

T.C.
DOKUZ EYLÜL ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
EKONOMETRİ ANABİLİM DALI
EKONOMETRİ PROGRAMI
YÜKSEK LİSANS TEZİ

ÇOK AMAÇLI HEDEF PROGRAMLAMANIN KARAR
PROBLEMLERİNE KATKISI VE BİR ÜRETİM
UYGULAMASI

Melike GÜNGÖR

Danışman
Prof. Dr. Mustafa GÜNEŞ

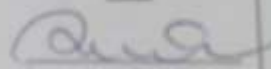
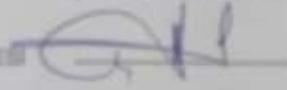
İZMİR – 2015

YÜKSEK LİSANS
TEZ PROJESİ ONAY SAYFASI

Üniversite : Düzce Eylül Üniversitesi
Enstitü : Sosyal Bilimler Enstitüsü
Adı ve Soyadı : Melike GÜNGÖR
Tez Başlığı : Çok Amaçlı Hedef Programlarının Karar Problemlerine Katkısı ve Bir Üretim Uygulaması
Savunma Tarihi : 23/07/2015
Danışmanı : Prof.Dr. Mustafa GÜNEŞ

2009800181

JÜRI ÜYELERİ

<u>Ünvanı, Adı, Soyadı</u>	<u>Üniversitesi</u>	<u>İmza</u>
Prof.Dr. Mustafa GÜNEŞ	GEDİZ ÜNİVERSİTESİ	
Doç.Dr. Mehmet AKSARAYLI	DÜZCE EYLÜL ÜNİVERSİTESİ	
Yrd.Doç.Dr. Şenel DEMİROĞLU	GEDİZ ÜNİVERSİTESİ	
Oybirliği (✓)		
Oy Çokluğu ()		

Melike GÜNGÖR tarafından hazırlanmış ve sunulmuş "Çok Amaçlı Hedef Programlarının Karar Problemlerine Katkısı ve Bir Üretim Uygulaması" başlıklı Tez () / Projesi () kabul edilmiştir.

Prof.Dr. Utku UTKULU
Enstitü Müdürü

YEMİN METNİ

Yüksek Lisans Tezi olarak sunduğum **“Çok Amaçlı Hedef Programlamanın Karar Problemlerine Katkısı ve Bir Üretim Uygulaması”** adlı çalışmanın, tarafımdan, akademik kurallara ve etik değerlere uygun olarak yazıldığını ve yararlandığım eserlerin kaynakçada gösterilenlerden oluştuğunu, bunlara atıf yapılarak yararlanılmış olduğunu belirtir ve bunu onurumla doğrularım.

Tarih

.../.../.....

Melike GÜNGÖR

İmza

ÖZET

Yüksek Lisans Tezi

Çok Amaçlı Hedef Programlamanın Karar Problemlerine Katkısı ve Bir Üretim Uygulaması

Melike GÜNGÖR

Dokuz Eylül Üniversitesi

Sosyal Bilimler Enstitüsü

Ekonometri Anabilim Dalı

Ekonometri Programı

Karar alma fonksiyonu, gerek kişisel yaşam gerekse işletmeler için büyük önem arz etmektedir. Birçok işletmenin ulaşmak istediği birden çok hedefi vardır. Bu hedeflerine ulaşırken verecekleri kararlarda çoğu zaman net rakamsal verilere değil, belirsizlik içeren verilere sahiptirler. İşletmeler hedeflerine, eldeki belirsiz verilerle işletme için hayati önem taşıyan kararlar alırlar. Bu kararları alırken bütün hedeflerinin tam olarak gerçekleşmesini yani optimal çözümü isterler. Ancak bu çoğu zaman mümkün olmamaktadır. Bununla birlikte hedef ve kısıtlardaki belirsizlikleri kullanarak bulanıklık modeller ile optimum çözümü bulmak mümkündür. Bunun için 1965 yılında L. A. Zadeh tarafından, belirsizliği incelemek için kullanılan ve çoklu mantığa dayalı olan, bulanık küme teorisi geliştirilmiştir. Bu teori günümüzde de birçok problemin çözümünde kullanılır hale gelmiştir.

Bu çalışmada İzmir’de yağ dolumunu ve satışını yapan bir işletmenin, mısır yağını işletme hedeflerine uygun olarak ne kadar üretmesi gerektiği bulanık mantık felsefesi ile Minmaks Hedef Programlama ve Bulanık Hedef Programlamanın MA yaklaşımı ile ayrı ayrı çözülmüştür ve sonuçlarının karşılaştırması yapılmıştır. Çıkan sonuçların yaklaşık olarak aynı olduğu gösterilmiş ve bu sonuçlar yorumlanarak işletmenin hedeflerine uygun önerilerde bulunulmuştur.

Anahtar Kelimeler: Hedef Programlama, Bulanık Hedef Programlama, Bulanık Mantık, Karar Verme.

ABSTRACT

Master Thesis

The Contribution of Multi-Objective Goal Programming for Decision Problems and an Application in Production Process

Melike GÜNGÖR

Dokuz Eylül University

Graduate School of Social Science

Department of Econometrics

Econometrics Program

Decision making mechanism has importance for both personal life and companies. Companies have goals to reach. Most of the time on course to reach these goals decisions that they have to make is related to uncertain data rather than explicit figures. They have to give vital decisions for their self's sake with those uncertain datas. By giving these decisions they expect optimal results in order to reach their goals, however usually, that is not exactly possible. Besides by using uncertainties and constraints on the objective, it is possible to develop fuzzy logic to find optimum solution. In order to evaluate the uncertainty; fuzzy set theory, which is predicated on many-valued logic, had developed by L. A. Zadeh in 1965. Nowadays this theory is being used to solve various problems.

In this work, corn oil amount that an oil company in İzmir has to produce for company goal had evaluated by using both “fuzzy logic in Minmaks Goal Programming” and “Fuzzy Goal Programming – MA approach” and the results have compared. It is showed that the results are close to each other and by commenting on these results suggestions had been made for the company's goals.

Keywords: Multi-Objective Goal Programming, Fuzzy Goal Programming, Fuzzy Logic, Decision-Making Process.

ÇOK AMAÇLI HEDEF PROGRAMLAMANIN KARAR PROBLEMLERİNE KATKISI VE BİR ÜRETİM UYGULAMASI

İÇİNDEKİLER

TEZ ONAY SAYFASI	ii
YEMİN METNİ	iii
ÖZET	iv
ABSTRACT	vi
İÇİNDEKİLER	vii
KISALTMALAR	x
TABLolar LİSTESİ	xi
ŞEKİLLER LİSTESİ	xii
 GİRİŞ	 1

BİRİNCİ BÖLÜM TEMEL KAVRAMLAR

1.1. KARAR VERME SÜRECİ VE BULANIK MANTIK	3
1.1.1. Karar Verme Sürecinin Adımları	5
1.1.2. Belirsizlik Altında Karar Verme ve Bulanık Mantığın Karar Verme Süreçlerine Katkıları	6
1.1.3. Bulanık Küme ve Temel Kavramlar	10
1.1.3.1. Üyelik Fonksiyonu	10
1.1.3.2. Bulanık Küme	10
1.1.3.3. α -Kesim Seviyesi	11
1.1.3.4. Kümenin Desteği	11
1.1.4. Bulanık Küme İşlemleri	12
1.1.4.1. Birleşim İşlemi	12
1.1.4.2. Kesişim İşlemi	13

1.1.4.3. Tümlleme İşlemi	13
1.1.5. Bulanık Sayılar ve Üyelik Fonksiyonları	15
1.1.5.1. Üçgensel Bulanık Sayı ve Üyelik Fonksiyonu	15
1.1.5.1.1. Toplama	16
1.1.5.1.2. Çıkarma	16
1.1.5.1.3. Çarpma	16
1.1.5.1.4. Bölme	16
1.1.5.2. Yamuk Tipli Bulanık Sayı ve Üyelik Fonksiyonu	17
1.1.5.2.1. Toplama	17
1.1.5.2.2. Çıkarma	18
1.1.5.2.3. Çarpma	18
1.1.5.2.4. Bölme	18
1.2. BULANIK ORTAMDA KARAR VERME	18
1.2.1. Bulanık Hedef	19
1.2.2. Bulanık Kısıt	19
1.2.3. Bulanık Karar	19

İKİNCİ BÖLÜM

HEDEF PROGRAMLAMA VE BULANIK HEDEF PROGRAMLAMA

2.1. KARAR VERMEDE ÇÖZÜM TİPLERİ	21
2.1.1. Optimal Çözüm	22
2.1.2. İdeal Çözümler	23
2.1.3. Üstün Olmayan Çözüm	23
2.2. HEDEF PROGRAMLAMA	24
2.2.1. Hedef Programlama ile İlgili Kavramlar	26
2.2.2. Hedef Programlama Formülasyonu	27
2.2.2.1. Ağırlıklı Hedef Programlama	31
2.2.2.2. Öncelikli Hedef Programlama	31
2.2.2.3. Minmaks Hedef Programlama	32
2.2.3. Hedef Programlama Modelinde Normalizasyon	33
2.2.3.1. Yüzde Normalizasyon	33

2.2.3.2. Sıfır-Bir Normalizasyon	34
2.2.3.3. Öklit Normalizasyon	34
2.2.3.4. Toplam Normalizasyon	34
2.2.3.5. Ölçekleme Fonksiyonu İle Normalizasyon	34
2.2.4. Bulanık Hedef Programlama Yaklaşımları	36
2.2.4.1. Narasimhan Yaklaşımı	37
2.2.4.2. Hannan Yaklaşımı	38
2.2.4.3. MA Yaklaşımı	39

ÜÇÜNCÜ BÖLÜM

UYGULAMA

3.1. KARAR PROBLEMİ	43
3.2. UYGULAMANIN ÇÖZÜM AŞAMALARI	47
3.2.1. İdeal Çözümlerinin Belirlenmesi	48
3.2.2. Minmaks Hedef Programlama Çözümü	49
3.2.3. Bulanık Hedef Programlama Çözümü	51
3.3. MİNMAKS HEDEF PROGRAMLAMA İLE BULANIK HEDEF PROGRAMLAMA SONUÇLARININ KARŞILAŞTIRILMASI	53
SONUÇ	55
KAYNAKÇA	58

KISALTMALAR

ÇADP	Çok Amaçlı Doğrusal Programlama
ÇAKV	Çok Amaçlı Karar Verme
ÇKKV	Çok Kriterli Karar Verme
ÇNKV	Çok Nitelikli Karar Verme
HP	Hedef Programlama
s.	Sayfa No
OR	Operational Research, Yöneylem Araştırması
GDO	Genetiği Değiştirilmiş Organizmalar

TABLÖLAR LİSTESİ

Tablo 1: Olaylar Arasındaki Sebep Sonuç İlişkisi	s. 7
Tablo 2: Klasik Mantık-Bulanık Mantık Arasındaki Temel Farklılıklar	s. 8
Tablo 3: Bulanık Denetim Uygulamaları	s. 9
Tablo 4: Hedef Tipine Göre Minimize Edilecek Değişken	s. 30
Tablo 5: Ambalajlı Mısır Yağlarının Dolum Hatları ve Dolum Süreleri	s. 43
Tablo 6: Ürünlerin Hacimleri(m^3)	s. 44
Tablo 7: Farklı Ambalajlı Ürünlerin 2013-2014 Yılları Mayıs Satış Rakamları	s. 45
Tablo 8: Farklı Ambalajlı Ürünlerin Birim Maliyet-Satış Fiyatı Tablosu(TL)	s. 46
Tablo 9: İdeal Çözüm Değerleri	s. 48
Tablo 10: Her Bir Amaç Fonksiyonuna Ait Karar Değişkenleri	s. 48
Tablo 11: Minmaks Yaklaşımı Çözümüne Göre Karar Değişkeni Değerleri	s. 50
Tablo 12: Minmaks Hedef Programlama Çözümü	s. 51
Tablo 13: Bulanık Yaklaşım Çözümüne Göre Karar Değişkenleri	s. 52
Tablo 14: Bulanık Hedef Programlama Çözümü	s. 53
Tablo 15: Minmaks Hedef Programlama ve Bulanık Hedef Programlama Çözümleri	s. 53
Tablo 16: Minmaks Hedef Programlama ve Bulanık Hedef Programlamaya Göre Amaç Fonksiyonlarının Değerleri	s. 54

ŞEKİLLER LİSTESİ

Şekil 1: Karar Verme Süreci Tablosu	s. 4
Şekil 2: Kümenin Desteği, Özü ve Sınırı	s. 12
Şekil 3: Birleşim ve Kesişim İşlemleri	s. 13
Şekil 4: Bulanık bir A kümesinin Tümleyeni	s. 14
Şekil 5: Bulanık Kümelerin Üyelik Fonksiyonu	s. 14
Şekil 6: Bulanık Üçgensel Üyelik Fonksiyonu	s. 15
Şekil 7: Yamuk Tipli Bulanık Sayı Üyelik Fonksiyonu	s. 17
Şekil 8: Üçgensel Üyelik Fonsiyonu	s. 37
Şekil 9: Bulanık Hedefler İçin Doğrusal Üyelik Fonksiyonları	s. 41

GİRİŞ

İnsan, hayatının her döneminde küçük veya büyük ölçekte karar vermek zorundadır. Küreselleşen ekonomide acımasız şartlarda üretim yapan işletmeler için ise doğru ve hızlı karar vermek son derece önemlidir. Karar vericilerin işletmeler için karar vermek zorunda olduğu problemlerden birkaçı: üretim planlama, pazarlama yönetimi, zaman yönetimi, alınacak ürün seçimi, kullanılacak materyal seçimi, tesis için yer seçimi, tedarikçi seçimi, kar ve verimlik artışı, maliyeti düşürme vb. şeklinde sıralayabiliriz. Özellikle büyük ölçekte ve karmaşık yapıdaki problemler için işletmeler karar alırken belirsiz, net olmayan ve göreceli verilere sahiptirler.

İşletmelerde karar vericiler problemi ne kadar detaylı tanımlarlar ve iyi analiz ederlerse o kadar geçerli sonuçlara ulaşırlar. Ancak problemlerin çözümü için genelde klasik yöntemleri kullanabilmek adına bazı veriler görmezden gelinip problem belli kalıplara sıkıştırılarak çözülmekte böylece sonuçları işletme için yetersiz olup işletmenin hedeflerini tatmin etmeyecektir. Daha detaylı ve dikkatli araştırmalar sonucunda elde edilen esnek verilerle yapılan üretim planlama sayesinde işletmeler gerek karın maksimize edilmesi, gerekse maliyetin minimize edilmesinin yanı sıra işletmenin sürdürülebilir olmasını sağlayacak prestiji de elde etmiş olurlar.

Hem üretim hem de satış yapan işletmeler üretim yapacakları miktarları, önceki yıllarda yaptıkları üretim ve satışları göz önünde bulundurarak ve de kişisel gözlemlerden yola çıkarak belirsizlik içeren verilerle oluştururlar. Bu verileri klasik Karar Verme Problemlerinde çözümlmek pek de mümkün değildir.

Gerçek hayata en yakın olan matematiksel karar verme yöntemlerinden olan Hedef Programlama ve Bulanık Hedef Programlama modelleridir. Bu iki programlamada da veriler esnek yapıdadır, yani bir şeyi olur ya da olmaz olarak nitelendirmek yerine olup olmaması konusunda derecelendirme yapmamıza olanak sağlar. Böylece karar verici daha esnek yapıda hedef ve kısıtlarını belirleyebilir.

Çalışmanın temel amacı farklı ağırlıklarda ambalajlanan mısır yağının üretim ve yüksek oranda toptan satışını yapan ve de hedeflerini optimum düzeyde gerçekleştirmek isteyen bir işletmenin üretim planlama problemi için Hedef Programlama ve Bulanık Hedef Programlama yaklaşımlarını araştırıp problemde uygulanarak karşılaştırılmasıdır.

Problemin çözümü ve programlama modellerinin araştırılması için yapılan bu çalışma üç bölümden oluşmaktadır. Birinci bölümde Karar Verme Süreci temel tanımları ve Bulanık Mantığın Karar Verme Sürecine katkıları araştırılmış, ikinci bölümde Hedef Programlamanın tanımı ve formülasyonu üzerinde durulmuş ayrıca Bulanık Hedef Programlama incelenmiştir. Üçüncü bölümde ise bir işletmenin üretim planlamasına dair problemi Minmaks Hedef Programlama ve Bulanık Hedef Programlamanın MA yaklaşımları ile çözümlenmiş ve sonuçları karşılaştırılıp işletme için yorumlanmıştır.

BİRİNCİ BÖLÜM

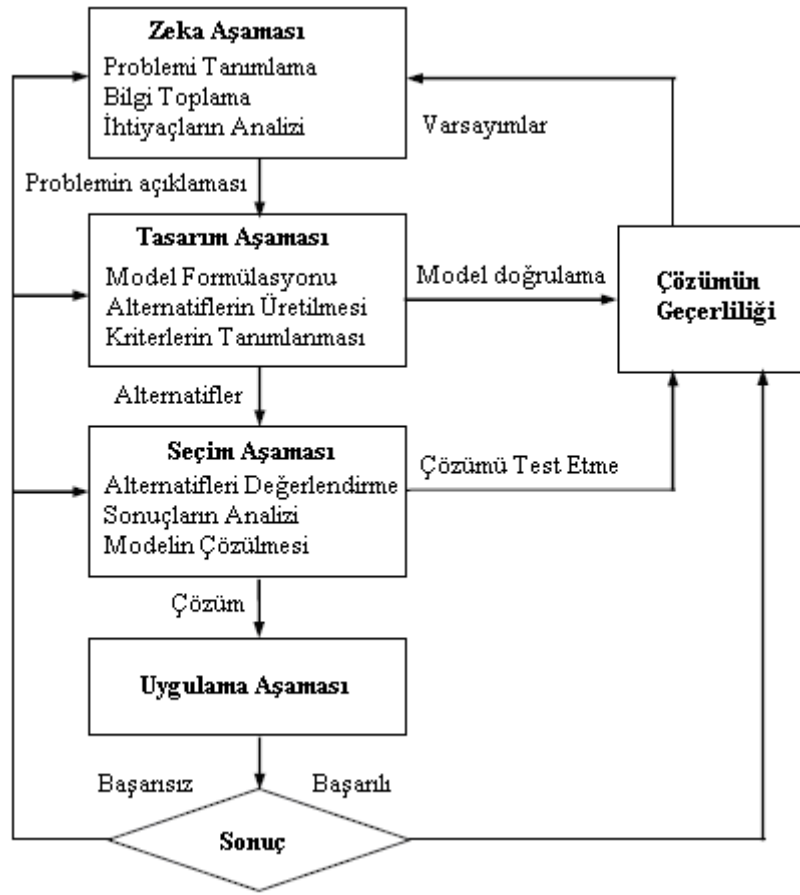
TEMEL KAVRAMLAR

1.1. KARAR VERME SÜRECİ VE BULANIK MANTIK

Karar verme, alternatifler arasında eylemlerin sıralamasının seçimi için yol gösteren bilişsel bir süreçtir. Her karar verme süreci, nihai seçimi üretir. Genel olarak, karar süreci bir çözüme ihtiyaç duyduğumuz fakat karar vericiler tarafından kabul edilen çözümün nasıl ve ne zaman olduğunun bilinmediği durumlarda başlar. Karar verme, aynı zamanda mantık süreci(rasyonel veya irrasyonel olabilir) olarak da düşünülebilir ve tanımlı veya tanımsız varsayımlar içerebilir.

