

**T.C.  
DOKUZ EYLÜL ÜNİVERSİTESİ  
SOSYAL BİLİMLER ENSTİTÜSÜ  
İŞLETME ANABİLİM DALI  
YÖNETİM BİLİMİ PROGRAMI  
YÜKSEK LİSANS TEZİ**

**ÇOK AMAÇLI DİNAMİK PROGRAMLAMA  
MODELİNİN ULAŞTIRMA SORUNLARINA  
UYGULANMASI**

**Gizem ERSÖZ**

**Danışman**

**Doç. Dr. Ali ÖZDEMİR**

**İZMİR - 2015**

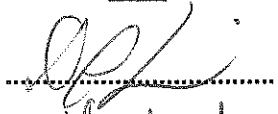
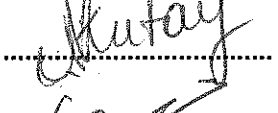
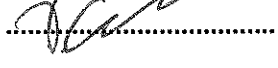
**YÜKSEK LİSANS**  
**TEZ/ PROJE ONAY SAYFASI**

2011800111

**Üniversite** : Dokuz Eylül Üniversitesi  
**Enstitü** : Sosyal Bilimler Enstitüsü  
**Adı ve Soyadı** : GİZEM ERSÖZ  
**Tez Başlığı** : Çok Amaçlı Dinamik Programlama Modelinin Ulaştırma Sorunlarına Uygulanması

**Savunma Tarihi** : 24.08.2015  
**Danışmanı** : Doç.Dr.Ali ÖZDEMİR

**JÜRİ ÜYELERİ**

| <b><u>Ünvanı, Adı, Soyadı</u></b> | <b><u>Üniversitesi</u></b> | <b><u>İmza</u></b>                                                                    |
|-----------------------------------|----------------------------|---------------------------------------------------------------------------------------|
| Doç.Dr.Ali ÖZDEMİR                | DOKUZ EYLÜL ÜNİVERSİTESİ   |  |
| Doç.Dr.Nilgün KUTAY               | DOKUZ EYLÜL ÜNİVERSİTESİ   |  |
| Yrd.Doç.Dr.Ibrahim GÜRLER         | GEDİZ ÜNİVERSİTESİ         |  |

Oybirliği ☒

Oy Çokluğu ( )

GİZEM ERSÖZ tarafından hazırlanmış ve sunulmuş "Çok Amaçlı Dinamik Programlama Modelinin Ulaştırma Sorunlarına Uygulanması" başlıklı Tezi ( ) / Projesi ( ) kabul edilmiştir.

**Prof.Dr. Utku UTKULU**  
**Enstitü Müdürü**

## YEMİN METNİ

Yüksek Lisans Tezi olarak sunduğum **“Çok Amaçlı Dinamik Programlama Modelinin Ulaştırma Sorunlarına Uygulanması”** 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.

30/07/2015

Gizem ERSÖZ

## **ÖZET**

**Yüksek Lisans Tezi**

### **ÇOK AMAÇLI DİNAMİK PROGRAMLAMA MODELİNİN ULAŞTIRMA SORUNLARINA UYGULANMASI**

**Gizem ERSÖZ**

**Dokuz Eylül Üniversitesi**

**Sosyal Bilimler Enstitüsü**

**İşletme Anabilim Dalı**

**Yönetim Bilimi Programı**

İşletmelerin birbiriyle çatışan birden fazla amaç ile karar vermesi gerektiğinde dikkate alınan matematiksel bir model olan çok amaçlı karar verme kantitatif karar verme tekniklerinden biridir. Söz konusu tekniğin kullanımında süreçte birden fazla amaç veya bir amacın çözümü için birden fazla seçenek karşımıza çıkmaktadır. Modelde hedef, çeşitli etkileşimleri göz önünde bulundurup, değer alabilen amaçların birleşimi ile arzu edilen düzeylere erişimi sağlamak ve bu sayede karar vericinin karşısına “en iyi” seçeneği sunmaktır.

Geleneksel en kısa yol problemlerinin geliştirilmiş bir şekli olan çok amaçlı en kısa yol problemi genel olarak çatışan iki veya daha fazla amaca ilişkin etkin yollar setinin bulunmasını sağlar. Çok amaçlı en kısa yol problemlerinin çözümünde kullanılan algoritmalarından biri olan Dinamik Programlama, probleme tek bir aşamada çözüm aramak yerine ana problemi daha küçük alt problemlere ayırarak, her yeni alt problem için bir çözüm üreterek ana problemi çözüme ulaştırır.

Bu çalışmada bir firmanın ulaştırma sorununda ikiden fazla amacı olması sebebiyle çok amaçlı dinamik programlama modeli ile karar vericinin optimum yolu belirlemesine yardımcı öneri geliştirilmiştir.

**Anahtar Kelimeler:** karar verme, çok amaçlı karar verme, en kısa yol problemi

## **ABSTRACT**

**Master's Thesis**

# **THE APPLICATION OF MULTI OBJECTIVE DYNAMIC PROGRAMMING MODEL TO TRANSPORTATION PROBLEMS**

**Gizem ERSÖZ**

**Dokuz Eylül University**

**Graduate School of Social Sciences**

**Department of Business Administration**

**Management Science Program**

In case of required to make decision with multiple conflicting objectives for business, multi objective decision-making which considered a mathematical model is one of the techniques of quantitative decision making. For using of this technique has encountered multiple objectives or more than one options to solve the objecti in the process. Target is that providing to reach on intended level by considering various interactions with the combination of objective which can take the value and thus offering the “best” option to decision-maker in model.

The multi-objective shortest path problem which kind of traditional shortest path problems usulally provide to find the set of efficient paths for conflicting two or more than two objectives. Dynamic Programming which is one of the algorithms used to solve multi –objective shortest path problems, instead of searching for solutions to the main problems in a single step, search the solutions by separating into smaller sub-problems and finding solutions for each of these sub-problems.

In this work, proposal improved with multiobjective dynamic programming model for finding the optimal path by decision maker because of more than two objectives in the transportation problem.

**Keywords:** decision making, multi-objective decision making, shortest path problem

# **ÇOK AMAÇLI DİNAMİK PROGRAMLAMA MODELİNİN ULAŞTIRMA SORUNLARINA UYGULANMASI**

## **İÇİNDEKİLER**

|                  |     |
|------------------|-----|
| TEZ ONAY SAYFASI | ii  |
| YEMİN METNİ      | iii |
| ÖZET             | iv  |
| ABSTRACT         | v   |
| İÇİNDEKİLER      | vi  |
| KISALTMALAR      | ix  |
| TABLolar LİSTESİ | x   |
| ŞEKİLLER LİSTESİ | xi  |
| GİRİŞ            | 1   |

## **BİRİNCİ BÖLÜM**

### **ÇOK AMAÇLI PROGRAMLAMA**

|                                                                                                         |    |
|---------------------------------------------------------------------------------------------------------|----|
| 1.1. ÇOK ÖLÇÜTLÜ KARAR VERME                                                                            | 4  |
| 1.1.1.Karar Verme                                                                                       | 5  |
| 1.1.2.Yönetmel Karar Verme Süreci                                                                       | 8  |
| 1.1.3.Çok Ölçütlü Karar Verme / Çok Kriterli Karar Verme Süreci ve Aşamaları                            | 11 |
| 1.1.4.Çok Ölçütlü Karar Verme Problemlerinin Sınıflandırılması: Çok Nitelikli ve Çok Amaçlı Karar Verme | 15 |
| 1.1.4.1.Çok Amaçlı Karar Verme                                                                          | 17 |
| 1.1.4.2.Çok Amaçlı Karar Vermede Temel Kavramlar                                                        | 19 |
| 1.2.ÇOK AMAÇLI PROGRAMLAMA                                                                              | 20 |
| 1.2.1.Çok Amaçlı Programlamanın Gelişimi Ve Uygulama Alanları                                           | 21 |

|                                                                             |    |
|-----------------------------------------------------------------------------|----|
| 1.2.2.Çok Amaçlı Programlama Problemlerinin Genel Formülasyonu ve Baskınlık | 25 |
| 1.2.2.1.Baskınlık                                                           | 28 |
| 1.2.3.Çok Amaçlı Programlama Sınıflandırması/Yöntemleri                     | 31 |
| 1.2.3.1.Tercih Bilgilerinin Kullanılmadığı Metot                            | 31 |
| 1.2.3.2.Tercih Bilgilerinin Öncelikli Kullanıldığı Metotlar                 | 32 |
| 1.2.3.3.Tercih Bilgilerinin Optimizasyon Sırasında Kullanıldığı Metotlar    | 32 |
| 1.2.3.4.Tercih Bilgilerinin Optimizasyon Sonrası Kullanıldığı Metotlar      | 33 |

## İKİNCİ BÖLÜM

### DİNAMİK PROGRAMLAMA

|                                                                        |    |
|------------------------------------------------------------------------|----|
| 2.1. DİNAMİK KARAR VERME                                               | 35 |
| 2.2. DİNAMİK PROGRAMLAMA                                               | 36 |
| 2.2.1.Dinamik Programlamanın Doğuşu ve Gelişimi                        | 40 |
| 2.2.2.Dinamik Programlamanın Uygulama Alanları                         | 41 |
| 2.2.3.Dinamik Programlama Kavramları                                   | 44 |
| 2.2.4.Dinamik Programlamanın Yinelenebilir Yapısı ve Optimallik İlkesi | 48 |
| 2.2.4.1.Optimallik İlkesi                                              | 49 |
| 2.2.5.Dinamik Programlama Formülü                                      | 50 |
| 2.2.6.Dinamik Programlama Çözüm Yöntemleri                             | 52 |
| 2.2.6.1.Geriye Doğru Hesaplama Yöntemi                                 | 53 |
| 2.2.6.2.İleriye Doğru Hesaplama Yöntemi                                | 54 |
| 2.2.6.3.Tablosal Çözüm Yöntemi                                         | 55 |
| 2.2.6.4.Analitik Çözüm Yöntemi                                         | 57 |
| 2.3. DETERMİNİSTİK DİNAMİK PROGRAMLAMA                                 | 57 |
| 2.4. STOKASTİK DİNAMİK PROGRAMLAMA                                     | 59 |

## ÜÇÜNCÜ BÖLÜM

### ÇOK AMAÇLI DİNAMİK PROGRAMLAMA MODELİNİN ULAŞTIRMA SORUNLARINA UYGULANMASI

|      |                                                                                     |    |
|------|-------------------------------------------------------------------------------------|----|
| 3.1. | ÇOK AMAÇLI DİNAMİK PROGRAMLAMA                                                      | 64 |
| 3.2. | LİTERATÜR TARAMASI VE UYGULAMA ALANLARI                                             | 65 |
| 3.3. | ÇOK AMAÇLI EN KISA YOL PROBLEMLERİ                                                  | 68 |
| 3.4. | ÇOK AMAÇLI DİNAMİK PROGRAMLAMA FORMÜLASYON                                          | 69 |
|      | 3.4.1.Ölçekleme Yaklaşımları                                                        | 71 |
|      | 3.4.1.1.Ağırlıklandırılmış Yaklaşım                                                 | 72 |
|      | 3.4.1.2.Kısıt Yaklaşımı                                                             | 72 |
|      | 3.4.1.3.Amaç Programlama Yaklaşımı                                                  | 73 |
| 3.4. | UYGULAMA                                                                            | 74 |
|      | 3.4.1.Firma Bilgileri                                                               | 75 |
|      | 3.4.2.Problemin Tanımlanması                                                        | 75 |
|      | 3.4.3.Problemin Notasyonu ve Varsayımları                                           | 76 |
|      | 3.4.4.Problemin Ağ Diyagramı ile Gösterimi ve Dinamik Programlama Modeli ile Çözümü | 79 |
|      | SONUÇ                                                                               | 89 |
|      | KAYNAKÇA                                                                            | 94 |

## KISALTMALAR

|              |                                     |
|--------------|-------------------------------------|
| <b>ÇKKVY</b> | Çok Kriterli Karar Verme Yöntemleri |
| <b>ÇKKV</b>  | Çok Kriterli Karar Verme            |
| <b>s.</b>    | Sayfa No                            |
| <b>fonk.</b> | Fonksiyon                           |
| <b>ÇNKV</b>  | Çok Nitelikli Karar Verme           |
| <b>ÇAKV</b>  | Çok Amaçlı Karar Verme              |
| <b>VMP</b>   | Vektör Minimizasyon Problemi        |
| <b>KV</b>    | Karar Verici                        |
| <b>DP</b>    | Dinamik Programlama                 |
| <b>min.</b>  | Minimum                             |
| <b>max.</b>  | Maksimum                            |
| <b>ss.</b>   | Sayfadan sayfaya                    |

## TABLÖLER LİSTESİ

|                                                                                                            |       |
|------------------------------------------------------------------------------------------------------------|-------|
| <b>Tablo 1:</b> Çok Nitelikli Karar Verme ve Çok Amaçlı Karar Verme Problemlerinin Karşılaştırılması       | s. 16 |
| <b>Tablo 2:</b> Baskınlık Örnek Problemi                                                                   | s. 29 |
| <b>Tablo 3:</b> Şehirlerarası Gerçekleştirilen Nakliye Başına Maliyetin Hesaplanması (ilk aşama için)      | s. 78 |
| <b>Tablo 4:</b> Şehirlerarası Gerçekleştirilen Nakliye Başına Maliyetin Hesaplanması (ikinci aşama için)   | s. 78 |
| <b>Tablo 5:</b> Şehirlerarası Gerçekleştirilen Nakliye Başına Maliyetin Hesaplanması (üçüncü aşama için)   | s. 79 |
| <b>Tablo 6:</b> Şehirlerarası Gerçekleştirilen Nakliye Başına Maliyetin Hesaplanması (dördüncü aşama için) | s. 79 |
| <b>Tablo 7:</b> Belirlenen Optimal Yolların Karşılaştırılması                                              | s. 86 |
| <b>Tablo 8:</b> Belirlenen Optimal Yolların Oransal Karşılaştırılması                                      | s. 87 |

## ŞEKİLLER LİSTESİ

|                                                                                                 |       |
|-------------------------------------------------------------------------------------------------|-------|
| <b>Şekil 1:</b> Yönetmel Karar Verme Süreci                                                     | s. 9  |
| <b>Şekil 2:</b> Karar Verme Problemlerinin Sınıflandırılması                                    | s.10  |
| <b>Şekil 3:</b> Tipik Çok Amaçlı (Kriterli) Karar Süreci Aşamaları                              | s. 13 |
| <b>Şekil 4:</b> Baskın Kümenin Görünümü                                                         | s. 30 |
| <b>Şekil 5:</b> Tablosal Çözüm Yöntemi                                                          | s. 56 |
| <b>Şekil 6:</b> Deterministik Dinamik Programlamanın Temel Yapısı                               | s.58  |
| <b>Şekil 7:</b> Olasılıklı Dinamik Programlamanın Temel Yapısı                                  | s. 62 |
| <b>Şekil 8:</b> Türkiye Karayolları Haritası'nda Firmanın Muhtemel Nakliye Yollarının Gösterimi | s.77  |
| <b>Şekil 9:</b> Çok Amaçlı Dinamik Programlama Yol Problemi                                     | s. 80 |

## GİRİŞ

Artan rekabet koşulları ile birlikte şirketlerin varoluş veya yok oluş eğilimleri bünyesinde bulundurdıkları yöneticilerin aldığı kararların verimliliği ile yakından ilgilidir. Şirket bünyesinde yönetim sorumluluğu almış birey şirketin ömrünü dahi etkileyebilecek kararlar alma durumundadır. Günümüzde bilim ve teknolojiye gelişmeler ve şirket yönetimlerinin geleneksel yapısının değişimi gibi etkenler karar alma konumunda bulunan bireyin, vereceği kararı kişisel deneyim kaynaklı alma sorumluluğundan kendini uzak tutup kantitatif yaklaşıma yönelme eğilimine geçmesine neden olmuştur.

İşletmeler doğası gereği karar sürecinde birbiriyle çatışan birden fazla amaç ile karşı karşıya kalırlar. Çok amaçlı karar vermede karşımıza çıkan ilk temel kavram amaçtır. Amaç genel anlamıyla karar vericinin gerçekleştirmeye çalıştığı nihai sonuçtur. Süreç karar vericinin istekleri yani bünyesinde bulunduğu işletmenin çıkarı doğrultusunda maksimize veya minimize edilmek istenen nitelikleri içinde bulundurmaktadır. Karar verici sonlu çözüm kümesinden tüm hedefler için kabul edilebilir bir sonuç sağlayacak çözümü seçmelidir. Amacın somutlaşmış haline hedef demek çok yanlış olmaz. Hedef ise amaca yönelik çalışmaların getireceği başarının belirli bir ölçütle sınırlandırılmış halidir. Bu iki kavrama bakarak karar alma süreci değerlendirildiğinde amacın sorunların çözümünde, hedefin ise sistematik çalışma ve motivasyon adına önemli etmenler olarak karşımıza çıktığını söyleyebiliriz.

Çok amaçlı bir optimizasyon problemi birden fazla amaç fonksiyonu ile ilgilenir. Bu yönü ile tek amaçlı bir optimizasyon probleminden farklıdır. Tek amaçlı bir optimizasyon problemi için sabit olarak hedef en iyi çözüm planıyken çok amaçlı bir optimizasyon için tek bir optimal çözümü olmayan bir çok hedef söz konusudur.

Çok kriterli karar verme problemleri yapısı itibarıyla dinamiklidir. Bunun nedeni problemlerde değişkenlerin durumlarını ve mevcut sistemin aşamalarının kontrolünü ifade eden kararların varlığıdır. Karar değişkenleri sisteminin dinamikleri arasında etkileşim içindedir. Dinamik kararlar çatışan amaçlar doğrultusunda ve

belirsiz zaman aralıklarında alınan kararların, süreçteki sonraki kararları etkilediği karar süreçleri olarak ifade edilmektedir.

Karar alma zamanlarının bir süreç içerisinde belirsiz bir dizin oluşturması sıralı amaçları belirlemek için ve buna bağlı olarak karar verme sistemi için büyük zorluklara neden olmaktadır. Bu zorluğu ortadan kaldırmaya en uygun biçimsel yaklaşım olarak Dinamik programlama göze çarpmaktadır. Doğrusal programlamadan daha ileride olan dinamik programlama bir işlem yerine birden çok işlemi sıra ile elde edilebilen bir planlama ve programlama yöntemidir.

Diğer tekniklerden farklı olarak karar problemine belirli bir aşamada çözüm aranmaz. Mevcut problemi küçük problemlere ayırarak, her yeni küçük problem için bir çözüm üretir. Bulunan her alt problemin optimum çözümünden sonra ana probleme ait optimum çözüm arayışı başlar.

Bu yönüyle dinamik programlama bir çok optimizasyon probleminin çözümünde kullanılabilen bir teknik olarak göze çarpmaktadır. Yöntemin içinde zaman değişkeninin açık olarak ele alınması yeni süreç içinde ardışık kararların yer alması yönüne “dinamik” isminin verilmesine neden olmuştur. 1920’den bu yana önemini arttırarak gelişmekte olup başlangıçta sadece ekonomik sistemin zaman içindeki durumunu incelemeye kullanılan bir yöntemken günümüzde zamanla ilgili alanların yanı sıra değişik nitelikli süreçlerin incelenmesinde de yaygınca kullanılmaya başlanmıştır. Dinamik programlama yaklaşımı ardışık kararlar içeren problemleri birbiriyle ilişkili alt problemlere ayrılarak kolay bir şekilde çözümlemeye imkan tanır. Matematiksel programlama problemlerine uygulanabilen esnek bir yapıya sahiptir. Bu özelliğiyle bir teknikten ziyade bir optimizasyon yaklaşımı olarak kendini gösterir. Deterministik ve stokastik süreçlere uygulanabilen dinamik programlama problemin çözümünde olası tüm seçeneklerin dikkate alınması yerine çok daha az hesaplama yapma kolaylığı sağlamaktadır.

En kısa yol problemleri genellikle tek amaçlı optimizasyon çerçevesinde yer alır ancak pek çok gerçek uygulamada tek bir amaç bulunmaması sebebiyle çok amaçlı en kısa yol problemleri olarak ele alınır. Çok amaçlı dinamik programlama

tekniđi mevcut dinamik programlama kavramlarının bir nevi genişletilmiş şeklidir. Çok amaç kavramında yer alan birden çok amaç için, durum uzayları tanımlama ve n aşamalı optimizasyon problem çözümlemesini dinamik programlamaya dahil edildiđi yaklaşımı ele alındığında problemin boyutunun ve hesaplama zorluđunun arttığı görölmektedir. Buna rağmen karar verme sorunlarının çözümünde oldukça etkindir.

Bu tez çalışmasında çok amaçlı dinamik programlama modeli teorik olarak incelenmiş ve işletmelerin çok amaçlı ulaştırma sorunlarına dinamik programlama modeli ile çözüm sunulmuştur.

Tezin birinci bölümünde, öncelikle yönetsel karar verme süreci üzerinde durulmuş ve karar verme probleminin elemanları açıklanmıştır. Daha sonra karar problemleri sınıflandırılarak, çok ölçütlü karar verme problemlerinin karar süreci aşamaları, sınıflandırılması ve temel kavramları ile birlikte ele alınmıştır. Birinci bölümün devamında çok amaçlı programlama ayrıntılı olarak ele alınmış, tarihsel gelişimi ile birlikte matematiksel formları ve geliştirilen modelleri teorik yapı çerçevesinde ortaya konulmuştur.

İkinci bölümde, dinamik karar verme ve yöneylem araştırmasının optimumlaştırma yöntemleri arasında yer alan dinamik programlama üzerinde durulmuştur. Dinamik programlamanın tarihsel gelişim süreci, uygulama alanları, temel kavramlarına değinilmiştir. Dinamik programlama modellerinin temel prensibi olan “optimallik ilkesi” açıklanarak dinamik programlama formülü ele alınmıştır. Son olarak dinamik programlama problemlerine yönelik çözüm yöntemleri ve deterministik ve stokastik dinamik programlama açıklanmıştır.

Tezin son bölümünde, çok amaçlı dinamik programlama literatür incelemesi, uygulama alanları ve çok amaçlı an kısa yol problemleri ile birlikte çok amaçlı dinamik programlama formülasyonuna yer verilmiştir. Bu bölümde son olarak, karar sürecinde üç amaca sahip firmanın ulaştırma sorunu için çok amaçlı dinamik programlama modeli ortaya konulmuş ve bu probleme çözüm sunulmuştur.

## **BİRİNCİ BÖLÜM**

### **ÇOK ÖLÇÜTLÜ KARAR VERME VE ÇOK AMAÇLI PROGRAMLAMA**

Karar problemlerinin çözümünde çoğunlukla optimal çözüm değeri elde edilmeye çalışılır. İşletmecilik problemlerinde kar maksimizasyonu optimal bir sonucu oluştururken, maliyet minimizasyonu da ayrı bir optimal sonuç oluşturmaktadır. Oysa gerçek hayattaki karar problemlerinde tek bir amaç bulunmaz. Pek çok işletme sorunu doğası gereği çok amaçlı ve doğrusal olmayan programlama problemleri olarak ortaya çıkmaktadır. Çünkü gerek kişiler gerekse kurumlar aynı anda birden çok amaca sahip olabilmektedir. Bu problemlere uygun optimum karar seçeneklerinin sağlanabilmesi için çok amaçlı karar verme teknikleri geliştirilmiştir.

#### **1.1. ÇOK ÖLÇÜTLÜ KARAR VERME**

Yönetim her zaman bir karar verme işlemidir. Şirket yaşamları, içinde barındırdığı bireylerin aldıkları kararların amaçlara uygunluğu veya uygunsuzluğu ile son derece yakından ilgilidir. Dolayısıyla, yönetim sorumluluğu almış bireyler, sonuçları az veya çok zararlı, amaca kısa veya uzun yoldan ulaşan, anlamlı veya anlamsız, şirketin parasını veya çalışanları gözetten, şirketin ömrünü uzatan veya kısaltan, vb. milyonlarca karar alma durumundadırlar (Emhan, 2007: 212). Yönetimlerde arzulanan sonuçlar veya amaçlar için en muhtemel yolu izleyecek çalışmaların seçimi ile seçenek ve durum değerlendirmesi yapan strateji veya prosedürlere karar alma denir (Köse, 2004: 10).

Karar verme, eldeki tüm bilgileri dikkate alınarak durumun kavranması, alternatif eylem biçimleri ile getirecekleri sonuçların gözden geçirilmesi ve uygun eylemin seçilerek uygulanma sürecidir. Artan rekabet koşulları ile beraber, işletmeler açısından karar verme uygulamaları çok daha büyük önem arz etmeye başlamıştır. Bilim ve teknolojiadaki gelişmeler sonucunda, artık karar verme problemlerinin

özmleri sezgisel deęil bilimsel yöntemlerle aranmaya başlamıştır (Kaya vd., 2007: 8).

Karar alma işleminde genellikle iki yaklaşımdan söz edilebilir. Birinci yaklaşım kantitatif yaklaşımdır. Kantitatif (yöneylem) yaklaşımda problem tanımlanarak analiz edilir, alternatif çözüm yolları belirlenerek belirli bir kritere göre bu çözüm yollarından birisi seçilir ve uygulanır. İkinci yaklaşımda ise, karar vericinin kişisel deneyim ve yargısı esas alınarak karar verilir. Bu durumda herhangi bir kantitatif analiz söz konusu değildir. Yöneylem araştırması, karar verme problemlerinin çözümünde yardımcı olan ve çözümünü kolaylaştıran çalışmaları geliştiren ve düzenleyen bir bilimdir. Yöneylem araştırmasının en önemli özelliklerinden birisi karar vermede kullanılan matematiksel tekniklere sahip olmasıdır. Yöneylem araştırması, bilimsel yönetim yaklaşımı esaslarına uygun olarak karar verme problemlerine rasyonel ilişkiler ışığında sistematik bir şekilde yaklaşır (Emhan, 2007: 215).

Ulaşılmak istenen hedefin birçok parametre tarafından belirlendięi ve deęerlendirilecek alternatiflerin her birinin kendine has avantajlarının bulunduęu durumlarda karar verme işi oldukça güçleşmektedir. Kararlar üzerine etki eden tüm kriterlerin deęerlendirilmesi ve analizi amacıyla çok kriterli karar verme yöntemleri (ÇKKVY) geliştirilmeye başlanmıştır (Kaya vd., 2007: 8).

#### **1.1.1. Karar Verme**

Karar verme işlemi, karar vericinin mevcut alternatiflerle karşı karşıya kaldığı durumda, bu alternatifler arasından kendisince belirlenmiş belli ölçütlere göre kendi amaç ve amaçlarına uygun, bir veya birkaçını seçme sürecidir (Çetin, 2002: 4). Eğer bir olaya ait tek bir olasılık ya da tek bir sonuç varsa o olaya ait bir karar verme işleminden söz edilemez (Turan, 2001:3).

Karar verme, Cosgrave (1996) tarafından bütün yöneticilerin en önemli görevlerinden biri olarak tanımlanmıştır. Karar verme davranışı, büyük oranda, yöneticilerin davranışlarını özetleyen ve yöneticileri toplumdaki dięer mesleklerden

ayırt eden bir eylemdir. Diğer taraftan bakılırsa yönetim faaliyeti tamamıyla karar verme faaliyeti olarak görülebilir. Hatta birçok yazar etkili kararlar verme eyleminin, yönetim faaliyetinin kalbinde yer aldığı konusunda hemfikirdirler. Mintzberg (1975), yönetimin on rolünden dördünü karar rolleri olarak tanımlamıştır. Karar verme yönetim faaliyetlerinden yalnızca bir tanesi değil aynı zamanda yönetim faaliyetinin bir parçasıdır (Tekin ve Ehtiyar, 2010: 3396).

Karar verme davranışının ortaya çıkabilmesi için şu üç koşulun bulunması gerektiği vurgulanmaktadır (Tekin, 2009:5):

- Karar verme gereksinimini ortaya çıkaran bir seçme sorununun varlığı ve bu sorunun birey tarafından hissedilmesi;
- Güçlüğü giderecek birden fazla seçeneğin bulunması;
- Bireyin seçeneklerden birine yönelme özgürlüğüne sahip olmasıdır. Bu koşullar sağlandığında bireyin bir karar vermesi beklenir.

Karar verme problemleri pek çok işletme ve karar verici için büyük önem arz etmektedir. Karar vericilerin, karar sırasında kararı etkileyecek olan tüm kriterleri göz önüne alması ve ona göre kararını vermesi gerekir (Kaya vd., 2007: 14).

Bir karar verme probleminin elemanları şu şekilde sıralanabilir (Türker, 1986:7):

1. Seçenekler kümesi (mümkün çözümler)
2. Ölçütler kümesi
3. Değerlendirme sonucu kümesi (ölçüt değerleri kümesi)
4. Karar verici tercihleri.

**1. Seçenekler kümesi (Mümkün seçenekler):** Seçenekler, bir sorunun çözümünde kullanılabilecek olan birbirinde farklı yaklaşımlardır. Seçenekler kümesi oluşturulurken mümkün olduğu kadar geniş tutulması gerekir. Seçenekler, karar vericinin kontrolü altındaki kaynaklara bağlıdır ve kontrol edilebilir değişkenlerdir (Türker, 1986:8).

**2. Ölçütler kümesi:** Ölçütler karar vericinin gerçekleşmesini istediği amaçlarla bir bütünlük gösterirler. Bir karar probleminin ölçütleri tanımlanmadan herhangi bir seçeneğin değerlendirilmesi yapılamamaktadır. Diğer yandan analizlerde ölçütler seçilirken aynı amacı birden çok ölçütle göstermekten kaçınmak gerekir. Ayrıca ölçütün sayısı da önemli olmaktadır. Ölçüt sayısı problemin önemli yanlarını görmezden gelecek kadar az sayıda olmamalı, analizi gereksiz yere zorlaştıracak kadar da çok sayıda olmamalıdır (Türker, 1986:8).

Ölçütler iki gruba ayrılırlar:

1. Nicel ölçütler: Sayısal değerlerle test edilebilen ölçütlerdir.
2. Nitel ölçütler: Bu gruptaki ölçütler sayısal değerlerle kolaylıkla test edilemez. Bazı ölçütler ise nicel ve nitel özellikleri birlikte taşırlar (Türker, 1986:8).

