

T.C.
İNÖNÜ ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ



**MALATYA BÜYÜKŞEHİR BELEDİYESİ İLAÇLAMA
ARAÇLARININ ROTALARININ METASEZGİSEL
ALGORİTMALARLA BELİRLENMESİ**
DOKTORA TEZİ

DANIŞMAN HAZIRLAYAN
Dr.Öğr.Üyesi Hasan SÖYLER Eda FENDOĞLU

Malatya - 2019

T.C.
İNÖNÜ ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ

MALATYA BÜYÜKŞEHİR BELEDİYESİ İLAÇLAMA
ARAÇLARININ ROTALARININ METASEZGİSEL
ALGORİTMALARLA BELİRLENMESİ

EKONOMETRİ ANA BİLİM DALI

DOKTORA TEZİ

Eda FENDOĞLU

MALATYA, 2019

T.C.
İNÖNÜ ÜNİVERSİTESİ
SOSYAL BİLİMLERİ ENSTİTÜSÜ

**MALATYA BÜYÜKŞEHİR BELEDİYESİ
İLAÇLAMA ARAÇLARININ ROTALARININ
METASEZGİSEL ALGORİTMALARLA
BELİRLENMESİ**

DOKTORA TEZİ

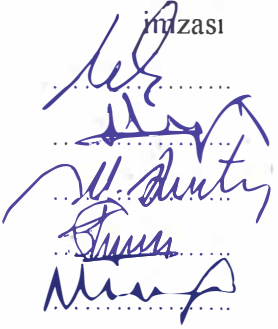
DANIŞMAN
Dr.Öğr.Üyesi Hasan SÖYLER

HAZIRLAYAN
Eda FENDOĞLU

Jürimiztarihinde yapılan savunma sınavı sonucunda bu doktora tezini (oybirliği / oyçokluğu) ile başarılı bulunarak Ekonometri Anabilim dalı Ekonometri Bilim dalında doktora tezi olarak kabul edilmiştir.

Jüri Üyelerinin Unvan Ad Soyadı

1. Prof. Dr. Mehmet GÜNGÖR
2. Dr.Öğr.Üyesi Hasan SÖYLER (Tez Danışmanı)
3. Dr.Öğr.Üyesi Mustafa DESTE
4. Dr.Öğr.Üyesi Oktay KIZILKAYA
5. Dr.Öğr.Üyesi Miraç EREN

imzası


İnönü Üniversitesi Sosyal Bilimler Enstitüsü Yönetim Kurulunun tarih ve sayılı kararıyla bu tezin kabulü onaylanmıştır.

Prof. Dr. Mehmet KUBAT
Sosyal Bilimler Enstitüsü Müdürü

ONUR SÖZÜ

Dr. Öğr. Üyesi Hasan SÖYLER'in danışmanlığında İnönü Üniversitesi Lisansüstü Eğitim ve Öğretim Yönetmeliğinin ilgili hükümleri uyarınca doktora tezi olarak sunduğum "MALATYA BÜYÜKŞEHİR BELEDİYESİ İLAÇLAMA ARAÇLARININ ROTALARININ METASEZGİSEL ALGORİTMALARLA BELİRLENMESİ" başlıklı bu tezin kendi çalışmam olduğunu, bilimsel ahlak ve geleneklere aykırı düşecek bir yardıma başvurmaksızın tarafımdan yazıldığını, sunduğum bütün sonuç, doküman, bilgi ve belgeleri bu tez çalışması kapsamında elde ettiğimi, elde edilemeyen ve yararlandığım bütün bilgi ve yorumlara atıf yaptığımı ayrıca kaynakçada yöntemine uygun biçimde gösterildiğini, bu tezin herhangi bir bölümünü bu üniversitede veya başka bir üniversitede diğer bir tez çalışması içinde sunmadığımı, bütün kısımlarında bilimsel etik kurallarına uyduğumu belirtir, bunu onurumla doğrularım.

Tarih: .01.2019

Ad-Soyad: Eda FENDOĞLU

İmza:

TEŞEKKÜR

Tez çalışmam boyunca öneri, bilgi ve desteğini esirgemeyen değerli hocam ve tez danışmanım Dr. Öğr. Üyesi Hasan SÖYLER' e,

Tezin en başından sonuna kadar yönlendirimlerini, tecrübelerini ve bilgilerini esirgemeyen değerli hocalarım Prof. Dr. Mehmet GÜNGÖR ve Dr. Öğr. Üyesi Mustafa DESTE' ye,

Tanımaktan büyük onur duyduğum, tez sürecimde bilgi ve önerilerini paylaşarak, desteklerini esirgemeyen Doç. Dr. Yalçın ALVER' e, Arş. Gör. M. Metin MUTLU' ya, Arş. Gör. M. Akif FINDIKLI' ya, Malatya Çevre Koruma ve Kontrol Dairesi Şube Müdürü M. Zeki DUMAN' a ve Maski CBS Şube Müdürü Yusuf KOCAMAN' a,

Sadece bu tez süreci boyunca değil, hayatımın her alanında, her konuda beni daima destekleyen, bu hayatta sevginin, merhametin ve iyiliğin önemini her şeyden önce benimseyerek bugünlere gelmemi sağlayan canım anneme, babama ve bir dost gibi hep yanımda olan biricik ağabeyime ve çok uzun yıllardır bizimle olduğu için ailemin bir parçası olup, bana huzur veren kedim Draşan' a,

Sonsuz TEŞEKKÜR EDERİM.

ÖZET

MALATYA BÜYÜKŞEHİR BELEDİYESİ İLAÇLAMA ARAÇLARININ ROTALARININ METASEZGİSEL ALGORİTMALARLA BELİRLENMESİ

Bu tez çalışmasında, Malatya Büyükşehir Belediyesi'nin ilaçlama araçlarının rotalarının optimizasyonu için metasezgisel yöntemler ile çok aşamalı bir uygulama geliştirilmiş ve sonuçları paylaşılmıştır.

Çok aşamalı uygulamanın ilk aşamasında, ilgili kısıtlar göz önünde bulundurularak mahallelerin araçlara optimum dağıtımı ve rotalanması yapılmıştır. Bu aşamada problem, araç rotalama problem çeşidi olan Zaman Pencereci Araç Rotalama Problemi olup, Genetik Algoritma (GA) ile çözüm elde edilmiştir. İkinci aşama ise ilk aşamada yapılan rotalama ile artık birer bölge olarak düşünülen alanlarda ilaçlama yapacak araçların caddeleri/sokakları minimum mesafe ile gezinmesi amaçlanmaktadır. Bu aşamada ise ayrıt rotalama problemlerinden olan Çinli Postacı Problemi (ÇPP) olup, çözümü için Yapay Arı Kolonisi Algoritması (YAKA) kullanılmıştır. ÇPP'nin en önemli adımı tek dereceli düğümlerin aralarında minimum eşleşmesidir. Bu çalışmada minimum eşleşmenin sağlanması için YAKA'nın parametre değişimi kısmına modifiye işlemi yapılmıştır. Böylece çok büyük boyutlu graflarda bile optimum eşleşme başarılı şekilde sağlanmış olup, YAKA'ya yapılan modifiye işlemi literatüre kazandırılmıştır. Optimum eşleşme ile bölgeler için elde edilen Eulerian graflar üzerinde, Euler tur oluşturacak rotalar elde edilmiştir.

Çok aşamalı uygulamanın verileri olan mahalleler arası mesafe matrisi ve cadde/sokak mesafe matrisleri, coğrafi bilgi sistemi (CBS) yazılımlarından QGIS yazılımı ile bizzat bu çalışma için oluşturulmuş gerçek ve orjinal verilerdir. Bu çok aşamalı uygulama, bu mesafe matrisleri üzerinde uygulanmış ve optimum sonuçlar elde edilmiştir.

ANAHTAR KELİMELELER: Araç Rotalama, Ayrıt Rotalama, Çinli Postacı Problemi, Eulerian Graf, Euler Tur, Metasezgisel Yöntemler, Genetik Algoritma, Yapay Arı Kolonisi Algoritması, Optimizasyon, Coğrafi Bilgi Sistemleri, QGIS Yazılımı.

ABSTRACT

DETERMINATION OF MALATYA METROPOLITAN MUNICIPALITY PESTICIDE VEHICLES WITH METAHEURISTIC ALGORITHMS

In this thesis, a multistage application has been developed to optimize the routes of the pesticide vehicles of Malatya Metropolitan Municipality with metaheuristic methods and the results have been shared.

In the first stage of multistage application, the optimum distribution and routing of the neighborhoods to the vehicles has been made considering the related constraints. At this stage, the problem is the Vehicle Routing Problem with Time Window which is the type of vehicle routing problem and it has been solved using Genetic Algorithm (GA). In the second stage, the areas which were obtained as a result of the routing in the first stage are considered as the region and it is aimed that the spraying vehicles will travel the streets / roads in these regions with minimum distance. At this second stage, the problem is the Chinese Postman Problem (CPP) which is one of the problems of arc routing and it has been solved using Artificial Bee Colony (ABC) Algorithm. The most important step of the CPP is the minimum matching of odd degree nodes. In this study, it is modified to the parameter change part of ABC algorithm to provide minimum matching. Thus, optimum matching is achieved successfully even in very large graphs and the modification made to ABC algorithm has been introduced into the literature. Later, on Eulerian graphs obtained for regions with the optimum matching, Euler tour routes have been obtained.

The neighborhoods distance matrix and the street / road distance matrices which is the data of the multistage application, were constituted for this study with the QGIS software from the geographic information system (GIS) software and are the real, original data. This multistage application has been applied on these distance matrices and optimum results have been obtained.

KEYWORDS: Vehicle Routing, Arc Routing, Chinese Postman Problem, Eulerian Graph, Euler Tour, Metaheuristic Methods, Genetic Algorithm, Artificial Bee Colony Algorithm, Optimization, Geographic Information Systems, QGIS Software.

İÇİNDEKİLER

ONAY SAYFASI.....	iii
ONUR SÖZÜ	iv
TEŞEKKÜR	v
ÖZET	vi
ABSTRACT.....	vii
KISALTMALAR.....	xii
TABLolar	xiii
ŞEKİL LİSTESİ	xv
GİRİŞ.....	1
BÖLÜM 1	4
1. DÜĞÜM ve AYRIT ROTALAMA PROBLEMLERİ.....	4
1.1. Gezin Satıcı Problemi	4
1.1.1. Kombinatoryel Optimizasyon Problemleri.....	4
1.1.2. Gezin Satıcı Probleminin Tanımı	4
1.1.3. Gezin Satıcı Probleminin Uygulamaları	6
1.1.4. GSP için Çözüm Yöntemleri	7
1.1.4.1. Dal - Sınır Yöntemi	7
1.1.4.2. Düzlem - Kesme Yöntemi	9
1.1.4.2.1. Dal - Kesme Algoritması	12
1.1.4.3. Dinamik Programlama	12
1.1.4.4. Sezgisellik (Heuristic).....	12
1.1.4.5. Metasezgisel (Metaheuristics);.....	13
1.2. Araç Rotalama Problemi.....	22
1.2.1. Araç Rotalama Probleminin Tanımı.....	22
1.2.2. Araç Rotalama Problemlerinin Uygulama Alanları	25
1.2.3. Araç Rotalama Probleminin Çeşitleri.....	26
1.2.3.1. Kapasite Kısıtlı Araç Rotalama Problemi (KKARP)	27
1.2.3.2. Mesafe Kısıtlı Araç Rotalama Problemi (MKARP).....	28

1.2.3.3. Zaman Pencere li Araç Rotalama Problemi (ZPARP)	28
1.2.3.4. Dağıt Topla Araç Rotalama Problemi (DTARP)	28
1.2.3.5. Eş Zamanlı ve Karışık Topla-Dağıt Araç Rotalama Problemi (EZKTDARP).....	29
1.2.3.6. Önce Dağıt Sonra Topla Araç Rotalama Problemi (ÖDSTARP)29	
1.2.3.7. Çok Depolu Araç Rotalama Problemi (ÇDARP).....	29
1.2.3.8. Periyodik Araç Rotalama Problemi (PARP)	30
1.2.3.9. Bölünmüş Talebe Sahip Araç Rotalama Problemi (BTARP).....	30
1.2.3.10. Açık Uçlu Araç Rotalama Problemi (AUARP)	30
1.2.4. Araç Rotalama Problemi Çözüm Yöntemleri.....	30
1.2.4.1. Kesin Çözüm Yöntemleri	31
1.2.4.2. Klasik Sezgisel Yöntemler.....	32
1.2.4.2.1. Tur Kurucu Sezgiseller	32
1.2.4.2.2. İki Aşamalı Metotlar	32
1.2.4.3. Tur Geliştirici Sezgiseller.....	33
1.2.4.4. Metasezgisel Yöntemler	33
1.3. Çinli Postacı Problemi	35
1.3.1. Çinli Postacı Probleminin Tanımı.....	35
1.3.2. Euler Tur ve Turun Bulunması.....	38
1.3.3. Çinli Postacı Probleminin Versiyonları	42
1.3.3.1. Yönlü ÇPP	43
1.3.3.2. Yönsüz ÇPP	44
1.3.3.3. Karma ÇPP	45
1.3.3.4. k – ÇPP	47
1.3.3.5. Min - Maks k -ÇPP.....	48
1.3.3.6. Hızlı ÇPP	49
1.3.3.7. Hiyerarşik ÇPP	50
1.3.3.8. Kırsal PP.....	51
1.3.3.9. Kapasite Kısıtlı AR	53
1.3.4. ÇPP, KPP ve KKR ile İlgili Literatürde Yapılmış Çalışmalar	56
1.3.5. Çinli Postacı Probleminin Uygulama Alanları	60
BÖLÜM 2	61

2. UYGULAMADA KULLANILAN METASEZGİSEL ALGORİTMALAR	61
2.1. Genetik Algoritma (GA)	61
2.1.1. Genetik Algoritma ile İlgili Temel Kavramlar	64
2.1.1.1. Gen	64
2.1.1.2. Kromozom	64
2.1.1.3. Popülasyon (Yığın)	64
2.1.2. Yeniden Üretim İşlemi	64
2.1.3. Başlangıç Popülasyonunun Oluşturulması	65
2.1.4. Uygunluk Değeri	65
2.1.5. Kodlama (Dizi Gösterimi)	65
2.1.6. Genetik Operatörler	67
2.1.6.1. Seçim (Çoğalma) Mekanizmaları	67
2.1.6.2. Çaprazlama	69
2.1.6.3. Mutasyon (Değişim)	71
2.1.6.4. Tamir	72
2.1.6.5. Elitizm (En İyinin Saklanması)	72
2.1.7. Durdurma Kriteri	72
2.1.8. Genetik Algoritmanın Uygulama Alanları	73
2.2. Yapay Arı Kolonisi Algoritması (YAKA)	74
2.2.1. Doğadaki Gerçek Arılar ve Görevleri	74
2.2.2. Yapay Arı Koloni Algoritması Tanımı	77
2.2.2.1. Başlangıç Besin Kaynağı Bölgelerinin Bulunması	77
2.2.2.2. Görevli Arıların Besin Kaynağı Bölgelerine Yönlendirilmesi	78
2.2.2.3. Gözcü Arıların Besin Kaynağı Seçiminde Kullanacakları Olasılık Değerlerinin Hesaplanması (Dans Benzetimi)	79
2.2.2.4. Gözcü Arıların Besin Kaynağı Bölgesini Seçmesi	79
2.2.2.5. Biten Besin Kaynağını Bırakma ve Kâşif Arının Meydana Gelmesi	79
2.2.3. Yapay Arı Kolonisi Algoritmasının Özellikleri ve Adımları	80
2.2.4. Çinli Postacı Probleminin Eşleşme Adımı için YAKA' da Yapılan Modifiye İşlemi	82
BÖLÜM 3	85

3. UYGULAMA	85
3.1. Uygulamanın Tanımlanması	85
3.1.1. Malatya Atık Yönetim ve İlaçlama Şube Müdürlüğü	86
3.1.2. İlaçlama Araçlarının Yapısı ve İşleyişi	86
3.2. Uygulamanın Önemi	92
3.3. Coğrafi Bilgi Sistemleri (CBS) ve QGIS Programı	93
3.3.1. Coğrafi Bilgi Sistemleri (CBS)	93
3.3.2. QGIS Programı	94
3.4. Verilerin Toplanması ve Düzenlenmesi	94
3.4.1. Mahalleler Arası Uzaklık Matrisinin Oluşturulması	95
3.4.2. Cadde/Sokak Uzaklık Matrislerinin Oluşturulması	95
3.5. Uygulamanın Matematiksel Modelleri	103
3.5.1. Zaman Pencereci Araç Rotalama Probleminin Matematiksel Modeli	103
3.5.2. Yönsüz Çinli Postacı Probleminin Matematiksel Modeli	105
3.6. Modellerin Uygulanması	107
3.6.1. Zaman Pencereci Araç Rotalama Modelinin Uygulanması	107
3.6.2. Yönsüz Çinli Postacı Problemi Modelinin Uygulanması	122
BÖLÜM 4	161
4. SONUÇ	161
KAYNAKÇA	165
EKLER	183
EK – 1: “KERNEK” Mahallesi Cadde/Sokak Düğümler Arası Uzaklıkları	183
EK – 2: Bölge 12 ve Bölge 34’ ün Rotaları	184
EK – 3: Çipli İlaçlama Araçlarının ARVENTO Programından Alınan 14 – 19 Ağustos 2017 tarihli Mevcut Gezinme Raporları	191

KISALTMALAR

AR	Ayrıt Rotalama
ARP	Araç Rotalama Problemi
AUARP	Açık Uçlu Araç Rotalama Problemi
BTARP	Bölünmüş Talebe Sahip Araç Rotalama Problemi
CBS	Coğrafi Bilgi Sistemleri
ÇDARP	Çok Depolu Araç Rotalama Problemi
ÇPP	Çinli Postacı Problemi
DTARP	Dağıt - Topla Araç Rotalama Problemi
EZKTDARP	Eş Zamanlı ve Karışık Topla-Dağıt Araç Rotalama Problemi
GA	Genetik Algoritma
GIS	Geographic Information System
GSP	Gezgin Satıcı Problemi
KKAR	Kapasite Kısıtlı Ayrıt Rotalama
KKARP	Kapasite Kısıtlı Araç Rotalama Problemi
KKO	Karınca Koloni Algoritması
KPP	Kırsal Postacı Problemi
MKARP	Mesafe Kısıtlı Araç Rotalama Problemi
ÖDSTARP	Önce Dağıt Sonra Topla Araç Rotalama Problemi
PARP	Periyodik Araç Rotalama Problemi
PP	Postacı Problemi
PSO	Parçacık Sürü Optimizasyonu
QGIS	Quantum Geographic Information System
TA	Tabu Arama
TB	Tavlama Benzetimi
YAKA	Yapay Arı Kolonisi Algoritması
ZPARP	Zaman Pencereli Araç Rotalama Problemi

TABLolar

Tablo 1.1. GSP Parametrelerinin Tavlama Benzetimindeki Karşılıkları	17
Tablo 1.2. Çinli Postacı, Kapasite Kısıtlı Ayrıt Rotalama ve Kırsal Postacı Problemlerinin Zaman Karmaşıklığı (Kaynak: Corberán ve Laporte,2013:24; Assad ve Golden,1995:377; Longo ve vd.,2004:5; Korteweg ve Volgenant,2006:47; Eiselt ve vd.,1995a:240)	55
Tablo 1.3. ÇPP, KPP ve KKR ile İgili Yapılan Çalışmalar	56
Tablo 3.1. Malatya Atık Yönetim ve İlaçlama Birimi'nin 2017 yılı için Planladığı İlaçlama Gün ve Ekip Çizelgesi (Kaynak: Malatya Büyükşehir Belediyesi Çevre Koruma ve Kontrol Dairesi Başkanlığı Atık Yönetim ve İlaçlama Birimi).....	89
Tablo 3.2. Önerilen Modellerin Çözüm Aşamaları.....	107
Tablo 3.3. Yeşilyurt İlçesine ait Mahalleler ile Bölünmüş veya Cadde/Sokak Uzunlukları Belli Olmayan Bazı Mahallelerinin Toplam Cadde/Sokak Uzunlukları (Kaynak: QGIS programı ile elle ölçülmüştür)	114
Tablo 3.4. Battalgazi İlçesine ait Mahalleler ile Bölünmüş veya Cadde/Sokak Uzunlukları Belli Olmayan Bazı Mahallelerinin Toplam Cadde/Sokak Uzunlukları (Kaynak: QGIS programı ile elle ölçülmüştür)	115
Tablo 3.5. Yeşilyurt İlçesi 1. Araç İçin Çıkan Rotalama Sonuçları.....	117
Tablo 3.6. Yeşilyurt İlçesi 2. Araç İçin Çıkan Rotalama Sonuçları.....	118
Tablo 3.7. Yeşilyurt İlçesi 3. Araç İçin Çıkan Rotalama Sonuçları.....	118
Tablo 3.8. Yeşilyurt İlçesi 4. Araç İçin Çıkan Rotalama Sonuçları.....	118
Tablo 3.9. Yeşilyurt İlçesi 5. Araç İçin Çıkan Rotalama Sonuçları.....	119
Tablo 3.10. Yeşilyurt İlçesi 6. Araç İçin Çıkan Rotalama Sonuçları.....	119
Tablo 3.11. Yeşilyurt İlçesi 7. Araç İçin Çıkan Rotalama Sonuçları.....	119
Tablo 3.12. Battalgazi İlçesi 1. Araç İçin Çıkan Rotalama Sonuçları	120
Tablo 3.13. Battalgazi İlçesi 2. Araç İçin Çıkan Rotalama Sonuçları	120
Tablo 3.14. Battalgazi İlçesi 3. Araç İçin Çıkan Rotalama Sonuçları	121
Tablo 3.15. Battalgazi İlçesi 4. Araç İçin Çıkan Rotalama Sonuçları	121
Tablo 3.16. Battalgazi İlçesi 5. Araç İçin Çıkan Rotalama Sonuçları	122
Tablo 3.17. Yeşilyurt İlçesi için Mahalle/Mahalle Gruplarına Verilen Bölge Numaraları	123

Tablo 3.18. Battalgazi İlçesi için Mahalle/Mahalle Gruplarına Verilen Bölge Numaraları	124
Tablo 3.19. Bölge 12 için Çinli Postacı Algoritması Çözümünün Sonucu	128
Tablo 3.20. Bölge 34 için Çinli Postacı Algoritması Çözümünün Sonucu	135
Tablo 3.21. Yeşilyurt İlçesi 1. Araç İçin Çıkan Yönsüz ÇPP Sonuçları	145
Tablo 3.22. Yeşilyurt İlçesi 2. Araç İçin Çıkan Yönsüz ÇPP Sonuçları	146
Tablo 3.23. Yeşilyurt İlçesi 3. Araç İçin Çıkan Yönsüz ÇPP Sonuçları	147
Tablo 3.24. Yeşilyurt İlçesi 4. Araç İçin Çıkan Yönsüz ÇPP Sonuçları	148
Tablo 3.25. Yeşilyurt İlçesi 5. Araç İçin Çıkan Yönsüz ÇPP Sonuçları	149
Tablo 3.26. Yeşilyurt İlçesi 6. Araç İçin Çıkan Yönsüz ÇPP Sonuçları	150
Tablo 3.27. Yeşilyurt İlçesi 7. Araç İçin Çıkan Yönsüz ÇPP Sonuçları	151
Tablo 3.28. Battalgazi İlçesi 1. Araç İçin Çıkan Yönsüz ÇPP Sonuçları	152
Tablo 3.29. Battalgazi İlçesi 2. Araç İçin Çıkan Yönsüz ÇPP Sonuçları	153
Tablo 3.30. Battalgazi İlçesi 3. Araç İçin Çıkan Yönsüz ÇPP Sonuçları	154
Tablo 3.31. Battalgazi İlçesi 4. Araç İçin Çıkan Yönsüz ÇPP Sonuçları	155
Tablo 3.32. Battalgazi İlçesi 5. Araç İçin Çıkan Yönsüz ÇPP Sonuçları	156
Tablo 3.33. Bu Çalışmadaki Uygulamanın Sonuçları ile Mevcut Araçların İşleyiş Sonuçlarının Karşılaştırılması (Bir Haftalık Sonuçlar)	158

ŞEKİL LİSTESİ

Şekil 1.1. 2 - Opt Algoritması (Kaynak: Davendra, 2010:15).....	15
Şekil 1.2. 3 - Opt Algoritması (Kaynak: Davendra, 2010:15).....	15
Şekil 1.3. Çift Köprü Hareketi (Kaynak: Davendra, 2010:15).....	16
Şekil 1.4. Araç Rotalama Problemi (ARP) (Kaynak: Machado vd., 2002:2).....	23
Şekil 1.5. Araç Rotalama Problemi Çeşitleri ve Birbirleri İle İlişkileri (Kaynak: Toth ve Vigo 2002a:6)	27
Şekil 1.6. Königsberg ‘in Yedi Köprüsü (Kaynak: Eroğlu, 2015:1).....	35
Şekil 1.7. Euler’in Königsberg Çizimi, 1736 (Kaynak: Grötschel ve Yuan, 2012:44)...	39
Şekil 1.8. Pregel Nehrindeki Yedi Köprü Çizgesi (Kaynak: Limon, 2015:2)	39
Şekil 1.9. Ayrıt Rotalama Problemleri Arasındaki İlişki (Kaynak: Padungwech, 2018:15).....	42
Şekil 1.10. (a) KPP’ de girdi grafi (b) KPP’ de çıktı grafi (Kaynak: Corberán ve Laporte, 2013:36).....	51
Şekil 2.1. Genetik Algoritma Akış Diyagramı (Kaynak: Elmas, 2016:425).....	62
Şekil 2.2. İkili Kodlanmış Kromozom Örneği.....	65
Şekil 2.3. Permütasyon (sıralı) Kodlanmış Kromozom Örneği.....	66
Şekil 2.4. Değer Kodlanmış Kromozom Örneği (Kaynak: Nabyev, 2010:590).....	66
Şekil 2.5. Ağaç Kodlanmış Kromozom Örneği (Kaynak: Taşkın ve Emel, 2009:36). ...	67
Şekil 2.6. Tek Noktalı Çaprazlama	69
Şekil 2.7. İki Noktalı Çaprazlama	69
Şekil 2.8. PMX Çaprazlama	70
Şekil 2.9. Mutasyon Operasyon Örneği	72
Şekil 2.10. Arıların Görev Dağılım Şeması (Kaynak: Akay, 2009:32)	75
Şekil 2.11. Arılarda Dans (Karaboğa, 2014:204).....	77
Şekil 2.12. YAKA Algoritmasının Akış Şeması (Kaynak: Akay, 2009:63)	81
Şekil 2.13. (a)Tek Dereceli Dügümlerden Oluşmuş Bir İşçi Arı.....	84
Şekil 3.1. İl – Star 18 ULV İlaçlama Makinası (Kaynak: https://images.google.com/ , (09.12.2017)).....	87
Şekil 3.2. Malatya Büyükşehir Belediyesi İlaçlama Aracı (Kaynak: https://images.google.com/ (09.12.2017)).....	88
Şekil 3.3. QGIS Programına Malatya İl Haritasının Yüklenmiş Hali.....	95

Şekil 3.4. QGIS’de “KERNEK” Mahallesi İçin Ekran Görüntüsü.....	96
Şekil 3.5. Her bir Mahalleye Verilen Dğümler Katmanının Gösterimi	97
Şekil 3.6. Her bir Mahalle İçerisinde ki Cadde/Sokak Başı ve Sonuna Dğümler Verilen Katmanın Gösterimi	97
Şekil 3.7. Her bir Mahalle İçinde ki Cadde/Sokak Arasında ki Ayrıtların Belirlendiği Katmanın Gösterimi	98
Şekil 3.8. Öznitelik Tablosundan Mahalle Seçimi.....	98
Şekil 3.9. Örnek Mahallemizin İsmine Yazılmasını Sağlayan Kısım.....	99
Şekil 3.10. Örnek Mahallemiz Olan Sadece “KERNEK” Mahallesi İçin Çağırılması	99
Şekil 3.11. QGIS Ekranında “KERNEK” Mahallesi İçin Cadde/Sokak Dğümlerinin Seçilmiş Hali.....	100
Şekil 3.12. QGIS Ekranında Cadde /Sokakların Baş ve Sonuna ID Verilmiş Hali	100
Şekil 3.13. “KERNEK” Mahallesi Sınırı Dışında Kalan Dğümlerinin Seçimi	101
Şekil 3.14. Seçilen Objelerin İlave Edilmesi	101
Şekil 3.15. İlave Edilen Dğümler İle Gösterim.....	102
Şekil 3.16. “KERNEK” Mahallesi İçin Elde Edilen Cadde/Sokak Mesafe Matrisi	102
Şekil 3.17. İlaçlama Aracının İlyas Mahallesi İçin Gezinmesi	108
Şekil 3.18. (a)Bulgurlu Mahallesi İçin Bölünmemiş Hali (b) Bulgurlu Mahallesi İçin İkiye Bölünmüş ve QGIS’ den Orta Noktalar Atanmış Hali (c) Dilek Mahallesi İçin Bölünmemiş Hali (d) Dilek Mahallesi İçin Üç Bölünmüş ve QGIS’ den Orta Noktalar Atanmış Hali	110
Şekil 3.19. Depodan Çıkan Aracın Bir Adet Kümelenmiş Mahalle Grubunun Olması Durumunda İzleyeceği Yol.....	112
Şekil 3.20. Depodan Çıkan Aracın Birden Çok Kümelenmiş Mahalle Grubunun Olması Durumunda İzleyeceği Yol.....	113
Şekil 3.21. (a) Bentbaşı –Kiltepe-Yeşilkaynak Mahalle Grubu (b) Bentbaşı –Kiltepe-Yeşilkaynak Mahalle Grubunun Bölge Hali.....	126
Şekil 3.22. Bölge 12 ile Deponun Haritada Gösterimi.....	126
Şekil 3.23. Bölge 12’nin Depoya En Yakın Dğümünün Gösterimi	127
Şekil 3.24. Bölge 12 İçin Minimum Maliyet Grafiği	129
Şekil 3.25. Bölge 12 İçin İlk Graf Hali	130

Şekil 3.26. Bölge 12 için Tek Dereceli Döğümler Arası Minimum Eşleşme İle Çıkan Eulerian Graf.....	131
Şekil 3.27. Bölge 12 için Tek Dereceli Döğümler Arası Minimum Eşleşme İle Çıkan Eulerian Graf'ın Rotası.....	132
Şekil 3.28. Bölge 34'ün Haritada Gösterimi ve Aracın Gezinme Sırası	133
Şekil 3.29. Şeyhbayram – Abdulgaffar Alanının Depoya En Yakın Döğümünün Gösterimi	134
Şekil 3.30. Şeyhbayram – Abdulgaffar Alanından Turgut Özal' a En Yakın Geçiş Döğümlerinin Gösterimi.....	135
Şekil 3.31. Bölge 34 Şeyhbayram - Abdulgaffar İçin Minimum Maliyet.....	137
Şekil 3.32. Bölge 34 Şeyhbayram - Abdulgaffar İçin İlk Graf.....	138
Şekil 3.33. Bölge 34 Şeyhbayram - Abdulgaffar İçin Tek Dereceli Döğümler Arası Minimum Eşleşme İle Çıkan Eulerian Graf.....	139
Şekil 3.34. Bölge 34 Şeyhbayram - Abdulgaffar için Tek Dereceli Döğümler Arası Minimum Eşleşme İle Çıkan Eulerian Graf'ın Rotası	140
Şekil 3.35. Bölge 34 Turgut Özal İçin Minimum Maliyet.....	141
Şekil 3.36. Bölge 34 Turgut Özal İçin İlk Graf	142
Şekil 3.37. Bölge 34 Turgut Özal İçin Tek Dereceli Döğümler Arası Minimum Eşleşme İle Çıkan Eulerian Graf	143
Şekil 3.38. Bölge 34 Turgut Özal için Tek Dereceli Döğümler Arası Minimum Eşleşme İle Çıkan Eulerian Graf'ın Rotası	144

GİRİŞ

Dağıtım yönetiminde günümüzün en önemli kararlarından biri, dağıtım noktalarına hizmet verecek araçların minimum maliyet, minimum mesafe ve minimum zaman ile rotalanmasıdır. İlk olarak Dantzig ve Ramser tarafından incelenen Araç Rotalama Problemi, bir veya birden çok depodan çıkan araçların her müşteriye yalnızca bir defa uğrayarak hizmet vermesi ve daha sonra tekrar çıktığı depoya optimum mesafe ve maliyetle tur oluşturacak şekilde dönmesidir.

Bir postacının, elindeki postaları sahiplerine ulaştırma işini en az yol kat ederek gerçekleştirme isteği üzerine düşünen Guan ise belki de önerdiği Çinli Postacı Probleminin günümüzde dahi birçok alanda kullanılabileceğini tahmin etmemiştir. Son yıllarda Çinli Postacı Problemine benzer dağıtım sistemleri giderek daha da karmaşık hale gelmiştir. Zamandan, kullanılan kaynaklardan ve iş gücünden tasarruf etme isteği, dağıtım firmalarının rekabeti ile birlikte oldukça önemli bir noktaya yerleşmiştir. Ekonomideki öneminden dolayı bu tür dağıtım problemleri literatürde Rotalama Problemleri olarak genişçe irdelenmiş ve bu problemler için önerilen çözümler birçok şirket ve kamusal oluşumlar tarafından kullanılmıştır.

Rotalama problemlerini genel olarak ikiye ayırmak mümkündür. Bunlardan ilki, verilecek hizmetin belli hedeflere uğranmasını gerektiren, ancak bu hedeflerin bulunduğu lokasyonun tamamen gezilmesini gerektirmeyen düğüm rotalama problemidir. Yani düğüm rotalama, bir dizi müşteri düğümüne hizmet vermek için rotaları optimize etmeyi amaçlar ve ayrıtlar sadece düğümleri bağlayan yolların öğeleri ile ilgilenir. Bir diğer rotalama problemi de, hizmetin lokasyona genel olarak uygulandığı, lokasyondaki caddelerin ve sokakların gezilmesinin gerektiği ayrıt rotalama problemidir. Yani ayrıt rotalama servis etkinliğinde bir ulaşım ağının (ayrıtların/kenarların) ziyaret edilmesi üzerine odaklanan, bir dizi ayrıtı sunmak için rotaları minimum maliyetle veya sürede bulmayı amaçlayan rotalama problemidir. Ayrıt rotalama probleminde eğer hizmetin verileceği lokasyonda sadece belli ana arterlerin (kenarlar/ayrıtlar) gezilmesi bekleniyorsa bu problemler Kırsal Postacı Problemi (KPP), tüm arterlerin (kenarlar/ayrıtlar) gezilmesi isteniyorsa da Çinli Postacı Problemi (ÇPP) olarak adlandırılmaktadır. Düğüm rotalamaya örnek olarak okullara öğrenci taşıyan servislerin, öğrencilerin evlerini gezerek okula ulaştırma rotasının hesaplanması

gösterilebilirken, ayrıt rotalamaya ise mahalledeki tüm sokaklardaki çöp konteynırlarını gezerek çöp toplayan belediye araçlarının rotalamaları örnek gösterilebilir.

Büyük kentlerimizde, belediyelerin hizmetlerini verebilmesi için gerekli iletişim, ulaşım ve altyapı gibi çalışmalarda rota planlamanın en iyilenmesi, belediyenin vermesi gereken hizmeti gereksiz masraflardan kaçınarak gerçekleştirmesini sağlamaktadır. Rotalamanın en uygun hale getirilmesi, hem hizmetin daha hızlı ulaştırılmasını hem de belediyelerimizin bütçelerinden tasarruf etmelerini sağlayacak büyük bir kaynak istifadesidir.

Bu tez çalışmasında, tüm ayrıtların gezinmesini gerektiren çinli postacı problemine benzer, Malatya Büyükşehir Belediyesi'nin ilaçlama araçlarının öncelikle mahallelere optimum bir şekilde dağıtımını sağlanmaya çalışılmış, ardından bu araçların, araçlara görev olarak verilen mahalleleri en düşük maliyet ile gezinmelerini sağlayan rotalar oluşturulmaya çalışılmıştır.

Rotalama işlemlerini optimizasyonları kesin metotlar ile çözülemeyecek kadar karmaşık olduğundan, bu probleme metasezgisel yöntemler ile yaklaşmıştır. Metasezgisel yöntemler, kesin metotlar ile çözülemeyen problemlerin sezgisel yöntemler ile optimum sonucunun tahminine dayalı yöntemlerdir ve başarıları günümüze kadar bir çok çalışmada gösterilmiştir.

Bu tez çalışmasında, ilaçlama araçlarının görev paylaşımı, Araç Rotalama probleminin bir çeşidi olan Zaman Pencere Araç Rotalama probleminin metasezgisel yöntemlerden olan Genetik Algoritma (GA) ile çözümlenmesiyle yapılmış, araçların mahalleler içerisinde optimum şekilde gezinme işleminde kullanılan Çinli Postacı Probleminin eşleşme adımına bir diğer metasezgisel yöntem olan Yapay Arı Kolonisi Algoritması (YAKA) uygulanarak optimum eşleşme ile elde edilen Eulerian grafa daha sonra Euler tur oluşturulması ile optimum rota elde edilmiştir.

Literatür taramasında ayrıca bu tez çalışması kadar kapsamlı çok fazla çalışma olmadığı gözlemlenmiştir. Çalışmanın boyutunun çok büyük olması nedeni ile çok aşamalı bir uygulama yapılmıştır. Bu çalışmada kullanılan bütün veriler (mahalleler arası mesafe matrisi ve cadde/sokak mesafe matrisleri) QGIS programı ile tarafınca

oluřturulmuř gerek ve orjinal verilerdir. PP'nin tek dereceli dğmler arası minumum eřleřme adımı iin YAKA'nın parametre deęiřimi kısmına yapılan modifiye iřlemi sonucu ok byk boyutlu graflarda bile optimum eřleřme saęlanmıřtır. YAKA'ya yapılan bu modifiye iřlemi literatre kazandırılmıřtır.

ok byk boyutlu bu alıřmanın hem verilerinin gerek ve orijinal olması, hem ilk kez Ynsz PP'ye YAKA ile zm aranması, hem YAKA'ya uygulanan modifiye iřlemi, hem nerilen ok ařamalı zm yntemi ve hem de alıřmanın sonucunun Malatya Bykřehir Belediyesi ilalama araları tarafından kullanılacak olması nedeniyle alıřma tamamen zgn bir hal almıřtır.



BÖLÜM 1

1. DÜĞÜM ve AYRIT ROTALAMA PROBLEMLERİ

1.1. Gezgin Satıcı Problemi

1.1.1. Kombinatoryel Optimizasyon Problemleri

Kombinatoryel Optimizasyon; nesnelere için belli bir sayıda bir sıralamanın mevcut olup olmadığından ziyade en iyi sıralamanın belirlenmesi ile ilgilidir. Yöneylem araştırmasında kombinatoryel eniyileme: “kesikli çözüm uzayına sahip problemler için en iyi çözümü arayan yöntemlerdir”. Kısacası dikkate aldığımız amaç fonksiyonunu en iyileyen kesikli karar değişkenlerinin bulunmasıdır.

Kombinatoryel Optimizasyon Problemleri iki başlık altında toplanmaktadır;

- P (tractable; erişilebilir); yani en iyi çözüm elde edilebilir.
- NP -zor (intractable; erişilemez) ; yani en iyi çözüm uygun zamanlarda elde edilemez.

P sınıfı problemler; çözüm zamanı problem genişliğinin polinom fonksiyonu olarak artan bir algoritma ile çözülebilir. Çözüm zamanı $O(n^2)$ 'dir. Örneğin; Atama Problemleri, Minimum Yayılan Ağaç, Şebeke Akış Problemleri v.s.

NP -zor gurubundaki problemlerin çözümü için polinom zamanlı algoritma yoktur. Çözüm zamanı $O(2^n)$ 'dir. Gezgin satıcı Problemi, Karesel Atama Problemi, Araç Rotalama Problemi, Çizelgeleme Problemi, Yer Seçimi Problemi v.s.

1.1.2. Gezgin Satıcı Probleminin Tanımı

Gezgin Satıcı Problemi (GSP) Hamilton çevrimini gerektirir. Düğümler, Gezgin Satıcı Probleminde bir gezgin satıcının uğramak zorunda olduğu şehirleri ifade eder. GSP' de satıcı her bir şehri ziyaret etmeli, her şehre sadece bir defa uğramalı ve her şehirden yalnızca bir kez çıkmalıdır. Problemden amaç; bütün düğümleri kapsayan ve başlangıç düğümünde son bulan minimum rotanın oluşturulmasıdır.

Maliyetler simetrik kabul edilmiş olup (1.2) deki amaç fonksiyonumuzda c_{ij} , i şehirden j şehrine yolculuk yapmanın maliyeti/mesafesi olarak alınmıştır.

Gezgin satıcı problemi tamsayı bir model olarak aşağıdaki şekildedir (Taşkın ve Emel, 2009:108-110);

Karar Değişkeni

$$x_{i,j} = \begin{cases} i \text{ şehrinden } j \text{ şehrine geçiş olursa,} & 1 \\ \text{aksi taktirde,} & 0 \end{cases} \quad (1.1)$$

Amaç Fonksiyonu

$$\min = \sum_{i \neq j} c_{ij} x_{ij} \quad (1.2)$$

Kısıtlar

$$\sum_j x_{ij} = 1 \text{ tüm } i\text{'ler için } i \neq j \quad (1.3)$$

$$\sum_i x_{ij} = 1 \text{ tüm } j\text{'ler için } i \neq j \quad (1.4)$$

(1.1)' de x_{ij} karar değişkenimiz olup i şehrinden j şehrine geçiş olursa 1, olmazsa 0'a eşittir. Kısıt (1.3) her şehirden bir çıkış olması gerektiğini, (1.4) ise her şehre bir giriş olması gerektiğini belirtmektedir. Bu formülasyon klasik atama yöntemine benzer. Tam bir GSP formülasyonu için alt tur engelleme kısıtları da modele katılmalıdır. Alt turlar iki, üç, dört ve daha fazla şehirden oluşabilir. Modele alt tur engel kısıtları eklenir ve problem tamsayı programlama ile modellenmiş olur.

$$x_{ij} + x_{ji} \leq 1 \quad i \neq j \quad \text{tüm } i \text{ ve } j \text{ için} \quad (1.5)$$

$$x_{ij} + x_{jk} + x_{ki} \leq 2 \quad i \neq j \neq k \quad \text{tüm } i, j \text{ ve } k \text{ için} \quad (1.6)$$

$$x_{ij} + x_{jk} + x_{kl} + x_{li} \leq 3 \quad i \neq j \neq k \neq l \quad \text{tüm } i, j, k \text{ ve } l \text{ için} \quad (1.7)$$

Kısıt (1.5), iki şehirden oluşabilecek alt turları, kısıt (1.6) üç şehirden oluşabilecek alt turları ve kısıt (1.7) dört şehirden oluşabilecek alt turları engelleme kısıtlarıdır. Böylece GSP probleminin modeli oluşmaktadır. Fakat problemdeki şehir sayısı arttıkça alt tur engelleme sayısı arttığı için çözüme ulaşma zamanı da artacaktır.

1.1.3. Gezgin Satıcı Probleminin Uygulamaları

GSP, daha çok yol planlaması ile rotalama üzerine yapılan çalışmalarda literatürde karşımıza çıkmaktadır. Bu nedenle günlük hayattaki problemler üzerinde uygulanabilen GSP, optimizasyon çalışmalarında birçok akademisyenin de ilgi alanına girmiştir. GSP' nin kullanım alanları şu şekilde sıralanabilir (Kuzu ve vd., 2014:3).

- GSM operatörlerinin baz istasyonlarının yerleşim yerlerinin belirlenmesi,
- Ulaşım ve lojistik uygulamaları,
- Malzeme akış sistem tasarımı,
- Araç rotalama problemleri,
- Depolardaki vinç güzergâhlarının programlaması,
- Stok alanındaki malzeme toplama problemleri,
- Uçaklar için havaalanı rotalaması,
- Elektronik devre tasarımı
- Röportaj zamanlama,
- Matbaa zamanlama,
- Bilgisayar kablolama,
- Ekip planlama,
- Misyona planlama,
- Navigasyon uydu sisteminin ölçme ağlarının tasarımı,
- Sipariş toplama.

GSP' ye göre formüle edilip en çok kullanılan uygulama alanları şunlardır;

- Bilgisayar Bağlantısı
- Devre Levhaların Delinmesi
- İşlerin Sıralanması
- Duvar Kağıdı Kesimi
- Kristalografi
- Gaz Türbünü Motorlarının Bakımı

1.1.4. GSP için Çözüm Yöntemleri

GSP için çözüm yöntemleri kesin ve yaklaşık (sezgisel) olmak üzere iki gruba ayrılmaktadır. Kesin çözüm sağlayan algoritmaların, büyük veri kümeleri için hesaplama karmaşıklığından ileri gelen yüksek çalışma sürelerine sahip olmaları sebebiyle literatürde GSP çalışmalarında sezgisel algoritmaların önemi artmıştır. Sezgisel algoritmalar, kesin çözümü garantilemezler ancak, yakın bir çözümü garantileyebilmektedirler. Sezgisel yöntemler GSP' ler için daha makul zamanlarda optimuma yakın sonuçlar sağlayabilmektedirler (Çevre, 2008:10).

Kesin çözüm yöntemleri genellikle, GSP' nin tamsayılı doğrusal programlama formülünden türeyen yaklaşımlardır. Ancak bu yöntemler, hesaplanabilirlik açısından maliyetli olmaktadır.

Kesin çözüm algoritmaları en iyi optimum çözümü garanti etmekle birlikte dal - kesme ve dal - sınır yöntemleri daha çok büyük algoritmalarında zamandan kazanç sağlamaktadır. Şimdiye kadar GSP için pek çok kesin çözüm algoritması önerilmesine rağmen, en iyi yöntem dal-sınır yöntemi olmuştur (Laporte, 1992:233-240). GSP'nin çözümünde kullanılan yöntemler aşağıda tanıtılmıştır;

1.1.4.1. Dal - Sınır Yöntemi

Dal-sınır yöntemi ilk defa Dantzig ve arkadaşları tarafından 1954 yılında 49 şehirlik problemi çözmek için kullanılmıştır.

Tamsayılı Programlama literatürün de en çok bilinen algoritmalarından birisi Dal-Sınır yöntemidir. Bu yöntem, problemin optimum çözümünü araştırırken, problemle ilgili bütün bölümleri de analiz eder. Öncelikle, problemin mümkün çözümlerinin toplam kümesi, daha küçük alt kümelere parçalanırken, bu her bir alt problem içinde üst ve alt sınır değerleri belirlenir. Daha sonra belirlenen bu değerlere göre bazı alt problemler çözümden çıkartılarak optimum çözüme ulaşılır. Yöntem bir alt problemin uygun bölgesindeki noktaları etkili ve zeki bir şekilde birer birer sayarak optimum çözümü bulur. Dal - Sınır yönteminde üç ana adım vardır: Dallandırma ya da ayrıştırma (Branching), Sınırlama (Bounding), Bağlama ya da Dügümleme (Fathoming) Bu üç ana kavram aşağıdaki gibi açıklanmaktadır (Öztürk, 2005:371-378);

Dallandırma ya da ayrıştırma; Çok boyutlu problemlerin direk çözümü çok zor olduğu için problem daha küçük alt problemlere ayrıştırılır (dallandırılır) ve alt problemler bağlanıncaya kadar ayrıştırma işlemi devam eder. Uygun çözümler kümesindeki her bir çözüme “aday çözüm” denir. Aday çözümler başlangıçtaki probleme dallandırılmış kısıtların ilave edilmesi ile ve bunu takip eden dallandırma işleminin gerçekleştirilmesiyle meydana gelen çözümlerdir.

Sınırlama; Bir problemde optimum çözüme ulaşmada kullanılan dallar ile ilgili yapılan azaltma işlemidir. Sınırlama, dallanma esnasında meydana gelen tüm alt problemlerin (en kaliteli uygun çözüm) optimum çözümünün alabileceği değer en küçük alt sınırını ve en büyük üst sınırını belirler. Gevşetilmiş alt sınırları çabucak çözüme işlemi ise bu sınırların elde etmenin en bilindik yoludur. Yani gevşetme; bir problemde optimum çözüme ulaşmada zorluk çıkaracak kısıtın birisinin atılması işlemine denir. Tüm alt problemlere sınırlandırma işlemi uyguladığımızda dal-sınır algoritması duru ve en iyi alt sınır probleminin çözümü olmaktadır.

Bağlama ya da düğümleme; Dallanmanın gereksiz olduğu durumlarda alt problem bağlanmalıdır. Yani bir düğümde çözüm elde edilmişse dallandırma işlemine devam edilmez ve bu düğüme “bağlandı” denir. Söz konusu olan aday çözüm uygun çözümler kümesinden çıkarılır ve artık dikkate alınmaya devam edilmez.

Aşağıdaki üç durumda bağlama işlemi yapılır;

- Alt problemin uygun çözümü olmadığı durumlarda (Uygun çözümün olmayışı optimum çözümün de olmaması demektir).
- Alt problemin tüm değişkenlerinin tamsayı değerli çözüm verdiği zaman.
- Alt problemin optimum çözüm değeri, maximizasyon (en büyükleme) problemlerinde diğer alt problemlerin çözümünden daha küçük olduğu zaman göz ardı edilip bağlanır.
- Aşağıdaki durumlarda da bir alt problem bağlanır (atılır);
- Uygun çözümlü bir alt problem olmadığı zaman,
- Aday çözüme sahip bir alt problemin z değeri, diğer aday çözüme sahip bir alt problemin z değerinden küçük olduğu zaman, (Minimizasyon (en küçükleme) probleminde de bunun tam tersi).

Dal- Sınır yöntemi optimum çözüme ait noktaların dışındaki bütün noktaları etkin bir şekilde elemesi sonucu optimum çözüme daha hızlı bir şekilde ulaşmaktadır. Dal ve Sınır algoritmasının adımları şöyledir (Bakır ve Altunkaynak, 2003:346);

Adım 1. Başlangıç; Verilen probleme alt sınır stratejisi uygulanıp minimizasyon (enküçükleme) fonksiyonu için alt sınır bulunur. Problem bağlanmışsa yani düğümlenmişse optimum çözüm elde edilmiş demektir ve algoritma durdurulur. Eğer gevşetme işlemi uygulanan alt sınır uygun çözüme sahip değilse ilk verilen problem de uygun çözüme sahip değildir ve durdurulur. Yukarıda bahsedilen iki durum dışında ise verdiğimiz ilk problem için bir sonraki adıma geçilir.

Adım 2. Genel adımlar; Dallandırma işlemi uygulayabilmek için arama stratejisi uygulanarak alt problemler oluşturulur ve bu üretilen alt problemlere alt problemler stratejisi uygulanır. Uygun olmayan çözümler listeden çıkartılır ve düğümlenmiş alt problemler var ise optimum çözüm güncellenir. Düğümlenmemiş çözümler listeye eklenerek yukarıdaki adımlar gerçekleştirilir.

1.1.4.2. Düzlem - Kesme Yöntemi

Düzlem - Kesme yöntemi, dal-sınır yöntemi ile çok yüzlü (polyhedral) yöntemin birleşimiyle oluşmaktadır. Bu yöntem büyük boyuttaki bazı zor problemleri başarı ile çözmüştür. Düzlem -Kesme metodunun en önemli yanı, GSP için olurlu çözümlerin ortaya çıkmasına yarayan yüzlem (*facet*) eşitsizliklerinin bulunmasıdır.

Güçlü bilgisayarların gelişmesiyle ve etkin Düzlem - Kesme algoritmalarının ortaya çıkmasıyla büyük boyuttaki GSP problemlerinin optimal sonuçları bulunmuştur. Crowder ve Padberg (1980) tarafından, 48 şehir ile 318 şehir arasındaki problemler alt tur elemesi ve tarak kısıtları sayesinde çözülmüştür. Padberg ve Rinaldi (1987) tarafından dal-kesme yöntemi kullanılarak, üst sınırı bulan sezgisel yöntem ve prosedür seti ile uymayan eşitsizliklerin saptanması yöntemiyle 532 şehirlik bir problem çözülmüştür. 1991 yılında büyük bir gelişme olmuş ve o zamana kadarki çözülebilen en büyük problem olan 2392 şehirlik problem, Padberg ve Rinaldi (1991) tarafından çözülmüştür.

Düzlem Kesme Algoritmasında tamsayı kısıtları gevşetilerek elde edilen doğrusal programlama problemini çözmeye dayanmaktadır. Ancak Dal - Sınır yöntemindeki kesikli değişkenler üzerine iteratif olarak kısıtlar koymak yerine, kısıtlar dizisi ya da “kesme” ler üretilerek probleme ilave edildikten sonra yeniden çözüm bulunur (Bakır ve Altunkaynak, 2003:386).

Düzlem Kesme yönteminde her iterasyonda bir veya daha fazla kısıt ilave edilirken aşağıdaki durumların sağlanması gerekmektedir;

- Orijinal doğrusal program çözüm kısıtlarını sağlayan uygun çözüm değildir.
- Herhangi bir uygun tamsayı, çözüm noktasını uygun olmayan hale getirmemelidir.

Kısıt ekleme süreci optimal çözüm elde edinceye kadar devam eder.

Dantzig'in Düzlem - Kesme Yöntemi; Düzlem - Kesme yönteminin ilk fikri Dantzig ve arkadaşları (1954) tarafından ortaya konmuştur ve 49 şehirlik bir problem çözülmüştür.

Gevşetilmiş bir doğrusal programlama modeli çözüldüğünde x_j değişkeninin tamsayı olmayan değer almış olduğunu varsayarsak orijinal çözümde temelde olmayan en az bir değişken pozitif değere sahip olmalı ve bu değer tamsayı olması gerektiğinden, orijinal çözümde temelde olmayan değişkenlerden en az biri “1” ya da daha büyük bir değere eşit olmalıdır. A temel çözümde olmayan değişkenler kümesi olsun ve 1 değeri en küçük pozitif değer olduğundan her tamsayılı çözüm için “Dantzig Kesme” kısıtı,

$$\sum_{j \in A} x_j \geq 1 \text{ sağlanmalıdır.}$$

Orijinal çözümde temelde olmayan değişkenler sıfırdır, yani yukarıdaki eşitsizliğin sol kısmı sıfıra eşit olacağından kısıt sağlanmamış olmaktadır. Bu yüzden kısıt doğrusal programlama gevşetme problemine eklenir ve böylece orijinal çözüm düzlemini keser.

Gomory Düzlem – Kesme Yöntemi; İlk kez Gomory tarafından 1958 yılında bir doğrusal programlama probleminin tam sayılı çözümlerini elde edebilecek hesaplama

yöntemi geliştirilmiştir. Bu yöntem sonlu sayıda işlemden sonra optimum tam sayılı çözümü sağlar ve simpleks yönteminden daha fazla işlem gerektirmektedir. Yani Gomory keseni Dantzig'e göre daha fazla bilgi kullanır ve doğrusal programlama gevşetmesinin uygun çözüm bölgesinden daha fazla kesme yapmaktadır.

Bu yöntemde, karar değişkenleri üzerindeki tamsayı kısıtları gevşetilerek simpleks yöntemi ile çözülür. Elde edilen çözüm tamsayılı ise orijinal Tamsayılı programlama probleminin çözümü de bulunmuş olur. Bunun dışında meydana gelen durumlarda ise Gomory kesenini tanımlayan kısıt eklenerek simpleks yöntemi ile bazı çözümler atılmaya çalışılır. Bu şekilde devam edilerek daraltılan uygun çözüm uzayında uygun bir tamsayılı programlama çözümüne ulaşılmış olunur.

Düzlem - Kesme yönteminde simpleks sonucu elde edilen kesirli karar değişkenlerinin tamsayılı kısımları denklemin sağına, kesirli kısımlar sol tarafına yazılarak, sağ taraftaki kısım atılır ve kesirli elemanlar bırakılır. Böylece eşitsizliğe dönüşen denkleminin sol kesirli kısmı sağ taraftaki değerden büyük veya eşit olacaktır. Ek kısıtlayıcılar da elde edilerek tamsayılı olmayan optimum çözüm bulunur.

Eğer kesirli kısımdaki karar değişkenlerinde negatif bir sayı varsa bu sayının eşleniği bulunarak yerine yazılmalıdır. Yani " $x - y = \text{tam sayı}$, x eşlenik y , gösterim olarak $x \equiv y$ " şeklindedir. Düzlem Kesme Yöntemi' nin adımları şöyledir (Bakır ve Altunkaynak, 2003:393-395);

Adım 1: Verilen tamsayılı programlama problemi kesirli kısıtlı bırakılarak doğrusal programlama gevşetmesi simpleks ile çözülür, sonuç tamsayı ise tamsayılı programlamanın çözümü bulunmuştur. Algoritma durdurulur.

Adım 2: Tablodaki son simpleks değerlerinden sıfırdan büyük olan i . satırın düzlem kesme kısıtı türetilir.

Adım 3: Meydana gelen yeni tamsayılı programlama problemi, ilk problemde Adım 2'de üretilen düzlem kesme kısıtı eklenerek türetilip tekrar Adım 2'ye dönlür.

1.1.4.2.1. Dal - Kesme Algoritması

Kesme düzlemi algoritması ile Dal – Sınır algoritmasının birleşiminden oluşur ve tamsayılı programlama problemlerinde etkili bir metottur. Burada da yine doğrusal programlamanın gevşetmesi ile yapılacak çözümü ile başlar.

Genel bir tamsayılı programlama problemini sadece kesme düzlemi yaklaşımı ile verimli olarak çözebilmek mümkün değildir, alternatif optimum çözümleri bulmak için ayrıca dallandırma yapmak gereklidir. Dal-sınır yaklaşımı, kesme düzlemi algoritmasının uygulanması ile oldukça hızlandırılabilir. Dallandırma yapılmadan kesme eklenebileceği gibi ağacın her düğümünün çözüm aşamasında da kesmeler kullanılabilir.

1.1.4.3. Dinamik Programlama

Dinamik Programlama, n değişkenli bir problemin optimum çözümünü, problemi n aşamaya ayrıştırarak ve her aşamada tek değişkenli bir alt problemi çözerek bulur. Dinamik programlamanın önemi ayrıştırmanın her aşamasında optimumluk ilkesini (kalan aşamalar için gelecekteki kararlar, önceki aşamalarda benimsenen optimum politikaya bakılmaksızın oluşturulur) göz önünde bulundurması ve n değişkenli alt problem yerine tek değişkenli alt problemi optimum yapıyor olmasıdır. Dinamik programlamanın uygulama alanları; Kargo yükleme, iş gücü miktarı, teçhizat yenileme, yatırım ve stok modelleri v.s. dir (Taha, 2005:403-427).

1.1.4.4. Sezgisellik (Heuristic)

Problemin durum uzayı çok büyük olduğunda çözümün aranmasını kesin biçimde sınırlayan herhangi kural, strateji, sadeleştirme ve diğer etmenlerin kullanımınıdır. Fakat çözüm kurucu sezgiseller ve yerel arama ve yerel arama algoritmaları gibi bir kere çalıştırmaya dayalı algoritmaların dezavantajı;

- Kısıtlı sayıda farklı çözümler üretirler.
- Düşük kaliteli yerel en iyi çözüm üretirler.

Yerel arama algoritmalarının bu dezavantajı algoritmanın farklı bir başlangıç çözüm ile yeniden çalıştırılması sayesinde ortadan kaldırılabilir. Ancak bu yol ile de çözüm kalitesi iyileştirilemez. Bu dezavantajları ortadan kaldırmak amacı ile çözüm uzayını etkin bir şekilde aramayı sağlayacak temel sezgisel yöntemleri birleştiren yeni yaklaşımlar geliştirilmiştir. Bu yöntemler metasezgiseller (metaheuristics) olarak adlandırılmaktadır.

1.1.4.5. Metasezgisel (Metaheuristics);

Metasezgisel, arama uzayında araştırma (exploring) ve işletme (exploiting) için farklı kavramları zeki bir şekilde birleştirerek alt seviye sezgisellere rehberlik eden iteratif üretim sürecidir (Osman ve Laporte, 1996:2-3). Kısacası metasezgiseller, farklı metotlar ile arama uzayının araştırılması için yüksek seviye stratejilerdir.

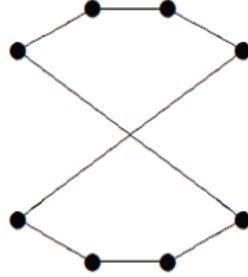
Sezgisel yöntemler hiçbir zaman kesin çözümü garantilemezler, kesin çözüme yakın sonuçlar verirler. Kesin çözüm algoritmaları daha uzun zamanda ve sadece küçük boyuttaki problemleri çözmeye yararken, sezgisel yöntemler bilgisayar ortamında daha makul bir zaman içinde kesin çözüme yakın çözümler elde eden yöntemlerdir. Optimizasyon ve yapay zeka alanlarında bilim adamları ve akademisyenler tarafından en çok araştırılıp, çözüm geliştirilen algoritmalar Gezgin Satıcı Problemi üzerinden yapılmaktadır. n boyutlu bir GSP' nin yaklaşık algoritma karmaşıklığı, $O(n(\log_2 n)^{O(c)})$ olup, GSP'nin çözümünde kullanılan sezgiseller aşağıdaki gibi üç ana gruba ayrılır (Davendra, 2010:13-17).

- Tur Oluşturucu Sezgiseller: Mevcut olan bir tura yeni bir şehir ilave edilerek tam bir turun oluşturulmasıyla çalışır. Bir sonuç bulunduğu sonuca daha da geliştirmek için iyileştirme yapılmayıp algoritmanın çalışması durdurulur ve en iyi çözümden %10-15 sapma söz konusudur. En iyi bilinen tur oluşturucu sezgisellerden bazıları şöyledir;
 - En Yakın Komşuluk: Bilinen en basit, en açık ve anlaşılır GSP sezgiselidir. Burada amaç rastgele seçilen başlangıç şehriden sonra daima en yakın şehri ziyaret etmektir. Bütün şehirler tura dahil olana kadar bu işlem devam eder. Fakat bu sezgisel her zaman iyi sonuçlar

vermez. Bu sezgisel yöntemin polinomsal karmaşıklık seviyesi $O(n^2)$ olur.

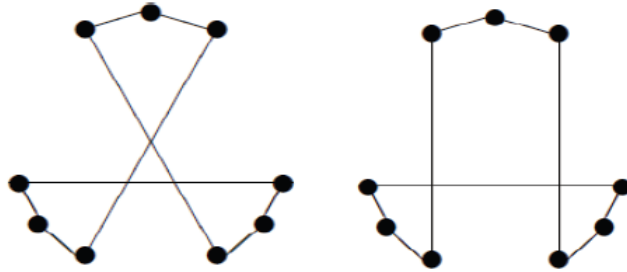
- Açgözlü: Açgözlü algoritmasında her adımda en iyi (minumum) seçilir. Algoritma, döngünün oluşmasını engelleyerek, ağırlıklandırılmış kenarları sıralayarak en kısa kenarı seçme işlemidir.
 - Ekleme: Ekleme algoritmasında bütün düğümlerin alt kümesi olan düğümler ile tur başlar ve diğer tüm düğümlerde bir sezgisel yöntemle göre tura katılırlar. Çoğu zaman başlangıç olan ilk alt tur üçgendir veya iki düğümlü bir kenardan meydana gelmektedir. Ekleme sezgiselinin karmaşıklığı $O(n^2)$ 'dir
 - Christofides: Minimum yayılan ağaç algoritmasına dayanan bu sezgisel, "Christofides Sezgiseli" olarak bilinmektedir. Bu algoritmanın karmaşıklığı $O(n^3)$ 'dür.
 - Clarke ve Wright: Tasarruf algoritması olarak da bilinen Clarke-Wright algoritmasında, k başlangıç şehri olmak üzere tüm $i - j$ ikilileri için tasarruflar hesaplanır. Tasarruflar, $s_{ij} = c_{ki} + c_{kj} - c_{ij}$ şeklinde hesaplanmaktadır. Sonra bu ikililer tasarruf değerlerine göre büyükten küçüğe doğru sıralanır. En büyük değere sahip ikiliden başlanmak üzere, kenarlar alt tur oluşturmayacak ve aynı düğümden ikiden fazla geçiş olmayacak şekilde ilave edilir. Böylece tur tamamlanarak nihai sonuca ulaşılır (Keskintürk v.d., 2016:8).
- Tur Geliştirici Sezgiseller: Tur maliyetini azaltmak için şehirlerin pozisyonunu değiştirerek turu geliştirir. Tur geliştirici sezgiseller; herhangi bir tur oluşturma yöntemiyle kısa sürede elde edilmiş olan yaklaşık çözümle başlayıp, bu çözüme göre optimuma daha yakın bir çözüm bulmayı amaçlar. Yani problem için elde edilmiş olan çözüm turunu sezgisel yöntemlerle iyileştirmeye çalışır. Tur geliştirici sezgiseller şunlardır;
- 2-opt Algoritması: Şekil 1.1 deki 2-opt algoritması, önceden oluşturulan turdan rasgele iki kenarı (bağlantı) kaldırır ve oluşturulan yeni ve geçerli yol yeniden birleştirilir. Yeniden birleştirme, turun geçerli kalması için yapılır. Bu ancak yeni turun eskisinden daha kısa olması durumunda yapılır ve bir

başka iyileştirme yapıncaya kadar devam etmektedir. Bu yöntem ile yaklaşık %5 oranında en iyi çözüme yakın çözümler elde edilmektedir.



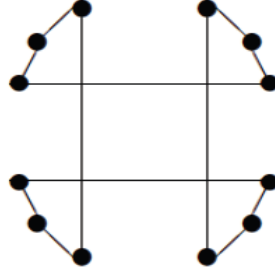
Şekil 1.1. 2 - Opt Algoritması (Kaynak: Davendra, 2010:15)

- 3-Opt Algoritması: 2-opt algoritmasına benzer şekilde çalışmakta olup, 3-opt bir ağ ya da tur içinde farklı üç kenarın (bağlantının) silinmesi ve turun farklı ve geçerli bir yol oluşturacak şekilde yeniden birleştirilmesidir. Yani bu, üç yoldan geçerli bir tura tekrar bağlanmanın iki yolu olduğu anlamına gelir (Şekil 1.2). Artık 3-opt taşımaları, tur kalitesini artırabildiğinde arama tamamlanır. Bu yöntem ile yaklaşık % 3 oranında en iyi çözüme yakın çözümler elde edilmektedir.



Şekil 1.2. 3 - Opt Algoritması (Kaynak: Davendra, 2010:15)

- k - Opt Algoritması: 2-opt ve 3-opt algoritmalarından sonra bu yöntem k için farklı değer alabilme durumu eklenerek geliştirilmiştir. k - Opt algoritması sezgisel olarak üretilen bir turun iyileştirmesini yapmak için uygulanabilir ve $k > 3$ için her zaman daha fazla süreye ihtiyaç vardır. 4-opt algoritmalara “çapraz köprüler” ismi verilmektedir (Şekil 1. 3).



Şekil 1.3. Çift Köprü Hareketi (Kaynak: Davendra, 2010:15)

- Lin – Kernighan Sezgiseli: Lin – Kernighan, 2-opt ve 3-opt sezgisellerinin genelleştirilmiş hali olup melez bir yöntemdir. Kısacası yeni bir tur meydana getirmek için alt turları değiştirirler yani, 2-opt ve 3-opt turu kısaltmak için iki veya üç güzergâhı değiştirirler. Yani Lin – Kernighan sezgiseli ise her bir iterasyonda kaç adet yolun değiştirileceğine karar veren uyarlanabilir bir sezgiseldir. 100 şehirden az problemler için optimum çözüme ulaşmaktadır. Lin – Kernighan sezgiseli Simetrik GSP’ nin çözümü için en etkili yöntemlerden biridir (Helsgaun, 2000:106). Lin – Kernighan’ nın zaman karmaşıklığı fazladır ve basit bir 2-opt uygulamasından daha yavaştır (Davendra, 2010:16).
- Tavlama Benzetimi (TB): Tavlama Benzetimi, bir metalin yüksek sıcaklıklara kadar eritilip, yavaş yavaş soğutma işleminden sonra donarak minimum enerji ile kristalize yapısına dönüşmesi (tavlama süreci) ile daha kapsamlı bir sistemde minimumun araştırılması arasındaki benzerlikten yararlanılmaktadır. Yüksek sıcaklıklarda yani algoritmanın başlarında küresel en iyiyi bulmak ve çözüm uzayını iyi araştırmak için belirli bir olasılıkla kötü çözümlere de izin verilir (Cura, 2008:43). Araştırmacılar komşu çözümleri bulmak için 2-opt yöntemini kullanmışlardır. Tavlama Benzetimi algoritmasının çalışma süresinin arttırılmasıyla daha iyi sonuçlar elde edilebilir olduğu ve sonuçların LK algoritması ile mukayese edilebilir olduğu bulunmuştur. Tavlama Benzetiminin en büyük avantajı, yerel minimumdan kurtulabilme yeteneğine sahip olmasıdır. Tavlama Benzetimi GSP için yakın sonuçlar verecek şekilde başarıyla uygulanmıştır. Tavlama Benzetimi ilk olarak

Kirkpatrick ve arkadaşları tarafından GSP'nin çözümü için kullanılmıştır (Kirkpatrick v.d., 1983). Daha sonra 1985 yılında Cerny tarafından büyük GSP problemlerine uygulanmıştır (Cerny, 1985). 1990 yılında ise Dueck ve Scheuer tarafından eşik değeri (Threshold) kabulü adıyla Tavlama Benzetimi'nin sadeleştirilmiş şekli yayınlanmıştır (Dueck ve Scheuer 1990). GSP' de kullanılan parametrelerin TB' de karşılık gelen hali Tablo 1.1' de verilmiştir.

