

**ÖNCE DAĞIT SONRA TOPLA
ARAÇ ROTALAMA PROBLEMİ
İÇİN BULANIK ORTAMDA
ÇOK AMAÇLI YAKLAŞIM
VE UYGULAMA**

Gülçin DİNÇ YALÇIN

Endüstri Mühendisliği Anabilim Dalı

Temmuz-2012

JÜRİ VE ENSTİTÜ ONAYI

Gülçin Dinç Yalçın'ın “Önce Dağıt Sonra Topla Araç Rotalama Problemi İçin Bulanık Ortamda Çok Amaçlı Yaklaşım ve Uygulama” başlıklı **Endüstri Mühendisliği** Anabilim Dalındaki, Yüksek Lisans Tezi 13.06.2012 tarihinde, aşağıdaki jüri tarafından Anadolu Üniversitesi Lisansüstü Eğitim Öğretim ve Sınav Yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

	Adı-Soyadı	İmza
Üye (Tez Danışmanı) :	Doç. Dr. Nihal ERGİNEL	
Üye :	Prof. Dr. Refail KASIMBEYLİ	
Üye :	Yard. Doç. Dr. İnci SARIÇİÇEK	

Anadolu Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun tarih ve sayılı kararıyla onaylanmıştır.

Enstitü Müdürü

ÖZET

Yüksek Lisans Tezi

ÖNCE DAĞIT SONRA TOPLA ARAÇ ROTALAMA PROBLEMİ İÇİN BULANIK ORTAMDA ÇOK AMAÇLI YAKLAŞIM VE UYGULAMA

Gülçin DİNÇ YALÇIN

Anadolu Üniversitesi
Fen Bilimleri Enstitüsü
Endüstri Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Nihal ERGİNEL
2012, 155 sayfa

Önce dağıtım sonra topla araç rotalama probleminde (ÖDST-ARP), müşteriler dağıtım yapılacak ve toplama yapılacaklar olmak üzere ikiye ayrılır. Depodan çıkan bir araç ürünleri dağıtım yapılacak müşterilere götürdükten sonra toplama yapılacak müşterilere uğrayarak ürünleri alır ve depoya geri döner. Bu çalışmada ÖDST-ARP'yi matematiksel modelle çözebilmek için iki aşamalı bir yöntem geliştirilmiştir. Bölge oluşturma olarak isimlendirilen ilk aşamada önerilen çok amaçlı model ile müşteriler araçlara atanmış ve modelin çözümü için bulanık çözüm yaklaşımları kullanılmıştır. Amaçların ağırlıklarının elde edilmesi için Belenson ve Kapur'un iki oyunculu sıfır toplamli oyun kuramına dayanan yöntemi bulanık kümeler ile yeniden düzenlenerek yeni bir metod önerilmiştir. Atama işleminden sonra araçların rotalarının oluşturulması önce dağıtım sonra topla gezgin satıcı problemine (ÖDST-GSP) dönüşmüştür. Rotalama aşaması olarak isimlendirilen bu ikinci aşamada, her bir aracın rotası Goetschalckx ve Jacobs-Belcha'nın ÖDST-ARP için önerdikleri modelden kapasite ve atama kısıtları çıkarılıp ÖDST-GSP için oluşturulan matematiksel model ile elde edilmiştir. Son olarak her iki aşama sonunda bulunan araç rotalarının iyileştirilmesi için yerel arama kullanılmıştır. Önerilen yöntem ile literatürde yer alan test problemleri çözülmüş ve sonuçlar literatürde bilinen en iyi sonuçlar ile karşılaştırılmıştır. Ayrıca seramik sağlık gereçleri üreten işletmenin bir haftalık rotalama problemi önerilen yöntem ile çözülmüştür. Araç kapasitesini aşan talep noktaları, araç kapasitesine göre bölünüp yapay noktalar elde edilerek tüm noktaların kapasitesinin araç kapasitesine eşit veya kapasiteden az olması sağlanmıştır.

Anahtar Kelimeler: Önce Dağıtım Sonra Topla Araç Rotalama Problemi, Çok Amaçlı Karar Verme, İki Aşama, Yerel Arama, Bulanık Yaklaşım

ABSTRACT

Master Of Science Thesis

A MULTI OBJECTIVE APPROACH FOR VEHICLE ROUTING PROBLEM WITH BACKHAULS UNDER FUZZY ENVIROMENT AND AN APPLICATION

Gülçin DİNÇ YALÇIN

**Anadolu University
Graduate School of Sciences
Industrial Engineering Program**

**Supervisor: Assoc. Prof. Dr. Nihal ERGİNEL
2012, 155 pages**

In Vehicle Routing Problem with Backhauls (VRPB), customers are divided into two categories as linehaul customers and backhaul customer. A vehicle that starts from depot deliveries all good to linehaul customers, then bring goods from backhaul customers and finally return to depot. A method that consists of two phase is proposed to solve the problem with mathematical programming in this study. Customers are clustered to vehicles with proposed multi objective mathematical programming in the first phase that is named as “clustering phase” and fuzzy approaches are used to solve the model. A method that is a modification of Belenson and Kapur’s two-person zero-sum game theory based approach with using fuzzy sets is proposed to determine weights of objectives. Constructing routes are converted into traveling salesman problem with backhauls (TSPB) after clustering. Every vehicle routes is obtained by proposed mathematical model of TSPB that is a modification of Goetschalckx ve Jacobs-Belcha’s model for VRPB in the second phase that is named as routing phase. Finally local search is used to improve the routes that are generated after two phases. Test problems from literature are solved by the proposed method and the solutions are compared with best known solutions in literature. Also weekly routing problem for a firm that produces ceramic sanitary ware , is solved by the proposed method. Demand nodes that exceed vehicle capacity are divided due to vehicle capacity and artificial nodes are generated so demand of all nodes are provided to be less or equal to vehicle capacity.

Keywords: Vehicle Routing Problem with Backhauls, Multi Objective Decision Making, Two Phase, Local Search, Fuzzy Approach

TEŞEKKÜR

Öğrencisi olduğum günden itibaren desteğini esirgemeyen değerli hocam sayın Doç. Dr. Nihal ERGİNEL'e ve Anadolu Üniversitesi Endüstri Mühendisliği Bölümü hocalarına teşekkürü bir borç bilirim.

Her konuda desteğini esirgemeyen eşim Taha Hasan YALÇIN'a, çalışmalarımda büyük yardımları olan arkadaşım Tahir ÇİFTÇİ'ye ve desteklerinden dolayı arkadaşlarım Araş. Gör. Erdener ÖZÇETİN, Araş. Gör. Zühal KARTAL, Araş. Gör. Beyzanur ÇAYIR ve Mümin SÖNMEZ'e teşekkür ederim.

Bu günlere gelmemi sağlayan ve her konuda desteklerini esirgemeyen aileme teşekkür ederim.

Yüksek Lisans Burs Programı ile desteklerinden dolayı TÜBİTAK'a teşekkürü bir borç bilirim.

Temmuz 2012

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET.....	i
ABSTRACT	ii
TEŞEKKÜR	iii
İÇİNDEKİLER	iv
ŞEKİLLER DİZİNİ	vii
ÇİZELGELER DİZİNİ	ix
SİMGELER VE KISALTMALAR DİZİNİ	x

1.GİRİŞ	1
---------	---

2.ÇOK AMAÇLI KARAR VERME	6
--------------------------	---

2.1. Bulanık Kümeler	9
2.2. Bulanık Kümelerin Çok Amaçlı Karar Vermede Kullanımı	12
2.3. Çok Amaçlı Matematiksel Modeller için Bulanık Çözüm Yaklaşımları ...	16
2.3.1. Min operatörü.....	16
2.3.2. İki aşamalı yaklaşım.....	19
2.3.3. Ağırlıklı max-min modeli	20
2.3.4. Ağırlıklı toplam operatörü.....	22
2.3.5. Bulanık ve operatörü	24

3.ÇOK AMAÇLI MATEMATİKSEL MODELLERDE AMAÇLARIN

AĞIRLIKLARININ BELİRLENMESİ	27
-----------------------------	----

3.1. Literatürde Kullanılan Yöntemler	27
3.2. Önerilen Metot	29
3.2.1. Oyun teorisi: İki oyunculu sıfır toplamlı oyun.....	29

3.2.2. Belenson ve Kapur'un yöntemi.....	33
3.2.3. Önerilen metodun adımları	35
4.ÖNCE DAĞIT SONRA TOPLA ARAÇ ROTALAMA PROBLEMİ	45
4.1. Araç Rotalama Problemi ve Türleri	45
4.2. Önce Dağıt Sonra Topla Araç Rotalama Problemi	46
4.2.1. Matematiksel modeller	48
4.2.2. Kesin çözüm yöntemleri	55
4.2.3. Sezgisel yaklaşımlar	56
4.2.4. Meta-Sezgisel yaklaşımlar	57
5.ÖNCE DAĞIT SONRA TOPLA ARAÇ ROTALAMA PROBLEMİ	
İÇİN ÖNERİLEN YÖNTEM	60
5.1. Bölge Oluşturma Aşaması.....	62
5.2. Rotalama Aşaması.....	65
5.3. Yerel Arama	67
6.TEST PROBLEMLERİNİN ÖNERİLEN YÖNTEM İLE ÇÖZÜLMESİ	69
6.1. Hesapsal Sonuçlar	69
6.2. Test Problemleri İçin Geliştirilen Ara Yüz	79
6.2.1. Bölge oluşturma aşamasında kullanılan ara yüz	79
6.2.2. Rotalama aşamasında kullanılan ara yüz	85
6.2.3. Yerel aramada kullanılan ara yüz.....	88
7.SERAMİK SAĞLIK GEREÇLERİ ÜRETİMİ YAPAN BİR	
İŞLETMEDE ÖNERİLEN YÖNTEMİN UYGULAMASI	91
7.1. İşletme Hakkında Genel Bilgi	91
7.2. İşletmenin Bir Haftalık Rotalama Problemi.....	94

7.3. Uygulama İçin Geliştirilen Ara Yüz	100
7.3.1. Bölge oluşturma aşamasında kullanılan ara yüz	101
7.3.2. Rotalama aşamasında kullanılan ara yüz	105
7.3.3. Yerel aramada kullanılan ara yüz.....	108
8. SONUÇ	111
KAYNAKLAR	113
EK-1 BÖLGE OLUŞTURMA AŞAMASI İÇİN YAZILAN GAMS	
YAZILIMININ KODLARI.....	120
EK-2 ROTALAMA AŞAMASI İÇİN YAZILAN GAMS YAZILIMININ	
KODLARI.....	127
EK-3 TEST PROBLEMLERİNİN ÖZELLİKLERİ.....	130
EK-4 TEST PROBLEMLERİNİN ÖNERİLEN YÖNTEM İÇİN CPU	
DEĞERLERİ.....	132
EK-5 TEST PROBLEMLERİNİN ÇÖZÜMÜNDE “BÖLGE	
OLUŞTURMA” AŞAMASINDA AMAÇLARA ATANAN	
AĞIRLIKLAR	134
EK-6 UYGULAMA PROBLEMİNİN SONUÇLARI	136

ŞEKİLLER DİZİNİ

2.1. İyi bir atıcı için iyiliğin derecesi (Lai ve ark., 1994)	11
2.2. Üyelik fonksiyonlarının şekilsel gösterimi (a) en büyüklenmek istenen amaç fonksiyonu için (b) en küçüklenmek istenen amaç fonksiyonu ve küçük eşittir şeklindeki kısıt için (c) eşitlik halindeki kısıt için (Lai ve ark., 1994).....	15
3.1. Belenson ve Kapur'un ağırlıkları ve önerilen metodun ağırlıkları kullanılarak ağırlıklı toplam yöntemi, min operatörü, iki aşamalı yaklaşım, ağırlıklı toplam operatörü ve ağırlıklı max-min operatörü kullanılarak elde edilen çözümlerde amaçların üyelik fonksiyonu değerleri ve toplam üyelik fonksiyonu değeri.....	40
3.2. Problemlerin toplam üyelik dereceleri ve uzaklık fonksiyonu değerleri	43
4.1. Önce dağıt sonra topla araç rotalama problemi örneği (Mingozzi ve Giorgi, 1999).....	47
6.1. Veri girişi için tasarlanan ara yüz.....	80
6.2. Veriler girildikten sonra ara yüzün görünümü	80
6.3. "Veri Düzenle" butonuna basıldığında "Veri" sayfasında elde edilen bilgiler	81
6.4. "Veri Düzenle" butonuna basıldığında "Bölge Veri" sayfasında elde edilen bilgiler	82
6.5. "Veri Düzenle" butonuna basıldığında "Tasarruf" sayfasında elde edilen bilgiler	82
6.6. GAMS yazılımında bölge oluşturma aşaması çözdürüldüğünde programa yazdırılan verilerin görünümü.....	83
6.7. "Bölgeler" sayfasının görüntüsü	84
6.8. "Bölgeleri Yaz" butonuna basıldıktan sonra verilerin düzenlenmiş hali	85
6.9. Rotalama aşaması için düzenlenecek veriler için ara yüz	86
6.10. Rotalama aşaması için düzenlenecek verilerin girilmiş hali	86
6.11. "Bölge Verilerini Düzenle" butonuna basıldıktan sonra elde edilen verilerden bir örnek	87
6.12. Rotalama aşamasından sonra GAMS yazılımı sonuçlarının yazdırılması ...	87
6.13. "Rota Yerel Arama" sayfasının görünümü	88
6.14. "Rota Yerel Arama" sayfasının "Rota Oluşturma ve Geliştirme" butonuna basıldıktan sonraki görünümü	89
6.15. "Yerel Arama" sayfasının veri girişi sonrası görünüşü.....	89
6.16. "Yerel Arama" sayfasının "Rota Geliştirme" butonuna basıldıktan sonraki görünüşü	90
7.1. "Ana Veri" sayfasının görüntüsü	100
7.2. "Veri Girişi" sayfasının gerekli bilgiler girildikten sonraki görünümü	101
7.3. "Verileri Düzenle" butonuna basıldıktan sonra "Veri Girişi" sayfasının	

görünümü	102
7.4. Bölge oluşturulma aşaması için GAMS yazılımı çalıştırıldıktan sonra verilerin EXCEL programına yazdırılan sonuçların görüntüsü	103
7.5. "Bölgeler" sayfasının "Bölgeleri Yaz" butonuna basıldıktan sonraki görüntüsü, 1. kısım.....	104
7.6. "Bölgeler" sayfasının "Bölgeleri Yaz" butonuna basıldıktan sonraki görüntüsü, 2. kısım.....	105
7.7. "Bölge Düzenle" sayfasına veri girildikten sonra görünümü.....	106
7.8. "Bölge verilerini düzenle" butonuna basıldıktan sonraki görünüm	107
7.9. "Bölge Oluşturma ve Geliştirme" butonuna basıldıktan sonraki görünüm..	109
7.10. "Rota" sayfasının görünümü	110

ÇİZELGELER DİZİNİ

2.1. Ödemeler Matrisi	13
3.1. İki oyunculu-sıfır toplamlı oyunun ödemeler matrisi	30
3.2. Ödemeler Matrisi	34
3.3. Önerilen yöntemin ödemeler matrisi.....	36
3.4.Örneğin Belenson ve Kapur'un kullandığı ödemeler matrisi	37
3.5. Örneğin önerilen ödemeler matrisi.....	38
3.6. Örneğin çözümleri.....	39
3.7. Problemlerin çözümü	43
3.8. Problemlerdeki amaçların ağırlıkları.....	43
6.1. Test problemlerinin sonuçları	71
6.2. A1 problemi için bulunan çözüm.....	78
7.1. İşletmenin müşteri ve tedarikçi bilgileri	92
7.3. İşletmenin bir haftalık müşteri verisi	95
7.5. İşletmenin bir haftalık tedarikçi verisi	98
7.6. Uygulama sonuçları	99

SİMGELER VE KISALTMALAR DİZİNİ

ARP	: Araç rotalama problemi
ÇAKV	: Çok amaçlı karar verme
GSP	: Gezgin satıcı problemi
ÖDST- ARP	: Önce dağıt sonra topla araç rotalama problemi
ÖDST- GSP	: Önce dağıt sonra topla gezgin satıcı problemi
VBA	: Visual Basic Application

Alt ve Üst İndisler

C	: Araç kapasitesi
Enb	: enbüyüklemek
Enk	: enküçüklemek
K	: Araç sayısı
M	: Toplama yapılacak müşteri sayısı
N	: Dağıtım yapılacak müşteri sayısı

1. GİRİŞ

Günümüzde firmaların ayakta kalabilmeleri ve rekabet halinde olabilmeleri için hem ürünlerini kaliteli bir biçimde üretmeleri hem de maliyetlerini düşürmeleri gerekmektedir. Ürünün toplam maliyetinin %20'sini ise dağıtım maliyeti oluşturmaktadır. Bu dağıtım maliyeti hem ürünün tedarik zinciri ile hem de ürünün son müşteriye dağıtım zinciri ile ilişkilidir (Reimann ve ark., 2004).

1959'da Danzig ve Ramser tarafından ortaya atılan araç rotalama problemi (ARP) , toplam mesafenin en küçük olması veya toplam maliyetin en küçük olmasını sağlayacak şekilde araç kapasitesi göz önüne alınarak ürünlerin müşterilere dağıtımı için rotaların oluşturulmasıdır. Bir araç depodan dolu olarak çıkar, müşterilere uğradıktan sonra tekrar depoya geri döner ve her müşteriye sadece bir araç hizmet eder.

Araç rotalama problemi farklı özelliklere sahip kapasiteli araç rotalama problemi, zaman pencereli araç rotalama problemi, önce dağıt sonra topla araç rotalama problemi, zaman pencereli önce dağıt sonra topla araç rotalama problemi, eş zamanlı dağıtım toplama araç rotalama problemi ve zaman pencereli eş zamanlı dağıtım toplama araç rotalama problemi olarak alt problemlere ayrılmaktadır (Toth ve Vigo, 2002).

Bu tezin çalışma konusu olan önce dağıt sonra topla araç rotalama problemi (vehicle routing problem with backhauls) (ÖDST-ARP) ise depodan çıkan araçların öncelikle ürünlerin dağıtılacak olan müşterilere gidilmesini, ürünlerin dağıtımı bittikten sonra depoya dönerken ürün toplanılacak müşterilere uğranılmasını sağlayacak rotaların oluşturulması problemidir. Dolayısı ile ÖDST-ARP'de müşteriler dağıtım yapılacak müşteriler/dağıtım noktaları ve toplama yapılacak müşteriler/toplama noktaları olmak üzere ikiye ayrılır. Rotalar oluşturulurken araç kapasitesi göz önüne alınır ve amaç toplam mesafeyi/maliyeti en küçük yapmaktır.

ÖDST-ARP, ARP gibi NP karmaşık bir yapıya sahiptir. Bu sebeple çözümü için literatürde matematiksel modeller önerilmesine rağmen bu modeller büyük çaplı problemlerin çözümü için yeterli olmamış, çeşitli sezgisel ve meta

sezgisel yöntemler geliştirilmiştir. Sezgisel ve meta sezgisel yöntemlerin kullanılabilmesi için de bu yöntemlerin algoritmalarının herhangi bir bilgisayar programlama dilinde kodlanması gerekmektedir.

Bu tezde ÖDST-ARP'yi matematiksel modelle çözebilmek ve programlamaya duyulan ihtiyacı azaltmak amacı ile iki aşamalı bir yöntem önerilmiştir. Birinci aşama bölge oluşturma aşaması olarak isimlendirilmiş ve önerilen çok amaçlı matematiksel model ile müşteriler alt bölgelere ayrılarak müşterilerin araçlara atanması sağlanmıştır. İkinci aşama rota oluşturma aşaması olarak isimlendirilmiş ve her bir araç Goetschalckx ve Jacobs-Belcha (1989)'nın ÖDST-ARP için önerdikleri matematiksel modelin yeniden düzenlenmesi ile elde edilen model ile rotalanmıştır. Son olarak yerel arama ile oluşturulan rotalar iyileştirilmeye çalışılmıştır.

Problemi alt bölgelere ayıran çok amaçlı model ile merkez olacak müşteri noktaları ve bu merkezlere atanacak müşteri noktaları belirlenir. Her bir müşterinin atandığı merkeze olan toplam uzaklık oranı minimum yapılmak istenirken, toplam tasarruf miktarı maksimum yapılmak istenmektedir. Buradaki tasarruf miktarı, araç rotalama probleminin çözümünde kullanılan tasarruf algoritmasından elde edilmektedir. Bölgeler o bölgedeki dağıtım yapılacak ürün sayısı araç kapasitesini aşmayacak, benzer şekilde toplama yapılacak ürün sayısı araç kapasitesini aşmayacak şekilde oluşturulmuştur. Bölge sayısı, dağıtım/toplama için kullanılacak araç sayısı kadardır.

Çok amaçlı doğrusal programlama modeli, amaçların üyelik fonksiyonlarını kullanan bulanık çözüm yaklaşımları ile çözülmüştür. Üyelik fonksiyonları amaçların problem kısıtları altında alabileceği en iyi ve en kötü değerlere göre oluşturulur ve amacın en iyi değerine olan yakınlığı gösterir. Bulanık çözüm yaklaşımları, min operatörü dışında, amaçlara ağırlıkların atanmasını gerektirmektedir. Literatürde genellikle amaçların ağırlıkları eşit kabul edilmiş veya karar vericiye sorularak belirlenmiştir. Bunun dışında AHP gibi çok kriterli karar yöntemleri de kullanılmıştır. Birkaç çalışmada ise ağırlıkların belirlenmesine ilişkin yöntemler önerilmiştir. Bu yöntemler de karar vericiden amaçlar hakkında detaylı bilgilerin alınmasını gerektirmektedir. Bu sebeple literatürde ağırlıkların belirlenmesi için karar vericiden bilgi alınmasına gerek

olmayan bir yöntemle ihtiyaç duyulmaktadır. Belenson ve Kapur (1973)'un iki oyunculu sıfır toplamli oyun kuramına dayanan yöntemi bulanık kümeler ile yeniden düzenlenerek amaçların ağırlıkları belirlenmesine ilişkin yeni bir metot önerilmiştir.

Alt bölgelere ayırma ile birlikte problem her bir araç için önce dağıt sonra topla gezgin satıcı problemine (ÖDST-GSP, travelling salesman problem with backhauls- TSPB) dönüştürülmüş , matematiksel model ile makul sürede çözülebilecek boyutlara indirgenmiştir. Goetschalckx ve Jacobs-Belcha (1989)'nın ÖDST-ARP için önerdikleri matematiksel modelden kapasite ve atama kısıtları çıkarılarak ÖDST-GSP için matematiksel model oluşturulmuştur. Her bir araç için rota bu yeniden düzenlenen matematiksel modelin çözümü ile elde edilmiştir.

Elde edilen rota, ekleme (insertation) ve değiştirme (interchance) operatörlerinin kullanıldığı yerel arama ile iyileştirilmeye çalışılmıştır. Ekleme operatöründe, bir müşteri bulunduğu araçtan alınır ve eğer başka bir aracın kapasitesi uygun ise bu araca eklenmeye çalışılır. Bu müşterinin yeni yeri ile birlikte rotanın toplam mesafesinde bir azalma oluyor ise yeni rota oluşturulmuş olur. Değiştirme operatöründe, iki farklı araçta yer alan iki müşteri yine araçların kapasiteleri uygun ise karşılıklı yer değiştirilir ve müşteriler yeni araçlarında en uygun yere yerleştirilmeye çalışılırlar. Müşterilerin bu yeni yerleri ile birlikte rotanın toplam mesafesinde bir azalma oluyor ise yeni rota oluşturulmuş olur.

Literatürde ÖDST-ARP ile ilgili olan Goetschalckx ve Jacobs-Belcha (1989) tarafından önerilen test problemleri önerilen yöntem ile çözülmüş ve elde edilen sonuçlar literatürde bilinen en iyi değerler (Brandão, 2006) ile karşılaştırılmıştır. Bölge oluşturma aşamasında, farklı bulanık çözüm yaklaşımı ile elde edilen çözümler birbirinden farklı olabileceğinden dolayı her bir bulanık çözüm yaklaşımı ve her bir amaç ile elde edilen her bir bölge için rotalar elde edilmiş, çözümler karşılaştırılmıştır.

Ayrıca seramik sağlık gereçleri üreten bir işletmenin bir haftalık rotalama problemi önerilen yöntem ile çözülmüştür. İşletmenin dağıtım yapılan veya toplama yapılan müşterilerinin talepleri araç kapasitesinden fazla olabilmektedir. Literatürde kullanılan problemlerde, müşterilerin talepleri araç kapasitesinden

küçük veya araç kapasitesine eşit olmaktadır. İncelenen çalışmalar içerisinde araç kapasitesini aşan talepler ile karşılaşılmamış, dolayısı ile böyle bir durum için çözüm önerisi görülmemiştir. Bu çalışmada, kapasiteyi aşan noktanın talebi araç kapasite uygun bir şekilde bölünerek yapay müşteriler oluşturulmuş, aynı nokta için oluşturulan yapay müşteriler arasındaki mesafe ise sıfır olarak ele alınmıştır.

Matematiksel modellerin çözümü için GAMS yazılımının CPLEX 9.0 çözücüsü kullanılmıştır. GAMS yazılımı için gerekli verilerin düzenlenmesi, GAMS yazılımından elde edilen verilerin daha kolay anlaşılır hale getirilmesi ve yerel arama algoritması için MS EXCEL VBA (Visual Basic Applicaiton) kullanılmıştır.

Kısaca, çalışmanın literatüre olan katkıları;

- Çok amaçlı matematiksel modellerde amaçların ağırlıklarının belirlenmesine ilişkin karar vericiden amaçlara ilişkin detaylı bilgi alınmasına gerek olmayan bir metot önerilmesi,
- ÖDST-ARP için hem müşterilerin araçlara atanmasında hem de her bir aracın rotalanmasında matematiksel modeller ile çözüm bulan bir yöntem önerilmesi,
- Önerilen yaklaşımın ilk aşamasında şehirlerin araçlara atanması için hem merkez noktaları belirleyen hem de bu merkezlere diğer noktaları atayan çok amaçlı matematiksel model önerilmesi,
- Önerilen yaklaşımın ikinci aşamasında gerekli olan ÖDST-TSP problemi için matematiksel model önerilmesi,
- Müşterilerin taleplerinin araç kapasitesini aştığı durumda müşteri taleplerini araç kapasitesine göre bölerek yapay müşteri noktaları oluşturma yaklaşımı önerisi,

şeklinde sıralanabilir.

Çalışmanın 2. bölümünde çok amaçlı karar verme (ÇAKV) ile ilgili temel kavramlar anlatılmış, bulanık mantık ile ÇAKV için kullanılan bulanık çözüm yaklaşımlarından bahsedilmiştir. 3. bölümde literatürde çok amaçlı matematiksel modellerde amaçların ağırlıklarına nasıl karar verildiği incelenmiş ve önerilen metot anlatılmıştır. 4. bölümde ÖDST-ARP'ye değinilmiş, çözümü için literatürde kullanılan matematiksel modellerden, sezgisel ve meta sezgisel yaklaşımlardan

bahsedilmiştir. 5. bölümde ÖDST-ARP için önerilen yöntem anlatılmıştır. 6. bölümde literatürde yer alan test problemlerine ilişkin hesapsal sonuçlar verilmiş, test problemlerinin çözümü için geliştirilen ara yüzden bahsedilmiştir. 7. bölümde sağlık gereçleri üretin firmada yapılan uygulamadan ve uygulama verileri için geliştirilen ara yüzden bahsedilmiştir. 8. bölümde ise sonuç ve öneriler ortaya konulmuştur.

2. ÇOK AMAÇLI KARAR VERME

Bireyler ve örgütler tanımlanmış amaçlarını gerçekleştirmek ve yaşamlarını sürdürebilmek çabası içerisinde iken karşılaştıkları sorunları çözmede, tarih boyunca üç temel yaklaşım göstermişlerdir. Sorunlar sistemini çözüme kavuşturmada ilk yaklaşım öncülüğünü Bacon'un yaptığı "Deneysel Yaklaşım"dır. Buna göre çözüm arayışları soyut düşüncelere değil gözlem ve deneylere bağlanmalıydı. İkinci yaklaşım ise Rene Descartes'in öncülüğünü yaptığı "Analitik Yaklaşım" olarak bilinir. Buradaki temel felsefe bir sorunlar sistemini elemanlarına ayırarak her birini ayrı ayrı analiz etmek ve sonuçta onları tekrar birleştirmek üzerine kurulmuştur. Zaman içerisinde sorunlar sisteminin ölçeğinin ve karmaşıklığının büyümesi her iki yaklaşımla çözüm üretme olasılığını azaltmış ya da ortadan kaldırmıştır. Günümüzde kısaca "Tümdengelim" veya "Bütünsellik" yaklaşımı olarak bilinen ve Bertalamfy'nin "Genel Sistem Teorisi" ile temelleri kurulan "Sistem Yaklaşımı" felsefesi söz konusu olanaksızlıkları aşmakta üçüncü yaklaşım olarak yaygın biçimde kullanılmaya başlanmıştır (Kuruüzüm, 1998).

Çok amaçlı karar verme süreci ve modelleri bu yaklaşımlar içerisinde üçüncü adımı oluşturan "Sistem Yaklaşımı" felsefesinin etkin uygulama alanlarından yöneylem araştırması tekniklerinin önemli bir grubunu içermektedir. Özellikle karar verme durumunda olan kişi ya da grupların belirli kısıtlamalar altında ve birbiriyle çelişen nitelikli birden fazla amacı tatminkar düzeyde gerçekleştirmek istemeleri halinde çözümler üretmeye çalışan bir karar modelleri kümesi olma niteliği söz konusudur (Kuruüzüm, 1998).

Hayatın bir parçası olan karar verme problemleri çoğunlukla birden fazla ve birbiriyle çelişen amaçları içerir. Bu tip problemlerin nasıl çözüleceği üzerine yapılan çalışmalar oldukça fazladır. Karar verme problemleri genel olarak "Çok Kriterli Karar Verme" ve "Çok Amaçlı Karar Verme" olarak ikiye ayrılabilir. Çok kriterli karar verme problemlerinde alternatifler baştan belirlenmiştir ve bu alternatifler karar verici tarafından belli kriterlere göre değerlendirilir. Çok amaçlı karar vermede ise alternatifler belirlenmemiştir. Belli kısıtlar altında en iyilenmek istenen amaçlar bellidir (Lai ve ark., 1994).

Çok amaçlı karar verme modellerinin temel kavramları ise şu şekildedir (Kuruüzüm, 1998):

Karar verici, sorunlar sistemini, tanımladığı amaçlar, kriterler ve hedefler doğrultusunda çözüme kavuşturmaya çalışan, yönlendiren veya denetleyen kişi ya da gruptur.

Analist, sorunlar sisteminin tanımlanmasından probleme ait modelin kurulmasına kadar karar verici ile etkileşim halinde bulunan, bazı durumlarda çözüm üretme sürecinin ara aşamalarında da bu etkileşimi sürdüren, problem çözme teknikleri konusunda uzman kişi veya gruptur. Bazen karar verici ve analist aynı kişi veya grup olabilir.

Kısıt, sorunlar sisteminin tanımlanmış amaç veya amaçlarını etkileyen varsayımlar ile içsel ve çevresel kaynaklar veya değişkenler üzerindeki sınırlamalardır.

Amaç, karar vericinin istekleri doğrultusunda maksimize veya minimize edilmek istenen özelliklerdir.

Hedef, istenilen bir seviye ile belirlenmiş amaçtır.

Kriter, performans etkinliğinin bir ölçüsüdür ve değerlendirme yapabilmenin temelini oluşturur.

Nitelik, karar vericinin istek ve ihtiyaçlarından nisbi olarak bağımsız bir şekilde tanımlanmış ve belirli bir kararın ne ölçüde gerçekleştirilebildiğinin değerlendirilmesine yarayan bir ölçüdür.

Karar değişkeni, karar verici tarafından verilen spesifik kararların her biridir.

Çok amaçlı karar verme problemleri doğrusal yapıda olabileceği gibi doğrusal olamayan yapıda da olabilir. Ancak bu çalışmada çok amaçlı doğrusal programlama modelleri ele alınmış ve incelenmiştir

K adet amaç fonksiyonuna sahip çok amaçlı doğrusal programlama modeli aşağıdaki gibi ifade edilebilir (Lai ve ark., 1994).

$$x \in X \quad (2.1)$$

$$X = \{x \in R^n : Ax \leq b, x \geq 0\} \quad (2.2)$$

kısıtları altında

$$enb/enk f_k(x) = C_k x, \quad k = 1, 2, \dots, K \quad (2.3)$$

Burada $C_k = (c_{k1}, c_{k2}, \dots, c_{kn})$ olmak üzere k. amaç fonksiyonunun kar/maliyet katsayısını, $b = (b_1, b_2, \dots, b_m)^T$ kaynak vektörünü, $x = (x_1, x_2, \dots, x_n)^T$ karar değişkenlerinin vektörünü ve $A = [a_{ij}]_{m \times n}$ teknik katsayılar matrisini göstermektedir.

Tek amaçlı doğrusal modellerde en iyi noktaya ulaşılabilirken, çok amaçlı doğrusal modellerde bu en iyilikten her bir amaç için ayrı ayrı bahsedebiliriz. Amaçlar genellikle birbiriyle çelişir durumda olduğu için hepsini en iyiye ulaştırmış tek bir noktanın var olması pek mümkün olmamaktadır. Bu sebeple baskın çözüm kavramı ortaya çıkmaktadır.

Baskın çözüm ya da baskınlık kavramı klasik iktisat teorisinin temel taşlarından olan Pareto Optimallik Prensibi üzerine kurulmuştur. Bu yaklaşıma göre en az bir amaç fonksiyonunda iyileşme sağlayan bir çözüm kombinasyonu, diğer amaçlarda herhangi bir kötüleşmeye neden olmuyorsa elde edilen bu ikinci çözüm birinci çözüme baskındır denir. İyileşme ile kastedilen maksimize edilmek istenilen amaç fonksiyonu için artış, minimize edilmek istenilen amaç fonksiyonu için azalıştır. Kötüleşme ile kastedilen ise maksimize edilmek istenilen amaç fonksiyonu için azalış, minimize edilmek istenilen amaç fonksiyonu için artıştır. Tüm amaçların maksimize edilmek istenildiği bir durumda baskınlığın basit bir vektörel gösterimi aşağıdaki gibi ifade edilebilir (Kuruüzüm, 1998).

$$x = (x_1, x_2, \dots, x_n) \quad (2.4)$$

$$y = (y_1, y_2, \dots, y_n) \quad (2.5)$$

olmak üzere aşağıdaki koşul geçerli olmak üzere x çözümü y çözümüne baskındır.

$$x_i \geq y_i, \quad i = 1, \dots, n \quad (2.6)$$

$$x_i > y_i, \text{ en az bir } i \text{ için} \quad (2.7)$$

Çok amaçlı doğrusal programlamada birden çok baskın çözüm söz konusu olduğu için baskın çözümler kümesinden söz etmek mümkündür. Karar vericinin istekleri doğrultusunda nihai çözüm baskın çözümler kümesinin içerisinde elde edilir.

Çok amaçlı doğrusal problemlerin çözümü için pek çok teknik geliştirilmiştir. Bu çalışmada bulanıklık kavramı kullanılarak geliştirilen çözüm yöntemleri ele alınmış ve bulanık çözüm yaklaşımları alt başlığı altında anlatılmıştır.

2.1. Bulanık Kümeler

Bulanık küme teorisi ilk defa Zadeh tarafından ortaya atılmıştır. Klasik mantıkta bir öneri doğru veya yanlıştır. Fakat gerçek dünyadaki olayların ne derecede iyi veya yanlış olmasının belirlenmesi gerekmektedir. Örneğin 100 °C suyun sıcaklığı sıcak olarak ifade edilirse 95 °C ve 80 °C 'lerdeki su için sıcak değildir ifadesi bu anlamda doğru olmadığı gibi yanlış da değildir. Bu nedenle önermelerin doğru (1) ve yanlış (0) arasındaki değerler (az sıcak, ılık, az soğuk vs.) kullanılarak bulanık küme kavramı ortaya atılmıştır. Bulanık küme teorisi az, sık, orta, düşük, çok, birçok gibi dilbilimsel yapıları kullanarak dereceli veri modellemesini gerçekleştirmektedir. Böylece olayların modellenmesinde daha gerçekçi ve doğala yakın sonuçlar elde edilmesini sağlar (Nabiyev, 2005).

Bulanık küme, kesin geçişleri elimine ederek belirsizlik kavramının tanımını yeniden verir ve evrendeki bütün bireylere üyelik derecesi değerini atayarak matematiksel olarak tanımlar. Bu derece, bulanık küme tarafından verilen kavram ile uyumludur ve benzer bir bireyin derecesine uyar. Böylece bireyler, bulanık küme içerisinde üyelik dereceleri tarafından gösterilen daha büyük ve daha küçük değerlere ait olabilirler. Bu üyelik dereceleri [0-1] aralığında gerçel değerler ile ifade edilir (Nabiyev, 2005).

Bulanık üyelik ile olasılık birbiriyle kıyaslanabilmektedir. Bu iki kavramın birbirinden farkı şunlardır (Nabiyev, 2005).

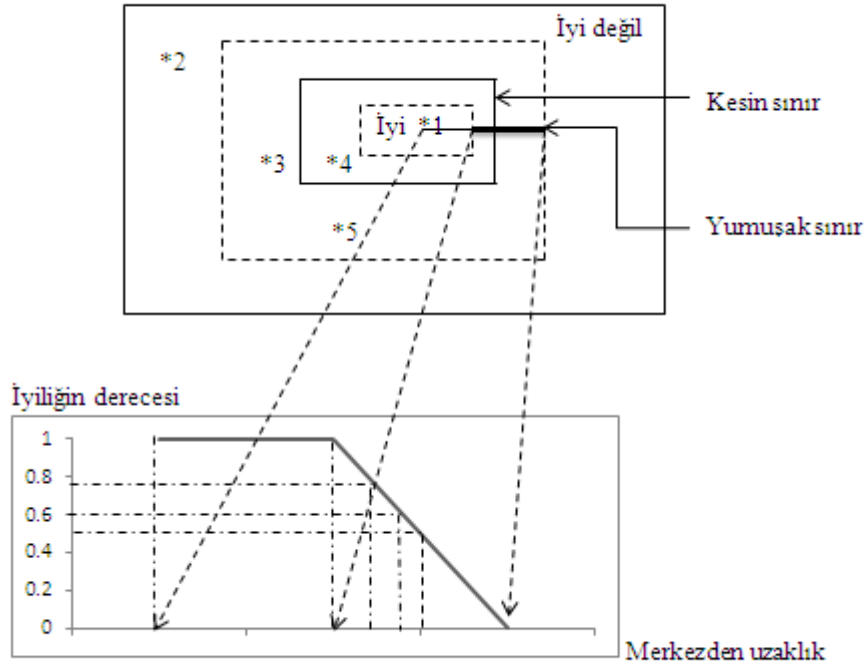
- Sonlu bir evrensel kümede olasılıklar toplamı 1'e eşit olmakta, bulanık üyelik derecelerinde ise böyle bir durumun gerçekliliği olmamaktadır.
- Olasılık, ayrık değerlere sahip olmakta, bulanık kümenin bireye ilişkin üyelik dereceleri süreklilik taşımaktadır.
- Klasik olasılık hesapları, bireyin tamamının temeline dayanır. Bulanık küme teorisinde ise bireyin üyelik derecesi, diğer bireylerin tamamının temeline ilişkin olmamakla birlikte üyelikler bakımından farklar görülmektedir.

“O iyi bir atıcıdır” durumunu ele alırsak, iyiliği tanımlamak için olasılıkla ilgilenen atıcı hedefi n defa vurmasını söyler ve iyi bölgeyi vurmasını (*1) sayısını (m) belirler. İyiliğin derecesi m/n olur. Eğer atıcı *1 ve *2 bölgesini vurmuşsa iyilik derecesi 1/2 olur. Eğer atıcı 3. Atışında 3* bölgesini vurursa iyilik derecesi 1/3 olur. Dördüncü atışında *1 bölgeyi vurduğunda 2/4 (1/2) ve beşinci atışında *5 bölgeyi vurduğunda 2/5 olarak iyilik oranı değişecektir. Bu deneyde küçük örnek büyüklüğünde iyiliğin derecesini ölçmek için olasılığın dezavantajını görülmektedir. İyilik derecesini belirlemek için kesin sınırlar belirlemek yerine bulanık kümeler kullanılıp, üyelik derecesinin yumuşak geçişleri tanımlanabilir. Bu örnek için parantezin içindeki ikinci sayının üyelik derecesini gösterdiği üyelikleri şu şekilde belirleyelim: (*1, 1), (*2, 0), (*3, 0.6), (*4, 0.7) ve (*5, 0.5). aynı atışlar için üyelik fonksiyonlarına bakıldığında, 0.5, 0.53, 0.58 ve 0.56 bulunur. Bu durum daha istikrarlı ve az dalgalanan sonuç üretmektedir. Üyelik derecesi iyiliğin derecesini ölçmek için olasılığa göre daha iyi bir araç sunmaktadır. Bu durum aşağıdaki Şekil 2.1’de görülmektedir (Lai ve ark., 1994).

Bulanık kümeler hakkında bu genel bilgilerden sonra kümelerle ilgili matematiksel gösteriminden ve kümeler üzerinde yapılabilen temel işlemlerden bahsedilecektir. Bulanık kümeler matematiksel olarak (2.8)’de olduğu gibi ifade edilebilir (Zimmermann, 2001).

$$A = \{ (x, \mu_A(x)) \mid x \in X \} \quad (2.8)$$

Burada $\mu_A(x)$, x’in üyelik fonksiyonudur.



Şekil 2.1. İyi bir atıcı için iyiliğin derecesi (Lai ve ark., 1994)

Bulanık kümeler üzerinde iki kümenin birleşimi, kesişimi, toplanması, farkı, çarpımı gibi birçok işlem yapılabilir. Bulanık kümeler çok amaçlı karar verme problemlerinin çözümü için bulanık kümelerin kesişimi özelliği kullanılmaktadır.

İki bulanık kümenin kesişimi matematiksel olarak (2.9)'da olduğu gibi ifade edilir.

$$\mu_C(x) = \min\{\mu_A(x), \mu_B(x)\}, \quad x \in X \quad (2.9)$$

İki bulanık kümenin birleşimi matematiksel olarak (2.10)'da olduğu gibi ifade edilir.

$$\mu_D(x) = \max\{\mu_A(x), \mu_B(x)\}, \quad x \in X \quad (2.10)$$

İki bulanık kümenin cebirsel toplamı matematiksel olarak (2.11) ve (2.12)'ye göre ifade edilir.

$$C = \{ (x, \mu_{A+B}(x)) \mid x \in X \} \quad (2.11)$$

$$\mu_{A+B}(x) = \mu_A(x) + \mu_B(x) - \mu_A(x) \cdot \mu_B(x) \quad (2.12)$$

İki bulanık kümenin sınırlı toplamı matematiksel olarak (2.13) ve (2.14)'de olduğu gibi ifade edilebilir.

$$C = \{ (x, \mu_{A \oplus B}(x)) \mid x \in X \} \quad (2.13)$$

$$\mu_{A \oplus B}(x) = \min\{ 1, \mu_A(x) + \mu_B(x) \} \quad (2.14)$$

İki bulanık kümenin sınırlı farkı matematiksel olarak (2.15) ve (2.16)'da olduğu gibi ifade edilebilir.

$$C = \{ (x, \mu_{A \ominus B}(x)) \mid x \in X \} \quad (2.15)$$

$$\mu_{A \ominus B}(x) = \max\{ 0, \mu_A(x) + \mu_B(x) - 1 \} \quad (2.16)$$

2.2. Bulanık Kümelerin Çok Amaçlı Karar Vermede Kullanımı

Bulanık kümeler yöneylem araştırması, yönetim bilimi, kontrol teorisi, yapay zeka/uzman sistemler insan davranışları gibi birçok disipline uygulanmıştır. Üretim, taşıma, atama/yer seçimi, lojistik, oyun, çevre yönetimi, ticari denge, ekonomi, finans, personel seçimi, pazarlama yönetimi, muhasebe, tarımsal planlamalar ve regresyon modelleri gibi birçok problem için kullanılmıştır (Lai ve ark., 1994).

Yöneylem araştırmasında, bulanık kümeler doğrusal ve doğrusal olmayan programlama, tam sayılı programlama, dinamik programlama, güvenilirlik, kalite kontrol, kuyruk teorisi, çok amaçlı karar verme, grup karar verme, karar destek sistemleri ve uzman sistemler gibi tekniklere uygulanmıştır. Bu durum modellerin geliştirilmesine yardımcı olmuş, karmaşık sistemler için esnek modeller sağlamıştır (Lai ve ark., 1994).

Bulanık kümeler, amaçlar ve bulanık kısıtlar için üyelik fonksiyonları oluşturularak çok amaçlı karar verme problemlerinin çözümünde kullanılmıştır.

