlarını dikkate alarak sera gazı salınımı maliyeti, yakıt tüketimi maliyeti ve sürücü ücretlerini içeren toplam seyahat maliyetini enazlayacak bir matematiksel

model oluşturmuşlardır. Figliozzi (2010), Kuo (2010), Koç, Bektaş, Jabali & Laporte (2014) ve Niu, Yang, Chen & Xiao (2018), yakıt tüketimi/CO2 salınımı miktarlarını hesaplarken hızı, seyehat mesafesini ve araç yükünü dikkate alan diğer önemli çalışmalardır.

Araçların yakıt tüketimine / CO2 salınımına etki eden diğer faktörler ise yolun eğimi ve aracın ivmesidir. Demir, Bektaş & Laporte (2012), yine hız seviyeleri belirleyerek ve her bir hız seviyesi için ortalama hızlar hesaplayarak araç hızlarını dikkate almışlardır. Araçların taşıdıkları yükler ve hızlarına ek olarak bu çalışmada motor yer değiştirmesi, motor hızı gibi motor özellikleri ve yolun eğimi yakıt tüketim miktarının hesaplanmasında göz önüne alınmıştır. Franceschetti, Hanhan, Woansel, Bektaş & Laporte (2013), araç yükünü, seyahat uzunluğunu, zamana bağlı araç hızlarını ve yolun eğimini dikkate almışlardır. Demir, Bektaş & Laporte (2014a), çalışmalarında araç yükü, seyahat mesafesi ve araç hızına ilaveten araçların ivmelerini ve yol eğimini de dikkate almıştır. Suzuki(2011), Alinaghian & Naderipour (2016), Naderipour & Alinaghian (2016), ve Franceschetti, Demir, Honhon, Woensel, Laporte & Stobbe (2017), yol eğimini dikkate alan diğer çalışmalardır.

Çizelge 3.1'de Y_ARP literatüründeki önemli çalışmalar için ele alınan ARP tipi, yeşil amaç fonksiyonlarında dikkate alınan faktörler ve kullanılan çözüm yöntemleri özetlenmiştir.

Çizelge 3.1'de görüldüğü gibi Y-ARP'ye ilişkin yapılan çalışmalarda ETDARP'yi ele alan iki çalışma vardır. Lin, Choy, Ho & Ng (2014), tüm taleplerin karşılanmak zorunda olmadığı ETDARP'yi ele almışlar ve yük miktarına dayalı CO2 salınımını enazlamışlardır. Literatürde tüm taleplerin karşılanmak durumunda olunduğu ETDARP için tek çalışma Elbasan (2015) tarafından yapılmıştır. Bu çalışmada, CO2 salınımının hesaplanmasında araçların yük miktarları ve yol eğimleri dikkate alınmış ve çözüm yöntemi olarak tavlama benzetimi algoritması kullanılmıştır. Y_ARP ile ilgili yapılan çalışmalarda büyük boyutlu problemler için geliştirilen çözüm yöntemlerine bakıldığında son dönemlerde iyi çözümler verdiği bilinen hipersezgisel yaklaşımların kullanılmadığı görülmektedir.

Bu çalışmada literatürde oldukça az çalışılmış olan yeşil ETDARP için hipersezgisel bir algoritma geliştirilmiş ve yüke dayalı yakıt tüketimi maliyetinin enazlanması amaçlanmıştır.

Çizelge 3.1. Y-ARP ile ilgili çalışmalar

Yazar (Yıl)	Problem Tipi	Çözüm Yöntemi	Seyahat Mesafesi	Aracın Yükü	Aracın Hızı	Yolun Eğimi
Figliozzi (2010)	ZPARP	Yinelemeli rota oluşturma ve geliştirme algoritması	X		X	
Kuo (2010)	ZBARP	Tavlama benzetimi algoritması	X	X	X	
Suzuki (2011)	ZPARP	Sıkıştırılmış tavlama algoritması	X	X	X	
Bektaş ve Laporte (2011)	ZPARP	Matematiksel model	X	X	X	
Ubeda ve diğerleri (2011)	KTDARP	Matematiksel model	X	X		
Demir ve diğerleri (2012)	ZPARP	Geniş komşu arama sezgiseli	X	X	X	X
Erdoğan ve Miller-Hooks (2012)	MKARP	Clarke ve Wright tasarruf sezgiseli, yoğunluğa dayalı kümeleme sezgiseli	X	X		
Jabali (2012)	ZBARP	Tabu Arama Algoritması	X	X	X	
Xiao vd. (2012)	KKARP	Tavlama benzetimi algoritması	X	X		
Franceschetti ve diğerleri (2013)	ZBARP	Matematiksel model	X	X	X	X
Oberscheider ve diğerleri (2013)	Çok depolu zaman pencereli TDARP	Matematiksel model	X			
Demir ve diğerleri (2014)	ZPARP	Adaptif geniş komşu arama sezgiseli	X	X	X	X
Koç ve diğerleri (2014)	Zaman pencereli HARP	Evrimsel algoritma ve adaptif geniş komşu arama sezgiselinden oluşan hibrit bir metasezgisel	X	X	X	
Kopfer ve diğerleri (2014)	HARP	Matematiksel model	X	X		
Lin ve diğerleri (2014)	ETDARP	Genetik algoritma	X	X		
Tajik ve diğerleri (2014)	Zaman pencereli KTDARP	Gürbüz matematiksel model	X	X	X	
Zhang ve diğerleri (2014)	KKARP	Hibrit yapay arı kolonisi algoritması	X	X		
Cinar ve diğerleri (2015)	Çok rotalı ARP	Tavlama benzetimi algoritması	X	X		
Küçükoğlu ve diğerleri (2015)	ZPARP	Tavlama benzetimi algoritması	X	X	X	X
Lang ve diğerleri (2015)	Zaman pencereli ZBARP	Karınca kolonisi algoritması	X	X		
Sosyal ve diğerleri (2015)	İki kademeli ZBARP	Matematiksel model	X	X	X	
Tiwari ve Chang (2015)	KKARP	Blok rekombinasyon algoritması	X	X		
Elbasan (2015)	ETDARP	Tavlama benzetimi algoritması	X	X		X
Alinaghian ve Naderipour (2016)	ZBARP	Gauss ateşböceği algoritması	X	X	X	X
Koç & Karaoglan (2016)	KKARP	Tavlama benzetimine dayalı kesin çözüm yaklaşımı	X	X	X	
Naderipour ve Alinaghian (2016)	Zaman bağımlı - AUARP	Parçacık sürüsü optimizasyonu	X	X	X	X
Suzuki (2016)	KKARP	Tavlama benzetimi algoritması ve tabu arama algoritmasından oluşan hibrit bir metasezgisel	X	X		
Xiao ve Konak (2016)	Heterojen zaman bağımlı araç rotalama ve çizelgeleme problemi	Karma tamsayılı programlama ve iteratif komşu arama algoritmasından oluşan hibrit bir metasezgisel	X	X		
Franceschetti ve diğerleri (2017)	ZBARP	Adaptif geniş komşu arama sezgiseli	X	X	X	
Jabir ve diğerleri (2017)	ÇDARP	Karınca kolonisi algoritması ve değişken komşu arama algoritmasından oluşan hibrit bir algoritma	X	X		

Çizelge 3.1. (devam) Y-ARP ile ilgili çalışmalar

Yazar (Yıl)	Problem Tipi	Çözüm Yöntemi	Seyahat Mesafesi	Aracın Yükü	Aracın Hızı	Yolun Eğimi
Soysal ve Çimen (2017a)	ZBARP	Simülasyon tabanlı sınırlandırılmış dinamik programlama yaklaşımı	X		X	
Soysal ve Çimen (2017b)	ZBARP	Yaklaşık dinamik programlama algoritması	X		X	
Niu ve diğerleri (2018)	Zaman pencereli AUARP	Hibrit tabu arama algoritması	X	X	X	
Poonthalir ve Nadarajan (2018)	KKARP	Parçacık sürüsü optimizasyonu	X		X	

Bu bölümde Y-ARP için yapılmış olan literatür araştırmasında, literatüre yön verdiği düşünülen önemli çalışmalar dikkate alınmıştır. Y-ARP literatürü ile ilgili daha ayrıntılı bilgi Demir, Bektaş & Laporte (2014b), Lin, Choy, Ho, Chung&Lam (2014) ve Belbağ (2017) çalışmalarından elde edilebilir.

4. Y-ETDARP İÇİN MATEMATİKSEL MODEL VE SEZGİSEL YAKLAŞIM

Bu bölümde Y-ETDARP için matematiksel model verildikten sonra, geliştirilen İYA_HS algoritması için yararlanılan sezgisel algoritmaların temel özelliklerinden bahsedilecektir.

4.1. Y-ETDARP için Matematiksel Model

Tez kapsamında ele alınan Y-ETDARP, yüke dayalı yakıt maliyetini enazlayacak şekilde müşterilerin topla ve dağıt faaliyetlerini eşzamanlı karşılayan araç rotalarının oluşturulması olarak tanımlanabilir. Y-ETDARP için önerilen matematiksel modelde yüke dayalı yakıt maliyeti Xiao, Zhao, Kaku & Xu'nun (2012) yaklaşımı dikkate alınarak tanımlanmıştır. Bu nedenle öncelikle bu yaklaşım kısaca anlatılacaktır.

Xiao ve diğerleri (2012), KKARP'de birim hacimdeki yakıt tüketimi oranının katedilen mesafe ve araç ağırlığıyla doğrudan ilişkili olduğunu göstermişler ve kurmuş oldukları modelde yakıt tüketim oranını araç ağırlığıyla ilişkilendirmişlerdir. Q_0, aracın yüksüz ağırlığı Q_1 aracın aracın taşıdığı yük miktarı olmak üzere yakıt tüketim oranını, Q_1'e bağlı doğrusal bir fonksiyon olarak Eşitlik 4.1'deki gibi tanımlamışlardır.

$$p(Q_1) = \alpha(Q_0 + Q_1) + b \tag{4.1}$$

Aracın taşıyabileceği enbüyük ağırlık Q ile ifade edilirse, aracın tam yüklüyken yakıt tüketim oranı p^* ve yüksüzken yakıt tüketim oranı p_0, Eşitlik 4.2 ve Eşitlik 4.3 ile elde edilmektedir.

$$P^* = \alpha(Q_0 + Q) + b \tag{4.2}$$

$$P_0 = \alpha Q_0 + b \tag{4.3}$$

α parametresi Eşitlik 4.2 ve Eşitlik 4.3 kullanılarak aşağıdaki gibi elde edilmektedir.

$$\alpha = (p^* - p_0)/Q \tag{4.4}$$

Böylece Q_1 miktarında yük taşıyan bir aracın yakıt tüketim oranı (YTO) Eşitlik 4.5 ile tanımlanabilir.

$$p(Q_1) = p_0 + \left(\frac{p^* - p_0}{Q}\right) Q_1 \tag{4.5}$$

Eşitlik 4.5,

1. YTO ile araç yükü arasında kesim noktası p_0 ve eğimi $(p^*-p_0)/Q$ olan doğrusal bir ilişki kurulabildiğini
2. Aracın yükü yakıt maliyetini en aza indirmede kat edilen mesafe ile eşit derecede öneme sahip olduğunu

göstermektedir (Xiao ve diğerleri, 2012).

ARP genellikle $G=(N,A)$ olarak tanımlanan bir graf üzerinde gösterilmekte ve bu grafa dayalı olarak modellenmektedir. $N=\{0, 1, 2,...., n\}$, grafın düğümleri olmak üzere deponun (0 düğümü) ve müşterilerin bulunduğu lokasyonları; A ise düğümler arası hatları yani müşteriler arası yolları ifade etmektedir. ETDARP için de bu graf gösterimi temel alınarak literatürde birçok matematiksel model geliştirilmiştir. Karaoğlan (2009) çalışmasında yardımcı değişkenleri graftaki düğümleri dikkate alarak tanımlamış ve ETDARP için bir matematiksel model önermiştir. Y-ATDARP için önerilen matematiksel modelde Karaoğlan'ın (2009) modeli temel alınmıştır.

Dizin Kümeleri ve Parametreler:

N_c : müşteriler kümesi
N : tüm düğümlerin kümesi ($N = '0' \cup N_c$)
c_{ij} : i ve j müşterileri arasındaki mesafe ($i, j \in N$)
c_0 : birim yakıt maliyeti ($i, j \in N$)
d_i : i müşterisinin dağıtım talebi ($i \in N_c$)
p_i : i müşterisinin toplama talebi ($i \in N_c$)
Q : araç kapasitesi

Karar Değişkeni:

$$x_{ij} = \begin{cases} 1, & (i,j) \text{ arkı herhangi bir tur üzerinde ise } (\forall i,j \in N) \\ 0, & \text{dd} \end{cases}$$

Yardımcı Karar Değişkenleri:

M : rota sayısı

U_{ij} : (i,j) hattı boyunca taşınan dağıtım talepleri miktarı $(\forall \in N)$

V_{ij} : (i,j) hattı boyunca taşınan toplama talepleri miktarı $(\forall \in N)$

Amaç Fonksiyonu:

$$\text{En küçük } Z = \sum_{i=0}^{n}\sum_{j=0}^{n} c_0 c_{ij}\left(p_0 x_{ij} + \alpha(U_{ij} + V_{ij})\right) \tag{4.6}$$

Kısıtlar:

$$\sum_{i \in N} x_{ij} = 1 \qquad \forall j \in N_c \tag{4.7}$$

$$\sum_{j \in N} x_{ij} = 1 \qquad \forall i \in N_c \tag{4.8}$$

$$\sum_{i \in N} x_{i0} \leq M \tag{4.9}$$

$$\sum_{J \in N} x_{0J} \leq M \tag{4.10}$$

$$U_{ij} + V_{ij} \leq Q x_{ij} \qquad \forall i \in N, \forall j \in N \tag{4.11}$$

$$\sum\nolimits_{j \in N_c} U_{0j} = Q_D \tag{4.12}$$

$$\sum\nolimits_{i \in N_c} U_{i0} = 0 \tag{4.13}$$

$$\sum\nolimits_{i \in N_c} V_{i0} = Q_P \tag{4.14}$$

$$\sum\nolimits_{J \in N_c} V_{0J} = 0 \tag{4.15}$$

$$U_{ij} \leq (Q - d_i)x_{ij} \qquad \forall i \in N_c, \forall j \in N \tag{4.16}$$

$$V_{ij} \leq (Q - p_j)x_{ij} \qquad \forall i \in N, \forall j \in N_c \tag{4.17}$$

$$U_{ij} \geq d_j x_{ij} \qquad \forall i \in N, \forall j \in N_c \tag{4.18}$$

$$V_{ij} \geq p_i x_{ij} \qquad \forall i \in N_c, \forall j \in N \tag{4.19}$$

$$\sum_{i \in N} V_{ji} - \sum_{i \in N} V_{ij} = p_j \qquad \forall j \in N_c \tag{4.20}$$

$$\sum_{i \in N} U_{ij} - \sum_{i \in N} U_{ji} = d_j \qquad \forall j \in N_c \tag{4.21}$$

$$x_{ij} \in \{0,1\} \qquad \forall i, j \in N \tag{4.22}$$

$$U_{ij}, V_{ij} \geq 0 \qquad \forall i, j \in N \tag{4.23}$$

$$M \geq 0 \tag{4.24}$$

Eşitlik 4.6 katedilen mesafeye ve taşınan yük miktarına bağlı olarak yakıt tüketimi maliyetinin enazlanması ile ilgili amaç fonksiyonunu ifade etmektedir. Eşitlik 4.7. ve Eşitlik 4.8. her müşteriye sadece bir kez hizmet verilmesini sağlamaktadır. Eşitlik 4.9. ve Eşitlik 4.10. depodan en fazla M adet araç çıkmasını ve depoya yine en fazla M adet aracın dönmesini garantilemektedir. Eşitlik 4.11. rotanın herhangi bir yerinde araçlardaki toplam yükün araç kapasitesini geçmesini engellemektedir. $Q_D = \sum_{i \in N_c} d_i$ ve $Q_P = \sum_{i \in N_c} p_i$ olmak üzere Eşitlik 4.12 depodan çıkan dağıtım taleplerinin toplamını, toplam dağıtım talebi miktarına eşitlemektedir. Eşitlik 4.13. ise rota sonunda depoya dönen araçtaki dağıtım talepleri toplamını "0"a eşitlemektedir. Bunun anlamı rota boyunca araçtaki dağıtım taleplerinin tamamının dağıtılmış olmasıdır. Benzer şekilde Eşitlik 4.14. ve Eşitlik 4.15' te depodan çıkan araçtaki toplama taleplerinin toplamını henüz toplama yapılmadığı için "0"a ve depoya dönen araçtaki toplama talepleri toplamını, toplam toplama talebi miktarına eşitlemektedir. Eşitlik 4.16. – Eşitlik 4.19. yardımcı karar değişkenlerinin alt ve üst sınırlarını belirlemektedir. Eş. 4.20. ve Eş. 4.21. numaralı kısıtlar ile rota üzerindeki toplama ve dağıtım taleplerinin dengesini sağlanmakta ve hatlar üzerinde taşınan yük miktarları belirlenmektedir. Eşitlik 4.22. , Eşitlik 4.23. ve Eşitlik 4.24. ise işaret kısıtlarıdır.

4.2. Sezgisel Yaklaşım

Bu tez kapsamında Y-ETDARP'ın çözümü için iteratif yerel arama (İYA) ve değişken komşu aramaya (DKA) dayalı bir hipersezgisel algoritma geliştirilmiştir. Bu nedenle bu kısımda kullanılan sezgisel algoritmaların temel özellikleri hakkında bilgiler verilecektir.

4.2.1. Hipersezgisel

İlk kez Cowling, P.,Kendall, G. and Soubeiga, E. (2000) tarafından literatüre kazandırılmış olan hipersezgiseller, karar noktalarında önceden tanımlanan alt düzey sezgisel kümesinden sezgisel seçmek ya da üretmek için geliştirilen yöntemlerdir. Amaç, arama sırasında alt düzey sezgisel kümesini etkili bir şekilde kontrol ederek algoritmanın etkinliğini arttırarak çözüm zamanını düşürmektir. Aramayı çözüm uzayı yerine sezgisel uzayında yapmakta olan hipersezgiseller, son yıllarda geniş bir yelpazedeki NP-zor problemler için makul çözüm zamanlarında kaliteli çözümler üreten güçlü yaklaşımlar haline gelmişlerdir.