Tablo 1.1. GSP Parametrelerinin Tavlama Benzetimindeki Karşılıkları

Fiziksel	Meta-Sezgisel
Enerji	Maliyet Fonksiyonu
Sıcaklık	Kontrol Parametreleri
Donma Hali	Yaklaşık Çözüm
Sistem Hali	Uygun Çözüm

- Tabu Arama (TA): Tabu Arama mevcut çözüm çevresinde daha iyi çözümü arayan bir komşu arama algoritmasıdır. Her çözümü araştırabilen bir yöntemdir. Genel olarak, daha iyi bir çözüm aranması için 2-opt değişim mekanizmasını kullanır. Hafıza (memory) kullanımı tabu aramanın önemli bir özelliğidir. Bu hafıza dört ana özelliğe bağlı kalarak işlem görmektedir, bunlar (Cura, 2008:65);
 - Çözümün yakın geçmişte incelenmiş olması
 - Çözümün incelenme sıklığı
 - Çözümün kalitesi
 - Çözümün etkisi

Tabu arama, çözümün ve son hareketlerin hafızasını yönetir. Algoritma öncelikle yerel minimuma doğru hareket ederek başlar. Döngüden kaçınmak için önceden ziyaret edilmiş komşuları ayırır ve yakın zamandaki arama yörüngeleri hafızada tutulur. Bu, "tabu listesi" olarak adlandırılır. Tabu listesi kısa dönem hafıza oluşturur. Tabu Arama'nın her iterasyonunda kısa dönem hafıza yenilenir. Bu yüzden ziyaret edilen tüm çözümleri saklamak zaman ve yer kaybına yol açar.

Olası bir başlangıç çözümü ile araştırmaya başlayan Tabu Arama yöntemi, daha iyi bir çözüme ulaşabilmek için peş peşe hareketler gerçekleştirmektedir ve bu yüzden yerel optimumdan kurtulmak ve diğer bölgeleri araştırabilmek için aşağıdaki stratejileri kullanır (Karaboğa, 2014:51-54);

- Yasaklama Stratejisi; Tabu listesini kontrol eder. Amaç; yeni bölgeler araştırmaktır.
- Aspirasyon Kriteri ve Tabu Sınırlama; Araştırma işleminin sınırlandırılmasında ve araştırmaya yol göstermesinde rol oynarlar.
- Orta ve Uzun Dönem Öğrenme Stratejileri; Orta dönem hafıza, algoritma çalıştığı sırada meydana gelen belirli sayıdaki iyi çözümlerin iyi özellikleri kaydetmek amacıyla çalışır. Uzun dönem hafıza da, tabu arama için farklı başlangıç çözümlerinin araştırılması için çalışır.
- Serbest Bırakma Stratejisi; Tabu listeden çıkacakların zamanını kontrol eder.
- Amaç; daha ileri ki dönemlerde yapılan araştırmalarda bazı tabu çözümlerinin tabu sınırlamalarını kaldırır.
- Kısa Dönem Stratejisi; Yukarıda bahsedilen farklı stratejilerin arasındaki ilişkiyi kontrol eder.
- Tabu aramanın özü olup araştırmanın geçmiş davranışları ile ilgili gerekli bilgileri kaydeder.
- Genetik Algoritma (GA): GA, daha çok fonksiyon optimizasyon problemlerinin ve özellikle birleşik optimizasyon problemlerinin çözümünde, klasik yöntemlere göre daha fazla üstünlükleri olduğu için yaygın bir kullanım alanına sahiptir. Özellikle çözüm uzayı geniş, süresiz ve karmaşık olan problem tiplerinde başarılı sonuçlar vermektedir. GA' nın işleyişindeki ilk adım popülasyonu oluşturup uyum değerini hesaplamaktır. Sonraki işlem genetik algoritma adımları olan; çoğalma, çaprazlama ve mutasyon uygulanıp her oluşan yeni nesil için

uyum değeri hesaplanır ve durdurma kriteri sağlanıncaya kadar işleme devam edilir.

GSP' yi çözmek üzere en sık kullanılan evrimsel hesaplama yöntemi Genetik Algoritmalarıdır. Birçok *NP-zor* problem için GA' nın oldukça başarılı meta-sezgisel bir yöntem olduğu ispatlanmıştır. Daha öncede belirtildiği gibi Hamilton çevrimini gerektiren GSP' nin GA' ya uygulanmasında ilk adım çözüm uzayının kodlanmasıdır. Rotalama problemleri sıralama problemleri olduğu için genellikle permütasyon kodlama yöntemi kullanılmaktadır. GSP' de dolaşılacak olan şehirlerin kromozomlarda tamsayı olarak kodlanması, ziyaret edilecek bir sonraki şehrin indisini ifade eder. Bu çalışma da kullanılan Genetik Algoritma Yöntemi daha detaylı şekilde Bölüm 3' de anlatılmıştır.

- Karınca Koloni Algoritması (KKA): Şimdiye kadar çok farklı çalışmalar yapılan KKO algoritmaları üzerine ilk çalışma Dorigo ve arkadaşları tarafından yapılmıştır. Çalışmalarında “Karınca Sistemi (KS)” ve “Karınca Algoritması” olarak bilinen kendi sistem ve algoritmalarını ortaya çıkarmışlardır (Dorigo v.d., 1996). Gerçek karınca davranışlarından biraz farklı yapıda olan yapay karıncaları kullanmışlardır. Yapay karıncalar, tamamen kör değillerdir ve hafızaya sahiptirler. Algoritmanın çalışma sürecinde temel işlemler, yapay karıncaların turları sonunda geçmiş oldukları yolların feromon miktarlarının artırılması, belirli bir oranda feromon buharlaşmasının gerçekleştirilmesi, en iyi çözümün bulunması, buna bağlı olarak global feromon güncellemesinin yapılması ve karıncaların yenilenen bu feromon miktarlarına bağlı olarak yeni turlarını gerçekleştirmeleridir. Tüm bu işlemlere ait hesaplama yöntemleri ayrıntılarıyla alt başlıklarda verilmiştir (Söyler ve Keskinürk, 2007:3):

Adım 1: Başlangıç feromon değerleri belirlenir

Adım 2: Karıncalar her düğüme rastsal olarak yerleştirilir

Adım 3: Her karınca, sonraki şehri denklemde verilen lokal arama olasılığına bağlı olarak seçmek suretiyle turunu tamamlar.

Adım 4: Her karınca tarafından katedilen yolların uzunluğunu hesaplanır ve lokal feromon güncellemesi yapılır.

Adım 5: En iyi çözüm hesaplanır ve global feromon yenilemesinde kullanılır.

Adım 6: Maksimum iterasyon sayısı ya da yeterlilik kriteri sağlanana kadar Adım 2' ye gidilir.

GSP, KKO algoritmaları için oldukça önemlidir. Karıncalar küçük sorunlara en iyi çözümleri vererek GSP' ye uygulandığında oldukça başarılı sonuçlar vermiştir. İlk KKO algoritması olan Karınca Sistemi, ilk olarak bu problem üzerinde denenmiştir (Dorigo ve Gambardella, 1997). Çünkü bu problem, önemli bir *Np-zor* eniyileme problemi olup. bu tür problemlerin çözüm uzayı çok büyüktür. Bu nedenle optimal çözüm uygun zamanda bulunamayacağından sezgisel yöntemlere başvurulur.

- Parçacık Sürü Optimizasyonu (PSO): Belki de en çok kullanılan ve varyasyonu bulunan metasezgisel yöntemlerden biridir. Eberhart ve Kennedy tarafından 1995 yılında literatüre eklenmiştir. Algoritma doğadaki kuş ve balık gibi sürüce hareket eden canlılardan esinlenmiştir. Aynı bu sürülerdeki canlılar gibi, hem bireysel hem de sürü zekasına dayalı yörüngesel hareket ederek arama uzayının optimum noktasını bulmayı hedefler.

PSO algoritması başlangıçta belirlenen popülasyon sayısına ulaşıncaya dek arama uzayında rasgele çözümleri ifade eden parçacıkları oluşturur. Oluşturulan bu parçacıklara yine rasgele ilk hızları atanır. Ardından bu parçacıkların her birinin kendi en iyi pozisyonları kaydedilir. Daha sonra bu kişisel en iyi pozisyonlardan en uygun olan çözüm kaydedilir. İlk iterasyonla beraber parçacıklar kaydedilmiş kişisel ve sürü geneli en iyi çözümlere göre hızlarını günceller ve yeni pozisyonlarında daha iyi çözümler üretebilmişler ise kişisel ve sürü geneli en iyi çözümü güncellerler. Bu işlem istenen gereksinimler tamamlanana kadar devam etmektedir.

$$v_{i,j}^{k+1} = wv_{i,j}^k + c_1rand_1(p_{ij}^k - x_{ij}^k) + c_2rand_2(g_j^k - x_{ij}^k) \quad (1.8)$$

Her parçacığın hızı Eşitlik (1.8)' e göre güncellenmektedir. $v_{i,j}^{k+1}$ i . parçacığın çözümünün j . parametresinin $k + 1$. İterasyondaki hız değerini ifade etmektedir. Fonksiyonun ilk terimi, $wv_{i,j}^k$, i . parçacığın bir önceki iterasyondaki hızının w katsayısı ile çarpılmış halini ifade etmektedir. w katsayısı, parçacığın önceki hızından alacağı momentumu, ataleti belirtir. p_{ij}^k , j . parçacığın bir önceki iterasyona kadar bulunduğu en iyi pozisyon, x_{ij}^k , parçacığın bir iterasyon önceki pozisyonu, g_j^k ise bir önceki iterasyona kadar bulunmuş genel en iyi çözümü ifade etmektedir. $rand_1$ ve $rand_2$, 0-1 arasında rasgele bir sayıdır. c_1 ve c_2 katsayıları ise bu parçacığın öğrenme katsayılarıdır. c_1 , parçacığın kendi tecrübesine verdiği önemi temsil etmekteyken, c_2 parçacığın sürünün genel tecrübesine verdiği önemi temsil etmektedir. Hızlar belirlendikten sonra her parçacığın bir sonraki pozisyonu Eşitlik (1.9)' a göre güncellenmektedir.

$$x_{i,j}^{k+1} = x_{i,j}^k + v_{i,j}^{k+1} \quad (1.9)$$

PSO algoritmasının başarılı bir optimizasyon işlemi gerçekleştirebilmesi c_1 , c_2 ve w katsayılarının uygun seçilmesine bağlıdır. Çalışmalarda genellikle $c_1=c_2=2$ ve $w > 1$ olarak alınmıştır.

- Melez Sezgiseller: Melez Yöntemler; tur oluşturma yöntemlerinden birisinin desteğiyle başlangıç turunun oluşturulması ve daha sonra tur geliştirme yöntemlerinin bir veya birkaçını kullanarak daha iyi bir tur oluşturmaya çalışan yöntemlerdir. Genellikle daha çok başarılı sonuçlar melez yöntemlerden elde edilmiştir (Özkan, 2010:28).

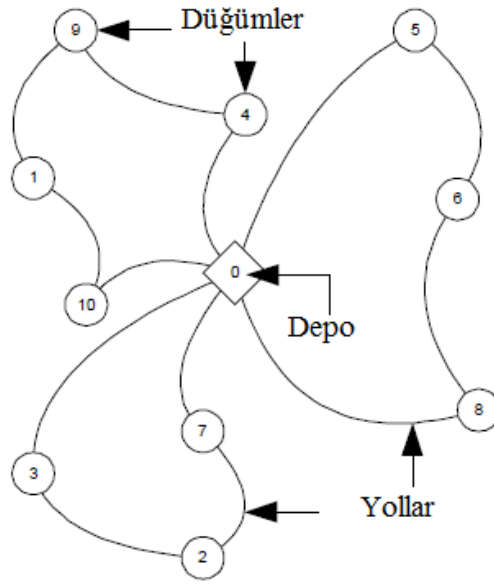
1.2. Araç Rotalama Problemi

1.2.1. Araç Rotalama Probleminin Tanımı

Araç Rotalama Problemi (ARP) ile ilgili ilk çalışma Dantzig ve Ramser tarafından yapılmıştır (Dantzig ve Ramser, 1959). Yapılan çalışmada, benzin istasyonlarına benzin dağıtımını ele alınmış ve başlangıç noktası depodan hareket eden ve yine bitiş noktası olan depoya dönen, müşterilerin ihtiyaçlarını belirlenen kısıtlar altında karşılayan ve taşıma maliyetlerinin veya kat edilen yolun her bir araç için minimize edildiği rotalar kümesi belirlenmiştir (Keskintürk vd., 2015:78). Daha sonra ise Clarke ve Wright ARP için en bilinen sezgisel yöntemlerden biri olan Clarke ve Wright tasarruf algoritması ortaya çıkarmışlardır (Clarke ve Wright, 1964). Üzerinde geliştirdikleri bu problem, araç sayısının sabit olmadığı (karar verme değişkeni olan) problemler için geçerlidir ve hem yönlendirilmiş hem de yönlendirilmemiş problemler için eşit derecede çalışmaktadır.

NP-zor (NP- hard) kombinatoriyal optimizasyon problemlerinden biri olan ARP, Gezgin Satıcı Problemine (GSP) ve Çinli Postacı Problemine (ÇPP) benzerliği ile bilinmektedir. GSP de amaç; bir düğümden başlayarak ve tekrar aynı düğüme geri dönerek, tüm müşterileri (düğümleri) bir kez ziyaret eden bir satıcı için en kısa seyahati bulmaktır. ARP, GSP' nin tek tip ve sınırlı kapasiteye sahip birden fazla aracın kullanıldığı ve bazı eklenmiş kısıtlar ile geliştirilmiş haline verilen isimdir.

Başka bir deyişle ile ARP bir takım kısıtlar beraberliğinde coğrafi olarak dağınık yapıya sahip şehir veya müşteriler için bir veya birkaç depoya dayalı araç filosuna ait rotaların belirlenmesine dayalı bir dizi problemin tümüne verilen genel bir isimdir (Rego, 2001:1). Şekil 1.4' de ARP modeli grafik gösteriminde bir örnek sunulmuştur. Düğümler müşterileri temsil etmektedir.



Şekil 1.4. Araç Rotalama Problemi (ARP) (Kaynak: Machado vd., 2002:2)

Araç Rotalamanın en temel amacı müşteri memnuniyetini arttırmak, mesafeyi, zaman ve maliyeti en aza düşürmektir. Araç rotalama probleminin temel kavramları işletme birimi/birimleri, müşteriler, yol şebekesi, araçlar ve sürücüler olarak tanımlayabilir (Göksal, 2010:4-5). ARP, pratikte bazı kısaltmaları da beraberinde getirir. Bunlar üç başlıkta toplanabilir (Düzakın ve Demircioğlu, 2009:69-70);

1) Araçlarla ilgili kısıtlar

- Araç kapasite (ağırlık ya da hacim)
- Toplam zaman
- Sürücünün çalışma saatleri için yasal sınırlamalar

2) Müşteriler ile ilgili kısıtlar

- Her bir müşterinin bir çeşit ürün talep etmesi ya da belirli türde ürün dağıtılması; Örneğin bu duruma lojistik firmaları düşünülebilir.
- Dağıtımın gerçekleştirilebilmesi için belirli zaman aralıklarının bulunması

3) Diğer kısıtlar

- Aynı araç ile aynı gün içerisinde, aracın depoya dönüp yeniden servise çıkması ile birden çok turun olması
- Bir turun bir günden fazla olması
- Birden çok depo olması

ARP' de dağıtım rotalarının aşağıdaki koşulları sağlaması gerekmektedir:

- Her bir müşterinin talebinin karşılanması gereklidir.
- Her bir müşteri yalnızca bir aracın rotasında bulunmalıdır.
- Bir dağıtım rotasında yer alan toplam müşteri talebi, o rotadaki aracın kapasitesinden düşük olmak zorundadır.
- Her rotanın başlangıç ve bitiş noktası depo olmalıdır.
- Herhangi bir rotadaki toplam alınan mesafe, daha önceden belirlenen maksimum rota mesafesini geçmemelidir.
- Bazı ARP türlerinde n araç sayısı sabit iken, bazı türlerinde sabit değildir.

Bir ARP çözüldüğü zaman tüm müşteri düğümlerine sadece bir kez gidildiği ve izlenen her güzergâhta sadece bir aracın olduğu rotalar kümesi meydana gelmektedir. Meydana gelen bu rotalar kümesi ile bütün kısıtlar sağlanmakta ve ayrıca taşıma maliyetinin, kat edilecek mesafenin veya harcanan zamanın minimize edilmesi gibi durumlar amaçlanmaktadır. Tek depolu klasik bir ARP' nin doğrusal modeli ise aşağıdaki gibidir (Fisher ve Jaikumar, 1981: 2-3);

Parametreler;

K : Araç sayısı

N : Müşteri sayısı, (müşteriler 1'den n 'ye kadar endekslenir, 0 endeksi ise merkez depoyu belirtir)

d_{ij} : i ve j noktaları arasındaki uzaklık

S : Düğümler setinin alt seti

Karar Değişkenleri;

$$x_{ijk} : \begin{cases} 1, & k \text{ aracı } i' \text{ den hemen sonra } j' \text{ ye hareket ederse} \\ 0, & \text{aksi halde} \end{cases} \quad (1.10)$$

$$y_{ik} : \begin{cases} 1, & k \text{ aracı } i' \text{ yi ziyaret ederse} \\ 0, & \text{aksi halde} \end{cases} \quad (1.11)$$

Amaç Fonksiyonu;

$$\text{Min}Z : \sum_{ijk} d_{ij} x_{ijk} \quad (1.12)$$

Kısıtlar;

$$\sum_i x_{ijk} = y_{jk} \quad j = 0, \dots, n, \quad k = 1, \dots, K \quad \text{için} \quad (1.13)$$

$$\sum_j x_{ijk} = y_{ik} \quad i = 0, \dots, n, \quad k = 1, \dots, K \quad \text{için} \quad (1.14)$$

$$\sum_k y_{ik} = \begin{cases} K, & i = 0 \\ 1, & i = 1, \dots, n \end{cases} \quad (1.15)$$

$$\sum_{i,j \in S \times S} x_{ijk} \leq |S| - 1 \quad S \subseteq \{2, \dots, n\} \quad k = 1, \dots, K \quad \text{için} \quad (1.16)$$

$$\begin{aligned} x_{ijk} &\in \{0,1\} & i, j = 0, \dots, n & \quad k = 1, \dots, K \\ y_{ik} &\in \{0,1\} & i = 0, \dots, n & \quad k = 1, \dots, K \end{aligned} \quad (1.17)$$

(1.10)' da x_{ijk} , (1.11)' de y_{ik} karar değişkenlerimizdir. (1.12) amaç fonksiyonu gidilecek toplam mesafenin minimizasyon olması gerektiğini göstermektedir. (1.13) ile (1.14) nolu kısıtlar ise bir araç ile bir düğüm en fazla bir kez ziyaret edilmelidir, (1.15) nolu kısıt her düğüm bir araca atanmalıdır ve her bir güzergâh depoda başlayıp depoda bitmelidir, (1.16) alt tur oluşumunu engelleyen kısıttır ve (1.17) ise değişkenlerin 0-1 tamsayı olmasını ifade etmektedir.

1.2.2. Araç Rotalama Problemlerinin Uygulama Alanları

Lojistik yönetiminin alt dallarından biri olan ARP, ekonomideki öneminden dolayı son zamanlarda çoğu bilim insanının odak noktası haline gelmiş olup yöneylem araştırması, matematik ve bilgisayar bilimlerinde bu soruna çözüm geliştirmek amaçlı zaman ve emeğin harcandığı görülmektedir. Geliştirilen bu çözüm yöntemleri ile dağıtım sektöründe ciddi şekilde harcanan maliyet, mesafe ve zamandan önemli ölçüde tasarruf sağlanması ile işletmelerin, firmaların ve kamu kurumlarının bu yöntemleri kullanmaya başladıkları gözlemlenmektedir.

ARP'nin uygulama alanları şunlardır (Kurul, 2013:14-15; Keskinürk, 2015:80);

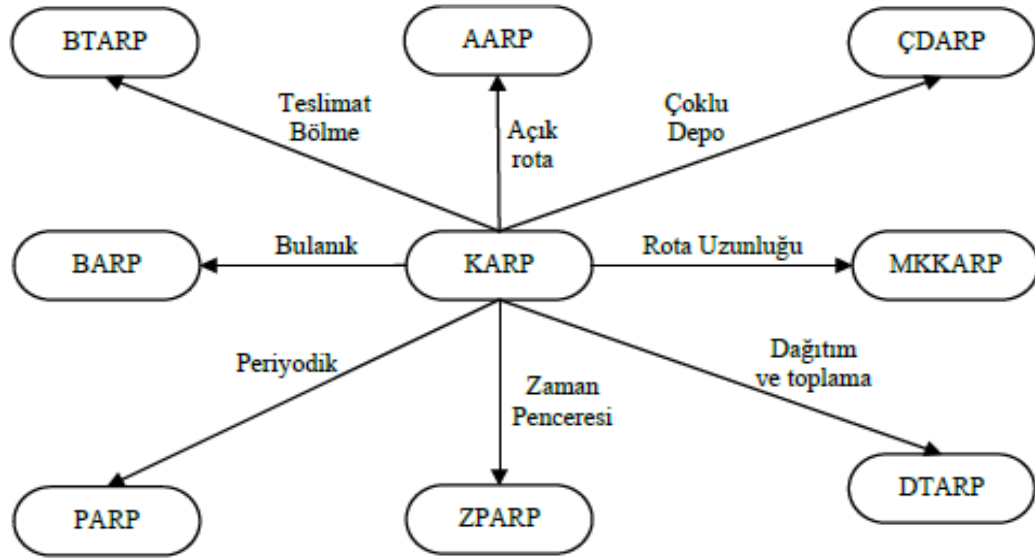
- Ürün ve hizmetlerin bir veya daha fazla sayıdaki depodan, çeşitli müşteri yerlerine dağıtımı,
- Üretim planlamasında hammadde, yarı mamul ve mamullerin fabrikalar arası taşınması,
- Restoran ve kafelere içecek dağıtımı,
- ATM makinelerine para dağıtımı ve zamanlaması,
- Benzin ve mazot dağıtımı,
- Lokantalardan atık yağların toplanması,
- Ev aletleri tamir servisi ve teslimatı,
- İnternet tabanlı ev şirketlerinin ürün dağıtımı,
- Süt toplama ve stok yönetimi,
- Evlerden hayır bağış toplama,
- Taşınabilir tuvalet teslimatı,
- Hapishane ve mahkemeler arası mahkum taşıma,
- Tıbbi ofislerden laboratuarlara idrar örneklerinin taşınması,
- Uçak rotalama,
- Servis araçlarının rotalanması,
- Devriye araçlarının rotalanması,
- Çöp toplama ve aktarma,
- Depolardan perakendecilere toptan dağıtım,
- Posta teslimat kamyonunun yönlendirilmesi,
- Havayolu şirketleri ile yolcu ve ürün taşınması,
- DVD film kiralama,
- Malzeme akış sistemi.

1.2.3. Araç Rotalama Probleminin Çeşitleri

ARP, kolay tanımlanabilmesine rağmen çözümü zordur ve çoğu yaklaşım uygulanmasına rağmen henüz gerçek hayatta karşılaşılan çoğu araç rotalama problemlerinde optimal çözüme ulaşılamamıştır. ARP' nin yalnızca matematiksel değeri olmayıp aynı zamanda gerçek hayatta da çok fazla rastladığımız problemler olan bir çok türde bulunmaktadır. Böylece araştırmacıların daha verimli ve ekonomik çözümlere

ulaşmayı sağlayan yeni algoritmalar ortaya çıkarmalarını sağlamıştır (Düzakın ve Demircioğlu, 2009:68).

ARP'de ağlar, depolar ve araçlar en genel bileşenlerdir. ARP ' nin farklı çeşitlerinin meydana gelmesi için her bir bileşene farklı kısıt ve durumlar ilave edilebileceği gibi her birinin belirli amaçları elde etmesi de sağlanabilir. ARP'nin çeşitleri: kapasite kısıtlı (KARP), mesafe ve kapasite kısıtlı (MKARP), zaman pencere (ZPARP), geri toplamalı (GTARP), dağıtım ve toplamalı (DTARP) araç rotalama ve bunların birleşimlerinden oluşan diğer farklı türlerdir. Son zamanlarda literatürde yoğun olarak çalışılan ARP türlerinin ise; açık (AARP), çoklu depo (ÇDARP), bölünmüş teslimatlı (BTARP), periyodik (PARP), heterojen filolu (HFARP) ve bulanık araç rotalama problemi (BARP) olduğu gözlemlenmiştir (Şahin ve Eroğlu, 2014:338). Araç rotalama probleminin çeşitleri ve aralarındaki ilişkileri Şekil 1.5' de gösterilmiştir.



Şekil 1.5. Araç Rotalama Problemi Çeşitleri ve Birbirleri İle İlişkileri (Kaynak: Toth ve Vigo 2002a:6)

1.2.3.1. Kapasite Kısıtlı Araç Rotalama Problemi (KKARP)

Araç rotalama probleminin en temel yapılarından biri olan kapasite kısıtlı araç rotalama probleminde (KKARP) talepler deterministik, bölünmemiş ve önceden bilinmektedir. Bütün araçların kapasiteleri belli ve birbirine eşittir. Kapasiteler gibi bütün müşteri talepleri de bellidir ve bütün müşteri talepleri karşılanmaktadır. Tek bir depo olup, araçlar harekete depodan başlayıp yeniden depoya dönerek rotalarını

tamamlarlar. Depo ve müşteriler arası mesafeler sabit ve bilinmektedir. KKARP' de amaç bütün müşterilere hizmet vermek için gerekli toplam maliyeti (rota sayısı ve/veya rotaların uzunluk veya seyahat süresi) minimize etmektir (Augerat ve v.d., 1998:546-547).

1.2.3.2. Mesafe Kısıtlı Araç Rotalama Problemi (MKARP)

KKARP' nin bir versiyonu olan Mesafe Kısıtlı Araç Rotalama Problemi (MKARP), belirlenmiş rotalar için kapasite kısıtı yerine aracın gidebileceği maksimum uzunluk (veya zaman) kısıtının kullanıldığı bir ARP çeşididir. Bu kısıt daha çok belirlenen süreyi aşmayacak yolculuklar için, gecikme yaşanmaması gereken durumlarda ve taşınan ürünün bozulabilir olması v.s. gibi durumlarda probleme eklenmesi gereken bir kısıttır (Keskintürk, 2015:83).

1.2.3.3. Zaman Pencere Araç Rotalama Problemi (ZPARP)

KKARP özel bir hali olan zaman pencere araç rotalama problemi (ZPARP), belirli bir zaman aralığında müşteriye uğranılması gereken kısıt durumunda araç rotasının belirlendiği bir problem çeşididir. Her müşterinin hizmet görmesi gereken belli bir zaman aralığının $[a_i, b_i]$ olması, müşteriye ulaşması gereken bu araçların bu zaman aralıklarında ulaşmasını zorunlu hale getirmektedir (Göksal, 2010:8).

1.2.3.4. Dağıt Topla Araç Rotalama Problemi (DTARP)

Dağıt - Topla Araç Rotalama Problemlerinde (DTARP) bir depo, belirli kapasiteye sahip araçlar ve talep miktarları bilinen belirli sayıda müşteri bulunmaktadır. Her bir müşteriye ait olan ürünler belirli zamanlarda alınıp belirli zamanlarda dağıtılmaktadır. Bu problemde planlamanın iyi yapılması gerekmektedir, çünkü araç depodan çıktığı zaman belirlenen yol güzergâhına göre müşterilere dağıtım yapılacak ürünleri dağıtacak ve müşterilerden topladığı ürünleri taşıyabilecek kapasiteye, uygun yol güzergâhına sahip olması gerekmektedir.

Minimum maliyetli rota birleşimini bulmak için DTARP kısıtları (Toth ve Vigo, 2002a:10):

- Her turda depoya uğranılmalıdır,

- Her müşteriye yalnızca bir araç uğramalıdır (yani her müşteri bir turda ziyaret edilmelidir) ,
- Tur boyunca aracın yükü negatif olmamalı ve kesinlikle aracın mevcut kapasitesini geçmemelidir,
- Aynı rota içerisinde depodan farklı dağıtım talebinin merkezi olan müşterisine, toplama talebinin gideceği müşterisinden önce hizmet verilmelidir,
- Her bir müşteri için, müşteri dağıtım talebinin başlangıç yeri depodan farklı ise, aynı tur içinde hizmet verilmeli ve bu hizmet müşteriden önce olmalıdır,

Her bir müşteri için, müşteri toplama talebinin ilk varış yeri depodan farklı ise, aynı tur içinde hizmet verilmeli ve bu hizmet müşteriden sonra olmalıdır.

1.2.3.5. Eş Zamanlı ve Karışık Topla-Dağıt Araç Rotalama Problemi (EZKTDARP)

Eş Zamanlı ve Karışık Topla-Dağıt Araç Rotalama Problemi isminden de anlaşıldığı üzere, toplama ve dağıtım aynı zamanda yapılırken, Karışık Topla-Dağıt Araç rotalama probleminde toplama veya dağıtım işlemi yapılacak müşterilerin karışık sırada ziyaret edilmesidir.

1.2.3.6. Önce Dağıt Sonra Topla Araç Rotalama Problemi (ÖDSTARP)

Burada, öncelikli olarak talep noktalarının ihtiyaçlarının karşılanması gerekmektedir. İhtiyaçların karşılanmasının ardından geri götürülecek ürünlerin veya yükün toplanması süreci başlatılmaktadır. Tam bütünleşik bir süreç yoktur. İleriye doğru ve tersine rotalama süreci kısmen bağımsız olarak yürütülmektedir (Deste, 2013:20). Gidilen mesafenin maliyeti, araçların sayısı ve toplanılmamış ürünlü müşterinin minimizasyonu söz konusu olduğundan problem çok amaçlı olarak ifade edilebilmektedir.

1.2.3.7. Çok Depolu Araç Rotalama Problemi (ÇDARP)

KKARP' nin özel bir hali olan ÇDARP, müşterilere hizmet vermek için dağıtımın birden fazla depoya sahip olduğu araç rotalama problemleridir. Her bir araç belirlenen

her bir depoya tayin edildikten sonra müşterilere hizmet verip tekrar aynı depoya geri döner (Keskintürk, 2009:5).

1.2.3.8. Periyodik Araç Rotalama Problemi (PARP)

Taleplerin belirli günlük periyotlarda her bir müşterinin en az bir kez en çok belirlenen periyotta ziyaret edilip, belirlenen rotanın tamamlanması ve başlangıç noktası olan depoya dönmesi durumudur. Burada araç kapasitesi söz konusu olup her bir müşterinin talebi yalnızca bir araçtan karşılanmaktadır.

Günümüzde rekabetçi bir ortamda bulunan firmaların ürünlerini müşterilerine daha çabuk ulaştırma isteği sebebiyle ortaya çıkan bir ARP türüdür. Amaç müşteriye ulaştırılması gereken ürünün maksimize edilip, toplam tur uzunluğunun minimize edilmesidir (Keskintürk, 2015:87).

1.2.3.9. Bölünmüş Talebe Sahip Araç Rotalama Problemi (BTARP)

Bu problem bir müşteriye bir çok aracın hizmet verebilmesine imkân sunan bir ARP çeşididir.

1.2.3.10. Açık Uçlu Araç Rotalama Problemi (AUARP)

Rotası belirlenen aracın ana depodan yola çıkarak ve kapasite kısıtlarını da göz önünde bulundurarak müşteriye hizmet verdikten sonra tekrar ana depoya geri dönmemesi, bu problemi bilinen klasik ARP' den ayıran en önemli unsurdur. Personel ve okullardaki servis araçlarının turlarının belirlenmesinde daha çok bu problem türü kullanılmaktadır.

1.2.4. Araç Rotalama Problemi Çözüm Yöntemleri

Literatürde Araç rotalama problemleri için bulunan çözüm yöntemleri, kesin çözüm yöntemleri ve sezgisel yöntemler olmak üzere iki şekilde ele alınmıştır. Araç rotalama problemlerinin çözümünde kullanılan kesin çözüm algoritmaları yol formülasyonu temelli olup, önemli ve başarılı etkileri olan bu yöntemler dal ve sınır, dal ve kesme, kesme düzlemi, sütun üretme, dal ve değer algoritması ve dinamik programlama olarak sıralanabilir (Rousseau ve vd., 2007:661; Jin ve vd., 2008:265; Courdeau vd., 2007:371-375). Kesin çözüm algoritmaları orta büyüklükteki problemleri

bile çözümede çok fazla bilgisayar zamanına ihtiyaç duyduğu için istenen beklentiyi pek fazla karşılayamamıştır. Literatürde az sayıda kesin çözüm algoritmaları ile çözülmüş çalışmalar olup araştırmacılar daha çok sezgisel çözüm yöntemlerine yönelmişlerdir.

ARP bir *NP-zor* problemler sınıfında olduğundan *NP-zor* problemleri optimal sonuca ulaştıracak bir algoritma bulunmamaktadır. Bu nedenle araştırmacılar optimal sonuca yakın ve hızlı çözüme götüren, sezgisel algoritmalar üzerindeki uygulamalara yoğunlaşmışlardır. ARP için çözüm arayan bu yöntemler klasik sezgisel algoritmalar ve metasezgisel yöntemler olmak üzere ikiye ayrılmaktadır.

Tur oluşumunu ve gelişimini kapsayan klasik sezgisel yöntemler, Clark ve Wright (1964) tarafından öne sürülen müşteriler arasındaki maliyet tasarrufunu hesaplamakla başlayan, anlaşılması kolay ve esnek olan tasarruf yöntemi, Gillet ve Miller (1974) tarafından önerilen orta ve büyük boyutlu ARP çözümleri için süpürme yöntemi, Christofides ve arkadaşları (1979) tarafından geliştirilen iki aşamalı yöntem ve kısa sürede optimale yakın sonuçlar veren Renaud ve arkadaşları (1996) tarafından önerilen petal yöntemi klasik sezgisel yöntemlerdir (Düzakın ve Demircioğlu, 2009:76).

ARP gibi karmaşık yapıdaki kombinatoriyal optimizasyon problemlerine çeşitli meta sezgisel yöntemler de uygulanmıştır. Klasik sezgisellere nazaran, çözüm alanının daha kapsamlı bir araştırmasını yaparlar ve yerel bir optimumla bitme olasılıkları daha azdır. Bunlar genel olarak üç sınıfa ayrılabilir: (Cordeau ve v.d., 2007:379):

- Tek Çözüm Temelli: Tavlama benzetimi, deterministik tavlama ve tabu arama
- Evrimsel Temelli: Genetik algoritma ve uyarlamalı bellek prosedürü
- Öğrenme mekanizması: Sinir ağları ve karınca kolonisi optimizasyonu

1.2.4.1. Kesin Çözüm Yöntemleri

GSP yöntemlerinin geliştirilmesi ile ARP için olan kesin çözüm yöntemleri oluşmuştur. ARP için optimum çözümler kesin çözüm yöntemleri ile elde edilebilmektedir (Demircioğlu, 2009:56). Bu kısım başlık 1.1.4.1, 1.1.4.2, 1.1.4.3 ve 1.1.4.4' de detaylı şekilde anlatılmıştır.

- Dal ve Sınır Algoritması
- Kesme Düzlemi Algoritması
- Dinamik Programlama

1.2.4.2. Klasik Sezgisel Yöntemler

1.2.4.2.1. Tur Kurucu Sezgiseller

Burada en iyi bilinen tur kurucu sezgisel Tasarruf Algoritmasıdır. Araç kapasitesi göz önünde bulundurularak her adımda turların birleştirilerek uygun çözüm elde edilmesidir.

Dantzig ve Ramser'in Yöntemi

1959 yılında ARP modeli ilk kez Dantzig ve Ramser tarafından ortaya atılmıştır. Dantzig ve Ramser benzin istasyonlarına benzin taşıyan tankerlerin rotasını belirleyen algoritma ile problemin optimal çözümü için ilk kez doğrusal matematiksel modeli kurarak algoritmik bir yaklaşım ortaya çıkarmışlardır (Atmaca, 2012:13).

Clarke ve Wright Tasarruf Algoritması

Tasarruf algoritması olarak bilinen, Clarke ve Wright tarafından 1964 yılında ortaya atılan amacın en iyi turu meydana getiren sezgiseli bulmak olan bir ARP'yi çözme yöntemidir. Bu yöntemde her bir talebe ayrı bir araç hizmet vermektedir. Daha sonra meydana gelen en büyük tasarrufa ve uygunluk değerine sahip iki farklı rota birleştirilir ve böylece iki ayrı noktaya hizmet verecek araç sayısı teke düşürülerek maliyetin minimizasyonu sağlanmış olur (Kosif ve Ekmekçi, 2012:43).

1.2.4.2.2. İki Aşamalı Metotlar

İki aşamalı metotların birinci aşamasında, noktalar araçlara kapasiteyi aşmayacak şekilde atanır. İkinci aşamada ise her bir araç için GSP sezgiselleri kullanılarak rota oluşturulur. Önce gruplama sonra rotalama tarzındaki problemler bu metoda örnek verilebilirler (Eryavuz ve Gencer, 2001:142).

Süpürme Algoritması

Gillett ve Miller tarafından geliştirilen Süpürme Algoritması önce grupta sonra rotala algoritmasının bilinen ilk örneğidir. Düzlemsel bir ARP' ye uygulanan Süpürme algoritması, müşterilerin düzleme yerleştirilmesi ve merkezi depo kabul eden bir algoritmadır. Depodan çıkan ışının kapasite kısıtı dolana kadar saat yönünde döndürülmesi ile belirlenen başlangıç uygun grupları depolar eklenerek GSP ile çözümler.

Fisher ve Jaikumar Algoritması

1981 yılında Fisher ve Jaikumar'ın çalışması geliştirilmiş atama metodu olarak da bilinir. Öncelikle müşteriler arası uzunluklar hesaplanır. Geliştirilmiş Atama Problemi (GAP) Lagranje gevşetme tekniği ile çözülür. İkinci kısımda elde edilem GAP çözümlerine göre GSP sezgiselleri ile çözülür (Önder, 2011:63).

1.2.4.3. Tur Geliştirici Sezgiseller

Tur kurucu sezgisellerden farkı daha elverişli bir çözümle aramaya başlaması, yani her adımda düzeltmeler yapılarak var olan çözümden daha iyi çözümler elde edilmesi amaçlanmaktadır. Bu kısımda 1.1.4.6' da aşağıdaki başlıklar altında detaylı şekilde anlatılmıştır.

- 2-Opt Algoritması
- 3-Opt Algoritması
- k -Opt Algoritması

1.2.4.4. Metasezgisel Yöntemler

Metasezgisel yöntemler, kesin çözüm yöntemleri ile makul bir sürede çözülemeyen karmaşık optimizasyon problemlerinin çözümü için genellikle doğadaki olaylardan esinlenerek tasarlanmış algoritmalarıdır. Metasezgisel yöntemler, esin kaynağı (doğal veya yapay), kullandığı başlangıç çözüm (popülasyon veya tek çözüm), kullanılan amaç fonksiyonu (dinamik, statik), komşuluk yapısı (tekli, çoklu) ve hafıza durumu (hafızalı, hafızasız) gibi kriterlere sınıflandırmaya tabi tutulabilir (Blum ve

Roli, 2003:272-291). Bu kısımda 1.1.4.6 'da ařađıdaki bařlıklar altında detaylı řekilde anlatılmıřtır.

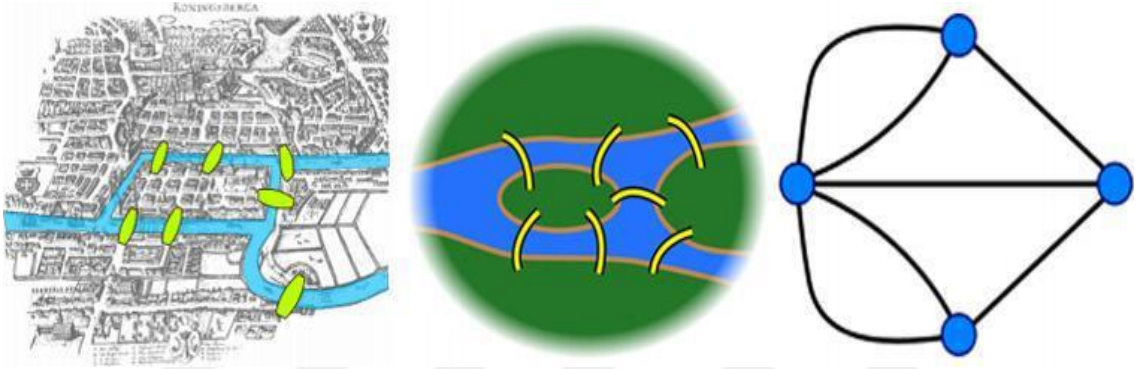
- Yerel Arama
- Tavlama Benzetimi
- Tabu Arama
- Parçacık Sürü Optimizasyonu
- Genetik Algoritma
- Karınca Koloni Algoritması



1.3. Çinli Postacı Problemi

1.3.1. Çinli Postacı Probleminin Tanımı

Ayrıt Rotalama (AR) problemleri, bir graf üzerindeki belirli bir alt kümenin minimum maliyetli geçişini belirlemeyi gerektirir. Königsberg Köprü Sorunu ayrıt rotalama ile ilgili en eski çalışmadır. Bu sorun İsviçreli matematikçi Leonhard Euler tarafından 1736 yılında çözülmüştür. Leonhard Euler Şekil 1.6' da görüldüğü gibi verilen yedi köprünün her birinden tam olarak bir kez geçilebilen kapalı bir yürüyüşün olup olmadığını belirlemeyi amaçlamıştır.



Şekil 1.6. Königsberg 'in Yedi Köprüsü (Kaynak: Eroğlu, 2015:1)

Şekil 1.6' da görüldüğü gibi, Königsberg'in köprüleri ayrıt ve her bir ayrıt adaları ve kıyıları ise dört bağlantı noktasının birleşimini düğümleri temsil etmektedir. Königsberg Köprüsü Sorunu sadece kapalı bir yürüyüşün varlığı ve belirlenmesi ile ilgilidir. Çinli Postacı Problemi temelini de Königsberg Köprüsü sorunundan almakta ve böyle bir çözümün var olmadığı durumlarla ilgilenmektedir. ÇPP her bir bölümü en az bir kez kapsayan bir minimum uzunluk yürüyüşü belirlemektedir.

Birçok pratik uygulamaya sahip iyi bilinen bir kombinatoriyal optimizasyon ve graf teorisinin bir matematiksel problemi olan Çinli Postacı Problemi (ÇPP), Gezgin Satıcı Probleminin (GSP) birleşimi en iyileme temeline dayanan problemlerden olmasına rağmen farklılıkları vardır. ÇPP'de çizge yerine bu çizgeyi bağlayan ayrıtlardan en az bir defa geçilmesi gerekirken (Euler tur) (Eiselt v.d., 1995a, 1995b), GSP' de çizgede bulunan her düğüme sadece bir defa uğraması şartı ile en kısa yolun (Hamilton tur) bulunması amaçlanır. Çinli postacı probleminde eğer Euler tur bulunamıyorsa tek dereceye sahip ayrıtlardan birden fazla geçilmesi gerekecektir ve bunun için de tek

dereceli düğümler aralarındaki toplamda en kısa mesafede (maliyet de) eşleşerek çift dereceli hale gelmesi gerekmektedir.

Çoğu yöneylem arařtırmacıları ve matematikçiler Çinli Postacı Problemi üzerinde birçok modellemeler yapmışlardır. Birçok çeşidi bulunan Çinli Postacı Probleminin üzerinde duran yöneylem arařtırmacıları daha çok klasik ÇPP' nin graflar üzerindeki yapısı ve uygulanabilir çözüm önerileri üzerinde çalışmışlardır.

Klasik ÇPP; her bir ayrıtın tek bir ağırlık ile temsil edildiğini varsayıp, hedefin minimum toplam ağırlık ile turu belirlemek olduğunu anlatmaktadır.

Optimum bir Çinli Postacı rotası için algoritmanın adımları aşağıdaki gibidir:

Adım 1: Tüm tek dereceye sahip düğümler listelenir

Adım 2: Tüm tek dereceli düğümlerin olası eşleşenleri listelenir

Adım 3: Her eşleşme de düğüm çiftleri arasındaki en kısa mesafeli (maliyetli) ayrıtlar belirlenir

Adım 4: Eşleşmede, mesafe (maliyet) toplamı en az olacak şekilde belirlenir

Adım 5: Orijinal grafa 4. adımında bulunan ayrıtlar eklenir

Adım 6: Optimum bir Çinli postacı rota uzunluğu, Adım 4'te bulunan ayrıtlar ile toplama eklenen tüm ayrıtların toplanması ile elde edilen uzunluktur.

Adım 7: Bu minimum uzunluğa karşılık gelen bir rota ve rotanın alternatifleri daha sonra kolayca bulunabilir.

Bir graf da çok fazla tek dereceli düğüm olduğu durumlarda n tek dereceli düğüm sayısı olmak üzere eşleşme sayısını gösteren formül Eşitlik (1.17) deki gibidir (Yüksel, 2013:122);

$$\text{Eşleşme sayısı} = 1 \times 3 \times \dots \times (n-3) \times (n-1) = \prod_{i=1}^{n/2} (2i-1) \quad (1.17)$$

Çinli Postacı Problemi:

A ayrıtlarından ve V düğümlerinden oluşan bir yönsüz $G = (V, A)$ grafını ele alalım. Ayrıt (i, j) i ve j düğümleri arasında bir bağlantı kurar ve c_{ij} ve d_{ij} olmak üzere iki ağırlık ile karakterize edilir. c_{ij} , ayrıt (i, j) 'yi geçme maliyetini gösterirken, d_{ij} onun seyahat süresi veya mesafesini gösterir.

Problemin ana karar değişkeni $x_{ij} = \text{ayrıt } (i, j)$ 'nin kaç kez geçildiğini gösteren değişkendir.

Problemin amacı, denklem (1.18)'de ifade edilen toplam maliyeti en aza indirmektir:

$$\text{Min}Z = \sum_{(i,j) \in A} c_{ij} \times x_{ij} \quad (1.18)$$

Aşağıda kısıtlamalar verilmiştir;

i) Her bir ayrıttan en az bir kez geçilmelidir.

$$x_{ij} + x_{ji} \geq 1 \quad \text{ve} \quad \text{tamsayı} \quad \forall (i, j) \in A \quad (1.19)$$

ii) Her girilen ayrıt için bir düğümdede akış korunmalıdır, yani bir düğüme giren ve çıkan ayrıt sayısı eşit olmalıdır.

$$\sum_{i:(i,j) \in A} x_{ij} = \sum_{i:(j,i) \in A} x_{ji} \quad i \in V \quad (1.20)$$

iii) Negatif olmama kısıtı sağlanmalıdır.

$$x_{ij} \in 0 \quad \text{ve} \quad \text{tamsayı} \quad \forall (i, j) \in A \quad (1.21)$$

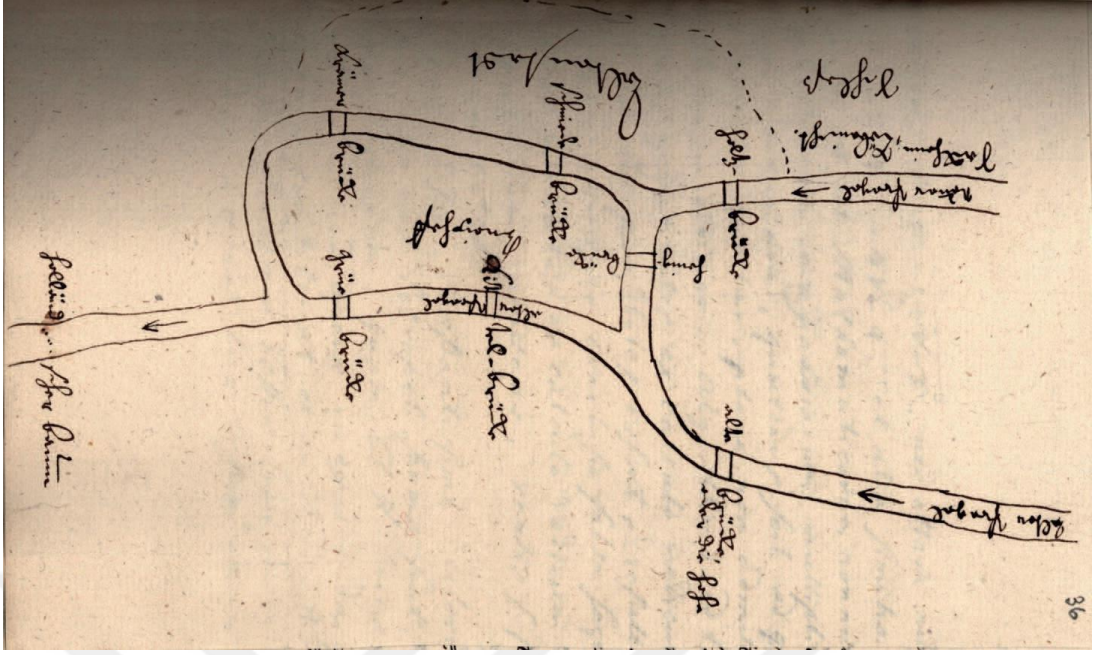
(1.19), (1.20) ve (1.21) nolu kısıtlar için (1.18) nolu minimizasyon problemi minimum maliyetli Çinli Postacı Problemidir.

Çinli postacı probleminin en önemli adımı eşleşme adıdır. Çok büyük graflarda çok fazla sayıda tek dereceli düğümler olduğundan tek dereceli düğümler arası minimum eşleşmenin Eşitlik (1.17)'de de anlaşılacağı gibi günümüz

bilgisayarlarında bile mümkün olmadığı gözlemlenmiştir. Bu nedenle özellikle çok fazla sayıda tek dereceli düğümler arası minimum eşleşmenin sağlanması için Out of Kilter algoritmasına ve sezgisel yöntemlere başvurulmuştur. Teze konu olan çalışmada her bir mahallenin boyutu farklı olup özellikle büyük mahallelerde çok sayıda tek dereceli düğümler vardır. Bu nedenle çalışmada tek dereceli düğümlerin birbirleri aralarında minimum eşleşmelerin sağlanmasında metasezgisel yöntem olan Yapay Arı Kolonisi Algoritması (YAKA) kullanılmıştır. Bunun için YAKA' da yapılan modifiye işlemi başlık 2.2.4' de detaylı olarak anlatılmıştır. Literatürde bu eşleşmenin sağlanmasında genetik algoritma, tabu araması, parçacık sürü optimizasyonu gibi yöntemlere başvurulmuştur. Burada minimum eşleşmenin sağlanmasında kullanılan YAKA yöntemine uygulanan modifiye işlemi ise literatüre kazandırılmıştır.

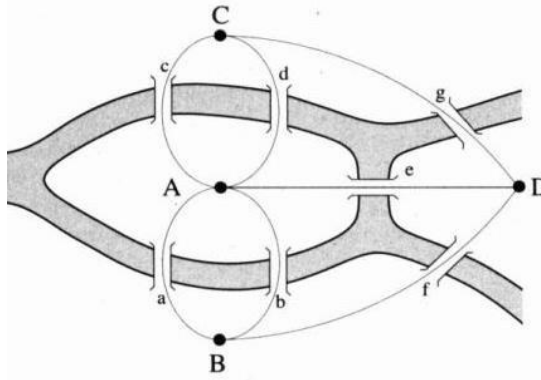
1.3.2. Euler Tur ve Turun Bulunması

Leonhard Euler, Rusya, St. Petersburg'da yayınlanan 21 kısa paragraflı, 13 sayfalık "Solution Problematis and Geometriam Situs Pertinentis" başlıklı bir makale yazdı (Euler,1741). Uzun bir süre ilgi görmeyen bu makale belli bir bulmacayı çözmeye benziyordu. Geçmişe bakıldığında, Euler'in Königsberg'in köprüleri üzerine yazdığı makalede, günümüzde güçlü bir modelleme dili ve aracı haline gelen graf teorisinin temelleri atılmıştır. Euler, kombinatoryel optimizasyon probleminin mimarı olabilecek iken bu fırsatı kaçırarak yerine genç Çinli bir matematikçi olan Guan 1962' de Euler'in köprüsü olarak isimlendirilen ve orijinal problemin bir versiyonu olan Çinli Postacı Problemini dikkate almıştır.



Şekil 1.7. Euler'in Königsberg Çizimi, 1736 (Kaynak: Grötschel ve Yuan, 2012:44)

Euler, Pregel nehri üzerinde bulunan iyi yaka ve ada üzerine kurulmuş olan Königsberg şehrinde (Şekil 1.7), herhangi bir noktadan başlayarak, yedi köprü'nün her birinden yalnız bir kez geçildikten sonra başlangıç noktasına dönülen problemi ele almış ve problemin çözümünün olmadığını göstermiştir. Problem aşağıda ki Şekil 1.8 deki çizgeyle gösterilmiştir.



Şekil 1.8. Pregel Nehrindeki Yedi Köprü Çizgesi (Kaynak: Limon, 2015:2)

Şehirler düğüm (node/nokta), eğri parçaları da kiriş (ayrıt/kenar/edge) olarak adlandırılırsa, bir düğümden başlayıp tüm kirişleri yalnız bir kez kullanarak ve bütün düğümlere uğrayarak başlanan noktaya ulaşmak istediğimizi varsayalım.

Yukarıdaki çizgede, Pregel nehrinin ayırdığı Königsberg şehrinin dört bölgesi A, B, C ve D düğümleriyle ve bu bölgeleri birbirine bağlayan eğriler (köprüler) de ayrıtlarla (kirişlerle) gösterilmektedir. Euler' e göre bir düğüme bir kenar ile geliniyorsa bu düğümden ayrılmak için farklı bir yol gerekmektedir. Böylece bir düğümden giren çıkan ayrıtların sayısına düğüm derecesi (*node order*) adını vermiştir. Bir düğümün derecesi tek ise tek düğüm (*odd node*), çift ise çift düğüm (*even odd*) olarak isimlendirilir. Bulunan düğüm derecelerine göre eğer derecesi tek olan düğüm varsa bu ya bitiş ya da başlangıç düğümü olmalı aksi halde tüm yollar ziyaret edilmiş olmayacaktır.

Euler yaptığı araştırmalar sonucu böyle bir yolun Königsberg'deki köprülerle oluşmasının mümkün olabilmesi için çizgenin bağlı (her bir bölgeye diğer bölgelerden ulaşılabilmesi) ve her bir düğümün çift dereceli olması (her bir bölgeye bağlı köprü sayısının çift olması) gerektiğini ortaya koymuştur. Böylece, bir turun başlangıç ve bitiş düğümü aynı ise ve bu tur her bir ayrıtı tam olarak bir kez içeriyorsa bu tur Euler tur (Euler tour) ve bir çizge Euler tur içeriyorsa bu çizge de Euler çizge (Euler graph) olarak adlandırılmıştır (Saoub, 2017:4-10). Leonhard Euler'in bu araştırmaları matematikte tamamıyla yeni bir dal olan çizge kuramının başlangıcı olmuştur.

Euler Tur Problemi: Bağlı bir graf $G = (N, E)$ verildiğinde, E' de ki her köşeyi tam olarak bir kez ziyaret eden bir tur bulabilir veya böyle bir tur olmadığını belirlenebilir. Euler, eğer G 'de her düğümün derecesi çift olursa, Euler turunun var olduğunu kanıtlamıştır. Euler tur/yol' un adımları şöyledir (Osterhues ve Mariak, 2005:3); $G = (N, E)$ bir Eulerian graf ve v_1 düğümü depo, T, E' deki tüm ayrıtları kapsayan bir turu göstermek üzere;

Adım 1: Ziyaret edilmemiş tüm ayrıtları ve ziyaret edilen v_1 düğümünü işaretleyin, v_1 , T turunun içine dâhil edin,

Adım 2: Tüm ayrıtlar ziyaret edilmemişken

Adım 2.1: Önceden ziyaret edilmiş v_i düğümü ile başlayan bir döngü bulun

Adım 2.2: T turunda, v_i 'nin ilk oluşumundaki döngüyü dâhil edin

Adım 2.3: Döngünün tüm ayrıtlarını ve tüm düğümlerini ziyaret edilmiş olarak işaretleyin

Adım 2.4: Döngüyü bitirin

Adım 3: E' deki tüm ayrıtları kapsayan bir T turu elde edilmiştir.

Euler yol/tur elde edilmesinde literatürde iki farklı algoritma kullanılmaktadır (Söyler ve Fendoğlu, 2018:17);

Euler yol/tur bulmayı gerçekleştiren iki algoritmadan birincisi 1883 yılına dayanan *Fleury* algoritmasıdır. *Fleury* algoritmasında; öncelikle grafın tek dereceli köşelere sahip olup olmadığından ya da sadece iki tane tek dereceli köşeye sahip olduğundan emin olunmalıdır:

Adım 1. Eğer grafın tek dereceli düğümü hiç yoksa herhangi bir nokta başlangıç noktası olarak seçilir. Sadece iki tane tek dereceli düğüm varsa başlangıç noktası bu tek dereceli düğümlerden biri olarak seçilir.

Adım 2. Her bir adımda eğer tek seçenek varsa, grafın henüz gezilecek kısmı için bir köprü seçilmemelidir. Ancak yalnız bir seçenek varsa o seçilmelidir. Geçilen her bir ayrıt grafdan çıkarılır. Bu şekilde devam edildiğinde artık gidilecek yer kalmadıysa Euler döngüsü veya Euler yol tamamlanmış demektir. Tek dereceli düğüm hiç yoksa başlangıç noktasına geri dönülür ve bir Euler döngüsü meydana gelmiş olur. İki tane tek dereceli düğüm varsa, algoritma diğer tek dereceli düğümde son bulacaktır ve bir Euler yol elde edilecektir.

İkinci algoritma, Euler döngüsü elde etmek için farklı bir 1873 yılında yöntem sunan *Hierholzer* algoritmasıdır. *Hierholzer* algoritmasının temel fikri, bağımlı olmayan çevreleri birleştirerek Euler döngüsünün kademeli olarak yapılandırılmasıdır. *Hierholzer* algoritması şöyledir:

Adım 1. Rastgele bir düğümle başlar ve ardından komşu da ziyaret edilmemiş keyfi bir yol izler. Bu adım, başlangıç düğümüne dönene kadar tekrarlanır.

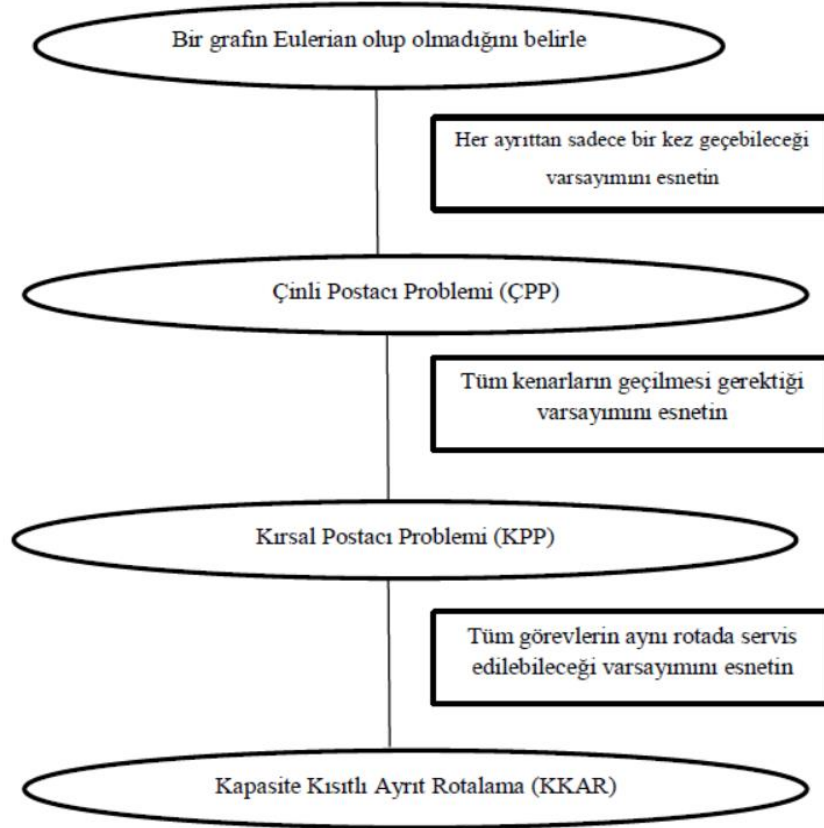
Adım 2. Bu, graf ilk daireyi oluşturur. Bu daire tüm düğümleri kapsıyorsa, bir Eulerian döngüsüdür ve algoritma tamamlanmıştır.

Adım 3. Aksi halde, döngülerin düğümleri arasında gezinilmeyen kenarları olan başka bir düğüm seçer ve 'subtour' adı verilen başka bir daire oluşturur. Yapıda

kenarların seçimi ile yeni daire ilk dairenin kenarını içermez, her ikisi de birbirinden ayrılmıştır. Bununla birlikte, her iki daire, ikinci dairenin başlangıç düğümünün seçimiyle en az bir düğümde kesişmelidir. Bu nedenle, her iki daire yeni bir daire olarak temsil edebilir. Bunu yapmak için, birinci daire düğümlerini yinelemekte ve alt turun düğüm dizgesi tarafından tamamlanan alt tur başlangıç düğümü ile yer değiştirmektedir. Böylece eklenen daireler ilk dairenin içine sokulur. Genişletilmiş daire tüm kenarları içeriyorsa, algoritma tamamlanır. Aksi takdirde, eklenecek başka bir çevrim bulunur.

1.3.3. Çinli Postacı Probleminin Versiyonları

Ayrıt rotalama problemlerinden Çinli Postacı Problemi, Kırsal Postacı Problemi ve Kapasite Kısıtlı Ayrıt Rotalama problemleri ile aralarında ki ilişki Şekil 1.9' da verilmiştir.



Şekil 1.9. Ayrıt Rotalama Problemleri Arasındaki İlişki (Kaynak: Padungwech, 2018:15)

Çinli postacı problemi ayrıt yönlerine bağlı olarak yönlü, yönsüz ve karma olmak üzere üçe ayrılmıştır, fakat daha sonra bu üç probleme farklı kısıtların eklenmesiyle yeni Çinli postacı problem çeşitleri ortaya çıkmıştır. Yönlü ve Yönsüz Çinli Postacı problemleri P sınıfına ait polinomal zamanda çözülen problemler iken, Karma Çinli Postacı Problemi is NP - zor sınıfına ait problemlerdir. Yönlü ve Yönsüz problemlerin zamanında çözülebilir olduğundan araştırmacıların çoğu yakın zamanda Karma ve Hızlı Çinli Postacı Problem türlerine odaklanmaktadır (Corberan ve Prins, 2010:51-52). Bunlar şöyledir;

1.3.3.1. Yönlü ÇPP

Yönlü Çinli postacı probleminde, yönlü bir çizge üzerindeki her bir ayrıta en az bir kez uğranılarak, başlanılan düğüme geri dönmek koşuluyla en kısa yolun bulunması istenmektedir (Ahuja vd., 1993:740).

Yönlü bir grafda büyüme problemini çözme prosedürü hemen hemen eş zamanlı olarak Edmonds ve Johnson (1973), Orloff (1974) ve Beltrami ve Bodin (1974) tarafından önerilmiştir. Karşılaşılmayan durumda yapıldığı gibi tek dereceli köşe noktalarının eşleştirilmesi yerine, grafın simetrik olmasını sağlamak için taşıma algoritması kullanır, yani her bir köşenin iç ve dış dereceleri eşittir (Dror, 2000:6).

Yöneltilen durumda, problem, grafın güçlü bir şekilde bağlanması koşuluyla, uygulanabilir bir çözüm bulur; bu, her düğüm çifti arasında yönlendirilmiş bir yol olması gerektiği anlamına gelir ve ayrıtın ağırlıkları negatif değildir. Optimal çözüm, Eiselt ve diğerleri (1995a) 'de önerdiği gibi bir polinom zaman algoritması olarak bulunmuştur. Thimbleby (2003:6-14) tarafından yapılan çalışmada ÇPP' nin yönlendirilmiş sürümünü çözmek için eksiksiz ve yürütülebilir Java kodu sunulmuştur.

$G = (V,A)$ bağlı grafın da A ayrıtlar kümesidir. G' nin güçlü bir şekilde bağlı olduğunu varsayarsak, aksi takdirde Yönlü ÇPP uygulanabilir değildir.

Her i köşesi için s_i , i den ayrılan ayrıtların sayısının eksisi olarak giren ayrıtların sayısıdır. $s_i < 0$ i köşelerinin kümesi T , $s_i > 0$ köşelerinin kümesi S olsun.

l_{ij} , i den j ye en kısa yol uzunluğunu belirtmek üzere Yönlü ÇPP aşağıdaki gibi formüle edilmiştir (Edmonds ve Johnson, 1973);

$$\text{Min} \sum_{i \in S} \sum_{j \in T} l_{ij} x_{ij} \quad (1.22)$$

$$\sum_{j \in T} x_{ij} = s_i, \quad \forall i \in S \quad (1.23)$$

$$\sum_{i \in S} x_{ij} = -s_j, \quad \forall j \in T \quad (1.23)$$

$$x_{ij} \geq 0, \quad \forall i \in S, \quad \forall j \in T \quad (1.24)$$

Bu Ulaştırma Probleminin Doğrusal Programlama formülasyonudur ve her zaman bir tamsayı optimal çözümüne sahip olduğu bilinmektedir (Dror, 2000:239).

1.3.3.2. Yönsüz ÇPP

Yönsüz Çinli Postacı Problemi P sınıfına ait olup, Yönlü Çinli Postacı Problemine göre daha karmaşıktır. İsminden de anlaşıldığı gibi yönsüz çizge üzerindeki her bir ayrıta en az bir kez uğramak koşuluyla en kısa yolun bulunması sağlanıp başladığı noktaya dönme problemidir. Tamsayılı doğrusal programlama ile de çözülebilen bu problemde, eğer G çizgesi bir Euler çizge ise, tur da tekrar edilen ayrıt olmadan Euler tur bulunup çözülebilir, fakat G bir Euler çizgesi değil ise, tekrarlanan ayrıtların toplamının en kısa mesafe olanı eşleştirme yöntemi ile minimize edilmeye çalışılır (Ahuja vd., 1993:742-743). Edmonds ve Johnson'un 1973' de önermiş olduğu bu matematiksel model de düğümler çift ise Euler turdur fakat Euler tur olmadığı durumlarda G grafının Euler tur olan G' grafına dönüştürülmesi yani, yeni ayrıtların eklenmesi gerekmektedir. G' grafında ki Euler turu G grafında ki Çinli postacı turuna denk gelir ki bu da Çinli postacı probleminin optimal çözümünü elde etmek için G grafına ilave edilecek ayrıtlar en düşük maliyetli olmalıdır. Edmonds ve Johnson, 1973 de ki çalışmasında gösterildiği gibi grafdaki tek dereceli düğümler arasındaki en düşük maliyetli eşleştirmeler belirlenmelidir.

$G=(V,E)$ yönsüz ve bağımlı bir graf, $e \in E$ her ayrıt için c_e negatif olmayan maliyet, x_e Euler graf elde etmek için G grafına eklenecek olan e 'nin kopya sayısını

temsil etmek üzere Yönsüz Çinli Postacının matematiksel modelini Edmonds ve Johnson, 1973 de aşağı da ki gibi formüle etmişlerdir;

$$\text{Min} \sum_{e \in E} c_e x_e \quad (1.25)$$

$$x(\delta(S)) \geq 1, \forall S \subset V, \quad |\delta(S)| \text{ tek} \quad (1.26)$$

$$x_e \geq 0, \quad \forall e \in E \quad (1.27)$$

$$x_e \text{ tamsayı}, \quad \forall e \in E \quad (1.28)$$

Edmonds ve Johnson, ÇPP' yi mükemmel eşleşme problemleri için blossom algoritmasının bir uyarlamasıyla çözmüştür (Edmonds, 1965) ve ilgili polihedronun negatif olmayan (1.27) numaralı eşitsizlik ve tekdüze veya blossom eşitsizliği ile tamamen tanımlandığı (1.26) eşitsizliği ile kanıtlamışlardır (Dror, 2000:236-237).

1.3.3.3. Karma ÇPP

Karma graflar hem yönlü hem de yönsüz ayrıt içerir ve G' nin her ayrıtı boyunca en az bir kez bir en düşük maliyetli geçişi belirlemekten oluşur. Bu *NP-zor* problem için Papadimitriou (1976), Christofides ve vd., (1984) her bir yay için bir değişken, her kenar için iki değişken ve her bir köşe için bir değişken içeren bir formülasyon önermişlerdir. Ardından, Lagrangian tarzda iki tür kısıtlamanın gevşetilmesiyle elde edilen iki farklı alt sınırın, arama ağacının her bir düğümünde hesaplandığı bir Dal - Sınır algoritması uygulanmıştır.

