



**HAVA HARP OKULU
HAVACILIK VE UZAY TEKNOLOJİLERİ
ENSTİTÜSÜ**



**BULANIK ÇOK MODLU KAYNAK KISITLI PROJE ÇİZELGELEME
PROBLEMLERİNİN ÇÖZÜMÜ İÇİN MATEMATİKSEL BİR MODEL**

DOKTORA TEZİ

Ömer ATLI

Endüstri Mühendisliği Ana Bilim Dalı Başkanlığı

Endüstri Mühendisliği Doktora Programı

Tez Danışmanı: Prof. Dr. Cengiz KAHRAMAN

TEMMUZ 2012

HAVA HARP OKULU
HAVACILIK VE UZAY TEKNOLOJİLERİ ENSTİTÜSÜ

**BULANIK ÇOK MODLU KAYNAK KISITLI PROJE ÇİZELGELEME
PROBLEMLERİNİN ÇÖZÜMÜ İÇİN MATEMATİKSEL BİR MODEL**

DOKTORA TEZİ

Ömer ATLI
(307203)

Endüstri Mühendisliği Ana Bilim Dalı Başkanlığı

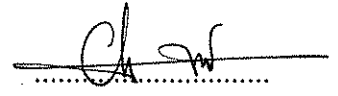
Endüstri Mühendisliği Doktora Programı

Tez Danışmanı: Prof. Dr. Cengiz KAHRAMAN

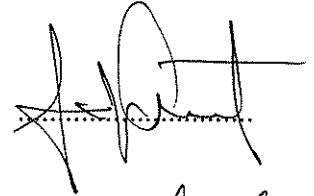
TEMMUZ 2012

Hava Harp Okulu Havacılık ve Uzay Teknolojileri Enstitüsünün 307203 numaralı Doktora Öğrencisi **Ömer ATLI**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “**BULANIK ÇOK MODLU KAYNAK KISITLI PROJE ÇİZELGELEME PROBLEMLERİNİN ÇÖZÜMÜ İÇİN MATEMATİKSEL BİR MODEL**” başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Prof. Dr. Cengiz KAHRAMAN**
İstanbul Teknik Üniversitesi



Jüri Üyeleri : **Yrd. Doç. Dr. Tufan DEMİREL**
Yıldız Teknik Üniversitesi



Yrd. Doç. Dr. Murat ERMIŞ
Hava Harp Okulu



Yrd. Doç. Dr. S. Kerem AYTULUN
Hava Harp Okulu



Yrd. Doç. Dr. Sezgin KILIÇ
Hava Harp Okulu



Teslim Tarihi : 18 Haziran 2012
Savunma Tarihi : 27 Temmuz 2012

Bu tez alışmasında belirtilen grüş ve yorumlar yazara aittir. Türk Silahlı Kuvvetleri'nin ya da diğerk kamu kuruluşlarının grüşlerini yansıtmaz. Ayrıca bu tez alışması bilimsel ahlak ve etik değerkere uygun olarak yazılmış olup, yararlanılan tüm eserler kaynaklarda gsterilmiştir.

Temmuz 2012

mer Atlı

Çok kıymetli aileme,

ÖNSÖZ

Tez çalışmam süresince bana araştırma olanağı sağlayan ve her konuda bilgi ve birikimleriyle bana yol gösteren tez danışmanım İTÜ İşletme Fakültesi, Endüstri Mühendisliği bölüm başkanı değerli hocam Sayın Prof. Dr. Cengiz KAHRAMAN'a ve YTÜ Mühendislik Mimarlık Fakültesi, Endüstri Mühendisliği bölüm başkanı yardımcısı Yrd. Doç. Dr. Tufan DEMİREL'e, tezimin her aşamasında yardım ve desteklerini benden esirgemeyen sevgili eşim Ceren ATLI'ya teşekkürlerimi sunarım.

Her zaman ve her koşulda varlıklarıyla bana moral veren, güç veren, maddi manevi desteklerini benden esirgemedi varlıklarını her an en içten duygularla hissettiğim çok değerli aileme ve sabrıyla, anlayış ve hoşgörüsüne sığındığım ve nice fedakarlıklara göğüs geren sevgili eşime teşekkürlerimi ve şükranlarımı bir kez daha ifade etmek isterim.

Bana bu imkanı tanıyan ve mensubu olmaktan her zaman için gurur duyduğum Hava Kuvvetleri Komutanlığına, desteği için teşekkür ve şükranlarımı ifade etmek isterim. Ayrıca değerli silah arkadaşlarıma ve çalışma arkadaşlarımla çalışmam süresinde görüş ve önerileri ile katkı sağlayan ve değerli zamanlarını bana ayırdıkları için Hava Harp Okulu Endüstri Mühendisliği bölümü personeline ve çok değerli tez izleme komitesi üyeleri Yrd. Doç. Dr. Hv. Müh. Yb. Murat ERMIŞ, Yrd. Doç. Dr. Hv. Müh. Yb. S. Kerem AYTULUN ve Yrd. Doç. Dr. Hv. Müh. Bnb. Sezgin KILIÇ'a ayrıca şükranlarımı sunarım.

Temmuz 2012

Ömer Atlı

İÇİNDEKİLER

	<u>Sayfa</u>
ÖNSÖZ.....	ix
İÇİNDEKİLER	xi
KISALTMALAR	xvii
ÇİZELGE LİSTESİ.....	xix
ŞEKİL LİSTESİ.....	xxi
SEMBOL LİSTESİ	xxiii
ÖZET.....	xxv
1. GİRİŞ.....	1
1.1 Çalışmanın Amacı ve Kapsamı.....	4
1.2 Varsayımlar	6
1.3 Çalışmanın Adımları	7
1.4 Özgün Katkıları	8
1.5 Tezin Organizasyonu	10
2. PROJE YÖNETİMİ VE ÇİZELGELEME	13
2.1 Proje ve Proje Yönetim Kavramı	13
2.2 Proje Planlama Yapısı ve Amaç.....	14
2.2.1 Deterministik yapı	15
2.2.2 Stokastik yapı	15
2.2.2.1 Çizelgelemede belirsizlik	15
2.2.2.2 Maliyetlerde belirsizlik	15
2.2.2.3 Teknolojik belirsizlik	16
2.2.3 Bulanık yapı	16
2.3 Proje Yönetiminde Temel Kavramlar	17
2.3.1 Faaliyet	17
2.3.2 Kaynak ve kaynak yapısı ve kaynakların sınıflandırılması...	18
2.3.2.1 Yenilenebilen kaynaklar.....	19
2.3.2.2 Yenilenemeyen kaynaklar	19
2.3.2.3 Kısmen yenilenebilen kaynaklar	19
2.3.3 Kapasite	20
2.3.4 Faaliyet süresi ile kaynak tahsis ilişkisi	20
2.3.5 Faaliyet süresi ile maliyet ilişkisi	22
2.3.6 Proje süresi ile maliyet ilişkisi	23
2.3.7 Kapasite yükleme ile maliyet ilişkisi	24
2.4 Proje Faaliyetlerinin Serim ve Düğümler Üzerinde Gösterimi.....	24
2.5 Öncülük İlişkileri	26
2.6 Proje Çizelgeleme	26
2.7 Çizelgeleme Problemlerinin Genel Yapısı.....	27
2.7.1 Amaçlar	27
2.7.2 Karar değişkenleri	28

2.7.3	Kısıtlar	28
2.7.3.1	Öncelik kısıtları	28
2.7.3.2	Kaynak kısıtları	29
2.7.3.3	Zaman kısıtları.....	29
2.8	Proje Çizelgelemede Çözüm Teknikleri	29
2.8.1	Kritik yol metodu (KYM)	29
2.8.2	Zaman - Maliyet dengeleme (ZMD)	30
2.8.3	Kaynak dengeleme (KD).....	30
2.8.4	Kaynak kısıtlı proje çizelgeleme yöntemleri (KKPÇ).....	31
2.8.5	Proje çizelgeleme problemlerinin sembolik gösterimi	31
2.9	Kaynak Kısıtlı Proje Çizelgeleme Modelleri (KKPÇ)	32
2.9.1	TM-KKPÇ modelinin özellikleri.....	32
2.9.2	ÇM-KKPÇ modelinin özellikleri	33
2.9.3	Tek proje (TP) ve çoklu proje (ÇP) modelleri.....	33
2.9.4	Tek kaynak (TK) ve çok kaynak (ÇK) modelleri.....	35
2.9.5	Sabit veya değişken kaynak kısıtlı KKPÇ modeli.....	35
2.9.5.1	Bölünebilir faaliyetli KKPÇ modeli.....	35
2.9.5.2	Sabit veya değişken kaynak tahsis yoğunluğu bulunan KKPÇ modelleri	36
2.9.6	KKPÇ modellerinin ortak özellikleri.....	36
2.9.7	TM-KKPÇ ve ÇM-KKPÇ modellerinin karşılaştırılması	37
2.10	Proje Çizelgeleme Problemlerinin Sınıflandırılması.....	40
2.11	Değişik Amaç İşlevleri.....	41
3.	LİTERATÜR ve BOŞLUK ANALİZİ	43
3.1	KKPÇM ile İlgili Yapılan Çalışmalar	44
3.2	Bulanık KYM'na Yönelik Yapılan Çalışmalar	50
3.3	Bulanık KKPÇ Yöntemlerine Yönelik Yapılan Çalışmalar.....	54
3.4	ÇM-KKPÇ ve B-ÇM-KKPÇ Yöntemlerine Yönelik Yapılan Çalışmalar.....	56
4.	BULANIK KÜME TEORİSİ.....	61
4.1	Bulanık Mantık.....	61
4.2	Bulanık Mantığın Avantajları.....	64
4.3	Bulanıklılık Kavramı.....	65
4.4	Bulanık Kümeler Teorisi.....	66
4.4.1	Klasik kümeler.....	68
4.4.2	Bulanık küme.....	70
4.4.3	Bulanık ve klasik kümelerin karşılaştırılması	72
4.5	Bulanık Sayılar.....	72
4.6	Olabilirlik ve Olasılık.....	72
4.7	Tanım Uzayı.....	74
4.8	Üyelik Fonksiyonu	74
4.8.1	Üçgen bulanık sayı ve üyelik fonksiyonu	75
4.8.2	Yamuk bulanık sayı ve üyelik fonksiyonu	76
4.9	Bulanık Kümeler Üzerinde İşlemler	78
4.9.1	Kesişim ($\tilde{A} \cap \tilde{B}$)	79
4.9.2	Birleşim ($\tilde{A} \cup \tilde{B}$).....	79
4.9.3	Tümleme ($\tilde{A}^c$).....	80

4.9.4	Niteleyiciler (Modifiers)	81
4.10	Bulanık Denetim	81
4.11	Bulanık Denetimin Temelleri.....	82
4.11.1	Bulanıklaştırıcı	83
4.11.2	Bulanık çıkarım	83
4.11.3	Bilgi tabanı	83
4.11.4	Kural tabanı	83
4.11.5	Durulaştırıcı.....	84
4.12	Bulanık Sayıların Sıralanması.....	85
4.12.1	Liou ve Wang'ın yöntemi	86
4.12.2	Abdel-Kader ve Dugdale'ın yöntemi	86
4.12.3	Yager sıralama indeksi	87
4.13	Endüstri Mühendisliğinde Bulanık Küme Uygulamaları.....	88
5.	BULANIK KYM.....	91
5.1	Bulanık KYM Problemlerinin DP Formülasyonu	92
5.2	TDP Çözüm Prosedürü	93
5.2.1	Klasik dönüşüm.....	94
5.2.2	Dual doğrulama sağlama	95
5.2.3	Kritik yolun belirlenmesi	96
5.3	Bulanık Aritmetik	97
5.3.1	Yamuk bulanık sayıların sıralanması	97
5.3.2	Bulanık kritik yol yöntemi (B-KYM)	98
5.3.3	Bulanık kritik yol yöntemi algoritması	100
5.4	Uygulamalı Sayısal Örnek 1	102
5.5	Uygulamalı Sayısal Örnek 2	106
6.	DETERMINİSTİK ve BULANIK FAALİYET SÜRELİ KAYNAK KISITLI PROJE ÇİZELGELEME PROBLEMLERİ.....	115
6.1	Kaynak Kısıtlı Proje Çizelgeleme.....	116
6.2	TM-KKPÇP'inin TDP Modeli.....	117
6.3	B-TM-KKPÇP'inin TDP Modeli.....	120
6.4	KKPÇP'lerinin Çözüm Yöntemleri	122
6.4.1	Kesin çözüm veren doğrusal programlama modelleri.....	122
6.4.2	Sezgisel Yöntemler	124
6.4.2.1	Paralel çizelgeleme.....	125
6.4.2.2	Seri çizelgeleme	126
6.4.2.3	Öncelik kuralları.....	126
6.4.2.4	Yaklaşık çözüm yöntemleri.....	126
6.4.2.5	Rot ve MinSlack algoritması.....	127
6.4.2.6	Rot algoritmasının örnek problem üzerinde gösterilmesi	128
6.4.2.7	B-TM-KKPÇP çözümü için sezgisel bir yöntem kanguru algoritması.....	134
6.4.2.8	Paralel kanguru algoritması adımları	135
6.4.2.9	Paralel kanguru algoritması örnek 1.....	135
6.4.3	Diğer sezgisel yöntemler	138
6.4.3.1	Yerel kısıt analizi	138
6.4.3.2	Yerel arama (YA).....	138
6.4.3.3	Budanmış dal-sınır yöntemi	139

6.4.3.4	Dal Sınır Yöntemi.....	139
6.4.3.5	Ayırgeç ayırıt yöntemi.....	139
6.4.4	Sezgisel kuralların etkinliği.....	140
6.4.5	Meta-sezgisel yöntemler.....	141
6.4.5.1	Genetik algoritma (GA).....	141
6.4.5.2	Tavlama benzetimi	142
6.4.5.3	Tabu arama (TA)	143
6.4.5.4	TA için notasyonlar	146
6.4.5.5	TA'nın uygulaması	149
6.4.5.6	TA için deterministik sayısal örnek 2.....	150
6.4.5.7	TA için bulanık sayısal örnek 3.....	160
7.	DETERMINİSTİK ve BULANIK FAALİYET SÜRELİ ÇOK MODLU KAYNAK KISITLI PROJE ÇİZELGELEME (B-ÇM-KKPÇP)	165
7.1	Problemin Tanımı ve Matematiksel Programlama Gösterimi.....	165
7.2	Belirsizlik Modelleme	169
7.3	Bulanık Aritmetik.....	171
7.4	İki Bulanık Sayının Kıyaslanması.....	172
7.4.1	Zayıf kıyaslama kuralı (WCR)	173
7.4.2	Güçlü kıyaslama kuralı (SCR).....	175
7.5	ÇM-KKPÇP'leri için Faaliyet Öncelik Kuralları.....	177
7.6	Sayısal Örnek 1	177
7.7	ÇM-KKPÇP'leri için Mod Seçimine Yönelik Öncelik Kuralları	179
7.8	Deterministik ve Bulanık ÇM-KKPÇP'lerinin Uygulaması.....	180
7.9	Çok Modlu durum için Sezgisel Yöntemler.....	181
7.10	ÇM-KKPÇM için Sayısal Örnek 2	181
7.11	Deterministik/Bulanık ÇM-KKPÇ Algoritması.....	183
7.12	Bulanık Sayısal Örnek 3.....	189
8.	UYGULAMA SONUÇLARI VE DEĞERLENDİRME.....	193
8.1	Veri Kümelerinin Kıyaslanmasında Uygulanan Yaklaşımlar	193
8.1.1	Etkinlik araştırması.....	193
8.1.2	Kullanılan yapay KKPÇP'leri ve özellikleri.....	195
8.1.3	Parametre kıyaslama.....	197
8.1.3.1	Proje şebekesinin karmaşıklığı	197
8.1.3.2	Kaynak faktörü	197
8.1.3.3	Kaynak gücü	198
8.1.4	En iyi çözümü bilinen problem ile test edilmesi	199
8.1.5	İstatistiksel analizlerin kullanılması	200
8.2	Uygulama Sonuçları ve Değerlendirme	201
8.3	Problem Büyüklüklerine göre Sonuçların Değerlendirilmesi	205
8.4	KKPÇ Problemlerinin Kıyaslanması	206
8.4.1	J30 KKPÇP veri kümesi.....	206
8.4.2	J60 KKPÇP veri kümesi.....	207
8.4.3	J90 KKPÇP veri kümesi.....	209
8.4.4	J120 KKPÇP veri kümesi.....	210
8.4.5	Patterson 110 KKPÇP veri kümesi.....	211
8.5	ÇM-KKPÇ Problemlerinin Kıyaslanması.....	212
8.6	Uygulama	217

8.7	Yapılan Çalışmalarda Sağlanan Sonuçlar	224
8.8	Tartışma ve Öneriler	225
KAYNAKLAR		237
EKLER.....		243
ÖZGEÇMİŞ.....		281

KISALTMALAR

AOA	: Faaliyetlerin Serimlerle Gösterimi
AON	: Faaliyetlerin Düzgünlerle Gösterimi
B-KKPC	: Bulanık Kaynak Kısıtlı Proje Çizelgeleme
B-KKPCP	: Bulanık Kaynak Kısıtlı Proje Çizelgeleme Problemi
B-ÇM-KKPC	: Bulanık Çok Modlu Kaynak Kısıtlı Proje Çizelgeleme
B-ÇM-KKPCP	: Bulanık Çok Modlu Kaynak Kısıtlı Proje Çizelgeleme Problemi
B-TM-KKPC	: Bulanık Tek Modlu Kaynak Kısıtlı Proje Çizelgeleme
B-TM-KKPCP	: Bulanık Tek Modlu Kaynak Kısıtlı Proje Çizelgeleme Problemi
BOA	: İki Bölge Yöntemi (bisector of area)
COA	: Ağırlık Merkezi Yöntemi (Center Of Area)
CPM	: Critical Path Method
ÇK	: Çok Kaynak Modelleri
ÇK-KKPC	: Çok Kaynaklı Kaynak Kısıtlı Proje Çizelgeleme
ÇM-KKPC	: Çok Modlu Kaynak Kısıtlı Proje Çizelgeleme
ÇM-KKPCP	: Çok Modlu Kaynak Kısıtlı Proje Çizelgeleme Problemi
ÇM-ÇP KKPC	: Çok Modlu Çok Projeli Kaynak Kısıtlı Proje Çizelgeleme
ÇM-ÇPKKPCP	: Çok Modlu Çok Projeli Kaynak Kısıtlı Proje Çizelgeleme Problemi
ÇP	: Çok Proje Modelleri
ÇP-KKPC	: Çok Projeli Kaynak Kısıtlı Proje Çizelgeleme
DP	: Doğrusal Programlama
E	: Evcil kanguru
EM	: Tuzak değer
EST	: En erken başlama zamanı
FF	: Bitiş-Bitiş tipi öncüllük
FIFO	: İlk giren ilk çıkar - First in First out
FS	: Bitiş-Başlangıç tipi öncüllük
GRD	: En büyük kaynak talebi (greatest resource demand)
GRPW	: En büyük konumsal ağırlık (greatest rank positional weight)
IHA-LBS	: Işık Hüzmesi Arama -Light Beam Search
KA	: Kanguru Algoritması
KKPC	: Kaynak Kısıtlı Proje Çizelgeleme
KKPCP	: Kaynak Kısıtlı Proje Çizelgeleme Problemi
KYM	: Kritik Yol Metodu
LCBA	: Yerel kısıt temelli analiz
LFT	: En geç bitirme zamanı (latest finish time)
LOM	: Maksimumun En Büyüğü (largest of maximum)
LST	: En geç başlama zamanı (latest start time)
MaxDM	: Maksimum Faaliyet Süreli Mod Seçim Kuralı- (Max duration mode)
MaxRDM	: Maksimum Kaynak Talebi Mod Seçim Kuralı- (Maximum Resource Demand mode) Kuralı
MDI	: Çok Projeli faaliyet süresi artışı- multiproject duration increase

MinDM	: Maksimum Faaliyet Süreli Mod Seçim Kuralı- (Min duration mode)
MinRM	: Minimum Kaynak Kullanımı Mod Seçim Kuralı- Minimum Resource Usage mode) Kuralı
MinRDM	: Minimum Kaynak Talebi Mod Seçim Kuralı- (Minimum Resource Demand mode) Rule
MinSlack	: En kısa boşluk süresi (minimum slack)
MOCO	: Çok Amaçlı Kombinatoral Optimizasyon
MOM	: Maksimumun Ortalaması (mean of maximum)
MPD	: Ortalama Proje Geçikmesi- mean project delay
MTS	: En çok sayıda toplam ardıl (most total successor)
NBD	: Net Bugünkü Değer
NŞD	: Net Şimdiki Değer
PBTM	: Pareto Benzetim Tavlaması Metodu
PERT	: Project Evaluation Review Techniques
PROGEN	: Proje Üreteci Program - Project Generator
PTZ	: Proje tamamlanma süresi
ÖK	: Öncelik kuralları
ROT	: Resource Over Time
RSM	: Kaynak çizelgeleme metodu (resource scheduling method)
SCR	: Güçlü kıyaslama kuralı
SF	: Başlangıç-Bitiş tipi öncüllük
SOM	: Maksimumun .En Küçüğü (smallest of maximum)
SPT	: En kısa işlem süresi (shortest processing time)
SS	: Başlangıç-Başlangıç tipi öncüllük
TA	: Tabu Arama
TDP	: Tamsayılı Doğrusal Programlama
TEARL	: Faaliyetin başlayabileceği en erken zaman değeri
TFIN	: Faaliyetin tamamlanma zamanı
TK	: Tek Kaynak Modelleri
TK-KKPC	: Tek Kaynaklı Kaynak Kısıtlı Proje Çizelgeleme
TM-KKPC	: Tek Modlu Kaynak Kısıtlı Proje Çizelgeleme
TM-KKPCP	: Tek Modlu Kaynak Kısıtlı Proje Çizelgeleme Problemi
TNOW	: İçinde bulunulan zaman
TP	: Tek Proje Modelleri
TP-KKPC	: Tek Projeli Kaynak Kısıtlı Proje Çizelgeleme
TSA	: Tabu Arama Algoritması - Tabu Search Algorithm
TSTART	: Gerçek başlama zamanı
ÜBS	: Üçgensel bulanık sayıdır
V	: Vahşi kanguru
YÇ	: Yaklaşık Çözüm
WCR	: Zayıf kıyaslama kuralı
WCS	: En kötü alternatif boşluk (worst case slack)

ÇİZELGE LİSTESİ

	<u>Sayfa</u>
Çizelge 5.1 : Örnek Projenin Bulanık Faaliyet Süreleri	102
Çizelge 5.2 : TDP ile Çözüm Yönteminin Kritik Yol Hesabı	104
Çizelge 5.3 : Bulanık Kritik Yol Analizi Sonuç Tablosu	105
Çizelge 5.4 : Bulanık Aritmetik Çözüm Yöntemi KYM Çözümü	106
Çizelge 5.5 : Örnek Proje Verileri	107
Çizelge 6.1 : Rot Algoritması Çözüm Algoritması	130
Çizelge 6.2 : Örnek problemin faaliyet süreleri ve kaynak ihtiyaçları	136
Çizelge 6.3 : Vahşi kanguru ile elde edilen çizelge	137
Çizelge 6.4 : Evcil kanguru ile elde edilen çizelge	137
Çizelge 6.5 : Sayısal Örnek Veri Yapısı	150
Çizelge 6.6 : Adım 1 Sonunda Elde Edilen Sonuçlar	151
Çizelge 6.7 : Minslack Algoritmasının İşleyişi	154
Çizelge 6.8 : ROT Algoritmasının İşleyişi	154
Çizelge 6.9 : Tabu İterasyonlarındaki Uygun (Olurlu) Komşuluklar	157
Çizelge 6.10 : Tabu İterasyonları	158
Çizelge 6.11 : Toplam Proje Çizelgeleme Maliyeti	159
Çizelge 6.12 : Yazılım proje çizelgeleme için örnek uygulama	161
Çizelge 6.13 : Bulanık KYM Sonuçları	162
Çizelge 6.14 : Önerilen TA İle Wang çalışmasının Kıyaslanması	163
Çizelge 6.15 : Örnek Çalışmanın Sonuçları	163
Çizelge 7.1 : Sayısal Örnek Verileri	178
Çizelge 7.2 : Sayısal Örnek Verileri	178
Çizelge 7.3 : J104_1.mm veri dosyası	181
Çizelge 7.4 : Örnek problemin Modları	184
Çizelge 7.5 : Minimum Faaliyet Süreli Mod Seçimi	184
Çizelge 7.6 : Maksimum Faaliyet Süreli Mod Seçimi	184
Çizelge 7.7 : Minimum Faaliyet Süreli Mod Seçimine göre KYM Sonuçları	186
Çizelge 7.8 : Maksimum Faaliyet Süreli Mod Seçimine göre KYM Sonuçları	186
Çizelge 7.9 : Minimum Faaliyet Süreli Mod Seçimine göre MinSlack Sonuçları	187
Çizelge 7.10 : Deterministik/Bulanık Bir MM-RCPS modelin Örnek Veri Kümesi	189
Çizelge 7.11 : Örnek Proje KYM/B-KYM Çözümü	190
Çizelge 7.12 : Örnek Proje MinDur Kuralına göre MinSlack Çözümü	191
Çizelge 7.13 : Örnek Proje Min/Maks Kaynak Talebi Kuralına Mod Seçimi	191
Çizelge 7.14 : Örnek Proje için Mod Seçim Kuralı	192
Çizelge 7.15 : Bulanık ÇM-KKPC Problemi Çözümü	192
Çizelge 8.1 : Literatürdeki Bazı Veri Kümeleri	203
Çizelge 8.2 : KKPC Veri Kümeleri Test Sonuçları	205
Çizelge 8.3 : J30 Veri Kümesi için Sezgisel Yöntemlerin Performansları	207
Çizelge 8.4 : J60 Veri Kümesi için Sezgisel Yöntemlerin Performansları	209

Çizelge 8.5	: J90 Veri Kümesi için Sezgisel Yöntemlerin Performansları.....	210
Çizelge 8.6	: J120 Veri Kümesi için Sezgisel Yöntemlerin Performansları.....	210
Çizelge 8.7	: 110 Patterson Problemleri Optimal Değerlerin diğer Sezgisel Yaklaşımlarıyla Bulunan Sonuçların Karşılaştırılması	211
Çizelge 8.8	: Patterson 110 Problemleri için Bulunan Sonuçlar.....	212
Çizelge 8.9	: J10, 12, 14, 16, 18, 20.mm için Algoritmanın Performansı	213
Çizelge 8.10	: Varyans Analizi Tanımlayıcı İstatistik	214
Çizelge 8.11	: Varyansların Homojenliği	214
Çizelge 8.12	: ANOVA Analizi Sonuçları.....	215
Çizelge 8.13	: J10.mm için Algoritmanın Performansı	218
Çizelge 8.14	: J12.mm için Algoritmanın Performansı	218
Çizelge 8.15	: J14.mm için Algoritmanın Performansı	219
Çizelge 8.16	: PSPLIB mm için Algoritmanın Performansı.....	223
Çizelge C.1	: TM-KKPÇP için Deterministik/Crisp Veri Kümelerinin Yapısı	249
Çizelge C.2	: ÇM-KKPÇP için Deterministik/Crisp Veri Kümelerinin Yapısı	249
Çizelge C.3	: B-KKPÇP için Bulanık Veri Kümelerinin Yapısı.....	250
Çizelge C.4	: B-ÇM-KKPÇP için Bulanık Veri Kümelerinin Yapısı	250
Çizelge C.5	: TM-KKPÇ Örnek Veri Yapısı.....	251
Çizelge C.6	: ÇM-KKPÇ Örnek Veri Yapısı	251
Çizelge H.1	: Uçak Yakıt Gaz Turbin Motoru Bakım Faaliyetleri Verileri	275
Çizelge H.2	: Uçak Yakıt Gaz Turbin Motoru Bakım Projesi B-KYM	276
Çizelge H.3	: Uçak Yakıt Gaz Turbin Motoru Bakım Projesi Kritik Faaliyetleri.	276
Çizelge I.1	: KKPÇ Problemleri için OPL-CPLEX Kodu	277
Çizelge İ.1	: ÇM-KKPÇ Problemleri için OPL-CPLEX Kodu.....	279

ŞEKİL LİSTESİ

	<u>Sayfa</u>
Şekil 2.1 : Faaliyet Süresi - Kaynak Tahsis İlişkisi	21
Şekil 2.2 : Faaliyet Süresi - Endirekt Maliyet İlişkisi.....	22
Şekil 2.3 : Proje Süresi - Maliyet İlişkisi.....	23
Şekil 2.4 : Kaynak - Yük Dağılımı	24
Şekil 2.5 : Sabit Kaynaklı TM-KKPC Örnek Problem	37
Şekil 2.6 : Sabit Kaynaklı TM-KKPC Örnek Probleminin Olurlu Çözüm	38
Şekil 2.7 : ÇM-KKPC için Örnek Problemin Olurlu Çözümleri.....	39
Şekil 2.8 : KKPC Problemlerinin Sınıflandırılması	41
Şekil 2.9 : Çözüm Uzaıı Büyüklüğü	42
Şekil 4.1 : Yeşil, siyah ve mavi bulanık renk kümeleri	65
Şekil 4.2 : Yaşlılar kümesinin kesin ve bulanık kümelerle gösterimi	66
Şekil 4.3 : Yaş genel uzayında tanımlı yaşlı kesin kümesi.....	67
Şekil 4.4 : Yaş uzayında tanımlı yaşlı bulanık kümesi.....	67
Şekil 4.5 : Klasik küme teorisi.....	69
Şekil 4.6 : Bulanık Küme	71
Şekil 4.7 : Bulanık küme teorisi	71
Şekil 4.8 : Üçgen Üyelik İşlevi.....	75
Şekil 4.9 : Yamuk Üyelik İşlevi	77
Şekil 4.10: $\tilde{A}$ kümesinin üyelik işlevi.....	78
Şekil 4.11: $\tilde{B}$ kümesinin üyelik işlevi.....	78
Şekil 4.12: Kesişim İşlemi.....	79
Şekil 4.13: Birleşim İşlemi	80
Şekil 4.14: Tümleme İşlemi.....	80
Şekil 4.15: Bulanık Denetim sistemi	82
Şekil 5.1 : Bulanık KYM Algoritması.....	101
Şekil 5.2 : Örnek proje şebekesi (Chanas ve Zelinski 2001).....	102
Şekil 5.2 : Örnek proje şebekesi	107
Şekil 6.1 : Örnek problem Şebeke Gösterimi	129
Şekil.6.2 : ROT Kriteri Faaliyet-Kaynak-Zaman Çizelgesi	131
Şekil 6.3 : Tabu Arama İçin Başlangıç Çözümün Bulunması	132
Şekil 6.4 : ROT ve MinSlack Algoritması Akış Diyagramı.....	133
Şekil 6.5 : Örnek problemin öncelik ilişkileri diyagramı	136
Şekil 6.6 : Evcil kanguru ile belirlenen çizelge yolu	137
Şekil 6.7 : TA Algoritmasının Genel Akış Şeması.....	146
Şekil 6.8 : KKPCP için Önerilen TA Algoritması.....	149
Şekil 6.9 : Örnek Şebeke Yapısı	150
Şekil 6.10: KYM Çizelgesi ve Kaynak Aşımı.....	152
Şekil 6.11: Minslack ve Rot Algoritmalarının Çözümü	155
Şekil 6.12: Yazılım proje şebekesi	161
Şekil 7.1 : 6 noktalı $\tilde{M}$ söbü bulanık sayı gösterimi	170
Şekil 7.2 : $\mu_{\tilde{A}}$ x yamuk bulanık sayı gösterimi	171

Şekil 7.3 :	Alan Takasını Temel Alan İki Bulanık Sayının Kıyaslanması	174
Şekil 7.4 :	Olurlu Çözümü olmayan Bulanık Çizelge Örneği	176
Şekil 7.5 :	K/B-ÇM-KKPCP için Önerilen TA Algoritması.....	180
Şekil 7.6 :	J104_1. mm örnek verisine ait Şebeke Diyagramı	182
Şekil 7.7 :	Bir MM-RCPS modelinin şebeke diyagramı	190
Şekil 8.1 :	J30.rcp için 1000 iterasyonlu GA ile Elde Edilen Sonuçlar	206
Şekil 8.2 :	J30.rcp için 1000 iterasyonlu DA ile Elde Edilen Sonuçlar	207
Şekil 8.3 :	J60.rcp için 1000 iterasyonlu GA ile Elde Edilen Sonuçlar	208
Şekil 8.4 :	J60.rcp için 1000 iterasyonlu DA ile Elde Edilen Sonuçlar	208
Şekil 8.5 :	Levene Testi Sonuçları	215
Şekil 8.6 :	ANOVA Analizi Sonucu Yöntemler Arasındaki Fark	216
Şekil 8.7 :	ANOVA Analizi Sonucu Yöntemler Farkın Kutu Gösterim	216
Şekil 8.8 :	ANOVA Analizi Grafik Sonuçlar	217
Şekil 8.9 :	Ana Arayüz Görünümü	224
Şekil A.1 :	Tezin Aşamaları.....	245
Şekil Ç.1 :	J30 Veri Seti için Optimal Sonuçlar Grafiği (1000 İterasyon).....	253
Şekil Ç.2 :	J30 Veri Seti için Optimal Sonuçlar Grafiği (1000 İterasyon).....	253
Şekil Ç.3 :	J30 Veri Seti için Optimal Sonuçlar Grafiği (1000 İterasyon).....	254
Şekil Ç.4 :	J30.rcp için Optimalden Yüzde Farkı Temel İstatistik Grafiği (1000 İterasyon).....	254
Şekil Ç.5 :	J30.rcp için Toplam Çizelge Maliyeti Grafiği (1000 İterasyon) ...	255
Şekil D.1 :	J60 Veri Seti için Optimal Sonuçlar Grafiği (1000 İterasyon).....	257
Şekil D.2 :	J60 Veri Seti için Optimal Sonuçlar Grafiği (1000 İterasyon).....	257
Şekil D.3 :	J60 Veri Seti için Optimal Sonuçlar Grafiği (1000 İterasyon).....	258
Şekil D.4 :	J60 Veri Seti için Optimalden Yüzde Farkı Temel İstatistik Grafiği (1000 İterasyon)	258
Şekil D.5 :	J60.rcp için Toplam Çizelge Maliyeti Grafiği (1000 İterasyon) ...	259
Şekil E.1 :	J90 Veri Seti için Optimal Sonuçlar Grafiği	261
Şekil E.2 :	J90 Veri Seti için Optimal Sonuçlar Grafiği	261
Şekil E.3 :	J90 Veri Seti için Optimal Sonuçlar Grafiği	262
Şekil E.4 :	J90 Veri Seti için Optimalden Yüzde Farkı Temel İstatistik Grafiği (100 İterasyon)	262
Şekil E.5 :	J90.rcp için Toplam Çizelge Maliyeti Grafiği (100 İterasyon)	263
Şekil F.1 :	Mindur Mod Seçim Kuralına göre J10.mm sonuçları	265
Şekil F.2 :	Maxdur Mod Seçim Kuralına göre J10.mm sonuçları	265
Şekil F.3 :	MinResUsage R3 Mod Seçim Kuralına göre J10.mm sonuçları...	265
Şekil F.4 :	MinResUsage R4 Mod Seçim Kuralına göre J10.mm sonuçları...	266
Şekil F.5 :	MaxResUsage R1 Mod Seçim Kuralına göre J10.mm sonuçları ..	266
Şekil F.6 :	MaxResUsage R2 Mod Seçim Kuralına göre J10.mm sonuçları ..	266
Şekil F.7 :	Mod 1 Seçim Kuralına göre J10.mm sonuçları	267
Şekil F.8 :	Mod 2 Seçim Kuralına göre J10.mm sonuçları	267
Şekil F.9 :	Mod 3 Seçim Kuralına göre J10.mm sonuçları	267
Şekil Ğ.1 :	Project Scheduling Progamının açılış Görünümü	271
Şekil Ğ.2 :	Dosya Yükleme Görünümü	271
Şekil Ğ.3 :	CPM Tabı Görünümü	271
Şekil Ğ.4 :	Min Slack Tabı Görünümü	272
Şekil Ğ.5 :	Tabu Tabı Görünümü	272
Şekil Ğ.6 :	ROT Tabı Görünümü	272
Şekil Ğ.7 :	MultiMode Tabı Görünümü	273
Şekil H.1 :	Uçak Yakıt Gaz Turbin Motoru Bakım Şebeke Diyagramı	275

SEMBOL LİSTESİ

a	: En iyimser süre,
A	: Çizelgelenen faaliyetler kümesi;
A_{ij}	: i ve j faaliyetleri arasındaki faaliyet.
{A}	: Serimler (arcs) kümesinin genel gösterimi
$\tilde{A}$	: A bulanık kümesi
$\bar{A}$	: Tümleme
b	: En kötümser süre,
c	: En olası süre,
C_{jr}	: j faaliyetinin r kaynağının birim zamanda kullanım maliyeti
d	: Faaliyet süresi
D	: Maksimum amaç değeri,
d_j	: “ j ” faaliyetinin süresi,
$\tilde{d}_i$	: $i \in A$ Faaliyetinin bulanık süresi
$\tilde{D}$	: Bulanık toplam faaliyet süresi,
$E \tilde{t}$	: $\tilde{t}$ zamanında uygun olan faaliyetler kümesi.
$E' \tilde{t}$	: $\tilde{\rho}_i \geq \tilde{t}$ için faaliyetlerin $E \tilde{t}$ kümesinin alt kümesidir.
EFT_j	: “ j ” faaliyetinin en erken bitiş zamanı,
$\tilde{f}_i$	: $i \in A$ Faaliyetinin bulanık tamamlanma süresi
FAT_{ij}	: A_{ij} ’nin bulanık faaliyet süresi.
$FCPM \ P_k$	: Bir proje şebekesinde P_k yolunun toplam aylık bulanık süresi.
$FEST_j$	: j seriminin en erken başlangıç bulanık süresi.
$FECT_j$	: j seriminin en erken tamamlanma bulanık süresi.
$FLST_j$	: j seriminin en geç başlangıç bulanık süresi.
$FLCT_j$	: j seriminin en geç tamamlanma bulanık süresi.
FTS_{ij}	: A_{ij} ’nin toplam aylık bulanık süre.
G=(N,A)	: Bir proje şebeke modeli notasyonu,
I $\tilde{t}$	: Sıralama indeksi,
j	: Faaliyet indisi $j = 1, \dots, J$
K	: Planlama periyodu
k_{jr}	: “ j ” faaliyetinin “ r ” kaynağından birim zaman kullanım miktarı,
K_r	: “ r ” yenilenebilir kaynağının birim zaman kullanım üst sınırı.
LFT_j	: “ j ” faaliyetinin en geç bitiş zamanı,
$\tilde{I}$	: Zayıf bulanık sayı kıyaslamayı temel alan B kümesinin en küçük değeri
m	: Mod indisi ($m=1, \dots, M_j$),
M	: Kaynak sayısı
M_j	: “ j ” faaliyetinin mod adedi,

n	: Faaliyet sayısı
N	: n düğümlerin kümesi,
NP_j	: j seriminin tüm öncül faaliyetleriyle bağlantılı tüm düğümlerin kümesi.
NR	: Yenilenemeyen kaynakların kümesi.
NS_j	: j seriminin tüm ardıl faaliyetleriyle bağlantılı tüm düğümlerin kümesi.
p_j	: Projede j faaliyetinin işlem süresi
P_j	: “ j ” faaliyetinin öncüllerinin kümesi,
PT	: Bir proje şebekesinin tüm yollarının kümesi.
PT_i	: i 'ninci yol.
r	: Kaynak tahsis miktarı
R	: Yenilenebilir kaynaklar kümesi,
r_{jk}	: J faaliyetinde kullanılan k kaynak tipinden herbir dönemde kullanılan miktar
$\Re_k^\rho$	: Sabit bir dönemde elde edilebilir sınırlı k tipi yenilenebilir kaynağının maksimum miktar
S	: p döneminde işlem gören faaliyetlerin kümesi
s_i	: Her bir i olayının bolluğu,
S_j	: j seriminin tüm ardıl faaliyetlerinin kümesi.
$S_{\tilde{t}}$	: $\tilde{s}_i \leq \tilde{t}$ için faaliyetler kümesi
$\tilde{s}_i$	: $i \in A$ Faaliyetinin bulanık başlangıç süresi
t	: Zaman indisi $t = 1, \dots, T$
$\tilde{t}$	: Bir faaliyetin kaynaklara tahsis edilmesi için kabul anlık şimdiki zaman
T_{Ai}	: En erken beklenen başlangıç zamanı,
T_{Li}	: En geç müsaade edilebilir başlangıç zamanı,
T_{ij}	: Faaliyet süreleri,
$\tilde{T}_{ij}$	: $i, j \in A$ bulanık sayıları için faaliyet süreleri,
$\{V\}$	: Düğümler (nodes) kümesinin genel gösterimi
$\tilde{\rho}_i$	: $i \in A$ Faaliyetinin bulanık hazırlık süresi
$\{\Phi\}$	: Düğümler ve serimleri ilişkilendiren ilişkiler kümesinin genel gösterimi
$X \subseteq Y$	: X kümesi Y kümesinin alt kümesidir
x	: Uzay kümesinin kesin küme elemanları,
X	: Uzay kümesi (kesin küme),
x_{ij}	: $i, j \in A$ faaliyetinde akış miktarını gösteren karar değişkeni,
$\mu_{\tilde{A}}^x$	: x kesin sayılarının $\tilde{A}$ bulanık kümesindeki üyelik derecesi,

BULANIK ÇOK MODLU KAYNAK KISITLI PROJE ÇİZELGELEME PROBLEMLERİNİN ÇÖZÜMÜ İÇİN MATEMATİKSEL BİR MODEL

ÖZET

Proje yönetimi hala yüksek oranda problemleri olan bir alan olma özelliğini sürdürmektedir. Çok sayıda büyük proje, bütçesini aşmakta, yavaş işlemekte veya diğer amaçları yerine getirmek için başarısız olmaktadır. Proje yönetiminde dünya lideri olan bir çok kurum-kuruluş ve özel teşebbüs tarafından bildirilen raporlara göre; projelerin % 30'u hiçbir zaman bitirilememekte ve yılda 75 milyar dolar boşa harcanmaktadır. Başlanan projelerin % 51'i, bütçelenenden % 189 daha fazlaya mal olurken; planlananın ise sadece % 74'ü kadar verim sağlamaktadır. Proje tipleri bakımından, bilgi teknolojileri, kamu kurum ve kuruluşları ile özel kurumların yeniden yapılanması amacıyla hazırlanan projeler yer almaktadır. Endüstriyel sektör bakımından projeler; finans, sigorta ve bankacılık endüstrisi, kamu yönetimi, bilgi teknolojileri, ulaşım, iletişim, imalat ve savunma sanayisi sektörlerinde yer alan endüstriyel kuruluşlar tarafından yoğun olarak kullanılmaktadır. Proje çizelgeleme, proje yönetiminin en önemli parçalarından biridir. Ancak günümüzde sürekli olarak ürün çeşitliliğinin artması ve ürünlerin kullanım sürelerinin kısalması, her seferinde farklı faaliyetleri ve süreçlerin kullanılmasını zorunlu kılmakla birlikte, belirsizliklerle dolu bir sürecin planlanmasını ve dolayısıyla da çizelgelenmesini zorunlu hale getirmektedir. Son yıllarda proje planlama, çizelgeleme ve kontrol yöntemi metotlarındaki sürekli gelişmeler, organizasyonlar için vazgeçilmez bir metodoloji olan proje yönetim tekniklerinin kullanımını zorunlu kılmıştır. Böylece zaman-maliyet-kaynak üçlüsünün olabildiğince rasyonel ve verimli bir şekilde ele alınmış, bu da giderek azalmakta olan kıt kaynakların daha verimli kullanılmasına imkan vermiştir.

Proje yönetimi yaklaşımlarının başarılı olarak uygulanması için; var olan zaman ve maliyet belirsizliklerinin uygun şekilde yönetilmesi ve projenin hedeflenen sürede ve maliyetle tamamlanması gereklidir. Proje yönetiminde klasik yaklaşımla mevcut belirsizliğin modellenmesi olasılık teorisine dayanmaktadır; ancak belirsizliğin modellenmesine olanak tanıyan yeni eğilim Bulanık Küme Teorisidir. Bulanık mantık; düzensizliğin, belirsizliğin, eksik ve çelişen bilgilerin olduğu, hem gerçeğin hem de olasılığın bir parçasının bulunduğu bir çevrede, kısacası mükemmel bilginin olmadığı bir çevrede, muhakeme etme ve mantıklı karar verme yeteneğidir. Modelin çözümüyle ilgili olarak literatürde, bulanık proje çizelgeleme probleminin çözümünde sezgisel yöntemler etkin şekilde kullanılmaktadır. Bu çalışmada ise literatürde yer alan pek çok öncelik kuralına bağlı sezgisel yöntemlerden sadece ikisi, MinSlack ve ROT sezgisel kuralları önerilmektedir. Bu yöntemlerin başlıca avantajı; kolay anlaşılabilir ve uygulanabilir olması ve hızlı şekilde çözüme ulaşabilmesidir. Ancak, elde edilen sonuçların optimal sonuçtan uzaklığı hakkında bulunan belirsizlikler sebebiyle farklı sezgisel yöntemler de araştırılmaktadır. Modelin çözümünde kullanılabilecek alternatif çözüm yöntemleri incelendiğinde, tabu arama algoritması değerlendirilmiştir. Tabu arama algoritmasının problemlerin çözümünde

hem deterministik hem de bulanık ortamda etkin bir şekilde kullanılabildiği ortaya çıkarılmıştır. Önerilen yöntem farklı bulanık sayı üyelik fonksiyonları kullanılarak, yenilenebilen ve yenilenemeyen proje kaynaklarının mevcut olduğu durumlarda ve farklı proje teknik karmaşıklığı sayısal analizlerle çalıştırılarak test edilmiştir. Elde edilen toplam proje süresi, toplam proje maliyeti, proje faaliyeti çizelgesi ve kaynak kullanımlarına dair model sonuçları, proje gerçekleştirmeleri ile karşılaştırılmış ve sonuçlar arasında büyük bir tutarlılık olduğu belirlenmiştir. Yapılan çalışma ile projelerde mevcut bulunan zaman ve maliyet risklerinin modellenmesinde kullanılabilecek matematiksel bir model hazırlanmış, oluşturulan model için çözüm yöntemi oluşturulmuş, sayısal analizler ve kıyaslamalı proje uygulaması yapılarak ele alınan KKPÇ probleminin etkin şekilde çözümünü gerçekleştiren ve gerçek projelerde kullanılabilecek bir yöntem oluşturulmuştur.

Genel ifadesiyle Proje Çizelgeleme Problemleri ile özel olarak bulanık faaliyet süreli çok modlu, kaynak kısıtlı proje çizelgeleme problemleri (B-ÇM-KKPÇP) üzerinde çalışılmıştır. Proje çizelgeleme problemleri içinde ileri konular arasında sayılan bu problemi ele alırken, basitten zora doğru bir yol izlenmiş, her bir problem hem deterministik/kesin (deterministic/crisp) hem de belirsizliklerle dolu süreçlerin çizelgelenmesine olanak tanıyan bulanık faaliyet sürelerinden oluşan projeler ele alınarak incelenmiştir. Proje çizelgeleme problemlerinin çözümü için Microsoft Visual C# 2008 programlama dili kullanılarak “Proje Çizelgeleme Programı” olarak isimlendirdiğimiz bir paket program üzerinde sayısal örnekler üzerinde test ve kıyaslamalar yapılmıştır. Kısıtlı kaynak, deterministik ve belirsizlik koşulları altında faaliyet süreleri minimize edilerek projenin çizelgelenmesi yapılmakta, proje çizelgeleme maliyetleri hesaplanmakta ve projenin toplam tamamlanma süresi ve kritik faaliyetler belirlenmektedir.

