

**GERÇEK HAYAT MALİYETLERİNİ
GÖZ ÖNÜNE ALAN AÇIK UÇLU
ARAÇ ROTALAMA PROBLEMİ İÇİN
BİR MELEZ GENETİK ALGORİTMA ÖNERİSİ**
Yüksek Lisans Tezi

Erhan TONBUL

Eskişehir, 2016

**GERÇEK HAYAT MALİYETLERİNİ GÖZ ÖNÜNE ALAN AÇIK UÇLU ARAÇ
ROTALAMA PROBLEMİ İÇİN
BİR MELEZ GENETİK ALGORİTMA ÖNERİSİ**

Erhan TONBUL

YÜKSEK LİSANS TEZİ

Endüstri Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Nihal ERGİNEL

Eskişehir

Anadolu Üniversitesi

Fen Bilimleri Enstitüsü

Aralık, 2016

Bu tez çalışması BAP Komisyonunca kabul edilen 1504F174 no.lu proje kapsamında desteklenmiştir.

JÜRİ VE ENSTİTÜ ONAYI

Erhan TONBUL'un "Gerçek Hayat Maliyetlerini Göz Önüne Alan Açık Uçlu Araç Rotalama Problemi İçin Bir Melez Genetik Algoritma Önerisi" başlıklı tezi, 28/12/2016 tarihinde aşağıdaki jüri tarafından değerlendirilerek "Anadolu Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliği"nin ilgili maddeleri uyarınca, Endüstri Mühendisliği Anabilim dalında Yüksek Lisans Tezi olarak kabul edilmiştir.

	<u>Unvanı-Adı Soyadı</u>	<u>İmza</u>
Üye (Tez Danışmanı)	: Prof. Dr. Nihal ERGİNEL
Üye	: Doç. Dr. İnci SARIÇİÇEK
Üye	: Yard. Doç. Dr. Haluk YAPICIOĞLU

.....

Enstitü Müdürü

ÖZET

GERÇEK HAYAT MALİYETLERİNİ GÖZ ÖNÜNE ALAN AÇIK UÇLU ARAÇ ROTALAMA PROBLEMİ İÇİN BİR MELEZ GENETİK ALGORİTMA ÖNERİSİ

Erhan TONBUL

Endüstri Mühendisliği Anabilim Dalı

Anadolu Üniversitesi, Fen Bilimleri Enstitüsü, Aralık, 2016

Danışman: Prof. Dr. Nihal ERGİNEL

Açık uçlu araç rotalama problemi (AUARP), belirli kısıtlar altında depoya geri dönmeyen araçların kat ettiği toplam mesafenin en küçüklenmesi olarak tanımlanmaktadır. Aslında gerçek hayatta birçok lojistik firması maliyet hesabını kat edilen toplam mesafeye göre yapmamaktadır. Çoğunlukla, belirli noktalar için tercih edilen standartlaştırılmış güzergahlar söz konusudur. Bu belirli standart güzergahlar izlendiğinde ortaya çıkan maliyetler de o noktaların standart maliyetleri olur. Standart maliyet yaratan bir noktaya sahip bir rotanın toplam maliyetinin belirlenmesi için, bu maliyete uğrama maliyeti ve standart rotadan sapma maliyeti de eklenmektedir. Bu çalışmada, standart rota maliyetleri, uğrama maliyetleri ve rotadan sapma maliyetleri adı verilen gerçek hayat maliyetleri, açık uçlu araç rotalama problemlerinde ana maliyetler olarak ele alınarak model oluşturulmuştur. Problem NP-Zor yapıda olduğu için kesin çözüm üreten yazılımlar problemin çözümünde yeterli olmamaktadır. Metasezgiseller, akla yatkın bir sürede yeterince iyi sonuçlar elde etmek için iyi bir yol olduğundan çözüm, bir melez genetik algoritma ile yapılmıştır.

Anahtar Kelimeler: AUARP, Melez Metasezgisel, Genetik Algoritma, Yerel Arama.

ABSTRACT

A HYBRID GENETIC ALGORITHM PROPOSAL FOR AN OPEN VEHICLE ROUTING PROBLEM WHICH CONSIDERS REAL LIFE COSTS

Erhan TONBUL

Department of Industrial Engineering

Anadolu University, Graduate School of Sciences, December, 2016

Supervisor: Prof. Dr. Nihal ERGİNEL

Open vehicle routing problem (OVRP) is regarded as minimizing the total distance travelled by vehicles which are not returning to the depot, under some constraints. Actually, most of the logistic firms do not do the costing according to the total distance travelled. Mostly, there are preferred standardized courses for certain nodes. The costs arising from following these standard courses are the standard costs of the related nodes. In a route which has a node creating a standart cost, stop-by costs and standard route deviation costs should be added in order to reach the total cost of the route. In this study, OVRP is modeled with these real life costs which are named as standart routing costs, stop-by costs and route deviation costs. The problem is an NP-Hard problem and softwares giving exact solutions are inadequate in solving these problems. Since metaheuristic approach is a good way of having good solutions within a reasonable period of time, the problem is solved via a hybrid genetic algorithm.

Keywords: OVRP, Hybrid Metaheuristic, Genetic Algorithm, Local Search.

TEŐEKKÜR

Bu alıőmanın t¼m aőamalarında g¼r¼ő, katkı ve yardımlarıyla bana desteęini hi esirgemeyen ve yoluma ışık tutan ok kıymetli danıőmanım Prof. Dr. Nihal ERĐİNEL' e teőekk¼r¼ bir bor bilirim.

Bilimsel alıőmalar yapabilmek adına bana hayat g¼r¼ő¼m¼ ve bakıő aımı kazandıran, beni bu hususta y¼nlendiren ve bilim iin alıőmaya heveslendiren Eskiőehir Osmangazi niversitesi End¼stri M¼hendislięi Anabilim dalındaki ve Anadolu niversitesi End¼stri M¼hendislięi Anabilim dalındaki t¼m deęerli hocalarıma teőekk¼rlerimi sunarım.

Bu g¼nlere gelmemi saęlayan ve desteklerini benden hi esirgemeyen annem Nil¼fer ERBİL'e ve kardeőim zge TONBUL'a sonsuz teőekk¼rlerimi sunarım.

Bana dostluęuyla sonsuz destek veren ve varlıęını hep yanımda hissettięim Doęa KURTULUŐ'a sonsuz teőekk¼r ederim.

Erhan TONBUL

28/12/2016

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilemeyen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; bu çalışmanın Anadolu Üniversitesi tarafından kullanılan “bilimsel intihal tespit programı”yla tarandığını ve hiçbir şekilde “intihal içermediğini” beyan ederim. Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçlara razı olduğumu bildiririm.

Erhan TONBUL

İÇİNDEKİLER

BAŞLIK SAYFASI	i
JÜRİ VE ENSTİTÜ ONAYI.....	ii
ÖZET	iii
ABSTRACT	iv
TEŞEKKÜR.....	v
ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ	vi
İÇİNDEKİLER	vii
TABLolar DİZİNİ.....	x
ŞEKİLLER DİZİNİ	xi
GÖRSELLER DİZİNİ	xii
SİMGELER VE KISALTMALAR DİZİNİ.....	xiii
1. GİRİŞ.....	1
2. ARAÇ ROTALAMA PROBLEMİ VE TÜRLERİ	6
2.1. Gezin Satıcı Problemi	6
2.2. Klasik Araç Rotalama Problemi (KARP)	7
2.3. Açık Uçlu Araç Rotalama Problemi (AUARP)	9
2.4. Heterojen Filolu Araç Rotalama Problemi (HFARP)	9
2.5. Heterojen Filolu Açık Uçlu Araç Rotalama Problemi (HFAUARP)	10
2.6. Zaman Pencerele Araç Rotalama Problemi	10
2.7. Mesafe Kısıtlı Araç Rotalama Problemi	10
2.8. Bölünebilir Talepli Araç Rotalama Problemi.....	11
2.9. Çoklu Depo Araç Rotalama Problemi	11
2.10. Önce Dağıt Sonra Topla Araç Rotalama Problemi	11
2.11. Eş Zamanlı Topla-Dağıt Araç Rotalama Problemi.....	11
2.12. Dinamik Araç Rotalama Problemi.....	12

2.13. Stokastik Araç Rotalama Problemi	12
3. GERÇEK HAYAT MALİYETLERİ İLE AUARP MODELLEMESİ	13
3.1. Klasik AUARP ve Matematiksel Modeli	13
3.2. Gerçek Hayat Maliyetli AUARP ve Matematiksel Modeli	16
3.2.1. Gerçek hayat maliyetleri	18
3.2.1.1. Standart rota maliyeti.....	18
3.2.1.2. Uğrama maliyeti.....	18
3.2.1.3. Rotadan sapma maliyeti.....	20
3.2.2. Boyut analizi	24
3.2.3. Küçük boyutlu problemlerin GAMS ile çözümü	25
3.3. Gerçek Hayat Maliyetli ve Doluluk Oranını Ele Alan AUARP ve Matematiksel Modeli	26
4. GENETİK ALGORİTMA VE YEREL ARAMA ALGORİTMASI.....	30
4.1. Genetik Algoritma.....	30
4.1.1. Gösterim (Encoding).....	33
4.1.1.1. İkili gösterim.....	33
4.1.1.2. Permütasyon gösterim	34
4.1.1.3. Değer gösterimi.....	34
4.1.1.4. Ağaç gösterimi	35
4.1.2. Başlangıç Popülasyonunun Oluşturulması.....	36
4.1.3. Uyumluluk fonksiyonu.....	36
4.1.4. Seleksiyon.....	37
4.1.5. Çaprazlama.....	38
4.1.6. Mutasyon.....	39
4.1.7. Durma kriteri	39
4.2. Yerel Arama Algoritması.....	40
4.2.1. Yerel arama hareketleri ve komşuluk tanımı.....	40
4.2.1.1. 1-1 değişim (Swap)	41
4.2.1.2. 1-0 değişim (Insert)	41

4.2.1.3. 2-Opt.....	42
5. ÖNERİLEN MELEZ GENETİK ALGORİTMA MODELİ.....	46
5.1. Toplumlaşmış Genetik Algoritma (TGA)	47
5.1.1. Çözüm gösterimi ve açılması	49
5.1.2. Popülasyon yapısı ve başlangıç.....	49
5.1.3. Seleksiyon ve çaprazlama	50
5.1.4. Mutasyon.....	51
5.1.5. Algoritmaya dair değişkenler ve durma kriteri.....	51
5.2. Test Problemleri İle Karşılaştırmalar	52
5.3. Geliştirilen Karar Destek Sistemi.....	57
5.3.1. Hipotetik bir problem	58
6. SONUÇ VE ÖNERİLER.....	64
KAYNAKÇA	66
EK-1. Müşteri Sayısı 5 Olan Bir Hipotetik Problem.....	69
EK-2. Müşteri Sayısı 10 Olan Bir Hipotetik Problem	70
EK-3. Müşteri Sayısı 15 Olan Bir Hipotetik Problem	71
ÖZGEÇMİŞ.....	72

TABLULAR DİZİNİ

Sayfa

Tablo 3.1. Numaralanmış Rotalar	19
Tablo 3.2. Önerilen Matematiksel Modelin Boyut Analizi	24
Tablo 3.3. Üç Hipotetik Problemin GAMS ile Çözümü	25
Tablo 4.1. İkili Gösterimli Kromozom Örnekleri	34
Tablo 4.2. Permütasyon Gösterimli Kromozom Örnekleri	34
Tablo 4.3. Değer Gösterimli Kromozom Örnekleri	35
Tablo 4.4. Mutasyon	39
Tablo 4.5. 9 ve 1 Talep Noktalarının 1-1 Değişim Hareketi	41
Tablo 4.6. 5 Numaralı Talep Noktasının 1-0 Değişim Hareketi	42
Tablo 5.1. Augerat vd. (Set A) ARP Test Problemleri Karşılaştırma Sonuçları	54
Tablo 5.2. Christofides, Mingozzi ve Toth ARP Test Problemleri Karşılaştırma Sonuçları	55
Tablo 5.3. Devrim Ltd. Şti.'nin Talep Noktaları ve Talep Miktarları	60
Tablo 5.4. Talep Noktaları İçin A-Logistics'in Belirlediği Standart Ücretler	61
Tablo 5.5. Devrim Ltd. Şti. İçin İki Ayrı Durumdaki Maliyetler	61

ŞEKİLLER DİZİNİ

	<u>Sayfa</u>
Şekil 3.1. Açık Uçlu Rotalar	14
Şekil 3.2. 5 Numaralı Düğümün Muhtemel Standart Rotası	21
Şekil 3.3. 5 Numaralı Düğüme Giderken Oluşan Gerçek Rota	21
Şekil 3.4. Depodan Tam Kapasite İle Çıkan Kamyon.....	27
Şekil 4.1. Genetik Algoritma Akış Diyagramı	33
Şekil 4.2. Ağaç Gösterimi.....	35
Şekil 4.3. Rulet Tekerı Yöntemi	38
Şekil 4.4. Çaprazlama Örneđi.....	38
Şekil 4.5. Aynı Rota Üzerinde 2-Opt Hareketi.....	42
Şekil 4.6. Farklı Rotalar Üzerinde 2-Opt Hareketi.....	43
Şekil 4.7. Aynı Rota Üzerinde 1-1 Deđişim Hareketi.....	43
Şekil 4.8. Farklı Rotalar Üzerinde 1-1 Deđişim Hareketi.....	44
Şekil 4.9. Aynı Rota Üzerinde 1-0 Deđişim Hareketi.....	44
Şekil 4.10. Farklı Rotalar Üzerinde 1-0 Deđişim Hareketi.....	45
Şekil 5.1. Toplumlaşmış Genetik Algoritma Akış Şeması	48
Şekil 5.2. Karar Destek Sistemi Bileşenleri	63

GÖRSELLER DİZİNİ

Sayfa

Görsel 5.1. Geliştirilen Karar Destek Sisteminin Arayüzü	58
Görsel 5.2. Gerçek Hayat Maliyetlerinin En Küçüklenmesi İle Bulunan Sonuç	62
Görsel 5.2. Kat Edilen Mesafenin En Küçüklenmesi İle Bulunan Sonuç	63

SİMGELER VE KISALTMALAR DİZİNİ

Kısaltmalar

ARP	: Araç rotalama problemi
AKARP	: Asimetrik klasik araç rotalama problemi
AUARP	: Açık uçlu araç rotalama problemi
HFARP	: Heterojen filolu araç rotalama problemi
HFAUARP	: Heterojen filolu açık uçlu araç rotalama problemi
KARP	: Klasik araç rotalama problemi
KDS	: Karar destek sistemi
OVRP	: Open vehicle routing problem
TGA	: Toplumlaşmış genetik algoritma
LJG-HGA	: Lui-Jiang-Geng melez genetik algoritma

Simgeler

N	: Talep noktası sayısı
t_{eni}	: Algoritmanın bulduğu en iyi sonucun CPU süresi

1. GİRİŞ

Açık uçlu araç rotalama problemleri, belirli talepleri olan dağıtım noktalarına araçların bazı kısıtlar altında dağıtım (veya toplama) yaparken gidilen toplam mesafenin ve filo büyüklüğünün en küçüklenmeye çalışıldığı ve araçların depoya geri dönmediği rota oluşturma problemidir. Bahsi geçen kısıtlar araç kapasite kısıtları, tüm taleplerin karşılanması zorunluluğu, taleplerin bölünmezliği vb. gibi kısıtlardır. Np-Zor yapıdaki bu problemler endüstri mühendisliğinin en çok çalışılan konuları arasında gelir. Standart araç rotalama problemlerinden en belirgin farkı araçların depoya geri dönme zorunluluğunun olmayışıdır. Yani oluşan rotalar açık rotalardır. Bu sebeple bu problemlere açık uçlu araç rotalama problemleri denilmiştir. Bu problemin incelenmesindeki temel sebep genelde lojistik firmalarının dağıtım yapılacak firmalarla gerçek hayatta yaptıkları anlaşma biçiminin tek yön üzerinden yapılmasıdır. Çünkü dağıtım yapılan firmayı aracın geri dönüşü ilgilendirmemektedir. Bununla birlikte dağıtım yapan firmanın da gidilen son noktadan depoya geri dönerken başka müşteri bulma olasılığı vardır. Bu şartlar altında açık uçlu bir araç rotalama problemi ortaya çıkmaktadır.

Literatüre açık uçlu araç rotalama problemi ismi ilk defa Sariklis ve Powell (2000, s. 564) tarafından katılsa da Schrage, 1981 yılında yaptığı sınıflandırma çalışmasında kapalı ve açık rotalar arasındaki farkı ilk ortaya koyan isim olmuştur (Li, Golden ve Wasil, 2007, s. 2919; Yu, Ding ve Zhu, 2011, s. 10568). Daha sonra Bodin vd. (1981, s. 108) tarafından bir özel kargo şirketinin kargo uçaklarının rotalanması hususunda yapılan çalışmada bu problem tipi çalışılmıştır ve Clarke ve Wright tasarruf algoritmasının bu probleme göre uyarlanmış bir versiyonu ortaya koyulmuştur. AUARP ismini ilk kez dile getiren Sariklis ve Powell, cezalı en küçük örten ağaç (minimum spanning tree) adlı bir sezgiselle çözmeye çalışmışlardır. Aynı problem, 2004 yılında bir tek noktalı metasezgisel arama yöntemi olan tabu arama algoritması ile çözülmeye çalışılmıştır (Brandão, 2004, s. 557). Bir başlangıç çözüm ile başlayan ve tek noktalı arama yapan bu yöntemin yerel iyiye sıkışmaması için kullandığı yöntem tabu listesi denir. Tabu listesi yardımıyla bazı hareketler belirli bir ardıştırma boyunca yasaklı kalırlar ve bu da çeşitliliği sağlar. AUARP, aynı yıl, temeli rotalar kümesindeki araçların, araç kapasite kısıtı altında müşterilere atanmasına dayanan BoneRoute adlı yeni bir metasezgisel yardımıyla

çözölmeye çalıřılmıřtır (Tarantilis vd., 2004, s. 439). Amaç, toplam kat edilen mesafenin en küçüklenmesidir. Fu ve diđerleri (2005, s. 270) ise bu probleme uygun bir tabu arama sezgiseli geliřtirmişlerdir. Maliyet kalemlerini, kat edilen toplam yol ve işletim maliyetleri adı altında kullanılan araç sayısının en küçüklenmesi olarak almışlardır. Yerel iyiden kaçabilmek için komşu seçimini rassal olarak almayı tercih etmişlerdir. Aksen, Özyurt ve Aras (2007, s. 1225) AUARP'nin farklı bir tipini çalışmışlardır. Bu problem tipinde araçlar müşteri noktalarındaki dağıtım veya toplama işlemlerini bitirdikten sonra sürücü düğümü adı verilen noktalarda rotalarını sonlandırmaktadırlar. Sürücü düğümleri, sürücünün evi veya aracın bırakılması gereken garaj vb. gibi son noktalardır. Bu problem tipini açık tabu arama algoritması adını verdikleri bir metasezgisel arama yöntemiyle çözmeye çalışmışlardır. Aynı problem için Derigs ve Reuter (2009, s. 1658-1669) özellik temelli tepe tırmanma özelliđi olan bir tabu arama sezgiselini tanıtmışlardır. Bu sezgisel, bilinen tabu arama algoritmasının parametresiz bir hali olarak tanımlanmıştır. Fleszar, Osman ve Hindi (2009, s. 804), araç rotalama problemini çözmek için kendi geliřtirdikleri etkili deđişken komşuluk arama sezgiselini kullanmışlardır. Amaç, kat edilen yolun en küçüklenmesi ve araç sayısının en aza indirilmesidir. Zachariadis ve Kiranoudis (2010, s. 712-723), zaman zaman çok fazla kaynak tüketen yerel arama işlemine farklı bir yaklaşım getirmiş ve daha geniş komşulukları tarayan bir metasezgisel türetmişlerdir. Erbao ve Mingyong (2010, s. 2405 - 2411), bu kombinatoriyel optimizasyon probleminin taleplerinin bulanık olduđu tipi üzerinde çalışmışlar ve stokastik simölasyon ve geliřtirilmiş diferansiyel evrimsel algoritmayı melezleyerek bir melez zeki algoritma türetmişlerdir. Repoussis ve diđerleri (2010, s. 445), AUARP için melez evrimsel strateji önermişlerdir. Bu çalışmada amaç yine toplam gidilen mesafeyi ve filoda kullanılan araç sayısını en küçüklemektir. Mirhassani ve Abolghasemi (2011, s. 11547 - 11551), kapasite kısıtlı AUARP için parçacık sürü optimizasyon metasezgiseli ile çözüm aramışlardır. Yu, Ding ve Zhu (2011, s. 10569), genetik algoritmanın güçlü çeşitlendirme yöntemi ve tabu aramanın etkin keskinleştirme özelliklerini bir araya getirerek bir melez metasezgisel oluşturmuşlardır. Amaç, kullanılan araç sayısını ve toplam mesafeyi en küçüklemektir. Liu, Jiang ve Geng (2012, s. 404), açık uçlu araç rotalama problemlerinin bir farklı kolunu

çalışmışlardır. Bu problem, çoklu depolu açık uçlu araç rotalama problemidir. Burada birden fazla depo vardır ve araçlar yine bu depolardan müşterilere gitmektedir ama geri dönmek durumunda değildirler. Bu problemi bir melez genetik algoritma ile çözmüşlerdir. Amaç yine kat edilen toplam yolun ve araç sayısının en küçüklenmesidir. Banos ve diğerleri (2013, s. 286-296), zaman pencereli ve kapasite kısıtlı AUARP üzerinde çalışmışlardır. Bu çalışmada, kat edilen toplam mesafenin en küçüklenmesi haricinde yüklerin araçlara dengeli dağıtımını da amaç edinilmiştir. Çözüm yöntemi olarak yine bir melez metasezgisel kullanılmıştır. López-Sánchez ve diğerleri (2014, s. 104 - 113) faaliyet gösteren bir şirketin servis rotalarının optimize edilmesini konu alan bir gerçek hayat problemi çözmüşlerdir. Bu problemde temel amaç bir kişinin serviste harcadığı en uzun sürenin en küçüklenmesi olarak verilmiştir. Ayrıca araç sayısının en küçüklenmesi ve rotaların dengelenmesi de problemin diğer amaçlarıdır. Çözüm için rekabetçi çoklu başlangıç algoritması önerilmiştir. Gerçek hayat verileriyle algoritma test edilmiştir. Brito ve diğerleri (2015 s. 153 -163), ARP'nin farklı bir kolu olan kapalı açık uçlu araç rotalama problemlerine çözüm aramışlardır. Bu problem tipinde araçların depoya geri dönmeleri zorunlu değildir fakat dönmeleri aynı zamanda mümkündür. Amaç, toplam gidilen mesafenin en küçüklenmesidir ve çözüm yöntemi olarak bulanık optimizasyon yöntemi ve bir melez metasezgisel önerilmiştir. Yu, Jewpanya ve Redi (2016, s. 6-17), çapraz sevkiyat (cross-docking) denilen sevkiyat şeklinin geçerli olduğu AUARP üzerinde çalışmışlardır. Amaç, toplam taşıma maliyetlerinin ve araç kiralama maliyetlerinin en küçüklenmesidir. Çözüm yöntemi olarak bir tavlama benzetimi algoritması önerilmiştir.