Hipersezgisel algoritmaların performansı, doğrudan seçilen alt düzey sezgisel havuzunun performansıyla ilişkilidir. Alt düzey sezgiseller genellikle yerel arama komşuluk yapılarını (hareket operatörleri) ifade etmektedir. Literatürdeki çalışmalar, geliştirilen sezgisel ve metasezgisellerde faklı komşuluk yapılarının kullanılması ile arama sürecinin iyileştiğini göstermiştir. Bununla birlikte, gereksiz büyüklükte bir alt düzey sezgisel havuzu algoritmanın çözüm zamanını uzatmakta ve sezgisellerin performans geçmişini kaydetmek için daha fazla hafızaya ihtiyaç duyulmasına neden olmaktadır. (Soria-Alcaraz, Ochoa, Sotelo-Figeroa & Burke, 2017). Bu sebeple alt düzey sezgisel havuzunun etkin bir şekilde oluşturulması ve alt düzey sezgisellerin karar noktalarında etkin bir şekilde kullanılması algoritmaların performansı açısından büyük önem taşımaktadır.

Hipersezgiseller, var olan sezgisel kümesinden sezgisel seçerek algoritmaya yön veren seçici hipersezgiseller ve yeniden bir sezgisel oluşturan kurucu hipersezgiseller olarak iki grupta ele alınmaktadır. Literatürde NP-zor problemler için geliştirilen hipersezgisel yaklaşımlar genellikle seçici hipersezgisel grubuna girmektedir. Son yıllarda ARP için de kısa zamanda daha iyi çözümler elde etmek amacıyla hipersezgisel yaklaşımlar kullanılmaya başlanmıştır.

Pisinger & Ropke (2007) zaman pencereli ARP, kapasite kısıtlı ARP, çok depolu ARP ve açık uçlu ARP için adaptif geniş komşu arama algoritması geliştirmişlerdir. Bu algoritmada aramayı yoğunlaştırmak ve çeşitlendirmek için bir dizi yerleştirme ve çıkarma sezgiselini arama boyunca atanan puanlara göre adaptif olarak seçilmiş ve algoritma daha güçlü bir hale getirilmiştir. Bu algoritma, ARP için hipersezgisele dayalı geliştirilen ilk algoritma olmuştur. Benzer şekilde Garrido & Costro (2009), Walker, Ochoa, Gendreau & Burke (2012) ARP için seçici hipersezgisele dayalı algoritmalar geliştirmişlerdir. Garrido & Riff (2010), kurucu ve karıştırıcı sezgisellerden oluşan 3 farklı alt düzey sezgiselin iyi bir kombinasyonunu oluşturmak için hipersezgisel algoritma kullanmışlardır. Garrido & Castro (2012), alt düzey sezgisellerin arasında etkili bir işbirliği oluşturmak için adaptif olarak sezgiseller için ödül ve ceza değerleri belirlemiştir ve operatör seçimini bu şekilde yönlendirmiştir.

ARP için geliştirilen hipersezgisele dayalı algoritmalar çoğunlukla seçici hipersezgisel grubuna girse de literatürde az da olsa kurucu hipersezgisele dayalı algoritmalar da geliştirilmiştir (Sim & Hart, 2016).

Çizelge 4.1. ARP için hipersezgisel kullanan çalışmalar

Yazar (yıl)	Problem Tipi	Çözüm Yöntemi
Pisinger and Ropke (2007)	Zaman pencereli ARP Kapasite kısıtlı ARP Çok depolu ARP Açık uçlu ARP	Adaptif geniş komşu arama algoritması
Garrido & Castro (2009)	Kapasite kısıtlı ARP	Tepe tırmanma algoritması
Garrido & Riff (2010)	Dinamik ARP	Evrimsel algoritma
Meigan, Koukam & Creput (2010)	Kapasite kısıtlı ARP Mesafe kısıtlı ARP	Koalisyon tabanlı metasezgisel
Garrido & Castro (2012)	Dinamik ARP	Evrimsel algoritma
Walker, Ochoa, Gendreau & Burke (2012)	Zaman pencereli ARP	Adaptif iteratif yerel arama
Mlejnek & Kubalik (2013)	Kapasite kısıtlı ARP	Evrimsel algoritma
Sabar, Zhang & Song (2015)	Zaman pencereli ARP	Mat - hipersezgisel
Sim &Hart (2016)	Zaman pencereli ARP Heterojen - çok kompartımalı ARP	Genetik algoritma

Çizelge 4.1'de, literatürde ARP için hipersezgisellere dayalı geliştirilen çalışmalara yer verilmiştir. hipersezgiseller aynı zamanda birer metasezgisel olsa da alt düzey sezgiselleri (hareket operatörlerini) çok daha akıllı kullanmaları sebebiyle algoritmanın daha kısa zamanda daha iyi çözümlere ulaşmasını sağlamaktadır. Bu yönüyle bu hipersezgiseller NP-zor problemler için oldukça güçlü çözüm yaklaşımları olmuşlardır.

4.2.2. İteratif yerel arama algoritması

İteratif yerel arama algoritması, ilk kez Martin, Otto &Felten (1991) tarafından büyük adım Markow zincirleri ismiyle gezgin satıcı problemini çözmek için geliştirilmiş ve literatüre kazandırılmıştır. Daha sonra, Lourenço, Martin & Stützle (2002), iteratif yerel arama (İYA) algoritması ismini kullanarak algoritmayı geliştirmişlerdir.

Yerel eniyi çözümde karışıklık yaparak bir sonraki iterasyon için başlangıç çözüm elde eden İYA, bu şekilde algoritmanın yerel eniyi çözüme takılı kalmasını engellemeyi ve çözüm uzayının farklı bölgelerinde arama yapabilmeyi amaçlamaktadır. İYA, her adımda mevcut çözüme karıştırma adımını uygular, karıştırma sonrası çözümü yerel arama ile iyileştirir ve yeni çözümü belli bir kabul kriterine göre mevcut çözüm olarak kabul ederek veya etmeyerek bir sonraki iterasyona geçiş yapar. İYA, karıştırma adımı ile arama sürecinde çeşitlendirmeyi sağlarken yerel arama uygulayarak belirli bir alanda yoğunlaştırmayı sağlamaktadır. İYA'nın genel işleyişi Şekil 4.1.' de verildiği gibidir.

---

begin
$s_0 \leftarrow$ Başlangıç çözümü üret
$\hat{s} \leftarrow s_0$ çözümüne yerel arama uygula
    repeat
        $s' \leftarrow$ Karıştır($\hat{s}$, arama geçmişi)
        $\hat{s}' \leftarrow$ Yerel arama (s')
        $\hat{s} \leftarrow$ Kabul kriterini uygula ($\hat{s}, \hat{s}'$, arama hafızası)
    until Durdurma koşulu sağlanana kadar
end

Şekil 4.1. İteratif yerel arama algoritmasının adımları

İYA'nın işleyişini ve performansını etkileyen üç temel eleman kullanılan yerel arama algoritması, karıştırma yöntemi ve yeni çözümlerin kabul edilip edilmeyeceğini belirleyen kabul kriteridir.

Yerel arama algoritması:

Yerel arama algoritmaları (YAA) , komşuluk yapıları ile yapılan yerel değişmelerle mevcut çözümü iyileştirmeye çalışır. YAA'nın performansında uygun komşuluk yapısının seçimi oldukça önemli rol oynamaktadır. YAA kullandığı komşuluk yapıları ile mümkün çözümler arasından yerel eniyi çözümü elde eder, fakat bu çözüm global eniyi çözümü vermeyebilir.

Karıştırma yöntemi:

Metasezgisel algoritmalar, tekrarlayan iterasyonlarda sürekli aynı yerel eniyi çözümü elde etmekten kaçınmak için çeşitlendirme stratejileri kullanırlar. İYA, yerel eniyi çözümlere takılmaktan kurtulmayı karıştırma adımıyla sağlamaktadır. Karıştırma yapılarak arama uzayının bir bölgesinden başka bir bölgesine geçiş yapılır ve bu şekilde algoritmanın çözüm uzayının farklı bölgelerini arayabilmesine imkan tanınır.

IYA'nın karıştırma işlemi belirli bir kurala göre veya rassal seçilmiş olan komşuluk yapılarını mevcut çözüme belirli sayıda tekrarla uygulayarak gerçekleştirilir. Tekrar sayısı karıştırma kuvveti olarak adlandırılır. Karıştırma kuvvetinin yüksek olması algoritmanın çeşitlendirme kuvvetini arttıracağı için iyi seçilmesi gerekir. Çok yüksek bir karıştırma kuvveti, algoritmayı iyi çözümlerden uzaklaştırırken, çok düşük bir karıştırma kuvveti ise algoritmanın yerel en iyi çözümlere takılmasına neden olacaktır.

Kabul kriteri:

Kabul kriteri yerel aramadan çıkan yeni çözümün bir sonraki aşamaya mevcut çözüm olarak taşınıp taşınmayacağını belirler (Talbi, 2009).

4.2.3. Değişken komşu arama algoritması

Değişken komşu arama algoritması ilk kez Mladenovic & Hansen (1997) tarafından geliştirilmiş olup, arama esnasında kullanılan komşuluk yapılarını sistematik bir şekilde değiştirerek çeşitlendirme sağmalamayı amaçlayan bir algoritmadır. DKA algoritması aşağıda verilen özelliklere dayanmaktadır.

- Bir komşuluk yapısı ile elde edilen yerel minimum çözüm, başka bir komşuluk yapısı için yerel eniyi çözüm olmayabilir.
- Global çözüm tüm komşuluk yapıları için yerel eniyi çözümdür.
- Yerel eniyi çözüm bir veya birkaç komşuluk yapısı için genellikle benzerdir.

Değişken komşu arama algoritmasının dört farklı versiyonu bulunmaktadır.

- Değişken komşu iniş (DKİ) algoritması
- Temel değişken komşu arama (TDKA) algoritması
- İndirgenmiş değişken komşu arama (İDKA) algoritması
- Genel değişken komşu arama (GDKA) algoritması

İYA'ya dayalı geliştirilen hipersezgisel algoritmada yerel arama stratejisi olarak DKİ algoritması kullanılmıştır. DKİ, DKA'nın temel özelliklerinin yerel arama algoritmasına uygulanması ile elde edilir. DKİ algoritmasının adımlar Şekil 4.2.' de verilmektedir. DKİ komşuluk kümesindeki ilk komşuluk yapısı ile aramaya başlar. İlk komşuluk yapısı ile üretilen komşu çözüm mevcut çözümden iyi ise, aynı komşuluk yapısı ile, değilse bir sonraki komşuluk yapısı ile aramaya devam edilir. Herhangi bir komşuluk yapısı ile mevcut çözüm iyileştirildiğinde arama işlemi ilk komşuluk yapısına geri döner. Tüm komşuluk yapıları ile arama yapıldı ve mevcut çözüm iyileşmediyse algoritma sonlanır. Böylece algoritmada farklı komşuluk yapılarının kullanılması aramada çeşitlilik sağlamaktadır (Talbi, 2009).

begin
$N_k \leftarrow$ Komşuluk yapıları kümesini seç $(k = 1,2, \ldots, k_{max})$
$s_0 \leftarrow$ Başlangıç çözümü üret
    $k = 1$
    repeat
        $\hat{s} \leftarrow s_0'$dan $k.$ komşuluk yapısına göre en iyi komşuyu seç
        $f(\hat{s}) < f(s_0)$ ise $s_0 \leftarrow \hat{s}$ ve $k = 1$
                değilse $k = k + 1$
    until $k = kmax$ oluncaya kadar
end

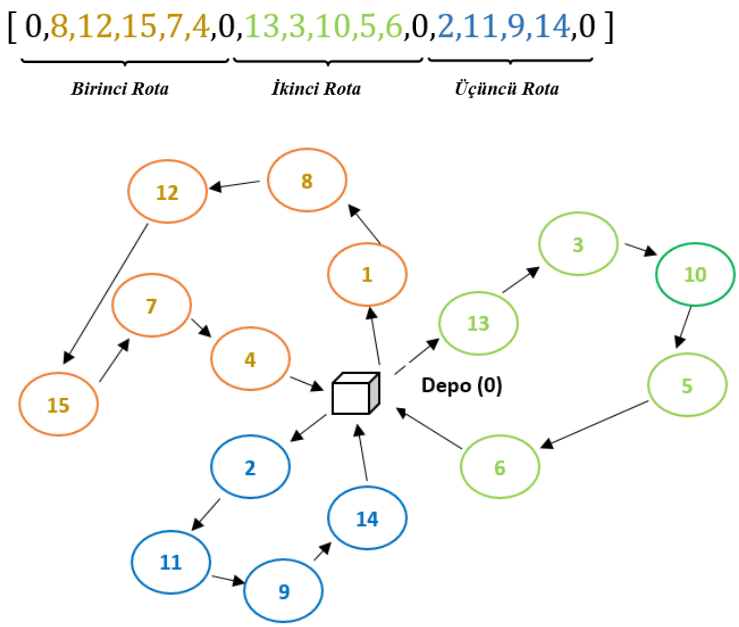
Şekil 4.2. Değişken komşu iniş algoritmasının adımları

5. Y-ETDARP İÇİN İYA_HS ALGORİTMASI

Bu tezde toplam yakıt tüketimi maliyetini enazlamayı amaçlayan Y-ETDARP için İYA ve DKİ tabanlı hipersezgisel bir algoritma geliştirilmiş ve İYA_HS olarak adlandırılmıştır. İYA_HS'de, İYA'nın çözüm uzayının farklı bölgelerinde arama yapabilme özelliği ile hipersezgisellerin komşuluk yapılarını daha etkili bir şekilde kullanabilme yeteneği birleştirilerek kısa zamanda daha iyi çözümlere ulaşabilmek amaçlanmıştır. Ayrıca İYA'da çözüm etrafında derinlemesine arama yapmak amacıyla DKİ algoritması kullanılmıştır. Bu bölümde İYA_HS'nin bileşenleri detaylı olarak incelenecektir.

5.1. Çözüm Gösterimi

İYA_HS'de çözümler bir dizi ile gösterilmiş, dizideki sıfırlar depoyu ifade etmek üzere her aracın rotası sıfırlar ile ayrılmıştır. Bir araç 0 noktasından yani depodan yola çıkar ve rota sonunda tekrar depoya geri döner. Gösterimde her bir rota 0 noktasında başlamakta ve 0 noktasında bitmektedir. Dizideki her iki sıfır arası bir aracın rotasını, sırayla hangi müşterilere uğrayacağını göstermektedir. Şekil 5.1.'de örnek bir kodlama ve karşılık gelen çözüm verilmiştir.



Şekil 5.1. İYA_HS'de çözüm gösterimi

Şekil 5.1.'de verilen örnek çözümde tüm müşterilere 3 araçla hizmet verilmekte, 1. araç, 1,4,7,15,12,8 numaralı müşterilere, 2. araç 2,11,9,14 numaralı müşterilere, 3. araç ise 13,3,10,5,6 numaralı müşterilere hizmet vermektedir.

5.2. Başlangıç Çözümünün Oluşturulması

Başlangıç çözümünün oluşturulmasında çözüm kurucu bir sezgisel olan en yakın komşu arama (EYK) sezgiseli kullanılmıştır. EYK sezgiselinde bir rota rassal olarak seçilen bir müşteriden başlayarak her seferinde mevcut müşteriye en yakın müşteri araç kapasitesine ulaşıncaya kadar rotaya eklenir. Araç kapasitesine ulaşıldığında yeni bir rota açılır ve tüm müşteriler bir rotaya yerleştirilinceye kadar aynı işleme devam edilir. *Nc* müşteriler kümesi olmak üzere en yakın komşu sezgiselinin adımları Şekil 5.2.'de verilmektedir.

---

begin
$N_c \leftarrow$ Müşteriler kümesi $(n = 1,2, \dots, N)$
$i \leftarrow$ Rassal bir müşteri seç
i müşterisini ilk rotaya ata
    repeat
        $j \leftarrow$ i müşterisine N_c kümesindeki en yakın müşteriyi seç
        j müşterisi için araç kapasitesi kontrolü yap
        Kapasite aşılmıyorsa j müşterisini mevcut rotaya ata
        Kapasite aşılıyorsa j müşterisini yeni bir rotaya ata
        j müşterisini N_c müşteriler kümesinden çıkar
        $i \leftarrow j$
    until $\{N_c = [\ ]\}$
end

Şekil 5.2. En yakın komşu sezgiselinin adımları

EYK sezgiseli ile problem için bir çözüm oluşturulduktan sonra, aramaya iyi bir başlangıç çözümü ile başlamak için, bu çözüm bölüm 5.5'de anlatılan yerel arama algoritması olarak kullanılan DKİ ile iyileştirilmiştir.

5.3. Komşuluk Yapıları

İYA_HS'nin karıştırma ve yerel arama adımında komşuluk yapıları kullanılmıştır. Karıştırma adımında mevcut çözümü değiştirerek arama uzayının farklı bölgelerinde yeni

bir çözüme, yerel arama adımında ise mevcut çözümden daha iyi komşu çözümlere ulaşabilmek için komşuluk yapıları uygulanmaktadır. Karıştırma adımında rotalararası komşuluk yapılarından, yerel arama adımında ise hem rotalararası hem de rotaiçi komşuluk yapılarından yararlanılmıştır. Daha önce bahsedildiği gibi yerel arama adımında DKİ algoritması kullanılmaktadır.

Bu kısımda algoritmada kullanılmış olan komşuluk yapıları ve bu komşuluk yapıları ile üretilen yeni çözümlerde kapasite aşım kontrol mekanizmaları anlatılmıştır. Oluşturulan yeni çözümlerin kapasite kontrolleri Belgin, Karaoglan & Altıparmak (2018) tarafından kullanılan yaklaşım ile gerçekleştirilmiştir. R bir çözümün sahip olduğu rotalar kümesi ve her bir rota r indisiyle gösterilmek üzere r rotasındaki i müşterisi c_i ile ve bulunduğu pozisyon ise t_i ile ifade edilmektedir. Eşitliklerde kullanılan notasyonlar aşağıda verilmektedir.