**3. Değerlendirme sonucu kümesi (Ölçüt değerleri kümesi):** Seçeneklerin ölçütlere göre değerlendirilmesi ile elde edilen değerler kümesidir. Ortaya çıkması beklenen olaylara göre 3 farklı gruba ayrılır. Ayrım karar verenin olaylar hakkındaki bilgi derecesini yansıtır (Türker, 1986:9):

1. Belirlilik altında çok ölçütlü karar verme: Seçeneklerin ölçütlere göre değerlendirilmesi sonucu elde edilen ölçüt değerleri kesin olarak bilinmektedir.
2. Risk altında çok ölçütlü karar verme: Her bir seçeneğin her bir ölçüte göre değerini tek bir değerle belirtmeye imkan yoktur. Çünkü riskli durumlar, her seçeneğin belli bir olasılık dağılımının bulunduğu durumlardır. Bu nedenle her bir seçeneğin birden fazla olası sonucu olabilmektedir. Her sonucun gerçekleşme olasılığı, belirli bir olasılık dağılımından elde edilebilmektedir.
3. Belirsizlik altında çok ölçütlü karar verme: Bu durumda her seçeneğin gerçekleşmesinin kesin olmadığı gibi olasılıkları da bilinmektedir.

**4. Karar verici tercihleri:** Çok ölçütlü karar verme problemlerinde amaç sayısı birden çoktur ve birbiriyle çelişmektedir. Bu ölçütlerin önemleri, karar verici açısından aynı derecede değildir. Karar verme durumunda olan kişi veya

kişiler tarafından her bir ölçüt için önem derecesi saptanmaktadır. Böylece ölçütler önem derecelerine göre sıraya sokulmaktadır. Burada; karar vericinin bilgileri, tercihleri, deneyimleri ve sezgisi önem kazanmaktadır (Türker, 1986:9).

İşletmelerde alınacak kararlar günlük yaşıntıda alınacak kararlardan çok daha zor ve karmaşıktır. Bunun başlıca nedenlerinden biri işletmelerde çok fazla seçenek arasından optimum seçimin yapılmasıdır. İşletmeler sahip oldukları karmaşık yapıyı ne kadar sadeleştirebilirlerse optimum seçime ulaşmak o derece kolay olacaktır (Köse, 2004: 11).

Karar alma aşamasında karar alıcının problemini çözmesine yardımcı olması için bir takım adımlar geliştirilmiştir. Bu adımların oluşturulmasındaki amaç, problemin optimum sonucuna en güvenilir ve en kısa zamanda belirli bir sistematik içerisinde ulaşmaktır. Karşılaşılan problemin çözümünü gerçekleştirebilmek veya iyi bir karar alabilmek için aşağıdaki adımlar izlenebilir (Köse, 2004: 18):

- Elimizdeki problemi kesin bir şekilde tanımlamak
- Olası alternatif çözümleri listelemek
- Olası sonuçları belirlemek
- Sonuçların ve alternatif çözümlerin bütün kombinasyonlarının kar ve maliyetini listelemek
- Matematiksel bir karar teori modelini seçmek
- Modeli uygulamak ve kararı almak

### **1.1.2. Yönetmel Karar Verme Süreci**

Genel anlamda karar verme, yöneticiyi belli bir hareket biçimini benimsemeye yönelten bir seçim süreci olarak tanımlanabilir. Buna göre karar verme değişik davranış biçimlerinden (eylem, alternatif, strateji vb.) birinin soruna çözüm getirmesi için seçilmesidir. Karar vermede, var olan olanak ve koşullara göre olağan olabilecek çeşitli hareket biçimlerinden en uygun olanını seçmek önemlidir. Yani karar vermede, optimum çözüm seçeneğinin aranması gereklidir. Ancak, gerçek

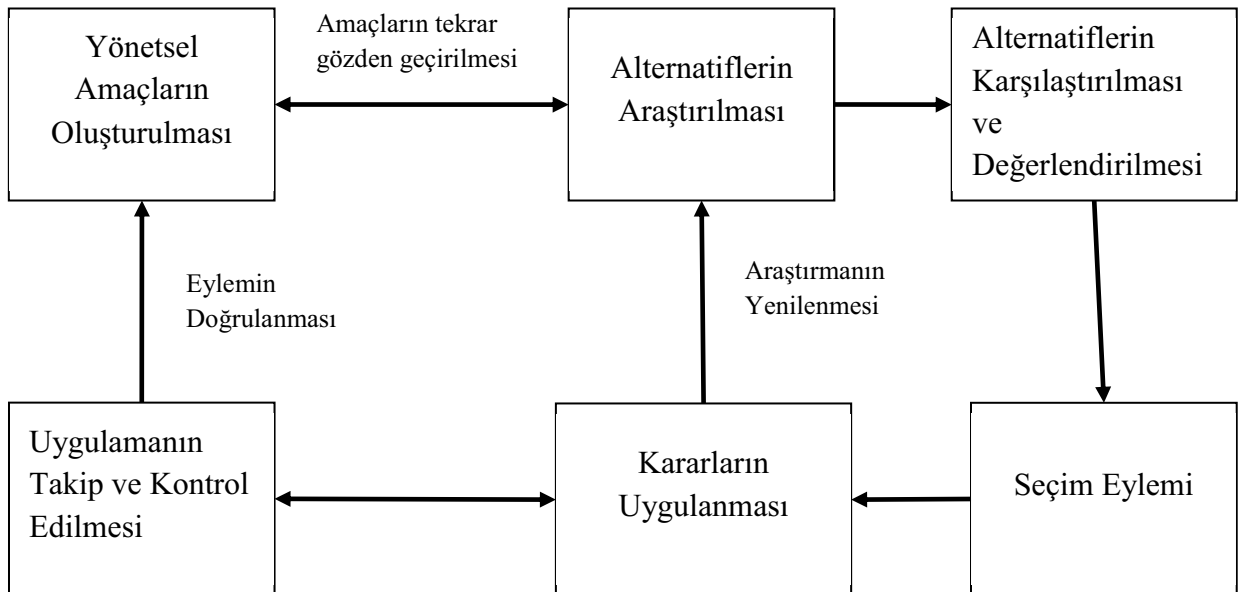
hayatta karar verme oldukça karmaşık yapıda olduğundan, muhtemel en rasyonel hareket biçimini seçmek gibi kısıtlı rasyonellik söz konusudur. Yani karar alanı olan gelecekle ilgili risk ve belirsizliği ortadan kaldırmak mümkün değildir, ancak azaltılabilir (Karakaya, 2004: 606).

Karar alma, bir amaca ulaşmak için eldeki olanak ve koşullara göre mümkün olabilecek çeşitli yöneylem hareket biçimlerinde en uygun görüneni seçmektir. Tanımdan da anlaşılacağı üzere, tüm karar problemlerinin bir amacı olmalıdır. Fakat en güç olanı ele alınacak özel amacın seçimidir (Köse, 2004: 11).

Tercih ve seçim biçiminde ortaya çıkan ve bir sonuç niteliğinde olan karar, kuskusuz düşünsel çaba ve çalışmanın ürünüdür. Bu düşünsel ürünün iyi olmasının önemi büyüktür. Çünkü kararın ilgili olduğu örgütün başarı oranı, kararın isabet derecesine bağlıdır. Karar vermenin anlam kazanması, konunun süreç olarak ele alınmasıyla ilgilidir (Koçoğlu, 2010: 79).

Harrison ve Pelletier (2000) yönetsel karar verme sürecini aşağıdaki Şekil 1’de görüldüğü gibi açıklamışlardır (Tekin ve Ehtiyar, 2010: 3398).

**Şekil 1: Yönetsel Karar Verme Süreci**



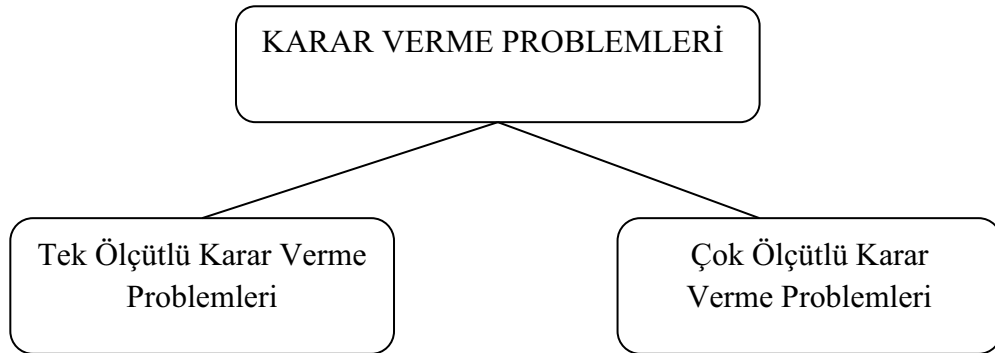
Kaynak: Tekin ve Ehtiyar, 2010: 3398.

Şekil 1’de de görüldüğü üzere yönetsel karar verme süreci aşağıdaki aşamalardan oluşmaktadır (Tekin ve Ehtiyar, 2010: 3398):

1. Yönetsel amaçların oluşturulması,
2. Amaçların tekrar gözden geçirilerek amaca ulaşabilmek için alternatiflerin araştırılması,
3. Alternatiflerin karşılaştırılması ve değerlendirilmesi,
4. Alternatiflerden birinin seçim eylemi,
5. Kararların uygulanması,
6. Uygulamanın takip edilmesi ve kontrol edilmesi.

Karar verme problemleri şekilde görüldüğü gibi iki gruba ayrılabilir (Türker, 1986:7):

**Şekil 2:** Karar Verme Problemlerinin Sınıflandırılması



Kaynak: Türker, 1986: 7.

1. Tek ölçütlü karar verme problemleri: karar vermenin en basit olanıdır. Karar verici tek bir amacı göz önüne almaktadır.
2. Çok ölçütlü karar verme problemleri: bu tür problemlerde genellikle amaçlar birbiriyle çelişkili olduğundan (yüksek oranda kar elde etmekle en çok sayıda işçi çalıştırmak gibi) her amacı en iyi şekilde karşılayan bir seçenek bulmak

zordur. Bu nedenle seçenekler arasından en uygununu seçmek için çeşitli yöntemler geliştirilmektedir (Türker, 1986:7).

### **1.1.3. Çok Ölçütlü Karar Verme / Çok Kriterli Karar Verme Süreci ve Aşamaları**

Çok kriterli karar verme, karar bilimlerinin bir alt dalı olup, karar sürecini kriterlere göre modelleme ve analiz etme sürecine dayanır. Karar vericinin sayılabilir sonlu ya da sayılamaz sayıda seçenekten oluşan bir küme içinde en az iki kritere dayalı değerlendirme yaparak seçim yapmasını sağlar. Bu nedenle de çok kriterli karar verme yöntemleri, karar vericiye topladığı bilgileri iyi bir şekilde analiz ederek amaç ya da amaçlarını gerçekleştirecek seçenekleri çeşitli kriterlere göre değerlendirmesine ve karar vericinin isteklerini de karşılayacak en uygun seçeneği belirlemesine yardımcı olur. Burada önemli olan karar sorunu ile ilgili bilgileri ÇKKV yöntemlerini uygulayacak şekilde düzenlemek yani uygun bir şekilde modeli kurmaktır. Ayrıca karar vericinin karar probleminin yapısına ve özelliklerine göre hangi ÇKKV yöntemini kullanacağına da modeli kurmadan önce karar vermesi gerekmektedir (Gülenç ve Bilgin, 2010: 98).

Çok ölçütlü karar verme, genellikle çelişen çok sayıda ölçütün bulunduğu karar verme durumlarını konu alır. “Çok ölçütlü” terimi, çoklu nitelikler veya çoklu amaçlar içeren kararlar için, çoğu kez de her ikisini birde kapsayan kararlarla ilgili olarak kullanılır (Günyaşar, 1990: 5).

Bir çok kriterli karar verme (Multiple Criteria Decision Making -MCDM) problemi, herhangi bir seçenekler kümesinden, bu seçenekleri etkileyen ve genelde birbiriyle çatışan kriterlerin performanslarına dayanarak, en iyi seçeneğin seçilmesi işlemidir. Çok amaçlı karar vermede ise (Multiple Objectives Decision Making-MODM), birbiriyle çatışan amaçlar altında en iyi alternatifin seçimi söz konusudur (Albayrak, 2004: 48).

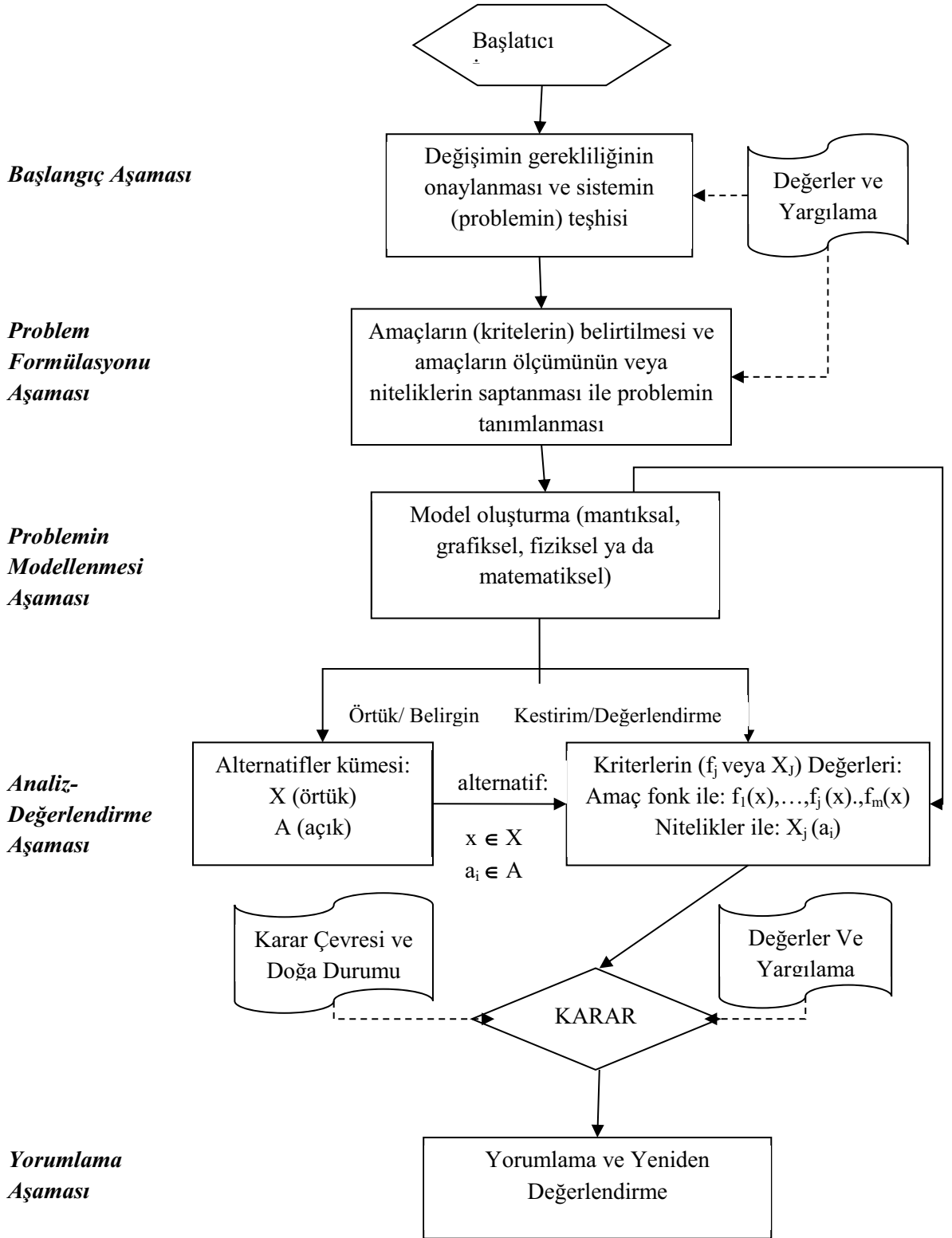
"İyi" ya da "rasyonel" karar verme sadece insana has bir özelliktir. Dolayısıyla, insan, karşısına çıkan ve giderek daha karmaşık bir hal alan karar

problemlerinde iyi kararlar verebilmek için sürekli olarak yollar ve araçlar geliştirmektedir. İşte Çok Kriterli Karar Verme süreçleri, karmaşık karar problemlerini bilimsel ve analitik bir çerçevede ele alarak karar vericiye en çok istediği çözüme ulaşmasında yardımcı olmaya çalışan prosedürler bütünü olarak ortaya çıkmıştır (Çınar, 2004: 17).

Çok Kriterli Karar Verme (Multiple Criteria Decision Making-MCDM), en kısa tanımıyla; "Çoklu ve birbiriyle çatışan amaçların (kriterlerin) gerçekleştirilmek istendiği problemlerin çözümüne verilen genel isimdir (Çınar, 2004: 17).

Çok Kriterli Karar Verme Süreci terimi, problem çözmenin aşağıdaki şekilde gösterilen ve beş aşamadan oluşan bütüncül yapısını tanımlar (Çınar, 2004 :26):

**Şekil 3:** Tipik Çok Amaçlı (Kriterli) Karar Verme Süreci Aşamaları



Kaynak: Çınar, 2004: 26.

Şekil 3’de görülen,

Başlangıç aşaması: Süreç, karar verici ilgilendiği sistemin (problemin akışını değiştirme ihtiyacını algıladığı anda başlar. Durum teşhis edilir ve nihai amaç ifadesi ortaya konulur (Çınar, 2004: 27).

Problemi Formüle Etme aşaması: Bu aşamada yapılması gereken farklı görevler/işler vardır. Bunlar (Çınar, 2004: 27);

- Soyut bir şekilde ortaya konmuş üst amacın daha işlevsel ve spesifik amaçlar halinde ifade edilerek alt amaçlar (kriterler) kümesinin oluşturulması,
- Sistemin tüm gerekli elemanlarının, problemin sınırlarının ve sistemin çevresel koşullarının açıkça ortaya konulmasıdır.

Model Oluşturma Aşaması: Sistemin çevresi ve amaçlar kümesi bir kere iyice tanımlandığında, probleme uygun modeller oluşturulabilir (Çınar, 2004: 27).

"Model", birlikte etkin ve anlamlı bir biçimde sistemin ilgili yönlerinin kapsamlı bir analizini sağlayacak anahtar değişkenlerin ve bunların mantıksal (veya fiziksel) ilişkilerinin toparlaması ile oluşan yapıdır. Modellerin çeşitli biçimleri vardır; basit mantıksal modeller, grafik modeller, karmaşık fiziksel modeller, matematiksel modeller vb. Modeller, alternatifler başlangıçta veri değilse, probleme uygun alternatif hareket tarzları üretmek işlevini yerine getirebilirler.

ÇKKV problemlerinde, alternatiflerin karşılaştırılması gerektiğinden, nitelikler (amaçlar/kriterler için bir ölçüm seti) açıkça belirlenmeli ve ortaya konulmalıdır. Bu ölçüm seti, “performans ölçümleri/kriterleri/endeksi” veya “amaç (kriter) fonksiyonu” gibi farklı adlar alabilir. Veri bir alternatif için niteliklerin ölçüm seviyeleri, uygun bir ölçekte/skalada belirlenir. Burada söz konusu ölçekler birer karşılaştırma veya ölçüm standardı olarak işlev görürken, belirlenen ölçüm seviyeleri bir önceki aşamada ifade edilmiş olan amaçlara ulaşma derecesi olarak atanırlar. Belli bir alternatif için ilgili niteliklerin ölçüm değerleri ya modelden çıkarılır

(kestirilir) ya da öznel yargıların değerlendirilmesi yoluyla doğrudan belirlenir (Çınar, 2004: 27).

**Analiz ve Değerlendirme Aşamaları:** Analiz etme ve değerlendirme aşamasında, her alternatif diğerlerine göre, önceden tanımlanmış ve alternatifleri derecelendirmek için kullanılan bir karar kuralı veya kurallar setine bağlı olarak değerlendirilir. Karar kuralına göre en üst dereceyi alan alternatif yorumlanmak üzere seçilir. Süreç bir açık döngüsel süreç ise, işlem adımları bu noktada son bulur. Eğer elde edilen sonuç karar vericiyi tatmin etmiyorsa, bir diğer deyişle yetersiz bulunuyorsa, gözlemlenen çıktı için elde edilen bilgi kullanılarak problem formülasyon aşamasına (ikinci adım) geri dönülür. Bu yapıdaki bir süreç ise kapalı döngüsel süreç olarak tanımlanır (Çınar, 2004: 27).

#### **1.1.4. Çok Ölçütlü Karar Verme Problemlerinin Sınıflandırılması: Çok Nitelikli ve Çok Amaçlı Karar Verme**

Çok ölçütlü karar verme problemleri, iki ana grupta toplanabilir: Çok nitelikli karar verme (ÇNKV) ve çok amaçlı karar verme (ÇAKV). ÇNKV seçim (değerlendirme), ÇAKV ise tasarım gerektiren problemlerle ilgilidir. Bu, genel anlamda kabul görmüş bir sınıflandırmadır (Günyaşar, 1990: 6). 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 (Umarusman, 2007:6).

ÇNKV' nin özellikleri belirlenirken, genellikle önceden belirlenen alternatifleri sınıflandırılmış (sayılabilir küçüklükte) bir sayı bulunmaktadır. Alternatifler, en son verilmesi gereken kararda esas alınan özelliklerin başarı seviyesine göre birleştirilir. Bir alternatifin seçiminde son karar, niteliklerin kendi içinde ve birbirleriyle kıyaslanmasından yararlanılarak verilir (Oflluoğlu vd., 2006: 115).

ÇAKV problemlerine ait modellerin aşaması bazı ölçülebilir amaçların kabul edilebilir seviyelerine ulaşmak yoluyla, karar vericiyi en çok tatmin eden model

içindeki çeşitli etkiler ele alınarak “en iyi” alternatifin bulunmasıdır (Ofloğlu vd., 2006: 115).

Çok nitelikli karar vermenin ayırıcı yönü, genellikle sınırlı sayıda önceden belirlenmiş seçeneğin varlığıdır. Her seçenek için ölçütler cinsinden başarımları düzeyleri belirlenmiş olup, son karar bu değerlere bağlı olarak verilecektir. Seçeneklere ilişkin son seçim, nitelik içi ve nitelikler arası karşılaştırmalar yardımıyla yapılacaktır (Günyaşar, 1990: 6).

Çok nitelikli ve çok amaçlı karar verme problemlerinin karşılaştırmalı yapısal özellikleri Tablo 1’ de özetlenmiştir:

**Tablo 1:** Çok Nitelikli Karar Verme ve Çok Amaçlı Karar Verme Problemlerinin Karşılaştırılması

|                           | ÇNKV                                         | ÇAKV                                                  |
|---------------------------|----------------------------------------------|-------------------------------------------------------|
| Ölçütlerin tanımlanması   | Nitelikler tarafından                        | Amaçlar tarafından                                    |
| Amaçların tanımlanması    | Örtülü tanımlanmış                           | Açık tanımlanmış                                      |
| Niteliklerin tanımlanması | Açık olarak                                  | Örtülü olarak                                         |
| Kısıtlılıklar             | Aktif değil (nitelikler içerisinde kapsanır) | Aktif                                                 |
| Seçenekler                | Sonlu sayıda, ayrık (Önceden belirlenmiş)    | Sonsuz sayıda, sürekli (Süreç ilerledikçe belirlenir) |
| Karar vericiyle etkileşim | Fazla değil                                  | Çoğu kez                                              |
| Kullanım amacı            | Seçim/ Değerlendirme                         | Tasarım                                               |

Kaynak: Dünyaşar, 1990: 7.

#### 1.1.4.1. Çok Amaçlı Karar Verme

Bir kararı gerektiren durumlarda, bazı amaçlar arasındaki doğal çelişme ya da kaynakların kıt olması nedeniyle aynı anda gerçekleştirilemeyen çoklu ve çelişen amaçlar söz konusudur. Burada hangi eylem tarzının seçileceğini saptamak için, karar vericinin amaçların göreceli önem düzeylerine ilişkin değer yargılarını belirtmesi gerekmektedir. Karar verici tercihlerinin temel göstergesi, bir amacı iyileştirmek için diğer amaçlardan verilebilecek ödün miktarlarını gösteren takas değerleridir. Bu değerler sabit ve kolayca erişilebilir değil, o andaki koşulların duruma bağlı, dinamik ve sürekli değişen bir göstergesidir (Günyaşar, 1990:4).

Çok amaçlı yönetim, yönetim biliminin en önemli konularından biri olarak bilinir. İşletmelerdeki büyüme, büyük ve karmaşık karar problemlerini ortaya çıkarmaktadır. Stratejik karar verme problemlerinde çok sayıda seçenekler, kısıtlayıcılar ve amaçlar söz konusu olmaktadır (Keçek, 2005: 111).

Karar verme, mevcut tüm alternatifler arasından amaç veya amaçlara en uygun ve mümkün olan bir veya bir kaçını seçme sürecidir. Özellikle son yıllarda yapılan bilimsel çalışmalar, karmaşık ve belirsiz bir karar ortamında karar verme zorunluluğu ile karşılaşan karar vericinin, bilgisini ve tecrübesini sistematik bir şekilde değerlendirerek, en iyi bir çözüme nasıl erişilebileceğine ilişkin yaklaşımlar getirmiştir. Bu nedenle de karar vericinin yapması gereken, amaç ya da amaçlarını gerçekleştirecek çeşitli alternatifler arasından belirlediği kriterlere göre bir değerlendirme yapmak ve birini seçmektir. Burada karar verici bir kişi veya amaçları aynı olan kişilerin oluşturduğu bir topluluk ya da amaçları farklı kişi ve toplulukların oluşturduğu bir grup olabilir (Gülenç ve Bilgin, 2010: 98).

Kantitatif karar verme tekniklerinden biri olan çok amaçlı karar verme modeli işletmenin birbiriyle çatışan birden çok amaç ile karar vermesi gerektiğinde dikkate alınan matematiksel bir modeldir (Özdemir ve Gümüsoğlu, 2006: 359).

Zeleny “çok amaçlı karar verme” için şunları söylemektedir: Çevrenizdeki dünyayı tek boyutlu görmek ve görülen her şeyi tek kritere göre değerlendirmek

giderek zorlaşmaktadır. Her zaman seçim kriterlerine göre amaçları karşılaştırır, derecelendirir ve sıralarız. Fakat yalnızca basit, açık ve alışılmış durumlarda tek bir seçim kriteri tamamıyla geçerli olabilir (Alp, 1990: 8).

Çok amaçlı karar alma, birden fazla amaç söz konusu olduğunda veya bir amacın çözümü için birden fazla seçenekle karşılaşıldığında ortaya çıkar (Köse, 2004: 41).

İşletmeler, klasik ekonomik amaçları olan “kar maksimizasyonu” nun yanı sıra, artık ekonomik olmayan fakat hayati derecede önemli birçok amacı bir arada gerçekleştirmek durumundadırlar. Bu amaçlar arasında ilk akla gelenler olarak; işletme organizasyonu içinde çalışan personelin, işletmenin hisse senedi sahiplerinin ve müşterilerinin tatmini, topluma karşı sorumluluk, ürün kalitesinin, tedarikçi ve dağıtıcılarla ilişkilerin iyileştirilmesi, kamusal ve hukuksal düzenlemelere itaat etme, işletme itibarının artırılması sayılabilir (Çınar, 2004: 20).

Çok amaçlı tekniklerin kullanımı, yatırım kaynaklarının sınırlı olduğu, etkin bir yatırım kararının çelişen sosyo politik koşullarla tanımlı amaçları sağlaması gerektiğinde özellikle önem taşır. Bu teknikler, literatürde çok amaçlı analiz, çok amaçlı optimizasyon, çok amaçlı karar verme ve vektör optimizasyonu olarak bilinir. Çok amaçlı karar verme (ÇAKV) pratik uygulamaları kadar teorik gelişmeleri ile de karar analizinin en hızlı gelişme gösteren alanlarından biridir. ÇAKV problemleri, yeni ürün geliştirme, fiyatlandırma kararları, araştırma projesi seçimi, işgücü planlaması gibi özel girişim işlerine ek olarak ulusal savunma giderlerini planlama, ulusal enerji planı geliştiren bir ülkenin politikasını belirleme gibi genel kararları da kapsar (Keçek, 2005: 111).

Çok amaçlı karar verme modellerinde hedef; tasarım kısıtları içerisindeki çeşitli etkileşimleri dikkate alıp, ölçülebilen bazı amaçlarda makul düzeylere erişimi sağlayarak karar vericiyi en iyi şekilde doyuma ulaştıran “en iyi” seçeneği tasarlamaktır. ÇAKV yöntemlerinde, ortak olarak şu bileşenler bulunur (Günyaşar, 1990: 6):

- Ölçülebilen amaçlar kümesi
- İyi tanımlanmış kısıtlar kümesi
- Açıklanmış ölçülebilir amaçlar arasındaki, ayrıca açıklanmış ya da açıklanmamış ölçülemeyen amaçlar arasındaki takaslama bilgilerinin örtülü veya açıkça elde edilmesini sağlayan bir süreç.

#### **1.1.4.2. Çok Amaçlı Karar Vermede Temel Kavramlar**

Çok amaçlı karar vermede karar verme işlemini yapan bir karar verici ya da karar vericiler vardır. Bunun yanında bir takım kavramlarda mevcuttur. Bu kavramlar, hedefler, kriterler, amaçlar, nitelikler, kısıtlar olarak tanımlanmaktadır. Sözlük tanımlaması olarak, hedef, kriter ve amaç kavramları genellikle aynı anlamı içermektedir. Bunların anlamı karar verme koşuluna göre ayırt edilebilmektedir (Çetin, 2002: 7).

Çok amaçlı karar verme modellerinin içerdiği kavramlardan başlıcaları aşağıda sıralanmaktadır:

**Amaç:** Amaç kavramı birçok yazar tarafından farklı şekillerde tanımlanmıştır. Spesifik bir tanım Zeleny tarafından “karar vericinin istekleri doğrultusunda maksimize ya da minimize edilmek istenen özellikler” olarak yapılmıştır. Amaç bir şeyi tamamlamak için onun peşine düşmektir. Bir işletmenin karını maksimize etmek, hizmetin kalitesini maksimize etmek ya da müşteri şikayetlerini minimize etmek istemesi gibi (Çetin, 2002: 8). Niteliklerin belirlenmesinin ardından, nitelikler ölçülür ve tanımlanır. Karar verici niteliklerin hangi seviyede maksimizasyonunu veya minimizasyonunu yapacağına kendi istek ve ihtiyaçları doğrultusunda kararlaştırır (Umarusman, 2007: 7).