Sistematik bir karar verme süreci, 1977 yılında Simon tarafından kurulmuştur ve üç aşamadan oluşur: Zeka, Tasarım ve Seçim aşamalarıdır. Dördüncü aşama olan Uygulama aşaması ise sonradan eklenmiştir. Şekil 1, karar verme sürecinin kavramsal bir tablosunu göstermektedir. Karar verme süreci, gerçeğin gözden geçirildiği Zeka aşamasıyla başlar. Bu aşamada problem tespit edilir ve problem ifadesi tanımlanır. Tasarım aşamasında, inşa edilen sistemi temsil eden bir model oluşturulur. Bu model bütün değişkenler arasındaki ilişkiyi yazarak ve gerçeği basitleştiren varsayımlar yapılarak düzenlenir. Daha sonra bu modelin geçerliliği denetlenir ve kısıtları, tanımlanan eylemlerin alternatif yollarının hesaplanması için koyulur. Çoğu kez, model yapım süreci, potansiyel alternatif çözümleri ve tam terslerini de tanımlar. Seçim aşaması ise model için önerilen çözümün seçimini içerir. Bu çözüm, uygulanabilirliğini belirlemek için test edilir. Önerilen çözüm mantıklı görünüyorsa, son aşama olan uygulama aşamasına geçilir. Başarılı bir uygulama, gerçek problemin çözümüyle sonuçlanır. Başarısızlık, sürecin daha önceki bir aşamasına geçmemize yol açar (aktaran Lu ve diğerleri, 2007: 5-6).

Şekil 1: Karar Verme Süreci Tablosu



Kaynak: Lu ve diğerleri, 2007: 6

Süreçten, kararın çeşitli alternatifler arasından bir seçim olduğu görülebilir. Her karar; bir problem ifadesi tarafından alternatiflerin bir kümesi ve karar kriterleri ile karakterize edilebilir. Karar vericiler, bir karara ulaşma sürecinde bütün bu aşamalardan geçerler. Karar verme sürecinin herhangi bir ortak tanımı yoktur. Fakat sistematik bir karar verme süreci, karar vermenin bütün durumlarında uygun karşılık ve bilgisayarlı destekle katkıda bulunarak anlamamızda yardım sağlar (Lu ve diğerleri, 2007: 39).

Karar verme süreci, işletmelerin hayati önem taşıyan fonksiyonudur. Günümüz modern örgütlerinde karar verme süreci; bireysel olmaktan çıkıp yöneticiyi aşan, grup, ekip ve hatta bilgisayar destekli bir süreç haline gelmiştir. Bu nedenle karar verme süreçlerindeki başarı; etkin ekip çalışmalarının dinamikliğine bağlıdır.

Klasik çözüm algoritmalarının yetersiz kaldığı modellerde kesinlikle bulanık mantığın matematiği devreye girmeli ve alternatif çözümler üretilmelidir.

Bulanık mantık ilk defa 1960 yılında, University of California, Berkeley'den Dr. Lotfi Zadeh tarafından, doğal dildeki belirsizliği modellemek için ortaya konmuştur. Zadeh, bulanık mantık teorisinin bağımsız ve tam bir teori olmaktan çok, bulanıklaştırma yönteminin, herhangi bir teorisin ayrık formdan sürekli forma dönüştürülmek suretiyle genelleştirilmesi için kullanılan bir metodoloji olarak ele alınmasını istemiştir (Zadeh, 1973: 38). Zadeh'e göre bulanık mantık çoklu değerliliklidir. Bulanık mantık, bir alternatifi iyi kötü, güzel çirkin, doğru yanlış, var yok, zor kolay veya bir kümeye ait "1" - ait değil "0" şeklinde sınıflandırmak yerine gerçek hayatta olduğu gibi iyi ve kötü arasında biraz iyi, biraz kötü, çok iyi, çok kötü gibi dilsel doğruluk dereceleri ile sınıflandırır yani klasik mantığın 0-1 önermelerine karşılık Bulanık Mantık, üç veya daha fazla sayıda önerme oluşturur. Bu da karar verme sürecinde daha esnek ve doğru kararlar almamızı sağlar.

Bulanık Mantık, bilimsel terminolojide ve teknolojide "Fuzzy Logic" kelimelerinin karşılığı olarak kullanılmaktadır. Bulanık Mantık Teorisi, düşünsel ve kavramsal işlev ve görelî sınıflandırılmış üyelik derecesi temeline dayanması nedeniyle, bilgisayarlarda ve bilgisayar destekli tasarımlarda rahatlıkla uygulanabilmesi bakımından büyük önem taşımaktadır (Güneş ve Yiğitbaşı'dan aktaran Ertuğrul, 2005: 48).

1.1.1. Karar Verme Sürecinin Adımları

Genel karar verme süreci tablosunda, farklı karar vericiler, aşamalara farklı ağırlıklar verebilir. Farklı karar verme problemleri, bir veya daha fazla aşamasında daha fazla bilgi veya alt aşamalar ve destek teknikler gerektirebilir. Karar vericilere karar verme sürecini anlama ve kolay bir şekilde takip etmeleri için, Şekil 1 'deki dokuz adım aşağıdaki gibidir (Lu ve diğerleri, 2007: 7-9):

Adım 1: Karar Probleminin Tanımlanması

Adım 2: Analiz İhtiyaçlarının Belirlenmesi

Adım 3: Amaç ve Hedeflerin Kurulması

Adım 4: Alternatiflerin Üretilmesi

Adım 5: Gerekliyse Kriterlerin Belirlenmesi

Adım 6: Bir Karar Verme Yöntemi veya Aracının Seçilmesi

Adım 7: Kriterlere göre Alternatiflerin Değerlendirilmesi

Adım 8: Problem İfadesine göre Çözümlerin Onaylanması

Adım 9: Probleme Uygulanması

1.1.2. Belirsizlik Altında Karar Verme ve Bulanık Mantığın Karar Verme Süreçlerine Katkıları

Son yıllarda yönetsel karar vermede görülen köklü değişim, kantitatif tekniklerin başarılı uygulanışı ile ilgili işlemlerin geniş kullanımına bağlanabilir. Günümüz toplumu farklı ve karmaşık kuruluşlardan oluşmaktadır. Karmaşık kuruluşlar da karmaşık karar problemlerini doğururlar. İşletmelerin karşılaştığı karmaşık karar problemlerinin çözümü yöneticilere düşer. Fakat kararlara etki eden koşulları ele almak ve tüm elverişli karar seçeneklerinden beklenen sonuçları belirlemek de pek kolay değildir. İşte bu noktada iyi bir karara ulaşılabilmesi amacıyla problemin sağlıklı analizi ve araştırılması için bilimsel yöntemler kullanılmalıdır (Öztürk, 1987: 1). Bu yöntemler kullanılırken karar vericiden belirsizlikler altında ‘hedeflenene en yakın’ kararlar vermesi beklenir. Günlük hayatta ve organizasyonlarda karar verme davranışının ortaya çıkış sürecinde kısıtlı ve kişisel bilgilere dayanan ya da geleceğe yönelik tahminde bulunarak elde edilen yeni veriler kullanılarak alternatif politikalar üretilir. Üretilen bu alternatif politikaların doğruluğu ya da uygulanabilirliği, söz konusu kararların istatistiksel modellere uygunluğu ile ölçülebilmektedir (Kökdemir, 2003: 4). Bu nedenle sayısal teknikler içinde yer alan modelleme çalışmaları her zaman kullanılmalıdır. Kullanılan verilerden biri belirli bir birim değiştiği zaman, diğer değişkenlerin nasıl bir tepki verdiğini de incelememiz gerekir, bunun için de regresyon analizi kullanılır. Regresyon analizi, aralarında sebep-sonuç ilişkisi bulunan iki veya daha fazla değişken arasındaki ilişkiyi belirlemek ve bu ilişkiyi kullanarak o konu ile ilgili tahminler ya da kestirimler yapabilmek amacıyla yapılır. Doğada, aşağıda olduğu gibi birçok bağımlı ve bağımsız değişken arasında sebep sonuç ilişkisine rastlamak mümkündür. Bu ilişki Tablo 1’ deki gibidir.

Tablo 1: Olaylar Arasındaki Sebep Sonuç İlişkisi

Sebep	Sonuç
Gelir	Harcama
Yaş	Boy
Gübre	Verim
Çalışma süresi	Alınan not

Yukarıda sıralanan modeller belirlendikten sonra, modelin parametrelerinin bulandırılması ve bulandırılmış modellerle alternatif çıktılar türetilmesi karar verme pozisyonunda bulunan kişi ya da kurumlara daha esnek davranma imkanı sağlar. Bu nedenle gerek regresyon modellerinde ve gerekse alternatif model tabanlı çalışmalarda bulanık mantığın katkıları mutlaka kullanılmalıdır. Örneğin kalite kontrol çalışmalarında klasik algoritmalarla belirlenen kontrol sınırları bulanık mantığın katkısı ile esnetilerek üretim süreçlerindeki riskler azaltılmaktadır. Bu çalışmaların yanı sıra yüksek yoğunluktaki trafik hatlarının optimizasyonu ve sermaye piyasalarında faaliyet gösteren yatırımcıların yatırım kararlarındaki risklerin analizi ve minimizasyonu da diğer uygulamalardır. Bütün bunlar bulanık mantığın insan merkezli karar problemlerinin getirdiği katkılardan sadece birkaçıdır. Bu alandaki uygulama sayısı her geçen gün artmakta olup yeni geliştirilen teknikler de farklı alanlarda uygulanmaktadır.

Klasik karar verme süreci, belirsizlikler altında gerçekleşecektir ve esneme payı yoktur. Bu noktada genel anlamda karar süreçlerinde belirsizliğin nasıl öngörüleceği ve nasıl karar süreçlerinin bir parçası haline getirilebileceği yolunda çalışmalar yapılmış ve bu çalışmalar sonucunda alternatif bilimsel yaklaşım düşüncesi ortaya atılmıştır. Bu süreçteki son nokta Loutfi Zadeh'in Bulanık Mantık Teorisi olmuştur. Klasik mantık ile bulanık mantık arasındaki temel farklılıklar Tablo 2'de gösterilmiştir.

Tablo 2: Klasik Mantık-Bulanık Mantık Arasındaki Temel Farklılıklar

Klasik Mantık	Bulanık Mantık
A <u>veya</u> A Değil	A-A Değil
Kesin	Kısmi
Hepsi veya Hiçbiri	Belirli Derecelerde
0 veya 1	0 ve 1 Arasında Süreklilik
İkili Birimler	Bulanık Birimler

Kaynak: Yaralıoğlu, 2004: 1

Bulanık mantık uygulamaları ilk olarak çimento sektöründe kullanılmaya başlanmıştır. Bu sektörde kireç taşı ve kil 1000-1400 derece sıcaklıkta reaksiyona girmektedir. Fırın içindeki sıcaklık ve oksijen oranı çimentonun kalitesini doğrudan etkilemektedir. Sadece bu konuda uzman operatörler, istenilen sınırlar dahilinde ürün elde edebilmektedirler. Ama vardiyalı bir sistemle çalışan bu fabrikada çok sayıda operatör vardır ve her operatörün uzmanlıklarının farklı olması nedeniyle farklı niteliklerde ve verimlilikte ürün elde edilmektedir. İstenilen kalitede ürün sadece bu işte yıllardır çalışan uzmanlar tarafından sağlanabilmektedir. Zira çimento üretimi bulanık bir yapıya sahiptir ve süreç kontrolünü bulanık kurallar sağlamaktadır. Örneğin ısıyı 10 derece yükselt veya 5 derece azalt gibi kesin kurallar değil biraz azalt, biraz yükselt gibi bulanık terimlerle ifade edilen kurallarla kontrol edilmektedir. Bir Danimarka firması bu sürecin kontrolü için uzman operatörlerin kullandığı 50-60 pratik kuraldan hareketle bir mikro kontrolör oluşturmuşlar ve sonuç olarak sabit ürün kalitesi ve yakıtta büyük tasarruf elde etmişlerdir. Daha sonraları bulanık mantık, insansız uçakların kontrolünde, tren frenleme sistemlerinde, ABS (otomatik fren sistemi) ve ASC (otomatik vites kontrolü) kontrolünde kullanılmıştır. Bulanık mantık, makinelere karar verme, algılama, öğrenme, değerlendirme ve çözüm üretme gibi insan zekasına benzer özellikler yükleyerek makineleri olduğundan daha zeki hale getirmekte ve böylece üretim süreçlerine çok yönlü katkı sağlamaktadır.

Tablo 3’ te pratikteki bulanık mantık uygulamalarından bazı örnekler verilmiştir.

Tablo 3: Bulanık Denetim Uygulamaları

ÜRÜN	FİRMA	BULANIK MANTIĞIN İŞLEVİ
Asansör Denetimi	Fujitec –Toshiba Mitsubishi	Yolcu trafiğini değerlendirir. Böylece bekleme zamanı azalır.
SLR Fotoğraf Makinesi	Sanyo –Fisher Canon Minolta	Ekranda birkaç obje olması durumunda en iyi fokusu ve aydınlatmayı belirler
Video Kayıt Cihazı	Panasonic	Cihazın elle tutulması nedeniyle çekim sırasında oluşan sarsıntıları ortadan kaldırır.
Çamaşır Makinesi	Matsushita	Çamaşırın kirliliğini, ağırlığını, kumaş cinsini sezer, ona göre yıkama programını seçer.
Klima	Mitsubishi	Ortam koşullarını değerlendirerek en iyi çalışma durumunu algılar, odaya birisi girerse soğutmayı artırır.
ABS Fren Sistemi	Nissan	Tekerleklerin kilitlenmeden frenlenmesini sağlar.
Sendai Metro Sistemi	Hitachi	Hızlanma ve yavaşlamayı ayarlayarak rahat bir yolculuk sağlanmasının yanı sıra durma konumunu iyi ayarlar, güçten tasarruf sağlar.
Çimento Sanayi	Mitsubishi Chem	Değirmende ısı ve oksijen oranı denetimi yapar.
Televizyon	Sony	Ekran kontrastını, parlaklığını ve rengini ayarlar.
El Bilgisayarı	Sony	El yazısı ile veri ve komut girişine olanak sağlar.

Kaynak: Yaralıoğlu, 2004: 7

Görüldüğü gibi bulanık mantık uygulamalarının bir çok elektronik alette, aletlerin yapay zekaya sahip olmalarını sağladığı ve ayrıca elektrik tasarrufu sağlamaya yardımcı olduğu hatta satışları artırıcı etkisi olduğu bile söylenebilir.

Bulanık mantığın, karar vermede avantajları ve kullanıldığı yerlerin oldukça yaygın olmasına karşın dezavantaj sayılabilecek özellikleri de mevcuttur, bunlar şu şekilde sıralanabilir:

- Bulanık karar vermede kullanılan kurallar deneyime çok bağlıdır ve karar vericinin subjektif fikirlerini içerir.
- Üyelik fonksiyonlarının seçiminde belirli bir yöntem yoktur. En uygun fonksiyon deneme ile bulunur. Bu da oldukça uzun zaman alabilir.
- Denetlenen sistemin bir kararlılık analizi yapılamaz ve sistemin nasıl cevap vereceği önceden kestirilemez. Yapılacak tek şey benzetim yani simülasyon çalışmasıdır.

Bu özellikleri ile de zaman ve para konusunda belli oranda kayıp olduğu düşünülse de teknolojinin ve insan beyninin aktif olarak kullanılması sebebiyle sonuçları net ve doyurucu olacaktır.

1.1.3. Bulanık Küme ve Temel Kavramlar

Bulanık mantığın klasik mantıktan en önemli farkı, üzerinde çalışılan karar probleminin modellenmesinde üyelik fonksiyonunun kullanılmasıdır. Klasik mantık olayların meydana geliş olasılığı ile ilgilenirken, bulanık mantık olayların belirli kümelere üyelik derecesi ile ilgilenir. Bulanık Küme Teorisinde kullanılan bazı temel kavramlar aşağıdaki gibidir.

Bulanık kümenin tanımlanabilmesi için öncelikle üyelik fonksiyonu tanımlanmalıdır.

1.1.3.1. Üyelik Fonksiyonu

X , elemanları x 'ler olan bir noktalar kümesi olsun, yani $X = \{x\}$. X 'in içinde bir $\tilde{A}$ bulanık kümesi bir üyelik fonksiyonu $\mu_{\tilde{A}}(x)$ ile karakterize edilir. Bu fonksiyon X içindeki her nesneyi, 0 ile 1 arasındaki bir reel sayıya $[0,1]$ tekabül ettirir ve bu fonksiyon 1.1'deki gibi tanımlanır.

$$\mu_{\tilde{A}}(x): X \rightarrow [0,1] \quad (1.1)$$

1.1.3.2. Bulanık Küme

Klasik küme teorisine göre küme elemanları, kümeye dahil ise 1, dahil değil ise 0 değerleri kullanılarak tanımlanmaktadır:

$$X_A(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases} \quad (1.2)$$

Bulanık küme kuramı ise bir elemanın bir kümeye kısmi üyeliğine olanak sağlar. Eğer üyelik derecesi olarak adlandırılan üyelik fonksiyonunun değeri 1'e eşitse x elemanı bulanık kümeye tamamen aittir. Eğer bu değer 0 ise, x bulanık kümeye ait değildir. Eğer üyelik derecesi 0 ile 1 arasında ise x, bulanık kümenin kısmi üyesidir. Bulanık bir $\tilde{A}$ kümesinin sıralı ikilileri;

$$\tilde{A} = \{x, \mu_{\tilde{A}}(x) : x \in X\} \quad (1.3)$$

şeklinde tanımlanır. Sıralı ikililerden oluşan elemanlardan birincisi kümenin elemanı, ikincisi ise bu elemanın üyelik derecesini belirten değerdir.

Bulanık küme literatüründe, genellikle 'kesin' terimi bulanık olmayan büyüklükleri belirtmek için kullanılır. Örneğin; kesin sayı, kesin küme, vb.