Papadimitriou 1976 yılında, karma graflar için optimal bir çözüm bulmanın *NP-zor* olduğunu göstermiştir. Bununla birlikte, graf eşitse, sorun polinom olarak çözülebilir (Edmonds ve Johnson, 1973).

Uygun çözümü bulma arayışlarına giren bilim adamları tamsayı doğrusal programlama formülasyonlarını kullanmışlardır (Christofides ve vd., 1984; Nobert, 1991). Büyük boyutlu Karma Çinli Postacı Problemini çözmek için ise dal ve kesme algoritması önerilmiştir (Corberan ve ark., 2013). Nobert ve Picard (1996), Karma ÇPP' yi tam olarak çözen bir kesme düzlemi algoritması sunmuştur.

Ford ve Fulkerson'da (1962), karma çoklu bir grafın sadece çift ve dengeli olduğunda Eulerian olduğunu kanıtlanmıştır.

$G = (V, E \cup A)$ bir güçlü bağlı grafi, V düğümler kümesi, E yönsüz ayrıtların kümesi, A ise yönlü ayrıtlar kümesi, $e \in E \cup A$ her ayrıt ve kenarlar için c_e negatif olmayan maliyet olmak üzere, her $S \subset V$, $\delta^+(S)$ $\delta^-(S)$ giren ve çıkan S ayrıtlar kümesini gösterir. $\delta(S)$ her zaman olduğu gibi, $V \setminus S$ deki diğer ve S deki son köşe ile kenarlar setini göstermektedir. $b(S)$, S 'nin dengesizliği olmak üzere $b(S) = |\delta^-(S)| - |\delta^+(S)| - |\delta(S)|$ şeklinde tanımlanmaktadır. Eğer i köşe noktasının derecesi tek ise $tek(i) = 1$, aksi takdirde $tek(i) = 0$ olarak tanımlanmaktadır. Sonra Nobert ve Picard 1996 yılında Karma ÇPP' nin Tamsayı Doğrusal Program modelini aşağıda ki gibi formüle etmişlerdir;

$$\text{Min} \sum_{e \in E \cup A} c_e x_e \quad (1.29)$$

$$x(\delta^+(i)) + x(\delta^-(i)) + x(\delta(i)) + tek(i) = 2z_i \quad \forall i \in V \quad (1.30)$$

$$x(\delta^+(S)) - x(\delta^-(S)) + x(\delta(S)) \geq b(S), \quad \forall S \subset V, \quad S \neq \emptyset \quad (1.31)$$

$$x_e, z_i \geq 0 \text{ ve tamsayı} \quad (1.32)$$

(1.30) ve (1.31) numaralı kısıtlar karma grafın eulerian olması içindir. (1.31) numaralı kısıt dengeli set kısıtlaması olarak isimlendirilir.

Ayrıca Nobert ve Picard 1996 yılında tekdüze-kesme kısıtlarını da aşağıdaki şekilde ifade etmişlerdir;

$$x(\delta^+(S)) + x(\delta^-(S)) + x(\delta(S)) \geq 1, \quad \forall S \subset V, \quad |\delta^-(S)| + |\delta^+(S)| + |\delta(S)| \text{ tek} \quad (1.33)$$

Bu kısıtlar ÇPP için tek düze – kesme kısıtlarını genellemektedir ve sadece bazı durumlarda bir Gomory kesmesinin üretilmesine izin vermek için kullanılan (1.30) denkleminin yanı sıra kesme düzlemi algoritmasında kullanılmaktadır (Dror, 2000:240-241).

1.3.3.4. k – ÇPP

ÇPP' nin daha pratik sürümleri yakın geçmişte tek ÇPP' nin gerçek hayatta pek çok uygulama için gerçekçi olmadığı perspektifi ile ortaya çıkmıştır. Posta teslimatı için genellikle birden fazla postacı vardır ve benzer şekilde çöp toplama için birden fazla araç gereklidir. ÇPP' ye birden fazla postacı eklemek için, k -ÇPP ilk önce toplam maliyetin asgariye indirilmesi amacıyla sunulmuştur. Bununla birlikte, maliyet minimizasyon hedefi, uygulamalar için süre sınırlaması göz önüne alındığında gerçek hayatı temsil etmek için yeterli olmayabilir.

1997'de Thomassen tarafından k -ÇPP kesinlikle bir *NP-zor* problem olarak tanımlanmıştır. Bununla birlikte, sorunun polinomiyle çözülebilir birkaç özel durumu vardır. Pearn (1994) simetrik döngülere sahip k -ÇPP'nin karışık ama eşit ve simetrik ağlarının ve hızlı ağların diğer iki versiyonunda polinom yoluyla çözülebildiğini gösteriyor.

k -ÇPP, klasik ÇPP' nin $k \geq 2$ postacı ile çoklu araç uzantısı olarak dikkate alınabilen Kapasite Kısıtlı Ayrıt Rotalama Problemidir. Sorun kısaca Pearn (1994) tarafından bir dizi k postacı turu bulmak olarak tanımlanmıştır ve şöyle ki;

- Her bir postacı rotası, rota merkezi adı verilen ortak bir düğümden başlar ve biter,
- G' de ki her ayrıta tam olarak bir postacı tarafından hizmet verilmesi gerekir,
- Tüm k postacıları teslimat hizmetine dahil olmalı ve toplam hareket mesafesi en aza indirilir.

k -ÇPP' nin Min-Maks k -ÇPP dışında üç çeşidi daha olup bunlardan ilki Degenhardt tarafından 2004 yılında tanımlanan temeli Dengeli k -ÇPP'ye dayanan Minimum Mutlak Sapma k -ÇPP (MMS k -ÇPP) dir. MMS k -ÇPP' nin amacı, tek bir postacı için izin verilen hizmet süresinden tüm sapmaların toplamını en aza indirmektir. Diğer ikisi ise Osterhues ve Mariak tarafından 2005 yılında tanımlanan Minimum Karesel Sapma k -ÇPP (MKS k -ÇPP) ve Minimum Mesaili k -ÇPP (MM k -ÇPP) dir. MKS k -ÇPP' nin amacı, tüm postacılar için daha da benzer hizmet sürelerine ulaşmaya çalışırken, MM k -ÇPP ise tüm postacılar için fazla mesai toplamını en aza indirmeyi amaçlamaktadır.

1.3.3.5. Min - Maks k -ÇPP

Klasik k -ÇPP'nin bir uzantısı olan Min-Maks k -ÇPP, tüm tur maliyetlerinin toplamını asgariye indirmek yerine maksimum maliyetle tur maliyetini en aza indirmeyi gerektirir. Min-Maks k -ÇPP' nin amacı, en uzun turun süresini en aza indirmektir. Yalnızca tek bir turla ilgilendiği için, diğer turlar arasında dengelenen iş yükü doğrudan incelenmez.

Min-Maks k -ÇPP problemi için ilk algoritma (FHK algoritması) Frederickson ve arkadaşları tarafından 1978 yılında ele alınmış olup NP -zor problem olduğu gösterilmiştir. Ahr 2004 yılında yazmış olduğu tezi ile MM k -ÇPP' yi çözmek için Kapasite Kısıtlı Ayrıt Rotalamaya dal-kesme algoritmasını adapte etmiştir. Bu algoritma Belenguer ve Benavent' in 1998 yılında vermiş olduğu seyrek formülasyon esasına dayanmaktadır. Ahr ve Reinelt (2002) tarafından alt sınırları kullanarak prosedürlerinin kalitesini değerlendirmişlerdir. Daha sonraları ise Ahr ve Reinelt (2006) tarafından bu probleme mevcut çözümlerden üstün olduğu bir tabu arama algoritması önermişlerdir. Belenguer ve Benavent değişkenlerin iki setini kullanmışlardır. Bunlar, y_e^k , her $e \in E$ ayrıtı ve her $k = 1, \dots, K$ için k aracı e ayrıtına servis verdiği zaman bir değerini alan ikili değişken, x_e^k , her $e \in E$ ayrıtı ve her $k = 1, \dots, K$ için k aracı tarafından e ayrıtından kaç defa geçtiğini gösteren tamsayı değişkeni olmak üzere, $1 \in V$ kenarı depoyu temsil etmektedir. $S \subseteq V$ köşelerinin alt kümesi için, $E(S)$, S ' de ki son noktalar kümesi iken $\delta(S)$, S ' de ki tam olarak bir son nokta ile ayrıtılarının kümesini gösterir. $F \subseteq E$ ayrıtıların bir alt kümesi için, $x^k(F) = \sum_{e \in F} x_e^k$ ve $y^k(F) = \sum_{e \in F} y_e^k$ dir.

Ahr (2004) aşağıdaki gibi formüle etmiştir;

$$MM\ k - \text{ÇPP} \quad \text{Min}z \quad (1.34)$$

$$\sum_{k=1}^K c_e (x_e^k + y_e^k) \leq z \quad \forall k = 1, \dots, K \quad (1.35)$$

$$\sum_{k=1}^K y_e^k = 1 \quad \forall e \in E \quad (1.36)$$

$$x^k(\delta(S)) + y^k(\delta(S)) \equiv 0 \pmod{2} \quad \forall S \subset V \setminus \{1\}, \forall k = 1, \dots, K \quad (1.37)$$

$$x^k(\delta(S)) + y^k(\delta(S)) \geq 2y_e^k \quad \forall S \subset V \setminus \{1\}, \forall e \in E(S) \quad (1.38)$$

$$x_e^k \geq 0 \text{ ve tamsayı} \quad \forall e \in E, \forall k = 1, \dots, K \quad (1.39)$$

$$y_e^k \in \{0,1\} \quad \forall e \in E, \forall k = 1, \dots, K \quad (1.40)$$

$$z \geq 0 \quad (1.41)$$

Min-Maks amaç fonksiyonu, çoğu z 'de K turlarının uzunluğunu tanımlayan (1.35) ve K eşitsizlikleri ve (1.34) minimize olması için tanımlanan bir ek değişken z tarafından modellenmektedir. Denklem (1.36) her bir ayrıta servis verilme zorunluluğu ve bağımlı eşitsizlik (1.38) depoya bağımlı her bir tek turu tanımlar. Doğrusal olmayan çözümde her köşenin eşitliğini garantileyen (1.37) kısıtıdır (Corberan ve Laporte, 2013:257).

1.3.3.6. Hızlı ÇPP

Hızlı graflar, bir ayrıtan geçme maliyetinin yönüne göre değişmesine rağmen yönsüz graflardır. 1979 yılında ilk kez Minieka tarafından tanıtılan Hızlı Çinli Postacı problemi, G 'nin her ayrıtını en az bir kez geçerek minimum maliyetli kapalı bir yürüyüş bulmasından oluşmaktadır.

Brucker (1981) ve Guan (1984), HÇPP' nin *NP-zor* problem olduğunu göstermiştir, fakat eğer G Eulerian ise, problem polinomal zamanda çözülebilir. Ayrıca özel durumlar ÇPP, Yönlü ÇPP ve Karma ÇPP içerir.

Win (1989) ve Grötschel & Win (1992), WPP için, daha önce test edilen problemlerin çoğunu çözmeyi başarabilen, daha önceki bir çok merkezli çalışmayı temel alan bir dal-kesme algoritması önermiştir.

Seyahatin yönüne bağlı olarak bir kenarı geçmenin maliyeti $(i, j) \in E$ her ayrıtan ilişkili $c_{ij}, c_{ji} \geq 0$ iki maliyetli yönsüz grafi $G = (V, E)$ olsun. x_{ij} , ayrıtan (i, j) için i ' den j ' ye geçiş sayısıdır.

Win (1989) ve Grötschel & Win (1992) yılında Hızlı ÇPP yi aşağıda ki gibi formüle etmişlerdir;

$$\text{Min} \sum_{(i,j) \in E} (c_{ij}x_{ij} + c_{ji}x_{ji}) \quad (1.42)$$

$$x_{ij} + x_{ji} \geq 1, \quad \forall (i, j) \in E \quad (1.43)$$

$$\sum_{(i,j) \in \delta(i)} (x_{ij} - x_{ji}) = 0, \quad \forall i \in V \quad (1.44)$$

$$x_{ij}, x_{ji} \geq 0 \quad (1.45)$$

$$x_{ij}, x_{ji} \text{ tamsayı} \quad (1.46)$$

Belirli koşullar altında, negatif olmayan eşitsizlikleri, çaprazlama eşitsizlikleri (1.43) ve aşağıdaki tekdüze-kesme eşitsizlikleri, ilgili çokyüzlü (Grötschel & Win (1988)) karşılamaktadır:

$$\sum_{(i,j) \in \delta(S)} (x_{ij} + x_{ji}) \geq |\delta(S)| + 1, \quad \forall S \subset V, \quad |\delta(S)| \text{ tek} \quad (1.47)$$

$$\sum_{i \in S, j \notin S} x_{ij} \geq \frac{1}{2} (|\delta(S)| + 1), \quad \forall S \subset V, \quad |\delta(S)| \text{ tek} \quad (1.48)$$

$$\sum_{i \in S, j \notin S} x_{ji} \geq \frac{1}{2} (|\delta(S)| + 1), \quad \forall S \subset V, \quad |\delta(S)| \text{ tek} \quad (1.49)$$

(1.47), (1.48) ve (1.49) tekdüze-kesme eşitsizlikleri eşdeğerlerdir. (1.47) nolu denklem ayırma algoritmasında kullanılmaktadır. Böyle bir ihlal edilen eşitsizlik bulunduğu, sıfır olmayan katsayıların minimum sayısına sahip (1.48) veya (1.49) biçiminde Doğrusal Programa eklenir (Dror,2000:243-244).

1.3.3.7. Hiyerarşik ÇPP

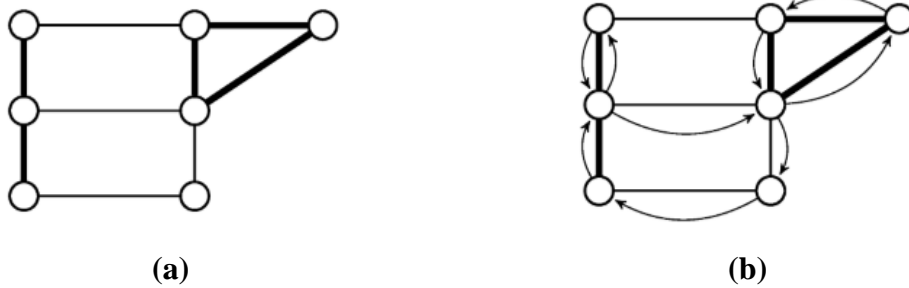
Klasik ÇPP' nin bir diğer çeşidi olan *NP- zor* biçimi Hiyerarşik ÇPP (HÇPP) dir. HÇPP' de, farklı ayırıt alt grupları arasında bir öncelik ilişkisi kurulmuştur. HÇPP genel olarak *NP-zor* problemi olsa da, öncelik ilişkisi doğrusal ise ve her küme bağlı olduğunda polinom zamanında çözülebilir olduğu Dror ve vd., tarafından 1987 yılında ifade edilmiştir. Bu problemde $G=(V,A)$ da ki yönlü ayırıtlar kümesinde öncelik ilişkileri tanımlanmış ve hangi yönlü ayırıtla hangi sırayla hizmet sunulacağı önceden tespit edilmiştir (Ghiani ve Improta, 2000:30). Diğer ÇPP problemlerine göre daha az ilgi

görmüş olan bu problemin çözümü için dinamik programlama algoritması önerilmiştir (Eiselt, 1995a:232).

1.3.3.8. Kırsal PP

1995b yılında Eiselt ve arkadaşlarının yaptığı çalışmaya göre; Kırsal Postacı Problemi, kentsel ortamlarda ortaya çıkma ihtimalinin daha yüksek olduğu Çinli Postacı Problemi' nin aksine, kırsal alanlarda posta gönderimi ile sıklıkla ilişkilidir. Eğer gerekli ayrıtlar bir bağlı grafa neden oluyorsa, yönlü ve yönsüz KPP' ler polinomal (P) olarak çözülür. Aksi takdirde, problem kesinlikle NP zordur. Genellikle Kırsal Postacı Problemi, karma grafların yanı sıra yönlü ve yönsüzler için de bir NP -zor problemidir. Son zamanlarda özellikle son on yılda bu problemin farklı versiyonları birçok araştırmacı tarafından incelenmiştir. Problemin en alakalı çeşitlerine Corberan ve Prins 2010 yılında genel bir bakış sunmuştur.

Kırsal Postacı Probleminde ayrıtlar ve kenarlar olarak isimlendirilen bir $G = (V, E \cup A)$ grafi ve bir $R \subseteq E \cup A$ seti dikkate alınır. Sonra, R ' de ki tüm kenarları ve ayrıtları kapsayan en az uzunlukta bir değişken belirlenir. KPP' yi çözmenin kolay yolu grafin ayrık bileşenleri arasında yolları kullanarak ÇPP' yi çözmektir. Bir bileşen ya servis kenarları ile bağlı köşeler setinden ya da yalnızca depo köşesinden oluşur (Şekil 1.10). Aşağıda verilen bağlı grafin (Adım 5) iki ayrık bileşeni bağlamak için açgözlü paradigma kullanılmıştır. Bu tek bir bileşen bulunana kadar yenilenmektedir, yani tüm servis kenarlarını ve depo noktasını kapsayan bir yayılan ağaç vardır.



Şekil 1.10. (a) KPP' de girdi grafi (b) KPP' de çıktı grafi (Kaynak: Corberán ve Laporte, 2013:36)

G grafin bağlanması için eklenen kenarların turun bir parçası olduğundan emin olmak için ÇPP çözücüne servis kenarları olarak işaretlenmelidir.

$G = (V, (S, C))$ grafi ve ayrıt maliyetleri; $s : S \rightarrow IR_0^+$ (servis kenarlarına ait maliyet), $c : C \rightarrow IR_0^+$ (seyir maliyetine sahip kenarlar) olmak üzere öncelikle bir bağlı graf'ın adımları şöyledir (Osterhues ve Mariak, 2005:7);

Adım 1. C kümesinin kenarlarındaki en kısa yollar $SP(v_i, v_j) \forall i, j \in \{1, \dots, n\}$ 'yi hesaplayın.

Adım 2. $\hat{C} = \emptyset$ olsun

Adım 3. $G' = (V', E')$ ile $V' = \left\{ v \mid v \in S, v \in \hat{C} \right\}$ ve $E' = S \cup \hat{C}$ bağlı değildir ve

$v_1 \notin V'$ dir

Adım 4. Farklı bileşenlerde v_i ve v_j ile minimum maliyetli $SP(v_i, v_j)$ bulun

Adım 5. \hat{C}' e $SP(v_i, v_j)$ de ki kenarları ekleyin

Adım 6. \hat{C} bağlı kenarlar kümesi oluşana kadar devam edin

ÇPP turu hesaplandıktan sonra tur maliyetine sahip kenarlara göre yeniden ayarlanmalıdır (adım 6). Bu şekilde tek bir tur hesaplanabilir.

$G = (V, (S, C))$ bir graf ve v_1 depodur. T, S' 'deki tüm kenarları kapsayan bir tur olmak üzere KPP' nin adımları şöyledir (Osterhues ve Mariak, 2005:7);

Adım 1. $\hat{C} = \text{Bağlı Graf } (G, v_1)$

Adım 2. $S' = S \cup C'$ ile $G' = (V, (S', C))$ ayarlayın

Adım 3. $T = \text{Çinli Postacı Problemi } (G', v_1)$

Adım 4. Tüm kenarlar için $e \in T$

Adım 5. Eğer $e \in \hat{C}$ ise

Adım 6. Kenar tipini servisten seyir maliyetli kenara değiştirin

Adım 7. S' 'deki tüm kenarları kapsayan bir T turu elde edilir.

1.3.3.9. Kapasite Kısıtlı AR

Kapasite Kısıtlı Ayrıt Rotalama problemi (KKAR), ilk kez 1973'te Christofides tarafından, ÇPP' nin genelleşmesi olarak dikkate alınmıştır. Talep kenarlarda yer alan yönsüz ağ üzerinde verildiğinde, hedef rotaların en düşük maliyetli programını belirlemektir. Araç kapasitesini aşmamak için gereken rota talepleri, tek bir depo düğümünde bulunan bir araç filosu tarafından yapılmaktadır.

Verilen yönsüz bir graf olan $G = (N, E, C, Q)$ göz önüne alındığında, burada C bir maliyet matrisi ve Q bir talep matrisi olup ve her birinin kapasitesi W olan (burada $W \geq \max q_{ij}$) özdeş araçların sayısı olmak üzere;

- (1) Pozitif talebe sahip her ayrıta tam olarak bir araçla servis verilmektedir,
- (2) Her araç tarafından servis verilen ağların toplamı W ' yi geçmez,
- (3) Turların toplam maliyeti en aza indirilir.

Kapasite Kısıtlı ARP' yi Golden ve Wong 1981' de şöyle formüle etmişlerdir;

$G=(V,A)$ yönlü grafın da

q_{ij} = ayrıt $(i, j) \in R \subseteq A$ boyunca talep miktarı,

W_v = v aracının kapasitesi,

c_{ij} = bir ayrıt $(i, j) \in A$ nın mesafesi (uzunluğu),

V = araç sayısının üst sınırı,

x_{ijv} = v aracının ayrıt $(i, j) \in A$ dan geçiş sayısı,

y_{ijv} = v aracının ayrıt $(i, j) \in R$ boyunca boşaltma işlemini yapıyorsa 1, aksi takdirde 0 değerini alan ikili bir değişkendir.

$$\text{Min} \sum_{(i,j) \in A} \sum_{v=1}^V c_{ij} x_{ijv} \quad (1.50)$$

$$\sum_{k \in N} x_{kiv} - \sum_{k \in N} x_{ikv} = 0 \quad i \in N, \quad v = 1, 2, \dots, V \quad (1.51)$$

$$\sum_{v=1}^V y_{ijv} = 1 \quad \forall (i, j) \in R, \quad (1.52)$$

$$\sum_{(i,j) \in R} q_{ij} x_{ijv} \leq W_v \quad v = 1, \dots, V, \quad (1.53)$$

$$x_{ijv} \geq y_{ijv}, \quad \forall (i, j) \in R, \quad (1.54)$$

$$M \sum_{i \in S, j \in S} x_{ijv} \geq \sum_{(j,k) \in A[S] \cap R} x_{jkv}, \quad \begin{cases} \forall S \subseteq N, 1 \notin S, \\ A[S] \cap R \neq \emptyset, \\ v = 1, \dots, V, \end{cases} \quad (1.55)$$

$$y_{ijv} \in \{0, 1\}, \quad \forall (i, j) \in R, \quad v = 1, \dots, V, \quad (1.56)$$

$$x_{ijv} \in Z^+, \quad \forall (i, j) \in A, \quad v = 1, \dots, V, \quad (1.57)$$

M , R 'de ki bütün ayrıtları içeren herhangi bir çevrimin toplam mesafesinden daha küçük olmayan büyük bir sabit ve $V[S]$, S de ki en az bir ayrıtlın bulunduğu düğümler kümesidir. Amaç fonksiyonu toplam mesafeyi minimize etmek istemektir. Ayrıtlardan birden fazla geçilebilmektedir. (1.51) nolu kısıt, akış koruma kısıtıdır. (1.52) nolu kısıt, R 'de ki her bir ayrıttan en az bir defa geçiş yapılmasını gerektiren kısıttır. (1.53) nolu kısıt ise araçlar için kapasite kısıtıdır. (1.54) nolu kısıt eğer $(i, j) \in R$ ayrıtlına talebin iletilmesi durumunda v aracının bu ayrıttan geçme durumudur. (1.55) nolu kısıt alt tur eliminasyon kısıtıdır (Dror, 2000:155-156).

Bu problemler $G = (V, A \cup E)$ grafi ve gerekli R kenarları altında farklılık göstermektedir. Düğümler V , yönlü ayrıtlar A , yönsüz ayrıtlar E ile ifade edilmek üzere, çözüm karmaşıklığına göre CPP türleri aşağıdaki Tablo 1.2' de göre özetlenmiştir.

Tablo 1.2. Çinli Postacı, Kapasite Kısıtlı Ayrıt Rotalama ve Kırsal Postacı Problemlerinin Zaman Karmaşıklığı (Kaynak: Corberán ve Laporte,2013:24; Assad ve Golden,1995:377; Longo ve vd.,2004:5; Korteweg ve Volgenant,2006:47; Eiselt ve vd.,1995a:240)

ÇPP Versiyonları	A	E	Gerekli ayrıtlar/ kenarlar R	Karmaşıklık
Yönlü ÇPP	√	-	$R = A$	$P - O(V ^3)$ ve $O((E - V) V ^2)$
Yönsüz ÇPP	-	√	$R = E$	$P - O(V ^3)$
Karma ÇPP	√	√	$R = A \cup E$	$NP - O(V ^3)$
Hızlı ÇPP	-	√	$R = E$	NP - bazı özel <u>durumlarda P</u>
Hiyerarşik ÇPP	√	√	$R = A \cup E$	$NP - O(V ^4)$ bazı özel durumlarda P
<u>Min-Maks</u> <u>k-ÇPP</u>	-	√	$R = E$	$NP - O(V ^3)$ zaman faktörü $(2 - 1/k)$
Kırsal PP Versiyonları	A	E	Gerekli ayrıtlar/ kenarlar R	Karmaşıklık
Kırsal PP	-	√	$R \subset E$	NP
Yönlü KPP	√	-	$R \subset A$	NP
Yönsüz KPP	-	√	$R \subset E$	NP
Kapasite Kısıtlı AR	-	√	$R = E$	$NP - O(V ^2 \cdot Q)$

1.3.4. ÇPP, KPP ve KKAR ile İlgili Literatürde Yapılmış Çalışmalar

Bölüm 1.3.3’ de anlatılan Çinli Postacı Problemi türleri, Kırsal Postacı Problemi ve Kapasite Kısıtlı Ayırıt Rotalama ile ilgili daha geniş bir literatür taraması Tablo 1.3’ de verilmiştir.

Tablo 1.3. ÇPP, KPP ve KKAR ile İlgili Yapılan Çalışmalar

Problem Çeşidi	Yazar İsmi(leri) ve Yılı	Çalışmanın İsmi
<i>Çinli Postacı Problemi</i>	1.Euler (1736/1741) 2.Meï-Ko, K. (Guan) (1962) 3.Edmonds (1965) 4.Eiselt, Gendreau ve Laporte (1995a) 5.Mitra, Ghoshal, Rahaman, Chakrabarty ve Bhattacharya (2010)	1.“Pozisyon geometrisi ile ilgili problemin çözümü” 2.“Tek veya çift noktaları kullanarak graf programlama” 3.“Çinli postacı problemi” 4.“Ayırıt rotalama problemleri, bölüm I: Çinli Postacı Problemi” 5.“Geliştirilmiş eulerizasyon teknikleri ve Çinli postacı problemini kullanarak dijital mikroakışkan biyoçiplerin test edilmesi”
<i>Yönlü Çinli Postacı Problemi</i>	1.Ford ve Fulkerson (1962) 2.Edmonds ve Johnson (1973) 3.Yaxiong ve Yongchang (1988) 4. Thimbleby (2003) 5. Erođlu (2015)	1.“Ađlarda Akışlar” 2.“Eşleştirme, Euler turları ve Çinli postacı” 3.“Yönlü Çinli postacı problemi için yeni bir algoritma” 4.“Yönlü Çinli postacı problemi” 5. “İki amaçlı tek Çinli postacı problemi”
<i>Yönsüz Çinli Postacı Problemi</i>	1.Edmonds ve Johnson (1973) 2.Sebö (1990) 3.Emel, Taşkın ve Dinç (2003) 4.Durucasu (2004) 5. Laporte (2013) 6. Söyler ve Fendođlu (2018)	1.“Eşleştirme, Euler Turları ve Çinli Postacı” 2.“Yönsüz mesafeler ve grafların postacı yapıları” 3.“Yönsüz Çinli Postacı Problemi: Polis Devriye Araçları İçin Bir Uygulama” 4.“Bir Polis Devriye Aracı Rotasının Elektronik Çalışma Sayfası Modeli Yardımıyla Belirlenmesi” 5.“Bölüm 3: Yönsüz Çinli Postacı Problemi. Ayırıt Rotalama: problemler, yöntemler ve uygulamalar” 6.“Malatya Büyükşehir Belediyesi İlaçlama Araçlarının Güzergâhlarının Optimizasyonu”

<p><i>Karma Çinli Postacı Problemi</i></p>	<ol style="list-style-type: none"> 1. Edmonds ve Johnson (1973) 2. Papadimitriou (1976) 2. Minieka (1979) 3. Christofides, Benavent, Campos, Corberán ve Mota, (1984) 4. Minieka (1979) 5. Brucker (1980) 6. Ralphs (1993) 7. Pearn ve Liu (1995) 8. Pearn ve Chou (1999) 9. Nobert ve Picard (1996) 10. Corberán, Martí ve Sanchis (2002) 11. Hua ve Li (2003) 12. Corberán Mejía ve Sanchis (2005) 13. Fernandes, Lee ve Wakabayashi (2009) 14. Gutin, Jones ve Wahlström (2015) 	<ol style="list-style-type: none"> 1. "Eşleştirme, Euler Turları ve Çinli Postacı" 2. "Ayrıttan geçmenin karmaşıklığı üzerine" 2. "Karma ağlar için Çinli postacı problemi" 3. "Karma postacı problemi için en uygun yöntem" 4. "Karma ağlar için Çinli postacı problemi" 5. "Karma graflar için Çinli postacı problemi" 6. "Karma Çinli postacı problemi" 7. "Karma ağlarda Çinli postacı problemi için algoritmalar" 8. "Karma ağlarda Çinli postacı problemi için geliştirilmiş çözümler" 9. "Karma Çinli postacı problemi için en uygun algoritma" 10. "Karma Çinli postacı problemi için bir GRASP sezgiseli" 11. "Çinli postacı problemleri için Genetik algoritma" 12. "Karma genel rotalama problemlerinde yeni sonuçlar" 13. "Sınırlı ağaç genişliğine sahip karma graflar için minimum döngü ve Çinli postacı problemleri" 14. "Karma Çinli postacı probleminin yapısal parametreleştirilmesi"
<p><i>k-Çinli Postacı Problemi</i></p>	<ol style="list-style-type: none"> 1. Zhang (1992) 2. Thomassen (1997) 3. Pearn (1994) 4. Degenhardt (2004) 5. Osterhues ve Mariak (2005) 6. Gutin, Muciaccia ve Yeo (2013). 7. Gutin, Jones ve Sheng (2014) 8. Gutin, Jones, Sheng ve Wahlström (2014) 	<ol style="list-style-type: none"> 1. "k-Çinli postacı problemi için polinomal algoritmalar" 2. "Bir grafın minimum döngüsünü bulma karmaşıklığı" 3. "k-Çinli postacı probleminin çözülebilir durumları" 4. "Dengeli k-Çinli postacı problemi için bir karınca-algoritması" 5. "k-Çinli postacı probleminin türleri" 6. "k-Çinli postacı probleminin parametrik karmaşıklığı" 7. "k-ayrıt Çinli postacı probleminin parametrik karmaşıklığı" 8. "Euler digraflarında parametrelili yönlü k-Çinli postacı problemi ve k-ayrıt parçalı döngüler problemi"

	<p>9.Akyurt, Keskin ve Kalkan (2015)</p> <p>10.Limon (2015)</p>	<p>9.“Kış bakım işlemleri için genetik algoritma kullanma: çoklu depo k-Çinli postacı problemi”</p> <p>10.“k-Çinli postacı dengeleme problemi üzerine”</p>
<i>Min-Maks k-Çinli Postacı Problemi</i>	<p>1.Frederickson, Hecht ve Kim (1976)</p> <p>2. Frederickson (1979)</p> <p>3. Ahr ve Reinelt (2002)</p> <p>4. Ahr (2004)</p> <p>5. Ahr ve Reinelt (2006)</p> <p>6. Arkin, Hassin ve Levin (2006)</p> <p>7. Willemse ve Joubert (2012)</p>	<p>1.“Bazı rotalama problemleri için yaklaşım algoritmaları”</p> <p>2.“Bazı postacı problemleri için yaklaşım algoritmaları”</p> <p>3.“Min-Maks k-Çinli postacı problemi için yeni sezgiseller ve alt sınırlar”</p> <p>4.“Çoklu postacı problemleri için katkılar”</p> <p>5.“Min-max k-Çinli postacı problemi için bir tabu arama algoritması”</p> <p>6.“Minimum ve min-maks araç rotalama problemleri için yaklaşımlar”</p> <p>7.“Güvenlik görevlilerinin yönlendirilmesine min-maks k postacı problemlerine uygulanması”</p>
<i>Hızlı Çinli Postacı Problemi</i>	<p>1. Minieka (1979)</p> <p>2. Win (1989)</p> <p>3.Grötschel ve Win (1992)</p> <p>4.Corberán, Plana ve Sanchis (2007)</p> <p>5.Corberán, Oswald, Plana, Reinelt ve Sanchis (2012)</p> <p>6. Dussault, Golden, GroëR ve Wasil (2013).</p>	<p>1.“Karma ağlar için Çinli postacı problemi”</p> <p>2.“Eulerian graflarda hızlı postacı problemi”</p> <p>3.“Hızlı postacı problemi için bir kesme düzlemi algoritması”</p> <p>4.“Hızlı genel rotalama problemi ve özel durumlar için bir dal kesme algoritması”</p> <p>5.“Hızlı postacı probleminde yeni sonuçlar”</p> <p>6.“Öncelikli kar kazıma: hızlı postacı probleminin bir çeşidi”</p>
<i>Hiyerarşik Çinli Postacı Problemi</i>	<p>1. Dror, Stern ve Trudeau (1987)</p> <p>2. Liu (1988)</p> <p>3.Alfa ve Liu (1988)</p> <p>4.Ghiani ve Improta (2000)</p> <p>5.Cabral, Gendreau, Ghiani ve Laporte (2004)</p> <p>6.Korteweg ve Volgenant (2006)</p>	<p>1.“Ayrıtlar üzerinde öncelikli ilişkili bir graf da postacı turu”</p> <p>2.“Kar temizleme aracı rotalaması: hiyerarşik bir ağda postacı problemi”</p> <p>3.“Hiyerarşik bir ağda postacı rotalama problemi”</p> <p>4.“Hiyerarşik Çinli postacı problemi için bir algoritma”</p> <p>5.“Kırsal bir postacı problemi olarak hiyerarşik Çinli postacı problemini çözme”</p> <p>6.“Doğrusal sıralı sınıflar ile hiyerarşik Çinli postacı problemi üzerine”</p>

<p><i>Kırsal Postacı Problemi</i></p>	<ol style="list-style-type: none"> 1.Christofides, Campos, Corberán ve Mota (1986) 2.Corberán ve Sanchis (1994) 3.Pearn ve Wu (1995) 4.Eiselt, Gendreau ve Laporte (1995b) 5. Benavent ve Soler (1999) 6.Ghiani ve Laporte (2000) 7.Rodrigues ve Ferreira (2001) 8.Corberán, Martí, Martinez ve Soler (2002) 9.Fernández, Meza, Garfinkel ve Ortega (2003) 10.Ghiani, Laganà ve Musmanno (2006) 11.Benavent, Carrota, Corberán, Sanchis ve Vigo (2007) 12.Benavent, Corberán, Plana ve Sanchis (2009) 13.Aráoz, Fernández ve Meza (2009) 14.Palma (2011) 15. Drexl (2014) 	<ol style="list-style-type: none"> 1.“Yönlü bir graf da kırsal postacı problemi için bir algoritma” 2. “Kırsal postacı problemine çok yüzlü bir yaklaşım” 3.“Kırsal postacı problemi için algoritmalar” 4.“Ayrıt rotalama problemleri, bölüm II: Kırsal postacı problemi” 5.“Dönüş cezaları ile kırsal postacı problemi” 6.“Yönsüz kırsal postacı problemi için dal kesme algoritması” 7.“Memetik algoritmalarla kırsal postacı problemini çözme” 8.“Dönüş cezaları ile karma graflarda kırsal postacı problemi” 9.“Yönsüz kırsal postacı probleminde: yeni bir formülasyona dayanan sıkı sınırlar” 10.“Yönsüz kırsal postacı problemi için yapıcı bir sezgisel” 11.“Hızlı kırsal postacı problemi için alt sınırlar ve sezgiseller” 12.“Min - Maks k-araçlar hızlı kırsal postacı problemi” 13.“Ödüllü toplayıcı kırsal postacı probleminin çözümü” 14.“Ödül toplayan kırsal postacı problemi için bir tabu arama sezgiseli” 15.“Genelleştirilmiş kırsal postacı problemi üzerine”
	<ol style="list-style-type: none"> 1.Golden ve Wong (1981) 2.DeArmon (1981) 3.Assad, Pearn ve Golden (1987) 4.Pearn (1991) 5.Benavent, Campos, Corbenin ve Mota (1992) 6.Eglese (1994) 	<ol style="list-style-type: none"> 1.“Kapasite kısıtlı ayrıt rotalama problemleri” 2.“Kapasite kısıtlı Çinli postacı problemi için sezgisel bir karşılaştırma” 3.“Kapasite kısıtlı Çinli postacı problemi:alt sınırlar ve çözülebilir durumlar” 4.“Kapasite kısıtlı ayrıt rotama problemi için augment-insert algoritmaları” 5.“Kapasite kısıtlı ayrıt rotama problemi. Alt sınırlar” 6.“Kışık öğütme araçlarının rotalanması” 7.“Kapasite kısıtlı ayrıt rotalama problemi: geçerli eşitsizlikler ve yüzeyler”

<i>Kapasite Kısıtlı Ayrıt Rotalama</i>	7. Belenguer ve Benavent (1998)	8. “Yönsüz kapasite kısıtlı ayrıt rotalama problemi için bir deęişkenli komşuluk iniş algoritması”
	8. Hertz ve Mittaz (2001)	9. “Kapasite kısıtlı ayrıt rotalama problemi için bir kesme düzlemi algoritması”
	9. Belenguer ve Benavent (2003)	10. “Büyük yönlü CARP örnekleri için Algoritmalar: kentsel katı atık toplama operasyonel desteęi”
	10. Maniezzo (2004)	11. “Karma kapasite kısıtlı ayrıt rotalama problemi için alt ve üst sınırlar”
	11. Belenguer, Benavent, Lacomme ve Prins (2006)	12. “Kapasite kısıtlı ayrıt rotalama problemi için deterministik bir tabu arama algoritması”
	12. Brandão ve Eglese (2008)	13. “Yeni Bir Kapasiteli Ayrıt Rotalama Problemi ve Çözüm Yaklaşımları”
	13. Kirlik (2009)	14. “Kapasite kısıtlı ayrıt rotalama yaklaşımını kullanarak çoklu robot sensör tabanlı kapsama alanı planlaması”
	14. Parlaktuna, Sipahioęlu, Kirlik ve Yazici (2009)	15. “Enerji kısıtlamalarını dikkate alarak çoklu robot sensör tabanlı kapsama için bir dinamik yol planlama yaklaşımı”
	15. Yazici, Kirlik, Parlaktuna ve Sipahioęlu (2009)	16. “Boş sefer talepleri ile kapasite kısıtlı ayrıt rotalama”
	16. Kirlik ve Sipahioęlu (2012)	17. “Bölüm 7: Kapasite kısıtlı ayrıt rotalama problemi: Sezgiseller. Ayrıt rotalama: problemler, yöntemler ve uygulamalar”
	17. Prins (2013)	18. “Stokastik taleplerle birlikte bir ayrıt rotalama problemini çözmek için bir simheuristik algoritma”
	18. Gonzalez ve Martin (2016)	19. “Dinamik kapasite kısıtlı ayrıt rotalama problemi için sezgisel algoritmalar”
	19. Padungwech (2018)	

1.3.5. Çinli Postacı Probleminin Uygulama Alanları

Çinli postacı problemi; mektupların dağıtımı, çöplerin toplanması, cadde ve otobanlarda ki kar ve buz kontrolleri, tuzlama, kar temizleme ve sokakların temizlenmesi çalışmaları, okul servisleri ve polis devriye araçları rotalarının çizelgelenmesi, süt ve gazete dağıtımı, etkili web sitesi kullanılabilirliğinin tespiti ve tesis içi planlama gibi bir çok alanda kullanılabilirliğini tespit etmiştir (Thimbleby, 2003:1082-1084). Özellikle şirketlerin ve kamu alanlarının bu bölümdeki plansız harcamaları nedeniyle büyük oranda kaynağın yanlış şekilde kullanıldığı tespit edilmiştir. Bundan dolayıdır ki bu tarz sorunların giderilmesi için bilim adamları bu problemlerin çözümü için çalışmalarını arttırmışlardır.

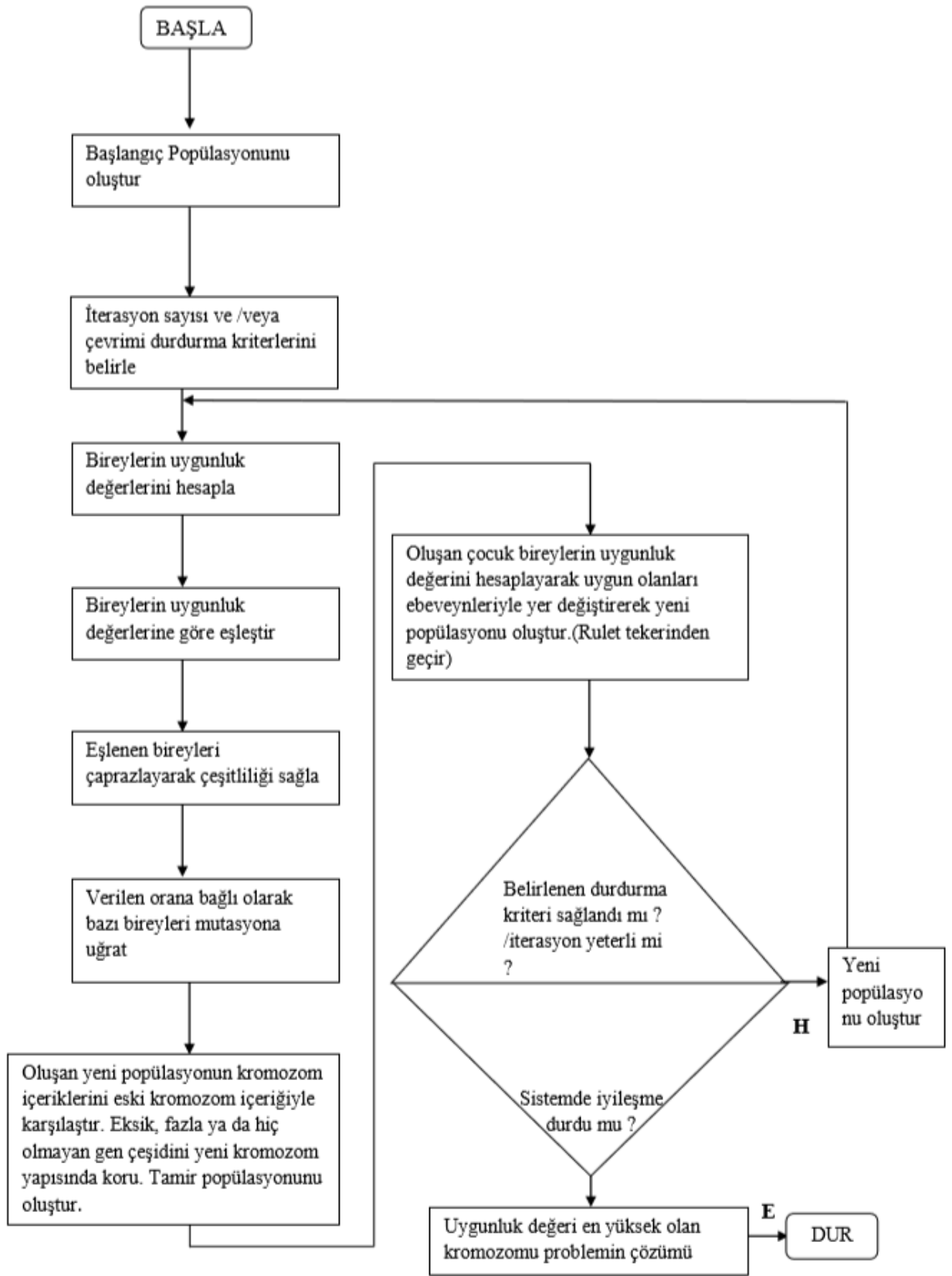
BÖLÜM 2

2. UYGULAMADA KULLANILAN METASEZGİSEL ALGORİTMALAR

2.1. Genetik Algoritma (GA)

Darwin'in evrim stratejisinden ortaya çıkan ve evrimsel algoritmaların başında gelen yapay zeka tabanlı Genetik Algoritma (GA), sezgisel arama ve sayısal optimizasyon algoritmasıdır. En iyinin hayatta kalması kuralından hareketle ve bilgisayar sisteminde doğal seçim ilkesinin uygulanması ile elde edilmiştir. Genetik algoritma ilk kez Michigan üniversitesinden bilgisayar uzmanı ve psikolog olan J.H. Holland tarafından 1975 yılında ortaya çıkarılmıştır. Goldberg (1989), Michalewicz (1996) ve Chambers (1998) optimizasyon problemlerinin çözümünde genetik algoritmayı kullanmışlardır.

Genetik algoritmayı, evrimselleşme stratejisini ve genetik programlamayı içeren evrimsel programlama I. Rechenberg'in (1973) "Evrimsel Stratejileri" isimli çalışmasıyla ortaya çıkmıştır. Bu iki kavram arasındaki en önemli fark ise; evrim stratejisinde sadece mutasyon uygulanırken, genetik algoritmada hem çaprazlama hem de mutasyon işleminin uygulanmasıdır. Genetik algoritma da başlangıçta belirli bir büyüklükte rastsal olarak başlangıç popülasyonu meydana getirilir ve çoğalma, çaprazlama ve mutasyon işlemlerinden geçirilerek yeni bir popülasyon (yavru nesil) ortaya çıkarılır. Bu işlemlerden geçecek olan kromozomlar amaç fonksiyonu değerine göre rastsal olarak belirlenir. Yani amaç fonksiyonuna yerleştirilen kromozomların vermiş olduğu sonuca göre değeri daha iyi olan kromozomların yeniden üretilme ya da çaprazlama olasılığı daha fazladır. Rastsal seçilen kromozomlar mutasyon işlemine de uğrar ve uyum değeri düşük olan kromozomlar bir sonraki nesile aktarılmazlar. Meydana gelen yeni popülasyon eski popülasyon ile yer değiştirir ve her yeni nesilde ortalama uyum giderek artar (Nabiyev, 2010:586-587). GA'nın işleyişi aşağıdaki Şekil 2.1' de verilmiştir.



Şekil 2.1. Genetik Algoritma Akış Diyagramı (Kaynak: Elmas, 2016:425)

Uygunluk fonksiyonu yardımıyla kromozomların uygunluklarının bulunmasına değerlendirme (evaluation) denir ve her bir (iterasyon) tekrarlama ile üretme sağlanmış olur.

Basit bir Genetik Algoritmanın temel adımları şöyledir (Karaboğa, 2014:76);

Adım 1: Olası çözümlerin kodlandığı bir başlangıç popülasyonunu oluşturulur,

Adım 2: Oluşturulan popülasyondaki her çözümün uygunluk değeri hesaplanır,

Adım 3: Durdurma kriteri sağlanıyorsa durdurulur,

Aksi durumda ;

Adım 3.1: Uygunluk değeri yüksek olan çözümler popülasyonda daha fazla yer alırlar kuralına uygun iki popülasyon seçilir,

Adım 3.2: Var olan iki çözümden yeni iki popülasyon üretilir, yani kısaca çaprazlama işlemi uygulanır,

Adım 3.3: Çözümlere rastgele değişim, yani mutasyon işlemi uygulanır,

Adım 3.4: Elde edilen kromozomlar yeni popülasyona eklenir,

Adım 4: Meydana gelen yeni popülasyon eski popülasyonla değiştirilir,

Adım 5: Adım 2' ye dönlür.

Birbirini takip eden yeni popülasyonların (jenerasyonların) elde edilmesi ve bunların da yeniden değerlendirilme döngüsü optimal bir çözüm elde edilinceye kadar devam etmektedir.

Genetik algoritmayı diğer sezgisel algoritmalarından ayıran özellikler ise şöyledir: (Elmas, 2016:408)

- GA, parametrelerinin kendisiyle değil, parametrelerin kodlarıyla ilgilenmektedir,
- GA, sadece tek bir alanda değil yığının tümünde çözümü aramaktadır,
- GA, sapma ve diğer hatalar ile ilgilenmemekte sadece amaçla ilgilenmektedir,
- GA, kesin değil, rassal (sezgisel) yöntemler kullanmaktadır.

2.1.1. Genetik Algoritma ile İlgili Temel Kavramlar

GA' nın çalışmasında ve başarılı çözümlere ulaşılmasında algoritmanın yapısında kullanılan kavramların iyi belirlenmesi gerekmektedir. Bu bölümde GA' nın temel kavramları üzerinde durulacaktır.

2.1.1.1. Gen

Kendi başına anlamlı ve bir bireyin tüm kalıtsal bilgilerini taşıyan en küçük genetik yapı birimine gen denir. Genlerin aldığı değerlerin her birine yani karakterleri belirleyen etmenlere alel denir.

2.1.1.2. Kromozom

Genler dizileri meydana getirir. Genlerin bir araya gelmesiyle oluşan ve problemin çözümüne ait bütün bilgileri taşıyan diziye kromozom denir. Yöneylem araştırmasında problemin amaç fonksiyonunda bulunan değişkenlerin alabileceği değerler kromozomlar ile ifade edilirler (Cura, 2008:88). Bu nedenle kromozomlar genetik algoritmalar için çok önemlidirler ve iyi ifade edilmelidirler.

2.1.1.3. Popülasyon (Yığın)

Kromozomların bir araya gelmesiyle popülasyon (yığın) meydana gelir. GA' nın işleyişi esnasında bazı kromozomlar yok olurken, bazı yeni kromozomlarda eklenerek popülasyon büyüklüğü dengelenmektedir. Popülasyon büyüklüğü için literatürde kesin bir ifade olmamakla birlikte genel olarak 30 – 100 arası kromozomdan oluşmaktadır. Popülasyonun büyüklüğü problemin çözüm zamanını da belirlemektedir. Problemin tipine uygun olarak programı yazan kişi tarafından popülasyon büyüklüğü belirlenebilmektedir (Elmas, 2016:412).

2.1.2. Yeniden Üretim İşlemi

Mevcut popülasyondan yeni popülasyona aktarılacak olan kromozomların seçilmesidir. Aktarılmış olan kromozomlar genetik yapı olarak en uygun değere sahip olan kromozomlardır. Böylelikle “iyinin hayatta kalması” kuralını temsilen en iyi uygunluk değerine sahip bireylerin gelecek jenerasyonlara aktarılması sağlanmış olur (Elmas, 2016:413).

2.1.3. Başlangıç Popülasyonunun Oluşturulması

GA' yı diğer sezgisel algoritmalarından ayıran en önemli fark çözümü noktadan noktaya değil noktaların oluşturduğu yığılda araması olduğu için öncelikle başlangıç yığınının (başlangıç popülasyonunun) oluşturulması gerekmektedir. Başlangıç popülasyonunu belirlemede en önemli konu ise kromozomların nasıl kodlanacağıdır. Aşağıda bir takım kodlama yöntemleri verilmiştir. Genel olarak literatürde en çok kullanılan yöntem yığını oluşturma işleminde genlerin tesadüfi olarak belirlenmesidir.

2.1.4. Uygunluk Değeri

Belli bir kromozom veya kromozomların amaç fonksiyonundaki başarı seviyesini belirleyen ve her problemde farklılık gösteren bir değerlendirme işlevidir. Uygunluk değeri uygunluk fonksiyonu ile belirlenir. Uygunluk fonksiyonu popülasyondaki her bir kromozoma bir uygunluk değeri verir. Kromozomlar ile taşınan genetik bilgi her jenerasyonda uyum değeri daha iyi kromozomların seçilmesine rehberlik etmektedir. Kromozomların uyum değeri ne kadar yüksek ise çoğalma ve yaşama şansı da o kadar yüksektir.

2.1.5. Kodlama (Dizi Gösterimi)

Genetik algoritmada ilk yapılacak işlem probleme en uygun arama uzayını temsil eden kromozomların kodlanma işlemidir. Literatürde bulunan kodlama çeşitleri şu şekildedir:

- İkili kodlama

En yaygın olan ve ilk GA uygulamalarında kullanılan bir kodlama çeşididir. Bu kodlamada her bir kromozom 1 ve 0' lardan oluşan bitlerle ifade edilen ikili dizilerdir (Şekil 2.2).

Kromozom A	010001101001
Kromozom B	101001011111

Şekil 2.2. İkili Kodlanmış Kromozom Örneği

İkili kodlama yöntemi çok sık kullanılmasına rağmen çok boyutlu (n- boyutlu) fonksiyonların optimizasyonunda uygun bir yöntem olmayabilir. Gezgin satıcı, karesel

atama, çizelgeleme gibi optimizasyon problemlerinde ikili kodlama gösterimi oldukça zordur.

- Permütasyon (Sıralı) Kodlama

Daha çok Gezgin Satıcı Problemi (GSP) ve iş sıralama gibi sıralama gerektiren permütasyon problemlerinde kullanılmaktadır. Şekil 2.3’ de gösterildiği gibi her bir kromozom bir numarayla ifade edilmektedir.

Kromozom A	837215064
Kromozom B	056174328

Şekil 2.3. Permütasyon (sıralı) Kodlanmış Kromozom Örneği

Bu kodlama biçimiyle tekrarlanan alel genler engellenmiş olup, uyum değerini hesaplama işleminde kolaylık sağlamaktadır (Taşkın ve Emel, 2009:34).

- Değer Kodlama

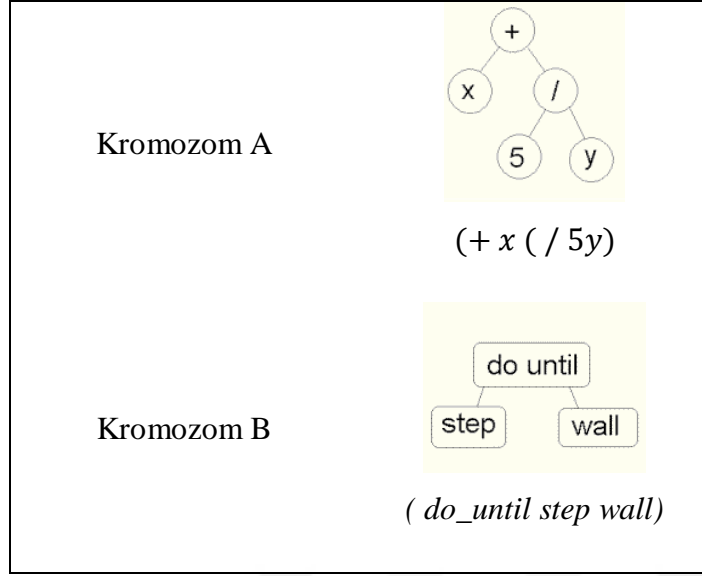
Daha çok yapay sinir ağlarında ağırlık katsayılarının bulunmasında kullanılan değer kodlaması, gerçel gibi karmaşık sayıların yer aldığı problemlerde kullanılan bir kodlama türüdür. Bu tarz problemlerde ikili kodlama kullanımı çok zordur. Şekil 2.4’ de değer kodlama örneği verilmiştir. Değer kodlaması probleme özgü olup gerçel sayılar, karakter veya kompleks nesnelere olabilir.

Kromozom A	3.274 6.121 2.456 0.299 2.115
Kromozom B	ABCJFKHDERJFDLFFEGHK
Kromozom C	(geri), (sağ), (ileri), (geri), (sol)

Şekil 2.4. Değer Kodlanmış Kromozom Örneği (Kaynak: Nabiyev, 2010:590)

- Ağaç Kodlama

Ağaç kodlama yöntemi, programların evrime tabi tutulduğu kromozomların nesnelere ve birbirleri arasındaki ilişkiler şeklinde ifade edildiği bir ağaç şeklinde kodlama türüdür. Yaygın olmayan bu kodlama türü aşağıdaki Şekil 2.5’ de gösterilmiştir.



Şekil 2.5. Ağaç Kodlanmış Kromozom Örneği (Kaynak: Taşkın ve Emel, 2009:36).

2.1.6. Genetik Operatörler

Genetik algortmada sonraki aşamada, gelecek nesillerin kalitesinin ve performansının artırılması yani iyi olanın hayatta kalması ilkesi gereği birtakım genetik operatörler uygulanır. Problemin çeşidine göre uygulanması gereken genetik operatörler; Çoğalma, Çaprazlama ve Mutasyon operatörleridir.

2.1.6.1. Seçim (Çoğalma) Mekanizmaları

Optimum çözümlere yaklaşmayı sağlamak için, en iyi yavruların yeni popülasyonda yer alması gerekmektedir. Seçim mekanizmaları, yeni nesilin oluşturulması için mevcut popülasyondan uygunluk değeri yüksek olan kromozomların bir sonraki nesle aktarılmasını sağlarlar (Kramer, 2017:16). Doğal yaşamdaki en iyinin yaşamasına benzer bir durum olan seçim işleminin amacı, ortalama uygunluk değeri üzerindeki dizilere çoğalma şansı sağlamaktır ve bu da hesaplanan uygunluk değerine bağlıdır (Jang ve v.d., 1997:1484). Bu seçim türlerinden bazıları ise şöyledir;

- Rulet Tekerleği (Çemberi) Seçimi

En yaygın ve en basit seçim mekanizmasıdır. Seçim işlemi bireylerin uygunluk değeriyle orantılı şekilde yapılmaktadır. Uygunluk değerinin yüksek olanın seçilme şansı daha fazla olacaktır. Rulet tekerleğinde çözümler kesinlikle pozitif değer almalıdır. Rulet tekerleği seçiminde bütün bireylerin uygunluk değerleri hesaplanarak

toplanır, sonra uygunluk deęerleri sırasıyla bu toplama bölünerek bireylerin [0,1] aralıęındaki seęilme ihtimalleri belirlenir.

- Rank Temelli Seęim

Popülasyondaki bireyler uygunluk deęerlerine göre iyiden kötüye doęru sıralanırlar. En kötüden en iyiye doęru bütün bireylere 1' den N' ye kadar deęerler atanır. Uygulanan bu atama işlemleri doęrusal olup iyi olan bireyler iki kopya edilirken kötü olan bireyler kopya edilememektedir.

Rulet tekerleęinde uygunluk deęerleri arasındaki fark fazla olması durumunda hep aynı çözümlerin seęilmesine neden olur ve çeşitlilięi azaltır. Fakat rank temelli seęimde GA 'nın bazı durumlarda yavaş çalışmasına sebep olsa da genellikle çeşitlilięi sağladığı için daha iyi bir arama uzayı elde etmemizi sağlar (Çevre, 2008:34).

- Turnuva Seęimi

Verimli bir seęim şekli olan turnuva seęimi, rastgele popülasyondan iki tane birey seęilip iki defa turnuvaya sokulması şeklinde yapılır ve bu iki bireyden en iyi uygunluk deęerine sahip olan birey turnuvayı kazanır. İki kez bu turnuvayı kazanan birey popülasyonun en iyi bireyi olduğundan seęilme şansı artar, böylece en iyinin hayatta kalma ilkesi gereęi bir sonraki kuşaęa aktarılma işlemleri sağlanmış olur (Chambers, 1999:79).

- Sabit Durum Seęimi

Sınırlı sayıdaki bireylerin yer deęiştirme işlemleridir. Seęim yöntemi ve yer deęiştirme yönteminin birlikte kullanılması ile bir birey birden fazla nesilde hayatta kalabilmektedir. Elimizdeki popülasyondan seęilen bireylerden genetik operatörler uygulanarak yeni nesiller üretilir ve yeniden popülasyona ilave edilir. Ancak popülasyon büyüklüğünün artmaması için popülasyondaki yavru bireyler mevcut popülasyondaki uygunluk deęeri düşük olan bireyler ile yer deęiştirirler (Taşkın ve Emel, 2009:47).

2.1.6.2. Çaprazlama

Çaprazlama işlemi, ebeveyn bireylere uygulanarak kaliteli ve performansı yüksek yeni nesiller elde etmek için uygulanmaktadır. Çoğalma işleminden sonra meydana gelen popülasyondan rastgele iki birey seçilir ve rastgele çaprazlama noktası belirlenerek karşılıklı değişim işlemi uygulanır. Burada amaç çözüm uzayını aramak ve ebeveyn bireylerde gizlenmiş bilgileri korumaktır. Gen çeşitliliğinin sağlanması açısından çaprazlama olasılığı için kesin bir sayı bulunmamakla birlikte genellikle 0.5 – 1.0 aralığında seçilmesi tavsiye edilmektedir (Michalewicz, 1996:59). Bireylerin çaprazlamaya tutulma olasılığı önemli olduğu gibi diğer önemli bir konuda ne tür bir çaprazlamaya tabi tutulacağıdır. Aşağıda bazı temel çaprazlama çeşitleri verilmiştir.

- Tek Noktalı Çaprazlama

En temel çaprazlama çeşididir. Aynı gen uzunluğuna sahip iki tane bireyin rastgele belirlenen çaprazlama noktasından sonraki kısımların yer değiştirmesidir. Böylece yeni iki birey elde edilir Şekil (2.6).

Eski birey 1	01001 101	Yeni birey1	0100 1110
Eski birey 2	00110 110	Yeni birey 2	00110 101

Şekil 2.6. Tek Noktalı Çaprazlama

- İki Noktalı Çaprazlama

Yöntem olarak tek noktalı çaprazlamaya benzemektedir, fakat tek fark isminden de anlaşıldığı gibi iki tane rastgele çaprazlama noktasının belirlenmesidir. Belirlenen bu iki çaprazlama noktaları arasında kalan genler yer değiştirir ve yeni iki birey elde edilir Şekil (2.7).

Eski birey 1	01 001 101	Yeni birey1	01 110 101
Eski birey 2	00 110 110	Yeni birey 2	00 001 110

Şekil 2.7. İki Noktalı Çaprazlama

- PMX Çaprazlama (Partially Matches Crossover)

Goldberg ve Lingle tarafından 1985 yılında önerilen sıra temelli PMX çaprazlama daha çok gezgin satıcı ve araç rotalama problemlerinde kullanılmaktadır. İki adet eski bireyden sadece bir adet yeni birey elde edilmektedir. Eski bireylerden rastgele seçilen alt birey, oluşacak yeni bireyin aynı konumuna kopyalanır. Açıkta kalan diğer kısımlar kullanılmamış genlerle doldurulur. Bu yöntem Şekil 2.8’ de verilmiştir (Taşkın ve Emel, 2009:52).

Eski birey 1	12 463 758	Yeni birey 5 1 4 6 3 7 8 2
Eski birey 2	54 172 683	
Eski birey 2	5* 463 *8*	

Şekil 2.8. PMX Çaprazlama

- OX Çaprazlama (Order Crossover)

Sıra temelli bir diğer çaprazlama yöntemidir. Oliver ve arkadaşları tarafından 1987’de geliştirilen bu yöntem, iki eski bireyden birinin sırası korunarak diğer eski bireyden bir alt birey seçilip iki yeni birey elde edilmesi esasına dayanır ve genlerin tekrarını önler (Banzhaf, 1999:141-142). Örneğin iki tane eski birey ve seçilen alt dizileri

Eski birey 1 (123 | 456 | 789)

Alt birey 1 (***) | 456 | (***)

Eski birey 2 (135 | 682 | 479)

Alt birey 2 (***) | 682 | (***)

şeklinde ele alalım. Birinci eski bireyin ikinci kesim noktasından alınan sıralama “789123456” olup Alt birey 2’ nin “682” olan alt dizisi çıkarılınca “791345” kalır. Bu da Alt birey 2’nin ikinci kesim noktasından eklenir ve Yeni birey 2 elde edilmiş olur. Aynı işlem Alt birey 1 içinde gerçekleştirilerek

Yeni birey 1 (382 | 456 | 791)

Yeni birey 2 (345 | 682 | 791)

şeklinde iki tane yeni birey elde edilir.

- CX Çaprazlama (Cycle Crossover)

Bir başka sıra temelli çaprazlama olan CX çaprazlama Oliver tarafından önerilmiştir. Seçilen her bir eski birey çiftlerinden iki tane yeni birey meydana gelir.

- LOX Çaprazlama (Linear Order Crossover)

İş çizelgeleme problemleri için 1991 yılında Falkenauer ve Bouffouix tarafından önerilen bu yöntemde, bireyler iki noktadan bölgelere ayrılır ve uygun şekilde çaprazlama yapılır.

- Ağaç Çaprazlama

Her iki bireyden belirlenen çaprazlama noktasından sonraki kısımlar kesilir ve birbirleri aralarında değiştirilerek yeni bireyler meydana getirilmiş olur (Taşkın ve Emel, 2009:55-56).

2.1.6.3. Mutasyon (Değişim)

Genetik algoritmada yapılan çaprazlama işlemi sonrası bireyler giderek birbirlerine benzemeye başlamaktadır. Çözüm uzayını daralttığı için bu benzerlik çözümün kalitesini ve performansını da düşürmekte tatmin edici bir çözüme ulaşılamamaktadır. İşte bu durumun giderilmesi için başka bir operatöre gereksinim duyulmuştur. Bireyin kendi içindeki genler arasındaki mutasyon oranına göre rastgele yapılan değişime mutasyon denilmektedir. Mutasyon işlemiyle çözümlerde çeşitlilik sağlandığından genetik algoritmanın yerel optimuma takılmasını engeller. Mutasyon işleminde mutasyon oranının çok yüksek olması fazla rastgelelik sağlayacağından popülasyonun gelişimine hasar vereceği gibi, çok düşük olması da araştırma uzayının tamamının araştırılmasına engel olacaktır. Bu nedenle mutasyon oranına dikkat edilmelidir. En uygun mutasyon olasılığı olarak 0.01 – 0.001 aralığı tavsiye edilmektedir (Michalewicz, 1996:311).

Mutasyon da çaprazlama gibi şifrelemeye tabidir. İkili kodlanmış bireylerde genler 1 ise 0'a, 0 ise 1'e dönüşür. Aşağıdaki şekilde ikili kodlanmış bireye uygulanan mutasyon örneği verilmiştir.

	Değişime uğrayan genler
Birey	1101011100011001
Mutasyona uğramış birey	1111011100001001

Şekil 2.9. Mutasyon Operasyon Örneği

2.1.6.4. Tamir

Tamir operatörü, genetik operatörlerin uygulanmasından sonra bireyden bazı bilgilerin yok olması veya aksine fazla bilgilerin gelmesi gibi durumları ortadan kaldırmak amaçlı başlangıçtaki bireylerin bilgilerine sadık kalmak koşuluyla özel bir algoritma geliştirir. Tamir operatörü, atölye çizelgeleme, ders programı çizelgeleme gibi kısıtlı eniyileme problemlerinde kullanılması gerekli bir operatördür (Elmas, 2016: 422-424).

2.1.6.5. Elitizm (En İyinin Saklanması)

Elitizm (Seçkinlik) kavramı, Kenneth De John (1975) tarafından ilk kez ortaya atılmıştır. Yeniden üretim, çaprazlama ve mutasyon işlemlerinden sonra elimizdeki popülasyonda bulunan en iyi uygunluk değerine sahip birey yeni popülasyona aktarılmadığı zaman elimizde bulunan popülasyondaki en iyi (elit) bireyin yeni popülasyona direkt aktarılması işlemine elitizm adı verilir. 1975 yılında açığa çıkan bu kavram GA için kullanılmış ve algoritmanın performansını ciddi şekilde arttırdığı gözlemlenmiştir.

2.1.7. Durdurma Kriteri

Literatürde belli başlı durdurma kriterleri şu şekildedir (Taşkın ve Emel, 2009:63-64) ;

- Hesaplama Zamanı Kriteri

Algoritma hesaplama işlemine başlamadan önce bir hesaplama süresi ve iterasyon sayısı belirlenerek algoritma belirlenen seviyeye ulaştığında durdurulmaktadır. Önceden belirlenen iterasyon sayısı bir erken duruş olabilir, belki daha ileri iterasyonlarda iyileşmenin olabileceği nesiller ortaya çıkabilir. Bu yüzden bir takım sakıncaları olabileceği gibi bunun tam tersi durum da söz konusudur.

- Optimizasyon Hedefi Kriteri

Önceden ulaşılmaması istenen amaç fonksiyonunun değeri belirlenmesi işlemidir. Genetik algoritmadaki uygunluk değeri bu değere geldiği zaman durdurulur.

- Minimum İyileşme Kriteri

Nesiller arasındaki iyileşmenin giderek azaldığı durumlarda iterasyonun durdurulması işlemidir. Genetik algorithmada iyileşme hızının yavaşlayıp sifira yaklaşması durumunda daha iyi bir sonuç beklenmemesi gerektiği sinyali verir ve çözüm için sarf edilen zaman ile kalite dengesi sağlanarak algoritma durdurulur.