Faaliyet sürelerinin yamuk bulanık sayılardan oluştuğu ve proje şebeke yapısının AON olarak gösterildiği model kullanılmıştır. Bulanık faaliyet süreli projelere ait, en erken tamamlanma süresi, en geç tamamlanma süresi ve aylak süreler belirlenmiştir. Önerilen yöntem, bulanık üyelik fonksiyonun yapısının bilinmesine ihtiyaç duyulmaması bakımından oldukça basittir. Oluşturulan optimizasyon modelinin mevcut hesaplama karmaşıklığı sebebiyle başlangıç çözümü sezgisel algoritmalarından MinSlack ve ROT öncelik kuralları kullanılarak oluşturulmuştur. Öncelik kuralı ile oluşturulan çözüm Tabu arama için başlangıç çözüm olarak kullanılmıştır. Tabu Arama algoritması ile çeşitli büyüklükte ki problem kümeleri çözülmüştür. Problem aşama aşama ele alınmış ve kolaydan zora doğru Bulanık Çok Modlu Kaynak Kısıtlı Proje Çizelgeleme Problemleri (B-ÇM-KKPÇP) için yeni bir matematiksel model oluşturulmuştur.

Çizelgeleme problemlerinin, NP-Zor (Polinom olmayan) problemler olması nedeniyle, çözümlerinde analitik yaklaşımlar yerine daha çok sezgisel algoritmaların kullanımı tercih edilmiştir. Tabu algoritması farklı boyutlardaki bu tarz çizelgeleme problemlerinde etkili olmakta ve iyi çözümlere ulaşmada başarılı sonuçlar vermektedir. Çizelgeleme problemlerinde kaynak kısıtları nedeniyle bütün işlerin yapılması gerektiği zamanlarda gerçekleştirilmeleri sağlanamıyorsa; hangi işlerin öne alınacağı, hangilerinin geciktirilebileceği ve hangilerinin de geciktirilemeyeceğinin kararını almakta doğru kriterlere göre davranmak gereklidir. Kaynak kısıtlı proje çizelgeleme problemleri, karmaşık yapısı gereği çözümü zor (NP-hard) problemler arasında yer almaktadır. Çözümü zor olan bu tür problemlerde, kesin çözüm veren bir algoritmanın hazırlanması oldukça zordur. Bu çalışmada, NP zor kapsamında, bulanık proje çizelgeleme problemlerinin çözümlerinde

kullanılabilecek öncelik kurallı algoritmalar ve sezgisel/meta-sezgisel yöntemler incelenmiştir. Ayrıca son yıllarda kullanılan ve başarılı sonuçlar veren Paralel Kanguru Algoritması metasezgisel yöntemi ile örnek bir bulanık proje çizelgeleme problemi çözülmüştür.

Bulanık proje çizelgeleme problemlerinin faaliyet sürelerinin bulanık sayılardan oluştuğu bir proje şebekesinde kritik yol analizine bulanık aritmetik yaklaşım ile doğrusal programlama yöntemi kıyaslanmıştır. Bu fikir bulanık sayıları sıralama (ranking method) ve doğrusal programlama formülasyonunu temel almaktadır. Bulanık kritik yol problemi, doğrusallık, eklenebilirlik özelliklerini temel alan ve amaç fonksiyonun bulanık katsayılı bir doğrusal programlama modeli olarak formülize edilmiş, Yager sıralama yöntemi geleneksel akış çizgisi çözüm yöntemleri kullanılarak çözümlenebilen klasik doğrusal programlama modellerinin, bulanık doğrusal programlama modeline uyarlanmıştır. Algoritma; literatürde bulunan çeşitli veri kümeleri üzerinde, toplam yaklaşık 27,000 çizelgeleme problemleri ile test edilmiştir. 27,000 problemin her biri, farklı iterasyonlarda olmak üzere test edilerek, iterasyon sayısının etkisi araştırılmaya çalışılmıştır. Algortimanın performansını ölçmek ve değerlendirmek amacıyla; her bir veri kümesinin ortalama standart sapması hesaplanarak, literatürde yayınlanmış olan diğer çalışmalarla karşılaştırılmaktadır. Geliştirilen algoritma, basit problemlerde evrensel (global) sonuçlar verirken, karmaşık problemlerde genellikle yerel sonuçlar üretebilmektedir.

Proje yönetimi yaklaşımlarının başarılı olarak uygulanması için; var olan zaman ve maliyet belirsizliklerinin uygun şekilde yönetilmesi ve projenin hedeflenen sürede ve maliyetle tamamlanması gereklidir. Projedeki zaman ve maliyet belirsizliklerinin uzman görüşüyle, bulanık üyelik fonksiyonları kullanılarak modellenmesi, toplam proje süresi ile toplam proje maliyetinin kaynak kısıtlarına bağlı olarak matematiksel modelinin oluşturulması ve oluşturulan modelin etkin şekilde çözülmesi amaçlanmıştır. Deterministik tekniğin daha çok kullanılmasının önündeki engel ise, gerçek hayatta kesin bilgiye ulaşmanın zorluğu ve maliyetidir. Bulanık küme teorisinin popüleritesinin gün geçtikçe artmasının sebebi; gerçek projelerin modellenmesinde daha başarılı olması ve özellikle faaliyet sürelerinin istatistiksel olarak hesaplanması için yeterli veri olmaması durumunda bile geniş bir uygulama sahası bulunmasıdır. Bulanık değişkenlerin bu alanda kullanılmasıyla, doğrusal programlama tekniği çok daha az maliyetle, gerçek hayat problemlerinin çözümünde kullanılabilmektedir. Kesin ve belirsizlik ortamında ele alınan kısıtlı kaynaklar altında B/K-ÇM-KKPÇ ve B/K-TM-KKPÇP'lerinin modellenmesinde tamsayılı matematiksel modelleme tekniği kullanılmıştır. Modelin amaç fonksiyonunda toplam proje çizelgeleme maliyeti bulunmuş ve toplam proje süresi minimize edilmiştir. Modelde farklı kaynak tipleri kullanılmış ve kaynak kullanım maliyetlerinin dönemler itibarıyla değişkenlik göstermediği varsayımına göre modelleme yapılmıştır. Her dönemde projede uygulanacak yenilenebilen kaynakların birim zamanda kullanılan miktarlarının kısıtlı ve sınırlı olduğu ve yenilenemeyen kaynakların ise proje süresince toplam miktarının kısıtlı olduğu varsayımı yapılmıştır. Hesaplama sonuçlarıyla belirlenen performans kriterleri karşılaştırılarak bilinen sezgisel yaklaşımların tabu arama yöntemiyle birlikte çok daha hızlı, optimal ve optimale yakın sonuçlar verdiği belirlenmiştir.

Anahtar Kelimeler : Kaynak Kısıtlı Proje Çizelgeleme Problemleri, Bulanık Kümeler, Proje Yönetimi, Sezgisel ve Tabu Arama Algoritması.

A MATHEMATICAL MODEL FOR THE SOLUTION OF THE FUZZY MULTI MODE RESOURCE-CONSTRAINED PROJECT SCHEDULING PROBLEMS

SUMMARY

In this study to understand the Resource Constrained Project Scheduling Problems in the Project Management possible explanations are examine. Because classic planning and scheduling methods fail to keep up in such an environment. When the project is scheduled with a given set of resources, it is difficult to find the optimal solution. Resource-constrained scheduling problems are generally Np-hard. A higher level heuristic procedure “tabu search” is proposed to provide good solutions to resource-constrained, determinated activitiy duration project scheduling problems. For the determininistic problems, most of the optimal schedules of the test projects investigated are found. Computaional results are presented which establish the superiority of tabu search over the existing heuristic algorithms. Good agreement was found between theoretical and experimental results. Moreover, An example discussed in some previous studies illustrates that the proposed approach is able to find the most critical path. The some example procedures, the project model and experimental work are explained.

We consider the fuzzy and crisp multi mode resource-constrained project scheduling problem (F/C-MM-RCPSP) with multiple execution modes for each activity. In this study, a software package named as “Project Scheduling Programming” was developed and its performance was tested on some sample projects from the PSPLib data sets. Microsoft Visual C# 2008 was used during the preparation of software. Project planning under constrained resource, deterministic and uncertainty conditions can be performed by minimizing activity; total project duration and total scheduling cost together with critical path can be determined by Project Scheduling software.

We propose a mathematical model to deal with project scheduling problem under vagueness and presenting the framework of a heuristic approach to fuzzy Resource Constrained Project Scheduling Problem (F-RCPSP) using heuristic and meta-heuristic scheduling methods. We use trapezoidal fuzzy numbers (TraFN), for activity times and Activity-on-Nodes (AON) representation in this method. We compute several project characteristics such as earliest, latest and slack times in term of TraFNs. Our approach is very simple to apply, and it does not require knowing the explicit form of the membership functions of the fuzzy activity times. We first identify two typical activity priority rules such as Resource over Time (ROT) and Minimum Slack (Minslack), which are used in the F-RCPS problem. These rules are used in the initial solution Taboo Search (TS) method. We improved the Taboo Search Algorithm Method to solve Fuzzy RCPSP. Our objective is to test the performance of these rules and meta-heuristic method in minimizing the project completion time for the F-RCPS problems. Through a numerical example, calculation steps in this method and the results are illustrated. The computational experiment shows that the performance of the proposed TS is better than other algorithms appearing in the literature.

Since scheduling is a NP-hard problem, heuristic algorithms is preferred in the solutions of scheduling problems instead of analytical approaches. Taboo algorithm is quite efficient in scheduling problems of different sizes and gives successful results in obtaining feasible and better solutions. If it is not possible to perform the tasks in the time they are supposed to be conducted because of resource constraints in scheduling problems, it is necessary to behave based on right criteria in deciding which tasks could be moved forward, which tasks could be postponed or which tasks can not be delayed.

This thesis presents a taboo algorithm for the resource constrained project scheduling problem using Microsoft Visual C# 2008. Resource constrained project scheduling is one of the NP-hard classes because of the complexity of their combinatorial nature. Any exact algorithm are difficult to handle such a problem. The heuristic algorithm generates parameterized active schedules and selects the best one which has minimum project completion time among the generated schedules. The representation of the problem is based on the minslack and Rot priority rule representation. The schedule is constructed using selected heuristic priority rule.

In this study develops a fuzzy algebra approach to critical path analysis in a project network with activity times being fuzzy numbers. The idea is based on the linear programming (LP) formulation and fuzzy number ranking method. The fuzzy critical path problem is formulated as an LP model with fuzzy coefficients of the objective function, and then on the basis of properties of linearity and additivity, the Yager's ranking method is adopted to transform the fuzzy LP formulation to the crisp one which can be solved by using the conventional streamlined solution methods. An example discussed in some previous studies illustrates that the proposed approach is able to find the most critical path, which is proved to be the same as that derived from an exhausted comparison of all possible paths. The proposed approach is very simple to apply, and it is not require knowing the explicit form of the membership functions of the fuzzy activity times.

This study use a resource allocation model to solve the project scheduling problem under fuzzy environment. It employs the uses of kangaroo algorithms and the fuzzy set theory to develop the resource-constrained project scheduling model under uncertainty, and gives an application of such fuzzy RCPSP analysis method. The work is some attempt at an interesting issue of proposing a mathematical model to deal with project scheduling problem under vagueness and presenting the framework of a heuristic approach to FRCPS using a fuzzy parallel kangaroo and minslack scheduling method. The objective of this work is to minimize project planning time with resource limitations and to show how to create a plan with critical path analyses under fuzzy environment. We use trapezoidal fuzzy numbers for activity times and Activity-on-Arcs (AOA) representation in fuzzy critical path method (FCPM). Resource-Constrained Project Scheduling (RCPS) is often a challenging issue in practice, due to its combinatorial nature and uncertainty. This approach can handle fuzzy numbers which commonly coexist in many realistic RCPS problems. Using this method for scheduling problems makes it possible to plan more efficiently.

Priorities of activities of scheduling problems are given by two types such as processors/successors of any activities. The algorithm works for two types of scheduling problems. So there is no need any conversion process for successors to processors (or vice versa) by human being. The algorithm was tested on the variously sets of standard problems from the literature. Each set has many project

scheduling problems. Total number of scheduling problems is 27000. Each of these scheduling problems are tested to observe the effects of different iteration number.

Many studies on the resource-constrained project scheduling problem (RCPSP) have been published, but literature further considering multi-mode issue often occurring in the real world is rather scarce. The fuzzy/crisp multi-mode resource-constrained project scheduling problem (F/C-MM-RCPSP) is a generalized case of the RCPSP. The MM-RCPSP consists of a number of projects, defined as collections of activities performed in one of several ways under given precedence relationships and limited amounts of various types of resources. The problem of scheduling activities under resource and precedence restrictions with the objective of minimizing the project duration (makespan) is referred to as resource-constrained project scheduling problem (RCPSP) in the literature. Project scheduling problem (PSP) as the core content of project management has been extensively studied in recent decades, but its complexity, dynamic randomness, fuzziness and multi-objective attributes have not yet been solved by a systematic method or theory. There is a big gap between the theoretical research and practical application of PSP. This work discusses the F/C-MM-RCPSP problems which exist widely in construction, software development, aircraft and ship manufacturing, designing a computer system and planning a military invasion, introducing a new product, and other single or small batch production mode of the enterprise. The resource constraints refer to limited renewable and non-renewable resources such as manpower, materials and machines which are necessary for carrying out the project activities. It can be clearly seen that the RCPSP is a generalization of the static job shop, flow shop, assembly line balancing, related scheduling problem and hence belongs to the class of NP hard problems.

A fuzzy taboo search algorithm (TSA) is first developed to solve the problem in a straight forward and robust manner. To improve its performance, fuzzy TSA avoids revisit of previous search areas, thus saving unnecessary search time for the optimal solution. The developed Fuzzy TSA can be applied to practical project scheduling problems of large sizes. A study has been conducted to develop a heuristic solution methodology for the fuzzy MM-RCPSP with generalized precedence relations. The MM-RCPSP problem was solved using a priority-based heuristic for heuristically solving the MM-RCPSP for a fixed mode assignment. Results have indicated that the taboo search procedure is capable of outperforming all other heuristic approaches in terms of solving MM-RCPSP problems. The objective is to determine a mode and a start time for each activity so that all constraints are obeyed and the project duration and the scheduling cost are minimized. Project scheduling problems of this type occur, e.g. in process industries. The solution approach is an integration approach where the modes and start times are determined simultaneously. Within an experimental performance analysis this procedure is compared with existing solution procedures. The algorithms are computationally compared, the results are analyzed and discussed, and some conclusions are given. Finally, the solutions obtained by the proposed TSA having the best activity- and mode- priority rule combination are compared on the test problems with those obtained by the proposed TSA. Finally, conclusions are drawn from the computational results. This study reports on a new solution approach for the well-known MM-RCPSP. This problem type aims at selecting a single activity mode from a set of available modes in order to construct precedence and a (renewable and non-renewable) resource feasible project schedule with a minimal makespan. The problem type is known to be NP-hard and has been solved using various exact as well as meta-heuristic procedures.

In this study, a real case of a project schedule is considered as a fuzzy RCPS problem and F-MM-RCPS's. Most of these activity duration times have to be estimated by the DM by using his/her knowledge and experience. This method is developed based on a number of assumptions and definitions in fuzzy sets and project scheduling. In the fuzzy project network considered in this work, we assume that the durations of activities are represented by trapezoidal fuzzy numbers (TraFNs). As activity durations are estimated by human experts, sometimes under unique circumstances, project management is often confronted with vague and imprecise judgmental statements. For example, the duration of an activity may be expressed as “more than 2 days and less than 5 days”. In those situations, which involve imprecision rather than uncertainty, the fuzzy scheduling literature recommends the use of fuzzy numbers for modeling activity durations, rather than stochastic variables. The project characteristics such as fuzzy earliest times and fuzzy project completion time are calculated as TraFNs by forward pass and backward pass in fuzzy environment. In this study we proposed a modified TSA to solve the MM-RCPS under fuzziness. The proposed approach aims to minimize the total project completion time. An extensive experiment was conducted. The advocates of the fuzzy activity duration approach argue that probability distributions for the activity durations are unknown due to the lack of historical data. In this study, we propose the TSA to be used for solving a large variety of combinatorial optimization problems, namely the MM-RCPS using integer values and fuzziness.

The model of the problem has been presented, and practical examples have been discussed. An application of a local search metaheuristics—taboo search for the considered problem has been described. This metaheuristics has been compared with SA, PSA and DE algorithms of a computational experiment performed on a set of instances, obtained from standard test problems constructed by the ProGen project generator. Our implementation of TSA is one of the best metaheuristics approaches to the classical MM-RCPS. The resolution of the MM-RCPS by TS algorithm is proposed in this study. We compared our results with those obtained through the simulated annealing algorithm of Bouleimen and Lecocq and the particle swarm optimization of Jarboui et al. and differential evolution algorithm of Weglarz. We could search for another search strategy, more efficient for the MM-RCPS. Summarizing, the proposed TS implementation seems to be quite a promising algorithm for solving the defined problem, evidently outperforming simpler methods: SA, PSA and DE algorithms. However, a more advanced approach, e.g. another metaheuristics, is needed to compare with taboo search in order to truly examine the efficiency of TS. The proposed method using taboo search makes it possible to improve fuzzy project scheduling under resource constraints with multi mode models. The proposed model can effectively provide the optimal fuzzy profiles of project duration and resource amounts under the constraint of limited resources. The computational experiment shows that the performance of the proposed TS is better than the other algorithms appearing in the literature, there is no significant difference between the proposed comparison method and weak-strong comparison rules on the performance of the algorithm. A taboo algorithm proposed to search a better solution of MM-RCMPSP is one of the objectives of the study, since so far there seems no good heuristic algorithm in public literature.

In this study, the multi-mode resource-constrained project scheduling problem with fuzziness has been considered. The model of the problem has been presented, and practical examples have been discussed. An application of a local search

metaheuristics—taboo search for the considered problem has been described. The results of the computational experiment show that taboo search seems to be an efficient algorithm for solving the considered problem, clearly outperforming simple search algorithms: multi-start iterative improvement and random sampling. Nevertheless, for better evaluation of the proposed TS implementation, it has to be compared with more advanced methods, as well as with optimal solutions (or at least lower bounds) for smaller numbers of activities. In the future we plan to implement other metaheuristics, e.g. simulated annealing, genetic algorithms, parallel kangaroo algorithm and orienteering algorithm or scatter search, to the defined problem. Our implementation of TSA is still one of the best metaheuristics approaches to the classical MM-RCPSP. It would be interesting to examine its efficiency for the MM-RCPSP and compare it with taboo search. It would be also useful to construct a branch-and-bound procedure in order to find optimal solutions. Comparing the results of the metaheuristics with optimal solutions will allow the full evaluation of their efficiency. Also other criteria can be taken into account from among which financial criteria, like the maximization of the net present value, seem to have good practical justification. Practical F-MM-RCPS is often a complex problem due to its NP-hardness and uncertainty. Its combinatorial nature is difficult to handle using the existing exact algorithms. While heuristic approaches appear to be attractive, they give reasonable practical solutions; traditional heuristic approaches suffer the major shortcoming of being unable to handle such uncertainty.

This study presents a fuzzy MM-RCPS heuristic approach that incorporates the fuzzy set theory to model the uncertain activity duration times estimated by the DM. To obtain a good solution for the minimization of the project completion time, a group of priority rules is employed in a fuzzy parallel scheduling method. This approach is simple and straight forward in practical applications. Therefore, this study provides the framework of a heuristic approach for solving F-MM-RCPS and deterministic/crisp MM-RCPS problems involving uncertain activity duration times modeled by others fuzzy numbers. There are still several unsolved issues in fuzzy CPM/PERT applications: The operation time of each activity is seldom available even using fuzzy numbers in any project. There are many ranking methods of fuzzy numbers. However a suitable method to compute the earliest starting times of each activity in project network has not been developed yet. Fuzzy subtraction method to compute the fuzzy latest starting times may get the unreasonable negative values of times. It is worth developing a project scheduling risk index (PSRI) to assist the decision maker to evaluate scheduling risk while they bid a potential project. A taboo algorithm proposed to search a better solution of MM-RCPSP is one of the objectives of the study, since so far there seems no good heuristic algorithm in public literature. Hence, a simple taboo algorithm is initialized; more efficient and robust TSA is left for future study. From the computational results, the effectiveness and performance of the proposed taboo algorithm is validated. To measure and evaluate the performance of the algorithm; the average deviation of each problem set is calculated to compare the algorithm with other approaches in the literature. The computational results are reported. Computational results relatively validate the effectiveness of the generated algorithm. Generally, the algorithm gives global results for simple problems and gives local results for complex problems.

Key Words : Fuzzy set theory, Fuzzy resource-constrained project scheduling, Heuristic and Taboo search algorithm.

1. GİRİŞ

Bu çalışmada bulanık faaliyet süreli, çok modlu, kaynak kısıtlı proje çizelgeleme problemleri (B-ÇM-KKPCP) üzerinde çalışılmıştır. Proje çizelgeleme problemleri içinde ileri konular arasında sayılan bu problemi ele alırken, basitten zora doğru bir yol izlenmiş, her bir problem hem deterministik/kesin (deterministic/crisp) hem de belirsizliklerle dolu süreçlerin çizelgelenmesine olanak tanıyan bulanık faaliyet sürelerinden oluşan projeler incelenmiştir. Ayrıca proje çizelgeleme maliyetlerinin modele dahil edilmesiyle, proje maliyeti ile proje süresi arasındaki ilişki incelenmiştir.

Proje yönetimi yaklaşımlarının başarılı olarak uygulanması için; var olan zaman ve maliyet belirsizliklerinin uygun şekilde yönetilmesi ve projenin hedeflenen sürede ve maliyetle tamamlanması gereklidir. Proje yönetiminde klasik yaklaşımla mevcut belirsizliğin modellenmesi olasılık teorisine dayanmaktadır; ancak belirsizliğin modellenmesine olanak tanıyan yeni eğilim Bulanık Küme Teorisidir. Bu teori, proje yönetiminin temel dayanak noktalarından biri olmaya başlamıştır. Projedeki zaman ve maliyet belirsizliklerinin uzman görüşüne dayalı bulanık üyelik fonksiyonları kullanılarak modellenmesi, toplam proje süresi ile toplam proje maliyetinin kaynak kısıtlarına bağlı olarak matematiksel modelinin oluşturulması ve oluşturulan modelin etkin şekilde çözülmesi amaçlanmıştır.

Bulanık küme teorisinin popüleritesinin gün geçtikçe artmasının sebebi, gerçek projelerin modellenmesinde daha başarılı olması ve özellikle faaliyet sürelerinin istatistiksel olarak hesaplanması için yeterli veri olmaması durumunda bile geniş bir uygulama sahası bulunmasıdır.

Ele alınan B-ÇM-KKPCP probleminin modellenmesinde karışık tamsayılı matematiksel modelleme tekniği kullanılmıştır. Modelin amaç fonksiyonunda toplam proje çizelgeleme maliyeti bulunmuş ve toplam proje süresi minimize edilmiştir. Modelde farklı kaynak tipleri kullanılmış ve kaynak kullanım maliyetlerinin dönemler itibarıyla değişkenlik göstermediği varsayımına göre modelleme

yapılmıştır. Her dönemde projede uygulanacak kaynakların birim zamanda kullanılacak miktarlarının kısıtlı ve sınırlı olduğu varsayılmıştır.

Son yıllarda proje planlama, çizelgeleme ve kontrol yöntemi metotlarındaki sürekli gelişmeler, organizasyonlar için vazgeçilmez bir metodoloji olan proje yönetim tekniklerinin kullanımını zorunlu kılmıştır. Böylece zaman-maliyet-kaynak üçlüsünün olabildiğince rasyonel ve verimli bir şekilde ele alınmış, bu da giderek azalmakta olan kıt kaynakların daha verimli kullanılmasına imkan vermiştir.

Günümüzde Kritik Yol Metodu (KYM- Critical Path Method-CPM), Proje Geliştirme ve Gözden Geçirme Tekniği (PERT- Project Evaluation Review Techniques) ve simülasyon gibi kantitatif serim teknikleri, karmaşık ve büyük ölçekli projelerin yönetiminde; yöneticilere planlama, çizelgeleme ve denetim konularında büyük yararlar sağlamakta ve dolayısıyla yaygın olarak kullanılmaktadır. Kaynak Kısıtlı Proje Çizelgeleme (KKPÇ) yaklaşımları organizasyonların kıt kaynakları daha verimli kullanarak, hedeflerine ulaşmasına ve dünya firmalarıyla rekabet edebilmek için modern yönetim felsefelerini yakından takip etmesine olanak tanımakta ve bu konuda farklı bir yaklaşım getirmektedir. Böylece çözümü zor problemler için işletmelere yeni bir metod sunmaktadır. Proje fizibilite çalışmasında; faaliyetlerin ve sürelerin belirlenmesi; nakit, insangücü, makine gibi kısıtlı kaynakların varlığında proje faaliyetlerin çizelgelenmesi; projenin öngörülen zaman ve maliyetle bitirilmesi için faaliyetlerin eş güdümü ve denetimi çoğu projede ancak bilgisayar desteği ile başarılı bir şekilde yapılabilmektedir.

Günümüzde proje yönetimi daha da gelişmiş ve profesyonel becerinin uygulama sahası, akademik araştırma ve tezler için oldukça rağbet gören bir alan olmuştur. Oluşumundan tamamlanışına kadar konuyla ilgili çok sayıda metot ve teknik, yönetsel projelerin bütün özelliklerini kapsayacak şekilde geliştirilmiştir. Bununla birlikte, proje yönetimi hala yüksek oranda problemleri olan bir alan olma özelliğini sürdürmektedir. Çok sayıda büyük proje, bütçesini aşmakta, yavaş işlemekte veya diğer amaçları yerine getirmek için başarısız olmaktadır.

Proje yönetiminde dünya lideri olan bir çok kurum-kuruluş ve özel teşebbüs tarafından onaylanan raporlara göre; projelerin % 30'u hiçbir zaman bitirilememekte ve yılda 75 milyar dolar boşa harcanmaktadır. Başlanan projelerin % 51'i, bütçelenenden % 189 daha fazlaya mal olurken; planlananın ise sadece % 74'ü kadar

verim sağlamaktadır (White 2002). Dolayısıyla bu ve benzeri nedenlerle; proje yönetimini kolaylaştıran metot, araç ve tekniklere, planlama ve bütçelemeye kaynak israfına son vermeye yönelik kesin çözümlere ihtiyaç duyulmaktadır.

Yapılan anket çalışmaları sonucu elde edilen verilere bakıldığında; proje tipleri bakımından baş sırayı, bilgi teknolojileri ile kamu kurum ve kuruluşları ile özel kurumların yeniden yapılanması amacıyla hazırlanan projeler almaktadır. Endüstriyel sektör bakımından projeler; finans, sigorta ve bankacılık endüstrisi, kamu yönetimi, bilgi teknolojileri, ulaşım, iletişim, imalat ve savunma sanayisi sektörlerinde yer alan endüstriyel kuruluşlar tarafından yoğun olarak kullanılmaktadır.

Proje çizelgeleme, proje yönetiminin en önemli parçalarından biridir. Ancak günümüzde sürekli olarak ürün çeşitliliğinin artması ve ürünlerin kullanım sürelerinin kısalması, her seferinde farklı faaliyetleri ve süreçlerin kullanılmasını zorunlu kılmakta birlikte bu da belirsizliklerle dolu bir sürecin planlanmasını ve dolayısıyla da çizelgelenmesini gerektirmektedir.

Çizelgeleme, verilen zaman periyotlarında kaynakların görevlere atanması işlemini yapmak; hedefi ise bir ya da birkaç amacı optimize etmektir. Çizelgeleme problemleri spesifik üretim görevlerine göre çok geniş bir alana yayılmıştır ve birçoğu NP (Nondeterministic Polynomial-time)-zor problemlerdir. Çizelgeleme problemleri, iş çizelgeleme ve proje çizelgeleme problemleri olarak sınıflandırılır.

Faaliyetlerin beklenen zamanda yapılması hem finansal (gerekli mali kaynağın temini) hem de operasyonel (gerekli kaynakların temini) açıdan çok önemlidir. Proje faaliyetlerinin çizelgelenmesi günlük çalışma saatleri, haftalık çalışma günleri, tatiller gibi unsurların, yani çalışma takviminin belirlenmesiyle başlar. Her faaliyetin ne kadar süreceği tahmin edilir ve faaliyetler arasındaki öncelik ilişkilerine ve her faaliyetin tahmini süresine göre çalışma takvimi yardımıyla proje çizelgesi oluşturulur.

Bulanık mantık, düzensizliğin, belirsizliğin, eksik ve çelişen bilgilerin olduğu, hem gerçeğin hem de olasılığın bir parçasının bulunduğu bir çevrede kısacası mükemmel bilginin olmadığı bir çevrede muhakeme etme ve mantıklı karar verme yeteneğidir. Başka bir tanım yapacak olursak; geniş bir alandaki fiziksel ve zihinsel görevleri, hiçbir ölçüm ve hesaplama yapmadan gerçekleştirebilme yeteneğidir.

Proje çizelgeleme konusunda yapılan çalışmaların büyük bir çoğunluğunda, problemle ilgili tüm bilgilere sahip olduğu ve problemin statik, deterministik bir ortamda çalıştığı varsayılmaktadır. Ancak gerçek hayatta proje faaliyetleri büyük ölçüde belirsizdir ve proje uygulanırken aşama aşama çözülmektedir. Bulanık küme teorisi, tam olarak tanımlanması zor olan sistemlerin modellenmesinde kullanılmaktadır. Kesin bilginin olmadığı ve özneliğin bulunduğu bir modeli formüle ederek çözüm sürecine alan bir yöntemdir. Belirsizlikle dolu bu problemin en uygun çözümünü bulmak için belirsizlikle uğraşabilecek çözüm tekniklerine ihtiyaç vardır. Günümüzde böyle bir problemin çözümünde kullanılabilecek en iyi yöntemlerden biri Bulanık Küme Teorisi yöntemidir.

Bu araştırmada, NP zor kapsamında, bulanık proje çizelgeleme problemlerinin çözümlerinde kullanılabilecek öncelik kurallı algoritmalar ve sezgisel/meta-sezgisel yöntemler incelenmiştir. Ayrıca son yıllarda kullanılan ve başarılı sonuçlar veren Paralel Kanguru Algoritması metasezgisel yöntemi ile örnek bir bulanık proje çizelgeleme problemi çözülmüştür. Ayrıca Çok Modlu Proje Çizelgeleme Problemlerine giriş yapılmış ve Hapke ve Wang'ın yapmış oldukları çalışmalar esas alınarak probleme daha etkin güvenilir çözümler bulunmuştur. Tezimde makine çizelgeleme problemlerinin özel bir durumu olan KKPÇP ele alınıp incelenmiştir. Proje kavramı, proje planlamadaki temel kavramlar tartışılmıştır.

1.1 Çalışmanın Amacı ve Kapsamı

Proje yönetimi yaklaşımlarının başarılı olarak uygulanabilmesi için projede var olan zaman ve maliyet belirsizliklerinin uygun şekilde yönetilmesi ve projenin hedeflenen süre ve maliyetle gerçekleştirilebilmesi gerekmektedir. Proje yönetiminde mevcut belirsizliğin modellenmesinde kullanılan iki temel yöntem olasılık teoremine ve bulanık kümeler teoremine dayanmaktadır. Bu çalışmada projedeki zaman ve maliyet belirsizliklerinin uzman görüşüne dayanan bulanık üyelik fonksiyonları kullanılarak modellenmesi, toplam proje süresinin ve toplam proje maliyetinin kaynak kısıtlarına bağlı olarak ele alınacak şekilde matematiksel modelin oluşturulması ve oluşturulan modelin etkin şekilde çözülmesi amaçlanmıştır.

Proje faaliyetlerinin modelde bulanık üyelik fonksiyonları kullanılarak dahil edilebilmesinde proje faaliyet sürelerinin stokastik modellerde faaliyet sürelerinin modellenmesinde kullanılan istatistiksel beta dağılımının yerine bulanık sayı üyelik

fonksiyonları kullanılması öngörülmektedir. Faaliyet sürelerinin beta dağılımında olduğu gibi faaliyetin iyimser, ortalama ve kötümser süreleri esas alınmaktadır. Ancak, bulanık faaliyet süresinin matematiksel ifadesinde iyimser, ortalama ve kötümser sürelerin gerçeği ne kadar temsil ettiği de uzman görüşüne göre ağırlık belirlenerek tespit edilmektedir. Araştırmacılar, bu yaklaşımın gerçek projelerin modellenmesinde daha başarılı olduğunu ve özellikle faaliyet süresinin istatistiksel olarak hesaplanması için yeterli veri olmadığı durumlarda da kullanılabileceği için geniş bir uygulama sahası olduğunu ifade etmektedir.

Ele alınan KKPC ve ÇM-KKPC probleminin modellenmesinde karışık tamsayılı matematiksel modelleme tekniği kullanılmıştır. Modelin amaç fonksiyonunda toplam proje süresi minimize edilmekte ve toplam proje çizelgeleme maliyeti hesaplanmaktadır. Modelde çoklu kaynak durumu öngörülmüştür ve kaynak kullanım maliyetlerinin dönemler itibarıyla değişkenlik göstermediği varsayımına göre modelleme yapılmıştır. Her dönemde projede kullanılabilecek kaynakların birim zamanda kullanılabilecek miktarlarının sınırlı olduğu varsayılmaktadır.

Proje faaliyetlerinin bulanık sayılar olarak modele dahil edilmesiyle oluşturulan bulanık modelin çözümünde Yager indisi kullanılarak kesinleştirme yapılması üzerinde durulmaktadır. Bu yöntemde konveks bulanık sayıların alan karşılığı yöntemi kullanılarak Yager (1981) tarafından önerilen yöntemle göre sıralanması önerilmekte ve sıralama yapılarak kesin hale getirilmiş olan modelin literatürdeki yöntemlerle çözülmesi önerilmektedir. Bu yaklaşım özellikle model hacminin büyük olduğu durumlarda bulanık operatörlerin kullanılmasına gerek kalmadan modelin çözülmesi için iyi bir yöntem olarak görülmektedir. Oluşturulan matematiksel modelde kullanılan bulanık parametreler bu yöntem kullanılarak kesinleştirilmiş ve ortaya çıkan hesaplama karmaşıklığına sahip modelin çözülmesi üzerinde durulmuştur.

Modelin çözümüyle ilgili olarak ise literatürde bulanık proje çizelgeleme probleminin çözümünde sezgisel yöntemler etkin şekilde kullanılmaktadır. Literatürde kullanılan ilk sezgisel yöntemler öncelik kurallarına bağlı sezgisel yöntemlerdir. Bu çalışmada ise literatürde yer alan bir çok öncelik kurallarına bağlı sezgisel yöntemlerine göre oluşturulan çözüm yöntemlerinden sadece ikisi MinSlack ve ROT kuralları önerilmektedir. Bu yöntemlerin başlıca avantajları kolay anlaşılabilir ve uygulanabilir olması ve hızlı şekilde çözüme ulaşabilmesi olarak

görülmektedir. Ancak, elde edilen sonuçların optimal sonuçtan uzaklığı hakkında bulunan belirsizlikler sebebiyle farklı sezgisel yöntemler de araştırılmaya devam etmektedir.

Modelin çözümünde kullanılabilecek alternatif çözüm yöntemleri incelendiğinde Paralel Kanguru ve Tabu Arama (TA) algoritmaları değerlendirilmiştir. TA algortimasının, problemlerin çözümünde hem deterministik hemde bulanık ortamda etkin bir şekilde kullanılabildiği ortaya koyulmuştur. Bundan başka genetik algoritma, benzetilmiş tavlama, bal arısı ve karınca kolonisi, kuş sürüsü, yapay zeka, grafik renklendirme, vs. algoritmalarının deterministik KKPC ve ÇM-KKPC problemlerinin çözümünde etkin sonuçlar verdiği gözlemlenmiştir.

Oluşturulan modelde kullanılacak olan bulanık sayıların sıralanması yöntemi konusunda literatürde değişik yöntemler mevcuttur. Fortemps ve Roubens (1996) tarafından yapılan çalışmada bu yöntemlerin çoğunun uzman yaklaşımını doğru şekilde ifade etmekte yetersiz kaldığı gösterilmektedir. Öte yandan alan karşılığı yönteminin doğrusallık ve toplanabilirlik özelliklerine sahip olduğu ve uzman yaklaşımını doğru şekilde yansıtabildiği ifade edilmektedir. Bu sebeple bulanık sayıların sıralanmasında Yager indeksi ile alan karşılığı yönteminin kullanılmasının uygun olacağı düşünülmüştür.

1.2 Varsayımlar

Yapılan çalışmada bir kaç KKPC probleminin ele alınması nedeniyle ileriki bölümlerin ilgili yerlerinde probleme özgü varsayımlar yapılacaktır. Ancak ortak olanlar bakımından aşağıdaki varsayımlar esas alınmıştır:

1. Proje faaliyet sürelerinin uzman görüşüne bağlı olarak yamuk bulanık sayı dağılımına göre modellenmesinin uygun olacağı öngörülmüştür. Faaliyet süreleri klasik ve yamuk bulanık sayılardan oluşmaktadır.
2. Amaç fonksiyonunda toplam proje süresinin minimize edilmesi hedeflenmiştir. Ayrıca toplam proje çizelgeleme maliyeti modele dahil edilmiştir.
3. Projenin faaliyetlerinin gerçekleştirilmesi için birden fazla kaynağın farklı miktarlarda kullanılması gerekeceği kabul edilmektedir.

4. Kaynak kullanım maliyetlerinin çizelgeleme dönemlerinde sabit ve brüt birim maliyeti olduğu varsayılmaktadır.
5. ÇM-KKPÇ problemlerinde yenilenemeyen kaynakların maksimum miktarı proje süresince sabit olduğu varsayımı yapılmıştır.
6. Bir proje; Başlangıç ve Bitiş faaliyetleri yapay faaliyetler olmak üzere, $N+2$ faaliyetten oluşmaktadır.
7. Mevsimsel ve enflasyonist etkiler bulunmamaktadır.
8. Projenin yürütülmesinde yenilenebilir ve yenilenemeyen kaynakların kullanılması gerektiği varsayılmaktadır.

1.3 Çalışmanın Adımları

Çalışmanın oluşturulmasında sırasıyla aşağıdaki adımlar izlenmiş olup çalışmada izlenen yönetime ait akış diyagramı EK-A'da gösterilmiştir:

1. Çalışmada ele alınan endüstri mühendisliği probleminin tanımlanması,
2. Problemin kavramsal çerçevesinin oluşturulması,
3. Literatürde yapılan önceki çalışmaların araştırılması,
4. Çalışmada kullanılan materyalin toplanması,
5. Matematiksel modelin oluşturulması,
6. Çözüm yöntemlerinin araştırılması ve uygun çözüm yönteminin belirlenerek modelin çözülmesi,
7. Örnek sayısal analizlerinin yapılması,
8. Literatürde yer alan örnek veri setleri kullanılarak modelin değişik durumlardaki performansının test edilebilmesi amacıyla Visual C# 2008 ile programlanması,
9. Literatürde yer alan örnek veri setleri kullanılarak modelin değişik durumlardaki performansının test edilmesi,
10. Elde edilen sonuçların problem çözümündeki yeterliliğinin değerlendirilmesi,
11. Çalışma sonucunda ortaya çıkan sonuçların özetlenmesi ve gelecek çalışmalara yönelik önerilerde bulunulması olarak düzenlenmiştir.

1.4 Özgün Katkılar

Çalışmada ele alınan endüstri mühendisliği problemi literatürde “Kaynak Kısıtlı Proje Çizelgelemesi (KKPÇ) Problemi” olarak tanımlanan problem sınıfına dahil bulunmaktadır. KKPÇ problemiyle ilgili literatürdeki çalışmalarda çoğunlukla deterministik veya stokastik yöntemler kullanılmıştır. Bu çalışmada proje faaliyet sürelerindeki belirsizliğin bulanık olarak ele alınması öngörülmüştür.

Faaliyet sürelerindeki belirsizliğin modellenmesinde sıklıkla kullanılan beta dağılımının ampirik bilgilere dayandığı ve teorik temelini bulmadığı ortaya konulmuştur. Bu çalışmada kullanılan bulanık üyelik fonksiyonları yaklaşımında faaliyet sürelerinin proje yöneticisi tarafından uzman görüşü esas alınarak kullanılması mümkün hale gelmekte ve pratik uygulamada önemli kolaylık sağlanmaktadır. Ayrıca, özellikle proje faaliyetleriyle ilgili geçmiş bilgilere sahip olunmadığı durumlarda bu yaklaşım önem kazanmaktadır (Wang, 2002). Kullanılan bulanık üyelik fonksiyonları yaklaşımı ile proje faaliyet sürelerinin belirli olduğunu varsayan deterministik yaklaşıma ve faaliyet sürelerinin bilinen bir istatistiki dağılıma uygun olduğunu varsayan stokastik yaklaşıma göre, pratik kullanımı daha etkin olan bir yaklaşım oluşturulması hedeflenmiştir.

Çalışmada kullanılan matematiksel model ile projenin hem toplam maliyeti hem de zaman kısıtları göz önünde bulundurmaktadır. Literatürdeki KKPÇ problemleri incelendiğine yapılan çalışmaların büyük çoğunluğunda zaman tabanlı amaç fonksiyonları kullanıldığı ve maliyet faktörüne yer verilmediği görülmektedir (Özdamar ve Ulusoy (1995)). Oysa, gerçek hayat problemlerinde genellikle proje maliyetleri en önemli faktör olarak ele alınmaktadır. Literatürdeki bazı çalışmalar, proje maliyetinin en aza indirilmesi veya proje nakit akışının en fazlalaştırılması amaçları üzerinde durmaktadır. Bu modellerde ise çoğunlukla projenin zaman kısıtlamaları ve proje süresinin maliyete etkileri göz ardı edilmektedir. KKPÇ problemlerinin özünde yer alan zaman-maliyet çatışmasına bu çalışmada yer verilmiştir. Kullanılan formülasyonda KKPÇ problemleri için toplam proje süresinin minimizasyonu dikkate alınmış, ÇM-KKPÇ problemleri için ise hem toplam proje maliyetinin en aza indirilmesi öngörülmüş hem de proje süresinin toplam maliyete etkisi göz önünde bulundurularak projenin mümkün olan en kısa sürede bitirilmesi amaçlanmıştır.

Kullanılan matematiksel modelde çoklu kaynak durumu öngörülmüştür. Yang ve ark. (1992) tarafından çoklu kaynak kısıtı altında proje net değerinin maksimize edilmesine yönelik önerilen karışık tamsayı programlama modeli bu alanda kullanılan benzer modellere örnek teşkil etmektedir. Ayrıca, kaynak maliyetlerinin sabit olduğu varsayımı yapılmış, enflasyonist ve özellikle mevsimsel etkiye bağlı olarak kaynak kullanımı maliyetinin proje süresince değişmediği durum göz önünde bulundurulmuştur. Bu şekilde projenin minimum maliyetle ve zamanında bitirilebilmesi için gereken kaynak dağılımının yapılması öngörülmektedir.

KKPÇ problemi literatürde hesaplama karmaşıklığı (NP-zor) olan bir problem olarak tanımlanmaktadır. Ancak ÇM-KKPÇ problemleri NP-zor olarak tanımlanmaya devam etmektedir. Buna göre proje faaliyet sayısı arttıkça problemin çözüm süresi geometrik olarak artmakta ve problemin makul süreler içerisinde optimal olarak çözülmesi mümkün olmamaktadır. Bu sebeple literatürde büyük ölçekli problemlerin çözümü için çoğunlukla dal-sınır algoritmaları ve sezgisel yöntemler kullanılmaktadır. Son yıllarda yapılan çalışmalar incelendiğinde özellikle genetik algoritma (Leu ve ark. 1999), hibrid genetik algoritma (Kim ve ark. 2004), TA (Mika ve ark. 2006), benzetilmiş tavlama (Hapke ve ark. 2000) yöntemlerinin problemin çözümünde sıklıkla kullanıldığı görülmektedir. Bu çalışmada oluşturulan modelin çözümü için sezgisel ve gelişmiş sezgisel yöntemler kullanılarak çalışılması ve problemin yapısına uygun bir algoritma ile etkin bir çözüm oluşturulması hedeflenmektedir.

Önerilen yöntem farklı bulanık sayı üyelik fonksiyonları kullanılarak, yenilenebilir ve yenilenemeyen proje kaynaklarının mevcut olduğu durumlarda ve farklı proje teknik karmaşıklığı sayısal analizlerle çalıştırılarak test edilmiştir. Sayısal analizlerden sonra, oluşturulan yöntem literatürde yer alan kıyaslamalı proje çalışmasında kullanılmış ve önerilen matematiksel modelin gerçek projenin modellenmesinde başarıyla kullanılabileceği tespit edilmiştir. Elde edilen toplam proje süresi, toplam proje maliyeti, proje faaliyeti çizelgesi ve kaynak kullanımlarına dair model sonuçları, proje gerçekleştirmeleri ile karşılaştırılmış ve sonuçlar arasında büyük tutarlılık olduğu belirlenmiştir. Bu şekilde, önerilen yöntemin gerçek projelerde uygulanabilirliği gösterilmiştir. Yapılan çalışma ile projelerde mevcut bulunan zaman ve maliyet risklerinin modellenmesinde kullanılabilecek matematiksel bir model hazırlanmış, oluşturulan model için çözüm yöntemi

oluşturulmuş, sayısal analizleri ve kıyaslamalı proje uygulaması yapılarak ele alınan KKPC probleminin etkin şekilde çözümünü gerçekleştiren ve gerçek projelerde kullanılabilecek bir yöntem oluşturulmuştur.

Proje yönetimiyle ilgili rapor edilmiş bilgiler ışığında “Gerçek Dünyayı” daha iyi modellemeye yönelik çabalardan birisini gerçekleştirmek üzere genel ifadesiyle KKPC yöntemleri incelenecek olup, projelerin genel yapısını inceleyerek konu anlatılmaya başlanacaktır. Çalışmanın ilerideki bölümleri şu şekilde düzenlenecektir. KKPC problemlerinin türleri, çizelgelemedeki yeri ve önemi izah edilmiştir. Materyal ve yöntemler bölümünde Tamsayılı Doğrusal Programlama (TDP) ile KKPC problemlerinin optimal çözüm önerileri üzerinde durulmuş, TDP’nin KKPC problemlerini çözmede ki yetersizlikleri açıklanmıştır. KKPC problemlerinin çözümü amacıyla daha basit ve kısa sürede çözümler bulan sezgisel yaklaşımlar çeşitli uygulamalarla tanıtılmış olup bu yaklaşımlardan TA prosedürünün Visual C# editörüyle geliştirilen modüler mimariye sahip bir program aracılığıyla çözümler aranmış ve literatürde yer alan örnek problemler vb daha bir çok problem üzerinde test ve karşılaştırmalar yapılmıştır.

1.5 Tezin Organizasyonu

Gerçekleştirilen tez çalışması sekiz bölümde sunulmaktadır. Giriş bölümünden sonra sırasıyla, ikinci bölümde proje yönteminde yer alan temel ve kavramlar bilgileri sunulmuş, üçüncü bölümde literatürde yapılan önceki çalışmalar incelenmektedir. Dördüncü bölümde bulanık ve bulanık küme kavramsal olarak ele alınmıştır. Çalışmada kullanılan materyal ve metod ise sırasıyla beşinci, altıncı ve yedinci bölümlerde tanıtılmıştır. Beşinci bölümde bulanık KYM, altıncı bölümde deterministik/kesin (crisp) KKPC ve B-KKPC modelleri incelenmiştir. Yedinci bölümde deterministik/kesin (crisp) ÇM-KKPC ve B-ÇM-KKPC modelleri incelenmiştir. Son bölümde yapılan sayısal analiz çalışmaları ve uygulama ile araştırmalar sonucunda elde edilen bulgular tartışılmakta ve çalışmanın özeti ve bundan sonraki dönemde yapılacak çalışmalara dair öneriler yer almaktadır.

İkinci bölümde, KKPC problemlerinin proje yönetiminde anlaşılabilmesine imkan tanıyacak açıklamalar ilk bölümde yapılmış ve terminolojide geçen konuyla ilgili terimler ve metodlar anlatılmıştır. KKPC problemlerinin tanımı yapılmış ve proje çizelgelemede ki yeri izah edilmiştir. Proje Kavramı, Proje Planlamadaki Temel

kavramlar tartışılmıştır. Proje çizelgelemenin yapısı üzerinde durulmuştur. KKPC modelleri tartışılmış, yapılacak çalışma için ÇM-KKPCP ele alınacağı belirtilmiştir. Proje çizelgelemede çözüm yöntemleri irdelenmiştir. Klasik yaklaşımlar, basit sezgisel yaklaşımlar, kesin çözüm veren (exact) algoritmalar ve metasezgisel yaklaşımlarıyla KKPC problemlerinin çözülebileceği belirtilmiştir. KKPCP'nin uygulanma alanı bulabildiği alanlar ve diğer üretim yönetim sistemleriyle ilgili karşılaştırmalar izah edilmiştir. KKPCP'nin ileri uygulamaları hakkında bilgiler verilmiştir.