1.1.3.3. α -Kesim Seviyesi

$\tilde{A}$ bulanık kümesinin A_α ile gösterilen α -kesimi, klasik X uzayında üyelik dereceleri α 'ya eşit veya büyük olan tüm elemanlarından oluşan kesin alt kümesidir ve 1.4 teki gibi gösterilir.

$$A_\alpha = \{x | \mu_A(x) \geq \alpha, \alpha \in [0,1]\} \quad (1.4)$$

Eğer $A_\alpha = \{x | \mu_A(x) > \alpha, \alpha \in [0,1]\}$ ilişkisi mevcutsa buna $\tilde{A}$ kümesinin kuvvetli α -kesimi adı verilir.

1.1.3.4. Kümenin Desteği

Bulanık bir $\tilde{A}$ kümesinin desteği, küme içerisinde üyelik derecesi 0'dan büyük olan elemanların oluşturduğu kümedir.

$$\text{supp}(\tilde{A}) = \{x | \mu_{\tilde{A}}(x) > 0\} \quad (1.5)$$

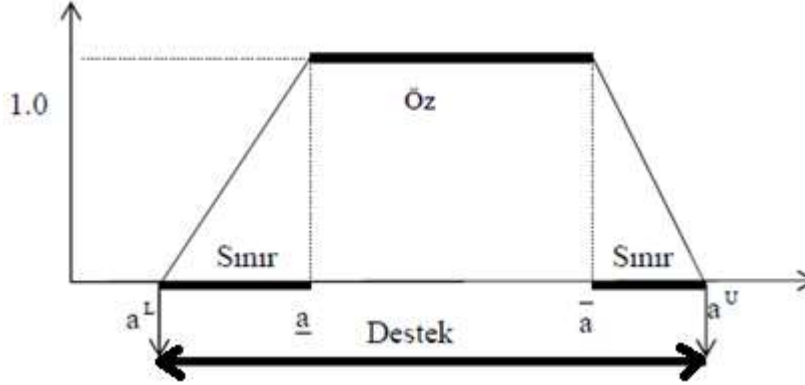
Üyelik derecesi 1'e eşit olan elemanların oluşturduğu kümeye bulanık $\tilde{A}$ kümesinin özü denir.

$$\text{core}(\tilde{A}) = \{x | \mu_{\tilde{A}}(x) = 1\} \quad (1.6)$$

Literatürde, $\tilde{A}$ bulanık kümesinin özü bazen $\ker(\tilde{A})$ olarak gösterilir.

0 ve 1 arasında kalan elemanların oluşturduğu kümeye ise $\tilde{A}$ kümesinin sınırı denir.

Şekil 2: Kümenin Desteği, Özü ve Sınırı



Kaynak: Şen, 2004: 28

1.1.4. Bulanık Küme İşlemleri

Klasik Küme Teorisinden farklı olarak Bulanık Küme Teorisinde küme işlemleri üyelik fonksiyonu yardımı ile aşağıdaki gibi tanımlanır (Zimmermann, 1987: 17).

1.1.4.1. Birleşim İşlemi

$\tilde{A}$ ve $\tilde{B}$ bulanık kümelerinin $\tilde{D} = \tilde{A} \cup \tilde{B}$ birleşim işlemi üyelik fonksiyonu yardımı ile aşağıdaki gibi gösterilir. Bu sonuç aynı zamanda “BULANIK KARAR (Fuzzy Decision)” ı ifade eder.

$$\mu_{\tilde{D}}(x) = \max\{\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)\}, x \in X \quad (1.7)$$

veya başka bir gösterim şekli ile;

$$\mu_{\tilde{D}}(x) = \mu_{\tilde{A}}(x) \vee \mu_{\tilde{B}}(x) \quad (1.8)$$

olarak tanımlanabilir.

1.1.4.2. Kesişim İşlemi

$\tilde{A}$ ve $\tilde{B}$ bulanık kümelerinin $\tilde{C} = \tilde{A} \cap \tilde{B}$ kesişim işlemi üyelik fonksiyonu yardımı ile aşağıdaki gibi gösterilir. Bu da bir başka “BULANIK KARAR KÜMESİ” olup üyelik derecesini göstermektedir.

$$\mu_{\tilde{C}}(x) = \min\{\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)\}, x \in X \quad (1.9)$$

veya başka bir gösterim ile;

$$\mu_{\tilde{C}}(x) = \mu_{\tilde{A}}(x) \wedge \mu_{\tilde{B}}(x) \quad (1.10)$$

tanımlanabilir.

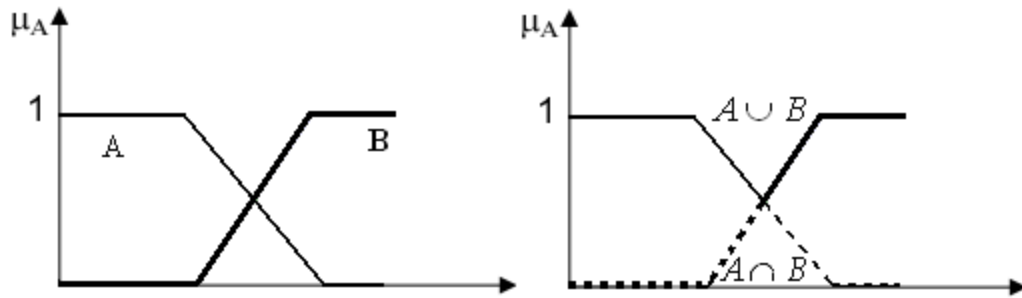
1.1.4.3. Tümlleme İşlemi

Bulanık A kümesinin tümleneni, A^c olarak gösterilir ve aşağıdaki gibi tanımlanır:

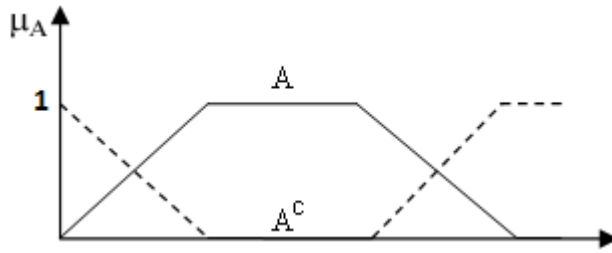
$$\mu_{A^c}(x) = 1 - \mu_A(x) \quad (1.11)$$

Şekil 3, bulanık kümelerin kesişimi ve birleşimini, Şekil 4’te bulanık kümelerin tümleneni grafiksel olarak göstermektedir.

Şekil 3: Birleşim ve Kesişim İşlemleri

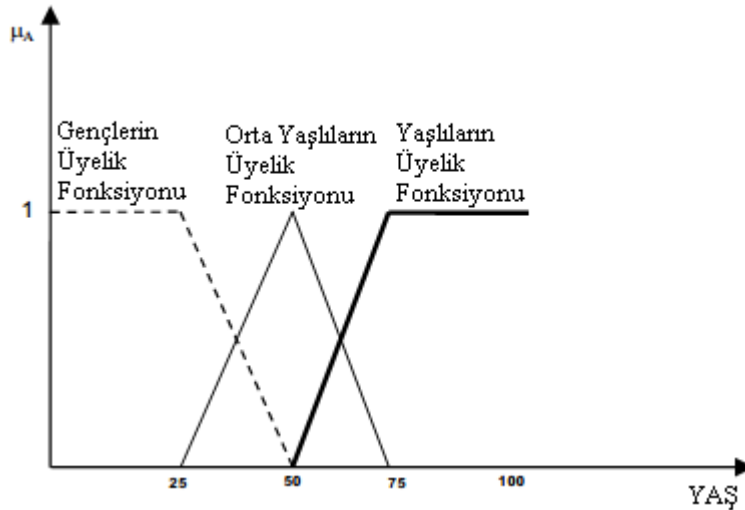


Şekil 4: Bulanık bir A kümesinin Tümleyeni



Şekil 5'te genç, orta yaşlı ve yaşlılar kümelerinin üyelik fonksiyonları yer almakta olup, yatay ekseninde yaş, dikey ekseninde kişinin “genç”, “orta yaşlı” veya “yaşlı” olarak sınıflandırılabilmesinin derecesini göstermektedir. Bu üç grubun üyelik fonksiyonlarının birlikte yer aldığı şekilde gruplar arası ilişkileri de grafik olarak görmek mümkündür. Yaş ilerledikçe, Gençler Kümesine Üyelik azalırken, Orta Yaşlılar Kümesine üyelik artmakta ve fonksiyonların kesim noktalarında ek çıkarımlar üretilebilmektedir.

Şekil 5: Bulanık Kümelerin Üyelik Fonksiyonu “Genç”(kesik çizgili), “Orta Yaşlı”(ince çizgili), “Yaşlı”(kalın çizgili)



Kaynak: Çelikyılmaz ve Türkşen, 2009: 13

Not: Şekil 5' teki kesik çizgiler diğer çizgilerden farklı olması içindir, yani yaşlar ya da üyelik derecesinde ayrıklık söz konusu değildir.

1.1.5. Bulanık Sayılar ve Üyelik Fonksiyonları

Klasik kümelerin tersine, bulanık kümelerin yapısal olarak sahip olduğu esneklikten dolayı, uygulama alanlarının farklılığına göre bu işlemleri temsil etmek için farklı fonksiyonların tanımlanması mümkündür. Bulanık küme teorisi üzerinde kurulan matematiksel analizlerde, sadece bulanık kümelerin üyelik fonksiyonları değil, fonksiyonlar arasında gerçekleşecek işlemlerin çalışılan alan ile yakından ilişkisi vardır. Uygun üyelik fonksiyonunu tanımlama ve anlamlı işlemleri belirleme kapasitesi, bulanık küme teorisinin pratik faydasını arttıran en önemli yönlerinden birisidir (Klir ve Youn, 1995: 12). Bulanık küme teorisinde kullanılan bulanık sayılar için farklı tipleri olmakla birlikte en çok kullanılan tipleri, üçgensel ve yamuk tipli bulanık sayılardır (Lai ve Hwang, 1992: 58). Bu iki bulanık sayı ve üyelik fonksiyonları aşağıda tanımlanmıştır.

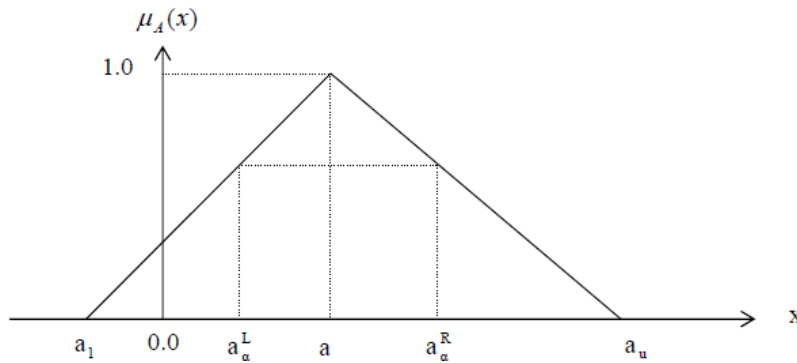
1.1.5.1. Üçgensel Bulanık Sayı ve Üyelik Fonksiyonu

Bulanık üçgensel sayı $A = [a_1, a, a_u]$ için üyelik fonksiyonu aşağıdaki gibi tanımlanır.

$$\mu_A(x) = \begin{cases} 0 & , x < a_1 \\ \frac{x-a_1}{a-a_1} & , a_1 \leq x \leq a \\ \frac{a_u-x}{a_u-a} & , a < x \leq a_u \\ 0 & , x > a_u \end{cases} \quad (1.12)$$

Üçgensel üyelik fonksiyonunun grafik gösterimi Şekil 6 da verilmiştir.

Şekil 6: Bulanık Üçgensel Üyelik Fonksiyonu



Bulanık üçgensel sayı $A = [a_1, a, a_u]$ için α -kesim seviyesi aşağıdaki gibi kapalı aralıkta tanımlanmıştır.

$$A = [a_\alpha^L, a_\alpha^R] = [(a - a_1)\alpha + a_1, -(a_u - a)\alpha + a_u] \text{ ve } \alpha \in [0,1].$$

Bulanık sayılarla yapılan matematiksel işlemler kendine has yapı içermektedir. Üçgensel tipli bulanık sayılarda dört işlemin nasıl yapıldığı aşağıda belirtilmiştir.

A ve B, bulanık üçgensel tipli iki sayı olsun:

$$A = [a_1, a_2, a_3] \text{ ve } B = [b_1, b_2, b_3].$$

1.1.5.1.1. Toplama

$A \oplus B = [a_1 + b_1, a_2 + b_2, a_3 + b_3]$ şeklindedir ve sonuç yine bir üçgensel bulanık sayıdır.

1.1.5.1.2. Çıkarma

$A \ominus B = [a_1 - b_3, a_2 - b_2, a_3 - b_1]$ şeklindedir ve sonuç bulanık üçgensel sayıdır.

1.1.5.1.3. Çarpma

$A \otimes B = [a_1 b_1, a_2 b_2, a_3 b_3]$ şeklindedir ve sonuç bulanık üçgensel sayıdır.

1.1.5.1.4. Bölme

$A \oslash B = [a_1 : b_3, a_2 : b_2, a_3 : b_1]$ dir. Bu işlemin sonucu da bulanık üçgensel sayıdır.

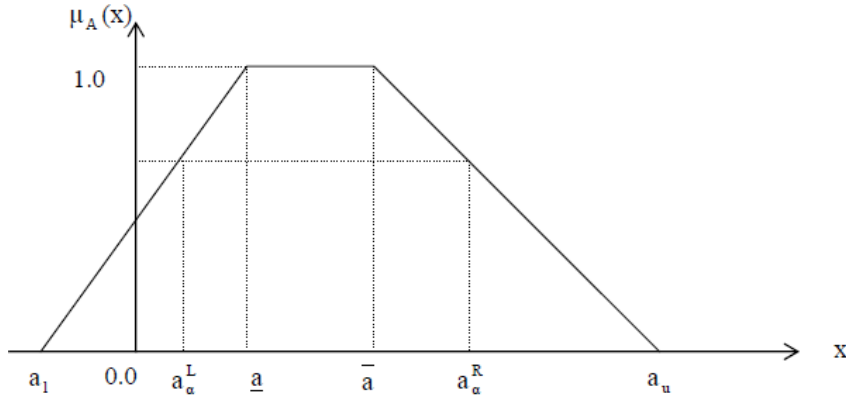
1.1.5.2. Yamuk Tipli Bulanık Sayı ve Üyelik Fonksiyonu

Yamuk tipli bulanık sayı $A = [a_1, \underline{a}, \bar{a}, a_u]$ için üyelik fonksiyonu aşağıdaki gibi tanımlanır.

$$\mu_A(x) = \begin{cases} 0 & , x < a_1 \\ \frac{x-a_1}{\underline{a}-a_1} & , a_1 \leq x \leq \underline{a} \\ 1 & , \underline{a} \leq x \leq \bar{a} \\ \frac{a_u-x}{a_u-\bar{a}} & , \bar{a} < x \leq a_u \\ 0 & , x > a_u \end{cases} \quad (1.13)$$

Yamuk tipli bulanık sayı için üyelik fonksiyonunun grafik gösterimi Şekil 7’de verilmiştir.

Şekil 7: Yamuk Tipli Bulanık Sayı Üyelik Fonksiyonu



Yamuk tipli bulanık sayı $A = [a_1, \underline{a}, \bar{a}, a_u]$ için α -kesim seviyesi aşağıdaki kapalı aralıkta tanımlanmıştır.

$$A = [a_\alpha^L, a_\alpha^R] = [(\underline{a} - a_1)\alpha + a_1, -(a_u - \bar{a})\alpha + a_u] \text{ ve } \alpha \in [0,1].$$

Yamuk tipli iki bulanık sayı için dört işlem şu şekildedir:

$$A = [a_1, a_2, a_3, a_4] \text{ ve } B = [b_1, b_2, b_3, b_4] \text{ olsun.}$$

1.1.5.2.1. Toplama

$$A \oplus B = [a_1 + b_1, a_2 + b_2, a_3 + b_3, a_4 + b_4] .$$

1.1.5.2.2. Çıkarma

$$A \ominus B = [a_1 - b_4, a_2 - b_3, a_3 - b_2, a_4 - b_1] .$$

1.1.5.2.3. Çarpma

$$A \otimes B = [a_1 b_1, a_2 b_2, a_3 b_3, a_4 b_4] .$$

1.1.5.2.4. Bölme

$A \oslash B = [a_1 : b_4, a_2 : b_3, a_3 : b_2, a_4 : b_1]$ şeklindedir ve yamuk tipli bulanık sayılarda toplama ve çıkarma işlemlerinin de sonuçları yamuk tipli bulanık sayıdır.

1.2. BULANIK ORTAMDA KARAR VERME

Bellman ve Zadeh 1970 yılında klasik karar verme modellerini incelemek sureti ile bulanık ortamda karar verebilmek için değişik modeller geliştirmişlerdir. Belirsizlik altında karar verme durumunda, amaç fonksiyonu ve kısıtlar belirsiz veya bulanıktır. Bu bulanıklık, amaç fonksiyonu ve kısıtların üyelik fonksiyonu yardımı ile karakterize edilebilir. Bulanık ortamda amaç fonksiyonu ve kısıtların değerlendirilmesi klasik optimizasyon yöntemleri kullanılarak gerçekleştirilir (Zimmermann, 1987: 16).

Bulanık Karar Verme, bulanık kısıtların ve bulanık hedef fonksiyonlarının kesişiminden meydana gelen alternatiflerin bulanık kümesidir. Bulanık hedefler, üyelik fonksiyonları kullanılarak nitelendirilirler. Bulanık küme teorisinde “logical and” kavramının karşılığı klasik kümede kesişimdir. Bulanık ortamda karar verme bulanık hedeflerin ve bulanık kısıtların kesişimi olarak açıklanır (Bellman ve Zadeh, 1970: 141-164). Tanımları ve matematiksel olarak ifadeleri verilecek olursa:

1.2.1. Bulanık Hedef

Bir bulanık hedef veya bulanık amaç $\tilde{G}$ olsun. $X = \{x\}$ alternatifler kümesi olmak üzere, Bulanık hedef $\tilde{G}$, X içerisinde bulanık küme şeklinde tanımlanır. Burada $\mu_{\tilde{G}}(x) \in [0,1]$ tir. Geleneksel yaklaşımlarda, karar verme süreci içerisinde bir performans fonksiyonu alternatifler kümesinde doğrusal bir düzende işlem görür.

1.2.2. Bulanık Kısıt

Bir bulanık kısıt $\tilde{C}$, X 'de tanımlanmış bir bulanık küme olsun. Bulanık hedef ve bulanık kısıtların yukarıda tanımlanan kavramlarının önemli bir özelliği, her ikisinin de alternatifler uzayında bulanık kümeler şeklinde tanımlanmasıdır. Bu şekilde yapılan bir tanımlamayla, kararın formülasyonunda bulanık hedefler ve bulanık kısıtlar aynı şekilde işlem görürler. Diğer yandan, bir karar problemi için geleneksel yaklaşımlarda alternatifler uzayında kısıt kümesi bulanık olmayan kümeler olarak alınırken performans fonksiyonu sadece bir fonksiyondur.