2.1.8. Genetik Algoritmanın Uygulama Alanları

GA, karesel atama, gezgin satıcı, sırt çantası, yerleşim, atölye çizelgeleme, görüntü işleme, uzman sistemlerin bilgi tabanını oluşturma ve sınav/ders programının hazırlanmasında başarılı bir şekilde uygulanmıştır. Geçmişte bu çok sayıda yapılan uygulamalar uygun modelleme, kolay uygulama ve verimli problem çözümlerinde GA'nın gücünü göstermişlerdir. Günümüzde ise üretim planlama, finans ve pazarlama, elektrik, tıp, ekoloji v.s. gibi çok farklı alanlarda da başarılı uygulamalara rastlamak mümkündür. Sezgisel bir arama yöntemi olan GA' da kesin çözüm elde edilemeyebilir, fakat çözüm zamanı problemin büyüklüğü ile üstel olarak artan problemlerde kesin çözüme çok yakın sonuçlar elde edilebilir. Bu sebeple GA, fonksiyon eniyilemesi, birleşim en iyileme, çizelgeleme, iş atölyesi çizelgeleme, montaj hattı dengeleme, yerleşim tasarımı, bilgisayar çipleri tasarımı, mekanik öğrenme, hücresele üretim gibi alanlarda da başarılı bir şekilde uygulanabilmektedir (Elmas, 2016:404 - 407).

Genetik Algoritmaların son uygulamaları arasında biyomedikal alan, sanat, mimarlık, müzik, oyunlar, enerji alanı, mühendislik ve makine öğrenimi uygulamaları yer almaktadır (Kramer, 2017:73).

2.2.Yapay Arı Kolonisi Algoritması (YAKA)

2.2.1. Doğadaki Gerçek Arılar ve Görevleri

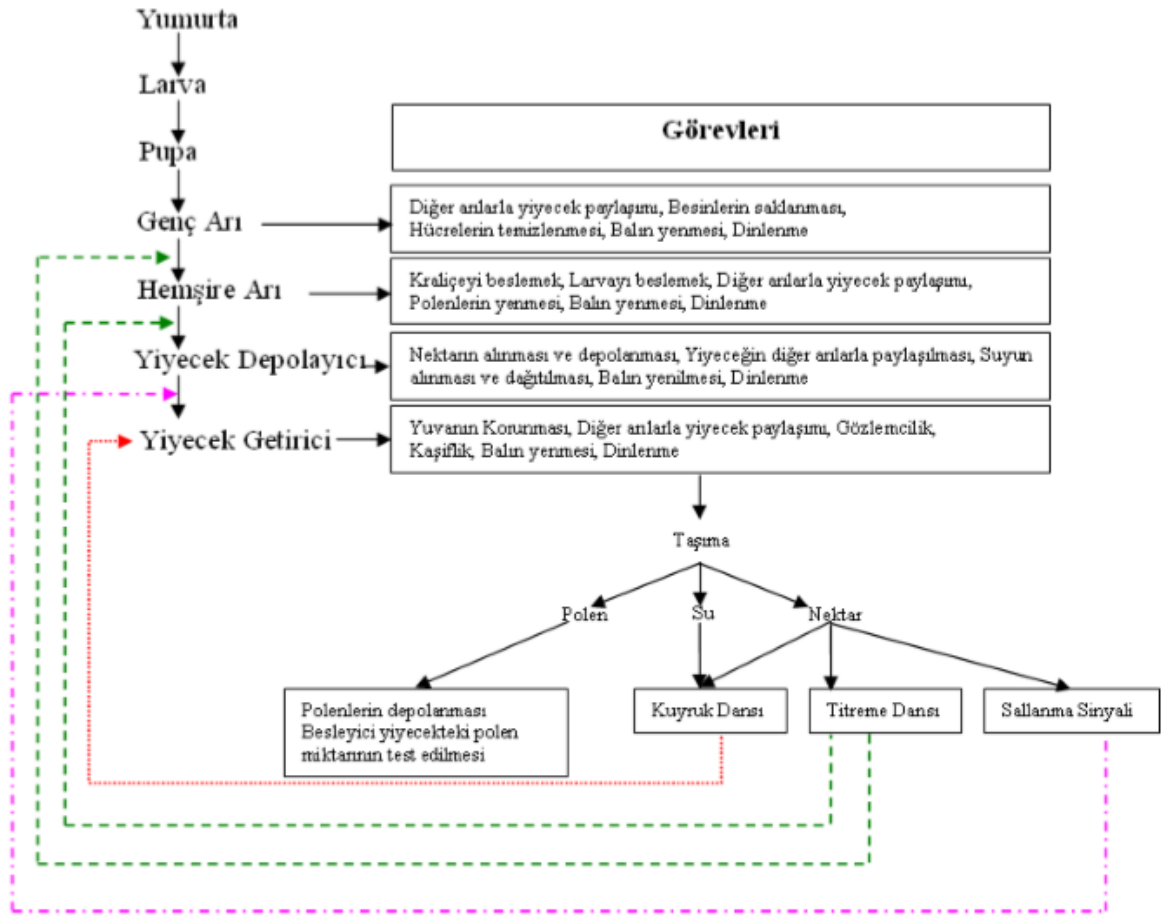
Doğadaki her bir arının farklı görevi olmasına rağmen çeşitli iletişim yolları vasıtasıyla karmaşık işlemleri tamamlamaktadırlar. Algılama davranışları stokastik yapıda olan ve kendi kendilerine organize olabilen sürü zekası sistemi olan Arı kolonisinde doğa üç tip arı vardır (Akay, 2009:28-30):

Kraliçe Arı: Kovanda sadece bir tane olan ve yumurtlama özelliğine sahip kraliçe arı ömrü boyunca sadece bir erkek arı ile çiftleşmektedir. Çiftleşme sonucu erkek arıdan aldığı spermleri vücudunda uzun süre depolamaktadır. Yılda 175000-200000, günde ise 2000 yumurta üretebilen kraliçe arı, kovanın kalabalıklaşma durumuna göre yumurtlamayı ayarlamaktadır. Kovandaki besin miktarı azaldığında yumurtlamaya başlamaktadır. Meydana gelen dişi arılardan biri kraliçe arı seçilene kadar bu durum devam etmektedir.

Erkek Arılar: Kolonideki görevi kraliçe arı ile çiftleşmek olan erkek arılar çiftleşmenin hemen ardından ölürlere ve toplamda altı aydan fazla yaşamamaktadırlar.

İşçi Arılar: Kolonide en fazla sayıya sahip arı çeşidi olan işçi arılar dişi olup üreme özelliğine sahip değildir ve larvalar ile kovandaki diğer arıları özel ürettikleri salgıları ile beslemektedirler. Aralarında görev dağılımı yaparak besin toplamak, toplanan bu besinleri saklamak ve kovanın temizliği gibi temel görevleri yerine getirirler.

İşçi arıların etrafını çevreleyerek besleyip temizlediği kraliçe arı, kovandan uzaklaşarak çiftleşme uçuşunu (mating-flight) gerçekleştirir. Erkek arı ile gerçekleşen çiftleşme sonucu oluşan spermler vücudunda bulunan spermetika organında depolanır. Larva evresinde hemşire arıların verdiği sütün zamanına göre yavruların dişi arı veya kraliçe arı olacakları belirlenmektedir. Yumurtadan çıkan genç arılar beslenirler. Yetişkin arılar kovanın dışında dururken hemşire ve genç arılar beslenme alanındadırlar. Yetişkin bir arı oluncaya kadar her arının besleme, depolama, besin arama, besinlerin elde edilmesi ve dağıtılması gibi görevleri vardır (Tokmak, 2011:10). Şekil 2.10'da arıların görev dağılımlarına ait şema verilmektedir.



Şekil 2.10. Arıların Görev Dağılım Şeması (Kaynak: Akay, 2009:32)

Besin, hayatın sürekliliği için bütün canlılarda önemli olduğu gibi zengin nektarlı besin kaynaklarını arama (foraging) arı kolonisi için oldukça önemlidir. Kovandan ayrılan arı besin arama sürecine öncelikle rastgele başlar. Besin arayan arılar bulduğu en uygun çiçeğin nektarını yaklaşık 30-120 dakika boyunca karnın bölgesinde depolar ve depolama işlemi bittikten sonra kovana dönerek, boş peteklere doldururlar. Daha sonra doldurdukları peteklerin üzerini bal mumu ile kapatırlar.

Arıların besin bulma davranışları, Treshko'nun 2000 yılında reaktif difüzyon denklemlerine dayanarak önerdiği kolektif zekânın meydana gelmesini sağlayan minimum besin arama modelindeki üç temel bileşene dayanmaktadır. Bunlar şöyledir (Karaboğa, 2014:201-205);

Besin Kaynakları: Kovana nektar, polen veya bal getirebilmek için gittikleri kaynaklardır. Bu besin kaynağının değeri için birden çok kriteri (besin kaynağının türü,

kovana yakınlığı, nektarın elde dilmesindeki kolaylığı ve nektar yoğunluğu v.s.) değerlendirebildikleri gibi yalnızca tek bir kriter de (nektarın zenginliği) ele alınabilmektedir.

Görevi Belirli İşçi Arılar: Daha önceden gidilmiş besin kaynağına ait nektarın kovana getirilmesinden sorumlu arılardır. Gidilmiş olan besinin yoğunluğu, kalitesi ve konumu ile ilgili bilgileri kovanda bekleyen diğer arılara aktarmaktadırlar. Görevi belirlenmiş işçi arılara görevli arılar denilmektedir.

Görevi Belirli Olmayan İşçi Arılar: Bunlar kâşif ve gözcü arı olmak üzere iki çeşittir. Kolonideki tüm arılara oranı %5 - %10 arasında olan kâşif arılar rastgele besin kaynağı aramaktadırlar. Kovanda bekleyen gözcü arılar ise görevli arılardan aldığı bilgiye dayanarak yeni besin kaynağına yönelen arılardır.

Arılar aşağıda ki temel durumlara göre kendi aralarında organize olmaktadır:

Pozitif Geribesleme: Besin kaynağındaki nektar miktarı arttıkça bu besin kaynağına yönelen gözcü arı sayısı da artmaktadır.

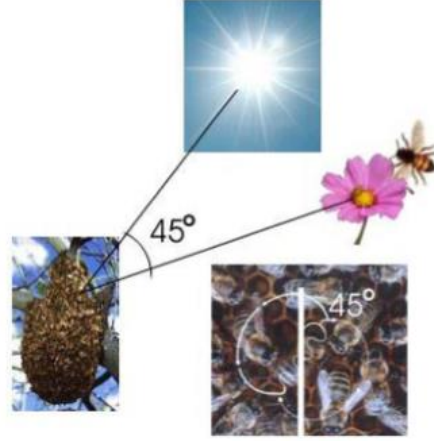
Negatif Geribesleme: Besin kaynağının tüketilmesi halinde o kaynak bırakılmaktadır.

Salınımlar: Besin kaynağı bulabilmek için kâşif arılar rastgele besin aramaya başlamaktadırlar.

Çoklu Etkileşim: Besin kaynağı hakkındaki bilgi alışverişini arılar dans alanı denilen yerde yapmaktadırlar.

Dans alanında yapılan bilgi alışverişi sonucunda nektarın tatlılığı dans eşik değerini belirlemektedir. Yani zengin besin kaynağı ile ilgili paylaşılan bilginin fazlaşması durumunda o besin kaynağının tercih edilme olasılığı da artmaktadır. Besinin kalitesi, hava durumu ve besin kaynağının kovana uzaklığına göre çeşitli dans şekilleri vardır: dairesel dans (round dance), kuyruk dansı (waggle dance) ve titreme dansıdır (tremble dance). Besinin kovana uzaklığı en fazla 50-100 metre ise daire dansı ile 100 metre ile 10 kilometre arasındaki alanda bulunan besin kaynağı hakkında ki bilgi aktarımı kuyruk dansı ile sağlanmaktadır. Besin kaynağının zengin olduğu ve kovana

işlenebilecek nektarın çok fazla olmasından dolayı nektarın işleme kısmına geçmek istediklerini belirten durumlarda ise arılar titreme dansı yapmaktadırlar.



Şekil 2.11. Arılarda Dans (Karaboğa, 2014:204)

2.2.2. Yapay Arı Koloni Algoritması Tanımı

Doğada bulunan hayvanların besin arayışları sırasındaki zeki davranışları ve kendi kendilerine organize olma durumları özellikle optimizasyon teknikleri alanında çalışan araştırmacıların ilgi odakları haline gelmiştir. Karaboğa 2005 yılında gerçek arıların besin arayışlarından yola çıkarak Yapay Arı Kolonisi Algoritmasını (YAKA) geliştirmiştir. Ele aldığı modelde iki tane varsayımı kabul etmektedir:

- Toplam besin sayısının görevli arı sayısına eşit olması, yani her bir besin kaynağından nektarın sadece bir görevli arının alması durumu
- Görevli arı sayısının gözcü arı sayısına eşit olma durumudur.

YAKA' da koloninin yarısını görevli arı, diğer yarısını ise gözcü arı oluşturmaktadır. Görevli arıların görevi bitmesi ile meydana kâşif arılar çıkarak rastgele besin kaynağı aramaktadırlar. YAKA optimizasyon algoritması çok fazla nektara sahip besin kaynağını bulmaya çalışıp çözüm uzayındaki problemin minimumunu ya da maksimumunu elde eden noktayı bulmayı hedeflemektedir (Kartal, 2015:66).

2.2.2.1. Başlangıç Besin Kaynağı Bölgelerinin Bulunması

i besin kaynağı, j parametre sayısı, x_j^{\min} ve x_j^{\max} sırasıyla j . parametrenin alt sınırı ve üst sınırı olmak üzere Eşitlik 2.1 ile her bir parametrenin alt ve üst sınırları

arasında rastgele değer üretmek algoritma arama uzayında ki çözümlere karşılık gelen rastgele besin kaynağı bölgeleri sağlamaktadır (Karaboğa ve Akay, 2009:114).

$$x_{ij} = x_j^{\min} + (x_j^{\max} - x_j^{\min}) * rand(0,1) \quad (2.1)$$

2.2.2.2. Görevli Arıların Besin Kaynağı Bölgelerine Yönlendirilmesi

Görevli arı sayısının besin kaynağı sayısına eşit kabul edilmesi ile yeni besin kaynağının daha iyi olması durumunda işçi arılar yeni besin kaynağını kaydederler. Eski besin kaynağı komşuluğunda yeni besin kaynağının belirlenmesi Eşitlik 2.2' de verilmiştir (Akay, 2009:59-60):

$$v_{ij} = x_{ij} + \varphi_{ij}(x_{ij} - x_{kj}) \quad (2.2)$$

Her bir x_i çözümü (besin kaynağı) için rastgele belirlenmiş olan j parametresinin değiştirilmesi ile x_i komşuluğunda v_i çözümü elde edilmektedir. Rastgele belirlenen x_k komşuluğunda çözüm ile mevcut çözümün j . parametresinin farkları alınıp $[-1,1]$ arasında rastgele değer alan φ_{ij} ağırlık faktörünün sayısı var olan kaynağın j . parametresine eklenerek komşuluk elde edilmektedir. Elde edilen v_{ij} ' nin belirlenen sınırları aşması halinde Eşitlik 2.3 ile çözüm uzayı belirlenmektedir.

$$v_{ij} = \begin{cases} x_j^{\min}, & v_{ij} < x_j^{\min} \\ v_{ij}, & x_j^{\min} \leq v_{ij} \leq x_j^{\max} \\ x_j^{\max}, & v_{ij} > x_j^{\max} \end{cases} \quad (2.3)$$

Daha sonra yeni besin kaynağını ifade eden v_i 'nin kalitesini belirlemek amacı ile uygunluk değeri Eşitlik 2.4 ile hesaplanmaktadır.

$$fit_i = \begin{cases} 1/(1 + f_i) & , f_i \geq 0 \\ 1 + |(f_i)| & , f_i < 0 \end{cases} \quad (2.4)$$

Burada f_i elde edilen komşu çözüm, yani yeni besin kaynağı v_i ' nin maliyet (uygunluk) değeridir.

2.2.2.3. Gözcü Arıların Besin Kaynağı Seçiminde Kullanacakları Olasılık Değerlerinin Hesaplanması (Dans Benzetimi)

Kovana dönen görevli arılar gözcü arılara besin kaynağı hakkında bilgi vermektedirler. Gözcü arılar daha sonra besin kaynaklarını nektar miktarları ile orantılı bir olasılıkla tercih ederler. YAKA' da bu tercih işlemi rulet tekerleği ile yapılmaktadır. Rulet tekerleği dışında sıra temelli, stokastik örnekleme ya da turnuva seçimi gibi yöntemlerde kullanılabilir. Fakat rulet çarkı seçimi nedeniyle YAKA' da bazı karmaşık modellerde yavaş veya erken yakınsama gibi durumlar oluşabilmektedir. Bu nedenle YAKA' nın yakınsama yeteneğini geliştirmek için Boltzmann seçim mekanizması kullanılabilir (Du ve Swamy, 2016:207). Besin kaynakları arasında bir karşılaştırma yapan gözcü arı uygunluk değeri arttıkça yani besin kaynağının nektarı arttıkça o kaynağı tercih edecektir. Bu olasılık değerini hesaplayan Eşitlik 2.5' de verilmiştir (Karaboga ve Basturk, 2008:690):

$$\mu_i = \frac{fit_i}{\sum fit_i} \quad (2.5)$$

Bu denklem, gözcü arıların yönlendiği besin kaynağının seçilme olasılıkları ile hesaplanan uygunluk değerinin toplam uygunluk değerine oranlanması ile elde edilmektedir.

2.2.2.4. Gözcü Arıların Besin Kaynağı Bölgesini Seçmesi

Besin kaynaklarının seçimi için olasılık değerlerinin hesaplanmasından sonra her bir besin kaynağı için [0,1] arasında rastgele sayı oluşturulur. μ_i olasılık değeri oluşturulan rastgele sayıdan büyük ise gözcü arılar yeni bir besin kaynağı için çözüm oluşturur. Var olan çözümle yeni çözüm kıyaslanır ve açgözlü seçim işlemi gerçekleştirilir. Yeni çözümün uygunluk değeri daha iyi ise eski çözüm hafızadan atılır, bunun tam aksi olması durumunda eski çözüm korunur ve geliştirememeye sayacı artırılır.

2.2.2.5. Biten Besin Kaynağını Bırakma ve Kâşif Arının Meydana Gelmesi

Görevli ve gözcü arıların besin kaynağı arama döngülerini tamandıktan sonra kontrol edilen geliştirememeye sayacına göre hangi besin kaynağından fayda elde edilip edilemediği, besin kaynağının nektarının bitip bitmediği hakkında hükme varılır. Besin kaynağının bitmesi veya nektarın azalması durumunda artık yeni besin

kaynaklarının rastgele aranması için kâşif arılar ortaya çıkmaktadır. YAKA’ da besin kaynağının bitip bitmediğine karar vermek için “limit” ismi verilen eşik değeri kullanılmaktadır (Karaboğa, 2014:210).

2.2.3. Yapay Arı Kolonisi Algoritmasının Özellikleri ve Adımları

Diğer metasezgisel yöntemlere göre daha iyi çalışan sürü zekâsı temelli YAKA algoritması az sayıda parametre içermesinden dolayı sade, esnek ve basittir. Algoritma gerçek arıların besin arama davranışlarına çok benzer şekildedir. Nümerik (sürekli) ve ayrık problemler için de kullanılabilen bu algoritma, kâşif arılar aracılığıyla bölgesel besin kaynağı arama özelliğine sahiptir. YAKA algoritmasına ait adımlar aşağıda verilmiştir (Akay, 2009:58):

Adım 1:..Bağlangıç besin kaynağı alanlarının üretilmesi

Tekrarla

Adım 2: İşçi arıların besin kaynağı alanlarına gönderilmesi

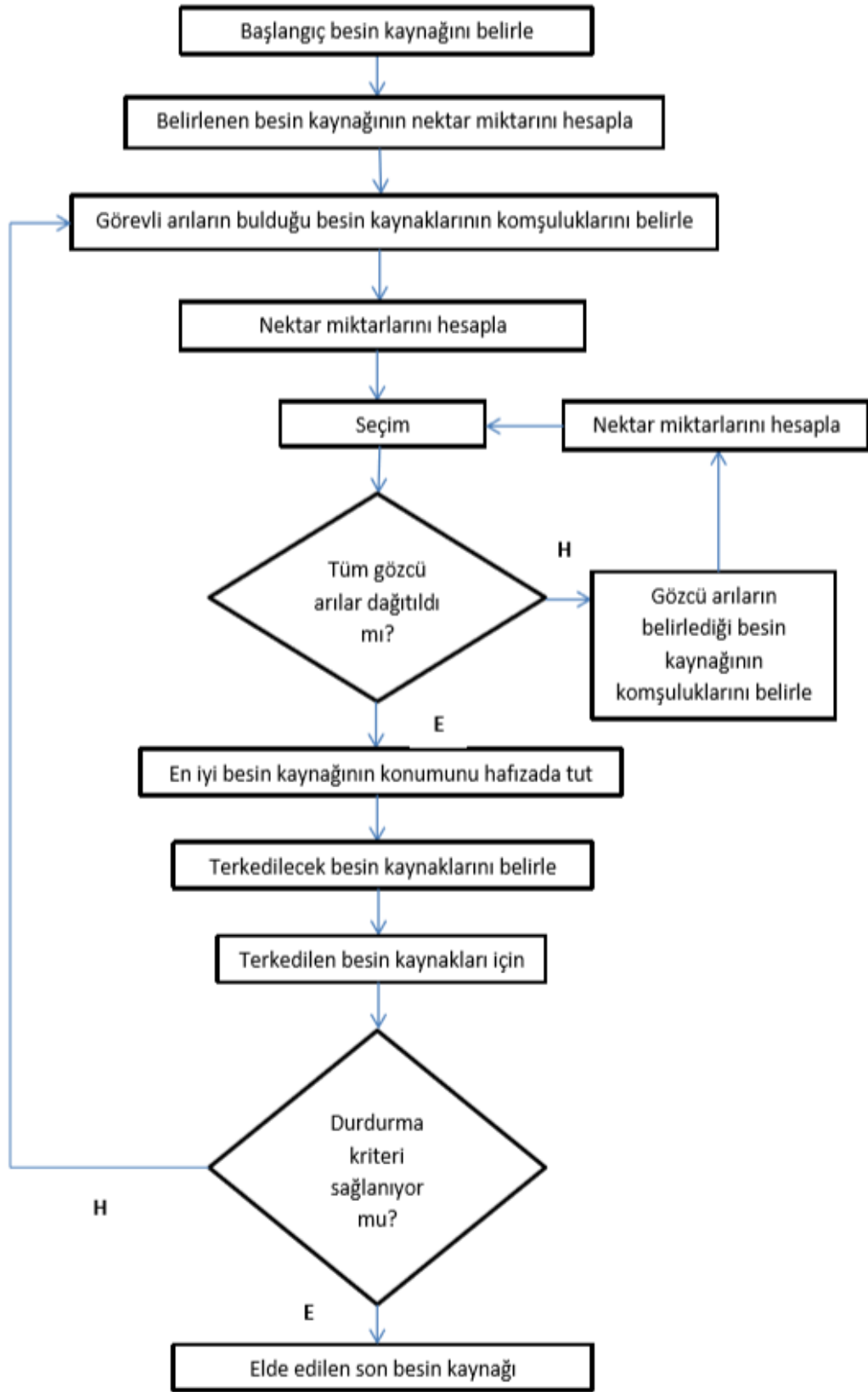
Adım 3: Görevli arılardan bilgi doğrultusunda seleksiyon işleminde kullanılacak olasılık değerlerinin hesaplanması

Adım 4: Adım 3’de hesaplanan olasılık değerlerine göre gözcü arıların besin kaynağı bölgesini seçmeleri

Adım 5: Terk edilecek besin kaynaklarının bırakılması ve kâşif arı üretimi

Olana kadar (Çevrim sayısı = Maksimum çevrim sayısı)

Şekil 2.12’ de YAKA için akış şeması verilmiştir.



Şekil 2.12. YAKA Algoritmasının Akış Şeması (Kaynak: Akay, 2009:63)

2.2.4. Çinli Postacı Probleminin Eşleşme Adımı için YAKA' da Yapılan Modifiye İşlemi

Çinli postacı probleminin çözümü için önerilen eşleşme adımı, bu problemin çözümünün en önemli parçasını oluşturmaktadır. Eşleşme adımının kesin metotlar ile çözümünün zaman karmaşıklığı n tek dereceli düğüm olmak üzere $O = n!!$ olmaktadır. Bu zaman karmaşıklığı ise eşleşme problemlerinin kesin metotlar ile çözümünü günümüz bilgisayarları için bile imkânsız hale getirmektedir. Günümüzde bu problem gibi kesin olarak çözümüne ulaşamayacağımız problemler bulunmaktadır. Bu gibi problemler için metasezgisel yöntemler kullanılarak kesin çözüme en yakın sonuç elde edilmeye çalışılmaktadır. Bu tez çalışmasında da kesin çözümü oldukça güç olan eşleşme problemine metasezgisel yöntemlerden Yapay Arı Kolonisi Algoritması (YAKA) önerilmiştir.

Temel Yapay Arı Kolonisi algoritması nümerik optimizasyonlar için önerilmiştir. Ancak nümerik olmayan problemler için de başarılı olduğu çalışmalarda gösterilmiştir. Küçüksille ve Tokmak (2011) yaptıkları çalışmalarında, nümerik olmayan ders çizelgeleme problemine yapay arı kolonisi algoritmasını uygularken, algoritmanın bazı özelliklerini probleme göre yorumlamıştır. Maliyet fonksiyonu olarak, çakışan derslere verdiği ceza puanlarını veya ardışık gelen derslere verdiği ödül puanlarını kullanmıştır. Temel Yapay Arı Kolonisi algoritması bu tür değişikliklerle uyarlanabilmeye müsaittir. Bu tez çalışmasında Çinli postacı probleminin eşleşme adımı yapay arı kolonisi algoritmasını etkin bir şekilde uygulayabilmek için, algoritmanın bazı adımları nümerik olmayan bu probleme göre adapte edilmiştir.

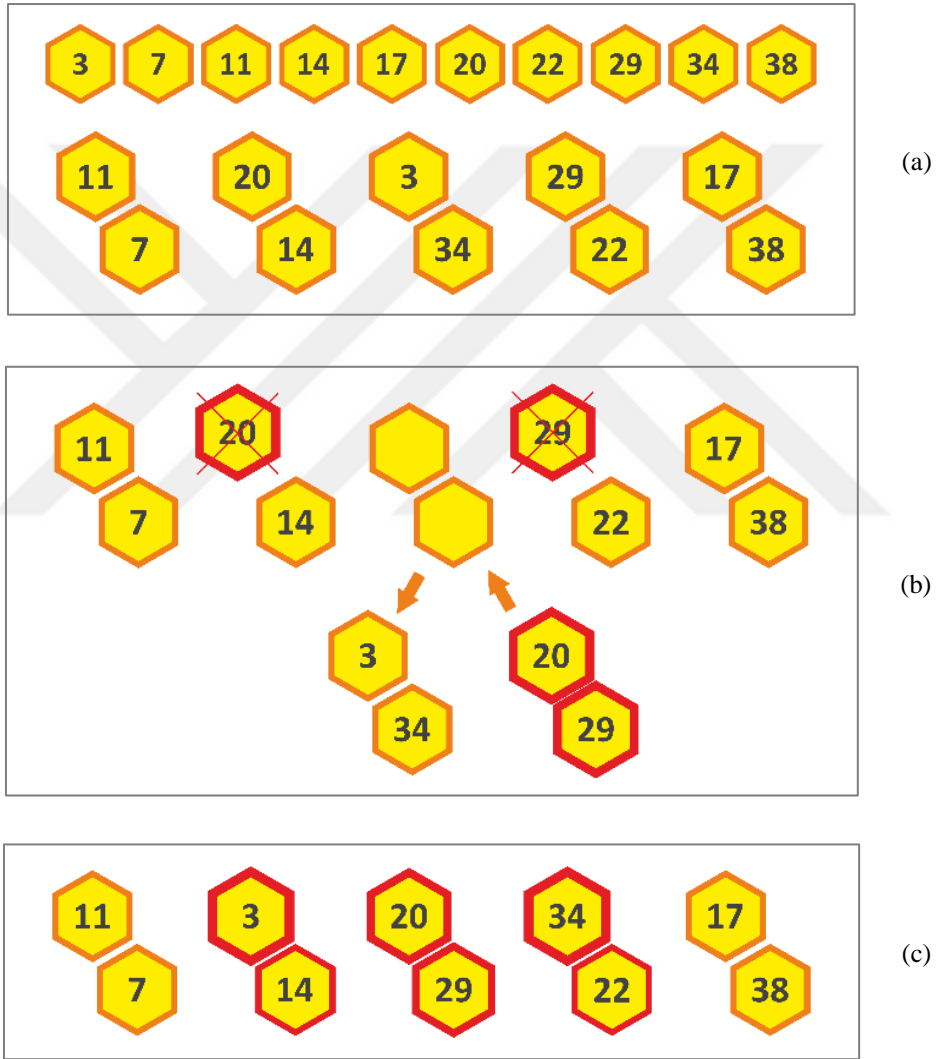
Arılarla temsil edilen arama uzayındaki her bir çözümün (besin kaynağı) uygunluğunu (nektar miktarı) ölçmek için kullanılacak maliyet fonksiyonu Eşitlik 2.6' da gösterilmiştir. Bu eşitlikte graf yapısında bulunan eşleşmiş olanlar dâhil tüm yolların toplam uzunluk değeri hesaplanmaktadır. n toplam düğüm sayısını, $d(i, j)$ ise i . ve j . düğümler arasındaki yolun uzunluğunu, $s_{i, j}$ ise bu iki düğüm arasındaki direkt yol sayısını temsil etmektedir.

$$f(x) = \sum_{i=1}^n \sum_{j=i}^n s_{i,j} * d(i,j) \quad (2.6)$$

Arama uzayındaki çözümler eşleşme problemi için nümerik değildir. Dolayısıyla bir arının arama uzayında başka bir arının çözümünün komşuluğunda yeni çözüm üretme işlemi Eşitlik (2.2)' deki gibi gerçekleştirilemeyecektir. Çünkü çözümler, ikili bağıntılar olarak ifade edilmektedir. İstenen çözümün hiçbir eşleşmemiş tek dereceli düğüm kalmayacak şekilde bir ikili bağıntılar kümesi oluşturması gerekmektedir. Ve her bir ikili bağıntıda kullanılan tek dereceli düğümler, diğer bağıntılarda bulunmamalıdır. Bir graf yapısında kesinlikle çift sayıda tek dereceli düğüm olması gerekliliği hatırlandığında böyle bir bağıntılar kümesinin varlığından bahsetmek mümkündür. V 'nin tek dereceli düğümler kümesini temsil ettiği düşünüldüğünde $V = \{3, 7, 11, 14, 17, 20, 22, 29, 34, 38\}$ örneği için oluşturulabilecek rasgele bir çözüm Şekil 2.13 (a)'da görüntülenmektedir. Yapay arı kolonisi algoritmasının en başta popülasyonunu rasgele oluşturduğu düşünüldüğünde V kümesi için oluşabilecek rasgele arı sayısı $10!$ olmaktadır. Bir kümede elemanların sırasının önemsizliği göz önünde bulundurulduğunda ise bu sayı azalacaktır.

İlk popülasyon oluşturulduğunda arıların buldukları kaynakların kalitesi Eşitlik (2.6)' ya göre hesaplanır, ve uygun olan arının etrafında yeni çözümlere bakılır. Uygun çözüm bulan arının komşuluğunda çözüm arama o arının çözüm kümesi içindeki bir bağıntının değiştirilmesi olarak ifade edilmiştir. Ancak bir arının bağıntılar kümesindeki bir elemanını rasgele bir ikili bağıntı ile değiştirmek, bu arının yapısındaki diğer bağıntıların da değişebilmesine sebep olmaktadır. Bu bağıntıların en az etkiyle değiştirilmesi, komşulukta bulunacak çözümün sağlığı açısından önemlidir. Bir bağıntı değişirken oluşabilecek üç durum vardır. Değişmesi için önerilecek bağıntı, hali hazırdaki bağıntı ile aynı olabilir, bu durumda arı yeni çözüm üretememiş kabul edilir. Çünkü arının bağıntı yapısında herhangi bir değişiklik söz konusu olmaz. İkinci durum ise değiştirilecek olan ikili bağıntı yerine önerilen yeni çiftin yalnızca bir tanesinin değişmesi gereken ikili bağıntının dışında bir bağıntıda bulunmasıdır. Bu durumda yalnızca iki bağıntının değişeceği öngörülebilir. Ancak yeni önerilen çiftin, değişecek olan bağıntıdan tamamen farklı olması durumunda 3 ikili bağıntı birden değişecektir.

Bu duruma örnek olarak Şekil 2.13 (b)'deki (3,34) bağıntısının yerine yeni gelen (20,29) bağıntısının arının çözümünün yapısında oluşturduğu değişiklik incelenebilir. Önerilen (20,29) bağıntısı ne olursa olsun önerilen yere geçmiş, ancak etkilediği diğer çiftlerde kalan boşluklara da eski bağıntı olan (3,34) ikilisi dağıtılmıştır. Bu durum Şekil 2.13 (c)'de incelenebilir. Şekil 2.13 (c)' de yapısı değişen ikili bağıntılar kalın çizgiler ile çerçeveslenerek ifade edilmiştir. Bu tasarım eşleşme adımı için önerilmiş ve başarılı sonuçlar elde edilmiştir (Fındıklı ve vd., 2018:1032).



Şekil 2.13. (a) Tek Dereceli Düğümlerden Oluşmuş Bir İşçi Arı
 (b) Rasgele Seçilmiş Bir Çiftin Yeni Çift İle Yer Değiştirmesi
 (c) Çiftler Değişiminin Diğer Çiftler Üzerindeki Etkisi

BÖLÜM 3

3. UYGULAMA

3.1. Uygulamanın Tanımlanması

Bu çalışma da Malatya ilinin Yeşilyurt ve Battalgazi ilçelerinin ilaçlama işlemi için ilaçlama araçların rotalanması ele alınmaktadır. Bunun için Malatya Büyükşehir Belediyesi Çevre Koruma ve Kontrol Dairesi Başkanlığı Atık Yönetim ve İlaçlama Birimi ile gerekli görüşmeler ve alınan izin sonucu araçların mevcut işleyişleri hakkında bilgi edinilmiştir. Kurum ile yapılan görüşme yılının 2017 yılı olması nedeniyle o dönemki (Tablo 3.1) plan ve EK- 3' deki raporlar dikkate alınmıştır.

Uygulamanın verileri coğrafi bilgi sistemi yazılımı olan QGIS 2.18.11 programına Malatya ili haritasının yüklenmesi ve gerekli katmanların oluşturulması ile elde edilmiş orijinal ve gerçek verilerdir. Burada öncelikle QGIS programı yardımı ile her iki ilçenin mahallelerine orta nokta atanmış ve mahalleler arası mesafe matrisi elde edilmiştir. Daha sonra ise her bir mahallenin içindeki (belirlenen bölgelerin) her bir cadde/sokağın başına ve sonuna ID verilerek, cadde/sokak mesafe matrisleri elde edilmiştir.

Uygulama her iki ilçe için düşünüldüğünde ve bazı büyük mahallelerin bölünme işlemi ile toplam mahalle sayısının 154 olması nedeniyle çok büyük boyutlu bir hal almıştır. Bu nedenle optimum çözümün sağlanması için çok aşamalı bir uygulama önerilmiş ve yine boyutunun çok büyük olması nedeni ile uygulamaların çözümünün de metasezgisel yöntemlere başvurulmuştur.

Uygulamanın ilk kısmında öncelikle her iki ilçe ayrı ayrı düşünülmüş olup araçlara mahalle dağıtımının ve rotalamanın yapılması sağlanmıştır. Bunun için öncelikle mahalleler yakınlık durumlarına göre kümelenip daha sonra gerekli kısıtlar dahilinde problem Zaman Pencere Araç Rotalama Problemi olup çözümünde Genetik Algoritma uygulanmıştır.

İlk kısımda belirlenen kümeleme ve rotalama sonucu ile artık mahalleler birer bölge olarak düşünülmüştür. İlaçlamayı yapacak araçların ilaçlama yapacağı bölgenin her bir cadde/sokağını ilaçlaması zorunluluğu nedeniyle problem Yönsüz Çinli Postacı Problemi olup problemin en önemli adımı olan ve özellikle büyük boyutlu graflarda

çözümü neredeyse imkânsız hal alan eşleşme adımının çözümüne Yapay Arı Kolonisi Algoritması uygulanmıştır. Bu eşleşmenin optimum olabilmesi için de Yapay Arı Kolonisi Algoritması'nın parametre değişimi kısmına modifiye işlemi yapılmıştır. Hem bu modifiye işlemi hem de Çinli Postacı Problemine Yapay Arı Kolonisi Algoritması ile çözüm yaklaşımı literatüre kazandırılmıştır. Optimum eşleşmelerin sağlanması ile bölge (mahalleler) içinde turların oluşturulmasında Euler tur elde edilmesinde kullanılan algoritmalarından biri olan Fleury algoritması kullanılmıştır.

Bu çalışmada ki uygulamanın sonuçları ile mevcut durumda ki araçların işleyiş sonuçları karşılaştırılarak Tablo 3.33'de paylaşılmıştır.

3.1.1. Malatya Atık Yönetim ve İlaçlama Şube Müdürlüğü

Malatya Büyükşehir Belediyesi Çevre Koruma ve Kontrol Dairesi Başkanlığı Atık Yönetim ve İlaçlama Birimi toplam 93 personel, 24 adet ilaçlama aracı ile Malatya'nın 13 ilçesinde toplam 716 mahalleye böcek ve haşerelerle mücadelede ilaçlama işlemi ile şehre hizmet vermektedir. İlaçlama işlemine yaz mevsiminin (Haziran ayı) gelmesi ile başlanmakta ve sonbaharın ortalarında (Eylül ayı) son bulmaktadır.

İlaçlama birimi haşerelerin üreme noktaları olan başta logarlar, bataklıklar, hayvansal gübre depoları olmak üzere haşere üreme potansiyeline sahip bütün yerleri erkenden ilaçlamakta ve haşerelerin en çok ürediği yerlerden olan on binlerce atık lastiği toplayarak, hem haşere üremesinin önüne geçmekte hem de ekonomiye kazanım sağlamaktadır.

3.1.2. İlaçlama Araçlarının Yapısı ve İşleyişi

İlaçlama araç tipi ve aracın taşıdığı ilaçlama makinası Şekil 3.1 ve Şekil 3.2 deki gibidir. İlaçlama araçları yarım araç olup arkasında İl-Star 18 ULV Jeneratörü ilaçlama makinaları taşımaktadır.



Şekil 3.1. İl – Star 18 ULV İlaçlama Makinası (Kaynak: <https://images.google.com/>, (09.12.2017))

İl – Star 18 ULV ağırlığı 175 kg, genişliği 95 cm, uzunluğu 115 cm ve yüksekliği 95 cm dir. İl – Star 18 ULV güçlü, sessiz ve çevreci Briggs Vanguard motor ile günde 250 hektar açık ve geniş alanları 10 metre esnek uzatma hortumu sayesinde kanal ve kapalı alanları ULV yöntemi ile ilaçlama yeteneğine sahiptir. Açık alanlarda araç üzerine, kapalı alanlarda çek çekler üzerine monte edilir. Tüm fonksiyonları araç şoförü tarafından kumanda edilerek kent yerleşim alanlarında, yeşil ve ormanlık dinlenme alanlarında, tarım alanlarında, depo antrepo şeklinde geniş kapalı alanlarda, seracılıkta uçan, yürüyen ve dinlenen her türlü zararlı haşere mücadelesinde kullanılabilir. İlave edilebilecek özel kanal aparatı ile kanalizasyon ve fosseptiklerde de kullanılabilir. İl – Star 18 ULV makinasının kapasitesi 67-100 litre olup ilaçların vizkozitesine bağlı olarak yaklaşık 3.5 saatte bitmektedir. Motor yakıt tankı polietilen malzemeden yakıt seviye göstergeli olup, kapasitesi 30 litredir. Benzin ile çalışmaktadır. 30 litre benzin ile bir defa doldurulan depo, ilaçlanacak bölgenin bölgenin büyüklüğüne göre 2 veya 3 gün yetmektedir. Yıkama tankı 8 litre, akü 12 V DC dir. 45-60 AH şarjlı aküye sahiptir. İl – Star 18 ULV cihazının tüm fonksiyonları uzaktan kumanda sistemi ile yönetilmektedir; Motor çalıştırma – durdurma, ULV başlatma – durdurma, sistemi geri yıkama v.s. (http://makina.il-ser.com.tr/TR/static_pages/89/il-star_18_ulv_jeneratuoru, Erişim Tarihi: 09.12.2017).

Sivrisinek erginlerine karşı araç üstü ULV (soğuk sisleme) uygulaması yapılmaktadır. Bunun için ilaçlama makinasına 60 lt suya 1.5 litre böcek ilacı verilerek 2-2.5 saat arası ilaçlama yapmaktadır. Araç ilaçlama işlemini tamamen bitiremediği

durumlarda en yakın yerden depoya tekrar su koyarak ve arabaya en başta aldığı yedek ilacı da ilave ederek ilaçlamaya devam etmektedir.



Şekil 3.2. Malatya Büyükşehir Belediyesi İlaçlama Aracı (Kaynak: <https://images.google.com/> (09.12.2017))

Uygulamamızda Malatya merkezinin bulunduğu Battalgazi ve Yeşilyurt merkez ilçeleri için ilaçlama yapan araçlar ve ilaçlama yapılan mahalleler ele alınmıştır. Atık Yönetim ve İlaçlama Birimi bu iki ilçe için toplam 12 adet ilaçlama aracını haftanın 6 günü akşam saat 19.00 - 23.00 saatleri içerisinde çalışacak şekilde tahsis etmiştir. Bu 12 araç dışında bir araç yedekte bekletilerek gelen şikâyet telefonları ile verilen adreslerde ilaçlama işlemini gerçekleştirmektedir. Araçlar ilaçlama işlemini yaparken hızları 10 - 15 km arasında olmak zorundadır.

2017 yılı Haziran – Eylül ayı için Tablo 3.1 deki gibi 12 araç 6 günde Malatya ilinin merkezine ait toplamda 143 mahalleyi ilaçlaması birim tarafından planlanmıştır. Mahallelerin günlere dağılımı Atık Yönetim ve İlaçlama Birimi tamamen rastgele belirlemiştir. Ayrıca ilaçlama araçlarının da mahalleler içerisinde cadde/sokak içerisinde gezinmeleri de rastsal olarak yapılmaktadır.

Tablo 3.1. Malatya Atık Yönetim ve İlaçlama Birimi'nin 2017 yılı için Planladığı İlaçlama Gün ve Ekip Çizelgesi (Kaynak: Malatya Büyükşehir Belediyesi Çevre Koruma ve Kontrol Dairesi Başkanlığı Atık Yönetim ve İlaçlama Birimi)

Günler Ekipler	PAZARTESİ	SALI	ÇARŞAMBA	PERŞEMBE	CUMA	CUMARTESİ	T o p l a m
	1.EKİP	Melekbaba	Kaynarca, Çarmuzu	Kiltepe, Çavuşoğlu, Salköprü, Bentbaşı, İlyas	Yeşilkaynak, Fatih (merkez), Gazi	Cumhuriyet, Atatürk, Koşu, Seyran	
2.EKİP	Fatih (organize), Duruldu, Kuşdoğan, Çayırköy	Tecde	Yavuz Selim, Cemal Gürsel	Yakınca	Yakınca	İnönü, Cevat Paşa	10
3.EKİP	Samanköy, Kendirli	Mullakasım, Gedik, Hıroğlu, Gündüzbey	Bahçebaşı, Su, Yeni (yeşilyurt)	Aşağıbağlar, Turgut Özal, Abdulgaffar, Samanlı	Özalper, Şeyhbayram	Koyunoğlu, Çukurdere, Zaviye	18
4.EKİP	Görgü, Gözene	Duranlar, Göktarla, Tohma	Tepeköy, Topraktepe	Suluköy	Sütlüce	Sütlüce, Mahmutlu	10
5.EKİP	Şişman, Yaygın	Karakavak	Bostanbaşı	Bostanbaşı	Çilesiz	Göztepe	6
6.EKİP	Yıldıztepe, Tandoğan	Battalgazi, Selçuklu	Şehit Fevzi, Beylerbaşı, Sarıcıoğlu, Hidayet	İskender, Cirikpınar, Taştepe	Nuriye, K.Mustafa P., B.Mustafa P., Aslanbey, İzzetiye, Cevherizade, Hamidiye, Ferhadiye, İstiklal, Saray, Kernek	Hasan Varol, Başharık	24

7.EKİP	Beydağı (merkez), Yamaç	Fırat, Çöşnük	Zafer	Uçbağlar, Sancaktar, Kırçuval, K.Hüseyinbey, B.Hüseyinbey, Akpınar, Halfettin, Yenihamam Dabakhane	Kavaklıbağ, İsmetiye, Niyazi, Şifa, Şıkşık	Paşaköşkü, Ataköy, Hacı Abdi	22
8.EKİP	Kuluşağı, Karatepe	Karaköy, Çolakoğlu	Fırıncılar, Çamurlu, Beydağı	Bulgurlu	Karabağlar, Toptaş	Bahçelievler, Dolamantepe	12
9.EKİP	H.Haliloğlu, Yenice, Yeniköy	H.yusuflar, Bağtepe, Göller	Karahan, Meydanbaşı	Alacakapı,	Boran, Toygara, Kadıçayırı		12
10.EKİP	KemerKöprü	Hasırcılar	H.çiftliği	H.çiftliği	Orduzu	Orduzu	4
11.EKİP	Yeni (battalgazi)	Alışar	Hatunsuyu	Hatunsuyu	Dilek	Dilek	4
12.EKİP	Bindal, Özal	Şahnahan	Şahnahan	Topsöğüt	Topsöğüt	Yaka	5

Araçlar çipli olup “Arvento” isimli araç takip sistemi ile izlenmektedir. Bu sistem ile araçlar, GPS uyduları aracılığıyla 7 gün 24 saat internet erişimi olan bilgisayarlar, akıllı telefonlar ve tabletler gibi iletişim cihazlarından takip ve kontrol edilebilmektedir. Arvento ile araçların hızları, rotaları, mola verdiği yerler, rölanti süreleri gibi çokça bilgi, haritalardan anlık veya geçmişe dönük olarak görülebildiği gibi anlık veya geçmişe dönük raporlar sayesinde de kayıt ve kontrol altına alınabilmektedir.

Araç takip sistemleri temel olarak GPS, GSM/GPRS, sayısal haritalar ve özel yazılımlardan oluşan bir sistem düzeneğinde çalışmaktadır. Araçlara takılan mobil veri cihazları, GPS uydularından aldıkları iki bilgiyi (Uydu sinyalinin gönderildiği gerçek zaman ve uydunun sinyal gönderildiği anda yörüngedeki konumu) ve bağlı sensörlerden gelen sıcaklık ve benzeri telemetri bilgilerini, GSM/GPRS şebekesi üzerinden kontrol ve iletişim merkezine gönderirler. Gönderilen bu bilgiler özel yazılımlar sayesinde derlenir ve sunucular üzerindeki veri bankasına depolanırlar. Bu sistemi kullananlar özel yazılımlar vasıtası ile internete erişimi olan bilgisayarlar, akıllı telefonlar ve tabletler ile anlık veya geçmişe dönük olarak araçları takip edebilirler, araçlardan gelen

bilgileri izleyebilirler, araçların alarm ve program kısımları üzerinde deęişiklik yapabilirler. Bu sistematik yapı ile araç takip sistemlerinin temel alıřma prensibini anlamak mümkündür (<https://www.arvento.com/tr/ arac-takip-nedir> (10.12.2017)).

Ara Takip Sistemlerinin Avantajları (<https://www.arvento.com/tr/neden-arac-takip-sistemi> (10.12.2017));

- Verimlilięi arttırıp, daha ok kazandırır.
- Aracın rölantide gereksiz alıřması, birden hızlanması, yavaşlaması, savrulması ve hızının takip edilmesi ile kötü araç kullanımını engelleyip, benzinden %30 civarı kar elde edilmesini saęlar.
- Özel yazılımlar sayesinde aracın bakım, servis ve onarımını takip ederek aracın yıpranma süresini uzatarak bakım maliyetini düşürür.
- Özel sensörler ve yazılımlar ile araçtan yakıtın izinsiz alınmasını engeller.
- Şoförlerin gün içinde hangi konumda olduklarına dair bilgi edinmek için gereksiz telefon görüşmeleri yapmayarak zamandan tasarruf ettirir.
- Aracın ne zaman işe başladığını, hangi güzergâh da gittiğini, ürün teslimini ne zaman ve hangi noktalara yaptığını ve işin ne zaman bittiğine dair bilgileri geçmişe dönük raporlama imkânı sunar.
- Ekibin iş saatleri içindeki performansını ölçmeye ve iş dışı araç kullanımını engellemeyi saęlar. Böylece harcanan zaman ve iş gücünden tasarruf sağlanarak verimliliğin üst düzeye çıkmasını saęlar.
- Alınan detaylı raporlar ile ekip ve şoför performansı istatistikleri elde edilerek gereksiz fazla mesai ödemelerinin önüne geçilir.
- Aracın takip de olması nedeniyle trafik kurallarına uyumluluęu sağlanıp, trafik kazası ve trafik cezası alma riskini azaltır.
- Kaza ve acil durumlarda araçtan alınan direkt uyarı mesajı ile en kısa sürede olaya müdahale sağlanır.
- Aracın alınması ve kaybolması halinde en kısa zamanda aracın konumunun belirlenmesini saęlar.

3.2. Uygulamanın Önemi

Malatya genelinde Haziran 2017 kayıtlarına göre 11 bin üreme noktası tespit edilmiş ve 11 bin üreme noktasında 20-25 günlük periyodlar ile vektörlerin larvalarına yönelik larvasit uygulamaları, 30-45 günlük periyodlarla da karasinek, tatarcık ve sivrisineklere karşı kalıcı (residüel) uygulamalar yapılmıştır. İlaçlama çalışmalarına tüm sezon süresince devam edilmektedir. İlaçlama çalışmasının yapılacağı mahallelerde vatandaşların kapı ve pencerelerini kapatmaları, çocuk, yaşlı ve astım hastalarının ilaçlama bölgelerinden uzak durması gerekmektedir. Haşere mücadelesinde kullanılan tüm ürünler Dünya Sağlık Örgütü'nün (WHO) onayladığı ve Sağlık Bakanlığı tarafından ruhsatlandırılan ilaçlar ve araçlar olup, halk sağlığı alanında güvenle kullanılan ürün ve malzemelerden olmak zorundadır.

Sivrisinekler ılıman iklim kuşağında olan ülkemiz için rahatça üreyip, çoğalabileceği uygun koşulları taşımaktadır. Beslenme ve üremek için kan emmeye ihtiyaç duyan sivrisinekler insanlarda sürekli kan emerler. Bu sebeple kan emdiği hastalıklı bir canlıdan aldığı kanı, diğer bir canlıyı sokması esnasında o canlının kanına bulaştırarak hastalık oluşturmada en önemli taşıyıcıdır. Ev, barınak ve ağaçların karanlık yerlerinde saklanırlar. Akşam karanlığından şafak vaktine kadar aktif olan sivrisinekler üreme alanlarından 500-1000 m' den fazla uçmayı tercih etmezler. Bu sebeple Malatya ili mülki sınırları içerisinde yer alan tüm mahallelerde yılın Mayıs ve Ekim ayları arasında 6 günlük periyodik aralıklarla ergin sivrisinek mücadelesi için Battalgazi ve Yeşilyurt ilçeleri için 12 ayrı ekip ile soğuk sisleme (ULV) uygulaması yapılmaktadır. Malatya il mülki sınırları içerisinde tespit edilen üreme noktalarında Mart ve Ekim ayları arasında, 20-30 günlük aralıklarla larva mücadelesi yapılmaktadır. Bu nedenle sivrisineklerle mücadele de harcanan paralar ülke ekonomisini de önemli ölçüde etkilemektedir.

Hem insan sağlığının ön planda olduğu ve hem de araçların optimal rotalarının bulunması ile mesafenin, maliyetin ve zamanın minimize edilmesi amacından dolayı uygulamamızın önemi artmaktadır.

Bu sebeple ilaçlama araçlarının şimdiye kadar yapılan rassal rotası ile Arvento programından yapılan incelemeler sonucunda hem araç tüm cadde/sokaklarda

gezinmemiş, hem de bazı yollardan gereksiz yere birçok defa geçerek fazladan kimyasal salınımına, satın alma gücü açısından akaryakıtın en pahalı olduğu ülkelerden biri olan Türkiye’ de fazla yakıt tüketimine neden olmaktadır. Ayrıca tüm caddelerin ilaçlanmaması nedeniyle telefonla gelen şikâyetler nedeniyle araçlar o mahalleye yönlendirilerek şikâyet edilen cadde/sokağa tekrar yönlendirilmesi ile zaman ve yakıt kaybına da neden olmaktadır.

Uygulamamızda araçların optimal rotaları elde edilerek mesafenin minimizasyonu ile maliyetten ve zamandan fayda sağlanabileceği gibi insan sağlığı için sivrisinek mücadelesi ve gereksiz kimyasal ilaç salınımının da önüne geçilmiş olacaktır.

3.3. Coğrafi Bilgi Sistemleri (CBS) ve QGIS Programı

3.3.1. Coğrafi Bilgi Sistemleri (CBS)

İlk olarak CBS’ nin tanımını 1998 yılında ifade eden Burrough’ a göre CBS belirli bir amaç için yeryüzüne ait gerçek verilerin toplanması, depolanması, analiz edilmesi, taşınması ve görüntülenme işlevlerini gerçekleştiren araçların tamamıdır. Günümüzde CBS; konumla ilişkili olsun olmasın geniş veri tabanı olan, seçme, transfer, sorgulama, analiz gibi özel fonksiyonları olan, modelleme ve analitik özellikli, farklı isteklerde kaliteli çıktı üretebilen ve her türlü kararı destekleyebilen bir bilgisayar sistemidir. Birçok farklı alanda kullanılan CBS temelde aşağıdaki üç hedefe ulaşmayı amaçlamaktadır (Tecim, 2008:51-56);

- Harita ve coğrafi bilgiler yardımı ile üretkenliği arttırmak
- Coğrafi veri tabanında yönetimi geliştirmek
- Coğrafi veriler ile daha güçlü stratejik yollar geliştirerek karar vermeyi desteklemektir.

Kimine göre araç, kimine göre bilgi sistemi olarak ifade edilen CBS; şehir planlama, bilgisayar tabanlı haritalama, ulaşım, sağlık, jeoloji, finans ve askeri alanlar gibi birçok farklı uygulama alanlarına sahip bir karar destek sistemidir (Mutlu ve Alver, 2015:550).

3.3.2. QGIS Programı

QGIS (Quantum GIS), WGIS GNU Genel Kamu Lisansı altında dağıtılan kullanıcı dostu bir Açık Kaynak Coğrafi Bilgi Sistemi (CBS) yazılımıdır. Açık Kaynak Coğrafya Vakfı (OSGeo) 'nın resmi projesi olup Linux, Unix, Mac OSX, Windows ve Android üzerinde çalışmaktadır ve çok sayıda vektör, raster ve veritabanı biçimlerini ve işlevlerini desteklemektedir. QGIS temel fonksiyonların ve eklentilerin yeteneklerinin sürekli iyileştirilmesini ve büyümesini sağlamaktadır. Görselleştirilebilir, yönetilebilir, düzenlenebilir, verileri analiz edilebilir ve yazdırılabilir haritalar oluşturabilmektedir (<https://qgis.org/tr/site/about/index.html> (20.06.2017)).

Haritaların meydana getirilmesinde, arazi çalışmalarında, işlenmiş verilerin elektronik ortama aktarılmasında, uydu fotoğraflarından elde edilemeyen tam görüntülerin yerinin tespitinde özellikle nokta elemanlarına ait verilerin elde edilmesinde, arazi çalışması öncesi bilgisayar aracılığıyla önceden oluşturulan verilerin saha çalışmasında CBS veri tabanına direkt Android işletim sistemli mobil cihazlarda kullanılabilen açık kaynaklı ve ücretsiz mobil CBS yazılımı olan “QGIS for Android” ile veri girişlerinin yapılabilmesini sağlamaktadır (Mutlu, 2015:36-37).

3.4. Verilerin Toplanması ve Düzenlenmesi

Bu çalışmada Çevre Koruma ve Kontrol Dairesi Başkanlığı Atık Yönetim ve İlaçlama Birimi'ne ait 12 adet ilaçlama aracının optimal rotaları belirlenmeye çalışılmıştır. İlaçlama araçlarının minimum rotalarının belirlenmesi ile ilaçlama sistemini oluşturan maliyetlerin azaltılması amaçlanmaktadır.

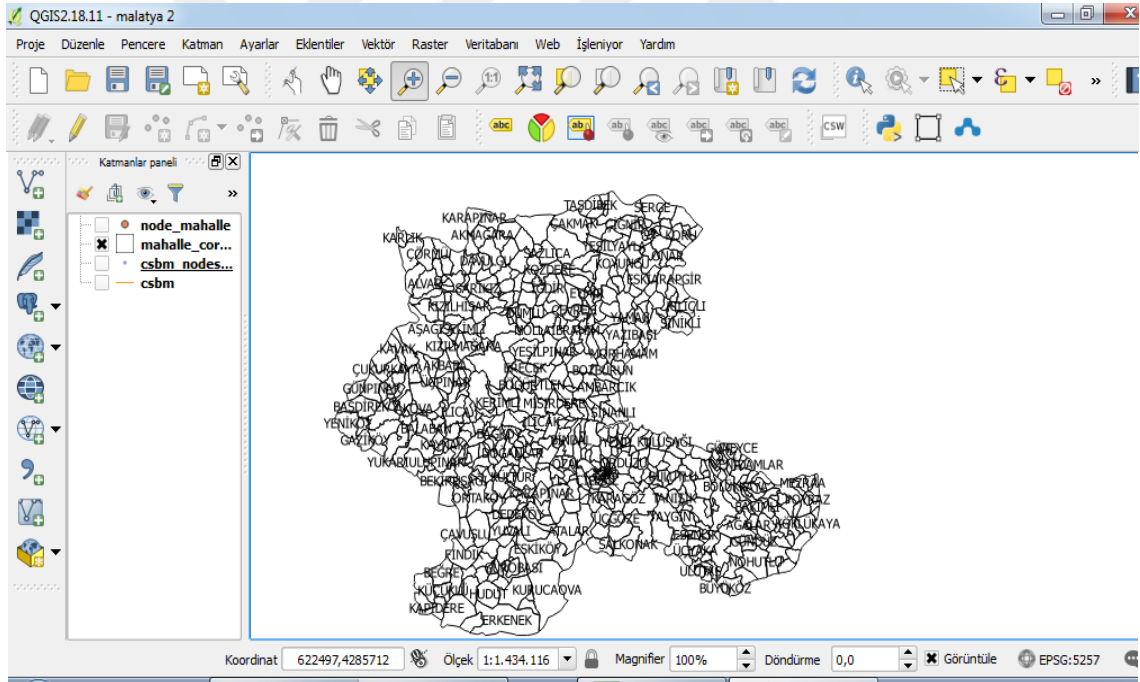
Verilerin toplanmasında öncelikle uygulamamızın yapılacağı Atık Yönetim ve İlaçlama Birimi ile gerekli görüşmeler yapılmıştır. Yapılan görüşmelerde ilaçlama işleminin hangi saatler aralığında, ne kadar süre içinde, aracın ne kadar hızda olması gerektiği, araçların depoları, kullanılan ilaçlama makinası ve ilacın özellikleri, ne kadar ilaç ve su gerektiğine kadar v.s. konularında gerekli bilgiler alınmıştır.

Daha sonra uygulama için gerekli olan Malatya ilinin Battalgazi ve Yeşilyurt ilçelerine ait tüm mahallelerin alan bilgileri ve her bir mahallenin sahip olduğu toplam cadde/sokak uzunlukları Malatya Büyükşehir Belediyesi Coğrafi Bilgi Sistemleri (CBS)

birimi ile görüşülerek elde edilmiştir. Mahalleler arası uzunluk matrisi ve cadde/sokak uzunluklar matrisi verileri ise Coğrafi Bilgi Sistemi yazılımı olan QGIS 2.18.11 programından gerekli işlemlerin yapılması ile bizzat bu çalışma için elde edilen gerçek verilerdir.

3.4.1. Mahalleler Arası Uzaklık Matrisinin Oluşturulması

Mahalleler arası uzaklık matrisi elde edilirken öncelikle QGIS programına Malatya il haritası Şekil 3.3’ de görüldüğü gibi yüklenmiştir. Daha sonra her bir mahallenin koordinatları belirlenmiştir. Bu belirlenen koordinatlar yardımı ile her mahalleye orta nokta atanmıştır. Elde edilen bu orta noktalar sayesinde QGIS programında mahalleler arası uzaklıklar oluşturulmuştur. Programdan elde edilen mahalleler arası mesafeler Matlab programında yazılan kod yardımı ile uzaklık matrisine dönüştürülmüştür.



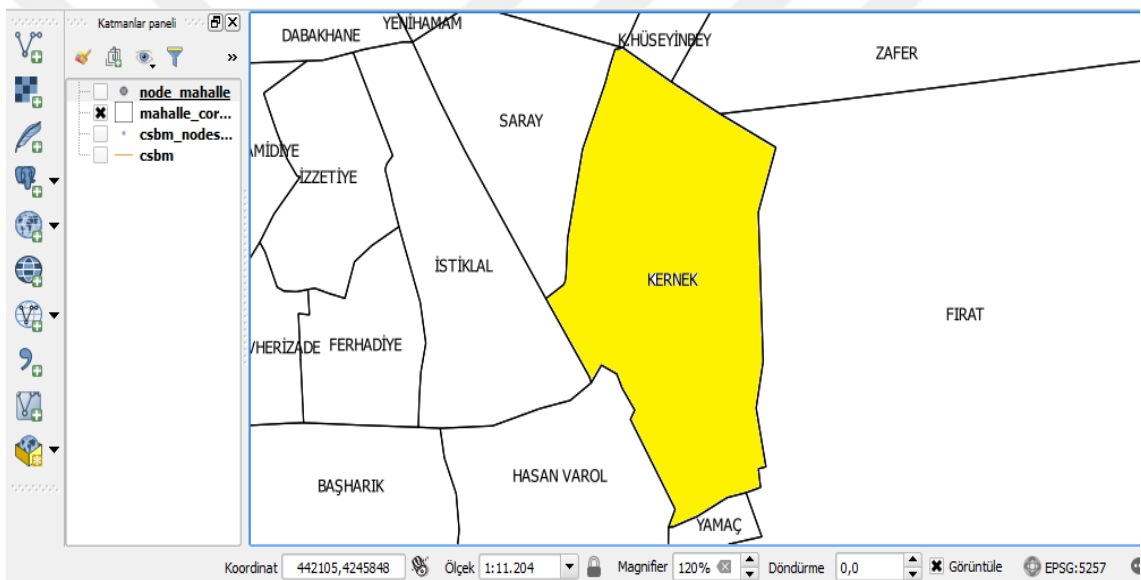
Şekil 3.3. QGIS Programına Malatya İl Haritasının Yüklenmiş Hali

3.4.2. Cadde/Sokak Uzaklık Matrislerinin Oluşturulması

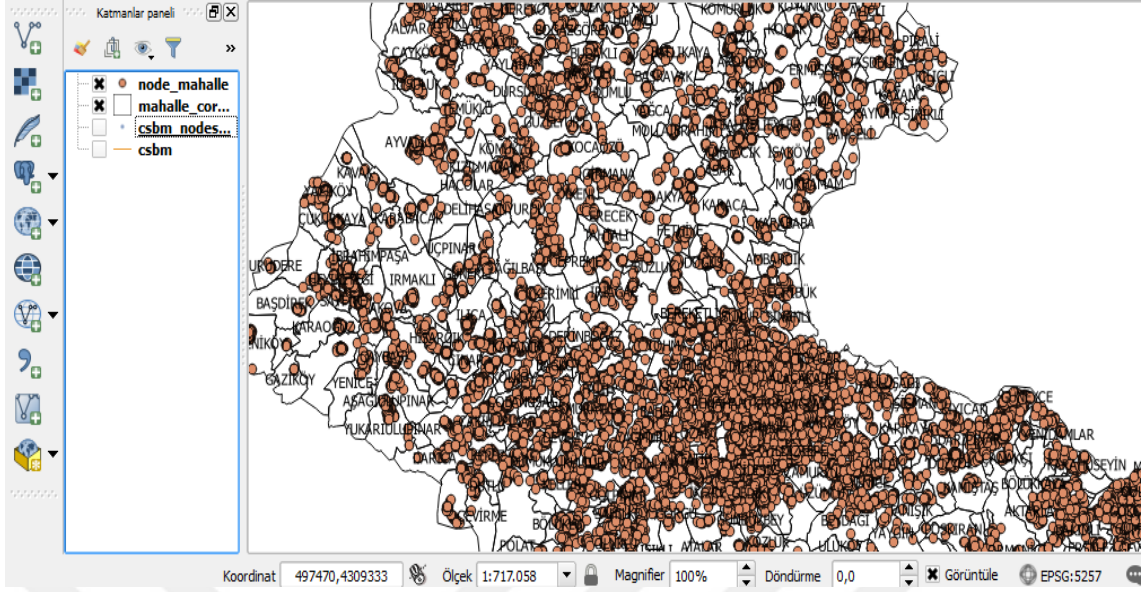
Her bir mahallenin içindeki caddeleri arası uzaklık matrislerini elde edebilmek için gerekli bağlantıların ve her bir bağlantı arası mesafenin elde edilmesi, yapılan düzenlemeler ile QGIS programından sağlanmıştır.

Cadde uzaklık matrislerinin nasıl elde edildiği Şekil 3.4’ de gösterilen “KERNEK” mahallesi üzerinden örneklendirilerek şekiller yardımıyla anlatılmıştır. Kullanılan bütün mesafeler metre cinsindedir. Ayrıca caddeler yönsüz olup, simetrik olduğu kabul edilmiştir.

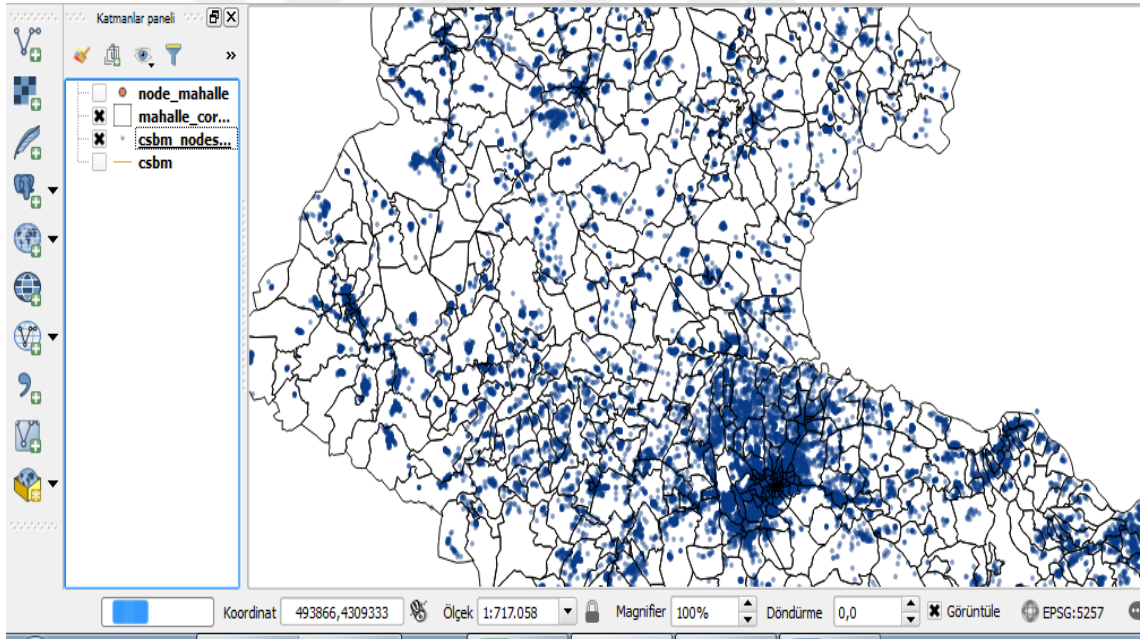
QGIS programına yüklenen Malatya il haritasına, programda yer alan “Katmanlar Paneli” kısmından Şekil 3.5 ’de mahallelere noktalar ataması, Şekil 3.6’ da mahalle içerisinde bulunan cadde sokak bağlantılarına düğümler verilmesi ve Şekil 3.7’ de caddelerin aralarındaki mesafeleri veren ayrıt katmanları haritaya yüklenmesi sağlanmıştır.