Üçüncü bölümde literatürdeki çalışmaların incelenmesine ayrılmıştır. Üçüncü bölümün birinci kısmında KKPC alanında yapılan çalışmalar irdelenmiştir. İkinci kısımda bulanık KYM'na yönelik yapılan çalışmalar irdelenmiş, bulanık kümeler teorisi, ve proje yönetimi alanlarında gerçekleştirilen önemli çalışmalar incelenmiştir. Üçüncü kısımda, bulanık kümeler teorisine dayanan bulanık KKPC problemiyle ilgili yapılan detaylı literatür taraması ve proje yönetimi alanındaki uygulamaları araştırılmıştır. Dördüncü kısımda ise ÇM-KKPC modelleri ile B-ÇM-KKPC modelleriyle ilgili literatürde yapılan araştırmalar ele alınmaktadır.

Çalışmanın dördüncü bölümü bulanıklık kavramından bahsedilmiş, Bulanık küme teorisi tanımlanmış, KKPC ve ÇM-KKPCP'nin bulanık bir ortamda ele alınması, problemlerin bulanıklıkla ifade edilebilecek kavramlar, Bulanık KYM için kaynak gereksinimlerinin belirlenmesi ve ÇM-KKPCP'nin bulanık bir ortamda ele alınması için bulanık sıralama konularına giriş yapılmıştır.

Tezin materyal ve metodu üç alt bölümden oluşturulmuştur. Bu kapsamda; beşinci bölümde ele alınan endüstri mühendisliği probleminin en basit formu olan KYM ve bulanık kritik yol analizi ve doğrusal programlama formülasyonu incelenerek matematiksel modelin oluşturulması üzerinde durulmuştur.

Altıncı bölümde deterministik ve bulanık faaliyet süreli KKPC problemlerinin tanımı yapılmış ve proje çizelgelemedeki önemi açıklanmıştır. KKPC problemlerinin çözümü amacıyla oluşturulan TDP modelinin küçük hacimli problemlerin en iyi çözümlerinin bulunmasında başarılı ile kullanılabileceği, daha kapsamlı projelerin modellenmesi ve çözümünde, TDP'nin yeterli olamayacağı açıklanmıştır. Matematiksel model kullanılarak yapılan örnek uygulamanın anlatılmasından sonra, ele alınan KKPC problemindeki hesaplama karmaşıklığı konusu detaylı olarak

incelenmektedir. KKPÇ problemleri için daha basit ve kısa sürede çözüm bulan öncelik kurallı sezgisel yaklaşımlarla birlikte gelişmiş iteratif özellikler taşıyan meta sezgisel yaklaşımlar tanıtılmaktadır. Bu yaklaşımlardan ROT ve MINSLACK öncelik kurallı sezgisel yaklaşımları, TA prosedürü için başlangıç çözüm olarak kabul edilmiştir. Seçilen başlangıç çözümlerini iyileştirmek üzere KKPÇ problemlerinin olurlu çözümleri arasından en iyisinin bulunmasını sağlayan TA prosedürü, Microsoft Visual C# programlama dilinde kodlanarak, işlem adımları gösterilerek probleme etkin yeni çözümler aranmıştır. Oluşturulan modelin çözümünde kullanılan analitik ve sezgisel çözüm yöntemleri incelendikten sonra yapılan deneysel tasarım çalışması anlatılmış ve modelin bilgisayar kodu sunulmuştur.

Yedinci bölümde proje yönetimi hakkında elde edilen bilgi birikimi ışığı altında, “gerçek dünyayı” daha iyi modellemeye yönelik çabalardan birini gerçekleştirmek üzere makine çizelgeleme problemlerinin özel bir durumu olan deterministik ve bulanık faaliyet süreli ÇM-KKPÇP irdelenmektedir.

Son bölümde yapılan çalışma sonucunda ortaya çıkan araştırma bulguları iki kısımda tartışılmaktadır ve son kısımda ise yeni araştırmacılar için önerilerde bulunulmuştur. Birinci kısımda problem büyüklüğü, kaynak kısıtları ve öncelik ilişkilerine göre elde edilen model örnek problemler sayısal analiz sonuçları ele alınmaktadır. Modelin çözümünde kullanılan doğrudan optimizasyon yöntemi, başlangıç çözümünün sezgisel algoritmayla belirlenmesi yöntemi ve çözüm uzayının sınırlandırılması yöntemlerinin uygulaması sunulmaktadır. Önerilen yöntemle ilgili yapılan sayısal analizler incelenmektedir. İkinci kısımda ise, Literatürde yer alan çeşitli kıyaslama problemlerine ait veri kümeleri, geliştirilen bilgisayar programında test edilmiştir. Deney sonuçlarıyla belirlenen performans kriterleri karşılaştırılarak bilinen sezgisel yaklaşımların TA yöntemiyle birlikte hem deterministik&crisp hemde bulanık ortamda çok daha hızlı, optimal veya optimale yakın iyi sonuçlar verdiği bulunmuştur. Elde edilen sonuçlar detaylı bir şekilde tezin eklerinde sunulmaktadır. Gerçekleştirilen çalışmalar ve elde edilen sayısal sonuçlar özetlenmiş ve ilgili alanlarda bundan sonraki dönemde yapılabilecek çalışmalar ele alınmıştır. Tez kapsamında ele alınan KKPÇP’lerinin literatürde yer alan kıyaslama problemleri tekrar çözülmesi ve optimalliği garanti edilmiş ve işlem sürelerinin kısılacağı öngörüsü tespit edilmiştir.

2. PROJE YÖNETİMİ VE ÇİZELGELEME

Bu bölüm, proje yönetimi kavramları hakkında bir bakış açısı sağlamayı amaçlar. Bu sebeple proje kavramı ve proje yönetiminin esasları incelenecektir. Proje yönetiminde çizelgeleme kavramı ele alınacak, çeşitli proje çizelgeleme teknikleri açıklandıktan sonra KKPC problemleri ve geliştirilen modeller tanıtılacaktır. Bu metotlar arasında çeşitli karşılaştırma ve değerlendirmeler yapılarak, kuramsal ve teorik bilgiler verilecektir. Proje çizelgeleme problemleri; proje, faaliyetler, kaynaklar, öncelik ilişkileri ve performans ölçülerinden oluşmaktadır.

2.1 Proje ve Proje Yönetim Kavramı

Proje, belli bir amaç için özel olarak hazırlanmış, karmaşık yapılı, sayılı ve sıralı faaliyetler kümesidir. Bir proje; tekrarlanmayan, belirli kaynaklarla, belirli bir zaman aralığı içinde tamamlanması gereken özel faaliyetler topluluğu olarak da tanımlanabilir (Barutçugil 1998). Bu tanım ışığında projelerin ortak özellikleri şöyle sıralanabilir.

- a. Projeler karmaşık ve özgün çabalarlardır. Birbirini izleyen veya birlikte yürütülen; ancak tekrarlanmayan faaliyetlerden oluşur.
- b. Projeyi oluşturan faaliyetler arasında, uygulanan teknolojiye kaynaklanan kısıtların oluşturduğu kompleks bir ilişki vardır. Bu ilişki, başlangıç–bitiş olarak adlandırılabilir karmaşık bir akış oluşturur.
- c. Proje, belirli bir sonuç yaratma sürecidir.
- ç. Her projenin belirli bir başlangıç, bitiş ve bu ikisi arasında geçen bir yaşam devresi vardır.
- d. Yaşam döneminin her aşamasında projenin niteliklerinde önemli değişiklikler olur.
- e. Tüm projeler sonludur ve belli bir amaç için geliştirilir. Amacın ortaya çıkışı ile başlar, gerçekleşip uygulamaya geçmesi ile sona erer.

f. Tm projeler zeldir. Her proje bulunduęu Őartlar, gereksinimler, amalar ve zorluklar altında sadece bir kere gerekleŐir. Seri retim ve ktle retimi, birer proje zellięi taŐımazlar. Buna karŐın zel imalat, atlye tipi retim, tekil sipariŐ retim gibi retim eŐitleri birer proje nitelięi taŐırlar.

g. Tm projeler birer iŐler kmesidir. Her proje analiz safhasında tanımlanmıŐ faaliyetlerin birer bileŐkesidir.

ę. Projeler nemli risk ve belirsizlikler taŐırlar. Bu belirsizlikler, proje ilerledike giderek artar veya azalır.

h. Bir projeyi hızlandırmanın maliyeti sonuca yaklaŐıka daha da ok artar.

Bu ortak zellikler, projelerin dikkatli ve zel yaklaŐımlarla ynetilmesi gerektięini ve baŐlangıtan bitiŐe kadar btnleŐik bir planlama ve denetleme sistemini zorunlu kıldıęını gstermektedir.

Proje yneticisi; projenin yrtlmesi sırasında ortaya ıkacak tm sorunlara, risk ve belirsizliklere karŐı hedeflenenin gerekleŐtirilmesini saęlamak iin alıŐan kiŐidir. Buna gre proje ynetimi, belirlenen bir amaca ulaŐmak zere biraraya getirilen maddi ve insani kaynakların faaliyetlerini planlama, yrtme, dzenleme ve denetleme fonksiyonlarının toplamıdır.

2.2 Proje Planlama Yapısı ve Ama

Proje planlamada ama ekonomiklik olduęuna gre, proje sresinin, maliyetinin ve kaynak kapasite kullanımının en kk kılınması ve emniyet unsurlarının en yksek dzeyde tutulması n planda olmalıdır. Ancak, bu temel parametrelerin drdn birden en kkleyen bir zmn bulunması ok zordur, hatta bazen de imkansızdır. nk bunlardan birini en kk kılan bir zm dięer n arttırma eęilimi gstermektedir.

Proje planlama  temel yapıda ele alınacaktır.

1. Deterministik Yapı
2. Stokastik Yapı
3. Bulanık Yapı

2.2.1 Deterministik yapı

Deterministik yapı, daha önceki projelerden veya tecrübelerden faydalanarak benzer bir proje için ayrıntılı ve kesin bir planlamayı gerektiren yapı biçimidir. Tekil sipariş üretimi ve tesis bakımı bu tür bir yapıya örnek olarak gösterilebilir.

2.2.2 Stokastik yapı

Stokastik yapıda, proje elemanları, süreleri ve kaynak gereksinimleri belirsizlik taşır. İlk kez yapılan projeler ve araştırma çalışmaları bu tür bir yapı gösterirler. Stokastik yaklaşım, projedeki olasılık elemanına etkide bulunur ve hem beklenen proje süresini hem de bundan sapmayı (*varyansı*) hesaplamaya ihtiyaç duyar. Böylece geçmiş bilgi ve istatistiki yöntemlerin kullanımını zorunlu kılar.

Proje yönetiminde en büyük problemlerden biri, risk kaynağı olarak yüksek miktarda belirsizliğin mevcut olmasıdır. Risk, bütün projelerde mevcuttur. Belirsizliğin temel kaynakları yetersiz ve geçersiz bilgi, bileşen ve alt sistemlerin performanslarında şansa bağlı değişimler ile bilgi eksikliği nedeniyle tahmin yapma güçlüğüdür. Özellikle üç çeşit belirsizlik olduğu kabul edilmektedir.

2.2.2.1 Çizelgelemede belirsizlik

Proje çevresinde meydana gelen ve daha önceden tahmini mümkün olmayan değişimler, bazı faaliyetlerin süresi üzerinde olumsuz etkide bulunabilirler. Örneğin bir taşeronun işi tamamlama süresi uzayabilir ve bu bütün sistemi etkiler. Bu etkiyle başa çıkmak için, olasılık hesapları ve simülasyon yaklaşımlarının kullanılması gereklidir.

2.2.2.2 Maliyetlerde belirsizlik

Faaliyetlerin süreleri hakkındaki bilginin sınırlı olması, onları tamamlamak için gerekli kaynak miktarının belirlenmesini de zorlaştırır. Bu durum, maliyetlerde belirsizlik olduğu anlamına gelir. Ayrıca, kaynakların beklenen kullanım oranı ve proje için gerekli malzemelerin fiyatları da büyük oranda değişkenlik gösterir.

2.2.2.3 Teknolojik belirsizlik

Bu çeşit belirsizlik yeni (henüz test edilip onaylanmamış) metotların, ekipmanların, teknolojilerin ve sistemlerin geliştirildiği veya kullanıldığı projelerde özellikle AR-GE projelerinde görülür. Teknolojik belirsizlikler çizelgeyi, maliyetleri ve projenin nihai başarısını etkiler. Organizasyonel ve politik belirsizlik kaynakları gibi başka belirsizlik kaynakları da vardır. Yeni kanuni düzenlemeler pazarı etkileyebilir, personelin işten ayrılması ve katılan organizasyonların politikalarındaki değişimler işlerin akışını yavaşlatabilir. Faaliyetlerin gerçek süresi ancak çok istisnai durumlarda kesin olarak bilinebilir. Bu yüzden, geçmiş verilerin analizi ve faaliyet sürelerinin frekans dağılımının hazırlanması, projenin olası tamamlanma zamanı hakkında bir fikir verebilir. Normal dağılım, simetrik ve çalışılması kolaydır. Fakat gerçek verilerin bu dağılıma uyup uymadığı ki-kare gibi istatistiksel bir test ile belirlenebilir.

2.2.3 Bulanık yapı

Belirtilen yöntemlerle proje çizelgeleme yapısının, deterministik ortamda ve belirsizlik ortamında kullanımına göre iki kısma ayrılacağı anlaşılmaktadır. Bu sebeple literatürde proje çizelgeleme problemlerinin yapısı iki şekilde sınıflandırılmaktadır. Ancak biz günümüzde, proje çizelgeleme yapısının üç şekilde sınıflandırılması gerektiğine inanmaktayız. KYM yöntemi deterministik yapıyı, belirsizlik ortamında istatistiki olarak olasılık teorisini temel alan PERT yöntemi ile temsil edilirken bu çalışma kapsamında proje çizelgelemenin belirsizlik yapısına bulanıklık yapı ifadesi ile bulanık FCPM ve FPERT yöntemleri ile belirsizlikler ifade edilmiştir. Her dört teknik ile bir projenin başlama ve bitişi ile ilgili bilgiler hesaplanabilir.

Proje çizelgeleme konusunda yapılan çalışmaların büyük çoğunluğunda, problemle ilgili tüm bilgilere sahip olduğu ve problemin statik, deterministik ortamda çalıştığı varsayılmaktadır. Ancak gerçek hayatta proje faaliyetleri büyük ölçüde belirsizdir ve proje uygulanırken aşama aşama çözülmektedir. Bulanık küme teorisi tam olarak tanımlanması zor olan sistemlerin modellenmesinde kullanılmaktadır. Bulanık küme teorisi kesin bilginin olmadığı ve özneliğin bulunduğu bir modeli formüle ederek çözüm sürecine alan bir yöntemler dizisidir.

Bulanık mantığın mimarı olan, Lotfi A. Zadeh bulanık mantığın yeteneğini şu şekilde açıklamaktadır. “Bulanık mantık, insanın iki mükemmel yeteneğinin resimleştirilmesi/makineleştirilmesi için yapılan bir girişim olarak görülebilir. Bunlardan ilki; düzgünlüğün olmadığı, belirsizliğin, eksik bilgilerin, çelişen bilgilerin, gerçeğin ve olasılığın bir parçasının bulunduğu bir çevrede kısaca mükemmel bilginin olmadığı bir çevrede muhakeme etme ve mantıklı karar verme yeteneğidir. İkincisi ise geniş bir alandaki fiziksel ve zihinsel görevleri hiçbir ölçüm ve hesaplama yapmadan gerçekleştirebilme yeteneğidir”. Dolayısıyla belirsizlikle dolu bu problemin en uygun çözümünü bulmak ancak belirsizlikle uğraşan çözüm teknikleriyle mümkün olabilir. Böyle bir problemin çözümünde günümüzde kullanılabilecek en iyi yöntemlerden biri, Bulanık Küme Teorisi yöntemidir. Bu çalışmada, bulanık proje çizelgeleme problemlerinin çözümlerinde kullanılan, kritik yol metodu, bulanık tek modlu ve çok modlu kaynak kısıtlı proje çizelgeleme (B-TM-KKPC ve B-ÇM-KKPC) problemleri incelenmiştir. Ayrıca son yıllarda kullanılan ve başarılı sonuçlar veren doğrusal programla ve bulanık aritmetik çözümler öneren, örnek bulanık proje çizelgeleme problemleri çözülmüştür. Bu çalışma deterministik ve bulanık planlama yapısına sahip projeler için tasarlanmıştır.

2.3 Proje Yönetiminde Temel Kavramlar

Proje analizi, bir projeyi meydana getiren faaliyetlerin bir akış yapısı içinde belirlenip tanımlanması için yapılan bir çalışmadır. Bu çalışma sonunda; bir faaliyetler listesi ve tahmini başlangıç zamanı türünden bir zaman listesi oluşur. Bu listeye daha yakından bakıldığında, bir projenin temel yapı taşlarının faaliyet, kaynak ve bu kaynağın kapasitesi olduğu ve bu kavramların açıklanması gerektiği anlaşılmaktadır.

2.3.1 Faaliyet

Faaliyet (etkinlik), kaynak ihtiyacı yaratan ve belirlenmiş başlangıç ve bitiş noktaları arasında bir zaman aralığını kapsayan proje yapı taşıdır. Tanımından da anlaşılacağı üzere, faaliyet, kaynak kullanımı ile gerçekleşen bir üretim faaliyetidir. Literatürde kavramsal olarak faaliyet (activity), iş (job), görev (Task), vb çok farklı terimler kullanılmıştır. Bu çalışma kapsamında ise, “faaliyet” terimini kullanmayı tercih ediyoruz.

2.3.2 Kaynak, kaynak yapısı ve kaynakların sınıflandırılması

Yöneylem araştırmasının temel işlevlerinden birisi de kaynak tahsisidir. Kısıtlı kaynakların mevcudiyeti, kaynakların kullanımında belirli amaç işlevini veya işlevlerini en iyileyecek kaynak tahsislerini önemli kılar. Kaynaklar, faaliyetlerin gerçekleştirilmesi için kullanılır. Faaliyetleri burada geniş anlamda tanımlıyoruz. Örneğin; yeni bir katın betonunun atılması, bir frezenin bir parçanın yüzeyini belirli bir ölçüye indirmesi, bir işlemcinin bir alt programı işlemesi gibi.

Kaynaklar ise;

- İş gücü,
- Makine, ekipman, tezgah, teçhizat vb. üretim araçları,
- Malzeme, yakıt, enerji vb. üretim girdileri,
- Para ve diğer finans araçları şeklinde tanımlanabilir.

Proje yönetiminde kaynak kısıtları açısından bakıldığında üç farklı yapı ortaya çıkmaktadır. Kaynaklar, kullanım ve tüketim özelliklerine göre üç ana grupta ele alınabilir: Faaliyetler tarafından kullanılan kaynaklar yenilenebilir (renewable), yenilenemeyen (non-renewable) kaynaklar ve kısmen yenilenebilir ya da çift yönden kısıtlı (doubly constrained) kaynaklar olarak sıralanır. Eğer kaynaklar sadece periyot tabanlı kısıtlanmışsa yenilenebilir kaynak adı verilir. Her yenilenebilir kaynak bütün periyotlarda mevcuttur. Diğer bir deyişle kaynakların varlığı proje uzunluğundan etkilenmemektedir.

- ◆ Yenilenebilen Kaynaklar
- ◆ Yenilenemeyen Kaynaklar
- ◆ Kısmen Yenilenebilir veya Çift Yönden Kısıtlı Kaynaklar

Kısmi yenilenebilir kaynaklar (partially renewable resources), modellemede belirli zaman dilimleri için, farklı kaynak kullanımı üst sınırını tanımlama olanağı vermektedir. Buna göre örneğin, vardiya düzenini problem gösterimine dahil etmek mümkün olmaktadır. Bu şekilde, daha gerçekçi problem tanımlarına olanak tanınmaktadır (Drexl vd., 1999). Bir faaliyetin uygulama süresi boyunca değişken kaynak kullanımı talebi olması (Sprecher, 1994) farklı bir kaynak kullanımı olarak gösterilebilir.

Diğer bir kaynak sınıflandırma da, kaynakların bölünebilirliğine dayandırılır. Adet olarak ifade edilen, yani bölünebilen kaynaklara, (örneğin beş adet iş makinesi) ayrık (discrete) kaynaklar; elektrik enerjisi gibi bölünemeyen kaynaklara da sürekli (continuous) kaynaklar denir.

Bu çalışmada kaynak kelimesi yenilenebilir ve yenilenemez kaynağı ifade etmek üzere kullanılacaktır. Yenilenebilir ve yenilenemeyen kaynak türünün dışında bir kaynak tipi kullanıldığında bu özel olarak ifade edilecektir.

2.3.2.1 Yenilenebilen kaynaklar

Projenin devam ettiği süre içinde, herhangi bir zaman diliminde, kullanılabilirliği kapasiteyle sınırlı olan işgücü ve üretim araçlarıdır. Bir diğer ifadeyle yenilenebilen kaynaklar; kullanımları sadece her bir dönemde kısıtlı olan kaynaklardır. Örneğin, işçilik sınırlı bir periyod içinde kullanılıyor ancak her dönem yeniden kullanılabiliriyorsa yenilenebilir bir kaynak olduğu değerlendirilir. Yenilenebilir kaynakların, bir zaman birimi içindeki toplam kullanım miktarı kısıtlı olmakla birlikte bunlar, kullanım ile tüketilmezler. Belirli bir faaliyet çerçevesinde kullanıldıktan sonra faaliyet sona erince tekrar kullanıma hazır duruma gelirler. Örneğin iş makinaları, bilgisayar işlemcileri gibi.

2.3.2.2 Yenilenemeyen kaynaklar

Yenilenemeyen kaynaklar kullanıldıklarında tüketilirler. Bir projede, toplam miktarı kısıtlı, eksildikçe bu miktardan kaybeden ve yeniden tanzim edilemeyen kaynaklardır. Malzeme, yakıt ve para gibi kaynaklar buna örnek olarak gösterebilir. Yenilenemeyen kaynakların proje süresi boyunca kullanılabilir toplam miktarı üzerinde kısıt vardır.

2.3.2.3 Kısmen yenilenebilen kaynaklar

Bir kaynağın, hem bir zaman birimi içinde kullanım miktarı, hem de proje süresi boyunca toplam tüketim miktarı üzerinde kısıt olması durumunda, bu kaynak çift yönden kısıtlı kaynak olarak nitelendirilir. Her zaman aralığına yerleştirilen kaynaklar, eğer o zaman aralığı için yenilenemezse bu kaynaklar (iş saatleri ya da makine saatleri gibi) sadece başlangıçta yenilenebilir. Örneğin para çift yönden kısıtlı

bir kaynak olabilir. Çünkü zaman birimi içindeki harcama miktarı kısıtlı olabileceği gibi, proje süresince toplam harcama miktarı da kısıtlı olabilir.

2.3.3 Kapasite

Kapasite, üretkenliğin bir göstergesidir. Kaynakların, bir zaman birimi içerisinde gerçekleştirdiği iş miktarıdır. İşletme standartları veya normal çalışma koşulları (bir kaynağın bulunduğu işletme standartları ile belirlenmiş efektif çalışma temposu ve mesai süresi), kaynağın teknolojik özelliklerine göre belirlenir ve adet/gün, ton/yıl, m³/saat, adam/saat vb. birimlerle ifade edilirler. Kullanılan birimlerin, farklılıklardan etkilenmeyen ve tüm kaynaklar için ortak olan bir ölçüte göre oluşturulması gerekir. Projelerde kullanılan bu ölçüt, programlama ve kontrol evresinde tespit edilen zaman birimi esas alınarak, bir kaynağın bu zaman aralığında, normal çalışma koşullarında kullanılabildiği süre ile örneğin saat/gün veya vardiya/hafta gibi birimlerle ifade edilir. Kapasite, çeşitli zorlamalarla kısmen artırılabilir. Bu aşamada; fazla mesai, ek vardiya gibi önlemlerle kaynağın çalışma süresini uzatmak, motivasyon ve teşvik ile kaynağın çalışma temposunu arttırarak verimi yükseltmek ve fason imalat yapmak, gibi kapasiteyi zorlayıcı sebepler örnek gösterilebilir.

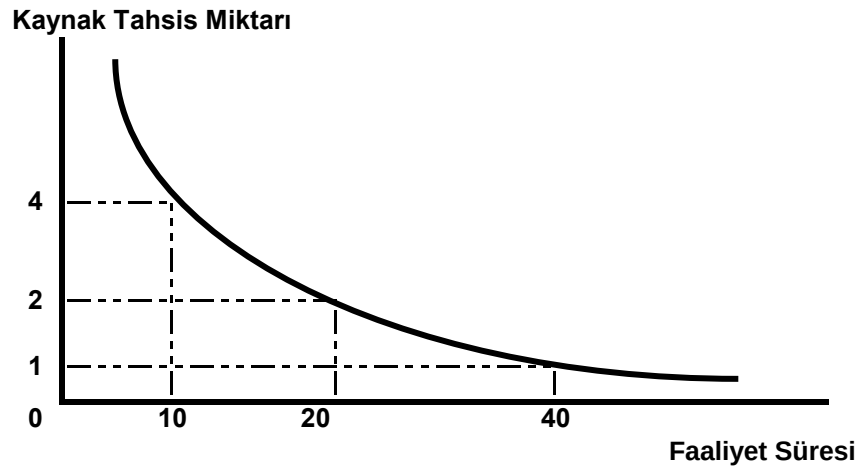
2.3.4 Faaliyet süresi ile kaynak tahsis ilişkisi

Bir projenin analiz safhasında faaliyetlerin tanımlanmasıyla hangi teknoloji ve ne tür kaynakların gerekeceği planlanır. Örneğin, bir tornalama işleminin “üniversal torna tezgahı ve kalifiye işgücü” kullanarak veya “nümerik ya da CNC kontrollü torna tezgahı ve teknisyen” kullanarak yapılabileceğinin belirlenmesi, bu işlemde farklı teknoloji seçeneklerinden birinin uygulanabileceğini gösterir. Faaliyetlerin tanımını ve hangi teknolojinin kullanılacağını tespiti ile kullanılan kaynak türünün nitel ve nicel özellikleri saptanmış olur. Bir faaliyetin belirlenmiş her teknoloji seçeneğinde, kaynaklara yarattığı iş miktarı, “iş hacmi” terimi ile ifade edilir. Faaliyet süresi; faaliyete birim zamanda tahsis edilen kaynak miktarı ve kaynak kapasitesine bağlı olarak şu şekilde ifade edilebilir.

$$\text{Faaliyet Süresi} = \frac{\text{İş Hacmi}}{(\text{Kaynak Tahsis Miktarı} \times \text{Kaynak Kapasitesi})} \quad (2.1)$$

Denk. (2.1)'de, iş hacmi ve kaynak kapasitesi sabit olarak düşünülürse faaliyet süresi ile kaynak tahsis miktarı arasında ters orantı olduğu açıkça görülür. Örneğin bir uçak boya yenileme işleminin faaliyet süresinin 320 adam/saat olduğu düşünülürse; bir işi, tek bir işçinin 8 saat/günlük bir kapasite ile 40 günde bitireceği, 2 işçinin 20 iş gününde tamamlayacağı, 4 işçinin ise 10 iş günü sonunda bitirebileceği açıktır.

Yukarıda belirtildiği üzere; iş hacminin sabit olduğu varsayımıyla, faaliyet süresi-kaynak tahsis miktarı arasındaki ters orantı, bir fonksiyonla ifade edilirse, bu fonksiyon Şekil 2.1' de görüldüğü gibi olacaktır.



Şekil 2.1 : Faaliyet Süresi - Kaynak Tahsis İlişkisi.

Özetle bir teknoloji seçeneğinde, faaliyet süresine etki eden faktörler şunlardır:

- Kaynak tahsis miktarı,
- Fazla mesai, ek vardiya,
- Kaynakların çalışma hızı ve temposu.

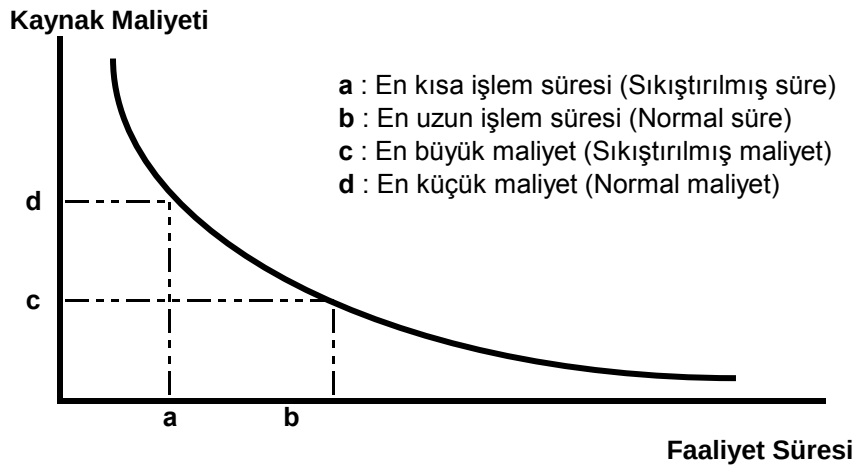
Projelerin analiz safhasında, faaliyet sürelerinin yukarıdaki unsurlarla nasıl ve hangi sınırlar içerisinde değiştiği belirlenir. Teknolojik alt sınır veya en kısa faaliyet süresi; en büyük kaynak tahsis miktarı ve kaynak kapasitesinin gerçekleştiği durumdur. Bölünemez ve tamsayı birimlerle tanımlı kaynaklar ve kaynak kapasitesindeki zorlamalar nedeniyle sınırlamalar arasındaki faaliyet süresi-kaynak tahsis miktarı arasındaki ilişki, ancak sürekli bir fonksiyonla tanımlanmakta veya oldukça ayrıntılı bir analizle tanımlanabilmektedir.

Bir faaliyetin gerçekleştirilme süresi ile kullanılan kaynaklar arasında bir ilişki vardır. Genel olarak, birim zaman içinde daha çok kaynak kullanımının maliyeti

yükseltmesi, faaliyet süresini ise azaltması beklenir. Bu öngöründen hareketle, bu uygulamayı, maliyet-süre ödünleşimi (time-cost trade-off) şeklinde modelleyebiliriz. Kullanılan kaynağın ayrık veya sürekli olmasıyla bağıntılı olarak maliyet-süre ödünleşimi ayrık veya sürekli bir işlev olarak ifade edilir. Ayrık işlev durumunda, işlevin her bir maliyet-süre çiftine karşı gelen noktası bir mod (mode) olarak nitelendirilir. Bir veya daha fazla sayıda faaliyetli ve birden fazla moda sahip proje çizelgeleme problemleri çok modlu (multi-mode) problemler olarak nitelendirilirler. Diğer bir ödünleşim türü de, kaynaklar arasındaki ödünleşimdir (resource-resource trade-off). Burada bir faaliyetin süresi sabittir, ancak değişik kaynakların farklı kullanımları söz konusudur. Örneğin bir kanal açma faaliyetinin; bir kazı makinası, bir operatör ve iki düz işçi yerine aynı süre içinde on altı düz işçi tarafından yapılması gibi. Proje yönetiminin uygulamalarında genellikle birden fazla projenin aynı kaynak havuzundan yararlanılarak yönetimi söz konusudur. Bu tür problemler, çok projeli (multi-project) çizelgeleme problemleri olarak nitelendirilirler.

2.3.5 Faaliyet süresi ile maliyet ilişkisi

Faaliyet süresi ile endirekt maliyet arasındaki ilişki kısaca, kaynağın faaliyette kullanılması durumunda ortaya çıkan, aksi bir durumda hiçbir şekilde görünmeyen bir maliyet kalemidir. Bu, bir faaliyetin süresinin kısaltılması ile artar. Bir faaliyete ayrılan her kaynağın kullanılmak için hazırlanması, fazla mesai, ek vardiya gibi kapasite zorlamaları, iş gücüne ödenen ücretlerin artmasına, dolayısıyla endirekt faaliyet maliyetinin yükselmesine sebep olur. Şekil 2.2’de değişken faaliyet süreleri ile faaliyet maliyeti arasındaki ilişki görülmektedir.



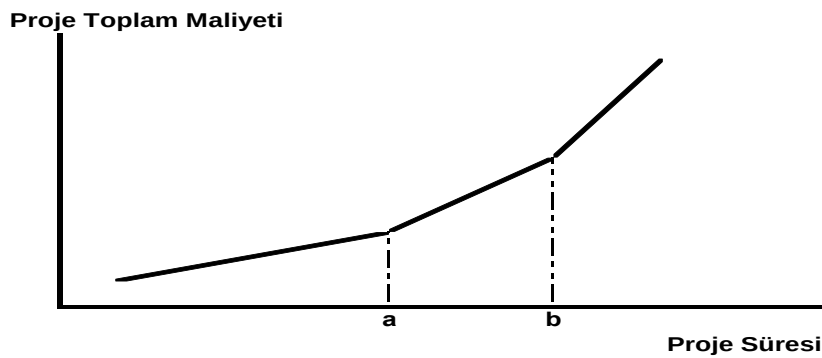
Şekil 2.2 : Faaliyet Süresi – Endirekt Maliyet İlişkisi.

2.3.6 Proje süresi - maliyet ilişkisi

Projenin toplam süresindeki değişimler, genel masraflar, teslim tarihine bağlı ceza veya ödüller ve fırsat maliyeti gibi unsurlar, maliyet kalemlerini etkiler. Proje süresinin uzamasıyla birlikte genel giderlerin önceden planlanandan fazla olacağı aşıkardır. Buna ek olarak projeye başlamadan taraflarca yapılan kontrat gereği her türlü süre uzatımının bir takım cezai tedbirler getireceği de bilinen bir gerçektir. Paralel olarak projeyi planlanandan önce bitirip, işletmeye almak ise birtakım ödüllendirici uygulamaları mümkün kılar. Bu durum ödülle birlikte, proje süresine bağlı yukarıda açıklanan maliyetlerin de azalmasına sebep olacaktır. Ancak dikkat edilecek önemli bir husus, projenin erken sürede tamamlanması ile ortaya çıkan “atıl kaynağı” olabildiğince kısa sürede rasyonel bir biçimde kullanma zorunluluğudur. Aksi bir durum, maliyetler üzerinde istenmeyen artışların olmasına sebep olur.

Proje süresinin uzamasıyla artan maliyetlere, “fırsat maliyetlerini” de katmak gerekir. Fırsat maliyeti kısaca, kaybedilen faydanın yarattığı maliyet olarak tanımlanabilir. Buna, bir demir çelik işletmesindeki yüksek dereceli fırınların bakımının uzamasından doğacak üretim kayıpları ve bundan doğan maliyetler örnek olarak gösterilebilir. Ya da uçak bakım faaliyetlerinin uzamasından kaynaklanan üretim kayıpları ile uçuş kayıp-kazanç oranları ve bundan kaynaklanan maliyetler de dikkate değerdir. Bu tip maliyetler genelde her projede benzer şekilde olmalarına karşın, bunları ortak bir fonksiyonla tanımlamak mümkün değildir. Her projeye göre farklı özellik gösterirler.

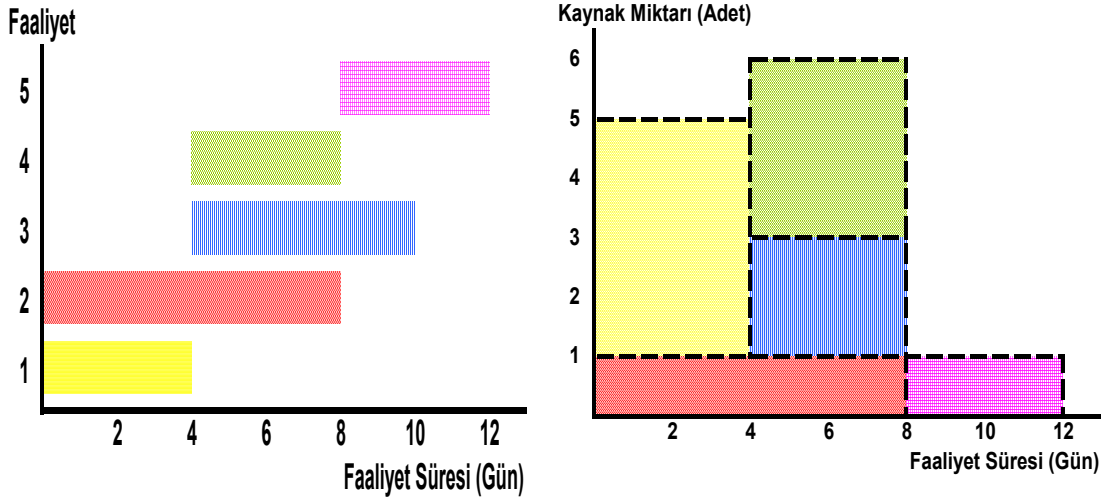
Proje süresi-maliyet ilişkisi Şekil 2.3’de görüldüğü gibi özetlenmiştir. Şekil 2.3’de a ve b noktalarında doğrudaki eğim değişiklikleri; gecikme cezası veya fırsat maliyetlerindeki artışların bir neticesidir.



Şekil 2.3 : Proje Süresi – Maliyet İlişkisi.

2.3.7 Kapasite yükleme ile maliyet ilişkisi

Kapasite yükleme veya kapasite kullanımı - maliyet ilişkisi; Şekil 2.4'teki "kaynak yük diyagramı" veya kısaca "yük diyagramı" ile açıklanabilir. Faaliyet süresi ve proje süresine bağlı maliyetlerin her iki durumda da aynı olduğu anlarda, kapasite kullanımından oluşacak maliyetler, kaynaklara verilen ani yük değişimlerinden meydana gelir. Eldeki kaynakların ani yük artışı gerektiren bir işe kaydırılması, nakil ve hazırlık maliyetlerini ortaya çıkarır. Yüklemelelerdeki bir düşüş ise, kaynakların atıl kapasite yaratmasına sebep olur. Özetle toplam proje maliyetini oluşturan elemanlar, yukarıda da açıklandığı gibi, faaliyet süresine, proje süresine ve kapasite yüklemesine bağlı olan maliyetler ve projenin tamamına etki eden genel gider kalemleridir. Projenin planlanmasında oluşturulan amaç fonksiyonu ise, tüm maliyet kalemlerinin belirlenip, programın ilgili değişkenleri ile fonksiyonel ilişki kurmayı sağlar. Kurulan bu fonksiyonel ilişki, faaliyet süresi ve proje süresi ile ilgili maliyet kalemleri üzerinde işlenmekte, kaynak yüklemesinden doğan maliyetler için böyle bir fonksiyonel tanımlama yapılamamaktadır. Bu sebeple toplam proje maliyetinin elde edilmesi aşamasında ilk iki maliyet kalemi, yaklaşık bir değer vermektedir.



Şekil 2.4 : Kaynak – Yük Dağılımı.

2.4 Proje Faaliyetlerinin Serim ve Düğüm Üzerinde Gösterimi

Proje şebekeleri, bir kümenin elemanları arasında kurulan ve matematiksel ifadelerle belirtilebilen karmaşık ilişkilerin "düğüm" ve "serim"ler ile elde edilen grafiksel görüntüsüdür. Bu grafiklerde yön gösteren bağlantı, "Serim" adını alır. Serim ve düğümlerden oluşan diyagram ise, "yönlendirilmiş diyagram"dır (Taha 1987).

Bağıntılar ve faaliyetler proje akışını belirttikleri için şebekede oklarla, olaylar ise durum değişikliklerini göstermesi sebebiyle düğümlerle yer alırlar. İşlem veya olay kendinden öncekinin bitmesi ile başlar ve kendinden sonra gelen ile biter. Bu sebeple başlangıçtan bitişe proje akışını gösterir bir ağ yapısı vardır.

Genel bir proje şebekesindeki özellikler şunlardır:

- Faaliyet, bağıntı ve olayların açık ve net bir şekilde gösterilmesi ve her faaliyete ait olayların ayrı ayrı düğümler halinde belirtilmesi şebeke mantığı içerisinde proje kontrolünü kolaylaştırıcı bir rol oynar.
- Şebeke genel bir akış modeli niteliği taşımaktadır. Bu çalışmada çeşitli ağ yapılarından ve hesaplamalarından bahsedilmeyecektir.

KKPÇ problemi bir serim (network) üzerinde tanımlanır. Bilindiği gibi bir serim, düğümler (nodes) kümesi $\{V\}$, serimler (arcs) kümesi $\{A\}$ ve düğümler ve serimleri ilişkilendiren ilişkiler kümesi $\{\Phi\}$ ile tanımlanır. İki tanımlama seçeneği vardır. Birinci uygulamada; faaliyetler, serimler kümesinin, olaylar ise düğümler kümesinin elemanları olarak tanımlanır. İlişkiler kümesi, hangi faaliyetin hangi olaylar arasında yer aldığı bilgisini içerir. Bu tanımlama seçeneğine, faaliyetlerin serimlerle gösterimi (AOA) diyoruz. AOA'da serimin oluşturulmasında yapay faaliyet tanımına gerek duyulabilmektedir. Proje çizelgeleme problemlerinin serim üzerinde tanımlanmasında ilk kullanılan gösterim şekli AOA'dır. Diğer tanımlama seçeneği, faaliyetlerin düğümler üzerinde gösterildiği tanımlama şeklidir. Serimler öncüllük ilişkilerini gösterir. Bu tanımlama seçeneğine, faaliyetlerin düğümlerde gösterimi (AON) diyoruz. AON, yapay faaliyet tanımına gerek olmadığı için daha basit bir gösterimdir. İki gösterimde de dikkat edilmesi gereken husus, proje seriminin sadece bir başlangıç ve sadece bir bitiş düğümü olması gereğidir. Aksi durumda, yapay bir başlangıç ve/veya bitiş düğümü tanımlanır. Projenin modellenmesinde faaliyetlerin tanımlanma ayrıntısı önemli bir yere sahiptir. Projenin iş kapsamı kendi içinde bir bütün oluşturan faaliyetlere bölüştürülür. Faaliyetler, belirli bir süre içinde ve belirli kaynaklar kullanılarak gerçekleştirilir. Faaliyetler, bir maliyet merkezi olarak tanımlanabilir. Faaliyet adedi projenin yönetiminde amaçlanan ayrıntı düzeyi tarafından belirlenir. Faaliyet sayısı artırıldıkça, toplanacak, depolanacak ve işlenecek veri miktarı da artacak; ayrıca takip edilecek ve raporlanacak faaliyet sayısı da artmış olacaktır. Diğer bir deyişle, faaliyet sayısının artması, proje yönetimi iş

yükünü de önemli ölçüde artırmaktadır. Faaliyet bazında, kapsam daha geniş tanımlanarak faaliyet adedi düşük tutulabilir.

2.5 Öncülük İlişkileri

Çizelgeleme, sıralamadan farklı olarak faaliyetlerin hangi sırada gerçekleştirileceği bilgisi dışında bu faaliyetlerin ne zaman başlayıp ne zaman biteceği bilgisini de içerir. Çizelgelenen faaliyetlerinin kullandığı kaynaklar kısıtlı olabildiği gibi faaliyetler arasında öncüllük ilişkileri de olabilir. Öncüllük ilişkileri teknolojik bir gereksinimi yansıttığı gibi, tamamen yönetsel bir kararın sonucu da olabilirler. Öncüllük ilişkileri, zaman bazında faaliyetler arasındaki başlangıç – bitiş ilişkilerini belirler. Bir faaliyet, öncül faaliyetlerinin hepsi bitmeden başlayamaz. Bu konuda genel kabul, bir faaliyetin, ancak bütün öncülleri tamamlanınca başlayabileceğidir. Fakat uygulamada farklı durumlar da söz konusu olabilmektedir. Örneğin, eşzamanlı (concurrent) tasarım faaliyetlerinde proje süresini kısaltmak için bazı faaliyetlerin paralel yürütülmesi istenir. Bir faaliyet sonucu elde edilen bozulabilir bir malzemeyi kullanacak olan başka bir faaliyetin, malzeme hazır olduktan sonra belirli bir süre içinde başlaması istenebilir. Bu değişik durumları modelleyebilmek için genelleştirilmiş öncüllük ilişkileri (generalized precedence relations) tanımlanmıştır. Bunlar; Başlangıç-Bitiş (SF), Başlangıç-Başlangıç (SS), Bitiş-Bitiş (FF) ve Bitiş-Başlangıç (FS) tipi öncüllük ilişkileridir. Bütün bu öncüllük ilişkileri için geçerli olmak üzere, en az bekleme süresi (minimal time lag) ve en çok bekleme süresi (maximal time lag) tanımlanmıştır. Örneğin, bir faaliyetin başlaması için diğer bir faaliyetin en azından belirli bir süre önce başlamış olması isteniyorsa, bu bir SS ilişkisidir ve en az bekleme süresi ile modellenir. Bir faaliyetin başlaması için diğer faaliyetin başlamasından sonra belirli bir süreden fazla geçmemiş olması gerekiyorsa, bu bir FF ilişkisidir ve en çok bekleme süresi ile modellenir.

2.6 Proje Çizelgeleme

Proje planlamasının ana çıktılarından biri de proje ana çizelgesidir. Bu çizelge, projeye ilişkili bütün faaliyetlerin zamana göre dağılımıdır. Proje çizelgesi projenin zaman parametrelerini belirler ve yöneticilere, proje dahilindeki çabaları doğru yöne kanalize etmede ve koordinasyon sağlamada yardımcı olur.

Çizelgeleme, kısıtlı kaynakların zaman üzerinde görevlere paylaştırılmasıyla ilgilidir. Bir ya da daha fazla amacı en iyileme hedefine sahip bir karar verme sürecidir. Kaynaklar ve görevler pek çok biçimde meydana gelebilir. **Kaynaklar**, bir iş atölyesindeki makineler, bir havaalanındaki kalkış pistleri, bir inşaat şantiyesindeki personel ya da bir bilgisayar sistemindeki işleme birimleri olabilir. **Faaliyetler veya görevler** ise bir üretim sürecindeki işlemler, bir havaalanındaki kalkış ve inişler, bir inşaat projesindeki aşamalar, bir bilgisayar programındaki komutlar vb olabilir. Her bir görev farklı öncelik düzeyine, en erken olası başlangıç zamanı ve teslim zamanına sahip olabilir. Amaçlar da farklı olabilir. Bir olası amaç, en son görevin tamamlanma zamanının en küçüklenmesi ve bir diğer söz verilmiş teslim tarihinden sonra tamamlanan işlerin sayısının en küçüklenmesidir (Pinedo 1995).

2.7 Çizelgeleme Problemlerinin Genel Yapısı

Proje çizelgeleme; akış yapısı belirlenerek şebekesi oluşturulan bir projenin sınırlı miktar ve kapasiteye sahip kaynaklar kullanılarak, en düşük toplam maliyetle gerçekleştirilmesi için her faaliyetin başlama zamanının, faaliyete tahsis edilen kaynak türlerinin (teknoloji seçeneğinin), kaynak tahsis miktarının ve kaynakların kapasite zorlamalarındaki çalışma hız ve sürelerinin belirlenmesi ile çözüme ulaşma çalışmalarının bir bütünüdür. Proje çizelgeleme en genel yapısı itibariyle, faaliyetlerin ne zaman başlayacağını tespitini ve kaynakların optimal kullanımını amaçlayan çok aşamalı bir “karar verme” problemidir. Yöneylem araştırmasında da çeşitli şekillerde tanımları yapılan karar problemlerinin yapısı şu üç temel eleman üzerine kurulmuştur.

1. Amaçlar
2. Karar değişkenleri
3. Kısıtlar

2.7.1 Amaçlar

Amaç; proje maliyetini en küçüklemek olduğuna göre bunu etkileyen faktörler; faaliyet süresi, proje süresi ve kapasite yükleme şeklinde sıralanabilir. Bir karar verme probleminde, çözüm seçeneklerinin hangi kriterlere göre ve nasıl değerlendirileceği, bir amacın oluşmasına ve bunun matematiksel bir terimle ifade

edilmesine bağlıdır. Uygulamalarda projeler, kaynak kullanımının yarattığı maliyetler, genel giderler ve projenin gecikmesinden doğan kayıpların toplamını en küçükleyecek bir amaçla yürütülmektedirler. Buna göre proje çizelgelemenin amacı; “en küçük toplam proje maliyetini elde etmek” şeklinde özetlenebilir.