1.2.3. Bulanık Karar

Basit olarak karar verme işlemi, alternatiflerden seçim yapma işlemidir. Bulanık bir karar, bulanık kısıtların ve bulanık hedeflerin kesişimlerinden elde edilen bulanık bir küme olarak tanımlanabilir. Bu tanımlama aşağıdaki gibi formüle edilebilir:

Alternatifler uzayında bulanık kısıt $\tilde{C}$ ve bulanık hedef $\tilde{G}$ için $\tilde{C}$ ve $\tilde{G}$ 'nin kesişiminden bulanık karar $\tilde{D} = \tilde{G} \cap \tilde{C}$ şeklinde ifade edilir. Bulanık karar kümesinin üyelik fonksiyonu,

$\mu_{\tilde{D}}(x) = \min\{\mu_{\tilde{G}}(x), \mu_{\tilde{C}}(x)\}$ şeklindedir.

Bulanık karar için bir genelleştirme yapılacaktır. n adet hedef ve m adet kısıt için hedefler $\tilde{G}_1, \tilde{G}_2, \dots, \tilde{G}_n$ ve kısıtlar $\tilde{C}_1, \tilde{C}_2, \dots, \tilde{C}_m$ şeklinde yazılabilir. Hedeflerin ve kısıtların kesişimleri,

$$\tilde{D} = (\cap_{j=1, \dots, n} \tilde{G}_j) \cap (\cap_{i=1, \dots, m} \tilde{C}_i) \quad (1.14)$$

şeklinde yazılır. Daha açık bir ifadeyle bulanık karar;

$$\mu_{\tilde{D}}(x) = \min\{\min_{j=1,\dots,n} \mu_{\tilde{G}}(x), \min_{i=1,\dots,m} \mu_{\tilde{C}}(x)\} \quad (1.15)$$

şeklinde ifade edilir (Babuska, 1998).

İKİNCİ BÖLÜM

HEDEF PROGRAMLAMA VE BULANIK HEDEF PROGRAMLAMA

2.1. KARAR VERMEDE ÇÖZÜM TİPLERİ

Karar verme işlemi, bazı kriterlerin optimizasyonu şartı altında, oluşturulan kısıtlara bağlı olarak alternatifler içerisinde optimal sonucun belirlenmesi sürecidir. Karar verme sürecinde ilk olarak problem çok iyi tanımlanmalı, amaçlar ve alternatifler kesin olarak bilinmelidir. Karar verme problemleri çoğunlukla birden fazla kriter içermektedir. Çok Kriterli Karar Verme problemlerinde karar verici tarafından oluşturulan amaçlar ve nitelikler karar sürecinin ilk aşamasıdır. Belirlenen amaçlar ve niteliklere göre problemin değerlendirilmesinde temel sorun, amaçların ve niteliklerin birden fazla ve birbirleriyle zıt olmasıdır ki, bu da karar problemlerinde sıkça karşılaşılan bir durumdur (Umarusman, 2007: 6).

Çok Kriterli Karar Verme (ÇKKV) problemleri, Çok Amaçlı Karar Verme (ÇAKV) ve Çok Nitelikli Karar Verme (ÇNKV) olarak iki sınıfa ayrılabilir (Tabucanon, 1988: 5). ÇAKV metotları matematiksel kısıtlar tarafından tanımlanan sınırsız sayıdaki alternatifleri içeren amaç problemleri için kullanılırken, ÇNKV metotları belirlenen kesin alternatifler içerisinde bir alternatifin seçilmesi için kullanılır. ÇAKV metotları kısıtları göz önünde bulundurarak karar verici tarafından oluşturulan amaçların kabul edilebilir seviyelerinin belirlenmesi sürecidir. ÇAKV metotlarının ortak özellikleri; çok iyi tanımlanmış kısıtlar, belirgin amaçlar ve amaçlar arasında belirgin etkileşimdir (Hwang ve Masud, 1979: 6). Diğer yandan, ÇNKV metotlarının ayırt edici özelliği önceden belirlenmiş sınırlı sayıda alternatifi içermesidir. Bu alternatifler içerisinde en iyi olanının seçilmesi nitelikler arasında birebir karşılaştırma yapılarak gerçekleştirilir (Hwang ve Yoon, 1981: 2).

ÇAKV problemlerinin klasik (tek amaçlı) optimizasyon problemlerinden en belirgin farkı matematiksel olarak sadece amaç fonksiyonu sayısından kaynaklanmaktadır. Tek amaçlı optimizasyon problemlerinde çözümün hedefi, amaç fonksiyonunun en iyi değerini veren değişkenlerin belirlenmesidir. Bu sebeple klasik optimizasyon problemlerinde amaç fonksiyonunun optimum değeri tektir. Diğer Çok Amaçlı optimizasyon problemlerinde fonksiyon sayısı birden fazla ve birbiriyle

ihtilaflı olması sebebiyle klasik optimizasyon problemlerin farklı olarak optimal çözüme sahip olmaları hemen hemen imkansızdır. Çünkü bu amaç fonksiyonları uygun çözüm bölgesinin farklı değişken ve bunların farklı değerlerine göre optimum değeri almaktadır. Çok Amaçlı Optimizasyon problemleri farklı tipteki amaç fonksiyonlarına göre aşağıdaki gibi açıklanır(Lai ve Hwang, 1994: 28). Maksimizasyon yönlü amaç fonksiyonları için,

$$Z_k(x) = [Z_1(x), Z_2(x), \dots, Z_1(x)] \quad (2.1)$$

ve minimizasyon yönlü amaçlar için,

$$W_s(x) = [W_1(x), W_2(x), \dots, W_r(x)] \quad (2.2)$$

olmak üzere matematiksel olarak;

$$\begin{aligned} \text{Maksimize } Z(x) &= \sum_{k=1}^1 Z_k(x) \\ \text{Minimize } W(x) &= \sum_{s=1}^r W_s(x) \end{aligned}$$

Kısıtlar; (2.3)

$$g_i(x) = \sum_{j=1}^n a_{ij} x_j \otimes 0$$

$$x_j \geq 0 \quad i = 1, 2, \dots, m \quad j = 1, 2, \dots, n \quad k = 1, 2, \dots, 1 \quad \text{ve } s = 1, 2, \dots, r$$

modeli ile ifade edilir. Burada $\otimes$ işaretiyle " $\leq$ ", " $=$ ", " $\geq$ " eşitsizlik durumları genelleştirilmiştir. Problemin uygun çözüm bölgesi X ile gösterilir ve $X = \{x \in R: g_i(x) \otimes 0, x \geq 0, \forall i, j\}$ şeklinde tanımlanır. Optimizasyon problemleri uygun çözüm alanı X içerisinde bir x^* ($x \in X$) değerini araştırır ve amaçların maksimum veya minimum değerlerini belirler. Çok Amaçlı optimizasyon problemlerinin çözümünde tek tip çözüm sonucu elde edilemez. Bu sebeple belirlenen çözümlerin yapısına göre farklı isimler kullanılır. Bu farklı çözümler aşağıdaki başlıklarda ifade edilmiştir (Lai ve Hwang, 1994: 28).

2.1.1. Optimal Çözüm

Bir vektör maksimum problemi için optimal çözüm, her bir amaç fonksiyonunun maksimum değerinin aynı değişkenlere göre belirlenmesidir. Çok

Amaçlı Karar Verme (ÇAKV) problemlerinde amaçların birbirleriyle zıt olması sebebiyle optimal çözüme genellikle ulaşılmaz. Bir optimal çözüm için aşağıdaki şart sağlanmalıdır.

$$\forall x \in X \text{ için } x^* \in X \text{ ve } Z_k(x^*) \geq Z_k(x) \quad (2.4)$$

2.1.2. İdeal Çözümler

İdeal çözümler pozitif ideal çözüm ve negatif çözüm olmak üzere iki farklı şekilde incelenir. Maksimizasyon ve minimizasyon yönlü amaçlar için pozitif ideal çözüm;

$$\begin{aligned} Z_k^* &= \underset{x \in X}{\text{Maximize}} Z_k(x) \\ W_s^* &= \underset{x \in X}{\text{Minimize}} W_s(x) \end{aligned} \quad (2.5)$$

olarak belirlenir Her iki amaç fonksiyonu için pozitif ideal çözüm kümesi

$$I^* = \{Z_1^*, Z_2^*, \dots, Z_r^*; W_1^*, W_2^*, \dots, W_r^*\} \quad (2.6)$$

şeklinde gösterilir. Maksimizasyon ve minimizasyon yönlü amaçlar için negatif ideal çözümler;

$$\begin{aligned} Z_k^- &= \underset{x \in X}{\text{Minimize}} Z_k(x) \\ W_k^- &= \underset{x \in X}{\text{Maximize}} W_r(x) \end{aligned} \quad (2.7)$$

olarak belirlenir. Negatif ideal çözümler kümesi;

$$I^- = \{Z_1^-, Z_2^-, \dots, Z_r^-; W_1^-, W_2^-, \dots, W_r^-\} \quad (2.8)$$

şeklinde ifade edilir.

2.1.3. Üstün Olmayan Çözüm

Bu çözüm, ÇAKV içerisinde ikincil çözüm, etkin çözüm olarak da kullanılır. Ekonomide Pareto çözüm, istatistiksel karar teorisinde kabul edilebilir alternatifler olarak adlandırılır. Üstün olmayan çözümlerin kümesi S olmak üzere,

$$\exists x' \in X$$

$$Z_{k'}(x') > Z_{k'}(x) \text{ ve } Z_k(x') > Z_k(x), k \neq k' \quad (2.9)$$

veya

$$W_{s'}(x') < W_{s'}(x) \text{ ve } W_s(x') < W_s(x), s \neq s'. \quad (2.10)$$

genel olarak üstün olmayan çözümlerin sayısı oldukça fazladır.

ÇADP (Çok Amaçlı Doğrusal Programlama) problemlerinde, üstün olmayan çözümler kümesinde çözüm sayısı oldukça fazladır. p- tane amaç fonksiyonunun birisine göre optimallik kavramı diğer “p-1” adet amaçlar için genellikle optimal değildir. Bu sebeple, tek bir optimal çözümün araştırılması yerine çok amaçlı programlama problemlerinde uygun çözüm alanı içerisinde üstün olmayan çözümlerin kümesinin tanımlanması ilk adımdır. Üstün olmayan çözüm noktalarının değerlendirilmesi için temel süreç, ideal çözüm noktalarına yakınlığın nasıl belirleneceğidir (Cohon, 1978: 69). Üstün olmayan çözümler arasından Step Metot, Geoffrion-Dyer-Freineberg Metot, Zouts ve Wallenius Metodu gibi İnteraktif Metotlar kullanılarak karar verici en tatminkâr çözümü belirler (Mollaghasemi ve Pet-Edwards, 1997: 55-58). Üstün olmayan çözümlerin değerlendirilmesinde diğer bir süreç, pozitif ve negatif ideal çözümlere göre yapılan değerlendirmedir. İdeal çözümlere bağlı olarak Uzlaşık Programlama, Hedef Programlama ve Bulanık Mantık modelleri ile uzlaşık çözüm belirlenir (Zimmerman, 1978).

2.2. HEDEF PROGRAMLAMA

Hedef Programlama, Doğrusal Programlamanın özel bir hali olup, tek hedef veya çoklu alt hedefli karar problemlerinde kullanıldığı gibi çoklu hedefli ve çoklu alt hedefli problemler için de kullanılır. Yöneticiler tarafından belirlenen bir hedefin gerçekleştirilmesi için bir veya birden fazla hedef göz ardı edilebilir. Ayrıca, hedefler farklı birimlerde olabildiği gibi, hedefler arasında önem sırasının oluşturulmasına zaman zaman ihtiyaç duyulmaktadır. Yüksek öncelikli hedeflerden başlanarak düşük öncelikli hedeflere doğru hedeflerin gerçekleştirilmesi sağlanır. Hedef Programlamanın, Doğrusal Programlamadan en önemli farkı amaç fonksiyonunun doğrudan maksimizasyonu veya minimizasyonu olmayıp, hedeflerin verilen kısıtlar altında mümkün olduğunca, planlanan hedefler ile gerçekleşen değerler arasındaki pozitif ya da negatif istenmeyen sapmaların minimize edilmesidir.

Bir Doğrusal Programlama modeli maksimize veya minimize edilmiş tek bir amaçtan ve kısıtlardan oluşurken, Hedef Programlama modeli karar vericiden sağlanan bilgilere bağlı olarak hedefler arasında öncelik sıralaması ya da hedeflerin

birbirlerine göre göreceli olarak ağırlıklandırıldığı bir hedef kümesi ve kısıtlardan meydana gelir. Hem Doğrusal Programlama hem de Hedef Programlama modellerinde, problemin içinden gelen kısıtlar modele doğru yansıtılmazsa, model için uygun çözümler bulmak çok zorlaşır. Hedef Programlamada, uygun çözümler kümesinde yer alan “herhangi bir çözüm” kısıtları sağlasa bile, farklı hedeflerin tamamını aynı anda tatmin etmeyebilir. Hedef Programlamanın amacı, tutarlı kısıtları tatmin edecek ve belirlenen hedefleri en iyi şekilde karşılayacak bir çözüm bulmaktır.

Hedef Programlamanın felsefi temelini Herbert A. Simon’un “amaçların tatmini” kavramı oluşturur. Simon’un varsayımına göre günümüz kompleks organizasyonlarında karar vericiler iyi tanımlanmış fayda fonksiyonunu maksimize etmeye çalışmazlar. Çünkü çıkarların çatıştığı ve bilginin yetersizliği sebebiyle Karar Vericilerin tercihlerini güvenilir bir matematiksel modele aktarmaları neredeyse mümkün değildir. Karar vermenin bu çeşidinde karar verici hedefleri en yakın değerinde gerçekleştirmeye çalışır (Jones ve Tamiz, 2002: 130).

Tatminkârlık kavramı “tatmin” ve “yeterli” kelimelerinin bir araya getirilmesinden oluşmuştur. Tatminkârlık, karar vericilerin tanımlanmış olan hedeflere ulaşmasını açıklayan bir davranış biçimidir. Eğer tanımlanmış hedeflere ulaşırlar ise karar durumu “yeterli” olur ve bu sebeple karar vericiler “tatmin” olurlar. Simon’un görüşüne göre insanoğlu karar probleminin her bir sonucunun optimize edilmesi soyut kavramı yerine hedeflere ulaşabilmekle ilgilenir. Karar verme kavramında optimize etmenin anlamı, muhtemel kararlar arasından bazı ölçümlerin en iyi değerini veren kararı bulmaktır. Çok Amaçlı Karar Verme’de optimizasyon teorisi, İtalyan ekonomist Vilfredo Pareto’nun karar verme alanındaki çalışmasından uyarlanarak tanımlanmıştır (Jones ve Tamiz, 2010: 6).

Kısıt fonksiyonlarının esneklik içermesi ve optimizasyonun karşıtı olan “tatminkârlık” felsefesine bağlılık, Hedef Programlamayı geleneksel optimizasyon metotlarından ayıran en belirgin özelliklerdir (Ignizio ve Romero, 2002: 490). Optimize etmek ve tatmin etmek kavramlarının karıştırılmasından dolayı Hedef Programlama yanlış anlaşılmaktadır. Bir çok geleneksel OR (Operational Research, Yöneylem Araştırması) tekniklerinde olduğu gibi Doğrusal Programlamanın esas amacı optimal çözümlerin elde edilmesi iken HP yaklaşımı tatminkar çözümleri

araştırır. Hedef Programlama arzu edilen seviyelere mümkün olduğunca yakın olarak önerilen çözümlerle kişilerin beklentilerini tatmin etmeyi hedefler. Bu yüzden, geleneksel yaklaşımların sağladığı çözümler tatminkâr olmadığında, Hedef Programlama karar vericilerin arzu edilen seviyelerini (hedeflerini) karşılamaya ve en iyi çözümlerin araştırılmasına imkân tanır (Min ve Storbeck, 1991: 302).

2.2.1. Hedef Programlama ile İlgili Kavramlar

Hedef Programlamayı daha iyi anlamamız için kullanılan literatürün tanımlarını vermekte fayda vardır (Jones ve Tamiz, 2010: 3-5)

Karar Vericiler: Bu çalışmada karar vericiler; karar probleminin ait olduğu kişiler, kuruluşlar ya da paydaşlardır.

Karar Değişkeni: Bir karar değişkeni, karar vericinin denetimi dışında olan bir faktör olarak tanımlanır. Örnek olarak; bir üretim şirketinin belirli bir ürünü gelecek ay için kaç tane üreteceğine karar vermesidir. Karar değişkenlerinin kümesi, problemi tamamen tanımlar ve verilecek kararı oluşturur. Hedef Programlama modelinin amacı, karar vericinin hedeflerini ve kısıtlarını en iyi şekilde tatmin eden noktayı belirlemek için, karar değişkenlerinin değerlerinin(karar uzayı olarak da bilinir) tüm olası kombinasyonlarının araştırılması olarak görülebilir.

Kriter: Bir kriter, karar problemi için herhangi bir çözüm değerinin ölçülebilir tek bir ölçüsüdür. Farklı uygulama alanlarından kaynaklanan birçok kriter bulunmaktadır. Bunlar; maliyet, kar, zaman, uzaklık, sistemin performansı, şirket veya organizasyonun stratejisi, karar vericilerin kişisel tercihleri, güvenlik faktörü gibi kriterlerdir. Bir karar problemi, birden fazla kritere sahip olabileceğinden, Çok Amaçlı Karar Verme veya Çok Amaçlı Karar Yardımı problemi olarak adlandırılır. Kriterler tarafından oluşturulan alan, kriter alanı olarak bilinir.

Amaç: Bu çalışmada amaç; karar vericinin kriter ölçeğinde tercihini hangi yönde kullandığının(maksimize veya minimize etme) ek bilgisini veren bir kriter olarak adlandırılır. Amaçları maksimize veya minimize edilen bir karar problemi, çok amaçlı bir optimizasyon problemi olarak adlandırılır. Uygulamada, bu amaçların aynı anda optimal değerlerine ulaşılamayacağından, bu amaçlar çelişkili olacaktır. Eğer amaçların her biri için optimallik sağlanırsa, model tek amaçlı bir problem

olarak çözümlenebilir. Bu amaçların değerleri ile oluşan bu alan amaç uzayı(alanı) olarak adlandırılır.

Hedef: Bu çalışmada hedef, bir kriter ve karar verici(ler)nin bu kriteri başarma arzusunun sayısal seviyesi olarak ifade edilmektedir.