Q : arac kapasitesi
d_{c_i} : c_i müşterisinin dağıtım talebi
p_{c_i} : c_i müşterisinin toplama talebi
DL_{rt}^{+} : r rotasındaki t pozisyonundaki müşteriden hemen sonra araçtaki toplam dağıtım yükü miktarı
PL_{rt}^{+} : r rotasındaki t pozisyonundaki müşteriden hemen sonra araçtaki toplam toplama yükü miktarı
DL_{rt}^{-} : r rotasındaki t pozisyonundaki müşteriden hemen önce araçtaki toplam dağıtım yükü miktarı
PL_{rt}^{-} : r rotasındaki t pozisyonundaki müşteriden hemen önce araçtaki toplam toplama yükü miktarı
ML_{rt}^{+} : r rotasındaki t pozisyonundaki müşteriden sonra araçtaki maksimum yük
ML_{rt}^{-} : r rotasındaki t pozisyonundaki müşteriden önce araçtaki maksimum yük

Karıştırma işleminde kullanılan rotalararası komşuluk yapıları:

Karıştırma işlemi için 8 farklı komşuluk yapısı kullanılmıştır. Bu yapılar aşağıda kısaca anlatılmaktadır.

i) Rassal rotalar arası yer değiştirme (1,1) : Rassal olarak seçilen r_1 rotasından rassal olarak seçilen c_1 müşterisi ile yine rassal olarak seçilen r_2 rotasından rassal olarak seçilen c_2 müşterisinin yerleri değiştirilir. Kapasite kontrolü aşağıda verilen eşitliklerle gerçekleştirilir.

$$max\left\{ML^{-}_{r_1 t_{c_1}} - d_{c_1} + d_{c_2},\ ML^{+}_{r_1 t_{c_1}} - p_{c_1} + p_{c_2}\right\} \leq Q \quad (5.1)$$

$$max\left\{ML^{-}_{r_2 t_{c_2}} - d_{c_2} + d_{c_1}, ML^{+}_{r_2 t_{c_2}} - p_{c_2} + p_{c_1}\right\} \leq Q \quad (5.2)$$

ii) Açgözlü rotalararası yer değiştirme (1,1) : Rassal olarak seçilen r_1 rotasından rassal olarak bir c_1 müşterisi seçilir. Rassal olarak seçilen r_2 rotasında bulunan tüm müşteriler denenerek c_1 müşterisinin yeri eniyi c_2 müşterisi ile değiştirilir. Kapasite kontrolü Eşitlik 5.1 ve Eşitlik 5.2 kullanılarak yapılır.

iii) Rassal rotalararası yer değiştirme (2,1) : Rassal olarak seçilen r_1 rotasından rassal olarak seçilen c_1 müşterisi ve c_1 müşterisinden hemen sonra gelen müşteri ile yine rassal olarak seçilen r_2 rotasından rassal olarak seçilen c_2 müşterisinin yerleri değiştirilir. Kapasite kontrolü Eşitlik 5.1 ve Eşitlik 5.2 kullanılarak yapılır.

iv) Açgözlü rotalararası yer değiştirme (2,1) : Rassal olarak seçilen r_1 rotasından rassal olarak bir c_1 müşterisi seçilir. Rassal olarak seçilen r_2 rotasında bulunan tüm müşteriler denenerek c_1 ve bu müşteriden hemen sonra gelen müşterinin yeri eniyi c_2 müşterisi ile değiştirilir. Kapasite kontrolü Eşitlik 5.1 ve Eşitlik 5.2 kullanılarak yapılır.

v) Rassal rotalararasi ekleme (1,0) : Rassal olarak seçilen r_1 rotasındaki rassal olarak seçilen bir c müşterisi, r_1 rotasından alınarak rassal olarak seçilen r_2 rotasındaki yine rassal olarak seçilen bir t_B pozisyonuna yerleştirilir. Kapasite kontrolü aşağıda verilen eşitlikle gerçekleştirilir.

$$max\left\{ML^{-}_{r_2 t_B} + d_c,\ ML^{+}_{r_2 t_B} + p_c\right\} \leq Q \quad (5.3)$$

vi) Açgözlü rotalararasi ekleme (1,0) : Rassal olarak seçilen r_1 rotasındaki rassal olarak seçilen bir c müşterisi, r_1 rotasından alınarak rassal olarak seçilen r_2 rotasındaki eniyi t_B pozisyonuna yerleştirilir. Kapasite kontrolü Eşitlik 5.3 kullanılarak yapılır.

vi) Rassal rotalararasi ekleme (2,0) : Rassal olarak seçilen r_1 rotasındaki rassal olarak seçilen bir c müşterisi ve bu müşteriden hemen sonra gelen müşteri r_2 rotasındaki yine rassal olarak seçilen bir t_B pozisyonuna yerleştirilir. Kapasite kontrolü Eşitlik 5.3 kullanılarak yapılır.

vii) Açgözlü rotalararasi ekleme (2,0) : Rassal olarak seçilen r_1 rotasındaki rassal olarak seçilen bir c müşterisi ve bu müşteriden hemen sonra gelen müşteri r_2 rotasındaki eniyi t_B pozisyonuna yerleştirilir. Kapasite kontrolü Eşitlik 5.3 kullanılarak yapılır.

DKİ algoritmasında kullanılan rotalararası komşuluk yapıları:

DKİ'de kullanılan 5 farklı rotalararası komşuluk yapısı aşağıda verilmiştir.

i) Rotalararası yer değiştirme (1,1) : Rassal olarak seçilen r_1 ve r_2 rotalarındaki tüm mümkün yer değiştirmeler denenerek eniyi yer değiştirme gerçekleştirilir. Kapasite kontrolü Eşitlik 5.1 ve Eşitlik 5.2 kullanılarak yapılır.

ii) Rotalararası ekleme (1,0) : Rassal olarak seçilen r_1 rotasındaki tüm mümkün müşterileri ve rassal olarak seçilen r_2 rotasındaki tüm mümkün pozisyonlar denenerek eniyi yerleştirme gerçekleştirilir. Kapasite kontrolü Eşitlik 5.3 kullanılarak yapılır.

iii) Rotalararası yer değiştirme (2,1) : Rassal olarak seçilen r_1 rotasındaki tüm mümkün ardışık müşteri ikilileri ve rassal olarak seçilen r_2 rotasındaki tüm mümkün müşteriler denenerek eniyi yer değiştirme gerçekleştirilir. Kapasite kontrolü Eşitlik 5.1 ve Eşitlik 5.2 kullanılarak yapılır.

iv) Rotalararası ekleme (2,0) : Rassal olarak seçilen r_1 rotasındaki tüm mümkün ardışık müşteri ikilileri ve rassal olarak seçilen r_2 rotasındaki tüm mümkün pozisyonlar denenerek eniyi yerleştirme gerçekleştirilir. Kapasite kontrolü Eşitlik 5.3 kullanılarak yapılır.

v) Çaprazlama : a_1 hattı rassal seçilen bir r_1 rotasındaki t_{c1} ve t_{c1+1} pozisyonlarındaki müşterileri bağlayan hat olsun. Aynı şekilde a_2 hattı da rassal seçilen bir r_2 rotasındaki t_{c2} ve t_{c2+1} pozisyonlarındaki müşterileri bağlayan hat olsun. a_1 ve a_2 hatları kesilir ve rotalar ikiye ayrılır. r_1 rotasının ilk kısmı ile r_2 rotasının ikinci kısmı ve $r_2$rotasının ilk kısmı ile r_1 rotasının

ikinci kısmı birleştirilerek iki yeni rota oluşturulur. r_1 ve r_2 rotasındaki tüm mümkün hatlar denenerek eniyi hareket gerçekleştirilir. Kapasite kontrolü aşağıdaki eşitlikler yardımı ile yapılmaktadır.

$$ML^{-}_{r_1,t_{c1}+1} - DL^{-}_{r_1,t_{c1}+1} + DL^{-}_{r_2,t_{c2}+1} \leq Q \quad (5.4)$$

$$ML^{-}_{r_2,t_{c2}} - PL^{+}_{r_2,t_{c2}} + PL^{+}_{r_1,t_{c1}} \leq Q \quad (5.5)$$

$$ML^{-}_{r_2,t_{c2}+1} - DL^{-}_{r_2,t_{c2}+1} + DL^{-}_{r_1,t_{c1}+1} \leq Q \quad (5.5)$$

$$ML^{-}_{r_1,t_{c1}} - PL^{+}_{r_1,t_{c1}} + PL^{+}_{r_2,t_{c2}} \leq Q \quad (5.7)$$

vi) Rotalararası yer değiştirme (2,2) : Rassal olarak seçilen r_1 rotasındaki tüm mümkün ardışık müşteri ikilileri ve rassal olarak seçilen r_2 rotasındaki tüm mümkün ardışık müşteri ikilileri denenerek eniyi yer değiştirme gerçekleştirilir. Kapasite kontrolü Eşitlik 5.1 ve Eşitlik 5.2 kullanılarak yapılır.

DKİ algoritmasında kullanılan rotaiçi komşuluk yapıları:

DKİ'de 3 farklı rotaiçi komşuluk yapısı kullanılmıştır. Bu komşuluk yapıları aşağıda verilmiştir.

i) Two-opt : Rassal olarak seçilen aynı r rotasındaki ardışık olmayan iki hat kesilir (i,j ve m,n) ve iki yeni hat oluşturulur (i,m ve j,n). Bu hareket kesilen hatlar arasında kalan rota parçasını ters çevirir. Kapasite kontrolü aşağıda verilen eşitlikle gerçekleştirilir.

$$DL^{+}_{rt_i} + PL^{+}_{rt_i} + \sum_{s=t_m-t}^{t_m} d_{r_s} - \sum_{s=t_m-t}^{t_m} p_{r_s} \leq Q \quad t = 0,1,\dots,t_m - t_j \quad (5.8)$$

ii) Rotaiçi yer değiştirme : Rassal olarak seçilen r rotasında tüm mümkün c_1 ve c_2 müşterileri ikilileri denenerek eniyi yer değiştirme yapılır. Kapasite kontrolü aşağıda verilen eşitlikle gerçekleştirilir.

$$if\ t_i < t_j DL^{+}_{rt} + PL^{+}_{rt} + d_{c_1} - p_{c_1} - d_{c_2} + p_{c_2} \leq Q \quad t = t_i, t_{i+1}, \dots, t_{j-1}$$
$$if\ t_i > t_j DL^{+}_{rt} + PL^{+}_{rt} + d_{c_2} - p_{c_2} - d_{c_1} + p_{c_1} \leq Q \quad t = t_j, t_{j+1}, \dots, t_{i-1} \quad (5.9)$$

iii) Rotaiçi ekleme: Rassal olarak seçilen *r* rotasındaki tüm mümkün müşteriler ve yine aynı rotadaki mevcut pozisyon dışındaki tüm mümkün pozisyonlar denenerek seçilen *c* müşterisi en uygun t_B pozisyonuna yerleştirilir. t_i, *c* müşterisinin mevcut pozisyonu olmak üzere kapasite kontrolü aşağıda verilen eşitlikle gerçekleştirilir.

$$if\ t_i < t_B DL_{rt}^{+} + PL_{rt}^{+} + d_c - p_c \leq Q \quad t = t_{i+1}, t_{i+2}, \dots, t_B$$
$$if\ t_i > t_B DL_{rt}^{-} + PL_{rt}^{-} - d_c + p_c \leq Q \quad t = t_B, t_{B+1}, \dots, t_{i-1} \qquad (5.10)$$

5.4. Komşuluk Yapısı Seçim Fonksiyonu

Hipersezgisel algoritmalar, arama sırasında, çözümler üzerinde büyük etkisi olan alt düzey sezgiselleri (komşuluk yapıları) sistematik bir şekilde kullanarak daha iyi çözümlere daha kısa sürede ulaşmayı sağlarlar. Arama sırasında hangi alt seviye sezgisellerin kullanıldığı kadar, bu sezgisellerin hangi sırada ve yöntemle kullanıldığı da algoritmanın performansı için oldukça önemlidir. Bu tezde, algoritmanın hem karıştırma adımında hem de yerel arama safhasında kullanılan komşuluk yapıları, operatörlerin geçmiş performansları göz önüne alınarak kullanılmıştır.

Her aşamada mevcut çözüme uygulanacak komşuluk yapısının belirlenebilmesi için Maashi, Kendall ve Özcan (2015) tarafından çok amaçlı optimizasyon için geliştirilen ve Seçim Fonksiyonu (CF) olarak adlandırılan seçim stratejisi tek amaçlı optimizasyona uyarlanarak kullanılmıştır. Seçim fonksiyonu komşuluk yapıları geçmiş performanslarını dikkate alarak iyi performans sergileyen komşuluk yapısını seçmeyi sağlarken aynı zamanda uzun zaman seçilmeyen komşuluk yapısını seçmek için fırsat tanımaktadır. Bu şekilde algoritmanın çeşitlendirme ve yoğunlaştırma dengesini sağlamaktadır. CF, her aşamada hareket operatörleri için bu dengeyi sağlayacak şekilde bir seçim puanı belirler ve bu puanlara dayanarak mevcut aşamada uygulanacak komşuluk yapısını belirler. $H=\{h_1,h_2,\dots,h_N\}$ komşuluk yapıları kümesi olmak üzere h_i operatörüne atanan seçim puanı (CF) Eşitlik 5.11 ile belirlenmektedir.

$$\mathrm{CF}(h_i) = \propto * f_1(h_i) + f_2(h_i) \quad \forall h_i \in H \qquad (5.11)$$

Bu eşitlikte f_1 yoğunlaştırmayı sağlayan fonksiyon bileşenidir ve *N* operatör havuzundaki komşuluk yapısı sayısı $\mathrm{SIRA}_{\mathrm{ENİYİ}}$, komşuluk yapılarının o aşamaya kadar buldukları eniyi

çözüme göre sıralanmış durumunu, $\mathrm{SIRA}_{\mathrm{ENSON}}$ komşuluk yapılarının son uygulamalarında bulmuş oldukları son çözümlerine göre sıralanmış durumunu ifade etmek üzere Eşitlik 5.12'de verildiği gibi hesaplanır.

$$f_1(h_i) = 2(N+1) - \{\mathrm{SIRA}_{\mathrm{ENİYİ}}(h_i) + \mathrm{SIRA}_{\mathrm{ENSON}}(h_i)\} \quad (5.12)$$

Çeşitlendirme ise komşuluk yapılarının en son uygulandığından beri geçen iterasyon sayısını ifade eden f_2 bileşeniyle sağlanmaktadır.

Seçim Fonksiyonu bu şekilde algoritmayı yüksek performanslı komşuluk yapılarını seçmeye yöneltirken uzun zamandır seçilmeyen komşuluk yapılarına da şans vererek aramayı çözüm uzayının farklı bölgelerine yönlendirmektedir. Eşitlik 5.11'de verilmiş olan $\propto$ parametresi çeşitlendirme ve yoğunlaştırmayı dengelemeye yarayan parametredir.

Bu tezde geliştirilen İYA_HS algoritmasında hem karıştırma aşamasında hem de DKİ aşamasında kullanılacak komşuluk yapıları Seçim Fonksiyonu ile akıllı bir şekilde seçilerek algoritmaya hipersezgisel özelliği kazandırılmıştır. Her aşamada Seçim Fonksiyonu ile komşuluk yapılarının puanları belirlenmiş, daha sonra ise bu puanlara dayalı seçim yapılmıştır. DKİ'de maksimum CF değerine sahip komşuluk yapısı seçilirken karıştırma adımında çeşitlendirmeyi arttırmak amacıyla CF puanlarına dayalı olasılıklar belirlenmiş ve rulet çemberi stratejisi ile seçim yapılmıştır.

5.5. İYA_HS algoritmasının adımları

Geliştirilen İYA_HS algoritması iteratif yerel arama tabanlı bir algoritmadır. Çözümler, her iterasyon başında komşuluk yapıları art arda belirli bir sayıda (karıştırma kuvveti) uygulanarak karıştırılır ve arama çözüm uzayının farklı bölgelerine sıçratılır. Daha sonra yerel arama ile mevcut çözümün iyi komşuları aranarak çözümde iyileştirme yapılır. Yerel arama ile elde edilen çözüm belirli bir kabul kriterine göre bir sonraki iterasyona mevcut çözüm olarak taşınır veya önceki mevcut çözümle algoritma bir sonraki iterasyona devam eder. Şekil 5.3'te geliştirilen İYA_HS algoritmasının genel işleyişi verilmiştir.

```
begin
s_0 ← Başlangıç çözümünü üret
s* ← Başlangıç çözümüne yerel arama uygula
id ← 0
    while (id < max _id) do
        s' ← Karıştır(s*)
        ya ← 0
        while (ya < max _ya) do
            s*' ← YAA1_HS(s')
            eğer f(s*') ≤ f(s') ise s* ← s*' ve ya ← 0
                                değilse ya ← ya + 1
        end
        eğer f(s*') ≤ f(s') ise s* ← s*' ve id ← 0
                            değilse id ← id + 1

    end
end
```

Şekil 5.3. İYA_HS algoritmasının adımları

Kullanılan yerel arama algoritması DKİ'ye dayalı olup, YAA1_HS olarak adlandırılmıştır. Bu algoritma, Subramanian ve diğerleri (2010) tarafından ETDARP için geliştirilen önce rotalararası daha sonra ise rotaiçi iyileştirmeye dayalı olan yerel arama algoritmasından yararlanılarak geliştirilmiştir. YAA1_HS'de önce rotalararası komşuluk yapılarıyla iyileştirme yapılmakta, rotalararası komşuluk yapılarıyla iyileştirilebilen çözümlere rotaiçi komşuluk yapılarıyla yerel arama uygulanmaktadır. Rotaiçi komşuluk yapıları ile yerel arama, YAA2_HS olarak adlandırılmıştır. YAA1_HS adımları Şekil 5.4'de, YAA2_HS adımları Şekil 5.5'de verilmektedir.