Çoğu araştırmacı bu probleme ilişkin maliyetleri kat edilen toplam yol ve işletim maliyetleri olarak görmüşlerdir. Oysaki, gerçek hayat problemlerinde biraz daha farklı bir maliyetlendirme kullanılmaktadır. Bilhassa, ürün sevkiyatı yapılacak olan firma, araçlarını lojistik hizmeti veren başka bir firmadan kiralyorsa genelde araçlarını kiraya veren firmanın ücretlendirme sistemi üzerinden rotaların maliyeti oluşmaktadır. Bu ücretlendirme sisteminde de genellikle bu çalışmada tanıtılacak olan maliyetler söz konusu olmaktadır. Bu şekilde araçlarını başka bir lojistik firmasında kiralayan şirketler için literatürde önerilen amaç fonksiyonu, maliyet kalemlerinin farklı olmasından dolayı doğru bir

en iyilemeye sebep olmamaktadır. Çünkü en küçüklenmesi gereken değer kuşkusuz maliyettir ve maliyeti etkileyen kat edilen toplam mesafe ve işletim giderleri (araç sayısı) dışında birçok unsur bulunmaktadır.

Literatürdeki araç rotalama problemlerinde maliyet, genellikle kat edilen toplam yoldur. Bu yaklaşım yanlış değildir ama eksikleri bulunmaktadır. Eskişehir’de bulunan bir depodan İstanbul’a direkt gitmek ve Kocaeli’ye dağıtım yapıp gitmek arasında literatürde tanıtılan klasik yaklaşımlara göre bir fark bulunmamaktadır. Çünkü mesafe aynıdır. Fakat gerçek hayatta bu durum biraz daha farklı gerçekleşmektedir. Bir şehrin anayolundan geçip gitmek ve o şehre uğramak aynı maliyeti ortaya çıkartmamaktadır. Çünkü dağıtım yapılacak üzere bir şehre gelindiğinde dağıtım yapılacak noktaya da ekstra bir mesafe kat edilmesi söz konusudur. Ayrıca, araçlarını kiraya veren lojistik firmaları, her bir rotadaki en yüksek maliyetli talep noktası için bir standart maliyet ve o noktaya giderken izlenecek standart güzergahtan sapmalar için bir ekstra maliyet belirlerler.

Bu çalışmaya konu olan maliyetler üç başlıkta toplanmıştır; **standart rota maliyetleri, uğrama maliyetleri ve rotadan sapma maliyetleri**. Ayrıca taşıma yapan firmaları ilgilendiren bir diğer önemli parametre ise araçların yol boyunca **doluluk oranlarıdır**. Yine, rotalama problemlerinin bu yönü bu çalışmadaki şekliyle literatürdeki çalışmalarda ele alınmamıştır. Bu parametre, bir aracın yol boyunca taşıdığı yükün aracın kapasitesine oranının ortalama değeri olarak tanımlanabilir. Bu çalışmada bu parametreye kısaca doluluk oranı denilmiştir. Tonbul vd. 2015 yılında bu maliyetlendirme sistemine temel olacak bir özet çalışma ortaya koymuşlardır. Tonbul, Tuna ve Erginel, 2016 yılındaki bir özet çalışmada, bu çalışmada bahsi geçen doluluk oranı kavramını modellemeye çalışmışlardır. Bu çalışmada tanımlanan maliyet kalemleri ve doluluk oranı kavramı, bahsi geçen özet çalışmaları bir çatı altında toplamaya çalışmıştır.

Bu çalışmada, literatürde yer alan klasik araç rotalama modellerinden farklı olarak, gerçek hayatta karşılaşılan ve karar vericilerin kullandıkları maliyetler modellenmiş ve araç rotalamanın bahsedilen maliyetleri dikkate alacak şekilde yapıldığı modeller oluşturulmuştur. Daha sonra oluşturulan modeller NP-zor yapıda olduğundan, metasezgiseller ile önerilen modellere çözüm aranmıştır. Modelin çözümü için melez yapıda bir metasezgisel önerilmiştir. Genetik algoritma

ile yerel arama yöntemlerinin birlikte çalıştığı bir metasezgisel yapı oluşturulmuştur.

Bu çalışmada 3 model ele alınmıştır. Birinci model literatürdeki klasik yaklaşımlara göre gidilen yol ve kullanılan kamyon sayısının en küçüklenmeye çalışıldığı bir model, ikinci model, gerçek hayatta karşılaşılan maliyetlerin göz önüne alındığı bir model ve üçüncü model ise gerçek hayat maliyetleri kullanılırken, doluluk oranı yüksek bir şekilde uzun mesafe gidilmesini hedefleyen bir modeldir.

Bu tezin ikinci bölümünde ARP modelleri, türleri ve literatürde yer alan çözüm yöntemleri tanıtılmıştır. Üçüncü bölümde açık uçlu araç rotalama problemi daha derin bir şekilde tanıtılmıştır ve önerilen açık uçlu rotalama modelleri verilmiştir. Dördüncü bölümde metasezgisel ailesinden genetik algoritma ve tek noktalı arama yapan tabu arama algoritması anlatılmıştır. Beşinci bölümde, önerilen melez metasezgisel tanıtılmış ve etkinliği araştırılmıştır. Bu çalışma kapsamında geliştirilen karar destek sistemi tanıtılmış ve bir örnek problem üzerinde sonuçları tartışılmıştır. Altıncı bölümde sonuç ve öneriler yer almaktadır.

2. ARAÇ ROTALAMA PROBLEMİ VE TÜRLERİ

2.1. Gezgin Satıcı Problemi

Gezgin satıcı problemi, literatürdeki en çok çalışılan, Np-Zor sınıfına ait, kombinatoryel problemlerden biridir. Hayali bir gezgin satıcı, bir sıfır noktasından (depo) dağıtımına başlayarak, n tane noktaya uğrar ve dağıtım yapar ve başlangıç noktasına geri döner. Dağıtım noktaları arasındaki mesafeler bellidir ve bir matriste gösterilmiştir. Her noktaya sadece bir kere uğramak suretiyle, gezgin satıcının amacı en küçük maliyetli rotayı oluşturmaktır. Rota maliyetini genel olarak rotada kat edilen toplam mesafe olarak alınır. Gezgin satıcı problemi matematiksel olarak şu şekilde ifade edilebilir:

Karar Değişkeni:

$$x_{ij} = \begin{cases} 1, & \text{Eğer } i \text{ şehriden } j \text{ şehrine gidiliyorsa} \\ 0, & \text{Diğer durumlarda} \end{cases}$$

Parametre:

c_{ij} : i noktasından j noktaya gitme maliyeti

$$\text{EnkZ} = \sum_{i=0}^n \sum_{j \neq i}^n c_{ij} x_{ij} \quad (2.1)$$

Kısıtlar:

$$\sum_{j=0}^n x_{ij} = 1 \quad \forall i, i \neq j \quad (2.2)$$

$$\sum_{i=0}^n x_{ij} = 1 \quad \forall j, j \neq i \quad (2.3)$$

$$u_i - u_j + nx_{ij} \leq n - 1 \quad 1 \leq i \neq j \leq n \quad (2.4)$$

Matematiksel modelden de anlaşılacağı gibi gezgin satıcı probleminde amaç, $c_{ij}x_{ij}$ çarpım toplamının en küçüklenmesidir. (2.1), amaç fonksiyonudur. Kısıt (2.2) her şehirden mutlaka bir yere gidilmesi gerektiğini belirtir. Kısıt (2.3)'e göre her şehre mutlaka bir yerden gelinmelidir. Kısıt (2.4) alt tur engellemek için yazılan bir kısıttır. n , problemdeki düğüm sayısını ifade etmektedir. u parametresi ise alt tur engelleme kısıtında yer alan, eksiksiz bir turun gerçekleşmesini sağlayacak yapay bir değişkendir. Araç rotalama problemleri, literatürde m-GSP olarak anılmaktadır. Araç rotalama problemlerinin gezgin satıcı problemlerinden temelde tek farkı, GSP'de bir gezgin varken, ARP'de birden fazla gezgin (araç) vardır. Bir başka deyişle, ARP, birden çok gezginin olduğu GSP olarak görülebilir.

2.2. Klasik Araç Rotalama Problemi (KARP)

İlk olarak 50'li yılların sonunda, Dantzig ve Ramser (1959, s. 80) tarafından ortaya atılmıştır. Klasik araç rotalama problemi, araç kapasite kısıtları ve diğer kısıtlar göz önünde bulundurularak, müşteri noktalarının talepleri karşılanırken toplam maliyetin en küçüklenmesi olarak tanımlanır. Klasik ARP'nin temel kısıtları şöyle ifade edilir. Her şehre sadece ve sadece bir kere uğranılmalıdır. Her müşterinin talebi karşılanmalıdır. Her müşteri sadece bir aracın rotasında var olabilir. Her rota mutlaka depoda başlar ve yine depoda sonlanır. Bir aracın uğradığı şehirlerdeki taleplerin toplamı o aracın kapasitesini aşamaz.

Her ne kadar literatürde tanımlanan klasik ARP bu kısıtlardan ibaret olsa da gerçek hayatta ele alınması gereken farklı kısıtlar da söz konusu olabilir. Bu kısıtlar yasalardan ileri gelen, sürücülerin maksimum çalışabileceği saat sayısı ya da müşterilerin bir sevkıyatı kabul etme zaman aralığı olabilir. Klasik araç rotalama problemi için Laporte (1992, s. 345), şu şekilde bir matematiksel model önermiştir:

$G = \{V, A\}$: Bir serim olsun.

$V = \{0, \dots, n\}$: Düğüm kümesi

A: Ayırt kümesi

0 noktası depoyu işaret etmektedir. Diğer düğümler de servis yapılacak müşteri noktalarını gösterir.

Parametreler:

c_{ij} : *i şehrinde j şehrine gitmenin maliyeti*

d_{ij} : *i şehri ile j şehri arasındaki mesafe*

$c_{ii} = 0$ kabul edilir.

Karar Değişkeni:

$x_{ij} : \begin{cases} 1, & \text{Eğer } i \text{ şehrinde } j \text{ şehrine gidiliyorsa} \\ 0, & \text{Diğer durumda} \end{cases}$

$$\text{Enk } Z = \sum_{i \neq j} d_{ij} x_{ij} \quad (2.5)$$

Kısıtlar:

$$\sum_j x_{ij} = 1 \quad \forall i \in V \quad (2.6)$$

$$\sum_i x_{ij} = 1 \quad \forall j \in V \quad (2.7)$$

$$\sum_{i,j \in S} x_{ij} \geq |S| - v(s) \quad \{S: S \in V \setminus \{1\}, |S| \geq 2\} \quad (2.8)$$

$$x_{ij} \in \{0,1\} \quad \forall i, j \in E; i \neq j \quad (2.9)$$

Bazı araç rotalama problemlerinde bir i noktasından bir j noktasına gitme mesafesi (maliyeti) ile aynı j noktasından aynı i noktasına gitme mesafesi birbirinden farklıdır. Bu durumda problem asimetrik klasik araç rotalama problemi (AKARP) ismini almaktadır. Amaç toplam mesafenin en küçüklenmesidir. (2.5), amaç fonksiyonunu temsil etmektedir ve gidilen mesafenin en küçüklenmesi anlamına gelir. Kısıt (2.6), her şehirden mutlaka başka bir şehre gidilmelidir ifadesini anlatır. Kısıt (2.7)'e göre her şehre mutlaka başka bir şehirden ya da depodan gelinmesi gerektiği ifade edilir. Kısıt (2.8), alt turların önlenmesi için gereken kısıttır.

2.3. Açık Uçlu Araç Rotalama Problemi (AUARP)

Açık uçlu araç rotalama probleminin klasik araç rotalama probleminden farkı, araçların yaptığı rotaların depoda değil de bir müşteri noktasında sonlanmasıdır. Başka bir deyişle, araçlar müşteri noktalarına dağıtım-toplama işlemlerini yaptıktan sonra depoya geri dönmezler. Bu problem tipi sanayide sık karşılaşılan bir problem türüdür. Çünkü birçok dağıtım veya toplama yapması gereken üretim firması bu işlemi kendi bünyesindeki araçlarla değil başka bir lojistik firması ile anlaşarak yapmaktadır. Bu durumda araçların başlangıç noktası olan depoya dönmesi anlamsız olmakla birlikte dağıtım bir firma için yapan lojistik firmasının da rotaların son noktaları olan şehirlerden başka firmaların dağıtım işlemlerini yapmak üzere dönüşte de farklı rotalar izlemesi tercih edilen bir durumdur. Bu sebeple anlaşmalar genelde açık uçlu rotalar üzerine yapılmaktadır. Bu durumun anlaşmanın iki tarafı için de daha karlı olması beklenmektedir.

2.4. Heterojen Filolu Araç Rotalama Problemi (HFARP)

Lojistik firmalarının ya da farklı noktalardan dağıtım veya toplama yapmakla yükümlü herhangi bir firmanın genellikle araçları tek tipte değildir. Farklı çeşitlerde araçlar bu tip işlemler için tahsis edilmektedir. Bu da araç rotalama problemlerinin bir başka sınıfını açığa çıkarmaktadır. Araç kapasite kısıtlarının tek tip olmadığı durumlardaki araç rotalama problemlerine, heterojen filolu araç

rotalama problemi (HFARP) denilmektedir. Heterojen filolu araç rotalama problemlerine ilk olarak Kirby (1959, s. 252) tarafından değinilmiştir. Bu problem türüne ait alt sınıflandırmalar da söz konusudur. Araçların sabit maliyetlerinin olup olmamasına, rotalama maliyetlerinin gidilen yere bağlı olup olmamasına ve filo boyutunun sınırlı ya da sınırsız olmasına göre problem alt sınıflara ayrılabilir.

2.5. Heterojen Filolu Açık Uçlu Araç Rotalama Problemi (HFAUARP)

Bu problem tipi endüstri alanında en sık karşılaşılan araç rotalama türlerinden biri olarak anılmaktadır. Çünkü lojistik firmalarının araçları genelde heterojen olup ve dağıtım veya toplama yapmak için anlaşma yaptıkları firmaları tek gidiş üzerinden ücretlendirmektedirler. HFAUARP, araçların depoya dönmedikleri ve farklı türlerde (farklı kapasitelerde) ve sınırlı sayıda aracın bulunduğu araç rotalama problemi tipidir. Rotalar bir müşteri noktasında son bulmaktadırlar.

2.6. Zaman Pencere Araç Rotalama Problemi

Bu problem tipinde, her bir müşteriye uğranılması gereken spesifik bir zaman dilimi söz konusudur. Buna zaman penceresi denir ve $[a_i, b_i]$ şeklinde ifade edilir. a_i : i . müşteriye dağıtım veya toplama yapmak için gelinmesi gereken en erken zamanı, b_i : i . müşteriye servis yapıldıktan sonra müşteriden ayrılması gereken en geç zamanı temsil eder. Araçlar depodan ayrıldıktan sonra müşteriye giderken belirli bir t zamanı harcayıp, $[a_i, b_i]$ zaman penceresi içerisinde s kadar sürede servislerini yapıp diğer müşteriye gitmek üzere yola koyulmaktadırlar.

2.7. Mesafe Kısıtlı Araç Rotalama Problemi

Özellikle hava kargo rotalamalarında çok kullanılan bir modeldir. Araçların kat edebileceği maksimum mesafe önceden belirlenmiştir ve bu mesafe aşılamaz durumdadır.

2.8. Bölünebilir Talepli Araç Rotalama Problemi

Klasik araç rotalama problemlerinin kısıtlarından olan müşterilerin taleplerinin bölünemezliği bu problem türü için geçerli olmamaktadır. Yani müşteri noktasının talebi birden fazla araç tarafından karşılanıyorsa araç rotalama problemlerinin bu modeli geçerli olur. Bütün müşterilerin talebinin bölünmesi zorunlu olmamaktadır.

2.9. Çoklu Depo Araç Rotalama Problemi

Endüstride çoğu zaman firmalar tek bir depo noktasından dağıtım yapılmamaktadır. Farklı illerdeki depolardan müşteri noktalarına ulaşılması söz konusudur. Bu durum da bu problem tipinin gündeme gelmesini zorunlu kılmıştır. Müşterilerin ve depoların yerlerinin bilindiği bir ortamda müşteri noktalarına depolardan yalnızca birinden çıkan bir araç tarafından servis yapılmaktadır. Bir araç hangi depodan yola çıktıysa daha sonra o depoya geri dönmek zorundadır.

2.10. Önce Dağıt Sonra Topla Araç Rotalama Problemi

Bu problem türü yine gerçek hayatta sık karşılaşılan bir durumu temsil etmektedir. Müşteri noktalarına dağıtım yapan firmalar genellikle dağıtım yaptıkları noktalardan boş kasa vb. gibi ürünleri geri teslim almak durumundadır. Bu durumda yapılan dağıtımın bir de toplaması söz konusu olur.

Müşteriler dağıtım yapılacaklar ve toplama yapılacaklar olarak iki kümeye ayrılmaktadır. Öncelikle dağıtım yapılacak müşterilere servis yapılır daha sonra toplama işlemleri rotalanır. Aynı rotada hem dağıtım hem toplama işlemi yapılacak müşteri noktası varsa dağıtım yapılacak noktaya öncelik verilir.

2.11. Eş Zamanlı Topla-Dağıt Araç Rotalama Problemi

Dağıtım için gidilen müşteri noktasında dağıtımdan sonra bir önceki dağıtım ürünlerinin kasaları vb. geri alınır. Her müşteri noktasının bir dağıtım talebi, d_i ve bir toplama talebi, c_i vardır. Bu dağıtım ve toplama işlemleri tek celsede yapılır.

2.12. Dinamik Araç Rotalama Problemi

Özellikle son otuz yıldaki teknolojik alandaki gelişmeler, araç rotalama problemlerinin bir başka türünü ortaya çıkarmıştır. Bu problem tipi statik araç rotalama türünden farklıdır. Statik araç rotalama problemlerinde önceden talepleri ve yerleri bilinen bir müşteri kümesinin talepleri, bir araç filosu tarafından en küçük maliyetle karşılanmak istenirken, dinamik araç rotalama problemlerinde araçlar dağıtım yapmak üzere yola çıktığında dahi yeni gelen taleplerin dinamik olarak değerlendirilir ve planlanması yapıp çevrim süresinde yeni rotaların oluşturulması sağlanır. Kısaca, meydana gelen değişimlere cevap veren bir araç rotalama problemi türüdür. Bu değişimler, yolların durumunda meydana gelen değişiklikler, yeni talep noktalarının oluşması gibi dinamik koşullardır.

2.13. Stokastik Araç Rotalama Problemi

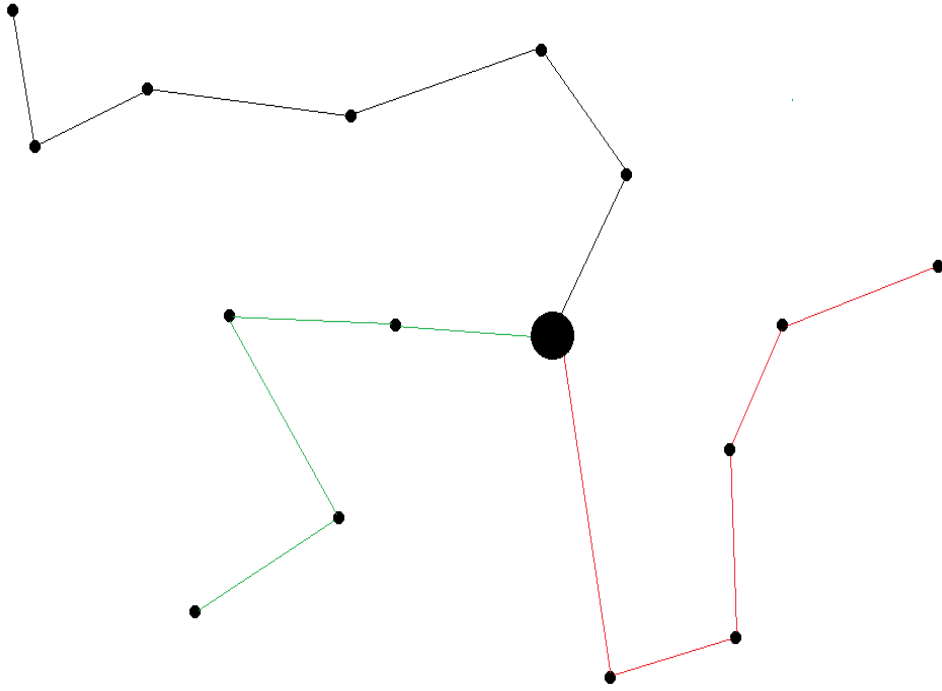
Araç rotalama probleminin bir ya da birden çok bileşeninin rassal olduğu durumlarda problem stokastik yapıda olmaktadır. Stokastik müşteriler, stokastik talepler ve stokastik süreler söz konusudur. Müşteriler belirli bir olasılıkla, belirli olasılıkla söz konusu olan bir talebe sahip olabilir ve bu olası müşterilere gidiş süreleri ve servis süreleri yine stokastik olarak belirlenebilecektir. Bu problemler genelde iki aşamada çözülmektedir. Öncelikle rassal değişkenlerin herhangi bir değeri belirlenmeden bir ilk çözüm türetilmektedir. İkinci aşamada ise bazı değişkenlerin değerleri ortaya çıkmaya başlayınca düzeltici hareketler yapılmaktadır.

3. GERÇEK HAYAT MALİYETLERİ İLE AUARP MODELLEMESİ

Bölüm 3.1.'de literatürde klasik anlamda tanımlanan araç rotalama problemi anlatılmış ve matematiksel modeli verilmiştir. 3.2. ve 3.3. bölümleri, bu problemin literatüre ilk defa bu çalışma aracılığı ile tanıtılacak olan gerçek hayat maliyetlerini ele alan versiyonlarını anlatmaktadır. Bölüm 3.2.'deki model standart rota maliyetleri, sapma maliyetleri ve uğrama maliyetlerini dikkate alan modeldir. Bölüm 3.3.'deki model ise amaç fonksiyonunda aynı gerçek hayat maliyetlerine ek olarak bir de doluluk oranını hesaba katmaktadır.