Bulanık kısıtlarda, teknik katsayılar matrisi, sağ taraf sabitleri veya her ikisi de bulanık olabilir. Üyelik fonksiyonları 0 ile 1 arasında bir değer alır ve bulunan çözümün en iyi çözüme olan yakınlığını gösterir. Böylelikle farklı değerler için benzer ifadeler elde edilir. 1 değeri en iyi değere ulaşıldığını gösterirken, 0 değeri en kötü değerinde elde edildiğini, aradaki değerler ise en iyi değere ne kadar ulaşıldığını gösterir. Üyelik fonksiyonları çeşitli formlarda tanımlanabilir. Bu çalışmada doğrusal olan üyelik fonksiyonları ile ilgilenilmiştir.

Amaçların üyelik fonksiyonları oluşturulurken amaç fonksiyonlarının en iyi ve en kötü değerlerinin belirlenmesi gerekmektedir. En iyi değer problemin kısıtları altında amacın tek başına en iyi noktasının bulunması ile elde edilir. En kötü değer, amacı minimum ise maksimum biçimine, maksimum ise minimum biçimine dönüştürülerek problem kısıtları altında çözülmesi ile elde edilebileceği gibi, ödemeler matrisinden de elde edilebilir veya karar verici tarafından da belirlenebilir. Ödemeler matrisi Çizelge 2.1’de görülmektedir. Burada z_k k. amaç fonksiyonu, x^k k. amaç fonksiyonun optimum değerini göstermektedir. $f_i(x^j)$ ise j. amacın en iyi çözümünde i. amacın aldığı değeri göstermektedir.

Çizelge 2.1. Ödemeler Matrisi

	x^1	x^2	...	...	x^k
z_1	$f_1(x^1)$	$f_1(x^2)$	...	...	$f_1(x^k)$
z_2	$f_2(x^1)$	$f_2(x^2)$	...	...	$f_2(x^k)$
...	...	...	$f_i(x^j)$	...	...
z_k	$f_k(x^1)$	$f_k(x^2)$	...	...	$f_k(x^k)$

Maksimum yapılmak istenilen amaç fonksiyonu için en iyi ve en kötü değer (2.17)’e göre elde edilebilir. (Lai, et al., 1994)

$$f_k^* = \text{enb}(f_k(x)) \quad f_k' = \text{enk}(f_k(x)) \quad (2.17)$$

Minimum yapılmak istenilen amaç fonksiyonu için en iyi ve en kötü değer (2.18)’e göre elde edilebilir.

$$f_k^* = \text{enk}(f_k(x)) \quad f_k' = \text{enb}(f_k(x)) \quad (2.18)$$

Burada f_k^* amaç fonksiyonunun problem kısıtları altında alabileceği en iyi değeri gösterirken, f_k' amaç fonksiyonunun alabileceği en kötü değeri göstermektedir.

Maksimize edilmek istenen amaç fonksiyonu için doğrusal üyelik fonksiyonunun matematiksel ifadesi aşağıdaki gibidir (Lai ve ark., 1994).

$$\mu_k(x) = \begin{cases} 1 & f_k > f_k^* \\ \frac{[f_k(x) - f_k']}{[f_k^* - f_k']} & f_k' \leq f_k(x) \leq f_k^* \\ 0 & f_k(x) < f_k' \end{cases} \quad (2.19)$$

Minimize edilmek istenen amaç fonksiyonu için doğrusal üyelik fonksiyonunun matematiksel ifadesi aşağıdaki gibidir.

$$\mu_k(x) = \begin{cases} 1 & f_k < f_k^* \\ \frac{[f_k' - f_k(x)]}{[f_k' - f_k^*]} & f_k^* \leq f_k(x) \leq f_k' \\ 0 & f_k(x) > f_k' \end{cases} \quad (2.20)$$

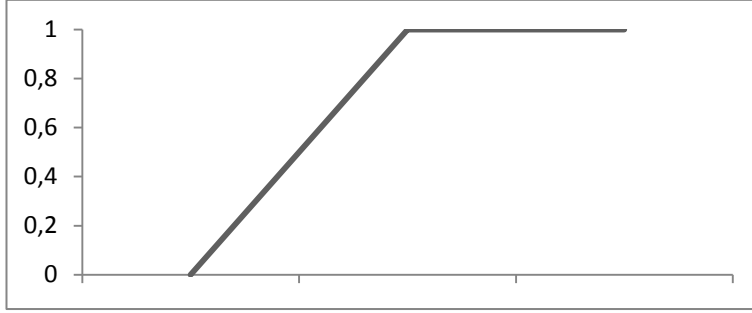
Eşitlik halindeki kısıt için doğrusal üyelik fonksiyonunun matematiksel gösterimi aşağıdaki gibidir.

$$\mu_i(x) = \begin{cases} [(Ax)_i - (b_i - d_i)]/d_i & b_i - d_i \leq (Ax)_i < b_i \\ [(b_i + d_i) - (Ax)_i]/d_i & b_i \leq (Ax)_i \leq b_i + d_i \\ 0 & b_i + d_i > (Ax)_i \text{ veya } (Ax)_i < b_i - d_i \end{cases} \quad (2.21)$$

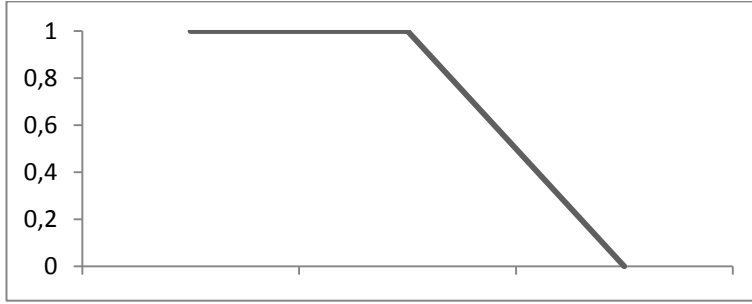
Küçük eşit halindeki kısıt için doğrusal üyelik fonksiyonunun matematiksel gösterimi aşağıdaki gibidir.

$$\mu_i(x) = \begin{cases} 1 & (Ax)_i < b_i \\ [(b_i + d_i) - (Ax)_i]/d_i & b_i \leq (Ax)_i \leq b_i + d_i \\ 0 & b_i + d_i > (Ax)_i \end{cases} \quad (2.22)$$

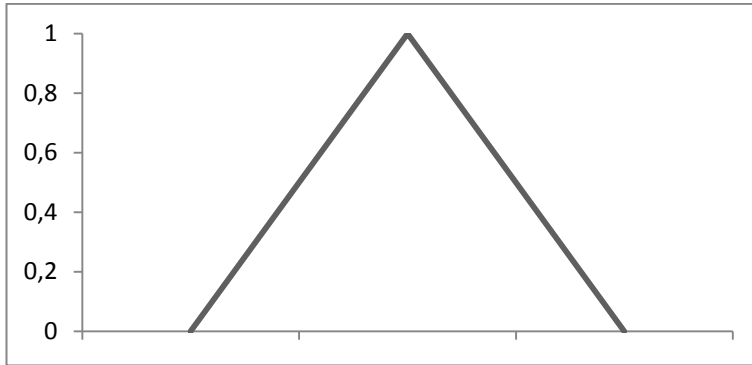
Burada d_i sağ taraf sabitinde olabilecek değişim miktarını göstermektedir. Yukarıda matematiksel ifadeleri verilen üyelik fonksiyonlarının şekilsel gösterimi ise Şekil 2.2'deki gibidir.



(a)



(b)



(c)

Şekil 2.2. Üyelik fonksiyonlarının şekilsel gösterimi (a) en büyüklenmek istenen amaç fonksiyonu için (b) en küçüklenmek istenen amaç fonksiyonu ve küçük eşittir şeklindeki kısıt için (c) eşitlik halindeki kısıt için (Lai ve ark., 1994)

2.3. Çok Amaçlı Matematiksel Modeller için Bulanık Çözüm Yaklaşımları

Bulanık kümeleri çok amaçlı yöntemlerin çözümü için ilk olarak Zimmermann (1977) kullanmıştır. Temeli Zimmermann'ın yaklaşımına dayanan başka yöntemler de literatürde geliştirilmiştir. Bu yöntemlerin ortak özelliği amaçların üyelik fonksiyonlarını oluşturmaları ve bu değerleri maksimize etmeyi amaçlamalarıdır. Böylelikle farklı katsayılarla ve minimize veya maksimize edilmek gibi farklı özelliklere sahip olan amaçların değerleri 0 ile 1 arasında getirilmiş olur.

2.3.1. Min operatörü

Zimmerman (1977), Zadeh ve Belmann (1970)'ın “min operatörünü” çok amaçlı doğrusal programlama problemlerinin çözümü için kullanmıştır. Amaç en küçük üyelik fonksiyonu değerini maksimum yapmak, bir başka deyişle üyelik fonksiyonlarının kesişim kümesini tanımlamaktır. μ_D olarak tanımlanan kesişim kümesi (2.23)'de olduğu gibi ifade edilir. Burada $\mu_k(x)$, k. amaç fonksiyonunun üyelik fonksiyonudur.

$$\mu_D = \min(\mu_1(x), \mu_2(x), \dots, \mu_k(x)) \quad (2.23)$$

Problemin kısıtları altında (2.24)'de görüldüğü gibi maksimum μ_D değeri elde edilmek istenir.

$$\max[\min_k \mu_k(x)] \quad (2.24)$$
$$x \in X$$

Eğer $\lambda = \min_k \mu_k(x)$ olarak ele alınırsa min operatörü (2.25)'de verilen matematiksel modele eşit olur.

$$\begin{aligned} \mu_k(x) &\geq \lambda & \forall k = 1, \dots, K \\ \lambda &\in [0,1] \\ x &\in X \\ \text{Kısıtları altında} \\ \text{enb } \lambda \end{aligned} \quad (2.25)$$

Min operatörü literatürde birçok çalışmada çok amaçlı matematiksel modellerin çözümü için kullanılmıştır ve ayrıca diğer bulanık çözüm yaklaşımlarının temelini oluşturmuştur.

Gupta ve Chakraborty (1998) iş değerlendirme ile ilgili olan çalışmalarında işleri değerlendirmek için belirlenen faktörlerin her bir seviyesinin kaç puan olması gerektiğine karar verecek model önermişlerdir. Bunun için işin karmaşıklığı, eğitim, denetim zorunluluğu ve zihinsel/görsel faaliyet olmak üzere 4 faktör ve her bir faktör için 6 seviye belirlenmiştir. Belirlenen 5 iş için faktör seviyelerinden oluşan amaçlar yazılmış, amaçların en iyi ve en kötü değerleri belirlenip doğrusal üyelik fonksiyonları oluşturulmuştur. Min operatörü ile çözüm elde edilmiştir.

Li ve Lai (2000) çalışmalarında çok amaçlı taşıma problemi için bulanık yaklaşım önermişlerdir. Amaçlar için doğrusal üyelik fonksiyonu kullanmışlardır. Çözüm aşamasında min operatörünü ve önerdikleri bulanık uzlaşık programlama yaklaşımını kullanmışlardır. Önerdikleri yaklaşım için amaçlara farklı ağırlıklar atayarak çözümler elde etmişlerdir. Amaçlara atanan farklı ağırlıklar ile farklı çözümlerin elde edildiğini göstermişlerdir.

El-Wahed (2001) çalışmasında çok amaçlı taşıma problemi için bulanık programlama yaklaşımı geliştirmiştir. Amaçlar için doğrusal üyelik fonksiyonu kullanılmıştır. Çözüm aşamasında min operatörü kullanılmış olup, karar verici çözümden memnun kalıncaya kadar üyelik fonksiyonundaki en kötü değer bir önceki aşamada elde edilen değer ile güncellenerek yeni üyelik fonksiyonu elde edilmiştir ve model yeniden çözülmüştür.

Arıkan ve Güngör (2001) çalışmalarında proje ağ tasarımı için tamamlanma zamanını ve aciliyet maliyetini (crashing time) minimum yapmak şeklinde belirlenmiş iki amaç ile gerçek hayat problemini çözmüşlerdir. Model her bir işin ne zaman başlaması gerektiğini bulmaktadır. Amaçların alt ve üst değerleri karar verici tarafından belirlenmiş ve üyelik fonksiyonları doğrusal olarak oluşturulmuştur. Çözüm aşamasında min operatörü kullanılmıştır.

Wang ve Liang (2004) çalışmalarında çok ürünlu toplam üretim planlaması problemi için bulanık çok amaçlı matematiksel model geliştirmişlerdir. Toplam üretim maliyetlerini minimum yapmayı, taşıma maliyetlerini minimum

yapmayı ve işgücü seviyesindeki değişmeyi minimum yapmayı amaçlamışlardır. Amaçların üyelik fonksiyonları parçalı üyelik fonksiyonu olarak ele alınmıştır. Çözüm aşamasında min operatörü kullanılmıştır.

Wang ve Liang (2005) çalışmalarında toplam üretim planlaması problemi için interaktif bulanık çok amaçlı matematiksel model geliştirmişlerdir. Amaçlar 2004 yılındaki çalışmaları ile aynıdır. Amaçların en iyi ve en kötü değerlerine karar verici karar vermektedir. Amaçların üyelik fonksiyonları için doğrusal üyelik fonksiyonları kullanılmış ve çözüm min operatörü ile elde edilmiştir. Çözüm karar verici tarafından yeterli bulunmazsa amaçlar için kabul edilebilecek en iyi ve en kötü değerler güncellenerek işlem prosedürü istenilen düzeyde sonuç elde edilinceye kadar devam edilmiştir.

Araz, Mizrak Özfiat ve Özkarahan (2007) çalışmalarında dış kaynak kullanımı için yöntem önermişlerdir. Öncelikle PROMETHEE yöntemi ile var olan firmalar değerlendirilmiştir. Daha sonra kurulan çok amaçlı model ile hangi firmaların seçilmesi gerektiğine ve bu firmalardan hangi ay ne kadar ürün alınması gerektiğine karar verilmektedir. Modelin amaçları PROMETHEE yöntemi ile elde edilen ağırlıkları maksimum yapmak, kabul edilen ürün miktarı oranını maksimum yapmak, zamanında gelen ürün miktarını maksimum yapmak ve toplam sipariş miktarını minimum yapmak olarak tanımlanmıştır. Amaçların en iyi değerleri her bir amaç tek tek optimize edilerek bulunurken, en kötü değerler karar verici tarafından belirlenmiştir. Üyelik fonksiyonları doğrusal olarak tanımlanmıştır. Türkiye'deki bir tekstil firmasında uygulama yapılmıştır. Çözüm aşamasında min operatörü kullanılmıştır.

Liang (2008) çalışmasında çok ürünlü, çoklu periot üretim/dağıtım planının kararları için envanter seviyeleri, makine kapasiteleri, her bir kaynaktaki işçi seviyeleri, pazarın talebi, her bir varış noktasındaki depolama alanları ve toplam bütçe kısıtları göz önüne alınarak toplam maliyeti ve toplam taşıma zamanını minimum yapmayı amaçlayan matematiksel model kurulmuştur. Amaçların üyelik fonksiyonları parçalı olarak tanımlanmış ve çözüm aşamasında min operatörü kullanılmıştır.

Liang ve Cheng (2009) çalışmalarında, Liang(2008) çalışmasında olduğu gibi çok ürünlü, çoklu periot üretim/dağıtım planının kararları problemini

modellenmiştir. Fakat amaçların üyelik fonksiyonları doğrusal olarak belirlenmiştir. Çözüm aşamasında aynı şekilde min operatörü kullanılmıştır.

2.3.2. İki aşamalı yaklaşım

Li, Zhang ve Li (2006) çalışmalarında, Zimmermann'ın min operatörüne bir aşama daha ekleyerek “iki aşamalı yaklaşımı” (two phase approach) önermişlerdir. İlk aşama min operatörü ile aynıdır. İkinci aşama ile amaçlanan eğer mümkünse amaçların üyelik fonksiyonu değerini yükseltebilmektir. İkinci aşama ise (2.26)'de matematiksel model olarak gösterilmiştir.

$$\begin{aligned} \mu_k(x) &\geq \lambda_k \geq \lambda_k^* & \forall k = 1, 2, \dots, K \\ \lambda_k &\in [0, 1] & \forall k = 1, 2, \dots, K \\ x &\in X \\ \text{kısıtları altında} \\ \text{enb } &\sum_{k=1}^K w_k \lambda_k \end{aligned} \quad (2.26)$$

Burada w_k k. amacın ağırlığıdır ve amaçların ağırlıklarının toplamı 1'e eşittir. λ_k^* ise ilk aşamadan elde edilen sonuç ile bulunan k. amacın üyelik fonksiyonu değeridir.

Gupta, Mehlawat ve Saxena (2008) çalışmalarında portföy seçimi için kısa vadede getiriyi, uzun vadede getiriyi ve yıllık temettüyü maksimum yapmak ve riski minimum yapmak şeklinde 4 amaçlı model kurmuşlardır. Amaçlar için S-şekilli üyelik fonksiyonlarını tanımlamışlar ve çözüm aşamasında iki aşamalı yaklaşımı kullanmışlardır.

Liang (2009) çalışmasında proje yönetimi için çok amaçlı model ve parçalı üyelik fonksiyonu kullanmıştır. Toplam proje maliyetini, toplam tamamlanma zamanını ve toplam erteleme maliyetini minimum yapmak üzere 3 amaç fonksiyonu belirlenmiştir. Çözüm aşamasında iki aşamalı yaklaşım kullanılmıştır.

Liang, Huang ve Yang (2011) çalışmalarında proje yönetimi kararları için toplam proje maliyetini minimum yapmayı amaçlayan matematiksel model ile çözüm bulmuşlardır. Amaçtaki parametrelerin değerleri en kötü değer, en iyi değer ve en olası değer şeklinde belirlendiği için, amaç üç farklı amaç olarak

tanımlanarak model çok amaçlı matematiksel modele dönüşmüştür. Amaçların üyelik fonksiyonları parçalı üyelik fonksiyonu olarak tanımlanmıştır. Çözüm aşamasında iki aşamalı yaklaşım kullanılmıştır.

Liang ve ark. (2011) çok ürünlü ve çoklu periyot toplam üretim planlama problemi için malzeme taşıma seviyeleri, işçi seviyeleri, makine kapasiteleri, depo alanı ve bütçe kısıtları altında, toplam üretim maliyeti ve işçi seviyelerinin değişim maliyetlerinin toplamını minimum olması amaçlanmıştır. Amaç için gerekli parametrelerin değerleri en kötü değer, en iyi değer ve en olası değer şeklinde belirlendiği için, amaç üç farlı amaç olarak tanımlanarak model çok amaçlı matematiksel modele dönüştürülmüş ve amaçların üyelik fonksiyonları parçalı üyelik fonksiyonu olarak tanımlanmıştır. Çözüm aşamasında iki aşamalı yaklaşım kullanılmıştır.

Rabbani ve ark. (2012) çalışmalarında stokastik üretim miktarları için hücre oluşturma problemini çok amaçlı matematiksel model ile çözmüşlerdir. Çeşitli maliyetlerin (sabit makine maliyetleri, değişken makine maliyetleri, hücre oluşturma maliyetleri ve hücre içi hareket maliyetleri) toplamını minimum yapmak ve hücre yükleme varyasyonlarını minimum yapmak üzere iki amaç tanımlanmıştır. Amaçların üyelik fonksiyonları doğrusaldır. Çözüm aşamasında iki aşamalı yaklaşımı, amaç fonksiyonunda değişiklik yaparak kullanmışlardır. Önerdikleri amaç fonksiyonu (2.27)'de gösterilmektedir.

$$\max \min(\mu_k(x)) + \frac{1}{K} \sum_{k=1}^K w_k(\lambda_k - \lambda_k^*) \quad (2.27)$$

2.3.3. Ağırlıklı max-min modeli

Lin (2004) “ağırlıklı max-min” modelini önermiştir. Amaç elde edilen çözüm ile bulunan amaçların üyelik fonksiyonu değerlerinin, amaçlara atanan ağırlıklar ile orantılı olmasıdır. Model (2.28)'de gösterilmiştir..

$$\begin{aligned}
\mu_k(x) &\geq \lambda w_k \quad \forall k = 1, 2, \dots, K \\
\lambda &\in [0, 1] \\
x &\in X \\
\text{kısıtları altında} \\
\text{enb } \lambda
\end{aligned}
\tag{2.28}$$

Burada w_k k. amacın ağırlığıdır ve amaçların ağırlıklarının toplamı 1'e eşittir.

Kongar ve Gupta (2006) çalışmalarında demontaj siparişi sistemi için kullanıcılardan ömrünün sonuna gelmiş olan ürünlerden kaç tanesinin geri alınması gerektiğine karar veren model önermişlerdir. Gelir maksimum, maliyet minimum ve toplam kar maksimum yapılmak istenmiştir. Amaçların üyelik fonksiyonları doğrusal olarak oluşturulmuş ve amaçların sınırları karar verici tarafından belirlenmiştir. Çözüm aşamasında ağırlıklı max-min modeli kullanılmıştır.

Chiang Lin ve Lin (2008) çalışmalarında kişisel finansal yatırım kararı için çok amaçlı matematiksel model önermişlerdir. Karar vericinin amaçlara verdiği ağırlıklar önemli olduğu için çözüm aşamasında ağırlıklı max-min modelini kullanmışlardır. Bir evin hangi dönemde alınması gerektiğine karar vermek amacı ile yaşam giderlerinin maksimum, kendi evini alma zamanının minimum, evin fiyatının maksimum ve emekli maaşının maksimum olması amaçlanan model kurulmuştur. Amaçların en iyi ve en kötü değerleri karar verici tarafından belirlenmiş ve üyelik fonksiyonları doğrusal olarak oluşturulmuştur.

Lin ve Liu (2008) çalışmalarında portföy seçimi için yatırımdan geri dönüşü maksimum yapmayı, riski minimum yapmayı amaçlayan çok amaçlı modelin çözümü için ağırlıklı max-min modelini kullanmışlardır. Ayrıca büyük problemlerde çözüm süresi uzayabileceğinden dolayı modelin çözümü için genetik algoritma geliştirmişlerdir.

Amid, Ghodsypour ve O'Brien (2011) çalışmalarında tedarik zincirinde tedarikçi seçiminde ürünün hangi tedarikçiden alınması gerektiğine karar veren (tek bir ürün için) çok amaçlı modelin çözümü için ağırlıklı max-min modelini kullanmışlardır. Amaçlar fiyatın minimum olması, servis kalitesinin maksimum olması ve hizmet kalitesinin maksimum olmasıdır. Amaçların üyelik fonksiyonu

olarak doğrusal üyelik fonksiyonu kullanılmıştır. Amaçlara atanacak olan ağırlıklar AHP yöntemi kullanılarak elde edilmiştir.

Dastkhan, Gharneh ve Golmakani (2011) çalışmalarında portföy seçimi için işlem maliyetlerini ve işlem miktarlarını minimum yapmayı amaçlayan matematiksel modellerinin çözümü için ağırlıklı max-min modelini kullanmışlardır ve S-şekilli üyelik fonksiyonu tanımlanmıştır. Modeli çözmek için genetik algoritma geliştirmişlerdir.

2.3.4. Ağırlıklı toplam operatörü

Tiwari, Dharmar ve Rao (1987) üyelik fonksiyonlarının ağırlıklı toplamlarını maksimum yapmayı amaçlayan “ağırlıklı toplam” (weighted additive) operatörünü önermişlerdir. Modelin matematiksel gösterimi (2.27)’de olduğu gibidir.

$$\begin{aligned} \mu_k &\in [0,1] \quad \forall k = 1, 2, \dots, K \\ x &\in X \\ \text{kısıtları altında} \\ \text{enb} \sum_{k=1}^K w_k \mu_k(x) \end{aligned} \quad (2.29)$$

Burada w_k k. amacın ağırlığıdır ve amaçların ağırlıklarının toplamı 1’e eşittir. Eğer modelin sağ taraf sabiti bulanık ise bu model (2.28)’de gösterilen modele eşdeğer olacaktır.

$$\begin{aligned} \mu_k(x) &\geq \lambda_k \quad \forall k = 1, \dots, K \\ \mu_r(x) &\geq \gamma_r \quad \forall r = 1, \dots, h \\ g_p(x) &\leq b_p \quad \forall b = h + 1, \dots, m \\ \lambda_k, \gamma_r &\in [0,1] \\ \text{kısıtları altında} \\ \text{enb} \sum_{k=1}^K w_k \lambda_k + \sum_{r=1}^h \beta_r \gamma_r \end{aligned} \quad (2.30)$$

Burada w_k amaçlara atanan ağırlıkları β_r bulanık kısıtlara atanan ağırlıkları göstermektedir ve tüm ağırlıkların toplamı 1’e eşittir. k amaç sayısını, r

sağ taraf sabiti bulanık olan kısıt sayısını, p ise sağ taraf sabiti bulanık olmayan kısıt sayısını göstermektedir. $\mu_r(x)$ ise r . bulanık kısıt için üyelik fonksiyonudur.

Amid, Ghodsypour ve O'Brien (2006) çalışmalarında tedarik zincirinde tedarikçi seçimi için bulanık çok amaçlı matematiksel model kullanmıştır. Amaçlar minimum fiyat, maksimum servis seviyesi ve maksimum kalite seviyesidir. Karar değişkeni, ürünün hangi tedarikçilerden ne kadar alınması gerektiğidir. Sağ taraf sabitinde belli bir esneklik olan kısıtlar için doğrusal üyelik fonksiyonları tanımlanmıştır. Bulanık kısıtlar için de karar verici tarafından ağırlıklar atanmıştır. Çözüm aşamasında ağırlıklı toplam operatörü kullanılmıştır.

Yücel ve Güneri (2011) çalışmalarında tedarikçi seçimi için ağırlıklı toplam operatörünü kullanmışlardır. Amaçlar ve bulanık kısıtlar için doğrusal amaç fonksiyonu kullanılmıştır. Ağırlıkların elde edilmesine ilişkin bulanık pozitif ideal ve bulanık negatif ideal değerleri arasındaki uzaklığa dayanan metodu önermişlerdir.

Araz, Selim ve Özkarahan (2007) acil servislerde araçların yer seçimi problemi için çok amaçlı model önermişlerdir. Modelde bir araç tarafından kapsanan nüfus ve yedek araç tarafından kapsanan nüfus maksimum yapılmak istenirken, toplam dolaşım mesafesi minimum yapılmak istenmektedir. Amaçların en iyi ve en kötü değerleri karar verici tarafından belirlenmiş ve üyelik fonksiyonları doğrusal olarak oluşturulmuştur. Çözüm aşamasında ağırlıklı toplam operatörü kullanılmıştır.

Selim, Araz ve Özkarahan (2008) işbirliğine dayalı üretim-dağıtım planlaması problemi için üreticinin karını, dağıtım merkezlerinin karını maksimum yapmak, perakendecilerin maliyetlerini ve perakendecilerin depo miktarını minimum yapmak şeklinde 4 amacı olan matematiksel model önermişlerdir. Bir firma tarafından tüm kararların alındığı merkezi tedarik zincirinin planlanması için modelin çözüm aşamasında ağırlıklı toplam operatörünü kullanmışlardır.

Amid, Ghodsypour ve O'Brien (2009) fiyat indirimlerini göz önüne alarak tedarikçi seçimi için maliyeti, reddedilen ürün sayısını ve geç gelen sevkiyat sayısını minimum yapmayı amaçlayan model önermişlerdir. Tedarikçiler, alınan

miktara göre fiyat vermektedirler. Model ile hangi fiyat seviyesinde hangi tedarikçiden ne kadar ürün alınmalı sorusu cevaplanmak istenmiştir.

Jazemi, Ghodsypour ve Gheidar-Kheljani (2011) çalışmalarında tedarikçi seçimine yönelik olarak hangi tedarikçiden ne kadar ürün alınmasını belirlemek için sağlam gelen ürün sayısını maksimum yapmayı, geç sevkiyatların sayısını minimum yapmayı ve alıcının ve tedarikçinin yıllık maliyetini minimum yapmayı hedefleyen çok amaçlı model önermişlerdir. Çözüm aşamasında ağırlıklı toplam operatörünü kullanmışlardır.

2.3.5. Bulanık ve operatörü

Werner (1988) min operatörü ile toplam operatörünü birleştiren “bulanık ve (fuzzy and)” operatörünü önermiştir. μ_D olarak belirtilen birleşim (2.31)’de gösterilmiştir. K amaç sayısını gösterirken, γ ise min operatörü ile toplam operatörüne verilen ağırlığı temsil etmektedir.

$$\mu_D(x) = \gamma \min_k \mu_k + (1 - \gamma) \left(\frac{1}{K} \right) \sum_k \mu_k(x) \quad (2.31)$$

Birleşim maksimize edilmek istenildiğinden, $\lambda = \min_k \mu_k(x)$ ve $\lambda_k = \mu_k(x)$ olarak ele alındığında bulanık ve operatörünün matematiksel gösterimi (2.32)’da olduğu gibidir.

$$\begin{aligned} \mu_k(x) &\geq \lambda + \lambda_k \quad \forall k = 1, 2, \dots, K \\ \lambda &\in [0, 1] \\ x &\in X \\ \text{kısıtları altında} \\ \text{enb } \gamma \lambda + (1 - \gamma) \left(\frac{1}{K} \right) \sum_k \lambda_k \end{aligned} \quad (2.32)$$

Topaloğlu ve Selim (2007) hemşire çizelgeleme problemi için çok amaçlı model önermişlerdir. Çizelgeleme için 2 haftalık periyot, günde 3 vardiya bulunmaktadır. Yasal düzenlemeler gereği hemşirelerin en az ve en çok çalışabilecekleri saatler bellidir. Amaçlar ise her bir gündeki her bir vardiya için

çalışan hemşire sayısını minimum yapmak ve hemşirelerin her bir isteğinden olan sapmayı minimum yapmaktır. Hemşirelerin istekleri toplam çalışma yükü, istenilen vardiya türü, izinli gün sayısı ve çalışma biçimidir. Çözüm aşamasında bulanık ve operatörü kullanılmıştır. γ değeri için 0 ile 1 arasında 0,1 aralıklarla bütün değerler verilmiş ve çözümler elde edilmiştir. γ değeri 0 olduğu zaman min operatörü ile çözüm, γ değeri 1 olduğu zaman toplam operatörü ile çözüm yapılmış olur. Çözümde bulanık ve operatörünün daha iyi sonuç verdiği söylenebilir ancak farklı γ değeri değerleri için bir baskınlık söz konusu değildir.

Selim ve Özkarahan (2008) çalışmalarında tedarik zinciri dağıtım ağı tasarımı için çok amaçlı matematiksel modeli kullanmışlardır. Amaçları toplam taşıma maliyetini minimum yapmak, yatırımları minimum yapmak ve toplam servis seviyesini maksimum yapmaktır. Birinci amaç için doğrusal, diğer amaçlar için üçgen üyelik fonksiyonu kullanılmıştır. Çözüm aşamasında bulanık ve operatörü kullanılmıştır. Çalışmada amaçların önemlerinin aynı olmayabileceği durumunu göz önüne alarak bulanık ve operatörüne amaçların ağırlıkları eklenmiştir. Modelin matematiksel ifadesi (2.30)'da gösterilmiştir.

$$\begin{aligned} \mu_k(x) &\geq \lambda + \lambda_k \quad \forall k = 1, 2, \dots, K \\ \lambda &\in [0, 1] \\ x &\in X \\ \text{Kısıtları altında} \\ \text{enb } \gamma\lambda + (1 - \gamma) \sum_k w_k \lambda_k \end{aligned} \tag{2.33}$$

Elde edilen çözüm karar verici tarafından yeterli bulunmazsa γ değeri ve amaçların en kötü değerleri bulunan çözüme göre güncellenerek çözüm işlemi tekrarlanır.

Selim, Araz ve Özkarahan (2008) çalışması ağırlıklı toplam operatörü başlığı altında bahsedilmişti. Bu çalışmada kararların çeşitli firmalar tarafından verildiği merkezi olmayan tedarik zinciri planlaması için modelin çözümünde bulanık ve operatörü kullanılmıştır.

Topaloğlu ve Selim (2010) hemşire çizelgeleme problemi için olan çalışmaları 2007 yılında yaptıkları çalışmaları ile birkaç yönden ayrılmaktadır. Her bir gündeki her bir vardiya için çalışan hemşire sayısını minimum yapmak

şeklinde olan amaç bu çalışmada istenilen hemşire miktarından sapmayı minimum yapmak olarak tanımlanmıştır. Diğer amaçlar aynıdır. Çözüm aşamasında bulanık ve operatörünün yanı sıra min operatörü ve iki aşamalı yaklaşım da kullanılmış ve sonuçlar karşılaştırılmıştır. Min operatörü kullanılarak bulunan sonuçlar diğer iki yöntem ile bulunan sonuçlara göre daha kötüyken, iki aşamalı yaklaşım ile bulanık ve operatörü kullanılarak bulunan sonuçlarda birbirine baskınlık söz konusu değildir.

Zarandi, Davari ve Sisakht (2011) çalışmalarında ana dağıtım üssü yer seçimi problemi için ağırlıklı güvenilirliğini ve ağırlık en küçük güvenilirliğini maksimum yapmayı amaçlayan model önermişlerdir. Çözüm aşamasında bulanık ve operatörünü Selim ve Özkarahan (2008)'in önerdiği şekilde kullanmışlardır.

Zarandi, Sisakht ve Davari (2011) hem ileri hem de geri akışların olduğu kapalı düzen tedarik zinciri modellemesi (closed-loop supply chain) için çok amaçlı model önermişlerdir. Birinci amaç fabrikalardan depolara, depolardan perakendecilere giden ürünün taşıma maliyetlerinin, perakendecilerden fabrikalara geri dönen ürünün taşıma maliyetlerinin ve fabrikaların ve depoların sabit maliyetlerinin toplamını minimum yapmaktır. İkinci amaç fabrika ve depo açma maliyetini minimum yapmaktır. Üçüncü amaç, perakendecilere sağlanan servis seviyesini maksimum yapmaktır. Dördüncü amaç ise fabrikalara geri dönüş akışında fabrikalara sağlanan servisi maksimum yapmaktır. Çözüm aşamasında bulanık ve operatörünü Selim ve Özkarahan (2008)'in önerdiği şekilde kullanılmıştır.

3. ÇOK AMAÇLI MATEMATİKSEL MODELLERDE AMAÇLARIN AĞIRLIKLARININ BELİRLENMESİ

Çok amaçlı karar verme problemleri için geliştirmiş çözüm yöntemlerinde amaçlara ağırlıkların atanması gerekebilmektedir. Bu yöntemlerde amaçların önemi eşit varsayılacağı gibi amaçlara farklı ağırlıklarda atanabilmektedir. Bu bölümde öncelikle literatürde yapılan çalışmalarda amaçların ağırlıkların belirlenmesi için nasıl yöntemler kullanıldığından bahsedilecek ardından önerilen metottan bahsedilecektir.

3.1. Literatürde Kullanılan Yöntemler

Literatürdeki çalışmalarda amaçların ağırlıkları belirleme yöntemleri incelenmiştir. Bu çalışmalarda ağırlıklar herhangi bir yöntem kullanılmaksızın karar verici tarafından belirlenebilmekte veya ağırlıkların belirlenmesi için literatürde var olan yöntemler kullanılmaktadır. Literatürde bu ağırlıkların nasıl belirlenmesi gerektiğine dair yeni yöntem öneren çalışmalar da bulunmaktadır.

El-Wahed ve Abo-Sinna (2001) çalışmalarında çok amaçlı doğrusal problemlerin çözümü için bulanık hedef programlamayı kullanmışlardır. Öncelikle amaçlar karar verici tarafından önceliklerine göre alt amaç gruplarına ayrılmaktadır. Her bir alt amaç grubu içindeki amaçların ağırlıkları belirlenmektedir. Bunun için amaçların üyelik fonksiyonu değerlerinden oluşan matris oluşturulmakta ve bu matris yardımı ile önerilen doğrusal model ile amaçların ağırlıkları belirlenmektedir. Bu ağırlıklar yardımı ile her bir alt amaç grubu kendi içinde toplanarak tek amaca indirgenmektedir. Yeni oluşturulan amaç fonksiyonlarının arasındaki açılar önerilen algoritma ile belirlenerek, bu açılar yardımı ile önerilen ikinci doğrusal programlama yöntemi kullanılarak bu amaçların ağırlıkları belirlenmektedir. Ağırlıklar belirlendikten sonra hedef programlama modeli kurularak problemin çözümü bulunmaktadır.

Amid, Ghodsypour ve O'Brien (2006) tedarikçi seçimine yönelik çalışmalarında ağırlıkları belirlemeye ilişkin bir yöntem kullanmamışlardır. Ağırlıkları karar verici kendi inisiyatifine göre belirlemiştir.

Kongar ve Gupta (2006) demontaj siparişi sistemi için olan çalışmalarında amaçların ağırlıkları karar verici tarafından belirlenmiştir.

Selim ve Özkarahan (2008) tedarik zinciri dağıtım ağı tasarımına yönelik çalışmalarında amaçların ağırlıklarını belirlemek için herhangi bir yöntem kullanmamışlar ve karar vericinin inisiyatifine göre belirlemişlerdir.

Chiang Lin ve Lin (2008) kişisel finansal planlamaya yönelik olan çalışmalarında amaçların ağırlıkları karar verici tarafından belirlenmiştir.

Selim, Araz ve Özkarahan (2008) işbirliğine dayalı üretim-dağıtım planlaması problemine yönelik olan çalışmalarında modelin çözümünde kullanılan ağırlıklı toplam operatörü için gerekli olan amaçların ağırlıklarını karar verici belirlemiştir.

Liang (2009) proje yönetimine yönelik çalışmasında çözüm aşamasında iki aşamalı yöntemi kullanmış ve ağırlıkların belirlenmesini karar vericiye bırakmıştır.

Yücel ve Güneri (2011) tedarikçi seçimine yönelik çalışmalarında amaçların ve bulanık kısıtların ağırlıklarının belirlenmesi için yeni bir yöntem önermişlerdir. Sözel değerleri yamuk bulanık sayı olarak ele alınmıştır. Karar vericiler tarafından her bir amaç ve bulanık kısıt çok yüksek, yüksek, orta yüksek, orta, orta düşük önemli, düşük ve çok düşük önemli olmak üzere puanlanmıştır. Bulanık en iyi değer ve bulanık en kötü değere olan uzaklıklar göz önüne alınarak ağırlıklar belirlenmiştir.

Amid, Ghodsypour ve O'Brien (2011) tedarikçi seçimine yönelik çalışmalarında amaçların ağırlıklarını AHP (Analytic Hierarchy Process) yöntemi ile belirlemişlerdir.

Liang, Huang ve Yang (2011) proje yönetimi kararları için kullandıkları modelin çözümü için iki aşamalı yaklaşımı kullanmışlardır ve amaçların ağırlıklarını karar vericiler belirlemiştir.

Liang ve ark. (2011) çok ürünlü ve çoklu periyotlu toplam üretim planlama problemini amaçların ağırlıklarını karar vericiden alarak iki aşamalı yaklaşım ile çözmüşlerdir.

Jazemi, Ghodsypour ve Gheidari-Kheljani (2011) tedarikçi seçimine yönelik olan çalışmalarında çözüm aşamasında ağırlıklı toplam operatörünü kullanmışlardır ve amaçların ağırlıklarını karar verici belirlemiştir.

Rabbani ve ark. (2012) hücre oluşturmaya yönelik olan çalışmalarında çözüm aşamasında iki aşamalı yaklaşımı kendi önerdikleri yeni amaç fonksiyonuna göre kullanmışlar ve amaçların ağırlıklarını karar vericinin inisiyatifine göre belirlemişlerdir.

3.2. Önerilen Metot

Literatürdeki çalışmalarda amaçların ağırlıkları eşit kabul edilmiş veya karar vericiye sorulmuştur. Ağırlıkların belirlenmesine ilişkin çok az yöntem önerilmiştir. Bu yöntemlerde karar vericinin tercihlerini içermektedir. Karar vericinin amaçlar hakkında detaylı bilgisi olmadığı durumlarda, amaçların ağırlıkları yanlış belirlenebilir.

Karar vericinin tercihleri olmaksızın amaçların ağırlıklarının belirleneceği bir yöntem ihtiyacı görülmesi sebebi ile Belenson ve Kapur (1973)'ün geliştirdikleri yöntemde ödemeler matrisinde normalleştirme bir adımda yapılmasını sağlamak için amaçların üyelik fonksiyonları kullanılarak yeni bir metot önerilmiştir. Yöntemde oyun teorisinin iki oyunculu sıfır toplamlı oyun kuramına dayandığı için öncelikle kısaca bu teoriden bahsedilecek ardından metot anlatılacaktır.

3.2.1. Oyun teorisi: İki oyunculu sıfır toplamlı oyun

a) Oyun teorisi:

Oyun teorisi karmaşık yararların mücadelesini açıklayan matematiksel bir yaklaşımdır. (Öztürk, 2007) Çelişen amaçları olan zeki karşıtların bulunduğu karar durumlarını inceler. Rekabet eden ürünler için reklam kampanyasının başlatılması, iki düşman ordunun savaş stratejilerinin planlanması tipik örneklerdir (Taha, 2007).

Oyunların şans kuramı 17. Yüzyılda ortaya atılmıř ve olasılık kuramı adı verilen matematik dalının gelişmesine kaynak olmuřtur. Oyun kuramına ilk değinen matematikçi Emile Barel'dir. Stratejik oyunlar kuramının bulucusu olan J. Von Neumann bu konu üzerinde ilk çalışmasını 1928 yılında yayımlamıřtır (Öztürk, 2007).

b) İki oyunculu sıfır toplamlı oyun:

Bir oyun çatıřmasında, oyuncu olarak bilinen iki rakibin her biri sonlu ya da sonsuz sayıda alternatif ya da stratejiye sahiptir. Her bir strateji çiftine baėlı sonuç, bir oyuncunun diėerine ödeme yapmasıdır. Bu oyunlar iki oyunculu-sıfır toplamlı oyunlar diye bilinir. Çünkü birinin kazancı diėerinin kaybına eşittir. Bu bakımdan oyunu, bir oyuncuya yapılan ödemeler cinsinden özetlemek yeterlidir (Taha, 2007).

1. Oyuncunun (A) kazancını 2. Oyuncunun (B) kaybını gösteren ve ödemeler matrisi olarak isimlendirilen matris Çizelge 3.1'de gösterilmektedir. (Öztürk, 2007) A oyuncusunun m adet stratejisi var iken B oyuncusunun n adet stratejisi bulunmaktadır. Burada a_{ij} A oyuncusunun i. stratejisini, B oyuncusunun j. stratejisini kullandığı zaman A oyuncusunun kazancını B oyuncusunun kaybını gösterir. Ödemeler matrisinin elemanları pozitif, negatif veya sıfıra eşit olabilir. Bu matrisin herhangi bir elemanı pozitif ise sütunda yer alan oyuncu satırda yer alan oyuncuya bu miktarda ödeme yaparken eėer bir eleman negatif ise satırdaki oyuncu sütundaki oyuncuya bu değerin mutlak değerine eşit bir ödeme yapar. Matrisin bu elemanı sıfır ise oyunculardan hiçbirisi birbirine ödeme yapmaz. Oyuna katılan her oyuncunun kendi kazancını en yüksek veya kaybını en az kılmak için akılcı hareket ettiėi kabul edilir (Öztürk, 2007).

Çizelge 3.1. İki oyunculu-sıfır toplamlı oyunun ödemeler matrisi

A oyuncusunun stratejileri	B oyuncusunun stratejileri				
	Strateji 2.1	Strateji 2.2	...	...	Strateji 2.n
Strateji 1.1	a_{11}	a_{12}	...	...	a_{1n}
Strateji 1.2	a_{21}	a_{22}			a_{2n}
.	...	...	a_{ij}	...	...
.	...	...	...	...	...
Strateji 1.m	a_{m1}	a_{m2}	...	...	a_{mn}

İki oyunculu-sıfır toplamlı oyunlarda oyunun çözümü, her bir oyuncu için kötünün iyisini garanti etmeye dayanır. Oyunlar çatışma üzerine kurulduğu için optimum çözüm her bir oyuncu için herhangi bir değişikliğin oyunculardan hiç birinin kazancını geliştirmediği bir veya daha fazla stratejiyi seçer. Bu çözümler önceden belirlene olasılıklara göre tekil saf strateji veya çeşitli stratejilerden oluşan karma strateji şeklinde olabilir (Taha, 2007). Eğer oyunun her oynanmasında oyuncular aynı stratejiyi kullanıyorlarsa tam strateji kullanıyorlar demektir. Bu strateji bazı oyunlar için optimal strateji olabilir. Eğer herhangi bir tam strateji bir oyuncu için optimal ise bu tam strateji diğer oyuncu için de optimaldir. Bu tam stratejiler maximin ve minimax kuralına göre ulaşılan değerleri veren stratejilerdir. Genellikle oyunlarda daha etkili olan karma strateji kullanılır. Karma strateji, tam stratejiler takımındaki olasılık dağılımı ile tanımlanır (Öztürk, 2007).

A oyuncusu B oyuncusunun hangi stratejiyi kullanacağını düşünmeden kendisi için x gibi optimal strateji vektörünü elde etmeye çalışır. x vektörü A oyuncusuna oyundan maksimum beklenen kazancı sağlar. Buna karşılık B oyuncusu da A oyuncusunun beklenen kazancını en aza indirecek kendi strateji vektörü y 'yi araştırır (Öztürk, 2007).

Oyun için iki temel teorem vardır (Öztürk, 2007).