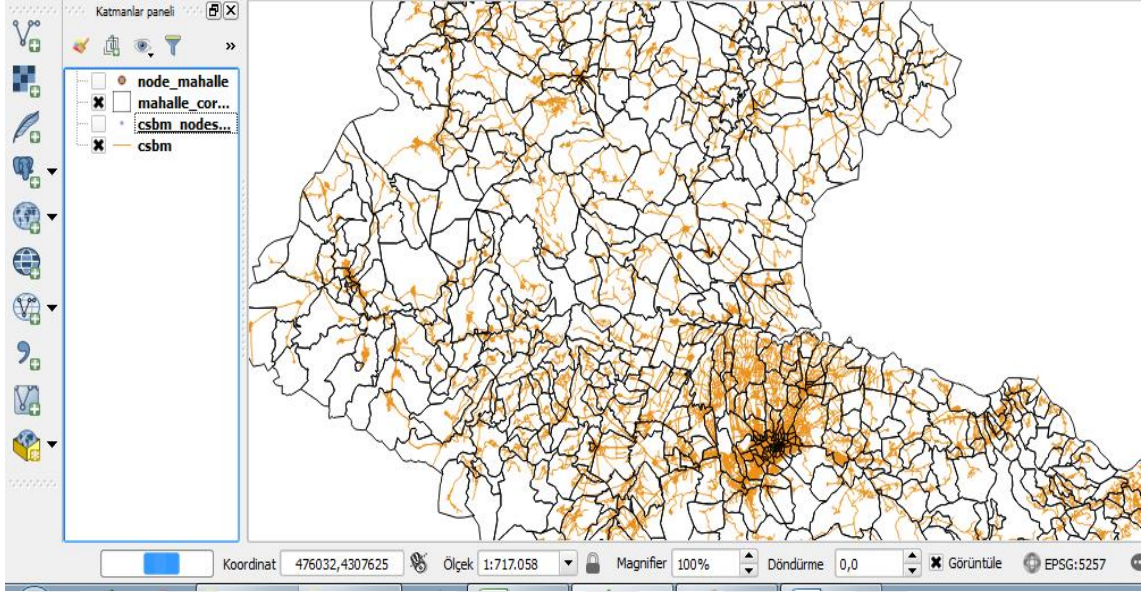
Şekil 3.4. QGIS’de “KERNEK” Mahallesi İçin Ekran Görüntüsü



Şekil 3.5. Her bir Mahalleye Verilen Dügümler Katmanının Gösterimi

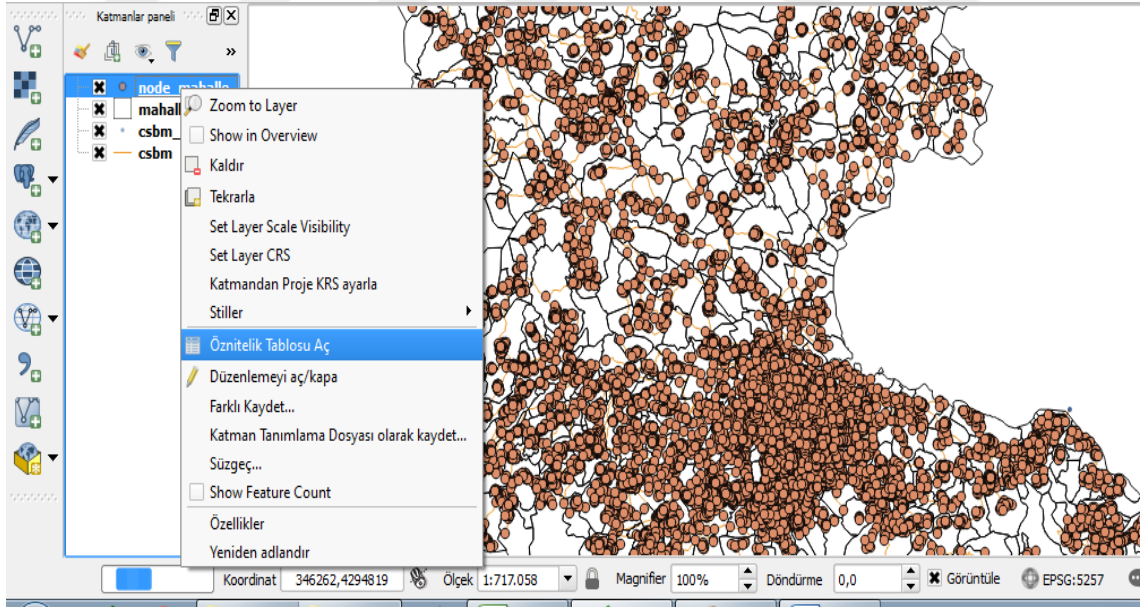


Şekil 3.6. Her bir Mahalle İçerisinde ki Cade/Sokak Başı ve Sonuna Dügümler Verilen Katmanın Gösterimi



Şekil 3.7. Her bir Mahalle İçinde ki Cadde/Sokak Arasında ki Ayrıtların Belirlendiği Katmanın Gösterimi

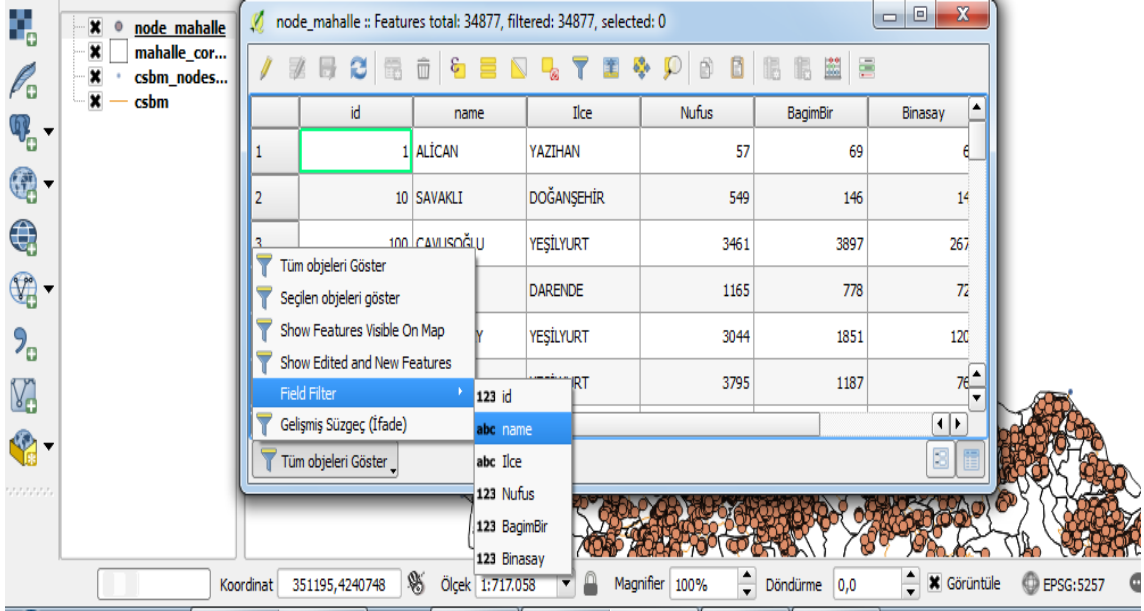
Bütün bu katmanlar QGIS programında belirlendikten sonra Şekil 3.8’ de görüldüğü gibi öncelikle “node_mahalle” kısmında sağ tıklayarak “Öznitelik Tablosunu Aç” sekmesi seçilir.



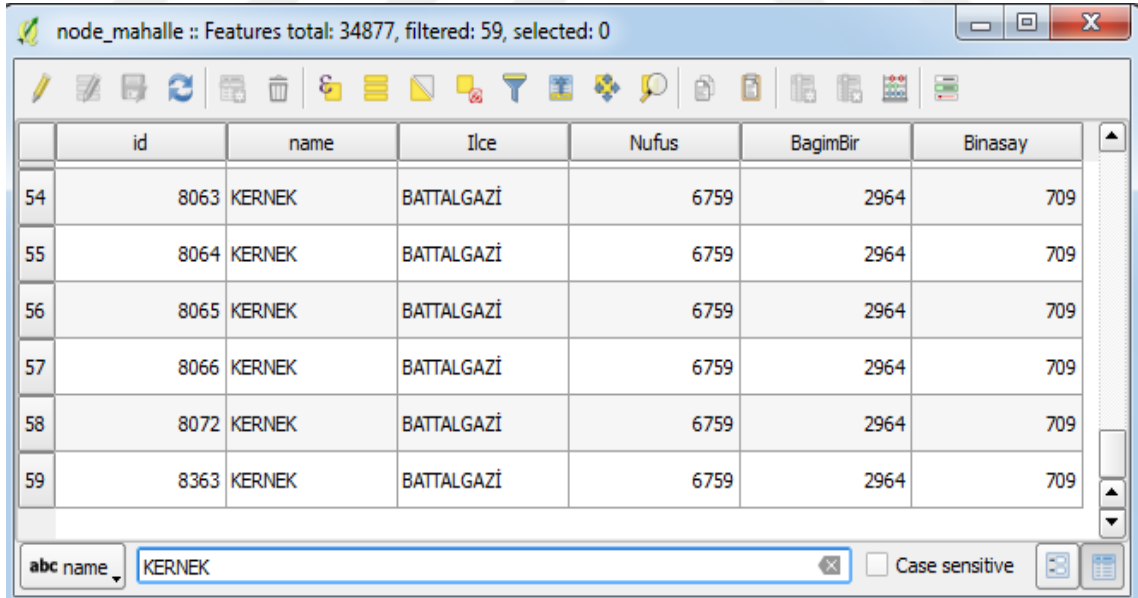
Şekil 3.8. Öznitelik Tablosundan Mahalle Seçimi

Şekil 3.9’ da görülen pencereden sol en alt köşede bulunan “Tüm Objeleri Göster”, “Field Filter” ve “name” seçilerek Şekil 3.10’ da ki gibi örnek olarak seçilen

“KERNEK” mahallesinin ismi yazılır. Böylelikle tekrar QGIS ana ekranına bakıldığında Kernek mahallesinin sınırları içerisinde bulunan cadde/sokak düğümlerinin seçildiği yani Şekil 3.11 deki gibi şeffaf renk olduğu görülmektedir.

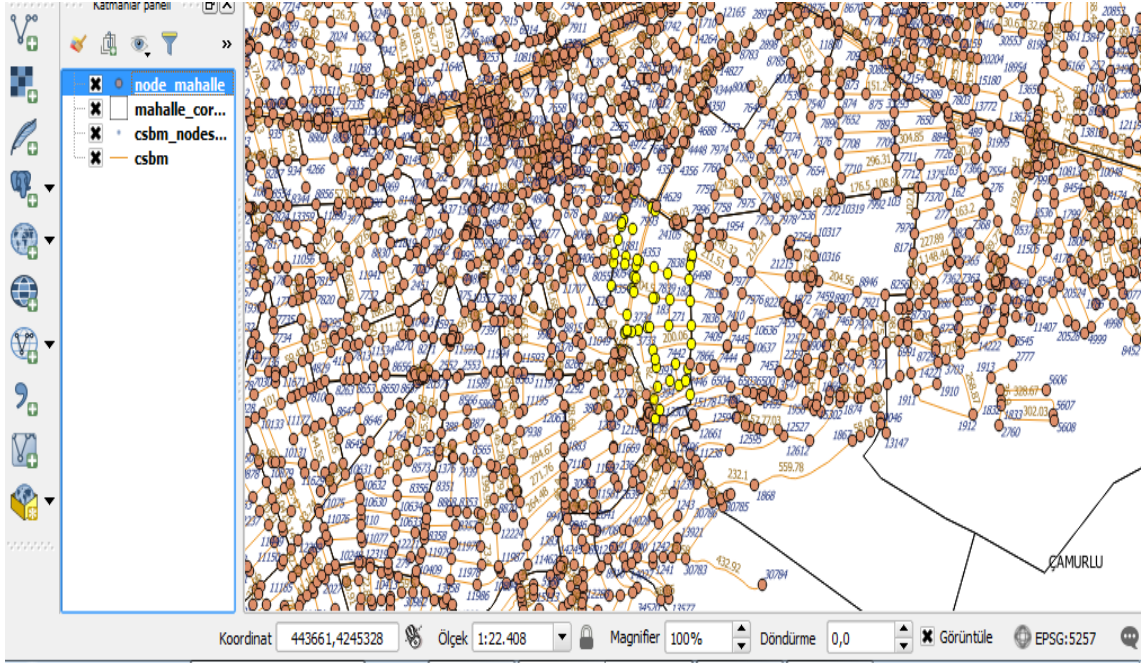


Şekil 3.9. Örnek Mahallemizin İsmi Yazılmasını Sağlayan Kısım



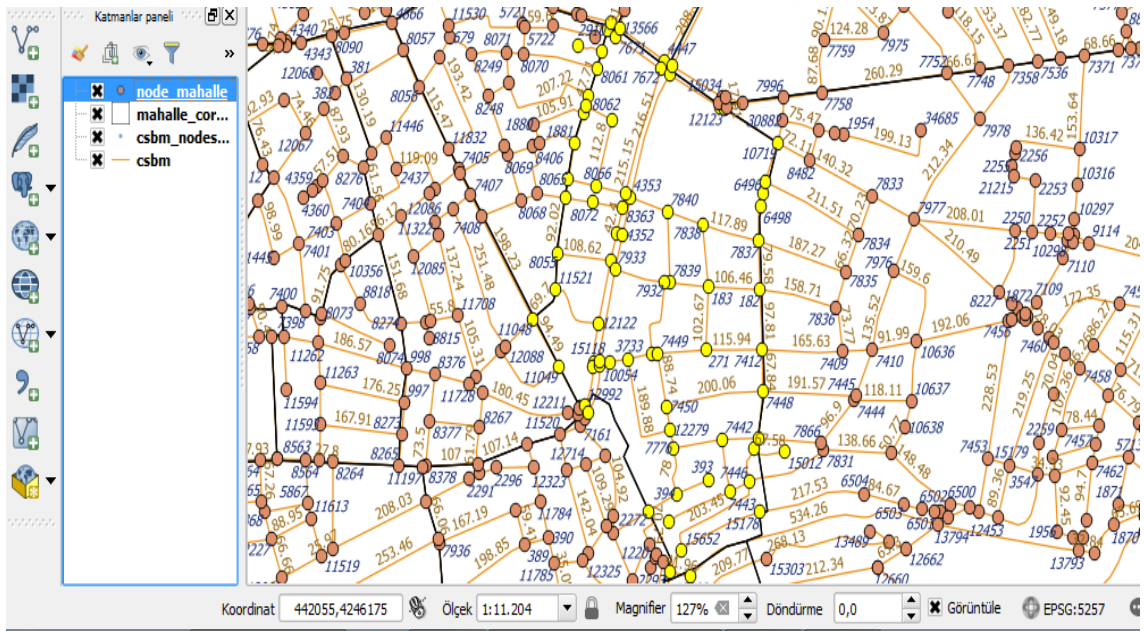
Şekil 3.10. Örnek Mahallemiz Olan Sadece “KERNEK” Mahallesinin Çağrılması

Şekil 3.10’ da görüldüğü gibi QGIS programı KERNEK mahallesi sınırları içerisinde toplam 59 cadde/sokak düğümü olduğu görülmektedir.



Şekil 3.11. QGIS Ekranında “KERNEK” Mahallesinin Cadde/Sokak Düğümlerinin Seçilmiş Hali

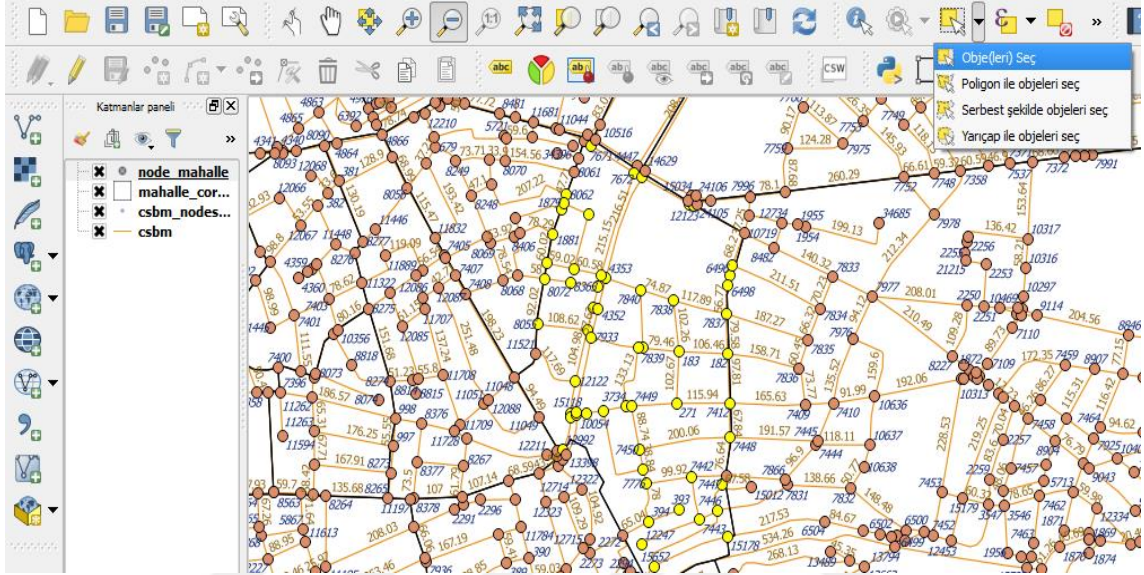
Her bir caddenin başına ve sonuna Şekil 3.12’ de görüldüğü gibi numaralar ilderde matris oluşumunda Excel de “ID” kısmına yazılacaktır.



Şekil 3.12. QGIS Ekranında Cadde /Sokakların Baş ve Sonuna ID Verilmiş Hali

Daha sonra Şekil 3.12’de görüldüğü gibi harita büyütüldüğünde ise bazı cadde/sokak düğümlerinin mahalle sınırları dışında kaldığı için programın o düğümlerin

seçili mahalle içine alınmadığı gözlemlenmiştir. Bu optimum rotanın ele edilmesinde doğru sonucu çıkarmadığı için mahallenin sınırı dışında kalan düğümler daha sonra Şekil 3.13 deki gibi “Objeler) Seç” sekmesi tıklandıktan sonra Ctrl tuşuna basılarak istenilen düğümler KERNEK mahallesi düğümlerine elle ilave edilmiştir.



Şekil 3.13. “KERNEK” Mahallesi Sınırı Dışında Kalan Düğümlerin Seçimi

Excel dökümüne aktaracağımız ID numaralarının dökümünü alabilmek için Öznitelik Tablosu açılarak Şekil 3.14’ de görüldüğü gibi “Seçilen objeleri göster” sekmesi tıklanmıştır.

node_mahalle :: Features total: 34877, filtered: 59, selected: 76

	id	name	İlce	Nufus	BagimBir	Binasay
54	8063	KERNEK	BATTALGAZİ	6759	2964	709
55	8064	KERNEK	BATTALGAZİ	6759	2964	709
		Tüm objeleri Göster	BATTALGAZİ	6759	2964	709
		Seçilen objeleri göster	BATTALGAZİ	6759	2964	709
		Show Features Visible On Map	BATTALGAZİ	6759	2964	709
		Show Edited and New Features	BATTALGAZİ	6759	2964	709
		Field Filter	BATTALGAZİ	6759	2964	709
		Gelişmiş Süzgeç (İfade)	BATTALGAZİ	6759	2964	709

abc name
abc name KERNEK

Şekil 3.14. Seçilen Objelerin İlave Edilmesi

Şekil 3.15 'de görüldüğü gibi “KERNEK” mahallemize elle seçilerek ilave edilen düğümlerden sonra düğüm sayısı 59 dan 71' ye çıkmıştır.

node_mahalle :: Features total: 34877, filtered: 71, selected: 71

	id	name	Ilce	Nufus	BagimBir	Binasay	Mahalle_Id
56	11049	İSTİKLAL	BATTALGAZİ	5086	2689	827	699
57	11050	KERNEK	BATTALGAZİ	6759	2964	709	537
58	11521	SARAY	BATTALGAZİ	2888	2118	861	497
59	12121	KERNEK	BATTALGAZİ	6759	2964	709	537
60	12122	KERNEK	BATTALGAZİ	6759	2964	709	537
61	12123	KERNEK	BATTALGAZİ	6759	2964	709	537
62	12247	KERNEK	BATTALGAZİ	6759	2964	709	537
63	12279	KERNEK	BATTALGAZİ	6759	2964	709	537
64	12992	KERNEK	BATTALGAZİ	6759	2964	709	537
65	13303	YAMAÇ	BATTALGAZİ	1600	914	744	424
66	13398	HASAN VAROL	BATTALGAZİ	5799	2487	553	521
67	14206	KERNEK	BATTALGAZİ	6759	2964	709	537
68	15012	FIRAT	BATTALGAZİ	14319	7945	1335	709
69	15118	KERNEK	BATTALGAZİ	6759	2964	709	537
70	15178	KERNEK	BATTALGAZİ	6759	2964	709	537
71	15652	KERNEK	BATTALGAZİ	6759	2964	709	537

Seçilen objeleri göster KERNEK Case sensitive

Şekil 3.15. İlave Edilen Düğümler İle Gösterim

QGIS' de oluşturulan noktalar ve çizgiler (ayrıntlar) dosyaları Excel' e aktarıldığında caddeler/sokaklar arasındaki mesafeyi gösteren tamamı Ek 1' de paylaşılan 71 düğüm, 92 ayrıttan oluşan düğümler arası uzaklık matrisi elde edilmiştir (Şekil 3.16).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	
1	id																							
2	182	0,00	106,46																				97,81	7442
3	183	106,46	0,00	102,67																				
4	271		102,67	0,00																			115,94	
5	393			0,00	70,13																			
6	394			70,13	0,00																			
7	1879				0,00	52,41																		
8	1881				52,41	0,00																		
9	2272				65,04				0,00															
10	2910								0,00	36,15														
11	2911								36,15	0,00														
12	3733										0,00	49,97											36,78	
13	3734										49,97	0,00												
14	4351												0,00	11,95			46,77							
15	4352												11,95	0,00	63,96									
16	4353												63,96	0,00										
17	4354												46,77		0,00									
18	6496																0,00	19,83	21,12					
19	6497																19,83	0,00						
20	6498																21,12		0,00					
21	7411																			0,00				
22	7412	97,81		115,94																		0,00		
23	7442																						0,00	
24	7443																						87,33	
25	7446																							
26	7447																							
27	7448																						67,84	
28	7449			99,51										11,46										
29	7450																							
30	7451																							75,66
31	7671									19,19														

Şekil 3.16. “KERNEK” Mahallesi Örneği İçin Elde Edilen Cadde/Sokak Mesafe Matrisi

Malatya ilinin Battalgazi ve Yeşilyurt ilçelerine ait toplam 154 mahallesi için (mahalle sayısının 143 den 154'e çıkmasındaki neden başlık 3.6.1 *Varsayım 2'* de anlatılmıştır) QGIS programında aynı işlem yapılarak çalışmanın ikinci kısmı olan Yönsüz Çinli Postacı Probleminde kullanılmak üzere cadde/sokak uzaklık matrisleri oluşturulmuştur.

3.5. Uygulamanın Matematiksel Modelleri

Uygulama iki kısımdan oluşmakta olup, birinci kısım mahalleleri gün ve zaman kısıtlarına bağlı olarak minimum araca göre dağıtıp minimum mesafede bu araçların mahallelere rotalanmasını sağlamaktadır. Bu aşama Zaman Pencereci Araç Rotalamadır. Rotalama işlemine başlamadan önce mahalleler yakınlık durumları ve günlük bir aracın 4 saatlik süre içerisinde toplam alabilecekleri mesafeye göre kümelenemiştir.

Uygulamanın ikinci kısmı ise araçlara ve günlere göre dağıtılmış olan mahallelerin içerisindeki her cadde/sokağın tamamında gezinerek ilaçlama işlemini gerçekleştirmek üzere mahalle içerisinde optimum rotayı belirlemektir. Problem Yönsüz Çinli Postacı Problemidir.

3.5.1. Zaman Pencereci Araç Rotalama Probleminin Matematiksel Modeli

Uygulamanın Birinci Kısım Olan *Zaman Pencereci Araç Rotalama Probleminin Matematiksel Modeli*;

$$G = (V, A) \quad V = (v_0, v_1, \dots, \dots, \dots, v_n) \quad (\text{Düğüm seti})$$

$$A = \{(v_i, v_j) \mid v_i, v_j \in V, i \neq j\} \quad (\text{Ayrıt seti})$$

Parametreler

$$N = \text{Toplam mahalle sayısı} \quad (154)$$

$$K = \text{Toplam araç sayısı} \quad (12)$$

$$T = \text{Toplam gün} \quad (6)$$

O = Depo

d_{ij} = i - j arası mesafe

q_{ij} = i . mahalle ile j . mahalle arasındaki seyahat süresi

$Q_{kt} = k$. aracın t . günde gidebileceği toplam cadde uzunluğu

$a_i = i$. mahalle için en erken servis zamanı (19:00)

$b_i = i$. mahalle için en geç servis zamanı (22:00)

$[a_i, b_i] = i$. mahalle için zaman penceresi

$f_i = i$. mahalle için servis süresi

Karar Değişkenleri

$$X_{ijk}^t = \begin{cases} 1, & \text{Eğer } k \text{ aracı } t \text{ gününde } i. \text{ mahalleden } j. \text{ mahalleye giderse} \\ 0, & \text{Aksi halde} \end{cases} \quad (3.1)$$

$$s_{ik}^t = t \text{ gününde } k \text{ aracı tarafından } i \text{ mahallesine servise başlama zamanı} \quad (3.2)$$

Amaç Fonksiyonu

$$\text{Min } Z = \sum_{k \in K} \sum_{t \in T} \sum_{(i,j) \in A} d_{ij} X_{ijk}^t \quad (3.3)$$

Kısıtlar

$$\sum_{t=1}^T \sum_{k=1}^K \sum_{j=0}^N X_{ijk}^t = 1 \quad \forall i \in N \quad (3.4)$$

$$\sum_{t=1}^T \sum_{k=1}^K \sum_{i=0}^N X_{ijk}^t = 1 \quad \forall j \in N \quad (3.5)$$

$$\sum_{j=1}^N X_{0jk}^t = 1 \quad \forall k \in K \quad (3.6)$$

$$\sum_{j=1}^N X_{i0k}^t = 1 \quad \forall k \in K \quad (3.7)$$

$$\sum_{i=0}^N X_{ihk}^t - \sum_{j=0}^N X_{hjk}^t = 0 \quad \forall k \in K \quad (3.8)$$

$$\sum_{k=1}^K \sum_{j=1}^N X_{0jk}^t \leq K \quad (3.9)$$

$$\sum_{j=0}^N \sum_{i=0}^N d_{ij} \sum_{j=0}^N X_{ijk}^t \leq Q_{kt} \quad (3.10)$$

$$a_i \leq s_{ik}^t + f_i \leq b_i \quad (3.11)$$

$$(s_{ik}^t + f_i + q_{ij} - s_{jk}^t) \leq (1 - X_{ijk}^t)B \quad (3.12)$$

$$\sum_{i \in P} \sum_{j \in P} X_{ijk}^t \leq |P| - 1 \quad P \subseteq V, |P| \geq 2 \quad (3.13)$$

$$X_{ijk}^t \in \{0,1\} \quad \forall (i,j) \in A, \forall k \in K, \forall t \in T \quad (3.14)$$

(3.1) ve (3.2) karar deęişkenleri olup, (3.3) ise amaç fonksiyonu olup mesafenin minimizasyonunu saęlar.

(3.4) ve (3.5) nolu kısıt; her bir mahallenin sadece bir araç tarafından ziyaret edilmesini saęlar.

(3.6) nolu kısıt; her bir aracın depodan bir kere çıkmasını saęlar.

(3.7) nolu kısıt; her bir aracın depoya bir kere dönmesini saęlar.

(3.8) nolu kısıt; h mahallesinden ayrılan aracın sadece o mahalleye gidilmişse gerçekleşmesini saęlar.

(3.9) nolu kısıt; maksimum K tane tur olmasını saęlar.

(3.10) nolu kısıt; her bir araç bir günde gidebileceęi toplam cadde uzunluęunu aşacak şekilde hizmet veremez.

(3.11) nolu kısıt; her bir mahallenin belirli bir zaman penceresinde servis görmesini saęlar.

(3.12) nolu kısıt; k aracı t gününde eęer i . mahalleden j . mahalleye seyahat ediyorsa aracın j . mahalleye $(s_{ik}^t + f_i + q_{ij})$ ' den önce ulaşamamasını saęlar.

(3.13) nolu kısıt; alt tur engelleme kısıtı

(3.14) nolu kısıt; tamsayı olma kısıtıdır.

3.5.2. Yönsüz Çinli Postacı Probleminin Matematiksel Modeli

Uygulamanın İkinci Kısmı Olan *Yönsüz Çinli Postacı Matematiksel Modeli*'nin uygulamamıza uyarlanmış hali aşağıda ki gibidir;

$G = (V, A)$ grafi olmak üzere,

Parametreler;

V = düğümler

A = ayrıtlar (yönsüz)

$O = \text{depo}$

$d_{ij} = \text{mesafe}$

$x_{ij} = i$ düğümü ile ayrıt bağlantısı olan j düğümü arasında var olan akışın gerçekleştirilme (i, j ayrıtını kullanma) sayısını gösterir.

Karar Değişkenleri;

$$x_{ij} = \begin{cases} 1, & i \text{ 'den } j \text{ 'ye geçilirse} \\ 0, & \text{aksi halde} \end{cases} \quad (3.15)$$

Amaç Fonksiyonu;

$$\text{Min } Z = \sum_{(i,j) \in A} d_{ij} \times x_{ij} \quad (3.16)$$

Kısıtlar;

$$\sum_{i \in V} x_{ij} = \sum_{i \in V} x_{ji} \quad \forall j \in V \quad (3.17)$$

$$\sum x_{ij} \geq 1 \quad \forall (i, j) \in A \quad (3.18)$$

$$\sum_{j \in V} x_{0j} \geq 1 \quad (3.19)$$

$$\sum_{j \in V} x_{i0} \geq 1 \quad (3.20)$$

$$\sum_{i,j \in S} x_{ij} - |S| \leq \sum_{i \in S, j \notin S} x_{ij} - 1 \quad \forall S \subseteq SS \quad (3.21)$$

$$x_{ij} \geq 0 \text{ ve tamsayı} \quad \forall (i, j) \in A \quad (3.22)$$

(3.15) karar değişkeni olup, (3.16) mesafenin minimizasyonunu sağlayan amaç fonksiyonudur.

(3.17) nolu kısıt; her düğüme giren ayrıt ile her düğümden çıkan ayrıt sayısı eşit olmalıdır. Akış her düğümden korunmalıdır.

(3.18) nolu kısıt; Her ayrıttan en az bir kez geçme zorunluluğudur.

(3.19) ve (3.20) nolu kısıtlar; tüm araçların bir depodan tura başlaması ve aynı depoda son bulma şartıdır.

(3.21) nolu kısıt; SS , V 'nin tüm alt kümeler setidir. Bir aracın her turu depoda başlamalı ve depoda bitmelidir. Depoyu ikamet etmeyen herhangi bir alt tur bulunmamalıdır.

(3.22) nolu kısıt; negatif olmama kısıtıdır.

3.6. Modellerin Uygulanması

Bu bölümde 3.5.1 ve 3.5.2' de verilen matematiksel modeller programlama üzerinde uygulanmıştır. Her iki model Malatya Büyükşehir Belediyesi Çevre Koruma ve Kontrol Dairesi Başkanlığı, Atık Yönetim ve İlaçlama Şube Müdürlüğü' ne ait homojen 12 adet ilaçlama araçlarının haftanın 6 günü ve 4'er saat çalışmak koşulu ile Battalgazi ve Yeşilyurt ilçelerine bağlı mahallelerinin tüm cadde/sokaklarını ilaçlama işleminde optimum rotalarının elde edilmesi için oluşturulmuştur.

Modeller oluşturulurken ilgili kurum ile gerekli görüşmeler yapılmış gerekli kısıtlar ve belirlenen istekler doğrultusunda model son halini almıştır.

Modellerimizin çözümü Tablo 3.2 deki gibi 3 aşama ile gerçekleştirilmektedir.

Tablo 3.2. Önerilen Modellerin Çözüm Aşamaları

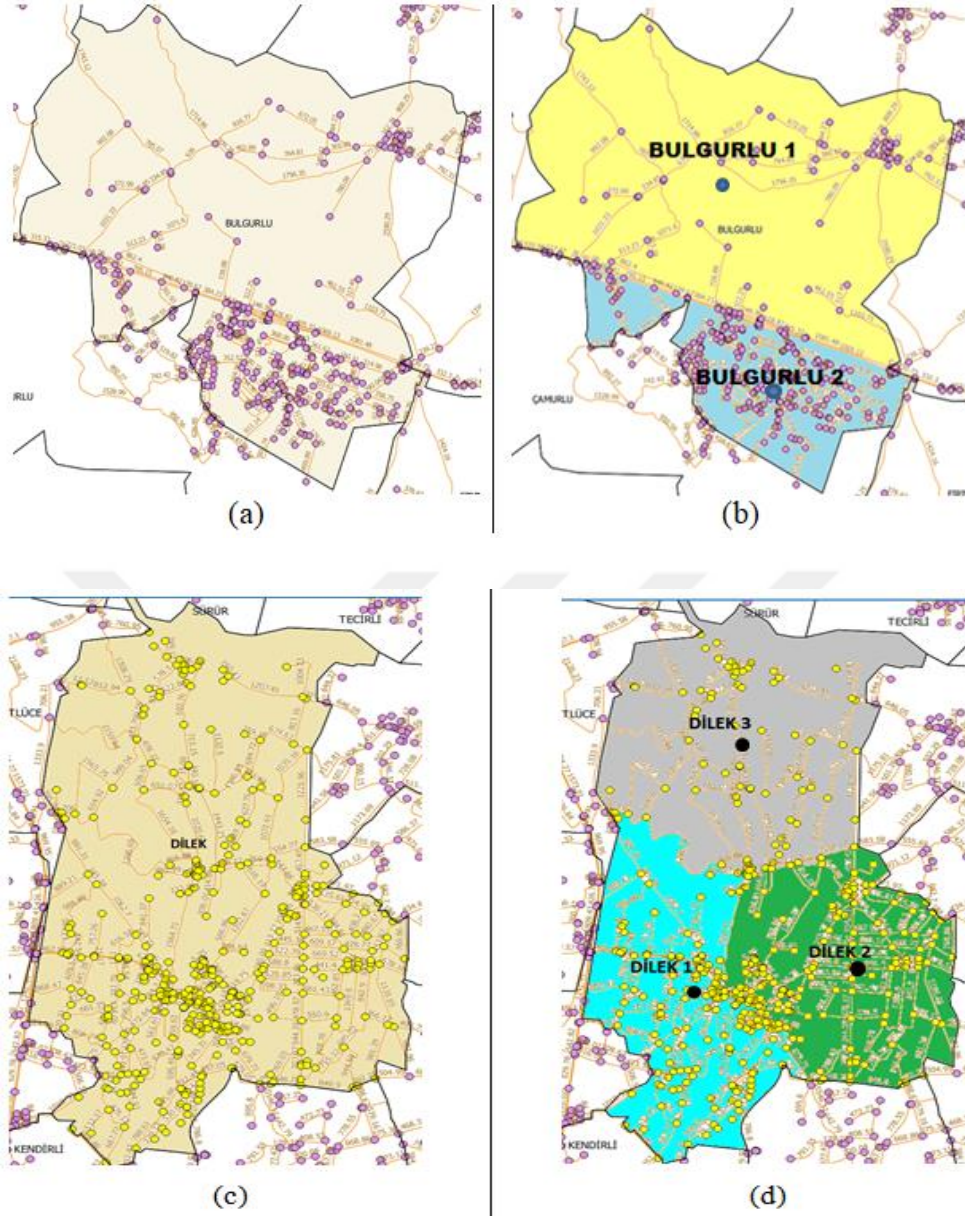
1. Aşama	Mahallelerin birbirlerine göre yakınlık durumları ve bir aracın günlük ilaçlayabileceği süre göz önüne alınarak kümeleme yapılması
2. Aşama	Araç rotalama probleminin Genetik Algoritma ile çözülmesi
3. Aşama	Çinli postacı probleminin Yapay Arı Kolonisi Algoritması ile çözülmesi

3.6.1. Zaman Pencereci Araç Rotalama Modelinin Uygulanması

Malatya ilinin Battalgazi ve Yeşilyurt ilçelerinin ilaçlama işleminde Atık Yönetim ve İlaçlama Şube Müdürlüğü' nün belirlediği 143 adet mahalleyi haftanın 6 günü saat 19:00 ile 23:00 arası çalışmak üzere Tablo 3.1. deki gibi 12 araca dağıtmıştır. Bu dağıtım işleminin belirlenmesinde ve ilaçlama işleminin gerçekleşmesinde aşağıda ki 3 durum gözlemlenmiştir;

1. Hiçbir durum veya kısıt göz önüne alınmadan tamamen rasgele güzergâh belirlenmektedir.

Varsayım 2. Hiçbir mahallenin bölünemez olduğudur. Yani bir araç için o gün belirlenen mahallenin/mahalle gruplarının tüm cadde/sokakları ilaçlamadan depoya dönmesi mümkün olmayacaktır. Bu varsayımımızın desteklenmediği büyük mahalleler diye isimlendirdiğimiz bu mahallelerde, bütün cadde/sokakların hem sürenin hem de ilaç haznesinin yetmemesinden dolayı ilaçlanamadığı gözlemlenmiştir. Bu nedenle bir günlük sürenin ve araçdaki ilacın yetmeyeceği durumda büyük mahalleler belirlenerek o mahalleler toplam cadde uzunlukları dikkate alınarak QGIS programı yardımıyla ikiye veya üçe bölünmüş ve her biri birer mahalle gibi düşünülmüştür. Örneğin; Bulgurlu mahallesi Bulgurlu 1 ve Bulgurlu 2 olmak üzere, Dilek mahallesi Dilek 1, Dilek 2 ve Dilek 3 olmak üzere Şekil 3.18 deki gibi bölünmüştür. Daha sonra QGIS programı ile bu ayrı mahalleye orta nokta atanarak diğer mahallelere olan uzaklıkları mesafe matrisine kolayca dâhil edilmiştir.



Şekil 3.18. (a)Bulgurlu Mahallesinin Bölünmemiş Hali (b) Bulgurlu Mahallesinin İkiye Bölünmüş ve QGIS' den Orta Noktalar Atanmış Hali (c) Dilek Mahallesinin Bölünmemiş Hali (d) Dilek Mahallesinin Üçe Bölünmüş ve QGIS' den Orta Noktalar Atanmış Hali

Varsayım 3. *Varsayım 2'* de bahsedilen bölme işleminde Malatya Belediyesi CBS biriminden aldığımız verilere göre bölme işlemi uyguladığımız mahallere ait toplam cadde/sokak uzunlukları bilgisi olduğundan bölme işlemi sonrası bu mahallelerin cadde/sokak uzunlukları QGIS programı ile elle belirlenmiştir. Belirleme işleminde ayrı birer mahalle gibi düşünülüp her kısma ayrı ayrı ölçüm yapılarak yazılmıştır.

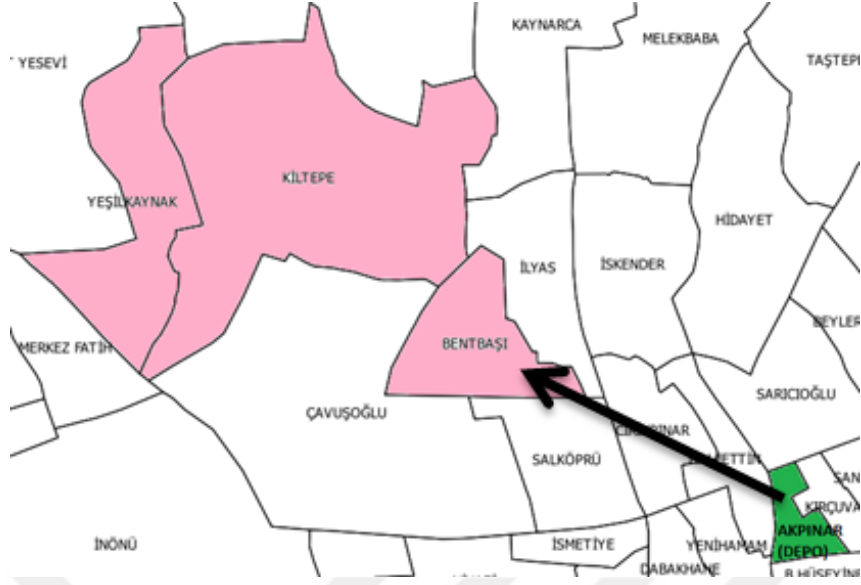
Örneğin CBS biriminden alınan bilgiye göre Bulgurlu mahallesinin toplam cadde uzunluğu 61640m olup bölme işlemi sonucu Bulgurlu 1 mahallesinin ve Bulgurlu 2 mahallesinin gerçek cadde uzunluklarını belirlenememekteydi. Bu nedenle bu mahallelerin cadde uzunluklarına QGIS programından elle ölçüm yapılmıştır. Bütün bölünmüş mahallelere bu aynı işlem uygulanarak Tablo 3.3 ve Tablo 3.4'e eklenmiştir.

Varsayım 4. *Varsayım 2'* de bahsedilen ikiye veya üçe bölünmüş büyük mahallelerin ilaçlanması aynı araç tarafından hemen bir sonraki gün yapılacaktır. Yani Tablo 3.14 deki gibi Bulgurlu 1 mahallesi 4. Araç tarafından pazartesi günü ilaçlanıyorsa, Bulgurlu 2 mahallesi gene 4. Araç tarafından hemen bir sonraki gün olan salı günü ilaçlanacaktır. Bulgurlu 1 mahallesini pazartesi günü ilaçlama işleminde ki aracı kullanan şoföre, hemen bir sonraki gün Bulgurlu 2 mahallesinin atanmasının sebebi, bölünmüş bölgeyi tanıyan şoförün bir önceki gün ilaçladığı yerlerden tekrar geçme olasılığını engellemektedir.

Varsayım 5. Araçların rotalama işleminden önce yapılan kümeleme sonucu elde edilen mahalle gruplarında yer alan mahallelerden hangisi depo olarak belirlenen Akpınar mahallesine yakınsa rotalama işleminde aracın önce o mahalleden giriş yaparak ilaçlamayı gerçekleştireceği düşünülmüştür. Bu varsayımın ele alınmasındaki sebep ise aracın ilaçlama haznesi dolu olduğu için yüklü olan aracın daha çok yakıt tüketeceği düşünülmüştür. Depodan çıkan aracın gireceği ilk mahalleye kadarki gelişi, mahalleler arası geçişi ve tekrar depoya dönüşündeki hızı, şehir içi kuralları göz önünde bulundurularak 60km olarak belirlenmiştir.

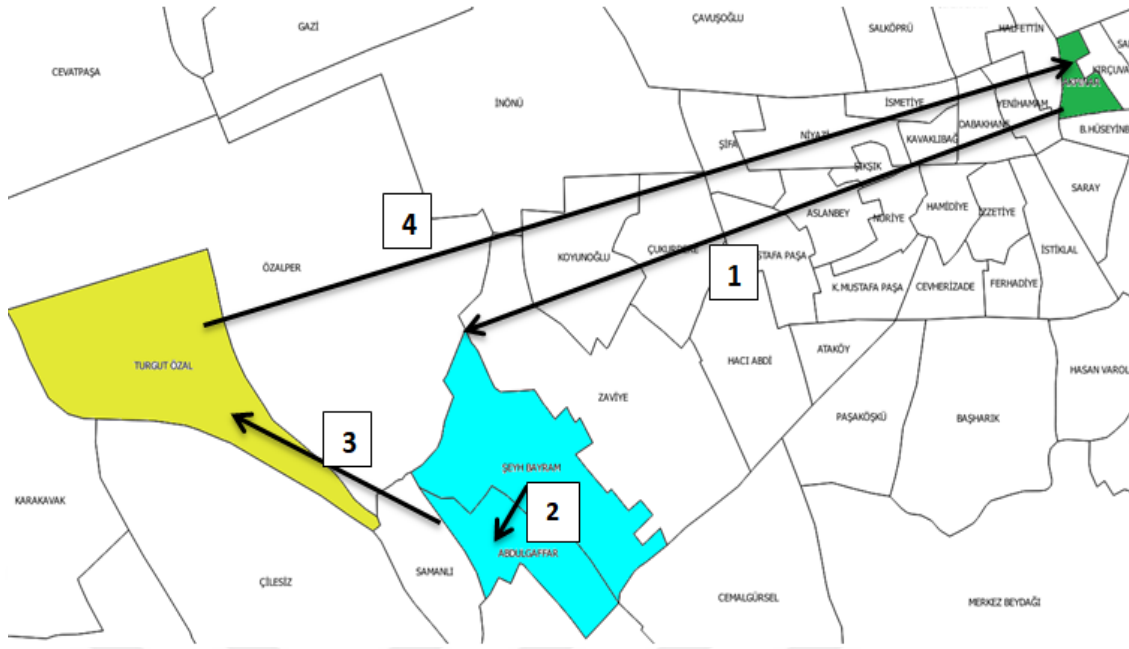
Varsayım 5 için oluşabilecek mahalle grubu sayısına göre iki durum söz konusudur;

Varsayım 5.1. Tek bir mahalle grubunun olması durumu: Örneğin; Yeşilyurt ilçesine ait Tablo 3.5' de görüldüğü gibi 2. aracın Cumartesi günü gideceği mahalle grubu 1 adet olup Bentbaşı – Kıltepe - Yeşilkaynak mahalleleridir. Yeşilyurt ilçesine ait mahalleler arası mesafe matrisine göre ve Şekil 3.19'da görüldüğü üzere Bentbaşı mahallesi Kıltepe ve Yeşilkaynak mahallerine göre Akpınar' a daha yakın olduğu için Akpınar mahallesinden çıkan araç öncelikle Bentbaşı mahallesinden giriş yapacaktır. Daha sonra sırasıyla Kıltepe ve Yeşilkaynak mahallerine gidecektir.



Şekil 3.19. Depodan Çıkan Aracın Bir Adet Kümelenmiş Mahalle Grubunun Olması Durumunda İzleyeceği Yol

Varsayım 5.2. Birden çok mahalle grubunun olması durumu: Örneğin; Yeşilyurt ilçesine ait Tablo 3.10' da görüldüğü gibi 6. aracın Perşembe günü gideceği mahalle grubu 2 adet olup, ilk mahalle grubu Şeyhbayram - Abdulgaffar, ikinci mahalle grubu Turgut Özal mahalleleridir. Yeşilyurt ilçesine ait mesafe matrisine göre ve Şekil 3.20'de de görüldüğü üzere araç öncelikle Şeyhbayram - Abdulgaffar olan ilk mahalle grubundan başlayacaktır. Bu gruptan ilk gireceği mahalle Akpınar' a en yakın olan Fatih mahallesidir. Şeyhbayram mahallesinden sonra Abdulgaffar mahallesine geçecektir. İlk mahalle grubunu bitiren araç ikinci mahalle grubunda bir adet mahalle bulunması nedeniyle en sonda da Turgut Özal mahallesine uğrayarak araç depoya dönecektir.



Şekil 3.20. Depodan Çıkan Aracın Birden Çok Kümelenmiş Mahalle Grubunun Olması Durumunda İzleyeceği Yol

Varsayım 6. Bir kutu böcek ilacı (1.5lt) ve 60 lt su ile dolu olan aracın ilaçlama haznesi yaklaşık 2-2.5 saat ilaçlama işlemini sürdürmeyi sağlamaktadır. Bu nedenle ilaçlama işlemi için ayrılan toplam 4 saatlik sürenin tamamlanabilmesi için araçların 2-2.5 saat sonunda ilaç haznesine yeniden böcek ilacı ve su ile doldurmaları gerekmektedir. Bu durum göz önüne alınarak yapılan programlamada 15 dk lık boş zaman dilimi ayrılmıştır.

Bu varsayımlar altında özellikle *Varsayım 2'* de bahsedilen büyük boyutlu mahalleler (bir araca ayrılan günlük süre ve ilaçlama anındaki hız ele alınarak belirlenen km' yi aşan mahalleler) olarak bahsedilen yani 4 saatlik ilaçlama süresi içinde tüm caddelerin ilaçlanamayacağı mahallelerdir. Bunlar; Yeşilyurt ilçesine bağlı olan *Tecde*, *Özal*, *Yaka* ikiye, *Yakınca* ve *Dilek* üçe, Battalgazi ilçesine bağlı olan *Bulgurlu* ve *Hanımünçiftliği* ikiye, *Orduzu* üçe bölünerek ayrı birer mahalle gibi ele alınmıştır. Bu nedenle toplam mahalle sayısı 143 den 154' e çıkmıştır.

Hem problemin boyutunun büyük olması, hem de QGIS programına yüklü Malatya ili haritasında uygulama sonuçlarının daha iyi incelenebilmesi açısından Battalgazi ve Yeşilyurt ilçelerine bağlı Tablo 3.3 ve Tablo 3.4' deki mahalleler ayrı ayrı uygulamaya tabi tutulmuştur.

Tablo 3.3. Yeşilyurt İlçesine ait Mahalleler ile Bölünmüş veya Cadde/Sokak Uzunlukları Belli Olmayan Bazı Mahallelerinin Toplam Cadde/Sokak Uzunlukları (Kaynak: QGIS programı ile elle ölçülmüştür)

Mahalle İsimleri	Hesaplanan Mahallelerin Cadde Uzunlukları	Mahalle Mahallelerin İsimleri	Hesaplanan Cadde Uzunlukları
ABDULGAFFAR		KOŞU	
AŞAĞI BAĞLAR		KOYUNOĞLU	
ATATÜRK		KUŞDOĞAN	
BAHÇEBAŞI		MAHMUTLU	
BENTBAŞI		MELEKBABA	
*BİNDAL	28912,79712	MERKEZ FATİH	
*BOSTANBAŞI	26586,9141	MULLAKASIM	
CEMALGÜRSEL		*ÖZAL	27533,0978
CEVATPAŞA		ÖZALPER	
CUMHURİYET		SALKÖPRÜ	
ÇARMUZU		SAMANKÖY	
ÇAVUŞOĞLU		SAMANLI	
ÇAYIRKÖY		SEYRAN	
ÇİLESİZ		SU	
ÇUKURDERE		*SULUKÖY	10249,6467
*DİLEK 1	24317,6496	*SÜTLÜCE	15509,2138
*DİLEK 2	35377,2183	*ŞAHNAHAN 1	25485,6229
*DİLEK 3	31596,5462	*ŞAHNAHAN 2	29269,6213
DURANLAR		ŞEYH BAYRAM	
DURULDU		*TECDE 1	27626,5195
FATİH		*TECDE 2	32612,5195
GAZİ		TEPEKÖY	
GEDİK		TOHMA	
GÖKTARLA		*TOPRAKTEPE	9391,661113
*GÖRGÜ	32909,05483	*TOPSÖĞÜT	27732,32491
*GÖZENE	27918,42435	TURGUT ÖZAL	
*GÜNDÜZBEY	30880,08253	*YAKA 1	24521,3995
HIROĞLU		*YAKA 2	31917,3995
HOCA AHMET		*YAKINCA 1	29453,535
YESEVİ		*YAKINCA 2	33486,238
İLYAS		*YAKINCA 3	17531,494
İNÖNÜ		YAVUZ SELİM	
KARAKAVAK		YENİ(yeşilyurt)	
KAYNARCA		YEŞİLKAYNAK	
KENDİRLİ		ZAVİYE	
KİLTEPE			

Tablo 3.4. Battalgazi İlçesine ait Mahalleler ile Bölünmüş veya Cadde/Sokak Uzunlukları Belli Olmayan Bazı Mahallelerinin Toplam Cadde/Sokak Uzunlukları
(Kaynak: QGIS programı ile elle ölçülmüştür)

Mahalle İsimleri	Hesaplanan Mahallelerin Cadde Uzunlukları	Mahalle İsimleri	Hesaplanan Mahallelerin Cadde Uzunlukları
AKPINAR(DEPO)		İZZETİYE	
*ALACAKAPI	16750,43785	K.HÜSEYİNBEY	
*ALİŞAR	24231,7233	K.MUSTAFA PAŞA	
ASLANBEY		KADIÇAYIRI	
ATAKÖY		KARABAĞLAR	
B.HÜSEYİNBEY		KARAHAN (battalgazi)	
B.MUSTAFA PAŞA		KARAKÖY	
BAĞTEPE		KARATEPE	
BAHÇELİEVLER		KAVAKLIBAĞ	
BAŞHARIK		KEMERKÖPRÜ	
BATTALGAZİ		KERNEK	
BEYDAĞI		KIRÇUVAL	
BEYLERBAŞI		*KULUŞAĞI	13921,2963
BORAN		MERKEZ BEYDAĞI	
*BULGURLU 1	20490,9429	MEYDANBAŞI	
*BULGURLU 2	30688,9429	NİYAZI	
CEVHERİZADE		NURİYE	
CİRİKPINAR		*ORDUZU 1	33612,59
ÇAMURLU		*ORDUZU 2	31364,49
*ÇOLAKOĞLU	25909,44901	*ORDUZU 3	22740,66
ÇÖŞNÜK		PAŞAKÖŞKÜ	
DABAKHANE		SANCAKTAR	
DOLAMANTEPE		SARAY	
FERHADİYE		SARICIOĞLU	
FIRAT		SELÇUKLU	
FIRINCI		ŞEHİTFEVZİ	
GÖLLER		ŞIKŞIK	
GÖZTEPE		ŞİFA	
HACI ABDİ		*ŞİŞMAN	12658,25069
HACİHALİLOĞLUÇİFTLİĞİ		TANDOĞAN	
HACİYUSUFLAR		TAŞTEPE	
HALFETTİN		TOPTAŞ	
HAMİDİYE (battalgazi)		TOYGAR	
*HANIMINÇİFTLİĞİ 1	32789,06895	ÜÇBAĞLAR	
*HANIMINÇİFTLİĞİ 2	28298,98758	YAMAÇ	
HASAN VAROL		YAYGIN	
*HASIRCILAR	32100,74212	YENİ(battalgazi)	
HATUNSUYU		YENİCE	
HİDAYET		YENİHAMAM	
İSKENDER		YENİKÖY	
İSMETİYE		YILDIZTEPE	
İSTİKLAL		ZAFER	

Tablo 3.3 ve 3.4' de görüldüğü gibi Yeşilyurt ilçesine bağlı toplam 70 mahalle, Battalgazi ilçesine bağlı toplam 84 mahalle olmak üzere uygulama için 154 mahalle ele alınacaktır. Burada başında * işaretli olup koyu renkli olan mahallelerden bazıları *Varsayım 2'* de bahsedilen bölünmüş mahalleler olup, diğerleri ise cadde/sokak uzunlukları ölçümü ilgili kurum tarafından tam olarak yapılmamış mahalleler olduğundan QGIS programı ile hesaplamaları tarafımca elle yapılmış olan mahalleleri belirtmektedir. Daha sonra uygulamamız her iki ilçe için ayrı ayrı çalıştırılmıştır.

Akpınar (depo) dâhil olmak üzere her bir mahalle için QGIS programı yardımı ile belirlediğimiz orta noktaların xy koordinatları alınarak yapılan kodlama sonucu mahallelerin birbirlerine olan uzaklık matrisleri her iki ilçe içinde ayrı ayrı oluşturulmuştur. Araçlar öncelikle birbirlerine yakınlık durumları ve bir günlük ayrılan ilaçlama süresinin karşılayacağı metre (m) baz alınarak kümelenebilir. Burada bahsedilen metreler uygulamanın 2. Bölümü olan Yönsüz Çinli Postacı Problemi' ni uygulamadan önceki mahalle/mahalle gruplarının toplam cadde uzunluklarıdır. Çinli postacı probleminde bütün yollardan geçmesi zorunluluğundan dolayı Eulerian grafin elde edilmesi (tüm cadde/sokaklardan bir kez geçme zorunluluğu) ve tam bir turun çıkması için tek dereceli düğümlerin çift dereceli hale getirilmesi ile bazı yollardan iki defa geçmesi gerekecektir. Haliyle mahalle/mahalle gruplarındaki çıkacak tek dereceli düğüm sayısına göre aracın o mahalle/mahalle grupları içinde alacağı toplam mesafenin artacağı anlamındadır. Bu nedenle uygulamanın bu bölümünde kümeleme kısmında dikkate aldığımız günlük toplam mesafe şartını etkileyecektir. Uygulamamızın Genetik Algoritma ile yapılan araçların rotalama işleminde bu durum göz önünde bulundurularak 1 saatlik bir zaman dilimi ayrılmış olup araçlara 3 saatlik (19:00-22:00 arası) süreleri varmış gibi uygulama yapılmıştır. Bu şartları sağlayarak kümelenen mahalleler listeden çıkıp tüm mahalleler kümelene kadar işleme devam edilmiştir. Kümeleme işlemi bittikten sonra meydana gelen mahalle gruplarının her birinin rotaları, belirlenen başlama ve bitiş süreleri dikkate alınarak, depodan çıkıp tekrar depoya dönecek şekilde Genetik Algoritma kullanılarak rotalama işlemi için problemimiz kodlanmıştır. Kodlar Matlab R2018a programında yazılmış olup Intel Core i7, 8 işlemci, 15.6" inç özellikli bilgisayarda çalıştırılmıştır.

Buna göre programın birçok defa çalıştırılması sonucu Yeşilyurt ve Battalgazi ilçeleri için belirlenen en uygun operatörler şöyledir;

Genetik Algoritma Operatörleri:

İterasyon Sayısı: **1000**

Birey Sayısı: **200**

Çaprazlama Türü: **OX Çaprazlama**

Çaprazlama Oranı: **0.09**

Mutasyon Oranı: **0.01**

Elitizm: **Var**

Diğer Operatörler:

Aracın ilaçlama anındaki hızı: **13 km** (Kurum tarafından verilen bilgi ile 10-15 km arası olması gerekmektedir)

Depodan bir mahalleye, mahalleden depoya veya mahalleler arası geçişteki aracın hızı: **60 km**

Uygulamanın her iki ilçe için ayrı ayrı çalıştırılması sonucu çıkan optimum sonuçlar Tablo 3.5, 3.6, 3.7, 3.8, 3.9, 3.10, 3.11, 3.12, 3.13, 3.14, 3.15 ve 3.16' da her bir araç ve gün için ayrı ayrı verilmiştir.

YEŞİLYURT İlçesi İçin Rotalama Sonuçları:

Tablo 3.5. Yeşilyurt İlçesi 1. Araç İçin Çıkan Rotalama Sonuçları

<i>Günler</i>	<i>Gittiği Mahalle Grubu Sayısı</i>	<i>Mahalle İsimleri</i>	<i>Mahalle/Mahalle Gruplarının Toplam Cadde Uzunlukları (m)</i>
<i>Pazartesi</i>	1	<i>Mahalle: Bahçebaşı – Su – Yeni (yeşilyurt)</i>	35914
<i>Salı</i>	2	<i>Mahalle: Fatih - Duruldu Mahalle: Çayırköy</i>	35395
<i>Çarşamba</i>	2	<i>Mahalle: Cumhuriyet Mahalle: Cevatpaşa</i>	32500
<i>Perşembe</i>	1	<i>Mahalle: Koşu - Seyran - Gazi - Atatürk</i>	33843
<i>Cuma</i>	1	<i>Mahalle: Hoca Ahmet Yesevi</i>	36539
<i>Cumartesi</i>	1	<i>Mahalle: Kaynarca – Çarmuzu</i>	33935

Tablo 3.6. Yeşilyurt İlçesi 2. Araç İçin Çıkan Rotalama Sonuçları

<i>Günler</i>	<i>Gittiği Mahalle Grubu Sayısı</i>	<i>Mahalle İsimleri</i>	<i>Mahalle/Mahalle Gruplarının Toplam Cade Uzunlukları (m)</i>
Pazartesi	1	Mahalle: İnönü – Merkez Fatih	28718
Salı	2	Mahalle: Yaka2 Mahalle: Topraktepe	33913
Çarşamba	1	Mahalle: Yaka 1	26203
Perşembe	1	Mahalle: Kuşdoğan	31742
Cuma	1	Mahalle: Duranlar – Göktarla – Tohma	15012
Cumartesi	1	Mahalle: Bentbaşı - Kiltpe - Yeşilkaynak	31012

Tablo 3.7. Yeşilyurt İlçesi 3. Araç İçin Çıkan Rotalama Sonuçları

<i>Günler</i>	<i>Gittiği Mahalle Grubu Sayısı</i>	<i>Mahalle İsimleri</i>	<i>Mahalle/Mahalle Gruplarının Toplam Cade Uzunlukları (m)</i>
Pazartesi	1	Mahalle: Bindal	28912
Salı	1	Mahalle: Tecde1	27626
Çarşamba	1	Mahalle: Tecde2	32612
Perşembe	1	Mahalle: Yakınca1	29453
Cuma	1	Mahalle: Yakınca2	33486
Cumartesi	1	Mahalle: Yakınca3 - Gedik	26178

Tablo 3.8. Yeşilyurt İlçesi 4. Araç İçin Çıkan Rotalama Sonuçları

<i>Günler</i>	<i>Gittiği Mahalle Grubu Sayısı</i>	<i>Mahalle İsimleri</i>	<i>Mahalle/Mahalle Gruplarının Toplam Cade Uzunlukları (m)</i>
Pazartesi	2	Mahalle: İlyas - Melekbaba Mahalle: Salköprü	29666
Salı	1	Mahalle: Gözene	27918
Çarşamba	1	Mahalle: Cemal Gürsel – Aşağıbağlar	31666
Perşembe	1	Mahalle: Dilek 1	24317
Cuma	1	Mahalle: Dilek 2	35377
Cumartesi	1	Mahalle: Dilek 3- Mahmutlu	34259

Tablo 3.9. Yeşilyurt İlçesi 5. Araç İçin Çıkan Rotalama Sonuçları

<i>Günler</i>	<i>Gittiği Mahalle Grubu Sayısı</i>	<i>Mahalle İsimleri</i>	<i>Mahalle/Mahalle Gruplarının Toplam Cadde Uzunlukları (m)</i>
Pazartesi	1	Mahalle: Karakavak	30624
Salı	1	Mahalle: Samanlı - Çilesiz	27872
Çarşamba	1	Mahalle: Çukurdere - Koyunoğlu - Zaviye	29905
Perşembe	1	Mahalle: Topsöğüt	27732
Cuma		Mahalle: Şahnahan 1 - Tepeköy	32937
Cumartesi	1	Mahalle: Şahnahan 2	29269

Tablo 3.10. Yeşilyurt İlçesi 6. Araç İçin Çıkan Rotalama Sonuçları

<i>Günler</i>	<i>Gittiği Mahalle Grubu Sayısı</i>	<i>Mahalle İsimleri</i>	<i>Mahalle/Mahalle Gruplarının Toplam Cadde Uzunlukları (m)</i>
Pazartesi	1	Mahalle: Yavuz Selim	24342
Salı	1	Mahalle: Mullakasım -Hiroğlu	16081
Çarşamba	1	Mahalle: Çavuşoğlu	26081
Perşembe	2	Mahalle: Şeyhbayram - Abdulgaffar Mahalle:Turgut Özal	32985
Cuma	1	Mahalle: Özal	27533
Cumartesi	1	Mahalle: Suluköy - Sütlüce	25758

Tablo 3.11. Yeşilyurt İlçesi 7. Araç İçin Çıkan Rotalama Sonuçları

<i>Günler</i>	<i>Gittiği Mahalle Grubu Sayısı</i>	<i>Mahalle İsimleri</i>	<i>Mahalle/Mahalle Gruplarının Toplam Cadde Uzunlukları (m)</i>
Pazartesi	1	Mahalle: Görgü	32909
Salı	1	Mahalle: Gündüzbey	30880
Çarşamba	1	Mahalle: Özalper	37377
Perşembe	1	Mahalle: Bostanbaşı	24921
Cuma	1	Mahalle: Samanköy - Kendirli	28279
Cumartesi			

BATTALGAZİ İlçesi İçin Rotalama Sonuçları:

Tablo 3.12. Battalgazi İlçesi 1. Araç İçin Çıkan Rotalama Sonuçları

<i>Günler</i>	<i>Gittiği Mahalle Grubu Sayısı</i>	<i>Mahalle İsimleri</i>	<i>Mahalle/Mahalle Gruplarının Toplam Cadde Uzunlukları (m)</i>
Pazartesi	1	Mahalle: Üçbağlar - Zafer – Fırat	37958
Salı	1	Mahalle: Kernek – Hasan Varol – Başharık - Yamaç	31705
Çarşamba	1	Mahalle: Çöşnük	28185
Perşembe	1	Mahalle: Dabakhane – Kavaklıbağ – İsmetiye - Niyazi –Şıkşık – Şifa - B.Mustafa Paşa – Hacı Abdi	30232
Cuma	1	Mahalle: HacıHaliloğlu Çiftliği – Yeniköy – Yenice	27497
Cumartesi	1	Mahalle: Merkez Beydağı	26262

Tablo 3.13. Battalgazi İlçesi 2. Araç İçin Çıkan Rotalama Sonuçları

<i>Günler</i>	<i>Gittiği Mahalle Grubu Sayısı</i>	<i>Mahalle İsimleri</i>	<i>Mahalle/Mahalle Gruplarının Toplam Cadde Uzunlukları (m)</i>
Pazartesi	1	Mahalle: Kemer Köprü – Alişar	26103
Salı	1	Mahalle: Kuluşağı - Şişman	26578
Çarşamba	1	Mahalle: Hatunsuyu	29635
Perşembe	2	Mahalle: Meydanbaşı – Karahan Mahalle: Alacakapı	49572
Cuma	1	Mahalle: Hanımın Çiftliği1	32789
Cumartesi	1	Mahalle: Hanımın Çiftliği2	28298

Tablo 3.14. Battalgazi İlçesi 3. Araç İçin Çıkan Rotalama Sonuçları

<i>Günler</i>	<i>Gittiği Mahalle Grubu Sayısı</i>	<i>Mahalle İsimleri</i>	<i>Mahalle/Mahalle Gruplarının Toplam Cadde Uzunlukları (m)</i>
Pazartesi	2	Mahalle: Bulgurlu 1 –Göller Mahalle: Karaköy	26291
Salı	1	Mahalle: Bulgurlu 2	30688
Çarşamba	1	Mahalle: Bağtepe – Hacı Yusufklar- Karatepe	19919
Perşembe	1	Mahalle: Boran – Toygar – Kadıçayırı -	26994
Cuma	1	Mahalle: Beydağı – Fırıncı - Yaygın	23525
Cumartesi	1	Mahalle: Karabağlar – Yeni (battalgazi)	31464

Tablo 3.15. Battalgazi İlçesi 4. Araç İçin Çıkan Rotalama Sonuçları

<i>Günler</i>	<i>Gittiği Mahalle Grubu Sayısı</i>	<i>Mahalle İsimleri</i>	<i>Mahalle/Mahalle Gruplarının Toplam Cadde Uzunlukları (m)</i>
Pazartesi	2	Mahalle: İskender – Hidayet – Taştepe Mahalle: Şehit Fevzi	36893
Salı	1	Mahalle: İzzetiye - Hamidiye – Nuriye – Aslanbey - K.Mustafa Paşa – Ataköy – Paşaköşkü – Cevherizade - Ferhadiye	29885
Çarşamba	3	Mahalle:Akpınar – Kırçuval – Sancaktar – B.Hüseyinbey –K.Hüseyinbey - Saray – İstiklal - Yenihamam – Mahalle:Cirikpınar Mahalle:Selçuklu	37017
Perşembe	1	Mahalle: Toptaş - Bahçelievler - Dolamantepe	28866
Cuma	2	Mahalle: Halfettin – Sarıcioğlu – Beylerbaşı Mahalle: Göztepe	38031
Cumartesi	1	Mahalle: Çamurlu	26204

Tablo 3.16. Battalgazi İlçesi 5. Araç İçin Çıkan Rotalama Sonuçları

<i>Günler</i>	<i>Gittiği Mahalle Grubu Sayısı</i>	<i>Mahalle İsimleri</i>	<i>Mahalle/Mahalle Gruplarının Toplam Caddeler Uzunlukları (m)</i>
<i>Pazartesi</i>	1	<i>Mahalle: Hasırcılar</i>	32100
<i>Salı</i>	1	<i>Mahalle: Orduzu1</i>	33612
<i>Çarşamba</i>	1	<i>Mahalle: Orduzu2</i>	31364
<i>Perşembe</i>	2	<i>Mahalle: Orduzu3</i> <i>Mahalle: Battalgazi</i>	43407
<i>Cuma</i>	1	<i>Mahalle: Tandoğan - Yıldıztepe</i>	36658
<i>Cumartesi</i>	1	<i>Mahalle: Çolakoğlu</i>	25909

Her iki ilçe için çıkan uygulama sonucuna göre Yeşilyurt ilçesi için 7 araç, Battalgazi ilçesi için ise 5 araç ilaçlama işlemini gerçekleştirecektir. Ancak haftanın 6 günü çalışması istenen ilaçlama araçlarının Yeşilyurt ilçesinin 7. aracı haftanın sadece 5 günü çalıştığı gözlemlenmiştir. Toplamda 12 araç ile 154 adet mahalle uygulamanın bu kısmında kümelenmiş ve Genetik Algoritma ile optimum şekilde rotalanmıştır.

Hem Yeşilyurt hem de Battalgazi ilçeleri için yapılan bu kümeleme ve rotalama işlemi sonucu elde edilen bu tablolardaki bilgiler sayesinde mahallelerin içerisinde optimum gezinme için bir sonraki uygulamamızda kullanılacaktır.

3.6.2. Yönsüz Çinli Postacı Problemi Modelinin Uygulanması

3.6.1. bölüm de mahallere yapılmış olan kümeleme ve Genetik Algoritma yardımı ile optimum rotalama işlemi sonucu elde edilen sonuçlar doğrultusunda mahalleler bu bölümde ki uygulamamız için bölge olarak düşünülecektir. Bu nedenle her iki ilçe için de verilen mahalle gruplarına bölge numaraları Tablo 3.17 ve Tablo 3.18 deki gibidir.