Sapan Değişkenler: Sapan değişken bir kriterdeki hedef seviyesi ile elde edilen çözümdeki değer arasındaki farkı ölçer. Elde edilen değer, hedeflenen seviyenin üzerinde ise aradaki fark pozitif sapan değişken değerini verir. Elde edilen değer, hedef değerın altındaysa aradaki fark negatif sapan değişkenin değerini verir. Hedef programlamanın esas amacı, istenmeyen sapan değişkenlerin minimize edilmesidir. 1.tip hedef (daha azı daha iyidir) için pozitif sapan değişkenin, 2.tip hedef (daha fazlası daha iyidir) için negatif sapan değişkenin, 3.tip hedef için ise hem pozitif hem de negatif sapan değişkenin istenmeyen sapma değişkeni olduğu söylenir.

Kısıt: Kısıt, pratikte uygulanabilir bir çözüm için yerine getirilmesi gereken karar değişkenleri üzerinde bir sınırlamadır. Kısıt, başarısızlığın otomatik olarak çözümü uygulanamaz hale getirmesi açısından hedef kavramından farklıdır. Kısıt genellikle birkaç karar değişkenin bir fonksiyonudur ve eşitlik ve eşitsizlik söz konusu olabilir.

Kısıt İşareti: Kısıt işareti, tek bir karar veya sapan değişkenini belirli değerler alanında sınırlar. En yaygın kısıt işareti negatif olmayan ve sürekli değişken içindir.

Uygun Çözüm Bölgesi: Karar uzayındaki çözümlerin kümesi bütün kısıtları ve kısıt işaretlerini sağlayan çözümlerin kümesidir. Uygun çözüm bölgesi içinde kalan herhangi bir çözümün pratikte uygulanabilir olduğu görülmüştür.

2.2.2. Hedef Programlama Formülasyonu

Doğrusal Programlamanın özel bir uzantısı olan Hedef Programlama, Doğrusal Programlamanın varsayımlarına bağlı olarak geliştirilmiştir. Kanonik formda bir Doğrusal Programlama modeli aşağıdaki gibidir:

$$\text{Minimize: } Z = \sum_{j=1}^n c_j x_j$$

Kısıtlar; (2.11)

$$\sum_{j=1}^n a_{ij} x_j \geq b_i, \quad i = 1, \dots, m$$

$$x_j \geq 0, \quad j = 1, \dots, n$$

Burada $x_1, x_2, \dots, x_n$ negatif olmayan karar değişkenleri, $c_1, c_2, \dots, c_n$ ise amaç fonksiyonu için sırasıyla her bir karar değişkenine ait, marjinal faydayı sağlayan katsayılarıdır. Bu modelde amaç fonksiyonu m tane kısıta bağlı olarak değerlendirilir. Bu kısıtlarda a_{ij} , $i = 1, 2, \dots, n$ ve $j = 1, 2, \dots, m$ olmak üzere her bir x_j ye ait teknolojik katsayılar olup sağ taraf sabiti b_i den kullanılan miktarı göstermektedir (Schniederjans, 1995: 2). Hedef Programlama, Doğrusal Programlamanın özel bir uzantısı olması sebebiyle Doğrusal Programlamanın varsayımlarını da içermektedir.

Hedef Programlama çalışmaları ilk kez Charnes, Cooper ve Ferguson (1955) tarafından başlatılmıştır. Daha sonra Charnes ve Cooper (1961) tarafından Hedef Programlama formülasyonu gerçekleştirilmiştir. Bu yaklaşımda Doğrusal Programlamanın her bir kısıtı ayrı fonksiyonlar olarak ele alınmış ve bunlar “fonksiyonel hedefler” olarak isimlendirilmiştir. Eşitlik durumunda verilen bir kısıtın mutlak değeri “fonksiyonel hedefler” olarak aşağıdaki gibi ifade edilir:

$$f_i(x) = \left| \sum_{j=1}^n a_{ij} x_j - b_i \right| \quad (2.12)$$

Buradaki amaç, mutlak sapmayı minimum kılmaktır. Yukarıdaki (2.12) eşitliğinde aşağıdaki iki durum söz konusudur:

- 1) $\sum_{j=1}^n a_{ij} x_j \geq b_i$ ise $\sum_{j=1}^n a_{ij} x_j - b_i$; buradaki fark d_i^+ ya eşit olur, aksi durumda d_i^+ sıfırdır.
- 2) $b_i \geq \sum_{j=1}^n a_{ij} x_j$ ise $b_i - \sum_{j=1}^n a_{ij} x_j$; buradaki fark d_i^- ye eşit olur, aksi durumda d_i^- sıfırdır.

Burada d_i^+ ve d_i^- sapma miktarları olup sırası ile sapmanın hedef üzerinde ve hedefin altında olduğunu gösterir. d_i^+ pozitif sapan değişken ve d_i^- negatif sapan değişken olarak isimlendirilir. Pozitif ve negatif değişkenler cebirsel olarak aşağıdaki gibi yazılır:

$$d_i^+ = \frac{1}{2} [|\sum_{j=1}^n a_{ij}x_j - b_i| + (\sum_{j=1}^n a_{ij}x_j - b_i)] \quad (2.13)$$

ve

$$d_i^- = \frac{1}{2} [|\sum_{j=1}^n a_{ij}x_j - b_i| - (\sum_{j=1}^n a_{ij}x_j - b_i)] \quad (2.14)$$

Hedef Programlamada doğrusallık varsayımının bozulmaması için,

$$d_i^- \cdot d_i^+ = 0 \quad (2.15)$$

olmalıdır. Çünkü negatif sapma ve pozitif sapmanın her ikisi temel değişken olmayacağından bu sapmalar aynı anda meydana gelemez. (2.13) ve (2.14) eşitliklerinin toplanması ile

$$d_i^- + d_i^+ = |\sum_{j=1}^n a_{ij}x_j - b_i| \quad (2.16)$$

ve çıkarılması ile

$$d_i^+ - d_i^- = (\sum_{j=1}^n a_{ij}x_j - b_i) \Rightarrow \sum_{j=1}^n a_{ij}x_j + d_i^- - d_i^+ = b_i \quad (2.17)$$

elde edilir. (2.12) eşitliğinde (2.16) ile (2.17) birlikte düşünülürse genel Hedef Programlama modeli aşağıdaki gibi yazılır (Charnes ve Cooper, 1977: 39-54):

$$\text{Minimize } Z = \sum_{i=1}^m (d_i^+ + d_i^-)$$

Kısıtlar; (2.18)

$$\sum_{j=1}^n a_{ij}x_j + d_i^- - d_i^+ = b_i$$

$$d_i^- \cdot d_i^+ = 0$$

$$x_j, d_i^-, d_i^+ \geq 0$$

Hedef Programlamada her bir hedefin eşitsizlik yönüne bağlı olarak amaç fonksiyonu oluşturulur. Amaç fonksiyonu, hedeflere ait sapan değişkenlerden meydana gelir. Bu sapan değişkenler aynı zamanda minimum yapılmak istenen, arzu edilmeyen sapmalardır. Bu düşüncenin aşağıdaki üç formda yapılması mümkündür (Ignizio, 1978: 1109-1119):

$$1) \quad \sum_{j=1}^n a_{ij}x_j \leq b_i$$

Eğer i-inci hedef belirlenen başarı düzeyinden küçük veya eşit ise oluşacak pozitif sapan değişken d_i^+ için mümkün olan en küçük pozitif değerin alınması gerekir.

$$2) \quad \sum_{j=1}^n a_{ij}x_j \geq b_i$$

Eğer i-inci hedef belirlenen başarı düzeyinden büyük veya eşit ise oluşacak negatif sapan değişken d_i^- için mümkün olan en küçük pozitif değerin alınması gerekir.

$$3) \quad \sum_{j=1}^n a_{ij}x_j = b_i$$

i-inci hedef belirlenen başarı düzeyini tam olarak karşılıyor ise hem pozitif sapan değişken d_i^+ hem de negatif sapan değişken d_i^- nin toplamalarının aynı anda minimize yapılması gerekir.

Tablo 4: Hedef Tipine Göre Minimize Edilecek Değişken

Hedef Tipi	Hedef Programlama Formu	Minimize Edilecek Değişken
$\sum_{j=1}^n a_{ij}x_j \leq b_i$	$\sum_{j=1}^n a_{ij}x_j + d_i^- - d_i^+ = b_i$	d_i^+
$\sum_{j=1}^n a_{ij}x_j \geq b_i$	$\sum_{j=1}^n a_{ij}x_j + d_i^- - d_i^+ = b_i$	d_i^-
$\sum_{j=1}^n a_{ij}x_j = b_i$	$\sum_{j=1}^n a_{ij}x_j + d_i^- - d_i^+ = b_i$	$d_i^- + d_i^+$

Kaynak: Ignizio, 1978: 1109-1119

Hedef Programlamada minimizasyon süreci farklı metotlarla gerçekleştirilir. Temel olarak üç farklı Hedef Programlama metodu vardır. Bu metotlar; Ağırlıklı Hedef Programlama, Öncelikli Hedef Programlama ve Minmax Hedef Programlamadır. Hedef Programlamanın bu üç metodunda sapmaların minimizasyonu araştırılır. Bu sebeple Hedef Programlama bir minimizasyon problemidir.

2.2.2.1. Ağırlıklı Hedef Programlama

Ağırlıklı Hedef Programlama aynı zamanda ‘Archimedian Hedef Programlama’ olarak da isimlendirilir. Bu yöntemde, hedeflerden ağırlıklı toplam sapmaların minimizasyonunu araştırılır. Ağırlıklı Hedef Programlama ile ilgili ilk çalışma 1965 yılında İjiri tarafından, öncelik ve ağırlık faktörlerinin bir arada düşünüldüğü modelle başlamıştır (Schniederjans, 1995: 6). Charnes ve Cooper (1977) Ağırlıklı Hedef Programlama modelini geliştirmiştir. Model matematiksel olarak aşağıdaki gibidir:

$$\text{Minimize } Z = \sum_{i=1}^m (w_i^+ d_i^+ + w_i^- d_i^-)$$

Kısıtlar; (2.19)

$$\sum_{j=1}^n a_{ij}x_j + d_i^- - d_i^+ = b_i$$

$$x_j, d_i^-, d_i^+ \geq 0$$

$$i = 1, \dots, m \text{ ve } j = 1, \dots, n$$

Burada w_i^- ve w_i^+ sırasıyla negatif ve pozitif sapan değişkenlere atanan göreceli ağırlıklar olup gerçel sayılardır. i-inci hedef, arzu edilen hedef değeri aşması durumunda pozitif sapan değişkenin ağırlığı w_i^+ sıfır olacaktır. Benzer düşünce ile i-inci hedef, arzu edilen hedef değerinin altında olması durumunda negatif sapan değişkenin ağırlığı w_i^- sıfır olacaktır (Romero, 1991: 3).

2.2.2.2. Öncelikli Hedef Programlama

Lexicographic Hedef Programlama olarak isimlendirilen bu model Ağırlıklı Hedef Programlamaya göre farklı bir modelleme yapısına sahiptir ve hedefler arasındaki öncelik sıralamasına bağlı olarak kurulur (Gal, Stewart ve Hanne, 1999: 8-1). Charnes ve Cooper (1977), İjiri tarafından önerilen modelden ağırlık faktörlerini çıkararak her bir hedef için yalnızca öncelik sıralamasının olduğu modeli önermiştir. Model matematiksel aşağıdaki gibidir:

$$\text{Minimize } Z = \sum_{i=1}^m P_i(d_i^+ + d_i^-)$$

Kısıtlar; (2.20)

$$\sum_{j=1}^n a_{ij}x_j + d_i^- - d_i^+ = b_i$$

$$x_j, d_i^-, d_i^+ \geq 0$$

$$i = 1, \dots, m \text{ ve } j = 1, \dots, n$$

$$P_i > P_{i+1} > P_{i+2} > \dots$$

İlk olarak birinci öncelikli i-inci hedefin sapan değişkeni/değişkenleri minimize edilir. Eğer bütün hedeflerin sapan değişkenleri bu öncelikte minimize oluyorsa tatminkarlık sağlanmıştır. İkinci olarak (i+1)-inci hedefe de uygulanarak ilgili sapan değişken/değişkenler minimize edilir. Bu süreç belirlenen öncelik seviyelerinde bütün sapan değişkenler minimize yapılmaya kadar devam eder. Buradan anlaşılacağı üzere Hedef Programlamada hedefler arasında bir hiyerarşi mevcuttur. Yüksek öncelikli hedeflerin optimum değerleri daha düşük öncelikli hedefler tarafından bozulmamalıdır.

2.2.2.3. Minmaks Hedef Programlama

Hedef Programlamanın bu tipinde ağırlıklı ve öncelikli yapılardan farklı olarak sapan değişkenlerin toplamalarının minimizasyonu yerine maksimum sapmanın minimizasyonu gerçekleştirilir. Modelde hedefler ayrı ayrı gösterilir ve öncelik sıralaması/ağırlıklandırma yapılmaksızın geleneksel simpleks algoritması kullanılarak çözüm yapılır. Modelin amaç fonksiyonu sadece maksimum sapmanın minimizasyonunu belirleyen uzaklık parametresinden oluşur (İgnizio ve Cavalier, 1994: 525). Bu yöntem ilk kez Flavell (1976) tarafından önerilmiştir. Matematiksel olarak aşağıdaki gibidir (Flavel, 1976: 731-732):

Min D

Kısıtlar; (2.21)

$$(w_i^- d_i^- + w_i^+ d_i^+) - D \leq 0$$

$$\sum_{j=1}^n a_{ij} x_j + d_i^- - d_i^+ = b_i$$

$$x_j, d_i^-, d_i^+ \geq 0$$

$$i = 1, \dots, m \text{ ve } j = 1, \dots, n$$

Burada;

D : minimum değeri araştırılan maksimum sapma miktarı

w_i^+ : pozitif sapan değişkenin ağırlığı

w_i^- : negatif sapan değişkenin ağırlığı

Minmaks Hedef Programlama Chebyshev Hedef programlama olarak da isimlendirilir. Bu yöntemin Öncelikli ve Ağırlıklı Hedef Programlama yöntemlerinin kullandığı D_1 metriği yerine D_∞ metriği kullanılır (Romero, 1985: 249-251).

2.2.3. Hedef Programlama Modelinde Normalizasyon

Çok amaçlı karar verme problemlerinde her bir amaç fonksiyonunun birimleri arasında farklılıkların bulunabileceği göz önünde bulundurularak normalleştirme işlemi yapılması gerekir. Normalleştirme işlemi yapmak için birkaç yöntem mevcuttur. Bunlardan bazıları yüzde, sıfır-bir, öklit ve toplam normalizasyondur (Jones ve Tamiz, 2010: 34-38).

2.2.3.1. Yüzde Normalizasyon

Yüzde normalizasyonda her bir sapma bir yüzde değerine dönüştürülür ve sapmalar benzer birimlerde ölçülür. Yüz ile bölünen hedef değer $N_i = b_i/100$ normalizasyon sapması olarak ifade edilir ve bütün sapmaların yüzde ölçekle ölçüldüğünü gösterir. Bu yolla hedeflerden sapmalar toplamının tamamının yüzdesi belirlenir. Ancak bu yöntemde hedef değerlerinin doğru bir şekilde ayarlanması gerekmektedir.

2.2.3.2. Sıfır-Bir Normalizasyon

Bu yöntemde arzu edilmeyen bütün sapmalar sıfır-bir dizisinde ölçülmektedir. Belirlenen normalizasyon sapması amaçlardan ziyade sınırlandırılan sapma değişkenlerle birlikte düşünülür. Normalizasyon sapması “sıfır” (hedef) ve “bir” (en kötü olasılık) arasındaki bir ölçüyle bütün sapmaları ölçer. Sıfır-Bir metodu hedeflerin başarısızlığının bir ölçümü olup amaç fonksiyonu değerine yeniden anlam kazandırır.

2.2.3.3. Öklit Normalizasyon

Bu yöntemde normalizasyon sabiti her bir hedefin teknik katsayılarının Öklit ölçüsüdür ve $N_i = \sqrt{\sum_j a_{ij}^2}$ şeklinde ifade edilir. Öklit normalizasyon çok hesaplı ve güvenilirliği olan bir metottur (Tamiz ve diğerleri, 1998: 569-581).

2.2.3.4. Toplam Normalizasyon

Bu metot da normalizasyon sapması amaçlardaki teknik katsayıları içerir ve $N_i = \sum_j |a_{ij}|$ şeklinde ifade edilir. Bu yöntem öklit metottan daha büyük bir bölene sahiptir ve ölçülebilirlik açısından daha uygun problemleri bulur. Öklit metotla yakın seviyede güvenilirliğe sahiptir ama başarı fonksiyonu değerini yeniden anlamlandırmaz.

2.2.3.5. Ölçekleme Fonksiyonu İle Normalizasyon

Yukarıda sıralanan normalizasyon yöntemlerinden başka pozitif ideal çözüm ve negatif ideal çözümler kullanılarak ölçekleme fonksiyonu yardımıyla farklı bir normalizasyonda gerçekleştirilebilir. Maksimizasyon yönlü amaçlar için ölçekleme fonksiyonu;

$$S_k(d_k) = \frac{Z_k^* - Z_k(x)}{Z_k^* - Z_k^-} \quad (2.22)$$

ve minimizasyon yönlü amaçlar için ölçekleme fonksiyonu;

$$S_s(d_s) = \frac{W_k(x) - W_s^*}{W_s^- - W_s^*} \quad (2.23)$$

tanımlanır(Zeleny, 1974: 479-496). Hedef Programlama yaklaşımlarının çözümünde ise normalizasyon için

$$d_k = Z_k^* - Z_k^- \quad (2.24)$$

ve minimizasyon yönlü amaçlar için

$$d_s = W_s^- - W_s^* \quad (2.25)$$

normalizasyon sabiti olarak kullanılabilir.

Jones ve Tamiz (2009), yukarda kısaca açıklanan normalizasyon yöntemlerini ve uygulanabilirliğini tanımlamışlardır. Doğrusal Hedef Programlama yaklaşımının temel yöntemlerinden birisi olan Minmaks Hedef Programlamada normalizasyon sabiti aşağıdaki gibi değerlendirilir.

Min D

Kısıtlar; (2.26)

$$\left(\frac{w_i^- d_i^-}{Z_k^* - Z_k^-} + \frac{w_i^+ d_i^+}{Z_k^* - Z_k^-} \right) - D \leq 0$$

$$(w_i^- d_i^- + w_i^+ d_i^+) - D \leq 0$$

$$\sum_{j=1}^n a_{ij} x_j + d_i^- - d_i^+ = Z_k^*$$

$$x_j, d_i^-, d_i^+ \geq 0$$

$$i = 1, \dots, m \text{ ve } j = 1, \dots, n$$

Burada;

D : minimum değeri araştırılan maksimum sapma miktarı,

w_i^+ : pozitif sapan değişkenin ağırlığı,

w_i^- : negatif sapan değişkenin ağırlığı,

k_i : ilgili normalizasyon yöntemlerinden elde edilen sabit.