Teorem 1: m satır ve n sütunu gösterirse $(m \times n)$ bir dikdörtgen oyundur. Her dikdörtgenin bir oyun değeri vardır. Dikdörtgen bir oyunda herhangi bir oyuncunun her zaman optimal bir stratejisi vardır. Bunu (3.1)'deki gibi ifade edebiliriz.

$$B.D.(x^*, y^*) = v \quad (3.1)$$

Burada $B.D.(x^*, y^*)$ A oyuncusunun beklenen değeridir ki bu da oyunun değeri v 'dir.

Teorem 2: Herhangi bir dikdörtgen oyunda A ve B oyuncuları için oyunun değeri v , optimal strateji vektörleri de x^*, y^* olsun.

- A oyuncusunun her tam strateji vektörü x_i için $B.D.(x_i, y^*) \leq v$ dir. Eğer B oyuncusu optimal stratejisini oynar ise, A oyuncusunu v değerinden daha fazla kazandırabilecek tam bir strateji yoktur.

- B oyuncusunun her tam strateji vektörü y_t için $B.D.(x^*, y_t) \geq v$ dir. Eğer A oyuncusu optimal stratejisini oynar ise B oyuncusunun oynayacağı hiçbir tam strateji A oyuncusunun kazancını v 'den daha aza indiremeyecektir.

c) İki oyunculu sıfır toplamlı oyunun doğrusal programlama ile çözümü:

İki oyunculu sıfır toplamlı oyunlar doğrusal program olarak ifade edilebildiği için oyun teorisi ile doğrusal programlama arasında güçlü bir ilişki vardır. Dantzig 1947'de simpleks yöntemini ilk ortaya koyduğunda J. Vn Neumann bu ilişkiyi fark ederek ve doğrusal programlamadaki dualite kavramına dayanarak doğrusal programlamayı (DP) karma stratejili iki oyunculu sıfır toplamlı oyunun çözümü için kullanılmıştır (Taha, 2007). 1. oyuncunun n adet stratejisi, 2. oyuncunun m adet stratejisi olduğu durumda, 1. oyuncu için doğrusal programlama modeli (3.2)'de, 2. Oyuncu için doğrusal programlama modeli (3.3)'de görülmektedir (Winston, 2004).

1. oyuncu için DP :

$$\begin{aligned} \sum_{i=1}^m a_{ij}x_i &\geq v \quad \forall j = 1, 2, \dots, n \\ \sum_{i=1}^m x_i &= 1 \\ x_i &\geq 0 \quad \forall i = 1, 2, \dots, m \\ \text{Kısıtları altında} \\ \text{enb } v \end{aligned} \tag{3.2}$$

2. oyuncu için DP :

$$\begin{aligned} \sum_{j=1}^n a_{ij}y_j &\leq w \quad \forall i = 1, 2, \dots, m \\ \sum_{j=1}^n y_j &= 1 \\ y_j &\geq 0 \quad \forall j = 1, 2, \dots, n \\ \text{Kısıtları altında} \\ \text{enk } w \end{aligned} \tag{3.3}$$

Burada x_i 1. oyuncunun i. stratejisini kullanma olasılığını, y_j 2. oyuncunun j. stratejisini kullanma olasılığını göstermektedir. 1. oyuncu, kazancını en az oyunun değeri v'den az olmamasını garanti ederken 2. oyuncu da, w'dan fazla olmayan bir ödemeyi sağlamaya çalışacaktır.

3.2.2. Belenson ve Kapur'un yöntemi

Belenson ve Kapur (1973) iki oyunculu sıfır toplamli oyun kuramını kullanarak ağırlıkların belirlenmesi için yöntem önermişler ve bu ağırlıkları ağırlıklı toplam metodunda kullanarak, çok amaçlı doğrusal problemler için çözüm bulmuşlardır. Oyunda 1. oyuncunun stratejileri amaçlar iken 2. oyuncunun stratejileri ise problem kısıtları altında her bir amacı tek tek optimize eden çözümlerdir. Eğer problem K adet amaç fonksiyonuna sahip ise oyunda her iki oyuncunun da K adet stratejisi olur. Çizelge 3.2'de görülen ödemeler matrisi oyun kuramında olduğu gibi oluşturulur.

Çok amaçlı karar vermenin yapısı gereği amaçlar birbiriyle çelişir durumda olabilirler ve alabildikleri sayısal değerler birbirinden çok farklı olabilir. Örneğin, ürün güvenilirliği maksimize edilmek isteniyorsa bu amacın değeri 0 ile 1 arasında değişiklik gösterecekken, aynı zamanda karlılık da maksimum yapılmak isteniyorsa bu amacın değeri de 0 ile milyonlar arasında değişecektir. Bu durumda da ödemeler matrisinde dengesizlik olacak bazı satırlar daha baskın görünecektir. Bu sebepten bazı normalleştirme teknikleri önerilmiştir. Ödemeler matrisi normalleştirildikten sonra oyun kuramının doğrusal programlama ile çözümü yardımı ile normalleştirilmiş ağırlıklar elde edilmiştir. Normalleştirilmiş ağırlıkların dönüşümü ile amaçların ağırlıkları elde edilir ve bu ağırlıklar ağırlıklı toplam metodunda kullanılarak problemin çözümü elde edilmiş olur.

Çizelge 3.2. Ödemeler Matrisi

1. oyuncunun stratejileri	2. oyuncunun stratejileri				
	x^1	x^2	...	...	x^k
z_1	$f_1(x^1)$	$f_1(x^2)$	...	...	$f_1(x^k)$
z_2	$f_2(x^1)$	$f_2(x^2)$	...	...	$f_2(x^k)$
·	...	...	$f_i(x^j)$	...	...
·	...	...	...	...	...
z_k	$f_k(x^1)$	$f_k(x^2)$	...	...	$f_k(x^k)$

Kısaca Belenson Kapur algoritmasının adımları şöyledir:

1. Çok amaçlı doğrusal programlama probleminin her bir amacını teker teker optimize ederek çöz.
2. Ödemeler matrisini oluştur.
3. Ödemeler matrisinde negatif olan değer varsa K sabitini (3.4)'e göre tanımla.

$$K = -\min_{k,l} f_{kl} \quad (3.4)$$

4. Her bir amaç için M_k değerini (3.5)'e göre belirle.

$$M_k = \max_l f_{kl} + K \quad (3.5)$$

Denklem (3.6)'e göre normalleştirilmiş ödemeler matrisini oluştur.

$$f'_{kl} = \frac{f_{kl} + K}{M_k} \quad (3.6)$$

5. İki oyunculu sıfır toplamli oyunun çözümü için kullanılan doğrusal programlamayı kullanarak optimal ağırlık vektörünü λ' elde et.
6. Orijinal ağırlıkları (3.7) ve (3.8)'e göre elde et.

$$n_k = \frac{\lambda'_k}{M_k} \quad \forall k = 1, \dots, p \quad (3.7)$$

$$\lambda_k^* = \frac{n_k}{\sum_{k=1}^p n_k} \quad \forall k = 1, \dots, p \quad (3.8)$$

7. Eşit doğrusal programlamayı (3.9)'e göre oluştur ve problem kısıtları altında çöz.

$$enb \ f_{p+1}(x) = \sum_{k=1}^p \lambda_k^* f_k(x) \quad (3.9)$$

3.2.3. Önerilen metodun adımları

Amaçların değerleri birbirinden çok farklı olabilir. Örneğin sistem güvenilirliği ile elde edilen kar aynı anda maksimize edilmek istenildiğinde sistem güvenilirliğinin değeri 0 ile 1 arasında değişirken kar miktarı milyonlar ile ifade edilecektir. Farklı değerleri olan ödemeler matrisi oyun kuramı ile çözülmek istenildiğinde sütunlar arasında baskınlık olmaması için bu matrisin normalleştirilmesi gerekmektedir. Bunun için bir önceki bölümde anlatıldığı gibi Belenson ve Kapur, normalleştirme önerilerinde bulunmuşlardır ve normalleştirilme birkaç adımda sağlanabilmiştir.

Amaç fonksiyonu hangi değerler ile ifade ediliyor olursa olsun ve ayrıca amaç ister maksimum ister minimum yapılmak istensin üyelik fonksiyonu değeri 0 ile 1 arasında değişeceği ve üyelik fonksiyonu değerinin anlamı her durumda aynı olduğu için ödemeler matrisinde amaçların üyelik fonksiyonlarının kullanıldığında ödemeler matrisinin normalleştirilmesi tek adımda olacaktır. Bu sebeple Belenson ve Kapur'un yönteminde kullanılan ödemeler matrisinde, amaçların değerleri yerine amaçların üyelik fonksiyonu değerlerinin kullanılması önerilmiştir. Bu durumda “0” değeri problem kısıtları altında amacın en kötü değerine ulaştığını gösterirken, “1” değeri amacın en iyi değerine ulaştığını göstermektedir. Diğer değerler ise amacın en iyi değere olan yakınlığını göstermektedir. Bu metot Dinç Yalçın, Erginel (2011) tarafından sunulmuştur.

Önerilen ödemeler matrisi Çizelge 3.3'te gösterilmiştir. Çizelgede z_k k. amaç fonksiyonunu gösterirken, x^k k. amaç fonksiyonunun en iyi değerini göstermektedir. $\mu_i(x^j)$ ise j. amaç fonksiyonunun en iyi değerinde i. amacın üyelik fonksiyonu değerini göstermektedir.

Çizelge 3.3. Önerilen yöntemin ödemeler matrisi

1. oyuncunun stratejileri	2. oyuncunun stratejileri				
	x^1	x^2	...	...	x^k
z_1	$\mu_1(x^1)$	$\mu_1(x^2)$	...	...	$\mu_1(x^k)$
z_2	$\mu_2(x^1)$	$\mu_2(x^2)$	...	...	$\mu_2(x^k)$
·	...	...	$\mu_i(x^j)$	...	...
·	...	...	...	...	...
z_k	$\mu_k(x^1)$	$\mu_k(x^2)$	...	...	$\mu_k(x^k)$

Eğer ağırlıklar $w_1, w_2, \dots, w_k$ şeklinde tanımlanırsa iki oyunculu sıfır toplamlı oyunun doğrusal programlama yöntemi (3.15)'de görüldüğü gibi revize edilmiş olur.

$$\sum_{i=1}^k w_i \mu_i(x^j) \geq v \quad \forall j = 1, 2, \dots, K$$

$$\sum_{i=1}^k w_i = 1$$

(3.15)

Kısıtları altında

enb v

Önerilen metodun adımları şöyledir:

1. Çok amaçlı doğrusal programlama probleminin her bir amacının teker teker en iyi ve en kötü değerlerini hesapla.
2. Amaçların üyelik fonksiyonlarını oluştur ve her bir ideal çözüm için amaçların üyelik fonksiyonu değerlerini hesapla.
3. Önerilen biçimde ödemeler matrisini oluştur.
4. Doğrusal programlama modelini kullanarak amaçların ağırlıklarını hesapla.
5. 4. Adımdan elde edilen ağırlıkları kullanarak bulanık çözüm yaklaşımlarından biri ile çok amaçlı doğrusal programlama problemi için çözüm bul.

a) Sayısal örnek:

Önerilen yöntem Belenson ve Kapur'un (1973) çalışmasından alınan örnek ile sayısal olarak gösterilecektir.

Örnek:

$$\begin{aligned} x_1 + x_2 &\leq 7 \\ x_1 &\leq 5 \\ x_2 &\leq 3 \\ x_1, x_2 &\geq 0 \\ \text{Kısıtları altında} \\ \text{enb } f_1(x) &= 0,1x_1 + 0,2x_2 \\ \text{enb } f_2(x) &= 10x_1 - 5x_2 \end{aligned} \tag{3.16}$$

Amaçların ideal ve anti ideal değerleri $f_1^* = 1$, $f_1' = 0$ ve $f_2^* = 50$, $f_2' = 0$. İdeal çözümler ise $x^1 = (4,3)$ and $x^2 = (5,0)$. Amaçların üyelik fonksiyonları ise aşağıdaki gibi doğrusal olarak belirlenmiştir.

$$\mu_1(x) = \begin{cases} 1 & f_k > 1 \\ \frac{[f_1(x) - 0]}{[1 - 0]} & 0 \leq f_1(x) \leq 1 \\ 0 & f_k(x) < 0 \end{cases} \tag{3.17}$$

$$\mu_2(x) = \begin{cases} 1 & f_k > 50 \\ \frac{[f_2(x) - 0]}{[50 - 0]} & 0 \leq f_2(x) \leq 50 \\ 0 & f_2(x) < 0 \end{cases} \tag{3.18}$$

Problem için Belenson ve Kapur'un kullandığı ödemeler matrisi Çizelge 3.4'de, ideal çözümler için amaçların üyelik fonksiyonları değerlerinin hesaplanması ile elde edilen önerilen ödemeler matrisi ise Çizelge 3.5'deki görülmektedir.

Çizelge 3.4.Örneğin Belenson ve Kapur'un kullandığı ödemeler matrisi

1. oyuncunun stratejileri	2. oyuncunun stratejileri	
	x^1	x^2
z_1	1	0.5
z_2	25	50

Çizelge 3.5. Örneğin önerilen ödemeler matrisi

1. oyuncunun stratejileri	2. oyuncunun stratejileri	
	x^1	x^2
z_1	1	0.5
z_2	0.5	1

Önerilen yöntem için kullanılan doğrusal programlama modeli çözülerek amaçların ağırlıkları elde edilmiştir. Örnek için doğrusal programla modeli aşağıda gösterilmiştir.

$$1w_1 + 0.5w_2 \geq v$$

$$0.5w_1 + 1w_2 \geq v$$

$$\sum_{i=1}^2 w_i = 1 \quad (3.19)$$

$$w_i \geq 0 \quad \forall i = 1,2$$

Kısıtları altında

enb v

Doğrusal modelin çözümü ile elde edilen ağırlıklar $w_1 = 0.5$ ve $w_2 = 0.5$ olurken Belenson ve Kapur çalışmalarında ağırlıkları $w_1 = 0.98$ ve $w_2 = 0.02$ olarak bulmuşlardır. Önerilen metot iki amacın ağırlığını aynı bulurken, Belenson ve Kapur'un yaklaşımı birinci amacın ağırlığını ikinci amaca göre çok büyük bulmuştur. Bu sonuç ödemeler matrisindeki normalleştirme tekniğinden kaynaklanıyor olabilir. İlk amacın değeri 1 ile 0,5 arasında değişiyor iken ikinci amacın değeri 25 ile 50 arasında değişmektedir ve iki amacın değerleri arasında önemli bir fark bulunmaktadır. Belenson ve Kapur'un normalleştirme tekniği ile değeri küçük olan amacın ağırlığı daha fazla çıkarken, üyelik fonksiyonu kullanılması durumunda ağırlıklar eşit çıkmıştır. Çünkü üyelik fonksiyonu değeri amaçların alacağı sayı değerlerinin büyüklüğüne bağlı kalmaksızın, en iyi durumda 1 değerini, en kötü durumda 0 değerini ve diğer durumlarda 0 ile 1 arasında değer almaktadır.

Her iki yönteme göre elde edilen ağırlıklara göre problem min operatörü, iki aşamalı yaklaşım, ağırlıklı mx-min modeli ve ağırlıklı toplam operatörüne göre çözülmüştür. Ayrıca Belenson ve Kapur'un çalışmalarında problemin çözümü için

kullandıkları yöntem olan ağırlıklı toplam yöntemine göre de çözülmüştür. Sonuçlar Çizelge 3.6.'da verilmiştir.

Belenson ve Kapur'un çalışmalarında ağırlıklı toplam yöntemi ile amaç fonksiyonlarının değerleri $f_1(x) = 0.9$ ve $f_2(x) = 40$ olarak bulunurken, üyelik fonksiyonu değerleri $\mu_1(x) = 0.9$ ve $\mu_2(x) = 0.8$ olarak bulunmuştur. Önerilen metot ile elde edilen ağırlıklarla çözülen ağırlıklı toplam yöntemine göre amaç fonksiyonunun değerleri $f_1(x) = 0.5$ ve $f_2(x) = 50$ olarak bulunurken, üyelik fonksiyonu değerleri $\mu_1(x) = 0.5$ ve $\mu_2(x) = 1$ olarak bulunmuştur. Benzer şekilde Belenson ve Kapur'un yöntemi ile elde eden ağırlıklar ve önerilen metot için elde edilen ağırlıklar kullanılarak bulanık çözüm yaklaşımlarına göre sonuçlar elde edilmiştir.

Çizelge 3.6. Örneğin çözümleri

	Ağırlıklı toplam yöntemi		Min operatörü	İki aşamalı yaklaşım	
	Önerilen metodun ağırlıkları ile	Belenson ve Kapur'un yaklaşımının ağırlıkları ile		Önerilen metodun ağırlıkları ile	Belenson ve Kapur'un yaklaşımının ağırlıkları ile
$f_1(x)$	0.5	0.9	0.834	0.834	0.834
$f_2(x)$	50	40	41.65	41.65	41.65
$\mu_1(x)$	0.5	0.9	0.834	0.834	0.834
$\mu_2(x)$	1	0.8	0.833	0.833	0.833
$\sum_{k=1}^2 \mu_k(x)$	1.5	1.7	1.667	1.667	1.667

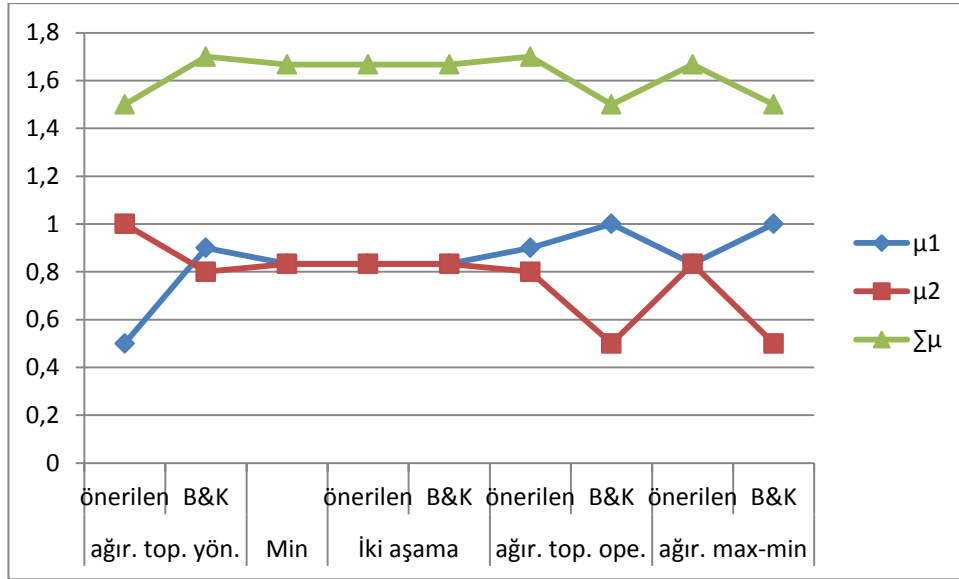
Çizelge 3.5. Örneğin çözümleri (devam)

	Ağırlıklı toplam operatörü		Ağırlıklı max-min modeli	
	Önerilen metodun ağırlıkları ile	Belenson ve Kapur'un yaklaşımının ağırlıkları ile	Önerilen metodun ağırlıkları ile	Belenson ve Kapur'un yaklaşımının ağırlıkları ile
$f_1(x)$	0.9	1	0.834	1
$f_2(x)$	40	25	41.65	25
$\mu_1(x)$	0.9	1	0.834	1
$\mu_2(x)$	0.8	0.5	0.833	0.5
$\sum_{k=1}^2 \mu_k(x)$	1.7	1.5	1.667	1.5

Çözümlere genel olarak baktığımızda çözümlerin birbirine baskın olmadığı görülebilir. İki aşamalı yaklaşım min operatörü ile aynı sonucu üretmiştir. İki aşamalı yaklaşımda her iki yöneme göre elde edilen ağırlıklar ile aynı sonuç elde edilmiş, ağırlıkların farklı olması sonucu değiştirmemiştir.

Eğer tüm sonuçlar toplam üyelik fonksiyonu değerine göre karşılaştırılırsa 1.7 değeri en iyi çözüm ile Belenson ve Kapur'un ağırlıkları kullanılarak, ağırlıklı toplam yöntemine ile ve önerilen metodun ağırlığı kullanılarak ağırlıklı toplam metodu ile elde edilmiştir. Fakat ağırlıklı max-min metodunda da önerilen metodun ağırlıkları kullanılarak elde edilen çözümün toplam üyelik derecesi değeri, Belenson ve Kapur'un ağırlıkları ile bulunan çözümünkine göre daha fazladır.

Örnek problemin çözümü ile elde edilen sonuçlar Şekil 3.1'de görülmektedir. İki amacın üyelik fonksiyonu değerleri (μ_1 ve μ_2) ile toplam üyelik fonksiyonu değerleri ($\sum\mu$) gösterilmiştir.



Şekil 3.1. Belenson ve Kapur'un ağırlıkları ve önerilen metodun ağırlıkları kullanılarak ağırlıklı toplam yöntemi, min operatörü, iki aşamalı yaklaşım, ağırlıklı toplam operatörü ve ağırlıklı max-min operatörü kullanılarak elde edilen çözümlerde amaçların üyelik fonksiyonu değerleri ve toplam üyelik fonksiyonu değeri

b) Bulanık çözüm yaklaşımlarında kullanılan problemlerin çözümü:

Ayrıca Li ve ark. (2006), Tiwari ve ark. (1987) ve Lin(2004) çalışmalarından alınan örnekler ilgili çalışmalarda önerilen bulanık çözüm yaklaşımları ile hem çalışmalardaki ağırlıklara göre hem de önerilen metotla belirlenmiş ağırlıklara göre çözülmüş ve sonuçlar karşılaştırılmıştır. Burada problemler birbirinden farklı olduğu için sonuçların karşılaştırılması için Zeleny (1982) tarafından önerilmiş olan uzaklık fonksiyonu (family distance function) kullanılmıştır. Ayrıca bu fonksiyon amaçlara verilen ağırlıklara göre farklılık göstermektedir. Denklem 3.24'te fonksiyon görülmektedir.

$$L_p(w, K) = \left[\sum_{k=1}^K w_k^p (1 - d_k)^p \right]^{\frac{1}{p}} \quad (3.20)$$

Burada p, uzaklığın ölçüsünü belirleyen parametredir. p 0,1 ve ∞ olarak alınmıştır. Böylelikle L_1, L_2, L_∞ değerleri hesaplanmıştır. L_1 lineer uzaklığı, L_2 Öklid uzaklığını ve L_∞ Chebychev uzaklığını göstermektedir. Bu değerler optimum çözüme olan uzaklığı gösterdiği için olabildiğince küçük olması istenir. Bu uzaklıkların hesaplanması (3.21), (3.22) ve (3.23)'de görülmektedir.

$$L_1(w, K) = 1 - \sum_{k=1}^K w_k d_k \quad (3.21)$$

$$L_2(w, K) = \left[\sum_{k=1}^K w_k^2 (1 - d_k)^2 \right]^{1/2} \quad (3.22)$$

$$L_\infty(w, K) = \max_k \{w_k (1 - d_k)\} \quad (3.23)$$

Burada d_k k. amaç için elde edilen çözümün k. amacın en iyi çözüme olan yakınlığıdır. Eğer amaç fonksiyonu maksimum yapılmak isteniyorsa (3.28)'e göre, eğer minimum yapılmak isteniyorsa (3.29)'ya göre hesaplanır.

$$d_k = \frac{f_k(x)}{f_k^*} \quad (3.24)$$

$$d_k = \frac{f_k^*}{f_k(x)} \quad (3.25)$$

Li ve ark. (2006)'nın çalışmalarında kullandıkları problem (3.26)'da görülmektedir.

$$\begin{aligned}
3x_1 + 4,5x_2 + 1,5x_3 + 7,5x_4 &= 150 \\
x_1, x_2, x_3, x_4 &\geq 0 \\
\text{Kısıtları altında} \\
\max z_1 &= 2x_1 + 5x_2 + 7x_3 + x_4 \\
\max z_2 &= 4x_1 + x_2 + 3x_3 + 11x_4 \\
\max z_3 &= 9x_1 + 3x_2 + x_3 + 2x_4 \\
\min z_4 &= 1,5x_1 + 2x_2 + 0,3x_3 + 3x_4 \\
\min z_5 &= 0,5x_1 + x_2 + 0,7x_3 + 2x_4
\end{aligned} \tag{3.26}$$

Tiwari ve ark. (1987)'nin çalışmalarında kullandıkları problem (3.27)'de görülmektedir.

$$\begin{aligned}
7x_1 + 5x_2 + 3x_3 + 2x_4 &\leq 98 \\
7x_1 + x_2 + 6x_3 + 6x_4 &\leq 117 \\
x_1 + x_2 + 2x_3 + 6x_4 &\leq 130 \\
9x_1 + x_2 + 6x_4 &\leq 105 \\
x_1, x_2, x_3, x_4 &\geq 0 \\
\text{Kısıtları altında} \\
\max z_1 &= 4x_1 + 2x_2 + 8x_3 + x_4 \\
\max z_2 &= 4x_1 + 7x_2 + 6x_3 + 2x_4 \\
\max z_3 &= x_1 - 6x_2 + 5x_3 + 10x_4 \\
\max z_4 &= 5x_1 + 3x_2 + 2x_4 \\
\max z_5 &= 4x_1 + 4x_2 + 4x_3
\end{aligned} \tag{3.27}$$

Lin (2004)'in çalışmasında kullandığı problem (3.28)'de verilmiştir.

$$\begin{aligned}
4x_1 + 2x_2 + 3x_3 &\leq 10 \\
x_1 + 3x_2 + 2x_3 &\leq 8 \\
x_3 &\leq 5 \\
x_1, x_2, x_3 &\geq 0 \\
\text{Kısıtları altında} \\
\max z_1 &= 3x_1 + x_2 + x_3 \\
\max z_2 &= x_1 - x_2 + 2x_3 \\
\max z_3 &= x_1 + 2x_2
\end{aligned} \tag{3.28}$$

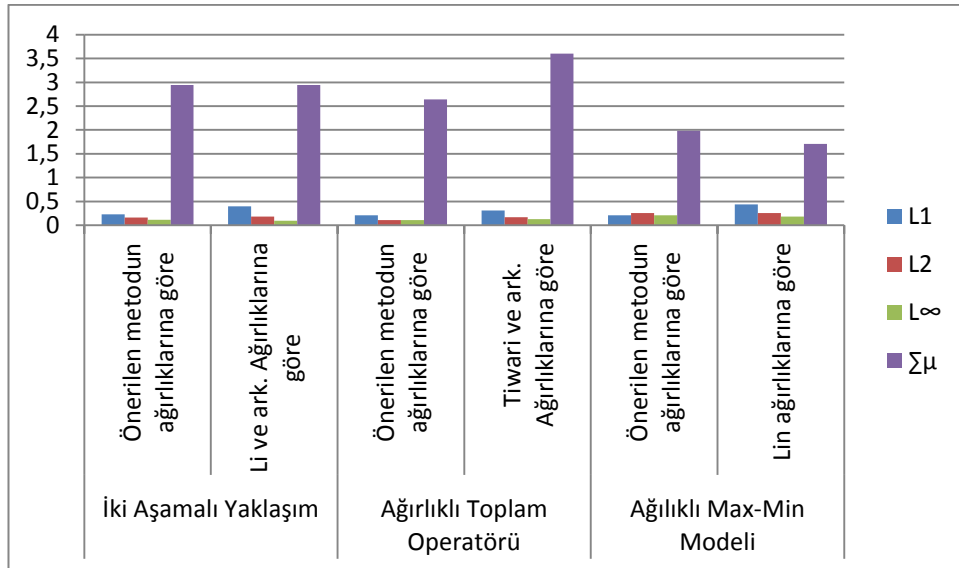
Problemler her iki ağırlık setine göre ilgili bulanık yaklaşım kullanılarak elde edilen çözümler için her bir fonksiyonun uzaklık fonksiyonu değerleri ve toplam üyelik fonksiyonu değeri hesaplanmıştır. Bu değerler Çizelge 3.7'de verilmiştir.

Çizelge 3.7. Problemlerin çözümü

	İki Aşamalı Yaklaşım'ındaki örnek		Ağırlıklı Toplam Operatörü'ndeki örnek		Ağırlıklı Max-Min Modelin'deki örnek	
	Önerilen metodun ağırlıklarına göre	Li ve ark. Ağırlıklarına göre	Önerilen metodun ağırlıklarına göre	Tiwari ve ark. Ağırlıklarına göre	Önerilen metodun ağırlıklarına göre	Lin ağırlıklarına göre
L_1	0.233	0.397	0.211	0.312	0.212	0.440
L_2	0.165	0.186	0.111	0.172	0.256	0.259
L_∞	0.117	0.1	0.111	0.128	0.212	0.186
$\sum_{k=1} \mu_k(x)$	2.94	2.94	2.64	3.6	1.985	1.708

Çizelge 3.8. Problemlerdeki amaçların ağırlıkları

	İki Aşamalı Yaklaşım'ındaki örnek		Ağırlıklı Toplam Operatörü'ndeki örnek		Ağırlıklı Max-Min Modelin'deki örnek	
	Önerilen metod	Li ve ark.	Önerilen metodun	Tiwari ve ark.	Önerilen metot	Lin
w_1	0	0,2	0	0,49	0,66	0,4
w_2	0,7	0,2	0	0,131	0,34	0,35
w_3	0,3	0,2	0,78	0,153	0	0,25
w_4	0	0,2	0	0,114	-	-
w_5	0	0,2	0,12	0,112	-	-



Şekil 3.2. Problemlerin toplam üyelik dereceleri ve uzaklık fonksiyonu değerleri

Çizelge 3.8’de önerilen metot ile bulunan ağırlıkların değeri ve çalışmalarda kullanılan ağırlıkların değerleri görülmektedir. Problemlerin çözümü ile elde edilen uzaklık fonksiyonu değerleri ve toplam üyelik fonksiyonu değerleri ayrıca Şekil 3.2’de görülmektedir.

İki aşamalı yaklaşımda, farklı ağırlıklara göre bulunan çözümler eşittir. Toplam üyelik derecesi aynı olurken, ağırlıklar farklı olduğu için uzaklık fonksiyonu değerleri farklı çıkmıştır. Ağırlıklı toplam operatöründe, Tiwari ve ark. (1987) kullandıkları ağırlıklara göre elde edilen çözümün toplam üyelik derecesi daha iyi iken, önerilen metot ile bulunan ağırlıklara göre elde edilen çözümün uzaklık fonksiyonu değerleri daha iyidir. Ağırlıklı max-min modelinde, hem toplam üyelik derecesine göre hem de amaçlar içindeki en uzak mesafeyi gösteren L_{∞} dışındaki uzaklık fonksiyonu değerlerine göre önerilen metot ile elde edilen çözüm daha iyidir.

4. ÖNCE DAĞIT SONRA TOPLA ARAÇ ROTALAMA PROBLEMİ

4. bölümde öncelikle araç rotalama probleminin genel tanımı yapıldıktan sonra problemin türlerinden kısaca bahsedilmiştir. Ardında bu tez çalışmasının konusu olan ÖDST-ARP tanıtılıp, literatürde yer alan çalışmalardan bahsedilmiştir.

4.1. Araç Rotalama Problemi ve Türleri

Taşıma, şirketler için ekonomik aktiviteleri içerisinde önemli bir faktördür. Müşteri beklentileri ve şirketin kısıtlı kaynakları altında araçların kullanımı bir problemdir ve bu problem araç rotalama problemi (ARP; vehicle routing problem-VRP) olarak bilinir (Tütüncü ve ark., 2009).

Klasik araç rotalama problemi, merkez depoda veya dağıtım merkezinde yer alan araçlar tarafından servis edilecek dağıtım noktalarından oluşur. Problemin amacı, araç kapasitesini aşmayacak şekilde talep noktalarından rotalar oluştururken tüm araçlar tarafından alınan yolu minimum yapmaktır (Goetschalckx ve ark., 1989).

Problemin temel kısıtları aşağıdaki gibi sıralanabilir.

- Araç kapasitesi aşılamaz.
- Bir müşteriye sadece bir araç uğrar.
- Tüm müşterilere uğranılmalıdır.
- Her bir aracın rotası depoda başlar ve biter.

ARP ilk olarak Dantzig ve Ramser (1959) tarafından ortaya atılmıştır ve daha sonra Clarke ve Wright (1964) tarafından ARP'nin çözümü için klasik tasarruf metodu önerilmiştir.

ARP'nin türleri aşağıdaki gibi sıralanabilir:

- a) Çok Depolu Araç Rotalama Problemi (Multi Depot Vehicle Routing Problem) :** Müşteri taleplerinin birden fazla depo ile karşılandığı ARP türüdür.

- b) Zaman Pencere Arac Rotalama Problemi (Vehicle Routing Problem with Time Windows) :** Müşterileri talebinin belli bir zaman aralığında karşılanması gereken ARP türüdür.
- c) Açık Araç Rotalama Problemi (Open Vehicle Routing Problem):** Araçların depoya dönmesi kısıtının olmadığı ARP türüdür.
- d) Dağıtım Toplamalı Araç Rotalama Problemi (Vehicle Routing Problem with Pickup and Delivery):** Araçların belli bir kısıt olmaksızın müşterilere hem ürünleri dağıttığı hem de müşterilerden ürün topladığı Arp türüdür.
- e) Eş Zamanlı Dağıtım Toplamalı Araç Rotalama Problemi (Vehicle Routing Problem with Simultaneously Pickup and Delivery):** Bir müşteriye hem ürünün bırakıldığı hem de aynı müşteriden ürün alındığı ARP türüdür.
- f) Önce Dağıt Sonra Topla Araç Rotalama Problemi (Vehicle Routing Problem with Backhaul):** Araçların öncelikle müşterilere ürünleri dağıttığı, ardından müşterilerden ürünleri topladığı ARP türüdür. Bu problem tipi ayrı başlık altında ayrıntılı olarak açıklanmıştır.

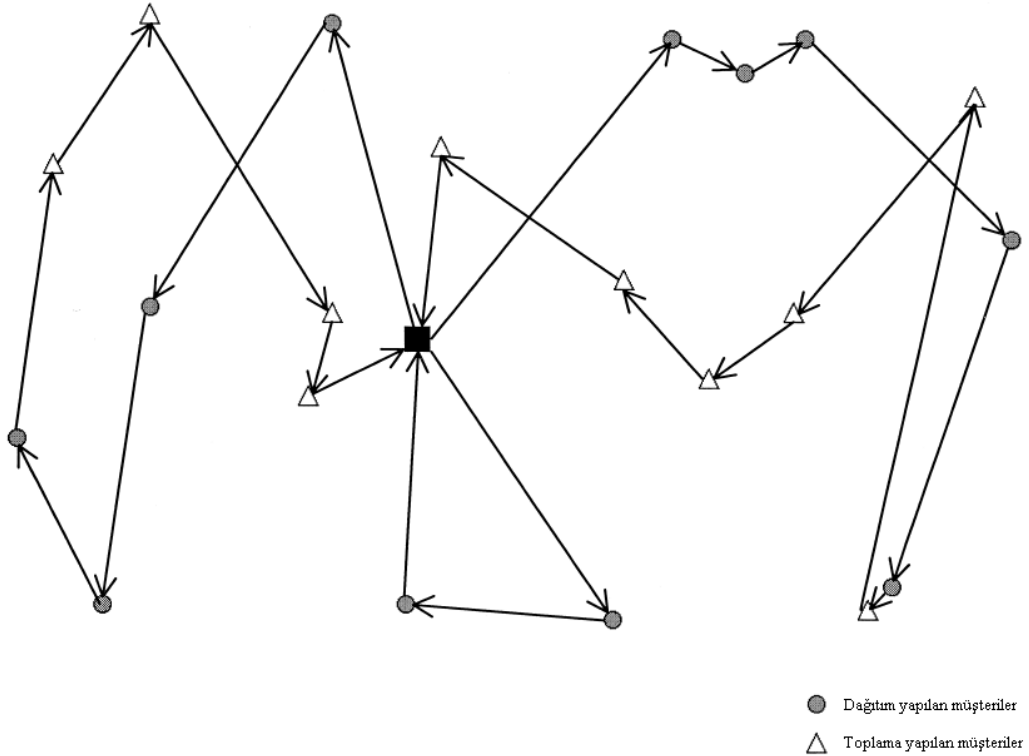
4.2. Önce Dağıt Sonra Topla Araç Rotalama Problemi

Önce dağıt sonra topla araç rotalama problemi (ÖDST-ARP; vehicle routing problem with backhauls-VRPB) dağıtım ve toplama noktalarını içeren ARP'nin genişletilmiş bir halidir. Dağıtım noktaları tek merkez noktadan ürünlerin dağıtıldığı bölgelerdir. Toplama noktaları ise ürünlerin toplanıp dağıtım merkezine getirildiği bölgelerdir (Goetschalckx ve ark., 1989). ÖDST-ARP için market endüstrisi yaygın bir örnektir. Süpermarketler ve mağazalar dağıtım yapılacak müşteriler, market tedarikçileri toplama yapılacak müşterilerdir. Yapılan çalışmalarda görülmüştür ki dağıtım rotalarında toplama yapılacak müşterilere de uğranılması taşıma maliyetlerinde kazanç sağlayacaktır (Toth ve ark., 1997).

ÖDST-ARP, aşağıda listelenen özellikleri taşıyan K adet aracın rotasının minimum maliyetlisini bulmaktan oluşur (Toth ve ark., 1997).

- Her araç bir tane rota oluşturur.
- Her rota dağıtım merkezini içerir.
- Her müşteriye bir kez uğranılır.
- Bir rotada uğranılan dağıtım ve toplama noktalarının taleplerinin toplamı, araç kapasitesini geçemez.
- Her bir rotada dağıtım noktalarından sonra toplama noktalarına gidilir.

Şekil 4.1’de ÖDST-ARP’nin şematik gösterimi görülmektedir. Yuvarlak ile gösterilenler dağıtım yapılacak müşteriler iken, üçgen ile gösterilenler toplama yapılacak müşterilerdir. Kare ile gösterilen ise depodur.



Şekil 4.1. Önce dağıtım sonra toplama araç rotalama problemi örneği (Mingozzi ve Giorgi, 1999)

4.2.1. Matematiksel modeller

Bu kısımda ÖDST-ARP için literatürde önerilmiş olan matematiksel yöntemlerden bahsedilmiştir. İlk üç modelde araçların kapasiteleri aynı olarak kabul edilirken son modelde araçların farklı kapasitelerde olduğu varsayılmıştır.

ÖDST-ARP'nin matematiksel modeli ilk defa Goetschalckx ve Jacobs-Blecha tarafından 1989 yılında önerilmiştir. Bu model Fisher ve Jaikumar'ın 1978 yılında ARP için geliştirdikleri modele dayanmaktadır. Modelin parametreleri, karar değişkenleri ve model aşağıda gösterildiği gibidir.

Parametreler

- K araç sayısı,
N dağıtım noktası sayısı, indeks; 1, 2, ..., N,
M toplama noktası sayısı, indeks; N+1, N+ 2, ..., N+M, (0 indeksi dağıtım merkezini göstermektedir.)
 a_i i. dağıtım noktasının talebi, $i = 1, 2, \dots, N$,
 b_i i. toplama noktasının talebi, $i = N+1, N+ 2, \dots, N+M$,
C dağıtım araçlarının sabit kapasitesi,
 c_{ij} i. noktadan j. noktaya yolculuğun maliyeti, $i, j = 1, 2, \dots, N, N+1, N+ 2, \dots, N+M$

Karar değişkenleri

$$u_{ik} = \begin{cases} 1 & \text{i. dağıtım noktasına k. araç tarafından ürün götürülüyorsa } i = 0, 1, \dots, N \\ 0 & \text{diğer durumlarda} \end{cases}$$

$$v_{jk} = \begin{cases} 1 & \text{j. toplama noktasından k. araç tarafından ürün alınıyorsa } j = N + 1, \dots, N + M \\ 0 & \text{diğer durumlarda} \end{cases}$$

$$x_{ijk} = \begin{cases} 1 & \text{i. noktasından j. noktaya k. araç tarafından geçiliyorsa } i, j = 0, 1, \dots, N, N + 1, \dots, N + M \\ 0 & \text{diğer durumlarda} \end{cases}$$

Model

$$\sum_{i=1}^N a_i u_{ik} \leq C \quad \forall k = 1, 2, \dots, K \quad (4.1)$$

$$\sum_{k=1}^K u_{ik} = 1 \quad \forall i = 1, 2, \dots, N \quad (4.2)$$

$$u_{0k} = 1 \quad \forall k = 1, 2, \dots, K \quad (4.3)$$

$$u_{ik} = 0 \text{ veya } 1 \quad \forall i = 1, 2, \dots, N \quad \forall k = 1, 2, \dots, K \quad (4.4)$$

$$\sum_{i=N+1}^{N+M} b_i v_{ik} \leq C \quad \forall k = 1, 2, \dots, K \quad (4.5)$$

$$\sum_{k=1}^K v_{ik} = 1 \quad \forall i = N+1, N+2, \dots, N+M \quad (4.6)$$

$$v_{0k} = 1 \quad \forall k = 1, 2, \dots, K \quad (4.7)$$

$$v_{ik} = 0 \text{ veya } 1 \quad \forall i = N+1, N+2, \dots, N+M \quad \forall k = 1, 2, \dots, K \quad (4.8)$$

$$\sum_{i=0}^{N+M} x_{ijk} = \begin{cases} u_{jk} & \text{eğer } j = 1, \dots, N \\ v_{jk} & \text{eğer } j = N+1, N+2, \dots, N+M \text{ ve } j = 0 \end{cases} \quad \forall k \quad (4.9)$$

$$\sum_{j=0}^{N+M} x_{ijk} = \begin{cases} u_{ik} & \text{eğer } i = 0, 1, \dots, N \\ v_{ik} & \text{eğer } i = N+1, N+2, \dots, N+M \end{cases} \quad \forall k \quad (4.10)$$

$$\sum_{i=0}^N \sum_{\substack{j=N+1 \\ \text{ve } j=0}}^{N+M} x_{ijk} = 1 \quad \forall k = 1, 2, \dots, K \quad (4.11)$$

$$x_{ijk} \in S \quad (4.12)$$

$$x_{ijk} = 0 \text{ veya } 1 \quad (4.13)$$

$$S = \{ x_{ijk} : \sum_{i \in Q} \sum_{j \in Q} \leq |Q| - 1 \} \quad (4.14)$$

Kısıtları altında

$$\text{enk} \sum_{k=1}^K \sum_{i=0}^{N+M} \sum_{j=0}^{N+M} c_{ij} x_{ijk} \quad (4.15)$$

Bu modelde (4.1) ve (4.5) numaralı kısıtlar hem dağıtım noktaları hem de toplama noktaları için araç kapasitesinin aşılamayacağını göstermektedir. (4.2) ve (4.6) numaralı kısıtlar her bir dağıtım ve toplama rotasına sadece bir aracın atanabileceğini ve K adetten fazla aracın dağıtım merkezinden ayrılıp geri dönemeyeceğini göstermektedir. (4.9) numaralı kısıt her bir noktaya sadece bir aracın gidebileceğini gösterirken, (4.10) numaralı kısıt her bir noktadan sadece bir

aracın ayrılabilceğini göstermektedir. (4.11) numaralı kısıt her bir rotada dağıtım noktaları ile toplama noktaları arasında sadece bir tane bağlantı olabileceğini göstermektedir. (4.14) numaralı kısıt ise gezgin satıcı probleminin klasik alt tur engelleme kısıtıdır.

ÖDST-ARP problemi için Toth ve Vigo 1997 yılında aşağıdaki matematiksel modeli önermişlerdir. Bu model asimetric ÖDST-ARP de için geçerlidir. Öncelikle model için gerekli kümeler tanımlanacaktır.

Kümeler

$$\begin{aligned}
G' &= (V'_0, A') \\
V_0 &= \{0\} \cup \{1, \dots, n\} \cup \{n+1, \dots, n+m\} \\
L &= \{1, \dots, n\} \\
B &= \{n+1, \dots, n+m\} \\
V &= V_0 \setminus \{0\} \\
L_0 &= L \cup \{0\} \\
B_0 &= B \cup \{0\} \\
G &= (V_0, A) \\
A &= A_1 \cup A_2 \cup A_3 \\
A_1 &= \{(i, j) \in A' : i \in L_0, j \in L\} \\
A_2 &= \{(i, j) \in A' : i \in B, j \in B_0\} \\
A_3 &= \{(i, j) \in A' : i \in L, j \in B_0\}
\end{aligned}$$

Bağlantı kümesi A, ayrık üç kümeye ayrılabilir. İlk alt küme A_1 depodan ve dağıtım noktalarından dağıtım noktalarına olan bağlantıları içerir. İkinci alt küme A_2 toplama noktalarından toplama noktalarına ve depoya olan bağlantıları içerir. Üçüncü alt küme ve ara bağ bağlantısı diye adlandırılan A_3 dağıtım noktalarından toplama noktalarına veya depoya olan bağlantıları içerir. A kümesi uygun çözüme ait olmayan bağlantıları içermez. Örneğin toplama noktalarından dağıtım noktalarına olan bağlantı, depodan toplama noktalarına bağlantı gibi.