3.1. Klasik AUARP ve Matematiksel Modeli

Açık uçlu araç rotalama problemi, araçların talep noktalarına gidip dağıtım yaptıktan sonra depoya dönmedikleri araç rotalama problemi türü olarak anılır (Bkz. Şekil 3.1). Klasik araç rotalama probleminde amaç, talepleri belirli olan dağıtım veya toplama noktalarına kullanılan araçların kapasite kısıtı göz önüne alınarak mümkün olan en az araç sayısı, en kısa mesafeyi kat eden rotalar oluşturmaktır. Literatürdeki çalışmalarda amaç genellikle gidilen mesafenin ve kullanılan araç sayısının en küçüklenmesi olarak verilmiştir. Bu problemde genel olarak, araçların kapasitelerinden daha fazla yük alamayacakları, bütün taleplerin karşılanması zorunluluğu, bir noktaya ya başka bir noktadan ya da depodan gelinmesi zorunluluğu, araçların depoya geri dönmemesi ve alt tur eleme kısıtları gibi kısıtlar altında optimum sonucun bulunması ile ilgilenilmektedir. Problemin matematiksel gösterimi aşağıdaki gibidir.



Şekil 3.1. Açık Uçlu Rotalar

Kümeler:

i, j : Talep noktaları ($i, j = 0, 1, \dots, N$)

k : Araçlar ($k = 1, 2, \dots, K$)

0 noktası depoyu gösterir.

Karar değişkenleri:

x_{ijk} : Eğer k aracı i noktasından j noktaya giderse 1; diğer durumlarda 0

y_{ik} : Eğer k aracı i noktasını ziyaret ediyorsa 1; diğer durumlarda 0

v_k : Eğer k aracı kullanıldıysa 1; diğer durumlarda 0

Parametreler:

c_{ij} : i noktasından j noktaya seyahat etmenin maliyeti

q_i : i noktasının talebi

cap : araçların kapasitesi

Amaç fonksiyonu:

$$Enk \sum_{k=1}^K \sum_{i=0}^N \sum_{j=1}^N c_{ij} x_{ijk} \quad \forall i \neq j \quad (3.1)$$

Kısıtlar:

$$\sum_{k=1}^K y_{ik} = 1 \quad \forall i, i \neq 0 \quad (3.2)$$

$$\sum_{i=1}^N q_i y_{ik} \leq (cap) v_k \quad \forall k \quad (3.3)$$

$$\sum_{i=0}^N x_{ijk} = y_{ik} \quad \forall j, k, j \neq 0 \quad (3.4)$$

$$\alpha_i - \alpha_j + (N + 1) \sum_{k=1}^K x_{ijk} \leq N \quad \forall i, j, i \neq j, i \neq 0, j \neq 0 \quad (3.6)$$

$$\sum_{j=1}^N x_{ijk} \leq y_{ik} \quad \forall i, k \quad (3.7)$$

Bu problemin amaç fonksiyonuna zaman zaman en küçüklenmek üzere kullanılan araç sayısı da belirli bir parametrik katsayı ile çarpılarak eklenilmektedir. Çünkü kullanılacak araç sayısı gidilen toplam mesafeye göre skaler olarak çok küçüktür. Dengeli bir en küçükleme için kullanılan araç sayısı bir parametreyle çarpılarak büyütülmektedir (Yu, Ding ve Zhu, 2011, s. 10570).

Görüldüğü üzere problemin genel amacı, gidilen toplam mesafenin en küçüklenmesi olarak verilmiştir. Fakat lojistik işletmeleri için amaç basitçe maliyetlerin en küçüklenmesidir. Her ne kadar gidilen toplam mesafenin en küçüklenmesinin maliyeti doğrudan düşüreceği düşünülse de gerçek hayatta bu probleme dair ilgilenilmesi gereken başka maliyetler de söz konusudur. Sadece gidilen yolun en küçüklenmesi gerçek hayatta ortaya çıkan maliyetlerin tümünü kapsamamaktadır.

3.2. Gerçek Hayat Maliyetli AUARP ve Matematiksel Modeli

Birçok üretim şirketi ürünlerinin sevkiyatı için lojistik firmalarından araç kiralayarak dağıtım yapmaktadırlar. Bu durumda sevkiyatı yapılacak olan şirket, aracını kiraladığı lojistik firması için rotaları belirlemekte ve aracını kiraya veren firma da bu rotalar üzerinden bir ücretlendirme yapmaktadır. Bu ücretlendirme sistemi, kat edilen toplam mesafe ile bağlantılı olsa da çoğu zaman rekabet koşulları, araçların dönüşte yük bulma olasılığı, mesafeden bağımsız oluşan ücretli yollar gibi bazı maliyetler, araçlarını kiraya veren firmalar için farklı ücretlendirme sistemlerine sebep olmaktadır. Literatürdeki çalışmalar, genel olarak mesafeye bağlı oluşan maliyeti dikkate almıştır ve bahsi geçen maliyetler göz ardı edilmiştir. Bu tezin literatüre katkısı, gerçek hayatta oluşan maliyetleri modelleyerek şirketlerin lojistik işlerini yaparken rotalarını bu maliyetler doğrultusunda belirleyebilecekleri bir sistematik oluşturmaktır.

Giriş bölümünde bahsedildiği üzere, bu tip bir AUARP'de karşılaşılan maliyetler, literatürde tanıtılan klasik AUARP modellerinde belirlenen maliyetlerden farklıdır. Gerçek hayatta karşılaşılan birçok durum sebebiyle maliyetin doğrudan kat edilen mesafeyle ve kullanılan araç sayısı ile doğru orantılı olarak alınması modelin eksik kurulmasına ve dolayısıyla çözümlerin uygulanabilir olmaktan uzaklaşmasına yol açmaktadır. Literatürdeki çalışmalarda bahsedilmeyen ama gerçek hayatta maliyet doğurabilecek durumlardan bazıları aşağıdadır:

- Araçlarını kiraya veren lojistik firmalarının dağıtım yaptıktan sonra bazı büyük şehirlerden talep bulma olasılığı nispeten küçük şehirlere göre çok daha fazladır. AUARP'nin doğası gereği lojistik firmaları genellikle bir firmanın taleplerinin dağıtımını tamamladıktan sonra dönüş yolu için başka bir firmanın ya da firmaların dağıtım veya toplama işlemini yapmaktadır. Dolayısıyla lojistik firmaları için, dönüşte iş alma ihtimali düşük olan bazı şehirlere gitmenin görünmeyen bir maliyeti vardır. Benzer şekilde, metropollere gitmenin maliyeti, aynı uzaklıktaki bazı küçük şehirlere gitmenin maliyetiyle aynı görünse de metropollere gitmek dönüşte farklı işler alabilmek adına çok daha avantajlıdır.

- Gerçek hayatta şehirlerarası yollardan, ulaşmak istenen şehirlere giderken belirli güzergahlar izlenmek zorunda kalınır. Örneğin, Eskişehir'den Samsun'a giderken yol koşulları da dikkate alınarak genellikle Ankara, Kırıkkale, Çorum ve Amasya şehirleri üzerinden geçen bir güzergahtan Samsun'a ulaşılmaktadır. Literatürdeki temel yaklaşıma göre bir şehre doğrudan gitmek veya o şehre giderken kullanılacak olan güzergah üzerindeki şehirlerden birine dağıtım yapıp sonra o şehre gitmek arasında bir fark yoktur. Gerçek hayatta durum böyle değildir. Gerçek hayatta bir şehrin anayolundan geçip gitmek ve o şehre dağıtım yapıp gitmek arasında hem mesafe bazında hem de bu çalışmanın konusu olmayan bazı hususlar açısından (şehir içi trafiğine girmek, kamyonların şehir içi yollarda manevra kabiliyetinin düşük olması sebebiyle daha yavaş hareket etmesi vb.) oldukça farklı durumlardır.
- İki şehir arasında bazen en kısa yol yerine bazen en stabil yolun veya en hızlı yolun tercih edilmesi söz konusu olabilir. Bu noktada doğrudan en kısa yol her zaman en avantajlı yol olmayabilir.
- Şehirlerarası yolculuklarda bazı farklı maliyetler de söz konusu olabilmektedir. Bu maliyetlerden bazıları gişe geçiş ücretleri, köprü geçiş ücretleri veya yasal düzenlemelerden dolayı bazı araç tiplerinin bazı yolları kullanamama durumudur. Bu maliyet kalemleri mesafeden bağımsızdır.
- Araçlarını kiralayan lojistik firmaları için rekabet çok önemli bir unsur olup belirli talep noktaları için firmanın belirleyeceği ücretin rekabet unsurundan fevkalade etkilenmesi söz konusu olabilmektedir. Bu durumda ürünlerinin sevkiyatı yapılacak olan şirket, kat edilen mesafeden ve kullanılan araç sayısından bağımsız bir maliyet ile karşı karşıya gelmiş olmaktadır.

Lojistik firmaları için yukarıda belirtilen sebepler gibi daha birçok gizli veya belirsiz maliyet kalemi söz konusudur ve bu maliyetlerin birçoğu kat edilen mesafeden bağımsız gerçekleşmektedir. Bu gibi sebeplerden ötürü lojistik firmalarının dağıtım noktalarına gidiş yolları önceden belirlenen bazı tercihlere (güzergah) göre standartlaştırılmaktadır. Bu şehirlere bu standart rotalardan gitmenin maliyeti, bütün maliyet kalemleri hesaba katılarak hesaplanmaktadır.

Gerçekte oluşan rotalar, söz konusu rotayla aynı son noktaya sahip standart rotalardan sapma durumuna ve standart rota üzerinde ve/veya üzerinde olmayan kaç noktaya dağıtım yapıldığına göre değerlendirilmektedir. Amaç fonksiyonunda en küçükleme gereken değer aslında bu maliyetler bütünüdür. Bunu modellemek adına bu çalışmada farklı maliyet kalemleri tanıtılacaktır.

3.2.1. Gerçek hayat maliyetleri

3.2.1.1. Standart rota maliyeti

Bir lojistik firması herhangi bir tekil noktaya giderken izlenecek güzergahı (standart rota) kat edilen mesafenin en kısa olmasının yanı sıra, yol koşullarını da dikkate alarak belirlemiş ve bu güzergahın maliyetini önceden hesaplamış olacaktır. Örneğin Eskişehir'deki bir depodan Adıyaman'a gitmenin maliyeti bellidir (araç işletim gideri, km cinsinden gidilen yol x km başına benzin maliyeti, gişeler, ekstra harcamalar ve diğer görünmeyen maliyet kalemleri). Dolayısıyla Adıyaman'ın son nokta olduğu bir rotada, en temel maliyetin bu standart rota maliyeti olması beklenmektedir.

3.2.1.2. Uğrama maliyeti

Bir rota üzerinde en temel maliyet bir son noktaya standart rotadan gitmenin maliyeti olsa da o standart rota üzerindeki (ya da üzerinde olmayan) noktalara eğer dağıtım veya toplama yapılacaksa bu da bir maliyet doğurmaktadır. Bu maliyete uğrama maliyeti denir. Yol üzerindeki bir şehre uğramanın dağıtım yapan firma açısından şehir içinde gidilecek ekstra mesafe gibi sebeplerden ötürü bir maliyet doğurması beklenmektedir. Uğrama maliyeti önceden belirlenen bir sabit maliyet olabileceği gibi farklı şehirler için farklı uğrama maliyetleri de düşünülebilir (Dağıtım yapılacak yerin çevre yoluna uzaklığı, şehrin trafik durumu vb. gibi sebeplerden). Karayolları Genel Müdürlüğü'nün resmi web sitesi (<http://www.kgm.gov.tr/Sayfalar/KGM/SiteTr/Root/Uzakliklar.aspx>) üzerinden alınan verilere göre Eskişehir'deki bir depodan bir aracın çıktığı dört farklı rotayı ikili olarak karşılaştırmak gerekirse;

Tablo 3.1. Numaralanmış rotalar

Numara	Rotalar	Kat edilen mesafe
1	Eskişehir - Kocaeli - İstanbul	324 km
2	Eskişehir - İstanbul	324 km
3	Eskişehir - Burdur - Antalya	424 km
4	Eskişehir - Antalya	424 km

1 numaralı rota ve 2 numaralı rotanın doğuracağı maliyetler literatürde verilen AUARP yaklaşımlarına göre aynıdır. Çünkü 1 numaralı rota ve 2 numaralı rotada üzerinden geçilen şehirler, dolayısıyla kat edilen toplam mesafeler aynıdır. Bir başka deyişle Kocaeli, Eskişehir - İstanbul yolu üzerinde olduğundan iki rotada kat edilen toplam mesafe aynıdır. Benzer şekilde;

3 numaralı rota ve 4 numaralı rotanın doğuracağı maliyetler de yine literatürde verilen yaklaşımlara göre aynıdır.

Gerçek hayatta 1 numaralı rota ve 2 numaralı rotanın bir firma açısından farklı maliyetler doğuracağı aşıkardır. Çünkü 1 numaralı rota ve 2 numaralı rotada aslında aynı şehirler üzerinden gidildiği için toplam kat edilen mesafe aynı olsa da 1 numaralı rotada ekstradan Kocaeli'ye dağıtım yapmanın firmaya bir bedeli olması beklenir. 4 numaralı rotada Eskişehir - Antalya arası yolda aynı zamanda Burdur üzerinden geçileceği için toplamda 3 numaralı rota ile aynı mesafe kat edilmiş olur. Fakat 3 numaralı rotada aslında Burdur'a da dağıtım yapılacağından ekstra bir maliyet söz konusudur. Bu maliyetin büyük kısmı şehir içi kat edilen mesafe ve Burdur-Antalya anayolundan ayrılıp daha yavaş ilerlenecek, şehir içi trafiğin olabileceği yollardan gidişten ileri gelecektir. Harcanan toplam zamanı en küçükmek bu çalışmanın konusu olmasa da araçlarını kiralayacak olan firma bütün bu durumları göz önünde bulundurup ücretlendirme yapacağından son şehir dışındaki her bir şehir için bir uğrama maliyeti belirlemek gerekmektedir.

Bir başka örnek vermek gerekirse Eskişehir'deki depodan sadece Adıyaman'a ürün sevkiyatı olduğunu, dolayısıyla izlenecek güzergahın bu iki şehir arasındaki en uygun (en kısa) rota olduğunu düşünelim. Bu güzergahın Eskişehir- Ankara- Adana- Gaziantep- Adıyaman olduğunu varsayalım. Bu rotanın gösterimi **Eskişehir - Adıyaman** şeklinde olsa da diğer şehirlerden de geçeceğimiz için aslında rota

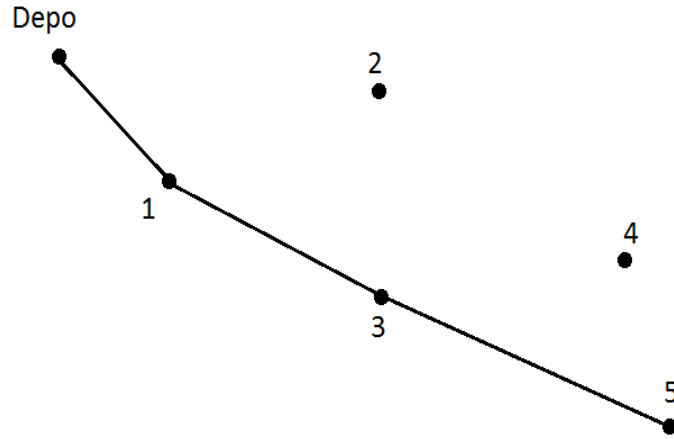
Eskişehir- Ankara- Adana- Gaziantep- Adıyaman şeklindedir. Bu rota üzerindeki bir yere dağıtım yapılacaksa, örneğin Adana'ya, bu bir uğrama maliyeti gerektirir. Bu maliyet de daha önce bahsedildiği gibi bir sabit maliyet olabileceği gibi şehirden şehre değişen bir maliyet matrisinden de alınabilir.

3.2.1.3. Rotadan sapma maliyeti

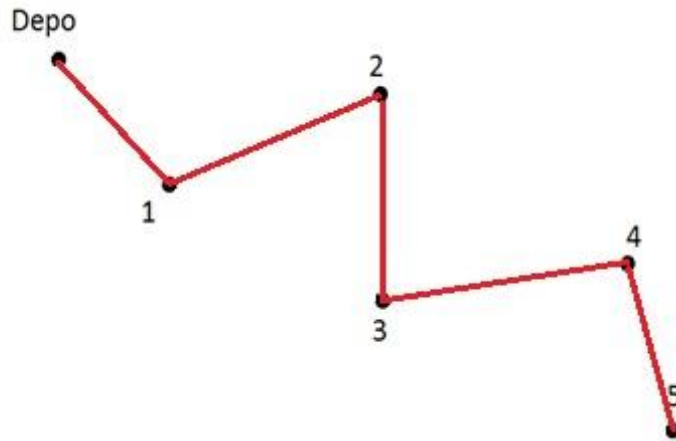
Bu maliyet, bir noktaya giderken gidilecek standart rotadaki şehirlerden farklı bir noktaya da gidiliyorsa ortaya çıkacak maliyettir. Bu maliyet matematiksel olarak bir noktaya standart rotadan giderken kat edilecek toplam yol ile sonucumuzda ortaya çıkan rotadan giderken kat edilen toplam yol arasındaki farkın km başına maliyet ile çarpılmasından ibarettir. Yukarıdaki örnekte Adıyaman'a giderken standart rota (**Eskişehir- Ankara- Adana- Gaziantep- Adıyaman**) 1026,115 kilometredir. Eğer bizim sonucumuzda oluşan rota **Eskişehir- Mersin- Gaziantep- Adıyaman** ise yani standart rotada olmayan bir nokta (Mersin) var ise, rotadan sapma maliyetimiz ortaya çıkar. Bu maliyet gerçekleşen rota uzunluğu ve standart rota uzunluğu arasındaki farkın kilometre başına yakıt maliyeti ile çarpılmasıyla bulunacaktır. Bu farka, rotadan sapma maliyeti denir. Şekil 3.2 ve Şekil 3.3'de sırasıyla hayali bir graftaki 5 numaralı düğümün standart rotası ve 5 numaralı düğüme giderken oluşan gerçek rota verilmiştir. 0 noktası depoyu temsil etmek üzere bu rotada oluşan sapma maliyeti:

$$C_{dev} = \{(d_{01} + d_{12} + d_{23} + d_{34} + d_{45}) - (d_{01} + d_{13} + d_{35})\}(Fuel_{cost}) \quad (3.8)$$

(3.8) eşitliğinde C_{dev} rotadan sapma maliyetini, $Fuel_{cost}$ km başına yakıt maliyetini, d_{ij} de i şehri ile j şehri arasındaki mesafeyi temsil etmektedir. Yani iki rota arasındaki mesafe farkının km başına maliyet ile çarpılması ile bulunur.



Şekil 3.2. 5 Numaralı Düğümün Muhtemel Standart Rotası



Şekil 3.3. 5 Numaralı Düğümüne Giderken Oluşan Gerçek Rota

Önerilen matematiksel model aşağıdaki gibidir:

İndisler:

i, j : Talep noktaları ($i, j = 0, 1, \dots, N$) (0 noktası depoyu gösterir.)

k : Araçlar ($k = 1, 2, \dots, K$)

Parametreler:

r_i : son şehri i .şehir olan rotanın standart rota maliyeti

q_i : i noktasının talebi

d_{ij} : i noktası ve j noktası arasındaki mesafe

cap : araçların kapasitesi

ent_{cost} : Bir şehre uğramanın maliyeti

$fuel_{cost}$: km başına yakıt maliyeti

Karar Değişkenleri:

$$x_{ijk} : \begin{cases} 1, & i. \text{şehirden } j. \text{ şehire } k. \text{ araç gidiyorsa} \\ 0, & \text{Diğer durumlarda} \end{cases}$$

$$y_{ik} : \begin{cases} 1, & i. \text{ şehre } k. \text{ araç uğruyorsa} \\ 0, & \text{Diğer durumlarda} \end{cases}$$

$$v_k : \begin{cases} 1, & k. \text{ araç kullanılıyorsa} \\ 0, & \text{Diğer durumlarda} \end{cases}$$

Amaç Fonksiyonu:

$$\min \sum_{k=1}^K \left(\sum_{i=1}^N r_i \left(y_{ik} - \sum_{j=1}^N x_{ijk} \right) + B + ent_{cost} \left(\sum_{i=1}^N y_{ik} - v_k \right) \right) \forall k, i \neq j \quad (3.9)$$

Kısıtlar:

$$\sum_{k=1}^K y_{ik} = 1 \quad \forall i, i \neq 0 \quad (3.10)$$

$$\sum_{i=1}^N q_i y_{ik} \leq cap(v_k) \quad \forall k \quad (3.11)$$

$$\sum_{i=0}^N x_{ijk} = y_{jk} \quad \forall j, k \quad j \neq 0 \quad (3.12)$$

$$\sum_{j=1}^N x_{ijk} \leq y_{ik} \quad \forall i, k \quad (3.13)$$

$$\sum_{j=1}^N x_{0jk} = v_k \quad \forall k \quad (3.14)$$

$$\alpha_i - \alpha_j + (N + 1) \sum_{k=1}^K x_{ijk} \leq N \quad \forall i, j \quad i \neq j \quad i \neq 0 \quad j \neq 0 \quad (3.15)$$

$$B \geq fuel_{cost} \sum_{k=1}^K \left(\sum_{i=0}^N \sum_{j=1}^N d_{ij} x_{ijk} - \sum_{i=1}^N d_{0i} \left(y_{ik} - \sum_{j=1}^N x_{ijk} \right) \right) \quad (3.16)$$

Yukarıdaki modelin amaç fonksiyonu (3.9) ile gösterilmektedir. Amaç fonksiyonunda rotaların standart rota maliyetleri toplamı $(\sum_{i=1}^N r_i (y_{ik} - \sum_{j=1}^N x_{ijk}))$ rotadan sapma maliyetleri toplamı $B \geq fuel_{cost} \sum_{k=1}^K \left((\sum_{i=0}^N \sum_{j=1}^N d_{ij} x_{ijk}) - \sum_{i=1}^N d_{0i} (y_{ik} - \sum_{j=1}^N x_{ijk}) \right)$ ve uğrama maliyetleri $(ent_{cost}) (\sum_{i=1}^N \sum_{k=1}^K y_{ik} - v_k)$ en küçüklenmek istenmektedir.