2.7.2 Karar değişkenleri

Tanım olarak, çözüm esnasında beliren kesin değerleriyle problemin bir çözümünü belirleyen tüm değişkenlerdir. Proje planlamada her faaliyetin başlama zamanı ve sürelerinin tespiti, problemin bir karar değişkenini oluşturur. Çözüm esnasında bu karar değişkenlerinden birinin değerinin değiştirilmesi, çözümün farklı şekilde oluşmasına yol açar. Faaliyet süresi, kaynak tahsis miktarı ile beraber bir değişim gösterir. Bu, bir faaliyetin süresinin belirlenmesinde, uygulanan teknoloji seçeneği, kaynak tahsis miktarı ve kaynak kapasitesindeki motive edici etkileri olan arttırıcı değişimlerin birlikte rol oynaması demektir.

Karar değişkenleri şöyle sıralanabilir;

1. Her faaliyetin ‘başlama zamanı’
2. Bir faaliyette uygulanan teknoloji seçeneğinin her bir adımı
3. Bir teknoloji seçeneğinin seçiminden sonra belirlenen kaynak tahsis miktarı
4. Kaynak tahsis miktarı gibi kaynak kapasitesi de bir karar değişkenidir.

2.7.3 Kısıtlar

Tanım olarak kısıt, karar değişkenlerinin alabileceği değerleri sınırlayarak yalnız olurlu çözüm seçeneklerinin belirlenmesini sağlar. Proje çizelgeleme problemlerinin başlıca üç kısıtı şunlardır;

2.7.3.1 Öncelik kısıtları

Projenin akış şeması içerisinde, faaliyetler arasında bulunan ilişkiler sonucu meydana gelir. Faaliyetlere atanan başlama zamanı değerlerini kısıtlar.

2.7.3.2 Kaynak kısıtları

Hiçbir projede kaynaklar, miktar ve kapasite bakımından sınırsız değildir. Projenin çizelgelenmesi esnasında bağıntı kısıtları ile paralel başlayabilecek faaliyetlerin bazıları, yeterli kaynak olmadığı için, ertelenir; yani gecikir. Kaynak tahsisinde ortaya çıkan bu sorun beraberinde proje çizelgelemeye bir de “sıralama problemi”ni getirmiştir. Bu sıralamaya göre faaliyetlerin öncelik dereceleri belirlenir.

2.7.3.3 Zaman kısıtları

Projenin kendisi ve proje sürecinde projeye doğrudan bağlı olmayan elemanların her birinin gerçekleşmesi, önceden belirlenmiş konumları bulunabilir.

2.8 Proje Çizelgelemede Çözüm Teknikleri

Bu tip bir problemin çözümü için çeşitli çözüm teknikleri ve yaklaşımları kullanılmaktadır. Bu tip yaklaşımlar;

- a. Kaynak kısıtları göz ardı edilerek,
- b. Teknoloji seçeneğinin ve kaynak tahsis miktarının önceden belirlenmiş ve sabit olduğu kabul edilerek,
- c. Kaynak yüklenmesinden doğan maliyetler ihmal edilerek oluşturulmaya çalışılmaktadır.

Bu yaklaşımlar dört ana grupta toplanır:

- 1) Kritik Yol Metodu (KYM)
- 2) Zaman - Maliyet Dengeleme Yöntemi (ZMD)
- 3) Kaynak Dengeleme Yöntemi (KD)
- 4) Kaynak Kısıtlı Proje Çizelgeleme Yöntemleri (KKPÇ)
 - a) Tek Modlu Kaynak Kısıtlı Proje Çizelgeleme Yöntemi (TM-KKPÇ)
 - b) Çok Modlu Kaynak Kısıtlı Proje Çizelgeleme Yöntemi (ÇM-KKPÇ)

2.8.1 Kritik yol metodu (KYM)

KYM modelleri şu temeller üzerine kurulur:

- Kaynak tahsisi ve kısıtları dikkate alınmaz.
- Faaliyet ve bağıntı süreleri proje analizi aşamasında belirlenmiştir.

Bu esasa göre proje, bağıntı kısıtları dikkate alınarak faaliyetlerin başlama zamanlarının tespiti şeklinde bir çözüm sistemine tabi tutulur. Amaç en kısa sürede projenin tamamlanmasıdır. KYM modellerinden bazıları çok büyük projelere uygulanmış, iyi ve etkin çözümler elde edilmiştir. Ancak modelin dayandığı temeller sebebiyle tam anlamıyla gerçekleştirilebilir olduğu iddia edilemez. Fakat proje yönetiminde ilk uygulanan sayısal yöntem olması sebebiyle geniş bir uygulama alanı bulmuştur. Bu çalışmada KYM modeli ve yöntemlerine geniş bir yer verilmeyecektir. Ancak, bulanık KYM konusunda ilerleyen bölümlerde detaylı açıklamalarda bulunulacaktır.

2.8.2 Zaman - maliyet dengeleme yöntemi (ZMD)

ZMD modelinde; faaliyet süresi - kaynak tahsisi ilişkilerinin ve buna bağlı olarak en kısa ve en uzun faaliyet süreleri arasındaki indirekt faaliyet maliyeti değişiminin proje analizi aşamasında belirlendiği varsayımıyla en düşük toplam maliyeti veren proje planlaması yapılır (Barutçugil 1998).

Ancak ZMD güncelliğini kaybetmiştir. Bunun nedenleri şunlardır:

1. Kaynak kısıtları ve kapasite yüklemesi, proje programının oluşturulması esnasında gözardı edildiğinden maliyet artışları göz önüne alınarak faaliyet ve proje süresinin kısaltılmasının kapasite yüklemesini nasıl etkileyeceği ve kaynak darboğazlarına yol açıp açmayacağı belirlenemez. Bu da ZMD yaklaşımına ters düşmektedir. Çünkü programlama öncesinde faaliyet sürelerinin kapasite zorlamaları da dikkate alınarak kısaltılabileceği tahmin edilmekte ve daha sonra kapasite yüklemeleri kontrol edilememektedir.

2. Faaliyet süresi ve maliyet ilişkisi pratikte oldukça güç bir biçimde belirlenmekte ve bunun için de gerçeklerden uzaklaştırıcı, basitleştirici varsayımlar yapılmaktadır.

2.8.3 Kaynak dengeleme yöntemi (KD)

KD'de faaliyet süreleri sabit kabul edilerek ve kaynak kısıtları gözetilmeden, kapasite yüklemesinde proje süresi boyunca oluşan değişimlerin maliyetlerini en

küçükleme amaçlanır. KD'nin ideal program modeli, proje süresi boyunca sabit kapasite yüklemesi sağlamaktır. Ancak öncelik kısıtları ve farklı kaynak gereksinimleri buna pek imkan tanımaz. KD probleminde en büyük eleştiriyi “denge ölçütleri” almaktadır. Kapasite yüklemesine bağlı maliyetlerin en küçük toplamını veren proje programının, amaç fonksiyonunda hangi ölçüt kullanılarak oluşturulabileceği henüz açık değildir. Farklı denge ölçütleri ile elde edilen çözümler farklı olabilmektedir. KD ile yalnız kapasite yüklemesine bağlı maliyetler etkilendiğinden, KD, ZMD veya KYM ile elde edilen proje programları kapasite yüklemesinin dengelenmesi ile sınırlı kalmaktadır.

2.8.4 Kaynak kısıtlı proje çizelgeleme yöntemleri (KKPÇ)

KYM, ZMD ve KD; kaynakların miktar ve kapasite kısıtlarına bağlı olduğunun gözardı edilmesi, kapasite darboğazlarının oluşumunun dengelemeyle de önlenememesi gibi sebeplerle kesin çözüm elde etmede yetersiz kalmıştır. Bu ihtiyaçtan, kaynak kısıtlarını açık bir biçimde ele alan bir model doğmuştur; “Kaynak Kısıtlı Proje Çizelgeleme Modelleri”. Bu yaklaşıma ait temel kavramlar aşağıdaki kısımda açıklanacaktır.

2.8.5 Proje çizelgeleme problemlerinin sembolik gösterimi

Makina çizelgeleme ve kuyruk problemlerinin sembolik gösteriminden esinlenilerek proje çizelgeleme problemlerinin de sembolik gösterimi için iki öneri geliştirilmiştir. Bunlardan bir tanesi Herroelen vd. (1999), diğeri ise Brucker vd. (1999) tarafından önerilmiştir. Bu iki öneriden sadece Herroelen vd.'nin önerisi üzerinde kısaca durulacak ve örnekler verilecektir. Sembolik gösterim üç alandan oluşmaktadır: a/b/c. Burada, a alanı; kaynak karakteristiklerini, b alanı; faaliyet karakteristiklerini ve c alanı; performans ölçütlerini göstermektedir. Aşağıda bazı örnekler sunulmuştur. İlk iki örnekte kaynak kısıtı olmadığından birinci alan yer almamıştır.

i. $c_{pm}, c_j / N\&D$: Kaynak kısıtsız bir ortamda N&D'nin en çoklanması problemi.

ii. $\min, c_j / N\&D$: Kaynak kısıtsız ve en az bekleme süresi olan bir ortamda N&D'nin en çoklanması problemi.

iii. $m,1 / \text{cpm}, c_j / \text{NŞD}$: Yenilenebilir kaynak kısıtlı bir ortamda NŞD'nin en çoklanması problemi.

iv. $m,1 / \text{cpm} / C_{\max}$: Tek modlu KKPÇP.

v. $m,1 / \text{gpr} / C_{\max}$: Genelleştirilmiş öncüllük ilişkileri altında KKPÇP.

Burada NŞD, net şimdiki değeri ifade etmektedir. İki gösterim de halen genel kabul görmemiştir; ancak Herroelen vd.'nin gösterim önerisi daha yaygın kullanılmaktadır. Uygulama ile eksiklikler görülecek ve bu gösterimler ve belki başka önerilecek gösterimler geliştirilecektir. Böyle bir gösterime ihtiyaç olduğu kesindir. Umalım ki zaman içinde araştırmacılar tek bir gösterim üzerinde fikir birliğine varabilsinler.

2.9 Kaynak Kısıtlı Proje Çizelgeleme Modelleri (KKPÇ)

KKPÇ modelleri genel olarak iki ana grupta incelenebilir;

1. Kaynak Tahsisi ve Faaliyet Sürelerinin sabit olduğu varsayımına dayanan modeller; Tek Modlu (TM-KKPÇ)
2. Kaynak Tahsisi ve Faaliyet Sürelerinin değişken olduğu varsayımına dayanan modeller; Çok Modlu (ÇM-KKPÇ)

2.9.1 TM-KKPÇ modelinin özellikleri

Faaliyet süreleri, proje analizi aşamasında belirlenmiş ve sabittir. Böylece proje planlama çalışması, faaliyet başlama zamanlarının tespiti ile devam etmiş olur. Faaliyet süresinin sabit kalmasıyla birlikte, teknoloji seçeneği, kaynak tahsis miktarı ve endirekt faaliyet maliyetinin de belirlenmiş olduğu kabul edilir.

TM-KKPÇ modelinin amacı, proje süresine bağlı maliyetleri en küçükleme veya proje süresini en küçükleme. Aslında her ikisi de aynı sonuca ulaşır. Zira proje süresini kısaltmakla, proje süresine bağlı maliyetleri azaltmak eş anlamlıdır. Burada tanımlanan TM-KKPÇ problemine, literatürde kaynak kısıtlı proje çizelgelemenin “klasik problemi” adı verilmektedir. Bu alanda, üzerinde en yoğun çalışmaların yapılmış olduğu problem TM-KKPÇ problemidir.

Bu çalışma kapsamında TM-KKPÇ problemleri bulanık yapıda ele alınmıştır. Ancak deterministik TM-KKPÇ problemlerinin çözümü için geliştirilen optimal ve sezgisel

yaklaşımlar bu tip modeller dikkate alınarak daha geniş bir biçimde (Atlı 2005)'de açıklanmıştır.

2.9.2 ÇM-KKPC modelinin özellikleri

ÇM-KKPC modelindeki en belirgin özellik faaliyet sürelerinin sabit olmamasıdır. Bu nedenle, bir faaliyetin süresi, kaynak tahsisinin belirlenmesi ve faaliyet başlama zamanının tespitiyle eş zamanlı olarak saptanır. ÇM-KKPC modelleri; belirli bir teknoloji ile yerine getirilen bir faaliyetin süresinin, kaynak tahsis miktarı ile değiştiği veya belirli bir kaynak tahsis miktarı için faaliyet süresinin, teknoloji seçeneği ile değiştiği durumlarda kullanılabilir. Açıklandığı üzere, kaynak tahsis miktarının değişken olması, problemi daha karmaşık bir yapıya getirmektedir. Proje amacı ise, yalnız proje süresine bağlı maliyetlerin (ya da proje süresinin) veya hem proje süresine hem de faaliyet sürelerine bağlı maliyetlerin toplamını en küçükmektir. Görüldüğü gibi karmaşıklığın artmasıyla, oluşturulabilecek model sayısında da bir artış olacaktır.

2.9.3 Tek proje (TP) ve çoklu proje (ÇP) modelleri

KKPC modelinde tek projenin (TP) dikkate alınması halinde şu ek varsayımlar söz konusudur:

- Bir işletmede aynı anda sadece bir proje yürütülür ve biri bitmeden diğerine başlanamaz,
- Bir işletmede aynı anda birden fazla proje yürütülürken, projeler arası kaynak kullanımı yapılmaz (kaynaklar önceden ayrılmıştır).

Örnek olarak, uçak bakımı yapılan bir hangarda faaliyetler art arda devam etmek zorundadır. Uluslararası ihalelere katılan bir firma, yurt dışından alacağı işlerde kullanacağı kaynağı için de aynı uygulamayı yapar. Uygulamada ise, bir işletmedeki kaynaklar o anki proje sayısına göre paylaşılarak kullanılır.

Her proje kendi içinde bir alt projeden oluşmaktadır. Bu yüzden temelde tek proje (TP) ile çoklu proje (ÇP) arasında bir fark olduğu söylenemez. Projenin planlanması esnasında TP veya ÇP varsayımları, karar değişkenlerini ve kısıtları etkilememektedir. Ancak amaç için aynı şey söylenemez. ÇP durumunda, tüm proje amaçlarının aynı olduğu düşünülürse (en düşük proje maliyeti gibi), işletmede proje

süresi boyunca statik bir yapı olacaktır. Yani işletme, bu süreç içerisinde yeni bir iş alamayacaktır. Ancak işletmedeki yapı statik değil dinamik ise, TP modelindeki en küçük toplam proje maliyeti amacı, ÇP modelinde geçersiz olacaktır. Çünkü bu durumda her projenin kaynak yükleme maliyeti ayrı olmak zorundadır. Bu yüzden ÇP modelinde projelerin süreye bağlı amaçlarının ayrılığı gözetilmekte; ancak aynı anda işletmede kapasite kullanımının en yüksek düzeyde olması istenmektedir.

Kaynak kısıtlı birden çok projenin çizelgelenmesi, uygulamadaki gerçek durumu yansıtması bakımından, KKPC'de önemli bir araştırma alanıdır. Örneğin, değişik şantiyelerde iş yapan bir inşaat şirketi, bu şantiyelerde ortak ve kısıtlı kaynaklarını kullandığından, tipik olarak bu proje çizelgeleme modeline uygun bir görünüm sergiler. Aynı şekilde, bir otomotiv firmasının aynı anda birden fazla ürün geliştirme projesinde, ortak ve kısıtlı kaynaklarının kullanıldığı ortamın planlanması da kaynak kısıtlı birden çok projenin çizelgelenmesini zorunlu kılar. Kaynak kısıtlı birden çok projenin çizelgelenmesi için iki temel yaklaşım vardır: (i). Her projenin kendi başına ele alınması (diğer bir deyişle, her projenin kendi başlangıç ve bitiş düğümünün olması); (ii). Tüm projelerin bir proje halinde bütünleştirilmesi (diğer bir deyişle, tüm projeler için bir başlangıç ve bir bitiş düğümü olması).

Çizelgeleme için amaç işlevi, iki tür modelleme için de temelde aynıdır: Kaynakların kısıtlı olması sonucu oluşan gecikmenin; yani kaynak kısıtlı çizelgeleme ile elde edilen proje süresi ile kaynak kısıtsız (KYM) proje süresi arasındaki farkın bir işlevidir. Her projenin kendi başına ele alınması durumunda, gecikme her bir proje için ayrı ayrı hesaplanır ve proje sayısına bölünerek ortalaması alınır (MPD- mean project delay).

Tüm projelerin tek bir proje halinde birleştirildiği durumda ise, tek proje boyutunda gecikme hesaplanır ve tek proje boyutundaki KYM, proje süresine bölünerek oransal bir değer olarak kullanılır (MDI- multiproject duration increase). Tüm projelerin tek bir proje halinde bütünleştirilmesi durumu için tek proje çizelgelemede, sürenin en küçüklenmesinde başarılı olunmuş ve LFT, LST, MinSlack gibi sezgisel kurallar kullanılmıştır. Her projenin kendi başına ele alındığı modelleme durumunda, tek proje için geliştirilmiş sezgisel kuralların, bu duruma uygulamasına göre, daha iyi sonuçlar verdiği belirtilmektedir. (Kurtulus ve Davis, 1982; Kurtulus ve Narula, 1985).

2.9.4 Tek kaynak (TK) ve çok kaynak (ÇK) modelleri

KKPÇ’de kullanılan kaynak türü sayısı modelin yapısını deęiřtirmmez. Ancak ÇK modelindeki deęiřken ve kısıtların sayısı bir hayli artarak problemin olurlu çözümlerinin bulunmasını zorlařtırır ve çözüm süresini uzatır.

KKPÇ modelini, TK veya ÇK ayırımına uygun tasarlarırken,

- projenin özelliklerine,
- kaynakların kısıtlılık özelliklerine, dikkat etmek gerekir.

Atölye tipi üretimde bir faaliyetin gerçekleştirilmesinde tek kaynak türü kullanılabilir. Ancak daha geniş kapsamlı işletmelerde projelerin yürütölmesi esnasında faaliyetler, sorumlu uzmanlar tarafından yapılır. Örneęin; kaynak faaliyetini yapan kaynakçı, elektrik tesisatını döřeyen elektrikçi gibi...

2.9.5 Sabit veya deęiřken kaynak kısıtlı KKPÇ modeli

Kullanılabilir kaynak miktarı ile belirlenen kaynak kısıtları sabit veya deęiřken olduęunda KKPÇ modelinin yapısı bundan etkilenmez. Ancak kurulan modeldeki kısıtların sayısı deęiřir. Gerçekçi bir modelin oluşumu için kaynak miktarında kısa ve orta vadede deęiřimlerin olabileceğini hesaba katmak gerekir.

2.9.5.1 Bölünebilir faaliyetli KKPÇ modeli

KKPÇ modellerinde faaliyetlerin başladıkları andan bitene kadar, kesintisiz, devam edecekleri düşüncesi gerçeęi tam olarak yansıtmamaktadır. Uygulamada bazı faaliyetlere çeřitli nedenlerle ara vermek gerekebilir veya teknolojik açıdan böyle bir zorunluluk bulunabilir. Bu yüzden modelin oluşumu esnasında, bölünebilecek faaliyetleri, hangi kořullarda bölünecekleri ve hangi faaliyetlerin kesintisiz yapılması gerektięi ayrıntılı biçimde belirtilmelidir. Böyle bir model, uygulamada maliyeti artıracaęından, kaynak darboęazının giderilmesi amacıyla faaliyetleri, çok kısa vadeli acil önlemler maksadıyla bölmek yerinde olacaktır. Böylelikle bir model, KKPÇ modelleri üzerinde ancak kısa periyotlarla yenilenen bir model oluşumu esnasında kullanılabilir.

2.9.5.2 Sabit veya deęişken kaynak tahsis yoğunluęu bulunan KKPC modelleri

Bir faaliyete birim zamanda tahsis edilen kaynak miktarının projenin uygulama devresi içerisinde azalması veya artması gerekebilir. Örnek olarak 1000 adam/saatlik bir kazı çalışması yapılırken;

$$2 \times 8 = 16 \text{ adam/saat}$$

$$5 \times 8 = 40 \text{ adam/saat}$$

$$20 \times 8 = 160 \text{ adam/saat}$$

$$30 \times 8 = 240 \text{ adam/saat}$$

..

..

..

1000 adam/saat şeklinde deęişik sayıda işçilerle tamamlanabilir.

Böyle bir varsayım KKPC modeline çözüm esneklięi kazandırması, kararlı bir model oluşturması, dengeli ve yüksek bir kapasite kullanımı yaratması açısından oldukça önemlidir. Ancak yapının bu şekilde gittikçe daha karmaşık bir hale geldięi durumlarda, olurlu çözümlerin sayısı da olabildiğince artmaktadır. Yukarıdaki açıklamalar ışığında ÇM-KKPC/ÇK modelinin gerçek uygulamalara en yakın model olduęu söylenebilir. Ancak herhangi bir modelin etkinlięini deęerlendirirken onu çözmede kullanılabilecek yöntemlerin varlıęı ve etkinlięi de dikkate alınmalıdır. Bu çalışmada ise bulanık ortamda ÇM-KKPC/ÇK modelleri üzerinde durulacaktır.

2.9.6 KKPC modellerinin ortak özellikleri

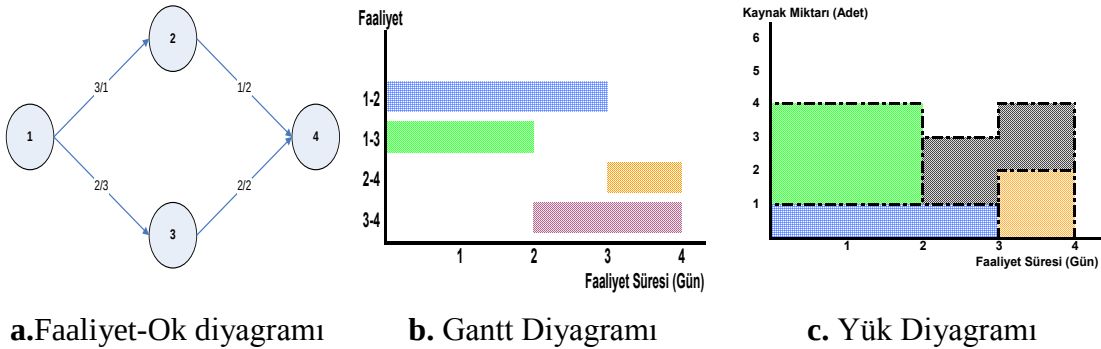
1. Tek Kaynak (TK) ve Çok Kaynak (ÇK) Modelleri; TK modellerinde bir faaliyetin sadece bir kaynak türünden çeşitli miktarda kaynak tahsis miktarı ile sağlanabileceęi kabul edilir. ÇK modellerinde ise faaliyetler farklı türden kaynakların birlikte kullanılması ile gerçekleşir.

2. Tek Proje (TP) ve Çok Proje (ÇP) Modelleri; ÇP modelinde, bir işletmede birden fazla proje aynı anda yürütölmektedir ve bu projeler işletmenin tüm kaynak varlıklarını birlikte kullanır. TP modelinde ise, işletmenin tüm kaynakları bir tek projeye ayrılmıştır.

3. Proje süresi boyunca kullanılabilir kaynak miktarı sabit veya değişkendir.
4. Bir faaliyet, aralıklarla veya aralıksız olarak tamamlanabilir.
5. Bir faaliyete tahsis edilen kaynak miktarı sabittir veya belirli aralıklar içinde değişebilir.

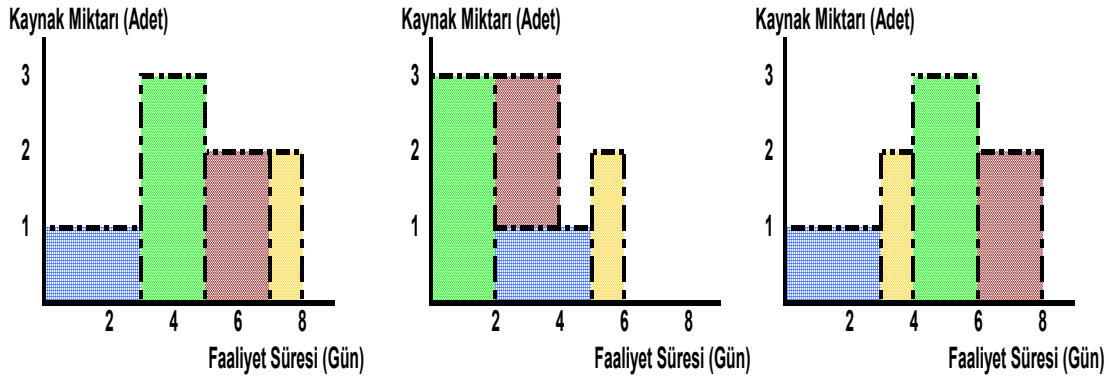
2.9.7 TM-KKPC ve CM-KKPC modellerinin karşılaştırılması

TM-KKPC modelinde, faaliyet sürelerinin sabit olduğu kabul edilmektedir. Bu kabul, modeli oldukça basitleştirmekte, değişken sayısını azaltmakta ve kaynak tahsisindeki esnekliği ve hassasiyeti büyük miktarda azaltmaktadır. TM-KKPC modelinde, kaynak gereksiniminin azaltılabileceği, kapasite zorlamaları ile kaynak darboğazlarının aşılabilmesi dikkate alınmadığı için kaynak darboğazlarının aşılması, darboğazı yaratan faaliyetin veya faaliyetlerin geciktirilmesi ile sağlanır. Örnek olarak Şekil 2.5’de dört faaliyetli bir şebekede KYM ile sağlanmış ve diyagramlarda görülen çözüm, 3 ile sınırlı olan kaynak miktarı 1., 2. ve 4. zaman dilimlerinde aşıldığı için gerçekleştirilememiştir.



Şekil 2.5 : Sabit Kaynaklı TM-KKPC Örnek Problem.

Yukarıdaki örnekte faaliyetlere tanınan önceliklerle birlikte olurlu çözüm sayısı dördü geçer. Bunlardan üç tanesi Şekil 2.6’da gösterilmiştir. Amaç en kısa proje süresi olduğuna göre 2. çözüm seçeneğinin optimal (en iyi) çözümü verdiği ortadadır. Örnekte de görüleceği üzere kaynak darboğazlarının giderilmesi oldukça kısıtlı önlemlerle gerçekleştirilir. Oluşturulan çizelgeler ise değişikliklere karşı oldukça duyarlıdır.



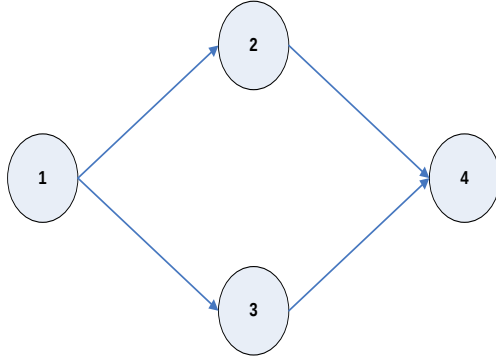
a. 1'inci Çözüm Seçeneği b. 2'nci Çözüm Seçeneği c. 3'üncü Çözüm Seçeneği

Şekil 2.6 : Sabit Kaynaklı TM-KKPC Örnek Probleminin Olurlu Çözümleri.

Öngörülme bir nedenden dolayı bir faaliyette, kaynak tahsis miktarında veya zamandaki bir değişiklik, çizelgenin tamamını etkilemekte ve son şartlar altında yeni bir çizelge hazırlama zorunluluğu doğmaktadır. Uygulamada kaynak darboğazları; faaliyetlerin kaydırılması, teknoloji değişikliği ile kaynak ikamesi yapılması, kaynak tahsis miktarlarının belirlenmiş limitler içinde değiştirilmesi ve kapasite zorlamaları gibi tedbirlerle de aşılabılır. ÇM-KKPC modeli kapsamına girmeye başlayan bu tedbirlerde, faaliyet süresi ve kaynak tahsisinin esneklik kazanması, kaynak kısıtları ile modelleme üzerinde daha çekici bir rol oynar.

Yukarıdaki örnek, belirtilen hususlar altında ÇM-KKPC modeline uyarlandığında, faaliyet sürelerinin (d), kaynak tahsis miktarının (r), iki seçeneğine göre belirli değerler aldığı görülmektedir. Şekil 2.7 (c), (ç), (d)'de problemin 3 olurlu çözümü görülmektedir. En büyük kapasite yüklemesi ve en kısa proje süresini veren (c), (ç) çözümleri problemin bunlara bağlı maliyet kalemlerinin en küçük toplamını veren optimal çözümleri sağlar. Bu iki çözüm arasındaki fark ise, faaliyet sürelerinin etkilediği maliyetlerin tanımlanması ile ortaya çıkar.

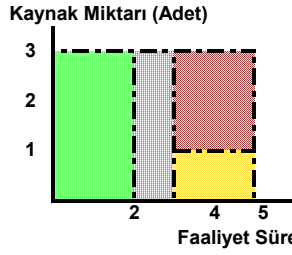
Bu örnekte de görüldüğü üzere ÇM-KKPC modelleri esnek bir yapıdadır. Çünkü 1-2 ve 1-3 faaliyetlerinde kaynak tahsis seçeneklerinin değişmesi, proje süresini ve kapasite yüklemesini değiştirmemektedir. Ancak bu esnekliğin yanında modelin bir zorluğu da, çıkabilecek olurlu çözüm seçeneklerinin çokluğudur. Yukarıdaki basit örnekte bile bu seçenek sayısı 10'u aşmaktadır.



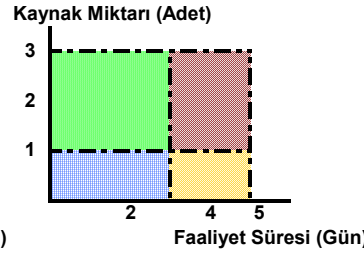
Faaliyetler	1'inci Seçenek		2'nci Seçenek	
	r	d	r	d
1-2	1	3	3	1
1-3	3	2	2	3
2-4	2	1	1	2
3-4	2	2	*	*

a. Faaliyet Şebekesi

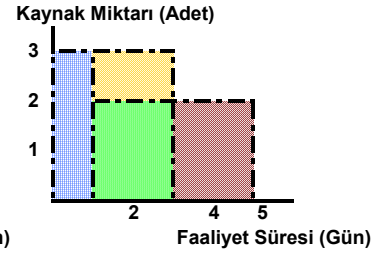
b. Faaliyet Süre ve Kaynak Listesi



c. 1'inci Çözüm



ç. 2'nci Çözüm



d. 3'üncü Çözüm

Şekil 2.7 : ÇM-KKPC için Örnek Problemin Olurlu Çözümleri.

Proje planlama safhasında oluşturulacak modelin;

- Gerçekleşebilir olması, önemli kısıt ve değişkenleri içermesi,
- Kararlı bir yapıya sahip olması, küçük farklılıklardan etkilenmeyecek biçimde oluşması,
- Tasarlanan amacı sağlaması gerekir.

Bunlara ek olarak bir modelin ne derece gerçekçi olduğu ve kaynak tahsisinde ne derece esnek olabildiği de incelenmelidir. Buna göre yukarıdaki iki farklı model aşağıdaki gibi karşılaştırılabilir:

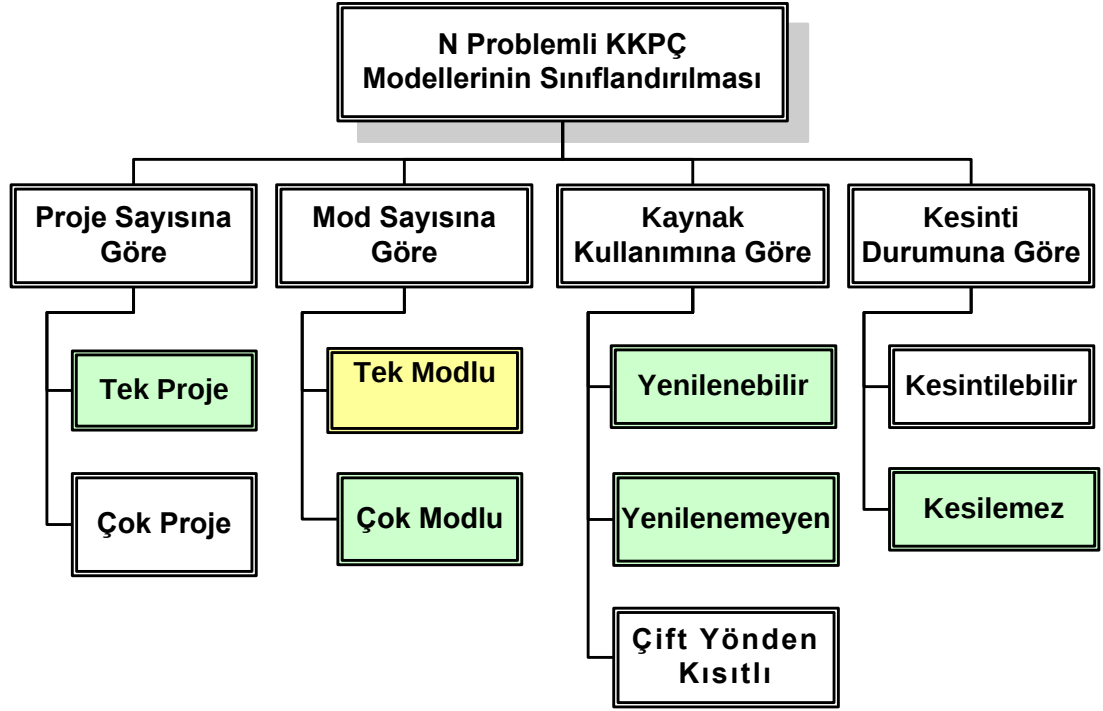
- TM-KKPC modelleri problemin çözümünde gerçekçi bir model oluşturamamıştır.
- TM-KKPC modelleri, kaynak tahsisinde bir esnekliğe sahip olmaması, amaç kriterinin sadece proje süresi veya proje süresine bağlı maliyetlerin en küçüklenmesi şeklinde olması ve endirekt faaliyet maliyetlerinin sabit olması veya ihmal edilmesi gibi sebeplerle ÇM-KKPC'ye göre yetersizdirler.
- ÇM-KKPC modelinde, modelin oluşturduğu esnek yapı sebebiyle çözüm seçeneklerinin bir hayli fazla olması, çözümü güçleştirmektedir.

2.10 Proje Çizelgeleme Problemlerinin Sınıflandırılması

Yukarıda açıklanan KKPC modelleri genel olarak iki ana gruba ayrılmakla birlikte literatürde farklı problem yapılarıyla karşılaşmak mümkündür. Bunlar;

- Klasik KKPC Problemleri
- Minimal ve Maksimal Zaman Kaydırmalı KKPC Problemleri
- Çok Modlu KKPC Problemleri
- Minimal ve Maksimal Zaman Kaydırmalı Çok Modlu KKPC Problemleri
- Minimal ve Maksimal Zaman Kaydırmalı Kaynak Yatırım Problemleri

KKPC modelinin birçok varyasyonu üzerinde yapılan testlerle son otuz yıldır incelenmektedir ve bu testler problem karakteristiklerini temel çeşitli sınıflandırmalar yapılabilir. Mesela projeler, işletim modu, kaynak tüketim ve kesikli sayısına göre. (Şekil 2.8'e bakınız). İşletim moduna göre; KKPC modelleri tek modlu ve çok modlu iki gruba ayrılabilir. Çok modlu sınıflandırmada, her bir faaliyet birkaç işletim modundan birinde icra edilebilir. Kaynak tüketimine göre, KKPC modellerinin kaynakları sınırlıdır fakat dönemsel itibariyle yenilebilir veya tüm proje boyunca miktarı sınırlı olan yenilemeyen kaynaklar ile hem dönemler itibariyle hemde tüm proje süresince çift taraflı sınırlı olan çift yönden sınırlı kaynaklar olabilir. Kesikli duruma göre KKPC modelleri faaliyetlerin bölünebilir veya bölünmez oluşuna göre gruplandırılır. Faaliyetlerin bölünemez olduğu durumda, faaliyet başlatıldığında yarıda kesilmesine müsaade edilmez. Bölünebilir faaliyetlerin olduğu durumda bunun tam aksine faaliyetlerin bölünmesine izin verilebilir. Ayrıca KKPC modellerinin çeşitli versiyonlarında değişik amaç fonksiyonlarına görede ayrı bir gruplandırma yapılabilir; örneğin proje tamamlanma süresinin minimizasyonu, proje toplam maliyetinin minimizasyonu ve proje net şimdiki değerinin minimizasyonu olarak da sınıflandırılabilir.



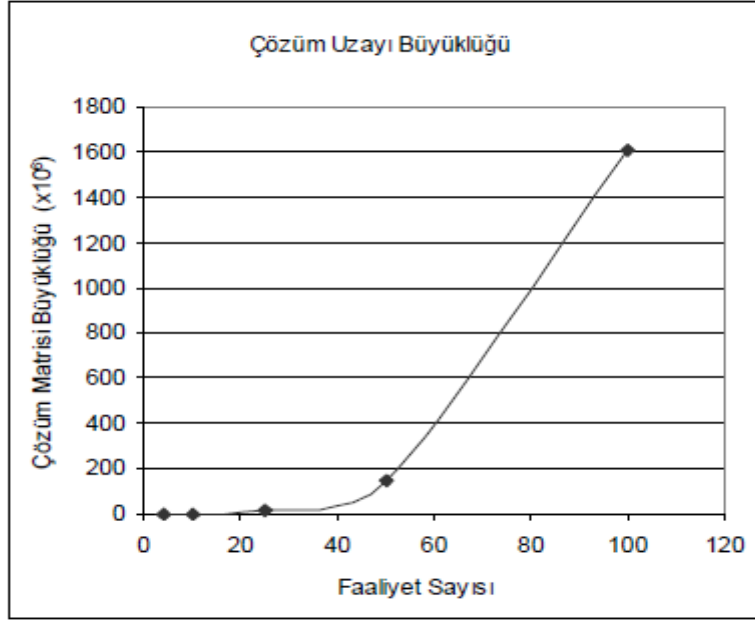
Şekil 2.8 : KKPC Problemlerinin Sınıflandırılması.

2.11 Değişik Amaç İşlevleri

KKPC probleminde değişik amaç işlevleri kullanılmıştır. Proje yöneticisi açısından bakıldığında en önemli performans ölçütlerinden birisi, projenin süresidir. Projenin elde mevcut kaynaklarla en az sürede bitirilmesi hedeflenir. Kaynak kısıtlarının olmadığı varsayılan ilk proje çizelgeleme gösterimlerinde kullanılan amaç işlevi, proje tamamlanma süresinin (PTZ) enazlanmasıdır (Malcolm, vd., 1959; Kelley, 1961). KKPCP’nde de en yaygın olarak kullanılan amaç işlevi proje süresinin enazlanmasıdır (Özdamar ve Ulusoy, 1995). Projenin bir termini olması durumunda, termin bazlı başka amaç işlevleri de incelenmiştir. Bunlardan bir tanesi, gecikmenin (tardiness) enazlanmasıdır. Bir diğeri de, tam-zamanında-üretim anlayışına yönelik olarak, erken bitirme ve gecikme toplamının enazlanması olan amaç işlevidir (Vanhoucke vd., 2000b). Burada, erken bitirme ve gecikmenin ayrı ayrı etkileri olması durumunda, ağırlıklı toplam enazlanır. Diğer sık kullanılan bir amaç işlevi de toplam maliyet veya toplam kârın en çoklanmasıdır (Herroelen vd., 1997). Bu amaç işlevi kullanılırken, nakit akışına paranın zaman değeri uygulanarak bugünkü değeri bulunur ve maliyet veya kâr, net bugünkü değer (NBD) olarak ifade edilir. Proje yönetimi ve toplam kalite anlayışını ilişkilendirmek amacı ile, projede yeniden işlem

süresi ve maliyetini tanımlayarak enazlayan bir amaç işlevi de kullanılmıştır (İçmeli-Tükel ve Rom, 1997; Erengüç ve İçmeli-Tükel, 1999).

Çözüm uzayı büyüklüğünün faaliyet sayısına (n), kaynak sayısına (m), ve planlama periyoduna (k) göre elde edilen verilere dayanarak proje faaliyet sayısındaki artışa bağlı olarak çözüm uzayı matrisinin büyüklüğünde oluşan artış grafiksel olarak Şekil 2.9'da gösterilmektedir.



Şekil 2.9 : Çözüm Uzayı Büyüklüğü.

Bu tez çalışmasında KKPC problemlerinde ileri konular ve çözüm yöntemleri ele alınmıştır. Bu konuların neler olduğundan bahsetmenin faydalı olacağı kanaatindeyiz. Elbetteki bu problemlerin her birinden, bu çalışma kapsamında detaylarına inmek mümkün olmadığından, kısaca bahsedilecektir. Uygulama bölümünde bulanık çok modlu KKPC problemleri, detaylarıyla ele alınacak olup, çözüm yöntemi ve geliştirilen Microsoft Visual C# 2008 programında modüler olarak geliştirilen uygulama tanıtılacaktır. Bu bölümde çeşitli KKPC problemleri, gelecek yeni araştırmacılara yol göstermesi ve geliştirdiğimiz modüler mimariye sahip programın geliştirilmesine olanak sağlaması açısından ayrıntılı bir biçimde açıklanmaya çalışılacaktır.

3. LİTERATÜR VE BOŞLUK ANALİZİ

Proje yönetimi, proje çizelgeleme ve en genel ifadeyle KKPÇM üzerine pek çok çalışma yapılmıştır. Bu kapsamda yer alan KKPÇ problemlerinin modellenmesi ve çözümlenmesine ilişkin çalışmalar halen devam etmektedir. KKPÇ problemlerinin çözüm yöntemleriyle ilgili yaklaşımlar, problemin çok geniş olurlu çözümleri arasından en iyisinin seçilmesi esasına dayanmaktadır. Yapılan araştırmalarda geliştirilen yaklaşımlar, KKPÇ problemlerine uygulanarak, kısa sürede optimal ve optimale yakın çözümler bulunmaya çalışılmaktadır. Ancak yapılan çalışmaların kapsamı, genel geçerliliği kabul edilmiş yaklaşımlarla yapılan karşılaştırmalardan ibarettir. KKPÇP'nin, optimizasyon yöntemleriyle, çözümünün zor olması ve herhangi bir sezgisel yaklaşımın henüz kabul görmemesi nedeniyle problem üzerindeki araştırmalar halen sürmektedir. KKPÇP, bir projedeki faaliyetlerin tamamlanma (ya da başlama) zamanının belirlenmesini öngörür; öncelik ve kaynak kısıtlarını dikkate alır ve projenin maliyeti ve tamamlanma zamanı en küçükmeye çalışır. 1970'lerin sonlarından itibaren proje çizelgeleme hakkında yapılan birçok çalışma, KKPÇP için çözüm teknikleri geliştirmek üzerine yoğunlaşmıştır ve çok sayıda optimal ve sezgisel çözüm teknikleri geliştirmiştir. Bu bölümde genel ifadesiyle KKPÇ problemleri için en iyi sonucu vermeye çalışan ve sezgisel yaklaşımlar sunan çalışmalardan bazıları özetlenmiş ve daha geniş özetler Atlı (2005)'da gösterilmiştir. Bu çalışma kapsamında ise, KKPÇ problemlerinde ileri konular ele alınacak olup, KKPÇ ve ÇM-KKPÇ problemlerine deterministik/crisp ve belirsizlik ortamında bulanık çözümler aranmıştır. Önerilen çözüm yönteminin tanıtımı; ileri bölümlerde test uygulamaları ile geliştirilen paket programa yönelik sayısal analiz ve uygulama bölümünde yapılacaktır.

Bu bölümde öncelikle bugüne kadar literatürde yapılan çalışmalar gözden geçirilmiş ve çalışmanın literatürdeki yerinin ve olası katkılarının tespit edilmesi amaçlanmıştır. Bu amaçla, proje çizelgeleme alanı bir bütün olarak araştırılmış ve literatürde yapılan çalışmalar sınıflandırılarak, proje çizelgeleme problemi taksonomisi daha da geliştirilmiştir. Bunu, genel ifadesiyle KKPÇ modellerine yönelik literatürde yapılan

alışmalar ve boşluk analizi alışmaları gruplandırılarak anlatılacaktır. Tez alışması kapsamında ele alınan konular safha safha her bir aşaması irdelenerek ve giriş bölümünde anlatıldığı şekilde kolaydan zora doğru yönlendirilerek incelenecektir. Literatür alışmaları da bu paralelde ele alınıp, açıklanacaktır. İlk kısımda, KKPC yöntemlerinde yapılan alışmalar; ikinci kısımda, bulanık KYM’na yönelik yapılan alışmalar; üçüncü kısımda, bulanık KKPC yöntemleri; dördüncü kısımda ÇM-KKPC modelleri ve bulanık çok modlu KKPC yöntemlerine yönelik yapılan alışmalar yer almaktadır. Üzerinde alışılmakta olan bulanık faaliyet süreli KKPC probleminin mevcut alışmalar içerisinde nerede yer aldığı ve problemin incelenmesinde ve özölmesinde hangi yöntemlerin kullanıldığı araştırılmıştır.

3.1 KKPCM ile İlgili Yapılan alışmalar

KKPC problemleriyle ilgili yapılan ilk alışmalar Elsayed ve Patterson tarafından yapılmıştır.

Elsayed (1982); literatürde “Brooks Algoritması” olarak tanınan ve bir faaliyetin çizelgelenmesinde proje başlangıç-bitiş düğümü arasındaki yolların süresini öncelik ölçütü olarak kullanan yöntem üzerinde, zaman ve kaynak bazlı eşitli sezgisel kurallar ve parametrik özüm yöntemleri geliştirerek bunların etkinliklerini araştırmıştır.

Talbot (1982), faaliyet listesi üzerinde tanımladığı bir sayılama yöntemi önermiştir. Bu yöntemde, faaliyet listesinde en yüksek önceliğe sahip bir sonraki faaliyet atanmakta ve en erken zamanda başlatılmaktadır. Eğer faaliyet, zaman aralığı içinde atanamazsa, yöntem bir önceki öncüle geri dönmekte ve o faaliyetin daha geç bir zamanda veya daha yavaş bir modda atanması olanağını araştırmaktadır.

Talbot (1982), Sprecher ve Drexl (1998) kendi dal-sınır yöntemlerinin uygulanmasına bir zaman sınırı koyarak, bunların sezgisel bir özüm yöntemi olarak kullanılmasını önermişlerdir.

Patterson vd. (1984), en fazla 30 işleme sahip 34 gerçek proje ve 13 kaynak türünden oluşan bir sistemde, belirli bir zaman aralığı içinde projelere rassal başlama zamanları verildiği varsayımıyla, öncelik kurallarının (ÖK) eşitli amaç fonksiyonlarına bağlı olarak etkinliklerini incelemiştir. Toplam proje süresinin minimizasyonu amacıyla yapılan çizelgelemede “en kısa işlem süresi” kuralını, 110

proje üzerinde, değişik kurallar ile sağlanan çözümleri kesin çözümlerle karşılaştırarak, “en küçük toplam bolluk” kuralının çoğunlukla daha iyi sonuç verdiğini, ayrıca “en küçük toplam bolluk” ile “en küçük geç başlama” kurallarının aslında birbirlerinden farklı olmadıklarını kanıtlamışlardır.

KKPÇ ile ilgili literatürde önerilen kıyaslama problemlerinin bir dönem en yaygın kullanılanı, Patterson'un (1984) 110 problemi olmuştur. Alvarez-Valdes ve Tamarit'in (1989) geliştirdiği sınaama problemleri de kıyaslama problemleri olarak yaygın kullanılmıştır. Ancak, daha genel bir sonuca ulaşmak için, sınaama problemleri türeteçlerinin geliştirilmesi üzerinde durulmuştur. Bunların en çok kabul göreni, Kolisch vd. (1995) tarafından geliştirilen ve “PROGEN” olarak ifade edilenidir. Kolisch ve Sprecher (1996), “PSPLIB” olarak isimlendirdikleri bir kıyaslama problemleri kümesi oluşturmuş ve bunu kullanıma açmışlardır. Gerek PSPLIB gerekse Patterson (1984), Alvarez-Valdes ve Tamarit (1989) ve Boctor (1993) kıyaslama problemlerine, <http://129.187.106.231/psplib/> web adresinden ulaşılabilir (Kolisch vd., 1999). Bugün için en çok kullanılan kaynak budur. Başka problem verisi türeteçleri de Demeulemeester vd. (1993) ve Vanhoucke vd. (2000) tarafından önerilmiştir.