$\mathcal{L}$ (aynı şekilde $\mathcal{B}$), L (aynı şekilde B)’deki köşelerin tüm alt kümelerini temsil ettiğinde $\mathcal{F} = \mathcal{L} \cup \mathcal{B}$ olur. Her bir $S \in \mathcal{F}$ için S ’deki tüm noktalara hizmet

vermek için gerekli olan amaç sayısını temsil etsin. Her $i \in V_0$ için $\Gamma_i^+ = \{j: (i,j) \in A\}$ ve $\Gamma_i^- = \{j: (j,i) \in A\}$ olarak tanımlandığında simetrik ve simetrik olmayan ÖDST-ARP için tam sayılı model aşağıdaki gibi tanımlanmıştır.

Parametreler

K araç sayısı,

c_{ij} i. noktadan j. noktaya yolculuğun maliyeti

Karar değişkenleri

$$x_{ij} = \begin{cases} 1 & \text{arc}(i,j) \in A \text{ en iyi çözümde kullanıldıysa} \\ 0 & \text{diğer durumlarda} \end{cases}$$

Model

$$\sum_{i \in \Gamma_j^-} x_{ij} = 1 \quad \forall j \in V \quad (4.16)$$

$$\sum_{j \in \Gamma_i^+} x_{ij} = 1 \quad \forall i \in V \quad (4.17)$$

$$\sum_{i \in \Gamma_0^-} x_{i0} = K \quad (4.18)$$

$$\sum_{j \in \Gamma_{s0}^+} x_{0j} = K \quad (4.19)$$

$$\sum_{j \in S} \sum_{i \in \Gamma_j^- \setminus S} x_{ij} \geq \sigma(S) \quad \forall S \in \mathfrak{F} \quad (4.20)$$

$$\sum_{i \in S} \sum_{j \in \Gamma_i^+ \setminus S} x_{ij} \geq \sigma(S) \quad \forall S \in \mathfrak{F} \quad (4.21)$$

$$x_{ij} \in \{0,1\} \quad \forall i,j \in V_0 \quad (4.22)$$

Kısıtları altında

$$v(P) = \text{enk} \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (4.23)$$

Bu modelde (4.16), (4.18) ve (4.17), (4.19) numaralı kısıtlar müşteri ve depo köşeleri için iç ve dış derecelerini göstermektedir. (4.20) ve (4.21) numaralı kısıtlar bağlantıyı ve kapasite sınırını göstermektedir.

Mingozzi ve Giorgi (1999) ÖDST-ARP için 0-1 tam sayılı yeni bir matematiksel model önermişlerdir. Modelde M gerekli araç sayısı, M_L dağıtım yapılacak müşteriler için gerekli minimum araç sayısı, M_B toplama yapılacak müşteriler için gerekli minimum araç sayısı olmak üzere $M \geq \text{Enb } [M_L, M_B]$ olur. Modelin kümeleri, parametreleri, karar değişkenleri ve model aşağıda gösterildiği gibidir.

Kümeler

$$G_L = (L_0, A_L)$$

$$G_B = (B_0, A_B)$$

$$L_0 = L \cup \{0\}$$

$$A_L = \{ (i, j): (i, j) \in A; i, j \in L_0 \}$$

$$B_0 = B \cup \{0\}$$

$$A_B = \{ (i, j): (i, j) \in A; i, j \in B_0 \}$$

$$A_0 = \{ (i, j): (i, j) \in A; i \in L, j \in B_0 \}$$

Parametreler

$$c_l \quad l \in L \cup B \text{ ait yolun maliyeti}$$

Karar değişkenleri

$$x_l, y_{l'} \text{ ve } \xi_{ij}$$

$$= \begin{cases} 1 & l \in L, l' \in B \text{ ve } (i, j) \in A_0 \text{ en iyi çözümde kullanılıyorsa} \\ 0 & \text{diğer durumlarda} \end{cases}$$

Model

$$\sum_{l \in L_i} x_l = 1 \quad \forall i \in L \quad (4.24)$$

$$\sum_{l \in B_j} y_l = 1 \quad \forall j \in B \quad (4.25)$$

$$\sum_{l \in L_i^E} x_l - \sum_{j \in B_0} \xi_{ij} = 0 \quad \forall i \in L \quad (4.26)$$

$$\sum_{l \in B_j^S} y_l - \sum_{i \in L} \xi_{ij} = 0 \quad \forall j \in B \quad (4.27)$$

$$\sum_{(i,j) \in A_0} \xi_{ij} = M \quad (4.28)$$

$$x_l \in \{0,1\}, \quad l \in L, \quad y_l \in \{0,1\}, \quad l \in B, \quad \xi_{ij} \in \{0,1\}, \quad (i,j) \in A_0 \quad (4.29)$$

Kısıtları altında

$$z(IP) = \text{enk} \sum_{l \in L} c_l x_l + \sum_{l \in B} c_l y_l + \sum_{(i,j) \in A_0} d_{ij} \xi_{ij} \quad (4.30)$$

Bu modelde (4.24) ve (4.25) numaralı kısıtlar her bir noktaya tek bir rotada uğranabileceğini göstermektedir. (4.26) numaralı kısıt $i \in L$ ile biten G_L 'de uygun yolu içeren çözümün $i \in L$ ile başlayan A_0 ayrıtını içermeye zorlamaktadır. (4.27) numaralı kısıt $j \in B$ ile başlayan G_B 'de uygun yolu içeren çözüm için $i \in L$ ve $j \in B$ 'den oluşan (i,j) ayrıtının çözümünü içermeyi gerektirir. (4.28) numaralı kısıt M adet rota oluşmasına zorlar.

Tavakkoli-Moghaddam, Saremi ve Ziaee (2006) araçların farklı kapasiteye ve farklı taşıma maliyetlerine sahip olduğu heterojen ÖDST-ARP için matematiksel model önermişlerdir. Önerilen matematiksel modelin parametreleri, karar değişkenleri ve model aşağıda açıklanmıştır.

Parametreler

C_{ijk} k. Aracın i. düğümünden j. düğüme geçme maliyeti

n dağıtım yapılacak düğüm sayısı

- m toplama yapılacak düğüm sayısı
M araç sayısı
 Q_k k. Aracın kapasitesi
 f_i i. noktada toplama yapılacak miktar
 d_i i. noktaya dağıtım yapılacak miktar

Karar Değişkenleri

$$x_{ijk} = \begin{cases} 1 & \text{eğer k. araç i. düğümünden j. düğüme geçerse} \\ 0 & \text{diğer durumlarda} \end{cases}$$

Model

$$\sum_{j=0}^{n+m} \sum_{k=1}^M x_{ijk} = 1 \quad \forall i = 1, \dots, n+m \quad i \neq j \quad (4.31)$$

$$\sum_{i=0}^{n+m} \sum_{k=1}^M x_{ijk} = 1 \quad \forall j = 1, \dots, n+m \quad j \neq i \quad (4.32)$$

$$\sum_{j=0}^{n+m} \sum_{k=1}^M x_{0jk} = M \quad (4.33)$$

$$\sum_{i=0}^{n+m} \sum_{k=1}^M x_{i0k} = M \quad (4.34)$$

$$\sum_{i=1}^n \sum_{j=1}^{n+m} d_i x_{ijk} \leq Q_k \quad \forall k = 1, \dots, M; j = 0, 1, \dots, n+m \quad (4.35)$$

$$\sum_{i=n+1}^{n+m} \sum_{j=1}^{n+m} f_i x_{ijk} \leq Q_k \quad \forall k = 1, \dots, M; j = 0, 1, \dots, n+m \quad (4.36)$$

$$\sum_{i=0}^{n+m} x_{ijk} - \sum_{l=0}^{n+m} x_{jlk} = 0 \quad \forall j = 1, \dots, n+m; k = 1, \dots, M \quad (4.37)$$

$$\sum_{j=0}^{n+m} x_{ijk} - \sum_{l=0}^{n+m} x_{lik} = 0 \quad \forall i = 0, 1, \dots, n+m; k = 1, \dots, M \quad (4.38)$$

$$\sum x_{ijk} \leq |S| - 1, \quad S \subseteq \{2, 3, \dots, n+m\} \quad (4.39)$$

$$\sum_{i=n+1}^{n+m} \sum_{j=1}^n \sum_{k=1}^M x_{ijk} = 0 \quad (4.40)$$

$$x_{ijk} = \{0,1\}, \quad i, j = 0,1, \dots, n+m; k = 1, \dots, M \quad (4.41)$$

Kısıtları altında

$$\text{enk} \sum_{i=n+1}^{n+m} \sum_{j=1}^n \sum_{k=1}^M C_{ijk} x_{ijk} \quad (4.42)$$

Bu modelde amaç taşıma maliyetini minimum yapmaktır. (4.31) ve (4.32) numaralı kısıtlar her bir düğümün bir rotada olmasını sağlamaktadır. (4.33) ve (4.34) numaralı kısıtlar depodan çıkacak ve depoya dönecek araç sayılarını göstermektedir. (4.35) ve (4.36) numaralı kısıtlar araca yüklenebilecek toplama ve dağıtım düğümlerinin toplam talepleri araç kapasitesini geçmemesini sağlamaktadır. (4.37) ve (4.38) numaralı kısıtlar rotanın devamlılığını sağlamaktadır. (4.39) numaralı kısıt alt turların oluşmasını engellemektedir. (4.40) numaralı kısıt toplama yapılan düğümlerden dağıtım yapılan düğümlere dönülmesini engellemektedir.

4.2.2. Kesin çözüm yöntemleri

Yano ve ark.(1987) ÖDST-ARP için küme kapsamaya dayanan kesin çözüm veren bir algoritma geliştirmişlerdir. Problemde her bir aracın gideceği müşteri sayısı her bir tür için de 4'ü aşmamalıdır.

Toth ve Vigo (1997), hem simetrik hem de simetrik olmayan uzaklık matsine sahip ÖDST-ARP için kesin çözüm yöntemi önermişlerdir. Önerdikleri matematiksel model “Matematiksel Modeller” başlığı altında verilmiştir. Kesme düzleminde kuvvetlendirilen lagrange alt sınır değerini önerilen matematiksel model ile bulmuşlardır. Ayrıca en iyi çözümü bulan dal sınır algoritması geliştirmişlerdir.

Mingozzi ve Giorgi (1999), yeni bir 0-1 tam sayılı matematiksel model önermişlerdir. Bu model “Matematiksel Modeller” başlığı altında verilmiştir. ÖDST-ARP’de geçerli alt sınırı elde etmek için tam sayılı modelin doğrusal programlama gevşetiminin ikil problemini çözen D olarak isimlendirilen sezgisel

yöntemi kullanmışlardır. İki farklı sezgisel yöntemi birleştirerek, ikil kısıtların tüm kümesine gerek kalmadan D için uygun çözüm bulunur. Elde edilen ikil çözüm ve ÖDST-ARP’de geçerli alt sınır en iyi çözümü oluşturabilen rotaların sayısını (tam sayılı modelin değişkenleri gibi) azaltmak için kullanılır. İndirgenmiş tam sayılı problemin boyutları bile dal-sınır algoritması ile çözmek için hala çok büyük olabilir. Bu sebeple EHP olarak isimlendirilen tam sayılı problemin değişken sayısını azaltan yöntemi önermişlerdir. Yöntem 100 noktaya kadar olan problemler üzerinde denenmiştir.

4.2.3. Sezgisel yaklaşımlar

Deif ve Bodin (1984) ÖDST-ARP için Clarke ve Wright’ın ARP sezgiseline dayanan yeni bir sezgisel önermişlerdir. Dağıtım yapılan ile toplama yapılan müşterilerin arasındaki tasarruf değerini düzenlemişlerdir.

Goetschalckx ve Jacobs-Blecha (1989) ÖDST-ARP için iki aşamalı çözüm yöntemini önermişlerdir. İlk aşamada boşluk doldurma eğrilerine dayanan yüksek kaliteli ilk uygun çözüm üretilir. İkinci aşamada problemin matematiksel modelinde tanımlanan alt problemlerin optimizasyonu ile bu çözüm iyileştirilir.

Thangiah ve Sun (1996) ÖDST-ARP’nin bir türü olan ve müşteriler için en erken ve en geç ulaşma kısıtı ile değişken talebi olan zaman pencereli önce dağıt sonra topla araç rotalama problemi (ZP-ÖDST-ARP) için rota yapım sezgiselini geliştirmişlerdir.

Toth ve Vigo (1999), hem simetrik hem de simetrik olmayan uzaklık matrisine sahip ÖDST-ARP için yeni bir kümeleme yaklaşımını kullandıkları önce kümele sonra rotala sezgisel yaklaşımını geliştirmişlerdir. Elde edilen rota, rota içi ve rotalar arası değişim ile geliştirilmişlerdir. Bu sezgiselin ismini TV olarak isimlendirmişlerdir.

Salhi ve Nagy (1999), klasik ekleme tabanlı sezgiselini ÖDST-ARP için genişletmişlerdir. Sezgisel aynı anda birden fazla toplama noktası ekleme düşüncesine dayanmaktadır. Bu metot bir ve birden fazla depolu ÖDST-ARP için test edilmiştir.

Wade ve Salhi (2002), ÖDST-ARP için yeni bir versiyon önermişlerdir. Dağıtım yapılan müşterilere belli bir oranda gidildikten sonra toplama yapılan müşterilere uğranılabilmektedir. Bu tip problemler için ekleme tipi sezgisel önerilmiştir. Ayrıca bu yeni tip problem için 0-1 tam sayılı matematiksel model önerilmiştir.

Ropke ve Pisinger (2006) çalışmalarında literatürde çeşitli ÖDST-ARP için önerilen yöntemler gözden geçirilmiş ve problemin çeşitli türlerini çözebilecek birleştirilmiş model geliştirmişlerdir.

Tütüncü, Carreto ve Baker (2009) çalışmalarında klasik ÖDST-ARP ve onun genişletilmiş versiyonu için yeni görsel interaktif yaklaşım önermişlerdir. Karma ÖDST-ARP olarak isimlendirilen genişletilmiş versiyonda toplama noktalarına uğranıldıktan sonra da dağıtım noktalarına uğranılmasına izin verir. Görsel, interaktif yaklaşım aç gözlü rassal adaptif hafıza programı aramasına (Greedy Randomized Adaptive Memory Programming Search) dayanmaktadır. Algoritmaya ilişkin karar destek sistemi geliştirmişlerdir.

Wang ve Wang (2009) çalışmalarında yeni iki aşamalı sezgisel metot geliştirmişlerdir. Şehirlerde araçların hızlarının farklı olduğu da düşünülüp zamana bağlı olarak araçların hızları da göz önüne alınmıştır. İlk aşamada orijinal çözüm genişletilmiş geleneksel sezgisel yöntemler ile oluşturulmakta, ikinci aşamada reaktif yasaklı arama algoritması orijinal çözümü en iyilemek için kullanılmıştır.

4.2.4. Meta-Sezgisel yaklaşımlar

Wade ve Salhi (2001) çalışmalarında ÖDST-ARP için karınca kolonilerini kullanmışlardır. Başlangıçta m adet karınca n adet müşteriye yerleştirilir ve araç kapasitesine ulaşıncaya kadar karınca bir sonraki şehir hareket eder. Bir sonraki tekrarlar da feromon miktarına göre hesaplanan olasılık ile bir sonraki şehir seçilir. Ayrıca rotanın iyileştirilmesi için 2-opt ve 3-opt yerel arama algoritmalarını kullanmışlardır.

Zhong ve Cole (2005), zaman pencereli önce dağıt sonra topla araç rotalama problemi için meta sezgisel önermişlerdir. Ayrıca müşteri kısıtının olup

olmadığı her iki durumda düşünülmüştür. Temel yaklaşım uygun olmayan başlangıç çözümü elde etmektir. Daha sonra rehberli yerel arama ile uygunluk ve kalite iletmiştir. Yeni bir teknik olan bölüm planlama, uygunluğu iletirmek için kullanılmıştır.

Tavakkoli-Moghaddam, Saremi ve Ziaee (2006) çalışmalarında ÖDST-ARP'ni çözmek için farklı yerel arama algoritmalarını kullanan memetik algoritma önermişlerdir. Araçlar heterojen olarak ele alınmıştır. Yani araçların kapasiteleri birbirinden farklıdır. Ulaşım ağı simetrik veya asimetrik olabilir. Önerilen memetik algoritmada ilk uygun çözüm Solomon ekleme sezgiselinin uyarlanmış versiyonu ile oluşturulmaktadır. Daha sonra yasaklı arama, yerel arama ve ilerleme algoritmaları çözümü iletirmek için kullanılmıştır.

Brandão (2006) çalışmasında ÖDST-ARP çözmek için yasaklı arama algoritmasını kullanmışlardır. Başlangıç çözümü “açık başlangıç çözümü” ve “K-ağaç başlangıç” çözümü ile elde edilmektedir. Açık başlangıç çözümünde her iki müşteri grubu ayrı ayrı rotaları oluşturulmakta daha sonra birleştirilmektedir. Dağıtım yapılacak müşteriler için rota oluşturulurken amaç toplam mesafeyi minimum yapmak, toplama yapılacak müşteriler için rota oluşturulurken birinci amaç rota sayısını minimum yapmakken ikinci amaç toplam mesafeyi minimum yapmaktır.

Ghaziri ve Osman (2006) çalışmalarında denetlenmeyen rekabetçi sinir ağları yapısını kullanan kendi kendine organize özellikli harita algoritması geliştirmişlerdir. Bu yöntem Ghazari ve Osman (2003) ÖDST-GSP için önerdikleri yöntemin genişletilmiş halidir.

Gajpal ve Abad (2009) çalışmalarında ÖDST-ARP problemi için araçlar tarafından toplam alınan yolun en küçüklenmesini amaçlayan çoklu karınca kolonisi algoritmasını önermişlerdir. Algoritmada iki grup karınca bulunmaktadır. Birinci grup karınca, müşterileri araçlara atarken ikinci grup karınca, her bir araç için rota oluşturur. Oluşturulan rotalar ekleme ve değiştirme operatörlerini içeren yerel arama ile iyileştirilmeye çalışılmıştır.

Tütüncü (2010) çalışmasında karma kapasiteli ARP ve karma kapasiteli sabit sayılı araçlı ÖDST-ARP problemi için açgözlü rassal adaptif hafıza programı aramasına dayanan görsel interaktif yaklaşım geliştirmiştir. Yöntemin kullanımı için bir karar destek sistemi geliştirmişlerdir.

5. ÖNCE DAĞIT SONRA TOPLA ARAÇ ROTALAMA PROBLEMİ İÇİN ÖNERİLEN YÖNTEM

Bir önceki bölümde görüleceği gibi ÖDST-ARP'nin matematiksel modelle çözüm süresi mevcut matematiksel modeller ile şehir sayısına bağlı olarak hızla arttığından dolayı bu modeller ile gerçek hayat problemlerine hızlı bir şekilde çözüm bulmak zorlaşmaktadır. Bu sebeple literatürde geliştirilen matematiksel model sayısı az iken önerilen sezgisel ve meta sezgisel çalışmaların sayısı fazladır. Sezgisel ve meta sezgisel yöntemler ile büyük çaplı problemlerin çözümü için algoritmalarının bir bilgisayar programlama dili ile kodlanması gerekmektedir. Başka bir deyişle bu yöntemler için programlamaya yoğun vakit ayrılması gerekmektedir.

Bu çalışmada ÖDST-ARP'nin çözümünü matematiksel modelle çözebilmek ve programlamaya ayrılan zamanı azaltabilmek amacı ile matematiksel modellere dayana iki aşamalı bir sezgisel önerilmiştir. Bu sezgisel “Önce kümele sonra rotala” olarak isimlendirilen ve ARP'nin çözümü için kullanılan algoritmalara benzemektedir. Bu algoritmalarda müşteriler öncelikle araçlara atanır, daha sonra her bir aracın rotası oluşturulur. İlk olarak Gillet ve Miller (1974) süpürme algoritmasını önermişlerdir. Bu algoritmada müşteriler araçlara atanırken en küçük açığa sahip müşteriden başlanır ve kapasite uygun ise müşteri araca atanır. Eğer j . müşteri kapasite yetmediği için araca atanamıyor ise $j+1$. müşterinin araca atanıp atanamayacağının kontrolü yapılmaz. Tüm müşteriler araçlara atanıncaya kadar atama işlemi devam eder. Fisher ve Jaikumar (1981) genelleştirilmiş atama problemine dayanarak müşterileri araçlara atayan bir algoritma önermişlerdir. Başlangıçta araç sayısı kadar merkez şehrin seçilmesi gerekmektedir. Bir müşterinin bir araca atanmasının maliyeti o müşteri ile merkez şehri arasındaki uzaklıktır. Bramel ve Simichi-Levi (1995) ise lokasyon tabanlı bir algoritma önermişlerdir. Başlangıçta aday merkez müşteriler seçilir ve kapasiteli lokasyon probleminin çözümü ile hem merkez şehirler belirlenir hem de diğer şehirler merkez şehirlere atanır.

Önerilen sezgiselin “Bölge oluşturma aşaması” olarak isimlendirilen birinci aşamasında çok amaçlı matematiksel model ile dağıtım ve toplama noktaları araçlara atanmıştır. Bölgeler içindeki toplam mesafenin minimize

edilmesi ve yine bölgeler içindeki tasarruf miktarının maksimize edilmesini amaçlayan çok amaçlı modelin çözümünde bulanık çözüm yaklaşımları kullanılmıştır. Kullanılan bulanık çözüm yaklaşımlarından min operatörü dışındaki yaklaşımlar için amaçlara ağırlıkların atanması gerekmektedir. Bu ağırlıkların belirlenmesi için ikinci bölümde bahsedilmiş olan ve karar vericiden amaçlar hakkında detaylı bilgi alınmasına gerek olmadığından dolayı iki oyunculu sıfır toplamli oyun kuramına dayanan ve ödemeler matrisinde amaçların üyelik fonksiyonu değeri kullanılan metot kullanılmıştır. Bölge oluşturma aşaması sonucunda her bir araç için problem ÖST-GSP'ye dönüşmüş olur. Bu aşamanın diğer önce kümele sonra rotalama algoritmalarındaki kümeleme aşamalarından farkı, hem merkez şehir ile diğer şehirler arasındaki mesafeyi hem de depoya olan uzaklığı göz önüne alan çok amaçlı matematiksel modelin kullanılması ve matematikselin model merkez şehirleri seçmesi böylelikle başlangıçta merkez şehir kümesinin belirlenmesine gerek kalmamasıdır.

Önerilen sezgiselin “Rotalama aşaması” olarak isimlendirilen ikinci aşamasında ise her bir aracın rotası Goetschalckx ve Jacobs-Blecha'nın ÖDST-ARP için önermiş olduğu matematiksel modelden kapasite ve atama kısıtları çıkarılarak ÖDST-GSP için elde edilmiş olan matematiksel model ile oluşturulur. Son olarak elde edilen rotalardan daha iyisini elde edilip edilemeyeceğine ekleme ve değiştirme operatörleri kullanılarak “yerel arama” ile bakılmıştır.

Önerilen matematiksel modeller GAMS yazılımında kodlanmış ve CPLEX 9.0 çözücüsü kullanılarak sonuçlar elde edilmiştir. GAMS yazılımındaki kodlar bölge oluşturma aşaması için EK 1’de, rotalama aşaması için EK 2’de verilmiştir. GAMS yazılımı için gerekli olan tüm veriler MS EXCEL’den alınmış ve GAMS yazılımından elde edilen çözümler yine MS EXCEL’e yazdırılmıştır. Önerilen yöntemin adımlarında gerekli olan işlemler MS EXCEL’de VBA kullanılarak kodlanmış ve esnek bir yapı oluşturularak değişen veriler için kolayca hesaplanabilmesi sağlanmıştır. Benzer şekilde GAMS yazılımı sonucunda MS EXCEL’e yazdırılan sayısal sonuçların kolayca anlaşılabilir şehir, dağıtım ve toplama noktası isimlerine dönüştürülmesi için VBA kullanılmıştır. Ayrıca yerel arama algoritması da VBA kullanılarak kodlanmıştır.

Önerilen yöntemin adımları kısaca aşağıda verilmiştir.

Bölge Oluşturma:

Adım 1: Bölgeleri oluşturmak için gerekli olan uzaklık matrisini ve tasarruf miktarı matrisini oluştur.

Adım 2: Amaç fonksiyonlarını oluştur.

Adım 3: Önerilen çok amaçlı matematiksel modeli kur.

Adım 4 : Amaçların ideal ve anti ideal değerlerini hesapla.

Adım 5: Amaçların üyelik fonksiyonlarını oluştur.

Adım 6: Önerilen metod ile amaçların ağırlıklarını belirle.

Adım 7: Bulanık çözüm yaklaşımlarını kullanarak modeli çöz.

Rotalama:

Adım 8: Her bir araca atanan müşterilere göre her bir araç için uzaklık matrisini oluştur.

Adım 9: Önerilen ÖDST-GSP matematiksel modelini kullanarak her bir araç için rotayı oluştur.

Yerel Arama:

Adım 10: Ekleme operatörünü kullanarak rotaları iyileştir.

Adım 11: Değiştirme operatörünü kullanarak rotaları iyileştir.

5.1. Bölge Oluşturma Aşaması

ÖDST-ARP'nin çözümü için önerilen modelin ilk aşamasında önerilen çok amaçlı matematiksel model ile toplama ve dağıtım noktalarının talepleri, araçların kapasiteleri ve gerekli araç sayısı göz önüne alınarak bu noktalar araçlara atanır. Araçlara atama işlemi için model merkez şehirleri seçer ve diğer şehirleri bu şehirlere atar. Bu atama işleminde birinci amaç için şehirlerin merkez şehirlere olan uzaklığını göz önüne alırken, ikinci amaçta şehirlerin merkez şehirlere olan tasarruf değerini (saving value) göz önüne alır. Buradaki tasarruf değeri, tasarruf algoritmasında kullanılan değerdir. Denklem 5.1'de bu değer nasıl hesaplanacağı görülmektedir. Birinci amaçta deponun konumu göz önüne

alınmazken, ikinci amaç ile deponun konumu önem kazanmaktadır. Böylelikle depodan çok uzakta bir müşteri kümesinin oluşması engellenmiş olur.

$$s_{ij} = d_{0i} + d_{0j} - d_{ij} \quad (5.1)$$

İlk aşama çok amaçlı bir yapıya sahip olduğundan dolayı bu çalışmanın ilk bölümünde bahsedilen bulanık çözüm yöntemleri modelin çözüm aşamasında kullanılmış ve amaçlara ağırlık atanmasını gerektiren bulanık çözüm yöntemleri için de yine ilk bölümde bahsedilen çok amaçlı matematiksel modellerde amaçların ağırlıklarının belirlenmesine ilişkin önerilen metoda göre amaçların ağırlıkları belirlenmiştir. Böylelikle ilk aşamanın çözümü ile birlikte her bir araca atanan dağıtım ve toplama noktası sayısı her bir araç için rotalama problemini matematiksel model ile çözülebilecek sayıya indirgemiş olur.

Birinci aşamanın matematiksel modeli :

Kümeler

i, j dağıtım ve toplama noktaları

Parametreler

- N dağıtım yapılacak müşteri sayısı
- M toplama yapılacak müşteri sayısı
- d_{ij} i. nokta ile j. nokta arasındaki uzaklık
- a_i toplama yapılacak i. müşterinin talebi
- b_i dağıtım yapılacak i. müşterinin talebi
- s_{ij} i. nokta ile j. nokta arasındaki tasarruf miktarı
- C araçların kapasitesi
- K toplam araç sayısı

Karar değişkenleri

$$x_{ij} = \begin{cases} 1 & \text{i. şehir j. merkeze atanmışsa} \\ 0 & \text{diğer durumlarda} \end{cases}$$

Model

$$\sum_{j=1}^{N+M} x_{ij} = 1 \quad \forall i = 1, \dots, N + M \quad (5.2)$$

$$\sum_{i=1}^N a_i x_{ij} \leq Cx_{jj} \quad \forall j = 1, \dots, N + M \quad (5.3)$$

$$\sum_{i=N+1}^{N+M} b_i x_{ij} \leq Cx_{jj} \quad \forall j = 1, \dots, N + M \quad (5.4)$$

$$\sum_{i=1}^{N+M} \sum_{j=1}^{N+M} x_{ij} = K \quad (5.5)$$

$$x_{ij} \in \{0,1\} \quad (5.6)$$

Kısıtları altında

$$\text{enk } z_1 = \sum_i \sum_j d_{ij} x_{ij} \quad (5.7)$$

$$\text{enb } z_2 = \sum_i \sum_j s_{ij} x_{ij} \quad (5.8)$$

(5.2) numaralı kısıt bir şehrin sadece bir araca atanması gerektiğini göstermektedir. (5.3) ve (5.4) numaralı kısıtlar bir araca atanan toplama ve dağıtım noktalarının taleplerinin aracın kapasitesini geçmemesini sağlamaktadır. (5.5) numaralı kısıt toplamda K adet merkez şehrin seçilmesini yani K adet araca şehirlerin atanmasını sağlamaktadır. (5.6) numaralı kısıt ise karar değişkenininin 0-1 tamsayı olduğunu göstermektedir. (5.7) numaralı birinci amaç fonksiyonu araçlara atama yapılması için seçilen merkez şehir ile bu şehre atanacak diğer şehirler için toplam mesafenin minimum olmasını hedefler. (5.8) numaralı ikinci amaç fonksiyonu seçilen merkez şehir ile bu şehre atanacak diğer şehirler için toplam tasarruf miktarının maksimum olmasını hedefler.

Bölge oluşturma aşaması kısaca aşağıdaki gibidir.

- Şehirlerin arasındaki mesafeler ve tasarruf miktarları belirlenir.
- Toplama ve dağıtım işlemleri için gerekli araç sayısı belirlenir.
- Amaçlar oluşturulur.

- Çok amaçlı model kurulur.
- Amaçların en iyi ve en kötü değerleri her bir amaç tek tek optimize edilerek bulunur.
- Amaçların en iyi ve en kötü değerlerine göre doğrusal üyelik fonksiyonları oluşturulur.
- Amaçların ağırlıkları önerilen metoda göre belirlenir.
- Bulanık çözüm yaklaşımlarından biri kullanılarak çok amaçlı matematiksel modelin çözümü elde edilir.

5.2. Rotalama Aşaması

Bölge oluşturma aşaması ile her bir dağıtım ve toplama noktası araçlara atamış olur. Bu atama ile her bir aracın rotalama problemi ÖDST-GSP dönüşmüş olur. Araçlara atanan şehir sayısı bu problemi matematiksel model ile çözüm bulmayı olanaklı hale getirir.

Rotalama aşamasında kullanılacak matematiksel model Goetschalckx ve Jacobs-Blecha'nın ÖDST-ARP için önermiş olduğu matematiksel modelden kapasite ve atama kısıtları çıkarılarak elde edilmiştir.

İkinci aşamanın matematiksel modeli :

Kümeler

i, j ilgili araç için toplama ve dağıtım yapılacak noktalar

Parametreler

n ilgili araçta dağıtım yapılacak müşteri sayısı

m ilgili araçta toplama yapılacak müşteri sayısı

c_{ij} i . nokta ile j . nokta arasındaki uzaklık

Karar deęiřkeni

$$y_{ij} = \begin{cases} 1 & \text{i. noktadan j. noktaya geiliyorsa} \\ 0 & \text{dięer durumlarda} \end{cases}$$

Model

$$\sum_{i=0}^n y_{ij} = 1 \quad \forall j = 1, \dots, n \quad (5.9)$$

$$\sum_{j=0}^{n+m} y_{ij} = 1 \quad \forall i = 0, \dots, n \quad (5.10)$$

$$\sum_{i=0}^{n+m} y_{ij} = 1 \quad \forall j = n+1, \dots, n+m \quad (5.11)$$

$$\sum_{j=n+1}^{n+m} y_{ij} = 1 \quad \forall i = n+1, \dots, n+m \quad (5.12)$$

$$\sum_{i=0}^n \sum_{j=n+1}^{n+m} y_{ij} \leq 1 \quad (5.13)$$

$$\alpha_i - \alpha_j + (n+m+1) y_{ij} \leq (n+m) \quad \forall i, j = 1, \dots, n+m \text{ and } i \neq j \quad (5.14)$$

$$y_{ij} \in \{0,1\} \quad (5.15)$$

Kısıtları altında

$$\text{enk } z = \sum_i \sum_j c_{ij} y_{ij} \quad (5.16)$$

(5.9) numaralı kısıt dağıtım yapılacak olan müşterilere depodan veya başka bir dağıtım yapılacak müşteriden sadece bir kere gelinmesini sağlamaktadır. (5.10) numaralı kısıt dağıtım yapılacak müşteriden veya depodan başka bir dağıtım yapılacak olan müşteriye sadece bir kez geilmesini sağlamaktadır. (5.11) numaralı kısıt toplama yapılacak olan müşteriye, depodan veya dağıtım yapılacak olan müşteriden veya başka bir toplama yapılacak olan müşteriden bir kez geiş

olmasını sağlamaktadır. (5.12) numaralı kısıt toplama yapılacak olan müşteriden depoya veya başka bir toplama yapılacak olan müşteriye geçişin sadece bir kez olmasını sağlamaktadır. (5.13) numaralı kısıt dağıtım yapılacak müşteriler ile toplama yapılacak müşteriler arasındaki geçişin en fazla bir kez olmasını sağlamaktadır.

Rotalama aşaması kısaca aşağıdaki gibidir.

- Her bir araç için araca atanan şehirlerin arasındaki mesafeleri hesaplanır.
- ÖDST-ARP için önerilen modeli oluştur ve her bir araç için çözülerek rotalar elde edilir.

5.3. Yerel Arama

İki aşamadan sonra elde edilen rotayı iyileştirmek için yerel arama kullanılmıştır. Bunun için ekleme ve değiştirme operatörlerinden yararlanılmıştır. Ekleme operatöründe, bir müşteri bulunduğu araçtan alınır ve eğer başka bir aracın kapasitesi uygun ise bu araca eklenmeye çalışılır. Bu müşterinin yeni yeri ile birlikte rotanın toplam mesafesinde bir azalma oluyor ise yeni rota oluşturulmuş olur. Değiştirme operatöründe, iki farklı araçta yer alan iki müşteri yine araçların kapasiteleri uygun ise karşılıklı yer değiştirilir ve müşteriler yeni araçlarında en uygun yere yerleştirilmeye çalışılırlar. Müşterilerin bu yeni yerleri ile birlikte rotanın toplam mesafesinde bir azalma oluyor ise yeni rota oluşturulmuş olur.

Bir şehir bulunduğu araçtan alınıp yeni bir araca yerleştiriliyorken ÖDST-ARP'nin özelliklerinin sağlanabilmesi için bazı kısıtların göz önüne alınması gerekmektedir. Dağıtım yapılan bir müşteri başka bir araca, depodan sonra veya dağıtım yapılan başka bir müşteriden sonra eklenebilir. Toplam yapılan bir müşteri ise başka bir araca depodan önce, toplama yapılan başka bir müşteriden sonra veya dağıtım yapılan müşteri ile toplama yapılan müşteri arasına eklenebilir.

Yerel arama kısaca aşağıdaki gibidir.

- Ekleme operatörü kullanılarak, toplam mesafede bir iyileşme oluyorsa ÖDST-ARP kuralları göz önüne alınarak bir müşteri orijinal aracından alınarak başka bir araca eklenir.
- Değiştirme operatörü kullanılarak, toplam mesafede bir iyileşme oluyorsa ÖDST-ARP kuralları göz önüne alınarak iki müşteri karşılıklı yer değiştirilerek bu müşteriler yeni araçlarında en iyi yere yerleştirilirler.

6. TEST PROBLEMLERİNİN ÖNERİLEN YÖNTEM İLE ÇÖZÜLMESİ

Önerilen yöntem ile Goeschalckx ve Jacobs (1989) tarafından önerilen, literatürdeki diğer çalışmalarda da kullanılan test problemleri çözülmüş ve literatürdeki bilinen en iyi değerler (Brandão, 2006) ile karşılaştırılmıştır. Problemlerde müşterilerin x koordinatları $[0, 2400]$ arasında düzgün dağılıma, y koordinatları ise $[0, 3200]$ arasında düzgün dağılıma uymaktadır. Deponun koordinatları ise tüm problemler için aynı olup (12000, 16000)'dir ve merkezde yer almaktadır. Uzaklıklar ise Öklid uzaklığına göre hesaplanmaktadır. Bu problem seti 14 ana problemten ve bu 14 ana problemin araç kapasitesi ve araç sayısı değiştirilerek elde edilmiş 62 alt problemten oluşmaktadır. Problemlerin özellikleri Ek-3'te verilmiştir.

Bölge oluşturma aşaması ve rotalama aşaması için GAMS yazılımının CPLEX 9.0 çözücüsü kullanılmıştır. GAMS yazılımına yazılan kodlar her iki aşama için sırası ile EK 1 ve EK 2'de verilmiştir. Daha önceki bölümde anlatıldığı üzere MS EXCEL VBA, GAMS yazılımı için gerekli verilerin düzenlenmesi, GAMS yazılımından elde edilen sonuçların yazdırılması ve yerel arama aşaması için kullanılmıştır. Problemler 2.20 GHz ve 1GB RAM özelliğine sahip olan bilgisayarda çözdürülmüştür.

6.1. Hesapsal Sonuçlar

Test problemleri önerilen yöntem ile çözülmüştür. Bölgelere ayırma aşamasında, her bir araca atanan şehirler hem her iki amaca göre hem de çok amaçlı model bulanık çözüm yaklaşımı olan min operatörü, iki aşamalı yaklaşım, ağırlıklı max-min modeli ve ağırlıklı toplam yaklaşımı ile çözümlenerek bulunmuştur. Amaçların ağırlıkları ikinci bölümde anlatılan metoda göre elde edilmiştir. Test problemleri için bulunan ağırlıklar ayrıntılı olarak EK-5'te verilmiştir. Rotalama aşamasında iki amaca ve bulanık çözüm yaklaşımlarına göre elde edilen bölgelere göre her bir araç için rotalar oluşturulmuş ve toplam taşıma mesafeleri her biri için hesaplanmıştır. Daha sonra yerel arama ile rotalar iyileştirilmiştir. Çizelge 6.1'de iki amaç fonksiyonuna göre ve bulanık çözüm yaklaşımlarına göre elde

edilen rotaların toplam mesafeleri, problemlerin şimdiye kadar bulunmuş olan en iyi değerleri (Brandão, 2006), elde edilen çözüm ile en iyi değer arasındaki fark ve çözüm için harcanan CPU zamanları saniye biriminde görülmektedir. Burada CPU-1 bölge oluşturma aşaması için harcanan zamanı, CPU-2 rotalama aşaması için harcanan zamanı ve CPU-3 yerel arama için harcanan zamanı göstermektedir. CPU-1 zamanı için bulanık çözüm aşamaları amaçların en iyi ve en kötü değerlerinin bulunması için harcanan zamanı da içermektedir. Ayrıca bu zaman min operatörü dışındaki diğer bulanık çözüm yaklaşımları için amaçlara atanacak ağırlıkların belirlenmesi için gereken süreyi de içermektedir. CPU zamanlarına ilişkin daha ayrıntılı tablo EK-4’de yer almaktadır. Ek-5’de üçüncü bölümde anlatılan ağırlıkların belirlenmesine ilişkin önerilen metoda göre belirlenen amaçların ağırlıkları ayrıntılı olarak verilmiştir.