Kısıt (3.10), her bir i şehrinin talebinin mutlaka karşılanması gerektiğini belirten kısıttır. Kısıt (3.11), araç kapasite kısıtı olup bir aracın rotası üzerindeki taleplerin toplamının aracın kapasitesini aşamayacağını belirtmektedir. Kısıt (3.12), bir şehre bir aracın yalnızca başka bir şehirden veya depodan gelebileceğini temsil eden kısıttır. Kısıt (3.13), rotaların açık uçlu olmasını sağlayan kısıttır ve araçların bir şehirden başka bir şehre gitmeyebileceğini belirtmektedir. Kısıt (3.14), rotaların depodan başlayacağını ve Kısıt (3.15), alt turların oluşmayacağını garanti eden kısıtlardır.

Kısıt (3.16), amaç fonksiyonunda en küçüklenen B değerinin rotadan sapma maliyetine eşit olmasını sağlayan kısıttır.

3.2.2. Boyut analizi

Boyut analizi, matematiksel olarak modellenen bir problemin zorluk seviyesinin, modeldeki kısıt sayısı ve değişken sayısına bağlı olarak değerlendirilmesi için uygun bir araçtır. Boyut analizi, problem boyutu ile değişkenlerin alacağı değer sayısı ve işlenen kısıt sayısı arasındaki bağıntıyı verir. Başlangıç noktası (depo) hariç dağıtım yapılacak talep noktası sayısının m ile, başlangıç düğümü dahil dağıtım yapılacak talep noktası sayısının n ve araç sayısının da k ile gösterildiği bir problem için boyut analizi sonuçları Tablo 3.2.'de gösterilmiştir.

Tablo 3.2. Önerilen Matematiksel Modelin Boyut Analizi

Kısıtlar	İndisler	Değişken Sayısı	Kısıt Sayısı
$\sum_{k=1}^K y_{ik} = 1$	$\forall i \in V$	i	m
$\sum_{i=1}^N q_i y_{ik} \leq \text{cap}(v_k)$	$\forall k \in A$	$i.k$	k
$\sum_{i=0}^N x_{ijk} = y_{jk}$	$\forall j \in V, \forall k \in A$	$i.j.k$	$m.k$
$\sum_{j=1}^N x_{ijk} \leq y_{ik}$	$\forall i \in V_0, \forall k \in A$	$i.j.k$	$n.k$
$\sum_{j=1}^N x_{0jk} = v_k$	$\forall k \in A$	$j.k$	k
$\alpha_i - \alpha_j + (N + 1) \sum_{k=1}^K x_{ijk} \leq N$	$\forall i \in V, \forall j \in V, i \neq j$	$i.(j-1).k$	$m.(m-1)$
Kısıt (3.16)	$\forall k \in A,$ $\forall i \in V_0, \forall j \in V$	$i.j.k$	1
Değişken	İndisler	Değişken Sayısı	
x_{ijk}	$\forall i \in V_0,$ $\forall j \in V, \forall k \in A$	$i.j.k$	$n.m.k$
y_{ik}	$\forall i \in V, \forall k \in A$	$i.k$	$m.k$
v_k	$\forall k \in A$	k	k
Toplam Kısıt Sayısı:	$k(m + n + 2) + m^2 + 1$		
Toplam Değişken Sayısı:	$k(n.m + m + 1)$		

Boyut analizi için küme tanımlamaları şu şekilde yapılmıştır:

$V_0: \{0,1,2, \dots, N\}$	Depo hariç talep noktaları kümesi
$V: \{1,2, \dots, N\}$	Depo dahil talep noktaları kümesi
$A: \{1,2, \dots, K\}$	Araçlar kümesi

Buna göre talep noktası sayısının (m) 15, araç sayısının (k) 8 olduğu bir problemde **490** adet kısıt bulunmaktadır. Toplam değişken sayısı da aynı problem için **2048**'dir.

3.2.3. Küçük boyutlu problemlerin GAMS ile çözümü

Bu çalışma kapsamında önerilen matematiksel model literatüre yeni tanıtıldığı için bu modele dair literatürde herhangi bir test problemi bulunmamaktadır. Bu nedenle önerilen matematiksel modelin çözümü için üç adet küçük boyutlu hipotetik problem tanımlanmıştır. N , talep noktası sayısı olmak üzere problemler, üç farklı boyut için oluşturulmuştur. Bu problemler, kesin sonuçlar üretmesi ile bilinen GAMS v24.8.2 yazılımının CPLEX çözücüsüyle çözüme ulaştırılmıştır. EK-1, EK-2 ve EK-3'te sırasıyla 1., 2. ve 3. problemin parametreleri ve matrisleri bulunmaktadır. Problemlere dair sonuçlar Tablo 3.3.'te verilmiştir.

AUARP, NP-Zor yapıda bir problem olduğundan nispeten büyük boyutlu problemler için kesin çözüm yöntemleri verimli çalışmamaktadır (Hertz ve Widmer, 2003, s. 247). Bu çalışmada önerilen model ise klasik anlamdaki AUARP'den çok daha karmaşık bir yapıya sahip olduğundan büyük boyutlu problemlerin verimli bir şekilde çözülebilmesi için metasezgisel kullanımının gerekliliği açıktır.

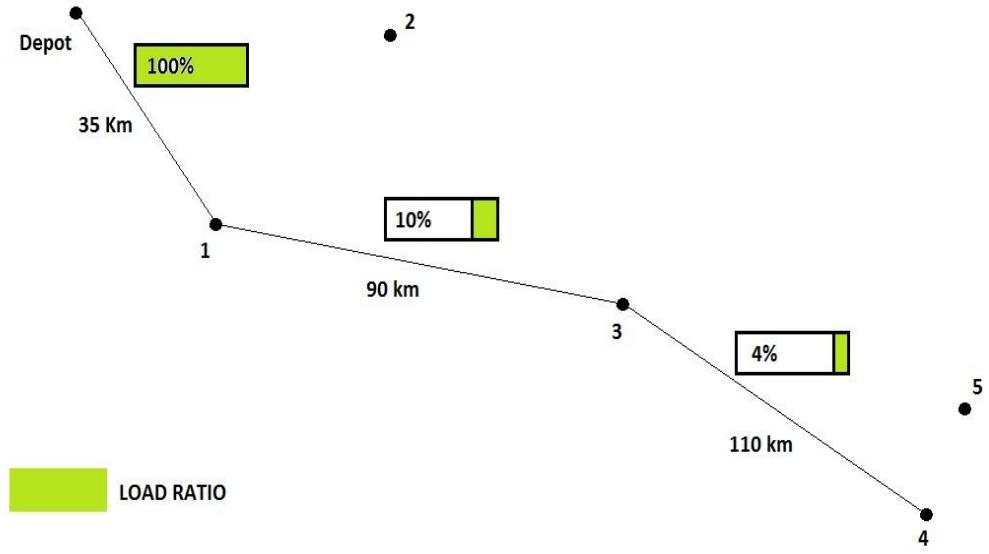
Tablo 3.3. Üç Hipotetik Problemin GAMS ile Çözümü

N	En iyi çözüm değeri	Süre (s)
5	4112,5	0,172
10	9075,5	0,450
15	12032,5	2,190

3.3. Gerçek Hayat Maliyetli ve Doluluk Oranını Ele Alan AUARP ve Matematiksel Modeli

Araç kiralayan lojistik departmanlarını ilgilendiren en önemli parametrelerden biri araçların yol boyunca doluluk oranlarıdır. Hatta gerçek hayatta araçların yeterli doluluk oranı sağlanamadığında çoğu zaman aracın dolmasını sağlayacak yeni taleplerin beklenmesi ve var olan taleplerin ertelenmesi söz konusu olur. Rotalama problemlerindeki doluluk oranı literatürdeki çalışmalarda ele alınmamıştır. Bu parametre, bir aracın yol boyunca taşıdığı yükün aracın kapasitesine oranının ortalama değeri olarak tanımlanabilir. Bu çalışmada bu parametreye kısaca doluluk oranı denilecektir. Rotalama problemlerinin bu yönünün neden taşıma yapan firmalar için önemli olduğunu görmek için aşağıdaki örnek verilebilir:

Depodan yüzde yüz doluluk oranı (tam kapasite) ile ayrılan bir aracın sırasıyla 35 km yol gidip ilk noktaya uğrayıp mal bırakacağı, sonra o noktadan 90 km sonra ikinci noktaya mal bırakacağı, ikinci noktadan 110 km sonra üçüncü noktaya mal bırakarak rotasını tamamlayacağını düşünelim. Şekil 3.4.'teki gibi bu kamyonun yükünün %90'ını 35 km sonra bırakıp kalan 200 kilometrenin ilk 90 kilometresinde kapasitesinin sadece %10'unu, kalan 110 kilometrede de kapasitesinin sadece %4'ünü taşıyarak rotasını tamamladığını düşünelim. Bu sonuç, kamyonun 235 kilometrelik rotasının % 80'ini neredeyse hiçbir şey taşımayarak geçirmesi anlamına gelmektedir. Bu, karlılık açısından mantıklı bir durum değildir. Bu istenmeyen durumla karşılaşmamak için uygulamada bazı yordamsal yöntemler kullanılmaktadır. Bu çalışmada bu parametrenin tersi, yani ortalama boşluk oranı standart rota maliyeti ile (ya da rotanın tüm maliyetiyle) çarpılarak zahiri bir ceza maliyeti olarak hesaplanıp amaç fonksiyonunda en küçüklenmesi sağlanmıştır. Bu durumda lojistik işletmelerinin değerlendirmek istediği bu doluluk oranı parametresi de modelde dikkate alınmıştır.



Şekil 3.4. Depodan Tam Kapasite İle Çıkan Kamyon

Aynı örnek üzerinden ortalama doluluk oranının veya ortalama boşluk oranının hesaplanması şu şekilde yapılmaktadır:

$$l_{avg} = \frac{(d_{01} \cdot l_{01}) + (d_{13} \cdot l_{13}) + (d_{34} \cdot l_{34})}{d_{01} + d_{13} + d_{34}} \quad (3.17)$$

l_{avg} , ortalama doluluk oranını, l_{ij} , i ve j noktaları arasındaki doluluk oranını, d_{ij} de i ve j noktaları arasındaki mesafeyi temsil etmektedir. Amaç fonksiyonunda en küçüklenecek değer haliyle $(1 - l_{avg})$ yani ortalama boş kalma oranı olmaktadır. Amaç fonksiyonunda bu değer hesaba katılan zahiri bir ceza maliyeti gibi görülebilir. Doluluk oranını da hesaba katan AUARP'nin önerilen matematiksel modeli aşağıdaki gibidir:

İndisler:

i, j : Talep noktaları ($i, j = 0, 1, \dots, N$) (0 noktası depoyu gösterir.)

k : Araçlar ($k = 1, 2, \dots, K$)

Parametreler:

r_i : son şehri i .şehir olan rotanın standart rota maliyeti

q_i : i noktasının talebi

d_{ij} : i noktası ve j noktası arasındaki mesafe

cap : araçların kapasitesi

ent_{cost} : Bir şehre uğramanın maliyeti

$fuel_{cost}$: km başına yakıt maliyeti

Karar Değişkenleri:

x_{ijk} : $\begin{cases} 1, & i. \text{şehirden } j. \text{şehire } k. \text{araç gidiyorsa} \\ 0, & \text{Diğer durumlarda} \end{cases}$

y_{ik} : $\begin{cases} 1, & i. \text{şehire } k. \text{araç uğruyorsa} \\ 0, & \text{Diğer durumlarda} \end{cases}$

v_k : $\begin{cases} 1, & k. \text{araç kullanılıyorsa} \\ 0, & \text{Diğer durumlarda} \end{cases}$

u_{ik} : k aracının i şehrine uğradıktan sonra kalan yük miktarı

Amaç Fonksiyonu:

$$\begin{aligned} \min \sum_{k=1}^K & \left(\sum_{i=1}^N r_i \left(y_{ik} - y_{ik} \sum_{j=1}^N x_{ijk} \right) + B + ent_{cost} \left(\sum_{i=1}^N y_{ik} - v_k \right) \right. \\ & + \sum_{i=1}^N r_i (y_{ik} \\ & - \sum_{j=1}^N x_{ijk}) \left(\left(\sum_{j=1}^n \sum_{i=0}^n d_{ij} * x_{ijk} (1 - (u_{ik}/cap)) \right) \right. \\ & \left. \left. / \left(\sum_{i=0}^n \sum_{j=0}^n d_{ij} x_{ijk} - v(k) + 1 \right) \right) \right) \quad \forall k, i \neq j \end{aligned} \quad (3.18)$$

Kısıtlar:

$$u_{ik} = \left(\sum_{j=0}^n y_{jk} q_j \right) - \left(\sum_{j=0}^i y_{jk} q_j \right) \quad \forall i, k \quad (3.19)$$

$$\sum_{k=1}^K y_{ik} = 1 \quad \forall i, i \neq 0 \quad (3.20)$$

$$\sum_{i=1}^N q_i y_{ik} \leq \text{cap}(v_k) \quad \forall k \quad (3.21)$$

$$\sum_{i=0}^N x_{ijk} = y_{jk} \quad \forall j, k \quad j \neq 0 \quad (3.22)$$

$$\sum_{j=1}^N x_{ijk} \leq y_{ik} \quad \forall i, k \quad (3.23)$$

$$\sum_{j=1}^N x_{0jk} = v_k \quad \forall k \quad (3.24)$$

$$\alpha_i - \alpha_j + (N + 1) \sum_{k=1}^K x_{ijk} \leq N \quad \forall i, j \quad i \neq j \quad i \neq 0 \quad j \neq 0 \quad (3.25)$$

$$B \geq \text{fuel}_{\text{cost}} \sum_{k=1}^K \left(\sum_{i=0}^N \sum_{j=1}^N d_{ij} x_{ijk} - \sum_{i=1}^N d_{0i} \left(y_{ik} - \sum_{j=1}^N x_{ijk} \right) \right) \quad (3.26)$$

4. GENETİK ALGORİTMA VE YEREL ARAMA ALGORİTMASI

4.1. Genetik Algoritma

Açık uçlu araç rotalama problemlerinin Np-Zor yapıda olmasından dolayı kesin çözüm yöntemleri günümüzün en hızlı bilgisayarları ile nispeten çok büyük boyutlu olmayan problemlerde bile büyük zaman harcamaktadırlar (Talbi, 2009, s. 11). Büyük boyutlu problemler için ise kesin çözüm yöntemleri pratikte kullanışlı değildir. Bu sebeplerden son 30 yılda metasezgisel yöntemler endüstriler için büyük önem kazanmıştır.

Metasezgiseller, kullanıldığı problemin optimum noktasına ulaşmak için ardıştırmalı olarak stokastik ve/veya deterministik kurallarla, çözüm uzayının belirli bölgelerinde arama yapan algoritmalara denir. Sezgisellerin bir üst sınıfı olarak görülebilirler ve sezgiseller gibi bir problem türüne özgü değildir. Farklı problem türlerinde kullanılabilirler ve bu açıdan daha çok bir problem çözme yöntemi olarak adlandırılabilirler. Metasezgiseller en iyiyi garanti etmez ama pratikte elverişli bir zaman harcayarak çözüm uzayını tarayıp yeterince iyi sonuçlar verebilirler. Metasezgiseller açısından en büyük problem yerel en iyiye takılma ihtimalidir. Yerel en iyilerden kaçmak için farklı algoritmalarca farklı yöntemler geliştirilmiştir.

İnsanoğlu, problem çözerken genelde gerçek hayatta karşılaştığı durumlardan esinlenerek çözüme ulaşmaya çalışan bir organizmadır. Kimi zaman içgüdülerimiz problem çözüm yöntemlerimize yön verir. Kimi zaman da doğal habitatımızda yaptığımız gözlemler sonucu elde ettiğimiz mantıksal kararlar ile sonuç bulmaya çalışırız. Arılar birbirleri ile yaptıkları bir dans aracılığıyla kilometrelerce öteden haberleşme yetilerine sahiptir. Karıncalar buldukları yiyecekleri yuvaya taşırken salgıladıkları feromon hormonu sayesinde diğer karıncalara yiyeceğin boyutu ve lokasyonu ile ilgili bilgi verir. Doğada her şey mükemmel bir uyum içerisindedir ve insanoğlu doğadaki yaşanan bu fenomenleri gözlemleyerek problemlerini çözmeye çalışır. Yüzyıllar boyunca büyük akıllar, kuşları inceleyerek insanoğlunun uçma hayalini gerçekleştirmeye çalışmışlardır. Günümüzde doğadan etkilenen binlerce algoritma vardır. Bu algoritmalar doğadaki mükemmel yapıyı taklit etmeye çalışarak sonuca ulaşmaya çalışan

yapılardır. Metasezgisellere son 20 yılda büyük adımlar attıran yaklaşımların birçoğu doğadan etkilenen algoritmalarıdır. Genetik algoritma da bunlardan biridir.

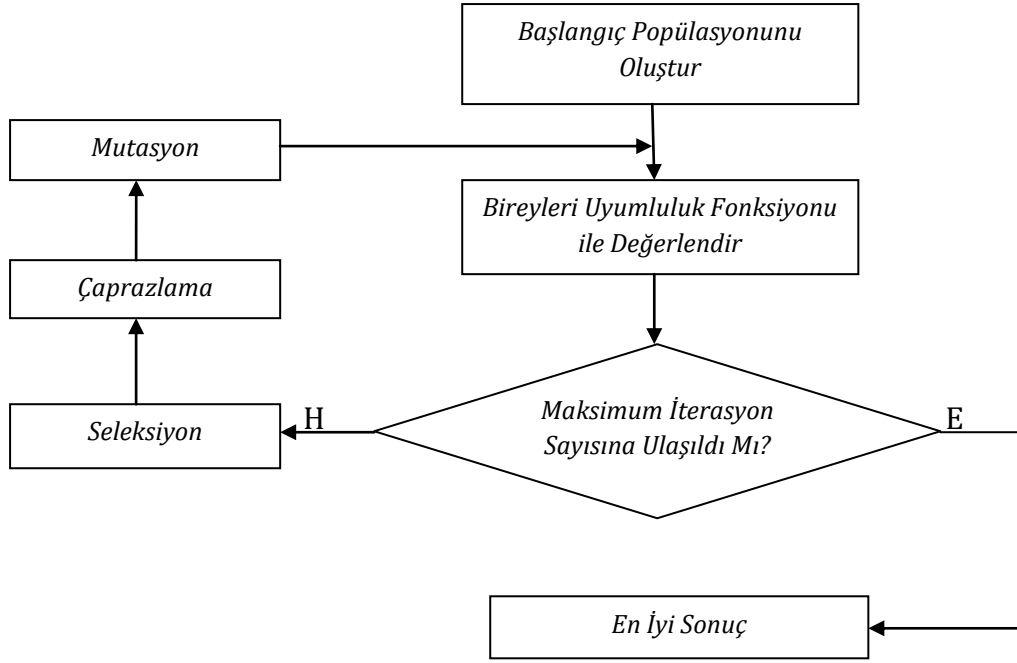
Genetik algoritma, ardıştırmalı metasezgisel arama yöntemlerinden çok noktalı arama yapabilen bir algoritma olarak John Holland (1975, s. 32-159) tarafından geliştirilmiştir. Popülasyon tabanlı bir algoritmadır ve tek noktada arama yerine birden çok noktada arama yaptığından genelde yerel en iyiye sıkışmaz. Arama uzayının büyük ve karmaşık olduğu problemler için etkin çalışır. Algoritmanın etkinliğini sağlayan en önemli faktör ise uyumluluk fonksiyonunun iyi belirlenmesidir. Doğal seleksiyon sürecini taklit eden bu yöntemde ilk aşamada oluşturulan her bir çözüm bir popülasyona ait bireyler olarak görülür ve her jenerasyonda bazı bireyler karakteristik özelliklerine yani uyumluluk fonksiyonu değerlerine ve bazı stokastik etmenlere bağlı olarak yeni jenerasyonlar üretmek üzere seçilir, belirli bir olasılıkla çaprazlanır ve oluşan yeni bireyler popülasyona eklenir. Sonra bazı bireyler yine bazı stokastik etmenlere ve uyumluluk fonksiyonu değerlerine bağlı olarak popülasyondan atılır. Böylece yeni bir jenerasyon oluşmuş olur. Nadiren bazı bireylerin genlerinde beklenmedik değişiklikler olur. Bu olaya mutasyon denir. Mutasyon aramanın yerel en iyilere takılmaması için nadiren de olsa farklı bölgelere de gitmesini sağlayacak bir genetik işlemcidir.

Genetik algoritma yapısı genel anlamda aşağıdaki şekilde ifade edilebilir.

- Arama uzayındaki çözümleri temsil edebilecek bir gösterim yapısının seçilmesi ve bu kodlama yöntemiyle başlangıç popülasyonunu oluşturacak bireylerin (aday çözümlerin) oluşturulması. Başlangıç popülasyonu genellikle rassal olarak seçilen çözümlerden oluşur ve bunlara birey ya da kromozom denir. Başlangıç popülasyonu büyüklüğü önemli bir parametredir ve büyük seçilmesi ilk ardıştırmalarda (jenerasyonlarda) global en iyi sonucun kapsama ihtimalini artırır fakat bu işlem yükünü arttıracaktır.
- Oluşturulan bireylerin uyumluluk fonksiyonu değerleri belirlenir. Genetik algoritmada bir bireyin hayatta kalma ve genlerini sonraki jenerasyonlara aktarma olasılığı o bireyin ne kadar *iyi* olduğu ile doğru orantılıdır. Bir bireyin ne kadar iyi olduğunun ölçüsü ise o uyumluluk fonksiyonunda

verdiği değer bağlıdır. Bu fonksiyon karar modelindeki amaç fonksiyonu ile doğrudan bağıntılıdır.

- Oluşan popülasyon arasından belirli sayıdaki birey diğer bireylerle çaprazlanıp yeni bireyler oluşturmak için seçilecektir. Seçim aşamasında her bireyin seçilme olasılığı olması ile birlikte bu olasılıklar bireylerin uyumluluk fonksiyonu değerlerine göre oranlıdır. Yani daha iyi bireylerin seçilme olasılığı her zaman daha fazladır. Nispeten kötü bireylerin ise seçilme olasılığı daha azdır.
- Seçilen bireyler eşlenerek çaprazlama ve mutasyon süreçlerine dahil olurlar. Çaprazlama süreci iki bireyin genlerinin karşılıklı değişim yoluyla yeni bireyler oluşturmaya denir. Mutasyon ise nadiren gerçekleşir (doğadaki gibi) ve bireylerin bazı genlerinde beklenmedik farklılıklar oluşmasına sebep olur.
- Oluşan yeni bireylerle büyüyen popülasyonun sabit büyüklükte kalması amacı ile popülasyondan bazı bireylerin çıkarılmaları (yok olmaları) gerekmektedir. Bu süreç yer değiştirme olarak tanımlanır. Bazı bireyler ölür (Kötü bireyler daha büyük ihtimalle). Yeni bireyler popülasyona katılır. Yeni nesil oluşmuş olur.
- Seçilme, çaprazlanma, mutasyon ve yer değiştirme işlemleri belirlenen bir ardıştırma sayısı boyunca veya belirlenen başka bir koşul sağlanana kadar tekrar edilir. Böylelikle her ardıştırmada yeni nesiller oluşmuş olur. En iyi birey, algoritmanın verdiği en iyi sonuçtur.