Tablo 3.17. Yeşilyurt İlçesi için Mahalle/Mahalle Gruplarına Verilen Bölge Numaraları

Mahalle/Mahalle Grup İsimleri	Bölge Numaraları
Bahçebaşı – Su – Yeni (yeşilyurt)	Bölge 1
Fatih – Duruldu ve Çayırköy	Bölge 2
Cumhuriyet ve Cevatpaşa	Bölge 3
Koşu - Seyran – Gazi - Atatürk	Bölge 4
Hoca Ahmet Yesevi	Bölge 5
Kaynarca – Çarmuzu	Bölge 6
İnönü – Merkez Fatih	Bölge 7
Yaka 2 ve Topraktepe	Bölge 8
Yaka 1	Bölge 9
Kuşdoğan	Bölge 10
Duranlar – Göktarla – Tohma	Bölge 11
Bentbaşı – Kiltape - Yeşilkaynak	Bölge 12
Bindal	Bölge 13
Tecde1	Bölge 14
Tecde2	Bölge 15
Yakınca1	Bölge 16
Yakınca2	Bölge 17
Yakınca3 - Gedik	Bölge 18
İlyas – Melekbaba ve Salköprü	Bölge 19
Gözene	Bölge 20
Cemal Gürsel – Aşağıbağlar	Bölge 21
Dilek 1	Bölge 22
Dilek 2	Bölge 23
Dilek 3 – Mahmutlu	Bölge 24
Karakavak	Bölge 25
Samanlı – Çilesiz	Bölge 26
Çukurdere – Koyunoğlu – Zaviye	Bölge 27
Topsöğüt	Bölge 28
Şahnahan 1 – Tepeköy	Bölge 29
Şahnahan 2	Bölge 30
Yavuz Selim	Bölge 31
Mullakasım – Hıroğlu	Bölge 32
Çavuşoğlu	Bölge 33
Şeyhbayram – Abdulgaffar ve Turgut Özal	Bölge 34
Özal	Bölge 35
Suluköy – Sütlice	Bölge 36
Görgü	Bölge 37
Gündüzbey	Bölge 38
Özalper	Bölge 39
Bostanbaşı	Bölge 40
Samanköy – Kendirli	Bölge 41

Tablo 3.18. Battalgazi İlçesi için Mahalle/Mahalle Gruplarına Verilen Bölge Numaraları

Mahalle / Mahalle Grup İsimleri	Bölge Numaraları
Üçbağlar – Zafer – Fırat	Bölge 42
Kernek – Hasan Varol – Başharık - Yamaç	Bölge 43
Çöşnük	Bölge 44
Dabakhane – Kavaklıbağ – İsmetiye - Niyazi –Şıkşık – Şifa - B.Mustafa Paşa – Hacı Abdi	Bölge 45
HacıHaliloğlu Çiftliği – Yeniköy – Yenice	Bölge 46
Merkez Beydağı	Bölge 47
Kemer Köprü – Alişar	Bölge 48
Kuluşağı - Şişman	Bölge 49
Hatunsuyu	Bölge 50
Meydanbaşı – Karahan ve Alacakapı	Bölge 51
Hanımın Çiftliği 1	Bölge 52
Hanımın Çiftliği 2	Bölge 53
Bulgurlu 1 – Göller ve Karaköy	Bölge 54
Bulgurlu 2	Bölge 55
Bağtepe – Hacı Yusufklar- Karatepe	Bölge 56
Boran – Toygar – Kadiçayırı	Bölge 57
Beydağı – Fırıncı - Yaygın	Bölge 58
Karabağlar – Yeni (battalgazi)	Bölge 59
İskender – Hidayet – Taştepe ve Şehit Fevzi	Bölge 60
İzzetiye - Hamidiye – Nuriye – Aslanbey - K.Mustafa Paşa – Ataköy – Paşaköşkü – Cevherizade - Ferhadiye	Bölge 61
Akpınar – Kırçoval – Sancaktar – B.Hüseyinbey – K.Hüseyinbey - Saray – İstiklal - Yenihamam ve Cirikpınar ve Selçuklu	Bölge 62
Toptaş - Bahçelievler - Dolamantepe	Bölge 63
Halfettin – Sarıcıoğlu – Beylerbaşı ve Göztepe	Bölge 64
Çamurlu	Bölge 65
Hasırcılar	Bölge 66
Orduzu 1	Bölge 67
Orduzu 2	Bölge 68
Orduzu 3 ve Battalgazi	Bölge 69
Tandoğan - Yıldıztepe	Bölge 70
Çolakoğlu	Bölge 71

Çinli postacı problemimizin özellikle tek düğümlerin çift olma durumundaki minimum eşleşmenin elde edilmesi için Yapay Arı Kolonisi Algoritması uygulanmış eşleşme durumunda meydana gelecek ihtimaller göz önünde bulunarak algoritmaya yapılan modifiye işlemi başlık 2.2.4’ de anlatılmıştır. Uygulamamızda kullanılan YAKA için parametreler aşağıda ki gibi olup her bir bölge için beşer defa çalıştırılmış olup ortalamaları yazılmıştır.

Birbirleri ile direkt bağlantısı olmayan düğümler arası en kısa yol hesaplama da ise Matlab programının “shortestpath” fonksiyonu içinde bulunan Dijkstra algoritması kullanılmıştır(<https://www.mathworks.com/help/matlab/ref/graph.shortestpath.html#buofmw7-1-Method>, Erişim Tarihi: 24/09/2018).

YAKA algoritması ile tek dereceli düğümler arası minimum eşleşme sağlanarak elde ettiğimiz Eulerian graflara, Fleury algoritması ile Euler tur oluşumu sağlanmıştır.

YAKA için belirlenen parametreler:

Bağlangıç Düzüm = “**ID no**” (her bölge için depoya en yakın düğümün ID’ si yazılır)

Populasyon = **500**

İterasyon=**1000**

Kâşif Sayısı= **4**

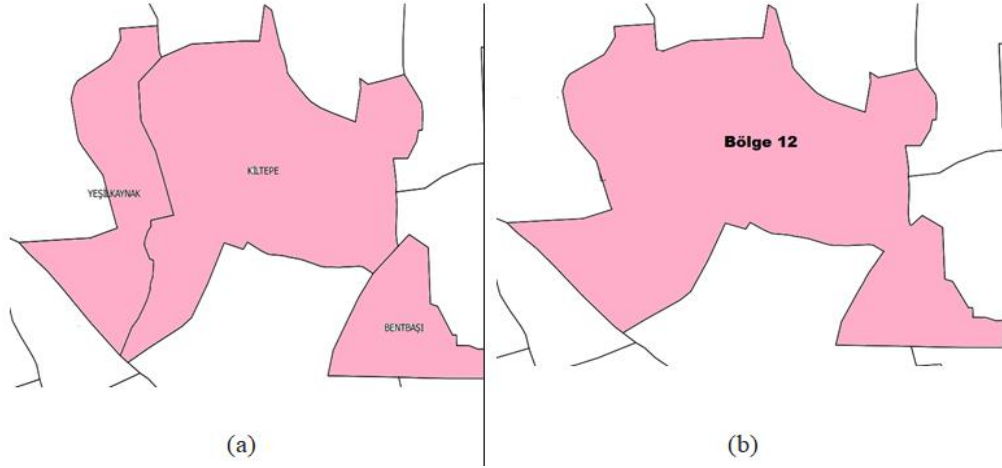
Deneme Limiti=**50**

Uygunluk Değerinin Hesaplanması: **Rulet Tekerleği Seçimi**

Yönsüz Çinli Postacı problemimizin uygulama adımıında mahallelerin dağılımı ve rotalanmasına göre iki olası durum söz konusudur. Bunlardan ilk olası durum mahalle grupları oluşumunda bu mahallelerin sınır çizgilerinin birbirlerine komşu olması, ikinci olası durum ise mahallerin sınır çizgilerinin birbirlerine komşu olmama durumudur. Bu her iki durumda aşağıda ki şekilde birer bölge için görsellerle ve elde edilen sonuçlarıyla ve EK-2 deki rotalarıyla paylaşılmıştır. Diğer bütün bölgeler için de aynı şekilde uygulama gerçekleştirilmiş olup çıkan sonuçlar Tablo 3.21, 3.22, 3.23, 3.24, 3.25, 3.26, 3.27, 3.28 3.29, 3.30, 3.31, 3.32’ de paylaşılmıştır.

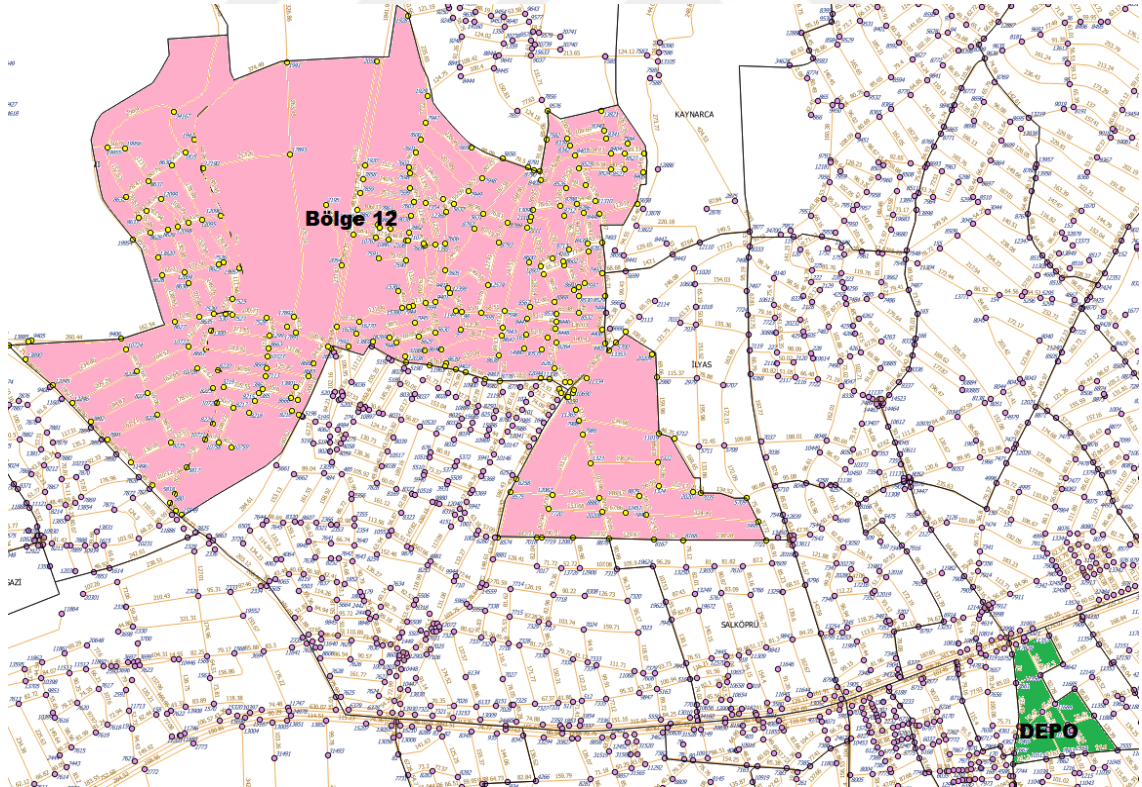
1. Durum: Mahalle Gruplarının Sınır Çizgileri Komşu Olması Hali

Şekil 3.21 (a) da görüldüğü gibi Bölge 12’ye ait mahalle grubumuz Bentbaşı- Kıltepe- Yeşilkaynaktır. Bu üç mahalle birbirine sınır komşusu olup artık Şekil 3.21 (b) de görüldüğü gibi Bölge 12 olarak adlandırılıp artık tek bir gezinilecek alanmış gibi düşünülerek Yeşilyurt ilçesine ait 2. araç tarafından cumartesi günü gezilecektir.



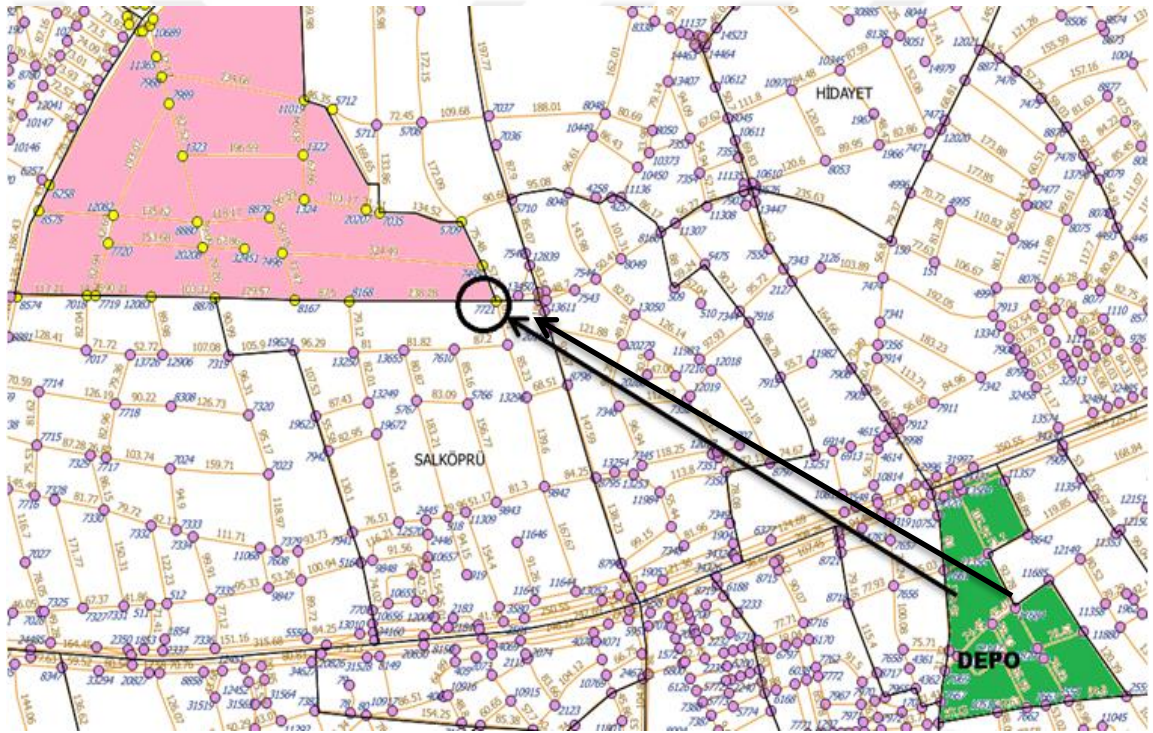
Şekil 3.21. (a) Bentbaşı –Kıltepe-Yeşilkaynak Mahalle Grubu (b) Bentbaşı –Kıltepe-Yeşilkaynak Mahalle Grubunun Bölge Hali

Araç Şekil 3.22’ de görüldüğü gibi Bölge 12 için Akpınar deposuna en yakın olan düğüm seçilerek uygulamamız çalıştırılacak ve bölge içinde ki optimum gezinme o düğümde başlayıp yine o düğümde bitecektir.



Şekil 3.22. Bölge 12 ile Deponun Haritada Gösterimi

Aracın ilaçlama haznesinin su ve ilaç ile dolu olmasından dolayı daha fazla yakıt harcayacağı düşüncesiyle depodan çıkan aracın gezinmesi gereken bölgeye kendisine en yakın noktadan giriş yapması gerektiği düşünülmüştür. Şekil 3.23’ de görüldüğü gibi Bölge 12 için depoya en yakın düğüm 7721 nolu düğümdür. Bu nedenle uygulama başlatılmadan önce gerekli parametrelerin yazıldığı kısımda “Başlangıç Düğüm = 7721” yazılacaktır. Böylece depodan çıkan araç gezinme işlemine 7721 nolu düğümden başlayıp 7721 nolu düğümden de tamamlayacaktır. Daha sonra depoya dönüş işlemi gerçekleşecektir. Rotaya depoya dönüş eklenmemiş olup 7721 nolu düğümden depoya olan uzaklık toplamı sonuç tablolarımızda “Depodan Gidiş - Dönüş Uzaklıkları (m)” kısmına yazılmıştır.



Şekil 3.23. Bölge 12'nin Depoya En Yakın Düğümünün Gösterimi

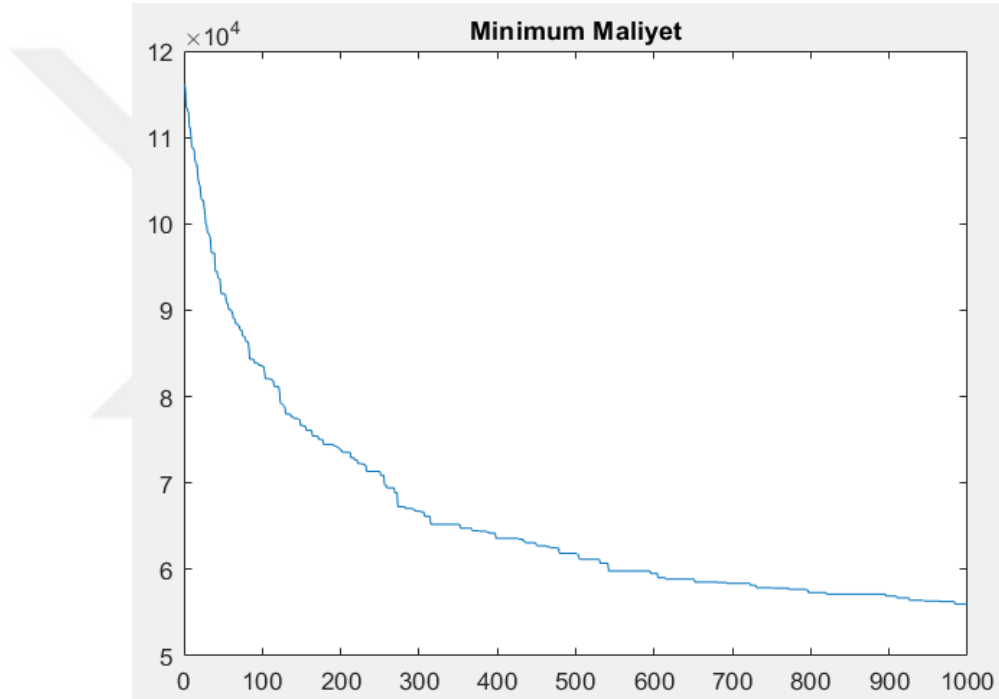
Bölge 12 için belirlenen YAKA parametrelerinin beşer defa çalıştırılması ile ortalaması alınan sonuçlar Tablo 3.19’ da verilmiştir.

Tablo 3.19. Bölge 12 için Çinli Postacı Algoritması Çözümünün Sonucu

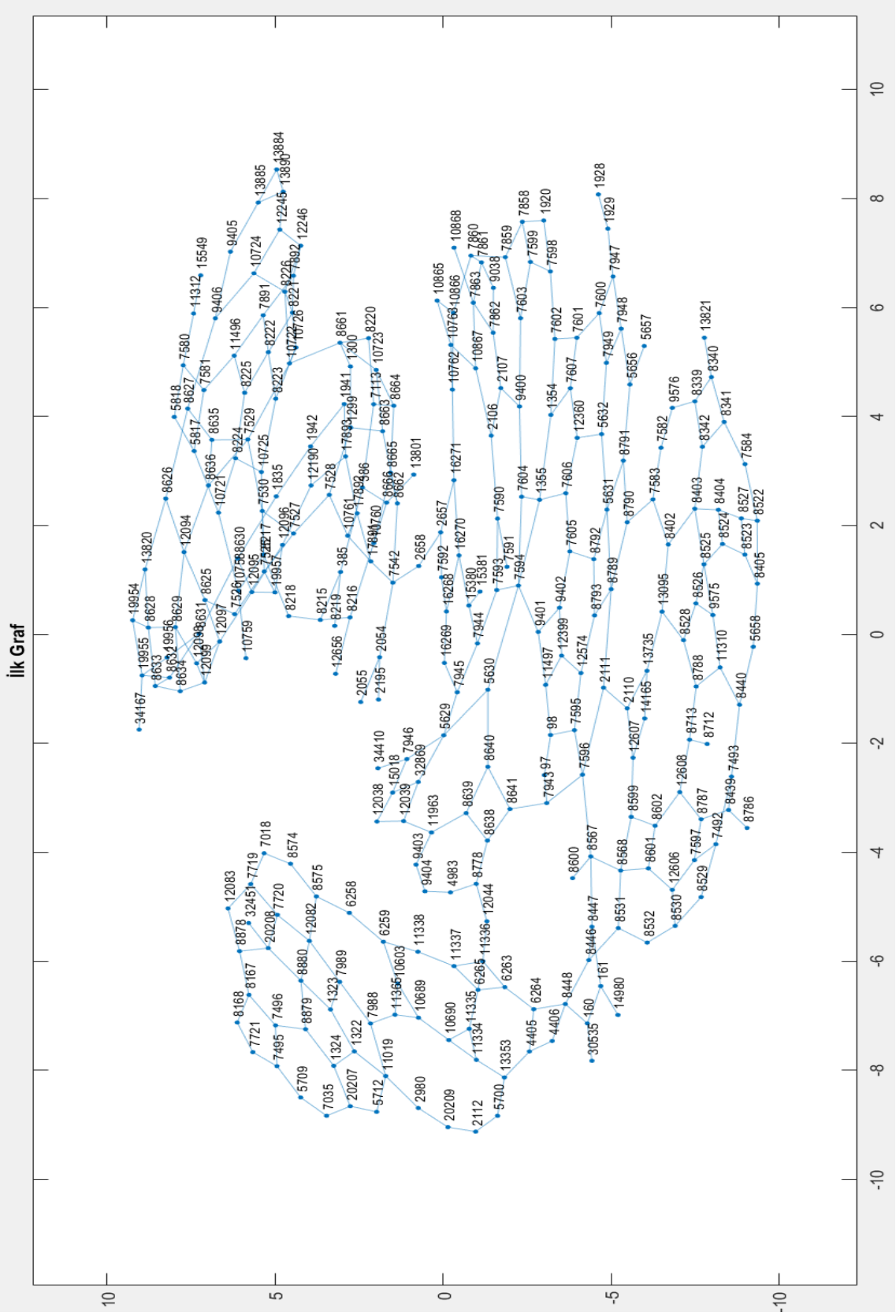
<i>Bölgeler</i>	<i>Grafın Düğüm (d) ve Ayrıt (a) Sayıları</i>	<i>Toplam Tek Dereceli Düğüm Sayısı</i>	<i>YAKA ile Tek Dereceli Dğümler Arası Eşleşmenin Global Minimum Uzunluğu (m)</i>	<i>İlk Grafın Uzunluğu (m)</i>	<i>Toplam Uzunluk (m)</i>	<i>Rota için Seçilen Başlangıç ID</i>	<i>Depodan Gidiş-Dönüş Toplam Uzaklığı (m)</i>	<i>Aracın Depoya Dönüş Saati (civarı)</i>
Bölge 12:								
Bentbaşı-Kiltepe-Yeşilkaynak	285 d 401 a	206	23458	32530	55890	7721	3400	22:20
			30 km hızda	13 km hızda			60 km hızda	

Bölge 12 için cadde/sokaklar arası mesafe matrisimizde 285 adet düğüm, 401 adet ayrıt (cadde/sokak) vardır. Bu 285 düğümün 206 tanesi tek derecelidir. Bu hali ile grafımızın eşleşme olmadan önceki uzunluğu 32530m dir. Bu 206 adet tek dereceli düğümler arası minimum eşleşme uzunluğu 23458 metredir. Minimum eşleşme sonucu bu uzunluğun ilk graf uzunluğuna dâhil edilerek ($32530+23458=55890$) aracın bu alan için alacağı toplam uzunluk 58390m olmaktadır. Depodan 7721 nolu düğüme olan uzaklık 1700 m olup, birde dönüş uzaklığının (1700 m) ilave edilmesi ile depodan gidiş dönüş toplam uzaklık 3400 m dir. Araç mahalle içerisinde gezinirken ilaçlama hızı 13 km olarak belirlenmiş bunun sebebi 3.6.1 deki uygulamada açıklanmıştır. Tek dereceli düğümlerin eşleşmesi sonucu aynı ayrıtın birden fazla gezinmesinden dolayı aracın bu caddeleri/sokakları tekrar ilaçlaması gerekmediğinden ilaç püskürtme düğmesini kapattığı için de hızının artırılabilceği düşüncesi ile o sırada ki hızı da ortalama 30 km olarak tarafımda belirlenmiştir. Yani araç, ilk graf uzunluğu olan 32530m lik mesafeyi 13km hız ile bunun dışında kalan yani bir caddeden birden çok geçmesi durumu ile oluşan minimum uzunluk olan 23458m lik mesafeyi de 30km hızla geçeceği düşünülmüştür. Çünkü araç ilaçlama esnasında 13km hızla giderken birden daha önce ilaçladığı düşünülen diğer bir caddeye hemen geçtiğinde birden hız artışının ortalama 30 km olacağı düşünülmüştür. Aracın depoya gidiş - dönüş hızı ise 60km olarak belirlenmiştir. Tüm bu hız ve mesafeler göz önüne alındığında 19:00' da depodan çıkan aracın depoya dönüş saati (trafikteki yoğunluk ve aksaklıklar göz önüne alınmamıştır) 22:20 civarı olarak belirlenmiştir.

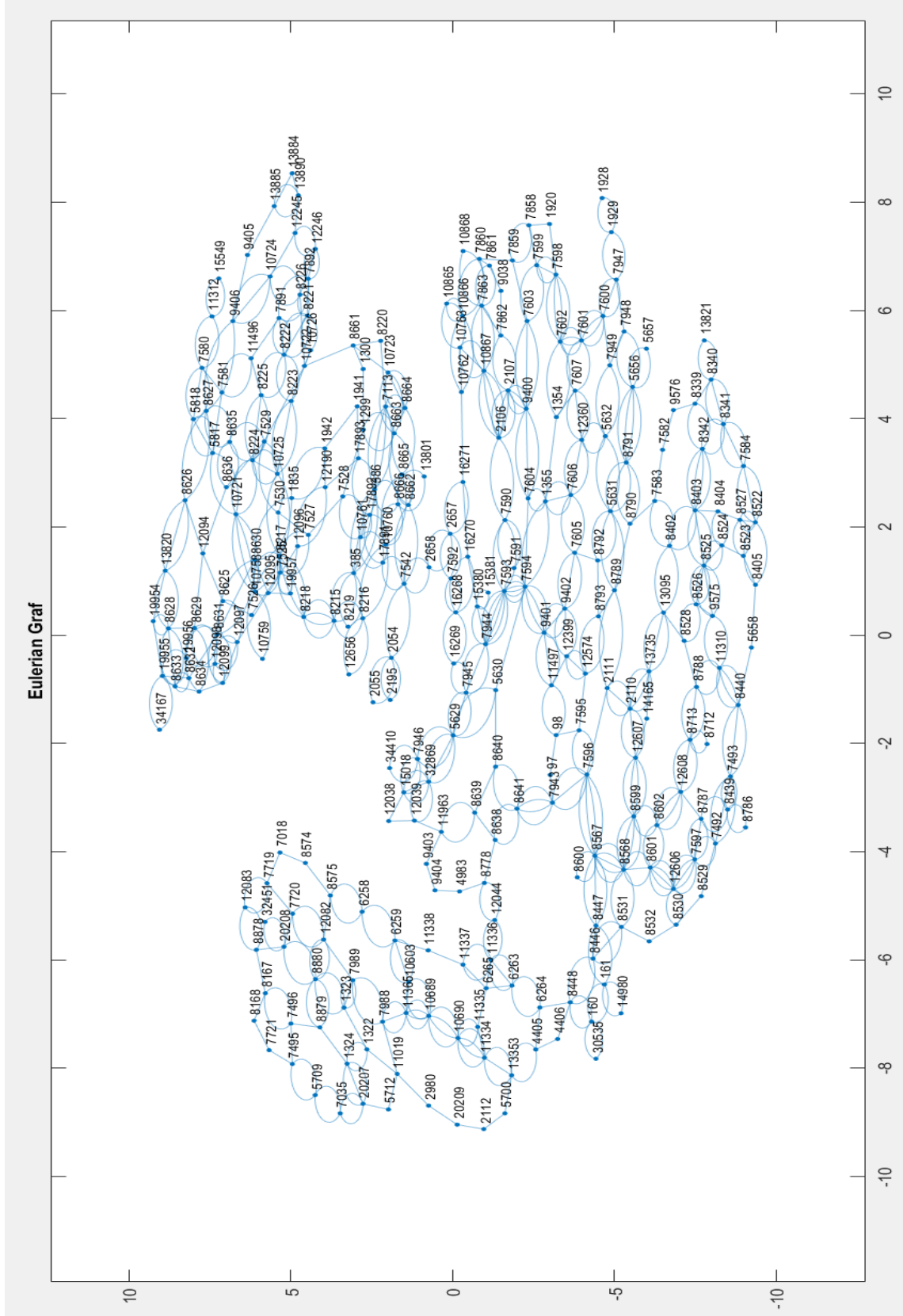
Aracın ilaçlama yapacağı alanın 32530m' yi 13 km hız ile 2.50 saatte bitirebilecektir. Yani araç bu bölgeyi tek depo ile ilaçlamayı bitirebileceğinden ilaçlama haznesini yeniden doldurması gerekmemektedir. Bu sebeple ilaçlama haznesinin ikinci defa doldurulması için uygulama 3.6.1'de bahsedilen ekstra 15 dk lık bir süre bu ve buna benzer bölgeler için toplam saate eklenmemiştir (Aracın ilaçlama haznesinin 2-2.50 saat yeteceği düşüncesi ile eğer bu aralıktan daha fazla sürede bir bölgeyi ilaçlaması gerekiyorsa sonuç tablolarına ekstra bir 15 dk lık süre ilave edilmiştir). Şekil 3.24, 3.25, 3.26 ve 3.27' de Bölge 12 için çıkan uygulama görselleri paylaşılmış olup EK-2 de ise bölgenin optimum rotası paylaşılmıştır.



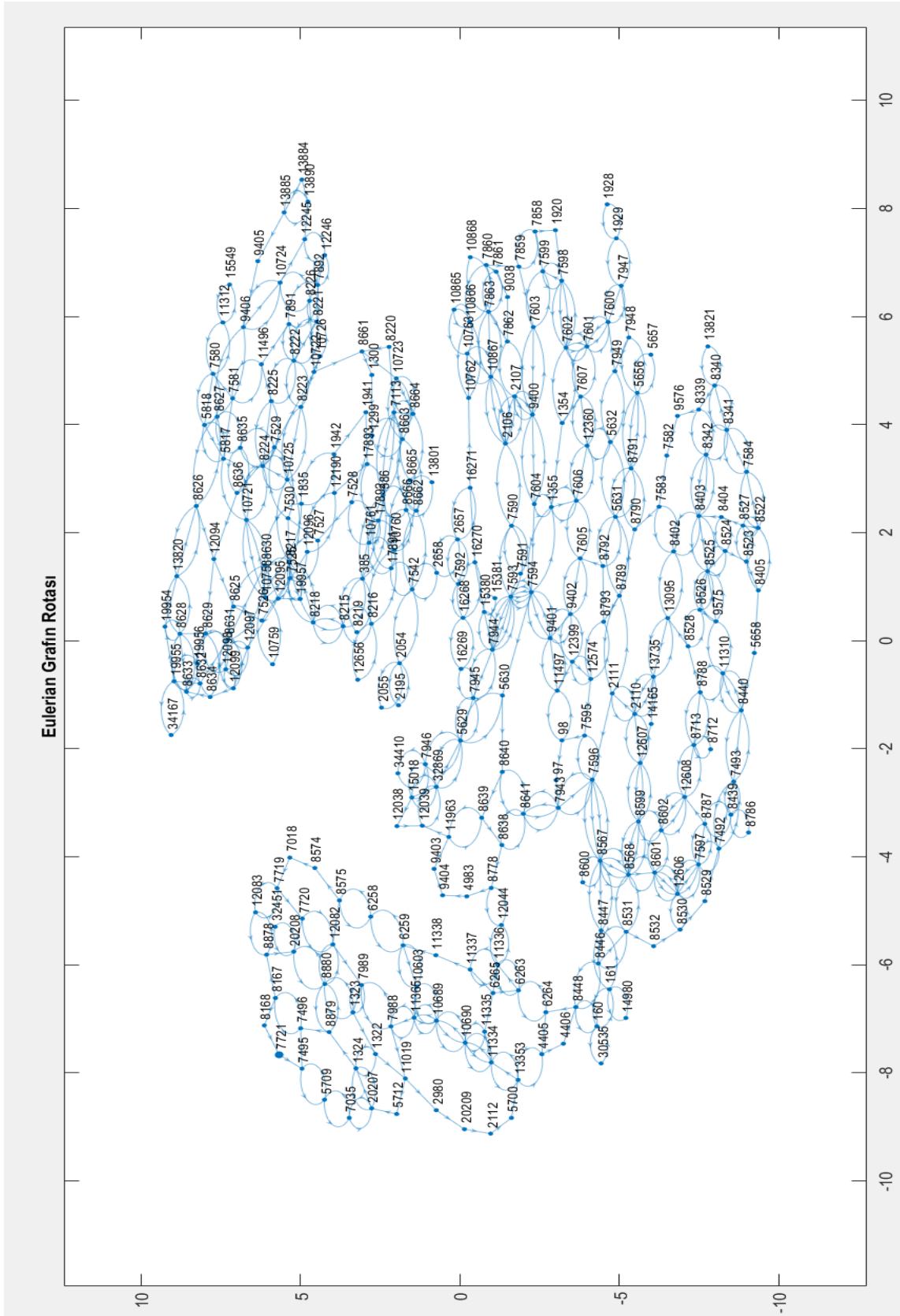
Şekil 3.24. Bölge 12 İçin Minimum Maliyet Grafiği



Şekil 3.25. Bölge 12 İçin İlk Graf Hali



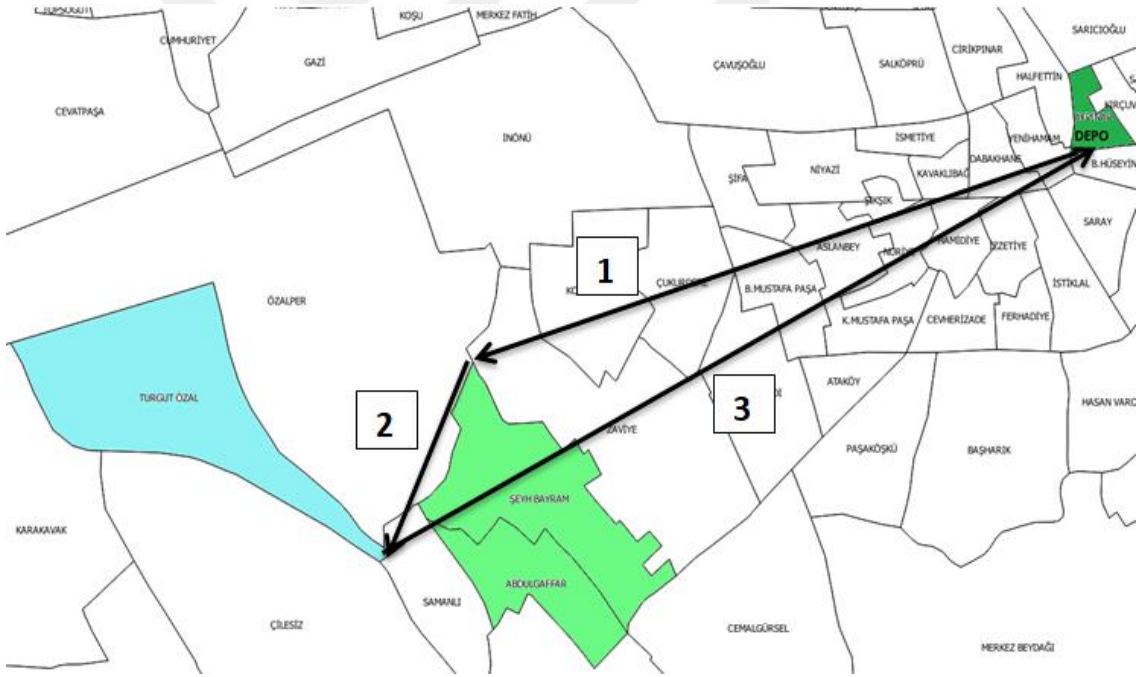
Şekil 3.26. Bölge 12 için Tek Dereceli Döğümmler Arası Minimum Eşleşme İle Çıkan Eulerian Graf



Şekil 3.27. Bölge 12 için Tek Dereceli Düğüm Arası Minimum Eşleşme İle Çıkan Eulerian Graf ın Rotası

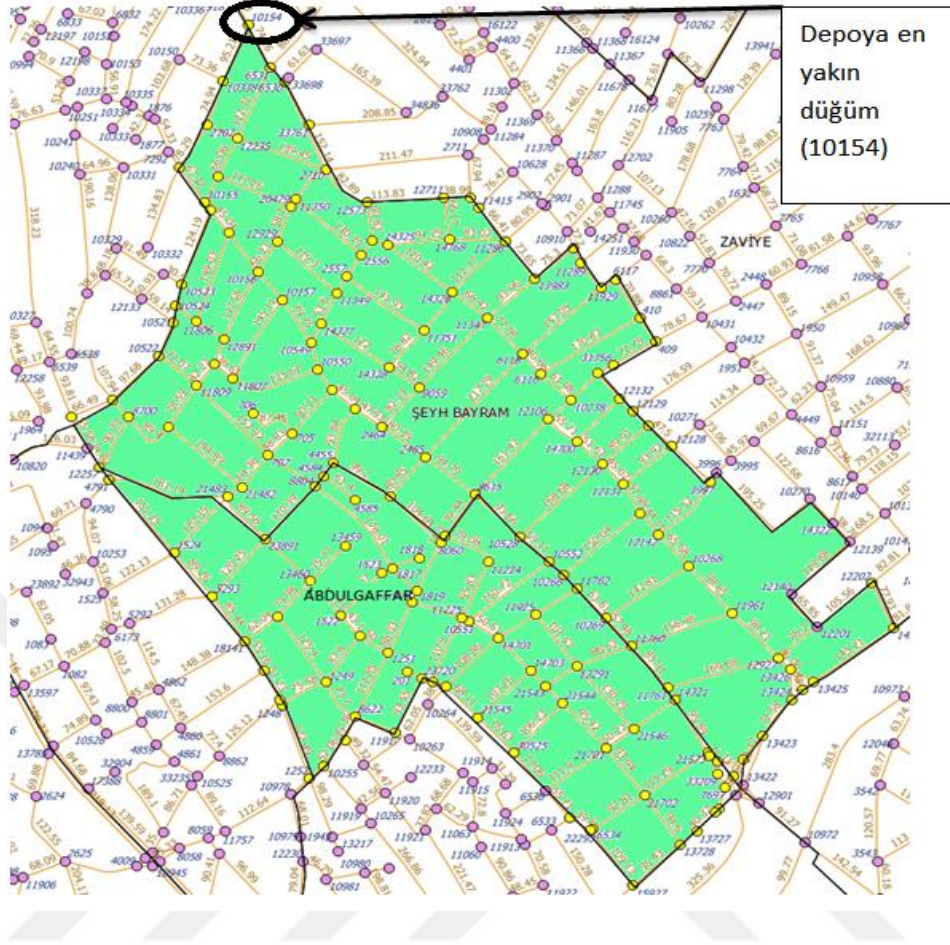
2. Durum: Mahalle Gruplarının Sınır Çizgileri Komşu Olmaması Hali

Bu durum için örneklendirerek anlatacağımız alan Bölge 34 olarak seçilmiştir. Bölge 34: Şeyhbayram–Abdulgaffar ve Turgut Özal mahallelerinden oluşmaktadır. Şekil 3.28’ de görüldüğü gibi Şeyhbayram–Abdulgaffar mahalleleri birbirlerine komşu iken Turgut Özal ise Şeyhbayram–Abdulgaffar’ a 2400m lik bir mesafe uzaklıklardır. Bu nedenle araç depoya yakın olan mahalle grubu Şeyhbayram–Abdulgaffar’ın en yakın düğümünden başlayacak ve oradaki ilaçlama işlemini bitirdikten sonra Bölge 34’ e ait olan ilaçlaması gereken bir diğer mahalle Turgut Özal’ a geçecektir. Bu geçiş sırasında depodan çıkan araç Şeyhbayram-Abdulgaffar alanına giriş yaptığı düğümde rotayı tamamladığı için bu düğümde Turgut Özal mahallesine en yakın düğüm belirlenerek giriş yapacaktır. Daha sonra Turgut Özal mahallesinde ki turunu da başladığı düğümde sonlandıran araç buradan depoya dönüş yapacaktır.



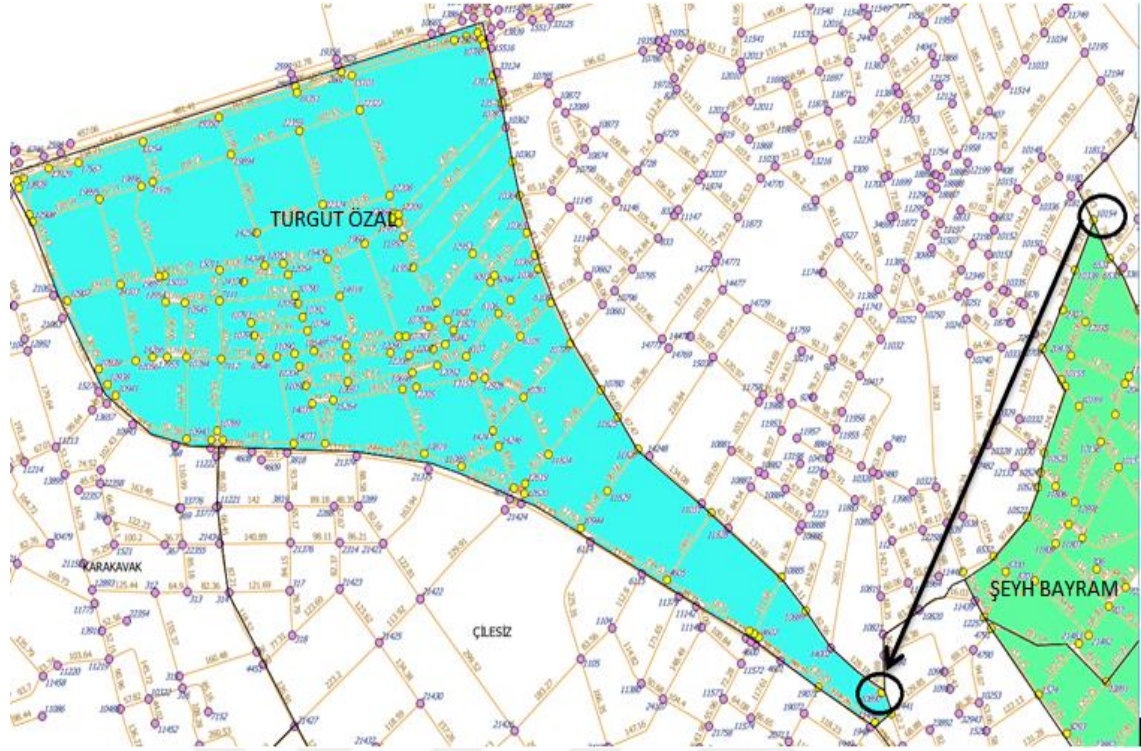
Şekil 3.28. Bölge 34'ün Haritada Gösterimi ve Aracın Gezinme Sırası

Bölge 34 için için depoya en yakın Şeyhbayram - Abdulgaffar olup önce burayı ilaçladıktan sonra araç Turgut Özal mahallesine geçecektir. Şeyhbayram - Abdulgaffar için depoya en yakın ilaçlama rotası başlangıç düğümü Şekil 3.29’ da görüldüğü gibi 10154 numaralı düğümdür.



Şekil 3.29. Şeyhbayram – Abdulgaffar Alanının Depoya En Yakın Düğümünün Gösterimi

Şeyhbayram - Abdulgaffar dan sonra Turgut Özal mahallesine giriş için en yakın düğüm Şekil 3.30’ da görüldüğü gibi ise 10890 olup ilaçlama işlemi bu düğümde başlayıp tekrar bu düğümde son bulacaktır. Daha sonra 10890 düğümünden de en yakın güzergâhtan araç depoya dönecektir.



Şekil 3.30. Şeyhbayram – Abdulgaffar Alanından Turgut Özal’ a En Yakın Geçiş Düzümlerinin Gösterimi

Bölge 34 için belirlenen YAKA parametrelerinin beşer defa çalıştırılması ile ortalaması alınan sonuçlar Tablo 3.20’ de verilmiştir.

Tablo 3.20. Bölge 34 için Çinli Postacı Algoritması Çözümünün Sonucu

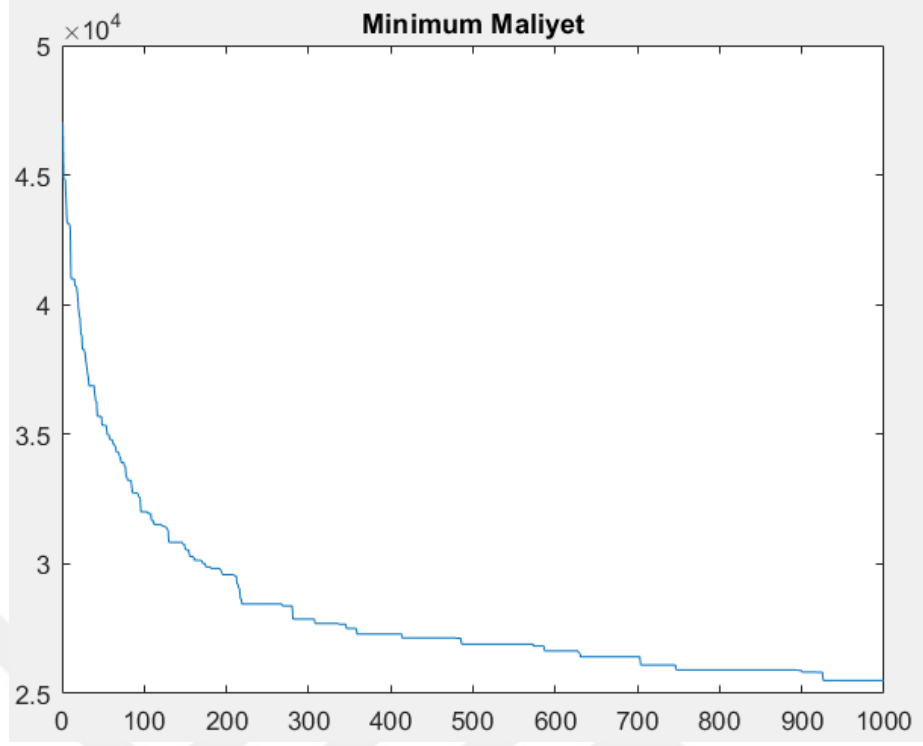
Bölgeler	Grafın Düzüm (d) ve Ayrıntı Sayıları (a)	Toplam Tek Dereceli Düzüm Sayısı	YAKA ile Tek Dereceli Düzümler Arası Eşleşmenin Global Minimum Uzunluğu (m)	İlk Grafın Uzunluğu (m)	Toplam Uzunluk (m)	Rota için Seçilen Başlangıç ID	Depodan Gidiş-Dönüş Uzaklıkları (m)	Aracın Depoya Dönüş Saati (civarı)
Bölge 34:								
Şeyhbayram-Abdulgaffar	169 d 240 a	110	7826	17680	25507	10154	4900	22:50
Turgut Özal	136 d 192 a	96	5713	15373	21087	10890	5000	
			30 km hızda	13 km hızda			60 km hızda	

Depodan 60km hızla çıkan araç öncelikle aracın ilaçlama haznesinin dolu olmasından dolayı daha az yakıt sarf edeceği düşüncesiyle kendisine en yakın düğüm olan 10154 nolu düğümden Şeyhbayram -Abdulgaffar alanına girecektir. Bu düğümün depoya uzaklığı 4900m dir. Şeyhbayram -Abdulgaffar alanı 169 düğüm, 240 ayrıttan (cadde/sokak) oluşmaktadır. Bu hali ile toplam cadde uzunluğu 17680m dir. 169 düğümün 110 tanesi de tek dereceye sahip olup bu tek dereceli düğümler arası minimum eşleşme 7930m dir. Minimum eşleşme sonu da ilk graf uzunluğuna dâhil edilerek ($17680+7826=25507$) aracın bu alan için gideceği toplam uzunluk 25507m olmaktadır. Araç 17680m' yi 13 km hızla, 7826m' yi 30km hızla gezinerek sonra 10154 düğümüne geri dönecektir.

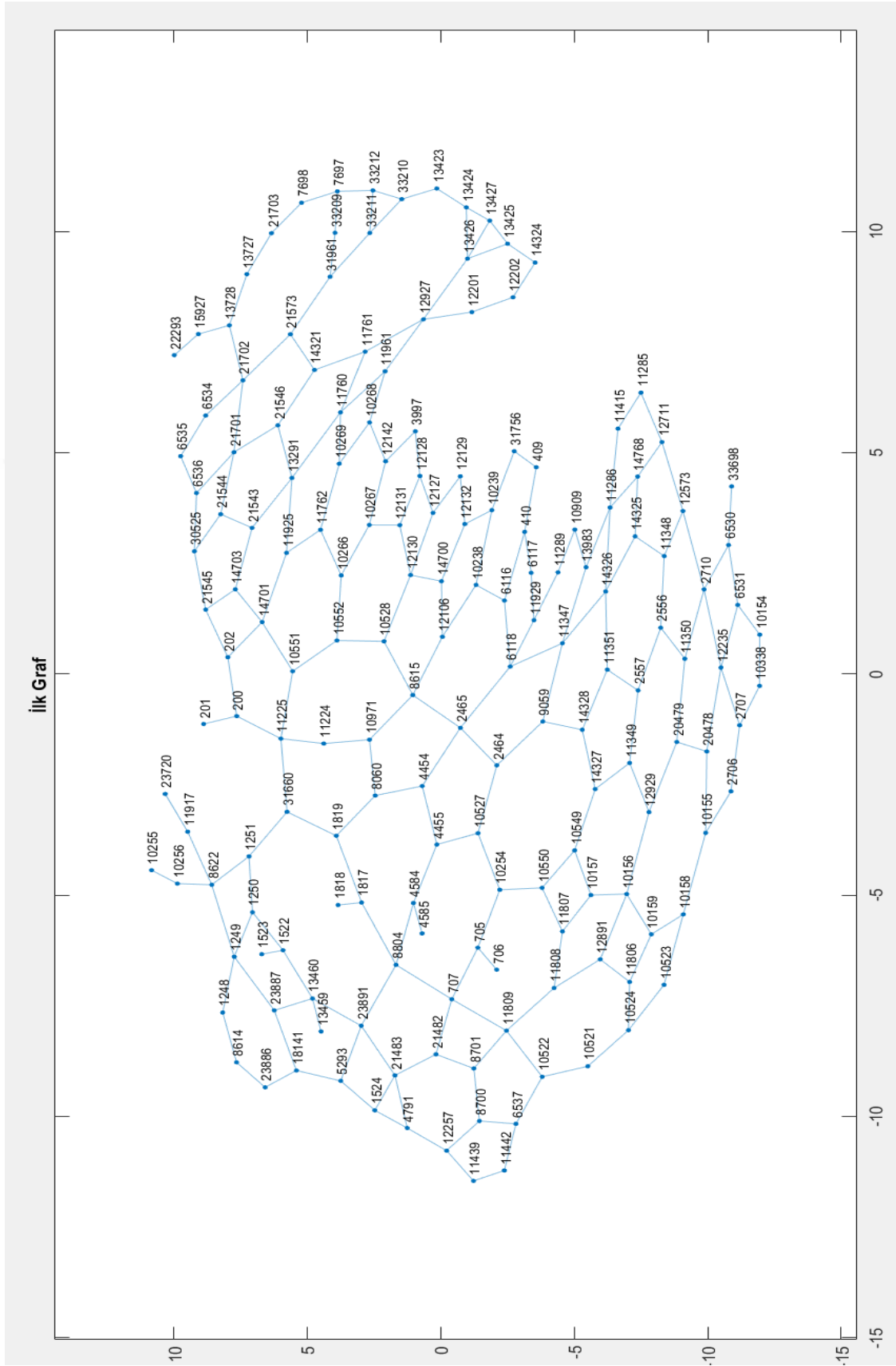
10154 düğümünde ki araç Bölge 34'e ait olan diğer mahallesi Turgut Özal' ı ilaçlamak için en yakın düğüm olan 10890 nolu düğüme tekrar 60km hızla giderek bu alana giriş yapacaktır. 10154 düğümünden 10890 düğümüne olan uzaklık 2100m dir. Turgut Özal mahallesi 136 düğüm ve 192 ayrıttan (cadde/sokak) oluşmaktadır. Bu hali ile mahallenin toplam cadde/sokak uzunluğu 15373m dir. Bu 136 düğümün 92 tanesi tek dereceli olup bu tek dereceliler arası minimum eşleşme 5713m dir. Minimum eşleşme sonucu da ilk graf uzunluğuna dâhil edilerek ($15373+5713=21087$) aracın bu alan için gideceği toplam uzunluk 21087m olmaktadır. Araç 15373m' yi 13km hızla, 5713m' yi ise 30km ile gezinerek tekrar başlangıç düğümü olan 10890 nolu düğüme dönüp burdan da depoya dönecektir. 10890 nolu düğümden depoya uzaklık ise 5000m dir.

Bölge 34' ü ilaçlamak için 19:00' da depodan çıkan araç yine tek ilaçlama haznesi (15dk lık süre dâhil) ile ilaçlayacağı bu bölge için depoya dönüş saati 22:50 olarak hesaplanmıştır.

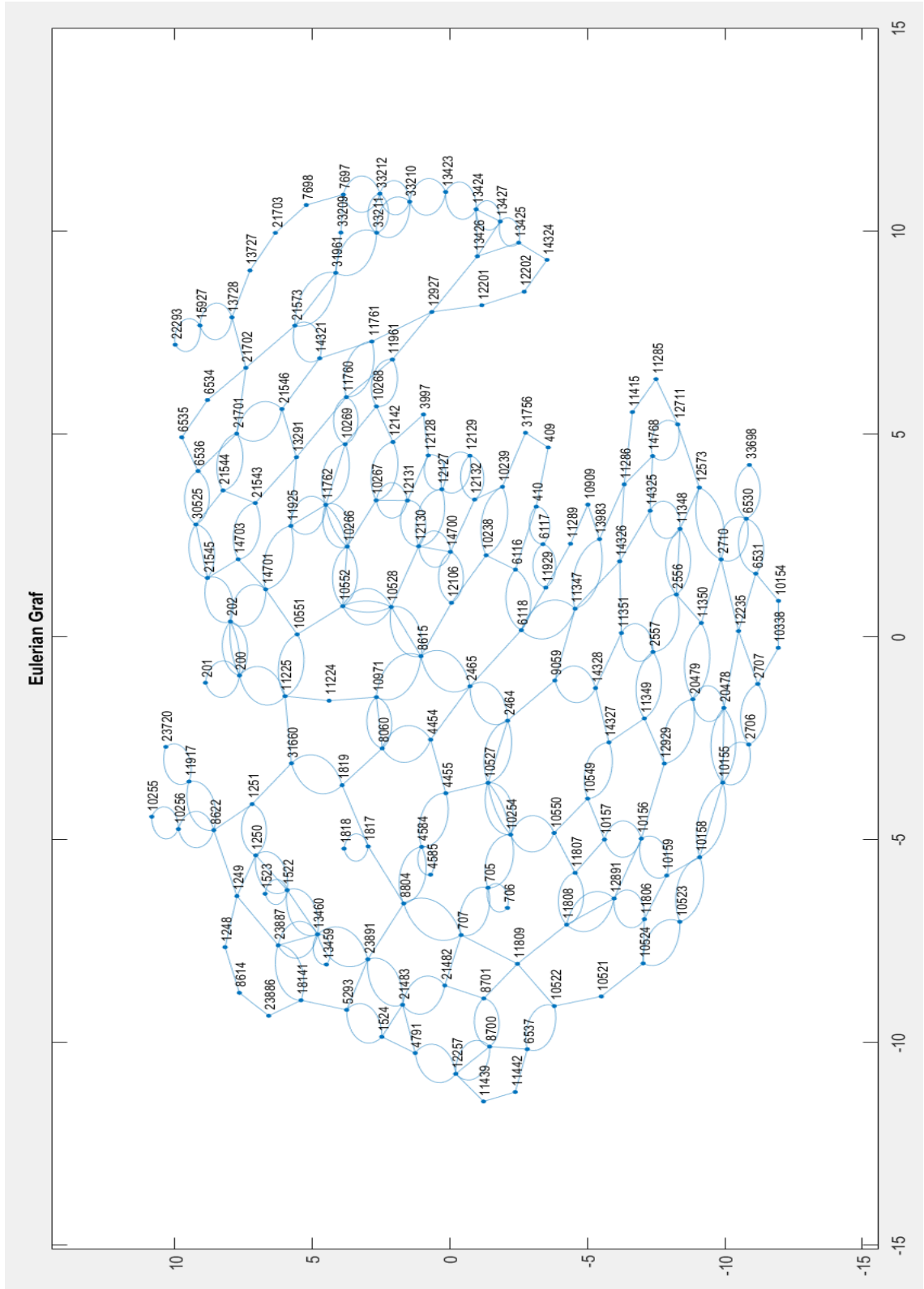
Bölge 34' e ait Şeyhbayram – Abdulgaffar alanı için uygulama da çıkan sonuçların gösterimi Şekil 3.31, 3.32, 3.33 ve 3.34' de, Turgut Özal mahallesi için uygulama da çıkan sonuçların gösterimi ise Şekil 3.35, 3.36, 3.37 ve 3.38' de paylaşılmış olup her iki alan için EK-2 de ise bölgenin optimum rotası paylaşılmıştır.



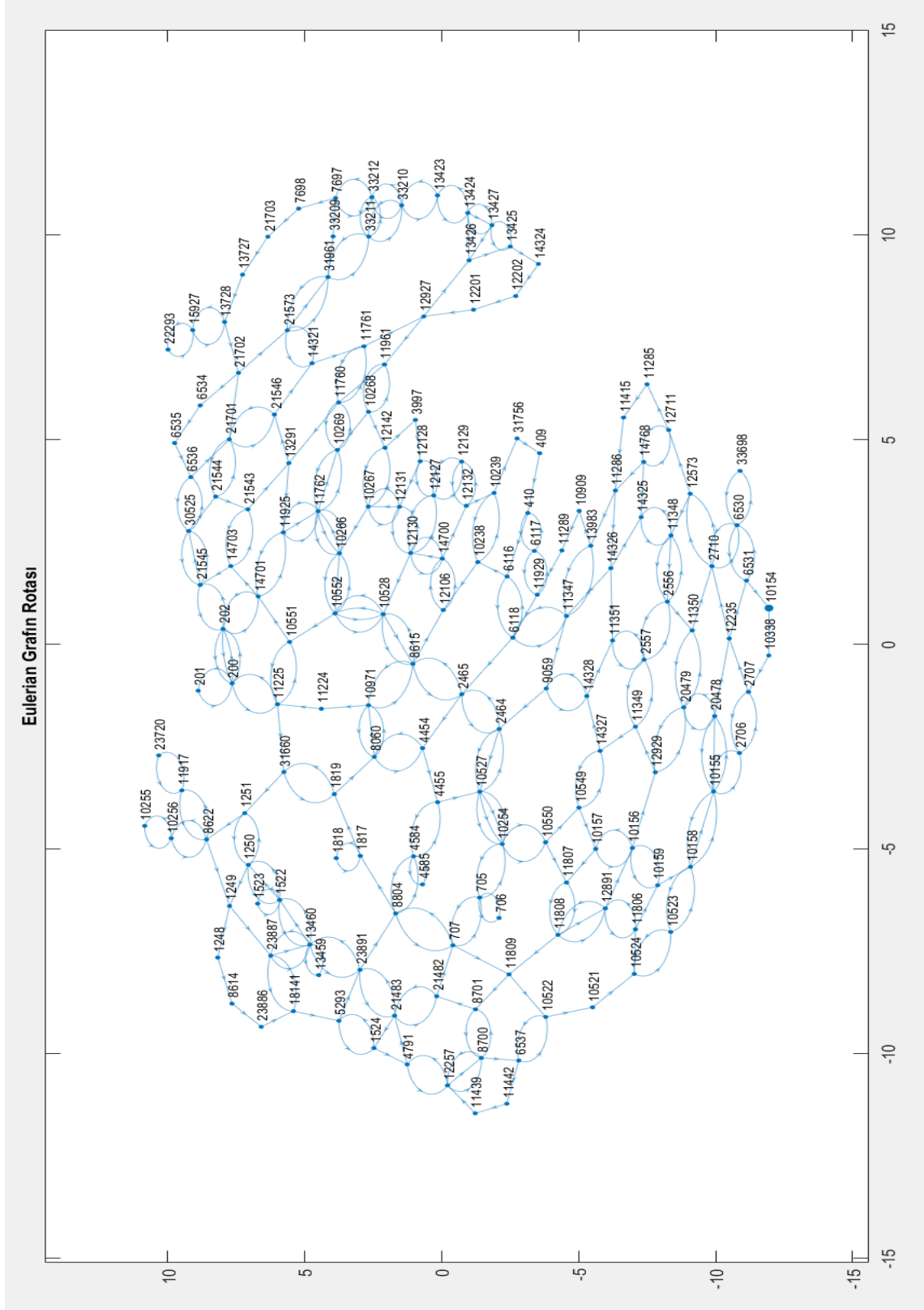
Şekil 3.31. Bölge 34 Şeyhbayram - Abdulgaffar İçin Minumum Maliyet



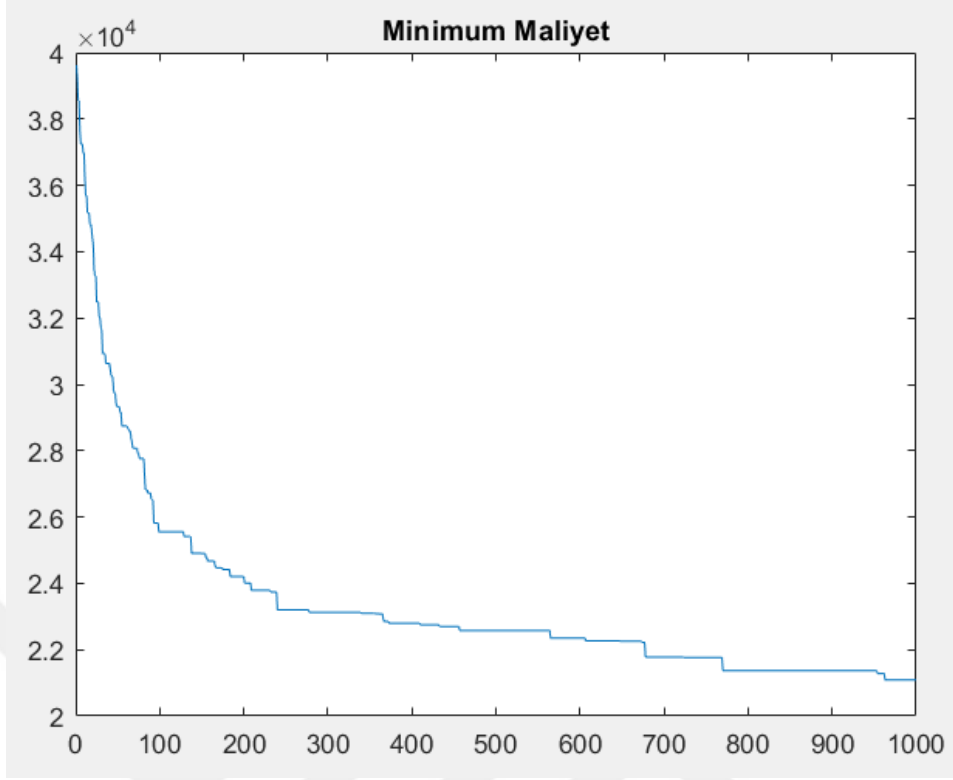
Şekil 3.32. Bölge 34 Şeyhbayram - Abdulgaffar İçin İlk Graf



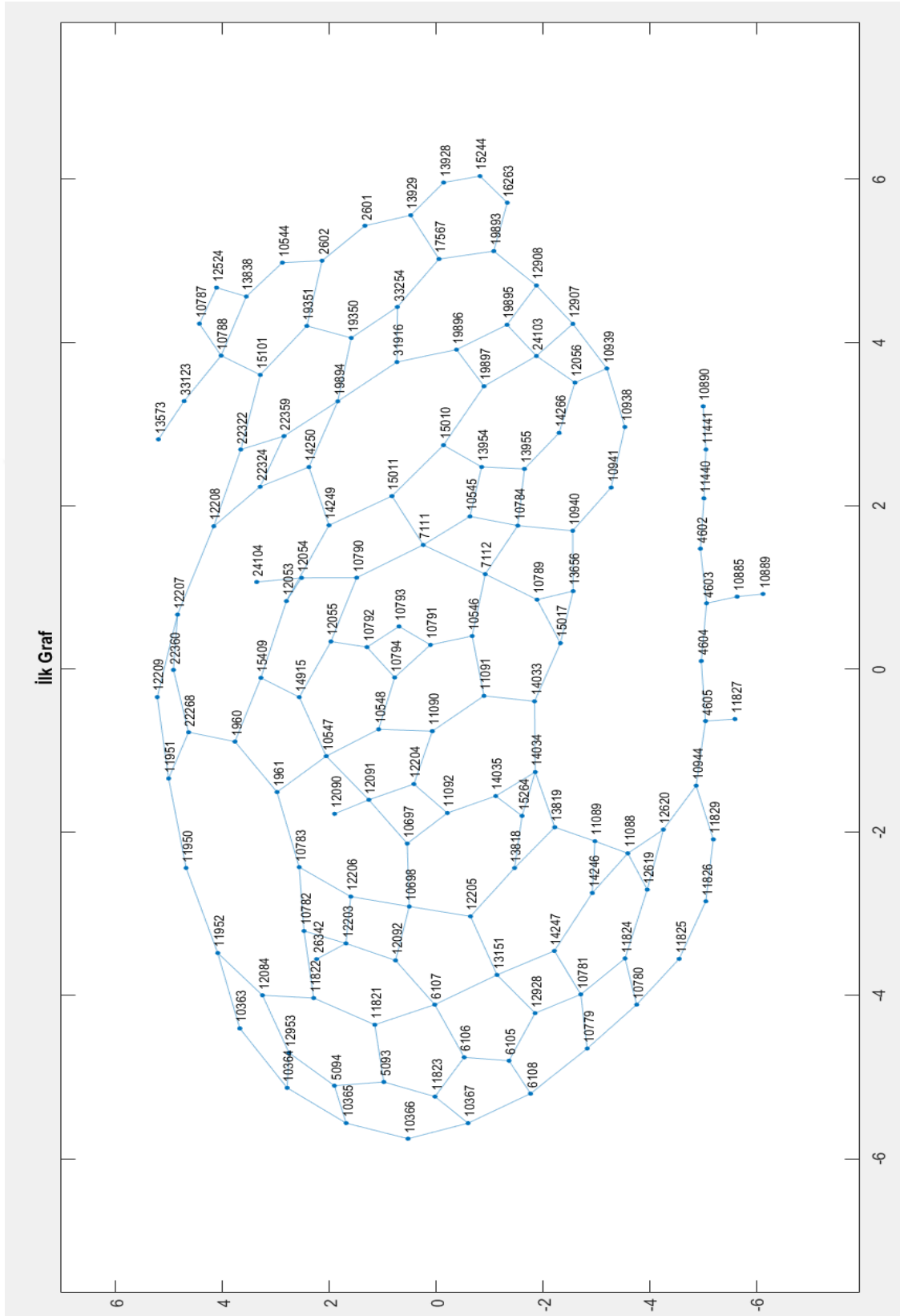
Şekil 3.33. Bölge 34 Şeyhbayram - Abdulgaffar İçin Tek Dereceli Düğümler Arası Minimum Eşleşme İle Çıkan Eulerian Graf



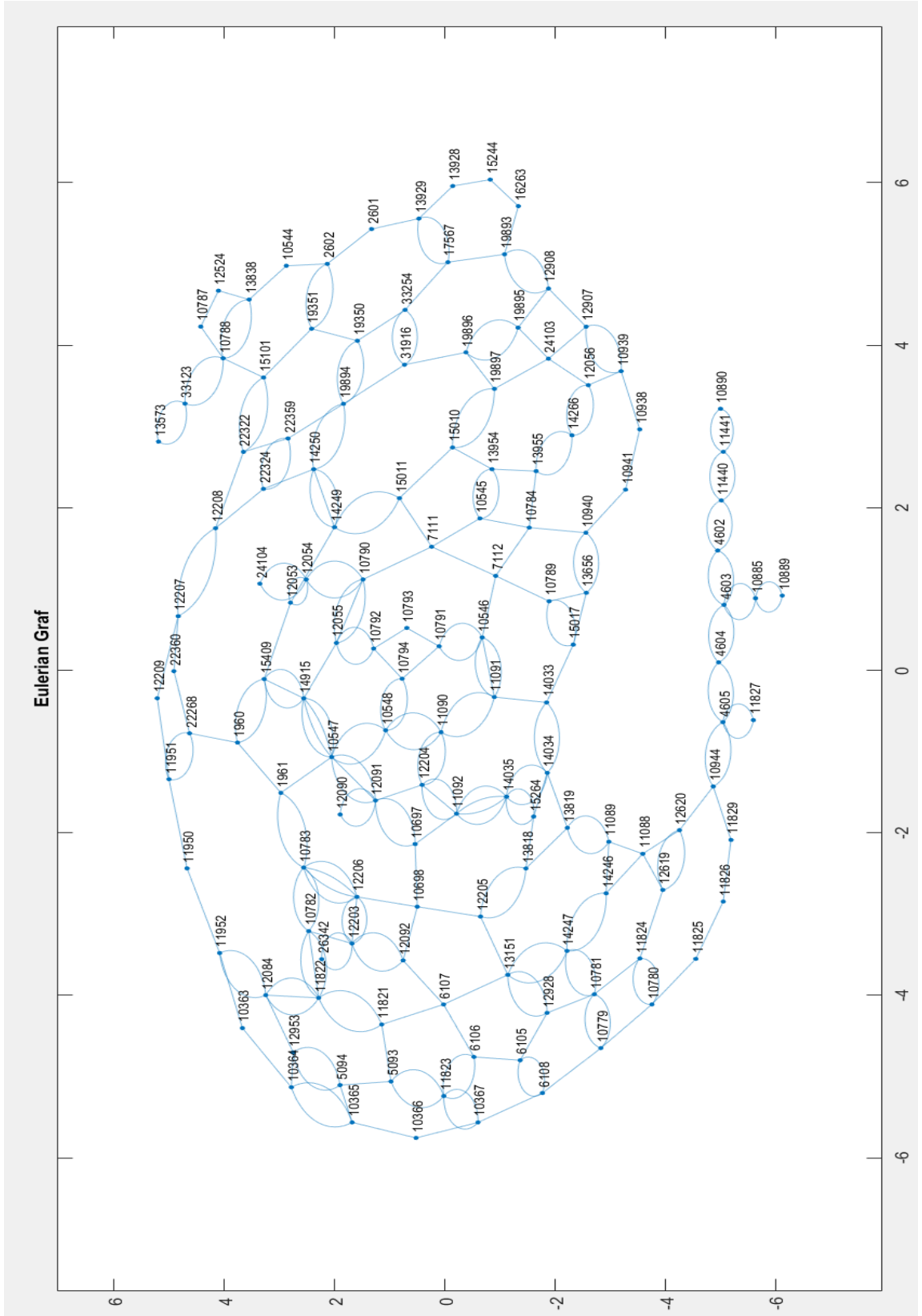
Şekil 3.34. Bölge 34 Şeyhbayram - Abdulgaffar için Tek Dereceli Döğümler Arası Minimum Eşleşme İle Çıkan Eulerian Graf'ın Rotası



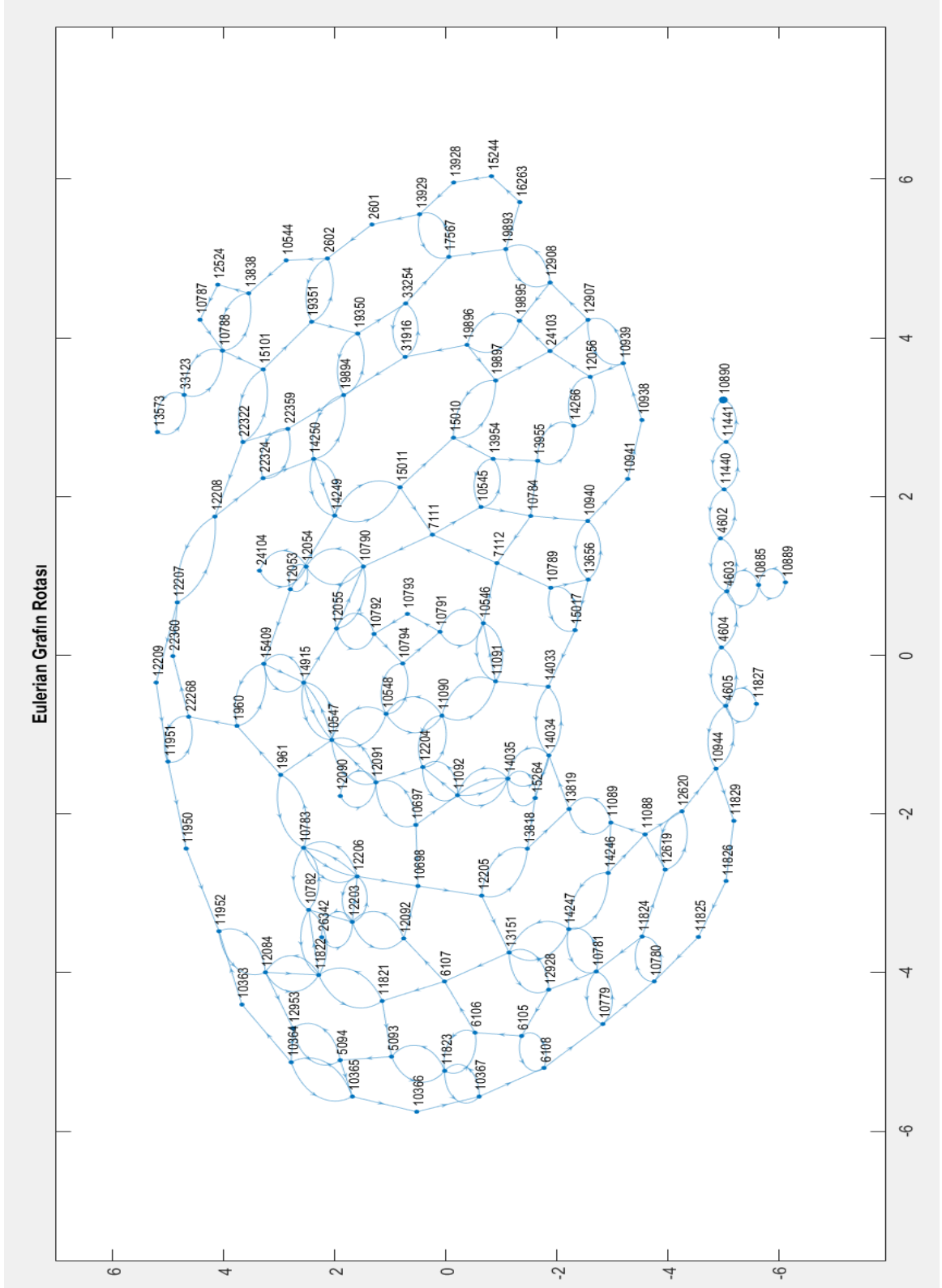
Şekil 3.35. Bölge 34 Turgut Özal İçin Minimum Maliyet



Şekil 3.36. Bölge 34 Turgut Özal İçin İlk Graf



Şekil 3.37. Bölge 34 Turgut Özal İçin Tek Dereceli Dügümler Arası Minimum Eşleşme İle Çıkan Eulerian Graf



Şekil 3.38. Bölge 34 Turgut Özal için Tek Dereceli Dügümler Arası Minimum Eşleşme İle Çıkan Eulerian Graf ın Rotası

Yönsüz Çinli Postacı Problemimiz için karşılaşılabileceğimiz olası her iki durumun nasıl olacağı ve bu olası durumlarda nasıl bir yol izlendiği görseller ve sonuçlarıyla örneklendirilerek anlatılmıştır. Uygulamanın boyutunun çok büyük olması ve toplamda 71 bölge olması sebebiyle hem Yeşilyurt hem de Battalgazi ilçeleri için geri kalan bölgelerin sonuçları aşağıdaki Tablo 3.21, 3.22, 3.23, 3.24, 3.25, 3.26, 3.27, 3.28 3.29, 3.30, 3.31, 3.32’de paylaşılmıştır.