Çizelge 6.1. Test problemlerinin sonuçları

No	İsim	Bilinen en iyi çözüm	Çözüm Yönt.	Öne. yön.	% Fark	CPU- 1	CPU- 2	CPU- 3	Top. CPU
1	A1	229886	z_1	233616,66	1,62	0,156	0,590	0,078	0,824
			z_2	229885,64	0	0,234	0,513	0,017	0,764
			Min. Ope.	230559,63	0,29	1,31	0,418	0,18	1,908
			İki aş.	230559,63	0,29	1,279	0,418	0,18	1,877
			Max. Min.	229885,64	0	1,232	0,513	0,017	1,762
			Ağ. Top.	230559,63	0,29	1,232	0,418	0,18	1,83
2	A2	180119	z_1	181521,93	0,78	0,031	0,295	0,031	0,357
			z_2	181883,37	0,98	0,031	0,435	0,02	0,486
			Min. Ope.	181883,37	0,98	0,233	0,435	0,02	0,688
			İki aş.	181883,37	0,98	0,201	0,435	0,02	0,656
			Max. Min.	181521,93	0,78	0,801	0,295	0,031	1,127
			Ağ. Top.	181883,37	0,98	0,264	0,435	0,02	0,719
3	A3	163405	z_1	168715,37	3,25	0,093	0,421	0,016	0,53
			z_2	169257,15	3,58	0,109	0,374	0,03	0,513
			Min. Ope.	168856,05	3,34	0,482	0,107	0,03	0,62
			İki aş.	170266,15	4,20	0,388	0,279	0,04	0,708
			Max. Min.	168715,37	3,25	0,435	0,421	0,016	0,873
			Ağ. Top.	170266,15	4,20	0,388	0,279	0,04	0,708
4	A4	155796	z_1	172890,69	10,97	0,234	0,124	0,047	0,405
			z_2	169002,29	8,48	0,109	0,217	0,03	0,356
			Min. Ope.	169002,29	8,48	0,983	0,217	0,03	1,23
			İki aş.	169002,29	8,48	0,717	0,217	0,03	0,964
			Max. Min.	169002,29	8,48	0,795	0,217	0,03	1,042
			Ağ. Top.	169002,29	8,48	0,764	0,217	0,03	1,011
5	B1	239080	z_1	253834,6	6,17	1,89	0,826	0,031	2,747
			z_2	242741,82	1,53	0,921	0,496	0,01	1,427
			Min. Ope.	248710,46	4,03	4,825	0,372	0,03	5,227
			İki aş.	248710,46	4,03	4,7	0,372	0,03	5,102
			Max. Min.	242741,82	1,53	5,231	0,496	0,01	5,737
			Ağ. Top.	248710,46	4,03	4,825	0,372	0,03	5,227
6	B2	198048	z_1	205067,2	3,54	0,62	0,669	0,031	1,32
			z_2	203046,83	2,52	0,453	0,295	0,03	0,561
			Min. Ope.	200905,25	1,44	1,603	0,326	0,03	1,959
			İki aş.	200677,67	1,33	1,509	0,17	0,03	1,709
			Max. Min.	203046,83	2,52	2,447	0,295	0,03	2,772
			Ağ. Top.	200677,67	1,33	1,509	0,17	0,03	1,709
7	B3	169372	z_1	184738,6	9,08	0,656	0,186	0,047	0,889
			z_2	176466,33	4,19	0,062	0,045	0,02	0,127
			Min. Ope.	194258,43	14,69	1,904	0,217	0,03	0,247
			İki aş.	194962,38	15,11	1,044	0,218	0,03	0,248
			Max. Min.	194962,38	15,11	1,232	0,218	0,03	0,248
			Ağ. Top.	194962,38	15,11	0,997	0,218	0,03	0,248
8	C1	249448	z_1	262136,83	5,09	1,765	0,528	0,031	2,324
			z_2	253317,56	1,55	2,593	0,592	0,048	3,233
			Min. Ope.	257462,82	3,21	12,06	0,45	0,035	12,55
			İki aş.	257462,82	3,21	13,03	0,373	0,03	13,43
			Max. Min.	253317,56	1,55	12,25	0,592	0,048	12,89
			Ağ. Top.	257462,82	3,213	13,12	0,373	0,03	13,53

No	İsim	Bilinen en iyi çözüm	Çözüm Yönt.	Öne. yön.	% Fark	CPU- 1	CPU- 2	CPU- 3	Top. CPU
9	C2	215020	z ₁	232302,36	6,93	1,734	0,341	0,031	2,106
			z ₂	237171,98	8,88	1,062	0,388	0,01	1,46
			Min. Ope.	232331,18	6,94	11,23	0,342	0,03	11,61
			İki aş.	238156,43	9,28	9,576	0,404	0,03	10,01
			Max. Min.	238156,43	9,28	10,03	0,155	0,03	10,21
			Ağ. Top.	238156,43	9,28	9,029	0,404	0,03	9,463
10	C3	199346	z ₁	216451,41	6,86	0,14	0,514	0,078	0,732
			z ₂	201945,79	1,04	0,093	0,248	0,08	0,421
			Min. Ope.	204892,29	2,22	4,091	0,264	0,06	4,415
			İki aş.	204892,29	2,22	1,451	0,264	0,06	1,775
			Max. Min.	216451,41	6,86	1,326	0,514	0,078	1,918
			Ağ. Top.	204892,29	2,22	1,685	0,264	0,06	2,009
11	C4	195366	z ₁	203982,76	3,45	0,796	0,374	0,078	1,248
			z ₂	199727,93	1,75	0,171	0,467	0,078	0,716
			Min. Ope.	202949,38	3,04	3,247	0,342	0,075	3,664
			İki aş.	206950,03	4,64	2,887	0,217	0,08	3,184
			Max. Min.	197283,44	0,77	2,809	0,248	0,08	3,137
			Ağ. Top.	206950,03	4,64	2,825	0,217	0,08	3,122
12	D1	322530	z ₁	322704,53	0,07	0,921	1,229	0,016	2,166
			z ₂	326234,94	1,49	0,359	0,433	0,02	0,812
			Min. Ope.	324502,41	0,79	4,137	0,979	0,016	5,132
			İki aş.	324502,41	0,79	4,168	0,979	0,016	5,163
			Max. Min.	322704,53	0,07	3,997	1,229	0,016	5,242
			Ağ. Top.	324502,41	0,79	4,278	0,979	0,016	5,273
13	D2	316709	z ₁	318475,7	0,71	3,296	0,869	0,031	4,196
			z ₂	325685,11	3,60	13,20	0,637	0,02	13,86
			Min. Ope.	320208,24	1,40	24,65	0,838	0,035	25,53
			İki aş.	318475,73	0,71	19,73	0,979	0,036	20,75
			Max. Min.	318475,7	0,71	21,81	0,869	0,031	22,71
			Ağ. Top.	318475,73	0,71	19,95	0,979	0,036	20,97
14	D3	239479	z ₁	239931,81	0,18	0,515	0,59	0,031	1,136
			z ₂	239801,21	0,13	0,14	0,454	0,027	0,621
			Min. Ope.	244003,03	1,81	3,764	0,591	0,037	4,392
			İki aş.	239801,21	0,13	3,201	0,622	0,037	3,86
			Max. Min.	239801,21	0,13	3,342	0,454	0,027	3,823
			Ağ. Top.	239801,21	0,13	3,185	0,622	0,037	3,844
15	D4	205832	z ₁	213785,8	3,86	0,718	0,373	0,047	1,138
			z ₂	208823,16	1,45	0,171	0,405	0,016	0,592
			Min. Ope.	213795,80	3,87	3,184	0,404	0,032	3,62
			İki aş.	213795,80	3,87	2,84	0,404	0,032	3,276
			Max. Min.	208823,16	1,45	2,824	0,405	0,016	3,245
			Ağ. Top.	213795,80	3,87	2,793	0,404	0,032	3,229
16	E1	238880	z ₁	244058,12	2,17	0,281	0,59	0,047	0,918
			z ₂	241697,19	1,18	1,937	0,387	0,045	2,369
			Min. Ope.	244307,7	2,27	9,232	0,449	0,04	9,721
			İki aş.	244345,78	2,29	6,904	0,466	0,066	7,436
			Max. Min.	246410,07	3,15	6,998	0,184	0,061	7,243
			Ağ. Top.	244345,78	2,29	6,607	0,466	0,066	7,139
17	E2	212263	z ₁	225697,2	6,33	1,875	0,2	0,078	2,153
			z ₂	214118,64	0,87	0,421	0,248	0,055	0,724
			Min. Ope.	227633,18	7,24	5,514	0,264	0,095	5,873
			İki aş.	227633,18	7,24	3,123	0,263	0,08	3,466
			Max. Min.	226604,61	6,76	4,389	0,28	0,076	4,745
			Ağ. Top.	227633,18	7,24	3,17	0,263	0,08	3,513

No	İsim	Bilinen en iyi çözüm	Çözüm Yönt.	Öne. yön.	% Fark	CPU- 1	CPU- 2	CPU- 3	Top. CPU
18	E3	206659	z ₁	219863,09	6,39	0,39	0,373	0,14	0,903
			z ₂	217882,27	5,43	0,171	0,185	0,14	0,496
			Min. Ope.	217882,27	5,43	2,263	0,294	0,14	2,697
			İki aş.	223114,34	7,96	1,122	0,184	0,15	1,456
			Max. Min.	217882,27	5,43	0,982	0,185	0,14	1,307
			Ağ. Top.	223114,34	7,96	1,029	0,184	0,15	1,363
19	F1	263173	z ₁	272825,72	3,67	177,1	0,512	0,16	177,7
			z ₂	280159,35	6,45	15,96	0,389	0,094	16,45
			Min. Ope.	269125,66	2,26	274,3	1,231	0,1	275,7
			İki aş.	275928,51	4,85	246,1	0,512	0,07	246,7
			Max. Min.	280159,35	6,45	225,2	0,389	0,094	225,7
			Ağ. Top.	275928,51	4,85	236,5	0,512	0,07	237
20	F2	265213	z ₁	282636,39	6,60	59,54	0,951	0,22	60,71
			z ₂	273679,56	3,19	2,968	0,545	0,2	3,713
			Min. Ope.	275569,71	3,91	82,69	0,466	0,1	83,25
			İki aş.	286217,2	7,92	76,08	0,544	0,11	76,73
			Max. Min.	273679,56	3,19	75,09	0,545	0,2	75,84
			Ağ. Top.	286217,2	7,92	78,58	0,544	0,11	79,23
21	F3	241120	z ₁	247463,58	2,63	0,187	0,56	0,28	1,027
			z ₂	242030,3	0,38	0,203	0,357	0,16	0,72
			Min. Ope.	249570,75	3,50	4,108	0,498	0,15	4,756
			İki aş.	243658,94	1,05	1,358	0,497	0,18	2,035
			Max. Min.	243658,94	1,05	2,124	0,357	0,16	2,641
			Ağ. Top.	247731,69	2,74	1,374	0,497	0,18	2,051
22	F4	233861	z ₁	249843,86	6,83	0,218	0,778	0,41	1,406
			z ₂	248337,73	6,19	0,203	0,733	0,26	1,196
			Min. Ope.	257448,59	10,09	7,967	0,78	0,29	9,037
			İki aş.	250967,86	7,31	3,936	109,5	0,25	113,8
			Max. Min.	248337,73	6,19	4,061	0,733	0,26	5,054
			Ağ. Top.	250967,86	7,31	3,92	109,5	0,25	113,8
23	G1	306305	z ₁	348475,39	13,77	162,8	0,84	0,047	163,7
			z ₂	316634,26	3,37	53,88	0,669	0,05	54,59
			Min. Ope.	321488,09	4,96	468,9	0,808	0,05	469,8
			İki aş.	323916,1	5,75	306,6	0,823	0,05	307,4
			Max. Min.	327307,3	6,86	521,6	0,732	0,04	522,4
			Ağ. Top.	323916,1	5,75	307,6	0,823	0,05	308,4
24	G2	245441	z ₁	257542,6	4,93	6,046	0,686	0,063	6,795
			z ₂	254202,52	3,57	5,078	0,639	0,07	5,787
			Min. Ope.	252095,46	2,71	26,67	0,466	0,08	27,22
			İki aş.	252095,46	2,71	26,08	0,466	0,08	26,62
			Max. Min.	254202,52	3,57	22,84	0,639	0,07	23,55
			Ağ. Top.	252095,46	2,71	25,03	0,466	0,08	25,58
25	G3	229507	z ₁	251096,55	9,41	2,109	0,608	0,109	2,826
			z ₂	242530,2	5,67	2,984	0,858	0,16	4,002
			Min. Ope.	251706,14	9,67	33,44	0,685	0,06	34,18
			İki aş.	246993,91	7,62	18,01	0,81	0,06	18,88
			Max. Min.	242530,2	5,67	14,29	0,858	0,16	15,32
			Ağ. Top.	246993,91	7,62	16,42	0,81	0,06	17,29
26	G4	232521	z ₁	249652,44	7,37	0,14	0,686	0,203	1,029
			z ₂	246865,20	6,17	0,125	0,498	0,35	0,973
			Min. Ope.	248958,17	7,07	1,967	0,872	0,16	2,999
			İki aş.	247488,87	6,44	1,796	0,809	0,16	2,765
			Max. Min.	245334,1	5,51	2,514	0,779	0,25	3,543
			Ağ. Top.	247488,87	6,44	1,764	0,809	0,16	2,733

No	İsim	Bilinen en iyi çözüm	Çözüm Yönt.	Öne. yön.	% Fark	CPU- 1	CPU- 2	CPU- 3	Top. CPU
27	G5	221730	z_1	237890,61	7,29	2,812	0,56	0,281	3,653
			z_2	232402,14	4,81	0,14	2,217	0,19	2,547
			Min. Ope.	244001,4	10,04	7,153	2,107	0,16	9,42
			İki aşı.	241586,53	8,96	4,372	0,904	0,19	5,466
			Max. Min.	232402,14	4,81	4,419	2,217	0,19	6,826
			Ağ. Top.	241586,53	8,96	4,403	0,904	0,19	5,497
28	G6	213457	z_1	232493,67	8,92	0,171	1,176	0,156	1,503
			z_2	227554,09	6,60	0,093	1,107	0,51	1,71
			Min. Ope.	235355,42	10,26	6,685	0,404	0,55	7,639
			İki aşı.	232826,11	9,07	2,7	172,2	0,21	175,1
			Max. Min.	227554,09	6,60	2,122	1,107	0,51	3,739
			Ağ. Top.	232826,11	9,07	2,513	172,2	0,21	174,9
29	H1	268933	z_1	278294,89	3,48	3,921	6,39	0,219	10,53
			z_2	278072,61	3,40	1,921	0,388	0,2	2,509
			Min. Ope.	277081,44	3,03	44,04	0,451	0,22	44,72
			İki aşı.	275089,18	2,29	43,29	0,606	0,23	44,13
			Max. Min.	278072,61	3,40	39,65	0,388	0,2	40,24
			Ağ. Top.	275089,18	2,29	36,92	0,606	0,23	37,76
30	H2	253365	z_1	268184,84	5,85	0,593	0,679	0,484	1,756
			z_2	265971,31	4,98	0,359	0,608	0,5	1,467
			Min. Ope.	263910,43	4,16	8,17	0,654	0,48	9,304
			İki aşı.	259990,48	2,61	4,107	0,827	0,48	5,414
			Max. Min.	262530,54	3,62	10,72	0,637	0,49	11,84
			Ağ. Top.	259990,48	2,61	3,31	0,827	0,48	4,617
31	H3	247449	z_1	257971,95	4,25	0,312	0,5	0,562	1,374
			z_2	252795,73	2,16	0,234	1,108	0,69	2,032
			Min. Ope.	257534,55	4,08	12,06	0,67	0,67	13,40
			İki aşı.	257534,55	4,08	9,404	0,67	0,67	10,74
			Max. Min.	255446,83	3,23	11,15	0,577	0,66	12,39
			Ağ. Top.	257534,55	4,08	9,326	0,67	0,67	10,67
32	H4	250221	z_1	262844,04	5,04	0,281	1,717	0,64	2,638
			z_2	261617,67	4,55	0,171	0,638	0,52	1,329
			Min. Ope.	261165,67	4,37	4,685	0,669	0,49	5,844
			İki aşı.	260948,26	4,29	1,248	0,576	0,51	2,334
			Max. Min.	265296,48	6,02	6,935	0,701	0,53	8,166
			Ağ. Top.	260948,26	4,29	1,107	0,576	0,51	2,193
33	H5	246121	z_1	251675,26	2,26	0,203	1,03	0,734	1,967
			z_2	250643,87	1,84	0,109	2,155	0,97	3,234
			Min. Ope.	252099,14	2,43	1,842	2,153	0,64	4,635
			İki aşı.	258909,06	5,20	1,404	2,062	0,65	4,116
			Max. Min.	251675,26	2,26	6,607	0,904	0,69	8,201
			Ağ. Top.	258909,06	5,20	1,31	2,062	0,65	4,022
34	H6	249135	z_1	269204,3	8,06	0,187	1,138	0,64	1,965
			z_2	262004,65	5,17	0,125	2,232	0,59	2,947
			Min. Ope.	260948,26	4,74	4,56	3,53	0,51	8,6
			İki aşı.	260705,99	4,64	0,904	2,342	0,51	3,756
			Max. Min.	263978,17	5,96	6,123	2,326	0,59	9,039
			Ağ. Top.	260705,99	4,64	0,795	2,342	0,51	3,647
35	I1	350246	z_1	377325,39	7,73	29,34	1,654	0,094	31,09
			z_2	360923,02	3,05	1000	0,557	0,08	1001
			Min. Ope.	375205,93	7,13	2924	0,527	0,08	2924
			İki aşı.	375205,93	7,13	2856	0,682	0,08	2857
			Max. Min.	386984,79	10,49	2865	1,654	0,094	2867
			Ağ. Top.	375205,93	7,13	2850	0,682	0,08	2851

No	İsim	Bilinen en iyi çözüm	Çözüm Yönt.	Öne. yön.	% Fark	CPU- 1	CPU- 2	CPU- 3	Top. CPU
36	I2	309943	z_1	350883,16	13,21	53,59	1,29	0,265	55,15
			z_2	335533,47	8,26	203,7	0,545	0,3	204,6
			Min. Ope.	333666,75	7,65	456,3	0,606	0,27	457,2
			İki aşı.	341503,55	10,18	473,9	0,513	0,27	474,6
			Max. Min.	335533,47	8,26	493,4	0,545	0,3	494,3
			Ağ. Top.	341503,55	10,18	450,6	0,513	0,27	451,5
37	I3	294507	z_1	317518,54	7,81	33,14	2,458	0,5	36,1
			z_2	307021,17	4,25	3,531	1,716	0,58	5,827
			Min. Ope.	308554,61	4,77	140,1	1,607	0,55	142,3
			İki aşı.	307690,87	4,48	123,8	1,843	0,53	126,1
			Max. Min.	307690,87	4,48	116,6	1,607	0,43	118,6
			Ağ. Top.	307690,87	4,48	126,8	1,843	0,53	129,2
38	I4	295988	z_1	313700,94	5,98	2,671	0,496	0,453	3,62
			z_2	305436,66	3,19	0,39	0,623	0,53	1,543
			Min. Ope.	315392,12	6,56	33,09	2,356	0,52	35,97
			İki aşı.	308916,65	4,37	14,75	1,012	0,58	16,34
			Max. Min.	305436,66	3,19	14,89	0,623	0,53	16,04
			Ağ. Top.	308916,65	4,37	14,58	1,012	0,58	16,17
39	I5	301226	z_1	333636,19	10,76	0,968	0,887	0,422	2,277
			z_2	306741,89	1,83	0,625	0,746	0,57	1,941
			Min. Ope.	323713,9	7,47	13,8	0,904	0,61	15,31
			İki aşı.	322094,39	6,93	6,108	0,887	0,48	7,475
			Max. Min.	308824,94	2,52	7,045	0,59	0,52	8,155
			Ağ. Top.	322094,39	6,93	5,905	0,887	0,48	7,272
40	J1	335006	z_1	358482,1	7,01	358,7	0,65	0,297	359,7
			z_2	343562,72	2,55	426,9	1,605	0,12	428,7
			Min. Ope.	357200,03	6,62	2425	0,683	0,13	2426
			İki aşı.	358509,93	7,02	2236	0,823	0,13	2237
			Max. Min.	358482,1	7,01	2548	0,65	0,297	2549
			Ağ. Top.	358509,93	7,02	2207	0,823	0,13	2208
41	J2	310417	z_1	328379,79	5,79	13,95	0,841	0,844	15,63
			z_2	319024,99	2,77	51,3	2,059	0,36	53,72
			Min. Ope.	327772,74	5,59	748,8	0,574	0,4	749,7
			İki aşı.	329007,84	5,99	654,5	0,73	0,47	655,7
			Max. Min.	328379,79	5,79	713,3	0,841	0,844	714,1
			Ağ. Top.	329007,84	5,99	657	0,73	0,47	658,2
42	J3	279219	z_1	314709,7	12,71	0,75	1,121	2,438	4,309
			z_2	308626,42	10,53	14,8	10,87	1,53	27,2
			Min. Ope.	309753,47	10,94	66,03	1,778	2,02	69,83
			İki aşı.	313152,67	12,15	56,97	2,342	2	61,31
			Max. Min.	314709,7	12,71	58,44	1,122	2,438	59,56
			Ağ. Top.	313152,67	12,15	56,67	2,342	2	61,01
43	J4	296553	z_1	335658,32	13,19	32,84	0,95	0,844	34,64
			z_2	302086,73	1,87	24,08	0,873	1,47	26,42
			Min. Ope.	321089,82	8,27	187,4	0,809	1,47	189,7
			İki aşı.	317035,65	6,91	152,9	0,762	1,34	155
			Max. Min.	320723,73	8,15	148,8	0,871	0,87	150,5
			Ağ. Top.	317035,65	6,91	126,5	0,762	1,34	128,6
44	K1	394376	z_1	416054,7	5,50	71,39	0,62	0,39	72,4
			z_2	409172,29	3,75	1000	0,995	0,33	1001
			Min. Ope.	410944,23	4,20	2941	0,574	0,27	2942
			İki aşı.	410474,67	4,08	2471	0,824	0,25	2472
			Max. Min.	409172,29	3,75	2940	0,995	0,33	2941
			Ağ. Top.	410474,67	4,08	2261	0,824	0,25	2262

No	İsim	Bilinen en iyi çözüm	Çözüm Yönt.	Öne. yön.	% Fark	CPU- 1	CPU- 2	CPU- 3	Top. CPU
45	K2	362130	z_1	380674,36	5,12	148,4	0,637	0,578	149,6
			z_2	378950,9	4,64	12,53	1,277	0,36	14,17
			Min. Ope.	382184,13	5,54	1030	0,934	0,27	1031
			İki aşı.	379925,14	4,91	1015	0,839	0,36	1016
			Max. Min.	378950,9	4,64	970,6	1,277	0,36	972,3
			Ağ. Top.	379925,14	4,91	993,4	0,839	0,36	994,6
46	K3	364693	z_1	384716,62	5,49	2,531	0,996	0,797	4,324
			z_2	382136,49	4,78	2,765	1,279	0,33	4,37
			Min. Ope.	397272,72	8,93	1132	1,574	0,59	1134
			İki aşı.	390705,85	7,13	1012	0,699	0,47	1013
			Max. Min.	382136,49	4,78	1029	1,279	0,33	1030
			Ağ. Top.	390705,85	7,13	1010	0,699	0,47	1011
47	K4	348950	z_1	358542,23	2,75	58,31	11,64	0,703	70,66
			z_2	363967,08	4,30	17,42	1,811	0,46	19,69
			Min. Ope.	372760,57	6,82	104,8	71,06	0,78	176,7
			İki aşı.	366357,52	4,98	100,3	0,965	0,62	101,9
			Max. Min.	363967,08	4,30	115	1,811	0,46	117,3
			Ağ. Top.	366357,52	4,99	104,8	0,965	0,62	106,4
48	L1	425772	z_1	454063,78	6,64	1000	2,465	1,031	1004
			z_2	446363,8	4,84	1000	2,495	1,01	1004
			Min. Ope.	482588,53	13,34	4022	7,353	1,05	4030
			İki aşı.	448803,19	5,41	4420	2,387	0,82	4423
			Max. Min.	448803,19	5,41	4008	2,387	0,82	4011
			Ağ. Top.	448803,19	5,41	4241	2,387	0,82	4244
49	L2	401228	z_1	455224,45	13,46	1000	14,58	1,203	1016
			z_2	433709,97	8,10	1000	11,50	1,22	1013
			Min. Ope.	461510,67	15,02	3237	14,40	1,5	3253
			İki aşı.	461194,2	14,95	4219	5,995	0,79	4225
			Max. Min.	461194,2	14,94	3234	5,995	0,79	3241
			Ağ. Top.	439555,22	9,55	4219	6,965	0,82	4226
50	L3	402720	z_1	455144,75	13,02	258,5	3,738	1,469	263,7
			z_2	433865,14	7,73	126,4	7,777	1,01	135,2
			Min. Ope.	457572,85	13,62	1443	18,83	1,19	1463
			İki aşı.	425656,25	5,70	1792	8,2	1,08	1801
			Max. Min.	425656,25	5,70	1440	8,2	1,08	1449
			Ağ. Top.	425656,25	5,70	1902	8,2	1,08	1911
51	L4	384637	z_1	415244,05	7,96	143,5	7,981	1,641	153,1
			z_2	415829,94	8,11	126,3	75,51	1,89	203,7
			Min. Ope.	448574,54	16,62	693,9	38,87	1,91	734,7
			İki aşı.	432122,46	12,35	1679	129,3	1,29	1810
			Max. Min.	432122,46	12,35	685,8	129,3	1,29	816,4
			Ağ. Top.	414850,14	7,85	1125	123,3	1,99	1250
52	L5	387928	z_1	421214,24	8,58	39,15	13,02	1,953	54,12
			z_2	413734,22	6,65	3,421	24,12	1,67	29,21
			Min. Ope.	442417,44	14,05	193,2	35,40	1,81	230,4
			İki aşı.	409964,77	5,68	187,2	3,716	1,47	192,4
			Max. Min.	413734,22	6,65	193,6	24,12	1,67	219,4
			Ağ. Top.	409964,77	5,68	185,6	3,716	1,47	190,8
53	M1	398593	z_1	437766,07	9,83	101,6	0,542	0,328	102,5
			z_2	413781,01	3,81	125,3	1,573	0,14	127
			Min. Ope.	419549,58	5,26	1997	0,715	0,17	1998
			İki aşı.	421512,4	5,75	1795	0,808	0,17	1796
			Max. Min.	437766,07	9,83	2041	0,542	0,328	2041
			Ağ. Top.	421512,4	5,75	1762	0,808	0,17	1763

No	İsim	Bilinen en iyi çözüm	Çözüm Yönt.	Öne. yön.	% Fark	CPU- 1	CPU- 2	CPU- 3	Top. CPU
54	M2	396917	z ₁	431226,02	8,64	1000	0,95	0,297	1001
			z ₂	411559,88	3,70	287,8	1,839	0,14	289,8
			Min. Ope.	478049,74	20,44	3476	5,012	1,13	3482
			İki aşı.	419809	5,77	3476	0,948	0,15	3477
			Max. Min.	419809	5,77	2493	0,948	0,15	2494
			Ağ. Top.	424466,9	6,94	3476	1,215	0,14	3477
55	M3	376309	z ₁	407302,83	8,24	254,5	2,218	0,375	257,1
			z ₂	398154,2	5,81	540,8	2,622	0,14	543,6
			Min. Ope.	435063,95	15,61	2002	1,012	0,25	2003
			İki aşı.	405274,59	7,70	2317	1,824	0,3	2319
			Max. Min.	398154,2	5,81	2256	2,622	0,14	2259
			Ağ. Top.	405274,59	7,70	2099	1,824	0,3	2101
56	M4	348418	z ₁	364889,49	4,73	17,86	1,091	0,469	19,42
			z ₂	366058,23	5,06	133,2	4,419	0,33	138
			Min. Ope.	410664,26	17,87	503,9	25,25	0,38	529,5
			İki aşı.	372530,06	6,92	656,7	0,903	0,44	658
			Max. Min.	366159,84	5,09	1041	1,576	0,38	1043
			Ağ. Top.	372530,06	6,92	607,3	0,903	0,44	608,6
57	N1	408101	z ₁	464954,44	13,93	16,16	1,495	2,375	20,03
			z ₂	429965,31	5,36	1000	1,92	1,18	1003
			Min. Ope.	488193,83	19,63	2654	93,1	2,59	2750
			İki aşı.	439585,7	7,71	2773	2,07	1,45	2776
			Max. Min.	439585,7	7,71	2649	2,07	1,45	2653
			Ağ. Top.	439585,7	7,71	2698	2,07	1,45	2702
58	N2	408066	z ₁	441412,86	8,17	1000	2,763	1,75	1005
			z ₂	431030,74	5,63	1000	2,528	0,89	1003
			Min. Ope.	483186,17	18,41	2846	14,50	1,59	2862
			İki aşı.	436992,32	7,09	3579	6,01	1,28	3586
			Max. Min.	436992,32	7,09	2843	6,01	1,28	2850
			Ağ. Top.	436992,32	7,09	3159	6,01	1,28	3166
59	N3	394338	z ₁	430575,08	9,19	614,2	4,355	3,453	622
			z ₂	414268,98	5,05	64,92	4,089	2,11	71,1
			Min. Ope.	459135,35	16,43	2711	191,7	2,24	2904
			İki aşı.	423336,6	7,35	2736	5,434	2,19	2743
			Max. Min.	423336,6	7,35	2698	5,434	2,19	2705
			Ağ. Top.	423336,6	7,35	2716	5,434	2,19	2724
60	N4	396055	z ₁	436029,96	10,09	3,562	5,479	3,781	12,82
			z ₂	414534,4	4,67	15,45	2,933	2,12	20,5
			Min. Ope.	479558,05	21,08	1921	20,06	3,21	1945
			İki aşı.	428429,48	8,17	1915	4,37	1,94	1921
			Max. Min.	428429,48	8,17	1919	4,37	1,94	1925
			Ağ. Top.	428429,48	8,17	1912	4,37	1,94	1918
61	N5	373477	z ₁	426575,27	14,22	84,43	99,81	5,39	189,6
			z ₂	398001,58	6,57	23,5	12,78	2,79	39,07
			Min. Ope.	428442,38	14,72	272,3	457,9	5,29	735,7
			İki aşı.	399772,34	7,04	350,0	23,22	5,64	378,9
			Max. Min.	399772,34	7,04	272,4	23,22	5,64	301,3
			Ağ. Top.	399772,34	7,04	295,9	23,22	5,64	324,8
62	N6	374691	z ₁	423357,66	12,99	10,88	84,14	5,5	100,5
			z ₂	400781,19	6,96	21,56	7,794	3,1	32,46
			Min. Ope.	447204,28	19,35	235,7	464,7	4,78	705,2
			İki aşı.	403932,29	7,80	212,1	3,482	5,33	220,9
			Max. Min.	403932,29	7,80	232,6	3,482	5,33	241,4
			Ağ. Top.	403932,29	7,80	205,6	3,482	5,33	214,5

Önerilen yöntem ile bilinen en iyi değerlere yakın çözümler elde edilmiştir. Ancak A1 test problemi için ikinci amaç ve ağırlıklı max-min operatörüne göre elde edilen çözüm ile en iyi değere ulaşılmıştır. Bu çözüm Çizelge 6.2’de görülmektedir.

Çizelge 6.2. A1 problemi için bulunan çözüm

Mesafe	Rota						
29735,83	0	13	2	16	0		
16130,01	0	10	3	22	0		
30425,82	0	4	15	12	25	0	
10758,42	0	6	0				
42904,26	0	8	20	7	5	23	0
35882,23	0	1	14	18	0		
24583,22	0	19	17	21	0		
39465,85	0	11	9	24	0		
229885,6							

Elde edilen çözümler ortalama olarak bilinen en iyi değerler ile olan ortalama fark %3,81 olarak bulunmuştur. En büyük fark %10,53 ile J3 test problemi için ikinci amaç fonksiyonuna göre bulunmuştur. En küçük fark %0,07 ile D1 test problemi için birinci amaç fonksiyonuna göre ve ağırlıklı max-min modeline göre bulunmuştur.

Genel olarak ikinci amaç fonksiyonu ile bulunan çözüm birinci amaç fonksiyonu ile bulunan çözümden daha iyidir. Benzer şekilde genel olarak iki aşamalı yaklaşım ile bulunan sonuç min operatörü ile bulunan sonuca göre daha iyidir.

En iyi sonuçlar 42 test problemi ile ikinci amaç fonksiyonuna göre elde edilen rotalar ile bulunmuştur. 22 test probleminde ağırlıklı max-min modeli ile bulunan sonuçlar daha iyi iken, 10’ar test problemi ile iki aşamalı yaklaşım ve ağırlıklı toplam operatörü, 7 test problemi ile birinci amaç fonksiyonu ve 5 test problemi ile min operatörü ile bulunan çözümler diğerlerine göre daha iyi olmuştur.

Problemlerin çözümü için gerekli CPU zamanları problem boyutuna bağlı olarak artış göstermiştir. En kısa CPU-1 zamanı A2 test problemi için 0,031sn

değeriyle birinci ve ikinci amaç fonksiyonları ile, en kısa CPU-2 zamanı B3 test problemi için 0,045sn değeriyle ikinci amaç fonksiyonu ile , en kısa CPU-3 zamanı C2 test problemi için 0,01sn değeriyle ikinci amaç fonksiyonu ile ve en kısa toplam CPU zamanı ise B3 test problemi için 0,127sn değeri ile ikinci amaç fonksiyonuna göre elde edilen çözümler için olmuştur. En büyük CPU-1 zamanı L1 test problemi için 4420sn değeri ile iki aşamalı yaklaşım ile, en büyük CPU-2 zamanı N6 test problemi için 464,7sn değeri ile min operatörü ile, en büyük CPU-3 zamanı N5 test problemi için 5,64sn değeri ile iki aşamalı yaklaşım, ağırlıklı max-min modeli ve ağırlıklı toplam operatörü ile ve en büyük toplam CPU değeri L1 test problemi için 4423sn değeri ile iki aşamalı yaklaşım ile elde edilen çözümde olmuştur.

Her bir problem için bakıldığında bulanık çözüm yaklaşımları ile bulunan çözümler için harcanan CPU zamanları birbirine yakındır. Bu konuda yöntemlerin birbirine olan herhangi bir baskınlığın olmadığı söylenebilir. Amaçlara göre çözüm elde etmek için daha az CPU zamanı harcanmaktadır. Bunun sebebi ise amaçların en kötü değerlerinin hesaplanmasına gerek olmamasıdır.

6.2. Test Problemleri İçin Geliştirilen Ara Yüz

Test problemleri için oluşturulmuş olan ara yüz bir kaç sayfadan oluşmaktadır.

6.2.1. Bölge oluşturma aşamasında kullanılan ara yüz

a) “Veri” ve “Bölge Veri” sayfaları:

Problemlerin çözümü için bölgeleri oluşturma aşamasında amaçların oluşturulmasında gerekli olan uzaklıklar ve tasarruf miktarının hesaplaması için tasarlanan ara yüzde “veri” sayfasına toplama yapılacak müşterilerin ve dağıtım yapılacak müşterilerin koordinatları ilgili sütünün altına girilmelidir. Toplama yapılacak müşterilerin ilki olarak depo eklenmelidir. Ayrıca bu aşamadaki modelin kısıtları için gerekli olan araç kapasitesi ve gerekli araç sayısı sayfada görülen ilgili hücreye yazılmalıdır. No kısmını program kendisi atar.

“Veri” sayfasının boş hali Şekil 6.1’de görülmektedir.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	KOORDİNATLAR		MÜŞTERİLER			KOORDİNATLAR		TEDARİKÇİLER		Kapasite		Araç Sayısı	
2	X	Y	NO	TALEP		X	Y	NO	TALEP				
3													
4													
5													
6													
7													
8													
9													
10													
11													
12													
13													
14													
15													
16													
17													
18													
19													
20													
21													
22													
23													
24													
25													
26													
27													
28													
29													
30													
31													
32													
33													
34													

Şekil 6.1. Veri girişi için tasarlanan ara yüz

“Veri” sayfasının bölge ve müşteri bilgileri, araç kapasitesi ile araç sayısı bilgileri girilmiş hali Şekil 6.2’de görülmektedir

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	KOORDİNATLAR		MÜŞTERİLER			KOORDİNATLAR		TEDARİKÇİLER		Kapasite	4050	Araç Sayısı	3	
2	X	Y	NO	TALEP		X	Y	NO	TALEP					
3	12000	16000		0		9795	21510			549				
4	21524	24879		483		7758	10895			558				
5	22337	17915		389		12927	2464			851				
6	7642	9722		435		23286	5538			388				
7	3707	20036		452		7416	15826			194				
8	15492	2721		343										
9	12430	10638		444										
10	13595	1918		350										
11	4290	2721		307										
12	18684	5268		810										
13	8686	9546		1002										
14	23140	10989		476										
15	5263	14289		674										
16	19557	17564		691										
17	23281	29684		675										
18	374	22683		410										
19	18804	9082		351										
20	5444	25004		786										
21	20390	25745		59										
22	4310	21692		362										
23	8526	2495		550										
24														
25														
26														
27														
28														
29														
30														
31														
32														
33														
34														

Şekil 6.2. Veriler girildikten sonra ara yüzün görünümü

“Verileri Düzenle” butonuna basıldığında program “Veri”, “Bölge Veri” ve “Tasarruf” sayfasına gerekli parametreleri yazdırır.

“Verileri Düzenle” butonuna basıldığında yerel aramada kullanılacağı için depoyu da içeren uzaklık matrisi veri ekranına hesaplar ve her defasında K2 hücresinden başlayarak uzaklık matrisini sayfaya yazdırır. Şekil 6.3’de gerekli hesaplamalar program tarafından sayfaya yazdırıldıktan sonraki sayfanın görünümü yer almaktadır.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	COORDİNATLAR	MÜŞTERİLER				COORDİNATLAR	TEDARİKÇİLER	Kapasite			4050	Araç Sayısı	3										
2	X	Y	NO	TALEP		X	Y	NO	TALEP														
3	12000	16000	1	0		12000	16000	1	0		1	0	130020,9	10512,9	7642,35	9222,97	13730,5	5379,21	14172	15355	12643,2	7255,12	12215,1
4	21524	24879	2	483		21524	24879	2	0		2	130020,876	0	7011,3	20553,5	18463,5	22964,4	16896,9	24291,5	28071,1	19815,6	19997,9	13983,7
5	22337	17915	3	389		22337	17915	3	0		3	10512,887	7011,3	0	16824,6	18750,3	16664,7	12292,4	18229,8	23591,4	13164	16012,2	6972,39
6	7642	9722	4	435		7642	9722	4	0		4	7642,3457	20553,5	16824,6	0	11039,1	10518,4	4874,83	9815,33	7762,08	11906,5	1058,73	15549,7
7	3707	20036	5	452		3707	20036	5	0		5	9222,9683	18463,5	18750,3	11039,1	0	20945,1	12822,4	20640,6	17324,8	21033,4	11611,7	21435,7
8	15492	2721	6	343		15492	2721	6	0		6	13730,474	22964,4	16664,7	10518,4	20945,1	0	8488,51	2059,96	11202	4085,63	9638,58	11262,8
9	12430	10638	7	444		12430	10638	7	0		7	5379,2141	16896,9	12292,4	4874,83	12822,4	8488,51	0	8797,48	11355,1	8240,14	3900	10715,8
10	13595	1918	8	350		13595	1918	8	0		8	14172,041	24291,5	18229,8	9815,33	20640,6	2059,96	8797,48	0	9339,58	6092,45	9071,09	13167,8
11	4290	2721	9	307		4290	2721	9	0		9	15354,997	28071,1	23591,4	7762,08	17324,8	11202	11355,1	9339,58	0	14617,6	8118,22	20583,5
12	18684	5268	10	810		18684	5268	10	0		10	12643,246	19815,6	13164	11906,5	21033,4	4083,63	8243,14	6092,45	14617,6	0	10874,8	7251,61
13	8686	9546	11	1002		8686	9546	11	0		11	7255,1163	19997,9	16012,2	1058,73	11611,7	9638,58	3900	9071,09	8118,22	10874,8	0	14525,9
14	23140	10989	12	476		23140	10989	12	0		12	12215,143	13983,7	6972,39	15549,7	21435,7	11262,8	10715,8	13167,8	20583,5	7251,61	14525,9	0
15	5263	14289	13	674		5263	14289	13	0		13	6950,8769	19405,4	17454,8	5149,48	5953,92	15441,9	8043,36	14915,2	11608,8	16171	5849,19	18179
16	19557	17564	14	691		19557	17564	14	0		14	7717,1462	7574,85	2802,07	14264,1	16041,6	15389,6	9937,99	16743,4	21293,1	12327	13508	7487,89
17	23281	29684	15	675		23281	29684	15	0		15	17734,509	5116,16	11806,8	25358,6	21822,6	28065,5	21920,2	29407	32979,7	24845	24870,7	18695,5
18	374	22683	16	410		374	22683	16	0		16	13409,935	21263,7	22474,6	14859,7	4256,23	25040,7	17042	24616,7	20342,5	25269,3	15545,7	25593,8
19	18804	9082	17	351		18804	9082	17	0		17	9703,2541	16029,5	9513,36	11180,3	18652,3	7171,59	6561,17	8857,57	15846,7	3815,89	10128,6	4736,83
20	5444	25004	18	786		5444	25004	18	0		18	11137,915	16080,5	18320,1	15439,3	5262,91	24443,7	15974,5	24482,7	22312,9	23765,7	15794,3	22573,6
21	20390	25745	19	59		20390	25745	19	0		19	12859,126	1426,85	8068,44	20475,3	17632,8	23539,2	17075,8	24777	28094,7	20547,9	19984,8	15010,1
22	4310	21692	20	862		4310	21692	20	0		20	8567,9907	17506,5	18418,4	12425,1	1762,37	22021,3	13715,9	21845,4	18971	21825,7	12510,3	21659,2
23	8526	2495	21	550		8526	2495	21	0		21	13944,666	25884,2	20700,7	7280,86	18190,9	6969,67	9030,49	5101,73	4242,02	10529,7	7052,82	16903,2
24	9795	21510	22	0		9795	21510	22	549		22	5934,8231	12203,3	13047,1	11983	6263,9	19633,7	11186,8	19957,1	19578,9	18515,3	12015,3	16993,5
25	7758	10895	23	0		7758	10895	23	558		23	6637,4384	19622,8	16181,1	1178,72	9999,42	11253	4679,06	10707,8	8879,26	12289,9	1637,37	15382,3
26	12927	2464	24	0		12927	2464	24	851		24	13567,705	24007,1	18090,9	8978,3	19844	2577,84	8189,1	862,751	8640,82	6403,55	8254,74	13303,4
27	23286	5538	25	0		23286	5538	25	388		25	15599,192	19421,1	12413,3	16193,8	24362,5	6287,46	11994,3	10345	19203,7	4609,91	15140,1	5452,95
28	7416	15826	26	0		7416	15826	26	194		26	4587,9012	16762,8	15066,5	6108,18	5610,77	15593,6	7214,95	15218,8	13472,7	15441,5	6407,13	16451,2
29																							
30																							
31																							
32																							
33																							
34																							

Şekil 6.3. “Veri Düzenle” butonuna basıldığında “Veri” sayfasında elde edilen bilgiler

“Verileri Düzenle” butonuna basıldığında, bölge oluşturma aşamasında şehirler araçlara atanırken depo göz önüne alınmadığı, depo dışındaki bilgilerin program tarafından yazdırıldığı ve depo dışında müşteriler arasındaki uzaklıkları gösteren her defasında K2 hücresinden başlayarak yazdırılan uzaklık matrisini içeren “Bölge Veri” sayfası Şekil 6.4’de görülmektedir.

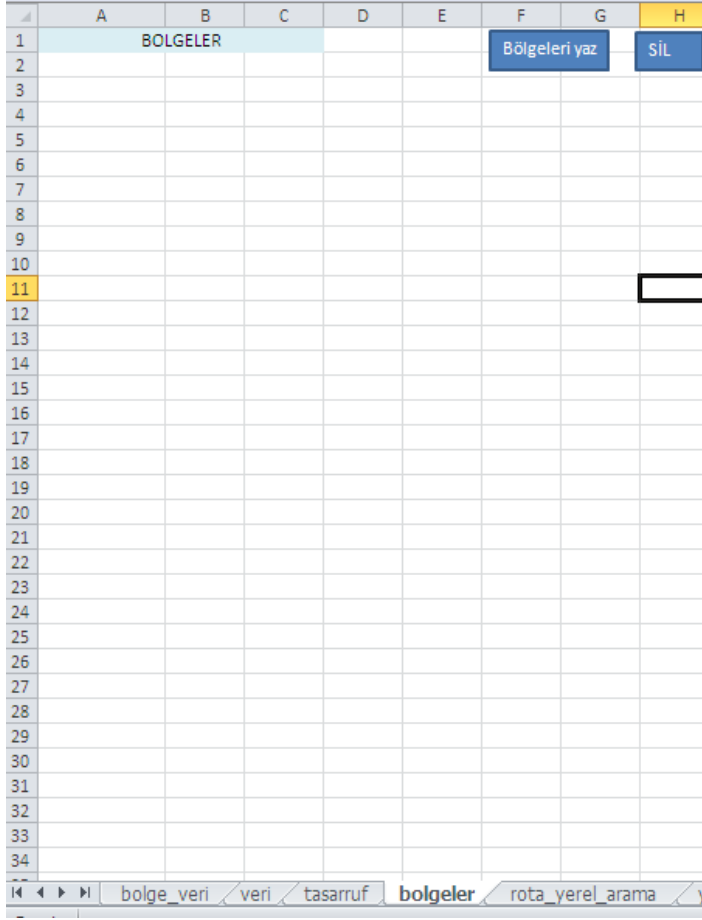
6.6’da GAMS yazılımından MS EXCEL programına verilerin nasıl yazdırıldığına ilişkin bir örnek görülmektedir.

	A	B	C	D	E	F	G	H	I	J	K
1		10	11	21							
2	1		1								
3	2		1								
4	3			1							
5	4			1							
6	5				1						
7	6				1						
8	7				1						
9	8				1						
10	9				1						
11	10		1								
12	11			1							
13	12			1							
14	13		1								
15	14		1								
16	15			1							
17	16				1						
18	17			1							
19	18		1								
20	19			1							
21	20				1						
22	21				1						
23	22				1						
24	23				1						
25	24				1						
26	25			1							

Şekil 6.6. GAMS yazılımında bölge oluşturma aşaması çözdürüldüğünde programa yazdırılan verilerin görünümü

c) “Bölgeler Sayfası”:

MS EXCEL’e farklı sayfalara yazdırılmış olan GAMS yazılımından elde edilen sonuçların hem toplu halde görülmesi hem de daha kolay anlaşılır hale gelmesi için “Bölgeler” sayfası hazırlanmıştır. Şekil 6.7’de bu sayfanın boş hali görülmektedir.



Şekil 6.7. “Bölgeler” sayfasının görüntüsü

“Bölgeleri Yaz” butonuna basıldığında ilk amaç fonksiyonuna göre elde edilen en iyi sonuçtan başlayarak, sırası ile ilk amaç fonksiyonunun en kötü değerine göre, ikinci amaç fonksiyonunun en iyi değerine ve en kötü değerine göre, min operatörüne göre, iki aşamalı yaklaşıma göre, ağırlıklı max-in modeline göre ve ağırlıklı toplam operatörüne göre elde edilen sonuçlar yazdırılmaktadır. Her bir araca atanan müşteriler sütun olarak görülmektedir. Ayrıca bu sayfanın D1 ve E1 sütunlarına GAMS yazılımından, hesaplanan amaçların ağırlıkları yazdırılır. Şekil 6.8’de “Bölgelere Yaz” butonuna basıldıktan sonraki “Bölgeler” sayfasının görünümü yer almaktadır.

	A	B	C	D	E	F	G	H
1	BÖLGELER			0,38435	0,61565	Bölgeleri yaz	SİL	
2	ilk amaç fonksiyonuna göre			en iyi				
3								
4		2	24	26				
5		3	6	4				
6		12	8	5				
7		14	9	7				
8		15	10	13				
9		19	11	16				
10			17	18				
11			21	20				
12			25	22				
13				23				
14	uzaklık	35112,8	50797,1	58537,8	144448			
15	saving	91030,3	159938	58630,8	309599			
16								
17								
18	ilk amaç fonksiyonuna göre			en kötü				
19								
20		15	16	25				
21		4	2	5				
22		7	3	13				
23		8	6	18				
24		9	10	19				
25		11	12	20				
26		21	14	22				
27		23	17					
28		24						
29		26						
30	uzaklık	240031	162402	137045	539477			
31	saving	8120,73	11011	10963,5	30095,2			
32								
33								
34								

Şekil 6.8. “Bölgeleri Yaz” butonuna basıldıktan sonra verilerin düzenlenmiş hali

6.2.2. Rotalama aşamasında kullanılan ara yüz

a) Bölge verilerinin düzenlenmesi:

Rotalama aşamasında her bir araç için matematiksel model çözdürülerek rotalar elde edilir. Bu aşamada kullanılan matematiksel model için gerekli olan uzaklık parametrelerin hesaplanabilmesi, GAMS yazılımının her bir araç için bu parametreleri okuyabilecek hale getirilmesi için “Bölgeler” sayfasına I4 hücresinden başlanılarak bölgeler yazılır. Şekil 6.9’da “Bölgeler” sayfasındaki bu kısım görülmektedir.

tarafından yazdırılır. Şekil 6.11’de “Bölge Verilerini Düzenle” butonuna basıldıktan sonraki oluşturulan sayfalardan bir örnek görülmektedir.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	KOORDİNATLAR		MÜŞTERİLER			TEDARİKÇİLER													
2	X	Y	NO	TALEP		NO	TALEP												
3	12000	16000	1	0		1	0		1			1	0	13020,88	10512,89	12215,14	7717,146	17734,51	12859,13
4	21524	24879	2	483		2	0		2			2	13020,88	0	7011,296	13983,69	7574,847	5116,158	1426,854
5	22337	17915	3	389		3	0		3			3	10512,89	7011,296	0	6972,394	2802,071	11806,8	8068,439
6	23140	10989	4	476		4	0		12			4	12215,14	13983,69	6972,394	0	7487,891	18695,53	15010,06
7	19557	17564	5	691		5	0		14			5	7717,146	7574,847	2802,071	7487,891	0	12679,22	8223,299
8	23281	29684	6	675		6	0		15			6	17734,51	5116,158	11806,8	18695,53	12679,22	0	4886,062
9	20390	25745	7	59		7	0		19			7	12859,13	1426,854	8068,439	15010,06	8223,299	4886,062	0

Şekil 6.11. “Bölge Verilerini Düzenle” butonuna basıldıktan sonra elde edilen verilerden bir örnek

b) GAMS yazılımının her bir araç için çalıştırılması:

Program kaydedilip kapatıldıktan sonra rotalama aşamasında kullanılan matematiksel modelin kodlandığı GAMS yazılımı her bir araç için çalıştırılır. GAMS yazılımı bulunduğu rotaları Şekil 6.12’de görüldüğü gibi MS EXCEL programına her bir araç için yeni bir sayfa ekleyerek yazdırır.