Şekil 4.1. Genetik Algoritma Akış Diyagramı

4.1.1. Gösterim (Encoding)

Genetik algoritmanın kritik konularından bir tanesi kromozomların gösterim şeklidir. Gösterim problem tipine bağlıdır ve problemin çözümünü temsil edecek şekilde olmalıdır. Gösterim, direkt ve endirekt şekilde yapılabilir. Endirekt gösterimde verilen birey doğrudan problemin bir çözümünü temsil etmekle birlikte çözümün kendisi değildir. Çözüme ulaşmak için bir dekoder kullanılır. Endirekt gösterimdeki veriler veya değerler bir dekoderde çeşitli işlemlere uğrayarak problemin çözümü haline gelir. En çok kullanılan genetik algoritma gösterimleri, ikili gösterim ve permütasyon gösterimidir.

4.1.1.1. İkili gösterim

Genetik algoritmanın ilk çalışılan hallerinin bu gösterim ile yapılmasından mütevellit, bu gösterim şekli, en çok kullanılan gösterim şeklidir.

İkili gösterimde, her bir kromozom 1 ya da 0 değerlerini alan bit'lerden oluşur.

Tablo 4.1. *İkili Gösterimli Kromozom Örnekleri*

Kromozom Numarası	Genler
1	1 0 0 1 1 1 0 0 1 0 1 0 0
2	0 1 0 1 1 1 1 0 0 0 1 0 1

Bu bitlerden her biri ya da n kadar grup bit, bir kromozomun bir özelliğini temsil etmektedir. Bu gösterim şeklinin en büyük dezavantajı, çoğu problem tipi için çaprazlama operatörleri veya mutasyon operatörleri uygulandığında sonucun olurlu olabilmesi için düzeltmeler yapılması gerekmesidir.

4.1.1.2. Permütasyon gösterim

En çok kullanılan gösterim şekillerinden biridir ve genelde sıralama problemleri için kullanılır. Gezgin satıcı problemi, araç rotalama problemi gibi sıralama problemleri için oldukça uygun bir gösterimdir.

Tablo 4.2. *Permütasyon Gösterimli Kromozom Örnekleri*

Kromozom Numarası	Genler
1	9 2 10 3 4 6 1 5 7 8
2	1 8 10 6 2 9 3 7 5 4

Bu gösterim türü en çok sıralama problemleri için uygundur. Fakat bu gösterim kullanıldığında eğer sıralama problemin gerçek bir çözümünü temsil ediyorsa, çaprazlama ve mutasyon operatörleri permütasyon gösterime uygun olan şekillerde işlemelidir. Aksi takdirde bu operatörlerin olurlu sonuçlar üretmesi için yine düzeltmeler yapılması gerekir.

4.1.1.3. Değer gösterimi

Bazen problem çözümleri içinde karmaşık verileri barındırır. Bir problemin çözümünde farklı şekiller, reel sayılar ve harfler aynı anda var olabilir. Bu değerler, problem çözümünün doğrudan gerçek verileridir. Bu tip problemlerde ikili gösterim gibi gösterim şekillerini kullanmak hem çok zor hem de hesap süresi

açısından çok maliyetli olabilir. Bu tip problemler için en uygun gösterim şekli çoğu zaman değer gösterimi olur.

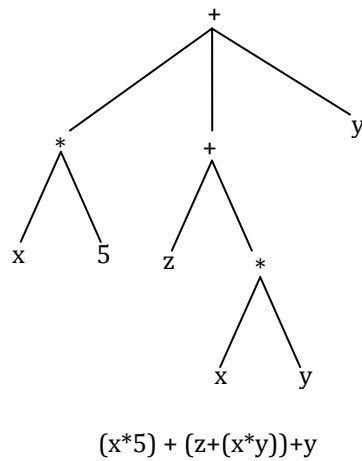
Tablo 4.3. Değer Gösterimli Kromozom Örnekleri

Kromozom Numarası	Genler
1	ABEJGHYUOKLHFNDJKLJNDKLABJ
2	0,13 0,21 0,86 0,01 0,07 1,01 0,74

Bu gösterim şekli genellikle bazı problemler için çok uygun olsa da değer gösterimi kullanıldığında, probleme uygun çaprazlama ve mutasyon operatörleri tanımlamak gerekliliği söz konusudur.

4.1.1.4. Ağaç gösterimi

Genellikle verilen bazı çıktı değerler üzerinden girdi denklemini bulmak üzere tasarlanan problemlerde kullanımı yaygındır. Regresyon denklemlerini temsil eden ağaçlar oluşturma fikri diğer gösterim biçimlerine göre daha avantajlı olmaktadır. Ağaç gösteriminde her bir kromozom, fonksiyonlar veya programlama dilindeki komutlar gibi nesnelerin birer ağacıdır.



Şekil 4.2. Ağaç Gösterimi

4.1.2. Başlangıç Popülasyonunun Oluşturulması

Başlangıç popülasyonunun oluşturulması, algoritmanın kritik adımlarından biridir. Aramanın başlayacağı noktaların yeterince farklı bölgelere dağılması, aramanın çeşitli alanlardan başlaması açısından oldukça önemlidir. İlk bireylerin çeşitlilik sahibi bir gen havuzu oluşturması, gelecek nesillerin bu çeşitlilik içerisinden amaç doğrultusunda uygun genleri kombine etmesi açısından aramanın belirli bir bölgeye sıkışmamasını sağlar. Genel yaklaşım, başlangıç popülasyonunun rassal olarak oluşturulmasıdır. Bu yaklaşım yanlış değildir fakat çeşitliliğin yeterince sağlanıp sağlanmayacağı ve bireylerin çözüm uzayına homojen bir şekilde dağılıp dağılmayacağı meçhul olduğundan, eksik bir yaklaşımdır.

Güçlü bir arama, mümkün merteye çözüm uzayının farklı bölgelerini araştırarak şekilde başlamalıdır. Özellikle algoritmanın ilk adımları, bu çeşitlilik içerisinde başlamalı ve daha sonra arama keskinleşmeli ve bazı umut veren bölgelere odaklanmalıdır. Bununla birlikte aramanın sadece çeşitli olması güçlü olması anlamına da gelmez. Bazı noktaların nispeten iyi uyumluluk değerine sahip olacak şekilde ayarlanması neslin gen kalitesi açısından da önemlidir. Bu açıdan bu çalışmanın başlangıç popülasyonu, bu üç özelliği kapsayacak şekilde meydana gelmektedir.

4.1.3. Uyumluluk fonksiyonu

Metasezgisel ailesinden genetik algoritmada sonuca doğrudan etki eden en kritik konu uyumluluk fonksiyonunun belirlenmesidir. Kuşkusuz, bireylerin karar modelindeki amaç fonksiyonu değeri karşılıkları, uyumluluk fonksiyonunda en büyük paydaya sahip etken olmalıdır. Ancak uyumluluk fonksiyonu değerinin doğrudan amaç fonksiyonu değerine eşit olması da bazen çok anlamlı olmayabilir. Problemin çözüm uzayı yapısına göre bir bireyin ne kadar *iyi* olduğunu belirleyecek faktörler değişebilir. Örneğin, çok fazla yerel en iyinin bulunduğu tepeli bir çözüm uzayı yapısında bir bireyin ne kadar iyi olduğu, onun diğer bireylerden ne kadar farklı olduğu ile de son derece alakalıdır. Çünkü böyle bir problem yapısında çeşitlendirme çok kritik rol oynamaktadır. Bir bireyin uyumluluk fonksiyonu değeri o bireyin ne kadar ardıştırma boyunca popülasyonda

kaldığı ile ters orantılı olarak da düşünülebilir. Uyumluluk fonksiyonundaki bu değişiklik aramanın farklı noktalara doğru yönelmesini sağlar. Ama problem yapısı daha az karmaşıklığa sahipse (kase şeklinde çözüm uzayı) aramanın farklı yerlere yönelmesi istenmeyen bir durum olur. Dolayısıyla, uyumluluk fonksiyonu problem tipine göre belirlenmelidir ama en büyük pay amaç fonksiyonu değerine ait olmalıdır.

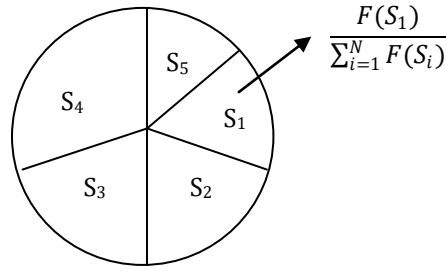
Bu çalışmada uyumluluk fonksiyonu doğrudan amaç fonksiyonu değeri olarak belirlenmiştir. Çeşitlendirme, ileriki bölümlerde anlatılacak prosedürlerle sağlanmıştır.

4.1.4. Seleksiyon

Bu işlem, yeni bireyler oluşturmak üzere var olan popülasyondan bireylerin seçilmesi işlemidir. Seleksiyon, genetik algoritmanın güçlü yapısını oluşturan en önemli özelliklerdendir. Popülasyondaki bütün bireylerin seçilme olasılığı olduğu için bu çeşitliliğin kaybolmamasını, bir diğer deyişle sonuçlarda erken yakınsama yaşanmamasını sağlar. Gayet tabi ki uyumluluk fonksiyonu değeri yüksek olan bireylerin seçilme olasılığı daha büyük olmalıdır. Bu yapıyı sağlayacak farklı yöntemler geliştirilmiştir. Rulet tekeri yöntemi ve turnuva metodu bunlardan en bilinenleridir.

Rulet tekeri yönteminde, her bir birey uyumluluk fonksiyonu değerine göre tekerde bir pay elde eder. Daha iyi değeri olan bireylerin tekerdeki hissesi daha büyük, daha küçük olan bireylerin tekerdeki hissesi daha küçüktür. Rulet tekeri çevrilir (çarkifelek) ve ibrenin gösterdiği birey seçilen bireydir. Tabi ki burada bir bireyin seçilme olasılığı tekerde kapladığı alan ile orantılıdır. Bu işlem seçilmesi gereken birey sayısı elde edilinceye dek tekrarlanır.

Turnuva yönteminde, her defasında rassal olarak iki veya daha çok birey seçilir ve uyumluluk fonksiyonu değeri büyük olan müsabakayı kazanır. Seçilmek istenen birey sayısı kadar müsabaka yapılır.



Şekil 4.3. *Rulet Tekeri Yöntemi*

4.1.5. Çaprazlama

Genetik algoritmada aday çözümler, birer kromozom olarak düşünülmektedir. Aday çözümlerin üzerindeki her bir özelliği temsil eden her bir bölüm de “gen” olarak adlandırılır. İkili gösterim üzerinden örnek vermek gerekirse, 1-0-0-1-1-1-0-0-0-1-0-1 bir problemin çözüm uzayındaki bir çözümü temsil etsin. Bu 0 ve 1’lerin bulunduğu her alan bir *gen* olarak düşünülür. Yani basit bir anlatımla ilk genin değerinin 1 olması bir bireyin göz renginin koyu renklerde olması, 0 olması da açık renklerde olması gibi düşünülebilir. İkinci gen bir bireyin 6 parmaklılık sendromuna sahip olup olmadığını belirleyebilir. Böylelikle her bir genin bir özelliği temsil ettiği düşünülür.

Çaprazlama işleminde, seleksiyon aşamasında seçilen bireylerden ikisi (ebeveynler), rassal olarak belirlenen noktalardan çaprazlanır ve genleri yer değiştirilerek yeni bireyler meydana getirirler.

1. Ebeveyn	1 0 0 1 1 0 0 0 1 0
2. Ebeveyn	1 1 0 1 1 1 0 1 1 1
	⇓
1. Çocuk	1 0 0 1 1 0 0 1 1 1
2. Çocuk	1 1 0 1 1 1 0 0 1 0

Şekil 4.4. *Çaprazlama Örneği*

Her bir birey problemin çözümü için belli nitelikler taşımaktadır. Çaprazlama işleminde bu uygun nitelikler bir araya gelerek daha güçlü bireyler meydana getirirler.

4.1.6. Mutasyon

Mutasyon, beklenmedik bir anda bir dış sebepten bir bireyin genlerinden bazılarının değişime uğraması işlemidir. Bu en basit haliyle ikil gösterimli bir çözümde bir genin 1 iken 0'a dönüşmesi ya da 0'ken 1'e dönüşmesi olarak tanımlanabilir. Bu işlem bireyin bir geninde meydana gelebileceği gibi bireyin birden fazla geni de bozulmaya (değişmeye) maruz kalabilir.

Mutasyon işlemi, genetik algoritmanın etkinliğini sağlayan en önemli faktörlerden biridir. Genel amacı çeşitliliğin sağlanmasıdır. Metasezgiseller için en büyük tehlike olan yerel en iyiye sıkışma durumu genetik algoritmada büyük oranda mutasyon işlemi sayesinde bertaraf edilmektedir. Eğer çözümler aynışmaya başlıyorsa ve erkenden bir noktada yoğunlaşmaya başlıyorsa o bölgeden kurtulmanın tek yolu mutasyon işlemleri ile popülasyonun çeşitliliğini tekrar kazanmasıdır. Bu aramanın tek bir noktaya yoğunlaşmasını engeller.

Tablo 4.4. *Mutasyon*

Kromozom	Genler
Mutasyondan önce	1 1 0 1 1 1 1 0 0
Mutasyondan sonra	1 1 0 0 1 1 1 0 0

4.1.7. Durma kriteri

Metasezgisellerin genel amacı, kesin çözüm yöntemleri ile bir problemi çözmek pratikte anlamlı olmadığında, yani kesin çözüm yöntemleri ile günümüz bilgisayarlarının çözüme ulaşması çok uzun sürdüğünde ya da pratikte mümkün olmadığında, aynı işlemci gücü ile akla yatkın bir süre içinde mantıklı bir sonuç elde edilmesidir. Bu sebeple aramanın belirli koşullar elde edildiğinde durması gerekmektedir. Sonlandırma kriteri basitçe belirlenmiş bir ardışırma sayısından sonra algoritmanın durması olabileceği gibi dinamik koşullara da bağlı olabilir. En sık uygulanan koşul algoritmanın belirli bir ardışırma sayısı boyunca elde edilen en iyi sonuçtan daha iyi bir sonuç elde edememesi durumunda sonlanmasıdır.

4.2. Yerel Arama Algoritması

Yerel arama, muhtemelen en eski ve basit arama algoritmalarından biri olarak bilinmektedir (Talbi, 2009, s. 121). Temel anlamda, bir başlangıç sonucun komşuluk tanımı içindeki diğer komşu çözümlerini araştırıp, aday çözümlerden en iyisini, ilk gelişme yaşanan çözümü ya da gelişme yaşanan çözümler kümesinden rassal olarak birini eldeki çözüm olarak güncelleyen ve eldeki çözümden aynı işlemi tekrarlayarak daha iyi sonuçlara gitmeye çalışan algoritmadır. Her bir ardıştırmada algoritma eldeki çözümden daha iyi bir komşu çözüme gitmeye çalışmaktadır. Daha iyi bir sonuç elde edilemediğinde de algoritma çalışmayı durdurmaktadır. Böylelikle, bir başlangıç çözümden bir yerel en iyiye gidilmesi sağlanmaktadır. Çok büyük bir komşuluk kümesi söz konusu olduğunda, komşuluk tanımı, komşuluk kümesinin bir alt kümesi olarak ayarlanabilmekte ve böylelikle arama daha hızlı bir süreçte ilerleyebilmektedir (Talbi, 2009, s. 122). Yerel aramadaki en önemli konu, komşuluk tanımının etkin bir şekilde yapılabilmesidir. Tanımlanabilecek sınırsız sayıda yerel arama metodu olmakla birlikte, yerel arama algoritmalarının, komşu çözümlerin stokastik veya deterministik kurallara göre türetilmesine veya komşu çözümlerin seçilme stratejisine göre sınıflandırılması mümkündür (Talbi, 2009, s. 122).

Temel olarak yerel arama algoritmaları, kısa sürede iyi sonuçlar veren, uygulaması kolay algoritmalar olarak bilinmektedirler. Bütün arama algoritmalarında olduğu gibi yerel aramada da en kritik konu, yerel en iyilerden kurtulma işleminin etkin bir şekilde yapılabilmesidir. Yerel arama algoritmaları başlangıç çözümler açısından oldukça hassas algoritmalar ve çok yerel en iyisi olan *multimodal* arama uzaylarında iyi sonuçlar vermeleri beklenmemektedir.

4.2.1. Yerel arama hareketleri ve komşuluk tanımı

$G=(S,V)$ olarak tanımlanan bir graf için, S bütün olurlu çözümler kümesi ve V de komşuluk ilişkisi olarak tanımlansın. (i, j) arkları bir s sonucunu s_i ve s_j sonuçlarına bağlayan arklar iken, bir s sonucu için toplam komşu sayısı $N(s)$, s sonucuna bağlı tüm arkların sayısı kadar olur. Bu durumda komşuluk tanımı bir sonuca uygulanabilecek yerel arama hareketlerinin her birinin oluşturacağı yeni sonuçlar kümesi birleşimi olur. Bir diğer deyişle, bir sonucun komşuları, o sonuca

uygulanması mümkün tüm hareketlerin ortaya çıkartacağı yeni sonuçlar bütünüdür.

Yerel aramanın en bilinen hareketlerinden bazıları 1-1 değişim (*swap*), 1-0 değişim (*insert*) ve 2-opt'tur. Bu hareketler farklı rotalar arasında ya da aynı rota içinde uygulanabilen hareketlerdir. Bu hareketlere dair daha ayrıntılı anlatım ve şekiller, Tarantilis ve Kiranoudis'in (2002, s. 87) çalışmalarında bulunabilir.

4.2.1.1. 1-1 değişim (Swap)

1-1 değişim işlemi eğer aynı rota içinde uygulanacaksa, bu aynı rotadaki iki elemanın karşılıklı olarak yer değiştirmesi anlamına gelmektedir. Örneğin, 0 noktası depo olmak üzere, 0-1-2-5-3-4-0 şeklinde oluşan bir rotada, 3. ve 6. elemanların değiştirilmesiyle oluşan yeni rota, 0-1-4-5-3-2-0 olmaktadır.

Eğer rotalararası 1-1 değişim uygulanacaksa, bu işlem, bir rotadaki bir elemanla diğer bir rotadaki bir elemanın karşılıklı değiştirilmesiyle gerçekleştirilir.

Tablo 4.5. 9 ve 1 Talep Noktalarının 1-1 Değişim Hareketi

Numara	1-1 Değişimden Önce	1-1 Değişimden Sonra
1	0 - 4 - 10 - 6 - 9 - 2 - 3	0 - 4 - 10 - 6 - 1 - 2 - 3
2	0 - 1 - 8 - 7 - 5	0 - 9 - 8 - 7 - 5

4.2.1.2. 1-0 değişim (insert)

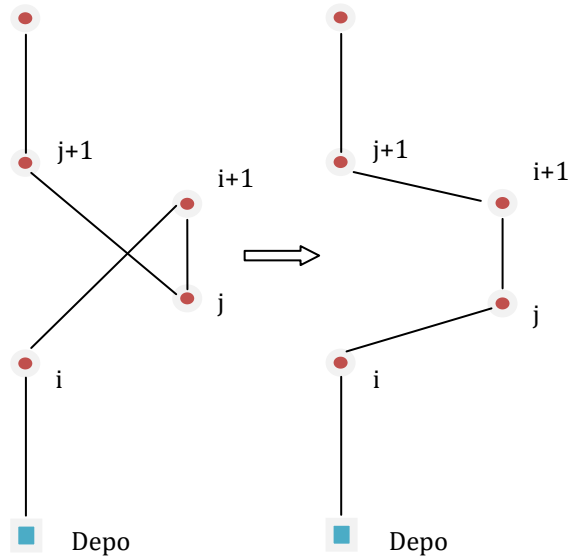
1-0 değişim hareketi de rotalararası ya da aynı rota içinde gerçekleştirilebilmektedir. 1-0 değişim hareketi, rotadaki bir elemanın yerinden kaldırılıp başka bir pozisyona getirilmesi işlemine denir. Örneğin, 0 noktası depoyu temsil etmek üzere, 0-5-1-4-2-3-0 rotasına 2. elemanın 5. pozisyona getirilmesi işlemi uygulanırsa, 0-1-4-5-2-3-0 rotası ortaya çıkmaktadır. 1-0 değişim hareketinin, yerel arama için oldukça etkin çalıştığı bilinmektedir.

Tablo 4.6. 5 Numaralı Talep Noktasının 1-0 Değişim Hareketi

Numara	1-0 Değişimden Önce	1-0 Değişimden Sonra
1	0 - 1 - 2 - 3 - 4 - 5 - 6	0 - 1 - 2 - 3 - 4 - 6
2	0 - 10 - 9 - 8 - 7	0 - 10 - 5 - 9 - 8 - 7

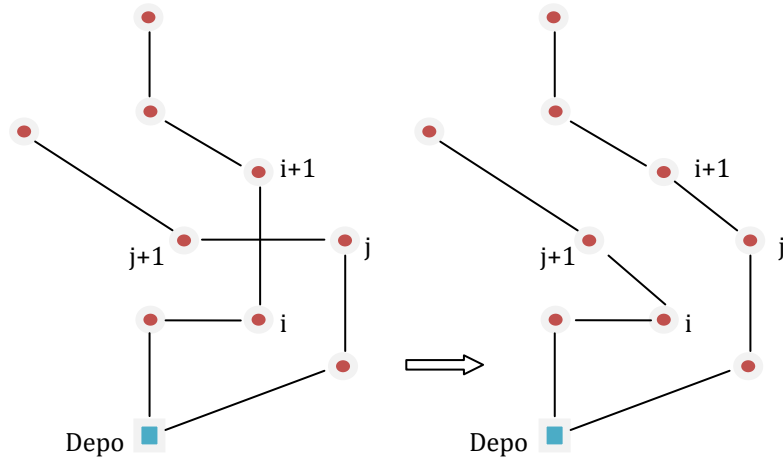
4.2.1.3. 2-Opt

2-Opt hareketi de, 1-1 değişim ve 1-0 değişim hareketleri gibi aynı rotada uygulanabildiği gibi rotalararası da uygulanabilen bir yerel arama yöntemidir. 2-Opt hareketi temel anlamda iki düğümün bağlı olduğu diğer düğümlerle bağlantılarının kesilip bu bağlantıların karşılıklı olarak değiştirilmesi hareketine denmektedir. Aynı rota içinde bir $i+1$ talep noktasına bağlı bir i talep noktası ve bir $j+1$ noktasına bağlı bir j talep noktası olduğunu düşünürsek, i noktasını j noktasına, $i+1$ noktasını da $j+1$ noktasına bağlayan hareket 2-Opt hareketine örnek olarak verilebilir.