Z_k^* : k-ıncı amacın pozitif ideal çözümü

Maksimizasyon yönlü amaçlarda pozitif ideal çözümler hedef değerler olarak atanması sebebiyle, bu amaç fonksiyonu bir hedef olarak;

$$\sum_{j=1}^n a_{ij} x_j + d_i^- - d_i^+ = Z_k^* \quad (2.27)$$

yazılabilir. Diğer yandan hedef pozitif ideal çözümün üzerine çıkamayacağı için pozitif sapma (d_i^+) sıfıra eşittir. Benzer şekilde minimizasyon yönlü amaç;

$$\sum_{j=1}^n a_{ij}x_j + d_i^- - d_i^+ = W_s^* \quad (2.28)$$

yazılabilir. Bu tip amaç fonksiyonlarında ise pozitif ideal çözümün altına düşülemeyeceği için negatif sapma (d_i^-) sıfır olmalıdır. Bu bilgilere göre (2.26) model aşağıdaki gibi yeniden düzenlenir.

Min D

Kısıtlar; (2.29)

$$\sum_{j=1}^n a_{ij}x_j + d_i^- - d_i^+ = Z_k^*$$

$$\left(\frac{d_i^-}{Z_k^* - Z_k^-} \right) - D \leq 0$$

$$\sum_{j=1}^n a_{ij}x_j + d_i^- - d_i^+ = W_s^*$$

$$\left(\frac{d_i^+}{W_s^- - W_s^*} \right) - D \leq 0$$

$$x_j, d_i^-, d_i^+ \geq 0$$

$$i = 1, \dots, m \text{ ve } j = 1, \dots, n$$

2.2.4. Bulanık Hedef Programlama Yaklaşımları

Standart bir Hedef Programlama formülasyonunda hedefler ve kısıtlar açık ve kesin olarak tanımlanarak verilen bir çevre yardımı ile birden fazla amacın optimal gerçekleşmesi araştırılır. Hedefler, kesin ve matematiksel eşitlikler kullanılarak belirlenen hedef değerlere dayanılarak formülasyonu yapılır. Hedef Programlama içerisine Bulanık Küme Teorisinin uygulanmasındaki en önemli avantaj, karar vericinin bulanık hedef değerlerinin belirlenmesidir. Çok Amaçlı Karar Vermenin bu tekniğindeki bir diğer önemli avantaj, hedefler ve kısıtların tamamen simetrik olarak oluşturulmasıdır (Lai ve Hwang, 1994: 58).

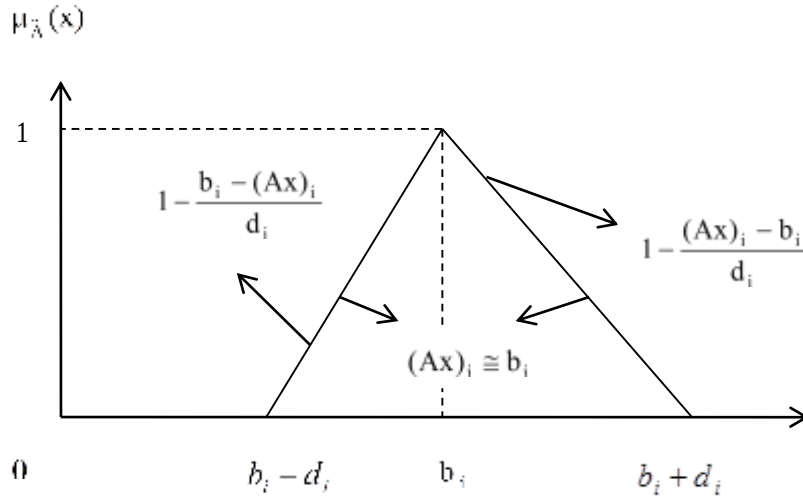
Bulanık Küme Teorisinin gelişimi, Bellman ve Zadeh (1970) tarafından önerilen Bulanık Ortamda Karar Verme süreci ile birlikte Doğrusal Programlamada olduğu gibi Hedef Programlama yöntemlerinde de birçok yaklaşım geliştirilmiştir.

Bulanık Hedef Programlamada ilk çalışma Narasimhan (1980) tarafından başlatılmış, Hannan (1981), R.N. Tiwari, S. Dharmar ve J. R. Rao (1987) T. Yang, J.P. Ignizio, H.J. Kim, (1991), Yaghoobi ve Tamiz (2007) tarafından farklı yaklaşımlar da önerilmiştir. Bu yaklaşımlar Zimmermann (1978) tarafından önerilen doğrusal üyelik fonksiyonlarına bağlı olarak açıklanmıştır.

2.2.4.1. Narasimhan Yaklaşımı

Narasimhan (1980) bulanık küme teorisini kullanarak ilk kez Bulanık Hedef Programlama formülasyonunu gerçekleştirmiştir. Narasimhan hedef değerleri bulanık hedefler olarak kabul ederek üçgensel üyelik fonksiyonu olarak tanımlamıştır. Narasimhan'ın yaklaşımında hedefler arasında bir öncelik sıralaması ya da ağırlıklandırma kullanılmamıştır. Narasimhan yaklaşımında üçgensel üyelik fonksiyonu artan ve azalan şeklinde iki alt probleme dönüştürmüştür. Bu yaklaşım için üçgensel üyelik fonksiyonunun grafiği Şekil 8' de gösterilmiştir.

Şekil 8: Üçgensel Üyelik Fonsiyonu



Kaynak: Özkan, 2003: 189

Şekil 8' de verilen üçgensel üyelik grafiğinin üyelik fonksiyonu aşağıda verilmiştir.

$$\mu_D(x^M) = \max_{x \geq 0}(\min[\mu_i, (x)]) = \begin{cases} \max_{x \geq 0} \left\{ \min \left[1 - \frac{b_i - (Ax)_i}{d_i} \right] \right\} & , & \max_{x \geq 0} \left\{ \min \left[1 - \frac{(Ax)_i - b_i}{d_i} \right] \right\} \\ K & , & K \\ b_i - d_i \leq (Ax)_i \leq b_i & , & b_i \leq (Ax)_i \leq b_i + d_i \\ i = 1, 2, \dots, m_1 & , & i = 1, 2, \dots, m_1 \end{cases} \quad (2.30)$$

(2.30)' a göre Narasimhan yaklaşımını matematiksel olarak

Maksimum λ

Kısıtlar; (2.31)

$$\left. \begin{aligned} 1 - \frac{b_i - (Ax)_i}{d_i} &\geq \lambda \\ b_i - d_i &\leq (Ax)_i \leq b_i \end{aligned} \right\} \text{Artan Kısım}$$

$$\left. \begin{aligned} 1 - \frac{(Ax)_i - b_i}{d_i} &\geq \lambda \\ b_i &\leq (Ax)_i \leq b_i + d_i \end{aligned} \right\} \text{Azalan Kısım}$$

$$\lambda \in [0, 1], x \geq 0$$

yazılır.

2.2.4.2. Hannan Yaklaşımı

Zimmerman modelinin artan parça(kısım) ismiyle nitelendirilen kısıtlayıcıları ele alınsın. Burada $(Ax)_i \leq b_i$ ifadesi d_i ile gösterilen tolerans miktarına bölüldüğü zaman $\frac{(Ax)_i}{d_i} \leq \frac{b_i}{d_i}$ ifadesi elde edilir. Bu eşitsizliğin sol tarafına pozitif sapma değişkenini gösteren p_i eklenirse $\frac{(Ax)_i}{d_i} + p_i = \frac{b_i}{d_i}$ eşitliğine ulaşılır. Bu eşitliği de p_i değişkenine göre düzenlenince $p_i = \frac{b_i}{d_i} - \frac{(Ax)_i}{d_i}$ ifadesi oluşturulur. Pozitif sapma değişkeni $1 - \frac{b_i - (Ax)_i}{d_i} \geq \lambda$ eşitsizliğinde yerine yazıldığı zaman, $\lambda + p_i \leq 1$ kısıtlayıcısı elde edilir. Yukarıda verilen modelin azalan parça ismiyle nitelendirilen kısıtlayıcıları için de benzer işlemler yapılabilir. Burada $b_i \leq (Ax)_i$ veya $(Ax)_i \geq b_i$ ifadesi tolerans miktarı d_i ye bölüldüğü zaman, $\frac{(Ax)_i}{d_i} \leq \frac{b_i}{d_i}$ ifadesi elde edilmiş

olur. Bu eşitsizliğin sol tarafından negatif sapma değişkeni olan n_i çıkarıldığı zaman, $\frac{(Ax)_i}{d_i} - n_i = \frac{b_i}{d_i}$ eşitliğine ulaşılır. Bu eşitlik de negatif sapma değişkenine göre düzenlenirse, $n_i = \frac{(Ax)_i}{d_i} - \frac{b_i}{d_i}$ ifadesi oluşur.

Negatif sapma değişkenini, $1 - \frac{(Ax)_i - b_i}{d_i} \geq \lambda$ eşitsizliğinde yerine koyulduğu zaman $\lambda + n_i \leq 1$ kısıtlayıcısı elde edilir. $(Ax)_i \leq b_i$ ve $(Ax)_i \geq b_i$ eşitsizlikleri için yapılan bu analizin $(Ax)_i = b_i$ durumuna genişletilmesi gerekmektedir. Hannan, bulanık bir eşitlik için belirlenen erişim düzeyine ne oranda ulaşıldığını belirlemek için $\lambda + p_i \leq 1$ ve $\lambda + n_i \leq 1$ kısıtlayıcılarını $\lambda + n_i + p_i \leq 1$ şeklinde bir araya getirmiştir. Bu düşünceden hareketle, eşitlik (2.32) da verilen bulanık hedef programlama modeli tek bir doğrusal programlama problemi olarak aşağıda verildiği gibi ifade edilir (Özkan, 2003: 190).

Maksimum λ

Kısıtlar; (2.32)

$$\frac{(Ax)_i}{d_i} + n_i - p_i = \frac{b_i}{d_i}$$

$$\lambda + n_i + p_i \leq 1$$

$$n_i \times p_i = 0$$

$$x_j, \lambda, n_i, p_i \geq 0 \quad i = 1, 2, \dots, m \quad j = 1, 2, \dots, n$$

2.2.4.3. MA Yaklaşımı

Klasik Hedef Programlama problemlerinde hedef değerler tam olarak belirlenir ve Hedef Programlama modelinde optimal çözümü bulmak için tam olarak belirlenen hedef değerlerinden sapma minimize edilir. Hedef Programlama problemlerini, Bulanık Hedef Programlama problemleriyle “~” sembolü dışında aynı olarak düşünebiliriz. MA yaklaşımı, Flavell (1976) tarafından önerilen Minmaks Hedef Programlamayı temel almıştır. Bu yaklaşım her bir hedeften maksimum sapmayı minimize eder. Bu farklı hedeflerin başarıları arasında en dengeli çözümü

yani optimal çözümü sağlar. (2.21) modeline bağlı olarak Minmaks tabanlı Bulanık Hedef Programlama modeli aşağıdaki gibi oluşturulur (Yaghoobi ve Tamiz, 2007: 1580-1590).

Bulanık Hedef Programlamada hedefler aşağıdaki gibi ifade edilir:

$$(AX)_i \lesseqgtr b_i, i = 1, 2, \dots, i_0 \quad (2.33)$$

$$(AX)_i \gtrless b_i, i = i_0 + 1, \dots, j_0 \quad (2.34)$$

$$(AX)_i \cong b_i, i = j_0 + 1, \dots, K. \quad (2.35)$$

Bu hedeflere göre doğrusal üyelik fonksiyonları aşağıdaki gibi tanımlanır.

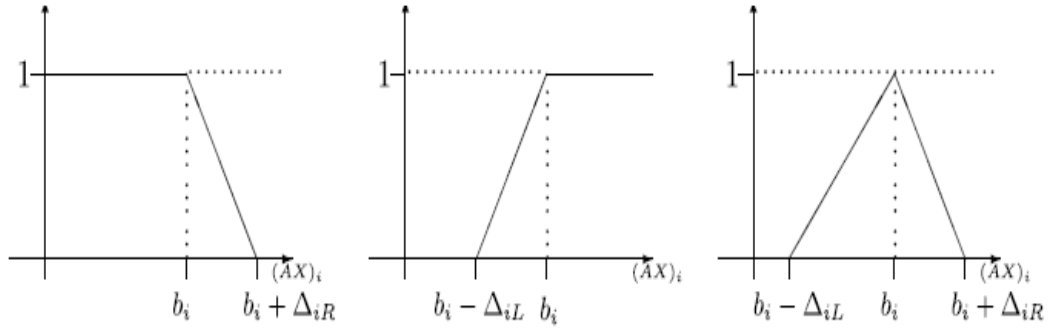
$$\mu_i(AX) = \begin{cases} 1 & , \quad (AX)_i \leq b_i \\ 1 - \frac{(AX)_i - b_i}{\Delta_{iR}} & , \quad b_i \leq (AX)_i \leq b_i + \Delta_{iR} \\ 0 & , \quad (AX)_i \geq b_i + \Delta_{iR} \end{cases} \quad (2.36)$$

$$\mu_i(AX) = \begin{cases} 1 & , \quad (AX)_i \geq b_i \\ 1 - \frac{b_i - (AX)_i}{\Delta_{iL}} & , \quad b_i - \Delta_{iL} \leq (AX)_i \leq b_i \\ 0 & , \quad (AX)_i \leq b_i - \Delta_{iL} \end{cases} \quad (2.37)$$

$$\mu_i(AX) = \begin{cases} 0 & , \quad (AX)_i \leq b_i - \Delta_{iL} \\ 1 - \frac{b_i - (AX)_i}{\Delta_{iL}} & , \quad b_i - \Delta_{iL} \leq (AX)_i \leq b_i \\ 1 - \frac{(AX)_i - b_i}{\Delta_{iR}} & , \quad b_i \leq (AX)_i \leq b_i + \Delta_{iR} \\ 0 & , \quad (AX)_i \geq b_i + \Delta_{iR} \end{cases} \quad (2.38)$$

Burada Δ_{iL} ve Δ_{iR} , karar vericiler tarafından belirlenen ve arzu edilen seviye b_i ’lerden kabul edilebilir maksimum sapma değeridir. Yukarıda matematiksel olarak verilen doğrusal üyelik fonksiyonları Şekil 9’de gösterilmiştir.

Şekil 9: Bulanık Hedefler İçin Doğrusal Üyelik Fonksiyonları



Kaynak: Yaghoobi ve Tamiz, 2007: 1580-1590

b_i , i. hedefe ait tam hedef değeridir, n_i, p_i tam hedef değerinden negatif ve pozitif sapmadır. Eğer $n_i(p_i)$ istenmeyen değerse $\alpha_i(\beta_i) = w_i/k_i$ istenmeyen sapmadır aksi halde $\alpha_i(\beta_i) = 0$ olacaktır. w_i ve k_i parametrelerinin i.hedefe ulaşmalarına bağlı olarak, öncelikli ve normalize edilmiş amaçlar ağırlıklandırılır. (2.21) modeli kullanılarak bulanık hedefler için aşağıdaki dönüşüm gerçekleştirilir: İlk olarak hem Bulanık A hem de Bulanık B hedefleri Bulanık C hedefinin özel bir durumudur. Bulanık C hedefi α_i ve β_i ye özel değerler atanarak (2.21) modeline eklenebilir.

$\alpha_i = \frac{1}{\Delta_{iL}}$ ve $\beta_i = \frac{1}{\Delta_{iR}}$ için (2.21) aşağıdaki gibi düzenlenir.

min D

Kısıtlar; (2.39)

$$\frac{1}{\Delta_{iL}} n_i + \frac{1}{\Delta_{iR}} p_i \leq D \quad i = 1, \dots, K$$

$$(AX)_i + n_i - p_i = b_i \quad i = 1, \dots, K$$

$$D \leq 1$$

$$D, n_i, p_i \geq 0 \quad i = 1, \dots, K$$

$$X \in C_s.$$

Şekil 9' daki bulanık hedefleri dikkate alınca $n_i \leq \Delta_{iL}$ ve $p_i \leq \Delta_{iR}$ koşulları sağlanmalıdır aksi halde istenmeyen çözümler elde edilebilir. Bulanık Hedef Programlama problemlerini çözmek için Minmaks Hedef Programlama yaklaşımı yukarıdaki modele sadece tek bir kısıt eklenmesini gerektirir. Bütün Hedef

Programlama modellerinin optimal çözümleri bulunurken n_i ve p_i sapmalarından sadece biri sıfırdan büyüktür. Bu durumda modele $D \leq 1$ in eklenmesiyle $\alpha_i = \frac{1}{\Delta_{iL}}$ ve $\beta_i = \frac{1}{\Delta_{iR}}$ olacak şekilde optimal çözüm $n_i \leq \Delta_{iL}$ ve $p_i \leq \Delta_{iR}$ şeklinde olur. Bu nedenle C bulanık hedefi gibi bütün bulanık hedeflerle Bulanık Hedef Programlama modelinin çözümü için yukarıdaki modelin bir uzantısı aşağıdaki gibidir. Diğer yandan Bulanık A ve B hedefleri için de benzer düşünce uygulanabilir. Bulanık B hedefi $\Delta_{iR} = \infty$ olduğunda bulanık C hedefinin özel bir hali olarak kabul edilebilir, bu şekilde (2.39) daki denklemlerde yerleştirilince bulanık B hedefi aşağıdaki gibi olur:

$$\begin{aligned} \frac{1}{\Delta_{iL}} n_i &\leq D \\ (AX)_i + n_i &\geq b_i . \end{aligned} \quad (2.40)$$

Önerilen modelde $D \leq 1$, D için basit bir sınırdır. Basit bir şekilde D yerine $\lambda = 1 - D$ yazılır. Çünkü $D \leq 1$ ve $1 - D \geq 0$ olduğundan $\lambda \geq 0$ olarak modele eklenmelidir. Ayrıca $\min 1 - \lambda$ yerine $\max \lambda$ olarak alınabilir. Sonuç olarak MA modeli olarak da adlandırılan Minmaks yaklaşımına göre Bulanık Hedef Programlama modeli:

MA

$$\max \lambda$$

$$\text{Kısıtlar;} \quad (2.41)$$

$$\begin{aligned} (AX)_i - p_i &\leq b_i \quad i = 1, \dots, i_0 \\ (AX)_i + n_i &\geq b_i \quad i = i_0 + 1, \dots, j_0 \\ (AX)_i + n_i - p_i &= b_i \quad i = j_0 + 1, \dots, K \\ \lambda + \frac{1}{\Delta_{iR}} p_i &\leq 1 \quad i = 1, \dots, i_0 \\ \lambda + \frac{1}{\Delta_{iL}} n_i &\leq 1 \quad i = i_0 + 1, \dots, j_0 \\ \lambda + \frac{1}{\Delta_{iL}} n_i + \frac{1}{\Delta_{iR}} p_i &\leq 1 \quad i = j_0 + 1, \dots, K \\ \lambda, n_i, p_i &\geq 0 \quad i = 1, \dots, K \\ X &\in C_s \end{aligned}$$

şeklinde yazılır.