Karar alma sürecinde amaçların bilinmesi sorunun çözümünde oldukça önemlidir. Çünkü yaptıkları uygulamanın hangi hususları ya da amaçları gerçekleştireceğinin farkında olacaklar ve faaliyetlerini kendileri denetleyip kontrol altına alabileceklerdir (Köse, 2004: 44).

Hedef: Evren ve Ülengil (1992) hedef kavramını “amaçların daha da somutlaşarak belli değerlere dönüşmüş şekilleri” olarak tanımlamaktadır. Bazen erişilebilen bazen erişilemeyen şeydir. Örneğin bir ürünün 1 yıl boyunca satışının önceki yıla göre %10 arttırmayı düşünmek bir hedeftir. Eğer bu hedefe ulaşmak zor ya da mümkün değilse amaçların içinde yer alabilir (Çetin, 2002: 8). Hedefler, tam olarak karar vericinin ihtiyaçları ve istekleri olarak tanımlanabilir. Amaçların veya özelliklerin herhangi birisinin özel değerinin veya seviyesinin önceliğini belirtir. Hedefler, basarının arzu edilen seviyesi olarak da ifade edilebilir (Umarusman, 2007: 7).

Kriter: Hwang ve Masud (1979) kriter kavramını kabul edilebilirliği test etmek için kararlar ve usullerin standartları şeklinde tanımlamışlardır (Çetin, 2002: 9). Değerlendirme yapmak için etkinliğin ve esasın ölçüsüdür. Kriter, problemin yapısı içerisinde özellikler veya amaçların bir formu olarak ortaya çıkar. Kriterler, karar vermeye rehberlik eden ölçümler, kurallar ve standartlar olarak da tanımlanır (Umarusman, 2007: 7).

Nitelik (karar bileşeni): karar vericinin istek ve ihtiyaçlarından nisbi olarak bağımsız bir şekilde tanımlanmış ve belirli bir kararın ne ölçüde gerçekleştirilebildiğinin değerlendirilmesine yarayan ölçüdür (Çetin, 2002: 9). Performans parametreleri, eleman, etken ve özellikler nitelik ile aynı anlamı taşımaktadır. Nitelik, bir amacın seviyesinin değer olarak anlamını belirtmektedir (Umarusman, 2007: 7).

Kısıtlar: sistemin amaçlarını etkileyen varsayımlar, mevcut kaynaklar ve problemde yer alan değişkenler üzerindeki sınırlardır (Köse, 2004: 44).

## **1.2. ÇOK AMAÇLI PROGRAMLAMA**

Bir çok gerçek dünya araştırmaları ve optimizasyon sorunları doğal haliyle çok amaçlı doğrusal olmayan programlama problemleri olarak ortaya çıkmaktadır. Uygun çözüm tekniklerinin olmayışı sebebiyle bu tür sorunlar yapay olarak tek hedefli problemlere dönüştürülmüş ve çözülmüştür. Bu tür problemlerin tek bir

optimal çözüm yerine bir dizi optimal çözümler sunması nedeniyle bir zorluk ortaya çıkmıştır (Deb, 2005: 273).

Çok amaçlı bir optimizasyon problemi birden fazla amaç fonksiyonu ile ilgilenir. Birçok pratik karar verme probleminde çoklu amaçlar ve çoklu kriterler belirgindir. Uygun çözüm metodolojilerinin eksikliği nedeniyle bir çok amaçlı optimizasyon problemi geçmişte çoğunlukla tek amaçlı optimizasyon problemi olarak düzenlendi ve çözüldü. Fakat çok amaçlı bir optimizasyon probleminin bir çok amaçlı optimizasyon tekniği kullanılarak çözümüne dayanması gerektiği için tek amaçlı ve çok amaçlı optimizasyon algoritmalarının çalışma prensipleri arasında bir dizi temel farklılıklar vardır. Bir tek amaçlı optimizasyon probleminde görev tek amaç fonksiyonunu optimize eden tek bir sonuç (birden fazla optimal çözümler aranan bazı özel optimizasyon problemleri hariç) bulmaktır. Çok amaçlı optimizasyon fikrini biraz genişletmek gerekirse bir çok amaçlı optimizasyon probleminde her bir amaç fonksiyonuna karşılık gelen optimal çözümü bulmak için bu tek amaç fonksiyonunu optimize etme görevini göz önünde bulundurmak yanlış olur. (Deb, 2005: 274).

Çok amaçlı bir optimizasyon problemi optimizasyonu sağlayacak çeşitli hedefler içerdiğinden tek amaçlı bir optimizasyon probleminden farklıdır. Tek amaçlı bir problemi optimize ederken, hedef en iyi çözüm planıdır. Fakat çok amaçlı bir optimizasyon problemi için var olan (muhtemelen çelişkili) bir çok hedefleri ile genellikle tek bir optimal çözümü yoktur. Bu nedenle karar vericinin uzlaşma yoluyla sonlu çözüm kümesinden bir çözüm seçmesi gereklidir. Uygun bir çözüm tüm hedefler üzerinde kabul edilebilir bir sonuç sağlamalıdır (Coello ve Lamont, 2004: 2).

### **1.2.1. Çok Amaçlı Programlamanın Gelişimi Ve Uygulama Alanları**

Çok amaçlı problemlerin analizindeki gelişim öncelikle üç disiplinde oluşmuştur: Yöneylem araştırması, ekonomi ve psikoloji. Yöneylem araştırması içinde Kuhn ve Tucker (1951)' in teorik çalışması, matematiksel programlamanın

daha sonraki algoritmik gelişmeleri için temel oluşturmuştur. Gass ve Satty (1955) çok amaçlı problemlere uygulanabilir ilk yaklaşımı sunmuşlardır. Yöneylem araştırmalarında çok amaçlı programlamaya ilgi devam etmiş ve artmıştır. Çok amaçlı programlamaya yönelik ilk konferans 1972’de South Caroline Üniversitesi’nde düzenlenmiştir. Bu konferansın bildiri kitabı Cochrane ve Zeleny (1973) tarafından yayınlanmıştır. Daha sonra Kyoto, Japonya’daki Yönetim Bilimleri Enstitüsü’nün uluslararası toplantısındaki 1975 yılı çok amaçlı oturumlarının bildiri kitabı yine Zeleny (1976) tarafından yayınlanmıştır. Ayrıca Ijiri (1965), Lee (1972) ve Ignizio (1976) hedef programlama kitabı hazırlamışlardır; Zeleny (1974) çok amaçlı simpleks metodu kitabı ve Haimes vd. ise (1975) yedek değer ikame yöntemine yönelik bir kitap sunmuşlardır. Starr ve Zeleny (1977) *Management Science* ’nin özel bir sayısını çok amaçlı programlama üzerine yayınlamışlardır (Cohon, 1978: 9).

Yayın patlaması seksenli yıllarda artarak devam etmiştir. ÇAKV konusunda düzenli yayın yapan dergi sayısı 10’u geçmiştir: Computer&Operation Research European Journal of Operational Research Society, Journal of Optimazation Theory and Applications, Omega Naval Research Quarterly, Mathematical Programming, Journal of Mathematical Analysis and Applications, Management Science, Operation Research ve Water Resourses Research (Alp, 1990: 6).

Son 30-40 yılda Çok amaçlı programlama; yönetim bilimi, yöneylem araştırması, uygulamalı matematik ve mühendislik alanında ki araştırmalarda artarak kullanılan bir yöntem durumuna gelmiştir. Çok amaçlı programlama, özellikle yöneylem araştırmasında hızla gelişen alanlardan birisidir (Atlas, 2008: 48).

Günümüzde yöneylem araştırması adreslerinde(kitap, makale, internet, vb.) matematiksel programlama modellerinin büyük çoğunluğu, tek amaçlı olarak kullanılmaktadır. Örneğin, işletmelerde kapasite artırma maliyetlerinin en küçüklenmesi gibi. Aynı problem için ikinci bir amaç, dış pazara açılma veya çalışanları mutlu kılmak olabilir. Böyle bir problem için tek amaçtan daha çoğuna

gereksinim vardır. Dolayısıyla çoğu problemde, olası seçeneklerden yalnız birini seçme zorunluluğu her zaman gerçekçi değildir (Atlas, 2008: 48).

Çok amaçlı matematiksel programlama, bir matematiksel programlama yapısı içinde çoklu amaçların aynı zaman da gerçekleşmesinin düşünüldüğü bir yoldur. Bu önemli alandaki çalışmaların çoğu 1970'den sonra görülmeye başlamıştır (Atlas, 2008: 48).

Çok amaçlı karar durumunda çoğu zaman genel kabul edilebilir bir çözüm yoktur. Bir sorunun herhangi bir çözümü karar vericinin tercihini yansıtır. En iyi çözümün tatmininin her karar verici için aynı olmadığı kabul edilir (Rakshit, 1988: 13).

Pek çok karar durumunda açıkça göz önünde bulundurulmuş çoklu amaçların faydası daha önceleri tanımlanmıştır ve bu tanımlamaların bu alandaki çalışmaları çok fazladır. Karar durumlarını sınıflandırmak için Stuart (1986), Szidarovszky vd. (1986), Yu (1985), Zeleny (1982), Goicoechea vd. (1982) tarafından pek çok yaklaşım sunulmuş, geliştirilmiş ve uygulanmıştır (Rakshit, 1988: 14).

Çok amaçlı karar verme metodolojisinin literatürde uygulamaları çok çeşitlidir. Bu uygulamalara örnekler; su kaynaklarının planlanması (Major, 1977), nükleer reaktör yeri seçimi (Cohon ve Eagles, 1982), eşit istihdam fırsatlarının planlanması (Charnes vd., 1985), askeri personel atama problemleri (Klingman ve Phillips, 1984), bitki yönetimi (Duckstein ve Gershon, 1983), şehir ve bölge planlanması (Voogd, 1983), araç flosu planlama (Couillard ve D'Avignon, 1988), enerji planlanması (Climaco vd., 1988) olarak verilebilir (Rakshit, 1988: 14).

Son yıllarda Yöneylem Araştırması alanında çok amaçlı programlamaya ilişkin uygulamalardan aşağıda bahsedilmiştir.

- Belirsizlik altında tedarikçi seçimine yardımcı olmak için çok amaçlı stokastik bir programlama modelinin kurulması (Bilsel ve Ravindran, 2011: 1284).

- Belirsiz bilgiler ile üretim planlaması için çok amaçlı programlama ve iç nokta Ellipsoid algoritmalarını temel alan bir yöntem geliştirilmesi (Trafalis, 2009: 631).
- Uluslararası hava alanı kod paylaşım ittifakını (Kod paylaşımı, ortak olan havaalanlarının işbirliğinde oldukları ortaklarıyla karşılıklı olarak genel karlılığı, piyasa için trafik kazanç ve kalite avantajlarını göz önüne alırken ayrı ayrı kendi karlarını maksimize etmelerini sağlar.) belirlemek için hava alanı ağ dizaynında etkileşimli çok amaçlı programlama prosedürünün geliştirilmesi (Wen, 2006: 404).
- Matematiksel programlama ve evrimsel algoritmalar kullanılarak çok amaçlı çok kısıtlı sırt çantası probleminin çözümlenmesi (Florios, 2010: 14).
- Çalışanların memnuniyet düzeylerinin belirlenmesi için çok amaçlı programlama modelinin kurulması (Gutierrez vd., 2010: 430).
- Televizyon reklam planlaması için bulanık veri tabanından veriler ile çok amaçlı programlama modeli kurulması (Gladish vd., 2010: 84).
- Dağıtım ağı bağlantıları boyunca arıtma tesislerinin kurulması ve katı atık taşınması için çok amaçlı programlama modelinin geliştirilmesi (Giannikos, 1998: 333).
- İş ortağı seçimi için bir çok amaçlı programlama modeli geliştirilmesi (Huang, 2010: 3530).
- Enerji üretim planlaması için çok amaçlı evrimsel programlama algoritmasının kullanılması (Meza, 2009: 1086).
- Çok ürünli parsel dağıtım zaman çizelgelemesi için çok amaçlı programlama modeli geliştirilmesi ( Ayed, 2012: 613).
- Portföy optimizasyonu için çok amaçlı programlama modeli ve çok nitelikli fayda fonksiyonu geliştirilmesi ( Ehrgott vd., 2009: 31).
- Çok amaçlı matematiksel programlama kullanarak özkaynak portföyü oluşturma ve seçimi (Xidonas, 2010: 185).
- Hastane yatak kapasite planlaması için çok amaçlı stokastik programlama modeli geliştirilmesi ( Abdelaziz, 2012: 530).

### 1.2.2. Çok Amaçlı Programlama Problemlerinin Genel Formülasyonu ve Baskınlık

Çok amaçlı programlama iki veya daha fazla amaç fonksiyonuna sahip optimizasyon problemleriyle ilgilenir. Çok amaçlı programlama problemleri tek amaçlı programlama problemlerinden yalnızca ayrı ayrı amaçlarının bulunması konusunda ayrılır. Genel olarak tek amaçlı optimizasyon problemleri  $n$  karar değişkeni ve  $m$  kısıtları ile şöyledir (Cohon, 1978: 68):

$$\text{Maksimum } Z(x_1, x_2, \dots, x_n)$$

$$g_i(x_1, x_2, \dots, x_n) \leq 0, \quad i=1, 2, \dots, m$$

$$x_j \geq 0, \quad i=1, 2, \dots, n$$

Çok amaçlı programlama problemlerini genel formülü ise  $n$  karar değişkeni,  $m$  kısıtları ve  $p$  amaçları ile şöyledir (Cohon, 1978: 69):

$$\text{Maksimum } Z(x_1, x_2, \dots, x_n)$$

$$= [Z_1(x_1, x_2, \dots, x_n),$$

$$Z_2(x_1, x_2, \dots, x_n),$$

$$\dots, Z_p(x_1, x_2, \dots, x_n)]$$

$$g_i(x_1, x_2, \dots, x_n) \leq 0, \quad i=1, 2, \dots, m$$

$$x_j \geq 0, \quad i=1, 2, \dots, n$$

Burada  $Z(x_1, x_2, \dots, x_n)$  çok amaçlı fonksiyonu ve  $Z_1(x_1, x_2, \dots, x_n)$ ,  $Z_2(x_1, x_2, \dots, x_n)$ ,  $\dots$ ,  $Z_p(x_1, x_2, \dots, x_n)$   $p$  tek tek amaç fonksiyonları bulunmaktadır. Amaç fonksiyonları tek tek amaç fonksiyonları olarak bu şekilde sıralanır, bu amaçlar toplanamaz ya da çarpılamaz (Cohon, 1978: 69).

Burada  $m$  adet amacı içeren bir vektörün maksimizasyonu söz konusudur. Tüm amaç fonksiyonlarını birlikte enbüyükleyen çözüme ulaşmak genellikle

mümkün olamamaktadır. Bu durumda her amaç için optimum çözümler, karar vericinin tercihleri doğrultusunda uzlaştırılır ve elde edilen çözüme “en iyi uzlaşık çözüm”(the best compromise solution) adı verilir. Çok amaçlı karar vermede sık karşılaşılan bir diğer çözüm ise baskın (nondominated) çözümdür. Baskın çözüm, Vektör Minimizasyon Probleminin (VMP) diğer amaç fonksiyonları arasında en az birinde gerileme söz konusu olmaksızın, diğer bir amaç fonksiyonunda gelişme sağlanamayan çözümdür (Keçek, 2005: 112).

Çok amaçlı programlamada, genellikle birbiriyle çelişen amaçlar söz konusudur ve p - amaçtan en az ikisi doğası gereği çelişir. Bu nedenle bütün amaçların en iyi şekilde karşılandığı tek bir optimum çözümün bulunması çok nadir bir durumdur. Bu gibi durumlarda karar verici en iyi çözüm yerine etkin çözümlere yönelir. Etkin çözümlerin her biri en az bir amaç için diğer aday çözümlerden daha iyi sonuç vermektedir. Başka bir ifade ile etkin çözüm; problemin optimum çözümü olmadığında, her bir amaca mümkün olduğunca yaklaşan ve amaçlardan en az birini sağlayan çözümdür. Problemin uygun çözüm kümesinde birden çok etkin çözüm olabilir. Bu etkin çözümlerden birisi, karar verici tarafından tercih edilir. Bu tercihin yapılabilmesi için en az iki etkin çözümün karar vericinin önüne konabilmesi gerekir. (Atlas, 2008: 50).

Çok Amaçlı Karar Verme metodlarının çözümünde tek tip çözüm sonucu elde edilemez. Bu sebeple belirlenen çözümlerin yapısına göre farklı isimler kullanılır. Bu farklı çözümler aşağıdaki başlıklarda ifade edilmiştir (Umarusman, 2007:9):

Uygun Çözüm Alanı: Uygun çözüm alanı X, karar değişkenlerinin bir kümesidir. Bu değişkenlere göre kısıtlar tatmin edilir.

Optimal Çözüm: Bir vektör maksimum problemi için optimal çözüm, her bir amaç fonksiyonunun maksimum değerinin aynı değişkenlere göre belirlenmesidir. ÇAKV problemlerinde amaçların birbirleriyle zıt olması sebebiyle optimal çözüme genellikle ulaşılmaz. Bir optimal çözüm için aşağıdaki şart sağlanmalıdır.

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

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

$$Z_k^* = \max_{x \in X} Z_k(x)$$

$$W_s^* = \min_{x \in X} W_s(x)$$

olarak belirlenir.

Her iki amaç fonksiyonu için pozitif ideal çözüm kümesi

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

şeklinde ifade edilir.

Maksimizasyon ve minimizasyon yönlü amaçlar için negatif ideal çözümler;

$$Z_k^- = \min_{x \in X} Z_k(x)$$

$$W_s^- = \max_{x \in X} W_s(x)$$

olarak belirlenir. Negatif ideal çözüm kümesi;

$$O^- = (Z_1^-, Z_2^-, \dots, Z_r^- ; W_1^-, W_2^-, \dots, W_r^- )$$

şeklinde ifade edilir.

Üstün Olmayan Çözüm: Bu çözüm, Çok Amaçlı Karar Verme 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,

$$\exists x' \in X'$$

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

veya

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

Genel anlamda üstün olmayan çözümlerin sayısı oldukça fazladır.

Çok Amaçlı Optimizasyon problemlerinde kısıtların sınırlı miktarları, amaç fonksiyonları arasında farklı birim ve yapıların olması beklenen bir durumdur. Bu sebeple “optimizasyon” kelimesi çok amaçlı programlama problemlerinin kapsamında bulunmaz. Çünkü, bütün amaç fonksiyonlarının eşzamanlı olarak optimum durumu hemen hemen imkansızdır (Umarusman, 2007:11).

#### 1.2.2.1. Baskınlık

Optimalite tek amaçlı problemlerin çözümünde önemli bir rol oynar. Bu karar vericilerin dikkatlerini tek bir çözüm ya da uygun çözüm kümesinin daha küçük bir alt kümesi ile sınırlamasına neden olur. Baskınlık kavramı çok amaçlı problemler için benzer fakat daha az sınırlanmış amaçlar sunmaktadır (Cohon, 1978: 69).

“Baskınlık” olarak adlandırılan kavram, çok amaçlı problemlerde benzer ancak daha az sınırlayıcı bir işleve sahiptir. Çok amaçlı programlama problemlerinde tek bir en iyi çözüm değil, “baskın çözümler kümesi” dikkate alınmaktadır. Baskın çözüm, hiçbir amacın erişim düzeyinin, diğer bir amacın erişim düzeyini azaltmaksızın arttırılamadığı bir çözümdür (Günyaşar, 1990:9).

İki amaçlı bir problemde tabloda gösterildiği gibi üç çözüm olduğunu varsayalım. A ve B çözüm seçenekleri  $Z_1$  ve  $Z_2$  amaçlarının her ikisi için C seçeneğine kıyasla daha yüksek değerler sunduğundan, C seçeneği basılgın niteliktedir. A ve B seçenekleri ise tabloda görüldüğü gibi baskın seçeneklerdir (Cohon, 1978: 69).

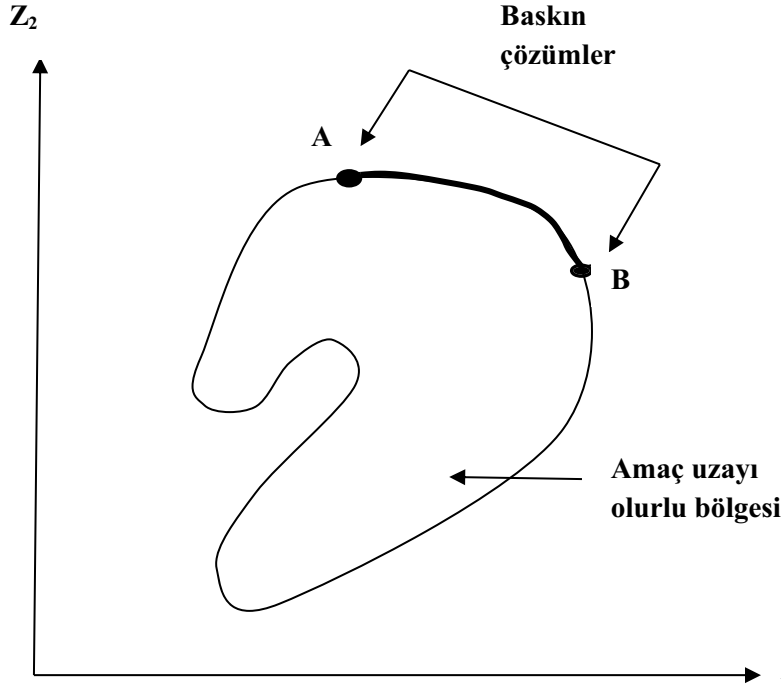
**Tablo 2:** Baskınlık Örnek Problemi

| Seenek | $Z_1$ | $Z_2$ |           |
|---------|-------|-------|-----------|
| A       | 10    | 11    | Baskın    |
| B       | 12    | 10    | Baskın    |
| C       | 9     | 8     | Basılıgın |

Kaynak: Cohon, 1978: 70.

Baskınlık kavramı grafik gösterimle daha iyi anlaşılabilir. Olurlu bölgedeki baskın çözümler, A ve B noktaları arasındaki taralı sınırda yer almaktadır (Cohon, 1978: 70).

**Şekil 4:** Baskın Kümenin Görünümü



Kaynak: Cohon, 1978: 70.

Çok amaçlı programlama genel modelinde,

$$X = \{x \mid g_i(x) \leq 0, x_j \geq 0; \forall i, j\}$$

Olarak tanımlanan bölge, karar uzayı olurlu bölgesidir. Bu bölgeye karşılık gelen amaç uzayı olurlu bölgesi ise  $Z(x)$  ile gösterilir. Her  $x \in X$  için, amaçların tümü  $\{Z_k(x), k=1, 2, \dots, p\}$  şeklinde değerler alır (Günyaşar, 1890: 10).

Baskın çözümlerin tümünü içeren baskın çözümler kümesi,

$$X^* = \{x \mid x \in X \text{ ve } x \text{ baskın çözümdür}\}$$

şeklinde tanımlanır. Her  $x \in X^*$  baskın çözümüne karşılık  $p$  tane  $Z(x)$  amaç işlevi değeri vardır.  $x \in X^*$  için elde edilen  $Z(x)$  değerlerinin tümü,  $Z(X^*)$  baskın

kümesini verir. Baskın çözümler kümesi karar uzayında, baskın küme ise amaç uzayında tanımlanmıştır (Günyaşar, 1890: 11).

Baskın kümede yer alan çözümler, karşılaştırılmaz niteliktedir. Tablo da yer alan A ve B çözümleri amaçlar cinsinde farklı değerleri yansıtmakta olup, hangi seçeneğin daha iyi olduğu doğrudan belirlenemez. A yerine B' yi seçmek,  $Z_1$  amacıyla iki birim artışa karşılık  $Z_2$  amacıyla bir birim azalışa yol açacaktır. Bir amaçta artış sağlamak için diğer bir amaçtan verilmesi gereken ödün miktarına “takas” adı verilir. Baskın seçeneklerin birinden diğerine geçişte amaçların birbiriyle takaslanması, baskın kümenin karakteristik özelliğidir (Günyaşar, 1890: 11).

### **1.2.3. Çok Amaçlı Programlama Sınıflandırması/Yöntemleri**

Cohon ve Marks çok amaçlı programlama probleminin çözümüne yönelik tekniklerin Yöneylem Araştırması alanında en popüler sınıflandırmalardan birini sunmuşlardır (Coello vd., 2007: 31):

1. Çözüm üretme teknikleri (Tercihlerin sonradan eklenmesi)
2. Tercihlerin önceden açıklanmasında dayalı teknikler (Etkileşimli olmayan yöntemler)
3. Tercihlerin etkileşimli olarak açıklanmasında dayalı teknikler ( Karar verici ile etkileşimli olan yöntemler).

ÇAKV problemleri, karar vericinin tercih bilgi yapılarının kullanıldığı zamana bağlı sınıflandırılması, içerdiği metotlar ve tercihlerin hangi aşamada kullanılmasına göre dört sınıfta incelenir (Umarusman, 2007: 11):

#### **1.2.3.1. Tercih Bilgilerinin Kullanılmadığı Metot**

Global Kriter metodu olarak da adlandırılan bu sınıflandırmada karar vericiden bilgiye ihtiyaç duyulmaz, ilk olarak problemin kısıtları ve amaçları tanımlanır. Global Kriter metodu, her bir amaç fonksiyonu için ideal noktalardan sapmaların toplamının ideal noktalara oranının minimizasyonunu içerir.

(Umarusman, 2007: 12). Bu yöntemlerde, problem ile ilgili kısıtlar ve amaçlar tanımlandıktan sonra Karar Verici'den (KV) amaçlar arası veya diğer tercihleriyle ilgili bilgiye ihtiyaç duyulmaz. Bu yaklaşımda klasik optimizasyon problemlerinde olduğu gibi, çözüm esnasında KV ile etkileşim yoktur ve KV ye yöntemin bulunduğu çözüm sunulur (Ersöz ve Kabak, 2010: 103).

#### **1.2.3.2. Tercih Bilgilerinin Öncelikli Kullanıldığı Metotlar**

Matematiksel formülasyondan önce analiste başarılmak istenen belirlenen hedefler hakkında bilgi verilir. Bilgilerin kardinal ve ordinal olmasına göre gerekli olan metotlar kullanılarak çözüm gerçekleştirilir. Bu metotlar da kendi arasında iki kategoride incelenmektedir. Bunlar; kardinal bilgiye göre değerlendirilen Fayda Fonksiyonu Yaklaşımı ile kardinal ve ordinal bilginin birlikte değerlendirildiği Hedef Programlama yaklaşımıdır (Umarusman, 2007: 12). Bu yöntemlerde, KV'nin amaçlarla ilgili tercihleri problemi çözecek kişiye baştan verilir. KV, tercihleriyle ilgili bilgiyi problemin formülasyonundan önce veya sonra verebilir. KV, amaçlarıyla ilgili bilgiyi sayısal veya kısmen sayısal kısmen sözel verebilir. İlk durumda, amaçlarıyla ilgili erişilmesini istediği hedefleri veya altına düşülmemesini istediği sınırları sayısal verir. Bir hedefte belli bir yükselme için diğer hedeften ne kadar fedakârlık yapabileceğini bildirir. İkinci durumda, amaçlarıyla ilgili alt ve üst sınırları sayısal ifade ederken amaçları önem sırasına göre sözel sıralandırabilir (Ersöz ve Kabak, 2010: 103).

#### **1.2.3.3. Tercih Bilgilerinin Optimizasyon Sırasında Kullanıldığı Metotlar**

Aynı zamanda interaktif metotlar olarak da bilinir. Çözüm süreci içerisinde karar verici-bilgisayar veya karar verici-analist arasında her aşamada karşılıklı diyalog oluşturulur. Her iterasyonda yeni çözümün belirlenmesi amacıyla karar vericiye ait tercih bilgilerini kullanır (Umarusman, 2007: 12). Bu yöntemler, KV'nin tercihlerinin ardışık tanımına dayanır. Her adımda KV ile bir diyalog

gerçekleştirilir. Diyalogda, yeni çözümü belirlemek için çözüme dayanarak KV' nin tercihleri veya erişilen değerlerden bir kısmının lehine diğerlerinden yapılabilecek fedakârlık miktarı sorulur. Soruların ifade şekli yöntemlere göre değişmektedir. Bu yöntemler, KV' nin problemin karmaşıklığı nedeniyle başlangıçta bilgi veremeyeceğini, bulunacak çözüme göre kısmen bilgi verebileceğini kabul eder. KV, çözüm aşamasında tercihlerini bildirir ve problem hakkında bilgi sahibi olur. Bu yöntemler KV' nin daha fazla gayret göstermesini gerektirir ama çözümün uygulanabilirliğini artırır (Ersöz ve Kabak, 2010: 106).

#### **1.2.3.4. Tercih Bilgilerinin Optimizasyon Sonrası Kullanıldığı Metotlar**

Karar verici üstün olmayan çözüm kümesinden bir alt küme belirleyerek bu kümenin içerisinde en tatminkar olan çözümü seçer. Mollaghasemi ve Pet-Edwards (1997), tercih bilgilerine bağlı olarak kullanılan metotların sahip olduğu avantajları ve dezavantajları ayrıntılı olarak tanımlamıştır (Umarusman, 2007: 12).

Çok amaçlı karar verme problemlerinin çözümüne yönelik tekniklerin çoğunlukla iki şekilde sınıflandırıldığı görülmektedir. Bunlardan birincisi, uygun seçenekler kümesi X üzerindeki varsayımlara göre yapılmaktadır. Buna göre çözüm teknikleri, X kümesinin (a) sonlu sayıda seçenekten oluştuğunun varsayıldığı teknikler ve (b) sonsuz sayıda seçenekten oluştuğunun varsayıldığı teknikler olarak iki gruba ayrılır. İkincisi ise, karar vericinin tercih bilgisini temel alan sınıflandırmadır. Hemen belirtmek gerekir ki, çok amaçlı problemlerinin çözüm tekniklerini çoğu karar vericinin tercih bilgisini temel alan tekniklerdir. Problemin çözüm aşamasında karar verici (KV) ile etkileşimlerine göre sınıflandırılır. Bu nedenle ikinci sınıflandırma şekli daha çok kullanılmaktadır (Ersöz ve Kabak, 2010: 106).

Karar vericilerin çok amaçlı karar verme problemlerine yönelik geliştirilen modellerden biri de çok amaçlı dinamik programlamadır. Çok amaçlı dinamik programlama modelinin teorik açıdan açıklanması ve formülasyonun sunulmasından

nce tezin ikinci blmnde dinamik karar verme ve dinamik programlama zerinde durulmuřtur.