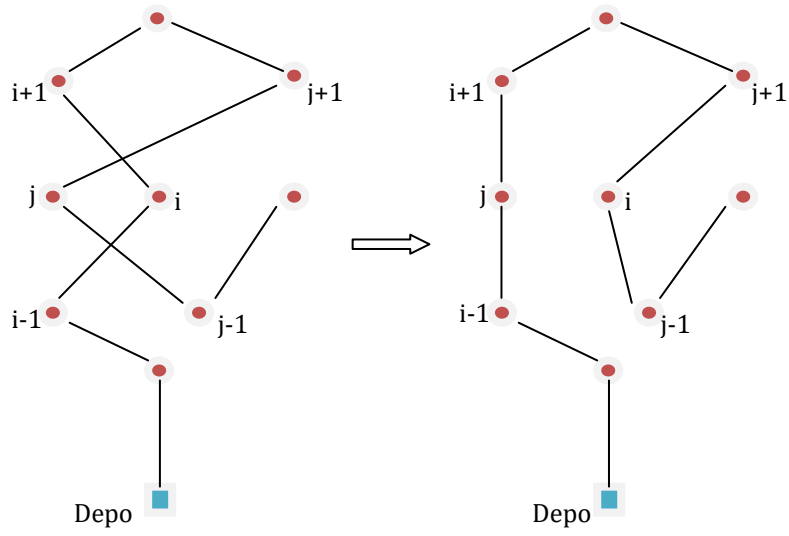
Şekil 4.5 Aynı Rota Üzerinde 2-Opt Hareketi

Kaynak: Tarantilis ve Kiranoudis, 2002, s. 87



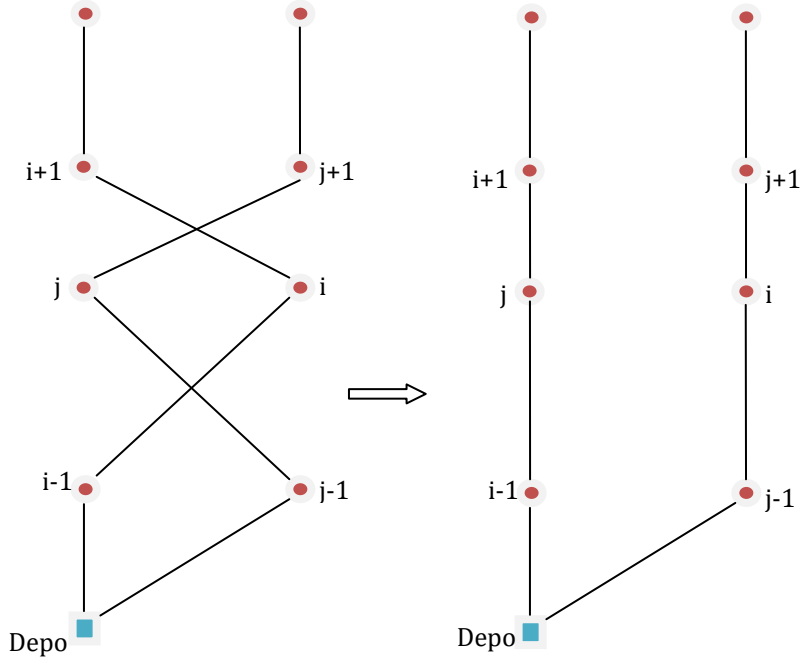
Şekil 4.6 Farklı Rotalar Üzerinde 2-Opt Hareketi

Kaynak: Tarantilis ve Kiranoudis, 2002, s. 87



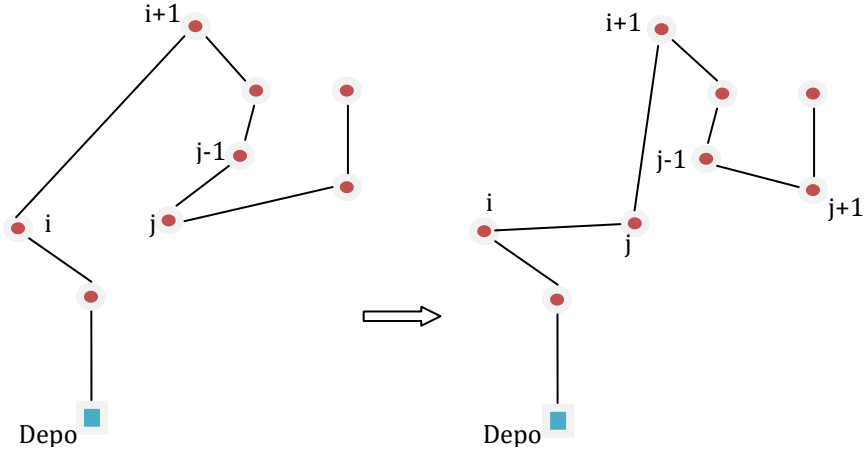
Şekil 4.7 Aynı Rota Üzerinde 1-1 Değişim Hareketi

Kaynak: Tarantilis ve Kiranoudis, 2002, s. 88



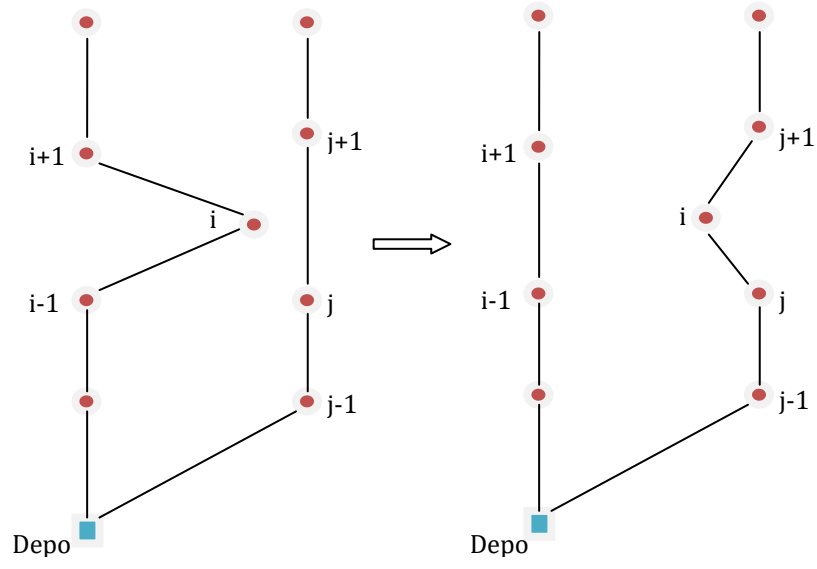
Şekil 4.8 Farklı Rotalar Üzerinde 1-1 Değişim Hareketi

Kaynak: Tarantilis ve Kiranoudis, 2002, s. 88



Şekil 4.9 Aynı Rota Üzerinde 1-0 Değişim Hareketi

Kaynak: Tarantilis ve Kiranoudis, 2002, s. 88



Şekil 4.10 Farklı Rotalar Üzerinde 1-0 Değişim Hareketi

Kaynak: Tarantilis ve Kiranoudis, 2002, s. 88

5. ÖNERİLEN MELEZ GENETİK ALGORİTMA MODELİ

Genetik algoritma ortaya atıldığı ilk yıllardan bu yana birçok farklı iyileştirme ile geliştirilmeye çalışılan, metasezgisel ailesinin en köklü ve popüler uygulamalarından biri olarak bilinmektedir. Doğal seleksiyon ve reproduksiyon tekniklerini temel alan genetik algoritmalar, kompleks çözüm uzaylarını etkin bir şekilde çözmesiyle bilinmesine rağmen, araç rotalama problemleri alanında genetik algoritmalar pek iyi çalışmayabilmektedirler (Lui, Jiang ve Geng, 2012, s. 405). Bununla birlikte, genetik algoritmalar çeşitli melezleme uygulamalarıyla araç rotalama problemleri için de iyi çalışır hale getirilebilmektedirler.

Bu çalışmada Lui, Jiang ve Geng'in (2012, s. 404) çoklu depolu açık uçlu araç rotalama problemleri için önerdiği melez genetik algoritma modeli, literatürde henüz bu çalışmadaki haliyle ele alınmayan bir yöntem ile iyileştirilmeye çalışılmıştır. Eshelmen ve Schaffer (1991, s. 116), bildirimlerinde, bireyler arasındaki enest ilişkileri kısıtlayan bir eşik değeri tanımlamıştır. Galán, Mengshoel ve Pinter (2013, s. 200) genleri benzer olan bireyler arasındaki eşleşme durumunu kısıtlayan bir genetik algoritma modeli önermişlerdir. Bu çalışmada ise kan bağı olan bireyler arasındaki ilişki doğrudan yasaklamıştır. Bu çalışmada önerilen model, "**toplumlaşmış genetik algoritma**" olarak isimlendirilmiştir. Toplumlaşmış genetik algoritmanın klasik genetik algoritmadan temel farkı, bireyler arasındaki kan bağının bellekte tutulması ve kan bağı olan bireylerin birbirleriyle eşlenmemesinin sağlanmasıdır. Bu özellik sayesinde, Lui, Jiang ve Geng'in yerel arama ile keskinleştirme yönü güçlendirilmiş modelinin, çeşitlendirme yönünün güçlendirilmesi hedeflenmiştir.

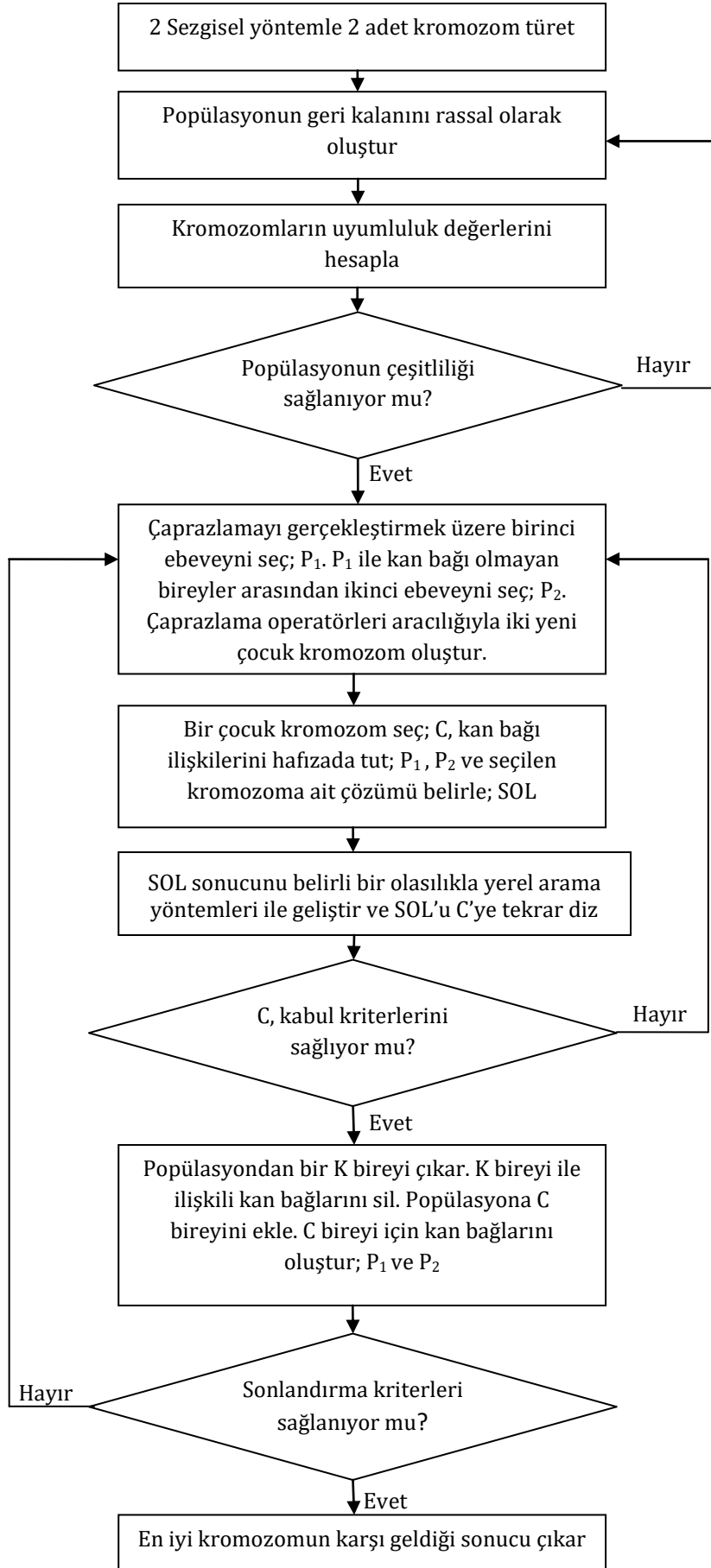
Gerçek hayatta akraba evliliklerinin çocuklarda sakat doğumlara ve ölümlere yol açmasının sebebi, kan bağı olan bireylerde bulunabilecek benzeri çekinik kötü genlerin çaprazlama sonrasında birleşerek hastalık gibi kötü özellikleri ortaya çıkarma ihtimali olarak bilinmektedir. Bu riskten korunmak için bireylerin kan bağı olmayan kişilerle eşleşmesi gelişmiş toplumlarca bir norm haline gelmiştir. Gelişmiş toplumlar, bireylerdeki aynı genlerin tekrarlanması sonucunda ortaya çıkabilecek olumsuzlukları engellemek için bu normu uygulamışlardır. Genetik algoritmada da korunması gereken en temel özelliklerden biri popülasyondaki bireylerin mümkün oldukça birbirlerinden farklı olması ve erken yakınsama olarak

bilinen durumun önlenmesidir. Bu durum, kısaca bireyler arasında çeşitliliğin sağlanması ile mümkün olmaktadır. Benzeri bir şekilde toplumlaşmış genetik algoritma, kan bağı olan bireylerin eşleşmesini engelleyerek meydana gelen yeni bireylerin benzeri gen özelliklerine sahip ebeveynlerden oluşmamasını, yani çeşitliliğin korunmasını sağlamaya çalışmaktadır.

5.1. Toplumlaşmış Genetik Algoritma (TGA)

Toplumlaşmış genetik algoritmanın ilk adımı başlangıç popülasyonunun oluşturulmasıdır. Daha sonra başlangıç popülasyonundaki bireylerin her birinin uyumluluk fonksiyonu değeri bulunur. Bireylerinin kan bağlarının tutulacağı liste başlatılır. Bu listede her birey adına iki ebeveyn, bir kardeş ve altı çocuk için yer tutulur. Daha sonra seleksiyon işleminde 2 adet birey çaprazlanmak üzere seçilir. Seçim aşamasında ilk birey, popülasyondan rassal olarak seçilen 3 bireyin turnuva yöntemi aracılığıyla uyumluluk fonksiyonu değeri en yüksek olan bireyin bulunması ile seçilir. İkinci birey, ilk birey ile kan bağı olmayan bireyler arasından ve ilk bireyin kendisi de hariç tutulmak suretiyle, yine aynı şekilde 3 bireyin seçilip turnuva metoduyla en iyi uyumluluk fonksiyonu değerine sahip olan bireyin bulunması ile seçilir. İlk ardıştırmanın öncesinde herhangi bir kan bağı olmadığı için seçilecek ikinci birey, sadece seçilen ilk bireyin olmadığı bireyler arasından seçilir. Seçilen iki birey çaprazlanır ve böylece iki yeni birey meydana gelir. Meydana gelen yeni bireylerden biri rassal olarak seçilir ve diğeri sonlandırılır¹. Seçilen yeni bireyin yerel arama yöntemleri ile bir yerel en iyiye gitmesi sağlanır. Popülasyondan bir birey eksiltir ve yeni birey popülasyona dahil edilir. Popülasyondan atılan bireyin diğer bireylerle kan bağları kaldırılır ve yeni birey için ilgili kan bağları oluşturulur. Kan bağlarının tutulduğu liste güncellenir. Sonlandırma kriteri veya maksimum ardıştırma sayısına ulaşıncaya kadar evrim süreci tekrar edilir. Aşağıda Şekil 4.7.'de geliştirilen algoritmanın akış şeması bulunmaktadır.

¹ Bu durum bireyler arasında kardeşlik ilişkilerinin hafızada tutulamamasına sebep olmaktadır.



Şekil 5.1. Toplumlaşmış Genetik Algoritma Akış Şeması

5.1.1. Çözüm gösterimi ve açılması

Önerilen genetik algoritmada bireyler, araç rotalama problemleri için en uygun gösterim olarak bilinen permütasyon gösterim ile gösterilir. Arada rota sonlarını belirten herhangi bir işaret yoktur. Bütün bir kromozom içinde bütün turları kapsayacak şekilde talep noktalarının numaraları ile sıralanır. Her bir kromozomun bir uyumluluk fonksiyonu değerinin hesaplanması için önce kromozomun bir sonucu temsil edecek şekilde açılması gerekmektedir. Kromozomların temsil ettiği sonuç, söz konusu kromozomun elemanları bir depodan başlamak üzere dizildiğinde, dizinin talep toplamları araç kapasitesini aştığı anda araya bir depo göstergesinin² gelmesi ve talebin aşılmasını sağlayan ilk elemanın yeni rotanın ilk elemanı olacak şekilde dizilme işleminin devam etmesi ile bulunur. Dizilecek talep noktaları bittiğinde her rota bir depo noktasından başlayacak şekilde oluşturulmuş olur.

Uyumluluk fonksiyonu değeri bu çalışmada önerilen şekilde gerçek hayat maliyetleri toplamıdır. Bu maliyetler, daha önceki bölümlerde de bahsedildiği gibi standart rota maliyetleri, rotadan sapma maliyetleri ve uğrama maliyetlerinin toplamıdır. Doluluk oranını dikkate alan modelde ise aynı maliyetlere, rotaları izleyecek araçların ortalama boş kalma oranlarının rotanın toplam maliyetiyle çarpılması ile oluşan zahiri bir maliyetin eklenmesi de gerekmektedir.

5.1.2. Popülasyon yapısı ve başlangıç

Toplumlaşmış genetik algoritmanın başlangıç popülasyonunda ilk iki kromozom, iki farklı sezgisel yöntemle bulunmaktadır. İlk uygulanan sezgisel, ARP uygulamalarında sıkça uygulanan en yakın komşu sezgiselidir. Uygulanan ikinci sezgisel ise Clarke ve Wright'ın (1964, s. 569) savings algoritmasıdır. Geri kalan kromozomlar ise talep noktası numaralarının rassal olarak dizilmesiyle bulunmaktadır. Algoritmanın tamamında aynı çözümlere izin verilmemektedir. Dolayısıyla başlangıç popülasyonda eğer iki kromozom birbirinin aynı uyumluluk değerine sahipse kromozomlardan biri popülasyondan atılmakta ve yerine başka

² Depo göstergesi, "0", "Depo" gibi depoyu işaret edebilecek herhangi bir simge olarak alınabilir.

bir rassal çözüm koyulmaktadır. Herhangi iki kromozomun uyumluluk fonksiyonu değeri birbiriyle aynı olmayıncaya kadar bu işleme devam edilmektedir.

5.1.3. Seleksiyon ve çaprazlama

Önerilen melez genetik algoritmada seçilecek 2 ebeveynden toplam 2 yeni birey oluşturulur. Ebeveynlerden ilki olarak atanacak ilk kromozom, turnuva metoduyla seçilmektedir. Turnuva metodu için öncelikle rassal olarak 3 adet kromozom seçilmektedir ve içlerinden en büyük uyumluluk fonksiyonu değerine sahip olan kromozom, turnuvanın kazananı olarak ilk ebeveyn olarak atanmaktadır. İkinci ebeveyni seçmek için öncelikle ilk ebeveyn ile birinci dereceden kan bağı olan bireylerin turnuvaya katılmaması sağlanmaktadır. Böylelikle, ilk ebeveyn ile benzeri gen dizilimlerine sahip bir ikinci ebeveynin seçilmemesi sağlanmaktadır. İlk bireyin kendisi ve ilk birey ile kan bağı olan kromozomlar hariç tutulmak üzere, yine 3 adet kromozom seçilmekte ve en büyük uyumluluk fonksiyonu değerine sahip kromozom, ikinci ebeveyn olarak atanmaktadır. Burada turnuva büyüklüğü önemli bir parametredir. Turnuva büyüklüğü arttıkça uyumluluk fonksiyonu değeri nispeten düşük olan kromozomların seçilme olasılığı azalmaktadır. Önerilen genetik algoritmada Lui, Jiang ve Geng'in (2012, s. 408) çalışmasında önerdiği turnuva büyüklüğü parametresi kullanılmıştır.

Önerilen melez genetik algoritmada 3 farklı çaprazlama operatörü kullanılmıştır. Bu operatörler permütasyon gösterim için geliştirilmiş *2-Point Crossover*, *Order Crossover* ve *Partially Mapped Crossover* operatörleridir. Bu çaprazlama operatörleri ile ilgili ayrıntılı bilgi El-Ghazali Talbi'nin (2009, s. 218) çalışmalarında bulunabilir. Algoritmadaki her ardıştırmada rassal olarak bir çaprazlama operatörü seçilmektedir. Her bir çaprazlama operatörünün seçilme olasılığı eşit olarak belirlenmiştir.

Türetilen 2 yeni kromozomdan rassal olarak biri seçilmektedir ve bu kromozom, belirli bir olasılıkla mutasyon operatörlerine maruz kalmaktadır. Diğeri ise doğrudan popülasyon dışında kalmaktadır.

5.1.4. Mutasyon

Toplumlaşmış genetik algoritmanın mutasyon parametresi, Lui, Jiang ve Geng'in (2012, s. 409) çalışmasındaki gibi 0,25 olarak belirlenmiştir. Çaprazlama operatörleriyle oluşturulan iki yeni kromozomdan birisi 0,25 olasılıkla mutasyon işlemine maruz kalmaktadırlar.