YEŞİLYURT İlçesi İçin:

Tablo 3.21. Yeşilyurt İlçesi 1. Araç İçin Çıkan Yönsüz ÇPP Sonuçları

<i>Bölgeler</i>	<i>Grafın Düğüm (d) ve Ayrıt (a) Sayıları</i>	<i>Toplam Tek Dereceli Düğüm Sayısı</i>	<i>YAKA ile Tek Dereceli Dğümler Arası Eşleşmenin Global Minimum Uzunluğu (m)</i>	<i>İlk Grafın Uzunluğu (m)</i>	<i>Toplam Uzunluk (m)</i>	<i>Rota için Seçilen Başlangıç ID</i>	<i>Depodan Gidiş-Dönüş Toplam Uzunluk (m)</i>	<i>Aracın Depoya Dönüş Saati (civarı)</i>
<i>Bölge 1:</i>								
Bahçebaşı – Su – Yeni (yeşilyurt)	183 d 224 a	142	17749	24863	42612	1513	20000	22:20
<i>Bölge 2:</i>								
Fatih - Duruldu	92 d 123 a	70	9229	25038	34267	1771	16800	23:00
Çayırköy	55 d 78 a	42	3199	9125	12645	14898	9700	
<i>Bölge 3:</i>								
Cumhuriyet	63 d 88 a	38	1952	7258	9210	8362	5400	22:45
Cevatpaşa	190 d 244 a	122	17981	24727	42708	2198	3600	
<i>Bölge 4:</i>								
Koşu - Seyran - Gazi - Atatürk	273 d 391 a	188	19183	27427	46610	10929	8000	22:15
<i>Bölge 5:</i>								
Hoca Ahmet Yesevi	236 d 369 a	132	17198	32896	50094	34405	12600	22:50
<i>Bölge 6:</i>								
Kaynarca - Çarmuzu	172 d 215 a	124	15793	21899	37693	2877	7800	21:35

Tablo 3.22. Yeşilyurt İlçesi 2. Araç İçin Çıkan Yönsüz ÇPP Sonuçları

<i>Bölgeler</i>	<i>Grafın Düğüm (d) ve Ayrıt (a) Sayıları</i>	<i>Toplam Tek Dereceli Düğüm Sayısı</i>	<i>YAKA ile Tek Dereceli Dğümler Arası Eşleşmenin Global Minimum Uzunluğu (m)</i>	<i>İlk Grafın Uzunluğu (İlaçlama yapılacak uzunluk) (m)</i>	<i>Toplam Uzunluk (m)</i>	<i>Rota için Seçilen Başlangıç ID</i>	<i>Depodan Gidiş-Dönüş Toplam Uzunluk (m)</i>	<i>Aracın Depoya Dönüş Saati (civarı)</i>
<i>Bölge 7:</i>								
İnönü – Merkez Fatih	250 d 340 a	168	21726	27897	49623	5377	6000	21:50
<i>Bölge 8:</i>								
Yaka 2	96 d 138 a	58	8158	24521	32680	12035	7200	22:45
Topraktepe	38 d 42 a	22	4824	9391	14215	27938	12500 21000	
<i>Bölge 9:</i>								
Yaka 1	68 d 87 a	48	8621	26203	34824	18138	14400	21:50
<i>Bölge 10:</i>								
Kuşdoğan	93 d 120 a	70	12676	27701	40376	6614	38200	22:35
<i>Bölge 11:</i>								
Duranlar - Göktarla - Tohma	107 d 146 a	74	13125	23609	36734	5217	54000	22:15
<i>Bölge 12:</i>								
Bentbaşı- Kiltepe- Yeşilkayna k	285 d 401 a	206	23458	32530	55890	7721	3400	22:20

Tablo 3.23. Yeşilyurt İlçesi 3. Araç İçin Çıkan Yönsüz ÇPP Sonuçları

<i>Bölgeler</i>	<i>Grafın Düğüm (d) ve Ayrıt (a) Sayıları</i>	<i>Toplam Tek Dereceli Düğüm Sayısı</i>	<i>YAKA ile Tek Dereceli Düşümler Arası Eşleşmenin Global Minimum Uzunluğu (m)</i>	<i>İlk Grafın Uzunluğu (m)</i>	<i>Toplam Uzunluk (m)</i>	<i>Rota için Seçilen Başlangıç ID</i>	<i>Depodan Gidiş-Dönüş Toplam Uzunluk (m)</i>	<i>Aracın Depoya Dönüş Saati (civarı)</i>
Bölge 13: Bindal	47 d 54 a	24	14287	28912	43199	18207	47000	22:45
Bölge 14: Tecde 1	181 d 226 a	114	19177	27626	46803	10374	21000	22:05
Bölge 15: Tecde 2	215 d 289 a	136	21097	32612	53709	33155	21000	22:50
Bölge 16: Yakınca 1	262 d 368 a	148	17123	29453	46576	6689	23000	22:15
Bölge 17: Yakınca 2	195 d 272 a	144	19505	33486	52991	5108	23000	22:55
Bölge 18: Yakınca 3 - Gedik	204 d 269 a	140	15786	25031	40817	10487	23000	22:10

Tablo 3.24. Yeşilyurt İlçesi 4. Araç İçin Çıkan Yönsüz ÇPP Sonuçları

<i>Bölgeler</i>	<i>Grafın Düğüm (d) ve Ayrıntı (a) Sayıları</i>	<i>Toplam Tek Dereceli Düğüm Sayısı</i>	<i>YAKA ile Tek Dereceli Düğüm Arası Eşleşmenin Global Minimum Uzunluğu (m)</i>	<i>İlk Grafın Uzunluğu (m)</i>	<i>Toplam Uzunluk (m)</i>	<i>Rota için Seçilen Başlangıç ID</i>	<i>Depodan Gidiş-Dönüş Toplam Uzunluk (m)</i>	<i>Araçın Depoya Dönüş Saati (civarı)</i>
Bölge 19:								
İlyas-Melekbaba	204 d 291 a	132	12585	22649	35234	10609	2100	22:05
Salköprü	67 d 93 a	48	2278	6695	8974	15237	900 1300	
Bölge 20:								
Gözene	80 d 101 a	64	12472	27918	40389	26325	75100	22:55
Bölge 21:								
Cemal Gürsel-Aşağıbağlar	228 d 312 a	148	16258	23933	40191	11188	9200	22:00
Bölge 22:								
Dilek 1	76 d 87 a	46	15713	24317	40030	31979	29400	22:15
Bölge 23:								
Dilek 2	179 d 222 a	120	21418	35375	56792	3384	27000	23:20
Bölge 24:								
Dilek3 - Mahmutlu	102 d 134 a	68	15491	34259	49749	30545	27100	23:05

Tablo 3.25. Yeşilyurt İlçesi 5. Araç İçin Çıkan Yönsüz ÇPP Sonuçları

<i>Bölgeler</i>	<i>Grafın Düğüm (d) ve Ayrıt (a) Sayıları</i>	<i>Toplam Tek Dereceli Düğüm Sayısı</i>	<i>YAKA ile Tek Dereceli Dğümler Arası Eşleşmenin Global Minimum Uzunluğu (m)</i>	<i>İlk Grafın Uzunluğu (m)</i>	<i>Toplam Uzunluk (m)</i>	<i>Rota için Seçilen Başlangıç ID</i>	<i>Depodan Gidiş-Dönüş Toplam Uzunluk (m)</i>	<i>Aracın Depoya Dönüş Saati (civarı)</i>
Bölge 25: Karakavak	239 d 332 a	174	21205	27517	48722	2754	14400	22:05
Bölge 26: Samanlı - Çilesiz	209 d 287 a	144	18805	28137	46943	4791	11400	22:05
Bölge 27: Çukurdere- Koyunoğlu- Zaviye	221 d 321 a	146	15069	25064	40133	9238	6400	21:55
Bölge 28: Topsöğüt	128 d 156 a	90	17247	27732	44979	32238	24800	22:15
Bölge 29: Şahnahan 1 -Tepeköy	103 d 121 a	54	16607	31936	48543	952	33100	22:55
Bölge 30: Şahnahan 2	110 d 138 a	76	13167	29269	42436	1592	34000	22:15

Tablo 3.26. Yeşilyurt İlçesi 6. Araç İçin Çıkan Yönsüz ÇPP Sonuçları

<i>Bölgeler</i>	<i>Grafın Dügüm (d) ve Ayrıt (a) Sayıları</i>	<i>Toplam Tek Dereceli Dügüm Sayısı</i>	<i>YAKA ile Tek Dereceli Dügümler Arası Eşleşmenin Global Minimum Uzunluğu (m)</i>	<i>İlk Grafın Uzunluğu (m)</i>	<i>Toplam Uzunluk (m)</i>	<i>Rota için Seçilen Başlangıç ID</i>	<i>Depodan Gidiş-Dönüş Toplam Uzunluk (m)</i>	<i>Aracın Depoya Dönüş Saati (civarı)</i>
Bölge 31: Yavuz Selim	172 d 221 a	126	16212	25346	41558	10558	12400	22:00
Bölge 32: Mullakasım – Hiroğlu	177 d 215 a	116	17075	25218	42293	12243	29700	22:15
Bölge 33: Çavuşoğlu	327 d 480 a	228	21141	28291	49432	34160	5000	21:55
Bölge 34: Şeyhbayram-Abdulgaffar	169 d 240 a	110	7826	17680	25507	10154	4900	22:50
Turgut Özal	136 d 192 a	96	5713	15373	21087	10890	2100	
Bölge 35: Özal	123 d 160 a	98	11160	27533	38693	20065	5000	22:05
Bölge 36: Suluköy Sütlice	85 d 104 a	62	12353	25758	38112	420	35300	22:30

Tablo 3.27. Yeşilyurt İlçesi 7. Araç İçin Çıkan Yönsüz ÇPP Sonuçları

<i>Bölgeler</i>	<i>Grafın Düğüm (d) ve Ayrıt (a) Sayıları</i>	<i>Toplam Tek Dereceli Düğüm Sayısı</i>	<i>YAKA ile Tek Dereceli Düğüm Arası Eşleşmenin Global Minimum Uzunluğu (m)</i>	<i>İlk Grafın Uzunluğu (m)</i>	<i>Toplam Uzunluk (m)</i>	<i>Rota için Seçilen Başlangıç ID</i>	<i>Depodan Gidiş-Dönüş Toplam Uzunluk (m)</i>	<i>Aracın Depoya Dönüş Saati (civarı)</i>
Bölge 37: Görgü	112 d 155 a	76	14492	32909	47401	2384	51800	23:10
Bölge 38: Gündüzbey	225 d 289 a	174	21131	29739	50870	6494	24000	23:00
Bölge 39: Özalper	286 d 404 a	190	20673	29674	50347	11616	8600	22:20
Bölge 40: Bostanbaşı	177 d 230 a	104	13940	24921	38861	30549	18800	22:30
Bölge 41: Samanköy - Kendirli	77 d 97 a	58	11325	28354	39679	9562	22600	22:10

Toplamda 41 adet bölgeye sahip olan Yeşilyurt ilçesi için çıkan sonuçlar Tablo 3.21, 3.22, 3.23, 3.24, 3.25, 3.26, 3.27’ de paylaşılmıştır. Bu 41 bölge 7 araç ile ilaçlanacaktır. İlk 6 araç haftanın 6 günü çalışırken sadece 7. Araç haftanın 5 günü çalışacağı gözlemlenmiştir. Böylece 7. aracın bir gün eksik ilaçlama yapacak olması ile kar edilmiştir. Ayrıca bu sonuçlara göre zemini koyu renkli olan bölgeler araçların yalnızca bir ilaçlama haznesi ile ilaçlanabileceği bölgelerdir. Bu nedenle ikinci ilaçlama haznesine gerek kalmayacağından tekrar doldurma işlemi için ayrılan ekstra 15 dk lık zamandan da tasarruf sağlandığı görülmektedir.

Yeşilyurt ilçesinin yalnızca üç bölge için (Bölge 23, Bölge 24 ve Bölge 37) dönüş saati 5 ila 20 dk arası olmak üzere 23:00’ ı geçtiği gözlenmiştir. Bu bölgeler incelendiğinde, mahalle içinde araç ilaçlama işlemini tüm cadde/sokaklara uyguladığı ancak bölgenin depoya olan uzaklıklarının çok fazla olması nedeniyle gidiş – dönüş sürelerinin bu süreyi aşmasına neden olduğu gözlemlenmiştir. Bu nedenle gidiş - dönüş mesafesinin tekrar kat edilmeyip, zaman ve yakıttan tasarruf sağlanması amacıyla bu bölgelere *Varsayım 2*’de bahsedilen bölme işlemi uygulanmamış olup dönüş süresini aşma süreleri göz ardı edilmiştir.

BATTALGAZİ İlçesi İçin Sonuçlar:

Tablo 3.28. Battalgazi İlçesi 1. Araç İçin Çıkan Yönsüz ÇPP Sonuçları

<i>Bölgeler</i>	<i>Grafın Düğüm (d) ve Ayrıt (a) Sayıları</i>	<i>Toplam Tek Dereceli Düğüm Sayısı</i>	<i>YAKA ile Tek Dereceli Düğüm Arası Eşleşmenin Global Minimum Uzunluğu (m)</i>	<i>İlk Grafın Uzunluğu (m)</i>	<i>Toplam Uzunluk (m)</i>	<i>Rota için Seçilen Başlangıç ID</i>	<i>Depodan Gidiş-Dönüş Toplam Uzunluk (m)</i>	<i>Aracın Depoya Dönüş Saati (civarı)</i>
Bölge 42: Üçbağlar - Zafer - Fırat	314 d 438 a	196	25170	32457	57717	11683	2200	22:30
Bölge 43: Kernek - Hasan Varol - Başharık - Yamaç	234 d 312 a	160	17255	24461	41716	7671	2800	21:45
Bölge 44: Çöşnük	249 d 360 a	176	19481	27288	46769	10951	7000	21:50
Bölge 45: Dabakhane-Kavaklıbağ - İsmetiye - Niyazi - Şıkşık - Şifa - B.Mustafa Paşa - Hacı Abdi	284 d 396 a	198	20351	25247	45597	6391	1800	21:50
Bölge 46: HacıHaliloğlu Çiftliği - Yeniköy - Yenice	95 d 131 a	68	7321	21877	29199	5844	28800	21:40
Bölge 47: Merkez Beydağı	277 d 377 a	204	22457	26975	49431	11989	7000	22:20

Tablo 3.29. Battalgazi İlçesi 2. Araç İçin Çıkan Yönsüz ÇPP Sonuçları

<i>Bölgeler</i>	<i>Grafın Düğüm (d) ve Ayrıt (a) Sayıları</i>	<i>Toplam Tek Dereceli Düğüm Sayısı</i>	<i>YAKA ile Tek Dereceli Dğümler Arası Eşleşmenin Global Minimum Uzunluğu (m)</i>	<i>İlk Grafın Uzunluğu (m)</i>	<i>Toplam Uzunluk (m)</i>	<i>Rota için Seçilen Başlangıç ID</i>	<i>Depodan Gidiş-Dönüş Toplam Uzunluk (m)</i>	<i>Aracın Depoya Dönüş Saati (civarı)</i>
Bölge 48: Kemer Köprü – Alışar	69 d 82 a	44	10013	26103	36115	20123	26000	22:15
Bölge 49: Kuluşağı – Şişman	123 d 152 a	88	12677	26579	39256	25866	57200	22:45
Bölge 50: Hatunsuyu	143 d 178 a	90	15981	34441	50421	31793	16200	23:00
Bölge 51: Meydanbaşı – Karahan	161 d 205 a	102	11914	24475	36390	2426	8700 2800	23:30
Alacakapı	155 d 198 a	110	9857	16750	26608	17957	9900	
Bölge 52: Hanımın Çiftliği 1	197 d 230 a	148	28047	32789	60836	4963	11200	23:00
Bölge 53: Hanımın Çiftliği 2	233 d 298 a	172	25499	28298	53797	34295	11200	22:30

Tablo 3.30. Battalgazi İlçesi 3. Araç İçin Çıkan Yönsüz ÇPP Sonuçları

<i>Bölgeler</i>	<i>Grafın Düğüm (d) ve Ayrıt (a) Sayıları</i>	<i>Toplam Tek Dereceli Düğüm Sayısı</i>	<i>YAKA ile Tek Dereceli Düğüm Arası Eşleşmenin Global Minimum Uzunluğu (m)</i>	<i>İlk Grafın Uzunluğu (m)</i>	<i>Toplam Uzunluk (m)</i>	<i>Rota için Seçilen Başlangıç ID</i>	<i>Depodan Gidiş-Dönüş Toplam Uzunluk (m)</i>	<i>Aracın Depoya Dönüş Saati (civarı)</i>
<i>Bölge 54:</i>								
Bulgurlu 1 –Göller	90 d 110 a	68	10686	23354	34040	2498	11400	22:00
Karaköy	24 d 28 a	16	2511	7922	10435	2423	4700 12300	
<i>Bölge 55:</i>								
Bulgurlu 2	219 d 306 a	142	19371	30688	50059	2501	22800	22:30
<i>Bölge 56:</i>								
Bağtepe – Hacı Yusufklar-Karatepe	128 d 159 a	104	21403	30228	51631	25881	29000	22:40
<i>Bölge 57:</i>								
Boran – Toygar – Kadiçayırı	62 d 76 a	40	11425	30686	42110	12284	37800	22:40
<i>Bölge 58:</i>								
Beydağı – Fırıncı - Yaygın	84 d 97 a	70	17163	35078	52241	16515	31800	23:00
<i>Bölge 59:</i>								
Karabağlar - Yeni	78 d 92 a	50	14083	32331	46413	3930	17400	22:35

Tablo 3.31. Battalgazi İlçesi 4. Araç İçin Çıkan Yönsüz ÇPP Sonuçları

<i>Bölgeler</i>	<i>Grafın Düğüm (d) ve Ayrı (a) Sayıları</i>	<i>Toplam Tek Dereceli Düğüm Sayısı</i>	<i>YAKA ile Tek Dereceli Dügümler Arası Eşleşmenin Global Minimum Uzunluğu (m)</i>	<i>İlk Grafın Uzunluğu (m)</i>	<i>Toplam Uzunluk (m)</i>	<i>Rota için Seçilen Başlangıç ID</i>	<i>Depodan Gidiş-Dönüş Toplam Uzunluk (m)</i>	<i>Aracın Depoya Dönüş Saati (civarı)</i>
<i>Bölge 60:</i>								
İskender – Hidayet – Taştepe	271 d 388 a	182	20297	28821	49118	13612	1500 2700	23:00
Şehit Fevzi	104 d 144 a	66	2413	8803	11217	4870	1700	
<i>Bölge 61:</i>								
İzzetiye - Hamidiye – Nuriye – Aslanbey - K.Mustafa Paşa –Ataköy – Paşaköşkü – Cevherizade – Ferhadiye	271 d 378 a	182	18284	26292	44576	8090	1900	21:45
<i>Bölge 62:</i>								
Akpınar – Kırçuval – Sancaktar – B.Hüseyinbey – K.Hüseyinbey - Saray – İstiklal - Yenihamam –	272 d 386 a	190	15037	20938	35975	7432	0 900	22:45
Cirikpınar	57 d 77 a	36	1520	5007	6528	34324	2600	
Selçuklu	96 d 142 a	56	3308	11139	14448	13394	1800	
<i>Bölge 63:</i>								
Toptaş - Bahçelievler - Dolamantepe	154 d 189 a	96	12416	28166	40582	16143	23200	22:10
<i>Bölge 64:</i>								
Halfettin – Sarcioğlu – Beylerbaşı	181 d 254 a	118	6996	15157	22153	7660	500 2600	22:40
Göztepe	173 d 244 a	96	7368	19515	26883	10456	2400	
<i>Bölge 65:</i>								
Çamurlu	88 d 98 a	40	19899	29485	49384	5950	19200	22:45

Tablo 3.32. Battalgazi İlçesi 5. Araç İçin Çıkan Yönsüz ÇPP Sonuçları

Bölgeler	Grafın Düğüm (d) ve Ayrıntı (a) Sayıları	Toplam Tek Dereceli Düğüm Sayısı	YAKA ile Tek Dereceli Dğümler Arası Eşleşmenin Global Minimum Uzunluğu (m)	İlk Grafın Uzunluğu (m)	Toplam Uzunluk (m)	Rota için Seçilen Başlangıç ID	Depodan Gidiş-Dönüş Toplam Uzunluk (m)	Aracın Depoya Dönüş Saati (civarı)
Bölge 66: Hasırıcılar	83 d 101 a	68	14891	32100	46991	15079	30600	22:55
Bölge 67: Orduzu 1	89 d 103 a	62	16534	33512	50146	13761	13000	22:45
Bölge 68: Orduzu 2	137 d 176 a	100	16152	31364	47517	5478	13000	22:35
Bölge 68: Orduzu 3	97 d 123 a	70	9500	22740	32240	8758	6500	23:30
Battalgazi	205 d 300 a	134	12637	21451	34088	10210	2900	
Bölge 70: Tandoğan Yıldıztepe -	336 d 485 a	220	29139	34618	63757	6837	7200	23:00
Bölge 71: Çolakoğlu	75 d 92 a	48	5818	25909	31727	4500	33600	22:00

Toplamda 30 adet bölgeye sahip olan Battalgazi ilçesi için çıkan sonuçlar Tablo 3.28, 3.29, 3.30, 3.31, 3.32’ de paylaşılmıştır. Bu sonuçlara göre zemini koyu renkli olan bölgeler araçların yalnızca bir ilaçlama haznesi ile ilaçlanabileceği bölgelerdir. Bu nedenle ikinci ilaçlama haznesine gerek kalmayacağından tekrar doldurma işlemi için ayrılan ekstra 15dk lık zamandan da tasarruf sağlandığı görülmektedir.

Battalgazi ilçesinin yalnızca iki bölgesi için (Bölge 51ve Bölge 68) dönüş saati 23:00’ ı geçtiği gözlenmiştir. Yine bu bölgeler incelendiğinde, bölgeye ait mahalle içinde araç ilaçlama işlemini tüm cadde/sokaklara uyguladığı ancak Bölge 51’in depoya olan uzaklığının çok fazla olması nedeniyle gidiş – dönüş sürelerinin bu süreyi aşmasına neden olduğu gözlemlenmiştir. Bu nedenle gidiş - dönüş mesafesinin tekrar kat edilmeyip zaman ve yakıttan tasarruf sağlanması amacıyla bu bölgelere *Varsayım 2*’de bahsedilen bölme işlemi uygulanmamış olup dönüş süresini aşma süreleri göz ardı edilmiştir.

Her iki ilçemiz için de mahalle içerisinde toplamda 12 araç ile 19:00 - 23:00 arası bölgelerde optimum gezinmeleri sağlanarak mesafe, zaman ve maliyetten kâr sağlandığı gibi gereksiz böcek ilacı salımında önüne geçilmiştir.

Yeşilyurt ilçesine ait 41 bölgenin 34'ü, Battalgazi ilçesine ait 30 bölgenin ise 21'ne ait araçlarının ilaçlama haznesini bir defa doldurarak (tabloda zemini koyu olan bölgeler) o bölgeyi tamamen ilaçlayacağı sonucuna varılmıştır. Böylece hem ikinci defa hazneyi doldurmak için ayrılan 15dk lık zamandan, hem de ikinci defa kullanılmak için yedek olarak alınan böcek ilacından tasarruf sağlanmıştır. Araçların belirlenen süre içerisinde bölge içerisindeki tüm cadde/sokaklara girerek ilaçlama işlemini yapmaları sağlanmıştır.

Yeşilyurt ilçesine ait üç bölgenin, Battalgazi ilçesine ait iki bölgenin aracının sadece dönüş süresi 23:00' ı aştığı gözlemlenmiştir. Ancak gidilen bölgenin depoya olan uzaklığından dolayı veya bu bölgenin bölünemez olmasından dolayı ilaçlama işlemini bitirmeden dönmemesi için toplamda 5 aracın ortalama 5 ila 30 dk arasındaki gecikmeleri göz ardı edilmiştir. Ayrıca Yeşilyurt ilçesinin 26 adet bölgesine ait araçların 21:35 - 22:30, Battalgazi ilçesinin 13 adet bölgesine ait araçların 21:40 - 22:30 saatleri arasında depoya erken döndüğü gözlemlenmiştir. İlgili kurum ile yapılan görüşmelerde gelen şikâyet telefonları üzerine bazen bölgede ilaçlama yapan araçları zaman ve mesafe durumu gözetmeksizin bu şikâyet gelen mahallelere yönlendirdikleri veya ellerinde fazladan bulunan 13. aracı şikâyet gelen bölgeye gitmek için çıkardıkları bilgisi edinilmiştir. Bu nedenle bu çalışma sonucunda depoya erken döneceği belirlenen her iki ilçe için toplamda 39 adet aracın şikâyetler üzerine yönlendirilmesi söz konusu olacaktır. Böylece fazladan bir aracın daha çalışmasına (13. araç) gerek kalmayacağı veya bölgede ilaçlama yapan aracın rotasının plansız bir şekilde değiştirilmesine imkân verilmemesi sağlanmıştır.

Uygulamanın sonuçları ile mevcut araçların gezinme sonuçları Tablo 3.33'de karşılaştırılmıştır. Kurumun 2017 yılı Haziran – Eylül ayları ilaçlama araçları için belirlediği çalışma planı Tablo 3.1'de ki gibidir. Karşılaştırma için mevcut araçların işleyişinde rasgele bir hafta seçilmiş olup 14 – 19 Ağustos 2017 tarihi seçilmiştir. Rasgele bir hafta belirlenmiştir, çünkü araçların çalışma durumlarında her hafta bir farklılık olduğu gözlemlenmiştir. Bu nedenle seçilen bu bir haftanın raporları EK – 3'de

paylaşmıştır. Araçlar mesai bitimi olan saat 17:00' da şoförlere teslim edildiğinden dolayı araç raporları bu satten itibaren alınmıştır. İlaçlamaya ayırdıkları süreler ayrıca hesaplanmıştır.

Tablo 3.33. Bu Çalışmadaki Uygulamanın Sonuçları ile Mevcut Araçların İşleyiş Sonuçlarının Karşılaştırılması (Bir Haftalık Sonuçlar)

	Bu Çalışmada Yapılan Uygulamanın Sonuçları	Mevcut Araçların İşleyiş Sonuçları (14-19 Ağustos 2017 tarihli)
İlaçlanmamış mahalle var mı?	Yok	Var *(28 tane)*
Kullanılan araç sayısı	12	13
Aldığı toplam mesafe (km)	4842,94	4500,79
Harcanan toplam süre	245 saat	315 saat
İlaçlama işlemi için aldığı toplam mesafe (km)	2110,85	1723,27
İlaçlama işlemi için harcanan toplam süre	162 saat	182 saat

EK-3' e göre 14-19 Ağustos 2017 tarihli ARVENTO program raporlarına göre 28 adet mahallenin ilaçlanmadığı tespit edilmiştir. Bunlar; *Abdulgaffar, Aşağıbağlar, Bağtepe, Bahçebaşı, Beydağı, Çarmuzu, Çayırköy, Duranlar, Duruldu, Fatih, Fırıncı, Göktarla, Göller, HacıHaliloğlu Çiftliği, HacıYusuflar, Karaköy, Karatepe, Kuşdoğan, Melekbaba, Samanköy, Samanlı, Su, Tohma, Toygar, Turgut Özal, Yeni (Yeşilyurt), Yenice ve Yeniköy* mahalleleridir. Ayrıca 2017 yılı için Tablo 3.1' de belirlenen mahalle ekip dağılımı planına da EK-3' deki raporlar incelendiğinde uyulmadığı gözlemlenmektedir.

Belirlenen bir haftalık (6 günlük çalışma) EK- 3'deki raporların incelenmesine göre kurumun mevcut araçları Yeşilyurt ve Battalgazi ilçelerine ait 28 adet mahalleyi hiç ilaçlamamıştır. Eksik ilaçlamaya rağmen 13 araç kullanılmış ve bir haftada toplamda 4500,79 km yolu 315 saatte almışlardır. Bu toplam mesafenin 1723,27 km'si ise ilaçlama işlemi için alınan mesafe olup 182 saat harcadığı gözlenmiştir. Araçların ilaçlama işlemi için gittiği mahallelerde ise tüm mahalleyi ilaçlamadan ayrıldığı da yine ARVENTO programında yapılan hareketli gezinme incelemelerinde gözlemlenmiştir.

Bu çalışma da yapılan uygulama sonuçlarına göre ise her iki ilçe için ilaçlanmamış mahalle hiç kalmamıştır. Bunun için 12 araç ile toplamda 4842,94 km yi 245 saatte almaktadır. Bu toplam mesafenin 2110,85 km si ilaçlama işlemi için olup 162 saatte alınmıştır. Bir haftalık karşılaştırmaya göre bu çalışmada uygulamanın sonucuna göre mevcut durumdan bir araç eksik çalışılmış ve toplam süreden 70 saat, ilaçlama süresinden 20 saat olmak üzere kâr elde etmiştir. Dört aylık (Haziran – Eylül) çalışma dönemleri düşünülürse eğer toplam süreden 1120 saat kâr elde edilmiştir. Alınan 4842,94 km' nin ilaçlama için harcanan 2110,85 km'si ile belirlenen bölgelerde ilaçlanmamış alan hiç kalmamıştır.

İlaçlama araçları motorin ile çalışmakta olup (bir depo 60-65lt), ULV ilaçlama makinesi ise benzin ile çalışmaktadır (bir depo 30 litre). ULV makinesinin bir depo benzini (gezindiği bölgenin büyüklüğüne göre) haftada 2 veya 3 gün gittiği gözlemlenmiştir. 2017 yılı Ağustos ayı motorin fiyatları ortalama 4.5 TL/lt, benzin fiyatlarının ise ortalama 5 TL/lt olduğu göz önünde bulundurulduğunda, bu çalışmada elde edilen haftalık 70 saat daha az çalışma sonucu ile mevcut durumun sonuçları (28 mahalle eksik olmasına rağmen) karşılaştırıldığında bir hafta için yaklaşık %22 iyileşme sağlanmıştır. Haliyle hem harcanan zaman, hem yakıt tüketimi, hem de alınan mesafeden tasarruf elde edilmiştir. Dolayısıyla çevre kirliliğinin azalması da fayda sağlanmaktadır.

Kurumun mevcut araçları eğer 28 tane mahalleyi belirlenmiş olan haftada ilaçlamış olsaydı Tablo 3.33'de görüldüğü gibi mevcut durumdaki alınan süre ve zaman da bir hayli artacaktı. 28 mahalle eksik ilaçlamasına rağmen 13 araç ile 315 saatte 4500,79 km yol kat etmişlerdir. Oysaki uygulamamız da eksik hiçbir mahalle olmaması haliyle bile 12 araç ile 245 saatte 4842,94 km almıştır. Bu nedenle bu çalışmanın

uygulama sonuçlarına göre ilaçlanmamış mahalle hiç kalmadığından Tablo 3.33'deki bu uygulamanın sonuçları net sonuçlardır.

Raporlanan hafta için ilaçlanmamış 28 mahalle mevcut durumda ilaçlanmış olsaydı, bu çalışmada ki uygulama ile elde edilen iyileşme oranının aslında %22'den çok daha fazla olduğu sonucu ortaya çıkmaktadır.



BÖLÜM 4

4. SONUÇ

Bu çalışmada, Malatya Büyükşehir Belediyesi, Çevre Koruma ve Kontrol Dairesi Başkanlığı, Atık Yönetim ve İlaçlama Şube Müdürlüğü' ne bağlı haşerelerle mücadele için Malatya ili sınırları içinde ilaçlama yapan araçların şimdiye kadar rastgele belirlenen güzergahlarının optimum rotalarının belirlenmesi ve böylece maliyetlerin minimizasyonu yapılmıştır. İlgili birim ile yapılan görüşme sonucu haftanın altı günü akşam 19:00-23:00 arasında ilaçlama yapan 13 adet homojen ULV(sisleme) araçlarının hem mahalle hem de cadde/sokak içerisindeki gezintileri tamamen rastgele yapılmaktadır. Rastgele hareket eden araçlar genellikle yanlış bir rota izledikleri için mahalle içerisinde tüm cadde/sokaklara girmeden ilaçlama işlemini bitirmektedirler. Bu durum maliyet, mesafe ve süre açısından büyük bir dezavantajdır. Ayrıca Türkiye yakıt fiyatlarının en yüksek olduğu ülkelerden biridir ve günümüzde de yakıt fiyatları bir hayli artmıştır. Bu sorun Türkiye'de kamu kurumları, işletmeler ve özel firmalar için Araç Rotalama Problemi çözümünün önemini arttırmaktadır. Dağıtımda, ulaştırma ve lojistikde yönetiminin etkin ve verimli gerçekleştirilmesi maliyet düşüşlerini doğrudan etkilemektedir. Uygun rotaların elde edilmesi sonucu oluşan kazanımlar, karlılığı arttırmaktadır.

Malatya Büyükşehir Belediyesinin bir matematiksel model kullanmadan yaptıkları ilaçlama araç rotaları, yüksek dağıtım maliyetlerinin ortaya çıkmasına neden olmaktadır. Malatya ilinin Battalgazi ve Yeşilyurt ilçelerine ait ilaçlama araçlarının mahalle ve caddelerde rastgele gezinerek ilaçlama işlemini yapmasından dolayı mesafe ve maliyet ciddi derecede göz ardı edilmiştir. Ayrıca ilaçlama işleminin bir zorunluluğu olan tüm cadde/sokakların ilaçlama zorunluluğu da büyük ölçüde ihlal edilmiştir. Araçlar her cadde/sokağa girme zorunluluğuna uymadan ilaçlama işlemini bitirmektedir (ARVENTO isimli program ile çipli olan ilaçlama araçların incelenmesi ile bu sonuca varılmıştır). Bu çalışma ile bu anlamda hem mahalle dağılımında optimum rotayı sağlayarak, hem de en kısa mesafe ve maliyet ile araçlar tüm cadde/sokakları gezinerek ilaçlama işlemini bitireceklerdir.

Sezgisel yaklaşımlar karmaşık bir optimizasyon problem olan rotalama problemlerinde sıklıkla kullanılan çözüm yöntemleridir. Rotalama çözümlerinin

geliştirilmesi ile ilgili akademik çalışmalarda kesin çözüm veren yöntemler yerine sezgisel yöntemler ağırlık kazanmaktadır. Tez çalışmasında, *NP* - zor sınıfına giren düğüm rotalama konusunun önemli türlerinden olan ve bu çalışmada kullanılan Araç Rotalama Problemi ve ayrıt rotalama da Çinli Postacı Problemi için sezgisel yöntemlerle çok aşamalı bir uygulama yapılarak çözüm aranmıştır.

Uygulamamızın ilk kısmı olan mahalleleri araçlara dağıtıp ve belirli süre içerisinde optimum rotama elde edilmesi gerektiğinden dolayı Zaman Pencereci Araç Rotalama Problemi kullanılmıştır. Yeşilyurt ilçesine ait 70, Battalgazi ilçesine ait 84 mahalle olmak üzere, toplam 154 (büyük mahallelerin bölünme işlemi sonucu elde edilen sayı) adet mahalle olmasından dolayı problemimizin çözüm yöntemi için sezgisel yaklaşıma ihtiyaç duyulmuştur. Her iki ilçe ayrı düşünülmüş olup Yeşilyurt ilçesinin 70 mahallesi, Battalgazi ilçesinde 84 mahallesi önce kendi aralarında yakınlık durumlarına göre kümelenmiş ve daha sonra metasezgisel yöntemlerden Genetik Algoritma ile haftanın altı günü, belirlenen süre içerisinde çalışacak şekilde optimum rotaları elde edilmiştir. Bu durumda çıkan sonuca göre Yeşilyurt ilçesine ait mahallelerin ilaçlama işlemi için 7 araç gerektiği ve 7. aracın ise beş gün çalışacağı gözlemlenmiştir. Battalgazi ilçesine ait mahalleler için ise 5 araç gerektiği ve tüm araçların haftanın altı günü çalıştığı gözlemlenmiştir.

Uygulamamızın ikinci kısmı için öncelikle ilk uygulamada yakınlık durumlarına göre kümelenen mahalle/mahalle grupları bu kısımda artık bölge olarak düşünülmüştür. Bu haliyle Yeşilyurt ilçesine ait ilaçlanması gereken bölge sayısı 41, Battalgazi ilçesine ait ilaçlanması gereken bölge sayısı ise 30 olmuştur. Belirlenen bölgelerin içerisindeki cadde/sokaklarda gezinme işlemi için önce QGIS programı yardımı ile uzaklık matrisleri oluşturulmuş ve Yapay Arı Kolonisi Algoritması ile Yönsüz Çinli Postacı Problemi' nin çözümü sağlanmıştır. Çinli Postacı Probleminin çözümünde en önemli kısım tek dereceli düğümlerin aralarında minimum şekilde eşleşmesidir. Bu eşleşme adımı tek dereceli düğüm sayısının az olduğu küçük boyutlu graflarda elle dahi çözülebilecek iken, tek dereceli düğüm sayısının çok fazla olduğu büyük graflarda ise günümüz bilgisayarlarında dahi neredeyse çözümü imkânsız olmaktadır. Bu nedenle bu çalışmada da oluşturulan bölgelerden dolayı graflarımızdaki tek dereceli düğüm sayısı çok fazla olduğundan sezgisel yaklaşıma ihtiyaç duyulmuş olup bunun için Yapay Arı

Kolonisi Algoritmasına modifiye işlemi sayesinde tek dereceli düğümler arası optimum eşleşme sağlanmıştır. Tek dereceli düğümlerin optimum eşleşmesi sonucu elde edilen Eulerian grafi için Fleury algoritması ile Euler turlar elde edilmiştir. Euler tur elde edilmesi sonucunda da graflarımız artık yönlü bir graf haline dönüşmüştür. Bu uygulamanın sonucuna göre bölge içerisinde araçların optimum gezinmesi sağlanmıştır. Depodan çıkan aracın bölgeye en yakın düğümden başlaması ile de ilaçlama haznesi dolu olan aracın daha az yakıt tüketeceği düşüncesinden yola çıkılarak yakıt tasarrufu sağlanmıştır.

Sonuç olarak bu çalışma, optimum rotaların elde edilmesi ile mevcut durumda 28 adet mahallenin hiç ilaçlanmamış olması halinde bile haftalık karşılaştırma sonucuna göre %22 lik bir iyileşme sağlamıştır. Eğer mevcut araçlar 14 – 19 Ağustos 2017 tarihinde bu 28 mahalleyi de ilaçlasalardı toplam mesafe (4500,79 km) ve süre (315 saat) daha da artacağından dolayı bu çalışmadaki uygulama ile elde edilen iyileşme oranının %22 den çok daha fazla olacağını göstermektedir. Çünkü bu çalışmada elde edilen uygulama sonucuna göre ilaçlanmamış hiç bir mahalle bulunmamaktadır, bu nedenle bu çalışmadaki uygulamanın toplam alınan mesafe ve süre sonuçları net sonuçlardır.

QGIS programı yardımı ile tarafımca oluşturulan tüm veriler gerçek ve orjinaldir. Ayrıca Yönsüz Çinli Postacı Probleminin eşleşme adımının çözümünde Yapay Arı Kolonisi Algoritmasına modifiye işlemi uygulanarak elde edilen çözüm şekli de literatüre kazandırılmıştır. Ve bu çalışma akşam ilaçlama yapan araçlar için en kısa mesafe, minimum maliyet, minimum zaman elde edilecek şekilde bulunan rotalar ile tez sonrası projelendirilerek Malatya Büyükşehir Belediyesi tarafından kullanılması sağlanabilecektir. Ayrıca uygulamamız dinamik bir model olup diğer illerin haritalarına göre adapte edilebilmektedir. Belediyelerin bu uygulamayı yazılım programıyla bir navigasyon cihazı oluşturularak kullanabilmeleri yine proje kapsamında hedeflenmektedir.

İleride aynı modele farklı sezgisel algoritmalar ile çözüm aranması ve ayrıca Çinli Postacı Probleminin diğer türlerine ait olan belediyelerin çöp kamyonlarının, kar araçlarının rotalanması, yol yapımı ve onarımı v.s. gibi yeni alanlara sezgisel yaklaşımlarla uygulanması planlanmaktadır.

Literatürde böyle kapsamlı bir çalışmanın fazla bulunmaması nedeniyle bu tarz çalışmalar çoğaltılarak, kamu ve özel firmalar için özellikle finansal açıdan tasarruf elde edilmesi ile ekonomiye büyük anlamda fayda sağlanmasına imkân verilmelidir.



KAYNAKÇA

- Ahr, D., & Reinelt, G. (2002, September). New Heuristics and Lower Bounds for The Min-Max K-Chinese Postman Problem. In *European symposium on algorithms* (pp. 64-74). Springer, Berlin, Heidelberg.
- Ahr, D. (2004). *Contributions to Multiple Postmen Problems*(Doctoral dissertation), Natural Sciences Faculty of Mathematics, Heidelberg University, Germany.
- Ahr, D., & Reinelt, G. (2006). A Tabu Search Algorithm for The Min–Max K-Chinese Postman Problem. *Computers & operations research*, 33(12), 3403-3422.
- Ahuja, R. K., Magnanti, T. L., & Orlin, J. B. (1993). *Network Flows: Theory, Applications and Algorithms*. Prentice-Hall, Englewood Cliffs, New Jersey, USA
- Arrow, KJ (1963). *Social Choice and Individual Values*, Wiley, New York.
- Gibbard, A.(1973). “Manipulation of Voting Schemes: A general result”, *Econometrica*, 41, 587-602.
- Akay, B. (2009). Nümerik Optimizasyon Problemlerinde Yapay Arı Kolonisi (Artificial Bee Colony) Algoritmasının Performans Analizi, Doktora Tezi, Erciyes Üniversitesi Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, Kayseri (Türkiye).
- Akyurt, İ. Z., Keskinurk, T., & Kalkancı, Ç. (2015). Using Genetic Algorithm for Winter Maintenance Operations: Multi Depot K-Chinese Postman Problem. *EMAJ: Emerging Markets Journal*, 5(1), 50-59.
- ALFA, A. S., & LIU, D. Q. (1988). Postman Routing Problem in A Hierarchical Network. *Engineering optimization*, 14(2), 127-138.
- Aráoz, J., Fernández, E., & Meza, O. (2009). Solving The Prize-Collecting Rural Postman Problem. *European Journal of Operational Research*, 196(3), 886-896.
- Arkin, E. M., Hassin, R., & Levin, A. (2006). Approximations for Minimum and Min-Max Vehicle Routing Problems. *Journal of Algorithms*, 59(1), 1-18.

- Assad, A. A., Pearn, W. L., & Golden, B. L. (1987). The Capacitated Chinese Postman Problem: Lower Bounds and Solvable Cases. *American Journal of Mathematical and Management Sciences*, 7(1-2), 63-88.
- Assad, A. A., & Golden, B. L. (1995). Arc Routing Methods and Applications. *Handbooks in operations research and management science*, 8, 375-483.
- Atmaca, E. (2012). Bir Kargo Şirketinde Araç Rotalama Problemi. *Tübvav Bilim Dergisi*, 5(2), 12-27.
- Augerat, P., Belenguer, J. M., Benavent, E., Corbéran, A., & Naddef, D. (1998). Separating Capacity Constraints in The CVRP Using Tabu Search. *European Journal of Operational Research*, 106(2-3), 546-557.
- Bakır, M. A., & Altunkaynak, B., (2003). Tamsayılı Programlama: Teori, Modeller ve Algoritmalar (1. Baskı), Nobel yayıncılık, Ankara.
- Banzhaf, W. (1999). *Foundations of Genetic Algorithms*. Morgan Kaufmann Publishers Inc..
- Belenguer, J. M., & Benavent, E. (1998). The Capacitated Arc Routing Problem: Valid Inequalities And Facets. *Computational Optimization and Applications*, 10(2), 165-187.
- Belenguer, J. M., & Benavent, E. (2003). A Cutting Plane Algorithm for The Capacitated Arc Routing Problem. *Computers & Operations Research*, 30(5), 705-728.
- Belenguer, J. M., Benavent, E., Lacomme, P., & Prins, C. (2006). Lower and Upper Bounds for The Mixed Capacitated Arc Routing Problem. *Computers & Operations Research*, 33(12), 3363-3383.
- Beltrami, E. J., & Bodin, L. D. (1974). Networks and Vehicle Routing for Municipal Waste Collection. *Networks*, 4(1), 65-94.
- Benavent, E., Campos, V., Corberán, A., & Mota, E. (1992). The Capacitated Arc Routing Problem: Lower Bounds. *Networks*, 22(7), 669-690.
- Benavent, E., & Soler, D. (1999). The Directed Rural Postman Problem with Turn Penalties. *Transportation Science*, 33(4), 408-418.

- Benavent, E., Carrota, A., Corberán, A., Sanchis, J. M., & Vigo, D. (2007). Lower Bounds and Heuristics for The Windy Rural Postman Problem. *European journal of operational research*, 176(2), 855-869.
- Benavent, E., Corberán, A., Plana, I., & Sanchis, J. M. (2009). Min-Max K-Vehicles Windy Rural Postman Problem. *Networks: An International Journal*, 54(4), 216-226.
- Blum, C., & Roli, A. (2003). Metaheuristics In Combinatorial Optimization: Overview and Conceptual Comparison. *ACM computing surveys (CSUR)*, 35(3), 268-308.
- Brandão, J., & Eglese, R. (2008). A Deterministic Tabu Search Algorithm for The Capacitated Arc Routing Problem. *Computers & Operations Research*, 35(4), 1112-1126.
- Brucker, P. (1980, June). The Chinese Postman Problem for Mixed Graphs. In *International Workshop on Graph-Theoretic Concepts in Computer Science* (pp. 354-366). Springer, Berlin, Heidelberg.
- Burrough, P. A. (1998). Principles of Geographical.
- Cabral, E. A., Gendreau, M., Ghiani, G., & Laporte, G. (2004). Solving The Hierarchical Chinese Postman Problem as A Rural Postman Problem. *European Journal of Operational Research*, 155(1), 44-50.
- Černý, V., (1985). Thermodynamical Approach to The Traveling Salesman Problem: An Efficient Simulation Algorithm. *Journal of optimization theory and applications*, 45(1), 41-51.
- Chambers, L. D. (1998). *Complex Coding Systems*. CRC Press, Inc..
- Chambers Lance, D. (1999). Practical Handbook of Genetic Algorithm Complex Coding System.
- Christofides, N. (1973). The Optimum Traversal of A Graph. *Omega*, 1(6), 719-732.
- Christofides, N., Mingozzi, A., & Toth, P. (1979). The Vehicle Routing Problem. In N. Christofides, A. Mingozzi, P. Toth, & C. Sandi (Eds.), *Combinatorial Optimization* (pp. 315–338). Chichester, UK: Wiley.

- Christofides, N., Benavent, E., Campos, V., Corberán, A., & Mota, E. (1984). An Optimal Method for The Mixed Postman Problem. In *System Modelling and Optimization* (pp. 641-649). Springer, Berlin, Heidelberg.
- Christofides, N., Campos, V., Corberán, A., & Mota, E. (1986). An Algorithm for The Rural Postman Problem on A Directed Graph. In *Netflow at pisa* (pp. 155-166). Springer, Berlin, Heidelberg.
- Clarke, G., & Wright, J. W. (1964). Scheduling of Vehicles From A Central Depot to A Number of Delivery Points. *Operations research*, 12(4), 568-581.
- Corberán, A., & Sanchis, J. M. (1994). A Polyhedral Approach to The Rural Postman Problem. *European Journal of Operational Research*, 79(1), 95-114.
- Corberán, A., Martí, R., Martínez, E., & Soler, D. (2002). The Rural Postman Problem On Mixed Graphs with Turn Penalties. *Computers & Operations Research*, 29(7), 887-903.
- Corberán, A., Martí, R., & Sanchis, J. M. (2002). A GRASP Heuristic for The Mixed Chinese Postman Problem. *European Journal of Operational Research*, 142(1), 70-80.
- Corberán, A., Mejía, G., & Sanchis, J. M. (2005). New Results on The Mixed General Routing Problem. *Operations Research*, 53(2), 363-376.
- Corberán, A., Plana, I., & Sanchis, J. M. (2007). A Branch & Cut Algorithm for The Windy General Routing Problem and Special Cases. *Networks: An International Journal*, 49(4), 245-257.
- Corberán, A., & Prins, C. (2010). Recent Results on Arc Routing Problems: An Annotated Bibliography. *Networks*, 56(1), 50-69.
- Corberán, A., Oswald, M., Plana, I., Reinelt, G., & Sanchis, J. M. (2012). New Results on The Windy Postman Problem. *Mathematical programming*, 132(1-2), 309-332.
- Corberán, A., Plana, I., Rodríguez-Chía, A. M., & Sanchis, J. M. (2013). A Branch-And-Cut Algorithm for The Maximum Benefit Chinese Postman Problem. *Mathematical Programming*, 141(1-2), 21-48.

Corberán, Á., & Laporte, G. (Eds.). (2013). *Arc Routing: Problems, Methods, and Applications*. Society for Industrial and Applied Mathematics.

Cordeau, J. F., Laporte, G., Savelsbergh, M. W., & Vigo, D. (2007). Vehicle Routing. *Handbooks in operations research and management science*, 14, 367-428.

Crowder, H., & Padberg, M. W. (1980). Solving Large-Scale Symmetric Travelling Salesman Problems to Optimality. *Management Science*, 26(5), 495-509.

Cura, T., (2008). *Modern Sezgisel Teknikler ve Uygulamaları*. Papatya Yayıncılık Eğitim.

Çevre, U., (2008). Çoklu Gezgin Satıcı Probleminin Çözümü İçin Bir Eniyileme Kütüphanesinin Tasarımı Ve Görsel Yazılım Geliştirme Ortamı İle Birlikte Gerçekleştirdim. Yüksek Lisans Tezi, Ege Üniversitesi, İzmir (Türkiye).

Dantzig, G., Fulkerson, R., & Johnson, S. (1954). Solution Of A Large-Scale Traveling-Salesman Problem. *Journal Of The Operations Research Society Of America*, 2(4), 393-410.

Dantzig, G. B., & Ramser, J. H. (1959). The Truck Dispatching Problem. *Management science*, 6(1), 80-91.

Davendra, D., (2010). *Travelling Salesman Problem, Theory and Applications*, Published by InTech, India.

Davis, L. (1989). Adapting Operator Probabilities in Genetic Algorithms. In *Proc. 3rd International Conference on Genetic Algorithms, 1989*.

DeArmon, J. S. (1981). *A Comparison Of Heuristics for The Capacitated Chinese Postman Problem* (Doctoral dissertation, University of Maryland).

Degenhardt, J. (2004). An Ant-Algorithm for The Balanced K-Chinese Postmen Problem. In *Operations research conference*.

De Jong, K.A., "An Analysis of the Behavior of a Class of Genetic Adaptive Systems", (Doctoral dissertation, University of Michigan), *Dissertation Abstract International*, 36(10), 5140B. (University Microfilms No 76-9381).

Demirciođlu, M., (2009). Ara Rotalama Probleminin Sezgisel Bir Yaklařım İle özömlenmesi Üzerine Bir Uygulama. Doktora Tezi, ukurova Üniversitesi, İřletme Anabilim Dalı, Adana (Türkiye).

Deste, M., (2013). Bütünleřik Lojistik Ađ Tasarımı ve Ara Rotalama Problemleri: Bir Model Önerisi ve Uygulama, Doktora Tezi, Erciyes Üniversitesi, İřletme Anabilim Dalı, Üretim Yönetimi ve Pazarlama Bilim Dalı Kayseri (Türkiye).

Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant System: Optimization by A Colony of Cooperating Agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 26(1), 29-41.

Dorigo, M., & Gambardella, L. M. (1997). Ant Colony System: A Cooperative Learning Approach to The Traveling Salesman Problem. *IEEE Transactions on evolutionary computation*, 1(1), 53-66.

Drexl, M. (2014). On The Generalized Directed Rural Postman Problem. *Journal of the Operational Research Society*, 65(8), 1143-1154.

Dror, M., Stern, H., & Trudeau, P. (1987). Postman Tour on A Graph With Precedence Relation On Arcs. *Networks*, 17(3), 283-294.

Dror, M. (Ed.). (2000). *Arc Routing: Theory, Solutions and Applications*. Springer Science & Business Media.

Du, K.-L., & Swamy, M. N. S. (2016). *Searched and Optimization by Metaheuristics: Techniques and Algorithms Inspired by Nature*, Springer International Publishing, Switzerland.

Dueck, G., & Scheuer, T. (1990). Threshold Accepting: A General Purpose Optimization Algorithm Appearing Superior to Simulated Annealing. *Journal of computational physics*, 90(1), 161-175.

Durucasu, H. (2004). Bir Polis Devriye Aracı Rotasının Elektronik alıřma Sayfası Modeli Yardımıyla Belirlenmesi.

- Düzakın, E., & Demircioğlu, M. (2009). Araç Rotalama Problemleri ve Çözüm Yöntemleri. *Çukurova Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 13(1).
- Dussault, B., Golden, B., GroëR, C., & Wasil, E. (2013). Plowing With Precedence: A Variant of The Windy Postman Problem. *Computers & Operations Research*, 40(4), 1047-1059.
- Eberhart, R., & Kennedy, J. (1995, October). A New Optimizer Using Particle Swarm Theory. In *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on* (pp. 39-43). IEEE.
- Edmonds, J. (1965). Maximum Matching and A Polyhedron With 0, 1-Vertices. *Journal of research of the National Bureau of Standards B*, 69(125-130), 55-56.
- Edmonds, J., & Johnson, E. L. (1973). Matching, Euler Tours and the Chinese Postman. *Mathematical programming*, 5(1), 88-124.
- Eiselt, H. A., Gendreau, M., & Laporte, G. (1995a). Arc Routing Problems, Part I: The Chinese Postman Problem. *Operations Research*, 43(2), 231-242.
- Eiselt, H. A., Gendreau, M., & Laporte, G. (1995b). Arc Routing Problems, Part II: The Rural Postman Problem. *Operations research*, 43(3), 399-414.
- Eglese, R. W. (1994). Routeing Winter Gritting Vehicles. *Discrete applied mathematics*, 48(3), 231-244.
- Eglese, R. W., & Li, L. Y. (1996). A Tabu Search Based Heuristic for Arc Routing With A Capacity Constraint and Time Deadline. In *Meta-Heuristics* (pp. 633-649). Springer, Boston, MA.
- Elmas, Ç., (2016). Yapay Zekâ Uygulamaları (3. Baskı), Seçkin Yayıncılık, Ankara.
- Emel, G. G., Taşkın, Ç., & Dinç, E. (2003). Yönsüz Çinli Postacı Problemi: Polis Devriye Araçları İçin Bir Uygulama.
- Eroglu E., (2015). A Single Chinese Postman Problem with Two Objectives, Middle East Technical University, Master Thesis, Industrial Engineering, Ankara, Turkey.

- Eryavuz, M., & Gencer, C. (2001). Araç Rotalama Problemine Ait Bir Uygulama. *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 6(1).
- Euler, L. (1736). Solutio Problematis Ad Geometriam Situs Pertinens. *Comm. Acad. Sci. Imper. Petropol.*, 8, 128-140.
- Falkenauer, E., & Bouffouix, S. (1991, April). A Genetic Algorithm for Job Shop. In *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on* (pp. 824-829). IEEE.
- Fernández, E., Meza, O., Garfinkel, R., & Ortega, M. (2003). On The Undirected Rural Postman Problem: Tight Bounds Based on A New Formulation. *Operations Research*, 51(2), 281-291.
- Fernandes, C. G., Lee, O., & Wakabayashi, Y. (2009). Minimum Cycle Cover and Chinese Postman Problems on Mixed Graphs With Bounded Tree-Width. *Discrete Applied Mathematics*, 157(2), 272-279.
- Findıklı, M.A., Altun, S., Fendoğlu, E. & Söyler H. (2018). Yapay Arı Kolonisi Algoritması ile Gezici Robotun Haritalanmış Sahada Gezinme Maliyetinin Optimizasyonu. ISAS 2018-Winter 2nd International Symposium on Innovative Approaches in Scientific Studies, Volume 3 (2018), 1029-1033.
- Fisher, M. L., & Jaikumar, R. (1981). A Generalized Assignment Heuristic for Vehicle Routing. *Networks*, 11(2), 109-124.
- Fleury, M. (1883). Deux Problemes De Geometrie De Situation. *Journal de mathematiques elementaires*, 2(2), 257-261.
- Ford, L. R., & Fulkerson, D. R. (1962). Flows in Networks. 1962. *Princeton U. Press, Princeton, NJ*.
- Frederickson, G. N., Hecht, M. S., & Kim, C. E. (1976, October). Approximation Algorithms for Some Routing Problems. In *Foundations of Computer Science, 1976., 17th Annual Symposium on* (pp. 216-227). IEEE.

- Frederickson, G. N. (1979). Approximation Algorithms for Some Postman Problems. *Journal of the ACM (JACM)*, 26(3), 538-554.
- Ghiani, G., & Improta, G. (2000). An Algorithm for The Hierarchical Chinese Postman Problem. *Operations Research Letters*, 26(1), 27-32.
- Ghiani, G., & Laporte, G. (2000). A Branch-And-Cut Algorithm for The Undirected Rural Postman Problem. *Mathematical Programming*, 87(3), 467-481.
- Ghiani, G., Laganà, D., & Musmanno, R. (2006). A Constructive Heuristic for The Undirected Rural Postman Problem. *Computers & operations research*, 33(12), 3450-3457.
- Gillett, B. E., & Miller, L. R. (1974). A Heuristic Algorithm for The Vehicle-Dispatch Problem. *Operations research*, 22(2), 340-349.
- Grötschel M. & Win Z. (1988): "On the Windy Postman Polyhedron". Report No. 75, Schwerpunktprogram der Deutschen Forschungsgemeinschaft, Universität Augsburg, Germany.
- Grötschel, M., & Win, Z. (1992). A Cutting Plane Algorithm for The Windy Postman Problem. *Mathematical Programming*, 55(1-3), 339-358.
- Goldberg, D. E., & Lingle, R. (1985, July). Alleles, Loci, and The Traveling Salesman Problem. In *Proceedings of an international conference on genetic algorithms and their applications* (Vol. 154, pp. 154-159). Lawrence Erlbaum, Hillsdale, NJ.
- Goldberg, D.E.,(1989). "Genetic Algorithms in Search, Optimisation and Machine Learning", Addison- Wesley Publ., Reading.
- Golden, B. L., & Wong, R. T. (1981). Capacitated Arc Routing Problems. *Networks*, 11(3), 305-315.
- Gomory, R. E. (1958). Outline of An Algorithm For Integer Solutions to Linear Programs. *Bulletin of the American Mathematical society*, 64(5), 275-278.

- Gonzalez-Martin, S., Juan, A. A., Riera, D., Elizondo, M. G., & Ramos, J. J. (2018). A Simheuristic Algorithm for Solving The Arc Routing Problem With Stochastic Demands. *Journal of Simulation*, 12(1), 53-66.
- Göksal, F., P., (2010). Eşzamanlı Topla-Dağıt Araç Rotalama Problemi İçin Sezgisel Yaklaşımlar: Genetik Algoritma ve Kuş Sürüsü Eniyileme. Yüksek Lisans Tezi, Gazi Üniversitesi, Endüstri Mühendisliği, Ankara (Türkiye).
- Grötschel, M., & Yuan, Y. X. (2012). Euler, Mei-Ko Kwan, Königsberg, and A Chinese Postman. *Optimization Stories*, 43.
- Guan, M. (1962). Graphic Programming Using Odd and Even Points. *Chinese Math.*, 1, 237-277.
- Guan, M. (1984). On The Windy Postman Problem. *Discrete Applied Mathematics*, 9(1), 41-46.
- Gutin, G., Jones, M., & Wahlström, M. (2015). Structural Parameterizations of The Mixed Chinese Postman Problem. In *Algorithms-ESA 2015* (pp. 668-679). Springer, Berlin, Heidelberg.
- Gutin, G., Muciaccia, G., & Yeo, A. (2013). Parameterized Complexity of K-Chinese Postman Problem. *Theoretical Computer Science*, 513, 124-128.
- Gutin, G., Jones, M., Sheng, B., & Wahlström, M. (2014, June). Parameterized Directed k -Chinese Postman Problem and k Arc-Disjoint Cycles Problem on Euler Digraphs. In *International Workshop on Graph-Theoretic Concepts in Computer Science* (pp. 250-262). Springer, Cham.
- Gutin, G., Jones, M., & Sheng, B. (2014, September). Parameterized Complexity of The k -Arc Chinese Postman Problem. In *European Symposium on Algorithms* (pp. 530-541). Springer, Berlin, Heidelberg.
- Holland, J. H., (1975). Adaptation in Natural And Artificial Systems. An Introductory Analysis With Application to Biology, Control, And Artificial Intelligence. Ann Arbor, MI: University of Michigan Press.