ÜÇÜNCÜ BÖLÜM

UYGULAMA

3.1. KARAR PROBLEMİ

Bu çalışma çerçevesinde, bir yağ fabrikasının bir aylık üretim döneminde, işletme hedefleri doğrultusunda 8 farklı ağırlıkta ambalajlı mısır yağı dolumu yapmaktadır. Farklı ağırlıktaki bu ürünler 1 litrelik pet şişe, 2 litrelik pet şişe, 4,5 litrelik pet şişe, 4,5 litrelik pet kavanoz, 5 litrelik pet şişe, 5 litrelik teneke, 10 litrelik teneke ve 18 litrelik teneke şeklindedir. Uygulamaya konu olan işletme başka yerde bulunan tesislerinde üretmiş olduğu mısır yağını dolum makineleri yardımıyla satışa hazır hale getirmektedir. İşletme yönetimi aylık taleplere göre dolum yapması gereken farklı tipteki ürünlerde kullandığı mısır yağı miktarını optimum seviyede belirlemek istemektedir. İşletme yönetimiyle yapılan görüşmeler sonucunda üç farklı amaç belirlenmiştir. Bunlar minimum maliyet, maksimum kar ve maksimum birim üretimdir. Sekiz farklı tipteki ürünler için dolum hatları ve dolum süreleri Tablo 5'te verilmiştir.

Tablo 5: Ambalajlı Mısır Yağlarının Dolum Hatları ve Dolum Süreleri

Ürün	Hat Adı	Üretim Süresi(saniye)/Adet
x_1	DM1(EM1)	0,2
x_2	DM9	0,75
x_3	DM3	1,36
x_4	DM3	1,36
x_5	DM8	1,25
x_6	DM10	3
x_7	DM4	6
x_8	DM5	4

İşletme yönetimi dolum yapacağı ürünler için tam olarak 216000 saniye makine kullanım kısıtı belirlemiştir. Makine kullanım kısıtı için iki farklı kısıt daha belirlenmiştir. Bunlar:

- a) 1 litrelik pet şişe, 2 litrelik pet şişe ve 5 litrelik tenekenin dolumu için makine kullanım süresi en az 177120 saniyedir.
- b) 4,5 litrelik pet şişe, 4,5 litrelik pet kavanoz ve 10 litrelik tenekenin dolumu için makine kullanım süresi en fazla 28080 saniyedir.

Dolumu yapılan ürünler için işletme yönetimi 5250 m^3 'lük bir alan ayırmıştır. Üretilen bu sekiz farklı tipteki her bir ürün hacimleri Tablo 6'da verilmiştir.

Tablo 6: Ürünlerin Hacimleri(m^3)

Ürünler	Hacim(m^3)
1 litrelik pet şişe(x_1)	0,00192
2 litrelik pet şişe(x_2)	0,003993
4,5 litrelik pet şişe(x_3)	0,006264
4,5 litrelik pet kavanoz(x_4)	0,006272
5 litrelik pet şişe(x_5)	0,009216
5 litrelik teneke(x_6)	0,005670
10 litrelik teneke(x_7)	0,011
18 litrelik teneke(x_8)	0,01925

İşletmenin pazarlama departmanı ile yapılan görüşmeler sonucunda 2013-2014 yılları arasındaki Mayıs ayı satış rakamları Tablo 7'de verilmiştir. Bazı bitkisel ürünlerin GDO'lu olabileceği tartışmalarından dolayı taleplerde düşüş olacağı öngörülmüş ve öngörülen talepler de Tablo 7'de verilmiştir.

Tablo 7: Farklı Ambalajlı Ürünlerin 2013-2014 Yılları Mayıs Satış Rakamları(Adet)

Ürün	2013	2014	2015
x_1	60.000	55.000	En az 40000 En Fazla 60000
x_2	46.400	44.000	En fazla 38080
x_3	2.140	2.040	En az 1792
x_4	2.128	2.080	En az 1792
x_5	11.400	10.400	En az 8960
x_6	37.000	31.500	En az 28000
x_7	2.004	1.500	En az 1120
x_8	13.500	12.000	En fazla 10388

Tablo 7’de önceki yılları baz alarak işletmenin 2015 öngörüsünün sebepleri şu şekildedir: değişen tüketici eğilimi ve işletmeye karı açısından 4,5 litrelik pet kavonoz yerine 4,5 litrelik pet şişe daha çok satılmaktadır. Fazladan üretilbilecek 4,5 litrelik pet şişeler, 4,5 litrelik pet kavonoz isteyen müşterilerin büyük çoğunluğuna verilebilmektedir. Ürün çeşitliliğini azaltmamak adına üretimden 4,5 litrelik pet kavonoz kaldırılmamaktadır. Benzer şekilde alım gücü düşen tüketici sebebiyle, 18 litrelik teneke alan tüketici sayısında ciddi azalış gözlemlenmiş, bu yüzden üretimi için bir alt sınır konmamıştır. 5 litrelik teneke için diğer ürünler kadar azalış öngörülmüş ancak 18 litrelik tenekedeki azalışın 5 litrelik tenekeye kayabileceğini düşünen işletme 5 litrelik teneke için bir üst sınır koymamıştır.

Bu bilgilerin yanı sıra dolum hatlarındaki süre kısıtı sebebiyle 1litrelik pet şişe, 2 litrelik pet şişe ve 5 litrelik teneke toplam üretim adetiyle, 4,5 litrelik pet şişe, 5 litrelik pet şişe ve 18 litrelik teneke toplam üretim adeti arasındaki fark 117000 olmalıdır. Pazarlama departmanı talepler doğrultusunda 4,5 litrelik pet şişe, 4,5 litrelik pet kavonoz ve 10 litrelik teneke ambalajlı ürünlerden toplamda en az 3750 adet üretim yapmak istemektedir.

İşletme yönetimi bu kısıtlara bağlı olarak üç farklı amacını gerçekleştirmek istemektedir. Tablo 8’de her bir ürüne ait birim maliyet ve kar bilgileri verilmiştir.

Tablo 8: Farklı Ambalajlı Ürünlerin Birim Maliyet-Satış Fiyatı Tablosu(TL)

Ürün	Birim Maliyet(TL)	Satış Fiyatı(TL)	Birim Kar(TL)
x_1	3,80	4,40	0,6
x_2	7,55	8,70	1,15
x_3	16,75	19,30	2,55
x_4	17,00	19,50	2,5
x_5	18,45	21,25	2,8
x_6	18,70	21,50	2,8
x_7	37,20	42,80	5,6
x_8	65,75	75,60	9,85

Tablo 8’ deki bilgilerden minimum maliyet, maksimum kar ve maksimum birim üretim amaçları aşağıdaki gibi oluşturulur.

Minimum Maliyet Fonksiyonu;

$$W_1 = 3.80x_1 + 7.55x_2 + 16.75x_3 + 17.00x_4 + 18.45x_5 + 18.70x_6 + 37.20x_7 + 65.75x_8 \quad (3.1)$$

Maksimum Kar Fonksiyonu;

$$Z_1 = 0.6x_1 + 1.15x_2 + 2.55x_3 + 2.5x_4 + 2.8x_5 + 2.8x_6 + 5.6x_7 + 9.85x_8 \quad (3.2)$$

Maksimum Birim Üretim Fonksiyonu;

$$Z_2 = x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 \quad (3.3)$$

şeklindedir.

Yukarıda verilen amaç fonksiyonları ile Tablo 5 ve Tablo 6’daki parametrelerin yanı sıra makine kullanım süresi kısıtı, üretim kısıtı bilgilerine göre uygulamanın Çok Amaçlı Doğrusal Programlama Modeli aşağıdaki gibidir:

$$W_1 = 3.80x_1 + 7.55x_2 + 16.75x_3 + 17.00x_4 + 18.45x_5 + 18.70x_6 + 37.20x_7 + 65.75x_8$$

$$Z_1 = 0.6x_1 + 1.15x_2 + 2.55x_3 + 2.5x_4 + 2.8x_5 + 2.8x_6 + 5.6x_7 + 9.85x_8$$

$$Z_2 = x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8$$

Kısıtlar; (3.4)

$$0.2x_1 + 0.75x_2 + 1.36x_3 + 1.36x_4 + 1.25x_5 + 3x_6 + 6x_7 + 4x_8 = 216000$$

$$0.2x_1 + 0.75x_2 + 3x_6 \geq 177120$$

$$1.36x_3 + 1.36x_4 + 6x_7 \leq 28080$$

$$0.00192x_1 + 0.003993x_2 + 0.006264x_3 + 0.006272x_4 + 0.009216x_5 + 0.00567x_6 + 0.011x_7 + 0.01925x_8 \leq 5250$$

$$x_1 \geq 40000$$

$$x_2 \geq 38080$$

$$x_3 \geq 1792$$

$$x_4 \geq 1792$$

$$x_5 \geq 8960$$

$$x_6 \geq 28000$$

$$x_7 \geq 1120$$

$$x_8 \leq 10388$$

$$x_1 \leq 60000$$

$$x_8 \geq 2780$$

$$x_1 + x_2 + x_6 - x_3 - x_5 - x_8 = 117000$$

$$x_3 + x_4 + x_7 \geq 3750$$

3.2. UYGULAMANIN ÇÖZÜM AŞAMALARI

(3.4)'ün çözümünün gerçekleştirilebilmesi için ilk olarak her bir amaç fonksiyonu için pozitif ideal çözümler ve negatif ideal çözümler belirlenmiştir. Bu çözüm değerleri kullanılarak Minmaks Hedef Programlama ve Bulanık Hedef Programlamanın MA Yaklaşımı ile problemin tatminkar çözümü belirlenmiştir. Uygulamanın bütün matematiksel modellerinde LİNDÖ paket programı kullanılmıştır.

3.2.1. İdeal Çözümlerinin Belirlenmesi

Pozitif ideal çözüm amaç fonksiyonlarının en iyi performanslarını, negatif ideal çözüm ise amaç fonksiyonlarının en kötü performanslarını gösterir. (3.4)'ün ideal çözümleri Tablo 9'da verilmiştir.

Tablo 9: İdeal Çözüm Değerleri

İdeal Çözümler	$W_1(\text{TL})$	$Z_1(\text{TL})$	$Z_2(\text{adet})$
Pozitif İdeal Çözümler	1807652	283636,6	154919,1
Negatif İdeal Çözümler	1875879	273156,4	146976

Tablo 9'daki her bir amaç fonksiyonunun pozitif ideal çözümü karar değişkenlerinin farklı değerlerinde gerçekleşmiştir. Tablo 10'da her bir amaç fonksiyonuna ait karar değişkenlerinin değerleri verilmiştir.

Tablo 10: Her Bir Amaç Fonksiyonuna Ait Karar Değişkenleri

Karar Değişkenleri	$W_1(\text{TL})$	$Z_1(\text{TL})$	$Z_2(\text{adet})$
x_1	44224.000000	40007.000000	40042.000000
x_2	38084.000000	50828.000000	50787.000000
x_3	1792.000000	1792.000000	1792.000000
x_4	1792.000000	1793.000000	1793.000000
x_5	8960.000000	12928.031250	12931.034180
x_6	48224.000000	43666.000000	43674.000000
x_7	1121.000000	1120.000000	1120.000000
x_8	2780.000000	2781.000000	2780.000000

Tablo 10'daki verilerden her bir amaç fonksiyonu karar değişkenlerinin farklı değerlerinde gerçekleşmiştir. Bu ise Çok Amaçlı Optimizasyon problemlerinde beklenen bir sonuçtur.

Bir başka ifadeyle optimal çözüme ulaşılmamıştır. (3.4)'ün tatminkar çözümünün belirlenmesi için Minmaks Hedef Programlama ve Bulanık Hedef Programlama yaklaşımları kullanılmıştır.

3.2.2. Minmaks Hedef Programlama Çözümü

(3.4)'ün Minmaks yaklaşımına göre tatminkar çözümünün belirlenebilmesi için pozitif ve negatif ideal çözümler kullanılmıştır. Minmaks yaklaşımına göre kurulan yeni problemin çözümü negatif ve pozitif ideal çözümler arasında gerçekleşecektir. (2.29) modeline göre pozitif ve negatif ideal çözümler kullanılarak (3.4) modeli aşağıdaki gibi kurulur.

Min D

Kısıtlar; (3.5)

$$3.80x_1 + 7.55x_2 + 16.75x_3 + 17.00x_4 + 18.45x_5 + 18.70x_6 + 37.20x_7 +$$

$$65.75x_8 - p_1 = 1807652$$

$$p_1 - 68227D \leq 0$$

$$0.6x_1 + 1.15x_2 + 2.55x_3 + 2.5x_4 + 2.8x_5 + 2.8x_6 + 5.6x_7 + 9.85x_8 + n_2 \\ = 283636.6$$

$$n_2 - 10480.2D \leq 0$$

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + n_3 = 154919.1$$

$$n_3 - 7943.1D \leq 0$$

$$0.2x_1 + 0.75x_2 + 1.36x_3 + 1.36x_4 + 1.25x_5 + 3x_6 + 6x_7 + 4x_8 = 216000$$

$$0.2x_1 + 0.75x_2 + 3x_6 \geq 177120$$

$$1.36x_3 + 1.36x_4 + 6x_7 \leq 28080$$

$$0.00192x_1 + 0.003993x_2 + 0.006264x_3 + 0.006272x_4 + 0.009216x_5 + \\ 0.00567x_6 + 0.011x_7 + 0.01925x_8 \leq 5250$$

$$x_1 \geq 40000$$

$$x_2 \geq 38080$$

$$x_3 \geq 1792$$

$$x_4 \geq 1792$$

$$x_5 \geq 8960$$

$$\begin{aligned}
x_6 &\geq 28000 \\
x_7 &\geq 1120 \\
x_8 &\leq 10388 \\
x_1 &\leq 60000 \\
x_8 &\geq 2780 \\
x_1 + x_2 + x_6 - x_3 - x_5 - x_8 &= 117000 \\
x_3 + x_4 + x_7 &\geq 3750 .
\end{aligned}$$

(3.5)'in çözümünden belirlenen karar değişkenleri ve bunların değerleri Tablo 11' de verilmiştir. Ayrıca (3.5)'in amaç fonksiyon değeri $D = 0.4938363$ şeklinde belirlenmiştir.

Tablo 11: Minmaks Yaklaşımı Çözümüne Göre Karar Değişkeni Değerleri

Karar Değişkenleri	Karar Değişkeni Değerleri(adet)
x_1	48816.000000
x_2	38080.000000
x_3	4448.000000
x_4	1793.000000
x_5	8960.000000
x_6	46293.000000
x_7	1331.000000
x_8	2781.000000
p_1	33692.972656
n_2	5175.229980
n_3	2417.101318

Tablo 11'deki karar değişkeni değerlerine göre her bir amaç fonksiyonunun değeri Tablo 12'de verilmiştir.

Tablo 12: Minmaks Hedef Programlama Çözümü

Amaç Fonksiyonu	Pozitif İdeal Çözüm	Negatif İdeal Çözüm	Minmaks Tatminkar Çözüm
$W_1(TL)$	1807652,000000	1875879	1841344,972656
$Z_1(TL)$	283636,600000	273156,4	278461,370020
$Z_2(adet)$	154919,100000	146976	152501,998682

Her bir amaç fonksiyonu için Minmaks tatminkar çözümler pozitif ve negatif ideal çözümler arasında gerçekleşmiştir. Ayrıca (3.5)'in amaç fonksiyonu değeri $D=0.4938363$ olması sebebiyle tatminkar çözüm değerleri pozitif ideal çözümlere çok az yaklaşmıştır. Bir başka ifadeyle tatminkar çözüm değerleri yaklaşık olarak pozitif ve negatif ideal çözümlerin orta noktasında gerçekleşmiştir.

3.2.3. Bulanık Hedef Programlama Çözümü

(3.4)'ün Bulanık Hedef Programlada MA yaklaşımına göre çözümünün belirlenebilmesi için pozitif ve negatif ideal çözümler kullanılmıştır. Bulanık Hedef Programlama yaklaşımına göre kurulan yeni problemin çözümü negatif ve pozitif ideal çözümler arasında gerçekleşecektir. (2.41) modeline göre pozitif ve negatif ideal çözümler kullanılarak (3.4) aşağıdaki gibi kurulur.

$Max \lambda$

Kısıtlar; (3.6)

$$3.80x_1 + 7.55x_2 + 16.75x_3 + 17.00x_4 + 18.45x_5 + 18.70x_6 + 37.20x_7 +$$

$$65.75x_8 - p_1 = 1807652$$

$$p_1 + 68227\lambda \leq 68227$$

$$0.6x_1 + 1.15x_2 + 2.55x_3 + 2.5x_4 + 2.8x_5 + 2.8x_6 + 5.6x_7 + 9.85x_8 + n_2 = 283636.6$$

$$n_2 + 10480.2\lambda \leq 10480.2$$

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + n_3 = 154919.1$$

$$n_3 + 7943.1\lambda \leq 7943.1$$

$$0.2x_1 + 0.75x_2 + 1.36x_3 + 1.36x_4 + 1.25x_5 + 3x_6 + 6x_7 + 4x_8 = 216000$$

$$0.2x_1 + 0.75x_2 + 3x_6 \geq 177120$$

$$1.36x_3 + 1.36x_4 + 6x_7 \leq 28080$$

$$0.00192x_1 + 0.003993x_2 + 0.006264x_3 + 0.006272x_4 + 0.009216x_5 +$$

$$0.00567x_6 + 0.011x_7 + 0.01925x_8 \leq 5250$$

$$x_1 \geq 40000$$

$$x_2 \geq 38080$$

$$x_3 \geq 1792$$

$$x_4 \geq 1792$$

$$x_5 \geq 8960$$

$$x_6 \geq 28000$$

$$x_7 \geq 1120$$

$$x_8 \leq 10388$$

$$x_1 \leq 60000$$

$$x_8 \geq 2780$$

$$x_1 + x_2 + x_6 - x_3 - x_5 - x_8 = 117000$$

$$x_3 + x_4 + x_7 \geq 3750$$

(3.6)'nın çözümünden belirlenen karar değişkenleri ve bunların değerleri Tablo 13'te verilmiştir. Ayrıca (3.5)'in amaç fonksiyon değeri $\lambda = 0.505314$ şeklinde belirlenmiştir.