Basit mutasyon operatörleri ile uygulanan klasik anlamdaki genetik algoritmanın, tabu arama algoritması ve tavlama benzetimi gibi modern çalışmalara göre daha kötü sonuçlar verdiği bilinmektedir (Lui, Jiang ve Geng, 2012, s. 409). Bu nedenle, bu çalışmada genetik algoritma, yerel arama yöntemleri ile melezlenmiş ve daha iyi sonuçların türetilmesi beklenmiştir. Bu çalışmada önerilen melez genetik algoritmada, çaprazlama operatörlerinden gelen yeni kromozomun öncelikle gerçek bir çözümü temsil edecek şekilde açılmış hale getirilmesi gerekmektedir. Daha sonra uyumluluk fonksiyonu değeri hesaplanmakta ve sırasıyla 1-1 değişim, 1-0 değişim ve 2-Opt yerel arama yöntemleri ile daha iyi sonuçlar aranmaktadır. Yerel arama stratejisi olarak en iyi sonuç veren komşu çözümü kabul etme stratejisi değil, daha iyi çözüm veren ilk komşu çözümü kabul etme stratejisi benimsenmiştir. Yani, elimizdeki çözüm daha iyi bir sonuç veren bir çözüm ile her güncellendiğinde yerel arama işlemi en başa dönmekte ve eldeki çözüm tanımlanan komşuluk üzerinde daha iyi bir sonuç vermeyinceye kadar yerel arama işlemlerine devam edilmektedir. Bütün komşu çözümler tarandığında ve daha iyi bir sonuç bulunamadığında mutasyon işlemi sona ermektedir. Son olarak güncellenen çözüm, depoyu gösteren elemanlar silinerek tekrar kapalı hale getirilmekte ve çaprazlamadan gelen yeni birey güncellenmektedir.

5.1.5. Algoritmaya dair değişkenler ve durma kriteri

Genetik algoritmanın etkinliğindeki en önemli faktörlerden biri popülasyon çeşitliliğinin korunmasıdır (Hertz ve Widmer, 2003, s. 251; Lui, Jiang ve Geng, 2012, s. 409). Popülasyon çeşitliliğini korumak ve erken yakınsama durumunun engellenmesi adına önerilen melez genetik algoritmada iki önemli adım uygulanmıştır. Bunlardan ilki, popülasyondaki bireylerin birbirleri ile aynı olmamasının sağlanması, diğeri de çaprazlanan bireylerin birinci derece kan

bağları (ebeveyn ve çocukları) hafızada tutularak kan bağı olan bireylerin eşlenmesinin önlenmesidir.

Prins (2004, s. 1996) ve Lui, Jiang ve Geng'in (2012, s. 409) çalışmaları ışığında popülasyon büyüklüğü nispeten küçük tutulmuş ve 30 olarak belirlenmiştir. Mutasyon parametresi önceden de belirtildiği gibi 0,25 olarak alınmıştır. İki adet durma kriteri üzerinde çalışılmıştır. Bunlardan biri eldeki en iyi çözümün güncellenmediği ardıştırma sayısı, diğeri de algoritmanın çalıştığı toplam ardıştırma sayısıdır. Her yeni bir çaprazlama gerçekleştiğinde algoritmada ardıştırma sayısı bir artmaktadır.

5.2. Test Problemleri İle Karşılaştırmalar

Kapasite kısıtlı ARP için nispeten AUARP'ye göre daha çok test problemi seti olduğu için öncelikle, önerilen toplulaşmış genetik algoritmanın kapasite kısıtlı ARP'ye uygun bir versiyonu geliştirilmiştir. Karşılaştırma 2 adet genetik algoritma modeli üzerinden yapılmıştır. Bunlardan ilki Lui, Jiang ve Geng'in (2012, s. 404) önerdiği modele oldukça yakın bir modeldir ve tabloda LJG-HGA olarak gösterilmektedir. Lui, Jiang ve Geng'in modelinde kromozomların bir çözüm haline gelmesi için *split* adlı algoritma çalışmaktadır. Bu çalışmada karşılaştırılan iki modelde ise kromozomlar açık hale bu algoritma kullanılmadan getirilmiştir. Lui, Jiang ve Geng'in (2012, s. 404) çalışmasında başlangıç popülasyonun oluşturulması aşamasında 3 adet sezgisel yöntemle 3 adet nispeten iyi uyumluluk fonksiyonu değeri veren kromozomlar türetiliyorken bu çalışmada karşılaştırılan modellerde 2 adet sezgiselle nispeten iyi çözümler türetilmiştir. Bu özellikler dışında karşılaştırmaya tabi ilk algoritma, Lui, Jiang ve Geng'in (2012, s. 404) algoritma adımlarını izlemektedir. İkinci algoritma ise bu çalışmada tanıtılan toplulaşmış genetik algoritmadır ve tablolarda TGA olarak geçmektedir.

Bu alanda iki test problemi serisi seçilmiştir. Bunlardan ilki Augerat ve diğerlerinin literatüre sunduğu test problemlerinden 27 problemlik A setidir. Karşılaştırma için seçilen diğer test problemi serisi ise Christofides, Mingozzi ve Toth'un (1979, s. 512) literatüre tanıttığı 14 problemden yükleme-boşaltma zamanı ve rotaların maksimum süresi üst limiti olmayan 1. (vrpnc1) , 2. (vrpnc2), 3. (vrpnc3), 4. (vrpnc4), 5. (vrpnc5), 11. (vrpnc11) ve 12. (vrpnc12) test

problemleridir. Maksimum rota süresi bu çalışmanın kapsamında olmadığı için geri kalan 7 test problemi dikkate alınmamıştır. Aşağıda Tablo 5.1. ve Tablo 5.2.'de test problemleri karşılaştırmaları yukarıda tanıtıldığı sırayla yer almıştır.

Tablo 5.1. Augerat vd. (Set A) ARP Test Problemleri Karşılaştırma Sonuçları

Test Problemi	N	LJG-HGA				TGA				Bilinen En İyi
		LJG-HGA _{eniği}	LJG-HGA _{ort}	t _{eniği} (CPU)	Yaklaşma	TGA _{eniği}	TGA _{ort}	t _{eniği} (CPU)	Yaklaşma	
A-n32-k5	32	784*	784*	4	%0	784*	784*	3	%0	784
A-n33-k5	33	661*	661*	2	%0	661*	661*	3	%0	661
A-n33-k6	33	742*	742*	3	%0	742*	742*	3	%0	742
A-n34-k5	34	778*	778*	7	%0	778*	778*	4	%0	778
A-n36-k5	36	803	806,1	21	%0,50	799*	799*	8	%0	799
A-n37-k5	37	669*	669*	6	%0	669*	669*	5	%0	669
A-n37-k6	37	949*	949*	6	%0	949*	949*	6	%0	949
A-n38-k5	38	730*	731,2	25	%0	730*	730*	6	%0	730
A-n39-k5	39	825	825	9	%0,36	822*	823	15	%0	822
A-n39-k6	39	831*	833	29	%0	831*	832,3	114	%0	831
A-n44-k7	44	937*	938,5	25	%0	937*	938,7	14	%0	937
A-n45-k6	45	948	954,1	201	%0,42	948	951,3	111	%0,42	944
A-n45-k7	45	1146*	1146*	16	%0	1146*	1146*	9	%0	1146
A-n46-k7	46	917	917,1	8	%0,32	914*	914*	11	%0	914
A-n48-k7	48	1074	1079,6	64	%0,09	1073*	1073,3	26	%0	1073
A-n53-k7	53	1017	1019,8	37	%0,69	1017	1017	25	%0,69	1010
A-n54-k7	54	1167*	1169,3	62	%0	1167*	1167*	12	%0	1167
A-n55-k9	55	1078	1082	54	%0,46	1075	1075	34	%0,18	1073
A-n60-k9	60	1354*	1358,1	275	%0	1354*	1357	280	%0	1354

Tablo 5.1. (Devam) Augerat vd. (Set A) Arp Test Problemleri Karşılaştırma Sonuçları

A-n62-k8	62	1283	1284	77	-%0,38	1262	1281	210	-%2,01	1288
A-n63-k9	63	1627	1633,3	408	%0,68	1627	1633,7	145	%0,68	1616
A-n63-k10	63	1320	1326	434	%0,45	1318	1324,2	148	%0,30	1314
A-n64-k9	64	1416	1424,2	159	%1,07	1415	1418,4	126	%0,99	1401
A-n65-k9	65	1181	1183,3	85	%0,59	1178	1178,1	44	%0,34	1174
A-n69-k9	69	1171	1173	240	%1,03	1168	1169,6	59	%0,77	1159
A-n80-k10	80	1740	1746,6	742	-%1,30	1738	1745,8	331	-%1,42	1763

Tablo 5.2. Christofides, Mingozi ve Toth ARP Test Problemleri Karşılaştırma Sonuçları

Test Problemi	N	LJG-HGA				TGA				Bilinen En İyi
		LJG-HGA _{eniyi}	LJG-HGA _{ort}	t _{eniyi} (CPU)	Yaklaşma	TGA _{eniyi}	TGA _{ort}	t _{eniyi} (CPU)	Yaklaşma	
vrpnc1	50	524,61*	524,61*	13	%0	524,61*	524,61*	16	%0	524,61
vrpnc2	75	841,97	845,39	240	%0,80	835,89	838,65	126	%0,07	835,26
vrpnc3	100	827,39	830,90	2050	%0,15	828,42	831,07	1918	%0,28	826,14
vrpnc4	150	1043,37	1047,47	3653	%1,45	1041,73	1044,87	1460	%1,29	1028,42
vrpnc5	199	1328,97	1339,87	3732	%2,91	1320,66	1342,34	3746	%2,27	1291,29
vrpnc11	120	1042,11*	1042,73	267	%0	1042,11*	1042,11*	81	%0	1042,11
vrpnc12	100	822,33	822,33	216	%0,33	821,46	822,12	200	%0,23	819,56

Tablo 5.1. ve Tablo 5.2.'deki karşılaştırmalarda, her bir test problemi, LJG-HGA ve TGA için 10'ar kere olmak üzere, aynı parametrelerle çalıştırılmış ve sonuçlar kaydedilmiştir. Aynı test problemi için kullanılan bütün parametreler LJG-HGA ve TGA için aynıdır. Bütün problemler için başlangıç popülasyon büyüklüğü 30, çaprazlama parametresi 1, mutasyon parametresi 0,25 seçilmiştir. Algoritmaların sonlandırma kriteri "gelişme olmayan a ardıştırma sonra" veya "toplamda b ardıştırma sonra" şeklinde tasarlanmıştır ($b > a$). Her bir test problemi için iki algortmada da a ve b aynı değerlerdir.

Tablo 5.1. ve Tablo 5.2.'deki karşılaştırmalardan görüleceği üzere toplam 34 test probleminin 15'inde bu çalışmada önerilen TGA'nın bulduğu en iyi sonuç, LJG-HGA'nın bulduğu en iyi sonuçtan daha iyidir. 18 test probleminde bulunan en iyi sonuçlar birbirinden farklılaşmamıştır. Buna karşılık 34 test probleminin 1'inde LJG-HGA, TGA'dan daha iyi bir sonuç üretmiştir.

34 test probleminin 20'sinde TGA'nın bulduğu sonuçların ortalaması, LJG-HGA'nın bulduğu sonuçların ortalamasından üstündür. 10 test probleminde iki algoritmanın bulduğu sonuçların ortalaması açısından bir farklılaşma olmamıştır. Buna karşılık, toplam 4 test probleminde LJG-HGA'nın sonuçlarının ortalaması TGA'dan üstün gelmiştir.

Test problemlerinin 24'ünde TGA, bulduğu en iyi sonucu LJG-HGA'ya göre daha hızlı üretmiştir (Tablo 5.1. ve Tablo 5.2.'da süreler saniye cinsinden verilmiştir). Hız açısından 8 test probleminde de LJG-HGA üstün gelmiştir.

Hem LJG-HGA hem de TGA aynı 2 test probleminde bilinen en iyi sonuçları geride bırakmıştır. Bilinen en iyi sonucun geliştirildiği test problemlerinin ikisinde de TGA'nın bulduğu sonuç, LJG-HGA'nın bulduğu sonuca üstün gelmiştir.

Sonuç olarak TGA, hem bulunan en iyi çözüm açısından, hem bulunan sonuçların aritmetik ortalaması bakımından, hem de sonuçların bulunma süresi açısından LJG-HGA'yı geride bırakmıştır. 2 test probleminde bilinen en iyi sonuç iyileştirilmiştir. Algoritmaların ikisinin de özellikle daha büyük ölçekli problemlerde daha iyi çalıştığı gözlenmiştir. Çözüm uzayının çok karmaşık olduğu bazı problemlerde de önerilen algoritmanın kabul edilebilirlik sınırları içerisinde nispeten yavaş çalıştığı gözlemlenmiştir.

5.3. Geliştirilen Karar Destek Sistemi

Bu çalışmanın kapsamında, bu çalışmada tanıtılan toplulaşmış genetik algoritmayı esas alan, Türkiye'deki illerden biri depo seçilmek üzere Türkiye'deki diğer illere açık uçlu araç rotalamasının etkin bir şekilde yapılabileceği bir karar destek sistemi (KDS) geliştirilmiştir. Bir KDS oluşturmaktaki temel amaç, bu projenin çıktılarının endüstriye tanıtılması ve endüstride uygulamalarının yapılabilmesidir.

KDS'nin arayüzü, oldukça kullanıcı dostu ve basit olmakla birlikte bir çok parametrenin değiştirilmesine de olanak vermektedir. Geliştirilen yazılımında kullanıcı ilk olarak verileri sisteme girmek durumundadır. Girilecek veriler, depo (şehir), sipariş noktaları ve o sipariş noktalarına özgü standart rota maliyetleri, kilometre başına yakıt maliyeti ve uğrama maliyetidir. Geliştirilen yazılımın iki ana en küçükleme modu vardır. Bu en küçükleme modlarından, "Gerçek Hayat Maliyetleri" veya "Kat Edilen Toplam Mesafe"den biri seçilmek suretiyle en küçükleme yöntemi belirlenebilmektedir. Bu projenin kapsamında tanıtılan gerçek hayat maliyetlerini kullanıp sipariş noktalarına göre en maliyetsiz rotaları bulmaya çalışmak ya da literatürde tanıtıldığı şekilde kat edilen mesafeyi en küçükmeye çalışmak kullanıcının insiyatifine bırakılmıştır.

Kullanıcı arayüzünün alt bölümünde, KDS'nin arka planında çalışan toplulaşmış genetik algoritmanın parametreleri değiştirilebilmektedir. Değiştirilebilir olarak ayarlanan parametreler, başlangıç popülasyonu büyüklüğü, çaprazlama parametresi ve mutasyon parametresidir. Bunun dışında bu çalışma kapsamında tanıtılan yerel arama metotlarından swap (1-1 değişim), insert (1-0 değişim) ve 2-opt yöntemlerinin her birinin de aktifleştirilip aktifleştirilmeyeceği kullanıcı tarafından belirlenebilmektedir. Bu durum, özellikle çok sipariş noktası olan problemler için programın çalışma süresinin kritik olduğu durumlar için düşünülmüştür. Çıkarılan her yerel arama metodu, programın daha hızlı çalışmasını sağlayacaktır. Fakat karşılığında çözüm kalitesinin düşmesi de daha muhtemel olacaktır. Yine programın aynı bölümünden "Akıllı Seeding Elemanları" da pasifleştirilebilir olarak ayarlanmıştır. Buradan savings algoritması ve/veya en yakın komşu sezgiseli ile türetilen çözümler iptal edilip popülasyonun geri

kalanının rassal olarak oluşturulması sağlanabilir. Aşağıda görsel 5.1.'de geliştirilen arayüzün bir ekran görüntüsü paylaşılmıştır.

Sipariş Noktaları	Talepler	Standart Rota Maliyetleri
ESKİŞEHİR	0	0
ANKARA	10	500
SİVAS	15	600
ERZURUM	25	800
ADANA	25	700

Görsel 5.1. Geliştirilen Karar Destek Sisteminin Arayüzü

Hipotetik bir problem, Geliştirilen KDS'nin iki ana moduyla (kat edilen mesafe en küçüklenerek ve gerçek hayat maliyetleri en küçüklenerek) iki farklı şekilde çözülmüştür ve sonuçlar karşılaştırılmıştır.

5.3.1. Hipotetik bir problem

Eskişehir'deki bir lojistik departmanı, bağlı oldukları *Devrim Ltd. Şti.*'nin ürünlerinin dağıtımının yapılması adına lojistik destek hizmeti veren *A-logistics* adlı başka bir firmadan araç kiralamaktadır. Bu lojistik departmanı, ürünlerinin sevkiyatı için *A-logistics* adlı firmanın ücretlendirme sistemine göre en maliyetsiz açık uçlu rotaların oluşturulması ile yükümlüdürler.

Tablo 5.3.'te *Devrim Ltd. Şti.*'nin talep noktalarının bulunduğu şehirler ve palet sayısı cinsinden talep sayıları verilmiştir. *A-logistics*, söz konusu lojistik departmanının oluşturduğu rotalar üzerinden *Devrim Ltd. Şti.*'ni ücretlendirecektir. *A-logistics*, belirlenen talep noktalarına doğrudan gidiş için ücretlendirmeyi Tablo 5.4.'teki gibi belirlemiştir. *A-logistics*'in Tablo 5.4.'te belirlediği fiyatlar, talep noktalarının standart ücretleridir. Bu fiyat tablosu, bir araç başka hiçbir yere uğramadan doğrudan Eskişehir'deki depodan talep noktasına gittiği zaman ortaya çıkacak olan standart ücrettir. *A-logistics*, *Devrim Ltd. Şti.*'nin oluşturduğu her bir rotadaki en pahalı standart ücrete sahip talep noktasını **standart rota ücreti** olarak belirlemektedir. Söz konusu standart rota ücretine neden olan noktaya doğrudan gitmeme bedelini de **sapma bedeli** olarak belirlemektedir. Yani, oluşturulan rotada toplam kat edilen mesafe ve o noktaya doğrudan giderken kat edilecek mesafe arasındaki farkın, kilometre başına yakıt maliyeti ile çarpılması, sapma bedeli olarak ücretlendirilmektedir. Bunun dışında *A-logistics*, oluşturulan bir rotada standart rota ücretine tabi nokta dışında dağıtım yapılan tüm noktalar için bir **uğrama bedeli** talep etmektedir.

Bu durumda *Devrim Ltd. Şti.* için maliyetler **standart rota ücretleri**, **sapma bedelleri** ve **uğrama bedelleri** olmaktadır. *A-logistics*, sapma bedeli için kat edilen her bir ekstra km'yi 0,25 TL'lik bir yakıt maliyetiyle ücretlendirmektedir. Uğrama bedeli her bir şehir için sabit ve 30 TL'dir. Kiralanan araçlar tek tip araçlardır ve araçların kapasitesi 40 palettir.

Örnek:

Eskişehir - Manisa - İzmir - Aydın rotası için Tablo 5.4.'e göre standart ücreti en pahalı olan nokta Aydın'dır (915 TL). Dolayısıyla belirlenen rotanın **standart rota ücreti**, 915 TL'dir.

Rotada kat edilen toplam mesafe 556 km'dir (395 km + 35 km + 126 km). Eğer Eskişehir'den Aydın'a direkt gidilseydi kat edilecek toplam mesafe 478 km olduğundan, Bu rota için **sapma bedeli**,

$$(556 \text{ km} - 478 \text{ km}) \times 0,25 \frac{\text{TL}}{\text{km}} = 19,5 \text{ TL olur.}$$

Belirlenen rotada Aydın'a giderken iki nokta'ya daha dağıtım yapılmaktadır. Bu noktalar için de *Devrim Ltd. Şti.* $30 \times 2 = 60$ TL'lik bir uğrama bedeline maruz kalmaktadır.

Belirlenen rotanın *Devrim Ltd. Şti.*'ne toplam maliyeti:

$$915 \text{ TL} + 19,5 \text{ TL} + 60 \text{ TL} = 994,5 \text{ TL} \text{ olmaktadır.}$$

Tablo 5.3. ve Tablo 5.4.'teki veriler ve problem tanımında belirtilen uğrama bedeli, km başına yakıt maliyeti ve araç kapasitesi parametreleri bu çalışma kapsamında geliştirilen karar destek sisteminde sırasıyla "Kat edilen mesafe" ve "Gerçek hayat maliyetleri" en küçüklenecek şekilde 2 kere çözülmüştür. Bu 2 ayrı sonucun verdiği rotaların, *Devrim Ltd. Şti.*'ne maliyeti, *A-logistics*'in belirlediği ücretlendirme sistemine göre değerlendirilmiş ve karşılaştırma yapılmıştır.