Yerel arama aşamalarında olduğu gibi algoritmanın karıştırma aşamasında da kullanılacak olan komşuluk yapıları seçim fonksiyonuyla belirlenmiştir. Böylece algoritmanın her adımında komşuluklar sistematik ve daha etkili bir şekilde kullanılmış, bu durum ileride verilecek olan deneysel çalışmalarda da görüleceği üzere algoritmanın arama kabiliyetini arttırmıştır. Algoritmanın karıştırma alt programı Şekil 5.6'da verilmiştir.

$N_{ra} = \{h_1, h_2, \dots, h_N\}$: Rotalar arasi komşuluk yapıları kümesi

$s^{*\prime}$: YAA1_HS sonucu elde edilen çözüm

begin

$s'' \leftarrow s'$
$i \leftarrow 1$

while $(N_{ra} \neq [\ ])$ **do**

$\text{SIRA}_{\text{ENİYİ}}(h_i) \leftarrow$ Komşuluk yapılarını en iyi çözümlerine göre sırala

$\text{SIRA}_{\text{ENSON}}(h_i) \leftarrow$ Komşuluk yapılarını son çözümlerine göre sırala

$\forall h_i$ için $f_1(h_i)$ ve $\text{CF}(h_i)$ değerlerini hesapla

$h_j \leftarrow$ maksimum CF skorlarına sahip komşuluk yapısını seç
s'' çözümünden rassal olarak iki rota (r_1 ve r_2) seç

$ei \leftarrow h_j$ komşuluk yapısına göre r_1 ve r_2 rotaları için mümkün tüm hareketleri gerçekleştir ve en iyi hareketi seç

$s''' \leftarrow s''$ çözümüne ei hareketini uygula

if $f(s''') \leq f(s'')$ **then**

$s'' \leftarrow s'''$

$s^{**} \leftarrow$ YAA2_HS ile s'' çözümünü iyileştir

$s'' \leftarrow s^{**}$

else

h_j komşuluk yapısını N_{ra} kümesinden çıkar

end

$s^{*\prime} \leftarrow s''$

end

Şekil 5.4. YAA1_HS adımları

$N_{ri} = \{h_1, h_2, \ldots, h_N\}$: Rota ici komşuluk yapıları kümesi

s'': YAA2_HS sonucu elde edilen çözüm

begin

s'' çözümünde değiştirilen rotaları (r_1 ve r_2) seç

r_1 ve r_2 rotaları için aşağıdaki adımları tekrarla

for $i = 1$ to 2 **do**

$s \leftarrow s''$

$r \leftarrow r_i$ $i = 1$ ise r_1 rotası, $i = 2$ ise r_1 rotası

$i \leftarrow 1$

while $(N_{ri} \neq [\ ])$ **do**

$\text{SIRA}_{\text{ENİYİ}}(h_i) \leftarrow$ Komşuluk yapılarını en iyi çözümlerine göre sırala

$\text{SIRA}_{\text{ENSON}}(h_i) \leftarrow$ Komşuluk yapılarını son çözümlerine göre sırala

$\forall h_i$ için $f_1(h_i)$ ve $\text{CF}(h_i)$ değerlerini hesapla

$h_j \leftarrow$ maksimum CF skorlarına sahip komşuluk yapısını seç

$ei \leftarrow h_j$ komşuluk yapısına göre r_i rotası için mümkün tüm mümkün hareketleri gerçekleştir ve en iyi hareketi seç

$s'' \leftarrow s$ çözümüne ei hareketini uygula

if $f(s'') \leq f(s)$ **then**

$s \leftarrow s''$

else

h_j komşuluk yapısını N_{ri} kümesinden çıkar

end while

$s'' \leftarrow s$

end for

end

Şekil 5.5. YAA2_HS adımları

$\{h_1, h_2, \ldots, h_N\}$: karıştırma adımında kullanılacak rotalar arası komşuluk yapıları

begin

$s^+ \leftarrow s^*$

$\text{SIRA}_{\text{ENİYİ}}(h_i) \leftarrow$ Komşuluk yapılarını en iyi çözümlerine göre sırala

$\text{SIRA}_{\text{ENSON}}(h_i) \leftarrow$ Komşuluk yapılarını son çözümlerine göre sırala

$\forall h_i$ için $f_1(h_i)$ ve $\text{CF}(h_i)$ değerlerini hesapla

$h_j \leftarrow$ CF skorlarına göre olasılık değerleri ata ve rulet çemberi ile bir komşuluk yapısı seç

Karıştırma kuvvetini (k) $[1 - K]$ arasında rassal olarak seç

$s' \leftarrow s^+$ çözümüne k defa h_j komşuluk yapısını uygula

end

Şekil 5.6. İYA_HS için karıştırma yaklaşımı adımları

6. DENEYSEL ANALİZ

Bu bölümde öncelikle İYA_HS, Y-ETDARP'ın çözümünde kullanılmış ve kullanılan amaç fonksiyonunun yakıt tüketim maliyetine etkisi incelenmiştir. Bir sonraki aşamada ise, literatürde üzerinde çok çalışılan ve Y-ETDARP'ın alt problemi olan ETDARP üzerinde İYA_HS'nin performans analizi gerçekleştirilmiştir. İYA_HS, C++ programlama dilinde kodlanmış ve denemeler Intel Core 2.6 GHz bilgisayar ortamında gerçekleştirilmiştir. Her test problemi için İYA_HS, 30 kez koşturulmuş ve elde edilen çözüm değerleri ile çözüm zamanları tutulmuştur. Sayısal analizler verilmeden önce kullanılan test problemleri ve İYA_HS'nin parametreleri hakkında bilgi verilecektir.

6.1. Test Problemleri

İYA_HS için sayısal analiz aşamasında Salhi ve Nagy (1999) ve Dethloff (2001) tarafından üretilen test problemleri kullanılmıştır.

Salhi ve Nagy (1999) test problemleri:

Bu test kümesi, Chirstofides, Mingozzi & Toth (1979) tarafındankapasite kısıtlı ARP için oluşturulmuş test problemlerinden türetilen 50-199 arası müşterininolduğu28 adet problemden oluşmaktadır. a müşterisinin koordinatları (X_a,Y_a),orijinal talep miktarı $t_a$ve toplam taleplerin bölünmesiyle elde edilen dağıtım ve toplama talepleri sırasıyla q_a *ve* p_a ile ifade edilmek üzere her bir müşteri için $r_a=min((X_a/ Y_a),(Y_a / X_a))$oranı hesaplanmıştır. Her a müşterisinin dağıtım talebi $q_a=r_a.t_a$,ve toplama talebi $p_a=(1-r_a).t_a$ eşitlikleri ile elde edilmiştir. Bu şekilde orijinal talep, dağıtım ve toplama taleplerine bölünmüştür. İlk 14 problem bu yaklaşım ile elde edilmiş ve CMTX grubu olarak adlandırılmıştır. Diğer 14 problem ise dağıtım ve toplama taleplerinin tam tersi şekilde hesaplanması ile elde edilmiştir ve CMTY grubu olarak adlandırılmıştır.

Dethloff (2001) test problemleri:

Bu test kümesi,her biri 50 müşteriden oluşan 40 adet test problemi içermektedir. Problemler iki farklı coğrafi senaryo üzerinden oluşturulmuştur. SCA olarak adlandırılan senaryoda

müşterilerin koordinatları [0, 100] aralığı içinde eşit olarak dağıtılmaktadır. CON senaryosunda ise müşterilerin ilk yarısı SCA senaryosunda olduğu gibi dağıtılmış, diğer yarısı ise [100/3, 200/3] aralığında daha sık bir biçimde dağıtılmıştır. Bu şekilde alanın bir kısmını daha kentsel bir yapıya benzetmek amaçlanmıştır. Her iki senaryoda da müşteriler arası uzaklıklar için Öklit uzaklıkları ölçülmüştür. Müşterilerin dağıtım talepleri (D_j), [0, 100] aralığında rassal olarak dağıtılmış, toplama talepleri ise dağıtım taleplerine bağlı olarak r_j ~ Düzgün[0, 1] olmak üzere $P_j = (0{,}5 + r_j)D_j$ eşitliği ile elde edilmiştir. Her bir coğrafi senaryo için gereken minimum araç sayısına dayalı olarak, farklı araç kapasitelerine sahip ikişer grup oluşturmuştur. Problemlerin isimlerinde harflerden sonra gelen 3 ve 8 problemlerin araç kapasitelerine göre oluşturduğu iki grubu ifade etmektedir.

6.2. İYA_HS Parametreleri

Geliştirilen İYA_HS'nin parametreleri; karıştırma kuvveti, mevcut çözüm iyileşmediğinde YAA1_HS'nin ardışık uygulanma sayısı ve YAA1_HS ve İYA_HS içiniyileşme olmadan yapılacak ardışık iterasyon sayısıdır. Bu parametreler yapılan ön denemeler sonucunda belirlenmiştir.

Algoritmanın her iterasyonunda değişken karıştırma kuvveti kullanılmıştır. Amaç, algoritmanın çözüm uzayının farklı bölgelerinde aramaya devam etmesini sağlamaktır. Bu nedenle, her iterayonda karıştırma kuvveti [1-K] aralığında rassal olarak seçilmiş ve K değeri 5 olarak alınmıştır.

Karıştırma işleminden sonra mevcut çözüme uygulanan YAA1_HS için durdurma koşulu iyileşme olmadan yapılacak ardışık iterasyon sayısı (*max_id*) olarak dikkate alınmıştır. Bunun yanısıra, YAA1_HS ile mevcut çözüm iyileştirilemiyor ise yerel arama süreci tekrar başlatılır. *max_id* değeri 20 olarak alınmıştır.

İYA_HS algoritması için durdurma koşulu, arama sırasında bulunan eniyi çözümün ardışık *n_rep* iterasyonda iyileşmemesi olup bu değer 300 olarak alınmıştır.

6.3. Y-ETDARP için Performans Analizi

Bu bölümde öncelikle, Y-ETDARP'da kullanılan amaç fonksiyonunun yakıt tüketim maliyetine etkisi analiz edilmiştir. Bu analizde, Y-ETDARP ve ETDARP çözümleri karşılaştırmalı incelenmiştir. Bilindiği gibi Y-ETDARP'da amaç fonksiyonu Eşitlik 6.1'de verilen toplam seyahat mesafesinin enazlanması olarak değiştirildiğinde ETDARP elde edilmektedir.

$$\text{En küçük } Z = \sum_{i=0}^{n} \sum_{j=0}^{n} c_{ij} x_{ij} \tag{6.1}$$

Bu analizde, Dethloff'ın (2001) test problemleri kullanılmıştır. Test problemleri için müşteri taleplerinin ve araç kapasitelerinin ağırlık cinsinden olduğu varsayılmıştır. Araçların yüksüzken normalleştirilmiş yakıt tüketim oranı 1, tam yüklüyken yakıt tüketim oranı ise 2 ve birim yakıt tüketimi maliyeti de 1 olarak alınmıştır.

İYA_HS ile Y-ETDARP ve ETDARP için çözümler elde edilmiş ve Y_ETDARP'ın çözümleri toplam yakıt tüketim maliyetleri ve toplam seyahat mesafeleri açısından ETDARP'ın çözümleri ile karşılaştırılmıştır. Amaç, toplam seyahat mesafesinin artması pahasına toplam yakıt tüketim maliyetinin ne kadar azaltılabileceğini görmektir. Bu amaçla, Y-ETDARP için elde edilen eniyi çözümün toplam seyahat mesafesi, ETDARP için elde edilen eniyi çözümün de toplam yakıt tüketim maliyeti hesaplanmıştır.

Çizelge 6.1'de Y-ETDARP ve ETDARP için IYA-HS ile elde edilen sonuçlar verilmektedir. Çizelgede ilk sütun ilgili test problemlerini, sonraki dört sütun Y-ETDARP için sonuçları vermektedir. Y-ETDARP ile ilgili sütunların birincisinde test probleminin 30 koşumu sonucunda elde edilen eniyi çözümün toplam tüketim maliyetini, ikinci sütun ilgili çözüm için hesaplanan toplam seyahat mesafesini, üçüncü sütun 30 koşuma dayalı toplam tüketim maliyeti ortalamasını ve dördüncü sütun da 30 koşumda elde edilen çözümlerin hesaplanan toplam seyahat mesafesi ortalamasını vermektedir. Çizelgede ETDARP başlığı altındaki dört sütunda benzer şekilde ETDARP için İYA-HS ile 30 koşum sonucunda eniyi çözümün

Çizelge 6.1. Y-ETDARP ve ETDARP için İYA_HS ile elde edilen sonuçlar

Problem	Y-ETDARP				ETDARP				Y-ETDARP için Yakıt Tüketimi Maliyeti % Sapma		Y-ETDARP için Toplam Mesafe % Sapma	
	Enküçük		Ortalama		Enküçük		Ortalama					
	Yakıt Tüketimi Maliyeti	İlgili Toplam Mesafe	Yakıt Tüketimi Maliyeti	İlgili Toplam Mesafe	Toplam Mesafe	İlgili Yakıt Tüketimi Maliyeti	Toplam Mesafe	İlgili Yakıt Tüketimi Maliyeti	Enküçük	Ortalama	Enküçük	Ortalama
SCA3-0	1119,80	640,55	1120,07	641,65	635,62	1120,31	640,05	1135,40	-0,05	-1,35	0,78	0,25
SCA3-1	1232,47	706,90	1232,51	706,95	697,84	1235,65	701,88	1249,98	-0,26	-1,40	1,30	0,72
SCA3-2	1166,84	659,34	1171,03	661,93	659,34	1166,84	660,47	1177,10	0,00	-0,52	0,00	0,22
SCA3-3	1178,32	682,01	1178,64	682,15	680,04	1180,11	681,72	1201,62	-0,15	-1,91	0,29	0,06
SCA3-4	1211,38	697,98	1211,38	697,98	690,50	1237,59	690,59	1261,79	-2,12	-3,99	1,08	1,07
SCA3-5	1194,27	682,10	1198,09	690,71	659,90	1244,68	660,87	1253,28	-4,05	-4,40	3,36	4,51
SCA3-6	1127,03	652,94	1129,81	655,02	651,09	1130,29	653,16	1140,42	-0,29	-0,93	0,28	0,29
SCA3-7	1191,62	671,67	1192,25	671,74	659,17	1194,23	669,73	1209,32	-0,22	-1,41	1,90	0,30
SCA3-8	1246,02	719,48	1246,02	719,48	719,48	1246,02	720,40	1258,45	0,00	-0,99	0,00	-0,13
SCA3-9	1205,91	681,64	1207,21	685,66	681,00	1211,31	681,11	1216,36	-0,45	-0,75	0,09	0,67
SCA8-0	1752,43	975,14	1800,08	999,32	961,50	1803,03	973,14	1839,11	-2,81	-2,12	1,42	2,69
SCA8-1	1944,37	1073,56	1964,60	1100,18	1049,65	1993,78	1068,47	2019,36	-2,48	-2,71	2,28	2,97
SCA8-2	1925,19	1050,37	1931,53	1055,82	1042,69	1933,65	1050,64	1943,05	-0,44	-0,59	0,74	0,49
SCA8-3	1810,98	1010,59	1832,74	1021,12	983,34	1831,88	1001,76	1876,87	-1,14	-2,35	2,77	1,93
SCA8-4	1972,05	1082,54	1986,47	1086,22	1065,49	2021,02	1071,66	2033,90	-2,42	-2,33	1,60	1,36
SCA8-5	1926,65	1039,91	1927,74	1080,14	1027,09	1941,71	1043,37	1966,30	-0,78	-1,96	1,25	3,52
SCA8-6	1806,29	973,27	1807,55	981,59	971,82	1808,59	974,85	1822,51	-0,13	-0,82	0,15	0,69
SCA8-7	1962,32	1078,19	1965,00	1080,63	1051,28	1983,77	1064,78	1994,89	-1,08	-1,50	2,56	1,49
SCA8-8	1958,42	1084,65	1963,87	1112,34	1071,18	1997,93	1082,35	2023,28	-1,98	-2,94	1,26	2,77
SCA8-9	1931,37	1067,42	1975,54	1075,72	1061,50	1975,98	1070,93	2006,92	-2,26	-1,56	0,56	0,45
CON3-0	1091,96	631,97	1093,76	636,54	616,52	1127,53	619,96	1148,13	-3,15	-4,74	2,51	2,67