Boctor (1993) ise, uyarlanmış bir paralel çizelgeleme uygulamasıdır. Buna göre, öncüllük ilişkileri açısından olurlu ve en az bir modu ile kaynak kısıtları açısından olurlu bir faaliyet karar kümesine dahil edilmiştir. Faaliyetler “MinSlack Kuralı”na, modlar ise “SPT Kuralı”na göre belirlenmiş ve birden çok geçiş uygulanmıştır.

Punne ve Aneja (1995), atama çizelgeleme problemleri için iyi nitelikli çözümler üreten bir tabu araması önermiştir. Kaynak kısıtlı tahsis problemleri için, bir TA algoritması sunarak, bunun o an var olan algoritmalarından daha üstün çözümler sağladığını göstermiştir.

İçmeli vd. (1995), “Subrutine Kaynak Optimizasyonu” diye adlandırılan genel amaçlı optimizasyon paketinin, büyük KKPÇ çözümü için geçerli bir yöntem olduğunu göstermiştir. Ayrıca bu çalışma, büyük nakit akışlı KKPÇP'lerin optimal çözümü için ilk çabayı teşkil etmektedir. Literatürde, KKPÇP ve nakit akışlı KKPÇP için, yaygın olarak, “0-1 Tamsayılı Doğrusal Programlama (TDP) Formülasyonu” kullanılmıştır. Burada 0-1 tamsayı değişkenleri, faaliyetlerin tamamlanma (veya başlama) zamanlarını belirlemek için tanımlamışlardır.

KKPÇ problemi hakkında, literatürdeki en kapsamlı çalışmalardan birisi, Özdamar ve Ulusoy (1995) tarafından yapılmıştır. Bu çalışmada yazarlar önemli çalışmalarını inceleyerek sınıflandırmışlar ve bu çalışmalarını, çok sayıda örnek problem kullanarak test etmişlerdir. Sonuç olarak ortaya KKPÇ problemini pek çok farklı yönden ele alan yaklaşımlar ve bu yaklaşımların detaylı analizlerini içeren kapsamlı bir çalışma çıkmıştır. Yazarların çalışmalarına göre; araştırmacıların çoğunluğu, zaman esaslı amaçları çözmeye yönelmişken, maliyet esaslı amaçları çözmeye çalışan araştırmacıların sayısı oldukça düşük olmuştur. Yazarlar, bunun sebebini çoğunlukla maliyet esaslı amaçların çözümünde karşılaşılan problemlere dayandırmışlardır. Bu sebeple bu konuda maliyet veya kar esaslı modellerin temel alınması gerektiği sonucuna varılmıştır.

Yine Özdamar ve Ulusoy'un (1995) çalışmasında maliyet esaslı modeller, üç gruba ayrılmıştır. Birinci gruptaki problemlerde; kaynak kısıtları dikkate alınmamış, sadece faaliyetler, öncelik sırasına göre, net şimdiki değer, en yüksek alınacak şekilde çizelgelenmiştir. İkinci grup modellerde; projede yenilenebilir kaynaklar olduğu, yani kaynakların her zaman biriminde yeniden tam kapasiteyle başladığı varsayılmıştır. Üçüncü grup modellerde ise yenilenemeyen kaynaklar kullanıldığı düşünülmüştür. Gerçek hayat projelerinde, her zaman kaynak kısıtlamaları bulunmakla birlikte, genellikle yenilenebilen ve yenilenemeyen kaynaklar bir arada bulunmaktadır.

Özdamar vd. (1998) varolan optimal teknikleri ve KKPÇ problemleri için farklı tiplerdeki çizelgeleme sezgisel araçlarının bir çoğunu incelemişlerdir. Proje süresi ve nakit akışlarının "Net Şimdiki Değer (NŞD)"i arasındaki ilişki, Özdamar vd. (1998) tarafından, ayrıntılı olarak incelenmiştir. Modelde nakit akışları, olaylara karşı gelen düğümlerde meydana gelmektedir ve proje serimi üzerine yayılmaktadır. Proje sonundaki ödeme ise, toplam ödemenin % 10-15'i dolaylarında tutulmaktadır ve birim zaman başına bir gecikme cezası uygulanmaktadır. Yine aynı çalışmada 16 sezgisel çizelgeleme kuralı kullanılmıştır. Bunların yedi tanesi, NŞD bazlı; üç tanesi, gecikme bazlı; dört tanesi, NŞD ve gecikme bazlarının karışımı ile elde edilmiş melez (hybrid) bazlı kuralları şeklindedir ve iki tanesi de LCBA uygulamasıdır. Kaynak kısıtlarının sıklığı ve termin sıklığı, ikişer düzeyde olmak üzere irdelenmiştir. Buna göre, gecikme ve NŞD arasındaki ödünleşim, kaynakların sıkı

olduğu durumda daha önem kazanmaktadır. En düşük gecikme değeri, daima en fazla NŞD çözümünün gecikme değerinden daha düşüktür.

Tsai ve Gemmill (1998), çalışmalarında kaynak kısıtlı rastsal hale getirilmiş faaliyet süreleri olan proje çizelgeleme problemlerinde, iyi çözümler bulmak için “Tabu Arama” yöntemini sunmuşlardır. TA çoklu tabu listelerini kullanmışlar, böylece rastsal hale getirilmiş kısa zamanlı hafıza ve çoklu başlangıç çizelgeleri ile, aramanın daha da genişlemesini sağlamışlardır. Sundukları yöntemin hem deterministik hem de stokastik problemlerde iyi çözüm bulmak için verimli bir yol olduğunu göstermişlerdir. Araştırdıkları örnek projelere göre, deterministik problemlerin çoğunluğunda optimal çizelgeler bulmuşlardır.

Brucker vd. (1999), literatürde “Kesen Düzlem Yöntemi”, “Dal-Sınır Yöntemi” gibi temel yöntemlerin KKPC problemlerinin çözümünde genel olarak yetersiz kaldığını belirtmiştir. Bu nedenle, bu yöntemlerin esaslarına bağlı kalınarak ve incelenen problemin özgün yapısı da dikkate alınarak geliştirilen özel sayılama yöntemleriyle çözüm etkinliğinin artırılmasına çalışılmıştır.

Khamooshi (1999), farklı bir yaklaşım olarak TM-KKPC problemlerinin çözümünün “dinamik programlama” ile sağlanabileceğini göstermiştir. Ancak, özellikle çok kaynak, çok proje özelliklerini taşıyan problemlerde, ardışık karar aşamalarında oluşturulması gereken karar seçeneklerinin sayısı son derecede arttığından, dinamik programlamanın uygulaması büyük bir önem taşımamakta, sayılama yöntemlerine üstünlük sağlayamamaktadır.

Nabrzyski ve Weglarz (1999), uzman sistem ve çok amaçlı tabuyu, arama yöntemi kullanarak, tanımlamış ve çok amaçlı bir proje çizelgeleme yöntemi geliştirmişlerdir. Bu çalışmada karar verici, amaçlarını bir amaçlar listesi içinden belirlemektedir. Bu listede yer alan amaçlar şunlardır: Proje süresi, kaynak profilinin düzgünlüğü, ortalama ağırlıklı terminden sapma, ortalama ağırlıklı akış süresi, toplam kaynak kullanımı, ağırlıklı kaynak kullanımı ve NŞD.

Seda (2002) çalışmasında; KKPC problemine uygulama açısından yaklaşarak, hangi faaliyetlerin hemen yapılması gerektiği ve hangilerinin ertelenmesi gerektiğine karar verecek uygun bir sezgisel yöntem seçmiştir. Faaliyetleri değiştirmek yerine, sürelerini uzatan verimli bir yöntemle faaliyetlerin sürelerini aktif olan ve olmayan olarak alt aralıklara ayırmak mümkün olmuştur. Daha sonra da klasik kritik yol

metodu, uygulanabilir hale gelen problemi kolayca çözmüştür. Geliştirdiği algoritmanın kolaylıkla kaynak kısıtlı çoklu proje çizelgelemeye uyarlanabildiğini ifade etmiştir.

Whitehouse ve Depuy (2000) de “COMSOAL Yöntemi”ni, birkaç kaynak tahsis sezgisel yaklaşımıyla karşılaştırmışlardır ve COMSOAL Yönteminin, geniş örnek büyüklükleri için, diğer geleneksel kaynak tahsis sezgisel yaklaşımlarından daha iyi sonuçlar verdiğini bulmuşlardır.

Drexl vd. (2000) Demeulemeester’in dal-sınır algoritmasından elde edilen çözümlerin kalitesine problem karakteristiklerinin etkisini göstermek amacıyla çalışma yapmışlardır. KKPÇ problemlerinin üretilmesi için “PROGEN” denilen bir algoritma tanımlamışlardır. Bu çalışmalarda, literatürde yaygın olarak kullanılan klasik kıyaslama problemlerinin, bütün muhtemel problemlerin küçük bir alt kümesine ait olduğu ve bu tür küçük hacimli problemlerin mantıklı hesaplamalı çaba ile optimal bir çözümünün olmayacağı sonucuna varılmıştır.

Kolisch ve Hartmann (2000), KKPÇP için yapılan araştırma sonuçlarına göre sezgisel bir çalışma yayınlamışlardır. Bu çalışmada; meta sezgisellerin farklı tipleri (tavlama benzetimi, genetik algoritmalar ve tabu aramalar) üzerine karşılaştırma sonuçları vermişler ve bu karşılaştırma sonucunda, meta sezgisel yaklaşımların diğerlerinden daha iyi sonuçlar verdiğini belirtmişlerdir.

Lova ve Tormos (2000), her projenin kendi başına ele alındığı modelleme durumu için, iki aşamalı bir sezgisel algoritma önermişlerdir. Atanacak faaliyetin saptanması için, ilk aşamada, toplam iş kapsamı (TWK) sezgisel kuralı kullanılarak proje seçilmiş; ikinci aşamada ise, GRD en erken başlama zamanı (EST), ilk giren ilk çıkar (FCFS) ve LST sezgisel karar kurallarından birisi kullanılarak atanacak faaliyet belirlenmiştir. Yapılan sayısal karşılaştırma çalışmalarında; iki aşamalı algoritma, “MAXTWK” sezgisel kuralını kullanan tek aşamalı sezgisel algoritma ile karşılaştırılmış ve iki aşamalı algoritmanın genelde daha iyi sonuçlar verdiği rapor edilmiştir. Yukarıda sayılan sezgisel kurallar arasında, ikinci aşamada LST’nin en iyi sonucu verdiği belirlenmiştir. Sezgisel kurallardan hiçbirisinin diğerlerinden daha çok öne çıkması sonucu, Lova ve Tormos, TWK-LST, TWK-FCFS ve TWK-EST kombinasyonlarını uygulayarak elde ettikleri en iyi çözümün nihai çözüm olduğu bir uygulama önermişlerdir. Tek proje çizelgeleme ortamındaki birden fazla geçiş

uygulamaya paralel bir uygulama olan bu uygulamanın, hesaplama süresi kullanılarak daha iyi sonuçlar verdiğini göstermişlerdir. Bahsedilen hesap süreleri, saniye mertebesinde CPU-süreleri olduğundan, “BEST-MP” olarak nitelendirilen bu algoritmanın, pratikteki uygulamalarda tercih edilmesinin gereği açıktır. Lova ve Tormos, sezgisel kurallarla birlikte seri ve paralel çizelgeleme yöntemlerinin katkısını da araştırmışlar ve paralel çizelgelemenin daha iyi sonuçlar verdiğini saptamışlardır.

Atlı (2005), var olan sezgisel algoritmalar üzerine çözüm niteliğini arttırmak için, “Tabu Araması”nı, KKPC problemine uygulamıştır. “Tabu Araması”, optimizasyon problemlerini çözmek için sezgisel bir süreçtir. Yerel optimum çözümde takılıp kalma olasılığını azaltan bir yöntem sağlamaktadır. Tabu Araması, bir çözüm durumunu, başka bir çözüm durumuna dönüştüren bir dizi yer değişimleri kullanmakta ve bu yer değişimlerinin çekiciliğini ölçmek için değerlendirme fonksiyonları sağlamaktadır. TA sürecinin yapısı epeyce esnektir ve sık sık farklı problemler için yeni yer değişimleri ve değerlendirme ölçütleri elde etmek için kullanılmaktadır. Bu çalışmada (Patterson 1984), arşivlenmiş olan ve çoklu kaynak problemlerinin temel türlerini temsil eden, Patterson’ın 110 adet örnek problemi test amaçlı olarak kullanmıştır ve test sonuçları OPL-CPLEX kesin çözüm veren (exact) yöntemiyle tekrar çözülerek karşılaştırma yapılmış ve problemin NP-tamam (NP-complete) olduğu bulunmuştur.

KKPC literatüründe genellikle uygulanan karşılaştırma kriterleri şunlardır; En iyi çözümünden ortalama sapma (lateness) ve bunun standart sapması; sezgisel algoritmanın en iyi çözümü kaç problemde bulduğu ve en iyi çözümün bilinmediği enazlama problemlerinde altsınırdan – ençoklama problemlerinde üstsınırdan - olmak üzere ortalama sapma ve bunun standart sapması. Bunlar, çözümün kalitesini değerlendirmeye yönelik kriterlerdir. Diğer bir kriter ise, elde edilen çözüm için harcanan çabadır. Bu çaba, genellikle CPU-saniye olarak ifade edilmektedir. Ancak raporlamanın değişik bilgisayarlara-işlemcilere dayandırılması karşılaştırmayı olanaksız kılabilir. Bu nedenle; giderek daha yaygınlaşan bir uygulama, algoritmaların yarattığı çizelgelerin, (örneğin ilk 1000 veya 5000 çizelgenin) çözüm kalitesini karşılaştırmaktır. Yukarıda da belirtildiği gibi, eniyi çözümün bilinmemesi durumunda, önerilen algoritmanın çözümü ile problemin altsınırı (üstsınırı) karşılaştırılmaktadır. Karşılaştırmının başarılı olması için, alt/üstsınırın eniyi çözüme

mümkün olduğunca yakın olması, diğer bir deyişle etkin olması istenmektedir. Dalsınır yöntemlerinin uygulanmasında, kullanılan alt/üstsınırların etkinliği, yöntemin verimliliğini önemli ölçüde etkilemektedir. Daha etkin alt/üstsınırların hesaplanması, genellikle daha çok hesaplama çabası istemektedir. Bu nedenle, daha az hesaplama ile elde edilebilecek daha etkin alt/üstsınır önerilerinin geliştirilmesi önemli bir araştırma alanı teşkil etmektedir.

3.2 Bulanık KYM'na Yönelik Yapılan Çalışmalar

Projelerde faaliyet sürelerinin kesin olarak bilindiği durumda, KYM'nin; mühendislik ve yönetim uygulamalarının geniş bir sahasında ve karmaşık projelerin planlama ve kontrolünde çok yararlı bir araç olduğu gösterilmektedir. Ancak uygulamada pek çok durumda, faaliyet sürelerinin belirlenmesi öznel olarak tahmin edilmek zorundadır. Bir proje şebekesini, bulanık küme kullanarak, modelleyen az sayıda araştırmacı hakkında aşağıda bilgi verilecek ve yapmış oldukları çalışmalardan bazılarını kısaca değinilecektir. Görüldüğü üzere, proje yönetiminde, bulanık programlama konusuyla ilgili olarak, geniş bir alanda, fazlaca değişik endüstri mühendisliği problemi mevcuttur. Bu problemler arasında en ilgi çekici ve gelişmeye açık konulardan birincisi bulanık KYM/PERT konusudur. Proje yönetimi dalında çok önemli bir yere sahip olan bu metodların, bulanık yöntemlerle incelenmesi sonucunda projelerde zaman yönetimi problemlerinin büyük ölçüde çözüleceği kanaatindeyiz. Proje faaliyetlerinin modele dahil edilmesi sırasında kullanılan bulanık KYM; çalışmanın önemli bir temelini teşkil etmektedir. Bu çalışma, daha sonra literatürde bu konuda yapılacak çalışmalara ışık tutacak niteliktedir. Yine bu çalışma ile, projelerde, faaliyet sürelerinin deterministik veya probabilistik şekilde belirlenmesi yerine, bulanık üyelik fonksiyonlarının kullanılmasının daha pratik ve anlamlı olduğu gösterilmektedir.

Bulanık sayıların sıralanması konusunda literatürde çok sayıda yöntem mevcuttur. Fortemps ve Roubens (1996) tarafından yapılan çalışmada, bu yöntemlerin çoğunun insan yaklaşımını doğru şekilde ifade etmediği, bu konuda yetersiz kaldığı gösterilmektedir. “Alan Karşılığı Yöntemi”nin doğrusallık ve toplanabilirlik özelliklerine sahip olması ve insan yaklaşımını doğru şekilde yansıtabilmesi de, üzerinde durulan bir diğer noktadır. Bütün bu sebepler dikkate alınarak, bulanık sayıların sıralanmasında Alan Karşılığı Yönteminin kullanılması uygun bulunmuştur.

Dubais ve Prade (1979), Chanas ve Kamburowski (1981), Chanas (1982), Buckley (1989), Mares (1989), Gazdik (1983), McCahon ve Lee (1988), Tsujimura vd. (1993), Chanas ve Zelinski (2001), Slyeptsov ve Tyshchuk (2003), Dubois ve ark. (1988), Chanas ve ark. (1988), Chanas ve Zelinski (2001), Kuchta, vd gibi çeşitli araştırmacılar bulanıklık kavramıyla ilgilenmişler ve çeşitli analiz yaklaşımı geliştirmişlerdir. Bu yaklaşımlardan çoğu ileri ve geriye doğru hesaplama için KYM formülasyonunu temel alan yaklaşımlardır. Aşağıda, literatürde yer alan bu çalışmalar, kısaca özetlenecektir. Ancak yapılan çalışmalar incelendiğinde “Bulanık KYM Yöntemleri”nin çözümü için genel olarak bulanık “doğrusal programlama çözümü”, klasik KYM yaklaşımını temel alan “bulanık KYM metodolojisi” ve “simpleks çözüm metodolojisinin” var olduğu gözlemlenmektedir. KKPC problemlerinin temel yapı taşı oluşturulan KYM yöntemi; “B-ÇM-KKPC Yöntemi”nin ilk basamağını oluşturmakta ve projelerin bulanık yapısı, projenin ilk aşamada oluşturulabilmesine imkan tanınması bakımından önem taşımaktadır. Bu nedenle, çalışmada ilk olarak bulanık KYM yöntemi üzerinde durulmuş, bir sonraki bölümde ise, bulanık KYM uygulaması ve hesaplama yöntemi açıklanmıştır.

Chanas ve Zelinski (2001), genel bir şebekede en geç başlangıç zamanının belirlenen aralıklar için polinom algoritmalar geliştirmiştir. Faaliyet bolluklarının ve en geç başlangıç zamanının olası değerleri kümesinin hesaplanmasında, geriye doğru hesaplama yaklaşımının hatalı olduğunu belirtmiştir. Ayrıca aynı yollar için bile, bulanık kritik yolun farklı tanımlamalarının, kritiklik derecesinin tahmin edilmesini zorlaştırdığını da eklemiştir. Bulanık kritiklik ve kritiklik aralığı notasyonları arasındaki bazı ilişkilerin tartışıldığı bir proje şebekesinde, kritiklik notasyonunun doğal bir genellemesini sunmuşlardır ve iki yöntemi, kritikliğin patika derecesinin hesaplanmasında uygulamışlardır.

Chanas ve Kamburowski (1981, 1982), geliştirdikleri metot KYM/PERT yöntemleriyle paralellik arz etmektedir. Fakat, faaliyet süreleri zaman uzayında bulanık kümelerden oluşturmuşlardır ve bu metodu FPERT olarak isimlendirmişlerdir. Amaç; bulanık proje tamamlanma süresinin belirlenmesidir. Bu nedenle, geriye doğru hesaplama ve kritik yol belirleme hesaplamalarını dikkate almamışlar, çalışmalarında bu konuya değinmemişlerdir. Bir “i” olayının, en erken başlama zamanının hesaplanmasında “genişletilmiş maksimum” kullanmışlardır.

Chanas (2001), son faaliyetin en erken başlanma ve en geç tamamlanma seçiminin, geriye doğru bulanık hesaplama icra edemediği tespitinde bulunmuştur. Chanas; geriye doğru hesaplamayı yürütmek adına, başlangıç hesaplamaları için deterministik veya bulanık hedef zamanını ayrı ayrı sabitlemeyi önermiş ve takip eden bir olayın kritiklik derecesini tanımlamıştır.

Dubois ve ark. (1988), faaliyet bolluğu ve en geç başlangıç zamanının olası değer kümelerini hesaplamak için, sezgisel bir yöntem geliştirmişlerdir. Son olay “e”nin, en geç tamamlanma süresi için bulanık bir değer sabitleyerek bunu başarmış olsalar da; geriye doğru hesaplamalarda en son müsaade edilebilir başlangıç zamanını hesaplamışlardır. Tam ve doğru olmayan (belirsiz) bir ortamda, “e”nin en geç tamamlanma süresinden farklı bir değer tanımlanmasının gerekliliğini kanıtlamışlardır. Çünkü, belirsizlikler yığılmak zorundadır ve $T_{Le} = T_{Ae}$ eşitliğiyle başladığında T_{Li} hesaplama esnasında iki kez sayılma riski vardır.

Buckley (1987); “(i,j)” faaliyetinin bolluk, en erken başlama zamanı ve en geç başlama zamanını ve “i” olayı için olasılık dağılımının tanımını vermiştir. Üçgensel olasılık dağılımına ve faaliyet sürelerine sahip proje süresi için olasılık dağılımlarını hesaplayan bir algoritma önermiştir. Bu hesaplama, bulanık zamanın α seviyeli bir kümesinde, KYM’nun sıralandırılmasında kullanılmaktadır. Bu girdiler, belirli bir α seviyesi kümesinde faaliyet sürelerinin alt sınırıdır. Bu, proje süresinin alt sınırını vermektedir. Daha sonra bu hesaplama, tüm üst sınır için tekrarlanmaktadır. Bulunan sonuç proje süresinin üst sınırı olmaktadır.

Mares (1989)’in düşüncesine bakılırsa, şebekenin tüm bileşenlerinin eş zamanlı olarak bulanık belirsiz olduğu görüşünden bahsedilebilir. Mares’in çalışması yalnızca faaliyet sürelerinin değil, aynı zamanda şebeke yapısının bulanıklaştırılmasını da önermektedir. Mares, “düğüm nesne şebekesi” kullanmaktadır. Problemlerde AON gösterimini kullanır.

Gazdik (1983) ise, şebekenin çeşitli şekillerde bulanıklaştırılmasıyla ilgilendiği söylenebilir. O, geriye doğru hesaplama yapmamış; olası tüm patikaların sayılama metoduyla bulunması için bulanık kritik bir yol önermiştir.

McCahon ve Lee (1988), ileri ve geriye doğru hesaplamalarını icra etmiş; genişletilmiş max. ve min kullanımı yerine, karşılaştırmalı temsili bir metodun uygulanmasını önermiştir. Chanas’ın tersine, McCahon ve Lee; geriye doğru

hesaplama için, başlangıç noktası olarak ileriye doğru hesaplamalarda son faaliyetin en erken tamamlanma süresini kullanmayı tavsiye etmişlerdir. Sebebinin belirtmemiş olsalar da, bir faaliyetin kritiklik derecesi olarak Chanas tarafından önerilen yöntemi vurgulamışlardır.

Tsujimura vd. (1993); birkaç uzman tarafından verildiği varsayılan bir şebekede, üçgensel bulanık sayıların kullanıldığı durumu ele almışlardır. Böylece bu üçgensel bulanık sayılar, birbirinden oldukça farklıdır. Her bir faaliyet süresi için major ve minör üçgensel bulanık sayılar olarak isimlendirdikleri, bazı kriterlere göre belirlediklerini kıyaslamışlardır. Faaliyetlerin süreleri hakkındaki uzman görüşünü, üçgensel bulanık sayıları bir tabloya koyarak ifade etmişlerdir. Kritikliği yüksek dereceli patikalar için hesaplama tekrarlanmakta; fakat proje tamamlanma zamanı ve kritiklik derecesini belirlemek için birleştirme(karma) metodu kullanılmaktadır.

Nasution (1994); bir olayın bolluğu ve en geç müsaade edilebilir başlama zamanı için operasyonel formüller sağlamıştır. Bundan başka faaliyetlerin bolluklarının hesaplanmasını da düşünmüştür. Bu çalışmada, tanımlanan P ve Q tabloları kullanılarak ve KYM sırasına paralel olarak, bir olayın bolluğu ve en geç müsaade edilen başlangıç hesaplama işlenecektir. Bir olayın bolluğu ve en geç müsaade edilebilir başlama zamanı için operasyonel formüller sağlamıştır.

Sharafi vd. (2008), çalışmalarında bulanık ortamda, bulanık proje çizelgelemek için bulanık teori tabanlı yeni bir yöntem sunmuşlardır. Literatürde ilk defa faaliyetler arasındaki ilişkileri, kesinleştirilmiş sayı değil de üçgensel bulanık sayı olarak ele almışlardır. Sundukları yöntem ise, doğrusal programlama tabanlıdır. Çalışmalarında bulanık ortamda, bulanık proje çizelgelemek için bulanık teori tabanlı yeni bir yöntem sunmuşlardır. Faaliyet zamanları üçgensel bulanık sayılardır ve literatürde ilk defa faaliyetler arasındaki ilişkileri kesinleştirilmiş sayı değil de üçgensel bulanık sayı olarak almışlardır. Sunulan yöntem doğrusal programlama tabanlıdır.

Chen ve Huang (2008) çalışmalarında; bulanık faaliyet zamanlarına sahip bir proje şebekesinde kritik yolun belirlenmesi için analitik bir yöntem sunmuşlardır. Proje şebekesindeki tüm faaliyetlerin işlem zamanlarında, üçgensel bulanık sayıları kullanmışlardır. “PERT” tekniği ve “Bulanık Küme Teorisi”ni birleştiren yeni bir modelle, faaliyetlerin ve yolların kritiklik derecesini belirlemişlerdir. Sundukları yöntemde bir olasılık indeksi tanımlayarak, projenin belirli bir zamanda tamamlanma

ihtimalini belirlemişlerdir. Geliştirdikleri yöntemi bir örnekte kullanarak, sonuçlarını da diğer yöntemlerle kıyaslamışlardır. Bu sonuçlar, sundukları yöntemin, faaliyetlerin kritikliğinin belirlenmesinde ve kritik yolun bulunmasında daha etkili olduğunu göstermiştir.

3.3 Bulanık KKPC Yöntemlerine Yönelik Yapılan Çalışmalar

Proje yönetiminde, geleneksel yöntemlerle çözümü çok zor olan, projenin karmaşıklığı ve projede arıza veya bekleme oluşması gibi durumlar nedeniyle, şebekedeki optimum ve altoptimum yolların başarıyla tahmin edilmesi çok önemlidir. Metasezgisel tabanlı optimizasyon algoritmaları, gerçek hayattaki uygulamalara yeni bir bakış açısı kazandırmıştır. Son yıllarda bulanık proje çizelgeleme problemlerinin metasezgisel yöntemlerle çözümü konusunda yapılan çalışmalar aşağıda verilmiştir.

Yousefli vd. (2008) çalışmalarında; KKPC problemlerinin bulanık ortamda çözümü için yeni bir yöntem sunmuşlardır. Sunulan modelde, her faaliyetin süresi, kaynağın kullanılabilirliği ve faaliyetlerin kaynak ihtiyaçları belirsizdir. Bu nedenle ilk etapta, üyelik dereceli bulanık sayılar kullanılarak öncelik listesi oluşturulmuştur. Proje çizelgeleme sonuçlarını göstermek için üç boyutlu Gantt şeması kullanılmıştır. Geliştirilen yöntem, sayısal bir örnek üzerinde gösterilmiştir.

Soltani ve Haji (2007) çalışmalarında; bulanık ortamda proje çizelgeleme problemlerinin çözümünde bulanık teori tabanlı yeni bir yöntem sunmuşlardır ve buna göre faaliyet sürelerini ikizkenar yamuk şeklindeki bulanık sayılar olarak kabul etmişlerdir. Doğrusal programlama tabanlı bu yeni yaklaşımla geriye doğru hesaplamalarda üretilen negatif ve uygun olmayan sonuçları elemişler ve bu yöntemle “değiştirilmiş geriye doğru hesaplama” adını vermişlerdir. Kullanıcıları, optimal sonucu basit bir tekrarlama ilişkisiyle hiçbir doğrusal programlama problemi çözmeden, kolayca çözüme götüren doğrusal programlama probleminin optimal sonucunun genel formunu kullanmışlardır. Sayısal bir örnekle de yöntemlerini açıklamışlardır. Buldukları sonuca göre yöntemin en büyük avantajı; doğrudan, bulanık aritmetik operatörler kullanarak, anlamlı ve hesaplanabilir sonuçların elde edilmesi olmuştur.

Liu (2003) çalışmalarında; faaliyet zamanlarını rastsal bulanık değişkenler olarak kabul edip, proje çizelgeleme problemlerini rastsalılık ve bulanıklıkla karışık bir belirsizlik içinde ele almışlardır. Farklı yönetim gereksinimlerini karşılamak için üç farklı rastsal bulanık model sunmuşlardır: beklenen maliyet minimizasyon modeli, (α, β) , maliyet minimizasyon modeli ve şans maksimizasyon modeli. Hibrit akıllı algoritmanın tasarımında bazı belirsiz fonksiyonlar için rastsal bulanık simulasyonlar yapmış ve bu fonksiyonları genetik algoritmaya eklemişlerdir. Sayısal örneklerle algoritmanın etkinliği göstermişlerdir. Oluşturulan modelin uygun şekilde çözümlenmesi için kullanılacak yöntemlerle ilgili olarak Liu (2003) tarafından yazılan “Fuzzy Activity Times in Critical Path and Project Crashing Problems” adlı çalışmada klasik PERT tekniğinin bulanık PERT tekniğine dönüşmesi ve oluşturulan bulanık modelin çözümünde “Yager İndisi” kullanılarak kesinleştirme yapılması üzerinde durulmaktadır. Bu yöntemde, konveks bulanık sayıların alan karşılığı yöntemi kullanılarak Yager (1981) tarafından önerilen yönteme göre sıralanması ve sıralama yapılarak kesin hale getirilmiş olan modelin klasik yöntemlerle çözülmesi önerilmektedir. Bu yaklaşım özellikle model hacminin büyük olduğu durumlarda bulanık operatörlerin kullanılmasına gerek kalmadan modelin çözülmesi için iyi bir yöntem olarak göze çarpmaktadır. Son yıllarda yapılan çalışmalar incelendiğinde, literatürde kullanılan yöntemler arasında benzetilmiş tavlama algoritmaları (Hapke vd. 2000), genetik algoritmalar (Zheng ve Ng 2005), hibrid genetik algoritmalar (Kim ve ark. 2005) ve TA algoritmaları (Mika vd. 2008) yer almaktadır.

Liu vd. (2008), bulanık KKPC problemlerinin çözümü için, bulanık küme teorisi tabanlı, özel bir genetik yerel arama algoritması tasarlamışlardır. Öncelikleri uygun olan bir faaliyet listesini çözüm örneği olarak uygulamışlar ve özel olarak tasarladıkları rekombinasyon operatörlerini ve yerel arama işlemlerini kullanmışlardır. Gelecek nesilleri oluştururken de rulet çemberi ve elit seçilimi modellerini birleştirmişlerdir. Değişik kaynak bulunabilirlik düzeyleri ile proje çizelgeleme deneyleri yapmışlar ve buldukları sonuçlara göre, bu tür problemlerin çözümünde geliştirdikleri algoritma etkili bir yöntem sunmuştur.

Wang ve Huang (2010) çalışmalarında; kaynak kısıtlı yazılım geliştirme projesinin çizelgelenmesini, bulanık programlama modelleriyle göstermişlerdir. Karar verme sürecinin farklı gereksinimleri için, beklenen maliyet modeli ve güvenilirlik maksimizasyonu modeli olmak üzere iki yeni model önermişlerdir. Modellerin

çözümü için genetik algoritma ve bulanık simülasyonu birleştirerek hibrit bir akıllı algoritma tasarlamışlardır. Sayısal örneklerle geliştirdikleri yöntemin etkinliğini göstermişlerdir.

Pan ve Yeh; bulanık KKPC yaklaşık optimal çözüm elde etmek için, bulanık genetik algoritmayı tabu mekanizmasıyla birleştirmişlerdir. Geliştirdikleri bu yöntemin genelde birçok gerçek KKPC’de birlikte bulunan, bulanık ve kesinleştirilmiş sayıları kullanabildiğini ileri sürmüşlerdir. Gerçek bir kontrol çizelgesini, bulanık KKPC problemi olarak ele alıp, gerçek bir örnek olayda algoritmayı test etmişlerdir.

Wang (2002) çalışmasında ürün geliştirme projelerinin yönetiminde karşılaşılan faaliyet sürelerindeki belirsizliklerinin çözümü için, bulanık çizelgeleme yöntemi geliştirmiştir. Belirsiz ve esnek olan geçici bilginin modellenmesinde olasılık teorisini kullanmıştır. Çizelge riski kavramını, çizelgenin performansının değerlendirilmesinde kullanmıştır. Minimum çizelge riskinin belirlenmesi için bulanık ışıın arama algoritmasını geliştirmiştir ve faaliyetlerin başlangıç zamanları olarak, bütün geçici kısıtların değerini en düşük, memnuniyet değerini en üst seviyeye çıkartacak şekilde seçmiştir. Çalışmasında ayrıca çizelge risklerinin özelliklerini de ele almıştır. Sunduğu yöntemin proje yöneticilerinin, belirsiz çizelgeleme ortamında, en düşük gecikme olasılığıyla çizelge seçimi yapmasına katkı sağladığını örnek bir elektronik ürün geliştirme projesinde göstermiştir.

3.4 ÇM-KKPC ve B-ÇM-KKPC Yöntemlerine Yönelik Yapılan Çalışmalar

Sprecher vd. (1997); Demeulemeester ve Herroelen (1992) tarafından tek modlu problemler için önerilen “dal-sınır” yöntemini çok modlu problemlere uyarlamıştır. Benzer şekilde, Sprecher ve Drexl (1998), önerdikleri yeni baskınlık kurallarını kullanarak “dal-sınır” yöntemi ile en fazla 20 faaliyet içeren problemleri çözebilmişlerdir. Hartmann ve Drexl (1998), Stinson vd.’nin (1978) tek modlu problemler için önerdiği yöntemi, çok modlu problemlere uyarlamışlardır. Sprecher (1994); Demeulemeester ve Herroelen (1992) ve Stinson vd.’nin (1978) dal-sınır yöntemlerinin ayrıntılı bir karşılaştırmasını yapmıştır. Hartmann ve Drexl, yaptıkları sayısal değerlendirmeler sonucu, Sprecher ve Drexl algoritmasının halen bu konudaki en etkin kesin çözüm yöntemi olduğunu rapor etmişlerdir.

Slowinski (1981, 1989); süre ve maliyet bazlı amaçları göz önüne alan ve yenilenemez kaynakları da içeren çok amaçlı kaynak KKPÇ üzerinde çalışmıştır. Faaliyetler çok modlu olarak tanımlanmıştır. Faaliyetlerin gerçekleştirilmesine ara verilebileceği (preemption) varsayılmıştır. Ancak ikincil bir amaç olarak da, bu ara verme sayısının enazlanması kabul edilmiştir. Slowinski'nin birinci algoritmasında (Slowinski,1981), öncülleri tamamlanan tüm faaliyetlerin oluşturduğu ve kaynak kısıtlarını ihlal etmeyen kümeler saptanır. Bu kümeler arasından atanacak kümenin seçimi çok amaçlı doğrusal programlama kullanılarak eniyileme yolu ile belirlenir. İkinci algoritmada (Slowinski, 1989) ise, küme seçimi karar verici ile etkileşimli olarak gerçekleştirilir.

Bouleimen ve Lecocq (1998) önerdikleri tek modlu problemler için tanıtılan “Tavlama Benzetimi Yaklaşımı”nı çok modlu probleme de uyarlamışlardır. Modlar faaliyetlerden ayrı bir liste olarak tutulmaktadır. Faaliyet listesindeki yer değişimi ile yeni bir faaliyet listesinin türetilmesinden sonra rasgele bir faaliyet için mod değişikliği uygulanmıştır. Rasgele seçilen faaliyet için rasgele bir mod seçilmiş ve olurluluğu araştırılmıştır. Olurlu ise; yeni bir komşu çözüm elde edilmiş ve amaç işlevi değeri hesaplanarak tavlama benzetimi uygulamasına devam edilmiştir. Olurlu değilse; mod değişikliği için rasgele yeni bir faaliyet seçilmiştir. Bu uygulama olurlu bir çözüme ulaşana veya belirli bir sayıda uygulama yapılana kadar devam edilmiştir. Son durumda, faaliyetlerin yer değiştirme aşamasına geri dönmüş ve yeni bir faaliyet listesi türetilmiştir. Faaliyet listesinin çizelgeye dönüştürülmesinde ise seri çizelgeleme uygulanmıştır.

Özdamar ve Ulusoy (1994), LCBA algoritmasının çok modlu durum için uyarlamasını sunmuşlardır. MSLK, LFT ve WRUP sezgisel kuralları ile 95 problem üzerinde karşılaştırılan LCBA'nın çok daha üstün bir performans sergilediği gösterilmiştir. Sınama problemlerinin faaliyet adetleri, 20-57; kaynak adetleri, 1-6 aralığında seçilmiş ve her kaynak için mod adedi, en fazla 3 olarak kabul edilmiştir. LCBA algoritmasının iteratif uygulaması, Ulusoy ve Özdamar (1995), Özdamar ve Ulusoy (1996) tarafından ayrıntılı olarak irdelenmiştir.

Özdamar (1999), yenilenemeyen kaynakların da yer aldığı probleme özel bilgilerin de yer aldığı bir melez GA yaklaşımı ile çözüm aramıştır. Bu yaklaşımda, dolaylı bir kodlama kullanarak, ilki faaliyetler için atanmış moda ait, diğeri bir grup öncelik kuralı tarafından belirlenen ve bir kural numarası olan iki boyutlu bir kromozom

yapısı tanımlamıştır. Çalışmada, bu yaklaşımla, uygun bir CPU süresi içinde, eniyi yaklaşık sonuçlara ulaşıldığı gösterilmiştir.

Mori ve Tseng (1997); sadece yenilenebilir kaynakların bulunduğu ÇM-KKPÇP'yi ele almışlardır. Çalışmada, kromozomun her geninde bir faaliyet numarası, bu faaliyet için atanmış mod, çizelgeleme sırası ve başlama-bitiş zamanları bilgilerini taşıyan bir kodlama uygulanmıştır. Sundukları bu GA yaklaşımını Drexl ve Grunewald'ın (1993) sunduğu stokastik bir yaklaşımla karşılaştırarak, GA'nın daha iyi sonuçlar verdiğini ifade etmişlerdir.

Hartmann (1997) ise; hem yenilenebilir hem de yenilenemeyen kaynakların yer aldığı bir GA yaklaşımı sunmuştur. Bu çalışmada kromozom, öncüllük ilişkileri bakımından olurlu bir faaliyet sıralaması ve bu faaliyetler için seçilmiş birer modun bulunduğu bir kodlama ile sunulmaktadır. Bu yaklaşım için farklı bir dizi deneysel çalışma yapılarak en iyi olan seçilmiş ve bu GA yaklaşımı literatürde raporlanmış bulunan üç sezgisel yöntemle karşılaştırılmıştır. Sunulan yeni yaklaşımın, diğer üç sezgisel yöntemden eniyi proje süresinden ortalama sapma kriterine göre daha iyi sonuçlar verdiği gösterilmiştir.

Hapke vd. (1996, 1997); kaynak kısıtı olan, çok modlu proje çizelgeleme problemlerinde faaliyetlerin zamanlarını, bulanık zaman parametreleri olarak, bulanık sayılar için güçlü-zayıf karşılaştırması yapan bir çizelgeleme yöntemi kullanmışlardır. Problem çok amaçlı olduğu için, Pareto kümesinin tahmini için "Pareto Tavlama Benzetimi"ni kullanmışlardır. Işık hüzmesi arama yönteminin kesikli sürümü, oluşturulan çözümlerin etkileşimli analizine ve karar vericinin uzlaşılan en iyi çizelgeyi seçmesine yardımcı olmuştur.

Hapke ve Slowinski (2000) KKPÇ problemlerinde kullanılan mevcut öncelikli sezgisel yöntemler için bir genelleme sunmuşlardır. Genellemede zaman parametrelerini kesin almak yerine, bulanık olarak ele almışlardır. Öncelik listeleri oluşturmak için, bulanık sıralama yöntemini ileri sürmüşlerdir. Yöntemin performansını bir örnek üzerinde göstermişlerdir.

Literatürde yer alan ikinci alan, B-KKPÇ problemi. Bu alan, 1994 yılında Hapke ve arkadaşları tarafından incelenmeye başlanmıştır. Problemin çözümü için literatürde çeşitli yöntemler kullanılmaktadır. Bunlardan birincisi; öncelik kurallı seri ve paralel çizelgeleme yöntemidir. Çok amaçlı B-KKPÇ probleminin çözümü için

ise, daha önce çok amaçlı kesin modellerin çözülmesinde kullanılan “Pareto Benzetilmiş Tavlama” yönteminin kullanılması önerilmektedir. Toplam beklenen proje süresinin en aza indirilmesi için ise, doğrusal olmayan ikili tamsayı programlama yöntemiyle model oluşturulması ve oluşturulan modelin çeşitli sezgisel yöntemlerle çözülmesi önerilmektedir. Bulanık ÇM-KKPÇ Problemi’nin çözümünde kullanılan diğer yöntemler ise, “Işın Arama Algoritması” ve “Genetik Algoritma”dır.

Drexl vd. (2000) işlemlerin kesintili yapılabileceği varsayımıyla, ÇM-KKPÇ modellerinin genel matematik formülasyonlarındaki kaynakları, kısmen yenilenebilen kaynaklar olarak tanımlamışlardır. Çözümün genişletilmiş simpleks Yöntemi adı verilen seçeneklerle gerçekleştirilebileceği esasına dayanan çok amaçlı doğrusal modeli geliştirmiş ve çözüm yöntemleri önermişlerdir. Bu yaklaşımda, işlem bölümlerinin sayısı kontrol edilememektedir. Elde edilen çözüm üzerinde, bölünme sayısını azaltan ve sınırlayan bir yöntem uygulanmaktadır.

4. BULANIK KÜME TEORİSİ

4.1 Bulanık Mantık

Bulanık mantık ve bulanık mantık tabanlı uygulamalar, son yıllarda hem üniversite çevrelerinde hem de üretici firmalar arasında ilgiyle izlenen bir konu haline gelmiştir. Bu bölümde bulanık mantık konusu basit ve anlaşılır bir üslupla ele alınıp açıklanmaya çalışılmıştır. Tez çalışmamın ileriki bölümlerinde kullanılacak olan bulanık küme teorisine ait temel kavramlar ve yöntemler de yine bu bölümde açıklanacaktır.

Bulanık mantık, düzgünlüğün olmadığı, belirsizliğin, eksik ve çelişen bilgilerin olduğu, gerçeğin ve olasılığın bir parçasının bulunduğu bir çevrede kısaca mükemmel bilginin olmadığı bir çevrede muhakeme etme ve mantıklı karar verme yeteneğidir. İkinci bir tanımı ise; geniş bir alandaki fiziksel ve zihinsel görevleri hiçbir ölçüm ve hesaplama yapmadan gerçekleştirebilme yeteneğidir. Bulanık küme teorisinin kurucusu Lotfi A. Zadeh'e göre bulanık küme teorisi; nesneler sınıfının sınırları kesin olarak tanımlanmadığı zaman ortaya çıkan belirsizlikleri sistematik bir şekilde ele alabilmek için bir kavramlar ve teknikler yapısı geliştirme girişimi olarak tanımlamıştır. Bulanık mantık temel olarak, evet/hayır, doğru/yanlış, siyah/beyaz gibi geleneksel olarak ve kesin şekilde tanımlanan ifadelerle ara değerler atamaya izin veren çok değerli bir mantık sistemidir. Bu sayede biraz sıcak, daha soğuk gibi kavramlar matematiksel olarak oluşturulabilir ve bilgisayarlarca anlaşılabilirler. Bu haliyle bulanık mantık temelde insan düşünüş tarzını örnek alan bir yapıya sahiptir.

Günlük hayatta rastgele kullandığımız bir çok terim genellikle bulanık bir yapıya sahiptir. Bir şeyi tanımlarken, bir olayı açıklarken, komut verirken ve daha bir çok durumda kullandığımız sözel veya sayısal ifadeler bulanıklık içerir. Bu terimlere örnek olarak; yaşlı, genç, uzun, kısa, sıcak, soğuk, ılık, bulutlu, parçalı bulutlu, güneşli, hızlı, yavaş, çok, az, biraz, fazla, çok az, çok fazla gibi daha pek çok sözel terim gösterilebilir. Biz insanlar bir olayı anlatıp, bir durum karşısında karar verirken bu tür kesinlik ifade etmeyen terimler kullanırız. Kişinin yaş durumuna göre ona

yaşlı, orta yaşlı, genç, çok yaşlı ve çok genç deriz. Yolun kayganlık ve rampa durumuna göre arabanın gaz veya fren pedalına biraz daha yavaş veya biraz daha hızlı basarız. Bütün bunlar insan beyninin belirsiz ve kesinlik içermeyen durumlarda nasıl davrandığına ve olayları nasıl değerlendirip, tanımlayıp, komut verdiğine dair birer örnektir.

Endüstriyel bir sürecin denetimi için tasarım yapılırken her şeyden önce o sürecin dinamik bir modeline gereksinim vardır. Ancak pratikte bu her zaman mümkün olmayabilir. Süreç içindeki olaylar matematiksel modellemeye el verecek ölçüde açıkça bilinmeyebilir veya bir model kurulsa bile bu modelin parametreleri zamanla büyük değişiklikler gösterebilir. Bazı durumlarda ise doğru bir model kurulsa bile bunun denetleyici tasarımda kullanılması karmaşık problemlere yol açabilir. Bu gibi sorunlarla karşılaşıldığı zaman genellikle bir uzman kişinin bilgi ve deneyimlerinden yararlanma yoluna gidilmektedir. Uzman karar verici dilsel değişkenler olarak, *uygun, çok uygun değil, yüksek, biraz yüksek, fazla, çok fazla*, gibi günlük yaşamda sıkça kullanılan sözcükler doğrultusunda esnek bir denetim mekanizması geliştirilir. İşte bulanık denetim de bu tür mantıksal ilişkiler üzerine kurulmuştur.

Bulanık mantığın ve bu mantık kurallarını kullanan bulanık küme teorisinin Lotfi A. Zadeh tarafından geliştirilip 1965 tarihli orijinal makalesinde yayınlanmasından sonra, belirsizlik içeren sistemlerin incelenmesi yeni bir boyut kazanmıştır. İlk olarak 1965 yılında dile getirilmesine rağmen bulanık küme kavramı ancak 1970’li yılların ikinci yarısından sonra kullanılmaya başlanmıştır. Bunda özellikle Zadeh’in 1965’deki ilk makalesinden (Zadeh 1965) daha çok bulanık mantığın belirsizlik içeren sistemlere uygulanabilirliğini açıklayan makaleleri (Zadeh 1973, 1975) etkili olmuştur. 1980’li yılların ikinci yarısından sonra ise Japonların ürünlerinde bulanık mantığı kullanmalarıyla hız kazanarak, günümüzde doruk noktasına ulaşmıştır. Artık hemen her alanda bulanık mantık uygulamalarına rastlamak mümkündür. Bu çalışma kapsamında Endüstri mühendisliğinde önemli bir yer tutan proje yönetimi, çizelgeleme ve optimizasyon konularında bulanık küme teorisinin nasıl uygulanacağı gösterilecektir.

Bulanık mantığın uygulama alanlarından bazıları :

- *Otomatik Kontrol Sistemleri* : Robotik, otomasyon, akıllı denetim, izleme sistemleri, ticari elektronik ürünler, vb.

- *Bilgi Sistemleri* : Bilgi depolama ve yeniden çağırma, uzman sistemler, bilgi tabanlı sistemler, vb.
- *Görüntü Tanımlama* : Görüntü işleme, makina görüntülemesi.
- *Optimizasyon* : Fonksiyon optimizasyonu, filtreleme, eğri uydurma, vb.