## İKİNCİ BÖLÜM

### DİNAMİK PROGRAMLAMA

Dinamik programlama, yöneylem araştırmasında kullanılan optimizasyon yöntemlerinden birisidir. Biri diğerini izleyen ve karşılıklı etkileri olan bir dizi kararın bütünüyle ele alındığı problemler için geliştirilen karar modelleridir. Amaç, mevcut kısıtlayıcı koşullar altında eldeki sorunla ilgili en iyi karara varmaktır.

Bazı ekonomik değişme ve gelişmeler, gelecek dönem için önceden yapılan planları geçersiz kılabilir. Bu durumda yeni bir planlamaya gereksinim vardır ya da önceki plan güncelleştirilmelidir. Koşullar bir zaman sürecinde değişiyorsa ve bunların alınan kararlara etkisi önemli ise, dinamik programlama modellerine gereksinim vardır.

#### 2.1. DİNAMİK KARAR VERME

Karar verme işlemi; belirli aşamalardan geçerek bu aşamaların her birinde amaç, duyarlılık veya ölçek göz önünde bulundurularak oluşturulacak seçenekler, uygun yöntem ve teknikle birlikte teknolojinin de kullanılarak ileri geri besleme ile en uygun çözümün bulunmasına kadar yapılacak olan bütün işlemlerin bir sistem çerçevesinde gerçekleştirilmesidir. Karar verme süreci, dinamik bir yapıya sahiptir. Bilgi ve planlar sürekli değişmektedir ve teknolojik gelişmeler bu değişimi tetiklemektedir (Acar ve Eker, 2001: 68).

Edward 1962’de dinamik karar vermeyi aşağıdaki özelliklere göre nitelendirmiştir (Fox vd., 2013:2):

1. Kararlar dizisi bir hedefe ulaşmak için gereklidir,
2. Kararlar bağımsız değildir (kararlar daha önceki kararlar ile kısıtlanmaktadır),
3. Çevredeki değişikliklerle ya da karar vericinin hareketlerinin sonucu olarak problemdeki değişimlerin durumu ve
4. Kararlar gerçek zamanda verilir.

Çok kriterli karar verme problemleri doğal yapısı itibari ile dinamiktir. Çünkü bu problemlerde değişkenlerin durumlarını, sistemin aşamalarını ve kontrolünü ifade eden kararlar vardır. Karar değişkenleri sistemin dinamikleri nedeni ile durumları etkilemektedirler. Bu tür problemlerin en belirgin örnekleri, üretim planlaması problemleri, kontrol problemleri ve su kaynakları planlaması v.b. problemlerdir (Özdemir ve Gümüsoğlu, 2006: 363). Dinamik karar vermenin, fabrika üretimi, klinik karar verme, hava trafik kontrolü, askeri komuta ve kontrol ve acil durum yönetimi gibi alanlarda da uygulanması önemlidir (Fox vd., 2013:2).

Dinamik kararlar, belirsiz çıktılar veya çatışan amaçlar dikkate alınarak farklı zaman aralıklarında verilen kararların, sonradan verilecek kararları etkilediği karar süreçleri olarak ifade edilmektedir. Zaman içinde gelişen kararlar eş zamanlı olmayan verilere dayanmaktadır ve sıralı amaçları belirtmek, her karar veya otomatik karar sistemi için çok büyük zorluklara neden olmaktadır. Bu zorluğu ortadan kaldırmaya en uygun biçimsel yaklaşım ise Dinamik Programlamadır. Dinamik programlama zorluğu ortadan kaldırmak için mükemmel fakat hesaplama açısından oldukça zor olan bir algoritma ile tüm olası alternatiflerin kapsamlı analizinin yapıldığı bir tekniktir. Eğer problemler sıralı kararlar dizisine dönüştürülebiliyorsa dinamik programlama, dinamik kararların çözümünde en önemli tekniktir (Özdemir, 2004:5).

## **2.2. DİNAMİK PROGRAMLAMA**

Dinamik programlama, yöneylem araştırmasının optimumlaştırma yöntemleri arasına giren, doğrusal programlamadan bir aşama daha ileride olan, bir işlem yerine birden çok işlemleri sıra ile elde edebilen, özel bir planlama ve programlama yönetimidir. Dinamik programlama yöntemi, tekrarlama faktörüne dayanır ve bir kritere bakılarak optimal karara ulaşıldıktan sonra, önceki kararın sonuçlarının benimsenilmesine göre bir sonraki karar için optimallik araştırmasını yapar (Demir, 1974: 298).

Diğer tekniklerden farklı olarak dinamik programlama yaklaşımında, karar problemine belli bir anda veya aşamada çözüm aranmaz. Bu yöntemle karşılaşılan problemi, önce alt bölümlere (aşamalara veya küçük problemlere) ayırarak her alt problem veya aşama için çözüm aranır. Daha sonra, her bir alt problemin optimum çözümüyle yetinilmeyip “tüm problem” için optimum çözüm bulmaya çalışılır (Doğan,1985: 122 ).

Koşulların zaman süreci içinde değiştiği ve bu koşulların verilecek kararlar üzerinde önemli etkilerinin olduğu biliniyorsa, dinamik programlama modellerine ihtiyaç duyulur. Dinamik programlamada, mevcut sistem birbiri ardından işlem gören parçalara ayrılmakta ve ardışık iki işlem arasında fonksiyonel bir bağıntı kurulması yoluna gidilmektedir. Bu yöntem daha çok birbirleriyle ilişkili bir dizi kararlar almayı gerektirdiğinden işlemler yineleme denklemleri kullanılarak yapılmaktadır. Yineleme denklemleri ile optimizasyonda, optimizasyon bir önceki kararın içerdiği bilgilerden yararlanarak adım adım gerçekleştirilir. Her adımda bulunan çözüm kendi başına problemin çözümü olmayıp, optimal çözümün bir parçasını belirleyen bilgiyi içermektedir. Her alt problemde verilen karar bir sonraki aşamada verilecek olan kararı etkileyeceğinden, her alt problemde verilen kararın sadece o alt probleme olan etkileri değil aynı zamanda sonraki bütün alt problemlere olan etkileri de göz önüne alınmalıdır. Alt problemler birbirleriyle bağlantılı olduklarından bir sonraki alt problem için gereken bilgi bir önceki alt problemde elde edilen bilgi olacaktır (Sevinç, 2008: 97).

En yalın anlatımıyla dinamik programlama, çözümü güç olabilecek bir problemi daha küçük alt problemlere ayrıştırarak çözüme ulaşmayı sağlayan bir optimizasyon tekniğidir (Durucasu, 2004: 116). Dinamik programlamanın ana fikri esas problemi daha küçük boyutlarda problemlere ayırarak çözmektir (Biyani vd., 2012: 421).

Dinamik programlama birçok optimizasyon probleminin çözümünde kullanılabilen bir tekniktir. Çoğu uygulamada dinamik programlama, problemi geriye doğru çözerek problemin sonundan başına doğru sonuçlar elde eder, böylece

büyük ve zor bir problemi çözülebilir daha küçük parçalara ayırır (Winston, 1991: 945). Çok aşamalı karar ve optimizasyon problemlerinin pek çok çeşidinde, her kararın etkilerini ayrı ayrı hesaba katarak optimum genel politikaya ulaşırız. Fakat bu ulaşılan her bir sürecin optimizasyonu yönteminden sağlanan tüm getiriye kesinlikle açıklamaz. Dinamik programlama ise her olasılığı araştıran bir yöntemdir (Zabransky, 1965: 2).

Dinamik programlama, büyük ve çok sayıda karar değişkeni olan sorunları, ardışık küçük sorunlara bölerek çözmek için geliştirilmiş bir yöntemdir. Yöntemin bir çok uygulamasında zaman içinde ardışık kararlar bulunması, diğer bir deyişle, modelde zaman değişkeninin açık olarak ele alınması, yönteme “dinamik” adının verilmesine neden olmuştur. Dinamik programlamada her alt sorun ayrı ayrı izlenerek ele alınır ve sonuçta tüm sorun optimal olacak biçimde çözülür (Tütek ve Gümüüşoğlu, 2005: 345).

Dinamik programlama birbiriyle ilişkili kararların bir dizi halinde çözümünü sağlayan matematiksel bir tekniktir. Optimal kararın belirlenmesi için sistematik bir yöntem sağlar. Doğrusal programlamanın aksine, dinamik programlama problemlerinin çözümünde kullanılacak standart bir matematiksel formülü yoktur. Buna rağmen, dinamik programlamanın problem çözümüne yaklaşımında genel bir modeli vardır ve her bir durum için probleme uygun özel denklemler geliştirilir (Hillier ve Lieberman, 2010: 424). Her problem farklı algılanır ve dinamik programlama tekniği ile çözümlenebilmesi için problemi tanımlayacak bir model kurulmasına ihtiyaç vardır (Sinha, 2006: 502).

Dinamik programlama, sistem analizi alanında yaygın olarak kullanılan bir yöntemdir ve çok aşamalı karar verme problemlerinde optimal bir silsileye karar vermede kullanılabilir. Dinamik programlama, özellikle karar aşamasının zaman periyodunda silsile halinde olan problemlere çok uygundur. Periyotlar birbirine öyle bir bağla bağlıdır ki bir zaman döneminde alınan kararlar sonraki karar verme aşamalarını etkilemektedir. Problem, alt problemlere bölünür ve her bir alt problem için optimal bir çözüm bulunur, n sayıda karar verme aşamalarına sahip bir problem,

n sayıda ve her biri tek bir karar deęiřkenine sahip, problemlere bölünür. Hesaplama süresi, bir problem içindeki deęiřkenler sayısınca eksponensiyal olarak büyürken, alt problemler sayısınca doğrusal olarak büyür (Çetin, 2005: 140).

Bir problemin tümü sistem ve alt problemler de basamak olarak düşünülebilir. Dinamik programlamada basamaklar, genellikle, bir zaman aralığını temsil eder. Bir sistemin her bir basamağında, problemin çözüm řanlarına karşılık gelen birden fazla durum vardır. Durumlar, tamamlanmamış çözümleri karakterize eder. Karar verici, her bir basamakta, o basamak için en iyi sonucu veren karar vermelidir. Bir karar, sistemi bir durumdan diğetine taşır. Bir sistemi bir durumdan diğetine taşıyan her bir aşamaya basamak denir (Çetin, 2005: 140).

Bir doğrusal programlama problemi zaman faktörü dikkate alınarak incelendiğinde, ancak geleneksel ve orijinal teknikler yerine zaman faktörünü uygulayan yeni bir yaklaşım ile objektif çözüm olanaklı görülebilir. Bir modelin parametreleri zamanla deęişme gösterdiğinde, o modele dinamiktir denebilir. Bu parametrelerin veya hiç olmazsa olasılık dağılımlarının ileriye doğru giden belirli zaman sınırı için zamanın fonksiyonu olduğu kabul edilebilir. Dinamik bir durum için uzun süreli optimal politikanın tamamlanması çok güçtür; çünkü kararların ileriye doğru durmadan deęişen bilgilere dayandırılması zorunluluęu vardır (Demir, 1974: 297).

Dinamik Programlama modelleri özellikle zaman periyotları üzerine yayılan prosesler için uygundur. Metot bütün periyotlarda alınan optimal kararlar kombinasyonunu arařtırmayı kapsar ve çok fazla bilginin hesaplanmasını gerektirir. (Halaç, 1972: 109). Dinamik programlamanın diğere tekniklere göre üstünlüğü, zaman faktörünü deęerlendirmesi yanında; çok boyutlu problemlerin çözümünde etkin hesaplama kolaylığı sağlamasıdır (Doğın,1985: 122 ). Bir problemin açıklanmasında zaman faktörü rol oynuyor ise, kullanılacak en uygun yaklaşımın adı «Dinamik Programlama» dır (Demir, 1974: 297).

Dinamik programlama problemleri ařağıdaki ilkeler uygulanarak çözüme ulařtırılır (Halaç, 1991: 162):

- Çok aşamalı (veya kademeli) problem alt parçalara, adımlara veya tek tek aşamalara bölünür. Buna ayrıştırma veya dekompozisyon denir.
- Her aşamada, her seferinde bir kez olmak üzere (veya yineleme ile) belirli bir en uygun şekle sokma amacına bağlı kalarak karar verilir.
- Problemin tümüne ait çözümü elde etmek için aşama sonuçları birleştirilir. Birleştirme sonucu politika olarak adlandırılan kararlar dizisi elde edilir.

Dinamik programlama problemlerinin en önemli prensibi R. Bellman'm Optimallik Prensibi'dir. Başlangıç değerleri ve başlangıçta alınan karar ne olursa olsun, bu prensip, herhangi bir adımda optimal olarak seçilen yolun özelliği göz önünde tutularak, ilk alınan kararın sonucu olarak varılan duruma göre geriye kalan kararların optimal olmasını gerektirecek biçimde olur. Yani izlenecek çözüm yolu öyledir ki, önceki kararlar ne olursa olsun, bunu izleyen adımlar yine optimum politikayı verir. Örneğin; birinci ve ikinci dönemde yanlış kararlar verilse bile, üçüncü, dördüncü ve sonraki adımlarda (basamaklarda) doğru kararlar verilebilir (Demir, 1974: 299).

Optimallik prensibi, dinamik programlamanın aslını oluşturur. Bu prensibe göre birbiri ardından gelen iki adım arasında optimal değeri verebilecek biçimde ilişki kurulabilmektedir. Dinamik programlama her problemi ayrı ayrı ele alır ve bunların fonksiyonel denklemini kurarak, bu denklemin çözümünü araştırır (Demir, 1974: 297).

### **2.2.1. Dinamik Programlamanın Doğuşu ve Gelişimi**

Bellman 2. Dünya savaşında görev almasına rağmen, 1946 yılında doktora tezini Princetand'da tamamlamıştır. Bellman matematikte ve fiziksel dünyadan kaynaklanan sorunlara çözüm bulmada var yeteneğini sergiledi. Başarılı akademik kariyerinin ardından daha sonraları Yöneylem Araştırması adıyla anılacak uygulamalı matematiğin bir türüyle ilgilenmeye başladı. 1949 yazı boyunca Santa Monica'da RAND şirketinde analitik sayılar teorisi üzerine uygulamalar yaptı ve RAND'da kendisinde çok aşamalı karar süreci ile ilgilenmesi istenildi. Bilimsel

açından matematiksel araştırmalar için pek uygun olmayan 1950’li yıllarda RAND şirketi hava kuvvetlerinden bir iş aldı ve bu iş için Bellman görevlendirildi. Yaptığı çalışmanın başlığının ne olması gerektiğini düşünen Bellman önce “Karar Vermede Planlama” yı düşünmüştür. Fakat çeşitli nedenlerle planlama kelimesinin uygun olmadığına ve bunun yerine programlamayı kullanmaya karar vermiştir. Dinamik kavramını çok aşamalı ve zaman değişkenli olduğu için insanların çok daha rahat anlayacağı bir kelime olarak düşünmüş ve bu kelimeyi programlama ile birleştirebileceğine karar vermiştir (Dreyfus, 2002: 48)

Bellman, optimizasyon problemlerinin çok fazla çeşidinde ortak özellikler gözlemledi ve dinamik programlama lisanını bulmuştur. Bellman optimallik denklemi yerine fonksiyonel denklem kullanmıştır. Bellman, kontrol teorisi, ekonomi, matematik, yöneylem araştırmaları, tıp ve fizik gibi çeşitli alanlarda görünüşte farklı yüzlerce problemi çözmek için bulduğu bu yöntemleri kullanmıştır (Denardo, 2011: 278-279).

Dinamik programlamanın özellikli sonuçlarından pek çoğu David Blackwell, Lester Dubins ve Leonard Savage ve Ronald Howard’ın çalışmalarında bulunabilir. Howard (1961)’ın kitabı konuya matematiksel olmayan mükemmel bir giriş yapmaktadır. Kısa, kullanışlı ve açık bir başlangıç değerlendirmesi Ross (1970)’un kitabında verilmektedir. Mine ve Osaki(1970)’nin kitabı yaklaşık olarak başlangıç seviyesinde fakat daha fazla detaya inmektedir. Matematiksel bir giriş ise Blackwell (1965,1966) ve Strauch (1966)’un çalışmalarında ve Dubins ve Savage (1965)’in kitaplarında bulunabilir. Bertsekas ve Shvere (1978)’ in kitabı ise yüksek matematik seviyesinde ve literatür için kullanışlı bir kılavuzdur. Dreyfus ve Law (1977)’ın kitabı ise örnekleri bakımından zengindir ve çok sayıda problem ve çözümlerini içermektedir (Balakrishnan, 2010: 137).

### **2.2.2. Dinamik Programlamanın Uygulama Alanları**

Dinamik Programlama, 1920’den bu yana önemini arttırarak gelişmiştir. Başlangıçta yalnızca bir ekonomik sistemin zaman içindeki durumunun

incelenmesinde kullanılmış olan yöntem, günümüzde zamanla ilgili olanların yanı sıra, daha değişik nitelikteki süreçlerin incelenmesinde de yaygınca kullanılmaktadır (Yücel ve Ulutaş, 2010: 275).

Dinamik programlama kuramı A. Wald'ın Sıralama Analizi Kuramı (Theory of Sequential Analysis) ile yakından ilgili bulunmaktadır. Daha geniş açıklamaları Amerikalı Matematikçi Richard Bellman'ın Dinamik Programlama adlı yapıtında ve çok sayıdaki makalelerinde, H. Simon, S. Karlin, F. Modigliani, F. Hohn ve K.J. Arrow'un çalışmalarında bulabiliriz. Uygulamada ise dinamik programlama ile ilgili en iyi örneklerden birini A. Vazsonyi vermektedir. Önemli problemlerin çözümlenmesinde dinamik programlama belirli bir teknik olarak birçok bilim adamları yakından incelenmekte ve geliştirilmektedir (Demir, 1974: 297).

Dinamik programlamanın en fazla uygulandığı işletme sorunları aşağıdaki biçimde sıralanabilir (Tütek ve Gümüşoğlu, 2005: 345):

- Yeniden sipariş kurallarının belirlenmesinde zaman ve nicelik değişkenlerinin saptanması,
- Değişen istem koşullarında üretim programlaması ve istihdam düzenlemesi,
- Pahalı araç ve gerecin etkin biçimde kullanımını garantilemek üzere yedek parça düzeyinin belirlenmesi,
- Yeni alanlara kaynak dağıtımını yapan sermaye bütçelemesi,
- Yapılanları halka geniş ölçüde tanıtabilmek için reklam araçlarının seçimi,
- Kıymetli bir kaynağın bulunmasında sistematik aramanın yapılması,
- Karmaşık makinelerin bakım ve onarımının programlanması,
- Eskiye donanım ve makinelerin yenilenmesi için uzun dönem stratejilerin saptanması,
- Çeşitli materyaller için (genellikle rulo türü kağıt, boru, vb.) kesme kalıplarının belirlenmesi.

Son yıllarda dinamik programlamanın uygulandığı işletme dışı alanlardan bazıları aşağıda belirtilmiştir:

- Beklenen kaza maliyetleri göz önünde bulundurularak minimum riske sahip yolların belirlenmesi (Serafini, 2006: 224).
- Hava yolu şirketinin, uçak satın alma, (finansal) kiralama ya da elden çıkarmaya ilişkin kararlarının optimizasyonunun sağlanması (Hsu vd., 2011: 41).
- Gemi filosu planlamasında her bir dönem için tamsayı doğrusal programlama formülasyonunun kullanılması ve modelin çevrimsiz ağda en kısa yol algoritması ile çözülmesi ile uzun ve kısa vadede gemi satın alma veya kiralama kararının verilmesi (Meng ve Wang, 2011: 401).
- Lojistik bir sistemde lojistik merkeze bağlı üç dağıtım merkezi boyunca taşıma kapasitesi bölümlenmesi için optimal politikanın belirlenmesi (Başanu vd., 2011: 19).
- Markov zinciri boyunca belirsizlik ve her bir zaman adımında belirsizlik hakkında bilgi veren dinamik işletme politikaları üreten stokastik dinamik programlamaya dayalı bir yaklaşım kullanılarak yeni bir stokastik optimizasyon algoritması ‘sezgisel sınırlı durum uzayında dinamik programlama’nın geliştirilmesi (Choi vd., 2006: 2473).
- Araç navigasyon cihazlarının en önemli özellikleri kullanıcı için optimal rotayı bulmasıdır. Araç navigasyon cihazları için trafik durumunun ve birden çok ölçütün(seyahat zamanı, yolun türü, kavşak dönüşleri ve trafik ışıkları) değişimi dikkate alınarak optimal rotanın bulunması (Mainali vd., 2011:14).
- Zamanlararası varlık fiyatlandırma modellerinin stokastik dinamik programlama ile belirlenmesi (Grüne ve Semmler, 2007: 233).
- Bir baraj işletmesinin iki aşamalı karar sürecine dayalı stokastik dinamik programlama modeli geliştirilerek uzun vadeli işletme ilkelerinin belirlenmesi (Liu vd., 2012:342).
- Pareto optimal tahsislerinin bulunması için dinamik programlama metodlarının kullanılması (Sitarz, 2009: 204).
- Mısır’da 20 yıllık bir planlama dönemi içinde optimal su kaynaklarının büyüklüğünün belirlenmesinde yüksek boyutlu dinamik programlama optimizasyon modelinin uygulanması (Mahmoud, 2006: 371).

İşletme sorunlarının çözümünde dinamik programlama uygulamaları:

- Stok sorununu çözmek için üç aşamalı kesim kalıplarının sırt çantası algoritması oluşturulması (Cui, 2013: 349).
- Firmaların optimum finansman politikalarını belirlemek için teorik bir stokastik dinamik optimizasyon modeli geliştirmek (Cerqueti, 2012: 539).
- Kapasite kısıtlı parti-boyutlandırma sorunu için dinamik programlama tabanlı eşitsizlikler belirlenmesi ( Hartman vd., 2010:915).
- Birden fazla öncelik seviyesi ile genel bir dinamik kapasite tahsisi sorunu için yaklaşık dinamik programlama yaklaşımı geliştirilmesi (Erdelyi ve Topaloğlu, 2010: 129).
- Çok kriterli kesikli dinamik programlama yaklaşımında karınca algoritmaları ve benzetimli tavlama yöntemlerinin karşılaştırılması (Sitarz, 2009: 433).
- Dinamik bir ortamda uluslararası portföy yönetimi için bir çok aşamalı stokastik programlama modeli geliştirmek (Topaloglu, 2008:1501).
- Endüstride gittikçe artan çok amaçlı ve iki boyutlu bir ambar yükleme sorununun dinamik programlama yaklaşımı ile optimizasyonunun sağlanması ( Liu, 2012: 4316).
- Araç yönlendirme sorunlarında ortaya çıkan fiyatlandırma alt problemleri için dinamik programlama ile kaynak kısıtlı en kısa yol problemlerinin çözülmesi (Righini ve Salani, 2007:155)

### **2.2.3. Dinamik Programlama Kavramları**

Dinamik programlama, bir dizi karar verme işlemini optimize eden bir matematik işlemleri bütünüdür. Temelde, dinamik programlama, problem çözümüne, problemin veya problemin bir kısmının parçalara bölünmesi ve bu parçaların çözülerek, bu çözümlerin depolanması şeklinde bir problem çözüm yaklaşımı sunmaktadır. Bu çözümler, ihtiyaç duyulduğunda, yeniden çözmek yerine, yeniden canlandırılmak suretiyle problemin genel çözümüne eklenerek, nihai çözüme ulaşılmaktadır (Çetin, 2005: 140).

Dinamik programlama ilişkili kararlar dizisini kapsayan problemlerin çözümünde kullanılır. Dinamik programlama problemlerinin formülasyonu, aşamalar, durum değişkenleri, karar değişkenleri, getiri fonksiyonları ve optimal karar politikalarını tanımlamayı gerektirir (Bierman vd., 1981: 559).

**Durum:** İncelenen problemin bir sonraki geleceği durum doğrudan şu andaki durumuna bağlıdır ve verilen problemin durumuna bağlı olarak ilgili safhadaki faaliyet seçenekleri tespit edildiğinden, bu seçeneklere durum değişkenlikleri denir. Yani, her safhanın başında incelenen problem bir durumdan diğer duruma değişir yeni durum sisteminin bir önceki devredeki durumunun fonksiyonu şeklindedir. Özet olarak, bir safhadaki durum, daha önceki safhada verilen kararların bir sonucudur. Durum değeri, sistemin mevcut durumundan başlayarak özel bir yol izlediğinde ortaya çıkan getirilerin toplam bir fonksiyonudur (Yücel ve Ulutaş, 2010: 280).

**Aşama:** Aşama büyük bir problem çözümlenmek üzere küçük alt problemlere ayrıştırıldığında, her alt problem çözümlenmesi gereken bir karar problemi olarak düşünülürse, karar verilmesi gereken her nokta safha olarak tanımlanır. Bundan dolayı her alt problem bir safhaya karşılık gelmektedir. Safhaların sayısı problemde yer alan düğüm sayısına ya da alt probleme göre değişebilir. Aşama sayısı sürecin uzunluğuna bağlıdır. Safha yapısını belirleyen önemli bir özellik, sürecin sürekli ya da kesikli olmasıdır. Tamsayılarla ifade edilen kesikli bir süreç söz konusu olduğunda safha değişkeni de kesiklidir. Sürekli bir süreçte safha tanımlaması yapmak, uygulama açısından bazı sorunlar ortaya çıkarmaktadır. Safhalar birbirleriyle bağlantılı bir sürecin bölmeleri olup, safhalardan birinin yanlış çözüme ulaştırılması bütün problemi yanlış çözüme iter (Yücel ve Ulutaş, 2010: 280).

**Karar:** Dinamik programlama herhangi bir problem için mümkün olan koşullar içinde ne yapılacağına karar vermeyi sağlar. Karar, problemdeki bir safhayı tamamlamak için ilgili alternatifler arasından bir seçim yapılması işlemidir. Belirli bir durum ve safhada verilen bir karar, sürecin hem durumunu hem de safhasını değiştirir. Dolayısıyla her karar, geçerli bir durumdan bir sonraki safhaya bağlı olan duruma geçişi etkilemektedir. Çok safhalı bir karar süreci safhalara ayrıldıktan sonra

her safha için bir dönüşüm fonksiyonu oluşturulur. Her safhada karar verme süreci, o safhanın seçeneklerinden birinin seçilmesiyle sonuçlanır. Buna safha kararı denir. Ayrıca her hangi bir safhadaki karar uygun çözüm ise bu karar, problemin en uygun çözümünün bir parçası olarak kullanılabilir (Yücel ve Ulutaş, 2010: 281).

**Politika:** Durumların hepsinin içinde olduğu hareketler topluluğudur. Verilen koşullarla uyum içinde olan hareketlerin en iyisi optimal politikadır. Diğer bir deyişle karar sürecinin tüm aşamalarını kapsayan belirli seçenekler dizisi veya seti “politika”, karar vericinin amacını gerçekleştirecek optimum seçenekler dizisi de “optimum politika” diye adlandırılmaktadır (Özdemir, 2004: 16).

**Getiri:** Her karar, amaç fonksiyonuna katkı sağlayan bir getiri ile sonuçlanır. Getiri bir sürecin her bir aşamasında meydana getirilen bir sistemdir. Getiri genellikle kazanç, maliyet, mamul üretimi ya da kaynak tüketimi olabilmektedir (Özdemir, 2004: 16).

**Durum Değeri:** Durum değeri, sistem o durumdan başlayıp özel bir politika izlendiğinde ortaya çıkan getirilerin toplam bir işlevidir. Değer, süreçteki aşamaların optimal uzaklıklarının toplamı ile belirlenmektedir. Optimal politika altında durum değeri optimal değerdir (Özdemir, 2004: 16).

Dinamik programlama uygulamalarının sahip olduğu ortak temel özellikler aşağıda verilmiştir (Hillier ve Lieberman, 2010: 429):

- Problemler, belirli seçenekleri veya karar politikalarını içeren bir dizi sıralı aşamalara bölünebilmektedir. Tüm dinamik programlama problemleri, sıralı karar aşamalarına ayrılarak, karar problemlerinin bir aşamaya karşılık geldiği birbiriyle ilişkili kararlar dizisinin oluşturulmasını gerektirmektedir.
- Her bir aşama o aşamanın başlangıcıyla ilgili birçok duruma sahiptir. Durumlar problemin ilgili aşamasında sistemin içinde bulunduğu olası koşullardır. Durumların sayısı sonlu ya da sonsuz olabilmektedir.
- Her aşamadaki politika kararının etkisi, belirli bir durumun bir sonraki aşamanın başlangıcına ilişkin bir duruma dönüşmesidir.

- Çözüm yöntemi tüm problem için optimal bir politika bulmak üzere modellenmiştir.
- Bir optimal politika kararının verildiği mevcut durumda, sonraki aşamalarda alınacak kararlar sonraki aşamalarda alınacak kararlardan bağımsızdır. Bu nedenle optimal karar, mevcut duruma ve bu mevcut duruma nasıl gelindiğine bağlıdır. Bu, dinamik programlama için optimallik prensibidir.
- Çözüm yöntemi son aşama için optimal politikanın bulunması ile başlar. Son aşama için optimal politika, o aşamadaki her bir olası durum için optimal politika kararını içermektedir.
- Aşama  $n+1$  için optimal politika belli ise, aşama  $n$  için optimal politikayı belirten tekrarlı yapıya ulaşılabilmektedir. Aşama  $n$ 'de ve  $s_n$  durumda başlanması halinde optimal politika kararının verilmesi  $x_n$ 'in değerinin bulunmasını gerektirmektedir. Dinamik programlama karar probleminin fonksiyonel eşitlik şeklindeki modeli aşağıdaki gibi gösterilebilir.

$$f_n^*(s_n) = \max_{x_n} [f_n(s_n, x_n)] \text{ ve ya } f_n^*(s_n) = \min_{x_n} [f_n(s_n, x_n)] (s_n, x_n)$$

Bu modelde;

$N$ : aşama sayısı,

$n$ : mevcut aşama ( $n=1,2,\dots,N$ ),

$s_n$ :  $n$ . aşamadaki mevcut durum,

$x_n$ :  $n$ . aşama için karar değişkeni,

$x_n^*$ :  $s_n$  verildiğinde  $x_n$ 'in optimal değeri,

$f_n(s_n, x_n)$ :  $n$ . aşamada ve  $s_n$  durumunda verilen karar  $x_n$  'in amaç fonksiyonuna katkısı,

$f_n^*(s_n)$  :  $f_n(s_n, x_n^*)$  olarak ifade edilir.