Çizelge 6.1. (devam) Y-ETDARP ve ETDARP için İYA_HS ile elde edilen sonuçlar

Problem	Y-ETDARP				ETDARP				Y-ETDARP için Yakıt Tüketimi Maliyeti % Sapma		Y-ETDARP için Toplam Mesafe % Sapma	
	Enküçük		Ortalama		Enküçük		Ortalama		Enküçük	Ortalama	Enküçük	Ortalama
	Yakıt Tüketimi Maliyeti	İlgili Toplam Mesafe	Yakıt Tüketimi Maliyeti	İlgili Toplam Mesafe	Toplam Mesafe	İlgili Yakıt Tüketimi Maliyeti	Toplam Mesafe	İlgili Yakıt Tüketimi Maliyeti				
CON3-1	938,02	573,15	964,14	578,84	554,47	1024,01	558,20	1044,72	-8,40	-7,71	3,37	3,70
CON3-2	889,64	535,90	892,14	537,39	518,00	925,60	521,09	937,86	-3,88	-4,88	3,45	3,13
CON3-3	1030,27	597,43	1032,12	599,63	591,19	1064,18	591,20	1088,93	-3,19	-5,22	1,06	1,43
CON3-4	1027,39	608,57	1027,47	608,94	588,79	1065,39	591,66	1097,60	-3,57	-6,39	3,36	2,92
CON3-5	976,33	598,58	977,30	598,70	563,70	1036,48	564,86	1056,32	-5,80	-7,48	6,19	5,99
CON3-6	885,52	521,11	886,84	525,97	499,05	920,10	502,39	943,70	-3,76	-6,03	4,42	4,69
CON3-7	1010,70	601,01	1010,70	601,01	576,48	1032,02	584,05	1068,11	-2,07	-5,38	4,25	2,90
CON3-8	924,83	537,28	928,38	544,59	523,05	968,06	523,25	982,72	-4,47	-5,53	2,72	4,08
CON3-9	1001,25	588,40	1003,35	595,75	578,25	1013,33	585,58	1068,14	-1,19	-6,07	1,76	1,74
CON8-0	1569,21	899,61	1569,70	903,81	857,17	1611,49	863,98	1629,61	-2,62	-3,68	4,95	4,61
CON8-1	1372,64	781,89	1380,83	789,19	740,85	1407,56	749,24	1425,37	-2,48	-3,13	5,54	5,33
CON8-2	1318,36	718,85	1320,94	745,63	712,89	1337,33	715,54	1348,55	-1,42	-2,05	0,84	4,21
CON8-3	1433,29	820,79	1486,28	842,25	811,07	1505,30	814,33	1519,14	-4,78	-2,16	1,20	3,43
CON8-4	1412,72	787,59	1418,59	809,35	772,25	1470,93	774,87	1478,59	-3,96	-4,06	1,99	4,45
CON8-5	1386,31	762,75	1390,87	792,04	754,88	1424,24	760,66	1436,75	-2,66	-3,19	1,04	4,13
CON8-6	1250,04	709,58	1253,14	720,36	678,92	1284,28	691,37	1299,87	-2,67	-3,59	4,52	4,19
CON8-7	1489,26	816,72	1491,69	854,51	811,96	1505,51	814,10	1525,46	-1,08	-2,21	0,59	4,96
CON8-8	1412,68	797,23	1422,08	813,28	767,53	1452,21	775,06	1470,27	-2,72	-3,28	3,87	4,93
CON8-9	1340,30	812,55	1475,41	835,56	809,00	1492,53	812,39	1522,96	-10,20	-3,12	0,44	2,85
ORTALAMA									-2,43	-3,16	2,04	2,57

toplam seyahat mesafesini, bu çözüm için hesaplanan toplam yakıt tüketim maliyetini, toplam seyahat mesafesi ortalamasını ve çözümler için hesaplanan toplam yakıt tüketim maliyeti ortalamasını vermektedir. Son dört sütunda ise Y-ETDARP için ETDARP sonuçlarına göre toplam yakıt tüketim maliyeti ve toplam seyahat mesafesi için enküçük ve ortalama % sapma değerleri verilmektedir. Toplam yakıt tüketimi maliyeti (YTM) ve toplam seyahat mesafesi (TSM) için % sapma değerleri sırasıyla eşitlik 6.2 ve 6.3 kullanılarak elde edilmiştir:

$$\%YTM = \left[\frac{Z_{YT}^{Y} - Z_{YT}}{Z_{YT}}\right].100 \tag{6.2}$$

$$\%TSM = \left[\frac{Z_{TS}^{Y} - Z_{TS}}{Z_{TS}}\right].100 \tag{6.3}$$

Bu eşitliklerde:

Z_{YT}^{Y} : Y-ETDARP çözümü için toplam yakıt tüketimi
Z_{TS}^{Y} : Y-ETDARP çözümü için hesaplanan toplam seyahat mesafesi
Z_{YT} : ETDARP çözümü için hesaplanan toplam yakıt tüketimi
Z_{TS} : ETDARP çözümü için toplam seyahat mesafesi

Çizelge 6.2'deki sonuçlar incelendiğinde, Y-ETDARP'ın çözümü ile toplam yakıt tüketim maliyeti ortalama %3,16 azaltırken, toplam seyahat mesafesi ortalama %2,57 artış göstermiştir. Bunun yanısıra, Y_ETDARP ile toplam yakıt tüketim maliyetindeki iyileşme %8'e kadar yükselirken, toplam seyahat mesafesindeki artış %6 civarında kalmıştır. Y-ETDARP'da toplam yakıt tüketimi maliyeti, seyahat mesafesi ve aracın taşıdığı yük dikkate alınarak hesaplanmaktadır. Dolayısıyla Y-ETDARP, yakıt tüketim maliyetini enazlamak için araçların daha az yük ile seyahat etmesini sağlamaya çalışmaktadır. Bu nedenle, araçları talebi büyük olan müşterilere öncelikle yönlendirerek fazla yükle uzun mesafeler katetmeyi önlemektedir ki bu durum, yükün yakıt tüketim miktarında oldukça etkili olduğunu göstermektedir. Modelde birim yakıt tüketim maliyeti ve araçların yakıt tüketim oranları normalize edilerek alınmıştır. Gerçekçi veriler düşünüldüğünde yakıt tüketiminde %3,16'lık bir azalma büyük bir maliyet farkını ifade edecektir.

6.4. ETDARP için İYA_HS Algoritmasının Performans Analizi

Literatürde Y-ETDARP için yakıt tüketimi maliyetini enazlamaya hedefleyen başka bir çalışma yoktur. Bu nedenle, İYA_HS'nin performans analizi, literatürde çok sık çalışılan ETDARP üzerinde gerçekleştirilmiştir. Bu analizde, ETDARP için geliştirilmiş ve performanslarının iyi olduğu bilinen aşağıdaki algoritmalar kullanılmıştır:

- LNS: Geniş Komşu Arama Algoritması (Ropke ve Pisinger, 2006).
- ACS: Karınca Kolonisi Algoritması (Gajpal ve Abad, 2009).
- AMM: Adaptif Hafıza Metodolijisi (Zachariadis ve diğerleri, 2010).
- h_PSO: HibritHesikli Parçacık Sürüsü Optimizasyonu (Goksal ve diğerleri, 2013).
- ALS: Adaptif Yerel Arama Algoritması (Avci ve Topaloglu, 2015).

İYA_HS'nin diğer algoritmalar ile karşılaştırmalı analizinde performans ölçütleri olarak çözüm kalitesi ve çözüm zamanı dikkate alınmıştır. Karşılaştırmada kullanılan tüm algoritmalar farklı bilgisayarda çalıştırılmıştır. Bilindiği gibi bilgisayar özellikleri bir algoritmanın çözüm zamanı üzerinde çok etkildir. Bu nedenle, algoritmaları çözüm zamanı açısından aynı ortamda karşılaştırabilmek için Dongarra'nın (2006) yaklaşımı kullanılmıştır. Dongarra (2006) çalışmasında, birçok bilgisayar için *Mflops* (million floating point calculation per second) değerini vermiştir. Dolayısıyla, karşılaştırmada kullanılan algoritmaların çözüm zamanları ilgili bilgisayarların Mflops değerleri kullanılarak, İYA_HS'nin koşumlarında kullanılan Intel Core 2.6 GHz bilgisayar ortamına çevrilmiştir.

Dethloff (2001) ve Salhi ve Nagy (1999) test problemleri için karşılatırma sonuçları sırasıyla Çizelge 6.2 ve Çizelge 6.3'de verilmiştir. Çizelgelerde, test problemleri için literatürde bilinen eniyi çözümleri (BEİÇ), her bir algoritma ile bulunan eniyi çözümler (EİÇ) , çözüm süreleri (ÇS) ve EİÇ'den % sapma değerleri verilmiştir. Avci ve Topaloğlu (2015) çalışmalarında geliştirmiş oldukları ALS için çözüm sürelerini raporlamamışlardır. Bu nedenle, çizelgelerde ALS için çözüm süreleri verilmemiştir.

Çizelge 6.2. ETDARP için Dethloff (2001) test problemleri sonuçları

Test Problemleri	BEİÇ	LNS			ACS			h_PSO			AMM			ALS		İYA_HS		
		EİÇ	ÇS	% sap	EİÇ	ÇS	% sap	EİÇ	ÇS	% sap	EİÇ	ÇS	% sap	EİÇ	% sap	EİÇ	ÇS	% sap
c0530	635,62	636,1	232,0	0,076	635,62	6,0	0,000	635,62	4,9	0,000	635,62	2,5	0,000	635,62	0,000	635,62	2,8	0,000
c0531	697,84	697,8	218,0	0,000	697,84	6,0	0,000	697,84	0,8	0,000	697,84	2,5	0,000	697,84	0,000	697,84	0,9	0,000
c0532	659,34	659,3	203,0	0,000	659,34	6,0	0,000	659,34	0,4	0,000	659,34	2,9	0,000	659,34	0,000	659,34	3,6	0,000
c0533	680,04	680,6	241,0	0,082	680,04	6,1	0,000	680,04	1,0	0,000	680,04	2,3	0,000	680,04	0,000	680,04	1,0	0,000
c0534	690,50	690,5	208,0	0,000	690,50	5,7	0,000	690,50	0,3	0,000	690,50	2,9	0,000	690,50	0,000	690,50	2,3	0,000
c0535	659,90	659,9	226,0	0,000	659,90	5,1	0,000	659,90	2,0	0,000	659,90	3,0	0,000	659,90	0,000	659,90	6,1	0,000
c0536	651,09	651,1	233,0	0,000	651,09	6,1	0,000	651,09	0,8	0,000	651,09	3,1	0,000	651,09	0,000	651,09	8,6	0,000
c0537	659,17	666,1	206,0	1,051	659,17	6,8	0,000	659,17	1,6	0,000	659,17	2,8	0,000	659,17	0,000	659,17	2,5	0,000
c0538	719,47	719,5	190,0	0,000	719,47	5,4	0,000	719,47	0,5	0,000	719,47	3,5	0,000	719,47	0,000	719,48	0,4	0,001
c0539	681,00	681,0	220,0	0,000	681,00	6,0	0,000	681,00	0,8	0,000	681,00	4,7	0,000	681,00	0,000	681,00	1,0	0,000
c0580	961,50	975,1	98,0	1,414	961,50	11,0	0,000	961,50	4,8	0,000	961,50	2,7	0,000	961,50	0,000	961,50	5,0	0,000
c0581	1049,65	1052,4	95,0	0,262	1049,65	11,5	0,000	1049,65	6,8	0,000	1049,65	3,8	0,000	1049,65	0,000	1049,65	9,6	0,000
c0582	1039,64	1044,5	94,0	0,467	1042,69	11,9	0,293	1039,64	10,2	0,000	1039,64	3,9	0,000	1039,64	0,000	1042,69	2,8	0,293
c0583	983,34	999,1	94,0	1,603	983,34	11,3	0,000	983,34	13,0	0,000	983,34	2,6	0,000	983,34	0,000	983,34	10,5	0,000
c0584	1065,49	1065,5	93,0	0,000	1065,49	11,1	0,000	1065,49	3,0	0,000	1065,49	2,4	0,000	1065,49	0,000	1065,49	1,2	0,000
c0585	1027,08	1027,1	96,0	0,000	1027,08	11,3	0,000	1027,08	4,1	0,000	1027,08	3,4	0,000	1027,08	0,000	1027,08	9,2	0,000
c0586	971,82	977,0	94,0	0,533	971,82	12,0	0,000	971,82	1,6	0,000	971,82	2,7	0,000	971,82	0,000	971,82	4,7	0,000
c0587	1051,28	1061,0	92,0	0,925	1052,17	12,5	0,085	1051,28	3,4	0,000	1051,28	5,1	0,000	1051,28	0,000	1051,28	2,6	0,000
c0588	1071,18	1071,2	98,0	0,000	1071,18	11,0	0,000	1071,18	0,8	0,000	1071,18	3,6	0,000	1071,18	0,000	1071,18	4,8	0,000
c0589	1060,50	1060,5	92,0	0,000	1060,50	11,5	0,000	1060,50	7,3	0,000	1060,50	4,8	0,000	1060,50	0,000	1060,50	11,0	0,000
cc0530	616,52	616,5	215,0	0,000	616,52	8,3	0,000	616,52	2,1	0,000	616,52	4,7	0,000	616,52	0,000	616,52	4,5	0,000
cc0531	554,47	554,5	245,0	0,000	554,47	7,1	0,000	554,47	1,3	0,000	554,47	2,2	0,000	554,47	0,000	554,47	4,4	0,000

Çizelge 6.2. (devam) ETDARP için Dethloff (2001) test problemleri sonuçları

Test Problemleri	BEİÇ	LNS			ACS			h_PSO			AMM			ALS		İYA_HS		
		EİÇ	ÇS	% sap	EİÇ	ÇS	% sap	EİÇ	ÇS	% sap	EİÇ	ÇS	% sap	EİÇ	% sap	EİÇ	ÇS	% sap
cc0532	518,00	521,4	232,0	0,656	518,00	6,9	0,000	518,00	1,3	0,000	518,00	3,1	0,000	518,00	0,000	518,00	3,1	0,000
cc0533	591,19	591,2	231,0	0,000	591,19	7,2	0,000	591,19	0,5	0,000	591,19	3,2	0,000	591,19	0,000	591,19	3,4	0,000
cc0534	588,79	588,8	221,0	0,000	588,79	6,0	0,000	588,79	3,2	0,000	588,79	2,3	0,000	588,79	0,000	588,79	15,5	0,000
cc0535	563,70	563,7	209,0	0,000	563,70	6,9	0,000	563,70	0,4	0,000	563,70	3,7	0,000	563,70	0,000	563,70	0,7	0,000
cc0536	499,05	500,8	225,0	0,351	499,05	7,3	0,000	499,05	2,3	0,000	499,05	3,7	0,000	499,05	0,000	499,05	8,4	0,000
cc0537	576,48	576,5	227,0	0,000	576,48	7,0	0,000	576,48	2,6	0,000	576,48	1,9	0,000	576,48	0,000	576,48	11,1	0,000
cc0538	523,05	523,1	237,0	0,000	523,05	7,4	0,000	523,05	1,0	0,000	523,05	3,8	0,000	523,05	0,000	523,05	3,3	0,000
cc0539	578,25	586,4	207,0	1,409	578,25	6,8	0,000	578,25	2,9	0,000	578,25	2,2	0,000	578,25	0,000	578,25	6,7	0,000
cc0580	857,17	857,2	94,0	0,000	857,17	12,3	0,000	857,17	5,2	0,000	857,17	4,4	0,000	857,17	0,000	857,17	8,5	0,000
cc0581	740,85	740,9	94,0	0,000	740,85	12,0	0,000	740,85	2,9	0,000	740,85	3,3	0,000	740,85	0,000	740,85	3,4	0,000
cc0582	712,89	716,0	94,0	0,436	712,89	13,0	0,000	712,89	2,1	0,000	712,89	2,7	0,000	712,89	0,000	712,89	4,5	0,000
cc0583	811,07	811,1	98,0	0,000	811,07	13,9	0,000	811,07	2,8	0,000	811,07	2,8	0,000	811,07	0,000	811,07	5,7	0,000
cc0584	772,25	772,3	95,0	0,000	772,25	11,9	0,000	772,25	3,6	0,000	772,25	2,8	0,000	772,25	0,000	772,25	0,7	0,000
cc0585	754,88	755,7	94,0	0,109	754,88	12,4	0,000	754,88	3,4	0,000	754,88	5,7	0,000	754,88	0,000	754,88	5,9	0,000
cc0586	678,92	693,1	96,0	2,089	678,92	12,4	0,000	678,92	7,9	0,000	678,92	3,4	0,000	678,92	0,000	678,92	7,5	0,000
cc0587	811,96	814,8	94,0	0,350	811,96	13,0	0,000	811,96	3,0	0,000	811,96	2,5	0,000	811,96	0,000	811,96	8,9	0,000
cc0588	767,53	774,0	94,0	0,843	767,53	12,5	0,000	767,53	3,2	0,000	767,53	3,2	0,000	767,53	0,000	767,53	8,9	0,000
cc0589	809,00	809,3	92,0	0,037	809,00	12,9	0,000	809,00	2,4	0,000	809,00	3,8	0,000	809,00	0,000	809,00	10,0	0,000
Bulunan EİÇ sayısı		22			38			40			40			40		38		
Ort ÇS			157,9			9,3			3,1			3,3					5,4	
Ort % sap.				0,317			0,009			0,000			0,000		0,000			0,007

Çizelge 6.2'de görüldüğü gibi İYA_HS, literatürdeki güçlü metasezgisellerle yarışabilecek performans sergilemiştir. 40 test probleminden 38'inde eniyi çözüme ulaşmıştır. Eniyi çözüme ulaşılamayan 2 test probleminin % sapma değeri ise sırasıyla 0,001 ve 0,293 olmak üzere çok küçük sapmalardır. Yığın tabanlı h_PSO ve AMM ile ALS 40 test probleminde de eniyi çözüme ulaşmıştır. İYA_HS ise ortalama %0,007'lik bir sapma ile bu üç algoritmaya oldukça yakın performans göstermiştir. Çözüm zamanı açısından ise ortalama 5,4 sn ile ACS'den (ortalama 9,3 sn) daha iyi performansa sahiptir. ACS de 40 test probleminden 38'inde eniyi çözüme ulaşmıştır. Ancak, ortalama % sapma değeri 0,009 olup İYA_HS'ye göre daha büyüktür. İYA_HS algoritması, LNS algoritmasından hem çözüm zamanı (ortalama 157,9 sn) hem de çözüm kalitesi (ortalama %0,317 sapma) açısından daha iyi performansa sahiptir.

Çizelge 6.3. incelendiğinde, Salhi ve Nagy (1999) test problemlerinde çözüm zamanlarının problem boyutlarının büyük olmasından dolayı Dethloff (2001) test problemlerine göre oldukça yüksek olduğu görülmektedir. Çözüm kalitesi açısından değerlendirildiğinde eniyi algoritmaların h_PSO, AMM, ALS ve İYA_HS olduğu görülmektedir. Toplamda çalıştırılan 14 problem içerisinde h_PSO ve ALS 6 problem için bilinen eniyi çözümlere ulaşırken, AMM ve İYA_HS algoritmaları 5 problem için eniyi çözümleri elde etmiştir. Çözüm kalitesi açısından dört algoritmanında birbirine yakın performanslar sergilediği görülmektedir. Çözüm zamanları karşılaştırıldığında ise ACS'nin oldukça iyi performans gösterdiği görülmektedir, ancak algoritma sadece 2 problemde eniyi çözüme ulaşmıştır. Çözüm kalitesi açısından iyi performans göstermiş olan 4 algoritmanın çözüm zamanları incelendiğinde yine birbirlerine çok yakın olduğu görülmektedir. Sonuç olarak İYA_HS algoritması, büyük boyutlu problemler için de literatürdeki güçlü algoritmalarla rekabet edebilecek performansa sahiptir.