Karmaşıklık ve belirsizlik içeren büyüklükler, bulanık sayılar olarak da isimlendirilen ve bulanık kümeleri karakterize eden üyelik fonksiyonları ile tanımlanmaktadır. Bulanık sayılar, yine bulanık bir ortamda insanın düşünce ve karar verme mekanizmasına benzer şekilde “**if ... then else**” biçimindeki önerme ve kural yürütme işlemlerine tabi tutularak yine bulanık bir sonuca ulaşılmaktadır.

Bulanık işlemcilerin daha genel bir adı olarak “approximate reasoning” terimi kullanılmaktadır. Yaklaşık düşünme, yaklaşık neden olma, yaklaşık sonuçlandırma gibi anlamları içine alan bu tanımlama, bulanık küme teorisini çağrıştıran bir ifadedir. Ancak “bulanık mantık” daha çok kabul gören bir adlandırmadır. “Approximate reasoning” ya da “yaklaşık düşünme” kavramı, bulanık mantık ve bulanık küme teorisi ile desteklendiğinde “fuzzy reasoning” ya da “bulanık düşünme” kavramını ortaya çıkarmaktadır.

Karar verme uygulamalarında bulanık kümeler değişik şekillerde kullanılabilir. Karar verme uygulamalarından birisi tek veya çok ölçütlü karar vermede bulanık tercih modelleridir. Burada klasik yaklaşımdaki kesin ölçütler yerine bulanık ölçütler kullanılarak daha iyi sonuçlar elde edilmesi amaçlanmaktadır. Aslında klasik yaklaşımlarla bulanık yaklaşımlar arasındaki temel fark, bilgideki belirsizliğin hangi safhada kaldırıldığıdır. Bütün gerçek sistemlerde bilgide belirsizlik mevcuttur. Halbuki sonuç olarak kesin bir karara varılması ve bunun uygulanması gerekmektedir. Klasik yaklaşımlarda önce belirsizlik ortadan kaldırılmakta, daha sonra da “belirli” bilgi kullanılıp analiz yapılarak kesin bir sonuca ulaşılmaktadır. Bulanık yaklaşımlarda ise belirsizlik korunarak analiz yapılmakta; analizden sonra belirsizlik kaldırılıp sonuca ulaşılmaktadır. Bu şekilde, sonucun daha sağlıklı olması ve gerçek durumu temsil etmesi sağlanır. Ancak bu iyileşmeye karşın, analiz safhasının karmaşık hale gelmesi kaçınılmaz olmaktadır.

Proje çizelgeleme konusunda yapılan çalışmaların büyük çoğunda, problemle ilgili tüm bilgilere sahip olunduğu ve problemin statik, deterministik ortamda çalıştığı varsayılmaktadır. Ancak gerçek hayatta proje faaliyetleri büyük ölçüde belirsizdir ve

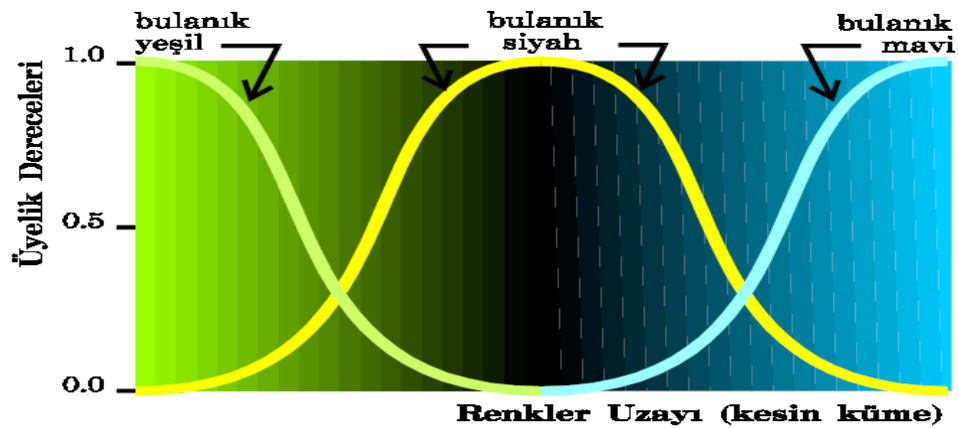
proje uygulanırken aşama aşama çözülmektedir. Bulanık küme teorisi tam olarak tanımlanması zor olan sistemlerin modellenmesinde kullanılmaktadır. Bulanık küme teorisi, kesin bilginin olmadığı ve özneliğin bulunduğu bir modeli formüle ederek çözüm sürecine alan bir yöntemdir. Dolayısıyla belirsizlikle dolu bu problemin en uygun çözümünü bulmak için belirsizlikle uğraşabilecek çözüm tekniklerine ihtiyaç vardır. İlk kez yapılacak bir projenin faaliyetlerinin tam ve doğru bir şekilde belirlenmesi zor olmaktadır. Bu durumda faaliyet sürelerinin özel olarak tahmin edilmesinde belirsizlik fikri vardır. Böyle bir problemin çözümünde günümüzde kullanılabilecek en iyi yöntemlerden biri bulanık küme teorisi yöntemidir.

4.2 Bulanık Mantığın Avantajları

- Bulanık mantık anlaşılması kolay temellere sahiptir. Çünkü arkasındaki matematik kuralları basittir ve yaklaşımı karmaşık olmaktan çok doğaldır.
- Bulanık mantık esnektir. Kullanılacağı sisteme herhangi bir seviyede kolaylıkla entegre olmaktadır.
- Hassas/kesin olmayan verileri kullanabilmektedir. Bütün ölçümler ve veriler yeteri hassaslıkta incelendiği zaman belli miktarda belirsizlik içermektedir. Bulanık mantık bu anlayışa göre probleme yaklaşmakta ve bundan faydalanarak çözüme ulaşmaktadır.
- Çok karmaşık, lineer olmayan sistemleri bile modelleyebilmektedir. Herhangi bir giriş-çıkış ilişkisini gerçekleyecek bir bulanık sistem oluşturmak mümkündür.
- Uzman bilgisi kullanarak çalışabilmektedir. Bulanık mantık, sistemi bilen uzman insanların tecrübesini ve bilgisini kullanmamızı sağlamaktadır.
- Var olan denetim teknikleriyle birlikte kullanılabilmektedir. Bulanık sistemler, klasik denetim yöntemlerinin yerine geçmektense çoğu uygulamada onları desteklemekte ve uygulanmalarını kolaylaştırmaktadır.
- Bulanık mantık konuşma diline daha yakındır. Kullanılan yaklaşım insanların iletişimini örnek almaktadır.

4.3 Bulanıklık Kavramı

Bulanık mantık konusunun temel elemanı bulanık kümedir. Bulanık kümeler, üyelik fonksiyonları ile karakterize edilmektedir. Bulanık mantık, üyelik fonksiyonu ve bulanık sayı gibi kavramların anlaşılması için öncelikle bulanıklık kavramının iyi anlaşılması gerekir. Şekil 4.1’de renkler uzayında tanımlı yeşil, siyah ve mavi değişik tonlar kullanılmıştır. Örneğin soldan sağa doğru ilerledikçe yeşilin renk tonu koyulaşmakta ve siyaha dönüşmektedir. Şeklin tam ortasında renk tam siyahken, sağa doğru ilerleme sürdürülürse, siyahın renk tonu açılıp mavi olmaktadır. Görüleceği gibi yeşilin bitip siyahın başladığı, siyahın bitip mavinin başladığı noktalar kesin bir şekilde ayrıştırılamamaktadır. Verilen üç renk bölgesi de kesin, sabit bir renk tonuna sahip değildir. Bu üç renk bölgesini birer bulanık küme ile temsil etmek uygun olacaktır. Verilen şekilde sadece yeşil, siyah ve mavinin tonları bulunduğundan, sadece bu üç rengi temsil eden yeşil, siyah ve mavi bulanık kümelerini tanımlamak yeterli olacaktır. Şekil 4.1.’in sol yarısındaki ilk bölgede yeşilden siyaha doğru bir geçiş vardır. Bu bölgede sağa doğru ilerledikçe, bölgenin yeşil bulanık kümesine ait olma derecesi azalırken, siyah bulanık kümesine ait olma derecesi de artmaktadır. Şeklin sağ yarısındaki bölgede ise siyahtan maviye bir geçiş vardır. Dolayısıyla, şeklin ortasından sağa doğru ilerledikçe, bu bölgenin siyah bulanık kümesine ait olma derecesi azalmakta, mavi bulanık kümesine ait olma derecesi de artmaktadır. Yeşil-siyah tonlarının hakim olduğu şeklin sol yarısının mavi bulanık kümesinde hiç bir üyeliği yoktur. Siyah mavi tonlarının hakim olduğu sağ yarının da yeşil bulanık kümesinde hiç bir üyeliği bulunmamaktadır.

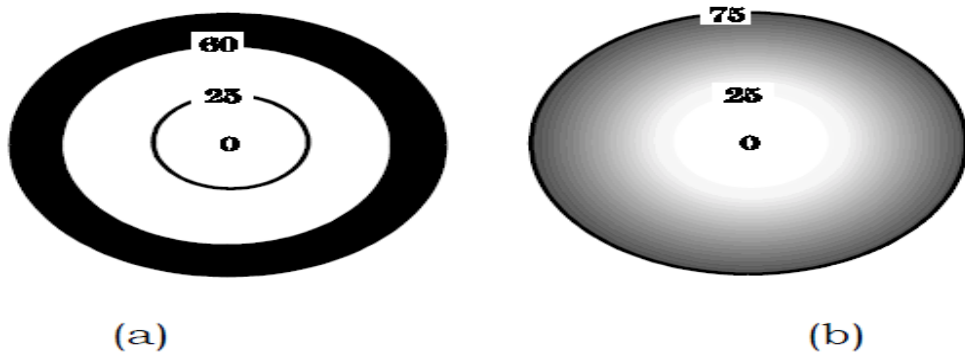


Şekil 4.1 : Yeşil, siyah ve mavi bulanık renk kümeleri.

4.4 Bulanık Kümeler Teorisi

Bulanık sistemlerin en temel elemanı bulanık kümedir. Bulanık bir küme, değişik üyelik (ait olma) derecelerine sahip elemanları olan bir küme türüdür. Böyle bir küme, elemanlarının her birine 0 ile 1 arasında üyelik değeri atayabilen bir üyelik fonksiyonu ile karakterize edilmektedir. Kümeye dahil olmayan elemanların üyelik değerleri 0, kümeye tam dahil olanların üyelik değerleri de 1 olarak atanmaktadır. Kümeye dahil olup olmadıkları belirsiz olan elemanlara ise belirsizlik durumuna göre 0 ile 1 arasında değerler atanır. Oysa kesin küme teorisinde belirsiz eleman diye bir şey söz konusu değildir. Bir eleman, ya kümeye dahildir ya da tamamı ile kümenin dışındadır. Dolayısıyla kesin kümelerde bir elemanın alabileceği üyelik değeri ya 0 ya da 1'dir.

Şekil 4.2'de yaşlı insanlar için kesin ve bulanık kümeler gösterilmiştir. Bu şekillerde siyah rengin tonu yaşlılık düzeyini belirtmektedir. Şekil 4.2.a'daki kesin kümeye göre yaşı 60 ve üzerinde olanlar yaşlı, 60'dan küçük olanlar yaşlı değildir. Oysa Şekil 4.2.b'de sadece yaşı 75'in üzerinde olanlar değil, yaşı 25 ile 75 arasında olanlar da yaşlılar kümesine dahildir.

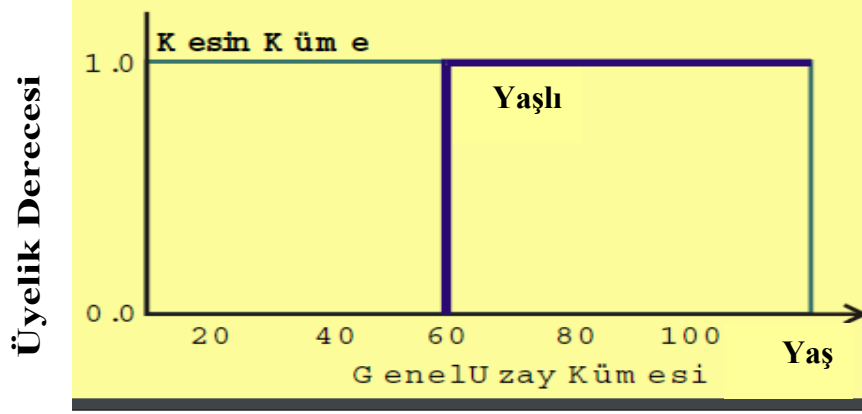


Şekil 4.2 : Yaşlılar kümesinin kesin ve bulanık kümelerle gösterimi.

Şekil 4.2.a ve b'de verilen kesin ve bulanık kümeler sırasıyla Şekil 4.3 ve Şekil 4.4'de gösterildiği gibi üyelik fonksiyonları (karakteristik fonksiyonlar) ile temsil edilmektedir.

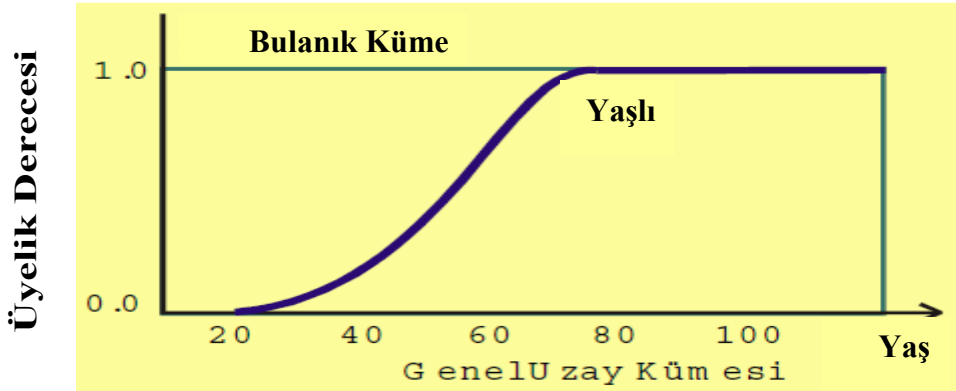
Şekil 4.2'deki kümeler yerine üyelik fonksiyonlarını kullanmak daha yararlı ve anlaşılır olacaktır. Görüldüğü gibi, üyelik fonksiyonlarının kullanılması, elemanların kümelere ait olma derecelerini, 0 ile 1 arasında değişen sayılara atama olanağı verir. Üyelik fonksiyonları kullanıldığında da, yaşlı kesin kümesine göre, yaşı 60 ve daha

büyük olanlar yaşlı, yaşı 60'dan daha küçük olanlar ise yaşlı değildirler. Yani 59 yaşındaki biri yaşlı sayılmazken 60 yaşındaki biri yaşlı sayılmaktadır. Bu da şu anlama gelmektedir. Yaşlı kesin kümesine göre 59 yaşındaki bir insan kesinlikle yaşlı değilken 60 yaşındaki bir insan kesinlikle yaşlıdır.



Şekil 4.3 : Yaş genel uzayında tanımlı yaşlı kesin kümesi.

Yaşlı insanlar bulanık bir küme ile temsil edilirse bu yeni küme Şekil 4.4'de verildiği gibi 20 ile 75 yaşları arasındakileri de kapsar. Ancak bu kapsama klasik kümede olduğu gibi tam bir kapsama değildir. Yani yaşı 20 ile 75 arasında olanlar belirli derecelerle bu kümenin elemanlarıdır. Örneğin yaşı 20'nin altında olanların, yaşlı bulanık kümesindeki üyelik dereceleri sıfır iken, yaşı 20'nin hemen üzerinde olanların üyelik derecesi sıfırın biraz üzerinde, yaşı 75'e gelmek üzere olanların üyelik derecesi de 1'e yakın olmaktadır. Örneğin, 25 yaşındaki birisinin yaşlı kümesindeki üyelik derecesi oldukça az iken, 65 yaşındaki birinin üyelik derecesi oldukça fazladır.



Şekil 4.4 : Yaş uzayında tanımlı yaşlı bulanık kümesi.

Şekil 4.3 ve Şekil 4.4’de verilen kümeler aslında yaş genel uzayında tanımlı olan ve yaşlı kümesini sırasıyla kesin ve bulanık biçimlerde tanımlayan birer üyelik fonksiyonudur. Herhangi bir bulanık küme, elemanlarının ait olma derecelerini gösteren bir karakteristik veya üyelik fonksiyon ile temsil edilmektedir. Örneğin bir $\tilde{A}$ bulanık kümesi genel anlamda aşağıdaki gibi gösterilebilir.

$$\tilde{A} = \{x, \mu_{\tilde{A}}(x) \mid x \in X\} \quad (4.1)$$

$$\tilde{A} = \sum_{x_i \in X} \frac{\mu_{\tilde{A}}(x_i)}{x_i} \quad (\text{Kesikli Biçim}) \quad (4.2)$$

$$\tilde{A} = \int_x \frac{\mu_{\tilde{A}}(x)}{x} \quad (\text{Sürekli Biçim}) \quad (4.3)$$

Bu denklemlerde;

X uzay kümesi (kesin küme)

x uzay kümesinin kesin küme elemanları

$\tilde{A}$ bulanık küme

$\mu_{\tilde{A}}(x)$ x kesin sayılarının $\tilde{A}$ bulanık kümesindeki üyelik dereceleridir.

Denklem (4.1-4.3) ile verilen ifadeler, bir bulanık kümenin üyelik fonksiyonları ile karakterize edilmesini göstermektedir. Bu denklemlerde kullanılan $\sum$ ve $\int$ işaretleri toplama ya da integral alma anlamında değil, üyelik fonksiyonlarının birleşimi anlamında kullanılmaktadır. Ayrıca bölme çizgisi de bölme yapmak amacıyla değil, sadece bir işaret olarak kullanılmaktadır.

4.4.1 Klasik kümeler

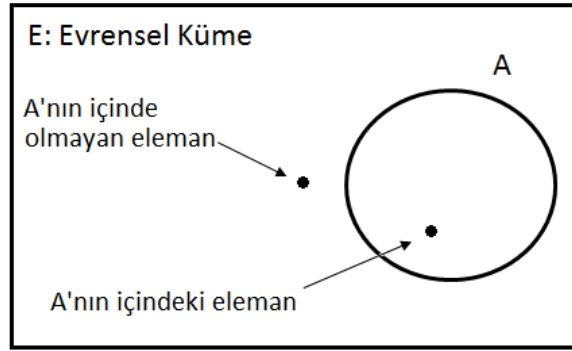
Küme, farklı elemanlar topluluğudur. Kesin küme, verilen bir evrensel kümede elemanların üyeliklerine göre ikiye ayrılması olarak tanımlanmaktadır. Bunlardan ilki üye olanlar, yani belirli bir kümeye ait olanlardır. İkincisi ise üye olmayanlar,

yani belirli bir kümeye ait olmayanlardır. Klasik küme teorisinde üyeler ve üye olmayanlar arasında kesin bir ayrım bulunmaktadır. Buna göre klasik küme teorisinde her eleman mutlaka 0 veya 1 değerini almalıdır. Denklem (4.4)'de gösterildiği gibi eğer bir eleman kümeye dahilse 1, dahil değilse 0 değerini almaktadır. Küme teorisinde bu, üyelik fonksiyonu olarak temsil edilmektedir.

$$\mu_A(x) = \begin{cases} 1, & \text{Eğer } x \in A \\ 0, & \text{Eğer } x \notin A \end{cases} \quad (4.4)$$

- 1 değeri tam üyeliği temsil etmektedir.
- 0 değeri tam olarak küme dışında kalmaya karşılıktır.

Şekil 4.5'te klasik küme teorisi gösterilmiştir.



Şekil 4.5 : Klasik küme teorisi.

Klasik kümeler pek çok değişik durumu modelleme konusunda başarılı olsa da gerçek hayatta çoğu zaman karşılaşılan kesin olmayan niceliklerle baş etme konusunda yetersiz kalmaktadır. O halde bir kümeyi tanımlamak için iki yol vardır; bir listeyle açıkça (*explicitly*) veya kural tanımlayarak (*implicitly*). Yukarıdaki örneklerden de görüleceği gibi klasik kümelerde, bir eleman bir kümeye ya tamamen dahildir (üyelik derecesi bir) ya da tamamen hariçtir (üyelik derecesi sıfır). Bu da küme tanımlamasında üçüncü bir yöntem olarak, sadece sıfır veya bir değerine sahip üyelik işlevi (karakteristik işlev olarak da bilinir) kullanmamıza olanak sağlamaktadır.

4.4.2 Bulanık küme

Bulanık küme, kesin geçişleri elemine ederek belirsizlik kavramının tanımını yeniden vermekte ve evrendeki bütün bireyleri, üyelik derecesi değerini atayarak matematiksel olarak tanımlamaktadır. Bu derece, bulanık küme tarafından verilen kavram ile uyumludur ve benzer bir bireyin derecesine uymaktadır. Böylece bireyler, bulanık küme içerisinde üyelik dereceleri tarafından gösterilen daha büyük ve daha küçük değerlere ait olabilmektedir. Bu üyelik dereceleri $[0-1]$ aralığında gerçel değerler ile ifade edilmektedir.

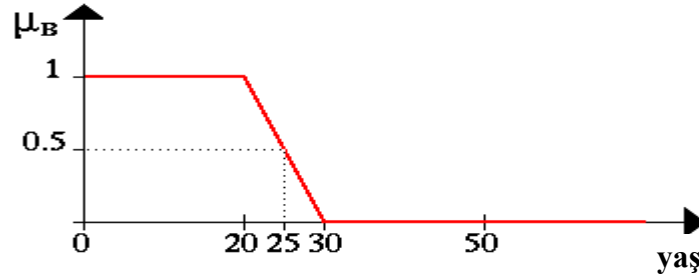
Zadeh'ten sonra kümeler üyeliğin tanımı tamamen değişmiştir. Bulanık kümelerde bir eleman, üyelik fonksiyonlarında belirlenen derecelerle bir kümeye dahildir. Zadeh'in önerdiği bu üyelik derecesi modelinde, üyelikten üye olmamaya geçiş, ani bir sıçrama değil, sürekli bir fonksiyon şeklindedir. Genç insanların kümesini tanımlamak istediğimizi varsayalım. Bunu formel olarak şu şekilde ifade edebiliriz:

$$B = \{\text{genç insanlar kümesi}\}$$

Bu küme için alt sınır açık olmakla beraber üst sınırın belirlenmesi hiç de açık değildir ve bu zorluk bulanık kümeler niye gereksinim duyulduğunu açıklamak için iyi bir başlangıçtır. Üst sınır olarak bir ilk değer vermeye çalışalım ve B'yi aşağıdaki gibi gösterelim: $B = [0,20]$. İşte bu noktada karşımıza; “*tam yirmi yaşındaki bir insan niye gençtir; fakat bir gün önce doğan bir başkası genç değildir?*” sorusu çıkar. Bu açıkça yapısal bir problemdir ve genç-genç olmayan sınırını nereye koyduğumuzdan tamamen bağımsızdır.

B kümesini oluşturmanın daha iyi bir yolu genç ve genç olmayanlar arasındaki sınırı biraz gevşetmek olacaktır. Bunu yapabilmek için yalnızca genç olan ve genç olmayan üyelikler yerine; genç insanlar kümesine biraz üye, daha az üye gibi esnek tanımlara da izin vermek gerekecektir. Bir önceki örnekte tanım uzayımızdaki bütün elemanlara ya 1 ya da 0 değeri atanmıştı; ve ikisinin arasındaki geçiş(ler) adım fonksiyonu şeklindeydi. Bu küme kavramını genelleştirmenin en açık yolu 0 ve 1 arasında başka üyelik değerlerine de izin vermektir. Bu durumda tanım uzayındaki elemanlara atanan sayıları yorumlamak biraz daha zorlaşacaktır. Tabii ki eğer elemanın değeri 1 ise B kümesinin üyesi olacak, 0 ise B kümesinin dışında olacaktır. Aradaki değerlerse her iki kümeye de (toplamları bir olmak üzere) kısmi üyeliği

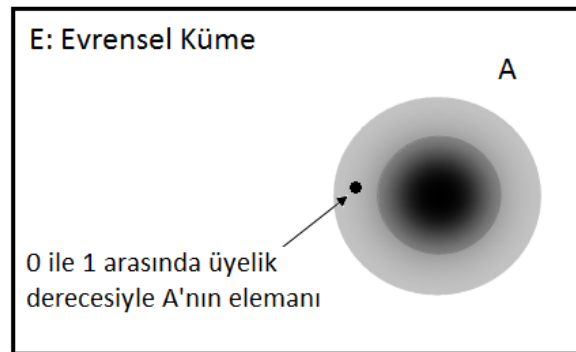
ifade edecektir. Şekil 4.6'da genç insanlar kümesi ve yeni tanımlanmış üyelik fonksiyonu görülmektedir.



Şekil 4.6 : Bulanık Küme.

Böyle bir üyelik tanımıyla 25 yaşındaki bir insan %50 oranında genç olacaktır. Burada verilen üyelik işlevini bulmak için tek bir yol veya tek bir analitik metot yoktur ve bu işlev(ler)in belirlenmesi tamamen sistemin tasarılayıcısına bırakılmıştır. Dolayısıyla bu; üyelik derecesi var olan, sistemin incelenmesinden sonra ortaya çıkarılmış kurallara göre kesin olarak belirlenebilen; fakat duruma bağlı olan ve nesnellik taşımayan bir ölçüdür.

Kesin kümelerin tersine bulanık kümelerde her elemana $[0,1]$ aralığında üyelik değerleri verilerek küme oluşturulmaktadır. Üyelik değeri, bir elemanın o bulanık kümeye olan aidiyet derecesini belirtmektedir. Bulanık küme teorisinde Şekil 4.7'de gösterildiği gibi A kümesinin kesin sınırları yoktur. Bunun yerine $\mu : E \rightarrow [0,1]$ üyelik fonksiyonuna göre 0 ile 1 arasında $\mu(x)$ üyelik derecesiyle x elemanı A kümesine dahildir (Şekil 4.7).



Şekil 4.7 : Bulanık Küme Teorisi.

4.4.3 Bulanık ve klasik kümelerin karşılaştırılması

Klasik küme kavramı ile bulanık kümeler arasında önemli bir farklılık bulunmaktadır. Klasik küme; kümeye kesinlikle üye veya kesinlikle üye değil biçiminde iki grubun oluşturulması ile anlamlı olmaktadır. Klasik kümede, kümeye üye olanlarla olmayanlar arasında kesin bir "geçiş" vardır ve her bireye evrensel kümede 1 veya 0 değerleri verilmektedir. Oysa doğal dilde de rastlanan birçok terim arasında bu sınırlar belirsiz olmakta ve üye olandan, üye olmayana geçiş "yumuşak" biçimde yapılmaktadır. Bulanık küme, evrende olası her mümkün bireye, bu bireyin bulanık küme içerisindeki üyelik derecesi değerini atayarak onu matematiksel olarak tanımlamaktadır. Bu üyelik derecelerinin değerlerine ise uzmanlar tarafından deneyler sonucunda ve tecrübelerle karar verilebilmektedir. Üyelik fonksiyonlarının belirlenmesi için ise genel bir yöntem bulunmamaktadır.

4.5 Bulanık Sayılar

Bulanık sayılar, reel sayıların bir bulanık alt kümesidir ve “güvenlik aralığı” fikrinin gelişmiş halini temsil etmektedir. Dubois ve Prade’e göre bulanık sayılar şu özelliklere sahip olmalıdır (Dubois ve Prade 1983).

Üyelik fonksiyonu $\mu_{\tilde{A}} x : R \in [0,1]$ olan “ $\tilde{A}$ ” bulanık sayısı için :

- $\mu_{\tilde{A}} x$, Reel sayılar kümesinden $[0,1]$ kapalı aralığında sürekli fonksiyondur.
- $\mu_{\tilde{A}} x$ konveks bulanık bir altkümedir.
- $\mu_{\tilde{A}} x_0 = 1$ yapan en az bir tane x_0 sayısı vardır.

4.6 Olabilirlik ve Olasılık

Olasılık teorisi ile bulanık mantık arasındaki ayrım, olasılık ve üyelik derecesi kavramları arasındaki farktan kaynaklanır. Bir elemanın üyelik derecesi çoğu zaman olasılık olarak anlaşıldığı için bunu açıklamakta fayda görüyoruz. Olasılık önermeleri bir olayın sonuçlarının olasılık dağılımı hakkında bilgi içermektedir. Bununla birlikte üyelik derecesi bir olayın olasılığı ile ilgili herhangi bir bilgi içermeyip; olayın var olan tanım uzayında nereye kadar uzanabileceğini, diğer bir ifadeyle *olabilirliğini* modellemeye çalışmaktadır.

Burada en çok merak edilen konu, bulanık üyelik derecelerinin olasılık değerleri ile kıyaslanmasıdır. Yani, üyelik dereceleri ile belirlenmiş bir bulanık kümenin, benzer değerler alan olasılık değerlerinden farkının ne olduğu sorusudur. Bulanık değerlendirme için bir ölçüm olan olasılıkla bulanık kümeler arasındaki farklılıklar aşağıda görülmektedir: Sonlu bir evrensel kümede olasılıklar toplamı 1'e eşit olmakla birlikte, bulanık üyelik derecelerinde böyle bir duruma ihtiyaç bulunmamaktadır.

Olasılık, kesikli değerlere sahip olmakta, bulanık kümenin bireye ilişkin üyelik dereceleri süreklilik taşımaktadır. Örneğin, bulanık kümelerde 70 kg olan bir kişi, %100 normal ağırlıkta ise, 71 kg için de normalliğin derecesi kolayca belirlenmektedir. Klasik olasılık hesapları, bireylerin tamamının temeline dayalıdır. Bulanık küme teorisinde ise bireyin üyelik derecesi, diğer bireylerin tamamının temeline ilişkin olmamakta, üyelikler bakımından farklar görülmektedir. Bu farklara rağmen, olasılıkla bulanık üyelik değerleri arasında kesin bir ilişki bulunmaktadır. Zadeh'e göre, bir olayın bulanık kümede olasılığı, bu olayın üyelik fonksiyonunun bir beklentisidir.

“Ahmet kahvaltıda X yumurta yedi” önermesini ele alalım. Ahmet'in 100 gün kahvaltıda yediği yumurta sayısını gözlemleyerek bir p olasılık dağılımı çıkarabiliriz:

$$U = [1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]$$

$$p = [.1 \ .8 \ .1 \ 0 \ 0 \ 0 \ 0 \ 0]$$

Ahmet'in kahvaltıda X yumurta yemesinin kolaylık derecesini belirten bulanık küme ise, aşağıdaki Π olabilirlik dağılımı ile gösterilmektedir:

$$U = [1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]$$

$$\Pi = [1 \ 1 \ 1 \ 1 \ .8 \ .6 \ .4 \ .2]$$

Burada $X=3$ için olabilirlik 1 iken (yani Ahmet aslında kolaylıkla üç yumurta yiyebilirken), olasılık 0.1'dir. Yani Ahmet kahvaltıda, gözlenen sürenin yalnızca onda birinde üç yumurta yemiştir. Olasılık ve bulanık mantık arasındaki bir diğer önemli fark da, ilkinde bütün olasılıkların 1'e normalize edilmesi gerekliliğidir. Yani, mümkün olan bütün sonuçların olasılıkları toplamı 1'e eşit olmalıdır. Bir üyelik fonksiyonunu bu şekilde tanımlamak mümkün olsa da bu, ne çok gerekli ne de çok yaygındır.

4.7 Tanım Uzayı

Bir bulanık kümenin elemanları, tanım uzayından alınmaktadır. Tanım uzayı, bir problem için olası bütün değerleri içerirken, aynı zamanda hatalı veya yanlış tanımlanmış verilerin de kullanılmasını engellemektedir. Görüldüğü gibi tanım uzayı, uygulamayla ve o anki bağlamla ilgilidir; aynı bulanık kümeler gibi öznel bir tanıma, daha doğrusu yoruma sahiptir. Sayısal olmayan bir nicelik ile ilgilenirken (örneğin yemeğin tadı), sayısal değerlere sahip bir tanım uzayı kullanamayız. Bu durumda elemanları fizyolojik bir devamlılıktan aldığımızı söyleriz. Örneğin acı, tatlı, ekşi, tuzlu, ya da sıcak gibi...

4.8 Üyelik Fonksiyonu

Tanım uzayında bulunan her eleman bir bulanık kümenin (bir dereceye kadar) üyesidir. Sıfırdan farklı bir üyelik derecesine sahip elemanlar, o bulanık kümenin “support”unu oluşturmaktadır. Tanım uzayının her elemanına belli bir sayı atayan işleve üyelik işlevi, bu sayıya da üyelik derecesi denmektedir. Üyelik derecesi 0 ile 1 arasında bir değerdir. Bu tanımla birlikte kesin kümeler, bu üyelik işlevlerinin kare şeklinde tanımlandığı özel durumlar olarak görülmektedir. Bu fonksiyon, elemanın küme içindeki üyelik derecesini veren ve özel bir aralıkta evrensel kümenin elemanlarına değer atayan fonksiyon olarak genelleştirilebilir. Daha büyük değerler, daha yüksek dereceli üyeliğe karşılık olurken, daha düşük değerler ise üyelik derecesinin küçük olduğunu göstermektedir. Üstte tanımlanan bu fonksiyona üyelik fonksiyonu, kümeye ise bulanık küme adı verilmektedir.

Üyelik fonksiyonlarını oluşturmada birçok yöntem bulunmaktadır. En gelişmiş yöntemler, uzman tecrübelerinden faydalanarak küme değerlerini noktalı olarak belirlemek ve analitik fonksiyon biçiminde ifade etmektir. Bulanık küme, kesin geçişleri elemine ederek belirsizlik kavramının tanımını yeniden vermekte ve evrendeki bütün bireylere üyelik derecesi değerini atayarak matematiksel olarak tanımlanmaktadır. Bu derece, bulanık küme tarafından verilen kavram ile uyumludur ve benzer bir bireyin derecesine de uymaktadır. Böylece bireyler, bulanık küme içerisinde üyelik dereceleri tarafından gösterilen daha büyük ve daha küçük değerlere ait olabilirler. Bu üyelik dereceleri $[0-1]$ aralığında gerçel değerler ile ifade edilmektedir.

Yaygın olarak kullanılan ve bu tez çalışmasının ilerleyen bölümlerinde ele alınacak olan özel üyelik işlevlerinden sadece üçgen, yamuk bulanık sayı ve üyelik fonksiyonlarından bahsedilecek, söbü, gauss, çan, sigmoid ve sigleton bulanık sayısı ve bunlara ait üyelik fonksiyonlarından bahsedilmeyecektir. Çeşitli uygulamalar için pek çok değişik üyelik işlevi önerilmiş olsa da yukarıdakiler en yaygın kullanılanlardır. Uyarlamanın önemli olduğu bulanık denetim uygulamalarında kullanılacak üyelik işlevi türevinin sürekli olması gereklidir. Dolayısıyla, üçgen ve yamuk üyelik işlevleri, bu gibi uygulamalar için pek kullanışlı değildir. Ayrıca, yukarıdaki üyelik işlevlerini aritmetik işlemlerle değişikliklere uğratarak farklı problem spesifik üyelik işlevleri üretilebilir.

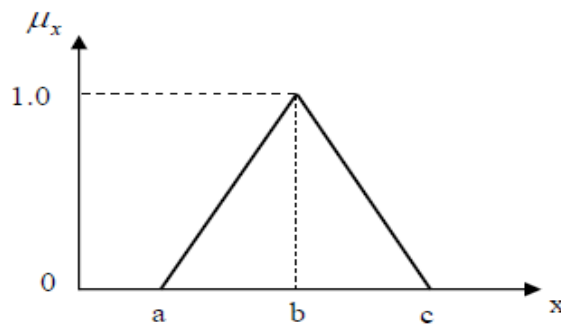
4.8.1 Üçgen bulanık sayı ve üyelik fonksiyonu

Üçgen üyelik işlevi aşağıdaki gibi tanımlanmaktadır:

$$\text{üçgen}(x, a, b, c) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 0, & c \leq x \end{cases} \quad (4.5)$$

Bunu hesaplamak için daha uygun hale getirilen bir diğer temsil şekli de aşağıdaki gibidir:

$$\text{üçgen}(x, a, b, c) = \max(\min(\frac{x-a}{b-a}, \frac{c-x}{c-b}), 0) \quad (4.6)$$



Şekil 4.8 : Üçgen Üyelik İşlevi.

Üyelik fonksiyonu Şekil 4.8’de görülen $\tilde{A}$ bulanık sayısı, bir üçgensel bulanık sayıdır (ÜBS). Burada $a \leq b \leq c$ ’dir ve a en küçük olası değeri, b en umut verici değeri, c ise en büyük olası değeri göstermektedir. Bir ÜBS genellikle (a,b,c) şeklinde gösterilmektedir. Özellikle çok ölçütlü bulanık karar verme problemlerinde yaygın bir biçimde kullanılan üçgensel bulanık sayılarla temel aritmetik işlemler şu şekildedir:

$\tilde{A}_1 = a_1, b_1, c_1$ ve $\tilde{A}_2 = a_2, b_2, c_2$ iki üçgen bulanık sayı olsun.

$$\text{Toplama} \quad \tilde{A}_1 \oplus \tilde{A}_2 = a_1 + a_2, b_1 + b_2, c_1 + c_2 \quad (4.7)$$

$$\text{Çıkarma} \quad \tilde{A}_1 \ominus \tilde{A}_2 = a_1 - c_2, b_1 - b_2, c_1 - a_2 \quad (4.8)$$

$$\text{Çarpma} \quad \tilde{A}_1 \otimes \tilde{A}_2 = a_1 \times a_2, b_1 \times b_2, c_1 \times c_2 \quad (4.9)$$

$$\text{Bölme} \quad \tilde{A}_1 \oslash \tilde{A}_2 = a_1 / c_2, b_1 / b_2, c_1 / a_2 \quad (4.10)$$

$$\text{Negatif} \quad -\tilde{A}_1 = -a_1, -b_1, -c_1 \quad (4.11)$$

$$\text{Tersi} \quad 1/\tilde{A}_1 \approx 1/c_1, 1/b_1, 1/a_1 \quad (4.12)$$

4.8.2 Yamuk bulanık sayı ve üyelik fonksiyonu

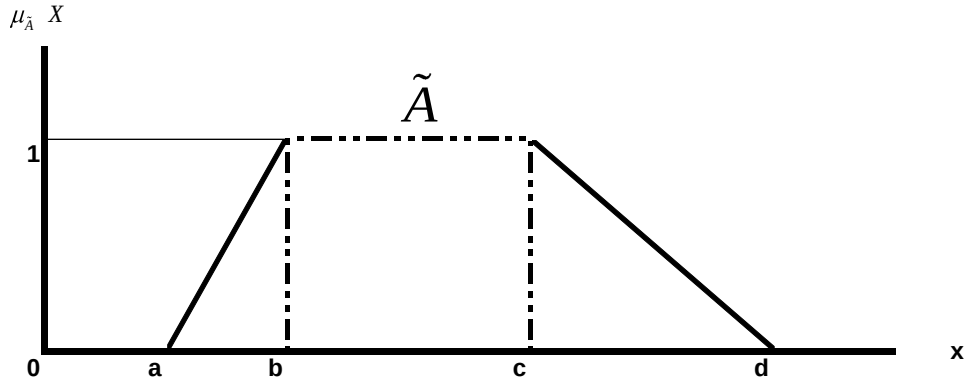
Yamuk üyelik işlevinin tanımı:

$$yamuk(x, a, b, c, d) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \\ 0, & d \leq x \end{cases} \quad (4.13)$$

Önceki örnekle paralel bir şekilde, programlama için kullanılan formülasyon:

$$yamuk(x, a, b, c, d) = \max(\min(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}), 0) \quad (4.14)$$

b ile c eşit oldukları zaman yamuk üyelik işlevi üçgen üyelik işlevine dönüşür.



Şekil 4.9 : Yamuk Üyelik İşlevi.

Yamuk bulanık sayılarla temel aritmetik işlemler şu şekildedir:

$\tilde{A}_1 = a_1, b_1, c_1, d_1$ ve $\tilde{A}_2 = a_2, b_2, c_2, d_2$ iki yamuk bulanık sayı olsun.

$$0 \leq a_1 \leq b_1 \leq c_1 \leq d_1 \quad 0 \leq a_2 \leq b_2 \leq c_2 \leq d_2$$

Toplama $\tilde{A}_1 \oplus \tilde{A}_2 = a_1 + a_2, b_1 + b_2, c_1 + c_2, d_1 + d_2$ (4.15)

Çıkarma $\tilde{A}_1 \ominus \tilde{A}_2 = a_1 - d_2, b_1 - c_2, c_1 - b_2, d_1 - a_2$ (4.16)

Çarpma $\tilde{A}_1 \otimes \tilde{A}_2 = a_1 \times a_2, b_1 \times b_2, c_1 \times c_2, d_1 \times d_2$ (4.17)

Bölme $\tilde{A}_1 \oslash \tilde{A}_2 = a_1 / d_2, b_1 / c_2, c_1 / b_2, d_1 / a_2$ (4.18)

Negatif $-\tilde{A}_1 = -a_1, -b_1, -c_1, -d_1$ (4.19)

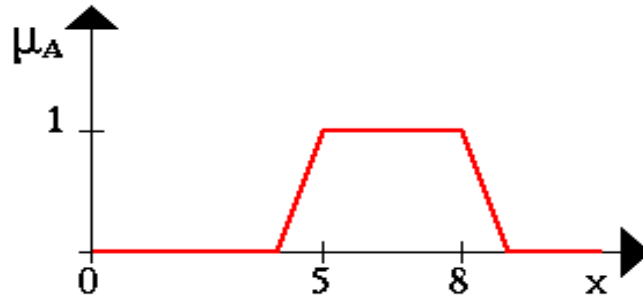
Tersi $1/\tilde{A}_1 \approx 1/d_1, 1/c_1, 1/b_1, 1/a_1$ (4.20)

Maksimum $\text{Max } \tilde{A}, \tilde{B} = \max a_1, a_2, \max b_1, b_2, \max c_1, c_2, \max d_1, d_2$ (4.21)

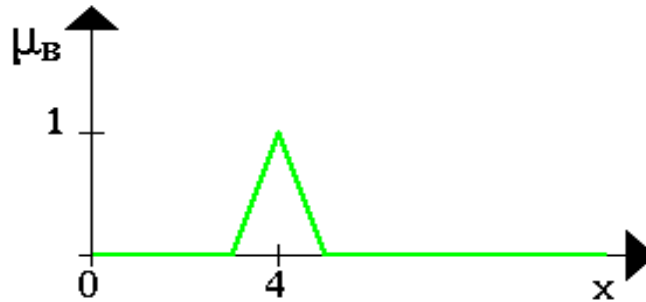
Minimum $\text{Min } \tilde{A}, \tilde{B} = \min a_1, a_2, \min b_1, b_2, \min c_1, c_2, \min d_1, d_2$ (4.22)

4.9 Bulanık Kümeler Üzerinde İşlemler

Yukarıdaki örneklerden sonra bulanık kümelerin formel tanımı ve bulanık kümeler üzerindeki işlemlerle devam etmek uygun olacaktır. Bu nedenle de bulanık kümeler üzerindeki işlemleri üyelik fonksiyonları üzerinde tanımlamak çok yerinde olacaktır. Aslında küme işlemi, bir veya birden fazla bulanık kümeden yeni bir küme oluşturma işinden başka bir şey değildir. Örneğin verilen $\tilde{A}$ ve $\tilde{B}$ bulanık kümelerinin kesişimi kendi üyelik işlevine sahip yeni bir bulanık kümedir. Klasik kümelerdeki temel işlemler bulanık kümeler için de geçerlidir. $\tilde{A}$ ve $\tilde{B}$ aynı tanım uzayında var olan ve üyelik işlevleri aşağıdaki gibi verilen iki bulanık küme olsun.



Şekil 4.10 : $\tilde{A}$ kümesinin üyelik işlevi.



Şekil 4.11 : $\tilde{B}$ kümesinin üyelik işlevi.

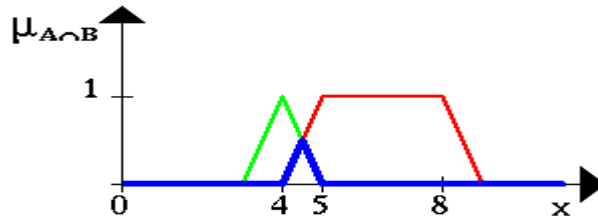
$\tilde{A}$ ve $\tilde{B}$ bulanık kümeleri için aşağıdaki işlemler tanımlanabilir.

4.9.1 Kesişim ($\tilde{A} \cap \tilde{B}$)

Verilen iki kümenin kesişimi hem $\tilde{A}$ 'da hem $\tilde{B}$ 'de var olan elemanların oluşturduğu kümedir. Kesişim işlemi aşağıdaki Şekil 4.12'de kalın çizgiyle gösterilmiştir.

$$\mu_{\tilde{A} \cap \tilde{B}}(x) = \min \mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x) = \mu_{\tilde{A}}(x) \text{ ve } \mu_{\tilde{B}}(x) \quad (4.23)$$

Buradaki **min** işlemi $\tilde{A}$ ve $\tilde{B}$ kümelerindeki elemanların karşılaştırması ve üyelik işlevinin en az olan değerinin kesişim kümesine atanmasıdır.



Şekil 4.12 : Kesişim İşlemi.

$$\mu_{\tilde{A} \cap \tilde{B}}(x) = \mu_{\tilde{A}}(x) \wedge \mu_{\tilde{B}}(x) = \min \mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x) \quad (4.24)$$

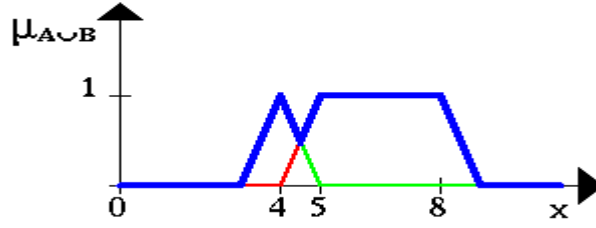
$$\text{veya } \tilde{A} \cap \tilde{B} = \sum_{i=1}^n \mu_{\tilde{A}}(u_i) \wedge \mu_{\tilde{B}}(u_i) / u_i \quad (4.25)$$

4.9.2 Birleşim ($\tilde{A} \cup \tilde{B}$)

Verilen iki kümenin birleşimi, her ikisinde bulunan bütün elemanların oluşturacağı kümedir. $\tilde{A}$ ve $\tilde{B}$ kümelerinin birleşimi aşağıda, Şekil 4.13'de, mavi çizgiyle gösterilmiştir.

$$\mu_{\tilde{A} \cup \tilde{B}}(x) = \max \mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x) = \mu_{\tilde{A}}(x) \text{ veya } \mu_{\tilde{B}}(x) \quad (4.26)$$

Buradaki **max** işlemi $\tilde{A}$ ve $\tilde{B}$ kümelerindeki elemanların birebir karşılaştırması ve maksimum değere sahip olanın seçilmesidir.



Şekil 4.13 : Birleşim işlemi.

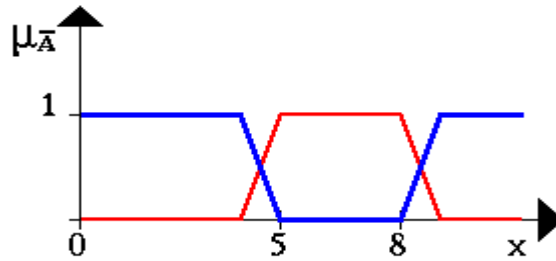
$$\mu_{\tilde{A} \cup \tilde{B}}(x) = \mu_{\tilde{A}}(x) \vee \mu_{\tilde{B}}(x) = \max \mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x) \quad (4.27)$$

$$\text{veya } \tilde{A} \cup \tilde{B} = \sum_{i=1}^n \mu_{\tilde{A}}(u_i) \vee \mu_{\tilde{B}}(u_i) / u_i \quad (4.28)$$

4.9.3 Tümleme ($\bar{\tilde{A}}$)

Verilen bir $\tilde{A}$ bulanık kümesinin tümlenyeni, tanım uzayında bulunan ancak $\tilde{A}$ kümesinde bulunmayan elemanların kümesidir. Yeni kümenin üyelik fonksiyonu $\tilde{A}$ kümesinin üyelik fonksiyonunun 1'den çıkarılmasıyla elde edilmektedir.

$$\mu_{\bar{\tilde{A}}}(x) = 1 - \mu_{\tilde{A}}(x) \quad (4.29)$$



Şekil 4.14 : Tümleme işlemi.