- Problemlerin çözümünde yineleme ilişkisi kullanılarak çözüm işlemi sondan başa doğru sürdürülür. Her aşamada her bir durum için optimal politika bulunarak, ilk aşama için optimal politika elde edilinceye kadar işlemlere devam edilir.

Dinamik programlama, çok aşamalı karar süreçlerinin optimizasyonunu sağlayan matematiksel bir tekniktir (Sinha, 2006: 502). Ardışık karar sorunlarını çözmeye etkili bir yöntem olan dinamik programlama, doğrusal programlama ile karşılaştırılacak olursa her iki yöntemin de amacı kaynakların verimli dağılımıdır; ancak doğrusal programlama varsayımları modeli oldukça kısıtlar. Buna karşın dinamik programlamayla çok daha çeşitli sorunlar ve ilişkileri modellenenabilir. Ancak dinamik programlama sorunun çözümünde genelde kabul edilmiş bir algoritmanın olmaması modelin çözümünde zorluklar ortaya koyabilir. Ele alınan her sorun için sorunun yapısına uygun eşitlikler geliştirmelidir. Bu yüzden yöntem belirli düzeyde yaratıcılık ve görüş yeteneği gerektirmektedir (Tütek ve Gümüšoğlu, 2005: 345).

#### **2.2.4. Dinamik Programlamanın Yinelenen Yapısı ve Optimallik İlkesi**

Karar değişkenlerinin amacı gerçekleyen değerlerinin belirlenmesine yönelik problemlerin çözümünde, sayıları ne olursa olsun, karar değişkenleri genellikle topluca veya eşanlı olarak işleme tabi tutulur. Oysa, zaman zaman birbiriyle ilişkili bir dizi karar alınmasını gerektiren problemlerle de karşılaşılabilen ve değişkenlerin topluca göz önünde bulundurulması durumunda problemi çözmek olanaksız veya çok zor olabilmektedir. Bu gibi durumlarda problemin küçük alt problemlere bölünmesi, bir başka deyişle ayrıştırılmasından sonra her bir alt problem için çözüm aranması ve alt problemlerinin birleştirilmesi sonucunda bütün problemin en iyi çözümünün bulunması daha uygun olabilmektedir. Kısaca açıklanan bu yaklaşım, “çok aşamalı problem çözme tekniği” olarak bilinir. Dinamik programlama, çok aşamalı problemlerin çözümünde etkin biçimde kullanılabilen uygun bir hesaplama yöntemidir (Cinemre, 2011:559).

Dinamik programlamada hesaplamalar yinelenerek yapılır, bu bakımdan bir alt problemin optimum çözümü bir sonraki alt problemin girdisidir. Son alt problemi çözdüğümüzde, problemin tamamının optimum çözümüne ulaşmış oluruz. Yinelenen hesaplamaların uygulanma biçimi orijinal problemi nasıl ayrıştırdığımıza bağlıdır. Özellikle, alt problemler bazı ortak kısıtlarla normalde birbirleriyle

ilişkilendirilmişlerdir. Bir alt problemden bir sonrakine ilerledikçe bu kısıtların uygunluğuna dikkat etmek zorunludur (Taha, 2009: 403).

Dinamik programlama çok yaygın bir kullanım alanı olan yineleme denklemi ile problemi en uygun şekle sokma tekniğidir. Yineleme denklemi ile optimizasyon yöntemi deyimi optimizasyonun bir önceki aşamanın içerdiği bilgilerden yararlanarak aşama aşama yapıldığını anlatır. Diğer matematik programlama yöntemlerinde de en iyi çözüme aşamalarla gidilmektedir, ancak yaklaşım yineleme değil iteratiftir. Örneğin doğrusal programlama iteratiftir. Dinamik programlama ise her aşama daha önceki aşamalar ile sırasal olarak ilişki içerisindedir. Her bir aşamadaki çözüm kendi başına problemin bir çözümü değil, ancak optimal bir çözümün bir parçasını belirleyen bilgiyi içermektedir ( Patır, 2009: 64).

Dinamik programlama yaklaşımında amaç, ardışık ve birbirlerini etkileyen alt problemler için ayrı ayrı optimum kararlar bulup bulunan sonuçlar toplayarak problemi sonuçlandırmak değil, asıl problem için optimum sonucu verecek şekilde alt problemler için en iyi karar bulmaktır. Yani bazı durumlarda alt problemlerde en iyi olmayan bir karar gerçek problem için optimum sonucu sağlayan karar olabilmektedir (Yücel ve Ulutaş, 2010: 273).

Dinamik programlama aşamalı veya sıralı bir şekilde çalışan sistemlerin matematiksel gösterimi ya da optimizasyonu için bir yöntemdir. Bu yöntemin temeli pek çok durumda kanıtlanmış, optimallik prensibi olarak açık bir şekilde ifade edilebilen optimallik teoremidir (Kaufmann, 1967:77).

#### **2.2.4.1. Optimallik ilkesi**

Dinamik programlama genel olarak, belirlenen kararların bir sıraya sokulmasını gerektiren optimizasyon problemlerinin çözülmesinde kullanılan bir yöntemdir, her karar için alt problemler aynı şekilde çözülür. Böylece ana problemin optimal çözümü alt problemlerin optimal çözümlerinden elde edilebilir. Bu yöntem Bellman' ın Optimallik Prensibine dayanır (Lew ve Mauch, 2007: 5). Optimal bir

politikanın özelliği, başlangıç durumu ve başlangıç kararları ne olursa olsun geri kalan kararların, ilk verilen kararların sonucuna göre optimal bir politika oluşturmasıdır (Sinha, 2006: 503).

Optimallik prensibi toplam problemi parçalara bölmeyi ve son karar aşamasını çözmeyi, daha sonra geriye doğru dönmeyi ve ikinci son karar aşamasını çözmeyi, ilk karar aşaması çözüme ulaştırılana kadar bu çözüm yöntemine devam etmeyi gerektirir (Bierman vd., 1981: 545).

Hiçbir optimal politika problemin diğer optimal alt politikalarından ibaret değildir (Başanu vd., 2011: 19). Optimal politika ya da karar öyle bir özellik taşımalıdır ki bir duruma nasıl erişildiği göz önüne alınmaksızın sonraki kararlar, o durumun terk edilmesinden sonra optimal politikayı oluşturmalıdır (Tütek ve Gümüšoğlu, 2005: 347).

#### **2.2.5. Dinamik Programlama Formülü**

Dinamik programlama (DP) ardı sıra yapılması zorunlu olan karar problemleri ile ilgili bir yaklaşımdır. DP, her kararın doğduğu gibi optimize edilmesi yerine, ilerdeki periyotlar üzerine mevcut kararların etkisini göz önüne alır ve en iyi performansı sağlayan her bir kararı düzenler (Halaç, 1972: 109). Dinamik programlama ana problemin küçük bir bölümüyle çözüme başlar ve bu küçük problemin optimal çözümünü bulur. Problem aşama aşama genişletilir, ana problem bütünüyle çözüme ulaştırılana kadar her aşamada bir önceki aşamanın optimal çözümü alınır (Hillier ve Lieberman, 2010: 425).

Dinamik programlama problemlerinin çözümü için kullanılacak standart bir model yoktur. Farklı türden problemler için ortak bir model mevcut değildir. Fakat benzer özellikler taşıyan problemler için aynı optimalite altında çözüm yaklaşımlarını genellemek mümkün olabilmektedir. Bu yüzden dinamik programlama problemlerinin her biri için ayrı ayrı çözüm yaklaşımları mevcuttur (Akdeniz ve Aksaraylı, 2003: 12). Bir başka deyişle, dinamik programlama bir kez

programlanmış bir algoritma sağlayarak dinamik programlama ile çözümü mümkün kılan özelliklere sahip bütün problemlerin çözümünde kullanılacak bir matematiksel model değil, bir hesaplama yöntemidir. Bu nedenle birbirine benzeyen problemler bir araya toplanarak her özel grup için ortak bir model dolayısıyla, çözüm yöntemi geliştirilmeye çalışılır (Cinemre, 2011: 560).

Her programlama probleminde; maksimum ve minimum yapılması istenilen bir amaç fonksiyonu ve ayrıca koşulları belirtilen bir denklemler veya eşitsizlikler grubu vardır. Bir problemde optimal olarak bulunması istenilen  $n$  sayıda değişken ve  $m$  sayıda sınırlayıcı koşullar var ise, bu probleme  $n$  değişkenli,  $m$  koşullu optimumlaştırma problemi denir. Doğrusal programlama böyle problemi bir adımda çözümleyebilir. Doğrusal programlamanın tersine, dinamik programlamada, mevcut sistem parçalara ayrılmış bulunur. Verilen bir optimumlaştırma problemi, her biri bir değişkenli  $n$  sayıda parça optimumlaştırma problemlerine bölünür ve her bir parça problem, adım adım çözülür. Doğrusal programlamadan değişkene genellikle simultan (hemzaman) optimal değerler verilerek bununla yetinilmesine karşılık; dinamik programlamada, çözüm prosedürünün her basamağıyla ilgili ve yalnızca o andaki parça probleminin değişkenini optimalleştirecek ikinci değerli çözüm ilişkisinin yerleştirilmesi yoluna gidilir (Demir, 1974: 298).

Dinamik programlamanın önemli özelliklerinden biri, her probleme ilişkin karar fonksiyonunun karar verici tarafından belirlenmesi gerektiğidir. Karar verici mevcut koşullara göre kendi amaç fonksiyonunu ve kısıtlarını belirleyerek zamana dayalı dinamik yapıdaki sıralı problemlerini dinamik programlama modeli ile çözmektedir (Özdemir, 2004: 103).

Genellikle herhangi bir durumda olabilecek  $x$  aktivitenin bir  $X$  seti vardır. Çoğu zaman özel bir  $(s,n)$  durumunda ulaşılabilir aktivitelerin seti tam olarak belirtmek istenmekte ve  $x_n(n=1,2,...,n)$  biçiminde ifade edilmektedir.

$$f_n^*(s_n) = \max [f_n(s_n, x_n)]$$

$f_n(s_n, x_n)$ : n. aşamada tüm politikaların toplam değeridir. Aşama  $n$  içinde  $s_n$  durumunda verilen optimal karar  $x_n^*$ ,  $f_n(s_n, x_n)$  değerinin optimal olmasını ve dolayısıyla  $f_n(s_n)$  değerinin optimal değerinin elde edilmesini sağlamaktadır. Sonuçta toplam getiri fonksiyonu, mevcut kararlar elde edilen getiri fonksiyonu ( $C_{s,xn}$ ) ile, mevcut kararlar sonuçlanan daha önceki kararların optimal getiri fonksiyonlarının [ $f_{n+1}^*(x_n)$ ] toplamından oluşmaktadır (Özdemir, 2004:103):

$$f_n(s_n, x_n) = (C_{s_n, x_n}) + f_{n+1}^*(x_n)$$

Dinamik programlamada önce problemin küçük bir kısmı çözülür ve daha sonra kademelerin tamamını kaplayıncaya kadar diğer kademeler çözüme alınır. İşlemler ileriye doğru veya geriye doğru yapılabilir (Halaç, 1972: 110). Dinamik programlama genellikle geriye doğru, yani son durumdan ilk duruma doğru, bir işlemler silsilesi şeklinde uygulanır. Bu geriye doğru endüksiyon tekniği, son durumdan, bir önceki basamağın durumlarına doğru yapılır (Çetin, 2005: 140).

#### 2.2.6. Dinamik Programlama Çözüm Yöntemleri

Dinamik programlama,  $n$  değişkenli bir problemin optimum çözümünü problemi  $n$  aşamaya ayrıştırarak ve her aşamada tek değişkenli bir alt problemi çözerek belirler. Bunun hesaplama avantajı,  $n$  değişkenli alt problemler yerine tek değişkenli alt problemleri optimum kılacak olmamızdır. Dinamik programlamanın asıl katkısı, problemleri aşamalara ayrıştırmasının çerçevesini oluşturan optimumluk ilkesidir. Optimizasyon problemine bağlı olarak aşamaların yapısı farklılık gösterdiğinden, dinamik programlama her bir aşamayı optimum kılmak için gereken hesaplamaların ayrıntısını vermez. Bu ayrıntılar problem çözücü tarafından doğaçlama olarak gerçekleştirilip tasarlanır (Taha, 2009: 403).

Dinamik programlama yaklaşımında, ilk çözüme problemin son aşamasından başlanıp her defasında bir önceki aşamaya dönülerek “Geriye Doğru En İyileme” biçiminde bir çözüm yolu benimsenebileceği gibi, problemin ilk aşaması birinci aşama olarak ele alınıp, her defasında bir sonraki aşamaya geçilerek “İleriye Doğru

En İyileme” biçiminde bir çözüm yolu da seçilebilir. Problemdeki aşamaların belirlenmesi ve dönüşüm fonksiyonlarının oluşturulması dinamik programlama yaklaşımının en önemli unsurudur (Durucasu, 2004: 116).

#### 2.2.6.1. Geriye Doğru Hesaplama Yöntemi

Birbirleriyle ilişkili alt problemlere ayrılmış ardışık karar problemi için, problemi eniyileyecek değerlerin, n aşamalı bir problem için en son aşamadan (n. aşama) başlanarak ilk aşamaya (1. aşama) doğru ilerlenerek bulunması geriye doğru çözüm yaklaşımıdır (Akdeniz ve Aksaraylı, 2003: 12).

Bu çözüm yönteminde, çözüme 1 inci safhadan başlanılarak 2, 3,...,n-1, n ve N inci safha için işlemler gerçekleştirilir, ancak işlem sırası akışın aksi yöndedir. Yani çözüme sondan başlanır. Örneğin 6 safhalı bir problem 6 ıncı safhasından başlanarak çözülmeye çalışılır (Yücel ve Ulutaş, 2010: 282).

Dinamik programlamaya esas olan model, bulunulan durumda benimsenecek eylemle erişilecek durumları belirleyen geçiş fonksiyonlarını kısıt olarak alıp, bunlara göre tüm kademelerdeki katkıları belirleyen amaç fonksiyonu ile genelleştirmektedir (Aksaraylı, 1998: 28).

Geriye doğru yineleme ile çözüm yönteminin formülasyonu aşağıdaki gibidir (Yücel ve Ulutaş, 2010: 283):

$$f_i(x_i) = \min_{x_i, x_{i+1}} \{d(x_i, x_{i+1}) + f_{i+1}(x_{i+1})\}, i=1,2,3$$

Genelde problem yapılarına göre değişen bu çözüm yönteminin genel formülasyonu aşağıdaki gibidir (Yücel ve Ulutaş, 2010: 283):

Çözüme başlanacak ilk safha için;

$$f_1(s_1) = \min (\max) [r_1(s_1, x_1)]$$

İkinci safha için;

$$f_2(s_2) = \min (\max) [r_2(s_2, x_2)] + f_1(s_1)$$

N inci safha için;

$$f_N(s_N) = \min (\max) [r_N(s_N, x_N)] + f_{N-1}(s_{N-1})$$

Verilen ifadelerde;

i = Aşama numarası

$x_i$  = i.aşamada karar değişkeni

$r_i$  = i.aşama sonunda sistemin durumu i.aşamada belirli bir durum-karar kombinasyonuna bağlı olarak ortaya çıkan getiri veya sonuç

$f_i$  = i.aşamada kriterin değeri

#### **2.2.6.2. İleriye Doğru Hesaplama Yöntemi**

Dinamik programlama problemleri geriye doğru çözüm yöntemi ile çözülebileceği gibi 1962’de Bhavnani ve Chen tarafından ortaya atılan optimalite prensibinin Dualine dayandırılan “ileriye doğru çözüm yöntemi” de kullanılabilir. Bhavnani ve Chen şöyle demektedir: “ bir optimal politikanın öyle bir özelliği vardır ki, takip eden durum ve kararlar ne olursa olsun, önce gelen kararlar son gelen kararı takip eden duruma göre bir optimal politika oluşturmalıdır.” (Çiçek, 2000: 38).

Çözümde baştan sona doğru mu yoksa sondan başa doğru mu gidileceğini belirleyen, problemin yapısı ve araştırmacıların probleme yaklaşım biçimleri olmaktadır. Sonu belli olan bir problemde sondan başa doğru yani tümünden gelim işlemi uygulanır. Devam etmekte olan bir faaliyetin bütün hesaplamalarını tekrarlamamak için baştan sona doğru yani tümevarım işlemini uygulamak yerinde olur (Çiçek, 2000: 39).

Bu çözüm yönteminde n-1 inci aşama ile ilgili bilgiler, n inci aşamanın karar girdilerini oluştururlar. Buna bağlı olarak çözüme 1 inci aşamadan başlanılarak 2, 3,..., n-1, n ve N inci aşamaya doğru gidileceğinden dönüşüm fonksiyonları da buna uygun olarak formüle edilecektir (Sevinç, 2008: 108).

Bu çözüm yönteminin genel formülü aşağıda verilmiştir (Yücel ve Ulutaş, 2010: 283):

Birinci safha için;

$$f_1(s_1) = \min (\max) [r_1(s_1, x_1)]$$

İkinci safha için;

$$f_2(s_2) = \min (\max) [r_2(s_2, x_2)] + f_1(s_1)$$

N inci safha için;

$$f_N(s_N) = \min (\max) [r_N(s_N, x_N)] + f_{N-1}(s_{N-1})$$

### 2.2.6.3. Tablosal Çözüm Yöntemi

Tüm dinamik programlama problemleri için her aşamayı gösteren aşağıdaki gibi bir tablo oluşturulabilir. Tablo son olarak ilk aşama için oluşturulduğunda (n=1), problem tamamen çözüme ulaşır. Çünkü ilk durum olarak bilinen, bu tabloda  $x_n^*$  ile gösterilen ilk karardır. Diğer karar değişkenlerinin optimal değeri bir önceki kararlardan elde edilen sonuçlara (durumlara) bağlı olarak oluşturulan tablolardan belirlenir (Hillier ve Lieberman, 2001: 541).

Tablosal çözüm yolunda her bir tablodaki satırlar, uygun durum değerlerini gösterirken sütunlar da mümkün karar seçeneklerini gösterecek biçimde düzenlenir. Tüm dinamik programlama problemlerinde her asama için aşağıdakine benzer bir tablo elde edilir (Sevinç, 2008: 111).

**Şekil 5:** Tablosal Çözüm Yöntemi

| $s_n$ \ $x_n$ | $f_n (s_n , x_n)$                       | $f_n^* (s_n)$ | $x_n^*$ |
|---------------|-----------------------------------------|---------------|---------|
|               |                                         |               |         |
|               | $\sum_{i=1}^s \{ C_n + f_{n+1}^*(i) \}$ |               |         |

Kaynak: Hillier ve Lieberman, 2001: 541.

$n$  = Mevcut asama ( $n = 1, 2, 3, \dots, N$ )

$s_n$  =  $n$  inci aşama için mevcut durum

$x_n$  =  $n$  inci aşama için karar değişkeni

$x_n^*$  =  $s_n$  durumunda  $x_n$  'in en iyi değeri

$f_n (s_n, x_n)$  =  $s_n$  durumundaki  $x_n$  kararı benimsendiğinde  $n$  aşamasında gerçekleşen kazanım

$$f_n^* (s_n) = \min_{x_n} [ f_n (s_n, x_n) ]$$

veya

$f_n^* (s_n) = \max [ f_n (s_n, x_n) ]$  biçiminde belirtilmektedir (Hillier ve Lieberman, 2001: 540).

Her aşama için benzer bir tablo düzenlenir. Çözümün ileriye veya geriye doğru en iyileme olmasına göre bir önceki aşamadaki tablo verileri bir sonraki aşamanın tablo çözümünde de kullanılacaktır. Son aşamaya ilişkin tablo çözümlendiğinde problemin tamamının çözümü de elde edilmiş olacaktır (Sevinç, 2008: 111).

Tablosal yöntemde her hangi bir süreçle ilgili tüm aşamalar için bütün durumlar göz önünde bulundurularak tüm seçenekler belirlenir her aşama ile ilgili

seenekler arasından en iyileri seilerek bir tabloya yerleřtirilmesi ve bu tablodan hareketle optimal politikanın belirlendiėi yntem olarak ifade edilir ( Patır, 2009: 69). Her ařamada dnřm fonksiyonları yardımı ile hesaplanan ilgili seenekler arasından en iyileri belirlenerek tabloya yerleřtirilir. Seenekler arasından seim yapılırken, seilen seeneėin uygun czm saėlayıp saėlamadıėı da czm sırasında gznnde bulundurulmalıdır. Uygun czm saėlamayan seenekler hesaplamalara dahil edilmez. Dolayısıyla, bu czm yolu ile elde edilen czmler de uygun czmler olmaktadır. Bu nedenle tablosal czm yolunun, yalnızca uygun seenekleri gznne alarak czm yapılmasına olanak saėladıėı sylenebilir (Sevin, 2008: 110).

#### **2.2.6.4. Analitik Czm Yntemi**

Analitik czm yolu, problem iin her ařamada oluřturulan dnřm fonksiyonlarının trevleri alınarak, her ařamada czm iin en iyi deėerin bulunmaya alıřıldıėı czm yoludur. Dnřm fonksiyonu her ařamada tek bir deėiřkene baėlı olarak en iyilenmeye alıřılmaktadır. Dinamik programlama ile bir problemin czmnde tablosal czmn kullanılabilmesi iin problemle ilgili parametrelerin sayısal deėerlerinin ve kısıtların aık olarak verilmesi gerekmektedir. Bunların verilmediėi durumlarda czm yolu olarak analitik czm yolu kullanılmaktadır (Sevin, 2008: 112).

### **2.3. DETERMİNİSTİK DİNAMİK PROGRAMLAMA**

Deterministik dinamik programlamada gelecek ařamadaki durumun ne olacaėı řu anki durum ve politika tarafından tamamen belirlenir. Diėer bir ifadeyle, ok ařamalı bir karar srecinde, karara etki eden tm dıřsal etmenler tam olarak bilinirse sre deterministiktir (Hillier ve Lieberman, 1990: 241).

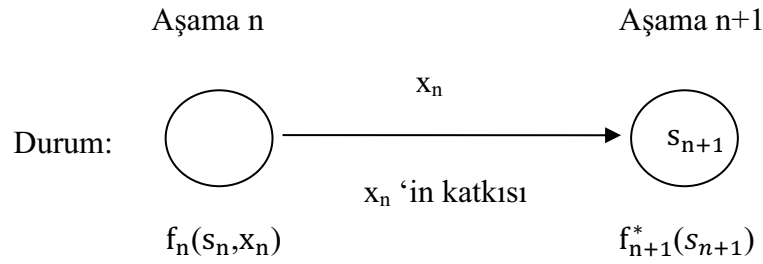
Deterministik dinamik programlama řematik olarak řekilde gsterildiėi gibi tanımlanabilir. Burada  $n$  ařamasındaki iřlem  $S_n$  durumunda oluřmaktadır. Politika kararı  $X_n$  olurken, sre  $(n+1)$ 'inci ařamada  $S_{n+1}$  durumuna gelir. Bylece optimal

politika çerçevesinde oluşan amaç fonksiyonu bir önceki durumda hesaplanmış  $f_{n+1}^*(s_{n+1})$  olacaktır. Politika kararı  $X_n$  de amaç fonksiyonuna katkı sağlar. Uygun bir yolla bu iki katkıyı birleştirmek  $f_n(s_n, x_n)$  verir,  $X_n$  için en iyi değeri bulmak;

$$f_n^*(s_n) = f_n(s_n, x_n^*) \text{ 'i verir.}$$

Her bir muhtemel  $s_n$  değeri için  $x_n$  ve  $f_n^*(s_n)$  ler bulunduktan sonra, çözüm prosedürü bir aşama geriye götürmeye hazır olmuş olur ((Hillier ve Lieberman, 2010: 431).

**Şekil 6:** Deterministik Dinamik Programlamanın Temel Yapısı



Kaynak: Hillier ve Lieberman, 2010: 432.

Optimizasyon problemlerinde seçilen değişkenler amaç fonksiyonunu minimize veya maksimize eden değerlere sahiptir. Fakat dinamik modellerde, bazı değişkenlerin nesnel politikaları bir varyasyon çerçevesine alınır. Dinamik programlamanın alt problemlerine uygulanacak optimizasyon teknikleri matematiksel ya da doğrusal programlama gibi basit karşılaştırmalar olabilir (Akdeniz ve Durmaz, 2003: 2).

Deterministik dinamik programlama farklı uygulama alanlarına sahiptir (Akdeniz ve Durmaz, 2003: 2):

- ✓ Yatırım problemleri,
- ✓ Yükleme problemleri,
- ✓ Üretim-Envanter problemleri,

- ✓ Dağıtım problemleri,
- ✓ Stoklama problemleri.

## **2.4. STOKASTİK DİNAMİK PROGRAMLAMA**

Dinamik programlama, risk altında karar durumlarına da uygulanabilir. Böyle durumlarda yineleme ilişkisinde rastlansal değişkenler bulunur. Bu rastlansal değişkenlere ilişkin olasılık dağılımlarının bilinmesi gerekmektedir (Tütek ve Gümüsoğlu, 2005: 368).

Stokastik dinamik programlamayı deterministik dinamik programlamadan ayıran nokta, şundaki durum ve kararlar bir sonraki durumun tam olarak belirlenememesidir. Burada bir sonraki durumun ne olacağı ile ilgili olarak bir olasılık dağılımı vardır. Bu olasılık dağılımı şu anki durum ve karardan tam olarak belirlenebilir (Cebeci, 2009: 18).

Stokastik dinamik programlama, rassal nitelik taşıyan çok aşamalı karar süreçlerinin optimizasyonu için geliştirilmiştir. Stokastik süreçler, değişken değerlerinin zaman fonksiyonuna göre veya aynı işlevi gören başka bir değişkene bağlı olarak rassal nitelikte değişmesi durumunda ortaya çıkan süreçler olarak tanımlanır (Cebeci, 2009: 18).

### **2.1.1. Stokastik Süreç ve Stokastik Programlama**

Bir rastgele değişkenin zaman içinde ardı ardına aldığı değerler birbirinden bağımsız değilse, bu değerlerden meydana gelen zaman serisine stokastik süreç denir. Bir stokastik süreci belirlemek için rastgele değişkenin olasılık dağılımından başka serinin ardışık elemanları arasındaki içsel bağımlılığı da belirtmek gerekir (İçağa vd., 2007:180).

Zaman içerisinde önceden kestirilemeyecek şekilde gelişen süreçlere stokastik (rastgelerassal) süreçler denir. Bu belirsizliğe olayların tutarsızlığından kaynaklanan ve kontrol edilemeyen değişimler neden olur. Bu değişiklikler kalitatif

olarak ele alınmaktansa, kurulacak bir matematiksel model içerisinde kantitatif olarak incelenebilir (Alp, 2007: 2).

Rastgele değişkenlerin bir  $\{X(t) : t \geq 0\}$  koleksiyonu stokastik süreç olarak adlandırılır. Stokastik süreç, verilen bir  $T$  kümesinden alınan  $t$  bir zaman parametresi olmak üzere  $\{X_t\}$  rasgele değişkenler kümesi ile tanımlanır (Alp, 2007: 2).

Burada  $t$  bilinen bir  $T$  kümesine ait zaman indisidir. Rastgele değişkenin aldığı her bir değere durum denir. Bu yüzden rastgele değişken  $X_t$  için durum değişkeni ifadesi de kullanılır. Rastgele değişkenin alabileceği değerlerin tanımlandığı  $S$  uzayı durum uzayı olarak adlandırılır. Durum uzayı  $S$  sürekli veya kesikli (tamsayı) değerlerden oluşabilir. Buna göre  $\{X_t\}$  süreci sürekli-durumlu stokastik süreç veya kesikli-durumlu stokastik süreç olarak adlandırılır. Aynı şekilde zaman kümesi  $T$  de sürekli veya kesikli olabilir.  $T$  sürekli değerler olabiliyorsa,  $\{X_t\}$  süreci sürekli-zamanlı stokastik süreç olarak ve eğer  $T$  tamsayı değerlerle sınırlanmış, yani  $T=\{0, 1, 2, \dots\}$  ise,  $\{X_t\}$  süreci kesikli-zamanlı stokastik süreç olarak adlandırılır (Alp, 2007: 2).

Belirsizlik içeren gerçek hayattaki optimizasyon problemlerinde, model parametrelerini doğru bir şekilde belirlemek oldukça zordur. Stokastik programlama bu tür, belirsizlik altında karar verme problemlerinin çözümü ile ilgilenir. Belirsizliğin varlığını göz önüne almak için problem parametreleri rassal veya stokastik değişkenler olarak düşünülür. Belirsizliğin etkisini korumak üzere genellikle orijinal stokastik programlama problemi, lineer olmayan deterministik eşdeğerine dönüştürülür. Daha sonra, lineer olmayan programlama problemleri için standart çözüm yöntemleri kullanılır. Tahmin edilemeyen ya da belirsiz problem parametrelerini kestirmek için belirsizliğin her kaynağı bir olasılık dağılımı ile gösterilmelidir (Akdemir ve Tiryaki, 2011: 157).