Çizelge 6.3. ETDARP için Salhi ve Nagy (1999) test problemleri sonuçları

Test Problemleri	BEİÇ	LNS		ACS		h_PSO		AMM		ALS	İYA_HS	
		EİÇ	ÇS	EİÇ	ÇS	EİÇ	ÇS	EİÇ	ÇS	EİÇ	EİÇ	ÇS
CMT1X	466,77	467	80,4	466,77	1,8	466,77	0,5	469,80	0,8	466,77	466,77	1,1
CMT1Y	466,77	467	85,5	466,77	1,8	466,77	0,5	469,80	1,4	466,77	466,77	2,5
CMT2X	668,77	704	106,9	684,21	7,6	684,21	12,8	684,21	2,0	684,89	684,21	14,6
CMT2Y	663,25	685	120,4	684,94	8,2	684,21	13,2	684,21	2,5	684,89	684,21	14,3
CMT3X	721,27	731	313,8	721,40	15,1	721,27	15,0	721,27	4,3	721,27	721,40	18,8
CMT3Y	721,27	738	257,5	721,40	16,1	721,27	20,0	721,27	4,0	721,27	721,40	17,3
CMT4X	852,46	879	609,5	854,12	48,3	852,83	136,9	852,46	10,7	852,46	852,46	147,9
CMT4Y	852,46	876	650,2	855,76	51,4	852,46	149,3	852,46	9,9	852,46	852,46	179,3
CMT5X	1029,25	1108	850,9	1034,87	138,4	1033,50	180,0	1030,55	22,7	1032,89	1034,87	198,2
CMT5Y	1029,25	1146	791,6	1037,34	144,3	1036,00	180,0	1030,55	17,3	1032,89	1036,00	191,2
CMT11X	833,92	837	662,2	839,66	21,0	833,92	88,2	833,92	7,7	874,85	833,92	85,9
CMT11Y	830,39	920	500,4	840,19	19,4	833,92	132,8	833,92	5,2	874,85	874,85	143,2
CMT12X	644,70	685	248,7	663,01	13,3	662,94	50,9	662,22	3,4	663,94	663,01	48,3
CMT12Y	659,52	675	196,0	663,50	14,4	663,50	37,9	662,22	1,7	663,69	663,50	42,5
Bulunan EİÇ sayısı		2		2		6		5		6	5	
Ort ÇS			391,0		35,8		72,7		6,7			78,9

6.5. İYA_HS için Duyarlılık Analizi

Son analiz, İYA_HS'de kullanılan komşuluk yapısı havuzunun, seçim fonksiyonunun ve yerel arama stratejisinin algoritmanın performansı üzerindeki etkisinin incelenmesi amacıyla yapılmıştır.

6.5.1. Komşuluk yapısı havuzu için analiz

Metasezgisel algoritmalarda farklı komşuluk yapılarının kullanılması algoritmanın performansında önemli bir etkiye sahiptir. Komşuluk yapılarının çeşitliliği mevcut çözüm üzerinde birbirinden farklı değişiklikler yaparak arama sürecinde çeşitliliği sağlar. Ancak, İYA_HS'de çok fazla komşuluk yapısının kullanılması ve komşuluk yapıları ile ilgili geçmişe dayalı bilginin tutulması ve sürekli güncellenmesi, algoritmanın çözüm zamanını artırabilmektedir. Aynı şekilde komşuluk yapısı sayısının az olması da algoritmanın arama yeteneğinin zayıflamasına ve iyi çözümlere ulaşmak için daha uzun süre çalışmasına sebep olabilmektedir. Bu nedenle, komşuluk yapısı havuzunun iyi bir şekilde belirlenmesi algoritmanın performansı ve çözüm süresi açısından oldukça önemlidir. İYA_HS'de komşuluk yapısı havuzunun etkinliğinin incelenebilmesi için yerel arama algoritmasında (yani YAA1_HS) kullanılan rotalararası komşuluk yapılarından aşağıda görüldüğü gibi ilk 4 komşuluk yapısı dikkate alınmış ve İHA_HS-1 algoritması elde edilmiştir.

İYA_HS_1'de yerel aramada kullanılan rotalararası komşuluk yapıları:

- Rotalar arası yer değiştirme (1,1)
- Rotalar arası ekleme (1,0)
- Rotalar arası yer değiştirme (2,1)
- Rotalar arası ekleme (2,0)

Çizelge 6.4' de, Dethloff (2001) test problemleri için İYA_HS ve İYA_HS-1'nin sonuçları verilmektedir. Çizelgede ilk sütun ilgili test problemini, ikinci sütun problemin literatürde bilinen eniyi çözümünü (BEİÇ), diğer sütunlar ise algoritmaların ulaşmış oldukları eniyi çözüm değerini (EİÇ) ve bu çözüme ulaştıkları çözüm zamanını (en iyi ÇS), 30 koşum için

Çizelge 6.4. Dethloff (2001) test problemleri için İYA_HS ve İYA_HS-1 sonuçları

Test Problemi	BEİÇ	İYA_HS					İYS_HS-1				
		EİÇ	en iyi ÇS	ort	%sap	ort ÇS	EİÇ	en iyi ÇS	ort	%sap	ort ÇS
SCA3-0	635,62	635,62	2,8	639,61	0,000	6,4	635,62	14,8	640,76	0,000	8,1
SCA3-1	697,84	697,84	0,9	700,83	0,000	9,1	697,84	1,7	706,94	0,000	9,9
SCA3-2	659,34	659,34	3,6	659,62	0,000	14,1	659,34	5,0	663,92	0,000	13,0
SCA3-3	680,04	680,04	1,0	682,19	0,000	9,0	680,04	3,9	682,59	0,000	8,5
SCA3-4	690,50	690,50	2,3	690,91	0,000	9,5	690,50	2,8	693,49	0,000	10,1
SCA3-5	659,90	659,90	6,1	663,01	0,000	11,9	659,90	3,1	676,73	0,000	9,4
SCA3-6	651,09	651,09	8,6	654,21	0,000	13,3	651,09	3,4	655,20	0,000	13,5
SCA3-7	659,17	659,17	2,5	667,17	0,000	6,7	659,17	2,2	670,36	0,000	8,1
SCA3-8	719,47	719,48	0,4	720,81	0,001	9,9	719,48	1,0	723,68	0,001	6,6
SCA3-9	681,00	681,00	1,0	681,55	0,000	8,8	681,00	1,8	685,97	0,000	8,3
SCA8-0	961,50	961,50	5,0	974,07	0,000	13,3	961,50	10,3	983,58	0,000	15,0
SCA8-1	1049,65	1049,65	9,6	1065,04	0,000	17,0	1049,65	32,7	1076,33	0,000	15,9
SCA8-2	1039,64	1049,08	2,8	1051,25	0,908	13,8	1049,08	0,6	1051,61	0,908	8,7
SCA8-3	983,34	983,34	10,5	1001,34	0,000	18,4	983,34	9,4	1009,43	0,000	13,9
SCA8-4	1065,49	1065,49	1,2	1072,24	0,000	14,6	1065,49	8,3	1077,33	0,000	11,8
SCA8-5	1027,09	1027,09	9,2	1042,33	0,000	18,8	1029,95	10,3	1052,24	0,278	16,0
SCA8-6	971,82	971,82	4,7	973,54	0,000	14,3	971,82	2,7	977,33	0,000	15,9
SCA8-7	1051,28	1051,28	2,6	1065,44	0,000	18,7	1058,90	36,9	1071,62	0,725	16,1
SCA8-8	1071,18	1071,18	4,8	1081,60	0,000	13,1	1071,18	2,2	1084,76	0,000	11,8
SCA8-9	1060,50	1060,50	11,0	1069,03	0,000	16,6	1060,50	7,2	1074,27	0,000	13,6
CON3-0	616,52	616,52	4,5	619,78	0,000	13,3	616,52	5,4	620,94	0,000	15,9
CON3-1	554,47	554,47	4,4	557,54	0,000	12,9	554,47	6,4	561,10	0,000	13,0
CON3-2	518,00	518,00	3,1	521,75	0,000	8,9	521,38	1,0	522,90	0,653	9,6
CON3-3	591,19	591,19	3,4	591,33	0,000	11,2	591,19	2,4	591,85	0,000	11,8
CON3-4	588,79	588,79	15,5	592,30	0,000	11,5	588,79	10,9	596,66	0,000	11,6
CON3-5	563,70	563,70	0,7	564,36	0,000	14,7	563,70	3,5	569,97	0,000	9,0
CON3-6	499,05	499,05	8,4	502,59	0,000	10,8	499,05	17,0	505,14	0,000	11,5
CON3-7	576,48	576,48	11,1	585,11	0,000	7,8	576,48	9,4	588,92	0,000	11,3
CON3-8	523,05	523,05	3,3	523,25	0,000	15,5	523,05	3,8	523,72	0,000	13,4
CON3-9	578,25	578,25	6,7	585,68	0,000	11,9	578,25	9,0	586,40	0,000	9,9
CON8-0	857,17	857,17	8,5	867,48	0,000	17,6	857,17	24,5	871,24	0,000	20,9
CON8-1	740,85	740,85	3,4	748,04	0,000	17,0	740,85	31,1	754,45	0,000	15,8
CON8-2	712,89	712,89	4,5	714,93	0,000	14,6	712,89	18,6	718,56	0,000	15,3
CON8-3	811,07	811,07	5,7	819,57	0,000	15,7	811,07	12,4	821,74	0,000	19,1
CON8-4	772,25	772,25	0,7	775,35	0,000	17,2	772,25	6,8	779,54	0,000	15,6
CON8-5	754,88	754,88	5,9	759,89	0,000	16,7	754,88	39,3	764,90	0,000	16,8
CON8-6	678,92	678,92	7,5	688,86	0,000	21,2	678,92	13,6	692,25	0,000	18,8
CON8-7	811,96	811,96	8,9	814,31	0,000	16,8	811,96	15,4	818,09	0,000	16,1
CON8-8	767,53	767,53	8,9	777,00	0,000	16,0	767,53	13,5	780,30	0,000	14,5
CON8-9	809,00	809,00	10,0	812,74	0,000	16,5	809,00	6,8	814,27	0,000	17,5
Ort. % Sap.					0,02					0,06	
Bulunan EİÇ sayısı		38					35				
Ort en iyi ÇS (sn)			5,4					10,3			
Ort ort ÇS (sn)						13,6					13,0

ortalama çözüm değerini (ort), eniyi çözümden % sapma değerini (% sap) ve 30 koşumda ortalama çözüm zamanını (ort ÇS) göstermektedir.

Çizelge 6.4. incelendiğinde, komşuluk yapısı çeşitliliği daha fazla olan İYA_HS'nın hem çözüm kalitesi hem de çözüm zamanı açısından çok daha iyi performans gösterdiği görülmektedir. İYA_HS-1 ile 35 test problemi için BEİÇ'ye ulaşılmışken, İYA_HS 38 test probleminde BEİÇ'ye ulaşmıştır. Aynı şekilde, İYA_HS-1 için ortalama sapma % 0,06 iken bu değer İYA_HS için %0,02'ye düşmüştür. Çözüm zamanları incelendiğinde ise İYA_HS'de kullanılan komşuluk yapıları ile çözüm zamanı 5,4 sn iken bu değerin İYA_HS-1 için 10,3 sn'ye yükseldiği görülmektedir. Sonuç olarak, kullanılan komşuluk yapısı arttıkça algoritmanın daha fazla bilgi tutması söz konusu olduğu için çözüm zamanının artması beklenirken, bu komşuluk yapıları algoritmanın arama performansını iyileştirmiş dolayısıyla çözüm zamanlarında azalma olmuştur.

6.5.2. Komşuluk seçim fonksiyonu için analiz

İYA_HS'de hem rotalararası iyileştirme (YAA1_HS) hem de rotaiçi iyileştirme (YAA2_HS) aşamasında kullanılacak komşuluk yapısı, bir seçim fonksiyonu ile belirlenmektedir. Bu seçim fonksiyonu, öncelikle geçmiş performanslarına ve kullanılma sıklıklarına göre komşuluk yapılarına puan ataması yapmakta, sonra da puanlara göre rassal olarak bir komşuluk yapısını seçmektedir. İYA_HS'nin performansı üzerinde seçim fonksiyonunun etkisini incelemek amacıyla YAA1_HS ve YAA2_HS'de kullanılan mevcut seçim fonksiyonu değiştirilmiştir. Yeni seçim fonksiyonu geçmiş bilgiyi kulanmaksızın her bir komşuluk yapısını eşit olasılıkla seçmektedir. Yeni seçim mekanizmasını yerel aramada kullanan algoritma, İYA_HS-2 olarak adlandırılmıştır. Çizelge 6.5' de, Dethloff (2001) test problemleri için İYA_HS ve İYA_HS-2'nin sonuçları verilmektedir.

Çizelge 6.5 incelendiğinde, çözüm zamanı İYA_HS için 5,4 sn iken İYA_HS-2 için 6,5 sn olduğu görülmektedir. Ayrıca, İYA_HS 38 problemde bilinen eniyi çözüme ulaşırken, İYA_HS-2 ise 34 problemde ulaşabilmiştir. İYA_HS ve İYA_HS-2 için ortalama % sapma değerleri ise sırasıyla %0,02 ve %0,05'dir. Dolayısıyla bu sonuçlar, komşuluk yapıları ile ilgili geçmişe yönelik bilgi tutulmadığında algoritmanın performansını hem çözüm kalitesi hem de çözüm zamanı açısından kötüleştiğini göstermektedir.

Çizelge 6.5. Dethloff (2001) test problemleri için İYA_HS ve İYA_HS-2 sonuçları

Test Problemi	BEİÇ	İYA_HS					İYS_HS-2				
		EİÇ	en iyi ÇS	ort	%sap	ort ÇS	EİÇ	en iyi ÇS	ort	%sap	ort ÇS
SCA3-0	635,62	635,62	2,8	639,61	0,000	6,4	635,62	5,6	640,12	0,000	7,4
SCA3-1	697,84	697,84	0,9	700,83	0,000	9,1	697,84	1,7	701,99	0,000	7,4
SCA3-2	659,34	659,34	3,6	659,62	0,000	14,1	659,34	2,8	661,72	0,000	12,4
SCA3-3	680,04	680,04	1,0	682,19	0,000	9,0	680,04	8,1	681,49	0,000	15,2
SCA3-4	690,50	690,50	2,3	690,91	0,000	9,5	690,50	1,7	691,59	0,000	9,2
SCA3-5	659,90	659,90	6,1	663,01	0,000	11,9	659,90	2,0	663,92	0,000	12,4
SCA3-6	651,09	651,09	8,6	654,21	0,000	13,3	651,09	1,6	654,36	0,000	12,2
SCA3-7	659,17	659,17	2,5	667,17	0,000	6,7	659,17	7,6	669,81	0,000	6,6
SCA3-8	719,47	719,48	0,4	720,81	0,001	9,9	719,48	3,3	721,51	0,001	10,3
SCA3-9	681,00	681,00	1,0	681,55	0,000	8,8	681,00	3,4	684,06	0,000	8,1
SCA8-0	961,50	961,50	5,0	974,07	0,000	13,3	961,50	14,5	976,22	0,000	11,1
SCA8-1	1049,65	1049,65	9,6	1065,04	0,000	17,0	1049,65	10,4	1066,49	0,000	16,5
SCA8-2	1039,64	1049,08	2,8	1051,25	0,908	13,8	1044,48	1,0	1051,11	0,466	15,3
SCA8-3	983,34	983,34	10,5	1001,34	0,000	18,4	983,34	8,5	1006,66	0,000	9,0
SCA8-4	1065,49	1065,49	1,2	1072,24	0,000	14,6	1065,49	8,8	1072,63	0,000	16,1
SCA8-5	1027,09	1027,09	9,2	1042,33	0,000	18,8	1029,95	8,0	1043,40	0,278	19,8
SCA8-6	971,82	971,82	4,7	973,54	0,000	14,3	971,82	10,0	976,57	0,000	16,1
SCA8-7	1051,28	1051,28	2,6	1065,44	0,000	18,7	1056,39	11,7	1068,05	0,486	15,8
SCA8-8	1071,18	1071,18	4,8	1081,60	0,000	13,1	1071,18	6,3	1084,11	0,000	8,5
SCA8-9	1060,50	1060,50	11,0	1069,03	0,000	16,6	1061,23	7,6	1068,39	0,069	14,0
CON3-0	616,52	616,52	4,5	619,78	0,000	13,3	616,52	1,7	620,82	0,000	10,5
CON3-1	554,47	554,47	4,4	557,54	0,000	12,9	554,47	14,1	558,32	0,000	14,0
CON3-2	518,00	518,00	3,1	521,75	0,000	8,9	521,38	1,9	522,41	0,653	12,3
CON3-3	591,19	591,19	3,4	591,33	0,000	11,2	591,19	4,7	591,20	0,000	11,6
CON3-4	588,79	588,79	15,5	592,30	0,000	11,5	588,79	8,1	591,65	0,000	11,4
CON3-5	563,70	563,70	0,7	564,36	0,000	14,7	563,70	3,3	565,88	0,000	16,6
CON3-6	499,05	499,05	8,4	502,59	0,000	10,8	499,05	2,8	502,89	0,000	10,4
CON3-7	576,48	576,48	11,1	585,11	0,000	7,8	576,48	11,4	583,86	0,000	14,1
CON3-8	523,05	523,05	3,3	523,25	0,000	15,5	523,05	3,1	523,32	0,000	14,0
CON3-9	578,25	578,25	6,7	585,68	0,000	11,9	578,25	6,7	586,41	0,000	14,2
CON8-0	857,17	857,17	8,5	867,48	0,000	17,6	857,17	9,5	867,75	0,000	13,0
CON8-1	740,85	740,85	3,4	748,04	0,000	17,0	740,85	4,9	752,83	0,000	17,9
CON8-2	712,89	712,89	4,5	714,93	0,000	14,6	712,89	5,3	714,21	0,000	16,0
CON8-3	811,07	811,07	5,7	819,57	0,000	15,7	811,07	5,4	817,44	0,000	14,7
CON8-4	772,25	772,25	0,7	775,35	0,000	17,2	772,25	7,7	778,31	0,000	9,0
CON8-5	754,88	754,88	5,9	759,89	0,000	16,7	754,88	7,3	761,26	0,000	18,2
CON8-6	678,92	678,92	7,5	688,86	0,000	21,2	678,92	13,8	690,68	0,000	12,9
CON8-7	811,96	811,96	8,9	814,31	0,000	16,8	811,96	8,1	815,25	0,000	17,9
CON8-8	767,53	767,53	8,9	777,00	0,000	16,0	767,53	11,2	776,39	0,000	8,6
CON8-9	809,00	809,00	10,0	812,74	0,000	16,5	809,00	5,9	812,26	0,000	16,9
Ort. % Sap. Bulunan EİÇ sayısı		38			0,02		34			0,05	
Ort en iyi ÇS (sn)			5,4					6,5			
Ort ort ÇS (sn)						13,6					12,9

6.5.3. Yerel arama algoritması için analiz

İYA_HS'nin yerel arama aşamasında, bir komşuluk yapısı bir havuzdan seçim fonksiyonu ile seçilmektedir. Seçilen komşuluk yapısı ile mevcut çözümde iyileşme sağlanmaz ise bu komşuluk yapısı havuzdan çıkarılır, iyileşme sağlanır ise havuz dışındaki tüm komşuluk yapıları havuza tekrar eklenir. Dolayısıyla, yerel arama algoritması ile arama işlemi havuz boşalıncaya kadar devam etmektedir. Bu stratejinin İYA_HS'nin performansı üzerindeki etkisini incelemek amacıyla, YAA2_HS'deki rotalariçi komşuluk yapılarının belirli bir sıraya göre kullanıldığı yeni bir YAA-3 algoritması tasarlanmıştır. Şekil 6.1'de YAA-3'ün adımları verilmektedir. YAA-3'ü kullanan algoritma ise İYA_HS-3 olarak adlandırılmıştır. Çizelge 6.6' da, Dethloff (2001) test problemleri için İYA_HS ve İYA_HS-3'ün sonuçları verilmektedir.