X kümesinin üyelik fonksiyonu tanım uzayı boyunca Y kümesinden küçükse, X kümesi Y kümesinin alt kümesidir denmekte ve $X \subseteq Y$ şeklinde gösterilmektedir. Yukarıdaki $\tilde{A}$ ve $\tilde{B}$ kümeleri için $(\tilde{A} \cap \tilde{B}) \subseteq (\tilde{A} \cup \tilde{B})$. Burada " $\vee$ " ve " $\wedge$ " simgeleri uygun olarak maksimum ve minimum işlemleri ifade etmektedir. Görüldüğü gibi bulanık kümelerdeki işlemler, klasik küme işlemlerine

benzemektedir. Fakat tümleme ile ilgili işlemlerde bazı farklılıklar vardır. Bunlar şu şekildedir:

$\tilde{A}$ bulanık kümesinin tersi ile birleşimi evrensel küme değildir : $\tilde{A} \cup \tilde{\tilde{A}} \neq X$

$\tilde{A}$ bulanık kümesinin tersi ile kesişimi boş küme değildir : $\tilde{A} \cap \tilde{\tilde{A}} \neq \emptyset$

Birleşim ve kesişim işlemlerinin özellikleri aşağıdaki tabloda özetlenmiştir. Bu özellikler uzun tanım cümlelerinin sonuçlarını kestirebilmek açısından önemlidirler. Yukarıda işlemler için başka tanımlar da mümkündür ancak en yaygın kullanılanlar **min** ve **max** işlemleridir.

4.9.4 Niteleyiciler (Modifiers)

Dilsel değiştirici, var olan bir terimin anlamını değiştiren (pekiştiren veya azaltan) bir işlemdir. Örneğin “*0’a çok yakın*” tümcesinde “*0’a yakın*” bulanık kümesi, “çok” kelimesi tarafından değişikliğe uğratılmıştır. Bu açıklamaya göre değiştirici de aslında bulanık küme üzerinde bir işlemden ibarettir. Buna benzer şekilde az, daha, neredeyse, oldukça, vb. gibi sözcükler de birer değiştirici olarak kullanılabilir. Çok değiştiricisinin etkisini tam olarak ifade etmek zor olsa da, güçlendirici bir etkisi olduğunu söylemek de lazımdır. Biraz değiştiricisi ise, tam tersi bir etkiye sahiptir. Bu işlemciler çoğunlukla aşağıdaki gibi temsil edilmektedir:

$$\text{çok } a \equiv a^2 \quad (4.30)$$

$$\text{biraz } a = a^{1/2} \quad (4.31)$$

Üstel fonksiyon, a kümesinin her elemanına etki etmektedir. Yalnız kare ve karekök alma yoluyla değil, farklı değiştiriciler için uygun olan üstel fonksiyon kullanılarak da verilen kümeyi değiştirmek mümkün olmaktadır .

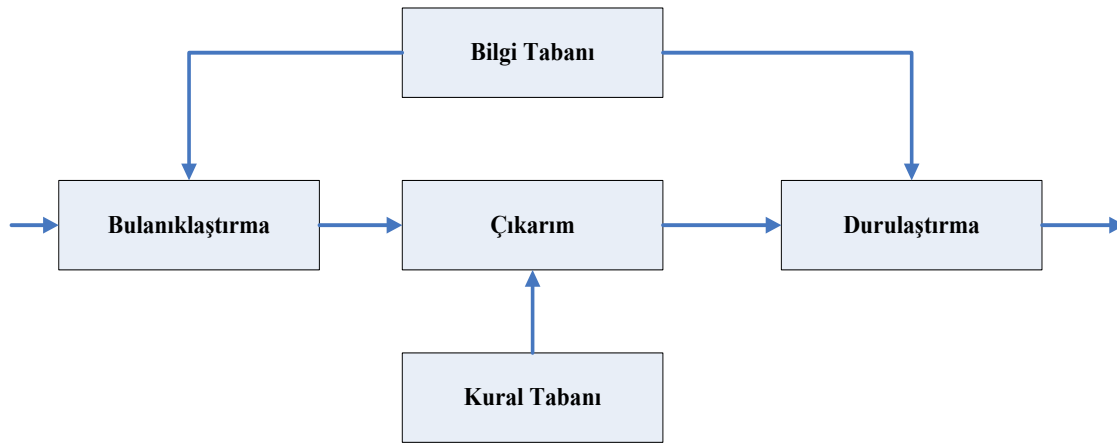
4.10 Bulanık Denetim

Bulanık mantık ve bulanık mantık tabanlı uygulamalar son yıllarda hızlı bir gelişme göstermişlerdir. Bulanık mantığın insan zihnine paralel bir yapıda olması ve bu paralellikte sistemleri tanıyıp karar vermesi, onun hızlı bir şekilde geniş uygulama alanları bulmasına neden olmuştur. Özellikle bulanık mantığın yapay sinir ağları ve genetik algoritmalarla birlikte kullanılması akıllı denetim sistemlerini ortaya

çıkarmıştır. Daha önceki bölümde bulanıklılık konusu ele alınıp, bulanık sistemlerin temel yapısı ve bulanık kümeler hakkında bazı bilgiler verilmişti. Bu nedenle burada bulanıklılık üzerinde fazla durmadan doğrudan bulanık denetim sistemlerinin temel kavramlar üzerinde açıklamalara geçilecektir.

4.11 Bulanık Denetimin Temelleri

Bulanık denetleyiciler bulanık kuramın en önemli alanıdır. Geleneksel denetimcilerden farklı olarak, türevsel denklemlerin yerine, uzman verisi kullanarak bir sistem tanımlanmaktadır. Bu bilgi, bulanık kümelerde belirtilen dilsel değişkenlerle ifade edilmektedir. Şekil 4.15’de verilen bulanık mantık tabanlı denetim sistemi, bulanık mantığın nasıl kullanıldığı ve bulanık işlemlerin nasıl gerçekleştirildiği konusunda yeterli bir fikir vermektedir. Şekil 4.15. bulanık mantık tabanlı herhangi bir sistemde bulunması gereken kısımları içermektedir. Dolayısıyla bu şekilde verilen sistem, genel amaçlı bir bulanık mantık sistemi gibi de düşünülmektedir. Bunların her biri aşağıda kısaca açıklanmaktadır. Bulanık uzman sistemlerin karar verme mekanizmaları 4 ana bloktan meydana gelmektedir: **bulanıklaştırma, çıkarım, birleştirme ve durulaştırma.**



Şekil 4.15 : Bulanık Denetim sistemi.

Denetimi, bulanık mantığa dayalı olan sistemler günümüzde çok yaygın olarak kullanılmaktadır. Akıllı olarak nitelendirdiğimiz aletler bu cinstendir. Bunlara örnek olarak otomatik çamaşır makinesi, kablosuz ütü, mikrodalga fırın, elektrik süpürgesi, ısıtıcı, televizyon, tıraş makinesi, hava temizleyici vb. gösterilebilir. Diğer örnekler ise, ters sarkaç (inverted pendulum), uçak iniş kontrol sistemi (aircraft landing

control), şoförsüz araba uygulamaları, artık klasikleşmiştir. Ayrıca bulanık mantık, ses ve örüntü tanıma problemlerinde de sıklıkla kullanılmaktadır.

4.11.1 Bulanıklaştırıcı

Bulanıklaştırıcı, bulanık işlemcinin ilk elemanı olarak devreye girmektedir. Çünkü o, bulanık işlemci girişlerinin uygulandığı birimdir. Bulanıklaştırıcı (fuzzifier) kendisine uygulanan kesin girişleri bulanıklaştırarak işe başlamaktadır. Giriş değişkenlerini ölçer, onlar üzerinde bir ölçek değişikliği yapar ve onları bulanık kümelere dönüştürür, burada amaç, giriş değerlerini dilsel niteleyicilerle tanımlamak, onlara terim setimizden birer dilsel değişken atamaktır. Bu girişlerden bazıları, denetlenen sistemden gelen geri besleme işaretleri olabileceği gibi, doğrudan dışardan verilen komutlar da olabilir. Bulanıklaştırıcıda elde edilen üyelik değerleri, aslında x ve y kesin girişlerinin ait oldukları bulanık kümedeki bulanıklılık dereceleridir. Kural işleme ünitesi, bulanıklaştırıcıdan gelen üyelik fonksiyonları burada depolanmış halde bulunan bilgi tabanına dayalı bilgi kümeleri ile birlikte kullanılarak bulanık bir sonuç elde edilmektedir.

4.11.2 Bulanık çıkarım

Bu blok kurallar üzerinde bulanık mantığı yürütmekle yükümlüdür. Bir şekilde insan beyninin düşünüş tarzını taklit etmeye çalışmaktadır.

4.11.3 Bilgi tabanı

Sistemin giriş ve çıkış değişkenleri için tanımlanmış olan üyelik işlevleri burada bulunmakta, çıkarım motoru bu üyelik işlevlerini kullanarak sonuca ulaşmaktadır.

4.11.4 Kural tabanı

Denetim amaçlarına uygun dilsel denetim kuralları (eğer su seviyesi çok azsa, vanayı çok aç) burada bulunmakta, ve çıkarım motoruna buradan verilmektedir. Kullanılan bilgi tabanı (knowledge-base) denetlenecek sistemle ilgili bilgilerin toplandığı bir veri tablosundan ibarettir. Burada sistemle ilgili bilgiler, sistem giriş ve çıkışını **if then else** biçimine sahip koşul cümleleriyle birbirine bağlamaktadır. Bu koşul cümlelerinin her biri, bir kural olarak isimlendirilmektedir. Bu nedenle de bilgi

tabanı yerine kural tabanı (rule-base) terimi de kullanılmaktadır. Bulanık işlemlerin bu aşamasında, kurallara dayalı işlemler yapıldığı için ilgili blok kural işleme ünitesi olarak da isimlendirilebilir. Yani x ve y değerlerine bağlı olarak z çıkışının bulanık kümelerinden hangisi veya hangilerinde üyeliğe sahip olması gerektiği konusundaki bilgi, bu sistemi iyi bilen uzmanlardan alınmakta ve kurallar biçiminde depolanmaktadır. Bu nedenle de böyle kuralların oluşturduğu tablolara bilgi tabanı (knowledge-base) adı verilmektedir.

4.11.5 Durulaştırıcı

Çıkarım motoru, bulanık küme çıkışları üzerinde gereken değişiklikleri yapmakta ve bunları gerçek (crisp) sayılara dönüştürmektedir. Durulaştırıcı, bulanık işlemcinin son elemanı olarak devreye girmektedir. Bulanık kural işleme ünitesinde elde edilen kural sonuçları, durulaştırıcıda değerlendirilip, kesin bir sonuca dönüştürülmektedir.

Mamdani tarafından kullanılan ve maksimumun ortalaması (MOM - mean of maximum) olarak adlandırılan yöntemdir. Bu yöntem bulanık mantık ile yapılan bulanık küme ve üyelik fonksiyonu işlemlerinin bir sonucu olarak doğrudan ortaya çıkmaktadır. Bu durulaştırma yöntemine alternatif olarak daha ileri düzeyde bazı ek yöntemler de geliştirilmiştir. Bunlardan en çok bilinenleri; durulama ile ilgilidir ve genellikle üyelik fonksiyonlarının birleşiminin ve ağırlıklarının hesaplanmasına dayalıdır. Örnek olarak aşağıdaki durulama yöntemleri gösterilebilir.

- Maksimum Üyelik İlkesi (Max-Membership Principle)
- Ağırlık Merkezi Yöntemi (COA-Center Of Area)
- Ağırlıklı Ortalama Yöntemi (Weighted Average Method)
- Ortalama Maksimum Üyelik Yöntemi (Mean Max Membership)
- Alanların Merkezi Yöntemi (COA - center of area)
- İki Bölge Yöntemi (BOA - bisector of area)
- Maksimumun Ortalaması (MOM - mean of maximum)
- Maksimumun .En Küçüğü (SOM - smallest of maximum)
- Maksimumun En Büyüğü (LOM - largest of maximum)
- Sugeno Bulanık Modelleri

- Tsukamoto Bulanık Modelleri

Bunlardan en sık kullanılan ağırlık merkezi yöntemi'nde tüm dilbilimsel çıkış değerlerine tek bir ortalama ağırlık verilmektedir. Merkezi durulama yönteminin karmaşıklığı, çıkış üyelik fonksiyonlarının biçimine bağlıdır ve kesin değerlerin elde edilmesinde aşağıdaki bağıntıdan faydalanılmaktadır.

4.12 Bulanık Sayıların Sıralanması

Bulanık sayılar muğlak, kesin olarak tanımlanamayan, bu değerleri sayılaştırabilmek için kullanıldığından, çeşitli uygulamalar açısından bulanık sayıların birbirleriyle kıyaslanabilmesi yada sıralanması oldukça önemlidir. Bulanık sayıların sıralanması yada derecelendirilmesi bulanık optimizasyon ve bulanık karar verme yöntemlerindeki temel problemidir.

Bulanık değerlerin sıralanması bulanık kümelerin değişik özelliklerine dayalı olarak yapılmaktadır. Bunlar; çekim merkezi, üyelik derecesi fonksiyonu altındaki alan veya bazı kesişim noktalarıdır. Doğal olarak farklı sıralama yöntemleri aynı veriler için değişik sıralama sonuçları verecektir (Chen ve Hwang, 1992).

Bulanık sayılar, reel sayılarda olduğu gibi doğal bir sıra oluşturmadıkları için, bulanık sayıları sıralamak için çok çeşitli yöntemler kullanılmaktadır. Literatürde bu konuda pek çok çalışmaya rastlanmaktadır. Her yöntemin kendine göre avantajları ve dezavantajları vardır ve hangi yöntemin eniyi olduğu konusunda karar verebilmek oldukça güçtür. McCahon ve Lee (1990), Bortolan ve Degani (1985), Chen ve Hwang (1992), Prodanovic ve Simonovic (2002) değişik sıralama yöntemlerini kıyaslamaktadırlar. Söz konusu problemin karmaşıklığına, hassasiyetine, yorumlanabilme kolaylığına ve bulanık sayıların şekline bağlı olarak değişik sıralama yöntemleri kullanılmaktadır. Bulanık sayıları sıralamak için yaygın bir şekilde kabul görmüş olan bir yöntem ilk olarak Baas ve Kwakernaak (1977) tarafından önerilmiştir. Tong ve Bonissone (1980), "baskınlık ölçüsü" kavramını tanımlayıp bunun Baas ve Kwakernaak'ın sıralama ölçüsüne denk olduğunu ispatlamışlardır. Daha sonra bu yöntem Buckley (1985) tarafından da kullanılmıştır.

Literatür taraması sonucu pek çok bulanık sıralama yöntemi olduğu görülmektedir. Bu yöntemlerin bazıları: Balwin ve Guild (1979), Chen (1985), Chen ve Klien (1997), Cheng (1998), Dubois ve Prade (1983), Kim ve Park (1990), Lee ve Li

(1988), Liou ve Wang (1992), Abdel- Kader ve Dugdale (2001), Matarazzo ve Munda (2001), Modarres and Sadi-Nezhad (2001), Peneva ve Popchev (1998), Yao ve Wu (2000), Yager (1981), Ibrahim ve Ayyub (1998), Ibanez ve Munoz (1989) ve Raj ve Kumar (1999) dır.

Bu çalışma kapsamında kullanılan yöntem ile literatürde yer alan bir kaç sıralama yöntemine açıklık getirilecektir.

4.12.1 Liou ve Wang’ın yöntemi

Liou ve Wang’ın toplam integral değer yönteminde; $\alpha \in 0,1$ iyimserlik indeksi olmak üzere; $\tilde{A} = (a,b,c)$ şeklinde verilen ÜBS için, toplam integral değer şu şekilde hesaplanmaktadır:

$$I_T^\alpha \tilde{A} = \frac{1}{2} \alpha (b+c) + \frac{1}{2} (1-\alpha) (a+b) = \frac{1}{2} [\alpha c + b + 1 - \alpha a] \quad (4.32)$$

Karar vericinin iyimserlik indeksi olarak tanımlanan α ; $0 \leq \alpha \leq 1$ ’dir. α büyüdükçe iyimser bir karar verici, küçüldükçe de karamsar bir karar verici söz konusudur.

$\tilde{A}_i$ ve $\tilde{A}_j$ bulanık sayıları için eğer;

$$I_T^\alpha \tilde{A}_i < I_T^\alpha \tilde{A}_j \quad \text{ise} \quad \tilde{A}_i < \tilde{A}_j \quad (4.33)$$

$$I_T^\alpha \tilde{A}_i = I_T^\alpha \tilde{A}_j \quad \text{ise} \quad \tilde{A}_i = \tilde{A}_j \quad (4.34)$$

$$I_T^\alpha \tilde{A}_i > I_T^\alpha \tilde{A}_j \quad \text{ise} \quad \tilde{A}_i > \tilde{A}_j \quad (4.35)$$

4.12.2 Abdel-Kader ve Dugdale’ın yöntemi

Abdel-Kader ve Dugdale’a göre, bir bulanık sayı; (1) tam üyelikler, (2) sağ taraftaki kısmi üyelikler ve (3) sol taraftaki kısmi üyelikler olmak üzere üç kısımdan oluşmaktadır. Mevcut sıralama yöntemleri ya sol taraftaki üyelikleri ya da her iki taraftaki üyelikleri yansıtmaktadır. Abdel-Kader ve Dugdale (2001), sıralama prosesinde bir bulanık sayının yukarıda sıralanan üç kısmını da dahil edecek yeni bir sıralama yöntemi önermektedirler. Proje değerlerini sıralamak için kullandıkları yöntemlerinde de $\alpha \in [0,1]$ iyimserlik indeksi kullanmaktadırlar.

$\tilde{A}_1 = a_1, b_1, c_1$, $\tilde{A}_2 = a_2, b_2, c_2$, $\tilde{A}_3 = a_3, b_3, c_3$ bulanık sayıları için;

$S = a_1, b_1, c_1, a_2, b_2, c_2, a_3, b_3, c_3$ ve $V \tilde{A}_k$ ise $\tilde{A}_k$ 'nın değeri olsun.

$$V \tilde{A}_k = b_k \left\{ \alpha \left[\frac{c_k - x_{\min}}{x_{\max} - x_{\min} + c_k - b_k} \right] + 1 - \alpha \left[1 - \frac{x_{\max} - a_k}{x_{\max} - x_{\min} + b_k - a_k} \right] \right\} \quad (4.36)$$

$$x_{\min} = \inf S \quad (4.37)$$

$$x_{\max} = \sup S \quad (4.38)$$

4.12.3 Yager sıralama indeksi

Bu bölümde tez çalışması kapsamında kullanılan bulanık sayı sıralama yöntemi hakkında genel bilgiler verilecek olup, yöntemin geniş açıklamaları ve hesaplamaları ilerleyen bölümlerde açıklanacaktır. Bulanık sayılar, doğrusallık ve toplanabilirlik özelliklerine sahiptir ve alan takasının (area compensation) gürbüz olduğuna işaret etmektedir. Alan takası tanımını temel alan Yager (1981), bulanık sayıların sıralanması için bir prosedür önermiştir. Sıralama indeksi $I \tilde{t}$ aşağıdaki formüle göre α -kesim, $\alpha_t = [t_\alpha^L, t_\alpha^U]$ dan konveks bulanık sayı $\tilde{t}$ hesaplanmaktadır.

$$I \tilde{t} = \int_0^1 \frac{1}{2} t_\alpha^L + t_\alpha^U d_\alpha \quad (4.39)$$

$\tilde{t}$ 'nin ortalama değerinin merkezidir. $\tilde{D}_1$ ve $\tilde{D}_2$ iki bulanık sayı olsun. $I \tilde{D}_1 \geq I \tilde{D}_2$ örneği $\tilde{D}_1 \geq \tilde{D}_2$ 'yi ifade etmekte ve $\max \tilde{D}_1, \tilde{D}_2 = \tilde{D}_1$ olmaktadır. Bu, indeks uygulama için çok basittir ve üyelik fonksiyonun yerine α -kesim t_α^L ve t_α^U 'nın uç değerlerinden $\tilde{t}$ konveks bulanık sayısı hesaplanmaktadır. Sıralama için, bulanık sayıların üyelik fonksiyonlarının belirtilmiş biçimlerinin bilinmesi gerekmemektedir. Bunun diğer sıralama yöntemlerinin çoğundan farkı; sıralama yapmak için tüm bulanık sayıların üyelik fonksiyonlarının bilinmesi gerekliliğidir. Bulanık faaliyet sürelerinin üyelik fonksiyonlarının şekli kesin olarak bilinmese bile, Yager'ın sıralama indeksi hala daha uygulanabilir geçerli bir metottur. Yager's yönteminin ayrıntılı tanımları (Fortemps ve Yager 1996)' da bulunabilir.

Yager'ın sıralama yöntemi; doğrusallık ve toplanabilirlik özelliklerine sahiptir ve alan takasını temel alan sıralama tekniklerinden biridir. $\tilde{B}$ ve $\tilde{C}$ konveks bulanık sayılarının doğrusal kombinasyonun $\tilde{A}$ konveks bulanık sayı olduğu düşünüldüğünde, $\tilde{A} = u\tilde{B} + v\tilde{C}$ burada u ve v sabitlerdir. Daha sonra $I \tilde{A} = uI \tilde{B} + vI \tilde{C}$ elde edilmektedir. Sonuç olarak Yager'ın sıralama yönteminin temeli şudur: Bulanık KYM, **problemni, klasik faaliyet süreli geleneksel KYM problemine dönüştürür.**

4.13 Endüstri Mühendisliğinde Bulanık Küme Uygulamaları

Bulanık kümelerle ilgili anlamsal yaklaşımlar genel olarak üç ayrı kategoriye ayrılabilir. Bunlardan ilki; bulanık kümelerin anlamların benzerliğinin ve yakınlığının bir ifadesi olarak kullanılmasıdır. Bu yaklaşım gruplama, tanımlama ve sınıflandırma uygulamalarında kullanılmaktadır. İkinci yaklaşımda ise, bulanık kümeler eksik veya bulanık bilginin olasılık dağılımları şeklinde ifade edilmesinde kullanılmaktadır. Bu yaklaşım, yöneylem araştırmasındaki karar verme problemlerindeki matematiksel modellerde, belirsiz bilgilerin ifade edilmesine olanak vermektedir. Matematiksel modellerde belirsizlik; modeldeki kesin katsayıların yerine, bulanık sayıların veya bulanık oranların kullanılmasıyla modele dahil edilmektedir. Bu uygulama bulanık aritmetik ve bulanık sayılar üzerindeki diğer işlemlerin kullanılmasına imkan vermektedir.

Bulanık kümeler teorisinin uygulamasıyla ilgili üçüncü yaklaşımda ise, esnek bir şekilde belirlenmiş olan kısıtların sağlanması veya hedeflerin gerçekleşmesi için tercihlerin ifadesinde bulanık kümeler kullanılmaktadır. Bu yaklaşım özellikle karar verme sırasındaki matematiksel kesinliği ortadan kaldırarak, yerine gerçek hayattaki gibi bir geçiş fonksiyonu oluşturur. Kısıtların karşılanması, optimizasyon, çok-faktörlü karar analizi gibi tekniklerdeki gerekli olan esneklik, bulanık kümeler kullanımı ile sağlanmaktadır. Özellikle yöneylem araştırması çalışmalarında ikinci ve üçüncü yaklaşımların birlikte kullanılmasını gerektiren pek çok problem mevcuttur. Bulanık kümelerin uygulamaları pek çok değişik disiplinde karşımıza çıkmaktadır.

Endüstri mühendisliği dalında yürütülen bulanık küme uygulamalarını ise dört ana başlık altında toplamak mümkündür.

Bunlardan ilki karar verme uygulamalarıdır. Karar verme uygulamalarında bulanık kümeler değişik şekillerde kullanılmaktadır. Karar verme uygulamalarından birisi tek veya çok ölçütlü karar vermede bulanık tercih modelleridir. Burada klasik yaklaşımdaki kesin ölçütler yerine bulanık ölçütler kullanılarak daha iyi sonuçlar elde edilmesi amaçlanmaktadır. Bu şekilde sonucun daha sağlıklı ve gerçek durumu temsil ediyor olması sağlanmaktadır. Ancak bu iyileşmeye karşın, analiz safhasının daha karmaşık hale gelmesi kaçınılmazdır.

Bir diğer karar verme uygulaması grup karar vermesi sırasında bulanık yöntemler uygulamasıdır. Burada kilit soru bir grup halindeki bireylerin tercihlerinin nasıl en iyi şekilde ifade edilebileceğinin bulunmasıdır. Bulanık oyun teorisi de karar verme uygulamalarının bir parçasını oluşturmaktadır. Oyun teorisiyle bulanık kümeler teorisi birleştirilerek ortaya çıkan bu alanda da çalışmalar sürdürülmektedir.

Bulanık küme uygulamalarının ikinci ana başlığı ise bulanık matematiksel programlamadır. Tek veya çok amaçlı bulanık doğrusal programlama yöntemleri konunun birinci kısmını oluşturmaktadır. Buna ek olarak, doğrusal programlamada kullanılan katsayıların bulanık hale getirilmesiyle ortaya çıkan tekniğe “esnek programlama” adı verilmiştir. Doğrusal programlama tekniğinin gerçek hayatta çok kullanılan tekniklerden birisi olduğu araştırmalar sonucu ortaya çıkmıştır. Bu güçlü tekniğin daha da çok kullanılmasının önündeki engel gerçek hayatta kesin bilgiye ulaşmanın zorluğu ve maliyetidir (Janko 1990). Bulanık değişkenlerin bu alanda kullanılmasıyla doğrusal programlama tekniği çok daha az maliyetle gerçek hayat problemlerinin çözümünde kullanılabilecektir.

Bulanık doğrusal olmayan programlama teknikleri de benzer şekilde kullanılan bir diğer tekniktir. Bulanık doğrusal olmayan programlamada özellikle Pareto en iyi çözümün bulunması üzerine çalışmalar yapılmaktadır. Bu alanda çok amaçlı doğrusal olmayan programlamada karar vericinin bulanık yaklaşımlarının gerçeğe en yakın şekilde modellenmesi hedeflenmektedir.

Bulanık değişkenlerin matematiksel programlamada kullanıldığı bir başka alan da tamsayılı bulanık en iyilemedir. Bunların arasında özellikle proje yönetimi ile de yakından ilgili olarak en iyileme ve çizelgeleme konuları dikkati çekmektedir. KYM ve PERT yöntemlerinin bulanık karşıtları üzerine yapılan çalışmalarda başarılı sonuçlar elde edilmiştir. Ayrıca bu alanda çizelgeleme, atama ve taşıma

problemleriyle ilgili çalışmalar da bulunmaktadır. Bulanık dinamik programlama yöntemleri de çok aşamalı karar verme problemlerinin bulanık programlama yöntemiyle çözümüne katkı sağlayan bir alandır. Bu yöntemde aşamalar arasındaki geçişler ve ulaşılan hedefler bulanık değişkenler kullanılarak modellenmektedir. Bunun sonucu olarak deterministik ve stokastik yaklaşımların ötesinde gerçekçi sonuçlar elde etmek mümkün olmaktadır (Lowen 1993).

Endüstri mühendisliği ile yakından ilgisi bulunan istatistiksel veri analizi yöntemlerinde de bulanık kümeler teorisinin faydalı uygulamaları yapılmıştır. Örneğin rasgele deneylerde bulanık verilerin kullanılmış ve olasılık teorisiyle bulanık kümeler teorisinin bir sentezi çıkarılmıştır. Özellikle parametre tahminleri ve hipotez testlerinde bulanık değişkenler kullanılarak çalışmalar yapılmıştır. Bunlara ek olarak istatistiksel karar verme yöntemlerinde bulanık değişkenler kullanılmış ve bulanık fayda fonksiyonlarının maksimizasyonu konusunda çalışılmıştır. Kullanılan bir diğer istatistiksel yöntem ise bulanık regresyon analizi olmuştur. Elde bulunan veriye bulanık doğrusal bir model uydurulması için iki ayrı yöntem kullanılmıştır. Bunlardan birincisi en küçük hata kareleri yöntemiyle bulanık kümeler arasındaki mesafenin en kısa hale getirilmesi yöntemidir. İkincisi ise olasılık dağılımları kullanarak en iyi doğrusal modelin bulunması yöntemidir (Kacprzyk 1993).

Güvenilirlik, bakım ve yenileme konuları ise bulanık yöntemlerin kullanıldığı dördüncü alandır. Bu alandaki temel yaklaşım sistemdeki bir nesnenin işler durumdan işlemez duruma geçişinin klasik modellerin aksine anlık değil, sürekli olmasıdır. Bu şekildeki bir yaklaşımla sistemlerin güvenilirliği yeniden ele alınmış ve faydalı sonuçlar elde edilmiştir. Bakım ve yenileme konularında da bulanık değişkenlerin ve sayıların kullanılmasıyla gelişmeler kaydedilmiştir. Özellikle makinelerin ekonomik ömürleri bu yeni yaklaşımda incelenmiş ve en iyi yenileme zamanları tespit edilmiştir (Janko 1990).

Genel olarak bulanık değişkenlerin uygulamaları endüstri mühendisliğinin hemen her alanında görülmektedir. Doğada var olan belirsizliğin modellenmesine olasılık teorisinden farklı ve yapıcı bir yaklaşım öngören bulanık kümeler teorisi hakkındaki çalışmalar büyük bir ilgiyle sürdürülmektedir.

5. BULANIK KYM

Bu bölümde, bulanık proje çizelgeleme problemlerinin çözümlerinde kullanılan, bulanık kritik yol analizi incelenmiştir. Bir proje şebekesinde, faaliyet süresi bulanık sayılardan oluşan bir bulanık proje çizelgeleme probleminin kritik yol analizi yapılarak, bulanık aritmetik yaklaşım ile doğrusal programlama yöntemi kıyaslanacaktır. Bu fikir, bulanık sayıları sıralama (ranking method) ve doğrusal programlama formülasyonunu temel almaktadır. Bulanık kritik yol problemi, doğrusallık ve eklenebilirlik (toplanabilirlik) özelliklerini temel alan ve amaç fonksiyonu bulanık katsayılı bir doğrusal programlama modeli olarak formülize edilmiştir. Yager sıralama yöntemi kullanılarak, geleneksel akış çizgisi çözüm yöntemleri kullanılarak çözümlenebilen klasik doğrusal programlama modellerinin, bulanık doğrusal programlama modellerine dönüşümü gösterilecektir. Önerilen yaklaşımın uygulanması oldukça basittir ve bu uygulamayla **bulanık faaliyet süreleri üyelik fonksiyonlarının kesin (açık olarak) bir biçimde bilinmesine ihtiyaç duyulmamaktadır**. Son yıllarda kullanılan ve başarılı sonuçlar veren; ayrıca doğrusal programlama ve bulanık aritmetik çözümler öneren örnek bir bulanık proje çizelgeleme problemi, sayısal analiz bölümünde test edilmektedir. Kritik yol metodu (KYM)'nin amacı, büyük ve kompleks projelerin planlanmasına ve kontrol edilmesine yardımcı olmaktır. Bu yaklaşım, şu gerekçeler için gereklidir: (a) Hangi faaliyetler icra edilecektir? (b) Faaliyetler hangi sırayla icra edilecektir? (c) Kaynak gereksinimleri nelerdir? (d) Her bir faaliyetin zaman gereksinimi nedir? KYM, proje için zaman çizelgesi oluşturmaya ve tasarlamaya yarayan, şebeke temelli bir yöntemdir. İki basit sonucu vardır; projenin tamamlanması için gereken toplam proje süresini ve kritik yolun elde edilmesini sağlamaktadır. Kritik yol; başlangıç olaydan, son olaya kadar tüm olayların bolluklarının tümünün sıfır olduğu yoldur. Kritik yol boyunca, bir faaliyetin süresindeki artış, toplam proje süresini de kesinlikle artırmaktadır.

Program geliştirme ve yeniden gözden geçirme tekniği (PERT); olasılık-yoğunluk fonksiyonunun bilindiği ve rassal değişkenli faaliyetlerin sürelerinin bulunduğu

durumda kullanılmaktadır. Beta, olasılık yoğunluk fonksiyonu olduğu kabul edildiğinde her bir faaliyet için c, a ve b gibi üç parametre gereklidir. Bir faaliyetinin sırasıyla c, en olası; a, en iyimser; ve b, en kötümser sürelerle sahip olduğu durumdur. Faaliyet süresi bu üç parametreden hesaplanmaktadır.

5.1 Bulanık KYM Problemlerinin Doğrusal Programlama Formülasyonu

Bir şebeke; yönlendirilmiş ve birbiriyle bağlantılı düğümlerden oluşan bir proje modeli $G=(N,A)$ notasyonu ile gösterilir. Burada N, n düğümlerin kümesidir ve $i, j \in A$ serimlerin kümesidir. “ T_{ij} ”, $i, j \in A$ faaliyetinin faaliyet süresini gösterir. KYM, proje için zaman çizelgesi oluşturmaya ve tasarlamaya yarayan şebeke temelli bir yöntemdir. Proje şebekelerinin toplam faaliyet süresini ve kritik yollarını, doğrusal programlama ile bulmak etkin yöntemlerden birisidir. KYM, en kısa yol probleminin zıttı olarak düşünüldüğü için; proje şebekesinde kritik yolu belirlemek, başlangıçtan sona kadar en uzun yolun bulunmasıyla sağlanmaktadır. Proje şebekesine giren başlangıç düğümü ve proje şebekesinden ayrılan bitiş düğümüne bir birim akış olduğu varsayılmaktadır. N düğümlü bir KYM’nin doğrusal programlama modeli aşağıdaki gibi ifade edilmektedir.

$$\begin{aligned}
 \max \quad & D = \sum_{i=1}^n \sum_{j=1}^n T_{ij} x_{ij} \\
 \text{s.t.} \quad & \sum_{j=1}^n x_{1j} = 1, \\
 & \sum_{j=1}^n x_{ij} = \sum_{k=1}^n x_{ki}, \quad i=2,\dots,n-1 \\
 & \sum_{k=1}^n x_{kn} = 1, \\
 & x_{ij} \geq 0, \quad i, j \in A \text{ ve } i < j
 \end{aligned} \tag{5.1}$$

Burada “ x_{ij} ”, $i, j \in A$ faaliyetinde akış miktarını gösteren karar değişkenidir ve kısıtlar, her bir düğümde akışın korunduğunu göstermektedir. Belirtilen bu akış, proje şebekesinde ya yaratılmakta; ya da yok edilmektedir. En kısa yol problemi olarak Model (5.1) için, her bir uygun çözümde tüm temel uygun değişkenler tamsayıdır (binarydir). Proje şebekesi için kritik yol Model (5.1) için, optimal çözümde $x_{ij}^* = 1$ optimal karar değişkenlerini kapsayan yollardaki her bir faaliyetin

başlangıçtan sona kadar $i, j \in A$ faaliyetini kapsamaktadır. Projenin tamamlanması için gereken toplam faaliyet süresi, Model (5.1)'in maksimum amaç değeri olan D olarak verilmektedir. T_{ij} faaliyet süreleri olduğu varsayıldığında, $i, j \in A$ tam ve kesin değildir ve $\tilde{T}_{ij}$, $i, j \in A$ bulanık sayılar olarak düşünülmektedir. Bulanık KYM probleminin doğrusal programlama formülasyonu aşağıdaki gibidir:

$$\begin{aligned}
\max \quad & \tilde{D} = \sum_{i=1}^n \sum_{j=1}^n \tilde{T}_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_{j=1}^n x_{1j} = 1, \\
& \sum_{j=1}^n x_{ij} = \sum_{k=1}^n x_{ki}, \quad i=2, \dots, n-1 \\
& \sum_{k=1}^n x_{kn} = 1, \\
& x_{ij} \geq 0, \quad i, j \in A,
\end{aligned} \tag{5.2}$$

Dikkat edilirse; toplam faaliyet süresi olan $\tilde{D}$, klasik bir sayı yerine, bulanık bir sayı haline gelmektedir. Sonuç olarak Model (5.2) direkt olarak çözülememektedir.

5.2 TDP Çözüm Prosedürü

B-KYM problemlerinin çözümüyle ilgili olan yaklaşımlardan biri de, bulanık sayıların klasik sayıya dönüştürülerek doğruluğunun kanıtlanmasıdır. Örneğin Buckley ve Feuring (1989), bulanık doğrusal programlama probleminin amaç fonksiyonunu, klasik çok amaçlı programlama problemine dönüştürmeyi temel alan, doğrusal programlama problemini tamamen bulanıklaştırmak için bir yaklaşım önermişlerdir. Ayrıca bu tür bir klasik dönüştürmenin doğruluğunu kanıtlamaya çalışmışlardır. Durulaştırma konseptini temel alan bu yaklaşım, Model (5.2)'yi amaç fonksiyonundaki bulanık sayıları durulaştırarak, bulanık sıralama yönteminin kullanımını basitleştirmekte ve bunları geleneksel klasik doğrusal programlama solverları ile hala çalışılabilir şekilde klasik modele dönüştürmektedir. Bu fikir, tüm olası uygun çözümlerin ve olası yolların hesaplanabilmesine yardım etmektedir. Model (5.2)'de, kısıtları tanımlamakta ve tüm bulanık amaç fonksiyonu değerlerini hesaplayarak kritik yolları basitçe bulmaktadır. En büyük amaç fonksiyonu değerli yol, kritik yol olarak belirlenmektedir. Ancak tüm olası yolların uzunluklarının

kıyaslanması, özellikle de olabilirli bir çok bulanık yol büyük bir proje şebekesi için, yüksek hızlı bilgisayarlarda elverişsizdir. Maalesef ki klasik bir çevrede KYM problemi gibi, bulanık bir çevrede kritik yol yerine tüm yolları bulmak istense de, tüm bulanık yolları bulmak gereksizdir ve hantal (kullanışsız ağır) bir görevdir. Bu nedenle bulanık bir ortamda kritik yolu bulmak için model (5.2)'yi klasik doğrusal programlamaya dönüştürerek çözmek etkili bir yol olacaktır. Burada model (5.2)'nin bulanık amaç fonksiyonu değeri, bulanık sayı sıralama yöntemiyle klasik bir modele durulaştırılır. Bulanık sayı sıralama yöntemlerinin popüler yaklaşımlarından birisi de Yager's sıralama yöntemidir. Proje şebekesinin yol uzunluklarını gösteren sıralanmış amaç fonksiyonu değerleri için bu yöntem uyarlanmıştır.

5.2.1 Klasik dönüşüm

M yollu model (5.2) olarak formüle edilen kritik yol problemini düşünelim $p_k, k=1,2,3,...,m$. k 'nıncı yoluna karşılık gelen k 'nıncı bulanık sayı $x_{ij}^k, i, j \in A$ olmak üzere : $\tilde{D}^k = \sum_{i=1}^n \sum_{j=1}^n \tilde{T}_{ij} x_{ij}^k$ k 'nıncı yolun bulanık toplam faaliyet süresidir. m yolundan en büyük toplam faaliyet süreli $\tilde{D}^* = \max \tilde{D}^k, k=1,2,...,m$ olanlarından biriside bulanık kritik yol, en büyük toplam faaliyet süresine sahip en kritik yol olarak belirlenmektedir. Yager'in sıralama yönteminin özelliklerine göre; $\tilde{D}^*$ 'yi bulmak için bu yöntem uygulanırsa en büyük (geniş) Yager sıralama indeksi $I \tilde{D} = \max I \tilde{D}^k, k=1,2,...,m$ bulmakta etkindir. Bundan başka Yager'in yönteminin doğrusallık ve toplanabilirlik özelliklerine sahip olmasının sonucu olarak aşağıdaki şu ifadeler, elde edilmektedir:

$$I \tilde{D}^k = I \left(\sum_{i=1}^n \sum_{j=1}^n \tilde{T}_{ij} x_{ij}^k \right) = \sum_{i=1}^n \sum_{j=1}^n I \tilde{T}_{ij} x_{ij}^k \quad (5.3)$$

İşte bu $\tilde{D}^*$ maksimum bulanık amaç fonksiyonu değeri, maksimum sıralama indeksi $I \tilde{D}^* = \max_k \left\{ \sum_{i=1}^n \sum_{j=1}^n I \tilde{T}_{ij} x_{ij}^k \right\}$ 'e karşılık gelir. Sonuç olarak; bulanık faaliyet süreli kritik yol problemi aşağıdaki gibi formulize edilmektedir:

$$\begin{aligned}
I \tilde{D}^* &= \max \left\{ \sum_{i=1}^n \sum_{j=1}^n I \tilde{T}_{ij} x_{ij} \right\} \\
\text{s.t.} \quad & \sum_{j=1}^n x_{1j} = 1, \\
& \sum_{j=1}^n x_{ij} = \sum_{k=1}^n x_{ki}, \quad i = 2, \dots, n-1, \\
& \sum_{k=1}^n x_{kn} = 1, \\
& x_{ij} \geq 0, \quad (i, j) \in A.
\end{aligned} \tag{5.4}$$

Bu problem, geleneksel doğrusal programlamanın temelidir. $I \tilde{T}_{ij}$, $i, j \in A$ amaç fonksiyonundaki katsayılar klasik gerçel sayılar yerine, bulanık sayılardır. Bir çok uzman görüşüne göre bu tür geleneksel doğrusal programlama çözümleri, model (5.2) için etkin bir yaklaşımdır. Optimal temel uygun çözümün x_{ij}^* belirlenmesiyle, Model (5.4)'ün $i, j \in A$ en kritik yol p^* ile belirlenebilmekte ve bulanık toplam faaliyet süresi $\tilde{D}^* = \sum_{i=1}^n \sum_{j=1}^n \tilde{T}_{ij} x_{ij}^*$ olarak hesaplanabilmektedir. Dikkat edilmelidir ki; bu klasik dönüşüm birleştirme, doğrusallık ve toplanabilirlik özelliklerini işleten Yager yönteminin kullanımıyla, bulanık faaliyet zamanlarının klasik zaman dönüşümünü temel alan bu klasik dönüşümün, Model (5.4)'ün doğruluğu kanıtlanabilmektedir. Bir sonraki bölümde bulanık KYM probleminin dual formülasyonu, bu klasik dönüşümün doğruluğunu kanıtlamak için kullanılacaktır.

5.2.2 Dual doğrulama sağlama

İyi bilinen doğrusal programlama modelinin dualite teoremine göre; primal ve dual modelleri aynı amaç fonksiyonu değerine sahiptir. Böylece yukarıdaki klasik dönüşümün geçerliliğini göstermenin bir yolu da, Model (5.1)'in dualinin formulüze edilmesidir.

$$\begin{aligned}
& \min y_n - y_1 \\
\text{s.t.} \quad & y_j - y_i \geq T_{ij}, \quad i, j \in A, \\
& y_i, y_j \text{ işareti belirlenmemişse, } \forall i, j \in A,
\end{aligned} \tag{5.5}$$

Burada y_i ve y_j karar değişkenleri sırasıyla i ve j düğümlerinin oluş zamanını göstermektedir. Her bir kısıt, bir faaliyetin karakteristiklerini farklı faaliyetler arasındaki öncelik ilişkileriyle birleştirdiğinde $y_j - y_i \geq T_{ij}$ 'nin kısıtı $y_i + T_{ij}$ zamanından önceki olamayan j düğümü için en erken oluş zamanını göstermektedir. Amaç, tüm öncelik ilişkilerini karşılayacak (sağlayacak) en kısa zaman çevrimini bulmaktır. Faaliyet süreleri bulanık sayılardan oluşuyorsa Model (5.5) aşağıdaki şekilde ifade edilmektedir:

$$\begin{aligned} \min & y_n - y_1 \\ \text{s.t.} & y_j - y_i \geq \tilde{T}_{ij}, \quad i, j \in A, \\ & y_i, y_j \text{ i\ddot{s}areti belirlenmemi\ddot{s}e,} \quad \forall i, j \in A, \end{aligned} \quad (5.6)$$

Kısıtların sağı yan deęerleri, bulanık sayılardan oluřan bir doęrusal programlama modelidir. Model (5.6)'yi ele almanın bir yolu da, yukarıdaki ifadeye klasik dönüşüm uygulamaktır. Uygulama sonucu, aşağıdaki şekilde gösterilmektedir:

$$\begin{aligned} \min & y_n - y_1 \\ \text{s.t.} & y_j - y_i \geq I \tilde{T}_{ij}, \quad i, j \in A, \\ & y_i, y_j \text{ i\ddot{s}areti belirlenmemi\ddot{s}e,} \quad \forall i, j \in A, \end{aligned} \quad (5.7)$$

Model (5.7)'in duali, model (5.4)'ünkiyle kesinlikle aynıdır; böylelikle önceki bölümdeki klasik dönüşümün doęruluęu kanıtlanmış olmaktadır.

5.2.3 Kritik yolun belirlenmesi

Model (5.5) için temel uygun çözüm düşünöldüğünde; herhangi bir temel uygun çözüm, proje řebekesinde bir yolda yerine konulmaktadır. En kritik yol, kritiklięinin yol derecesi 1.0 olarak ayarlandığında, $\deg_{Cr}^R p^* = 1$ ile gösterilmektedir. $p_k, k=1, 2, \dots, m$. k 'nıncı yol için, kritik yolun izafı derecesi aşağıdaki gibi tanımlanabilir:

$$\deg_{Cr}^R p_k = \frac{\sum_{i=1}^n \sum_{j=1}^n I \tilde{T}_{ij} x_{ij}^k}{\sum_{i=1}^n \sum_{j=1}^n I \tilde{T}_{ij} x_{ij}^*}, \quad k=1, 2, \dots, m. \quad (5.8)$$

5.3 Bulanık Aritmetik

Bu bölümde bir proje şebekesinde A_{ij} faaliyetinin bulanık faaliyet süresi, FAT_{ij} olarak gösterilmiştir. $FAT_{ij} = a_{ij}, b_{ij}, c_{ij}, d_{ij}$ ile temsil edilen yamuk bulanık sayılar, A_{ij} faaliyeti için sırasıyla minimum ve maksimum değerleri a_{ij}, d_{ij} ve birinci çeyrek değeri ile ikinci çeyrek değeri b_{ij} ve c_{ij} 'dir. Eğer bunlar istatistiki verilere dayalı ise; $a_{ij}, b_{ij}, c_{ij}, d_{ij}$, minimumdan maksimuma doğru sıralanmalıdır. Örneğin, faaliyetin 4 istatistiki verisi 6,9,3 ve 8 ise; yamuk bulanık sayısı (3, 6, 8, ve 9) olarak oluşturulmaktadır. Bunun tersine A_{ij} faaliyeti hakkında hiç bir bilgi bulunmuyor ise; $FAT_{ij} = a_{ij}, b_{ij}, c_{ij}, d_{ij}$ bulanık faaliyet zamanı bir uzmanın bilgi, deneyim ve subjektif yargılarıyla oluşturulabilmektedir. Herhangi iki bulanık faaliyet süresinin genişletilmiş aritmetik işlemleri, sırasıyla toplama ($\oplus$) ve çıkarma ($\ominus$) işlemleri şu şekilde ifade edilebilir;

$$FAT_1 \oplus FAT_2 = a_1, b_1, c_1, d_1 \oplus a_2, b_2, c_2, d_2 = a_1 + a_2, b_1 + b_2, c_1 + c_2, d_1 + d_2 \quad (5.9)$$

$$FAT_1 \ominus FAT_2 = a_1, b_1, c_1, d_1 \ominus a_2, b_2, c_2, d_2 = a_1 - d_2, b_1 - c_2, c_1 - b_2, d_1 - a_2 \quad (5.10)$$

5.3.1 Yamuk bulanık sayıların sıralanması

Sıralama yöntemleri, bulanık KYM'de temel zorluktur. Bulanık sayıları sıralamak için bir çok yöntem önerilmiştir. Bu çalışmada uygulama kolaylığı için kullanışlı bir sıralama yöntemi, bulanık yol analizinde, mevcut sıralama sorunlarını çözmek için bu kısımda açıklanacaktır. A_{ij} faaliyeti için $FAT_{ij} = a_{ij}, b_{ij}, c_{ij}, d_{ij}$ bulanık faaliyet süresi olsun. Karar vericinin risk tutum indeksi β , şu şekilde ifade edilebilir:

$$\beta = \left[\sum_i \sum_j \frac{b_{ij} - a_{ij}}{b_{ij} - a_{ij} + d_{ij} - c_{ij}} \right] / t \quad (5.11)$$

$f_{A_i} x$ üyelik fonksiyonlu A_{ij} bulanık sayısını $m_i = \min x | f_{A_i} x = 1 + \max x | f_{A_i} x = 1$. tanımladığımızda, aşağıdaki kurallara göre $A_i = a_1, b_1, c_1, d_1$ ve $A_j = a_2, b_2, c_2, d_2$ olmak üzere n adet bulanık sayıyı sıralayabiliriz. Proje yönetiminde, aktivitelerin kritikliğini risk faktörü olarak değerlendiren bulanık sayılar kullanılarak, riskli aktivitelerin

belirlenmesi ve bunlara göre genel proje riskinin ölçülmesi için, bir metod önerilmiştir. Proje yönetiminde risk kavramının değerlendirilmesinde, değişik yöntemler kullanılmaktadır. Bu çalışmada, bir aktivitenin kritikliğinin, o aktivitenin riskliliğini gösterdiği kabul edilmektedir. Benzer şekilde, projenin genel riskliliğinin hesaplanması için, proje ağında bulunan riskli (kritik) aktivitelerin sayısı, bunların risk dereceleri ve proje ağındaki yerleri esas alınmaktadır.

$$A_i > A_j \Leftrightarrow R_{A_i} > R_{A_j}, \text{ veya,} \quad (5.12)$$

$$R_{A_i} = R_{A_j} \text{ ve } m_i > m_j, \quad (5.13)$$

$$A_i = A_j \Leftrightarrow R_{A_i} = R_{A_j} \text{ ve } m_i = m_j \quad (5.14)$$

Buna göre, A_i yamuk bulanık sayısının R_{A_i} sıralama değeri aşağıdaki gibi elde edilebilir:

$$R_{A_i} = \beta \left[\frac{d_i - x_1}{x_2 - x_1 - c_i + d_i} \right] + 1 - \beta \left[1 - \frac{x_2 - a_i}{x_2 - x_1 + b_i - a_i} \right] \quad (5.15)$$

Burada β karar verici risk tutum indeksi denk. (5.11) ve denk. (5.15)'de $x_1 = \min a_1, a_2, \dots, a_n$ ve $x_2 = \max d_1, d_2, \dots, d_n$ kullanılarak ve β değeri alınarak, n adet yamuk bulanık sayının sıralama değerleri, kolayca hesaplanabilmektedir. Yukarıda açıklanan sıralama kuralına dayalı olarak, n adet yamuk bulanık sayısının sıralaması etkili bir şekilde tespit edilmektedir.