	A	B	C	D	E	F	G	H	I	J
1		1	2	3	4	5	6	7		
2	1								1	
3	2			1						
4	3				1					
5	4					1				
6	5	1								
7	6		1							
8	7							1		

Şekil 6.12. Rotalama aşamasından sonra GAMS yazılımı sonuçlarının yazdırılması

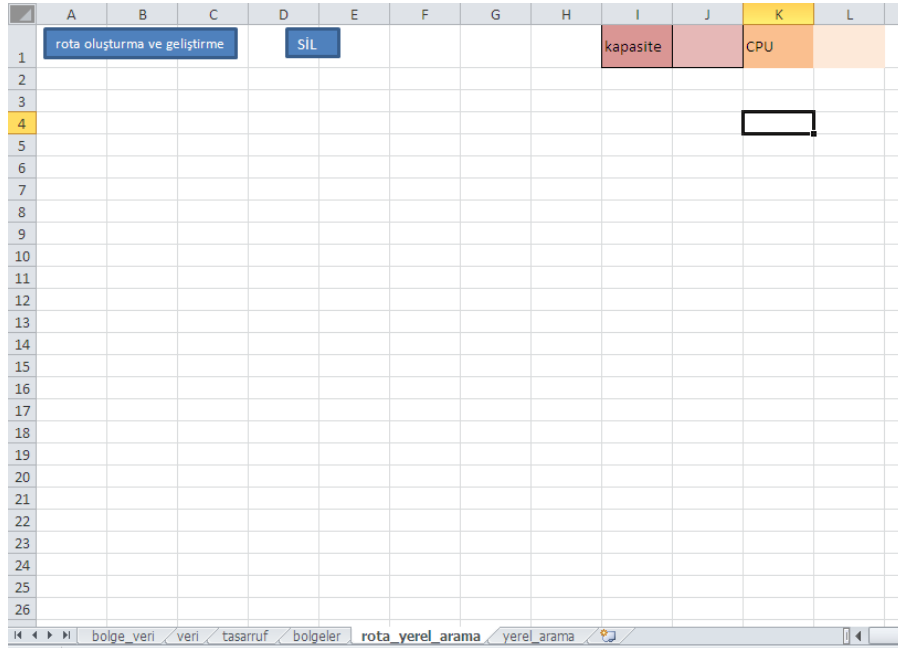
c) Rotaların bütün haline getirilmesi:

“Rota Yerel Arama” sayfası kullanılarak GAMS yazılımından her bir araç için ayrı sayfalara yazdırılan sonuçlar hem bir bütün haline getirilir hem de yerel arama algoritması çalıştırılır.

6.2.3. Yerel aramada kullanılan ara yüz

a) Rota yerel arama sayfası:

Önerilen modelin yerel arama algoritması her iki aşama ile elde edilen rotaların iyileştirilmesinde kullanılması için MS EXCEL VBA kullanılarak kodlanmıştır. Bunun için “Rota Yerel Arama” sayfası oluşturulmuştur. Bu sayfaya sadece J1 hücresine araç kapasite bilgisinin girilmesi yeterlidir. Şekil 6.13’de sayfanın boş hali görülmektedir.



Şekil 6.13. “Rota Yerel Arama” sayfasının görünümü

“Rota Oluşturma ve Geliştirme” butonuna basıldığında, ilk öncelikle her iki aşama sonucunda elde edilen rota yazdırılır, daha sonra yerel arama çalıştırılarak elde edilen yeni rota yazdırılır. Ayrıca yerel arama için harcanan CPU zamanı L1

hücrelerine yazdırılır. Şekil 6.14’de “Rota Oluşturma ve Geliştirme” butonuna basıldıktan sonra “Rota Yerel Arama” sayfası görülmektedir.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	rota oluşturma ve geliştirme			SİL					kapasite	4050	CPU	0,0475	
2	52050,07	1	19	15	2	3	12	14	1				
3	64763,62	1	11	9	21	8	6	10	17	25	24	1	
4	56076,98	1	7	4	13	5	20	16	18	22	26	23	1
5													
6	172890,7												
7													
8													
9	52050,08	1	19	15	2	3	12	14	1				
10	66124,61	1	7	9	21	8	6	10	17	25	24	1	
11	54136,79	1	11	4	13	5	20	16	18	22	26	23	1
12													
13	172311,5												
14													
15													
16													
17													
18													
19													
20													
21													
22													
23													
24													
25													
26													

Şekil 6.14.“Rota Yerel Arama” sayfasının “Rota Oluşturma ve Geliştirme” butonuna basıldıktan sonraki görünümü

b) Yerel arama sayfası:

GAMS çözümü olmayan bir rota üzerinde yerel arama yapılabilmesi için “Yerel Arama” sayfası oluşturulmuştur. Bu sayfanın görünümü “Rota Yerel Arama” sayfası gibidir. Şekil 6.15’de bu sayfaya veri girildikten sonraki görünümü yer almaktadır.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	rota geliştirme			SİL					kapasite	4050	CPU		
2	52050,07	1	19	15	2	3	12	14	1				
3	64763,62	1	11	9	21	8	6	10	17	25	24	1	
4	56076,98	1	7	4	13	5	20	16	18	22	26	23	1
5													
6	172890,7												
7													
8													
9													
10													
11													
12													
13													
14													
15													
16													
17													
18													
19													
20													
21													
22													
23													
24													
25													
26													

Şekil 6.15. “Yerel Arama” sayfasının veri girişi sonrası görünüşü

J1 hücresine araç kapasitesi girildikten ve “Rota Geliştirme” butonuna basıldıktan sonra yerel arama ile elde edilen yeni rotanın da görüldüğü sayfa Şekil 6.17’de yer almaktadır. Yerel arama için gerekli olan CPU zamanı L1 hücresine yazdırılır.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	rota geliştirme		SİL						kapasite	4050	CPU	0,0325	
2	52050,07	1	19	15	2	3	12	14	1				
3	64763,62	1	11	9	21	8	6	10	17	25	24	1	
4	56076,98	1	7	4	13	5	20	16	18	22	26	23	1
5													
6	172890,7												
7													
8													
9	52050,08	1	19	15	2	3	12	14	1				
10	66124,61	1	7	9	21	8	6	10	17	25	24	1	
11	54136,79	1	11	4	13	5	20	16	18	22	26	23	1
12													
13	172311,5												
14													
15													
16													
17													
18													
19													
20													
21													
22													
23													
24													
25													
26													

Şekil 6.16. “Yerel Arama” sayfasının “Rota Geliştirme” butonuna basıldıktan sonraki görünüşü

7. SERAMİK SAĞLIK GEREÇLERİ ÜRETİMİ YAPAN BİR İŞLETMEDE ÖNERİLEN YÖNTEMİN UYGULAMASI

ÖDST-ARP için önerilen yöntem seramik sağlık gereçleri üretimi yapan bir işletme için uygulanmıştır. Bu işletmede dağıtım yapılacak noktalar işletmenin ürünlerini sevk ettiği müşterileri, toplama yapılacak olan noktalar ise işletmenin ürün aldığı tedarikçileridir. Her hafta başında müşterilerden gelen talepler doğrultusunda araçların rotaları oluşturulmaktadır. Tedarikçiler ise gerekli miktardaki ürünü işletmeye kendi araçları ile göndermektedir. İşletmede lojistik ile ilgili birimde yer alan araçların rotalarının oluşturulmasından sorumlu endüstri mühendisi bu iki süreci birleştirerek hem zaman hem de maliyet açısından kazanmak istemektedir. Maliyetler açısından işletmeden herhangi bir bilgi alınamamıştır. Ancak bu tür bir sisteme geçildiğinde maliyetlerde %30 oranında azalmanın olacağını hesapladıklarını belirtmişlerdir.

Önerilen yöntem işletmenin bir haftalık rotalama problemi için uygulanmıştır. müşterilerin talepleri ve tedarikçilerin talepleri bir aracın kapasitesini aşabilmektedir. Literatürde taleplerin kapasiteyi aştığı bir problem ve dolayısı ile bir çözüm yöntemine rastlanamamıştır. Talebin kapasiteyi aştığı noktalar, araç kapasitesine erişinceye kadar yapay noktalara ayrılmışlardır. Aynı noktaya ait yapay noktalar arasındaki uzaklık sıfır olarak ele alınmıştır. Örneğin müşterinin talebi 50birim , araç kapasitesi 22birim iken bu müşteri sırası ile 22, 22 ve 6 birim talepleri olan üç yapay müşteri noktasına ayrılır ve bu üç nokta arasındaki mesafe 0 olarak uzaklık matrisine yazılır.

7.1. İşletme Hakkında Genel Bilgi

Seramik sağlık gereçleri üreten işletmenin yurt içinde ve yurt dışında müşterileri ve tedarikçileri bulunmaktadır. Ancak ürünlerin sevkiyatından sorumlu mühendis, yurt içindeki müşteriler ve tedarikçiler için araç rotalama işlemini birleştirmek istemektedir. Bu sebeple bu çalışma yurt içi verileri ile yapılmıştır.

İşletmenin Türkiye genelinde 73 ilde müşterisi ve 17 ilde tedarikçisi bulunmaktadır. İllerdeki müşteri ve tedarikçi sayısı çeşitlilik göstermektedir.

İşletme her hafta her müşterisine ürün göndermediği gibi her tedarikçisinden de ürün almamaktadır. Dolayısı ile toplamda 530 müşterisi yani dağıtım noktası, 108 tedarikçisi yani toplama noktası var iken haftalık bazda bu sayılar azalmasına rağmen geçmiş verilere göre en az 60 müşteriye en fazla 184 müşteriye sevkiyat yapılmış, en az 7 tedarikçiden ürün gelirken en fazla 40 tedarikçiden ürün gelmiştir. Hem müşteri hem de tedarikçi bir bütün olarak bakıldığında toplamda haftalık en az 67 en fazla 212 nokta olmaktadır. İşletmenin toplam müşteri ve tedarikçi sayıları il bazında Çizelge 7.1’de verilmiştir.

Çizelge 7.1. İşletmenin müşteri ve tedarikçi bilgileri

Şehir	Müşteri Sayısı	Tedarikçi Sayısı
Adana	15	1
Adıyaman	5	0
Afyon	8	0
Aksaray	7	0
Amasya	5	0
Ankara	27	1
Antalya	19	1
Artvin	1	0
Aydın	8	1
Balıkesir	14	1
Batman	4	0
Bayburt	1	0
Bilecik	8	6
Bitlis	1	0
Bolu	1	1
Burdur	2	0
Bursa	11	9
Çanakkale	7	0
Çankırı	5	0
Çorum	4	0
Denizli	8	0
Diyarbakır	6	0
Düzce	2	0
Edirne	7	0
Elazığ	1	0

Çizelge 7.2. (Devam) İşletmenin müşteri ve tedarikçi bilgileri

Şehir	Müşteri Sayısı	Tedarikçi Sayısı
Erzincan	3	0
Erzurum	5	0
Eskişehir	11	16
Gaziantep	9	1
Giresun	7	0
Gümüşhane	1	0
Hatay	0	0
Isparta	3	0
İçel	10	0
İstanbul	58	55
İzmir	24	5
Kahramanmaraş	6	0
Karabük	3	0
Karaman	5	0
Kars	1	0
Kastamonu	3	0
Kayseri	14	0
Kırıkkale	2	0
Kırklareli	5	0
Kırşehir	1	0
Kocaeli	18	6
Konya	12	0
Kütahya	6	1
Malatya	6	0
Manisa	6	1
Mardin	1	0
Muğla	25	0
Muş	5	0
Nevşehir	6	0
Niğde	3	0
Ordu	8	0
Osmaniye	3	0
Rize	7	0
Sakarya	6	0
Samsun	12	1

Çizelge 7.2. (Devam) İşletmenin müşteri ve tedarikçi bilgileri

Şehir	Müşteri Sayısı	Tedarikçi Sayısı
Siirt	1	0
Sinop	5	0
Sivas	5	0
Şanlıurfa	8	0
Şırnak	2	0
Tekirdağ	9	0
Tokat	8	0
Trabzon	6	0
Uşak	2	0
Van	2	0
Yalova	4	1
Yozgat	2	0
Zonguldak	4	0

7.2. İşletmenin Bir Haftalık Rotalama Problemi

İşletmenin bir haftalık rotalama problemi önerilen yöntem ile çözülmüştür. Gerekli olan uzaklık parametreleri T.C. Karayolları Genel Müdürlüğü'nün internet sitesinden elde edilmiştir. Bu veriler şehirlere ait olup şehirlerin ilçelerini kapsamamaktadır. Aynı şehir içinde iki nokta arasındaki mesafe 0 olarak kabul edilmiştir. Ayrıca daha önce de açıklandığı gibi talebin kapasiteyi aştığı noktalar, araç kapasitesine erişinceye kadar yapay noktalara ayrılmışlardır. Aynı noktaya ait yapay noktalar arasındaki uzaklık sıfır olarak ele alınmıştır.

Önerilen yöntem uygulanmak üzere işletmeden bir haftalık veri elde edilmiştir. İşletmeden alınan veriler;

- Müşterilerin ve tedarikçilerin bulunduğu şehirler
- Müşterilerin ve tedarikçilerin talepleri
- Araç kapasitesidir.

Bir haftalık veri müşteriler için çizelge 7.2'de, tedarikçiler için çizelge 7.3'te verilmiştir. Bu çizelgelerdeki müşteri ve tedarikçi kodları bilgi güvenliği açısından değiştirilmiş, müşteriler için sırası ile M1, M2 vb olarak, tedarikçiler için T1, T2 vb. olarak isimlendirilmiştir. Müşterilerin ve tedarikçilerin talepleri

palet cinsindendir ve işletmede ürünlerin taşınmasında kullanılan araçların kapasitesi 22 palettir.

Gerekli verilerin düzenlenmesi, matematiksel modellerde gerekli olan parametrelerin elde edilmesi ve yerel arama algoritması için MS EXCEL programında VBA uygulaması kullanılmıştır. Test problemlerinde olduğu gibi matematiksel modellerin çözümüne GAMS yazılımı ve CPLEX 9.0 çözücüsü kullanılmıştır. GAMS yazılımından elde edilen verilerin düzenlenmesinde de MS EXCEL VBA ortamında kodlanan programdan yararlanılmıştır.

Çizelge 7.3. İşletmenin bir haftalık müşteri verisi

Şehir	Müşteri Kodu	Talep
Adana	M1	1
Adana	M2	110
Adana	M3	46
Afyon	M4	5
Afyon	M5	9
Ankara	M6	9
Ankara	M7	5
Ankara	M8	1
Ankara	M9	4
Ankara	M10	1
Ankara	M11	2
Ankara	M12	68
Ankara	M13	1
Ankara	M14	41
Antalya	M15	8
Antalya	M16	21
Antalya	M17	3
Antalya	M18	21
Antalya	M19	1
Aydın	M20	3
Balıkesir	M21	2
Bursa	M22	4
Bursa	M23	41
Bursa	M24	30
Bursa	M25	21

Çizelge 7.4. (Devam) İşletmenin bir haftalık müşteri verisi

Şehir	Müşteri Kodu	Talep
Bursa	M26	4
Çanakkale	M27	10
Diyarbakır	M28	40
Diyarbakır	M29	48
Edirne	M30	19
Edirne	M31	1
Edirne	M32	21
Elazığ	M33	4
Gaziantep	M34	52
Giresun	M35	4
Giresun	M36	1
Giresun	M37	4
Hatay	M38	22
İçel	M39	52
İçel	M40	1
İçel	M41	1
İstanbul	M42	20
İstanbul	M43	23
İstanbul	M44	1
İstanbul	M45	36
İstanbul	M46	1
İstanbul	M47	45
İstanbul	M48	220
İstanbul	M49	4
İstanbul	M50	117
İstanbul	M51	3
İstanbul	M52	65
İstanbul	M53	8
İstanbul	M54	1
İzmir	M55	25
İzmir	M56	14
İzmir	M57	13
İzmir	M58	2
İzmir	M59	20
İzmir	M60	11

Çizelge 7.4. (Devam) İşletmenin bir haftalık müşteri verisi

Şehir	Müşteri Kodu	Talep
Kahramanmaraş	M61	14
Kayseri	M62	3
Kayseri	M63	59
Kayseri	M64	4
Kocaeli	M65	4
Kocaeli	M66	21
Konya	M67	6
Konya	M68	9
Konya	M69	6
Kütahya	M70	5
Manisa	M71	2
Muğla	M72	1
Muğla	M73	2
Muğla	M74	2
Muğla	M75	2
Ordu	M76	2
Ordu	M77	2
Ordu	M78	3
Ordu	M79	12
Ordu	M80	12
Rize	M81	1
Sakarya	M82	1
Samsun	M83	48
Samsun	M84	37
Samsun	M85	16
Siirt	M86	4
Tokat	M87	14
Tokat	M88	5
Tokat	M89	3
Trabzon	M90	8
Trabzon	M91	1
Trabzon	M92	3
Trabzon	M93	17

Çizelge 7.5. İşletmenin bir haftalık tedarikçi verisi

Şehir	Tedarikçi Kodu	Talep
Adana	T1	7,8
Bursa	T2	18,5
Bursa	T3	630,5
Bursa-İnegöl	T4	101,2
Bursa-İnegöl	T5	792,8
Eskişehir	T6	61,5
Eskişehir	T7	3
Eskişehir	T8	3,6
Eskişehir	T9	61
Eskişehir	T10	69,5
Eskişehir	T11	154
Gaziantep-Başpınar	T12	5
İstanbul-Ayazağa	T13	1
İstanbul-B.Çekmece	T14	6,9
İstanbul-B.Çekmece	T15	1
İstanbul-Beykoz	T16	2
İstanbul-K.Çekmece	T17	11,2
İstanbul-Kartal	T18	5,7
İstanbul-Maltepe	T19	2,6
İstanbul-Pendik	T20	3,1
İstanbul-Pendik	T21	10
İstanbul-Pendik	T22	0,1
İstanbul-Tuzla	T23	5
İstanbul-Tuzla	T24	145,8
İstanbul-Tuzla	T25	77,3
İzmir	T26	2
Samsun-Kutlukent	T27	6,6

Müşterilerin ve tedarikçilerin talepleri ve şehirler arası mesafeler kullanılarak önerilen yöntem ile işletmenin bir haftalık verisi çözülmüştür. Bölge oluşturma aşaması ile hem birinci ve ikinci amaç fonksiyonunu göre hem de bulanık çözüm yaklaşımları olan min operatörü, iki aşamalı yaklaşım, ağırlıklı max-min modeli ve ağırlıklı toplam operatörüne göre şehirler araçlara atanmıştır.

Bölge oluşturma aşamasında elde edilen her bir çözüm, rotalama aşamasında her bir araç için çözülmüştür. Son olarak yerel arama ile rotalar iyileştirilmiştir.

Problemin hem birinci ve ikinci amaç fonksiyonuna göre hem de bulanık çözüm yöntemlerine göre elde edilen ayrıntılı çözümleri EK-6'da sırası ile verilmiştir. Özet sonuçlar ve CPU zamanları Çizelge 7.4'de verilmiştir. Amaçlara atanan ağırlıklar ise sırası ile 0,94 ve 0,06'dır. En iyi sonuç birinci amaç fonksiyonuna göre 93553km değeri ile elde edilmiştir.

Çizelge 7.6. Uygulama sonuçları

Çözüm yöntemi	Sonuç (km)	CPU-1	CPU-2	CPU-3	Toplam CPU
1. Amaç fonksiyonu	93552	1000	3,89	3,47	1007,36
2. amaç fonksiyonu	93719	1000	3,97	2,05	1006,02
Min operatörü	99410	3000	4,37	1,92	3006,29
İki aşamalı yaklaşım	99410	3053,687	4,37	1,92	3059,98
Ağırlıklı max-min modeli	99410	2052,046	4,37	1,92	2058,34
Ağırlıklı toplam operatörü	94154	3000	4,6	1,32	3005,92

CPU-1, bölge oluşturma aşaması için harcanan zamanı, CPU-2 rotalama aşaması için, CPU-3 yerel arama için harcanan zamanı gösterirken toplam CPU bu 3 değer toplamıdır. Toplam CPU zamanına bakıldığında birinci amaç fonksiyonu ile ikinci amaç fonksiyonunki birbirine çok yakındır ve diğerlerine göre daha iyidir. Bulanık çözüm yaklaşımlarının CPU zamanının yüksek olmasının sebebi amaçların en kötü değerlerinin bulunması için gereken süreyi de içermesidir. Sürelerin büyük bir kısmı bölge oluşturma aşaması için harcanmıştır. Rotalama aşaması ve yerel arama için harcanan süre oldukça azdır.

7.3. Uygulama İçin Geliştirilen Ara Yüz

Uygulama problemi için Excel VBA ortamında kodlanan ara yüz test problemlerinde olduğu gibi birkaç sayfadan oluşmaktadır. Uzaklıklar, müşterilerin koordinatlarına göre değil, şehirlerin arasındaki karayolları uzunluğuna göre hesaplandığından ve talepleri araç kapasitesini aşan noktalar için yapay noktaların oluşturulması gerektiğinden dolayı uygulama problemi için yeniden ara yüz tasarlanmasına gerek duyulmuştur. “Ana Veri” sayfası Türkiye’nin tüm şehirleri arasındaki uzaklık matrisini içermektedir. A sütununda il plakası, B sütununda ise şehir ismi yazmaktadır. Bu sayfada yer alan veriler sabit olup, diğer sayfalarda gerekli olan parametrelerin elde edilmesinde kullanılmaktadır. Şekil 7.1’de “Ana Veri” sayfası görülmektedir.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE
1	Il	plaka kodu	il Adı		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
2		1	Adana		1	329	573	961	613	490	558	1030	893	894	768	628	727	677	671	837	1101	576	575	767	518	1169	486	675	804	688	
3		2	Adıyaman		2	329		902	648	636	756	887	755	1222	1223	1059	349	414	947	1000	1128	1399	785	700	1096	205	1439	283	548	529	979
4		3	Afyon		3	573	902		1308	591	256	292	1238	351	321	210	1102	1295	423	170	273	528	387	499	225	1091	690	960	939	1128	144
5		4	Ağrı		4	961	648	1308		734	1052	1429	396	1651	1567	1359	359	234	1143	1429	1416	1687	982	826	1525	443	1635	497	369	183	1285
6		5	Amasya		5	613	636	591	734		335	875	694	938	833	625	636	829	409	757	682	953	248	92	812	702	901	549	365	554	568
7		6	Ankara		6	490	756	256	1052	335		544	982	603	530	313	900	1093	191	422	382	653	131	243	477	911	683	758	683	872	233
8		7	Antalya		7	558	887	292	1429	875	544		1467	344	510	474	1186	1285	690	122	537	717	675	783	222	1076	919	1044	1060	1249	424
9		8	Artvin		8	1030	755	1238	396	694	982	1467		1585	1463	1255	406	543	1039	1404	1312	1583	895	739	1459	550	1531	544	407	226	1215
10		9	Aydın		9	893	1222	351	1651	938	603	344	1585		293	523	1422	1615	715	272	442	449	734	846	126	1411	658	1280	1282	1471	483
11		10	Balıkesir		10	894	1223	321	1567	833	530	510	1463	293		246	1420	1613	272	397	151	207	659	776	288	1412	411	1278	1198	1387	495
12		11	Bilecik		11	768	1059	210	1359	625	313	474	1255	523	246		1203	1396	216	352	95	366	444	556	397	1214	480	1061	990	1179	80
13		12	Bingöl		12	628	349	1102	359	636	900	1186	406	1422	1420	1203		197	1045	1200	1272	1543	884	728	1296	144	1537	142	271	180	1123
14		13	Bitlis		13	727	414	1295	234	829	1093	1285	543	1615	1613	1396	197		1238	1393	1465	1736	1077	921	1489	209	1730	335	464	330	1316
15		14	Bolu		14	677	947	423	1143	409	191	690	1039	715	424	216	1045	1238		568	273	544	235	352	613	1102	492	949	774	963	296
16		15	Burdur		15	671	1000	170	1429	757	422	122	1404	272	397	352	1200	1393	568		415	604	553	665	150	1189	806	1058	1060	1249	302
17		16	Bursa		16	837	1128	273	1416	682	382	537	1312	442	151	95	1272	1465	273	415		271	508	625	437	1283	420	1130	1047	1236	149
18		17	Çanakkale		17	1101	1399	528	1687	953	653	717	1583	449	207	366	1543	1736	544	604	271		779	896	495	1554	217	1401	1318	1507	420
19		18	Çankırı		18	576	785	387	982	248	131	675	895	734	659	444	884	1077	235	553	508	779		156	608	922	727	769	613	802	364
20		19	Çorum		19	575	700	499	826	92	243	783	739	846	773	556	728	921	352	665	625	896	156		720	766	844	613	457	646	476
21		20	Denizli		20	767	1096	225	1525	812	477	222	1459	126	288	397	1296	1489	613	150	437	495	608	720		1285	697	1154	1156	1345	357
22		21	Dişyarbakır		21	518	205	1091	443	702	911	1076	550	1411	1412	1214	144	209	1102	1189	1283	1554	922	766	1285		1594	153	406	324	1134
23		22	Edirne		22	1169	1439	690	1635	901	683	919	1531	658	411	480	1537	1730	492	806	420	217	727	844	697	1594		1441	1266	1455	560
24		23	Elazığ		23	486	283	960	497	549	758	1044	544	1280	1278	1061	142	335	949	1058	1130	1401	769	613	1154	153	1441		265	318	981
25		24	Erzincan		24	675	548	939	369	365	683	1060	407	1282	1198	990	271	464	774	1060	1047	1318	613	457	1156	406	1266	265		189	916
26		25	Erzurum		25	804	529	1128	183	554	872	1249	226	1471	1387	1179	180	330	963	1249	1236	1507	802	646	1345	324	1455	318	189		1105
27		26	Eskişehir		26	688	979	144	1285	568	233	424	1215	483	297	80	1123	1316	296	302	149	420	364	476	357	1134	560	981	916	1105	
28		27	Gaziantep		27	205	150	778	756	609	672	763	863	1098	1099	973	457	522	863	876	1042	1306	701	630	972	313	1355	345	610	637	893
29		28	Giresun		28	727	712	868	545	324	612	1112	370	1215	1093	885	545	692	669	1034	942	1213	525	369	1089	669	1161	558	293	365	845
30		29	Gümüşhane		29	786	679	1009	383	435	753	1171	333	1356	1255	1047	383	530	831	1171	1104	1375	683	527	1230	527	1323	396	131	203	986
31		30	Hakkari		30	900	667	1473	434	1141	1367	1458	770	1793	1794	1668	509	340	1550	1571	1737	2001	1389	1233	1667	474	2042	627	776	617	1588
32		31	Hatay		31	191	320	764	952	705	681	749	1021	1084	1085	959	619	718	868	862	1028	1292	767	726	958	509	1360	477	742	795	879
33		32	İsparta		32	620	949	169	1378	756	421	130	1403	293	396	351	1149	1342	567	51	414	603	552	664	167	1138	805	1007	1009	1198	301
34		33	Mersin		33	69	398	565	1030	639	483	489	1099	833	886	760	697	796	670	611	829	1093	569	568	711	587	1162	555	744	873	680
35		34	İstanbul		34	939	1209	460	1405	671	453	724	1301	681	390	250	1307	1500	262	602	243	320	497	614	647	1364	230	1211	1036	1225	330
36		35	İzmir		35	900	1229	327	1631	914	579	446	1561	126	173	417	1429	1622	595	374	322	325	710	822	224	1418	534	1287	1262	1451	412
37		36	Kars		36	1007	732	1328	216	754	1072	1449	209	1671	1587	1379	383	413	1163	1449	1436	1707	1002	846	1545	527	1655	521	389	203	1305
38		37	Kastamonu		37	690	888	501	986	252	245	789	864	848	670	462	888	1081	246	667	519	790	114	195	722	954	738	801	617	806	478
39		38	Kayseri		38	333	437	521	810	348	319	619	848	841	839	622	581	774	510	619	691	962	348	277	715	592	1002	439	441	630	542
40		39	Kırklareli		39	1150	1420	671	1616	882	664	935	1512	675	428	461	1518	1711	473	813	437	234	708	825	714	1575	62	1422	1247	1436	541
41		40	Kırşehir		40	375	571	431	940	308	185	573	952	778	705	488	715	908	376	573	557	828	214	216	652	726	868	573	571	760	408
42		41	Kocaeli		41	828	1098	349	1294	560	342	613	1190	570	279	139	1196	1389	151	491	132	399	386	503	536	1253	341	1100	925	1114	219
43		42	Konya		42	356	685	223	1114	560	258	323	1152	541	544	418	885	1078	445	315	487	751	389	468	415	874	898	743	745	934	338
44		43	Kütahya		43	673	1002	100	1363	646	311	364	1293	413	221	110	1201	1394	326	242	173	428	442	554	287	1191	590	1059	994	1183	78
45		44	Malatya		44	388	185	862	595	469	660	946	642	1182	1180	963	240	433	851												

7.3.1. Bölge oluşturma aşamasında kullanılan ara yüz

a) “Veri Girişi” sayfası:

“Veri Girişi” sayfasına müşterilere ve tedarikçilere ilişkin şehir, şehir plaka kodu, müşteri no veya kodu ve talep miktarları ilgili sütunların altına girilir. No ile belirtilen kısım GAMS yazılımının verileri doğru okuması için gerekli olup, bu kısma numaralar program tarafından atanır. N1 hücrelerine deponun bulunduğu şehri plaka numarası ve P1 hücrelerine araç kapasitesi girilir. Şekil 7.2’de “Veri Girişi” sayfasının verilerin yazılmış görüntüsü yer almaktadır.

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Plaka no	İl Adı	Müşteri no	No	Talep		Plaka no	İl Adı	Tedarikçi no	No	Talep		depo plaka no		11 araç kapasitesi		22
2	1	ADANA	M1		1		1	ADANA	T1		7,8						
3	1	ADANA	M2		110		16	BURSA	T2		18,5						
4	1	ADANA	M3		46		16	BURSA	T3		630,5						
5	3	AFYON	M4		5		16	BURSA-İNEGÖL	T4		101,2						
6	3	AFYON	M5		9		16	BURSA-İNEGÖL	T5		792,8						
7	6	ANKARA	M6		9		26	ESKİŞEHİR	T6		61,5						
8	6	ANKARA	M7		5		26	ESKİŞEHİR	T7		3						
9	6	ANKARA	M8		1		26	ESKİŞEHİR	T8		3,6						
10	6	ANKARA	M9		4		26	ESKİŞEHİR	T9		61						
11	6	ANKARA	M10		1		26	ESKİŞEHİR	T10		69,5						
12	6	ANKARA	M11		2		26	ESKİŞEHİR	T11		154						
13	6	ANKARA	M12		68		27	GAZİANTEP-BASPINAR	T12		5						
14	6	ANKARA	M13		1		34	İSTANBUL-AYAZAĞA	T13		1						
15	6	ANKARA	M14		41		34	İSTANBUL-B.ÇEKMECE	T14		6,9						
16	7	ANTALYA	M15		8		34	İSTANBUL-B.ÇEKMECE	T15		1						
17	7	ANTALYA	M16		21		34	İSTANBUL-BEYKOZ	T16		2						
18	7	ANTALYA	M17		3		34	İSTANBUL-K.ÇEKMECE	T17		11,2						
19	7	ANTALYA	M18		21		34	İSTANBUL-KARTAL	T18		5,7						
20	7	ANTALYA	M19		1		34	İSTANBUL-MALTEPE	T19		2,6						
21	9	AYDIN	M20		3		34	İSTANBUL-PENDİK	T20		3,1						
22	10	BAKİRKESİR	M21		2		34	İSTANBUL-PENDİK	T21		10						
23	16	BURSA	M22		4		34	İSTANBUL-PENDİK	T22		0,1						
24	16	BURSA	M23		41		34	İSTANBUL-TUZLA	T23		5						
25	16	BURSA	M24		30		34	İSTANBUL-TUZLA	T24		145,8						
26	16	BURSA	M25		21		34	İSTANBUL-TUZLA	T25		77,3						
27	16	BURSA	M26		4		35	İZMİR	T26		2						
28	17	ÇANAKKALE	M27		10		55	SAMSUN-KUTLUKENT	T27		6,8						
29	21	DIYARBAKIR	M28		40												
30	21	DIYARBAKIR	M29		48												
31	22	EDİRNE	M30		19												
32	22	EDİRNE	M31		1												
33	22	EDİRNE	M32		21												
34	23	ELAZIĞ	M33		4												
35	27	GAZİANTEP	M34		52												
36	28	ĞİRESUN	M35		4												
37	28	ĞİRESUN	M36		1												
38	28	ĞİRESUN	M37		4												
39	31	HATAY	M38		22												
40	33	İÇEL	M39		52												
41	33	İÇEL	M40		1												
42	33	İÇEL	M41		1												
43	34	İSTANBUL	M42		20												
44	34	İSTANBUL	M43		23												
45	34	İSTANBUL	M44		1												
46	34	İSTANBUL	M45		36												
47	34	İSTANBUL	M46		1												
48	34	İSTANBUL	M47		45												
49	34	İSTANBUL	M48		220												
50	34	İSTANBUL	M49		4												
51	34	İSTANBUL	M50		117												
52	34	İSTANBUL	M51		3												
53	34	İSTANBUL	M52		65												
54	34	İSTANBUL	M53		8												
55	34	İSTANBUL	M54		1												
56	35	İZMİR	M55		25												
57	35	İZMİR	M56		14												
58	35	İZMİR	M57		13												
59	35	İZMİR	M58		2												
60	35	İZMİR	M59		20												
61	35	İZMİR	M60		11												
62	46	KAHRAMANMARAŞ	M61		14												
63	38	KAYSERİ	M62		3												
64	38	KAYSERİ	M63		59												
65	38	KAYSERİ	M64		4												
66	38	KOCAELİ	M65		4												

Şekil 7.2. "Veri Girişi" sayfasının gerekli bilgiler girildikten sonraki görünümü

“Verileri Düzenle” butonuna basıldığında, öncelikle araç kapasitesinden talepleri fazla olan noktaların talepleri araç kapasitesinden az oluncaya kadar alt noktalara bölünür. Örneğin talebi 45 olan bir nokta, i araç kapasitesi 22 olduğu için 22-22-1 şeklinde talep miktarları olan alt noktalara bölünür. Bu noktalar arasındaki mesafe uzaklık matrisinde 0 olarak hesaplanır. P2 hücresine gerekli olan araç sayısı yazdırılır. “Uzaklık” ve “Tasarruf” sayfalarına uzaklık matrisi ve tasarruf miktarı matrisi yazdırılır. “Veri Girişi” sayfasının “Verileri Düzenle” butonuna basıldıktan sonraki görüntüsü Şekil 7.3’te yer almaktadır.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Plaka no	il Adı	Müşteri no	No	Talep		Plaka no	il Adı	Tedarikçi no	No	Talep		depo plaka no	11	araç kapasitesi	22
2	1	ADANA	M1		1		1	ADANA	M1		1	0				100
3	1	ADANA	M2	2	22		1	ADANA	M2		2	0				
4	1	ADANA	M2	3	22		1	ADANA	M2		3	0				
5	1	ADANA	M2	4	22		1	ADANA	M2		4	0				
6	1	ADANA	M2	5	22		1	ADANA	M2		5	0				
7	1	ADANA	M2	6	22		1	ADANA	M2		6	0				
8	1	ADANA	M3	7	2		1	ADANA	M3		7	0				
9	1	ADANA	M3	8	22		1	ADANA	M3		8	0				
10	1	ADANA	M3	9	22		1	ADANA	M3		9	0				
11	3	AFYON	M4	10	5		3	AFYON	M4		10	0				
12	3	AFYON	M5	11	9		3	AFYON	M5		11	0				
13	6	ANKARA	M6	12	9		6	ANKARA	M6		12	0				
14	6	ANKARA	M7	13	5		6	ANKARA	M7		13	0				
15	6	ANKARA	M8	14	1		6	ANKARA	M8		14	0				
16	6	ANKARA	M9	15	4		6	ANKARA	M9		15	0				
17	6	ANKARA	M10	16	1		6	ANKARA	M10		16	0				
18	6	ANKARA	M11	17	2		6	ANKARA	M11		17	0				
19	6	ANKARA	M12	18	2		6	ANKARA	M12		18	0				
20	6	ANKARA	M12	19	22		6	ANKARA	M12		19	0				
21	6	ANKARA	M12	20	22		6	ANKARA	M12		20	0				
22	6	ANKARA	M12	21	22		6	ANKARA	M12		21	0				
23	6	ANKARA	M13	22	1		6	ANKARA	M13		22	0				
24	6	ANKARA	M14	23	19		6	ANKARA	M14		23	0				
25	6	ANKARA	M14	24	22		6	ANKARA	M14		24	0				
26	7	ANTALYA	M15	25	8		7	ANTALYA	M15		25	0				
27	7	ANTALYA	M16	26	21		7	ANTALYA	M16		26	0				
28	7	ANTALYA	M17	27	3		7	ANTALYA	M17		27	0				
29	7	ANTALYA	M18	28	21		7	ANTALYA	M18		28	0				
30	7	ANTALYA	M19	29	1		7	ANTALYA	M19		29	0				
31	9	AYDIN	M20	30	3		9	AYDIN	M20		30	0				
32	10	BALIKESİR	M21	31	2		10	BALIKESİR	M21		31	0				
33	16	BURSA	M22	32	4		16	BURSA	M22		32	0				
34	16	BURSA	M23	33	19		16	BURSA	M23		33	0				
35	16	BURSA	M23	34	22		16	BURSA	M23		34	0				
36	16	BURSA	M24	35	8		16	BURSA	M24		35	0				
37	16	BURSA	M24	36	22		16	BURSA	M24		36	0				
38	16	BURSA	M25	37	21		16	BURSA	M25		37	0				
39	16	BURSA	M26	38	4		16	BURSA	M26		38	0				
40	17	ÇANAKKALE	M27	39	10		17	ÇANAKKALE	M27		39	0				
41	21	DIYARBAKIR	M28	40	18		21	DIYARBAKIR	M28		40	0				
42	21	DIYARBAKIR	M28	41	22		21	DIYARBAKIR	M28		41	0				
43	21	DIYARBAKIR	M29	42	4		21	DIYARBAKIR	M29		42	0				
44	21	DIYARBAKIR	M29	43	22		21	DIYARBAKIR	M29		43	0				
45	21	DIYARBAKIR	M29	44	22		21	DIYARBAKIR	M29		44	0				
46	22	EDİRNE	M30	45	19		22	EDİRNE	M30		45	0				
47	22	EDİRNE	M31	46	1		22	EDİRNE	M31		46	0				
48	22	EDİRNE	M32	47	21		22	EDİRNE	M32		47	0				
49	23	ELAZIĞ	M33	48	4		23	ELAZIĞ	M33		48	0				
50	27	GAZİANTEP	M34	49	8		27	GAZİANTEP	M34		49	0				
51	27	GAZİANTEP	M34	50	22		27	GAZİANTEP	M34		50	0				
52	27	GAZİANTEP	M34	51	22		27	GAZİANTEP	M34		51	0				
53	28	GİRESUN	M35	52	4		28	GİRESUN	M35		52	0				
54	28	GİRESUN	M36	53	1		28	GİRESUN	M36		53	0				
55	28	GİRESUN	M37	54	4		28	GİRESUN	M37		54	0				
56	31	HATAY	M38	55	22		31	HATAY	M38		55	0				
57	33	İÇEL	M39	56	8		33	İÇEL	M39		56	0				
58	33	İÇEL	M39	57	22		33	İÇEL	M39		57	0				
59	33	İÇEL	M39	58	22		33	İÇEL	M39		58	0				
60	33	İÇEL	M40	59	1		33	İÇEL	M40		59	0				
61	33	İÇEL	M41	60	1		33	İÇEL	M41		60	0				
62	34	İSTANBUL	M42	61	20		34	İSTANBUL	M42		61	0				
63	34	İSTANBUL	M43	62	1		34	İSTANBUL	M43		62	0				
64	34	İSTANBUL	M43	63	22		34	İSTANBUL	M43		63	0				
65	34	İSTANBUL	M44	64	1		34	İSTANBUL	M44		64	0				
66	34	İSTANBUL	M45	65	14		34	İSTANBUL	M45		65	0				

Şekil 7.3. "Verileri Düzenle" butonuna basıldıktan sonra "Veri Girişi" sayfasının görünümü

b)GAMS yazılımının çalıştırılması:

Program kaydedilip kapatıldıktan sonra bölge oluşturma aşaması için oluşturulmuş olan GAMS yazılımı çalıştırılır. GAMS yazılımından elde edilen veriler MS EXCEL programına her bir çözüm için ayrı bir sayfaya yazdırılır. Bu sayfalardan bir örnek Şekil 7.5’te verilmiştir.

Şekil 7.4. Bölge oluşturulma aşaması için GAMS yazılımı çalıştırıldıktan sonra verilerin EXCEL programına yazdırılan sonuçların görüntüsü

c) “Bölgeler” Sayfası:

“Bölgeler” sayfası elde edilen GAMS yazılımı verilerinin daha anlaşılır hale getirmek için hazırlanmıştır. “Bölgeleri Yaz” butonuna tıklandığında sırası ile birinci amaç fonksiyonunun en iyi ve en kötü değeri, ikinci amaç fonksiyonunun en iyi ve en kötü değeri, min operatörü, iki aşamalı yaklaşımı, ağırlıklı max-min operatörü, ağırlıklı toplam operatörüne göre elde edilen bölgeler yazdırılır. D1 ve E1 hücrelerinde amaçlara atanan değerler yer almaktadır. Şekil 7.5 ve 7.6’da görülmektedir.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1		BÖLGELER		0,937317	0,062683	Bölgeleri yaz		SİL						
2		ilk amaç fonksiyonuna göre		en iyi		Ctrl								
3														
4		4	10	17	22	23	24	26	27	31	33	34	37	40
5		142	11	12	13	14	228	210	25	96	188	181	170	131
6			112	15	128	18			29	154			175	224
7			196	16	192	222			200				233	
8				159									234	
9														
10														
11	uzaklık	837	373	382	798	233	233	537	537	324	0	0	486	13:
12	saving	26	672	1904	1238	1412	160	32	1928	680	190	190	584	25:
13														
14														
15		ilk amaç fonksiyonuna göre		en kötü										
16														
17		1	3	6	7	34	36	40	41	42	43	44	48	49
18		37	157	169	61	245	251	38	233	15	142	236	35	13
19		208			203			170	250	31		240	39	92
20								237		32			156	200
21								238		81				
22										88				
23										140				
24										239				
25														
26														
27														
28														
29														
30														
31														
32														
33														
34														
35														
36														
37														
38														
39														
40														
41														
42	uzaklık	1674	837	837	1776	243	243	5294	2728	8981	1283	2728	3661	28:
43	saving	52	26	26	105	102	102	252	200	1016	26	200	78	7:
44														
45														
46														
47		ikinci amaç fonksiyonuna göre		en iyi										
48														
49		5	9	11	16	17	19	20	21	27	29	31	32	33
50		219	207	100	108	22	176	226	197	25	28	45	38	146
51				187	229	23				111	196	149	96	
52						212				147			178	
53						220								
54														
55														
56														
57														
58														
59	uzaklık	688	837	600	552	466	382	233	382	860	537	562	322	
60	saving	160	26	222	775	1573	26	360	26	1540	880	505	570	11
61		ana veri	veri girişi	uzaklık	tasarruf	bolgeler	bolge düzenle	rota	verel arama	rota				

Hedefi belirleyip ENTER tuşuna basın veya Yapıştır'ı seçin

Şekil 7.5. "Bölgeler" sayfasının "Bölgeleri Yaz" butonuna basıldıktan sonraki görüntüsü, 1. kısım

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Bölgeleri A4'den başlayarak giriniz.														
2							Bölge verilerini düzenle								
3															
4	1	2	3	4	13	16	18	25	27	28	29	34	40	43	48
5	35	147	184	205	31	156	33	62	45	68	47	139	209	178	153
6	171				32		197	67	246	248	236	255			
7					161			81			240				
8								88							
9								93							
10								247							
11															
12															
13															
14															
15															
16															
17															
18															
19															
20															
21															
22															
23															
24															
25															
26															
27															
28															
29															
30															
31															
32															
33															
34															
35															
36															
37															
38															
39															
40															
41															
42															
43															
44															
45															
46															
47															
48															
49															
50															
51															
52															
53															
54															
55															
56															
57															
58															
59															
60															

Şekil 7.7. "Bölge Düzenle" sayfasına veri girildikten sonra görünümü

“Bölge Verilerini Düzenle” butonuna basıldıktan sonra her bir araca atanmış olan noktaların şehir isimleri, şehir plaka numaraları, müşteri numaraları, talep miktarları, müşterilere ilk aşamada program tarafından atanan numaralar, depo plaka numarası, araç kapasitesi ve uzaklık matrisi her bir araç için ayrı bir

sayfaya yazdırılır. Butona basıldıktan sonra bir araç için yazdırılan sayfa örneği Şekil 7.8’de görülmektedir.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Plaka no	İl Adı	Müşteri nı No	Talep			Plaka no	İl Adı	Tedarikçi i No	Talep			depo plak	11	araç kapas	22		
2	11 Bilecik		0	1	0	depo	11 Bilecik		0	1	0							
3	1 ADANA	M1		2	1		1 ADANA	M1		2	0							
4	16 BURSA	M24		3	8	35	16 BURSA	M24		3	0							
5	16 BURSA-İN	T4		4	0	171	16 BURSA-İN	T4		4	22				1	2	3	4
6														1	0	768	95	95
7														2	768	0	837	837
8														3	95	837	0	0
9														4	95	837	0	0
10																		
11																		
12																		
13																		
14																		
15																		
16																		
17																		
18																		
19																		
20																		
21																		
22																		
23																		
24																		
25																		
26																		
27																		
28																		
29																		
30																		
31																		
32																		
33																		
34																		
35																		
36																		
37																		
38																		
39																		
40																		
41																		
42																		
43																		
44																		
45																		
46																		
47																		
48																		
49																		
50																		
51																		
52																		
53																		
54																		
55																		
56																		
57																		
58																		
59																		
60																		

Şekil 7.8. "Bölge verilerini düzenle" butonuna basıldıktan sonraki görünüm

b) GAMS yazılımının çalıştırılması:

Veriler düzenlendikten sonra rotalama aşaması için oluşturulan GAMS yazılımı kodları çalıştırılır. GAMS yazılımının sonucu her bir araç için MS EXCEL programının ayrı bir sayfasına yazdırılır. Sayfaların görünümü test problemleri için geliştirilmiş olan ara yüzde anlatıldığı gibidir.

c) “Rota Yerel Arama” sayfası:

GAMS yazılımından elde edilen çözümlerin daha anlaşılır hale gelmesi için ve yerel arama aşamasının yapılması için “Rota Yerel Arama” sayfası oluşturulmuştur. J1 hücresine araç kapasitesinin girilmiş olması yeterlidir.