### 2.1.2. Stokastik Dinamik Programlama

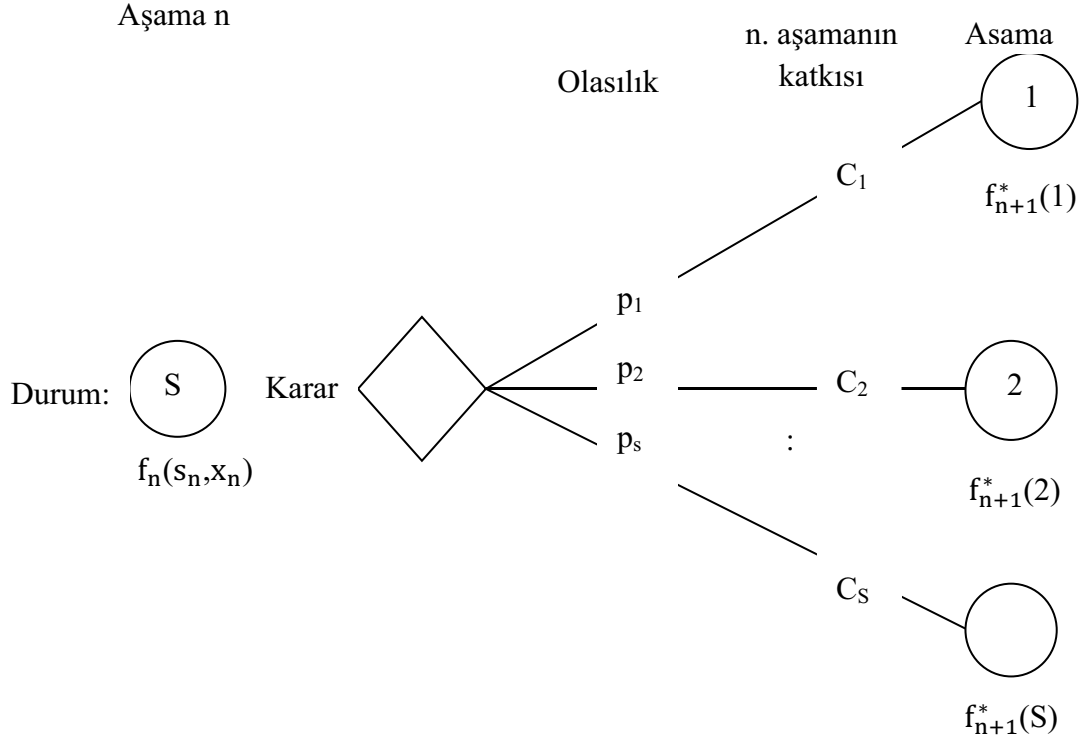
Belirsizlik altında dinamik programlamada, karar vericinin geçmişe ilişkin verileri yok ya da yok denecek kadar azdır. Diğer taraftan geçmişe ilişkin veriler var olmasına rağmen, karar verici bu verinin kullanılmasıyla önemli bir yanılmanın olacağı hususunda bir kaniya sahipse, yine belirsizlik durumu söz konusu olacaktır. Karar verici böyle durumlarda, ilgili olaylara ilişkin objektif olasılık dağılımlarını elde edemeyecektir. Olasılık dağılımlarının objektif olarak saptanamadığı koşullarda, belirsizlik durumları olduğundan belirsizlikle karşı karşıya kalınacaktır. Belirsizlik ortamında ilgili olaylara ilişkin subjektif olasılık dağılımlarının belirlenmesi söz konusudur. Subjektif olasılıklar, karar vericinin kendisinin ya da bilgisine güvendiği uzmanların değer yargıları, kişisel görüşleri, düşünce ve tecrübelerine dayanılır (Çiçek, 2000: 48).

Olasılıksal dinamik programlamada karar vericinin amacı, genellikle belirli bir büyüklükle ilgili beklenen değerin en büyük veya en küçük olmasını sağlayan değişken değerlerinin belirlenmesidir (Cebeci, 2009: 20).

Burada gelecek aşamadaki durum, şu anki durum ve karar tarafından tamamen belirlenemez. Daha doğrusu gelecek durumun ne olacağı ile ilgili olasılık dağılımı vardır. Bu olasılık dağılımı da şu anki durum ve politik karar tarafından tamamen belirlenir (Hillier ve Lieberman, 1990: 259).

Şekil 1 tüm aşamalarda ki olasılıklı durum ve kararları içermektedir. Bu şekil karar ağacı olarak da bilinmektedir. Eğer karar ağacı çok büyük değil ise, meydana gelecek çeşitli olasılıkların özetlenmesinde kullanışlı bir yol sağlar (Hillier ve Lieberman, 2001: 562).

**Şekil 7:** Olasılıklı Dinamik Programlamanın Temel Yapısı



Kaynak: Hillier ve Lieberman, 2001: 563.

Olasılıklı yapı nedeniyle  $f_n(s_n, x_n)$  ve  $f_{n+1}^*(s_{n+1})$  arasındaki ilişki deterministik dinamik programlamadan daha karmaşıktır. Bu ilişkinin kesinliği genel amaç fonksiyonunun şekline bağlı olacaktır. Açıklamak için, her aşamanın katkısının beklenen miktarının minimizasyonunun amacımız olduğunu farz edelim. Bu durumda  $f_n(s_n, x_n)$ , n. aşamadan ileriye doğru beklenen minimum miktarı temsil eder, n. aşamada ki durum ve politika kararı  $f_n$   $s_n$  ve  $x_n$  olarak verir. (Hillier ve Lieberman, 2001;562) Buna bağlı olarak eşitlikler aşağıdaki gibi olacaktır,

$$f_n(s_n, x_n) = \sum_{i=1}^S p_i \{ C_n + f_{n+1}^*(i) \}$$

$$f_{n+1}^*(i) = \min_{x_{n+1}} f_{n+1}(i, x_{n+1})$$

Burada minimizasyon  $x_{n+1}$  değerinden oluşacaktır.

Sonlu bir optimizasyon probleminin çözümü için Stokastik dinamik programlama yaklaşımı, şimdiki zaman periyodundan zaman serisinin sonuna dek sistemi çalıştırmanın optimal maliyetine karşılık gelen bir zaman periyodu için verilen değer fonksiyonunun tanımına dayanmaktadır. Bu değer fonksiyonu durum değişkenlerinin fonksiyonudur, örneğin durumlara dayanan sistemi işletmenin optimal maliyetidir ( Cervellera vd., 2006: 1146).

İşletmelerin zaman sürecinde sıralı kararlarını belirlerken birden çok amaca göre karar vermesinde, tek başına dinamik programlama veya çok amaçlı programlama modelleri yeterli olmamaktadır. Statik sürecin aksine, dinamik karar verme sürecinde çok amaçlı işletme sorunlarını çözmede çok amaçlı dinamik karar verme modelleri kullanılabilmektedir (Özdemir ve Gümüšoğlu, 2006: 360). Tezin son bölümünde çok amaçlı en kısa yol problemine ve çok amaçlı dinamik programlama yaklaşımına yer verilmiştir. Son olarak, karar sürecinde ikiden fazla amaca sahip firmanın ulaştırma sorunu için çok amaçlı dinamik programlama modeli geliştirilmiştir.

## ÜÇÜNCÜ BÖLÜM

### ÇOK AMAÇLI DİNAMİK PROGRAMLAMA MODELİNİN ULAŞTIRMA SORUNLARINA UYGULANMASI

Pek çok gerçek dünya problemleri sıralı ve çok aşamalı karar vermeyi gerektirir. Dinamik programlama özellikle bir dizi birbiriyle ilişkili kararları içeren karmaşık problemlere uygulanabilen güçlü bir optimizasyon tekniğidir. Geleneksel dinamik programlamaya dayanan çok amaçlı dinamik programlama metodolojisi, dinamik programlama karakteristiğine uyan çatışan amaçlara sahip problemleri çözmek üzere geliştirilmiş bir tekniktir (Abo-Sinna, 2004:861).

En kısa yol problemleri genellikle tek amaçlı optimizasyon teknikleri arasında yer alır. Ancak işletme sorunlarında karar verme sürecinde iki veya daha fazla amacın dikkate alınması söz konusu olduğunda çok amaçlı en kısa yol problemlerine ihtiyaç duyulur. Çok amaçlı dinamik programlama modeli ile çoklu amaca sahip bu problemlere çözüm sunulabilmektedir.

#### 3.1. ÇOK AMAÇLI DİNAMİK PROGRAMLAMA

Global rekabete uyum sağlamayı hedefleyen firmaların karar verme sürecinde karşılaştıkları sorunları çoklu amaç faktörünü dikkate alarak, dinamik bir süreçte çözmeleri gerekmektedir. Yönetmel karar verme sürecinde, çok amaçlı sorunlar, çok amaçlı sorunların çözümünde kullanılması gereken matematiksel modeller ve karar sonuçlarının kontrolü yer almaktadır (Özdemir ve Gümüšoğlu, 2006: 360).

Dinamik programlama 1950'lerin ortasında Bellman tarafından ortaya atılmasından bu yana pek çok alanda sıralı ve ya çok aşamalı karar vermeyi içeren karmaşık yapıda problemlerin çözümüne yönelik güçlü bir optimizasyon tekniği olmuştur (Lai ve Li, 1999: 293).

Karar verme dinamik bir süreç olup; bilgiye yönelik karmaşık bir arama, yinelemeler, tüm yönlerde geri beslemeler, bilgileri toplama ve eleme, değişen

oranda belirsizlik, karışıklık ve çelişkiler söz konusudur. Karar vermenin yapısı işlevsel olup, karara giden yolu türetme yeteneğine sahiptir. Son karara; bir öğrenme, anlama, bilgi işleme, değer biçme, problemi ve özelliklerini tanımlama sürecinden geçerek ulaşılır. Ağırlık karar verme eylemi ya da sonucu üzerinde değil, süreç üzerinde olmalıdır (Günyaşar, 1990: 15).

Çoklu amaçlar ve dinamik kontrol, gerçek dünyada pek çok sıralı karar verme problemlerini nitelendirir ve bu problemlerin çözümü birbiriyle çatışan ve rekabet içinde olan amaçların optimizasyonunu gerektirir. Genel olarak, çok amaçlı dinamik optimizasyon, çoklu amaçları olan dinamik sistem problemlerinin optimizasyonu anlamına gelir (Abo-Sinna, 2004: 862).

Dinamik programlamada ağ sorunu tek bir amaç için planlanmaktadır. Yani; tek bir minimum amaç fonksiyonuna göre ağ planlaması yapılmaktadır. Oysa gelişen dünya koşullarında karar vericiler bir çok amacı aynı plan çerçevesinde dikkate almak zorundadır. Çok amaçlı ağ yapısında gösterilen plan, çok amaçlı en kısa yol problemi olarak ifade edilmektedir (Özdemir, 2004: 97).

### **3.2. LİTERATÜR TARAMASI VE UYGULAMA ALANLARI**

Çok amaçlı dinamik programlama problemleri için dinamik programlama temeline dayandırılmış bazı çalışmalar yapılmıştır. Caplin ve Kornbluth (1975) çok amaçlı dinamik programlama yatırım problemine dinamik programlama uygulamıştır. Tauxe vd. (1979) su kaynakları yönetimi problemine çok amaçlı dinamik programlama uygulamışlardır. DCruz vd. (1983) ve Rosenman ve Gero (1983) bazı mühendislik problemlerine çok amaçlı dinamik programlama uygulamışlardır. Sancho (1986) ve Sneidovich (1988) rotamala problemleri için çok amaçlı dinamik programlamaya başvurmuşlardır (Abo-Sinna, 2004: 862).

Gerçek hayatta karar problemleri sıklıkla çatışan amaçlar ve birden fazla aşamalardan meydana gelen özelliktedir. Bu gibi durumların modellenmesinin bir yolu da çok amaçlı yaklaşımdır. Bu alanda pek çok araştırma yayınlanmıştır. Çok

amaçlı dinamik programlama araştırmaların incelemesi Li ve Haimes (1989) ve Trzaskalik (1997,1998) tarafından sunulmuştur. Trzaskalik ve Sitarz'ın 2007'da, Özdemir ve Gümüšoğlu'nun 2006'da ve Tarun'un 2008'de yapmış oldukları çalışmalardan derlenen uygulamalar şu şekilde özetlenebilir;

- Klotzer (1978) kesikli dinamik sistemler ve vektör değerli amaç fonksiyonları için optimal değerler kümesinin dinamik programlama tekrarlama hesapları üzerine dayanan bir yöntem geliştirmiştir.
- Yu ve Seiford (1981) dinamik çok aşamalı sistemlerde vektör optimizasyon problemlerini ele almışlardır.
- Etki analizi Gomide ve Haimes (1984) tarafından sunulmuştur.
- Çok amaçlı en kısa yol problemleri Heing (1985), Warburton (1987) ve Getachew (1992) tarafından ele alınmıştır .
- Çok amaçlı dinamik programlamada süreye bağlılık Kostreva ve Wiecek (1993) ve Kostreva ve Lancaster (2002) tarafından sunulmuştur.
- Çok amaçlı dinamik programlama problemlerini skalar amaçlı optimizasyon tekniklerine dönüştürmek için çeşitli skalarizasyon ve çözüm üretme teknikleri kullanılmaktadır. Ağırlıklandırılmış toplam yaklaşımı ve  $\epsilon$ - kısıt metodu Tauxe (1979)' in çalışmasında bulunabilir.
- Etkileşimli ikame yaklaşımına Hu ve Wu (1989) başvurmuştur.
- Ağırlıklandırma kuralı kullanılarak parametrik alanın ayrıştırılması için Abo-Sinna ve Hussein (1994) bir algoritma geliştirmişlerdir.
- Wierzbicki (1979) referans yollarına dayanan çok amaçlı yol optimizasyonu geliştirmişlerdir.
- Trzaskalik (1997) ve Caballero vd. (1998) tek ve çok dönemli hedef değerleri uygulamaları ile dinamik amaç programlama modelleri ve lexicographic hedef programlama modelleri ortaya koymuşlardır.
- Lexicographic çok amaçlı dinamik programlama Szidarovszky ve Duckstein (1986) tarafından tanımlanmıştır .
- Opricovic (1993) uzlaşmalı çok amaçlı dinamik programlama metodunu geliştirmiştir.

- Çok amaçlı dinamik programlama Bellman (1957)' ın geliştirdiği tek kriterli dinamik programlamanın genellemesi olarak kabul edilebilir. Genellemenin diğer yolu dikkate alınan getirileri kısmi sıralı kriterler setine sokmaktır. Mitten (1974) çok aşamalı karar problemlerinin bir çeşidini çözmek için gerçek değerli amaç fonksiyonunun önem derecelerine göre tercih ilişkileri ile gösterildiği bir yöntem tanımlamıştır.
- Sobel (1975) Mitten'in sonuçlarını, deterministik ve stokastik problemler için sonsuz-aralık problemlerine doğru geliştirmiştir.
- Steinberg ve Parks (1979) tercih sıralı dinamik programlama kavramını tanımlamışlardır .
- Henig (1985) getirilerin kısmi sıralı kümede olduğu genel bir ardışık model tanımlamıştır .
- Trzaskalik ve Sitarz (2002) kısmi sıralı karar seti ile kesikli dinamik programlamayı ele almıştır. Optimallik prensibi ve geriye doğru süreç türetilmiştir .
- Daellenbach ve Kluyver (1980), doğrusal, doğrusal olmayan, deterministik ve stokastik çok kriterli karar problemleri gibi çok çeşitli karar problemlerinin ele alınabildiği, çok kriterli karar problemlerine minitoplam ve minimaks çözümler bulmaya yönelik “Çok Amaçlı Dinamik Programlama” olarak adlandırılan bir teknik ortaya koymaktadırlar.
- Gorelik (1991), çok kriterli dinamik problemi çözmede kullanılan parametrik ölçekleme fonksiyonları üzerinde durmaktadır. Çok kriterli dinamik problemlerin kesikli zaman sürecinde ve sürekli zaman sürecinde formülasyonunu göstermektedir.
- Abo-Sinna, Hussein (1995), çok amaçlı matematiksel programlama problemlerinin etkin çözümünü bulmak için algoritma geliştirmişlerdir. Bu çalışmada etkin çözümün bulunması dinamik programlamadaki optimalite prensibine dayanmaktadır .
- Klotzer (1978), Bellman'ın ortaya attığı dinamik programlamaya ayrıştırılabilirlik ve tekdüzelik koşulu ve yineleme bağıntılarını ekleyerek vektör ağırlıklı bir amaç fonksiyonu olarak geliştirmiştir. Yineleme seti

bağıntıları olarak anılan değiştirilmiş yineleme bağıntıları Neumann-Morgenstern özelliğine uymayı gerektirmektedir. Bu çalışma çok amaçlı kesikli dinamik programlamaya iyi bir teorik hesaplama sunmaktadır .

- Villarreal ve Karwan (1982), temel dinamik programlama yenileme ilişkilerini çok kriterli çerçeveye taşımışlardır. İkili çoklu kriterli sırt çantası problemlerinin hesaplama sonuçlarını sunmuşlardır. Dezavantajları sınırlı uygulanabilirlik ve etkin olmayışdır .
- Li ve Haimes, çok amaçlı dinamik programlamanın altındaki teorik yapıyı ve bu fikrin ortaya çıkışından itibaren teori ve metodolojisinin evrimini incelemişlerdir. Li (1990), stokastik çok amaçlı dinamik programlama temeline dayanan çözüm üretme yaklaşımının kullanımında Pareto optimal çözüm setini bulmuştur .

### **3.3. ÇOK AMAÇLI EN KISA YOL PROBLEMLERİ**

Bir ağ yapısındaki yolların planlanması pek çok işletme uygulamalarında önem taşımaktadır. Yol problemleri genellikle ulaştırma, telekomünikasyon ve bilgisayar tasarımı konularına uygulanmaktadır. Çok amaçlı en kısa yol problemleri ilk olarak Bellman (1957) tarafından çözülmüştür. Yoldaki her bir aktivite zamanının sabit olduğu varsayımı altında, başlangıç durumundan en son duruma minimum zamanda ulaşılan yolun bulunmasında, dinamik programlama modeli kullanılmaktadır (Kostreva ve Wiecek, 1993:289). En kısa yol problemleri esasen tek amaçlı optimizasyon çerçevesinde yer alır. Daha spesifik olarak her bir yay ile ilgili olarak bazı değerler (örneğin uzunluk ya da ulaşım zamanı) kabul edilir ve toplam mesafe ya da toplam varış zamanını minimize eden uygun yolu belirleyecek bir hedef vardır. Pek çok gerçek uygulamada genellikle tek bir amaç bulunmaması bu problemlerin yetersiz kalmasına neden olmaktadır. Bu gibi durumlarda, çok amaçlı en kısa yol problemleri kullanılır (Tarapata, 2007: 269).

Çok amaçlı en kısa yol problemleri genellikle çatışan iki veya daha fazla amaca ilişkin etkin yollar setinin bulunmasını sağlayan geleneksel en kısa yol problemlerinin geliştirilmiş şeklidir. Örneğin iletişim ağlarındaki optimal yolların

bulunması problemi eş zamanlı olarak seyahat maliyetini, yolların uzunluğunu ve seyahat zamanını minimize eden ulaşım planlamasındaki etkin yollar bulunurken ya da verimlilik maksimize edilirken gecikmelerin minimize edilmesini sağlar. Çok amaçlı en kısa yol problemlerinde ki optimizasyon kavramı ya da çok amaçlı problemlerin tek amaçlı optimizasyon problemlerinden genellikle farkı tek amaçlı optimizasyon problemlerinde görev yönünden tek bir amaç fonksiyonunu optimize eden bir çözüm bulunmasıdır. Çok amaçlı problemlerde her bir amaç fonksiyonu için optimal bir çözüm bulunmaz fakat aynı anda tüm amaçları optimize eden optimum bir çözüm bulunur. Çoğu durumda tek bir çözüm yoktur, etkin yada baskın çözümlerin bir seti vardır (Pangilinan ve Janssens, 2007: 322).

Çok amaçlı en kısa yol problemlerinde bazı algoritma ve metodlar incelenmekte ve uygulanmaktadır. Dinamik programlama yöntemi de bu algoritmalarından biridir.

### 3.4. ÇOK AMAÇLI DİNAMİK PROGRAMLAMA FORMÜLASYONU

Dinamik programlama sıralı kararlara ilişkin fonksiyonları çözmektedir. Çok amaçlı dinamik programlama, dinamik programlamayı vektör değerli kriter fonksiyonlarına dönüştürmektedir. Vektör dinamik programlama optimizasyonu ile kriter uzayında etkin çözümlerin kümesi ve pareto-optimal vektörleri bulunabilmektedir (Özdemir, 2004: 114).

Aşağıdaki vektör minimizasyon problemini (VMP) düşünelim (Abo-Sinna, 2004:863):

$$\begin{aligned} \text{Minimum} \quad & F_{\ell} (f_{\ell 1} (x_1), \dots, f_{\ell N} (x_N)), \quad \ell = 1, 2, \dots, L, \quad (L \geq 2) \\ \text{Kısıtlar} \quad & G_m (g_{m1} (x_1), \dots, g_{mN} (x_N)) \leq 0, \quad m = 1, 2, \dots, M, \\ & x_n \in X_n, \quad n = 1, 2, \dots, N \end{aligned}$$

Burada amaç fonksiyonları  $F_{\ell}$ ,  $\ell = 1, 2, \dots, L$  ve kısıt fonksiyonları  $G_m$ ,  $m = 1, 2, \dots, M$  ve  $R^N$  gerçekte değerli fonksiyonlardır.

Dinamik programlamanın yinelemeli formülünün türetilmesi için vektör optimizasyonu problemlerinde kullanılan ayrıştırılabilirlik ve tekdüzelik kavramlarının açıklamaları aşağıda verilmektedir.

Eğer  $F_\ell^n$ ,  $n=1, \dots, N$  fonksiyonları mevcut,  $R^n$  ve  $\emptyset_\ell^n$ ,  $n=2, \dots, N$  fonksiyonlarında tanımlı ve  $n=2, \dots, N$  için  $R^n$  sağlanıyorsa  $F_\ell$  amaç fonksiyonunun ayrıştırılabilir olduğu söylenir (Abo-Sinna, 2004:863).

$$F_\ell^n((f_{\ell 1}(x_1), \dots, f_{\ell n}(x_n))) = \emptyset_\ell^n[F_\ell^{n-1}((f_{\ell 1}(x_1), \dots, f_{\ell n}(x_{n-1})), f_{\ell n}(x_n))] \text{ ve}$$

$$F_\ell^N((f_{\ell 1}(x_1), \dots, f_{\ell N}(x_N))) = F_\ell(f_{\ell 1}(x_1), \dots, f_{\ell N}(x_N))$$

Benzer olarak eğer  $G_m^n$ ,  $n=1, \dots, N$  fonksiyonları mevcut,  $R^n$  ve  $\psi_m^n$ ,  $n=2, \dots, N$  fonksiyonlarında tanımlı ve  $n=2, \dots, N$  için  $R^n$  sağlanıyorsa  $G_m$  kısıt fonksiyonunun ayrıştırılabilir olduğu söylenir (Abo-Sinna, 2004:863).

$$G_m^n((g_{m1}(x_1), \dots, g_{mn}(x_n))) = \psi_m^n[G_m^{n-1}((g_{m1}(x_1), \dots, g_{mn-1}(x_{n-1})), g_{mn}(x_n))] \text{ ve}$$

ve

$$G_m^N((g_{m1}(x_1), \dots, g_{mN}(x_N))) = G_m(g_{m1}(x_1), \dots, g_{mN}(x_M))$$

Eğer tüm amaç ve kısıt fonksiyonları ayrıştırılabilir ise vektör minimizasyon probleminin ayrıştırılabilir olduğu söyleyebiliriz. Ayrıca eğer tüm  $\emptyset_\ell^n$  ve  $\psi_m^n$  fonksiyonları ilk çıkarım bakımından her sabit ikinci çıkarım için kesinlikle artıyorsa vektör minimizasyon probleminin ayrıştırılmasının tekdüze olduğu söylenebilir. Özellikle her  $y \in R$  için,

$$\emptyset_\ell^n(s, y) > \emptyset_\ell^n(s', y) \quad s > s' \text{ ise} \quad \text{ve} \quad \emptyset_m^n(s, y) > \emptyset_m^n(s', y) \quad s > s' \text{ ise}$$

her  $\ell = 1, \dots, L$ ,  $m = 1, \dots, M$  ve  $n = 2, \dots, N$  için.

Herhangi bir karışıklık ortaya çıkmadığı durumda formül (2) aşağıdaki gibi yazılabilir(Abo-Sinna, 2004:864):

$$F = f_1 \bullet f_2 \bullet \dots \bullet f_n$$

Vektör minimizasyonu probleminin ayrıştırılabilir olduğunu varsayarsak her  $n=1, \dots, N$  ve her  $s = (s_1, \dots, s_2) \in R^m$  için bir  $n$ . alt problem  $P_n(s)$  tanımlanır:

$$P_n(s) : \text{minimum} \quad F_\ell^n((f_{\ell 1}(x_1), \dots, f_{\ell n}(x_n)), \quad \ell = 1, 2, \dots, L,$$

$$\text{Kısıtlar} \quad G_m^n((g_{m1}(x_1), \dots, g_{mn}(x_n)) \leq s_1, \quad m = 1, 2, \dots, M,$$

$$x_1 \in X_1, \dots, x_n \in X_n.$$

$n = N$  ve  $s = o$  iken yukarıdaki problemin vektör minimizasyonu problemi ile örtüştüğü açıktır. Her  $s \in R^m$  için  $n$ . alt problem  $P_n(s)$  bir çok amaçlı programlama probleminde tekrarlanır.

$(x_1^0, \dots, x_n^0)$ ' in  $P_n(s)$  problemi için uygun bir çözüm olduğunu düşünelim. Eğer hiçbir uygun çözüm  $(x_1, \dots, x_n)$  yok ise buna etkin denilir(Abo-Sinna, 2004:864).

$$F_\ell^n((f_{\ell 1}(x_1), \dots, f_{\ell n}(x_n)) \leq F_\ell^n((f_{\ell 1}(x_1^0), \dots, f_{\ell n}(x_n^0))) \text{ tüm } \ell \text{ için ve}$$

$$F_k^n((f_{k1}(x_1), \dots, f_{kn}(x_n)) \leq F_k^n((f_{k1}(x_1^0), \dots, f_{kn}(x_n^0))) \text{ en az bir dizi için } k \in \{1, \dots, L\}.$$

### 3.4.1. Ölçekleme Yaklaşımları

Ölçekleme yaklaşımında çok amaçlı dinamik problem, tek amaçlı dinamik programlama problemine dönüştürülebilmektedir. Karar verici ölçekleme yöntemlerini uygulayarak etkin çözümlerin alt kümesini tanımlamakta ve bu çözümün elemanları arasından seçim yapmaktadır. Bu metotların en önemli yararı karar vericinin sistem hakkında tam bilgiye ve daha iyi bir anlayışa sahip olmasıdır. En genel anlamda yaklaşımları; 1. Ağırlıklandırılmış yaklaşım, 2. Kısıt yaklaşımı ve

3. Amaç programlama yaklaşımı olarak üç farklı biçimde ele alınabilmektedir (Özdemir, 2004: 126).

#### 3.4.1.1. Ağırlıklandırılmış Yaklaşım

Bu yaklaşımda problem (Vektör minimizasyon problemi) sayısal amaçlarla birlikte aşağıdaki doğrusal olmayan programlama ile çözülecektir.

$$\begin{aligned}
 (P\lambda): \text{minimum} \quad & \sum_{\ell=1}^L \lambda_{\ell} F_{\ell}((f_{\ell 1}(x_1), \dots, f_{\ell N}(x_N))) \\
 \text{Kısıtlar} \quad & G_m(g_{m1}(x_1), \dots, g_{mN}(x_N)) \leq 0, m=1, \dots, M, \\
 & x_1 \in X_1, \dots, x_n \in X_n, n=1, \dots, N, \\
 & \lambda_{\ell} \geq 0
 \end{aligned}$$

Amaç vektörün her bir bileşenin ağırlıklı toplamının bulunduğu yaklaşım olarak bilinmektedir.  $\lambda_{\ell}$ ,  $\ell$ . amacın ağırlığıdır (Özdemir, 2004:126).

#### 3.4.1.2. Kısıt Yaklaşımı

Etkin bir dizide kesikli noktaların belirlenmesi için kullanılan yaygın metodolojilerden biri kısıt yaklaşımıdır. Bu yaklaşım çok amaçlı dinamik programlama problemlerinin gelişimi için bir temel oluşturmaktadır. Kısıt yaklaşımında bir amaç temel amaç olarak ele alınır ve geriye kalan  $\ell-1$  amaç kısıt olarak ele alınır. Kısıt yaklaşımın formülasyonu aşağıda verilmiştir (Abo-Sinna, 2004:867):

$$\begin{aligned}
 P_1(\varepsilon) : \text{minimum} \quad & F_1((f_{11}(x_1), \dots, f_{1N}(x_n))) \\
 \text{Kısıtlar} \quad & F_{\ell}((f_{\ell 1}(x_1), \dots, f_{\ell N}(x_N))) \leq \varepsilon_{\ell}, \quad \ell = 1, 2, \dots, L, \\
 & G_m((g_{m1}(x_1), \dots, g_{mN}(x_N))) \leq 0, \quad m= 1, 2, \dots, M, \\
 & x_n \in X_n, n=1, \dots, N,
 \end{aligned}$$

burada  $\varepsilon = (\varepsilon_2, \dots, \varepsilon_L)$ ,  $P_I(\varepsilon)$ 'nin olurlu olması koşulu ile  $(L-1)$  kısıt vektörünü göstermektedir.

Problemin dinamik programlama ile gösterimi

$$B_1^n(\varepsilon, s) = \min \{F_1^n(f_{11}(x_1), \dots, f_{1n}(x_n)) | F_\ell^n(f_{\ell 1}(x_1), \dots, f_{\ell n}(x_n)) \leq \varepsilon_\ell, \ell = 2, \dots, L; \\ G_m^n(g_{m1}(x_1), \dots, g_{mn}(x_n)) \leq s_m, m = 1, 2, \dots, M;$$

$$x_1 \in X_1, x_n \in X_n\}$$

şeklindedir ve formülasyonun sıralı biçimi aşağıdaki gibidir(Abo-Sinna, 2004:868):

$$B_1^n(\varepsilon_2^n, \dots, \varepsilon_\ell^n, s_1^n, \dots, s_m^n) =$$

$$\min_{x_n \in X_n} \phi_1^n \{ [f_{1n}(x_n)], B_1^{n-1}[\varepsilon_\ell^{n-1}(f_{\ell n}(x_n)), s_m^{n-1}(s_m^n, g_{mn}(x_n))] \}$$

$$\text{Kısıtlar} \quad f_{\ell n}(x_n) \leq \varepsilon_\ell^n, \quad \ell = 2, \dots, L,$$

$$g_{mn}(x_n) \leq s_m^n, \quad m = 1, \dots, M.$$

### 3.4.1.3. Amaç Programlama Yaklaşımı

Problemin ilgili kriterleri için önceden atanmış amaç düzeyleri ve hedef değerlerinin düzenlenmesi ile hedef programlama formülasyonu aşağıdaki gibidir:

$$\text{Minimum} \quad Z_t = \sum_{j=1}^J P_j \sum_{t=1}^T r_t(D_t, x_t, P_j),$$

$$\text{Kısıtlar} \quad g_{it}(\mu_t) + d_{it}^- - d_{it}^+ = G_{it}, \quad i=1, 2, \dots, k, t=1, 2, \dots, T,$$

$$x_{t+1} = S_t(x_t, \mu_t),$$

$$d_{it}^- \cdot d_{it}^+ = 0, \quad i=1, 2, \dots, k, t=1, 2, \dots, T,$$

$$d_{it}^-, d_{it}^+ \geq 0,$$

burada,  $x_t$  durum,  $\mu_t$  kontrol,  $G_{it}$  t. aşamada i. hedef,  $d_{it}^-$  hedeflenenin altında kalan değer,  $d_{it}^+$  hedefin üzerine çıkan değer,  $D_t$  t. aşamada istisnai değişkenlerin vektörü,  $P_j$  ise sonsuz öncelikli seviyedir. Her öncelik seviyesi için sayısal amaçlı dinamik programlama uygulanabilir. Böylece, hedef programlama ve sayısal amaçlı dinamik programlamayı bir araya getiren bir algoritma sunulabilir.  $Q_t(D_t, x_t, P_j)$  t. aşamadaki amaç programlama fonsiyonumuz olsun;

$$r_t(x_t, P_j) = \min Q_t(D_t, x_t, P_j), \quad t=1,2,\dots,T, \quad j=1,2,\dots,J$$

Burada  $r_t(x_t, P_j)$  hedef programlama fonsiyonunun t. aşamadaki optimal getiri değeridir (Abo-Sinna, 2004: 869).