Çizelge 6.6 incelendiğinde, İYA_HS'nin tüm performans ölçütlerine göre daha iyi bir performansa sahip olduğu görülmektedir. YAA2_HS'de rotaiçi yerel arama stratejisi algoritmanın performansı üzerinde oldukça etkilidir. Bu strateji, rotalararası komşuluk yapıları ile iyileştirilmiş olan çözümler üzerinde derinlemesine aramanın gerçekleştirilmesini sağlamaktadır. İYA_HS, 38 problemde eniyi çözüme ulaşırken İYA_HS-3 ile 29 problemde eniyi çözüme ulaşılmıştır ve ortalama sapma değeri bu algoritma için %0,16'ya çıkmıştır. Aynı şeklide, İYA_HS ortlama 5,4 sn'de çözümlere ulaşırken IYA_HS-3 ortalama 17,1 sn harcamaktadır. Sonuç olarak, YAA2_HS'de kullanılan komşuluk seçim stratejisinin İYA_HS'nin iyi performans göstermesinde büyük etkiye sahip olduğu söylenebilir.

$N_{ri} = \{h_1, h_2, \dots, h_N\}$: Rota ici komşuluk yapıları kümesi
s'': YAA2 − 3 sonucu elde edilen çözüm
h_1: twoopt operatörü
h_2: rotaiçi yerdeğiştirme operatörü
h_3: rotaiçi ekleme operatörü

begin

s'' çözümünde değiştirilen rotaları (r_1 ve r_2) seç
r_1 ve r_2 rotaları için aşağıdaki adımları tekrarla

for $i = 1$ to 2 **do**

$s \leftarrow s''$
$r \leftarrow r_i$ $\quad$ $i = 1$ ise r_1 rotası, $\quad$ $i = 2$ ise r_2 rotası
$i \leftarrow 1$

while $(j \leq 3)$ **do**

$ei \leftarrow h_j$ komşuluk yapısına göre r rotası için mümkün tüm gerçekleştir ve en iyi hareketi seç

$s'' \leftarrow s$ çözümüne ei hareketini uygula

if $f(s'') \leq f(s)$ **then**

$s \leftarrow s''$ ve $i \leftarrow 1$

else

$i \leftarrow i + 1$

end while

$s'' \leftarrow s$

end for

end

Şekil 6.1. YAA-3 adımları

Çizelge 6.6. Dethloff (2001) test problemleri için İYA_HS ve İYA_HS-3 sonuçları

Test Problemi	BEİÇ	İYA_HS					İYS_HS-3				
		EİÇ	en iyi ÇS	ort	%sap	ort ÇS	EİÇ	en iyi ÇS	ort	%sap	ort ÇS
SCA3-0	635,62	635,62	2,8	639,61	0,000	6,4	640,55	14,1	654,01	0,775	18,1
SCA3-1	697,84	697,84	0,9	700,83	0,000	9,1	697,84	9,1	718,93	0,000	20,4
SCA3-2	659,34	659,34	3,6	659,62	0,000	14,1	659,34	20,7	683,97	0,000	19,9
SCA3-3	680,04	680,04	1,0	682,19	0,000	9,0	681,16	12,0	707,18	0,165	83,4
SCA3-4	690,50	690,50	2,3	690,91	0,000	9,5	690,50	18,1	704,73	0,000	20,5
SCA3-5	659,90	659,90	6,1	663,01	0,000	11,9	659,90	20,4	693,25	0,000	15,4
SCA3-6	651,09	651,09	8,6	654,21	0,000	13,3	651,09	14,9	672,61	0,000	12,7
SCA3-7	659,17	659,17	2,5	667,17	0,000	6,7	659,17	9,3	684,64	0,000	21,6
SCA3-8	719,47	719,48	0,4	720,81	0,001	9,9	719,48	28,2	744,99	0,001	19,9
SCA3-9	681,00	681,00	1,0	681,55	0,000	8,8	681,00	16,7	697,59	0,000	16,3
SCA8-0	961,50	961,50	5,0	974,07	0,000	13,3	961,50	32,2	992,77	0,000	22,8
SCA8-1	1049,65	1049,65	9,6	1065,04	0,000	17,0	1049,65	10,4	1083,41	0,000	17,4
SCA8-2	1039,64	1049,08	2,8	1051,25	0,908	13,8	1044,48	23,9	1048,55	0,466	19,7
SCA8-3	983,34	983,34	10,5	1001,34	0,000	18,4	983,34	23,2	1033,13	0,000	14,7
SCA8-4	1065,49	1065,49	1,2	1072,24	0,000	14,6	1065,49	7,8	1092,09	0,000	13,0
SCA8-5	1027,09	1027,09	9,2	1042,33	0,000	18,8	1027,09	21,2	1057,72	0,000	14,2
SCA8-6	971,82	971,82	4,7	973,54	0,000	14,3	971,82	7,5	996,62	0,000	20,1
SCA8-7	1051,28	1051,28	2,6	1065,44	0,000	18,7	1063,33	29,9	1079,46	1,146	17,9
SCA8-8	1071,18	1071,18	4,8	1081,60	0,000	13,1	1071,18	21,1	1096,35	0,000	17,7
SCA8-9	1060,50	1060,50	11,0	1069,03	0,000	16,6	1064,40	22,8	1091,25	0,368	13,0
CON3-0	616,52	616,52	4,5	619,78	0,000	13,3	616,52	34,0	632,73	0,000	16,7
CON3-1	554,47	554,47	4,4	557,54	0,000	12,9	554,47	8,1	575,52	0,000	17,6
CON3-2	518,00	518,00	3,1	521,75	0,000	8,9	521,38	21,3	531,46	0,653	24,7
CON3-3	591,19	591,19	3,4	591,33	0,000	11,2	591,19	6,1	605,80	0,000	10,8
CON3-4	588,79	588,79	15,5	592,30	0,000	11,5	588,79	11,8	603,64	0,000	12,7
CON3-5	563,70	563,70	0,7	564,36	0,000	14,7	563,70	38,5	577,63	0,000	26,3
CON3-6	499,05	499,05	8,4	502,59	0,000	10,8	502,29	25,3	511,95	0,649	26,0
CON3-7	576,48	576,48	11,1	585,11	0,000	7,8	576,48	44,1	598,61	0,000	24,5
CON3-8	523,05	523,05	3,3	523,25	0,000	15,5	523,05	7,3	534,29	0,000	16,3
CON3-9	578,25	578,25	6,7	585,68	0,000	11,9	578,25	17,9	597,31	0,000	14,9
CON8-0	857,17	857,17	8,5	867,48	0,000	17,6	857,17	2,8	889,15	0,000	19,3
CON8-1	740,85	740,85	3,4	748,04	0,000	17,0	740,85	11,8	774,91	0,000	16,5
CON8-2	712,89	712,89	4,5	714,93	0,000	14,6	712,89	8,8	727,05	0,000	16,4
CON8-3	811,07	811,07	5,7	819,57	0,000	15,7	813,68	7,6	832,92	0,322	24,7
CON8-4	772,25	772,25	0,7	775,35	0,000	17,2	772,25	15,0	792,26	0,000	22,5
CON8-5	754,88	754,88	5,9	759,89	0,000	16,7	761,48	14,0	778,46	0,874	17,9
CON8-6	678,92	678,92	7,5	688,86	0,000	21,2	686,14	8,1	701,18	1,063	13,2
CON8-7	811,96	811,96	8,9	814,31	0,000	16,8	811,96	21,3	825,74	0,000	19,7
CON8-8	767,53	767,53	8,9	777,00	0,000	16,0	767,53	6,1	792,52	0,000	14,0
CON8-9	809,00	809,00	10,0	812,74	0,000	16,5	809,00	11,8	825,11	0,000	23,8
Ort. % Sap. Bulunan EİÇ sayısı		38			0,02		29			0,16	
Ort en iyi ÇS (sn)			5,4					17,1			
Ort ort ÇS (sn)						13,6					19,9

7. SONUÇ VE DEĞERLENDİRME

Belirli kapasitede birden fazla aracın, bir grup müşterinin talebini karşılamak amacıyla rotalanması problemi araç rotalama problemi (ARP) olarak adlandırılır. Günümüzde lojistik ağın büyümesi ve büyük miktarlarda ürünlerin uzun mesafelere taşınması ihtiyacı, ARP'ye olan ilgiyi artırmıştır. Müşterilerin, kullanılan araçların ve taleplerin tipine, depo sayısına vb. diğer kriterlere göre literatürde ARP'nin farklı türleri tanımlanmış ve çözüm yaklaşımları önerilmiştir. Bu türlerden birisi de, müşterilere ürün dağıtımının yanısıra müşterilerden ürünlerin toplandığı topla-dağıt ARP (TDARP) olarak bilinir. TDARP'da müşterilerin bir kısmının dağıtım talebi varken diğer kısmının toplama talebi olabilir. Önce dağıtım sonra da toplama taleplerinin karşılandığı durumda problem önce dağıt-sonra topla TDARP (ÖDSTDARP) adlandırılırken, toplama ve dağıtım taleplerinin karışık sırada karşılandığı durumda problem karma TDARP (KTDARP) olarak adlandırılır. Müşteriler hem dağıtım hem de toplama talebine sahipse ve bu taleplerin aynı anda karşılanması söz konusu a ise problem, eşzamanlı TDARP (ETDARP) olarak adlandırılır. Bu tez kapsamında ETDARP ele alınmıştır.

Günümüzde çevreye olan duyarlılığın artması, hükümetlerin bu konudaki yasal kısıtlamaları işletmeleri de operasyonlarını gerçekleştirirken çevresel etkilerini göz önüne almak zorunda bırakmıştır. Özellikle lojistik faaliyetlerinde kullanılan araçların çevreye olan zararlı gaz salınımlarının fazla olması, lojistik işletmeleribu konuda daha da duyarlı olmaya zorlamıştır. Bu durum literatürede yansımış ve ARP'de çevresel etkileri göz önüne alan çalışmalar yapılmaya başlanmıştır. Araçların yakıt tüketimini veya zararlı gaz salınımlarını enazlayarak rotalamayı hedefleyenARP, literatürde yeşil ARP (Y-ARP) olarak adlandırılmaktadır. Bu tezde, ETDARP'de çevreye olan zararı azaltmak amacıyla araç yüklerine bağlı yakıt tüketim maliyetini enazlayacak şekilde araç rotalarını oluşturmayı hedefleyenyeşil ETDARP (Y-ETDARP) ele alınmıştır. Problemin çözümü için, literatürden yararlanılarak öncelikle bir matematiksel model geliştirilmiştir. Y-ETDARP, NP-zor problemler sınıfında yer aldığı için çözüm zamanı problem boyutuna bağlı olarak üstel artış göstermektedir. Bu nedenle, orta ve büyük boyutlu Y-ETDARP'ın çözümü içinmetasezgisel algoritmalara dayalı bir algoritma geliştirilmiştir.

Metasezgisel algoritmalar, NP-zor problemler için güçlü arama stratejileri ile makul sürelerdeeniyi çözüme yakın çözümler bulabilen sezgisel yöntemlerdir. Son yıllarda, metasezgisel algoritmaların performansını artırmak amacıyla iki ya da daha fazla metasezgiselin güçlü yönlerini birleştiren algoritmalar geliştirilmeye başlanmış ve bu algoritmalar karma metasezgisel algoritmalar olarak adlandırılmıştır. Metasezgisel algoritmalar, çözüm uzayında arama yapmak için komşuluk yapılarını kullanırlar ve bu komşuluk yapıları algoritmanın performansı üzerinde büyük etkiye sahiptir. Son yıllarda, komşuluk yapılarının daha akıllı bir şekilde kullanarak algoritmanın performansını arttırmayı amaçlayan hipersezgisel algoritmalar geliştirilmeye başlanmıştır. Hipersezgisel algoritmalar, aramanın her iterasyonunda kullanılacak olan komşuluk yapısınıarama sırasında elde edilen bilgiyi kullanan bir seçim stratejisi ile seçer. Bu tür seçim stratejisinin kullanılmasındaki amaç, aramayı daha iyi yönlendirerek kaliteli çözümlere daha kısa sürelerde ulaşmaktır. Bu tezde, Y-ETDARP'ınçözümü için iteratif yerel arama ve değişken komşu arama algoritmalarının güçlü yönlerini birleştiren ve arama sırasında komşuluk yapılarının seçiminde hipersezgisel algoritmanın özelliğini kullanan bir karma sezgisel algoritma geliştirilmiştir. Bu algoritma, İYA_HS olarak adlandırılmıştır.

Tez kapsamında üç aşamalı deneysel analiz yapılmıştır. Birinci aşamada, Y-ETDARP'da kullanılan amaç fonksiyonunun yakıt tüketim maliyetine etkisi analiz edilmiştir. Bu amaçla, İYA_HS ile elde edilen Y-ETDARP ve ETDARP çözümleri karşılaştırılmıştır. Bu karşılaştırma sonucunda toplam seyahat mesafesinde ortalama % 2,57 artışa karşın, toplam yakıt tüketim maliyetinde ortalama %3,16 azalma sağlandığı görülmüştür ki bunun sonucunda çevreye salınan sera gazlarının miktarında da azalma olacaktır. İkinci aşamada, İYA_HS'nin performansı literatürde ETDARP için geliştirilmiş olan diğer sezgisel algoritmalar ile karşılaştırmalı incelenmiş ve çözüm kalitesi ve çözüm zamanı açısından bu algoritmalar ile benzer performans gösterdiği görülmüştür. Son olarak, İYA_HS için duyarlılık analizi yapılarak komşuluk yapılarının, hipersezgisel yaklaşımın ve yerel arama algoritmasının İYA_HS'nin performansı üzerindeki etkileri incelenmiştir.

Bu tez kapsamında gerçekleştirilen çalışmalar göz önünde bulundurularak ileride yapılabilecek çalışmalar aşağıda özetlenmektedir:

- Y-ETDARP'da sadece araç kapasiteleri dikkate alınmıştır. Zaman penceresi, heterojen araç ve mesafe kısıtı gibi farklı operasyonel kısıtlar dikkate alınarak Y-ETDARP'ın farkli türleri üzerine çalışma yapılabilir.
- Yeşil lojistik kapsamında ETDARP'da araç yüküne bağlı yakıt tüketimi maliyetinin en azlanması ele alınmıştır. Araç yükü dışında yakıt tüketimi maliyetini etkileyen hız, yolun eğimi gibi başka kriterlerde dikkate alınarak ETDARP için daha gerçekçi yakıt tüketim maliyeti modeli oluşturulabilir.

- Son yıllarda literatürde çok amaçlı problemler üzerine çalışmalar yapıldığı görülmektedir. Dolayısıyla, toplam yakıt tüketim maliyeti, toplam seyahat mesafesi, araç yükü dengesi vb amaçlar birlikte ele alınarak çok amaçlı Y-ETDARP ele alınabilir ve yeni çözüm yaklaşımları geliştirilebilir ya da İYA_HS bu problem için uyarlanabilir.

- Y-ETDARP için matematiksel modelin ve metasezgisellerin birlikte kullanıldığı matematiksel sezgisel algoritmalar geliştirilebilir.

- İYA_HS'nin performansı diğer ARP türleri için incelenebilir.

KAYNAKLAR

Ai, T. J. and Kachitvichyanukul, V. (2009). A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery. *Computers & Operations Research*, 36(5), 1693–1702.

Alfredo Tang Montané, F. and Galvão, R. D. (2006). A tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service. *Computers & Operations Research*, 33(3), 595–619.

Alinaghian, M. and Naderipour, M. (2016). A novel comprehensive macroscopic model for time-dependent vehicle routing problem with multi-alternative graph to reduce fuel consumption: A case study. *Computers & Industrial Engineering*, 99, 210–222.

Avci, M. and Topaloglu, S. (2015). An adaptive local search algorithm for vehicle routing problem with simultaneous and mixed pickups and deliveries. *Computers & Industrial Engineering*, 83, 15–29.

Bektaş, T. and Laporte, G. (2011). The Pollution-Routing Problem. *Transportation Research Part B: Methodological*, 45(8), 1232–1250.

Belbağ, S. (2017). Yeşil kapasite kısıtlı araç rotalama problemi: Bir literatür taraması. *Gazi Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 19(1), 345-366.

Belgin, Ö., Karaoglan, I. and Altiparmak, F. (2018). Two-echelon vehicle routing problem with simultaneous pickup and delivery: Mathematical model and heuristic approach. *Computers & Industrial Engineering*, 115, 1–16.