Tablo 5.3. *Devrim Ltd. Şti.*'nin talep noktaları ve talep miktarları

Numara	Talep Noktaları	Talep Miktarı (Palet)
0	Eskişehir (Depo)	0
1	Ankara	10
2	Sivas	15
3	Erzurum	25
4	Adana	25
5	Kilis	12
6	Amasya	10
7	Samsun	7
8	Erzincan	6
9	Manisa	11
10	İzmir	22
11	Aydın	7
12	Konya	9
13	Antalya	13

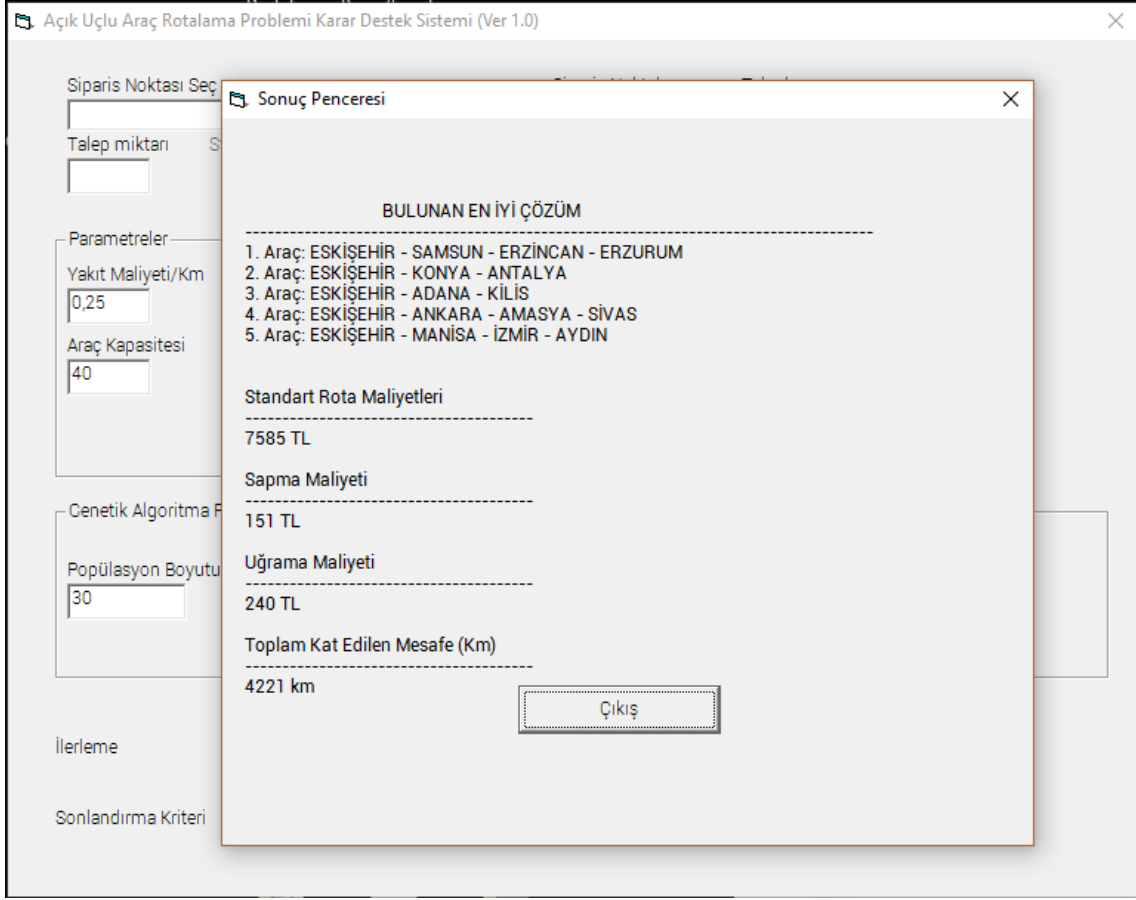
Tablo 5.4. Talep Noktaları İçin A-Logistics'in Belirlediği Standart Ücretler

Numara	Talep Noktaları	Std. Ücret (TL)
0	Eskişehir (Depo)	0
1	Ankara	490
2	Sivas	1450
3	Erzurum	2350
4	Adana	1400
5	Kilis	1990
6	Amasya	1200
7	Samsun	1370
8	Erzincan	1960
9	Manisa	820
10	İzmir	875
11	Aydın	915
12	Konya	720
13	Antalya	880

Bu çalışma kapsamında geliştirilen karar destek sisteminde "Kat edilen toplam mesafe" modunun seçilmesi ile program çalıştırılmış ve Görsel 5.2.'deki sonuç elde edilmiştir. Görsel 5.3.'teki sonuç ise programın "Gerçek hayat maliyetleri" modu seçilerek aynı verilerle tekrar çalıştırılması sonucu elde edilen sonuçtur. Bu sonuçlara göre, eldeki verilerle "Kat edilen toplam mesafe" en küçüklendiğinde *Devrim Ltd. Şti.*'nin maruz kalacağı maliyetler ve "Gerçek hayat maliyetleri" en küçüklendiğinde *Devrim Ltd. Şti.*'nin maruz kalacağı maliyetler Tablo 5.5.'te gösterilmiştir.

Tablo 5.5. Devrim Ltd. Şti. İçin İki Ayrı Durumdaki Maliyetler

Mod	Mesafe (km)	Maliyetler			
		Std. Ücreti	Rota Bedeli	Sapma Bedeli	Ugrama Bedeli
Enk(kat edilen toplam mesafe)	4221	7585	151	240	7976
Enk(gerçek hayat maliyetleri)	4581	7375	258,5	240	7873,5

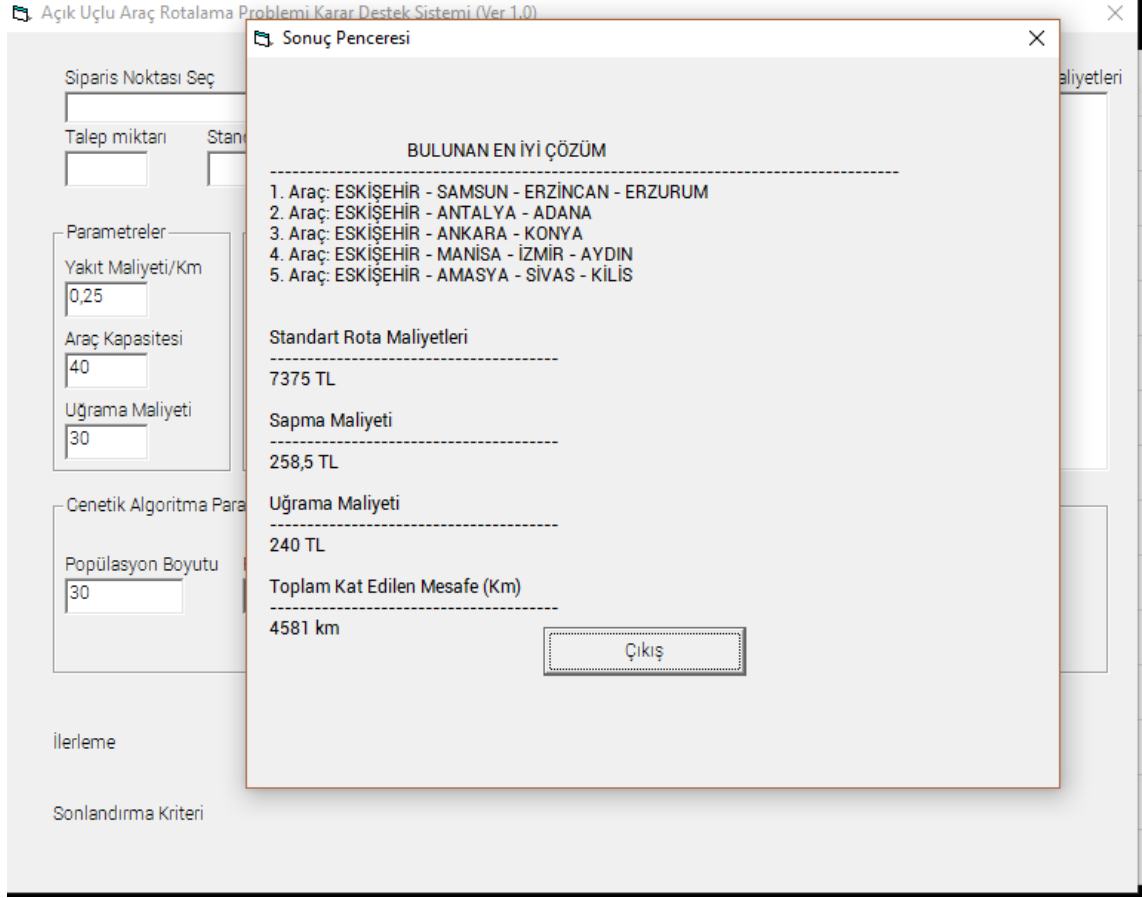


Görsel 5.2. Kat Edilen Toplam Mesafenin En Küçüklenmesi ile Bulunan Sonuç

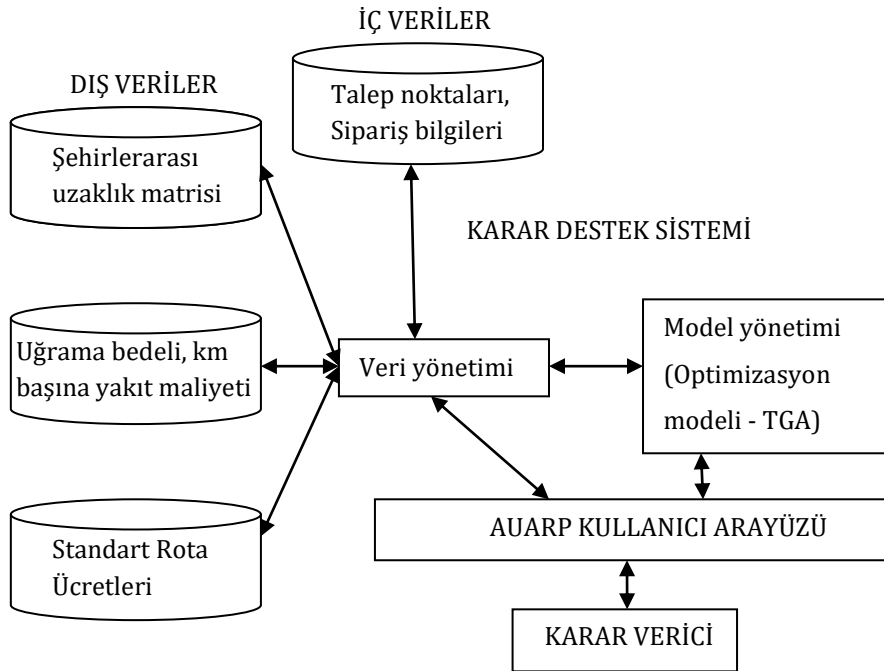
Geliştirilen KDS iki durumda da (genetik algoritma parametreleri dahil) tamamen aynı parametrelerle çalıştırılmıştır.

Buna göre, eğer *Devrim Ltd. Şti.* bu problemi literatürdeki yaklaşımlarla çözmek isterse ekstra 102,5 TL'lik bir maliyete maruz kalır. Geliştirilen KDS'nin "Gerçek hayat maliyetleri" modu seçildiğinde oluşan güzergahların toplam maliyeti 7873,5 TL iken, "Kat edilen mesafe" modu ile oluşan rotaların toplam maliyeti, 7976 TL'dir.

Bu kurgusal problemin çözümü, mesafeden bağımsız maliyetler söz konusu olduğunda literatürdeki "kat edilen toplam mesafenin en küçüklenmesi" yaklaşımının doğru sonuçlar vermeyebileceğini göstermektedir.



Görsel 5.2. Gerçek Hayat Maliyetlerinin En Küçüklenmesi İle Bulunan Sonuç



Şekil 5.2. Karar Destek Sistemi Bileşenleri

6. SONUÇ VE ÖNERİLER

Bu çalışma kapsamında, literatüre yeni maliyet kalemleri tanıtılmıştır. Bu maliyet kalemlerinin uygulamada lojistik firmalarına daha maliyetsiz rotalar olarak geri dönüşünün olması beklenmektedir. Standart rota maliyetleri, sapma maliyetleri ve uğrama maliyetleri aracılığıyla kat edilen toplam mesafeden bağımsız maliyetlerin de hesaba katılması sağlanmıştır. Literatüre tanıtılan bu maliyetlendirme yöntemi sayesinde, firmalar talep noktalarına giderken en kısa yolu değil, yerine göre en uygun (hızlı) veya en maliyetsiz yolları tercih edebileceklerdir. Bu çalışma, bu maliyetlendirme yöntemi ile literatüre katkı yapmıştır.

Bu çalışmanın çıktılarında birisi de, modern metasezgiseller karşısında etkinliğini yitirmekte olan genetik algoritma ile çağdaş metasezgiseller arasındaki farkın kapanmasını sağlayacak yeni bir melez genetik algoritma modeli olan toplulaşmış genetik algoritmanın literatüre tanıtılmasıdır. Bu algorithmada tanıtılan birinci dereceden kan bağı ilişkisi olan bireylerin çaprazlanmaması kuralı sayesinde Liu, Jiang ve Geng'in (2012, s. 404) literatüre tanıttığı melez genetik algoritmanın çeşitlendirme yönü kuvvetlendirilerek algoritma etkinliği arttırılmıştır. Test problemleri ile yapılan karşılaştırmalarda, bu çalışmada önerilen toplulaşmış genetik algoritma, daha kısa sürede daha iyi çözümler üretebilmesiyle ön plana çıkmıştır.

Liu, Jiang ve Geng'in (2012, s. 404) önerdiği algoritma her ne kadar bu çalışmanın metasezgisel kısmına taban sağlasa da, o algorithmadaki bazı kısımlar bu çalışmada göz önüne alınmamıştır. Bunlardan ilki, Liu, Jiang ve Geng'in (2012, s. 404) önerdiği algorithmada başlangıç popülasyonu oluşturulurken 3 adet sezgisel ile 3 *nispeten iyi* çözüm türetilmekte ve başlangıç popülasyonunda iyi genlere sahip olan kromozomların bulunması sağlanmaktadır. 2012'de tanıtılan algorithmada savings algoritması, sweep algoritması ve ilk ikisinden daha karmaşık bir üçüncü sezgisel ile 3 adet çözüm oluşturulmaktadır. Bu çalışmada önerilen toplulaşmış genetik algorithmada ise 2 adet daha basit sezgisel yöntem kullanılmıştır. Bu durumun ileriki çalışmalar için üzerine düşünülmesi gereken bir unsur olduğunu söylemek doğru olacaktır. Ayrıca bahsedilen çalışmada kromozomların çözüm

haline getirilmesi için kullanılan *split* algoritmasının da sonuçlara olumlu etki edeceği düşünülmektedir. *Split* algoritması bu çalışmada kullanılmamıştır.

Bu çalışmada geliştirilen eşleme yasaklama sisteminde kardeşlik ilişkileri geliştirilen algoritma adımları gereği hafızada tutulamamaktadır. Çünkü çaprazlama sonucu oluşan iki yeni kromozomun biri doğrudan popülasyon dışında kalmaktadır. Oluşan yeni bireylerin ikisinin de tutulduğu algoritmalarda bu çalışmada anlatılan yasaklı eş kavramının çeşitlendirme açısından daha etkili olabileceği düşünülmektedir.

KAYNAKÇA

- Aksen, D., Özyurt, Z. and Aras, N. (2007). The open vehicle routing problem with driver nodes and time deadlines. *Journal of the Operational Research Society*, 58(9), 1223–1234.
- Baños, R., Ortega, J., Gil, C., Márquez, A. L., De Toro, F. (2013). A hybrid meta-heuristic for multi-objective vehicle routing problems with time windows. *Computers and Industrial Engineering*, 65(2), 286–296.
- Bodin, L. D., Golden, B. L., Assad, A., Ball, M. (1981). *The state of the art in the routing and scheduling of vehicles and crews*. Washington, DC: US Department of Transportation.
- Brandão, J. (2004). A tabu search algorithm for the open vehicle routing problem. *European Journal of Operational Research*, 157(3), 552–564.
- Brito, J., Martinez, F. J., Moreno, J. A., Verdegay, J. L. (2015). An ACO hybrid metaheuristic for close-open vehicle routing problems with time windows and fuzzy constraints. *Applied Soft Computing Journal*, 32, 154–163.
- Chrisofides N., Mingozzi A. and Toth P. (1979) *Combinatorial Optimization*. Chichester: John Wiley.
- Clarke, G. U. and Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12(4), 568-581.
- Dantzig, G. B. and Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, 6(1), 80-91.
- Derigs, U. and Reuter, K. (2009). A simple and efficient tabu search heuristic for solving the open vehicle routing problem. *Journal of the Operational Research Society*, 60, 1658–1669.
- Erbao, C. and Mingyong, L. (2010). The open vehicle routing problem with fuzzy demands. *Expert Systems with Applications*, 37(3), 2405–2411.
- Eshelmen, L. J. and Schaffer, J. D. (1991). Preventing premature convergence in genetic algorithms by preventing incest. *Proceedings of the fourth international conference on genetic algorithms*, 115-121.

- Fleszar, K., Osman, I. H. and Hindi, K. S. (2009). A variable neighbourhood search algorithm for the open vehicle routing problem. *European Journal of Operational Research*, 195(3), 803–809.
- Fu, Z., Eglese, R. and Li, L. Y. O. (2005). A new tabu search heuristic for the open vehicle routing problem. *Journal of the Operational Research Society*, 56(3), 267–274.
- Galán, S. F., Mengshoel, O. J. and Pinter, R. (2013). A novel mating approach for genetic algorithms, *Evolutionary computation*, 21(2), 197-229.
- Hertz, A. ve Widmer, M. (2003). Guidelines for the use of meta-heuristics in combinatorial optimization. *European Journal of Operational Research*, Cilt(151), 247-252.
- Holland, J. H. (1975). *Adaptation in natural and artificial systems*. Ann Arbor: University of Michigan Press.
- Kirby, D., (1959). Is your fleet the right size?. *Operational Research Quarterly*, 10(4), 252.
- Laporte, G. (1992). The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(3), 345-358.
- Li, F., Golden, B. and Wasil, E. (2007). The open vehicle routing problem: Algorithms, large-scale test problems, and computational results. *Computers and Operations Research*, 34(10), 2918–2930.
- Liu, R., Jiang, Z. ve Geng, N. (2012). A hybrid genetic algorithm for the multi-depot open vehicle routing problem. *Springer-Verlag*, 36(2), 401-421.
- López-Sánchez, A. D., Hernández-Díaz, A. G., Vigo, D., Caballero, R., Molina, J. (2014). A multi-start algorithm for a balanced real-world Open Vehicle Routing Problem. *European Journal of Operational Research*, 238(1), 104–113.
- MirHassani, S. A. and Abolghasemi, N. (2011). A particle swarm optimization algorithm for open vehicle routing problem. *Expert Systems with Applications*, 38(9), 11547–11551.
- Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers and Operations Research*, 31(12), 1985-2002.

- Repoussis, P. P., Tarantilis, C. D. and Ioannou, G. (2007). The open vehicle routing problem with time windows. *Journal of Operational Research Society*, 58(3), 355-367.
- Sariklis, D. and Powell, S. (2000). A heuristic method for the open vehicle routing problem. *Journal of the Operational Research Society*, 51(5), 564–573.
- Schrage, L. (1981). Formulation and structure of more complex/realistic routing and scheduling problems. *Networks*, 11(2), 229-232.
- Talbi, E. -G. (2009). *Metaheuristics: From design to implementation*. New Jersey: John Wiley & Sons.
- Tarantilis, C. D., Ioannou, G., Kiranoudis, C. T., Prastacos, G. P. (2005). Solving the open vehicle routing problem via a single parameter metaheuristic algorithm. *Journal of the Operational Research Society*, 56(5), 588-596.
- Tarantilis, C. D. ve Kiranoudis, C. T. (2002). Distribution of fresh meat. *Journal of Food Engineering*, 51(1), 85-91.
- Tonbul, E., Tuna G., Erginel, N. and Alpaslan, M. (2015). Solving open vehicle problems with real life costs via genetic algorithm. 27th European conference on operational research.
- Tonbul, E., Tuna G. and Erginel, N. (2016). A new approach: Maximizing load ratios of vehicles to maintain profitability in open vehicle routing problems. 23rd international academic conference.
- Yu, S., Ding, C. and Zhu, K. (2011). A hybrid GA-TS algorithm for open vehicle routing optimization of coal mines material. *Expert Systems with Applications*, 38(8), 10568–10573.
- Yu, V. F., Jewpanya, P. and Redi, A. A. N. P. (2016). Open vehicle routing problem with cross-docking. *Computers and Industrial Engineering*, 94, 6–17.
- Zachariadis, E. E. and Kiranoudis, C. T. (2010). An open vehicle routing problem metaheuristic for examining wide solution neighborhoods. *Computers and Operations Research*, 37(4), 712–723.
- <http://www.kgm.gov.tr/Sayfalar/KGM/SiteTr/Root/Uzakliklar.aspx> (Erişim Tarihi: 21.12.2016)

EK-1. Müşteri Sayısı 5 Olan Bir Hipotetik Problem

Talep noktaları uzaklık matrisi (km)

	0	1	2	3	4	5
0	0	336	573	966	611	490
1	336	0	909	646	632	755
2	573	909	0	1310	589	256
3	966	646	1310	0	738	1054
4	611	632	589	738	0	333
5	490	755	256	1054	333	0

Standart rota maliyetleri (TL)

0	0
1	672
2	1146
3	1932
4	1222
5	980

Talepler (br)

0	0
1	11
2	6
3	8
4	7
5	7

Araç kapasitesi=20 br

Uğrama maliyeti=36 TL

Km başına yakıt maliyeti=1,5 TL

EK-2. Müşteri Sayısı 10 Olan Bir Hipotetik Problem

Talep noktaları uzaklık matrisi (km)

	0	1	2	3	4	5	6	7	8	9	10
0	0	336	573	966	611	490	557	1034	883	901	770
1	336	0	909	646	632	755	893	755	1219	1237	1059
2	573	909	0	1310	589	256	292	1237	346	328	212
3	966	646	1310	0	738	1054	1429	397	1642	1571	1360
4	611	632	589	738	0	333	825	696	931	833	622
5	490	755	256	1054	333	0	544	981	598	535	315
6	557	893	292	1429	825	544	0	1466	342	505	476
7	1034	755	1237	397	696	981	1466	0	1579	1464	1253
8	883	1219	346	1642	931	598	342	1579	0	296	520
9	901	1237	328	1571	833	535	505	1464	296	0	246
10	770	1059	212	1360	622	315	476	1253	520	246	0

Standart rota maliyetleri (TL)

0	0
1	672
2	1146
3	1932
4	1222
5	980
6	1114
7	2068
8	1766
9	1802
10	1540

Talepler (br)

0	0
1	11
2	6
3	8
4	7
5	7
6	5
7	4
8	9
9	12
10	10

Araç kapasitesi=20 br

Uğrama maliyeti=36 TL

Km başına yakıt maliyeti=1,5 TL

EK-3. Müşteri Sayısı 15 Olan Bir Hipotetik Problem

Talep noktaları uzaklık matrisi (km)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	336	573	966	611	490	557	1034	883	901	770	632	732	677	666	839
1	336	0	909	646	632	755	893	755	1219	1237	1059	349	412	946	1002	1128
2	573	909	0	1310	589	256	292	1237	346	328	212	1100	1292	420	170	277
3	966	646	1310	0	738	1054	1429	397	1642	1571	1360	358	234	1147	1425	1420
4	611	632	589	738	0	333	825	696	931	833	622	640	832	409	755	682
5	490	755	256	1054	333	0	544	981	598	535	315	897	1089	191	422	384
6	557	893	292	1429	825	544	0	1466	342	505	476	1189	1289	684	122	541
7	1034	755	1237	397	696	981	1466	0	1579	1464	1253	406	562	1040	1403	1313
8	883	1219	346	1642	931	598	342	1579	0	296	520	1410	1602	718	272	445
9	901	1237	328	1571	833	535	505	1464	296	0	246	1421	1613	424	397	151
10	770	1059	212	1360	622	315	476	1253	520	246	0	1201	1393	213	354	95
11	632	349	1100	358	640	897	1189	406	1410	1421	1201	0	196	1049	1193	1270
12	732	412	1292	234	832	1089	1289	562	1602	1613	1393	196	0	1241	1385	1462
13	677	946	420	1147	409	191	684	1040	718	424	213	1049	1241	0	562	273
14	666	1002	170	1425	755	422	122	1403	272	397	354	1193	1385	562	0	419
15	839	1128	277	1420	682	384	541	1313	445	151	95	1270	1462	273	419	0

Standart rota maliyetleri (TL)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	672	1146	1932	1222	980	1114	2068	1766	1802	1540	1264	1464	1354	1332	1678

Talepler (br)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	11	6	8	7	7	5	4	9	12	10	3	1	6	15	8

Araç kapasitesi=20 br

Uğrama maliyeti=36 TL

Km başına yakıt maliyeti=1,5 TL