Böylece dinamik amaç programlama modelinin sıralı şekli

$$Q_t(D_t, x_t, P_j) = \begin{cases} Z_1(D_1, x_1, P_1), & t=1, j=1,2,\dots,J, \\ Z_t(D_t, x_t, P_j * r_{t-1}(x_{t-1}, P_j), & t=2,3,\dots,T, j=1,2,\dots,J \end{cases}$$

ifadesi ile gösterilir(Abo-Sinna, 2004: 869).

### 3.4. UYGULAMA

Çok amaçlı dinamik programlama, çok amaçlı doğrusal programlama gibi mevcut çok amaçlı yaklaşımların uygun olmadığı, özellikle de kesikli değişkenleri içeren problemleri çözümde ulaştırmada kullanılabilir. Çok amaçlı dinamik programlama tekniği mevcut dinamik programlama kavramlarının nispeten genişletilmiş halidir. (Daellebbach ve Kluyver, 1980: 591). Günümüz rekabet ortamı gereği hızla gelişen koşullara ayak uydurmak için çoğu zaman çoklu amaçlara sahip olan işletmeler için çözüme ulaşmada çok daha uygun bir tekniktir.

Çalışmanın bu kısmında ikiden fazla amaca sahip ulaştırma sorununda çok amaçlı dinamik programlama modelinin kurulacağı firmaya ilişkin bilgiler verilecek ardından firmanın mevcut ulaştırma sorunu ortaya konulacaktır. Soruna amaçlı

dinamik programlama modeli ile çözüm sunulacak ve karar vericiye çoklu amaçları doğrultusunda çözüm alternatifleri sunulacaktır.

#### **3.4.1. Firma Bilgileri**

Çalışmayı gerçekleştirmiş olduğumuz firma ana merkezi Japonya olan 10 ülkede üretim ve 54 ülkede satış merkezleri bulunan uluslararası bir şirketin Türkiye şubesidir. Metalden prefabrik yapıların, köprülerin, köprü parçalarının, otomobil üretiminde kullanılan fabrikasyon metal parçaların ülke içinde satış ve dağıtımını İzmir merkezli olarak sağlamaktadır.

2012 yılında faaliyete geçen firma yurtdışından ithal ettiği ürünlerin İzmir ve Kocaeli Limanları'ndan ülkeye girişini sağlamaktadır. İzmir ve Kocaeli'de mevcut depolarında tuttıkları ürünler talebe göre ülkenin çeşitli illerindeki müşterilere, şirket bünyesinde bulunan tırlar ile nakliye edilmektedir.

#### **3.4.2. Problemin Tanımlanması**

Firma, gelen talebe göre İzmir ve Kocaeli illerinde iki ayrı depoda bulunan ürünlerinden ülke içinde çeşitli illerde bulunun müşterilerine gönderim yapmak istemektedir. Ancak gelen talebi karşılayacak ürünlerin her iki depoda da mevcut olması firmayı etkin bir karar verme konusunda zorlamaktadır.

Firma, müşteri tarafından talep edilen ürünün hangi ildeki depodan gönderiminin yapılacağına etkin bir şekilde karar vermelidir. Bu nedenle gönderimi gerçekleştirirken en az maliyet ile ulaştırma zamanının minimumda tutmayı ve yol güvenliğini maksimum seviyede tutmayı amaçlamaktadır.

### 3.4.3. Problemin Notasyonu ve Varsayımları

Karar verici İzmir veya Kocaeli depolarından çıkıp varış yeri Malatya olacak ürün gönderimi için en az maliyete, en az ulaştırma zamanına ve maksimum yol güvenliğine sahip güzergahı belirlemelidir. Dolayısıyla karar vermek için 3 amacı vardır. Bu amaçlar; ulaştırma maliyetinin ve ulaştırma zamanının minimumda, seçilen yol güvenliğinin ise maksimum seviyede tutulmasıdır.

Bu problemin çok amaçlı dinamik programlama ile geriye doğru çözümünde aşağıdaki notasyon kullanılmaktadır:

$n$  = aşama,  $n= 1,2,\dots, N$

$i_n$  =  $n$ . aşamada içinde bulunulan durum

$j_n$  =  $n$ . aşamadan ulaşılabilecek durumlar veya kararlar seti

$C_{ij}$  =  $i$ 'den  $j$ 'ye yapılan hareketin amaç fonksiyonundaki değeri

$W_k$  =  $k$ . amacın ağırlığı

Soruna uygun model geliştirilirken aşağıdaki varsayımlar göz önünde tutulmuştur;

✓ Talep edilen ürün her iki depoda da mevcuttur. Dolayısıyla firma nakliye için amacına uygun olarak İzmir veya Kocaeli depolarından herhangi biri seçilebilir.

✓ Seçilen güzergahlarda her ilden her ile yol bulunmamaktadır. Örneğin Kocaeli'den yola çıkan tır Denizli'ye gidememektedir. Tırın nakliye sırasında kullanabileceği güzergahlar şu şekildedir;

- Kocaeli depodan çıkıp Eskişehir veya Afyon'a gidebilir,
- İzmir depodan çıkıp Eskişehir, Afyon veya Denizli'ye gidebilir,
- Eskişehir'den Ankara veya Konya'ya gidebilir,
- Afyon'dan Ankara veya Konya'ya gidebilir,
- Denizli'den Ankara veya Konya'ya gidebilir,

- Ankara'dan Kayseri veya Adana'ya gidebilir,
- Konya'dan Kayseri veya Adana'ya gidebilir,
- Kayseri ve Adana'dan ise varış noktası olan Malatya'ya gidebilir.

✓ Belirlenen güzergahta iller arasındaki ulaşım zamanı ile ulaşım maliyetleri arasında doğrusal bir ilişki yoktur.

Kocaeli veya İzmir'den yola çıkması öngörülen ürünleri taşıyacak tırın Malatya'ya ulaşması için izlenecek muhtemel yollar şeklindeki haritada işaretlenmiştir. Ulaşım zamanları, haritada işaretlenen güzergahları takip edecek tırın iller arasındaki nakliye için geçen süreler olarak belirlenmiştir.

**Şekil 8:** Türkiye Karayolları Haritası'nda Firmanın Muhtemel Nakliye Yollarının Gösterimi



— Muhtemel güzergahı

✓ Taşınan ürün tipi ve ağırlığı önemsenmeyerek, yük taşıyan tırın şehirler arasında katlandığı ulaştırma maliyetinin unsurları aşağıda belirtilmiştir;

- Yakıt gideri,
- Araç sigorta gideri,
- Tır kirası,
- Personel (Tır Şoförü) ücreti,

- Tünel vb. geçiş ücretleri,
- Harcırak.

✓ Güzergahta seçili şehirler arasında son bir yıl içinde tır ile yapılan gönderimlerin sayısı ve gönderimlerin toplam maliyeti aşağıdaki tabloda özetlenmiştir. Maliyeti oluşturan gider kalemleri ile ilgili ayrıntılı hesaplamalar firma gizliliği bakımından bu çalışmada sunulmamaktadır.

Toplam maliyetlerin gönderim sayısına bölünmesi ile problemin çözümünde dikkate alınacak nakliye başına maliyetler TL cinsinden hesaplanmıştır. Nakliye 4 aşamadan oluşmaktadır. Aşağıdaki tablolarda dört aşamaya için ortalama maliyet hesaplaması mevcuttur ;

**Tablo 3:** Şehirlerarası Gerçekleştirilen Nakliye Başına Maliyetin Hesaplanması (ilk aşama için)

|                | Eskişehir    |                |                  | Afyon        |                |                  | Denizli      |                |                  |
|----------------|--------------|----------------|------------------|--------------|----------------|------------------|--------------|----------------|------------------|
|                | Sefer Sayısı | Toplam Maliyet | Ortalama Maliyet | Sefer Sayısı | Toplam Maliyet | Ortalama Maliyet | Sefer Sayısı | Toplam Maliyet | Ortalama Maliyet |
| <b>Kocaeli</b> | 20           | 3.400          | <b>170</b>       | 12           | 3.720          | <b>310</b>       | -            | -              | -                |
| <b>İzmir</b>   | 11           | 3.630          | <b>330</b>       | 32           | 7.488          | <b>234</b>       | 21           | 3.675          | <b>175</b>       |

**Tablo 4:** Şehirlerarası Gerçekleştirilen Nakliye Başına Maliyetin Hesaplanması (ikinci aşama için)

|                  | Ankara       |                |                  | Konya        |                |                  |
|------------------|--------------|----------------|------------------|--------------|----------------|------------------|
|                  | Sefer Sayısı | Toplam Maliyet | Ortalama Maliyet | Sefer Sayısı | Toplam Maliyet | Ortalama Maliyet |
| <b>Eskişehir</b> | 22           | 3.960          | <b>180</b>       | 8            | 2.160          | <b>270</b>       |
| <b>Afyon</b>     | 13           | 2.665          | <b>205</b>       | 23           | 3.450          | <b>150</b>       |
| <b>Denizli</b>   | 11           | 3.960          | <b>360</b>       | 22           | 6.710          | <b>305</b>       |

**Tablo 5:** Şehirlerarası Gerçekleştirilen Nakliye Başına Maliyetin Hesaplanması (üçüncü aşama için)

|               | <b>Kayseri</b> |                |                  | <b>Adana</b> |                |                  |
|---------------|----------------|----------------|------------------|--------------|----------------|------------------|
|               | Sefer Sayısı   | Toplam Maliyet | Ortalama Maliyet | Sefer Sayısı | Toplam Maliyet | Ortalama Maliyet |
| <b>Ankara</b> | 23             | 5.704          | <b>248</b>       | 14           | 5.208          | <b>372</b>       |
| <b>Konya</b>  | 21             | 4.305          | <b>205</b>       | 20           | 5.560          | <b>278</b>       |

**Tablo 6:** Şehirlerarası Gerçekleştirilen Nakliye Başına Maliyetin Hesaplanması (dördüncü aşama için)

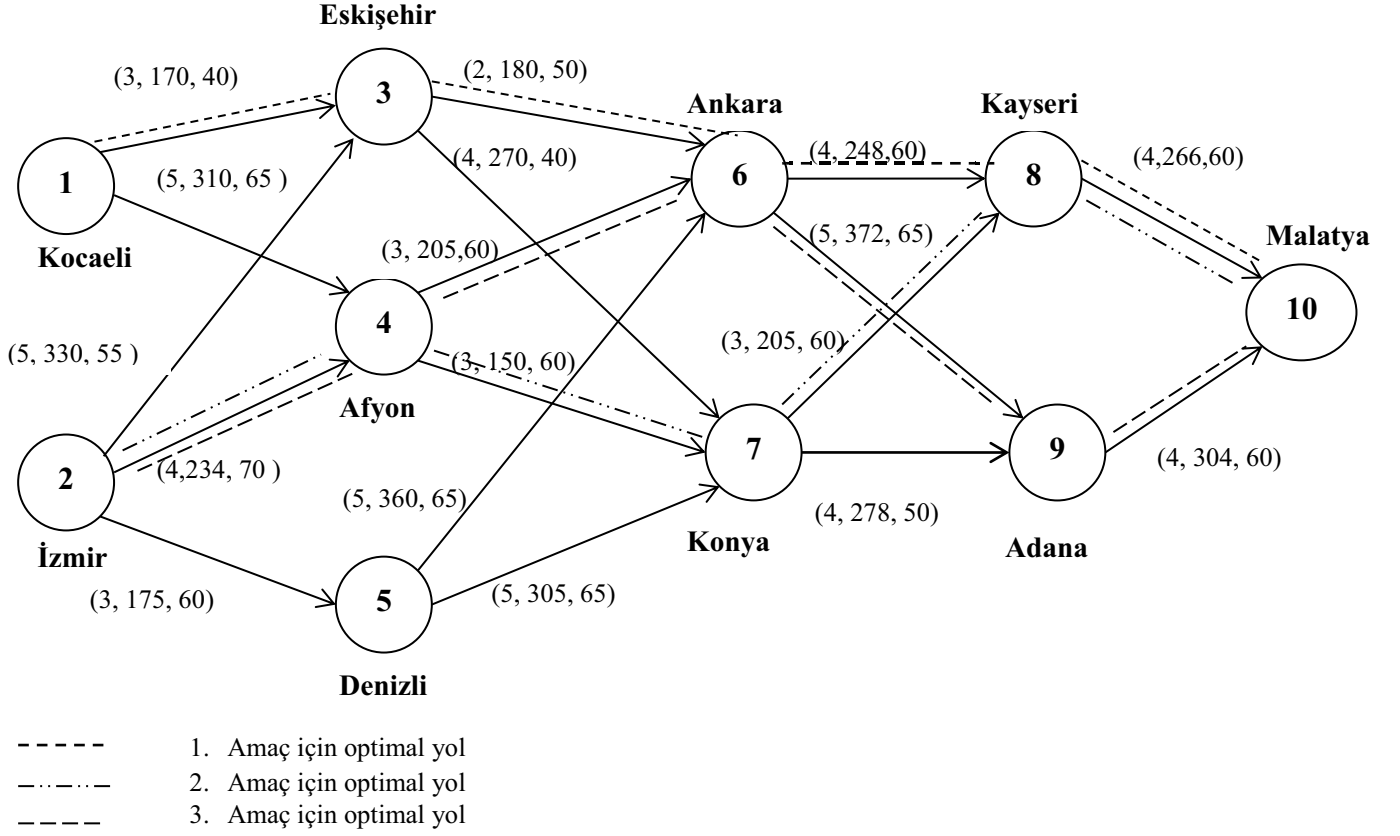
|                | <b>Malatya</b> |                |                  |
|----------------|----------------|----------------|------------------|
|                | Sefer Sayısı   | Toplam Maliyet | Ortalama Maliyet |
| <b>Kayseri</b> | 26             | 6.916          | <b>266</b>       |
| <b>Adana</b>   | 22             | 6.688          | <b>304</b>       |

✓ Yol güvenliği amacının ilgilendiği konu, ürünün hasarsız ve kazasız bir şekilde müşteriye teslim edilmesidir. Firmadan temin edilen kaza riski haritasında belirlenen güzergaha ilişkin ortalama güvenlik puanları hesaplanmıştır. Bulunan kaza riski 100 üzerinden puanlanmıştır. Kaza ve hasar riskli yollar için verilen puan 0'a yaklaşırken, güvenli yollar için 100'e yaklaşmaktadır.

#### **3.4.4. Problemin Ağ Diyagramı ile Gösterimi ve Dinamik Programlama Modeli ile Çözümü**

Üç amaca yönelik oluşturulan şekildeki yol problemi ele alınarak çok amaçlı dinamik programlama modeli kurulmuştur. Şekilde iller arasındaki mesafe saat olarak, maliyetler TL cinsinden sunulmuştur. Yol güvenliği ise kaza ve hasar riskine göre 100 üzerinden verilen puan ile gösterilmektedir.

**Şekil 9:** Çok Amaçlı Dinamik Programlama Yol Probleminin Ağ Diyagramı ile Gösterimi



Şekildeki yol probleminin çözümünde öncelikle her amaç için ayrı ayrı optimal yollar belirlenir. Çözüme problemin son aşamasından başlayıp her seferinde bir önceki aşamaya geçerek, geriye doğru çözüm yolu izlenecektir.

1.amaç için optimal yolun belirlenmesi;

n=0 için

$$n_0(10) = 0^*$$

n=1 için

$$n_1(9) = C_{910} + f_0(10) = 4+0= 4^*$$

$$n_1(8) = \mathbf{C_{810} + f_0(10) = 4+0= 4^*}$$

n=2 için

$$n_2(7) = \min \begin{cases} C_{79} + f_1(9) = 4+4= 8 \\ C_{78} + f_1(8) = 3+4= 7^* \end{cases}$$

$$n_2(6) = \min \begin{cases} C_{69} + f_1(9) = 5+4= 9 \\ \mathbf{C_{68} + f_1(8) = 4+4= 8^*} \end{cases}$$

n=3 için

$$n_3(5) = \min \begin{cases} C_{57} + f_2(7) = 5+7= 12^* \\ C_{56} + f_2(6) = 5+8= 13 \end{cases}$$

$$n_3(4) = \min \begin{cases} C_{47} + f_2(7) = 3+7= 10^* \\ C_{46} + f_2(6) = 3+8= 11 \end{cases}$$

$$n_3(3) = \min \begin{cases} C_{37} + f_2(7) = 4+7= 11 \\ \mathbf{C_{36} + f_2(6) = 2+8= 10^*} \end{cases}$$

n=4 için

$$n_4(2) = \min \begin{cases} C_{25} + f_3(5) = 3+12= 15 \\ C_{24} + f_3(4) = 4+10= 14^* \\ C_{23} + f_3(3) = 5+10= 15 \end{cases}$$

$$n_4(1) = \min \begin{cases} C_{14} + f_3(4) = 5+10= 15 \\ C_{13} + f_3(3) = 3+10= 13^* \end{cases}$$

Karar verici varış süresini minimize etmek istediğinde sözkonusu ürünü varış şehri olan Malatya' ya Kocaeli'deki depodan göndermelidir. Bu sayede tırın varış süresi 13 saat ve kullanacağı güzergah Kocaeli – Eskişehir – Ankara - Kayseri – Malatya olacaktır.

2.amaç için optimal yolun belirlenmesi;

n=0 için

$$n_0(10) = 0^*$$

n=1 için

$$n_1(9) = C_{910} + f_0(10) = 304+0= 304^*$$

$$n_1(8) = C_{810} + f_0(10) = 266+0= 266^*$$

n=2 için

$$n_2(7) = \min \begin{cases} C_{79} + f_1(9) = 278+304= 582 \\ C_{78} + f_1(8) = 205+266= 471^* \end{cases}$$

$$n_2(6) = \min \begin{cases} C_{69} + f_1(9) = 372+304= 676 \\ C_{68} + f_1(8) = 248+266= 514^* \end{cases}$$

n=3 için

$$n_3(5) = \min \begin{cases} C_{57} + f_2(7) = 305 + 471 = 776^* \\ C_{56} + f_2(6) = 360 + 514 = 874 \end{cases}$$

$$n_3(4) = \min \begin{cases} C_{47} + f_2(7) = \mathbf{150 + 471 = 621^*} \\ C_{46} + f_2(6) = 205 + 514 = 719 \end{cases}$$

$$n_3(3) = \min \begin{cases} C_{37} + f_2(7) = 270 + 471 = 741 \\ C_{36} + f_2(6) = 180 + 514 = 694^* \end{cases}$$

n=4 için

$$n_4(2) = \min \begin{cases} C_{25} + f_3(5) = 175 + 776 = 951 \\ C_{24} + f_3(4) = \mathbf{234 + 621 = 855^*} \\ C_{23} + f_3(3) = 330 + 694 = 1.024 \end{cases}$$

$$n_4(1) = \min \begin{cases} C_{14} + f_3(4) = 310 + 621 = 931 \\ C_{13} + f_3(3) = 170 + 694 = 864^* \end{cases}$$

Yol maliyetinin minimize edilmesi amaçlandığında seçilecek olan güzergah İzmir – Afyon – Konya – Kayseri – Malatya olacaktır. Böylece katlanılan ulaşım maliyeti 855 TL olacaktır ve gönderim İzmir’den gerçekleştirilecektir.

3.amaç için optimal yolun belirlenmesi;

n=0 için

$$n_0(10) = 0^*$$

n=1 için

$$n_1(9) = C_{910} + f_0(10) = 60+0 = 60^*$$

$$n_1(8) = C_{810} + f_0(10) = 60+0 = 60^*$$

n=2 için

$$n_2(7) = \max \begin{cases} C_{79} + f_1(9) = 50+60 = 110 \\ C_{78} + f_1(8) = 60+60 = 120^* \end{cases}$$

$$n_2(6) = \max \begin{cases} C_{69} + f_1(9) = 65+60 = 125^* \\ C_{68} + f_1(8) = 60+60 = 120 \end{cases}$$

n=3 için

$$n_3(5) = \max \begin{cases} C_{57} + f_2(7) = 65+120 = 185 \\ C_{56} + f_2(6) = 65+125 = 190^* \end{cases}$$

$$n_3(4) = \max \begin{cases} C_{47} + f_2(7) = 60+120 = 180 \\ C_{46} + f_2(6) = 60+125 = 185^* \end{cases}$$

$$n_3(3) = \max \begin{cases} C_{37} + f_2(7) = 40+120 = 160 \\ C_{36} + f_2(6) = 50+125 = 175^* \end{cases}$$

n=4 için

$$n_4(2) = \max \begin{cases} C_{25} + f_3(5) = 60+190 = 250 \\ C_{24} + f_3(4) = 70+185 = 255^* \\ C_{23} + f_3(3) = 55+175 = 230 \end{cases}$$

$$n_4(1) = \max \begin{cases} C_{14} + f_3(4) = 65 + 185 = 250 \\ C_{13} + f_3(3) = 40 + 175 = 215^* \end{cases}$$

Karar vericinin son amacı olan maksimum düzeyde yol güvenliğini sağlayacak güzergah ise İzmir - Ankara – Adana – Malatya’dır. Dolayısıyla yol güvenliğini maksimum düzeyde karşılayan gönderim İzmir’den yapılmalıdır.

Firmanın ulaştırma sorununda her amacını ayrı ayrı optimum düzeyde karşılayan güzergahlar belirlenmiştir. Ancak depolardan herhangi birinden gönderim yapılması sözkonusu olmadığı durumda diğer depodan alternatif olarak karar vericinin amacına uygun gönderim yapılacak nakliye güzergahları da bu çalışmada sunulmuştur.

İzmir deposundan gönderim gerçekleştirilemeyeceği bir durumda karar vericinin amacı minimum maliyete katlanmak ise tır Kocaeli depodan çıkıp Eskişehir – Ankara – Kayseri güzergahını izleyerek Malatya’ya ulaşabilir. Böylece maliyeti 864 TL olacaktır. Eğer amacı yol güvenliğinin maksimum seviyede olması ise Kocaeli’den çıkan tırın güzergahı Eskişehir – Ankara – Adana - Malatya olmalıdır.

Firmanın Kocaeli deposundan gönderim yapması sözkonusu değil ise karar verici minimum sürede nakliyeyi gerçekleştirmeyi amaçladığında İzmir deposundan Malatya’ya gönderim yapılacak nakliye güzergahı şu şekilde olmalıdır: Afyon – Konya – Kayseri – Malatya. Tır bu güzergahı kullandığında ulaştırma süresi 14 saat olacaktır.

Her bir amaç için ayrı ayrı belirlenen optimal yollar Tablo 7’de kriterler bazında karşılaştırılmıştır;

**Tablo 7:** Belirlenen Optimal Yolların Karşılaştırılması

|                         | Zaman     | Maliyet      | Güvenlik   |
|-------------------------|-----------|--------------|------------|
| 1.amaç için optimal yol | 13        | 864          | 210        |
| 2.amaç için optimal yol | 14        | 855          | 250        |
| 3.amaç için optimal yol | 16        | 1.115        | 255        |
| <b>Toplam</b>           | <b>43</b> | <b>2.834</b> | <b>715</b> |

Tablo 7’de karşılaştırılan optimal yollara ilişkin her üç amaç için oranlar hesaplanmıştır.

1.amaç için oran hesaplaması;

$$r_1 = \frac{\text{1.amaç için optimal zaman değeri}}{\text{zaman değerlerinin toplamı}}$$

2.amaç için oran hesaplaması;

$$r_2 = \frac{\text{2.amaç için optimal maliyet değeri}}{\text{maliyet değerlerinin toplamı}}$$

3.amaç için oran hesaplaması;

Firmanın ilk iki amacında mevcut değerlerin minimizasyonunu sağlamak hedeflenirken iken üçüncü amacında güvenlik puanı maksimizasyonu sağlamaktır. Dolayısıyla bu üç amacın oranlarının toplanabilmesi için ilk iki amaca uygun olarak maksimizasyonu amaçlanan güvenlik puanına ilişkin aşağıdaki oran formülü kullanılmıştır;

$$r_3 = \frac{1-i}{2} \implies i = \frac{\text{3.amaç için optimal güvenlik değeri}}{\text{güvenlik değerlerinin toplamı}}$$

Mevcut formülasyona göre yapılan hesaplam sonuçları aşağıdaki tabloda verilmiştir;

**Tablo 8:** Belirlenen Optimal Yolların Oransal Karşılaştırılması

|                         | Zaman    | Maliyet  | Güvenlik |
|-------------------------|----------|----------|----------|
| 1.amaç için optimal yol | 0,30     | 0,31     | 0,36     |
| 2.amaç için optimal yol | 0,33     | 0,30     | 0,32     |
| 3.amaç için optimal yol | 0,37     | 0,39     | 0,32     |
|                         | <b>1</b> | <b>1</b> | <b>1</b> |

Sunulan tüm çözümlerde optimal yollar seçilirken tüm amaçların aynı önem düzeyine sahip olduğu varsayıldı. Ancak karar verici konumundaki firma yöneticisi her amacın aynı düzeyde olmadığını düşünmekte olup her amaca yönelik önem düzeyini bildiren ağırlıklar saptamıştır. Ağırlıklar 0'dan uzaklaştıkça amacın firma açısından önem düzeyi artmaktadır ve verilen ağırlıklar aşağıdadır;

$$W_1= 0,30$$

$$W_2= 0,45$$

$$W_3= 0,25$$

Optimum sonuca ulaşmak için aşağıdaki formül kullanılmaktadır. Formülasyonda  $W_k$  amaçların ağırlığını,  $r_k$  ise daha önce hesaplanmış olan yolların ilgili amaçtan aldıkları nisbi değerlerdir.

$$Z_{min}^* \left[ \sum_{k=1}^3 (W_k * r_k) \right]$$

1.amaç için optimal yol

$$(0,30 \times 0,30) + (0,31 \times 0,45) + (0,36 \times 0,25) = 0,32$$

### 2.amaç için optimal yol

$$(0,33 \times 0,30) + (0,30 \times 0,45) + (0,32 \times 0,25) = 0,31$$

### 3.amaç için optimal yol

$$(0,37 \times 0,30) + (0,39 \times 0,45) + (0,32 \times 0,25) = 0,37$$

Bulunan sonuçlara göre minimizasyonu sağlayan optimum seçim 2. amaç için optimal yolu sağlayan İzmir - Afyon – Konya – Kayseri –Malatya güzergahıdır. Bu yolun seçimi ile ulaşım süresi 14 saat, ulaşım maliyeti 855TL ve toplam güvenlik puanı 250 olacaktır.

Firma sahip olduğu amaçlar doğrultusunda tek bir amaca yönelik optimum kararı verebileceği gibi deneyimlerine dayanarak belirlediği amaçlarına verdiği ağırlıklar ile bulunan optimal kararda seçebilir. Değişen şartlara uygun olarak verilen bu ağırlık düzeylerinde değişiklik yapabilir ve yeni optimal sonuçlar elde edebilir.

## SONUÇ

İşletmeler alınacak her bir karar için pek çok alternatif arasından seçim zorunda kalmaktadır. Çoğu zaman verilen kararlar işletmelerin devamlılığı ve karlılığını etkileyecek hayati önem taşır. Bu yüzden karar verme süreci stratejik bir öneme sahiptir. Çünkü İşletmelerin geleceği, verilecek bu kararların doğruluğu ve uygun zamanda verilmesi ile yakından ilgilidir.

Optimum seçime ulaşmak işletmelerin sahip olduğu karmaşık yapısı, birbiriyle çatışan çoklu amaçları nedeniyle bir hayli zordur. Karmaşıklığı olabildiğince sadeleştirmek optimum seçime ulaşmada ve aksinde oluşabilecek kayıpların tümünün engeline karşı karar vericinin işini daha kolay kılacaktır. Ancak karar vericinin deneyim ve sezgileri önemli olmakla birlikte, karar verme sürecinde yetersiz kalabilmektedirler. Karar vericiler bu tür karar süreçlerinde, deneyimlerinin yanı sıra, kendilerini en iyi çözüme ulaştıracak bilimsel yöntemlere ihtiyaç duymaktadırlar.

Dinamik programlama genel olarak optimizasyon problemlerinin çözülmesinde kullanılan bir yöntemdir. Bu problemlerin çözümü, öncelikle kararların belli bir sıraya sokulmasını ve “Optimallik prensibi” gereği , ana problemi parçalara bölmeyi, son karar aşamasını çözmeyi, daha sonra geriye dönmeyi ardından ikinci son karar aşamasını çözmeyi ve ilk karar aşaması çözüme ulaşana kadar bu yönteme devam etmeyi gerektirir.

Dinamik programlama 1950’lerin başında Belman tarafından ortaya atılmıştır. Geçen süreçte pek çok alanda sıralı veya çok aşamalı karar vermeyi içeren karmaşık problemlerin çözümüne yönelik güçlü bir optimizasyon tekniği olarak kullanılmıştır. Ancak günümüzde işletmeler için belirli zaman süreçlerinde vereceği sıralı kararları ve birden çok amaca yönelik etki beklentileri dinamik programlamanın tek başına yeterli olmasını mümkün kılmamaktadır. Buna bağlı olarak geleneksel dinamik programlamaya dayanan çok amaçlı dinamik programlama metodolojisi dinamik programlama karakteristiğine uyan, çatışan amaçlara sahip problemlerin çözümünü sağlamaya yönelik geliştirilmiş bir tekniktir.

Gerçek dünyada pek çok sıralı karar verme problemi ve bu problemlerin çözümü birbirleriyle çatışan veya rekabet içinde olan amaçların optimizasyonunu gerektirir. Çoklu amaçlar ve dinamik kontrol genel olarak çok amaçlı optimizasyon, çoklu amaçlı dinamik sistem problemlerinin optimizasyonu anlamına gelir. Dinamik programlamada ağ sorunu tek bir amaç için planlanırken gelişen dünya koşullarında karar vericiler için bir çok amacı aynı plan içinde içinde uygulamak mecburiyet halindedir. Bu çalışma çoklu amaçlara sahip işletmelerin karar verme sürecine yardımcı bir dinamik programlama modeli ile çözüm getirmek amacıyla yapılmıştır.