Berbeglia, G., Cordeau, J.-F., Gribkovskaia, I. and Laporte, G. (2007). Static pickup and delivery problems: a classification scheme and survey. *TOP*, 15(1), 1–31.

Braekers, K., Ramaekers, K. and Van Nieuwenhuyse, I. (2016). The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering*, 99, 300–313.

Chand, P. and Mohanty, J. R. (2013). Multi objective genetic approach for solving vehicle routing problem. *International Journal of Computer Theory and Engineering*, 5(6), 846-849.

Chen, J. F. and Wu, T. H. (2006). Vehicle routing problem with simultaneous deliveries and pickups. *Journal of the Operational Research Society*, 57(5), 579–587.

Cinar, D., Gakis, K. and Pardalos, P. M. (2014). Reduction of CO2 emissions in cumulative multi-trip vehicle routing problems with limited duration. *Environmental Modeling & Assessment*, 20(4), 273–284.

Clarke, G. and Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*,12(4), 568-581.

Cowling, P., Kendall, G. and Soubeiga, E. (2000). A hyperheuristic approach for scheduling a sales summit. *Third International Conference on the Practice And Theory of Automated Timetabling*, 176–190.

Crispim, J. A. and Brandao, J. (2005). Metaheuristics applied to mixed and simultaneous extensions of vehicle routing problems with backhauls. *Journal of the Operational Research Society,* 56(11), 1296-1302.

Çatay, B. (2010). A new saving-based ant algorithm for the vehicle routing problem with simultaneous pickup and delivery. *Expert Systems with Applications*, 37(10), 6809–6817.

Çimen, M. and Soysal, M. (2017b). Time-dependent green vehicle routing problem with stochastic vehicle speeds: An approximate dynamic programming algorithm. *Transportation Research Part D: Transport and Environment*, 54, 82–98.

Dantzig, G. B. and Ramser J. H. (1959). The truck dispatching problem. *Institute for Operations Research and the Management Sciences,* 6(1), 80-91.

Dell'Amico, M., Righini, G. and Salani, M. (2006). A branch-and-price approach to the vehicle routing problem with simultaneous distribution and collection. *Transportation Science,* 40(2), 235-247.

Demir, E., Bektaş, T. and Laporte, G. (2012). An adaptive large neighborhood search heuristic for the pollution-routing problem. *European Journal of Operational Research*, 223(2), 346–359.

Demir, E., Bektaş, T. and Laporte, G. (2014a). The bi-objective pollution-routing problem. *European Journal of Operational Research*, 232(3), 464–478.

Demir, E., Bektaş, T., and Laporte, G. (2014b). A review of recent research on green road freight transportation. *European Journal of Operational Research*, 237(3), 775-793.

Dethloff, J. (2001). Vehicle routing and reverse logistics: The vehicle routing problem with simultaneous delivery and pick-up. *OR-Spektrum,* 23(1), 79-96.

Dişkaya, F. ve Dinçer, S. E. (2018). Yeşil lojistik yönetiminde araç rotalama optimizasyonu için bir model önerisi. *Beykoz Akademi Dergisi,* 6(1), 29-46.

Dongarra, J. J. (1992). Performance of various computers using standard linear equations software. *Association for Computing Machinery Special Interest Committee on Computer Architecture News,* 20(3), 22–44.

Eksioglu, B., Vural, A. V. and Reisman, A. (2009). The vehicle routing problem: A taxonomic review. *Computers & Industrial Engineering*, 57(4), 1472–1483.

Elbasan, S. (2015). *Karbon ayak izini dikkate alan eşzamanlı topla-dağıt araç rotalama,* Yüksek Lisans Tezi, Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü, Eskişehir, 32-33.

Erdoğan, S. and Miller-Hooks, E. (2012). A green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 48(1), 100–114.

Figliozzi, M. (2010). Vehicle routing problem for emissions minimization. *Transportation Research Record: Journal of the Transportation Research Board*, (2197), 1-7.

Franceschetti, A., Demir, E., Honhon, D., Van Woensel, T., Laporte, G. and Stobbe, M. (2017). A metaheuristic for the time-dependent pollution-routing problem. *European Journal of Operational Research*, 259(3), 972–991.

Franceschetti, A., Honhon, D., Van Woensel, T., Bektaş, T. and Laporte, G. (2013). The time-dependent pollution-routing problem. *Transportation Research Part B: Methodological*, 56, 265–293.

Gajpal, Y. and Abad, P. (2009). An ant colony system (ACS) for vehicle routing problem with simultaneous delivery and pickup. *Computers & Operations Research*, 36(12), 3215–3223.

Gan, X., Wang, Y., Li, S. and Niu, B. (2012). Vehicle routing problem with time windows and simultaneous delivery and pick-up service based on MCPSO. *Mathematical Problems in Engineering*, 2012, 1–11.

Garrido, P. and Castro, C. (2009). Stable solving of CVRPs using hyperheuristics. *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation*, 255-262.

Garrido, P. and Castro, C. (2012). A flexible and adaptive hyper-heuristic approach for (dynamic) capacitated vehicle routing problems. *Journal Fundamenta Informaticae - Emergent Computing*, 119(1), 29-60.

Garrido, P. and Riff, M. C. (2010). DVRP: A hard dynamic combinatorial optimisation problem tackled by an evolutionary hyper-heuristic. *Journal of Heuristics*, 16(6), 795-834.

Goksal, F. P., Karaoglan, I. and Altiparmak, F. (2013). A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery. *Computers & Industrial Engineering*, 65(1), 39–53.

Göksal, F. P. (2010). *Eşzamanlı topla-dağıt araç rotalama problemi için sezgisel yaklaşımlar: genetik algoritma ve kuş sürüsü eniyileme*, Yüksek Lisans Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara, 51-52.

Jabali, O., Van Woensel, T. and de Kok, A. G. (2012). Analysis of travel times and CO2 emissions in time-dependent vehicle routing. *Production and Operations Management*, 21(6), 1060–1074.

Jabir, E., Panicker, V. V. and Sridharan, R. (2017). Design and development of a hybrid ant colony-variable neighbourhood search algorithm for a multi-depot green vehicle routing problem. *Transportation Research Part D: Transport and Environment*, 57, 422–457.

Karaoğlan, İ. (2009). *Dağıtım ağları tasarımında yer seçimi ve eşzamanlı topla-dağıt araç rotalama problemleri*, Doktora Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara, 17-20.

Koç, Ç. and Karaoglan, I. (2016). The green vehicle routing problem: A heuristic based exact solution approach. *Applied Soft Computing*, 39, 154–164.

Koç, Ç. and Laporte, G. (2018). Vehicle routing with backhauls: Review and research perspectives. *Computers & Operations Research*, 91, 79–91.

Koç, Ç., Bektaş, T., Jabali, O. and Laporte, G. (2014). The fleet size and mix pollution-routing problem. *Transportation Research Part B*: *Methodological*, 70, 239–254.

Kopfer, H. W., Schönberger, J. and Kopfer, H. (2013). Reducing greenhouse gas emissions of a heterogeneous vehicle fleet. *Flexible Services and Manufacturing Journal*, 26(1-2), 221–248.

Kuo, Y. (2010). Using simulated annealing to minimize fuel consumption for the time dependent vehicle routing problem. *Computers & Industrial Engineering*, 59(1), 157-165.

Küçükoğlu, İ., Ene, S., Aksoy, A. and Öztürk, N. (2014). A memory structure adapted simulated annealing algorithm for a green vehicle routing problem. *Environmental Science and Pollution Research*, 22(5), 3279–3297.

Lin, C., Choy, K. L., Ho, G. T. and Ng, T. W. (2014). A genetic algorithm-based optimization model for supporting green transportation operations. *Expert Systems with Applications*, 41(7), 3284–3296.

Lin, C., Choy, K. L., Ho, G. T., Chung, S. H., and Lam, H. Y. (2014). Survey of green vehicle routing problem: Past and future trends. *Expert Systems with Applications*, 41(4), 1118-1138.

Lourenço, H. R., Martin, O. C., and Stützle, T. (2002). Iterated local search. *International Series in Operations Research & Management Science*, 320–353.

Martin, O. C., Otto, S. W. and Felten, E. W. (1991). Large-step Markov chains for the traveling salesman problem. *Complex Systems*, 5(3), 299–326.

Meignan, D., Creput, J.-C. and Koukam, A. (2008). A coalition-based metaheuristic for the vehicle routing problem. *2008 IEEE Congress on Evolutionary Computation*, 1176-1182.

Meignan, D., Koukam, A. and Créput, J.-C. (2009). Coalition-based metaheuristic: a self-adaptive metaheuristic using reinforcement learning and mimetism. *Journal of Heuristics*, 16(6), 859–879.

Min, H. (1989). The multiple vehicle routing problem with simultaneous delivery and pick-up points. *Transportation Research Part A: General*, 23(5), 377-386.

Mladenovic, N. and Hansen, P. (1997). Variable neighbourhod search. *Computers & Operations Research*, 24(11), 1097-1100.

Mlejnek, J. and Kubalik, J. (2013). Evolutionary hyperheuristic for capacitated vehicle routing problem. *Proceeding of the Fifteenth Annual Conference Companion on Genetic and Evolutionary Computation*, 219-220.

Mlejnek, J. and Kubalik, J. (2013). Evolutionary hyperheuristic for capacitated vehicle routing problem. *Proceeding of the Fifteenth Annual Conference Companion on Genetic and Evolutionary Computation Conference Companion - GECCO '13 Companion*, 219-220.

Naderipour, M. and Alinaghian, M. (2016). Measurement, evaluation and minimization of CO2, NOx, and CO emissions in the open time dependent vehicle routing problem. *Measurement*, 90, 443–452.

Nagy, G. and Salhi, S.(2005). Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries. *European Journal of Operational Research*, 162(1), 126-141.

Niu, Y., Yang, Z., Chen, P. and Xiao, J. (2018). Optimizing the green open vehicle routing problem with time windows by minimizing comprehensive routing cost. *Journal of Cleaner Production*, 171, 962–971.

Oberscheider, M., Zazgornik, J., Henriksen, C. B., Gronalt, M. and Hirsch, P. (2013). Minimizing driving times and greenhouse gas emissions in timber transport with a near-exact solution approach. *Scandinavian Journal of Forest Research*, 28(5), 493–506.

Parragh, S. N., Doerner, K. F. and Hartl, R. F. (2008). A survey on pickup and delivery problems. *Journal Für Betriebswirtschaft*, 58(1), 21–51.

Pisinger, D. and Ropke, S. (2007). A general heuristic for vehicle routing problems. *Computers & Operations Research*, 34(8), 2403–2435.

Poonthalir, G. and Nadarajan, R. (2018). A fuel efficient green vehicle routing problem with varying speed constraint (F-GVRP). *Expert Systems with Applications*, 100, 131–144.

Rieck, J. and Zimmermann, J. (2009). A branch-and-cut approach to the vehicle routing problem with simultaneous delivery and pick-up. *Operations Research Proceedings*, 301–306.

Ropke, S and Pisinger, D. (2006). A unified heuristic for a large class of vehicle routing problems with backhauls. *European Journal of Operational Research*, 171(3), 750–775.

Sabar, N. R., Zhang, X. J. and Song, A. (2015). A math-hyper-heuristic approach for large-scale vehicle routing problems with time windows. *2015 IEEE Congress on Evolutionary Computation (CEC)*, 830-837.

Salhi, S. and Nagy, G. (1999). A cluster insertion heuristic for single and multiple depot vehicle routing problems with backhauling. *Journal of the Operational Research Society*, 50(10), 1034-1042.

Sim, K. and Hart, E. (2016). A combined generative and selective hyper-heuristic for the vehicle routing problem. *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference*, 1093-1100.

Soria-Alcaraz, J. A., Ochoa, G., Sotelo-Figeroa, M. A. and Burke, E. K. (2017). A methodology for determining an effective subset of heuristics in selection hyper-heuristics. *European Journal of Operational Research*, 260(3), 972–983.

Soysal, M. and Çimen, M. (2017). A simulation based restricted dynamic programming approach for the green time dependent vehicle routing problem. *Computers & Operations Research*, 88, 297–305.

Soysal, M., Bloemhof-Ruwaard, J. M. and Bektaş, T. (2015). The time-dependent two-echelon capacitated vehicle routing problem with environmental considerations. *International Journal of Production Economics*, 164, 366–378.

Subramanian, A. and dos Anjos Formiga Cabral, L. (2008). An ILS based heuristic for the vehicle routing problem with simultaneous pickup and delivery and time limit. *Lecture Notes in Computer Science*, 135–146.

Subramanian, A. and Ochi, L. S. (2008). An efficient ILS heuristic for the vehicle routing problem with simultaneous pickup and delivery. *Instituto de Computaçao Technical Report*, RT 07/08.

Subramanian, A., Drummond, L. M. A., Bentes, C., Ochi, L. S. and Farias, R. (2010). A parallel heuristic for the vehicle routing problem with simultaneous pickup and delivery. *Computers & Operations Research*, 37(11), 1899–1911.

Subramanian, A., Uchoa, E. and Ochi, L. S. (2013). A hybrid algorithm for a class of vehicle routing problems. *Computers & Operations Research*, 40(10), 2519–2531.

Subramanian, A., Uchoa, E., Pessoa, A. A. and Ochi, L. S. (2011). Branch-and-cut with lazy separation for the vehicle routing problem with simultaneous pickup and delivery. *Operations Research Letters*, 39(5), 338–341.

Subramanian, A., Uchoa, E., Pessoa, A. A. and Ochi, L. S. (2013). Branch-cut-and-price for the vehicle routing problem with simultaneous pickup and delivery. *Optimization Letters*, 7(7), 1569–1581.

Suzuki, Y. (2011). A new truck-routing approach for reducing fuel consumption and pollutants emission. *Transportation Research Part D: Transport and Environment*, 16(1), 73-77.

Suzuki, Y. (2016). A dual-objective metaheuristic approach to solve practical pollution routing problem. *International Journal of Production Economics*, 176, 143–153.

Talbi, G. (Editors). (2002). *MetaHeuristics From Design to Implementation*, New Jersey: John Wiley & Sons, 146-150.

Tajik, N., Tavakkoli-Moghaddam, R., Vahdani, B. and Meysam Mousavi, S. (2014). A robust optimization approach for pollution routing problem with pickup and delivery under uncertainty. *Journal of Manufacturing Systems*, 33(2), 277–286.

Tasan, A. S. and Gen, M. (2012). A genetic algorithm based approach to vehicle routing problem with simultaneous pick-up and deliveries. *Computers & Industrial Engineering*, 62(3), 755–761.

Tiwari, A. and Chang, P. C. (2015). A block recombination approach to solve green vehicle routing problem. *International Journal of Production Economics*, 164, 379–387.

Toth, P. and Vigo, D. (Editors). (2002). *The Vehicle Routing Problem*, Philadelphia: Society for Industrial and Applied Mathematics, 1-7.

Ubeda, S., Arcelus, F. J. and Faulin, J. (2011). Green logistics at Eroski: A case study. *International Journal of Production Economics*, 131(1), 44–51.

Walker, J. D., Ochoa, G., Gendreau, M. and Burke, E. K. (2012). Vehicle routing and adaptive ıterated local search within the HyFlex Hyper-heuristic framework. *Learning and Intelligent Optimization*, 265–276.

Wassan, N. A., Wassan, A. H. and Nagy, G. (2008). A reactive tabu search algorithm for the vehicle routing problem with simultaneous pickups and deliveries. *Journal of Combinatorial Optimization*, 15(4), 368–386.

Wei, R., Zhang, T. and Tang, H. (2010). An improved particle swarm optimization algorithm for vehicle routing problem with simultaneous pickup and delivery. *Information Computing and Applications*, 430–436.

Xiao, Y. and Konak, A. (2016). The heterogeneous green vehicle routing and scheduling problem with time-varying traffic congestion. *Transportation Research Part E: Logistics and Transportation Review*, 88, 146–166.

Xiao, Y., Zhao, Q., Kaku, I. and Xu, Y. (2012). Development of a fuel consumption optimization model for the capacitated vehicle routing problem. *Computers & Operations Research*, 39(7), 1419–1431.

Yao, E., Lang, Z., Zhang, Y. and Yang, Y. (2015). Vehicle routing problem solution considering minimising fuel consumption. *IET Intelligent Transport Systems*, 9(5), 523–529.

Zachariadis, E. E., Tarantilis, C. D. and Kiranoudis, C. T. (2009). A hybrid metaheuristic algorithm for the vehicle routing problem with simultaneous delivery and pick-up service. *Expert Systems with Applications*, 36(2), 1070–1081.

Zachariadis, E. E., Tarantilis, C. D. and Kiranoudis, C. T. (2010). An adaptive memory methodology for the vehicle routing problem with simultaneous pick-ups and deliveries. *European Journal of Operational Research*, 202(2), 401–411.

Zhang, S., Lee, C. K. M., Choy, K. L., Ho, W. and Lp, W. H. (2014). Design and development of a hybrid artificial bee colony algorithm for the environmental vehicle routing problem. *Transportation Research Part D: Transport and Environment*, 31, 85–99.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı	: OLGUN, Büşra
Uyruğu	: T.C.
Doğum tarihi ve yeri	: 06.07.1992, Ankara
Medeni hali	: Evli
Telefon	: 0 (545) 537 23 90
e-mail	: busrakutlu@sdu.edu.tr



Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Yüksek lisans	Gazi Üniversitesi / Endüstri Mühendisliği	Devam ediyor
Lisans	Gazi Üniversitesi / Endüstri Mühendisliği	2015
Lise	Seferihisar Anadolu Lisesi	2010

İş Deneyimi

Yıl	Yer	Görev
2016-Halen	Süleyman Demirel Üniversitesi	Araştırma Görevlisi

Yabancı Dil

İngilizce

Yayınlar

Kutlu, B., and Altıparmak, F. (2018). A hyper-heuristic algorithm based on simulated annealing for vehicle routing problem with simultaneous pickup and delivery. *16th International Logistics and Supply Chain Abstracts.*

Olgun, B., and Altıparmak, F. (2019). A hyper-heuristic algorithm based on iterated local search for vehicle routing problem with simultaneous pickup and delivery. *ICAIAME Abstracts.*

Hobiler

Yüzme



GAZİ GELECEKTİR..