Tablo 13: Bulanık Yaklaşım Çözümüne Göre Karar Değişkenleri

Karar Değişkenleri	Karar Değişkeni Değerleri
x_1	48507.000000
x_2	38084.000000
x_3	4541.000000
x_4	1792.000000
x_5	8963.022461
x_6	46693.000000
x_7	1120.000000
x_8	2780.000000
p_1	33710.273438
n_2	5184.404785
n_3	2439.047119

Tablo 13’ teki karar değişkeni değerlerine göre her bir amaç fonksiyonunun değeri Tablo 14’ te verilmiştir.

Tablo 14: Bulanık Hedef Programlama Çözümü

Amaç Fonksiyonu	Pozitif İdeal Çözüm	Negatif İdeal Çözüm	Bulanık Tatminkar Çözüm
$W_1(TL)$	1807652,000000	1875879	1841362,273438
$Z_1(TL)$	283636,600000	273156,4	278452,1952
$Z_2(adet)$	154919,100000	146976	152480,0529

Tablo 14’ten de görüleceği üzere her bir amaç fonksiyonu için Bulanık çözümler pozitif ve negatif ideal çözümler arasında gerçekleşmiştir. Ayrıca (3.5)’in amaç fonksiyonu değeri $\lambda=0.505314$ olması sebebiyle çözüm değerleri pozitif ideal çözümlere çok az yaklaşmıştır. Yani çözüm değerleri yaklaşık olarak pozitif ve negatif ideal çözümlerin orta noktasında gerçekleşmiştir.

3.3. MİNMAKS HEDEF PROGRAMLAMA İLE BULANIK HEDEF PROGRAMLAMA SONUÇLARININ KARŞILAŞTIRILMASI

Tablo 15: Minmaks Hedef Programlama ve Bulanık Hedef Programlama Çözümleri

Karar Değişkenleri	Minmaks Hedef Programlama Çözümü	Bulanık Hedef Programlama (MA Yaklaşımı ile)
D/λ	$D = 0.4938363$	$\lambda = 0.5053144$
x_1	48816.000000	48507.000000
x_2	38080.000000	38084.000000
x_3	4448.000000	4541.000000
x_4	1793.000000	1792.000000
x_5	8960.000000	8963.022461
x_6	46293.000000	46693.000000
x_7	1331.000000	1120.000000
x_8	2781.000000	2780.000000
p_1	33692.972656	33710.273438
n_2	5175.229980	5184.404785
n_3	2417.101318	2439.047119

Her iki modele göre amaç fonksiyonlarının değerleri, pozitif ve negatif ideal çözümler arasında çıkmıştır ve Tablo 16’da gösterilmiştir

Tablo 16: Minmaks Hedef Programlama ve Bulanık Hedef Programlamaya Göre Amaç Fonksiyonlarının Değerleri

Amaç Fonksiyonu	Pozitif İdeal Çözüm	Minmaks Tatminkar Çözüm	Bulanık Tatminkar Çözüm	Negatif İdeal Çözüm
$W_1(TL)$	1807652,000000	1841344,972656	1841362,273438	1875879
$Z_1(TL)$	283636,600000	278461,370020	278452,1952	273156,4
$Z_2(adet)$	154919,100000	152501,998682	152480,0529	146976

MA yaklaşımı, Minmaks Hedef Programlamayı esas alarak geliştirilmiştir. Bu iki yaklaşım arasındaki en belirgin fark amaç fonksiyonu yönünden ortaya çıkmaktadır. Minmaks Hedef Programlamada pozitif ideal çözümlere uzaklığın minimizasyonunu araştırırken, MA yaklaşımı ise pozitif ideal çözümlere maksimum seviyede yaklaşmayı gerçekleştirir. Bu açıdan bakıldığı zaman her iki yaklaşım sadece matematiksel modelleme tekniği olarak birbirinden farklıdır. Bu sebeple analitik anlamda iki yaklaşım birbirinin aynısıdır. Uygulama probleminin her iki yaklaşım açısından birbirine çok yakın sonuçlar vermesinin temel sebebi aynı ideal çözümler ve aynı normalizasyon işleminin kullanılmasıdır.

Uygulama probleminde Minmaks Hedef Programlama modelindeki üyelik dereceleri $D = 0.4938363$ ve Bulanık Hedef Programlamanın MA yaklaşımı modelindeki $\lambda = 0.5053144$ değerleri arasında $\lambda = 1 - D$ eşitliğinin de neredeyse tam olarak sağladığı da gösterilmiştir. Aradaki küçük farkında karar probleminde fazla kısıt olmasından kaynaklanan doğal bir durumdur.

SONUÇ

Belirsizlik içeren problemleri ele alırken klasik karar verme problemlerinin yetersiz kaldığı yerlerde bulanık modellerle çözüm sıklıkla kullanılır hale gelmiştir. Bulanık modeller ikili karar yerine çoklu kararı baz aldığı için modelin gerçek hayata daha yakın kurulmasına ve çözümü daha esnek bir yapıda bulmaya olanak sağlar. Üretim yapan firmalar hangi üründen ne kadar üretmesi gerektiğini daha gerçekçi temsil edebilecek bulanık verileri nicel verilere çevirerek karar verme problemini çözebilirler. Birçok alanda da sıkça kullanılan Hedef Programlama ve Bulanık Hedef Programlama modelleri, birden çok hedef ve birçok kısıt içeren problemlerin çözümünde kolaylık sağlayan matematiksel modellerdir.

Bu çalışmada, İzmir’de mısır, kanola, fındık, çiçek ve zeytinyağının dolumunu ve satışını yapan işletmenin 2015 Mayıs ayındaki karar problemi incelenmiştir. İşletme için, üretiminin kısıt oluşturduğu öğrenilen mısır yağının durumu incelenmelidir. Ayrıca bazı bitkisel ürünlerin GDO’lu olabileceği tartışmalarından dolayı taleplerde düşüş olacağı öngörülmüş ve problemin kısıtları buna uygun olarak çözüm istemiştir. İşletme başka ilçedeki fabrikasında üretmiş olduğu mısır yağını ambalajlamaya hazır halde, dolum hattının başındaki kazana taşıma yoluyla getirmektedir. Dolum hatlarında 8 farklı ağırlıkta ambalajlanan mısır yağının perakende satışı asgari düzeyde olup büyük ve orta ölçekli marketlere ve otellere toptan satışı yapılmaktadır. İşletmeden ambalajlı mısır yağını en az maliyetle, en fazla kar elde etmek ve siparişlerin tümünün karşılanacağı bir üretimi gerçekleştirmek istediği bilinmektedir. Üreteceği ambalajlı mısır yağına dair hedefleri ve kısıtları işletmeden alınmış olup problem Minmaks Hedef Programlama ve Bulanık Hedef Programlamanın MA yaklaşımı ile ayrı ayrı çözülmüştür. Zaten bu iki yaklaşımın felsefesi aynıdır: hedeflerde oluşabilecek sapmanın minimizasyonunu gerçekleştirmektir. Bu iki yaklaşımla çözülen problemin çözüm değerleri yaklaşık olarak aynı çıkmıştır, çözüm değerleri Tablo 15’te yer almaktadır.

Problemin çözümünde pozitif ve negatif ideal çözümler arasındaki fark, tolerans değeri olarak belirlendi. Pozitif ideal çözüm amaç fonksiyonlarının en iyi performanslarını, negatif ideal çözüm ise amaç fonksiyonlarının en kötü performanslarını göstermektedir. Yani çıkan sonuç pozitif ideal çözümden daha iyi,

negatif ideal çözümden daha kötü bir performans sergileyemez. Bu yüzden ideal çözümlerin kullanılması ile çözüm hesaplanırken, sapan değişkenlerin daha büyük olmasının önüne geçilmiş oldu. Ele aldığımız problemin çözümü de, beklendiği gibi pozitif ve negatif ideal çözümlerin arasında Tablo 16'daki değerlerde elde edilmiştir.

İşletme, minimum maliyet hedefinden yaklaşık 33.700,00 TL daha pahalı bir dolum gerçekleştirmiştir. Diğer yandan, kısıtlar altındaki en yüksek kardan da yaklaşık 5.180,00 TL daha az kar elde edildiği görülmektedir. Bu da bir üretim döneminde işletme kısıtları altında, birden fazla olan hedefi için maliyetten artarak kasada kalabilecek ve daha fazla kar ile kasaya girebilecek olan toplam para miktarının yaklaşık 38.880,00 TL olduğunu gösterir. İşletmenin 1.841.350,00 TL'lik yani toplamda 152.500,00 tane ürün hazır hale getirebildiğini düşünürsek, bunun çok da büyük bir kayıp olmayacağı anlaşılır.

Problemin çözümünden elde edilen, hazır hale gelmiş her ürünün adet sayısı baz alındığında, dolum hattının en başında bulunan kazanda bu üretim dönemi için yaklaşık 492.694,00 litre mısır yağı bulundurmasının yeterli olacağı düşünülmektedir. Bu sayede, mısır yağını taşıyan aracın fazla yakıt kullanımının ve kazandaki gereğinden fazla yağın önüne geçilecektir. Aynı zamanda, her bir ürün çeşidinden kaçar tane satışa hazır olması gerektiğinin bilinmesi, işletmenin gereğinden fazla stok kullanmamasını, buna bağlı olarak depolama maliyetini düşürmesini ve son kullanma tarihleri yaklaşımadan satışını sağlayacaktır. Gıda ürünleri için son kullanma tarihlerinin, son tüketici için çok önemli olduğu da dikkate alınırsa bir anlamda daha uzun ömürlü ürünleri satmak, ürününün marka değerinin de arttırmasını sağlayacaktır.

Ayrıca, işletmenin 2015 Mayıs ayı için kurulan her iki model, sonraki yıllarda da değişen kısıtlar için ekle-çıkartma işlemi yapılarak uygun çözümler üretmesini sağlayacaktır. İşletme, bu çalışma kapsamında sayısal veriler desteği ile kurulan Hedef Programlama ve Bulanık Hedef Programlama modelleri kullanarak, dolum hızı kapasitesini daha net görebilecek ve tahmin edebilecektir.

İşletme, bu problemin dışında kalan darboğazlarını, örneğin mısır yağını ürettiği tesisindeki verimliliği de arttırabilirse, problemdeki iki hedef olan maliyetin düşmesini ve karın otomatik olarak artışı da sağlamış olur.

Uygulama içinde, mısır yağı için kullanılan çözüm algoritması, benzer şekilde diğer yağların dolumu için de kullanılması halinde, işletmenin hangi yağ için ne kadar kapasite kullanacağını bilmesine ve verimliliğini maksimize etmesine destek bulabilecektir. İşletmede, dolum hattının başında tek bir kazan olduğu düşünülürse, kazanın içindeki yağ çeşidinin daha az değiştirilip daha az vakit kaybı ile verimliliğin artışı da sağlanmış olacaktır.

Bu uygulama ile, bir işletme için hayati öneme sahip tipik bir karar problemine çözümler üretilmiştir. İşletmeler bütün faaliyetlerinde çoklu amaçlar taşıdıkları için, Bulanık Hedef Programlama Algoritmalarının güncel tutulması ve geliştirilecek uygun yazılımlar ile aktif çözümler üretilmesi yerinde olacaktır. Bu sayede, işletmelerin önemli karar problemlerini istedikleri kadar modifikasyona tabi tutabilmeleri de sağlanacağından, her türlü karar süreçleri de hızlanacaktır.

KAYNAKÇA

Babuska, R. (1998). *Fuzzy Modeling for Control*. USA: Kluwer Academic Publisher.

Bellman, R.E. ve Zadeh, L.A. (1970). Decision-Making İn A Fuzzy Environment. *Management Science*. 17(4): 141-164.

Charnes, A. ve Cooper W.W. (1961). *Management Models and Industrial Applications of Linear Programming*. New York: John Wiley & Sons.

Charnes, A. ve Cooper W.W. (1977). Goal Programming and Multiple Objective Optimization (Part I), *European Journal of Operational Research*, 1(1): 39-54.

Charnes, A., Cooper, W.W. and Ferguson, R.O. (1955). Optimal Estimation of Executive Compensation by Linear Programming. *Management Science*. 1(2): 138-151.

Cohon, J. (1978). *Multiobjective Programming and Planning*. New York: Academic Press.

Çelikyılmaz A. ve Türkşen, İ. B. (2009). *Modelling Uncertainty with Fuzzy Logic*. Berlin: Springer-Verlag.

Ertuğrul, İ. (2005). Bulanık Hedef Programlama ve Bir Tekstil Firmasında Uygulama Örneği. *Eskişehir Osmangazi Üniversitesi Sosyal Bilimler Dergisi*. 6(2): 45-79.

Flavell, R.B. (1976). A New Goal Programming Formulation. *Omega*. 4: 731-732.

Gal, T., Stewart, T.J. ve Hanne, T. (Eds.) (1999). *Multicriteria Decision Making: Advances in MCDM Models, Algorithms, Theory and Applications*. Boston: Springer.

Hannan, E.I. (1981). On Fuzzy Goal Programming. *Decision Sciences*. 12: 522–531.

Hwang, C.L. ve Masud A.S.M. (1979). Multiple Objective Decision Making - Methods and Applications: A State-of-the-Art Survey. New York: Springer-Verlag Berlin Heidelberg.

Hwang, C.L. ve Yoon, K. (1981). Multiple Attribute Decision Making - Methods and Applications: A State-of-the-Art Survey. New York: Springer-Verlag Berlin Heidelberg.

Ignizio, J. P. (1978). A Review of Goal Programming: A Tool for Multiobjective Analysis. *The Journal of the Operational Research Society*. 29(11): 1109-1119.

Ignizio, J.P. ve Cavalier, T.M. (1994). Linear programming. New Jersey: Prentice-Hall.

Ignizio, J. P. ve Romero, C. (2002). Goal Programming, Encyclopedia of Information Systems. *Elsevier Science*. 2: 489-500.

Jones, D. ve Tamiz, M. (2002). Goal Programming in the Period 1990-2000. *Multiple Criteria Optimization: State of the Art Annotated Bibliographic Surveys*(ss.130). Edited by Matthias Ehrgott and Xavier Gandibleux. USA: Kluwer Academics Publishers.

Jones, D. ve Tamiz, M. (2010). *Practical Goal Programming*. New York: Springer.

Jones, D. ve Tamiz, M. (2010). Formulating Goal Programmes. *Practical Goal Programming*(ss.34-38). Edited by Frederick S. Hillier. New York: Springer.

Klir, G.J. ve Yuan, B. (1995). *Fuzzy Set Theory: Foundations And Applications*. New York: Prentice-Hall.

Kökdemir, D. (2003). *Belirsizlik Durumlarında Karar Verme ve Problem Çözme*. (Yayınlanmamış Doktora Tezi). Ankara: Ankara Üniversitesi Sosyal Bilimler Enstitüsü.

Lai, Y.J. ve Hwang, C.L. (1992). *Fuzzy Mathematical Programming*. Berlin: Springer-Verlag.

Lai, Y.J. ve Hwang C.L. (1994). *Fuzzy Multiple Objective Decision Making: Methods and Applications*. Berlin: Springer-Verlag.

Lu, J., Zhang, G., Ruan, D. ve Wu, F. (2007). *Multi-Objective Group Decision Making*. London: Imperial College Press.

Min, H. ve Storbeck, J. (1991). On The Origin and Persistence of Misconceptions in Goal Programming. *Journal of The Operational Research Society*. 2(4): 301-312.

Mollaghasemi, M. ve Pet-Edwards, J. (1997). Technical Briefing Making Multiple Objective Decisions, Los Alamitos, IEEE Computer Society Press: 55-58.

Narasimhan, R. (1980). Goal programming in a Fuzzy environment. *Decision Science*. 11: 325-338.

Öztürk, A. (1987). *Yöneylem Araştırması*. Bursa: Uludağ Üniversitesi.

Özkan, M. M. (2003). *Bulanık Hedef Programlama*. Bursa: Ekin Kitabevi.

Romero, C. (1985). Multi-Objective And Goal Programming As A Distance Function Model. *Journal of the Operational Research Society*. 36(3): 249-251.

Romero, C. (1991). *Handbook of Critical Issues in Goal Programming*. New York: Pergamon Press.

Schniederjans, M. J.(1995). *Goal Programming: Methodology and Applications*. New York: Kluwer Academic Publishers.

Şen, Z. (2004). *Mühendislikte Bulanık (Fuzzy) Mantık ile Modelleme Prensipleri*. İstanbul: Su Vakfı Yayınları.

Tabucanon, M.T. (1988). *Multiple Criteria Decision Making In Industry*. New York: Elsevier.

Tamiz, M., Jones, D. ve Romero, C. (1998). Goal programming for decision making: an overview of the current state-of-the-art. *European Journal of Operational Research*. 111: 569-581.

Tiwari, R.N. , Dharmar, S. ve Rao, J.R. (1987). Fuzzy Goal Programming - An Additive Model. *Fuzzy Sets and Systems*. 24: 27-34.

Umarusman, N. (2007). *Çok Amaçlı Karar Problemlerinde Duyarlılık Analizi ve Bulanık Mantık İlişkisi: De Novo Programlama Uygulaması*. (Yayınlanmamış Doktora Tezi). İzmir: Dokuz Eylül Üniversitesi Sosyal Bilimler Enstitüsü.

Yaghoobi, M.A. ve Tamiz, M. (2007). A Method For Solving Fuzzy Goal Programming Problems Based on MINMAX Approach. *European Journal of Operational Research*. 177: 1580-1590.

Yang, T., Ignizio, J.P. ve Kim, H.J. (1991). Fuzzy Programming With Nonlinear Membership Functions: Piecewise Linear Approximation. *Fuzzy Sets and Systems*. 41: 39-53.

Yaralıoğlu, K. (2004). *Bulanık Mantık, Soft Yöneylem Teknikleri, Kişisel Sayfalar: Kaan Yaralıoğlu, Dokuz Eylül Üniversitesi*.
http://www.deu.edu.tr/userweb/k.yaralioglu/dosyalar/bul_man.doc, (14.11.2010).

Zadeh, L.A. (1973). Outline of a New Approach to the Analysis of Complex Systems and Decision Processes. IEEE Trans. Syst., Man. and Cybern. SMC-3(1).

Zeleny, M. (1974). A Concept Of Compromise Programming Solutions and The Method of The Displaced Ideal. Comput. & Ops. Res. 1: 479-496.

Zimmermann, H.J. (1978). Fuzzy Programming and Linear Programming with Several Functions. Fuzzy Sets and Sysytems. 1: 45-55.

Zimmermann, H.J. (1987). *Fuzzy Sets, Decicion Making and Expert Systems*. Boston: Kluwer Academic Publishers.