Tezin birinci bölümünde, karar verme kavramı açıklanmış ve yönetsel karar verme süreci aşamaları sunulmuştur. Çalışmada çok amaçlı bir model kurulacağı için öncelikle karar verme problemleri tek ölçütü karar verme problemleri ve çok ölçütlü karar verme problemleri olarak sınıflandırılmıştır.

Çok ölçütlü karar verme problemlerinin karar süreci aşamaları ayrıntılı olarak ele alınmış ve çok ölçütlü karar verme problemleri, çok nitelikli ve çok amaçlı karar verme problemleri olarak ikiye ayrılmıştır. Çok amaçlı karar problemleri; ölçütlerin, amaçların ve niteliklerin tanımlanması, kısıtları, seçenekleri, karar verici etkileşimi ve kullanım amaçları bakımından Çok nitelikli karar problemlerinden farklılık göstermektedir.

Çok amaçlı karar verme kavramları açıklanmış ve optimizasyon problemlerine çözüm sağlayan çok amaçlı programlamanın gelişimi ve uygulama alanlarına değinilmiştir. Çok amaçlı programlamanın özellikle Yöneylem Araştırmalarında hızla geliştiğine ve bu çalışmanın amacı doğrultusunda Yöneylem Araştırması alanında sunulmuş uygulamalara yer verilmiştir.

Çalışmanın ilerleyen bölümlerinde problemin çözümüne ilişkin çok amaçlı bir model kurulacağı için birinci bölümde ayrıca çok amaçlı programlama formülasyonu sunulmuştur. Çok amaçlı programlama yöntemleri çözüm üretme teknikleri, etkileşimli olmayan teknikler ve karar verici ile etkileşimli teknikler olarak sınıflandırılmıştır.

Tezin ikinci bölümünde, dinamik programlama ayrıntılı olarak ele alınmıştır. Rekabet ortamında işletmelerin sürekli değişen koşullara ayak uydurmak zorunda olması karar verme sürecinin dinamik bir yapıya sahip olmasına neden olmaktadır.

Dinamik programlamada amaç ardışık ve birbirini etkileyen alt problemler için ayrı ayrı optimum kararı bulup, bulunan sonuçları toplayarak problemi sonuçlandırmak değil, asıl problem için optimum sonucu verecek şekilde alt problemler için en iyi kararı bulmaktır. Bu sebeple işletmelerin birbirini etkileyecek sıralı kararlarında ve zaman faktörü ele alındığında uygun bir yöntemdir.

Dinamik programlama problemlerinin en önemli ilkesi Bellman'ın "Optimallik prensibi" dir. Dinamik programlama ana problemi alt problemelere ayırır ve her alt problem için bir çözüm üreterek ana problemi çözüme ulaştırır.

Dinamik programlama önceleri ekonomik sistem üzerine yapılan çalışmalarda kullanılan bir yöntem iken günümüzde pek çok karar sürecinde yardımcı bir yöntem olarak karşımıza çıkmaktadır. Tezin ikinci bölümünde Dinamik programlamanın en fazla uygulandığı işletme sorunlarıyla birlikte işletme dışı uygulama alanlarına da değinilmiştir.

Dinamik programlama problemleri genellikle geriye doğru hesaplama yöntemi ile çözülür. Ancak ileriye doğru çözüm yöntemi, analitik çözüm yöntemi ve tablosal çözüm yöntemi kullanılan diğer yöntemlerdir. Bu çalışmada çok amaçlı sorunun çözümü geriye doğru çözüm yöntemi kullanılarak bulunmuştur.

Dinamik programlama problemleri karara etki eden dışsal etmenlerin tam olarak bilinip bilinmemesine göre ikiye ayrılmaktadır. Eğer dışsal etmenler tam olarak belirlenebiliyorsa süreç deterministiktir. Ancak belirsizlik altında karar verilmesi gereken durumlarda gelecek aşamadaki durum, şuan ki durum ve karar tarafından belirlenemiyor demektir. Gelecek durumun ne olacağı hakkında olasılıklar bellidir ve problemin çözümü olasılıklar dikkate alınarak gerçekleştirilir. Bu problemler stokastik dinamik programlama problemleridir.

Tezin son bölümünde, çok amaçlı dinamik programlama ve çok amaçlı en kısa yol problemleri ile birlikte uygulamaya ışık tutacak çok amaçlı dinamik programlama formülasyonunu sunulmuştur. Uygulama kısmında, işletmenin nakliye sorununa çözüm olacak optimal yolun seçilmesi amacıyla üç ayrı amaç belirlenmiştir. Bu amaçlar; ulaştırma maliyetini ve ulaşım zamanını minimize ederken yol güvenliğinin maksimize etmektir. Problemin çözümü için çok amaçlı dinamik programlama modeli geliştirilmiştir ve karar vericiye çözüm önerileri sunulmuştur.

Uygulama yapılan firma geçmiş yıllarda nakliye yapılacak güzergahları seçerken bu çalışmada belirlenen amaçlar doğrultusunda karar vermemiştir. Uygulamada amaçlanan karar vericiye belirlenmiş bu üç amacına yönelik optimum karar seçeneklerini sunmaktır.

Öncelikle, firmanın mevcut iki deposundan müşterinin bulunduğu şehire gönderim yapılabilecek muhtemel güzergahlar belirlenmiştir. Bu güzergahlarda ulaşılması gereken iller arası ulaşım süreleri, iller arasında nakliye gerçekleştirilirken katlanılacak maliyetler ve yol ile ilgili kaza ve hasar riski bilgileri firmadan temin edilmiştir. Daha sonra seçili güzergahlara ilişkin ağ diyagramı gösterilmiştir.

Firmanın üç amacı için mevcut ulaştırma problemi dinamik programlamanın geriye doğru çözüm yolu izlenerek çözülmüş ve karar vericiye optimal yol alternatifleri sunulmuştur. Karar verici eğer ulaşım süresini minimum düzeyde tutmak istiyorsa talep edilen ürünü Kocaeli depodan çıkarmalı ve varış noktası olan Malatya'ya, Eskişehir - Ankara – Kayseri üzerinden ulaştırmalıdır. Böylece ulaştırma süresi  $3+2+4+4=13$  saat olacaktır.

Karar vericinin amacı ulaştırma maliyetini en düşük seviyede olması ise seçeceği güzergah İzmir - Afyon – Konya – Kayseri – Malatya olmalıdır. Dolayısıyla ürün İzmir'den gönderilmelidir. Böylece nakliye maliyeti  $234+ 150+205+266=855$  TL olacaktır.

Firmanın üçüncü amacı olan yol güvenliğini maksimum düzeyde olmasını istemesi halinde karar verici, İzmir – Afyon – Ankara – Kayseri – Malatya güzergahını seçmelidir.

Karar vericiye sunulan bu çözüm önerileri her bir amaç için ayrı ayrı optimum seçeneklerden oluşmaktadır. Ancak karar verici her amacın farklı düzeylerde önem düzeyine sahip olduğu düşünmektedir. Ve önem düzeylerine göre ağırlıklar saptamıştır. Verilen ağırlıklara göre firma açısından birinci öncelikli amaç 0,45 önem düzeyine sahip maliyetin minimize edilmesi amacıdır. İkinci öncelikli amacı 0,35 önem düzeyi ile ulaştırma zamanını minimize etmek ve son olarak 0,25'lik önem düzeyi ile güvenlik puanının maksimum düzeyde tutmaktır.

Verilen ağırlıklar ile çözümün tamamlanması sonucu optimum seçenek olarak en düşük sonuca sahip İzmir - Afyon – Konya – Kayseri –Malatya güzergahı belirlenmiştir.

Tüm sonuçlar değerlendirildiğinde karar verici, firmanın o an içinde bulunduğu şartları gereği belli bir amaca yönelik karar vermek istediğinde güzergah seçimi o amaç için belirlenen optimal yol olmalıdır. Ancak verdiği önem düzeyine göre tüm amaçlarını göz önünde bulundurduğunda seçimi İzmir deposundan gönderim yapmak olmalıdır.

İşletmelerde çoğu zaman sezgisel yöntemlerle alınan kararlar sebebiyle zaman ve maliyet kayıpları ve güvenlik zaafiyetleri ortaya çıkmaktadır. Karar süreçlerinde özellikle çoklu amaçlar sözkonusu olduğunda kantitatif karar tekniklerinin kullanımı da gerçek işletme ortamlarındaki karmaşıklık nedeniyle kolay olmayabilir. Uygulamada sunulan işletme verilerinden yola çıkarak firmanın çoklu amaçlarını optimum düzeyde sağlayacak dinamik programlama modelinin önerilmesi, modelle optimum sonuçların bulunması ile firmaya güzergah seçiminde kolaylık sağlayacak alternatifler sunulmuştur.

## KAYNAKÇA

Abdelaziz, F.B. ve Masmoudi, M. (2012). A Multiobjective Stochastic Program For Hospital Bed Planning. *Journal of the Operational Research Society*, 63: 530–538.

Abo-Sinna, M. (2004). Multiple Objective (Fuzzy) Dynamic Programming Problems: a Survey and Some Applications. *Applied Mathematics and Computation* 157: 861–888.

Acar, H. H. ve Eker, M. (2001). Ormancılıkta Karar Verme Süreçlerinde Orman Yol ve Üretim Planlarının Değerlendirilmesi. *Kafkas Üniversitesi Artvin Orman Fakültesi Dergisi*. 1: 67-74.

Akdemir, H. G. ve Tiryaki, F. (2011). İki Seviyeli Kesikli Stokastik Taşıma Problemi. *Electronic Journal of Vocational Colleges*. 1(1): 156-167.

Akdeniz, A. ve Aksaraylı, M. (2003). Bir İmalat İşletmesinde Çok Aşamalı Üretim - Stok Planlaması. *Yönetim ve Ekonomi Celal Bayar Üniversitesi İ.İ.B.F. Dergisi*. 10(1): 11-30.

Akdeniz, A. ve Durmaz, F. (2003). The Application of Production-Inventory Balance Model on WMI. *Yönetim ve Ekonomi Celal Bayar Üniversitesi İ.İ.B.F. Dergisi*. 10(1): 1-9.

Aksaraylı, M.(1998). *Batı Makine Sanayii'nde Çok Aşamalı Üretim Stok Planlaması*. Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi Sosyal Bilimler Enstitüsü, İzmir.

Albayrak, Y. E. (2004). *Hizmet Sektöründe Performans Odaklı Çok Amaçlı Karar Verme: Banka Performans Ölçümünde Analitik Hiyerarşi Süreci Uygulaması*. Doktora Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.

Alp, İ. (1990). *Çok Amaçlı Karar Vermede Etkileşimli Programlama Ve Kaliteli Kanatlı Yemleri İçin Bir Uygulama*. Doktora Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara.

Alp, S. (2007). Türkiye’de Eğitim Sürecinin Markov Geçiş Modeli. 8. *Türkiye Ekonometri ve İstatistik Kongresi*, Düzenleyen İnönü Üniversitesi Malatya. 24-25 Mayıs 2007.

Atlas, M. (2008). Çok Amaçlı Programlama Çözüm Tekniklerini Sınıflandırılması. *Anadolu Üniversitesi Sosyal Bilimler Dergisi*. 8(1): 47–68.

Ayed, O.B. ve Hamzaoui, S. (2012). Multiobjective Multiproduct Parcel Distribution Timetabling: A Real-World Application. *International Transactions in Operational Research*. 19: 613–629.

Balakrishnan, N. (2010). *Methods and Applications of Statistics in Business, Finance, and Management Science*. New Jersey: John Wiley & Sons. Second Edition.

Başanu, G., Teleaşa, V. ve Armeanu, E. (2011). Optimum Repartition of Transport Capacities in the Logistic System using Dynamic Programming. *Theoretical and Applied Economics*. 18(561): 17-30.

Bierman, H., Bonini, C.P. ve Hausman, W.H. (1981). *Quantitative Analysis for Business Decisions*. Homewood, Illinois: Richard D. Irwin. Sixth Edition.

Bilsel, R. U. ve Ravindran, A. (2011). A Multiobjective Chance Constrained Programming Model For Supplier Selection Under Uncertainty. *Transportation Research Part B*. 45: 1284-1300.

Biyani, P., Prakriya, S., Bagchi, A. ve Prasad, S. (2012). Dynamic Programming Based Multi-User Resource Allocation for Partial Crosstalk Cancellation in VDSL. *Ieee Communications Letters*. 16(3): 420-423.

Cebeci, S. (2009). *Dinamik Programlama İle Üretim Planlaması Ve Bir İşletme Uygulaması*. Yüksek Lisans Tezi, Marmara Üniversitesi Sosyal Bilimler Enstitüsü, İstanbul.

Cerquetti, R. (2012). Financing Policies Via Stochastic Control: A Dynamic Programming Approach. *Journal Of Global Optimization*. 53: 539-561.

Cervellera, C., Chen, V. C. P. ve Wen, A. (2006). Optimization Of Large –Scale Water Reservoir Network By Stochastic Dynamic Programming With Efficient State Space Discretization. *European Journal Of Operation Research*. Volume 171 (3): 1139-1151.

Çetin, E. (2005). Dinamik Programlama ile Sınır Tenörü Optimizasyonu. *Doğu ve Güneydoğu Anadolu Maden Kaynaklarının Değerlendirilmesi Sempozyumu Bildiriler Kitabı* (ss. 139-143), Düzenleyen Maden Mühendisleri Odası. Diyarbakır. 21-23 Nisan 2005.

Çetin, E. İ. (2002). Çok Amaçlı Karar Verme Konusunda Yeni Yaklaşımlar Ve Okul Öncesi Eğitime İlişkin Bir Uygulama. Yüksek Lisans Tezi, Akdeniz Üniversitesi Sosyal Bilimler Enstitüsü, Antalya.

Choi, J., Realff, M. J. ve Lee, J.H. (2006). Approximate Dynamic Programming: Application to Process Supply Chain Management. *American Institute of Chemical Engineers Journal*. 52(7): 2473-2485.

Cinemre, N. (2011). *Yöneylem Araştırması*. İstanbul: Evrim Yayınevi. 2. Basım.

Coello, C. A. C. ve Lamont, G. B. (2004). *An Introduction To Multi-Objective Evolutionary Algorithms And Their Applications*. World Scientific. Singapore.

Coello, C. A. C., Lamont, G. B. ve Veldhuizen, D. A. V. (2007). *Evolutionary Algorithms for Solving Multi-Objective Problems*. Second Edition, New York: Springer Science+Business Media.

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

Cui, Y. (2013). A New Dynamic Programming Procedure For Three-Staged Cutting Patterns, *Journal Of Global Optimization*. 55(2): 349-357.

Çınar, Y. (2004). *Çok Nitelikli Karar Verme ve 'Bankaların Mali Performanslarının Değerlendirilmesi' Örneği*. Yüksek Lisans Tezi, Ankara Üniversitesi Sosyal Bilimler Enstitüsü, Ankara.

Çiçek, D.S. (2000). *Otel İşletmelerinde Dinamik Programlamaya Dayalı Oda Fiyatının Belirlenmesi*. Yüksek Lisans Tezi, Anadolu Üniversitesi Sosyal Bilimler Enstitüsü, Eskişehir.

Daellenbach, H.G. ve Kluyver, C.A. (1980). Note on Multiple Objective Dynamic Programming. *Journal of Operational Research Society*. 31: 591-594.

Deb, K. (2005). Multi -Objective Optimization. *Search Methodologies*. (ss. 273-316). Editors Edmund K. Burke and Graham Kendall. ABD: Springer Science+Business Media, LLC.

Demir, M. H. (1974). Dinamik Programlama. *İstanbul Üniversitesi İşletme Fakültesi Dergisi*. 3(1), 296-303.

Denardo, E. V.(2011). *Linear Programming and Generalizations A Problem-based Introduction with Spreadsheets*. USA: Springer Science+Business Media. vol.149 chapter 8.

Doğan, M. (1985). *İşletmelerde Karar Verme Teknikleri*. İzmir: Bilgehan Basımevi.

Dreyfus, S. (2002). Richard Bellman on the Birth of Dynamic Programming. *Operation Research*. 50(1), 48-51.

Durucasu, H. (2004). Stok Planlamada Wagner-Whitin Dinamik Programlama Algoritmasının Kullanımı. *Dumlupınar Üniversitesi, Sosyal Bilimler Dergisi*. 11: 115-139.

Ehrgott, M., Waters, C., Kasımbeyli, R. ve Üstün, Ö. (2009). Multiobjective Programming and Multiattribute Utility Functions in Portfolio Optimization. *Infor*, 47(1): 31-42.

Emhan, A. (2007). Karar Verme Süreci ve Bu Süreçte Bilişim Sistemlerinin Kullanılması. *Elektronik Sosyal Bilimler Dergisi*, 6(21): 212-224.

Erdelyi, A. ve Topaloğlu, H. (2010). Approximate dynamic programming for dynamic capacity allocation with multiple priority levels. *IIE Transactions*. 43(2): 129-142.

Ersöz, F. ve Kabak, M. (2010). Savunma Sanayi Uygulamalarında Çok Kriterli Karar Verme Yöntemlerinin Literatür Araştırması. *Savunma Bilimleri Dergisi*. 9(1): 97-125.

Florios, K., Mavrotas, G. ve Diakoulaki, D. (2010). Solving Multiobjective, Multiconstraint Knapsack Problems Using Mathematical Programming And Evolutionary Algorithms. *European Journal of Operational Research*. 203: 14–21.

Fox, J., Cooper, R.P ve Glasspool, D.W. (2013). A Canonical Theory Of Dynamic Decision-Making. *Front Psychol*. (4)150, 1-19.

Grüne, L. ve Sammler, W. (2007). Asset Pricing With Dynamic Programming. *Comput Econ*. 29: 233–265.

Gülenç, İ. F. ve Bilgin, G. A. (2010). Yatırım Kararları İçin Bir Model Önerisi: Ahp Yöntemi. *Marmara Üniversitesi Sosyal Bilimler Enstitüsü Hakemli Dergisi: Öneri*. 9(34): 97-107.

Günyaşar, V. (1990). *Karar Vermede Çok Amaçlı Programlama Yaklaşımı: Etkileşimli Bir Model Uygulama*. Yayımlanmış Doktora Tezi, Marmara Üniversitesi Sosyal Bilimler Enstitüsü, İstanbul.

Hallaç, O. (1972). Bir Karar Verme Vasıtası Olarak Harekat Araştırması. *İstanbul Üniversitesi İşletme Fakültesi Dergisi*. 1(2), 103-116.

Hallaç, O. (1991). *Kantitatif Karar Verme Teknikleri (Yöneylem Araştırması)*. İstanbul: Evrim Basım Yayım Dağıtım.

Hartman, J. C., Büyüktaktın, İ.E. ve Smith, J.C. (2010). *IIE Transactions*. 42(12): 915-939.

Hillier, F.S., Lieberman, G.J. (1990). *Introduction to Operations Research*. Fifth Edition, New York: McGraw-Hill.

Hillier, F.S., Lieberman, G.J. (2001). *Introduction to Operations Research*. Seventh Edition, New York: McGraw-Hill.

Hillier, F.S., Lieberman, G.J. (2010). *Introduction to Operations Research*. Ninth Edition, New York: McGraw-Hill.

Hsu, C.I., Li, H.C., Liu, S.M. ve Chao, C.C. (2011). Aircraft Replacement Scheduling: A Dynamic Programming Approach, *Transportation Research*, Part E 47: 41–60.

Huang, J.J., Chen, C. Y., Liu, H. H. ve Tzeng, G. H. (2010). A Multiobjective Programming Model For Partner Selection-Perspectives Of Objective Synergies And Resource Allocations. *Expert Systems with Applications*. 37: 3530–3536.

İçağa, Y., Yurteu, Ş. ve Ulutürk, Y. (2007). Yeraltı Suyu Seviye Degisiminin Stokastik Modellemesi: Akarçay Afyon Alt Havzası Örneği, *Süleyman Demirel Üniversitesi, Fen Bilimleri Enstitüsü Dergisi*, 11(2): 180-186.

Karakaya, A. (2004). Yönetmel Bilginin Özellikleri Üzerine Bir Araştırma, *Teknoloji*, 7(4): 605-614.

Kaufmann, A. (1967). *Graphs, Dynamic Programming and Finite Game*. New York-London, Academic Press, Çeviri: Henry C. Sneyd. Vol.36.

Kaya, İ., Kılınç, M.S. ve Çevikcan, E. (2007). Makine-Techizat Seçim Probleminde Bulanık Karar Verme Süreci, *Mühendis ve Makine*, 49(576): 8-14.

Keçek, G. (2005). Bir Dişli Fabrikasında Tamsayılı Hedef Programlama Uygulama Denemesi, *Dumlupınar Üniversitesi Sosyal Bilimler Dergisi*. 13: 111-130.

Koçoğlu, E. (2010). *İşletmelerde Yöneticilerin Karar Verme Süreci ve Bu Süreçte Bilişim Sistemlerinin Kullanımı : Ankara İli Örneği*. Yayımlanmış Yüksek Lisans Tezi, Atılım Üniversitesi Sosyal Bilimler Enstitüsü, Ankara.

Kostreva, M. M. ve Wiecek, M.M. (1993). Time Dependency in Multiple Objective Dynamic Programming. *Journal of Mathematical Analysis and Applications*. 173: 289-307.

Köse, A. (2004). *Çok Amaçlı Karar Alma Yöntemleri Ve Türk Sigorta Sektörüne Yönelik Bir Uygulama*. Yayımlanmış Doktora Tezi, Marmara Üniversitesi Sosyal Bilimler Enstitüsü, İstanbul.

Lai, K.K ve Li, L. (1999). A Dynamic Approach to Multiple-Objective Resource Allocation Problem. *European Journal of Operational Research*. 117: 293-309.

Lew, A. ve Mauch, H. (2007). *Dynamic Programming: A Computational Tool*. Berlin: Springer Berlin Heidelberg.

Liu, P., Zhao, J. ve Shen, Y. (2012). Optimal Reservoir Operation Using Stochastic Dynamic Programming, *Journal of Water Resource and Protection*. 4: 342-345.

Liu, Y., Chu, C. Ve Yu, Y. (2012). Aggregated State Dynamic Programming For A Multiobjective Two-Dimensional Bin Packing Problem. *International Journal of Production Research*. 50(15): 4316–4325.

Mahmoud, M. R. (2006). High Dimension Dynamic Programming Model For Water Resources Expansion Projects. *Engineering Optimization*. 38(3): 371-389.

Mainali, M. K., Mabu, S., Yu, S., Eto, S. ve Hirasawa, K. (2011). Dynamic Optimal Route Search Algorithm for Car Navigation Systems with Preferences by Dynamic Programming, *IEEJ Transactions on Electrical and Electronic Engineering*. 6(1): 14–22.

Meng, Q. ve Wang, T.(2011). A Scenario-Based Dynamic Programming Model for Multi-Period Liner Ship Fleet Planning, *Transportation Research. Part E* 47: 401–413.

Meza, J.L.C., Yıldırım, M. B. ve Masud, A. S. M. (2009). A Multiobjective Evolutionary Programming Algorithm and Its Applications to Power Generation Expansion Planning. *IEEE Transactions On Systems, Man, And Cybernetics- Part A: Systems And Humans*, 39(5): 1086- 1096.

Ofluoğlu, G., Büyükyılmaz, O. ve Koltan, Ş. (2006). İnsan Kaynağı Seçiminde Çok Ölçütlü Karar Verme Yöntemleri: Etkileşimli Beklenti Düzeyi Yaklaşımı, *Kamu-İş.* 9(1): 105-125.

Özdemir, A. (2004). Üretim-Envanter Düzeyinin Belirlenmesinde Dinamik Programlama Modelinin Uygulanması, *Dokuz Eylül Üniversitesi Sosyal Bilimler Enstitüsü Dergisi.* 6(3): 98-112.

Özdemir, A. (2004). *Yönetmel Karar Verme Sürecinde Dinamik Amaç Programlama Yaklaşımı ve Bir Uygulama.* Yayımlanmamış Doktora Tezi, Dokuz Eylül Üniversitesi Sosyal Bilimler Enstitüsü, İzmir.

Özdemir, A. (2009). Markov Karar Süreci İle Modellenen Stokastik Ve Çok Amaçlı Üretim/Envanter Problemlerinin Hedef Programlama Yaklaşımı İle Çözülmesi, *Dokuz Eylül Üniversitesi Sosyal Bilimler Enstitüsü Dergisi.* 11(3): 75-96.

Özdemir, A. ve Gümüsoğlu, Ş. (2006). Dinamik Amaç Programlama Modelinin Ürünleri Makinelere Atama Problemine Uygulanması, *Review of Social, Economic & Business Studies.* 7(8): 359-383.

Pangilinan, J.M.A. ve Janssens, G.K. (2007). Evolutionary Algorithms for the Multiobjective Shortest Path Problem. *International Journal of Computer and Information Engineering.* 1(5): 322- 327).

- Patır, S. (2009). Dinamik Programlama ve Bir Ecza Deposunun Şehir İçi İlaç Dağıtımına Alternatifli Bir Çözüm Önerisi, *Atatürk Üniversitesi İktisadi ve İdari Bilimler Dergisi*. 23(2): 63-79.
- Pérez-Gladisha, B., Gonzalezb, I., Bilbao-Terola, A. ve Arenas-Parraa, M. (2010). Planning A TV Advertising Campaign: A Crisp Multiobjective Programming Model From Fuzzy Basic Data. *Omega*. 38: 84 -94.
- Righini, G. ve Salani, M. (2007). New Dynamic Programming Algorithms for the Resource Constrained Elementary Shortest Path Problem. *Networks*. 51(3). 155-170.
- Serafini, P. (2006). Dynamic Programming and Minimum Risk Paths, *European Journal of Operational Research*. 175: 224–237.
- Sevinç, C. (2008). *Lojistik Planlama Çok Ürünlü Çok Kademeli İkmal Zincirindeki Depolarda Bulundurulması Gerekli Minimum Malzeme Miktarının Dinamik Programlama Yöntemiyle Optimizasyonu*. Yüksek Lisans Tezi, Dokuz Eylül Üniversitesi Sosyal Bilimler Enstitüsü, İzmir.
- Sinha, S. M. (2006). *Mathematical Programming: Theory and Methods*. India: Elsevier Science.
- Sitarz, S. (2009). Ant Algorithms And Simulated Annealing For Multicriteria Dynamic Programming. *Computers&Operation Research*. 36(2): 433-441.
- Sitarz, S. (2009). Pareto optimal allocations and dynamic programming. *Ann Oper Res*. 172: 203–219.
- Taha, H. A. (2000). *Yöneylem Araştırması*. Çev. Ş. Alp Baray ve Şakir Esnaf. İstanbul: Literatür Yayıncılık.
- Tarapata, Z. (2007). Selected Multicriteria Shortest Path Problems: an Analysis of Complexity, Models And Adaptation of Standard Algorithms. *Int. J. Appl. Math. Comput. Sci*. 17(2): 269–287.

Tarun, P.K. (2008). *A Dynamic Multiple Stage, Multiple Objective Optimization Model with an Application to a Wastewater Treatment System*. Doktora Tezi, The University of Texas at Arlington, Arlington, Texas.

Tekin, Ö. A. (2009). *Yönetimde Karar Verme: Batı Antalya Bölgesinde Bulunan Beş Yıldızlı Otel İşletmelerindeki Çeşitli Departman Yöneticilerinin Karar Verme Stillerini Tespit Etmeye Yönelik Uygulamalı Bir Araştırma*. Yüksek Lisans Tezi, Akdeniz Üniversitesi Sosyal Bilimler Enstitüsü, Antalya.

Tekin, Ö. A. ve Ehtiyar, V.R. (2010). Yönetimde Karar Verme: Batı Antalya Bölgesindeki Beş Yıldızlı Otellerde Çalışan Farklı Departman Yöneticilerinin Karar Verme Stilleri Üzerine Bir Araştırma. *Journal of Yasar University*. 20(5): 3394-3414.

Topaloglou, N., Vladimirov, H. ve Zenios, S.A. (2008). A Dynamic Stochastic Programming Model For International Portfolio Management. *European Journal of Operational Research*. 185(3): 1501–1524.

Trafalis, T. B., Mishina, T. ve Foote, B.L. (1999). An Interior Point Multiobjective Programming Approach For Production Planning With Uncertain Information. *Computers & Industrial Engineering*. 37: 631- 648.

Trzaskalik, T. ve Sitarz, S. (2007). Discrete Dynamic Programming with Outcomes in Random Variable Structures. *European Journal of Operational Research*. 177: 1535–1548.

Turan, G. (2001). *İki Amaçlı Portföy Seçimi Problemi*. Yüksek Lisans Tezi, İstanbul Üniversitesi Sosyal Bilimler Enstitüsü, İstanbul.

Türker, A. (1986). *Ağaçlandırmada Çok Ölçütlü Karar Verme*. Doktora Tezi, İstanbul Üniversitesi Fen Bilimleri Enstitüsü.

Tütek, H. H. ve Gümüsoğlu, Ş. (2005). *Sayısal Yöntemler Yönetimsel Yaklaşım*. İstanbul: Beta Basım Yayım Dağıtım A.Ş.

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

Xidonas, P., Mavrotas, G. ve Psarras, J. (2010). Equity Portfolio Construction And Selection Using Multiobjective Mathematical Programming. *J Glob Optim*, 47:185–209.

Wen, Y. H. ve Hsu, C. I. (2006). Interactive Multiobjective Programming In Airline Network Design For International Airline Code-Share Alliance. *European Journal of Operational Research*. 174: 404–426.

Winston, W.L. (1991). *Operations Research-Applications and Algorithms*. Second Edition, USA: Brooks/Cole.

Yücel, M. (2004). Pazarlama Problemlerinin Çözümünde Yöneylem Araştırmasının Önemi. *Anadolu Bölgesi Araştırma Dergisi*, 2(3): 70-76.

Yücel, M., Ulutaş, A. (2010). Dinamik Programlamanın İşçilik Maliyetlerinin Minimizasyonunda Uygulanması, *Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*. 15(3): 271-290.

Zabransky, F. (1965). *Basic Algorithms of Dymanic Programming*. Yüksek Lisans Tezi, Faculty of Graduate Studies The University of Western Ontario, London.