- Helsgaun, K., (2000). An Effective Implementation of The Lin–Kernighan Traveling Salesman Heuristic. *European Journal of Operational Research*, 126(1), 106-130.
- Hertz, A., & Mittaz, M. (2001). A Variable Neighborhood Descent Algorithm for The Undirected Capacitated Arc Routing Problem. *Transportation science*, 35(4), 425-434.
- Hierholzer, C., & Wiener, C. (1873). Über Die Möglichkeit, Einen Linienzug Ohne Wiederholung Und Ohne Unterbrechung Zu Umfahren. *Mathematische Annalen*, 6(1), 30-32.
- Hua, J., & Li-shan, K. (2003). Genetic Algorithm for Chinese Postman Problems. *Wuhan University Journal of Natural Sciences*, 8(1), 316-318.
- Jang, J. S. R., Sun, C. T., & Mizutani, E. (1997). Neuro-Fuzzy and Soft Computing-A Computational Approach to Learning And Machine Intelligence [Book Review]. *IEEE Transactions on automatic control*, 42(10), 1482-1484.
- Jin, M., Liu, K., & Eksioğlu, B. (2008). A Column Generation Approach for The Split Delivery Vehicle Routing Problem. *Operations Research Letters*, 36(2), 265-270.
- Karaboga, D. (2005). *An Idea Based On Honey Bee Swarm For Numerical Optimization* (Vol. 200). Technical report-tr06, Erciyes university, engineering faculty, computer engineering department.
- Karaboga, D., & Basturk, B. (2008). On The Performance of Artificial Bee Colony (ABC) Algorithm. *Applied soft computing*, 8(1), 687-697.
- Karaboga, D., & Akay, B. (2009). A Comparative Study of Artificial Bee Colony Algorithm. *Applied mathematics and computation*, 214(1), 108-132.
- Karaboğa D., (2014). *Yapay Zekâ Optimizasyon Algoritmaları* (3. Basım), Nobel Yayıncılık, Ankara.
- Kartal, B. (2015). *Yapay Arı Kolonisi Algoritması ile Finansal Portföy Optimizasyonu* (Yayınlanmış Doktora Tezi), İstanbul Üniversitesi Sosyal Bilimler Enstitüsü, İşletme Anabilim Dalı Sayısal Yöntemler Bilim Dalı, İstanbul, Türkiye.

Keskintürk, T., (2009). Araç Rotalama Problemlerinin Global Karınca Koloni Optimizasyonu İle Çözümü, Doktora Tezi, İstanbul Üniversitesi, İşletme Anabilim Dalı, Sayısal Yöntemler Bilim Dalı, İstanbul (Türkiye).

Keskintürk, T., Topuk, N., & Özyeşil, O. (2015). Araç Rotalama Problemleri ve Çözüm Yöntemleri. *The Journal of Business Science*, 3(2), 77-107.

Keskintürk, T., Kiremitci, B., & Kiremitci, S. (2016). 2-Opt Algoritması ve Başlangıç Çözümünün Algoritma Sonuçları Üzerindeki Etkisi. *Journal of Industrial Engineering (Turkish Chamber of Mechanical Engineers)*, 27(3).

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P., (1983). Optimization by Simmulated Annealing. *science*, 220(4598), 671-680.

Kirlik, G., (2009). Yeni Bir Kapasiteli Ayırıt Rotalama Problemi ve Çözüm Yaklaşımları. Yüksek Lisans Tezi, Endüstri Mühendisliği Anabilim Dalı, Eskişehir Üniversitesi, Eskişehir (Türkiye).

Kirlik, G., & Sipahioglu, A. (2012). Capacitated Arc Routing Problem with Deadheading Demands. *Computers & Operations Research*, 39(10), 2380-2394.

Korteweg, P., & Volgenant, T. (2006). On The Hierarchical Chinese Postman Problem With Linear Ordered Classes. *European Journal of Operational Research*, 169(1), 41-52.

Kosif, B., & Ekmekçi, İ. (2012). Araç Rotalama Sistemleri ve Tasarruf Algoritması Uygulaması. İstanbul Üniversitesi, Fen bilimleri Dergisi, Sayı:21, s.41-51.

Kramer, O. (2017). *Genetic Algorithm Essentials* (Vol. 679). Springer.

Kurul, F. C., (2013) Araç Rotalama Problemi, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi, Matematik Mühendisliği Anabilim Dalı, İstanbul (Türkiye)..

Kuzu, S., Önay, O., Şen, U., Tunçer, M., Yıldırım, B. F., & Keskintürk, T. (2014). Gezgin Satıcı Problemlerinin Metasezgiseller İle Çözümü. *Istanbul University Journal of the School of Business*, 43(1), 1-27.

- Küçüksille, E. U., & Tokmak, M. (2011). Yapay Arı Kolonisi Algoritması Kullanarak Otomatik Ders Çizelgeleme. *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 15(3).
- Laporte, G., (1992). The Traveling Salesman Problem: An Overview of Exact and Approximate Algorithms. *European Journal of Operational Research*, 59(2), 231-247.
- Laporte, G. (1997). Modeling and Solving Several Classes of Arc Routing Problems as Traveling Salesman Problems. *Computers & operations research*, 24(11), 1057-1061.
- Laporte, G. (2013). Chapter 3: The Undirected Chinese Postman Problem. In *Arc routing: problems, methods, and applications*(pp. 53-64). Society for Industrial and Applied Mathematics.
- Limon, Y., (2015). On The Balanced K-Chinese Postmen Problems, Middle East Technical University, Master Thesis, Industrial Engineering, Ankara, Turkey.
- Liu, D. Q. (1988). Snow Clearing Vehicle Routing: The Postman Problem in A Hierarchical Network.
- Machado, P., Tavares, J., Pereira, F. B., & Costa, E. (2002, July). Vehicle Routing Problem: Doing it the evolutionary way. In *Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation* (pp. 690-690). Morgan Kaufmann Publishers Inc.
- Maniezzo, V. (2004). Algorithms for Large Directed CARP Instances: Urban Solid Waste Collection Operational Support. *UBLCS Technical Report Series, Bolonha, Italy: University of Bolonha*, 27.
- Michalewicz, Z. (1996). Gas: What Are They?. In *Genetic Algorithms+ Data Structures= Evolution Programs* (pp. 13-30). Springer Berlin Heidelberg.
- Minieka, E. (1979). The Chinese Postman Problem for Mixed Networks. *Management Science*, 25(7), 643-648.
- Mitra, D., Ghoshal, S., Rahaman, H., Chakrabarty, K., & Bhattacharya, B. B. (2010, December). Testing of Digital Microfluidic Biochips Using Improved Eulerization

Techniques and The Chinese Postman Problem. In *2010 19th IEEE Asian Test Symposium* (pp. 111-116). IEEE.

Mutlu, M. M., Alver Y., (2015). Ulaşım Planlama Destek Sistemi olarak Coğrafi Bilgi Sistemleri ve Ege Üniversitesi Ulaşım Ana Planı Örneği, TMMOB İnşaat Mühendisleri Odası, 11. Ulaştırma Kongresi.

Mutlu M. M., (2015). Üniversite Yerleşkesi Ulaşım Ana Planı Hazırlanması: Ege Üniversitesi Örneği, Yüksek Lisans Tezi, Ege Üniversitesi, İnşaat Mühendisliği Anabilim Dalı, İzmir (Türkiye).

Nabiyev N., V. (2010). “Yapay Zeka, İnsan – Bilgisayar Etkileşimi, 3.Basım, Seçkin Yayıncılık, , Ankara.

Nobert, Y. (1991). An Optimal Algorithm for The Mixed Chinese Postman Problem. *Centre De Recherche Sur Les Transports Publication*, (799).

Nobert, Y., & Picard, J. C. (1996). An Optimal Algorithm for The Mixed Chinese Postman Problem. *Networks: An International Journal*, 27(2), 95-108.

Oliver, I. M., Smith, D., & Holland, J. R. (1987). Study of Permutation Crossover Operators on The Traveling Salesman Problem. In *Genetic algorithms and their applications: proceedings of the second International Conference on Genetic Algorithms: July 28-31, 1987 at the Massachusetts Institute of Technology, Cambridge, MA*. Hillsdale, NJ: L. Erlbaum Associates, 1987..

Orloff, C. S. (1974). A Fundamental Problem in Vehicle Routing. *Networks*, 4(1), 35-64.

Osman, I. H., & Laporte, G. (1996). Metaheuristics: A Bibliography.

Osterhues, A., & Mariak, F. (2005). On Variants of The K-Chinese Postman Problem. *Operations Research und Wirtschaftinformatik*, (30).

Önder E., (2011). “Araç Rotalama Probleminin Parçacık Sürü ve Genetik Algoritma ile Optimizasyonu”, Doktora Tezi, İstanbul Üniversitesi, İşletme Anabilim Dalı, İstanbul (Türkiye).

- Özkan, S., (2010). Gezin Satıcı Probleminin Çözümüne Yönelik Algoritmik Yaklaşımlar. Yüksek Lisans Tezi, Gazi Üniversitesi, Endüstri Mühendisliği, Ankara (Türkiye).
- Öztürk, A., (2005). Yöneylem Araştırması, Ekin Kitabevi (Onuncu Baskı), Başak Matbaacılık, Ankara.
- Padberg, M., & Rinaldi, G. (1987). Optimization of A 532-City Symmetric Traveling Salesman Problem by Branch and Cut. *Operations Research Letters*, 6(1), 1-7.
- Padberg, M., & Rinaldi, G. (1991). A Branch-And-Cut Algorithm for The Resolution of Large-Scale Symmetric Traveling Salesman Problems. *SIAM review*, 33(1), 60-100.
- Padungwech, W. (2018). *Heuristic Algorithms for Dynamic Capacitated Arc Routing* (Doctoral dissertation, Cardiff University).
- Palma, G. (2011). A Tabu Search Heuristic for The Prize-Collecting Rural Postman Problem. *Electronic Notes in Theoretical Computer Science*, 281, 85-100.
- Papadimitriou, C. H. (1976). On The Complexity of Edge Traversing. *Journal of the ACM (JACM)*, 23(3), 544-554.
- Parlaktuna, O., Sipahioglu, A., Kirlik, G., & Yazici, A. (2009, July). Multi-Robot Sensor-Based Coverage Path Planning Using Capacitated Arc Routing Approach. In *Control Applications,(CCA) & Intelligent Control,(ISIC), 2009 IEEE* (pp. 1146-1151). IEEE.
- Pearn, W. L. (1991). Augment-Insert Algorithms for The Capacitated Arc Routing Problem. *Computers & Operations Research*, 18(2), 189-198.
- Pearn, W. L. (1994). Solvable Cases of The K-Person Chinese Postman Problem. *Operations Research Letters*, 16(4), 241-244.
- Pearn, W. L., & Wu, T. C. (1995). Algorithms for The Rural Postman Problem. *Computers & Operations Research*, 22(8), 819-828.
- Pearn, W. L., & Liu, C. M. (1995). Algorithms for The Chinese Postman Problem on Mixed Networks. *Computers & operations research*, 22(5), 479-489.

- Pearn, W. L., & Chou, J. B. (1999). Improved Solutions for The Chinese Postman Problem On Mixed Networks. *Computers & Operations Research*, 26(8), 819-827.
- Prins, C. (2013). Chapter 7: The Capacitated Arc Routing Problem: Heuristics. In *Arc routing: problems, methods, and applications* (pp. 131-157). Society for Industrial and Applied Mathematics.
- Ralphs, T. K. (1993). On The Mixed Chinese Postman Problem. *Oper. Res. Lett.*, 14(3), 123-127.
- Rechenberg, I. (1973). *Evolutionsstrategie--Optimierung Technischer Systeme Nach Prinzipien Der Biologischen Evolution*.
- Rego, C. (2001). Node-Ejection Chains for The Vehicle Routing Problem: Sequential and parallel algorithms. *Parallel Computing*, 27(3), 201-222.
- Renaud, J., Boctor, F. F., & Laporte, G. (1996). An Improved Petal Heuristic for The Vehicle Routing Problem. *Journal of the operational Research Society*, 47(2), 329-336.
- Rodrigues, A. M., & Ferreira, J. S. (2001, July). Solving The Rural Postman Problem by Memetic Algorithms. In *Proceedings of the 4th Metaheuristic International Conference (MIC'2001), Porto, Portugal* (pp. 679-684).
- Rousseau, L. M., Gendreau, M., & Feillet, D. (2007). Interior Point Stabilization for Column Generation. *Operations Research Letters*, 35(5), 660-668.
- Saoub, K. R. (2017). *A Tour Through Graph Theory*. Chapman and Hall/CRC.
- Sebő, A. (1990). Undirected Distances and The Postman-Structure Of Graphs. *Journal of Combinatorial Theory, Series B*, 49(1), 10-39.
- Söyler H., ve Keskinürk T., (2007). Karınca Kolonisi Algoritması ile Gezen Satıcı Probleminin Çözümü, 8. Türkiye Ekonometri ve İstatistik Kongresi, İnönü üniversitesi, Malatya.
- Söyler, H., & Fendoğlu, E. (2018). Malatya Büyükşehir Belediyesi İlaçlama Araçlarının Güzergâhlarının Optimizasyonu. *Alphanumeric Journal*, 6(1), 13-24.

Şahin, Y., & Erođlu, A. (2014). Kapasite Kısıtlı Araç Rotalama Problemi İçin Metasezgisel Yöntemler: Bilimsel Yazın Taraması. *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 19(4), 337-355.

Taha, H. A., (2005), Yöneylem Araştırması, çev. Ş. Alp Baray & Şakir Esnaf, (6. Basımdan Çeviri), Literatür Yayıncılık. İstanbul.

Taşkın, Ç., & Emel, G. G. (2009). *Sayısal Yöntemlerde Genetik Algoritmalar*. Alfa Aktüel.

Tecim, V. (2008). *Coğrafi Bilgi Sistemleri: Harita Tabanlı Bilgi Yönetimi* (1.Basım), Renk Form Ofset Matbaacılık, Ankara.

Tereshko, V. (2000). Reaction-Diffusion Model of A Honeybee Colony's Foraging Behaviour, M. Schoenauer, Et Al, Eds., Parallel Problem Solving From Nature VI (Lecture Notes in Computer Science, Vol. 1917) SpringerVerlag: Berlin, p. 807-816.

Thimbleby, H. (2003). The Directed Chinese Postman Problem. *Software: Practice and Experience*, 33(11), 1081-1096.

Thomassen, C. (1997). On The Complexity of Finding A Minimum Cycle Cover of A Graph. *SIAM Journal on Computing*, 26(3), 675-677.

Tokmak, M. (2011). Yapay Arı Kolonisi Algoritması ile Ders Çizelgeleme Probleminin Çözümü, Yüksek Lisans Tezi, Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü, Elektronik Bilgisayar Eğitimi Anabilim Dalı, Isparta (Türkiye).

Toth, P., & Vigo, D. (2002a). An Overview of Vehicle Routing Problems. In *The vehicle routing problem* (pp. 1-26). Society for Industrial and Applied Mathematics.

Win, Z. (1989). On The Windy Postman Problem on Eulerian Graphs. *Mathematical Programming*, 44(1-3), 97-112.

Yaxiong, L., & Yongchang, Z. (1988). A New Algorithm for The Directed Chinese Postman Problem. *Computers & operations research*, 15(6), 577-584.

Yazici, A., Kirlik, G., Parlaktuna, O., & Sipahioglu, A. (2009, October). A Dynamic Path Planning Approach for Multi-Robot Sensor-Based Coverage Considering Energy

Constraints. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on* (pp. 5930-5935). IEEE.

Yüksel O., (2013). Çizge ve Serim Algoritmalar Uygulamalar, Ege Üniversitesi Yayınları Mühendislik Fakültesi Yayın No:57, İzmir.

Zhang, L. (1992). Polynomial Algorithms for The K-Chinese Postman Problem. *IFIP Trans. A, Comput. Sci. Technol.*, 12, 430-435.

<https://images.google.com/> , Search Term: İl- Star İlaçlama Makinası, Erişim Tarihi: 09.12.2017.

http://makina.il-ser.com.tr/TR/static_pages/89/il-star_18_ulv_jeneratoru, Erişim Tarihi: 09.12.2017.

<https://images.google.com/> , Search Term: Malatya Büyükşehir Belediyesi İlaçlama Araçları, Erişim Tarihi: 09.12.2017.

<https://www.arvento.com/tr/ arac-takip-nedir>, Erişim Tarihi:10.12.2017.

<https://www.arvento.com/tr/neden-arac-takip-sistemi>, Erişim Tarihi:10.12.2017.

<https://qgis.org/tr/site/about/index.html>, Erişim Tarihi:20.06.2017.

<https://www.mathworks.com/help/matlab/ref/graph.shortestpath.html#buofmw7-1-Method> , Erişim Tarihi: 24/09/2018.

EKLER

EK – 1: “KERNEK” Mahallesi Cadde/Sokak Dügümler Arası Uzakhlıkları

(Kernek Mahallesi 71 düğüm, 92 ayrıttan oluşmaktadır)

Başlangıç ID	Bitiş ID	Uzakhlık (m)	Başlangıç ID	Bitiş ID	Uzakhlık (m)
182	183	106,456023	7448	7450	200,064961
182	7412	97,812586	7448	7451	76,641655
182	7837	79,58398	7449	7450	88,735191
183	271	102,666409	7450	12279	38,836878
183	7838	102,258742	7671	7672	107,833763
183	7839	79,461356	7672	7993	19,688219
271	7412	115,935022	7672	12121	215,148384
271	7449	99,512553	7776	12279	30,303935
393	394	70,128354	7837	7838	117,887545
394	2272	65,036354	7838	7840	74,874873
394	7776	78,001507	7839	7840	115,580405
1879	1881	52,412735	7839	7932	10,68474
1879	8064	12,517577	7932	7933	104,501301
1881	8065	60,018766	7933	15118	153,127295
2910	2911	36,149378	7993	7994	115,349777
2910	7671	19,194097	7994	12123	10,268139
2911	8061	29,551037	8054	8055	92,018012
3733	3734	49,968495	8054	8065	31,104164
3733	7411	36,777508	8054	8072	58,078345
3733	12279	189,883065	8055	11521	58,476533
3734	7449	11,463717	8061	8062	47,707049
3734	7932	133,128855	8062	8064	17,912479
4351	4352	11,946631	8063	8064	58,405071
4351	4354	46,7722	8063	8066	112,803751
4351	8363	42,40385	8065	8066	59,016582
4352	4353	63,956763	8066	12121	60,5808
4352	7933	57,623456	8072	8363	60,648419
4353	7840	79,677244	8363	12121	24,44912
4353	7993	216,513394	10052	10053	5,555218
4354	8055	108,622417	10052	12122	65,632799
4354	12122	104,983071	10052	15118	12,461408
6496	6497	19,930677	10053	10054	13,122882
6496	6498	21,1164	10053	12992	68,2231
6497	10719	68,202873	10054	13398	82,708572
6498	7837	56,765964	10054	15118	5,669335
7411	10054	22,508508	10719	12123	123,791498
7411	15118	22,304641	11048	11049	94,493324
7412	7448	67,840554	11048	11521	69,704582
7442	7443	87,330618	11049	11050	79,117573
7442	7451	75,662849	11050	12992	9,298701
7442	7776	99,922155	11521	12122	117,68583
7443	7446	41,858873	12992	13398	13,038882
7443	12247	135,889075	13303	14206	41,961383
7446	7447	58,288556	14206	15652	44,086952
7446	15652	203,452526	15178	15652	173,122421
7447	7451	20,499503			
7447	15012	67,580727			

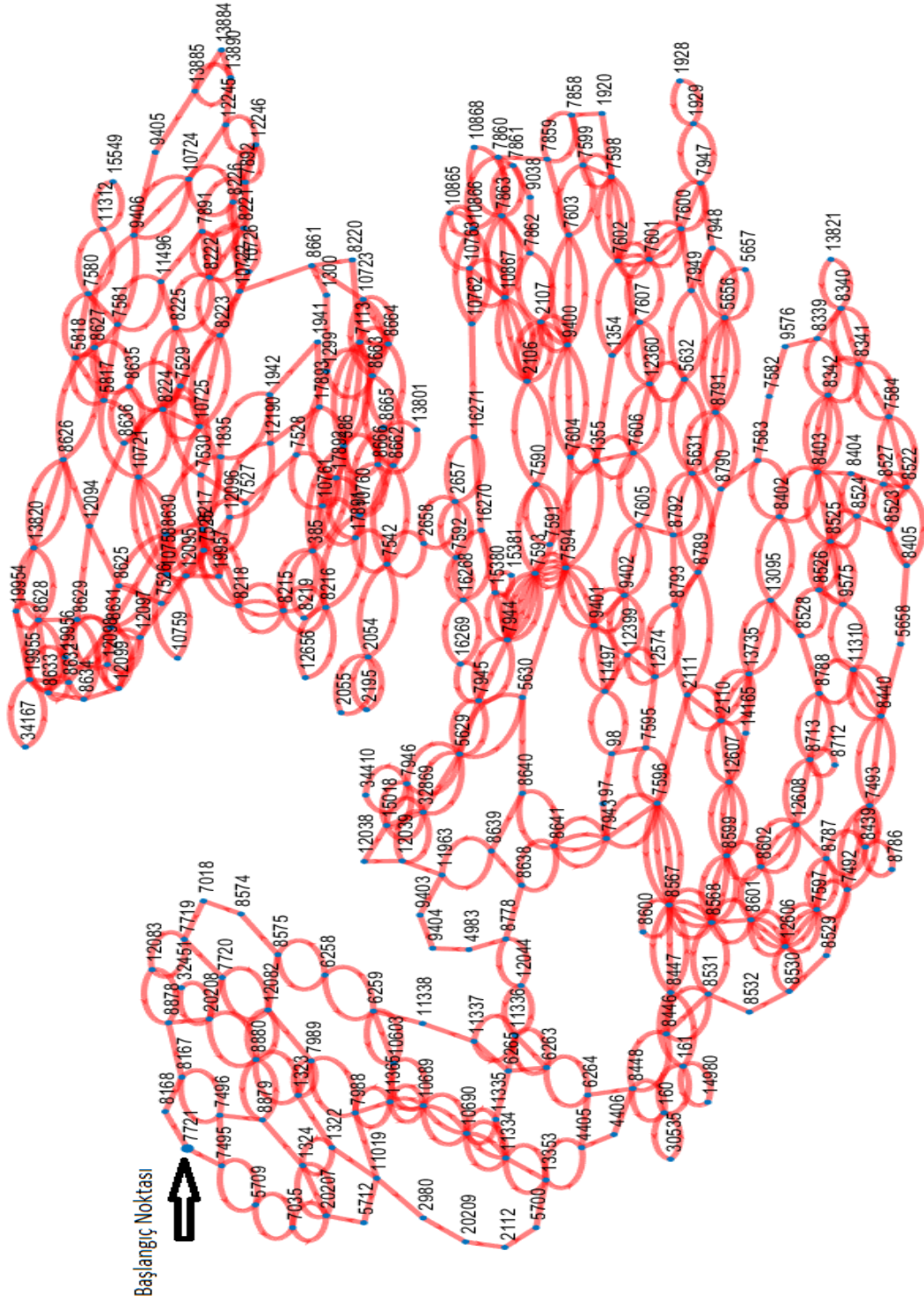
EK – 2: Bölge 12 ve Bölge 34' ün Rotaları

Bölge 12: Bentbaşı – Kıltepe - Yeşilkaynak Alanının ID' li Rotası;

7721	7495	5709	7035	5709	7495	7496	8167	8878	12083	7719	12083
	8878	20208	7720	12082	8575	6258	8575	8574	7018	7719	7720
	12082	7989	7988	7989	1323	7989	1323	1322	11019	2980	20209
	2112	5700	13353	11334	13353	4405	13353	11334	11335	10690	11334
	10690	11334	10690	10689	10690	10689	10603	6259	6258	6259	10603
	10689	10690	11335	6265	6263	11336	12044	8778	4983	9404	9403
	11963	8639	11963	12039	32869	15018	32869	15018	7946	34410	7946
	15018	12038	12039	32869	5629	32869	5629	7946	5629	7945	16269
	16268	16270	16268	7592	16268	16269	7945	7944	15380	15381	15380
	7944	7945	5629	7945	7944	7593	7944	7593	7944	7593	7944
	15380	16270	16271	2657	7592	2657	2658	7542	17891	17892	17893
	12190	12096	19957	7525	19957	12095	12097	12098	12099	12098	8629
	12098	12097	8625	12097	12099	8631	19956	19955	34167	19955	19956
	8631	12099	8634	8633	8632	19955	19954	8628	19954	13820	8628
	8633	8628	8633	8632	8631	8630	8631	8630	1835	8630	12095
	12096	12190	1942	1835	1942	1941	17893	17892	10761	17892	17891
	10761	7528	7527	7525	7527	7530	7529	8635	8636	8635	8627
	9406	10724	9406	10724	8226	10722	10726	8221	10726	8223	10725
	8224	10725	8223	10725	8217	10758	10759	10758	10721	10758	10721
	10758	8217	10758	8217	8218	8217	8218	8215	8218	8215	8216
	12656	8216	10760	8666	13801	8666	386	8666	10760	8665	10760
	8216	8215	385	8219	385	8215	8218	8217	10725	8224	10721
	5817	10721	8224	8225	8224	8225	8222	8223	8222	8221	8222
	8225	11496	7581	7580	11312	15549	11312	7580	5818	7580	5818
	5817	5818	5817	7581	11496	7891	8222	7891	8222	8221	8226
	12246	12245	12246	7892	12246	7892	7891	7892	8221	8226	10724
	12245	13890	13885	13890	13884	13885	9405	9406	8627	8635	7529
	8636	12094	8629	8634	8629	8628	13820	8626	13820	8626	8627
	8626	12094	8625	7526	7525	7530	7529	10722	8661	10723	8664
	10723	8663	10723	8664	8665	8662	8663	1299	8663	8662	8665
	8664	7113	8664	7113	386	7113	386	385	386	385	386
	7113	8220	8661	1300	1299	7528	10761	17891	7542	8662	7542
	2054	2195	2054	2055	2054	7542	2658	2657	16271	10762	10866
	10865	10763	10867	10763	10865	10866	7860	10866	10762	10763	10867
	7863	10868	7863	10867	2106	10867	2106	10867	7863	7862	9038

7861	7863	7861	7860	7859	7858	7859	7603	9400	7604	9400
7603	7599	7603	9400	2107	9400	2107	7862	2107	9400	7604
1355	7606	12360	7607	12360	7606	7605	9402	12399	12574	12399
11497	9401	11497	98	11497	9401	9402	9401	7594	9401	7594
9401	9402	7605	9402	12399	12574	8793	8792	5631	8791	8790
8791	5656	8791	5656	7948	7949	7600	7947	7600	7947	1929
1928	1929	7947	7948	5656	5657	5656	8791	5631	8792	7605
7606	1355	7604	7594	7593	7594	7593	7594	7593	7590	7591
7590	2106	2107	2106	2107	2106	7590	7593	7594	1355	7606
12360	5632	7949	5632	12360	7607	7601	7602	7601	7602	7598
7602	7598	7599	7598	7599	7858	1920	7598	7602	7601	7600
7601	7600	7601	7607	1354	7602	1354	1355	7594	5630	5629
5630	8640	8639	8638	8641	7943	8641	7943	7596	8567	8600
8567	8568	8601	12606	8601	12606	8601	12606	8530	12606	7597
12606	7597	12606	7597	8787	8439	8786	8439	7493	8440	11310
9575	11310	8788	11310	8440	11310	8440	7493	8439	8787	7597
8787	12608	8713	12608	8602	12608	8602	8601	8568	8601	8602
12608	8713	8788	8713	8712	8713	8788	8528	8526	9575	8525
9575	8525	8526	8525	8524	8525	8403	8525	8524	8523	8527
8522	8527	8523	8405	8522	8527	8404	8524	8404	8403	8525
8526	8528	13095	13735	13095	8402	8403	8342	8403	8342	8341
8340	13821	8340	8341	8342	8339	8342	8403	8402	7583	8402
13095	13735	2110	13735	2110	13735	14165	12607	8599	12607	8599
8602	8599	8568	8599	8568	8567	8568	8567	8447	8567	7596
7943	7596	8567	8568	8531	8446	8448	6264	6263	6264	4405
4406	8448	160	30535	160	8448	8446	8531	8532	8530	8529
7492	7597	7492	7493	7492	7493	8440	5658	8405	8522	7584
8522	7584	8341	7584	8341	8340	8339	9576	7582	7583	8790
8789	5631	5632	5631	8789	2111	8789	8793	12574	7595	7596
8567	8447	161	14980	161	160	161	8447	8446	8531	8568
8599	12607	2110	12607	2110	2111	2110	2111	7596	7595	98
97	7943	8641	8640	8641	8638	8778	12044	11336	11337	11336
6263	11336	6265	11337	11338	6259	10603	11365	10689	11365	10689
11365	7988	11365	7988	11019	5712	20207	7035	20207	1324	20207
1324	1322	1324	8879	8880	20208	32451	20208	8880	12082	8880
1323	8880	8879	7496	8167	8168	7721				

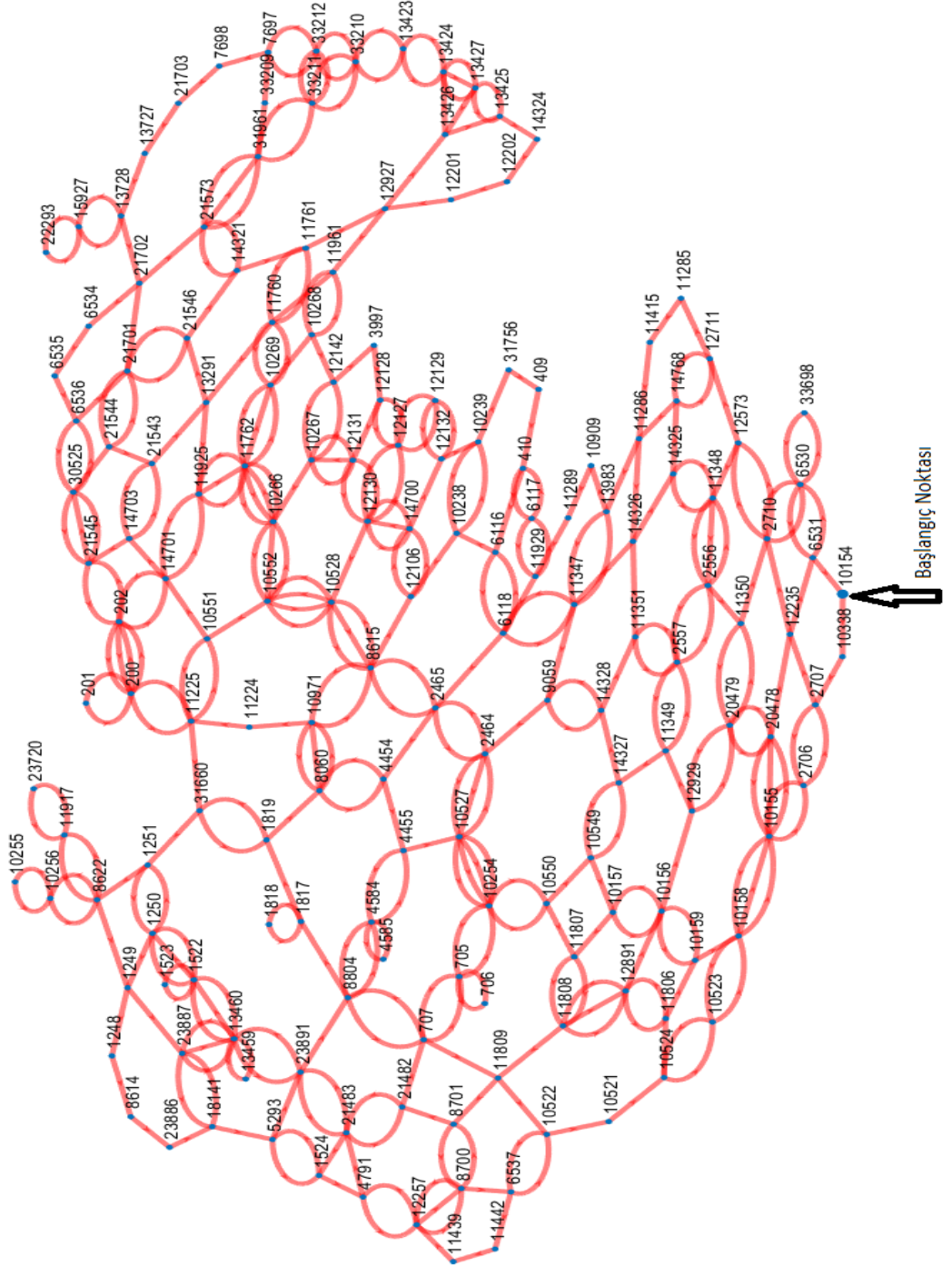
Bölge 12: Bentbaşı – Kıltepe - Yeşilkaynak Alanının Rotasının Programdan Çıkan Hareketli Gezinme Sonucu



Bölge 34: Şeyhbayram – Abdulgaffar Alanının ID'li Rotası;

10154	10338	2707	2706	2707	12235	2710	12573	11348	2556	11348	2556
	2557	2556	11350	2710	6530	2710	12573	12711	11285	11415	11286
	13983	10909	11289	11929	6117	410	6117	11929	6118	11347	6118
	6116	10238	12106	14700	12130	14700	12130	12131	12130	12127	12130
	10528	10552	10528	10552	10528	8615	10971	8060	4454	8060	10971
	8615	10528	8615	2465	8615	12106	14700	12132	12129	12127	12128
	12127	12129	12132	10239	10238	10239	31756	409	410	6116	6118
	2465	2464	2465	4454	4455	4584	8804	707	705	706	705
	707	8804	1817	1818	1817	1819	8060	10971	11224	11225	10551
	14701	11925	14701	202	21545	30525	21545	30525	6536	30525	21544
	21701	21546	21701	21702	21573	31961	33211	33212	33210	33212	7697
	33212	33211	33210	13423	13424	13427	13425	13427	13424	13427	13426
	13425	14324	12202	12201	12927	13426	13424	13423	33210	33211	31961
	21573	14321	21573	31961	33209	7697	7698	21703	13727	13728	15927
	22293	15927	13728	21702	6534	6535	6536	21701	21544	21543	14703
	21545	202	14701	14703	21543	13291	21546	14321	11761	12927	11961
	10268	11961	11760	11761	11760	13291	11925	11762	11925	11762	10269
	11762	10266	11762	10266	10552	10266	11762	10269	11760	10269	10268
	12142	10267	12142	3997	12128	12131	10267	12131	10267	10266	10552
	10551	11225	200	202	200	202	200	201	200	11225	31660
	1819	31660	1251	8622	11917	23720	11917	8622	10256	10255	10256
	8622	1249	23887	18141	23887	13460	23891	21483	21482	21483	23891
	13460	23887	13460	13459	13460	1522	13460	1522	1523	1522	1250
	1522	1250	1251	1250	1249	1248	8614	23886	18141	5293	23891
	8804	4584	4585	4584	4455	10527	10254	10527	2464	10527	10254
	10550	11807	11808	12891	11808	12891	11806	12891	10156	12929	20479
	20478	10155	20478	20479	11350	20479	12929	11349	14327	10549	14327
	14328	9059	14328	11351	2557	11349	2557	11351	14326	14325	11348
	14325	14768	12711	14768	11286	14326	11347	13983	11347	9059	2464
	10527	10254	705	10254	10550	10549	10157	11807	11808	11809	8701
	21482	707	11809	10522	6537	11442	11439	12257	8700	12257	4791
	21483	1524	5293	1524	4791	12257	8700	8701	8700	6537	10522
	10521	10524	10523	10158	10523	10524	11806	10159	10156	10157	10156
	10159	10158	10155	10158	10155	2706	10155	20478	12235	6531	6530
	33698	6530	6531	10154							

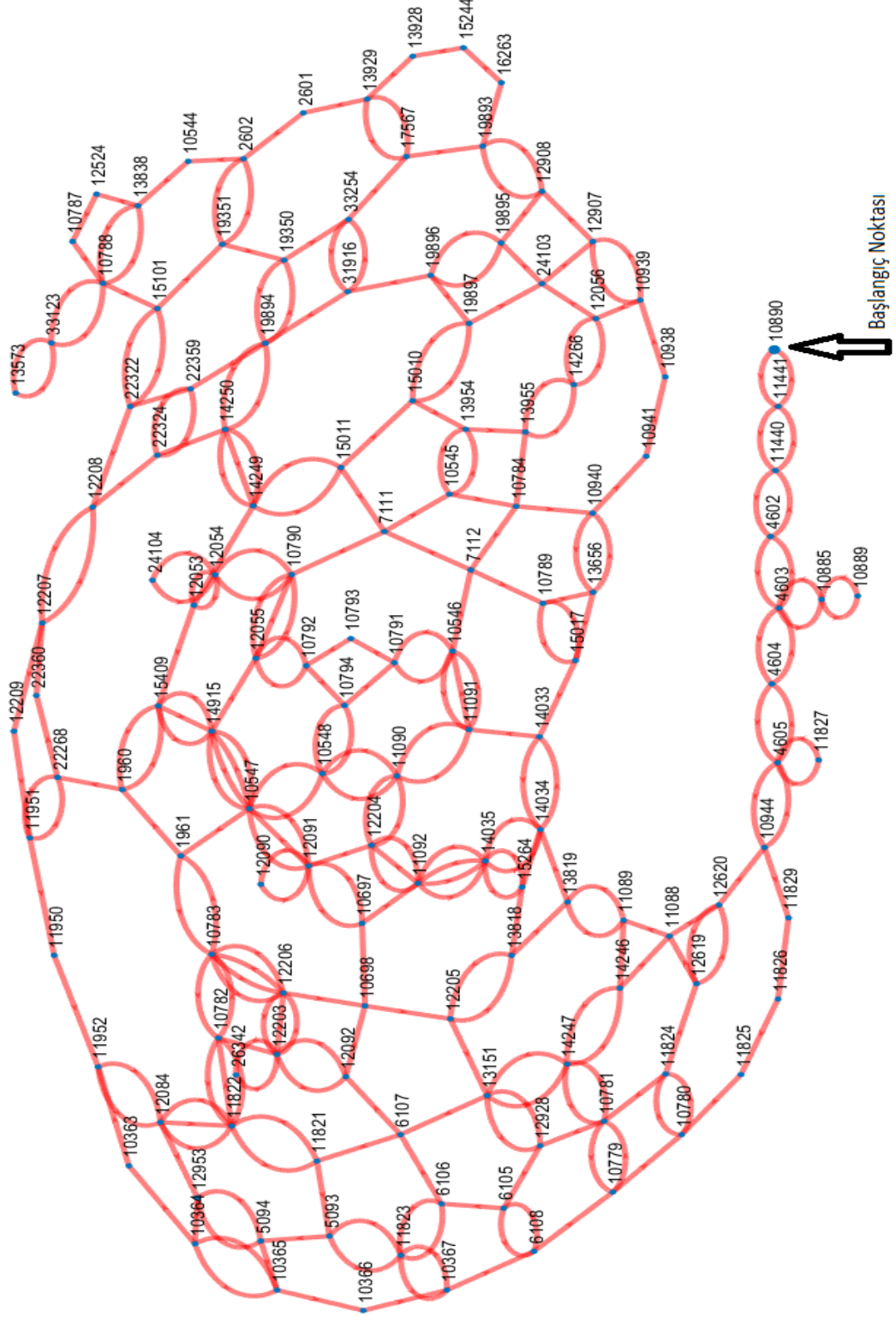
Bölge 34: Şeyhbayram – Abdulgaffar Alanının Rotasının Programdan Çıkan Hareketli Gezinme Sonucu



Bölge 34: Turgut Özal Alanının ID'li Rotası;

10890	11441	11440	4602	4603	4604	4605	10944	11829	11826	11825	10780
	11824	10781	10779	10780	11824	12619	11088	11089	13819	14034	14033
	11091	11090	10548	10794	10548	11090	11091	10546	11091	10546	10791
	10546	7112	7111	10790	12054	12053	15409	1960	22268	11951	22268
	22360	12207	12208	12207	12209	11951	11950	11952	12084	12953	5094
	12953	10364	10365	10364	10363	11952	12084	11822	12084	11822	11821
	11822	10782	11822	10782	12203	26342	12203	12206	12203	12092	12203
	12206	10783	12206	10783	10782	10783	1961	10783	12206	10698	12205
	13151	6107	11821	5093	11823	10367	11823	6106	11823	5093	5094
	10365	10366	10367	6108	10779	10781	12928	6105	6108	6105	6106
	6107	12092	10698	10697	12091	10547	12091	10547	10548	10547	1961
	1960	15409	14915	15409	14915	10547	14915	10547	14915	12055	10792
	10794	10791	10793	10792	12055	10790	12055	10790	12054	24104	12054
	12053	14249	15011	15010	19897	24103	12907	10939	12056	14266	13955
	14266	12056	24103	19895	19896	31916	33254	31916	19894	19350	19894
	22359	22324	22359	22322	15101	19351	2602	19351	19350	33254	17567
	13929	17567	19893	12908	19895	19896	19897	15010	13954	10545	13954
	13955	10784	10940	13656	15017	10789	13656	10940	10941	10938	10939
	12907	12908	19893	16263	15244	13928	13929	2601	2602	10544	13838
	10788	33123	13573	33123	10788	13838	12524	10787	10788	15101	22322
	12208	22324	14250	19894	14250	14249	14250	14249	15011	7111	10545
	10784	7112	10789	15017	14033	14034	14035	15264	14035	11092	14035
	11092	12204	11092	12204	11090	12204	12091	12090	12091	10697	11092
	14035	14034	15264	13818	12205	13818	13819	11089	14246	14247	13151
	12928	13151	14247	10781	14247	14246	11088	12620	12619	12620	10944
	4605	11827	4605	4604	4603	10885	10889	10885	4603	4602	11440
	11441	10890									

Bölge 34: Turgut Özal Alanının Rotasının Programdan Çıkan Hareketli Gezinme Sonucu



**EK – 3: Çiqli İlaçlama Araçlarının ARVENTO Programından Alınan 14 – 19
Ağustos 2017 tarihli Mevcut Gezinme Raporları**

	Toplam Aldığı Mesafe (km)	Mahalle(ler) içi Aldığı Mesafe(km)	İlk Mesafe Sayacı Değeri Tarihi	Son Mesafe Sayacı Değeri Tarihi	İlaçlamaya başlağı İlk saat	İlaçlamayı bitirdiği son saat
1. EKİP (651) (276)						
PAZARTESİ						
ARBAÇ HIÇ ÇALIŞMAMIŞ						
SAU						
ÇAŞSAMBBA	59,17	32,74	16 Ağustos 2017 Çarşamba 17:12:31	16 Ağustos 2017 Çarşamba 22:59:35	16 Ağustos 2017 Çarşamba 18:09:05	16 Ağustos 2017 Çarşamba 22:55:13
PERŞEMBE	51,6	23,77	17 Ağustos 2017 Perşembe 17:00:16	17 Ağustos 2017 Perşembe 22:27:12	17 Ağustos 2017 Perşembe 18:31:09	17 Ağustos 2017 Perşembe 22:22:10
CUMA	53,76	25,03	18 Ağustos 2017 Cuma 17:10:35	18 Ağustos 2017 Cuma 22:06:46	18 Ağustos 2017 Cuma 18:36:41	18 Ağustos 2017 Cuma 22:00:39
CUMARTESİ	52,33	19,22	19 Ağustos 2017 Cumartesi 17:00:55	19 Ağustos 2017 Cumartesi 21:26:46	19 Ağustos 2017 Cumartesi 18:44:55	19 Ağustos 2017 Cumartesi 21:15:11
	216,86	100,76				
2. EKİP (647) (272)						
PAZARTESİ	52,36	21,01	14 Ağustos 2017 Pazartesi 17:03:23	14 Ağustos 2017 Pazartesi 22:23:05	14 Ağustos 2017 Pazartesi 19:36:45	14 Ağustos 2017 Pazartesi 22:11:08
SAU	42,69	25,47	15 Ağustos 2017 Salı 17:13:38	15 Ağustos 2017 Salı 21:42:57	15 Ağustos 2017 Salı 18:38:16	15 Ağustos 2017 Salı 21:20:00
ÇAŞSAMBBA	30,8	18,03	16 Ağustos 2017 Çarşamba 17:10:45	16 Ağustos 2017 Çarşamba 20:31:25	16 Ağustos 2017 Çarşamba 18:37:38	16 Ağustos 2017 Çarşamba 20:24:32
PERŞEMBE	52,62	20,93	17 Ağustos 2017 Perşembe 17:03:00	17 Ağustos 2017 Perşembe 21:19:22	17 Ağustos 2017 Perşembe 17:22:27	17 Ağustos 2017 Perşembe 20:15:28
CUMA	53,8	13,44	18 Ağustos 2017 Cuma 17:00:11	18 Ağustos 2017 Cuma 21:16:00	18 Ağustos 2017 Cuma 19:17:18	18 Ağustos 2017 Cuma 20:27:15
CUMARTESİ	34,62	19,11	19 Ağustos 2017 Cumartesi 17:26:45	19 Ağustos 2017 Cumartesi 21:26:09	19 Ağustos 2017 Cumartesi 19:03:14	19 Ağustos 2017 Cumartesi 21:20:45
	266,89	117,99				
3. EKİP (123) (270)						
PAZARTESİ						
SAU						
ÇAŞSAMBBA	40,89	17,32	16 Ağustos 2017 Çarşamba 17:00:11	16 Ağustos 2017 Çarşamba 21:24:46	16 Ağustos 2017 Çarşamba 17:11:57	16 Ağustos 2017 Çarşamba 21:22:55
PERŞEMBE						
CUMA						
CUMARTESİ	16,74		19 Ağustos 2017 Cumartesi 17:02:16	19 Ağustos 2017 Cumartesi 17:29:49	İLAÇLAMA YAPILMAMIŞ	
	57,63	17,32				
4. EKİP (648) (273)						
PAZARTESİ	91,06	25,52	14 Ağustos 2017 Pazartesi 17:01:24	14 Ağustos 2017 Pazartesi 21:36:48	14 Ağustos 2017 Pazartesi 18:05:03	14 Ağustos 2017 Pazartesi 21:14:23
SAU	94,79	26,78	15 Ağustos 2017 Salı 17:00:07	15 Ağustos 2017 Salı 21:34:18	15 Ağustos 2017 Salı 18:45:00	15 Ağustos 2017 Salı 20:59:52
ÇAŞSAMBBA	59,23	16,1	16 Ağustos 2017 Çarşamba 17:00:11	16 Ağustos 2017 Çarşamba 21:37:52	16 Ağustos 2017 Çarşamba 19:49:33	16 Ağustos 2017 Çarşamba 21:34:49
PERŞEMBE	60,81	26,39	17 Ağustos 2017 Perşembe 17:00:18	17 Ağustos 2017 Perşembe 21:36:29	17 Ağustos 2017 Perşembe 18:26:08	17 Ağustos 2017 Perşembe 21:34:04
CUMA	83,47	27,83	18 Ağustos 2017 Cuma 17:00:12	18 Ağustos 2017 Cuma 21:27:25	18 Ağustos 2017 Cuma 19:03:55	18 Ağustos 2017 Cuma 21:12:17
CUMARTESİ	85,24	14,95	19 Ağustos 2017 Cumartesi 17:00:13	19 Ağustos 2017 Cumartesi 21:21:37	19 Ağustos 2017 Cumartesi 19:31:00	19 Ağustos 2017 Cumartesi 21:01:03
	474,6	137,57				

EK – 3 Devamı

5. EKİP (24) (271)		İlk Mesafe Sayarı Değeri Tarihi	Son Mesafe Sayarı Değeri Tarihi	İlaçlamaya başlangıç ilk saat	İlaçlamayı bitirdiği son saat
PAZARTESİ	Şişman - Pağın	14 Ağustos 2017 Pazartesi 17:00:11,14 Ağustos 2017 Pazartesi 18:48:15	14 Ağustos 2017 Pazartesi 21:13:14 Ağustos 2017 Pazartesi 18:48:15	14 Ağustos 2017 Pazartesi 18:48:15	14 Ağustos 2017 Pazartesi 20:44:05
SAU	Karacanak	4,69	15 Ağustos 2017 Salı 17:40:57	15 Ağustos 2017 Salı 18:17:08	15 Ağustos 2017 Salı 21:37:36
ÇARŞAMBA	Sarıy-Keremkİstİkİa-Hacıbol-Şeyhbayram-Zavİye-Celetpaşa-Bostanbasi	54,66	16 Ağustos 2017 Çarşamba 17:11:16 Ağustos 2017 Çarşamba 21:28:16 Ağustos 2017 Çarşamba 18:08:48	16 Ağustos 2017 Çarşamba 18:08:48	16 Ağustos 2017 Çarşamba 21:15:41
PERŞEMBE	K.Hüseynbey-B.Hüseynbey-Yenihamam-Dabakhane-Koyunlu-Çukurdere-Bostanbasi - Gedik	71,51	17 Ağustos 2017 Perşembe 17:01:27 Ağustos 2017 Perşembe 21:24:17 Ağustos 2017 Perşembe 18:07:29	17 Ağustos 2017 Perşembe 18:07:29	17 Ağustos 2017 Perşembe 21:15:43
CUMA	Üpöğar-K.Hüseynbey-Dabakhane-Kavaklıbağ-Niyazi-Şifa-Tecde	57,69	18 Ağustos 2017 Cuma 17:00:11	18 Ağustos 2017 Cuma 21:58:49	18 Ağustos 2017 Cuma 18:14:48
CUMARTESİ	İronu-Çatalper - Ceveipaşa	63,66	19 Ağustos 2017 Cumartesi 17:00:19 Ağustos 2017 Cumartesi 21:29:19 Ağustos 2017 Cumartesi 18:19:01	19 Ağustos 2017 Cumartesi 18:19:01	19 Ağustos 2017 Cumartesi 20:50:30
		373,91			
					116,2
6. EKİP (073) (278)		İlk Mesafe Sayarı Değeri Tarihi	Son Mesafe Sayarı Değeri Tarihi	İlaçlamaya başlangıç ilk saat	İlaçlamayı bitirdiği son saat
PAZARTESİ	Yeni (batağazi) -Kemerköpü	110,82	14 Ağustos 2017 Pazartesi 17:00:14 Ağustos 2017 Pazartesi 21:47:14 Ağustos 2017 Pazartesi 19:06:22	14 Ağustos 2017 Pazartesi 19:06:22	14 Ağustos 2017 Pazartesi 21:23:54
SAU	Alıgar	80,73	15 Ağustos 2017 Salı 17:00:12	15 Ağustos 2017 Salı 21:28:45	15 Ağustos 2017 Salı 18:39:02
ÇARŞAMBA	Dilek-Yaka-Hatunsuyu	88,24	16 Ağustos 2017 Çarşamba 17:00:16 Ağustos 2017 Çarşamba 22:08:16 Ağustos 2017 Çarşamba 18:52:48	16 Ağustos 2017 Çarşamba 18:52:48	16 Ağustos 2017 Çarşamba 21:13:39
PERŞEMBE	Hatunsuyu - Yaka - Alıgar	81,08	17 Ağustos 2017 Perşembe 17:00:17 Ağustos 2017 Perşembe 21:30:17 Ağustos 2017 Perşembe 18:33:50	17 Ağustos 2017 Perşembe 18:33:50	17 Ağustos 2017 Perşembe 20:15:43
CUMA	Dilek	68,63	18 Ağustos 2017 Cuma 17:00:12	18 Ağustos 2017 Cuma 21:27:21	18 Ağustos 2017 Cuma 18:24:02
CUMARTESİ	Dilek - Sütlüce	91,17	19 Ağustos 2017 Cumartesi 17:00:19 Ağustos 2017 Cumartesi 21:54:19 Ağustos 2017 Cumartesi 18:36:56	19 Ağustos 2017 Cumartesi 18:36:56	19 Ağustos 2017 Cumartesi 21:22:24
		515,67			
					187,3
7. EKİP (122) (049)		İlk Mesafe Sayarı Değeri Tarihi	Son Mesafe Sayarı Değeri Tarihi	İlaçlamaya başlangıç ilk saat	İlaçlamayı bitirdiği son saat
PAZARTESİ	Merkez Beşdiği -Yamaç	73,56	14 Ağustos 2017 Pazartesi 17:46:11,14 Ağustos 2017 Pazartesi 23:01:14 Ağustos 2017 Pazartesi 17:54:48	14 Ağustos 2017 Pazartesi 17:54:48	14 Ağustos 2017 Pazartesi 22:26:12
SAU	Firat -Merkez Beşdiği-Yamaç	64,26	15 Ağustos 2017 Salı 17:01:41	15 Ağustos 2017 Salı 21:58:49	15 Ağustos 2017 Salı 18:59:00
ÇARŞAMBA	Zafer -Çöplük-Fırat - K.Hüseynbey - B.Hüseynbey	67,55	16 Ağustos 2017 Çarşamba 17:00:16 Ağustos 2017 Çarşamba 22:30:16 Ağustos 2017 Çarşamba 19:18:12	16 Ağustos 2017 Çarşamba 19:18:12	16 Ağustos 2017 Çarşamba 22:29:24
PERŞEMBE	Üpöğar - K.Hüseynbey - B.Hüseynbey - Akınlar - Halİttin - Çirİkpar - Yenihamam - Dabakhane - Sarıoğlu - Kırcıval - Sancaktar	63,18	17 Ağustos 2017 Perşembe 17:02:17 Ağustos 2017 Perşembe 22:15:17 Ağustos 2017 Perşembe 18:51:31	17 Ağustos 2017 Perşembe 18:51:31	17 Ağustos 2017 Perşembe 21:39:48
CUMA	Sarıoğlu-Halİttin - Cemherade - H.Abdİ - Paşaköşkü -Ataköy -B.mustafapaşa - K.Mustafapaşa - Dabakhane	45,48	18 Ağustos 2017 Cuma 17:00:19	18 Ağustos 2017 Cuma 22:10:44	18 Ağustos 2017 Cuma 17:46:46
CUMARTESİ	Üpöğar - Niyazi - Şişik - Aslanbey - Şifa - İsmetİye - Yenihamam - Dabakhane - Karaklıbağ	47,75	19 Ağustos 2017 Cumartesi 17:00:19 Ağustos 2017 Cumartesi 21:01:19 Ağustos 2017 Cumartesi 18:53:59	19 Ağustos 2017 Cumartesi 18:53:59	19 Ağustos 2017 Cumartesi 21:00:21
		361,78			
					170,06
8. EKİP (52) (279)		İlk Mesafe Sayarı Değeri Tarihi	Son Mesafe Sayarı Değeri Tarihi	İlaçlamaya başlangıç ilk saat	İlaçlamayı bitirdiği son saat
PAZARTESİ	Alacakapı - Dolamantepe - Çolakoğlu	102,63	14 Ağustos 2017 Pazartesi 17:02:24 Ağustos 2017 Pazartesi 23:05:14 Ağustos 2017 Pazartesi 18:53:19	14 Ağustos 2017 Pazartesi 18:53:19	14 Ağustos 2017 Pazartesi 22:45:28
SAU	Bulgurlu	95,08	15 Ağustos 2017 Salı 17:00:46	15 Ağustos 2017 Salı 22:09:52	15 Ağustos 2017 Salı 19:08:30
ÇARŞAMBA	Düncü - Bahri	47,68	23 Ağustos 2017 Çarşamba 18:29:52 Ağustos 2017 Çarşamba 20:52:23 Ağustos 2017 Çarşamba 18:47:07	23 Ağustos 2017 Çarşamba 18:47:07	23 Ağustos 2017 Çarşamba 20:08:32
PERŞEMBE	Çolakoğlu - Topaş - Dolamantepe - Meydanbasi - Karahan	60,42	17 Ağustos 2017 Perşembe 17:06:17 Ağustos 2017 Perşembe 20:37:17 Ağustos 2017 Perşembe 18:48:32	17 Ağustos 2017 Perşembe 18:48:32	17 Ağustos 2017 Perşembe 20:20:35
CUMA	Bulgurlu - Çamlıru	55,2	18 Ağustos 2017 Cuma 17:28:35	18 Ağustos 2017 Cuma 21:01:39	18 Ağustos 2017 Cuma 18:51:20
CUMARTESİ	Dolamantepe - Topaş-Çolakoğlu - Bağıçteİleler -İadıçayır - Alacakapı	59,64	19 Ağustos 2017 Cumartesi 17:01:19 Ağustos 2017 Cumartesi 21:25:19 Ağustos 2017 Cumartesi 19:07:36	19 Ağustos 2017 Cumartesi 19:07:36	19 Ağustos 2017 Cumartesi 21:13:08
		420,65			
					164,05
9. EKİP (62) (277)		İlk Mesafe Sayarı Değeri Tarihi	Son Mesafe Sayarı Değeri Tarihi	İlaçlamaya başlangıç ilk saat	İlaçlamayı bitirdiği son saat
PAZARTESİ	Kerdİnli -Hoca Ahmet Yesevİ - Seyvan	77,85	14 Ağustos 2017 Pazartesi 17:00:14 Ağustos 2017 Pazartesi 21:00:14 Ağustos 2017 Pazartesi 18:44:29	14 Ağustos 2017 Pazartesi 18:44:29	14 Ağustos 2017 Pazartesi 20:21:13
SAU		26,46	15 Ağustos 2017 Salı 17:01:41	15 Ağustos 2017 Salı 17:36:38	15 Ağustos 2017 Salı 17:36:38
ÇARŞAMBA		2,56	16 Ağustos 2017 Çarşamba 17:06:51 Ağustos 2017 Çarşamba 17:43	16 Ağustos 2017 Çarşamba 17:43	16 Ağustos 2017 Çarşamba 17:43
PERŞEMBE		17,12	17 Ağustos 2017 Perşembe 17:01:47 Ağustos 2017 Perşembe 17:22	17 Ağustos 2017 Perşembe 17:22	17 Ağustos 2017 Perşembe 17:22
CUMA		8,21	18 Ağustos 2017 Cuma 17:18:03	18 Ağustos 2017 Cuma 17:21:35	18 Ağustos 2017 Cuma 17:21:35
CUMARTESİ	ARACIÇALŞANMİŞ		18 Ağustos 2017 Cuma 17:21:35	18 Ağustos 2017 Cuma 17:21:35	18 Ağustos 2017 Cuma 17:21:35
		132,2			
					14,02

EK – 3 Devamı

	10. EYLÜL (50) (175)																				
	Yeni Battalgası - Keme Köprü	75,92	21,91	14 Ağustos 2017 Pazartesi 17:00:08	İlk Mesafe Sırayı Dağıtım Tarihi	14 Ağustos 2017 Pazartesi 22:48:53	Son Mesafe Sırayı Dağıtım Tarihi	14 Ağustos 2017 Pazartesi 18:51:18	İlaçlamaya Başlangıç İlişkisi	14 Ağustos 2017 Pazartesi 21:08:07											
PALZARTEŞİ																					
SALI	Yeni Battalgası - Keme Köprü - Hasırcılar - Boren	84,29	34,22	15 Ağustos 2017 Salı 17:00:09	İlk Mesafe Sırayı Dağıtım Tarihi	15 Ağustos 2017 Salı 22:31:45	Son Mesafe Sırayı Dağıtım Tarihi	15 Ağustos 2017 Salı 18:41:38	İlaçlamaya Başlangıç İlişkisi	15 Ağustos 2017 Salı 22:02:46											
ÇARŞAMBA	Hamim Çiftliği - Battalgası	66,97	35,36	16 Ağustos 2017 Çarşamba 17:00:08	İlk Mesafe Sırayı Dağıtım Tarihi	16 Ağustos 2017 Çarşamba 22:59:36	Son Mesafe Sırayı Dağıtım Tarihi	16 Ağustos 2017 Çarşamba 18:46:53	İlaçlamaya Başlangıç İlişkisi	16 Ağustos 2017 Çarşamba 21:56:51											
PEŞKEBİRE	Hamim Çiftliği - Battalgası - Ordu	52,87	33,86	17 Ağustos 2017 Perşembe 17:00:19	İlk Mesafe Sırayı Dağıtım Tarihi	17 Ağustos 2017 Perşembe 22:57:24	Son Mesafe Sırayı Dağıtım Tarihi	17 Ağustos 2017 Perşembe 18:51:05	İlaçlamaya Başlangıç İlişkisi	17 Ağustos 2017 Perşembe 21:50:38											
CUMA	Yıldıztepe - Ordu	61,65	39,4	18 Ağustos 2017 Cuma 17:00:16	İlk Mesafe Sırayı Dağıtım Tarihi	18 Ağustos 2017 Cuma 23:01:03	Son Mesafe Sırayı Dağıtım Tarihi	18 Ağustos 2017 Cuma 18:56:47	İlaçlamaya Başlangıç İlişkisi	18 Ağustos 2017 Cuma 22:58:37											
CUMARTESİ	Ordu - Yamıncahan	69,71	39,02	19 Ağustos 2017 Cumartesi 17:00:16	İlk Mesafe Sırayı Dağıtım Tarihi	19 Ağustos 2017 Cumartesi 22:43:34	Son Mesafe Sırayı Dağıtım Tarihi	19 Ağustos 2017 Cumartesi 18:44:43	İlaçlamaya Başlangıç İlişkisi	19 Ağustos 2017 Cumartesi 21:29:18											
		411,41	203,77																		
	11. EYLÜL (20) (167)																				
	Yıldıztepe - Tanoğan	80,55	37,04	14 Ağustos 2017 Pazartesi 17:00:08	İlk Mesafe Sırayı Dağıtım Tarihi	14 Ağustos 2017 Pazartesi 22:57:50	Son Mesafe Sırayı Dağıtım Tarihi	14 Ağustos 2017 Pazartesi 19:22:57	İlaçlamaya Başlangıç İlişkisi	14 Ağustos 2017 Pazartesi 22:41:42											
PALZARTEŞİ																					
SALI	Sarıçuklu - Hamim Çiftliği - Üçbağlar - Zeker - Keme - Hasanlar	59,77	28,05	15 Ağustos 2017 Salı 17:00:11	İlk Mesafe Sırayı Dağıtım Tarihi	15 Ağustos 2017 Salı 21:06:32	Son Mesafe Sırayı Dağıtım Tarihi	15 Ağustos 2017 Salı 18:44:36	İlaçlamaya Başlangıç İlişkisi	15 Ağustos 2017 Salı 21:06:32											
ÇARŞAMBA	Şehit Fırat - Beylerbaşı - Hidayet	57,31	26,79	16 Ağustos 2017 Çarşamba 17:00:07	İlk Mesafe Sırayı Dağıtım Tarihi	16 Ağustos 2017 Çarşamba 21:39:31	Son Mesafe Sırayı Dağıtım Tarihi	16 Ağustos 2017 Çarşamba 18:46:15	İlaçlamaya Başlangıç İlişkisi	16 Ağustos 2017 Çarşamba 21:39:50											
PEŞKEBİRE	Hidayet - Tazetepe - Şehit Fırat - İskender - Çiğlipınar - Halilçetin - Sarıoğlu - Sarıçuklu - Üçbağlar	66,91	37,48	17 Ağustos 2017 Perşembe 17:00:20	İlk Mesafe Sırayı Dağıtım Tarihi	17 Ağustos 2017 Perşembe 22:54:51	Son Mesafe Sırayı Dağıtım Tarihi	17 Ağustos 2017 Perşembe 18:25:32	İlaçlamaya Başlangıç İlişkisi	17 Ağustos 2017 Perşembe 21:43:47											
CUMA	Ferihşahine - İzniye - Hamim Çiftliği - Kavaklıbağ - Alanbey - Nurije - Şeyh - B. Mustafa Paşa - Şifa - Çarşıbaşı - K. Mustafa Paşa	68,96	33,13	18 Ağustos 2017 Cuma 17:00:47	İlk Mesafe Sırayı Dağıtım Tarihi	18 Ağustos 2017 Cuma 23:31:15	Son Mesafe Sırayı Dağıtım Tarihi	18 Ağustos 2017 Cuma 18:46:00	İlaçlamaya Başlangıç İlişkisi	18 Ağustos 2017 Cuma 23:15:27											
CUMARTESİ	Hasanlar - Ordu - Beşparmak - Mevlana Beşbağ - İstiklal - Saray - B. Hüseyinbey	64,14	34,98	19 Ağustos 2017 Cumartesi 17:01:25	İlk Mesafe Sırayı Dağıtım Tarihi	19 Ağustos 2017 Cumartesi 21:39:21	Son Mesafe Sırayı Dağıtım Tarihi	19 Ağustos 2017 Cumartesi 18:03:37	İlaçlamaya Başlangıç İlişkisi	19 Ağustos 2017 Cumartesi 21:26:08											
		397,64	197,47																		
	12. EYLÜL (48) (174)																				
	Bindi - Çab - Şimşirhan	110,61	45,89	14 Ağustos 2017 Pazartesi 17:00:58	İlk Mesafe Sırayı Dağıtım Tarihi	14 Ağustos 2017 Pazartesi 23:24:03	Son Mesafe Sırayı Dağıtım Tarihi	14 Ağustos 2017 Pazartesi 19:06:52	İlaçlamaya Başlangıç İlişkisi	14 Ağustos 2017 Pazartesi 22:56:11											
PALZARTEŞİ																					
SALI	Kendirli - Şimşirhan - Topraktepe - Tepeliköy	98,53	42,15	15 Ağustos 2017 Salı 17:00:18	İlk Mesafe Sırayı Dağıtım Tarihi	15 Ağustos 2017 Salı 23:18:47	Son Mesafe Sırayı Dağıtım Tarihi	15 Ağustos 2017 Salı 19:00:48	İlaçlamaya Başlangıç İlişkisi	15 Ağustos 2017 Salı 22:58:29											
ÇARŞAMBA	Şahmatan - Binal	91,7	28,73	16 Ağustos 2017 Çarşamba 17:01:16	İlk Mesafe Sırayı Dağıtım Tarihi	16 Ağustos 2017 Çarşamba 22:13:13	Son Mesafe Sırayı Dağıtım Tarihi	16 Ağustos 2017 Çarşamba 19:02:12	İlaçlamaya Başlangıç İlişkisi	16 Ağustos 2017 Çarşamba 21:50:05											
PEŞKEBİRE	Topraktepe	76,73	25,65	17 Ağustos 2017 Perşembe 17:03:16	İlk Mesafe Sırayı Dağıtım Tarihi	17 Ağustos 2017 Perşembe 22:28:47	Son Mesafe Sırayı Dağıtım Tarihi	17 Ağustos 2017 Perşembe 19:00:12	İlaçlamaya Başlangıç İlişkisi	17 Ağustos 2017 Perşembe 22:03:47											
CUMA	Topraktepe	80,42	12,17	18 Ağustos 2017 Cuma 17:06:08	İlk Mesafe Sırayı Dağıtım Tarihi	18 Ağustos 2017 Cuma 21:37:28	Son Mesafe Sırayı Dağıtım Tarihi	18 Ağustos 2017 Cuma 19:55:14	İlaçlamaya Başlangıç İlişkisi	18 Ağustos 2017 Cuma 21:11:46											
CUMARTESİ	Yala - Hatunsuyu	79,82	31,3	19 Ağustos 2017 Cumartesi 17:01:13	İlk Mesafe Sırayı Dağıtım Tarihi	19 Ağustos 2017 Cumartesi 22:10:51	Son Mesafe Sırayı Dağıtım Tarihi	19 Ağustos 2017 Cumartesi 19:05:14	İlaçlamaya Başlangıç İlişkisi	19 Ağustos 2017 Cumartesi 21:52:46											
		537,81	185,89																		
	13. EYLÜL (87) (164)																				
	K. Hüseyinbey - Tazetepe	70,12	49,24	14 Ağustos 2017 Pazartesi 17:03:57	İlk Mesafe Sırayı Dağıtım Tarihi	14 Ağustos 2017 Pazartesi 22:24:13	Son Mesafe Sırayı Dağıtım Tarihi	14 Ağustos 2017 Pazartesi 18:17:56	İlaçlamaya Başlangıç İlişkisi	14 Ağustos 2017 Pazartesi 22:21:38											
PALZARTEŞİ																					
SALI	Yalınca - Karabağlar - Hamim Çiftliği	104,43	22,13	15 Ağustos 2017 Salı 17:04:56	İlk Mesafe Sırayı Dağıtım Tarihi	15 Ağustos 2017 Salı 22:00:35	Son Mesafe Sırayı Dağıtım Tarihi	15 Ağustos 2017 Salı 19:17:47	İlaçlamaya Başlangıç İlişkisi	15 Ağustos 2017 Salı 21:08:06											
ÇARŞAMBA		11,52		16 Ağustos 2017 Çarşamba 17:03:23	İlk Mesafe Sırayı Dağıtım Tarihi	16 Ağustos 2017 Çarşamba 17:25:34	Son Mesafe Sırayı Dağıtım Tarihi	16 Ağustos 2017 Çarşamba 17:25:34	İlaçlamaya Başlangıç İlişkisi												
PEŞKEBİRE	Kaynarca	40,48	16,62	17 Ağustos 2017 Perşembe 17:00:05	İlk Mesafe Sırayı Dağıtım Tarihi	17 Ağustos 2017 Perşembe 20:25:47	Son Mesafe Sırayı Dağıtım Tarihi	17 Ağustos 2017 Perşembe 16:18:50	İlaçlamaya Başlangıç İlişkisi	17 Ağustos 2017 Perşembe 20:06:30											
CUMA	Çilesiz	59,64	9,24	18 Ağustos 2017 Cuma 17:00:09	İlk Mesafe Sırayı Dağıtım Tarihi	18 Ağustos 2017 Cuma 21:14:18	Son Mesafe Sırayı Dağıtım Tarihi	18 Ağustos 2017 Cuma 19:28:16	İlaçlamaya Başlangıç İlişkisi	18 Ağustos 2017 Cuma 20:48:52											
CUMARTESİ	Üçbağlar - Göztepe	47,55	13,64	19 Ağustos 2017 Cumartesi 17:04:09	İlk Mesafe Sırayı Dağıtım Tarihi	19 Ağustos 2017 Cumartesi 21:06:14	Son Mesafe Sırayı Dağıtım Tarihi	19 Ağustos 2017 Cumartesi 19:23:41	İlaçlamaya Başlangıç İlişkisi	19 Ağustos 2017 Cumartesi 20:59:55											
		333,74	110,87																		