7.3.3. Yerel aramada kullanılan ara yüz

“Rota Yerel Arama” sayfası, hem GAMS yazılımından elde edilen çözümler daha toplu bir halde ve daha anlaşılır bir biçimde görülmesi için hem de yerel arama algoritması için oluşturulmuştur. Aracın kapasitesinin J1 hücresine girilmesi gerekmektedir. “Rota Oluşturma ve Geliştirme” butonuna basıldığında öncelikle bölge oluşturma ve rotalama aşamaları sonucu elde edilen rota yazdırılır, ardından yerel arama ile iyileştirilmiş rota yazdırılır. Ayrıca yerel arama için harcana CPU zamanı L1 hücresinde belirtilir. Ayrıca sayısal veriler şehir ve müşteri/tedarikçi numaraları ile birlikte sözel olarak “Rota” sayfasına yazdırılır. “Rota Oluşturma ve Geliştirme” butonuna basıldıktan sonraki “Rota Yerel Arama” sayfasının görüntüsü Şekil 7.9’da ve “Rota” sayfasının görüntüsü ise Şekil 7.10’da yer almaktadır. “0” değeri depoyu temsil etmektedir.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	rota oluřturma ve geliřtirme				sil				kapasite	22	CPU	2,54	
2	1700	0	4	142	0								
3	578	0	11	10	112	196	0						
4	790	0	12	16	17	15	159	0					
5	1570	0	22	13	128	192	0						
6	626	0	18	14	23	222	0						
7	626	0	24	228	0								
8	1106	0	26	210	0								
9	1106	0	27	25	29	200	0						
10	836	0	31	96	154	0							
11	190	0	33	188	0								
12	190	0	34	181	0								
13	588	0	37	175	170	233	234	0					
14	2802	0	40	131	224	0							
15	2592	0	41	204	0								
16	2592	0	44	185	0								
17	995	0	45	46	171	0							
18	995	0	47	163	0								
19	3015	0	48	42	135	161	0						
20	2367	0	50	231	139	239	0						
21	1946	0	51	223	0								
22	1922	0	122	53	52	54	166	0					
23	2171	0	7	1	49	60	59	56	207	0			
24	500	0	68	61	93	245	0						
25	588	0	65	62	81	169	0						
26	588	0	66	178	0								
27	588	0	79	157	0								
28	500	0	80	252	0								
29	500	0	85	253	0								
30	500	0	67	92	64	82	88	251	0				
31	834	0	97	94	98	180	0						
32	1408	0	104	186	0								
33	1408	0	103	106	147	0							
34	1954	0	102	101	107	167	0						
35	836	0	110	109	111	218	0						
36	834	0	99	113	209	0							
37	1184	0	117	116	114	115	30	100	143	0			
38	1834	0	118	120	121	119	179	0					
39	1500	0	108	124	206	0							
40	1530	0	127	203	0								
41	1530	0	125	130	255	141	0						
42	1384	0	133	132	134	213	0						
43	2344	0	138	123	136	137	205	0					
44	190	0	144	0									
45	588	0	75	145	0								
46	190	0	146	0									
47	588	0	71	148	0								
48	190	0	36	149	0								
49	190	0	150	0									
50	588	0	72	151	0								
51	1700	0	6	152	0								
52	588	0	73	153	0								
53	190	0	155	0									
54	588	0	69	156	0								
55	190	0	158	0									
56	190	0	160	0									
57	1684	0	58	162	0								
58	588	0	78	164	0								

řekil 7.9. "Bölge Oluřturma ve Geliřtirme" butonuna basıldıktan sonraki görünüml

	A	B	C	D	E	F	G	H	I	J	K
1											
2	1700	Bilecik	ADANA-M2	BURSA-T3	Bilecik						
3	578	Bilecik	AFYON-M5	AFYON-M4	KÜTAHYA-M70	BURSA-İNEGÖL-T5	Bilecik				
4	790	Bilecik	ANKARA-M6	ANKARA-M10	ANKARA-M11	ANKARA-M9	BURSA-T3	Bilecik			
5	1570	Bilecik	ANKARA-M13	ANKARA-M7	SAMSUN-M84	BURSA-İNEGÖL-T5	Bilecik				
6	626	Bilecik	ANKARA-M12	ANKARA-M8	ANKARA-M14	ESKİŞEHİR-T10	Bilecik				
7	626	Bilecik	ANKARA-M14	ESKİŞEHİR-T11	Bilecik						
8	1106	Bilecik	ANTALYA-M16	BURSA-İNEGÖL-T5	Bilecik						
9	1106	Bilecik	ANTALYA-M17	ANTALYA-M15	ANTALYA-M19	BURSA-İNEGÖL-T5	Bilecik				
10	836	Bilecik	BALIKESİR-M21	İZMİR-M56	BURSA-T3	Bilecik					
11	190	Bilecik	BURSA-M23	BURSA-İNEGÖL-T5	Bilecik						
12	190	Bilecik	BURSA-M23	BURSA-İNEGÖL-T5	Bilecik						
13	588	Bilecik	BURSA-M25	BURSA-İNEGÖL-T5	BURSA-İNEGÖL-T4	İSTANBUL-B,ÇEKMECE-T14	İSTANBUL-B,ÇEKMECE-T15	Bilecik			
14	2802	Bilecik	DIYARBAKIR-M28	SİİRT-M86	ESKİŞEHİR-T11	Bilecik					
15	2592	Bilecik	DIYARBAKIR-M28	BURSA-İNEGÖL-T5	Bilecik						
16	2592	Bilecik	DIYARBAKIR-M29	BURSA-İNEGÖL-T5	Bilecik						
17	995	Bilecik	EDİRNE-M30	EDİRNE-M31	BURSA-İNEGÖL-T4	Bilecik					
18	995	Bilecik	EDİRNE-M32	BURSA-T3	Bilecik						
19	3015	Bilecik	ELAZIĞ-M33	DIYARBAKIR-M29	TRABZON-M90	BURSA-T3	Bilecik				
20	2367	Bilecik	GAZİANTEP-M34	GAZİANTEP-BAŞPINAR -T12	ADANA-T1	İSTANBUL-PENDİK-T20	Bilecik				
21	1946	Bilecik	GAZİANTEP-M34	ESKİŞEHİR-T10	Bilecik						
22	1922	Bilecik	ORDU-M80	GİRESUN-M36	GİRESUN-M35	GİRESUN-M37	BURSA-T3	Bilecik			
23	2171	Bilecik	ADANA-M3	ADANA-M1	GAZİANTEP-M34	İÇEL-M41	İÇEL-M40	BURSA-İNEGÖL-T5	Bilecik		
24	500	Bilecik	İSTANBUL-M47	İSTANBUL-M42	İSTANBUL-M54	İSTANBUL-TUZLA -T24	Bilecik				
25	588	Bilecik	İSTANBUL-M45	İSTANBUL-M43	İSTANBUL-M49	BURSA-T3	Bilecik				
26	588	Bilecik	İSTANBUL-M45	BURSA-İNEGÖL-T5	Bilecik						
27	588	Bilecik	İSTANBUL-M48	BURSA-T3	Bilecik						
28	500	Bilecik	İSTANBUL-M48	İSTANBUL-TUZLA -T25	Bilecik						
29	500	Bilecik	İSTANBUL-M50	İSTANBUL-TUZLA -T25	Bilecik						
30	500	Bilecik	İSTANBUL-M46	İSTANBUL-M53	İSTANBUL-M44	İSTANBUL-M50	İSTANBUL-M51	İSTANBUL-TUZLA -T25	Bilecik		
31	834	Bilecik	İZMİR-M57	İZMİR-M55	İZMİR-M58	BURSA-İNEGÖL-T5	Bilecik				
32	1408	Bilecik	KAYSERİ-M63	BURSA-İNEGÖL-T5	Bilecik						
33	1408	Bilecik	KAYSERİ-M63	KAYSERİ-M64	BURSA-T3	Bilecik					
34	1954	Bilecik	KAYSERİ-M62	KAHRAMANMARAŞ-M61	KOCAELİ-M65	BURSA-T3	Bilecik				
35	836	Bilecik	KONYA-M68	KONYA-M67	KONYA-M69	ESKİŞEHİR-T9	Bilecik				
36	834	Bilecik	İZMİR-M59	MANİSA-M71	BURSA-İNEGÖL-T5	Bilecik					
37	1184	Bilecik	MUĞLA-M75	MUĞLA-M74	MUĞLA-M72	MUĞLA-M73	AYDIN-M20	İZMİR-M60	BURSA-T3	Bilecik	
38	1834	Bilecik	ORDU-M76	ORDU-M78	ORDU-M79	ORDU-M77	BURSA-İNEGÖL-T5	Bilecik			
39	1500	Bilecik	KOCAELİ-M66	SAKARYA-M82	BURSA-İNEGÖL-T5	Bilecik					
40	1530	Bilecik	SAMSUN-M83	BURSA-İNEGÖL-T5	Bilecik						
41	1530	Bilecik	SAMSUN-M83	SAMSUN-M85	SAMSUN-KUTLUKENT-T27	BURSA-T3	Bilecik				
42	1384	Bilecik	TOKAT-M88	TOKAT-M87	TOKAT-M89	ESKİŞEHİR-T6	Bilecik				
43	2344	Bilecik	TRABZON-M93	RİZE-M81	TRABZON-M91	TRABZON-M92	BURSA-İNEGÖL-T5	Bilecik			
44	190	Bilecik	BURSA-T3	Bilecik							
45	588	Bilecik	İSTANBUL-M48	BURSA-T3	Bilecik						
46	190	Bilecik	BURSA-T3	Bilecik							
47	588	Bilecik	İSTANBUL-M48	BURSA-T3	Bilecik						
48	190	Bilecik	BURSA-M24	BURSA-T3	Bilecik						
49	190	Bilecik	BURSA-T3	Bilecik							
50	588	Bilecik	İSTANBUL-M48	BURSA-T3	Bilecik						
51	1700	Bilecik	ADANA-M2	BURSA-T3	Bilecik						
52	588	Bilecik	İSTANBUL-M48	BURSA-T3	Bilecik						
53	190	Bilecik	BURSA-T3	Bilecik							
54	588	Bilecik	İSTANBUL-M47	BURSA-T3	Bilecik						
55	190	Bilecik	BURSA-T3	Bilecik							
56	190	Bilecik	BURSA-T3	Bilecik							
57	1684	Bilecik	İÇEL-M39	BURSA-T3	Bilecik						
58	588	Bilecik	İSTANBUL-M48	BURSA-T3	Bilecik						
59	1530	Bilecik	SAMSUN-M83	BURSA-T3	Bilecik						
60	500	Bilecik	İSTANBUL-M46	BURSA-T3	Bilecik						

Şekil 7.10. "Rota" sayfasının görünümü

8. SONUÇ

Literatür incelendiğinde ÖDST-ARP için matematiksel modeller önerilmiş, sezgisel ve meta sezgisel yaklaşımlar geliştirilmiştir. Önerilen matematiksel modellerde çözüm süresi problemin boyutuna bağlı olarak arttığı için etkin bir şekilde kullanılamamıştır. Sezgisel ve meta sezgisel yaklaşımlar da bir bilgisayar programlama dilinde algoritmaların kodlanmasını gerektirmektedir.

Bu çalışmanın amacı ÖDST-ARP'ni matematiksel model kullanarak çözebilmek ve programlamaya duyulan ihtiyacı azaltmaktır. Bu amaçla iki aşamalı bir yaklaşım geliştirilmiştir. Bölge oluşturma olarak isimlendirilen ilk aşamada çok amaçlı matematiksel model ile müşterilerin araçlara atanması sağlanmış ve modelin çözümü için bulanık çözüm yaklaşımları kullanılmıştır. Amaçların ağırlıkları, Belenson ve Kapur'un iki oyunculu sıfır toplamli oyuna dayanan yöntemlerinde yapılan düzenlemeler ile önerilen metotla belirlenmiştir. Bölge oluşturma aşamasının sonucunda her bir araç için problem ÖDST-GSP'ne dönüştürülmüştür. Goetschalckx ve Jacobs-Belcha (1989)'nın ÖDST-ARP için önerdikleri matematiksel model yeniden düzenlenerek ÖDST-GSP için matematiksel model önerilmiştir. Rota oluşturma olarak isimlendirilen ikinci aşamada ise her bir aracın rotası bu modelin çözümü ile bulunmuştur. Son olarak ekleme ve değiştirme operatörlerinin kullanıldığı yerel arama ile oluşturulan rota iyileştirilmiştir.

Literatürde yer alan test problemleri önerilen yöntem ile çözülmüştür. Bölge oluşturma aşamasında birinci amaç fonksiyonu, ikinci amaç fonksiyonu ve bulanık çözüm operatörlerine göre elde edilen bölgeler için rotalar oluşturulmuştur. Her zaman en iyi bilinen sonuca ulaşamamasına rağmen, en iyi bilinen çözümden olan farklar kabul edilebilir düzeydedir. Ancak A1 test problemi için ikinci amaç ve ağırlıklı max-min operatörüne göre elde edilen çözüm ile en iyi değere ulaşılmıştır. Problemlerin çözümü için gereken CPU zamanı da kabul edilebilir sınırlar içerisinde.

Ayrıca seramik sağlık gereçleri üreten bir işletmenin bir haftalık rotalama problemi önerilen yöntem ile çözülmüştür. Bazı müşterilerin talepleri araç kapasitesini aştığı için bir düzenleme yapılarak bu müşterilerin talepleri araç

kapasitesine göre bölüştürülerek yapay müşteriler oluşturulmuştur. Birinci amaç fonksiyonuna göre elde edilen çözüm diğerlerine göre daha iyi bulunmuştur. Ayrıca çözüm için harcanan CPU zamanı makul ölçüdedir. Önerilen yöntem gerçek hayat problemleri için de uygundur.

Daha sonraki çalışmalarda, önerilen yöntemin bölge oluşturma aşamasında başka amaçlar göz önüne alınarak çözümler elde edilebilir. Müşteri taleplerinin ve/veya müşteri noktaları arasındaki mesafelerin bulanık olması durumu göz önüne alınarak çözüm aşaması üzerinde çalışılabilir. Uygulama probleminde şehir içindeki dolaşımlar dikkate alınabilir. Ayrıca önerilen yöntem başka araç rotalama problemi türleri için de uyarlanabilir. Bölge oluşturma aşamasında ve yerel aramada toplama yapılacak müşterilerle ilgili kısıtlar çıkarılarak, rota oluşturma aşamasında GSP modeli kullanılarak özellikle kapasiteli araç rotalama problemine uygulanması kolaylıkla sağlanabilir.

KAYNAKLAR

- Amid, A., Ghodsypour, S., ve O'Brien, C., (2006), "Fuzzy multiobjective linear model for supplier selection in a supply chain," *International Journal of Production Economics*, 104, 394-407.
- Amid, A., Ghodsypour, S., ve O'Brien, C., (2009), "A weighted additive fuzzy multiobjective model for the supplier selection problem under price breaks in a supply chain," *International Journal of Production Economics*, 121(2), 323-332.
- Amid, A., Ghodsypour, S., ve O'Brien, C., (2011), "A weighted max-min model for fuzzy multi-objective supplier selection in a supply chain", *International Journal of Production Economics*, 131(1), 139-145.
- Araz, C., Mızrak Ozfırat, P., ve Ozkarahan, İ., (2007), "An integrated multicriteria decision-making methodology for outsourcing management", *ComputerveOperations Research*, 34, 3738-3756.
- Araz, C., Selim, H., ve Ozkarahan, İ., (2007), "A fuzzy multi-objective covering-based vehicle location model for emergency services", *ComputersveOperations Research*, 34, 705-726.
- Arıkan, F., ve Güngör, Z., (2001), "An application of fuzzy goal programming to a multi objective project network problem", *Fuzzy Sets and Systems*, 119, 49-58.
- Bramel, J. ve Simchi-Levi, D., (1995), "A location based heuristic for general routing problems", *Operations Research*, 43, 649-660.
- Belenson, S., ve Kapur, K., (1973), "An algorithm for solving multicriterion linear programming problems with examples", 24(1), 65-77.
- Belmann, R. E., ve Zadeh, L. A., (1970), "Decision making in a fuzzy enviroment", *Management Science*, 17, 141-164.
- Brandão, J., (2006), "A new tabu search algorithm for the vehicle routing problem with backhauls", *European Journal of Operational Research*, 173, 540-555.
- CihangLin, C-Y., ve Lin, C-C., (2008), "Personal financial planning based on fuzzy multiple objective programming", *Expert Systems with Applications*, 35, 373-378.

- Clarke, G. and Wright, J.V., (1964) "Scheduling of vehicles from a central depot to a number of delivery points" *Operations Research*, 12, 568-581.
- Dantzig, G. B. ve Ramser, J. H., (1959), "The Truck dispatching problem" *Management Science*, 6(1), 80-91.
- Dastkhan, H., Gharneh, N. S., ve Golmakani, H. R., (2011), "A linguistic-based portfolio selection model using weighted max-min model operator and hybrid genetic algortihm", *Expert System with Applications*, 38, 11735-11743.
- Deif, I., ve Bodin, L., (1984), "Extension of the Clarke and Wright algorithm for solving the vehicle routing problem with backhauls", *The Babson Conference on Software Uses in Transportation and Logistic Management*, 75-96.
- Dinç Yalçın, G., ve Erginel, N., (2011), "Determining weights in multi-objective linear programming under fuzziness", *Word Congress on Engineering 2011*, London, İngiltere, 6-8 Temmuz.
- El-Wahed, W. A., (2001), "A multi objective transportation problem under fuzziness", *Fuzzy Sets and System*, 117, 27-33.
- El-Wahed, W., ve Abo-Sinna, M., (2001), "A hybrid fuzzy goal programming approach to multiple objective decision making problems", *Fuzzy Sets and Systems*, 119, 71-85.
- Fisher, M. L. ve Jaikumar, R., (1978), *A decomposition algorithm large-scale vehicle routing*, University of Pennsylvania, wharton Department of Decision Sciences.
- Fisher, M. L. ve Jaikumar, R., (1981), "A generalized assignment heuristic for the vehicle routing problem", *Networks*, 11, 109-124.
- Gajpal, Y. ve Abad, P. L., (2009), "Multi-ant colony system (MACS) for a vehicle routing problem with backhauls", *European Journal of Operational Research*, 196, 102-117.
- Ghaziri, H. ve Osman, I. H., (2006), "Self-organizing feature maps for the vehicle routing problem with backhhauls", *Journal of Scheduling*, 9, 97-114.
- Ghaziri, H. ve Osman, I. H., (2003), "A neural network algorithm for travelling

- salesman problem with backhauls”, *Computers and Industrial Engineering*, 44, 267-281.
- Gillett, B.E. ve Miller L.R., (1974), “A heuristic algorithm for the vehicle dispatch problem”, *Operations Research*, 22, 340-349.
- Goetschalckx, M., ve Jacobs-Blecha, C., (1989), “The vehicle routing problem with backhauls”, *European Journal of Operational Research*, 42, 39-51.
- Gupta, P., Mehlawat, M, K., ve Saxena, A., (2008), “Asset portfolio optimization using fuzzy mathematical programming”, *Information Sciences*, 178, 1734-1755.
- Gupta, S., ve Chakraborty, M., (1998), “Job evaluation in fuzzy environment”, *Fuzzy Sets and System*, 100, 71-76.
- Jazemi, R., Ghodsypour, S., ve Gheidar-Kheljani, J., (2011), “Considering supply chain benefit in supplier selection problem by using information sharing benefits”, *IEEE transactions on Industrial Informatics*, 7(3), 517-526.
- Kongar, E., ve Gupta, S., (2006), “Disassembly to order system under uncertainty”, *The International Journal of Management Science*, 34, 550-561.
- Kuruüzüm, A, (1998), *Karar destek sistemlerinde çok amaçlı yöntemler*, Antalya, Türkiye: Akdeniz Üniversitesi Basımevi.
- Lai, Y-J., ve Hwang, C-L, (1994), *Fuzzy Multiple Objective Decision Making: Methods and Applications*, Newyork: Springer.
- Li, L., ve Lai, K., (2000), “A fuzzy approach to the multi objective transportation problem”, *Computers and Operations Research*, 27, 43-57.
- Li, X-Q., Zhang, B., ve Li, H., (2006), “Computing efficient solutions to fuzzy multiple objective linear programming problems”, *Fuzzy Sets and System*, 157, 1328-1332.
- Liang, T-F., (2008), “Fuzzy multi-objective production/distribution planning decisions with multi-product and multi-time period in a supply chain”, *Computers and Industrial Engineering*, 55, 676-694.
- Liang, T-F., (2009), “Fuzzy multi objective project management decisions using two phase fuzzy goal programming approach”, *Computer and Industrial Engineering*, 57(4), 1407-1416.

- Liang, T.-F., ve Cheng, H-W, (2009), "Application of fuzzy sets to manufacturing/distribution planning decisions with multi-product and multi-time period in supply chains", *Expert Systems with Applications*, 36, 3367-3377.
- Liang, T-F., Cheng, H-W., Chen, P-Y., ve Shen, K-H, (2011), "Application of Fuzzy sets to aggregate production planning with multiproducts and multitime periods", *Transactions on Fuzzy Systems*, 19(3), 465-477.
- Liang, T-F., Huang, T-S., ve Yang, M-F., (2011), "Application of fuzzy mathematical programming to imprecise project management decisions", *Quality and Quantity*.
- Lin, C-C., (2004), "A weighted max-min model for fuzzy goal programming", *Fuzzy Sets and Systems*, 142, 407-420.
- Lin, C-C., ve Liu, Y-T., (2008), "Genetic algorithms for portfolio selection problems with minimum transaction lots", *European Journal of Operational Research*, 185, 393-404.
- Mingozi, A., ve Giorgi, S., (1999), "An exact algorithm for the vehicle routing problem with backhauls", *Transportation Science*, 33(3), 315-329.
- Nabiyev, V., (2005), *Yapay zeka, problemler-yöntemler-algoritma*, Ankara, Türkiye: Seçkin Yayıncılık.
- Öztürk, A., (2007), *Yöneylem araştırması*, Bursa, Türkiye: Ekin Kitabevi.
- Rabbani, M., Jolai, F., Manavizadeh, N., Radmehr, F., ve Javadi, B., (2012), "Solving a bi-objective cell formation problem with stochastic production quantities by a two phase fuzzy linear programming approach", *International Journal of Manufacturing Technology*, 58, 709-722.
- Reimann, M., Doerner, K. ve Hartl, R.F., (2004), "DANTS: Savings based ants divide and conquer the vehicle routing problem", *Computers and Operations Research*, 31, 563-591.
- Ropke, S., ve Pisinger, D., (2006), "A unified heuristic for a large class of vehicle routing problems with backhauls", *European Journal of Operational Research*, 171, 750-775.

- Salhi, S., ve Nagy, G., (1999), "A cluster insertion heuristic for single and multiple depot vehicle routing problems with backhauling", *Journal of Operational Research Society*, 50, 1034-1042.
- Selim, H., ve Ozkarahan, I., (2008), "A supply chain distribution network design model: An interactive fuzzy goal programming-based solution approach", 36, 401-418.
- Selim, H., Araz, C., ve Ozkarahan, İ., (2008), "Collaborative production-distribution planning in supply chain: a fuzzy goal programming approach", *Transportation research Part E*, 44, 396-419.
- Taha, H., (2007), *Yöneylem Araştırması*, (Ş, Baray, ve Ş, Esnaf, Çev,) İstanbul: Literatür Yayınları.
- Tavakkoli-Moghaddam, R., Saremi, A. R., ve Ziaee, M. S., (2006), "A memetic algorithm for a vehicle routing problem with backhauls", *Applied Mathematics and Computations*, 181, 104-1060.
- Thangiah, S. R., ve Sun, T., (1996), "Heuristic approaches to vehicle routing with backhauls and time windows", *Computers and Operations Research*, 23(11), 1043-1057.
- Tiwari, R., Dharmar, S., ve Rao, J., (1987), "Fuzzy goal programming- An additive model", 24(1), 27-54.
- Topaloğlu, S., ve Selim, H., (2007), "Nurse scheduling using fuzzy multiple objective programming", H, Okuno, ve M, Ali içinde, *New Trends in Applied Artificial Intelligence* (s, 54-63), Kyoto, Japonya: Springer.
- Topaloğlu, S., ve Selim, H., (2010), "Nurse scheduling using fuzzy modeling approach", *Fuzzy Sets and Systems*, 161, 1543-1563.
- Toth, P., ve Vigo, D., (1996), "A heuristic algorithm for the vehicle routing problem with backhauls", *Transportation Analysis*, 585-608.
- Toth, P., ve Vigo, D., (1997), "An exact algorithm for the vehicle routing problem with backhauls", *Transportation Science*, 31(4), 372-385.
- Toth, P., ve Vigo, D., (1999), "A heuristic algorithm for the symmetric and asymmetric vehicle routing problems with backhauls", *European Journal of Operational Research*, 113, 528-543.

- Toth, P., ve Vigo, D., (2002), *The vehicle routing problem*, Philadelphia, Amerika: Society for Industrial and Applied Mathematics.
- Tütüncü, G. Y., (2010), “An interactive GRAMPS algorithm for the heterogeneous fixed fleet vehicle routing problem with and without backhauls”, *European Journal of Operational Research*, 201, 593-600.
- Tütüncü, G. Y., Carreto, C. A., ve Baker, B. M., (2009), “A visual interactive approach to classical and mixed vehicle routing problems with backhauls”, *The International Journal of Management Science*, 37, 138-154.
- Wade, A., ve Salhi, S., (2001), “An ant system algorithm for the vehicle routing problem with backhauls”, 4th Metaheuristics International Conference, Porto, Portekiz, 16-20 Temmuz, 199-203.
- Wade, A., ve Salhi, S., (2002), “An investigation into a new class of vehicle routing problem with backhauls”, *The International Journal of Management Science*, 30, 479-487.
- Wang, R-C., ve Liang, T-F., (2004), “Application of fuzzy multi objective linear programming to aggregate production planning”, *Computers and Industrial Engineering*, 46, 17-41.
- Wang, R-C., ve Liang, T-F., (2005), “Aggregate production planning with multiple fuzzy goals”, *International Journal of Advance Manufacturing Technology*, 25, 589-597.
- Wang, Z., ve Wang, Z., (2009), “A novel two-phase heuristic method for vehicle routing problem with backhauls”, *Computer and Mathematics with Applications*, 57, 1923-1928.
- Werner, B., (1988), “Aggregation models in mathematical programming”, G, Mitra, H, Greenberg, F, Lootsma, M, Rijckaert, ve H, Zimmermann içinde, *Mathematical models for decision support*, Newyork: Springer, 295-305.
- Winston, W., (2004), *Operations research, applications and algorithms*, Duxbury Pres Thomson Learning.
- Yano, C. A., Chan, T. J., Richter, L., Culter, T., Murty, K., ve McGettingan, D., (1987), “Vehicle routing at quality stores”, *Interfaces*, 17(2), 52-63.

- Yücel, A., ve Güneri, A. F., (2011), “A weighted additive fuzzy programming approach for multi criteria supplier selection”, *Expert System with Applications*(38), 6281-6286.
- Zarandi, M., Davari, S., ve Sisakht, A., (2011), “Design of a reliable hub-and-spoke network using an interactive fuzzy goal programming”, *IEEE International Conferance on Fuzzy Systems*, Taipei/Taiwan, 2955-2959.
- Zarandi, M., Sisakht, A., ve Davari, S., (2011), “Design of a closed-loop supply chain (CLSC) model using an interactive fuzzy goal programming”, *International Journal of Manufacturing Technology*, 56, 809-821.
- Zeleny M., (1982), *Multiple Criteria Decision Making*. Newyork : McGraw-Hill.
- Zhong, Y., ve Cole, M. H., (2005), “A vehicle routing problem with backhauls and time windows: a guided local search solution”, *Transportation Research Part E*, 41, 131-144.
- Zimmermann, H. J., (1977), “Fuzzy Programming and Linear Programming with Several Objective Functions”, *Fuzzy Sets and Systems*, 1, 45-55.
- Zimmermann, H. J., (2001), *Fuzzy set theory and its applications*, USA: Kluwer academic Publishers.

EK-1 Bölge Oluşturma Aşaması için Yazılan GAMS Yazılımının Kodları

```
option optcr=0;
```

```
sets i(*) ;
```

```
alias (i,j);
```

```
parameter d(i,j);
```

```
Parameter a(i);
```

```
Parameter b(i);
```

```
Parameter s(i,j);
```

```
Scalar araç_kapasite//;
```

```
scalar arac_sayisi//;
```

```
scalar k, eniyi1, eniyi2, enkotu1, enkotu2, a1, a2, lam1, lam2, w1, w2;
```

```
$CALL GDXXRW Case.xls set=i rng=bolge_veri!C3 Rdim=1 par=d
```

```
rng=bolge_veri!K2 rdim=1 cdim=1 par=a rng=bolge_veri!C3 Rdim=1 par=b
```

```
rng=bolge_veri!F3 Rdim=1 par=s rng=tasarruf!A1 Rdim=1 cdim=1
```

```
$GDXIN Case.gdx
```

```
$LOAD i d a b s
```

```
$GDXIN
```

```
d(i,i)=100000000
```

```
display i, d, a, b, s;
```

```
variable z1, z2, z3, z4, z5, z6;
```

```
binary variable x(i,j);
```

```
positive variable lamda, lamda1, lamda2, ww1,ww2, v;
```

```
equations
```

```
amac1
```

```
amac2
```

```
amac_agr
```

agr1
 agr2
 agr3
 min
 uyelik1
 uyelik2
 twophase
 two1
 two2
 uyelik1_two
 uyelik2_two
 uyelik1_maxmin
 uyelik2_maxmin
 agirlikli_toplam
 toplamAtama
 kisit
 kapasite
 arackapasite
 toplambolge;

amac1..

$\text{sum}((i,j) \$ (\text{ord}(i) \neq \text{ord}(j)), d(i,j) * x(i,j)) = e = z1;$

amac2..

$\text{sum}((i,j) \$ (\text{ord}(i) \neq \text{ord}(j)), s(i,j) * x(i,j)) = e = z2;$

min..

$\text{lamda} = e = z3;$

uyelik1..

$(\text{enkotu1} - \text{sum}((i,j) \$ (\text{ord}(i) \neq \text{ord}(j)), d(i,j) * x(i,j))) / (\text{enkotu1} - \text{eniyi1}) = g = \text{lamda};$

uyelik2..

$$(\sum((i,j)\$(\text{ord}(i) \neq \text{ord}(j)),s(i,j)*x(i,j))-\text{enkotu2})/(\text{eniyi2}-\text{enkotu2})=g=\text{lamda};$$

twophase..

$$\text{lamda1}*w1+\text{lamda2}*w2=e=z4;$$

uyelik1_two..

$$(\text{enkotu1}-\sum((i,j)\$(\text{ord}(i) \neq \text{ord}(j)),d(i,j)*x(i,j)))/(\text{enkotu1}-\text{eniyi1})=g=\text{lamda1};$$

uyelik2_two..

$$(\sum((i,j)\$(\text{ord}(i) \neq \text{ord}(j)),s(i,j)*x(i,j))-\text{enkotu2})/(\text{eniyi2}-\text{enkotu2})=g=\text{lamda2};$$

two1..

$$\text{lamda1}=g=\text{lam1};$$

two2..

$$\text{lamda2}=g=\text{lam2};$$

uyelik1_maxmin..

$$(\text{enkotu1}-\sum((i,j)\$(\text{ord}(i) \neq \text{ord}(j)),d(i,j)*x(i,j)))/(\text{enkotu1}-\text{eniyi1})=g=\text{lamda}*w1;$$

uyelik2_maxmin..

$$(\sum((i,j)\$(\text{ord}(i) \neq \text{ord}(j)),s(i,j)*x(i,j))-\text{enkotu2})/(\text{eniyi2}-\text{enkotu2})=g=\text{lamda}*w2;$$

agirlikli_toplam..

$$(\text{enkotu1}-\sum((i,j)\$(\text{ord}(i) \neq \text{ord}(j)),d(i,j)*x(i,j)))/(\text{enkotu1}-\text{eniyi1})*w1+(\sum((i,j)\$(\text{ord}(i) \neq \text{ord}(j)),s(i,j)*x(i,j))-\text{enkotu2})/(\text{eniyi2}-\text{enkotu2})*w2=e=z5;$$

toplamAtama(i)..

sum(j,x(i,j))=e=1;

kisit(i,j)\$(ord(i) ne ord(j))..

x(i,j)=l=x(j,j);

kapasite(j)..

sum(i,b(i)*x(i,j))=l=arac_kapasite;

arackapasite(j)..

sum(i,a(i)*x(i,j))=l=araç_kapasite ;

toplambolge..

sum((i,j)\$(ord(i) eq ord(j)),x(i,j))=e=arac_sayisi;

model bolgeler/amac1, amac2, toplamAtama, kisit, kapasite, arackapasite,
toplambolge/;

model minoperatoru /min, uyelik1, uyelik2, toplamAtama, kisit, kapasite,
arackapasite, toplambolge/;

model two_phase /twophase, uyelik1_two, uyelik2_two,two1, two2,
toplamAtama, kisit, kapasite, arackapasite, toplambolge/;

model max_min / min, uyelik1_maxmin, uyelik2_maxmin, toplamAtama, kisit,
kapasite, arackapasite, toplambolge/;

model agr_top / agirlikli_toplam, toplamAtama, kisit, kapasite, arackapasite,
toplambolge/;

for(k=1 to 4,

if(k eq 1,

solve bolgeler minimizing z1 using mip ;

eniyi1=z1.l ;

a2=z2.l ;

display x.l, a2, eniyi1;

execute_unload "Case.gdx" x.L ;

```

        execute 'gdxxrw.exe Case.gdx var=x.L rng=bolge_eniyi1!A1' ;
    );
    if(k eq 2,
        solve bolgeler maximizing z1 using mip ;
        enkotu1=z1.l ;
        display x.l, enkotu1;
        execute_unload "Case.gdx" x.L ;
        execute 'gdxxrw.exe Case.gdx var=x.L rng=bolge_ekotu1!A1' ;
    );

    if (k eq 3,
        solve bolgeler maximizing z2 using mip ;
        eniyi2=z2.l ;
        a1=z1.l
        display x.l, eniyi2, a1;
        execute_unload "Case.gdx" x.L ;
        execute 'gdxxrw.exe Case.gdx var=x.L rng=bolge_eniyi2!A1' ;
    );

    if(k eq 4,
        solve bolgeler minimizing z2 using mip;
        enkotu2=z2.l ;
        display x.l, enkotu2 ;
        execute_unload "Case.gdx" x.L ;
        execute 'gdxxrw.exe Case.gdx var=x.L rng=bolge_ekotu2!A1' ;
    );
);

amac_agr..
v=e=z6;

agr1..

```

```
ww1+(enkotu1-a1)/(enkotu1-eniyi1)*ww2=g=v;
```

```
agr2..
```

```
(a2-enkotu2)/(eniyi2-enkotu2)*ww1+ww2=g=v;
```

```
agr3..
```

```
ww1+ww2=e=1;
```

```
model agirlik_bul / amac_agr, agr1, agr2, agr3/;
```

```
solve agirlik_bul maximizing z6 using lp ;
```

```
w1=ww1.l;
```

```
w2=ww2.l;
```

```
display ww1.l, ww2.l, w1, w2 ;
```

```
execute_unload "Case.gdx" ww1.L , ww2.L;
```

```
execute 'gdxxrw.exe Case.gdx var=ww1.L rng=bolgeler!D1 var=ww2.L  
rng=bolgeler!E1' ;
```

```
for(k=1 to 4,
```

```
  if( k eq 1,
```

```
    solve minoperatoru maximizing z3 using mip ;
```

```
    display x.l;
```

```
    lam1=uyelik1.l;
```

```
    lam2=uyelik2.l
```

```
    execute_unload "Case.gdx" x.L ;
```

```
    execute 'gdxxrw.exe Case.gdx var=x.L rng=bolge_min!A1' ;
```

```
  );
```

```
  if(k eq 2,
```

```
    solve two_phase maximizing z4 using mip ;
```

```
    display x.l;
```

```
    execute_unload "Case.gdx" x.L ;
```

```
    execute 'gdxxrw.exe Case.gdx var=x.L rng=bolge_two!A1' ;
```

```
  );
```

```

if (k eq 3,
    solve max_min maximizing z3 using mip ;
    display x.l;
    execute_unload "Case.gdx" x.L ;
    execute 'gdxxrw.exe Case.gdx var=x.L rng=bolge_max_min!A1' ;
);

if(k eq 4,
    solve agr_top maximizing z5 using mip;
    display x.l ;
    execute_unload "Case.gdx" x.L ;
    execute 'gdxxrw.exe Case.gdx var=x.L rng=bolge_agr_top!A1' ;
);
);

```

EK-2 Rotalama Aşaması için Yazılan GAMS Yazılımının Kodları

```
option optcr=0;
```

```
option optca=0;
```

```
sets i(*) ;
```

```
alias (i,j);
```

```
Parameter c(i,j), a(i), b(i);
```

```
scalar q/0/;
```

```
scalar w/0/;
```

```
scalar n/0/;
```

```
$CALL GDXXRW Case.xls set=i rng=Bolge1!C3 Rdim=1 par=c
```

```
rng=Bolge1!K2 Rdim=1 Cdim=1 par=a rng=Bolge1!C3 Rdim=1 par=b
```

```
rng=Bolge1!F3 Rdim=1
```

```
$GDXIN Case.gdx
```

```
$LOAD i c a b
```

```
$GDXIN
```

```
c(i,i)=10000000000;
```

```
display i, c;
```

```
loop(i,
```

```
    if( b(i) eq 0,
```

```
        q=q+1
```

```
    );
```

```
        n=n+1
```

```
);
```

```
w= q+1;
```

```
display q, w, n;
```

```
variable z;
```

```
positive variable alfa(i);
```

binary variable $x(i,j)$;

equations

amac1

gecis1

gecis2

gecis3

gecis4

gecis5

depoyadonus

alttur;

amac1..

$\sum((i,j), c(i,j) * x(i,j)) = e = z;$

gecis1(j)\$(ord(j) lt w and ord(j) ne 1)..

$\sum(i$(ord(i) lt w), x(i,j)) = e = 1;$

gecis2(j)\$(ord(j) gt q)..

$\sum(i, x(i,j)) = e = 1 ;$

depoyadonus(j)\$(ord(j) eq 1)..

$\sum(i, x(i,j)) = e = 1 ;$

gecis3(i)\$(ord(i) lt w)..

$\sum(j, x(i,j)) = e = 1 ;$

gecis4(i)\$(ord(i) gt q)..

$\sum(j$(ord(j) gt q or ord(j) eq 1), x(i,j)) = e = 1 ;$

gecis5..

$\sum((i,j)$(ord(i) lt w and ord(j) gt q), x(i,j)) = l = 1 ;$

alftur(i,j)\$ (ord(i) gt 1 and ord(j)gt 1 and ord(i) ne ord(j))..

alfa(i)-alfa(j)+n*x(i,j)=l=(n-1) ;

model rotalama/all/;

solve rotalama minimizing z using mip ;

display x.L, z.L;

execute_unload "Case.gdx" z.L x.l

execute 'gdxxrw.exe Case.gdx var=z.L rng=rota_heuristic!A2 var=x.l

rng=rota1!A1'

EK-3 Test Problemlerinin Özellikleri

No	İsim	N	M	C	K
1	A1	20	5	1550	8
2	A2	20	5	2550	5
3	A3	20	5	4050	4
4	A4	20	5	4050	3
5	B1	20	10	1600	7
6	B2	20	10	2600	5
7	B3	20	10	4000	3
8	C1	20	20	1800	7
9	C2	20	20	2600	5
10	C3	20	20	4150	5
11	C4	20	20	4150	4
12	D1	30	8	1700	12
13	D2	30	8	1700	11
14	D3	30	8	2750	7
15	D4	30	8	4075	5
16	E1	30	15	2650	7
17	E2	30	15	4300	4
18	E3	30	15	5225	4
19	F1	30	30	3000	6
20	F2	30	30	3000	7
21	F3	30	30	4400	5
22	F4	30	30	5500	4
23	G1	45	12	2700	10
24	G2	45	12	4300	6
25	G3	45	12	5300	5
26	G4	45	12	5300	6
27	G5	45	12	6400	5
28	G6	45	12	8000	4
29	H1	45	23	4000	6
30	H2	45	23	5100	5
31	H3	45	23	6100	4
32	H4	45	23	6100	5
33	H5	45	23	7100	4
34	H6	45	23	7100	5
35	I1	45	45	3000	10
36	I2	45	45	4000	7
37	I3	45	45	5700	5
38	I4	45	45	5700	6
39	I5	45	45	5700	7

No	İsim	N	M	C	K
40	J1	75	19	4400	10
41	J2	75	19	5600	8
42	J3	75	19	8200	6
43	J4	75	19	6600	7
44	K1	75	38	4100	10
45	K2	75	38	5200	8
46	K3	75	38	5200	9
47	K4	75	38	6200	7
48	L1	75	75	4400	10
49	L2	75	75	5000	8
50	L3	75	75	5000	9
51	L4	75	75	6000	7
52	L5	75	75	6000	8
53	M1	100	25	5200	11
54	M2	100	25	5200	10
55	M3	100	25	6200	9
56	M4	100	25	8000	7
57	N1	100	50	5700	11
58	N2	100	50	5700	10
59	N3	100	50	6600	9
60	N4	100	50	6600	10
61	N5	100	50	8500	7
62	N6	100	50	8500	8

EK-4 Test Problemlerinin Önerilen Yöntem için CPU Değerleri

No	İsim	CPU							
		Z_1^+	Z_1^-	Z_2^+	Z_2^-	Min ope	İki aşa. yak.	Ağ. max- min	Ağ. top. ope.
1	A1	0.156	0.484	0.234	0.171	0.265	0.234	0.187	0.187
2	A2	0.031	0.062	0.031	0.031	0.078	0.046	0.646	0.109
3	A3	0.094	0.062	0.109	0.078	0.14	0.046	0.093	0.046
4	A4	0.234	0.078	0.109	0.156	0.406	0.14	0.218	0.187
5	B1	1.89	0.64	0.921	0.593	0.781	0.656	1.187	0.781
6	B2	0.62	0.156	0.453	0.14	0.234	0.14	1.078	0.14
7	B3	0.656	0.109	0.062	0.062	1.015	0.14	0.328	0.093
8	C1	1.765	0.812	2.593	3.953	2.937	2.937	2.937	2.937
9	C2	1.734	0.406	1.062	3.781	4.25	2.593	3.046	2.046
10	C3	0.14	0.14	0.093	0.828	2.89	0.25	0.125	0.484
11	C4	0.796	0.062	0.171	1.64	0.578	0.218	0.14	0.156
12	D1	0.921	1.265	0.359	0.296	1.296	1.296	1.125	1.406
13	D2	3.296	0.89	13.203	1.281	5.984	1.062	3.14	1.281
14	D3	0.515	0.531	0.14	1.953	0.625	0.062	0.203	0.046
15	D4	0.718	0.109	0.171	1.593	0.593	0.234	0.218	0.187
16	E1	0.281	2.015	1.937	1.453	3.546	1.218	1.312	0.921
17	E2	1.875	0.093	0.421	0.25	2.875	0.484	1.75	0.531
18	E3	0.39	0.078	0.171	0.14	1.484	0.343	0.203	0.25
19	F1	177.062	1.1	15.962	18.562	61.656	33.39	12.515	23.765
20	F2	59.453	1.859	2.968	8.75	9.64	3.046	2.062	5.546
21	F3	0.187	0.328	0.203	0.5	2.89	0.14	0.906	0.156
22	F4	0.218	1.031	0.203	2.281	4.234	0.203	0.328	0.187
23	G1	162.843	16.671	53.875	41.109	194.45	32.062	247.13	33.062
24	G2	6.046	2.859	5.078	2.375	10.312	9.718	6.484	8.671
25	G3	2.109	1.078	2.984	3.14	24.125	8.703	4.987	7.109
26	G4	0.14	1.031	0.125	0.375	0.296	0.125	0.843	0.093
27	G5	2.812	1.015	0.14	0.171	3.015	0.234	0.281	0.265
28	G6	0.171	1.375	0.093	0.281	4.765	0.765	0.187	0.578
29	H1	3.921	15.625	1.921	11.484	11.093	10.343	6.703	3.968
30	H2	0.593	1.25	0.359	0.218	5.75	1.687	8.296	0.89
31	H3	0.312	0.234	0.234	8.328	2.953	0.296	2.046	0.218
32	H4	0.281	0.234	0.171	0.203	3.796	0.359	6.05	0.22
33	H5	0.203	0.171	0.109	0.656	0.703	0.25	5.45	0.16
34	H6	0.187	0.14	0.125	0.187	3.921	0.265	5.48	0.16

No	İsim	CPU							
		Z_1^+	Z_1^-	Z_2^+	Z_2^-	Min ope	İki aşa. yak.	Ağ. max- min	Ağ. top. ope.
35	I1	29.343	1000	1000	802.671	91.843	23.859	32.875	18.015
36	I2	53.59	10.421	203.718	59.906	128.67	146.22	165.77	123.03
37	I3	33.14	2.234	3.531	51.531	49.688	33.328	26.109	36.39
38	I4	2.671	3.281	0.39	7.781	18.968	0.625	0.765	0.453
39	I5	0.968	3.25	0.625	0.437	8.515	0.828	1.765	0.625
40	J1	358.718	1000	426.968	292.812	346.67	157.27	469.42	128.031
41	J2	13.546	113.265	51.296	446.14	124.52	30.25	89.031	32.734
42	J3	0.75	24.562	14.796	16.187	9.734	0.671	2.14	0.375
43	J4	32.843	57.781	24.078	9.468	63.265	28.75	24.625	2.343
44	K1	71.39	1000	1000	134	736	265.8	734.109	55.562
45	K2	148.421	653.187	12.531	127.484	88.562	73.015	29	51.812
46	K3	2.531	1000	2.765	2.312	124.64	4.437	20.984	2.703
47	K4	58.312	14.312	17.421	3.859	10.906	6.406	21.109	10.843
48	L1	1000	1000	1000	1000	21.546	420.05	7.796	241.03
49	L2	1000	218.546	1000	1000	18.14	1000	15.14	1000
50	L3	258.453	398.109	126.437	642.765	16.734	366.06	14.218	475.88
51	L4	143.5	226.765	126.312	182.843	14.468	1000	6.375	445.58
52	L5	39.15	93.484	3.421	48.187	8.906	2.89	9.265	1.265
53	M1	101.578	452.609	125.281	1000	317.25	115.88	361.39	82.453
54	M2	1000	188.187	287.781	1000	1000	1000	17.062	1000
55	M3	254.531	1000	540.843	192.343	14.484	329.56	268.47	110.921
56	M4	17.859	136.296	133.203	202.968	13.562	166.33	550.55	116.937
57	N1	16.156	1000	1000	614.312	23.843	142.22	18.609	67.609
58	N2	829.125	668.64	1000	335.984	11.703	745	9.171	324.703
59	N3	614.156	1000	64.921	1000	31.453	56.453	18.64	36.828
60	N4	3.562	1000	15.45	891	11.313	4.46	9.14	1.64
61	N5	84.43	31.062	23.5	126.078	7.31	84.93	7.32	30.84
62	N6	10.875	100.281	21.562	66.484	36.515	12.795	33.375	6.39

EK-5 Test Problemlerinin Çözümünde “Bölge Oluşturma” Aşamasında Amaçlara Atanan Ağırlıklar

No	İsim	w_1	w_2
1	A1	0.41	0.59
2	A2	0.58	0.42
3	A3	0.59	0.41
4	A4	0.38	0.62
5	B1	0.43	0.57
6	B2	0.38	0.61
7	B3	0.45	0.55
8	C1	0.37	0.63
9	C2	0.47	0.53
10	C3	0.57	0.43
11	C4	0.47	0.53
12	D1	0.57	0.43
13	D2	0.73	0.27
14	D3	0.48	0.52
15	D4	0.48	0.52
16	E1	0.51	0.49
17	E2	0.51	0.49
18	E3	0.51	0.49
19	F1	0.38	0.62
20	F2	0.44	0.56
21	F3	0.44	0.56
22	F4	0.44	0.56
23	G1	0.52	0.48
24	G2	0.46	0.54
25	G3	0.43	0.57
26	G4	0.49	0.51
27	G5	0.46	0.54
28	G6	0.44	0.56
29	H1	0.46	0.54
30	H2	0.5	0.5
31	H3	0.48	0.52
32	H4	0.49	0.51
33	H5	0.48	0.52
34	H6	0.48	0.52

No	Isim	w ₁	w ₂
35	I1	0.54	0.46
36	I2	0.46	0.54
37	I3	0.49	0.51
38	I4	0.43	0.57
39	I5	0.49	0.51
40	J1	0.54	0.46
41	J2	0.56	0.44
42	J3	0.55	0.45
43	J4	0.49	0.51
44	K1	0.46	0.54
45	K2	0.41	0.59
46	K3	0.47	0.53
47	K4	0.4	0.6
48	L1	0.49	0.51
49	L2	0.46	0.54
50	L3	0.43	0.57
51	L4	0.41	0.59
52	L5	0.5	0.5
53	M1	0.55	0.45
54	M2	0.54	0.46
55	M3	0.47	0.53
56	M4	0.53	0.47
57	N1	0.54	0.46
58	N2	0.56	0.44
59	N3	0.52	0.48
60	N4	0.56	0.44
61	N5	0.48	0.52
62	N6	0.51	0.49

EK-6 Uygulama Probleminin Sonuçları

1. Amaç fonksiyonuna göre elde edilen çözüm:

Mesafe	Rota								
1700	Bilecik	Adana-M2	Bursa-T3	Bilecik					
578	Bilecik	Afyon-M5	Afyon-M4	Kütahya-M70	Bursa-İnegöl-T5	Bilecik			
790	Bilecik	Ankara-M13	Ankara-M6	Ankara-M10	Ankara-M11	Ankara-M9	Bursa-T3	Bilecik	
1570	Bilecik	Ankara-M7	Samsun-M84	Sakarya-M82	Bursa-İnegöl-T5	Bilecik			
626	Bilecik	Ankara-M12	Ankara-M8	Ankara-M14	Eskişehir-T10	Bilecik			
626	Bilecik	Ankara-M14	Eskişehir-T11	Bilecik					
1106	Bilecik	Antalya-M16	Bursa-İnegöl-T5	Bilecik					
1557	Bilecik	Antalya-M17	Antalya-M15	Antalya-M19	Çanakkale-M27	Bursa-İnegöl-T5	Bilecik		
836	Bilecik	Balıkesir-M21	İzmir-M56	Bursa-T3	Bilecik				
190	Bilecik	Bursa-M23	Bursa-İnegöl-T5	Bilecik					
190	Bilecik	Bursa-M23	Bursa-İnegöl-T5	Bilecik					
588	Bilecik	Bursa-M25	Bursa-İnegöl-T5	Bursa-İnegöl-T4	İstanbul-B.Çekmeci-T14	İstanbul-B.Çekmeci-T15	Bilecik		
2428	Bilecik	Diyarbakır-M28	Eskişehir-T11	Bilecik					
2592	Bilecik	Diyarbakır-M28	Bursa-İnegöl-T5	Bilecik					
2592	Bilecik	Diyarbakır-M29	Bursa-İnegöl-T5	Bilecik					
995	Bilecik	Edirne-M30	Edirne-M31	Bursa-İnegöl-T4	Bilecik				
995	Bilecik	Edirne-M32	Bursa-T3	Bilecik					
3440	Bilecik	Elazığ-M33	Diyarbakır-M29	Siirt-M86	Rize-M81	Trabzon-M90	Bursa-T3	Bilecik	
2570	Bilecik	Gaziantep-M34	Gaziantep-Başpınar-T12	Adana-T1	İzmir-T26	Eskişehir-T7	Bilecik		
1946	Bilecik	Gaziantep-M34	Eskişehir-T10	Bilecik					
1922	Bilecik	Ordu-M80	Giresun-M36	Giresun-M35	Giresun-M37	Bursa-T3	Bilecik		
2171	Bilecik	Adana-M3	Adana-M1	Gaziantep-M34	İçel-M41	İçel-M40	İçel-M39	Bursa-İnegöl-T5	
	Bilecik								

Mesafe		Rota					
500	Bilecik	İstanbul-M47	İstanbul-M42	İstanbul-M54	İstanbul-Tuzla - T24	Bilecik	
588	Bilecik	İstanbul-M45	İstanbul-M43	İstanbul-M49	Bursa-T3	Bilecik	
790	Bilecik	Ankara-M12	Bursa-İnegöl-T5	Bilecik			
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik			
500	Bilecik	İstanbul-M48	İstanbul-Tuzla - T25	Bilecik			
500	Bilecik	İstanbul-M50	İstanbul-Tuzla - T25	Bilecik			
500	Bilecik	İstanbul-M46	İstanbul-M53	İstanbul-M44	İstanbul-M50	İstanbul-M51	İstanbul-Tuzla -T25
834	Bilecik	İzmir-M57	İzmir-M55	İzmir-M58	Bursa-İnegöl-T5	Bilecik	
1408	Bilecik	Kayseri-M63	Bursa-İnegöl-T5	Bilecik			
1408	Bilecik	Kayseri-M63	Kayseri-M64	Bursa-T3	Bilecik		
1954	Bilecik	Kayseri-M62	Kahramanmaraş-M61	Kocaeli-M65	Bursa-T3	Bilecik	
836	Bilecik	Konya-M68	Konya-M67	Konya-M69	Eskişehir-T9	Bilecik	
834	Bilecik	İzmir-M59	Manisa-M71	Bursa-İnegöl-T5	Bilecik		
1184	Bilecik	Muğla-M75	Muğla-M74	Muğla-M72	Muğla-M73	Aydın-M20	İzmir-M60
1834	Bilecik	Ordu-M76	Ordu-M78	Ordu-M79	Ordu-M77	Bursa-İnegöl-T5	Bilecik
1408	Bilecik	Kocaeli-M66	Bursa-İnegöl-T5	Bilecik			
1530	Bilecik	Samsun-M83	Bursa-İnegöl-T5	Bilecik			
1530	Bilecik	Samsun-M83	Samsun-M85	Samsun-Kutlukent-T27	Bursa-T3	Bilecik	
1384	Bilecik	Tokat-M88	Tokat-M87	Tokat-M89	Eskişehir-T6	Bilecik	
2196	Bilecik	Trabzon-M93	Trabzon-M91	Trabzon-M92	Bursa-İnegöl-T5	Bilecik	
190	Bilecik	Bursa-T3	Bilecik				
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik			
190	Bilecik	Bursa-T3	Bilecik				

Mesafe		Rota			
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik	
190	Bilecik	Bursa-M24	Bursa-T3	Bilecik	
190	Bilecik	Bursa-T3	Bilecik		
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik	
1700	Bilecik	Adana-M2	Bursa-T3	Bilecik	
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik	
190	Bilecik	Bursa-T3	Bilecik		
588	Bilecik	İstanbul-M47	Bursa-T3	Bilecik	
190	Bilecik	Bursa-T3	Bilecik		
190	Bilecik	Bursa-T3	Bilecik		
1684	Bilecik	İçel-M39	Bursa-T3	Bilecik	
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik	
1530	Bilecik	Samsun-M83	Bursa-T3	Bilecik	
588	Bilecik	İstanbul-M50	Bursa-T3	Bilecik	
190	Bilecik	Bursa-İnegöl-T4	Bilecik		
190	Bilecik	Bursa-İnegöl-T4	Bilecik		
190	Bilecik	Bursa-M22	Bursa-M24	Bursa-İnegöl-T4	Bilecik
190	Bilecik	Bursa-İnegöl-T5	Bilecik		
190	Bilecik	Bursa-İnegöl-T5	Bilecik		
190	Bilecik	Bursa-M26	Bursa-İnegöl-T5	Bilecik	
588	Bilecik	İstanbul-M48	Bursa-İnegöl-T5	Bilecik	
1700	Bilecik	Adana-M3	Bursa-İnegöl-T5	Bilecik	
190	Bilecik	Bursa-İnegöl-T5	Bilecik		
834	Bilecik	İzmir-M55	Bursa-İnegöl-T5	Bilecik	
190	Bilecik	Bursa-İnegöl-T5	Bilecik		
190	Bilecik	Bursa-İnegöl-T5	Bilecik		
1530	Bilecik	Samsun-M84	Bursa-İnegöl-T5	Bilecik	
1106	Bilecik	Antalya-M18	Bursa-İnegöl-T5	Bilecik	
190	Bilecik	Bursa-İnegöl-T5	Bilecik		

Mesafe		Rota			
190	Bilecik	Bursa-İnegöl-T5	Bilecik		
588	Bilecik	İstanbul-M52	Bursa-İnegöl-T5	Bilecik	
190	Bilecik	Bursa-İnegöl-T5	Bilecik		
588	Bilecik	İstanbul-M50	Bursa-İnegöl-T5	Bilecik	
190	Bilecik	Bursa-İnegöl-T5	Bilecik		
190	Bilecik	Bursa-İnegöl-T5	Bilecik		
190	Bilecik	Bursa-İnegöl-T5	Bilecik		
626	Bilecik	Ankara-M12	Eskişehir-T6	Bilecik	
588	Bilecik	İstanbul-M45	İstanbul-Pendik-T20	Bursa-T2	Bilecik
1536	Bilecik	Adana-M3	Eskişehir-T8	Eskişehir-T6	Bilecik
1536	Bilecik	Adana-M2	Eskişehir-T9	Bilecik	
1536	Bilecik	Adana-M2	Eskişehir-T10	Eskişehir-T9	Bilecik
1918	Bilecik	Hatay-M38	Eskişehir-T10	Bilecik	
626	Bilecik	Ankara-M12	Eskişehir-T11	Bilecik	
2428	Bilecik	Diyarbakır-M29	Eskişehir-T11	Bilecik	
1520	Bilecik	İçel-M39	Eskişehir-T11	Bilecik	
1536	Bilecik	Adana-M2	Eskişehir-T11	Bilecik	
1244	Bilecik	Kayseri-M63	Eskişehir-T11	Bilecik	
500	Bilecik	İstanbul-M52	İstanbul-Pendik-T21	İstanbul-K.Çekmeci-T17	Bilecik
500	Bilecik	İstanbul-M47	İstanbul-Tuzla - T24	İstanbul-Tuzla - T23	İstanbul-Ayazağa-T13
500	Bilecik	İstanbul-M50	İstanbul-Tuzla - T24	Bilecik	
500	Bilecik	İstanbul-M43	İstanbul-Tuzla - T24	Bilecik	
500	Bilecik	İstanbul-M52	İstanbul-Tuzla - T24	Bilecik	

Mesafe		Rota						
500	Bilecik	İstanbul-M48	İstanbul-Tuzla - T24	Bilecik				
500	Bilecik	İstanbul-M50	İstanbul-Tuzla - T24	Bilecik				
500	Bilecik	İstanbul-M48	İstanbul-Beykoz-T16	İstanbul-Maltepe-T19	İstanbul-Kartal-T18	İstanbul-Pendik-T22	İstanbul-Tuzla-T25	Bilecik
93552								

2. amaç fonksiyonuna göre elde edilen çözüm:

Mesafe		Rota					
1536	Bilecik	Adana-M2	Eskişehir-T9	Bilecik			
1700	Bilecik	Adana-M3	Bursa-İnegöl-T5	Bilecik			
956	Bilecik	Afyon-M5	İzmir-M60	Balıkesir-M21	Bursa-İnegöl-T5	Bilecik	
1254	Bilecik	Ankara-M8	Kocaeli-M66	Eskişehir-T11	Bilecik		
626	Bilecik	Ankara-M13	Ankara-M14	Ankara-M11	Eskişehir-T10	Eskişehir-T6	Bilecik
790	Bilecik	Ankara-M12	Bursa-İnegöl-T5	Bilecik			
626	Bilecik	Ankara-M12	Eskişehir-T11	Bilecik			
790	Bilecik	Ankara-M12	Bursa-İnegöl-T5	Bilecik			
1134	Bilecik	Afyon-M4	Antalya-M15	Antalya-M17	Bursa-T3	Bilecik	
1106	Bilecik	Antalya-M18	Antalya-M19	Bursa-İnegöl-T5	Bilecik		
995	Bilecik	Edirne-M30	Bursa-T3	Bilecik			
834	Bilecik	Bursa-M26	Bursa-M22	İzmir-M56	Bursa-İnegöl-T5	Bilecik	
190	Bilecik	Bursa-M23	Bursa-T3	Bilecik			
190	Bilecik	Bursa-M24	Bursa-T3	Bilecik			
190	Bilecik	Bursa-M25	Bursa-T3	Bilecik			
2592	Bilecik	Diyarbakır-M28	Diyarbakır-M29	Bursa-İnegöl-T5	Bilecik		
2428	Bilecik	Diyarbakır-M28	Eskişehir-T11	Bilecik			
2592	Bilecik	Diyarbakır-M29	Bursa-İnegöl-T5	Bilecik			
995	Bilecik	Edirne-M31	Edirne-M32	Bursa-İnegöl-T5	Bilecik		
2112	Bilecik	Gaziantep-M34	Kahramanmaraş-M61	Bursa-İnegöl-T4	Bilecik		
1946	Bilecik	Gaziantep-M34	Gaziantep-Başpınar - T12	Eskişehir-T9	Bilecik		
1946	Bilecik	Gaziantep-M34	Eskişehir-T6	Bilecik			
1922	Bilecik	Giresun-M35	Ordu-M77	Ordu-M80	Ordu-M78	Bursa-İnegöl-T4	Bilecik
1922	Bilecik	Sakarya-M82	Giresun-M36	Giresun-M37	Samsun-M85	Bursa-İnegöl-T5	Bilecik
1918	Bilecik	Hatay-M38	Eskişehir-T9	Bilecik			

Mesafe		Rota					
500	Bilecik	İstanbul-M45	İstanbul-Tuzla -T25	Bilecik			
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik			
500	Bilecik	İstanbul-M48	İstanbul-Tuzla -T24	Bilecik			
500	Bilecik	İstanbul-M48	İstanbul-Tuzla -T25	Bilecik			
588	Bilecik	İstanbul-M44	İstanbul-M53	İstanbul-M47	İstanbul-M49	İstanbul-M46	
	İstanbul-M43	İstanbul-M51	Bursa-İnegöl-T5	Bilecik			
588	Bilecik	İstanbul-M50	Bursa-T3	Bilecik			
500	Bilecik	İstanbul-M52	İstanbul-Tuzla -T24	Bilecik			
500	Bilecik	İstanbul-M50	İstanbul-M45	İstanbul-M54	İstanbul-Tuzla -T24	Bilecik	
1066	Bilecik	Aydın-M20	İzmir-M55	İzmir-M57	İzmir-M58	Bursa-T3	Bilecik
1244	Bilecik	Kayseri-M63	Kayseri-M62	Kayseri-M64	Eskişehir-T11	Bilecik	
1408	Bilecik	Kayseri-M63	Bursa-T3	Bilecik			
1244	Bilecik	Kayseri-M63	Eskişehir-T10	Bilecik			
1153	Bilecik	Ankara-M10	Ankara-M6	Konya-M69	Konya-M67	Bursa-T3	Bilecik
1456	Bilecik	Kütahya-M70	Muğla-M75	Muğla-M73	Muğla-M72	Muğla-M74	Çanakkale-M27
	Bursa-İnegöl-T5	Bilecik					
1530	Bilecik	Samsun-M83	Bursa-T3	Bilecik			
2868	Bilecik	Elazığ-M33	Siirt-M86	Tokat-M87	Eskişehir-T11	Bilecik	
1570	Bilecik	Ankara-M12	Ankara-M9	Samsun-M84	Bursa-İnegöl-T5	Bilecik	
2344	Bilecik	Ordu-M79	Trabzon-M90	Rize-M81	Trabzon-M91	Bursa-İnegöl-T5	Bilecik
2196	Bilecik	Ordu-M76	Trabzon-M92	Trabzon-M93	Bursa-T3	Bilecik	
2103	Bilecik	Konya-M68	İçel-M40	İçel-M41	İçel-M39	Adana-M3	Adana-M1
	Adana-T1	Eskişehir-T8	Eskişehir-T7	İstanbul-Tuzla -T23	Bilecik		
190	Bilecik	Bursa-T3	Bilecik				
190	Bilecik	Bursa-T3	Bilecik				
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik			

Mesafe		Rota		
588	Bilecik	İstanbul -M48	Bursa-T3	Bilecik
190	Bilecik	Bursa- T3	Bilecik	
1106	Bilecik	Antalya- M16	Bursa-T3	Bilecik
588	Bilecik	İstanbul -M48	Bursa-T3	Bilecik
588	Bilecik	İstanbul -M43	Bursa-T3	Bilecik
1530	Bilecik	Samsun- M84	Bursa-T3	Bilecik
588	Bilecik	İstanbul -M50	Bursa-T3	Bilecik
190	Bilecik	Bursa- T3	Bilecik	
190	Bilecik	Bursa- T3	Bilecik	
1700	Bilecik	Adana- M3	Bursa-T3	Bilecik
190	Bilecik	Bursa- T3	Bilecik	
190	Bilecik	Bursa- M24	Bursa-T3	Bilecik
190	Bilecik	Bursa- T3	Bilecik	
834	Bilecik	İzmir- M55	Bursa-İnegöl- T4	Bilecik
1700	Bilecik	Adana- M2	Bursa-İnegöl- T4	Bilecik
190	Bilecik	Bursa- İnegöl- T5	Bilecik	
1700	Bilecik	Adana- M2	Bursa-İnegöl- T5	Bilecik
190	Bilecik	Bursa- İnegöl- T5	Bilecik	
588	Bilecik	İstanbul -M50	Bursa-İnegöl- T5	Bilecik
190	Bilecik	Bursa- İnegöl- T5	Bilecik	
190	Bilecik	Bursa- İnegöl- T5	Bilecik	
190	Bilecik	Bursa- İnegöl- T5	Bilecik	
190	Bilecik	Bursa- İnegöl- T5	Bilecik	
588	Bilecik	İstanbul	Bursa-İnegöl-	Bilecik

Mesafe		Rota					
		-M50	T5				
190	Bilecik	Bursa-İnegöl-T5	Bilecik				
190	Bilecik	Bursa-İnegöl-T5	Bilecik				
190	Bilecik	Bursa-M23	Bursa-İnegöl-T5	Bilecik			
790	Bilecik	Ankara-M14	Bursa-İnegöl-T5	Bilecik			
1530	Bilecik	Samsun-M83	Bursa-İnegöl-T5	Bilecik			
190	Bilecik	Bursa-İnegöl-T5	Bilecik				
190	Bilecik	Bursa-İnegöl-T5	Bilecik				
190	Bilecik	Bursa-İnegöl-T5	Bilecik				
190	Bilecik	Bursa-İnegöl-T5	Bilecik				
190	Bilecik	Bursa-İnegöl-T5	Bilecik				
190	Bilecik	Bursa-İnegöl-T5	Bilecik				
588	Bilecik	İstanbul-M48	Bursa-İnegöl-T5	Bilecik			
588	Bilecik	İstanbul-M48	Bursa-İnegöl-T5	Bilecik			
1536	Bilecik	Adana-M2	Eskişehir-T6	Bilecik			
1520	Bilecik	İçel-M39	Eskişehir-T10	Bilecik			
1536	Bilecik	Adana-M2	Eskişehir-T10	Bilecik			
1520	Bilecik	İçel-M39	Eskişehir-T11	Bilecik			
2428	Bilecik	Diyarbakır-M29	Eskişehir-T11	Bilecik			
588	Bilecik	İstanbul-M42	İstanbul-Maltepe-T19	İstanbul-Ayazağa-T13	İstanbul-B.Çekmece-T15	İstanbul-Beykoz-T16	Bursa-T3
500	Bilecik	İstanbul-M48	İstanbul-Tuzla-T25	İstanbul-Kartal-T18	İstanbul-Pendik-T20	Bilecik	
500	Bilecik	İstanbul-M50	İstanbul-Pendik-T21	İstanbul-K.Çekmece-	Bilecik		

Mesafe		Rota				
				T17		
500	Bilecik	İstanbul-M47	İstanbul-Tuzla -T24	İstanbul-B.Çekmece-T14	İstanbul-Pendik-T22	Bilecik
500	Bilecik	İstanbul-M48	İstanbul-Tuzla -T24	Bilecik		
500	Bilecik	İstanbul-M52	İstanbul-Tuzla -T24	Bilecik		
500	Bilecik	İstanbul-M47	İstanbul-Tuzla -T24	Bilecik		
500	Bilecik	İstanbul-M52	İstanbul-Tuzla -T25	Bilecik		
834	Bilecik	Manisa-M71	İzmir-M59	İzmir-T26	Bursa-T2	Bursa-İnegöl-T5
1969	Bilecik	Ankara-M7	Kocaeli-M65	Tokat-M89	Tokat-M88	Samsun-M83
	Samsun-Kutluken-t-T27	Bursa-İnegöl-T4	Bilecik			
93719						

Min operatörü, iki aşamalı yaklaşım ve ağırlıklı max-min modeline göre çözüm:

Mesafe			Rota					
1690	Bilecik	İçel-M41	İçel-M40	Konya-M69	Konya-M67	Bursa-M24	Bursa-İnegöl-T4	Bilecik
834	Bilecik	İzmir-M55	Bursa-T3	Bilecik				
500	Bilecik	İstanbul-M52	İstanbul-Tuzla - T24	Bilecik				
500	Bilecik	İstanbul-M52	İstanbul-Tuzla - T24	Bilecik				
1904	Bilecik	Ankara-M12	Tokat-M88	Tokat-M89	Ordu-M80	Bursa-İnegöl-T5	Bilecik	
626	Bilecik	Ankara-M14	Ankara-M13	Eskişehir-T9	Bilecik			
1184	Bilecik	Muğla-M72	Muğla-M75	İzmir-M57	Bursa-İnegöl-T5	Bilecik		
2106	Bilecik	Ankara-M10	Ankara-M8	Ankara-M7	Giresun-M37	Giresun-M36	İstanbul-M46	İstanbul-M54
	İstanbul-M43	İstanbul-M49	İstanbul-M51	İstanbul-Tuzla - T24	Bilecik			
995	Bilecik	Sakarya-M82	Edirne-M31	Edirne-M30	Bursa-İnegöl-T5	Bilecik		
1744	Bilecik	Konya-M68	Kayseri-M64	Kayseri-M62	İstanbul-M47	İstanbul-Tuzla - T24	Bilecik	
500	Bilecik	İstanbul-M47	İstanbul-Ayazağa-T13	İstanbul-Pendik-T21	İstanbul-K.Çekmeci-T17	Bilecik		
1946	Bilecik	Gaziantep-M34	Gaziantep-Başpınar-T12	Adana-T1	Eskişehir-T8	Eskişehir-T7	Bilecik	
500	Bilecik	İstanbul-M45	İstanbul-Tuzla - T24	Bilecik				
995	Bilecik	Edirne-M32	Bursa-İnegöl-T5	Bilecik				
1134	Bilecik	Afyon-M5	Antalya-M17	Antalya-M15	Bursa-T3	Bilecik		
1918	Bilecik	Hatay-M38	Eskişehir-T11	Bilecik				
500	Bilecik	İstanbul-M48	İstanbul-Tuzla - T25	Bilecik				
1244	Bilecik	Kocaeli-M66	Eskişehir-T11	Bilecik				
190	Bilecik	Bursa-M25	Bursa-İnegöl-T5	Bilecik				
1244	Bilecik	Kayseri-M63	Eskişehir-T11	Bilecik				

Mesafe			Rota		
1244	Bilecik	Kayseri-M63	Eskişehir-T11	Bilecik	
1530	Bilecik	Samsun-M84	Bursa-T3	Bilecik	
2428	Bilecik	Diyarbakır-M29	Eskişehir-T6	Bilecik	
1946	Bilecik	Gaziantep-M34	Eskişehir-T10	Bilecik	
1536	Bilecik	Adana-M2	Eskişehir-T9	Bilecik	
1520	Bilecik	İçel-M39	Eskişehir-T11	Bilecik	
626	Bilecik	Ankara-M12	Eskişehir-T6	Bilecik	
588	Bilecik	İstanbul-M50	Bursa-T3	Bilecik	
1536	Bilecik	Adana-M2	Eskişehir-T10	Bilecik	
1536	Bilecik	Adana-M2	Eskişehir-T10	Bilecik	
2428	Bilecik	Diyarbakır-M29	Eskişehir-T11	Bilecik	
500	Bilecik	İstanbul-M50	İstanbul-Tuzla - T24	Bilecik	
588	Bilecik	İstanbul-M48	Bursa-İnegöl-T5	Bilecik	
588	Bilecik	İstanbul-M50	Bursa-İnegöl-T5	Bilecik	
588	Bilecik	İstanbul-M47	Bursa-İnegöl-T5	Bilecik	
1106	Bilecik	Antalya-M16	Bursa-T3	Bilecik	
588	Bilecik	İstanbul-M53	İstanbul-M50	Bursa-İnegöl-T5	Bilecik
588	Bilecik	İstanbul-M45	İstanbul-M44	Bursa-T3	Bilecik
500	Bilecik	İstanbul-M52	İstanbul-Tuzla - T25	Bilecik	
500	Bilecik	İstanbul-M42	İstanbul-Tuzla - T25	Bilecik	
819	Bilecik	Çanakkale-M27	Balıkesir-M21	Bursa-İnegöl-T5	Bilecik
190	Bilecik	Bursa-T3	Bilecik		
190	Bilecik	Bursa-İnegöl-T5	Bilecik		
834	Bilecik	İzmir-M58	İzmir-M55	İzmir-M60	İzmir-T26
834	Bilecik	İzmir-	Manisa-	Bursa-	Bilecik
					Bursa-T2

Mesafe		Rota					
		M59	M71	İnegöl-T5			
190	Bilecik	Bursa-M22	Bursa-İnegöl-T5	Bilecik			
190	Bilecik	Bursa-T3	Bilecik				
190	Bilecik	Bursa-T3	Bilecik				
1309	Bilecik	Ankara-M9	Ankara-M11	İzmir-M56	Bursa-T3	Bilecik	
190	Bilecik	Bursa-T3	Bilecik				
190	Bilecik	Bursa-T3	Bilecik				
1106	Bilecik	Antalya-M18	Bursa-T3	Bilecik			
190	Bilecik	Bursa-T3	Bilecik				
190	Bilecik	Bursa-M23	Bursa-T3	Bilecik			
190	Bilecik	Bursa-T3	Bilecik				
190	Bilecik	Bursa-T3	Bilecik				
190	Bilecik	Bursa-T3	Bilecik				
1536	Bilecik	Adana-M2	Eskişehir-T11	Bilecik			
2368	Bilecik	Gaziantep-M34	Adana-M1	Adana-M3	İçel-M39	Antalya-M19	Bursa-T3 Bilecik
1743	Bilecik	Tokat-M87	Kocaeli-M65	Bursa-M26	Bursa-T3	Bilecik	
190	Bilecik	Bursa-T3	Bilecik				
1700	Bilecik	Adana-M3	Bursa-T3	Bilecik			
588	Bilecik	İstanbul-M43	Bursa-T3	Bilecik			
2196	Bilecik	Ordu-M77	Trabzon-M91	Trabzon-M93	Bursa-T3	Bilecik	
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik			
2344	Bilecik	Ordu-M79	Rize-M81	Giresun-M35	Ordu-M78	Ordu-M76	Bursa-T3 Bilecik
1530	Bilecik	Samsun-M83	Samsun-M84	Bursa-İnegöl-T4	Bilecik		
790	Bilecik	Ankara-M14	Bursa-İnegöl-T4	Bilecik			
2592	Bilecik	Diyarbakır-M28	Bursa-İnegöl-T4	Bilecik			
588	Bilecik	İstanbul-M48	Bursa-İnegöl-T5	Bilecik			
3012	Bilecik	Kahramanmaraş	Siirt-M86	Diyarbakır-M28	Bursa-İnegöl-T5	Bilecik	

Mesafe		Rota					
		-M61					
1530	Bilecik	Samsun-M83	Bursa-İnegöl-T5	Bilecik			
1684	Bilecik	İçel-M39	Bursa-İnegöl-T5	Bilecik			
190	Bilecik	Bursa-İnegöl-T5	Bilecik				
1530	Bilecik	Samsun-M83	Bursa-İnegöl-T5	Bilecik			
190	Bilecik	Bursa-M23	Bursa-İnegöl-T5	Bilecik			
1700	Bilecik	Adana-M2	Bursa-İnegöl-T5	Bilecik			
588	Bilecik	İstanbul-M48	Bursa-İnegöl-T5	Bilecik			
588	Bilecik	İstanbul-M48	Bursa-İnegöl-T5	Bilecik			
190	Bilecik	Bursa-İnegöl-T5	Bilecik				
790	Bilecik	Ankara-M12	Bursa-İnegöl-T5	Bilecik			
588	Bilecik	İstanbul-M50	Bursa-İnegöl-T5	Bilecik			
190	Bilecik	Bursa-İnegöl-T5	Bilecik				
2236	Bilecik	Ankara-M6	Trabzon-M90	Trabzon-M92	Bursa-İnegöl-T5	Bilecik	
1700	Bilecik	Adana-M3	Bursa-İnegöl-T5	Bilecik			
790	Bilecik	Ankara-M12	Bursa-İnegöl-T5	Bilecik			
1913	Bilecik	Kayseri-M63	Samsun-M85	Bursa-İnegöl-T5	Bilecik		
2592	Bilecik	Diyarbakır-M29	Elazığ-M33	Bursa-İnegöl-T5	Bilecik		
660	Bilecik	İstanbul-M48	İstanbul-Pendik-T22	İstanbul-B.Çekmeci-T15	Eskişehir-T10	Eskişehir-T6	Bilecik
588	Bilecik	İstanbul-M48	İstanbul-Kartal-T18	İstanbul-Pendik-T20	Bursa-İnegöl-T4	Bilecik	
500	Bilecik	İstanbul-M48	İstanbul-Maltepe-T19	İstanbul-Tuzla -T24	İstanbul-Beykoz-T16	Bilecik	
660	Bilecik	İstanbul-M48	İstanbul-Tuzla -T23	Eskişehir-T9	Bilecik		
500	Bilecik	İstanbul-M50	İstanbul-B.Çekmeci-T14	İstanbul-Tuzla -T25	Bilecik		

Mesafe		Rota						
190	Bilecik	Bursa-T3	Bilecik					
190	Bilecik	Bursa-T3	Bilecik					
190	Bilecik	Bursa-İnegöl-T5	Bilecik					
1682	Bilecik	Bursa-M24	Bursa-T3	Samsun-Kutlukent-T27	Bursa-İnegöl-T5	Bilecik		
190	Bilecik	Bursa-İnegöl-T5	Bilecik					
190	Bilecik	Bursa-İnegöl-T5	Bilecik					
1216	Bilecik	Kütahya-M70	Afyon-M4	Muğla-M74	Muğla-M73	Aydın-M20	Bursa-İnegöl-T5	Bilecik
99410								

Ağırlıklı toplam operatörüne göre çözüm :

Mesafe		Rota					
1700	Bilecik	Adana-M2	Bursa-İnegöl-T5	Bilecik			
1700	Bilecik	Adana-M2	Bursa-İnegöl-T5	Bilecik			
1536	Bilecik	Adana-M2	Eskişehir-T9	Bilecik			
1700	Bilecik	Adana-M3	Bursa-T3	Bilecik			
578	Bilecik	Afyon-M5	Afyon-M4	Kütahya-M70	Bursa-İnegöl-T5	Bilecik	
626	Bilecik	Ankara-M6	Ankara-M8	Ankara-M7	Ankara-M13	Eskişehir-T10	Bilecik
790	Bilecik	Ankara-M11	Ankara-M10	Ankara-M14	Bursa-T3	Bilecik	
1570	Bilecik	Ankara-M9	Ankara-M12	Samsun-M84	Bursa-İnegöl-T4	Bilecik	
790	Bilecik	Ankara-M12	Bursa-İnegöl-T4	Bilecik			
626	Bilecik	Ankara-M14	Eskişehir-T6	Bilecik			
1557	Bilecik	Antalya-M15	Antalya-M17	Çanakkale-M27	Bursa-İnegöl-T5	Bilecik	
1106	Bilecik	Antalya-M16	Antalya-M19	Bursa-T3	Bilecik		
1106	Bilecik	Antalya-M18	Bursa-İnegöl-T5	Bilecik			
836	Bilecik	Balıkesir-M21	İzmir-M59	Bursa-İnegöl-T5	Bilecik		
324	Bilecik	Bursa-M22	Bursa-İnegöl-T5	Bursa-İnegöl-T4	Eskişehir-T7	Eskişehir-T10	Bilecik
190	Bilecik	Bursa-M23	Bursa-İnegöl-T5	Bilecik			
190	Bilecik	Bursa-M24	Bursa-İnegöl-T5	Bilecik			
190	Bilecik	Bursa-M26	Bursa-İnegöl-T5	Bilecik			
190	Bilecik	Bursa-T3	Bilecik				
2592	Bilecik	Diyarbakır-M28	Diyarbakır-M29	Bursa-İnegöl-T4	Bilecik		
2592	Bilecik	Diyarbakır-M28	Bursa-İnegöl-T5	Bilecik			
2428	Bilecik	Diyarbakır-M29	Eskişehir-T11	Bilecik			
2428	Bilecik	Diyarbakır-M29	Eskişehir-T11	Bilecik			
995	Bilecik	Edirne-M31	Edirne-M32	Bursa-İnegöl-T5	Bilecik		
3150	Bilecik	Trabzon-M90	Siirt-M86	Elazığ-M33	Eskişehir-T11	Bilecik	

Mesafe			Rota					
2110	Bilecik	Gaziantep-M34	Bursa-T3	Bilecik				
1946	Bilecik	Gaziantep-M34	Gaziantep-Başpınar - T12	Eskişehir-T9	Bilecik			
1922	Bilecik	Ordu-M79	Giresun-M36	Giresun-M35	Giresun-M37	Bursa-İnegöl-T5	Bilecik	
2082	Bilecik	Hatay-M38	Bursa-İnegöl-T4	Bilecik				
1684	Bilecik	İçel-M39	Bursa-İnegöl-T5	Bilecik				
1784	Bilecik	Kocaeli-M65	Adana-M3	Adana-M1	İçel-M41	İçel-M40	İçel-M39	
	Eskişehir-T10	Bilecik						
500	Bilecik	İstanbul-M45	İstanbul-M49	İstanbul-Tuzla -T25	Bilecik			
588	Bilecik	İstanbul-M47	Bursa-İnegöl-T5	Bilecik				
500	Bilecik	İstanbul-M48	İstanbul-Tuzla -T24	Bilecik				
500	Bilecik	İstanbul-M50	İstanbul-Tuzla -T24	Bilecik				
1957	Bilecik	İstanbul-M50	İstanbul-Pendik-T22	İstanbul-Tuzla -T25	Adana-T1	Bilecik		
834	Bilecik	İzmir-M55	İzmir-M56	İzmir-M58	İzmir-T26	Bursa-T2	Bilecik	
1408	Bilecik	Kayseri-M63	Kayseri-M62	Kayseri-M64	Bursa-T3	Bilecik		
836	Bilecik	Konya-M68	Konya-M67	Konya-M69	Eskişehir-T11	Bilecik		
834	Bilecik	İzmir-M57	Manisa-M71	Bursa-İnegöl-T5	Bilecik			
1184	Bilecik	Muğla-M75	Muğla-M74	Muğla-M72	Muğla-M73	Aydın-M20	İzmir-M60	
	Bursa-İnegöl-T5	Bilecik						
1834	Bilecik	Ordu-M76	Ordu-M78	Ordu-M80	Ordu-M77	Bursa-İnegöl-T5	Bilecik	
588	Bilecik	Sakarya-M82	İstanbul-M52	Bursa-İnegöl-T5	Bilecik			
1530	Bilecik	Samsun-M84	Samsun-Kutlukent-T27	Bursa-T3	Bilecik			
1530	Bilecik	Samsun-M83	Samsun-M85	Bursa-T3	Bilecik			
1548	Bilecik	Tokat-M88	Tokat-M87	Tokat-M89	Bursa-T3	Bilecik		
2344	Bilecik	Trabzon-M93	Rize-M81	Trabzon-M91	Trabzon-M92	Bursa-İnegöl-T5	Bilecik	
995	Bilecik	Edirne-M30	Bursa-T3	Bilecik				

Mesafe		Rota		
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik
588	Bilecik	İstanbul-M52	Bursa-T3	Bilecik
190	Bilecik	Bursa-T3	Bilecik	
190	Bilecik	Bursa-T3	Bilecik	
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik
1530	Bilecik	Samsun-M83	Bursa-T3	Bilecik
190	Bilecik	Bursa-T3	Bilecik	
190	Bilecik	Bursa-T3	Bilecik	
190	Bilecik	Bursa-T3	Bilecik	
190	Bilecik	Bursa-T3	Bilecik	
190	Bilecik	Bursa-T3	Bilecik	
190	Bilecik	Bursa-T3	Bilecik	
190	Bilecik	Bursa-T3	Bilecik	
588	Bilecik	İstanbul-M48	Bursa-T3	Bilecik
190	Bilecik	Bursa-T3	Bilecik	
1530	Bilecik	Samsun-M83	Bursa-T3	Bilecik
1408	Bilecik	Kayseri-M63	Bursa-T3	Bilecik
190	Bilecik	Bursa-İnegöl-T5	Bilecik	
190	Bilecik	Bursa-İnegöl-T5	Bilecik	
190	Bilecik	Bursa-M25	Bursa-İnegöl-T5	Bilecik
588	Bilecik	İstanbul-M48	Bursa-İnegöl-T5	Bilecik
190	Bilecik	Bursa-İnegöl-T5	Bilecik	
1684	Bilecik	İçel-M39	Bursa-İnegöl-T5	Bilecik

Mesafe		Rota						
190	Bilecik	Bursa-M23	Bursa-İnegöl-T5	Bilecik				
588	Bilecik	İstanbul-M50	Bursa-İnegöl-T5	Bilecik				
190	Bilecik	Bursa-İnegöl-T5	Bilecik					
588	Bilecik	İstanbul-M52	Bursa-İnegöl-T5	Bilecik				
190	Bilecik	Bursa-İnegöl-T5	Bilecik					
190	Bilecik	Bursa-M24	Bursa-İnegöl-T5	Bilecik				
2112	Bilecik	Gaziantep-M34	Kahramanmaraş-M61	Bursa-İnegöl-T5	Bilecik			
190	Bilecik	Bursa-İnegöl-T5	Bilecik					
190	Bilecik	Bursa-İnegöl-T5	Bilecik					
588	Bilecik	İstanbul-M50	Bursa-İnegöl-T5	Bilecik				
834	Bilecik	İzmir-M55	Bursa-İnegöl-T5	Bilecik				
1536	Bilecik	Adana-M2	Eskişehir-T8	Eskişehir-T6	Bilecik			
626	Bilecik	Ankara-M12	Eskişehir-T6	Bilecik				
1536	Bilecik	Adana-M3	Eskişehir-T9	Bilecik				
1244	Bilecik	Kocaeli-M66	Eskişehir-T10	Bilecik				
1536	Bilecik	Adana-M2	Eskişehir-T11	Bilecik				
626	Bilecik	Ankara-M12	Eskişehir-T11	Bilecik				
1244	Bilecik	Kayseri-M63	Eskişehir-T11	Bilecik				
500	Bilecik	İstanbul-M43	İstanbul-Ayazağa-T13	İstanbul-Pendik-T20	İstanbul-Maltepe-T19	İstanbul-Pendik-T21	İstanbul-Tuzla - T23	
500	Bilecik	İstanbul-M51	İstanbul-M46	İstanbul-M44	İstanbul-M53	İstanbul-M43	İstanbul-M50	
	İstanbul-Kartal-T18	İstanbul - K.Çekmece-T17	İstanbul-Beykoz-T16	Bilecik				

Mesafe		Rota				
500	Bilecik	İstanbul -M45	İstanbul- Tuzla -T24	İstanbul- B.Çekmece- T14	İstanbul- B.Çekmece- T15	Bilecik
500	Bilecik	İstanbul -M47	İstanbul- Tuzla -T24	Bilecik		
500	Bilecik	İstanbul -M48	İstanbul- Tuzla -T24	Bilecik		
500	Bilecik	İstanbul -M47	İstanbul- M42	İstanbul- M54	İstanbul- Tuzla -T24	Bilecik
500	Bilecik	İstanbul -M48	İstanbul- Tuzla -T24	Bilecik		
500	Bilecik	İstanbul -M48	İstanbul- Tuzla -T25	Bilecik		
500	Bilecik	İstanbul -M50	İstanbul- Tuzla -T25	Bilecik		
94154						