5.3.2 Bulanık kritik yol yöntemi (B-KYM)

- N Bir proje şebekesindeki bütün serimlerin kümesi.
- A_{ij} i ve j faaliyetleri arasındaki faaliyet (AOA'a göre).
- A_i i faaliyeti (AON'e göre).
- A_j j faaliyeti (AON'e göre).
- FAT_{ij} A_{ij} 'nin bulanık faaliyet süresi.

$FEST_j$	j seriminin en erken başlangıç bulanık süresi.
$FECT_j$	j seriminin en erken tamamlanma bulanık süresi.
$FLST_j$	j seriminin en geç başlangıç bulanık süresi.
$FLCT_j$	j seriminin en geç tamamlanma bulanık süresi.
FTS_{ij}	A_{ij} 'nın toplam aylak bulanık süre.
S_j	j seriminin tüm ardıl faaliyetlerinin kümesi.
NS_j	j seriminin tüm ardıl faaliyetleriyle bağlantılı tüm düğümlerin kümesi. Örneğin, $NS_j = \{k A_{jk} \in S_j, k \in N\}$.
P_j	j seriminin tüm öncellerinin kümesi.
NP_j	j seriminin tüm öncül faaliyetleriyle bağlantılı tüm düğümlerin kümesi. Örneğin, $NP_j = \{i A_{ij} \in P_j, i \in N\}$.
PT_i	i 'ninci yol.
PT	Bir proje şebekesinin tüm yollarının kümesi.
$FCPM_{P_k}$	Bir proje şebekesinde P_k yolunun toplam aylak bulanık süresi.

Burada, B-KYM'de kullanılan önemli özellikler ve teorem kısaca tanıtılacaktır. Başlangıç için, başlangıç serimi $FEST_1 = FECT_1 = 0,0,0,0$ sıfır olarak belirlenir. B-KYM'nun özellikleri aşağıda sunulmuştur :

Özelik 1

1.a.) $FEST_j = \max_{i \in NP_j, j \neq 1, j \in N} FEST_i \oplus FAT_{ij}$

1.b.) $FECT_j = FEST_i \oplus FAT_{ij}$

Özelik 2

2.a.) $FLCT_j = \min_{k \in NS_j, j \neq n, j \in N} FLCT_k \ominus FAT_{jk}$

$$2.b.) FLST_j = FLCT_k \ominus FAT_{jk}$$

Özelik 3 : $FTS_{ij} = FLCT_j \ominus FEST_i \oplus FAT_{ij}$, $1 \leq i < j \leq n$; $i, j \in N$

$$\text{Özelik 4 : } FCPM_{P_K} = \sum_{\substack{1 \leq i < j \leq n \\ i, j \in P_K}} FTS_{ij}, P_K \in PT$$

Bir proje şebekesinde, P_C yolu; $FCPM_{P_C} = \min FCPM_{PT_i} | PT_i \in PT$ bulanık kritik yoldur. Teoremin aşağıdaki şekilde doğru olduğu belirtilmektedir.

Teorem : Bir proje şebekesinde tüm faaliyetlerin yamuk bulanık sayılardan oluşan bulanık faaliyetlere sahip olduğunu varsayarsak, bu şebekede bir bulanık kritik yol olduğunu söyleyebiliriz.

5.3.3 Bulanık kritik yol yöntemi algoritması

Bu bölümde, bulanık bir kritik yol algoritması, bulanık bir ortamda bir proje şebekesinin kritik yolunu bulmak için kullanılmaktadır. Algoritmanın açıklamaları aşağıda sunulmuştur.

1. Proje faaliyetlerinin tanımlanması.
2. Tüm faaliyetlerin öncelik ilişkilerinin belirlenmesi.
3. Her bir faaliyetin bulanık faaliyet süresinin belirlenmesi
4. Proje şebekesinin oluşturulması.

5.a.) Başlangıç (1) seriminin en erken başlangıç bulanık süresinin $FEST_1 = 0,0,0,0$ olarak alındığında ve özellik 1.a kullanılarak $FEST_j$, $j = 2, 3, \dots, n$ hesaplanması.

5.b.) Başlangıç (1) seriminin en erken tamamlanma bulanık süresinin $FECT_1 = 0,0,0,0$ olarak alınarak ve özellik 1.a kullanılarak $FECT_j$, $j = 2, 3, \dots, n$, hesaplanması.

6.a.) Son faaliyet (n) olmak üzere, $FLCT_n = FEST_n$ olarak eşitlenmesi. Ve özellik 2.a kullanılarak $FLCT_j$, $j = n - 1, n - 2, \dots, 2, 1$. hesaplanması.

6.b.) Son faaliyet (n) olmak üzere, $FLST_n = FECT_n$ ve , özellik 2.b kullanılarak $FLST_j$, $j = n - 1, n - 2, \dots, 2, 1$, hesaplanması.

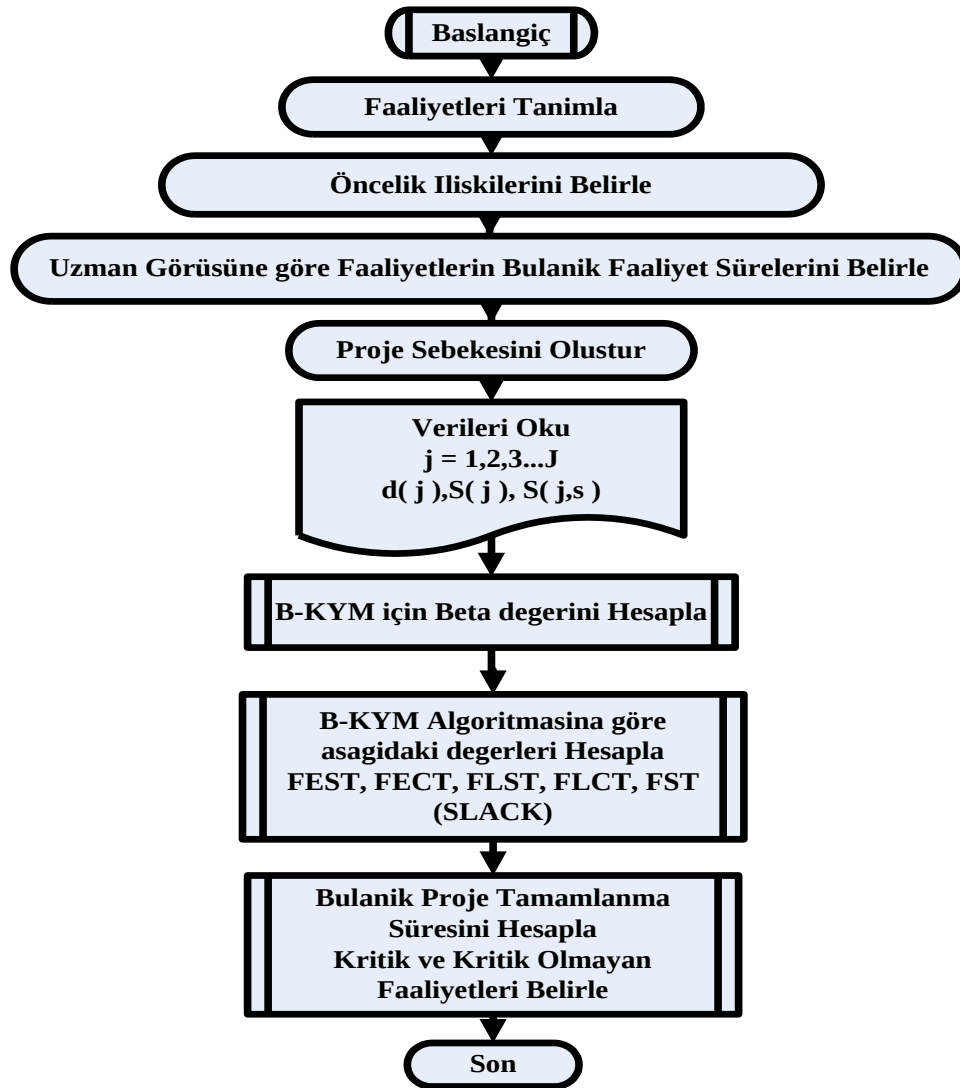
7. Bir proje şebekesinde özellik 3 kullanılarak her bir faaliyetin sırasıyla toplam aylık bulanık sürelerinin FTS_{ij} değerinin hesaplanması.

8. Olası tüm yolların bulunması ve özellik 4 kullanılarak $FCPM P_k$ değerinin hesaplanması.

9. Teorem kullanılarak bulanık kritik yolun bulunması.

10. Üyelik derecesi bulunarak proje çizelgelemenin zamanında tamamlanması.

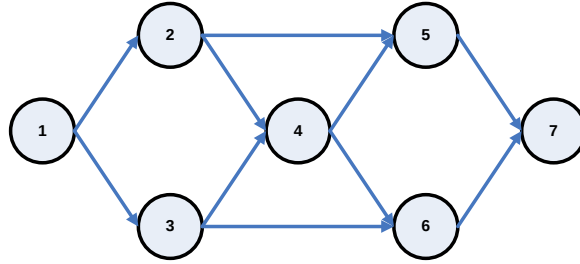
Belirtilen bulanık KYM algoritmasına ait akış diyagramı Şekil 5.1’de sunulmuştur.



Şekil 5.1 : Bulanık KYM Algoritması.

5.4 Uygulamalı Sayısal Örnek 1

Chang ve Lee (1999)'nin yapmış oldukları çalışmayı kullandık. Şekil 5.2'de düğüm 1 den düğüm 7 arasında kritik yolu bulmaya çalışmışlardır. Tüm faaliyetler, üçgensel bulanık sayılardan oluşsa da biz bu çalışmaya yamuk bulanık sayılar olarak uyarlanmıştır.



Şekil 5.2 : Örnek Proje Şebekesi (Chanas ve Zelinski 2001).

Çizelge 5.1 : Örnek Projenin Bulanık Faaliyet Süreleri.

	Faaliyetler A_{ij}	Bulanık Faaliyet Süresi $\tilde{T}_{ij}$		Faaliyetler A_{ij}	Bulanık Faaliyet Süresi $\tilde{T}_{ij}$
B		(0,0,0,0)	6	a_{36}	(85,95,95,103)
1	a_{12}	(45,55,55,58)	7	a_{45}	(88,95,95,105)
2	a_{13}	(52,55,55,65)	8	a_{46}	(107,115,115,120)
3	a_{25}	(90,100,100,112)	9	a_{57}	(113,120,120,140)
4	a_{24}	(55,70,70,75)	10	a_{67}	(95,100,100,105)
5	a_{34}	(62,65,65,75)	S		(0,0,0,0)

Şekil 5.2'deki proje şebekesi üzerinde Çizelge 5.1'de verilen bulanık faaliyet süreli düğüm 1'den düğüm 7'ye kadar en kritik yolun bulunma problemi TDP ile bulanık aritmetik yöntemle karşılaştırmalı çözümü gösterilmiştir.

Sayısal örneğin TDP çözümü

Model (5.2)'ye göre bu problem formülize edilirse;

$$\max \tilde{T}_{12}x_{12} + \tilde{T}_{13}x_{13} + \tilde{T}_{24}x_{24} + \tilde{T}_{25}x_{25} + \tilde{T}_{34}x_{34} + \tilde{T}_{36}x_{36} + \tilde{T}_{45}x_{45} + \tilde{T}_{46}x_{46} + \tilde{T}_{57}x_{57} + \tilde{T}_{67}x_{67}$$

$$\begin{aligned}
x_{12} + x_{13} &= 1, & x_{12} &= x_{24} + x_{25}, \\
x_{13} &= x_{34} + x_{36}, & x_{24} + x_{34} &= x_{45} + x_{46}, \\
x_{25} + x_{45} &= x_{57}, & x_{36} + x_{46} &= x_{67}, \\
x_{57} + x_{67} &= 1, & x_{12}, x_{13}, x_{24}, x_{25}, x_{34}, x_{35}, x_{45}, x_{46}, x_{57}, x_{67} &\geq 0,
\end{aligned}$$

Model (5.4)'ü temele alan birleştirilmiş matematik programlama modeli ise aşağıdaki gibidir :

$$\max I \tilde{T}_{12} x_{12} + I \tilde{T}_{13} x_{13} + I \tilde{T}_{24} x_{24} + I \tilde{T}_{25} x_{25} + I \tilde{T}_{34} x_{34} + I \tilde{T}_{36} x_{36} + I \tilde{T}_{45} x_{45}$$

+ I $\tilde{T}_{46} x_{46} + I \tilde{T}_{57} x_{57} + I \tilde{T}_{67} x_{67}$ Kısıtlar ise aynıdır. Görüleceği gibi, belirlenmiş olan Yager yönteminin kullanımı için T_{ij} bulanık faaliyet süreleri için sıralama indekslerinin hesaplanması gerekmektedir. Öncelikle T_{ij} 'nin α -kesim'ini bulmak zorundayız. Bu çalışmada önerilen yöntem uygulanarak bulanık faaliyet süreleri için sıralama indeksi hesaplandığında; T_{ij} için Yager'in sıralama indeksleri şu şekilde hesaplanmıştır :

$$\begin{aligned}
I \tilde{T}_{12} &= 53.25, I \tilde{T}_{13} = 56.75, I \tilde{T}_{24} = 67.5, I \tilde{T}_{25} = 100.5, I \tilde{T}_{34} = 66.75, I \tilde{T}_{36} = 94.5, \\
I \tilde{T}_{45} &= 95.75, I \tilde{T}_{46} = 114.25, I \tilde{T}_{57} = 123.25, I \tilde{T}_{67} = 100.
\end{aligned}$$

Model (5.3) sonuçları geleneksel kritik yol probleminde bu değerler yerine konulduğunda, çözüme ulaşmak kolaydır. Matematiksel programlama çözücüsü olan “Lindo”, bu tür doğrusal programlama problemlerini çözmekte kullanılmıştır. Optimal çözüm karar değişkenlerinin $x_{13} = x_{34} = x_{45} = x_{57} = 1$ ve $x_{12} = x_{24} = x_{25} = x_{36} = x_{46} = x_{67} = 0$ aldığı bu sonuçlara göre $I \tilde{D}^* = 315,335,335,385$ olarak bulunur. Bu yolun kritik yol olduğunu doğrulamak için, diğer Yager sıralama indeksleriyle kıyaslamak için tüm yolları buluruz. Düğüm 1'den başlayan ve düğüm 9'a kadar giden yalnızca 6 yol vardır. Bu yollar Çizelge 5.2'de gösterilmiştir. Sıralama indeksleri $I \tilde{D}_{pi}, i=1,2,...,6$. Yager yaklaşımı kullanılarak hesaplanır sırasıyla; 123.25, 219, 0, 56.75, 342.5 ve 123.5 bu 6 sıralama indekslerinin maksimumu $I \tilde{D}_{p1} = 342.5$ ve $p_5 = 1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 7$ olarak gösterilen gerçek kritik yollardır ve önerilen yaklaşımın çözümüyle aynıdır. Denklem (2)'ye uygulandığında en kritik yolun kritikliğinin izafi yol derecesi $p_5 = 1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 7$ 1'e eşit olmalıdır. En kritik yol ise $p^* = 1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 7$ olarak bulunur. Bu çözüm Chanas

ve Lee'nin bulduğu sonuçla aynıdır. Diğer yolların kritikliğinin izafi yol dereceleri Çizelge 5.2'deki gibi hesaplanabilir.

Çizelge 5.2 : TDP ile Çözüm Yönteminin Kritik Yol Hesabı.

Yollar	P_k Değer
$p_5 = 1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 7$	1
$p_2 = 1 \rightarrow 2 \rightarrow 4 \rightarrow 5 \rightarrow 7$	0.639
$p_6 = 1 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 7$	0.361
$p_1 = 1 \rightarrow 2 \rightarrow 5 \rightarrow 7$	0.359
$p_4 = 1 \rightarrow 3 \rightarrow 6 \rightarrow 7$	0.1657
$p_3 = 1 \rightarrow 2 \rightarrow 4 \rightarrow 6 \rightarrow 7$	0

Sayısal örneğin bulanık aritmetik çözümü

Buraya kadar bulanık faaliyet süreli kritik yol probleminin doğrusal programlama ile nasıl çözümlendiği gösterilmiştir. Bundan sonraki kısımda, önerilen bulanık aritmetik yöntem ile çözümü gösterilecektir. Ancak tezdeki sınırlı alan nedeniyle tüm sayısal işlemleri burada göstermemiz mümkün değildir. Bu sebeple her bir adıma ait bir örnek ile çözüm gösterilecek ve sonuçları sunulacaktır. Öncelikle denklem 5.11'e göre toplam risk indeksinin hesaplanması gerekmektedir. Toplam risk indeksi, $\beta = 0.47773$ olarak bulunur. Sonuçlar Çizelge 5.2'de azalan sırada listelenmiştir.

Adım 1. $FEST_1 = 0,0,0,0$ ve özellik 1.a kullanılarak

$$FEST_1 = FEST_B = FEST_2 = 0,0,0,0$$

$$FEST_3 = FEST_1 \oplus FAT_{12} = 0,0,0,0 \oplus 45,55,55,58 = 45,55,55,58$$

$$FEST_7 = \max \{ FEST_4 \oplus FAT_{24}, FEST_3 \oplus FAT_{34} \}$$

$$= \max [45,55,55,58 \oplus 55,70,70,75], [52,55,55,65 \oplus 62,65,65,75]$$

$$FEST_7 = \max \{ 00,125,125,133, (14,120,120,140) \}$$

$$\max \{ 100,125,125,133, 114,120,120,140 \} \quad x_1 = 100, x_2 = 140$$

$$R_{100,125,125,133} = 0.529 \quad R_{114,120,120,140} = 0.546$$

Adım 1 Özelik 1.b

$$\begin{aligned}
 FECT_B &= FEST_B \oplus FAT_S = (0, 0, 0, 0) \oplus (0, 0, 0, 0) = (0, 0, 0, 0) \\
 FECT_1 &= FEST_1 \oplus FAT_1 = (0, 0, 0, 0) \oplus 45, 55, 55, 58 = 45, 55, 55, 58 \\
 FECT_2 &= FEST_2 \oplus FAT_2 = (0, 0, 0, 0) \oplus 52, 55, 55, 65 = 52, 55, 55, 65 \\
 FECT_3 &= FEST_3 \oplus FAT_3 = 45, 55, 55, 58 \oplus 90, 100, 100, 112 = (135, 155, 155, 170)
 \end{aligned}$$

Adım 2.a $FLCT_n = FECT_n$ ve $FLCT_j$ hesaplanır, $j = n-1, n-2, \dots, 2, 1$.

$$\begin{aligned}
 FLCT_S &= FECT_S = 315, 335, 335, 385 \\
 FLCT_{10} &= FLCT_S \ominus FAT_S = 315, 335, 335, 385 \ominus 0, 0, 0, 0 = 315, 335, 335, 385 \\
 FLCT_5 &= \min FLCT_7 \ominus FAT_7, FLCT_8 \ominus FAT_8 \\
 FLCT_5 &= \min [175, 215, 215, 272 \ominus 88, 95, 95, 105], [210, 235, 235, 290 \ominus 107, 115, 115, 120] \\
 FLCT_5 &= \min \{ (0, 120, 120, 184) \ominus (0, 120, 120, 183) \} \ominus (0, 120, 120, 184) \\
 \min [70, 120, 120, 184], [90, 120, 120, 183] \quad x_1 = 70, x_2 = 184 \\
 R \quad 70, 120, 120, 184 &= 0.4987 \quad R \quad 90, 120, 120, 183 = 0.5154
 \end{aligned}$$

Adım 2.b. $FLST_S = FLCT_S \ominus FAT_S = 315, 335, 335, 385 \ominus 0, 0, 0, 0 = 315, 335, 335, 385$

Adım 3. $FTS_S = FLCT_S \ominus FECT_S = 315, 335, 335, 385 \ominus 315, 335, 335, 385 = -70, 0, 0, 70$

Buraya kadar yapılan hesaplama sonuçları Çizelge 5.3’de sunulmuştur.

Çizelge 5.3 : Bulanık Kritik Yol Analizi Sonuç Tablosu.

		FEST				FECT				FLST				FLCT				FTS			
B	B	0	0	0	0	0	0	0	0	-70	0	0	70	-70	0	0	70	-70	0	0	70
1	1-2	0	0	0	0	45	55	55	58	-63	0	0	77	-5	55	55	122	-63	0	0	77
2	1-3	0	0	0	0	52	55	55	65	-70	0	0	70	-5	55	55	122	-70	0	0	70
3	2-5	45	55	55	58	135	155	155	170	63	115	115	182	175	215	215	272	5	60	60	137
4	2-4	45	55	55	58	100	125	125	133	-5	50	50	129	70	120	120	184	-63	-5	-5	84
5	3-4	52	55	55	65	114	120	120	140	-5	55	55	122	70	120	120	184	-70	0	0	70
6	3-6	52	55	55	65	137	150	150	168	107	140	140	205	210	235	235	290	42	85	85	153
7	4-5	114	120	120	140	202	215	215	245	70	120	120	184	175	215	215	272	-70	0	0	70
8	4-6	114	120	120	140	221	235	235	260	90	120	120	183	210	235	235	290	-50	0	0	69
9	5-7	202	215	215	245	315	335	335	385	175	215	215	272	315	335	335	385	-70	0	0	70
10	6-7	221	235	235	260	316	335	335	365	210	235	235	290	315	335	335	385	-50	0	0	69
S	S	315	335	335	385	315	335	335	385	315	335	335	385	315	335	335	385	-70	0	0	70

Adım 4.

$$PT = \min \left\{ \begin{array}{l} B,1,3,6,7,S \text{ , } B,1,3,4,6,7,S \text{ , } B,1,3,5,6,7,S \\ B,1,2,5,7,S \text{ , } B,1,2,4,6,7,S \text{ , } B,1,3,5,6,7,S \end{array} \right\} \quad PT_1 = B,1,3,6,7,S$$

$$FCPM \ P_1 = FTS_B \oplus FTS_{13} \oplus FTS_{36} \oplus FTS_{67} \oplus FTS_S$$

$$= -70,0,0,70 \oplus (-70,0,0,70) \oplus (42,85,85,153) \oplus (-70,0,0,70) = (-218,85,85,432)$$

$$R(FCPM \ P_1) = R(-218,85,85,432) = 0,595301351$$

Olası tüm yollar bulunur ve özellik 4 kullanılarak $FCPM \ P_k$ değeri Çizelge 5.3'deki gibi hesaplanır. $x_1 = -420, x_2 = 441$ ise $R(-268,60,60,424) = 0,539986323$

Çizelge 5.4 : Bulanık Aritmetik Çözüm Yöntemlerinin Kritik Yol Hesabı.

Yollar	$FCPM \ P_k$ Değeri
$p_5 = 1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 7$	0,507667
$p_2 = 1 \rightarrow 2 \rightarrow 4 \rightarrow 5 \rightarrow 7$	0,515443
$p_6 = 1 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 7$	0,51848
$p_3 = 1 \rightarrow 2 \rightarrow 4 \rightarrow 6 \rightarrow 7$	0,526554
$p_1 = 1 \rightarrow 2 \rightarrow 5 \rightarrow 7$	0,539986
$p_4 = 1 \rightarrow 3 \rightarrow 6 \rightarrow 7$	0,595301

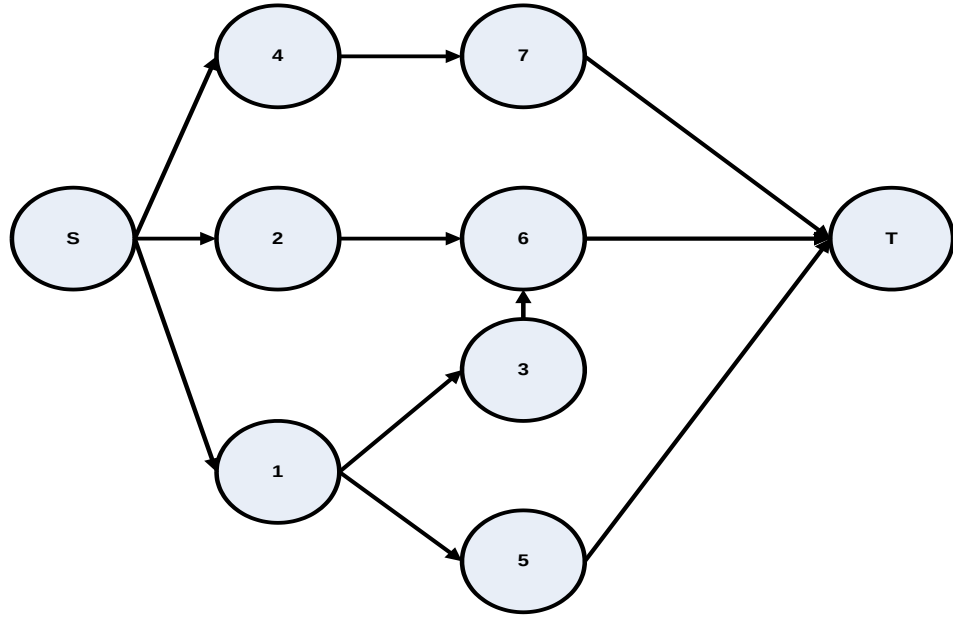
Bu örnekte, klasik sayılardan çok daha gerçekçi, bulanık faaliyet süreli KYM problemini çözmek için doğrusal programla ile bulanık aritmetik çözüm yöntemleri geliştirilmiş ve birbiriyle kıyaslanarak önerilen yöntemin geçerliliği ispat edilmiştir. Yager'in sıralama yöntemini temel alan bulanık KYM, problemi geleneksel akış doğrusal programlama çözüm yaklaşımı kullanılarak çözülebilecek klasik sayıya dönüştürür. Bulanık kritik yol incelendiğinde, Yager'in görüşünün en kritik yol olduğu garantilenmektedir. Ayrıca, bulanık sayıların üyelik fonksiyonlarının şekilsel biçimlerinin bilinmesine ihtiyaç duyulmaması, problemin çözümde büyük bir kolaylık sağlamaktadır.

5.5 Uygulamalı Sayısal Örnek 2

Sayısal örneğin bulanık aritmetik yöntemle çözümü ise aşağıda sunulmuştur.

Çizelge 5.5 : Örnek Proje Verileri.

No	Faaliyetler (i,j)	Faaliyet Süreleri (D_{ij})
S		(0,0,0,0)
1	(1-2)	(10,15,15,20)
2	(1-3)	(30,40,40,50)
3	(2-3)	(30,40,50,60)
4	(1-4)	(15,20,25,30)
5	(2-5)	(60,100,150,180)
6	(3-5)	(60,100,150,180)
7	(4-5)	(60,100,150,180)
T		(0,0,0,0)



Şekil 5.3 : Örnek Veri Şebeke Yapısı.

$$\beta = \left[\sum_i \sum_j \frac{b_{ij} - a_{ij}}{b_{ij} - a_{ij} + d_{ij} - c_{ij}} \right] / t$$

$$= \left[\frac{(15-10)}{(15-10)+(20-15)} + \frac{(40-30)}{(40-30)+(50-40)} + \frac{40-30}{(40-30)+(60-50)} + \frac{(20-15)}{(20-15)+(30-25)} \right. \\ \left. + \frac{(100-60)}{(100-60)+(180-150)} + \frac{(100-60)}{(100-60)+(180-150)} + \frac{(100-60)}{(100-60)+(180-150)} \right] / 7$$

$$= \left[\frac{5}{10} + \frac{10}{20} + \frac{10}{20} + \frac{5}{10} + \frac{40}{70} + \frac{40}{70} + \frac{40}{70} \right] / 7 = (2 + \frac{12}{7}) / 7 = 26 / 49 = 0.5306$$

Denk. (5.11) kullanılarak, toplam proje risk indeksi $\beta = 0.5306$ olarak hesaplanır.

Adım1. $FEST_A = 0,0,0,0$ olmak üzere $FEST_j = \max FEST_i \oplus FAT_{ij} | i \in NP \ j, j \neq 1, j \in N$,

$j = 2,3,4,\dots,T$ özellik 1.a. kullanılarak hesaplanır.

$$FEST_s = (0,0,0,0)$$

$$FEST_1 = FEST_s = 0,0,0,0$$

$$FEST_2 = FEST_s = 0,0,0,0$$

$$FEST_3 = FEST_1 \oplus FAT_1 = 0,0,0,0 \oplus 10,15,15,20 = 10,15,15,20$$

$$FEST_4 = FEST_s = 0,0,0,0$$

$$FEST_5 = FEST_1 \oplus FAT_1 = 0,0,0,0 \oplus 10,15,15,20 = 10,15,15,20$$

$$FEST_6 = \max FEST_2 \oplus FAT_2, FEST_3 \oplus FAT_3$$

$$= \max \{ (0,0,0,0) \oplus (60,40,40,50), [(0,15,15,20) \oplus (60,40,50,60)] \}$$

$$FEST_6 = \max \{ (30,40,40,50), (40,55,65,80) \}$$

$$FEST_6 = (40,55,65,80)$$

$$FEST_7 = FEST_4 \oplus FAT_4 = (0,0,0,0) \oplus (5,20,25,30) = (5,20,25,30)$$

$$FEST_T = \max \{ FEST_5 \oplus FAT_5, FEST_6 \oplus FAT_6, FEST_7 \oplus FAT_7 \}$$

$$FEST_T = \max \left\{ [(0,15,15,20) \oplus (60,100,150,180)], [(40,55,65,80) \oplus (60,100,150,180)], [(5,20,25,30) \oplus (60,100,150,180)] \right\}$$

$$FEST_T = \max \{ (70,115,165,200), (100,155,215,260), (75,120,175,210) \}$$

$$FEST_T = (100,155,215,260)$$

$$R_{A_1} = \beta \left[\frac{d_i - x_1}{x_2 - x_1 - c_i + d_i} \right] + 1 - \beta \left[1 - \frac{x_2 - a_i}{x_2 - x_1 + b_i - a_i} \right]$$

$$30,40,40,50, 40,55,65,80$$

$$x_1 = 30, x_2 = 80$$

$$R_{30,40,40,50} = 0.5306 \left[\frac{50-30}{80-30-40+50} \right] + (1-0.5306) \left[1 - \frac{80-30}{80-30+40-30} \right]$$

$$R_{30,40,40,50} = 0.5306 \left[\frac{20}{60} \right] + 0.4694 \left[1 - \frac{50}{60} \right] = 0.2551$$

$$R_{(40,55,65,80)} = 0.5306 \left[\frac{(60-30)}{(60-30-65+80)} \right] + (1-0.5306) \left[1 - \frac{(60-40)}{(60-30+55-40)} \right]$$

$$R_{(40,55,65,80)} = 0.5306 \left[\frac{50}{65} \right] + 0.4694 \left[1 - \frac{40}{65} \right] = 0.5887$$

$$R_{A_i} = \beta \left[\frac{d_i - x_1}{x_2 - x_1 - c_i + d_i} \right] + (1 - \beta) \left[1 - \frac{x_2 - a_i}{x_2 - x_1 + b_i - a_i} \right]$$

$$70, 115, 165, 200, 100, 155, 215, 260, 75, 120, 175, 210$$

$$x_1 = 70, x_2 = 260$$

$$R_{70, 115, 165, 200} = 0.5306 \left[\frac{200 - 70}{260 - 70 - 165 + 200} \right] + (1 - 0.5306) \left[1 - \frac{260 - 70}{260 - 70 + 115 - 70} \right]$$

$$R_{(70, 115, 165, 200)} = 0.5306 \left[\frac{130}{225} \right] + 0.4694 \left[1 - \frac{190}{235} \right] = 0.3965$$

$$R_{100, 155, 215, 260} = 0.5306 \left[\frac{260 - 70}{260 - 70 - 215 + 260} \right] + (1 - 0.5306) \left[1 - \frac{260 - 100}{260 - 70 + 155 - 100} \right]$$

$$R_{100, 155, 215, 260} = 0.5306 \left[\frac{190}{235} \right] + 0.4694 \left[1 - \frac{160}{245} \right] = 0.5918$$

$$R_{75, 120, 175, 210} = 0.5306 \left[\frac{210 - 70}{260 - 70 - 175 + 210} \right] + (1 - 0.5306) \left[1 - \frac{(260 - 75)}{(260 - 70 + 120 - 75)} \right]$$

$$R_{(75, 120, 175, 210)} = 0.5306 \left[\frac{140}{225} \right] + 0.4694 \left[1 - \frac{185}{235} \right] = 0.43$$

$$FECT_5 = FEST_5 \oplus FAT_5 = (0, 0, 0, 0) \oplus (0, 0, 0, 0) = (0, 0, 0, 0)$$

$$FECT_1 = FEST_1 \oplus FAT_1 = (0, 0, 0, 0) \oplus 10, 15, 15, 20 = 10, 15, 15, 20$$

$$FECT_2 = FEST_2 \oplus FAT_2 = (0, 0, 0, 0) \oplus 30, 40, 40, 50 = 30, 40, 40, 50$$

$$FECT_3 = FEST_3 \oplus FAT_3 = 10, 15, 15, 20 \oplus 30, 40, 50, 60 = (40, 55, 65, 80)$$

$$FECT_4 = FEST_4 \oplus FAT_4 = (0, 0, 0, 0) \oplus (5, 20, 25, 30) = (5, 20, 25, 30)$$

$$FECT_5 = FEST_5 \oplus FAT_5 = (0, 15, 15, 20) \oplus (60, 100, 150, 180) = (70, 115, 165, 200)$$

$$FECT_6 = FEST_6 \oplus FAT_6 = (40, 55, 65, 80) \oplus (60, 100, 150, 180) = (100, 155, 215, 260)$$

$$FECT_7 = FEST_7 \oplus FAT_7 = (5, 20, 25, 30) \oplus (60, 100, 150, 180) = (75, 120, 175, 210)$$

$$FECT_T = FEST_T \oplus FAT_T = (00, 155, 215, 260) \oplus (0, 0, 0, 0) = (00, 155, 215, 260)$$

$$\text{Adım 2.a. } FLCT_j = \min FLCT_k \ominus FAT_{jk} \mid k \in NS \setminus j, j \neq n, j \in N$$

2. a.) $FLCT_n = FEST_n$ olarak alınır ve $FLCT_j, j = n - 1, n - 2, \dots, 2, 1$, özellik 2.a.

kullanılarak hesaplanır.

$$FLCT_T = FECT_T$$

$$FLCT_T = 100, 155, 215, 260$$

$$FLCT_7 = FLCT_T \ominus FAT_T = 100, 155, 215, 260 \ominus 0, 0, 0, 0 = 100, 155, 215, 260$$

$$FLCT_6 = FLCT_T \ominus FAT_T = 100, 155, 215, 260 \ominus 0, 0, 0, 0 = 100, 155, 215, 260$$

$$FLCT_5 = FLCT_T \ominus FAT_T = 100, 155, 215, 260 \ominus 0, 0, 0, 0 = 100, 155, 215, 260$$

$$FLCT_4 = FLCT_7 \ominus FAT_7 = (00, 155, 215, 260) \ominus (60, 100, 150, 180) = (80, 5, 115, 200)$$

$$FLCT_3 = FLCT_6 \ominus FAT_6 = (00, 155, 215, 260) \ominus (60, 100, 150, 180) = (80, 5, 115, 200)$$

$$FLCT_2 = FLCT_6 \ominus FAT_6 = (00, 155, 215, 260) \ominus (60, 100, 150, 180) = (80, 5, 115, 200)$$

$$FLCT_1 = \min \{ FLCT_3 \ominus FAT_3, FLCT_5 \ominus FAT_5 \}$$

$$FLCT_1 = \min \left[-80, 5, 115, 200 \ominus 30, 40, 50, 60 \right], \left[100, 155, 215, 260 \ominus 60, 100, 150, 180 \right]$$

$$FLCT_1 = \min \left[-140, -45, 75, 170 \right], \left[-80, 5, 115, 200 \right] = -140, -45, 75, 170$$

$$FLCT_s = \min \quad FLCT_1 \ominus FAT_1, \quad FLCT_2 \ominus FAT_2, \quad FLCT_4 \ominus FAT_4$$

$$FLCT_s = \min \left\{ \left[-140, -45, 75, 170 \right]_1 \ominus (10, 15, 15, 20) \right\}, \left[-80, 5, 115, 200 \right]_2 \ominus (30, 40, 40, 50) \right\}, \left[-80, 5, 115, 200 \right] \ominus (15, 20, 25, 30) \right\}$$

$$FLCT_s = \min \left\{ \left[-160, -60, 60, 160 \right], \left[-130, -35, 75, 170 \right] \ominus (30, 40, 40, 50) \right\}, \left[-110, -20, 95, 185 \right] \right\}$$

$$FLCT_s = \langle -160, -60, 60, 160 \rangle$$

$$R_{A_i} = \beta \left[\frac{d_i - x_1}{x_2 - x_1 - c_i + d_i} \right] + 1 - \beta \left[1 - \frac{x_2 - a_i}{x_2 - x_1 + b_i - a_i} \right]$$

$$\left[-140, -45, 75, 170 \right], \left[-80, 5, 115, 200 \right]$$

$$x_1 = -140, x_2 = 200$$

$$R_{-140, -45, 75, 170} = 0.5306 \left[\frac{170 + 140}{200 + 140 - 75 + 170} \right] + (1 - 0.5306) \left[1 - \frac{200 + 140}{200 + 140 - 45 + 140} \right]$$

$$R_{\langle -140, -45, 75, 170 \rangle} = 0.5306 \left[\frac{310}{435} \right] + 0.4694 \left[1 - \frac{340}{435} \right] = 0.4806$$

$$R_{\langle -80, 5, 115, 200 \rangle} = 0.5306 \left[\frac{\langle 200 + 140 \rangle}{\langle 200 + 140 - 115 + 200 \rangle} \right] + (1 - 0.5306) \left[1 - \frac{\langle 200 + 80 \rangle}{\langle 200 + 140 + 5 + 80 \rangle} \right]$$

$$R_{\langle -80, 5, 115, 200 \rangle} = 0.5306 \left[\frac{340}{425} \right] + 0.4694 \left[1 - \frac{280}{425} \right] = 0.5846$$

$$R_{A_i} = \beta \left[\frac{d_i - x_1}{x_2 - x_1 - c_i + d_i} \right] + 1 - \beta \left[1 - \frac{x_2 - a_i}{x_2 - x_1 + b_i - a_i} \right]$$

$$\left[-160, -60, 60, 160 \right], \left[-130, -35, 75, 170 \right], \left[-110, -20, 95, 185 \right]$$

$$x_1 = -160, x_2 = 170$$

$$R_{-160, -60, 60, 160}$$

$$= 0.5306 \left[\frac{160 + 160}{170 + 160 - 60 + 160} \right] + (1 - 0.5306) \left[1 - \frac{170 + 160}{\langle 170 + 160 - 60 + 160 \rangle} \right]$$

$$R_{\langle -160, -60, 60, 160 \rangle} = 0.5306 \left[\frac{320}{430} \right] + 0.4694 \left[1 - \frac{330}{430} \right] = 0.50402$$

$$R_{-130, -35, 75, 170}$$

$$= 0.5306 \left[\frac{170 + 160}{170 + 160 - 75 + 170} \right] + (1 - 0.5306) \left[1 - \frac{170 + 130}{170 + 160 - 35 + 130} \right]$$

$$R_{-130, -35, 75, 170} = 0.5306 \left[\frac{330}{425} \right] + 0.4694 \left[1 - \frac{300}{425} \right] = 0.55005$$

$$\begin{aligned}
R & -110, -20, 95, 185 \\
& = 0.5306 \left[\frac{170 + 160}{170 + 160 - 95 + 185} \right] + (1 - 0.5306) \left[1 - \frac{170 + 110}{170 + 160 - 20 + 110} \right] \\
R & -110, -20, 95, 185 = 0.5306 \left[\frac{330}{420} \right] + 0.4694 \left[1 - \frac{280}{420} \right] = 0.5734
\end{aligned}$$

$$\begin{aligned}
FLST_T &= FLCT_T \ominus FAT_T = 100, 155, 215, 260 \ominus 0, 0, 0, 0 = 100, 155, 215, 260 \\
FLST_7 &= FLCT_7 \ominus FAT_7 = 100, 155, 215, 260 \ominus (60, 100, 150, 180) = -80, 5, 115, 200 \\
FLST_6 &= FLCT_6 \ominus FAT_6 = 100, 155, 215, 260 \ominus (60, 100, 150, 180) = -80, 5, 115, 200 \\
FLST_5 &= FLCT_5 \ominus FAT_5 = (100, 155, 215, 260) \ominus (60, 100, 150, 180) = (-80, 5, 115, 200) \\
FLST_4 &= FLCT_4 \ominus FAT_4 = (-80, 5, 115, 200) \ominus (5, 20, 25, 30) = (-110, -20, 95, 185) \\
FLST_3 &= FLCT_3 \ominus FAT_3 = (-80, 5, 115, 200) \ominus (60, 40, 50, 60) = (-140, -45, 75, 170) \\
FLST_2 &= FLCT_2 \ominus FAT_2 = (-80, 5, 115, 200) \ominus (30, 40, 40, 50) = (-130, -35, 75, 170) \\
FLST_1 &= FLCT_1 \ominus FAT_1 = (-140, -45, 75, 170) \ominus (10, 15, 15, 20) = (-160, -60, 60, 160) \\
FLST_S &= FLCT_S \ominus FAT_S = (-160, -60, 60, 160) \ominus (0, 0, 0, 0) = (-160, -60, 60, 160)
\end{aligned}$$

Adm 3

$$FTS_{ij} = FLCT_j \ominus FECT_i \oplus FAT_{ij}, \quad 1 \leq i < j \leq n; \quad i, j \in N$$

$$\begin{aligned}
FTS_S &= FLCT_S \ominus FECT_S = -160, -60, 60, 160 \ominus (0, 0, 0, 0) = -160, -60, 60, 160 \\
FTS_1 &= FLCT_1 \ominus FECT_1 = -140, -45, 75, 170 \ominus 10, 15, 15, 20 = -160, -60, 60, 160 \\
FTS_2 &= FLCT_2 \ominus FECT_2 = -80, 5, 115, 200 \ominus 30, 40, 40, 50 = (-130, -35, 75, 170) \\
FTS_3 &= FLCT_3 \ominus FECT_3 = (-80, 5, 115, 200) \ominus (40, 55, 65, 80) = (-160, -60, 60, 160) \\
FTS_4 &= FLCT_4 \ominus FECT_4 = (-80, 5, 115, 200) \ominus (5, 20, 25, 30) = (-110, -20, 95, 185) \\
FTS_5 &= FLCT_5 \ominus FECT_5 = (100, 155, 215, 260) \ominus (70, 115, 165, 200) = (-100, -10, 100, 190) \\
FTS_6 &= FLCT_6 \ominus FECT_6 = (100, 155, 215, 260) \ominus (100, 155, 215, 260) = (-160, -60, 60, 160) \\
FTS_7 &= FLCT_7 \ominus FECT_7 = (100, 155, 215, 260) \ominus (75, 120, 175, 210) = (-110, -20, 95, 185) \\
FTS_T &= FLCT_T \ominus FECT_T = (100, 155, 215, 260) \ominus (100, 155, 215, 260) = (-160, -60, 60, 160)
\end{aligned}$$

Adm 4

$$FCPM \ P_K = \sum_{\substack{1 \leq i < j \leq n \\ i, j \in P_K}} FTS_{ij}, \quad P_K \in PT$$

$$PT = S, 4, 7, T, \quad S, 2, 6, T, \quad S, 1, 3, 6, T, \quad S, 1, 5, T$$

$$PT_1 = S, 4, 7, T$$

$$\begin{aligned}
FCPM \ P_1 &= FTS_s \oplus FTS_4 \oplus FTS_7 \oplus FTS_T \\
&= -160, -60, 60, 160 \oplus (-110, -20, 95, 185) \oplus (-110, -20, 95, 185) \oplus (-160, -60, 60, 160) \\
&= (-540, -160, 310, 690)
\end{aligned}$$

$$PT_2 = S, 2, 6, T$$

$$\begin{aligned}
FCPM \ P_2 &= FTS_s \oplus FTS_2 \oplus FTS_6 \oplus FTS_T \\
&= -160, -60, 60, 160 \oplus (-130, -35, 75, 170) \oplus (-160, -60, 60, 160) \oplus (-160, -60, 60, 160) \\
&= (-610, -215, 255, 650)
\end{aligned}$$

$$PT_3 = \{1, 3, 6, T\}$$

$$\begin{aligned}
FCPM \ P_3 &= FTS_s \oplus FTS_1 \oplus FTS_3 \oplus FTS_6 \oplus FTS_T \\
&= -160, -60, 60, 160 \oplus -160, -60, 60, 160 \oplus -160, -60, 60, 160 \\
&\oplus (-160, -60, 60, 160) \oplus (-160, -60, 60, 160) \\
&= (-800, -300, 300, 800)
\end{aligned}$$

$$PT_4 = S, 1, 5, T$$

$$\begin{aligned}
FCPM \ P_4 &= FTS_s \oplus FTS_1 \oplus FTS_5 \oplus FTS_T \\
&= -160, -60, 60, 160 \oplus (-160, -60, 60, 160) \\
&\oplus (-100, -10, 100, 190) \oplus (-160, -60, 60, 160) \\
&= (-580, -190, 280, 670)
\end{aligned}$$

$$R \left(FCPM \ P_1 \right) R \left(-540, -160, 310, 690 \right) = 0.5511$$

$$R \left(FCPM \ P_2 \right) R \left(-610, -215, 255, 650 \right) = 0.5233$$

$$R \left(FCPM \ P_3 \right) R \left(-800, -300, 300, 800 \right) = 0.516$$

$$R \left(FCPM \ P_4 \right) R \left(-580, -190, 280, 670 \right) = 0.5358$$

Bütün olası yollar bulunur ve $FCPM \ P_k$ değeri özellik 4 kullanılarak hesaplanır.

$$R \ A_i = \beta \left[\frac{d_i - x_1}{x_2 - x_1 - c_i + d_i} \right] + 1 - \beta \left[1 - \frac{x_2 - a_i}{x_2 - x_1 + b_i - a_i} \right]$$

$$x_1 = -800, x_2 = 800$$

$$R \ (-540, -160, 310, 690)$$

$$= 0.5306 \left[\frac{690 + 800}{800 + 800 - 310 + 690} \right] + (1 - 0.5306) \left[1 - \frac{800 + 540}{800 + 800 - 160 + 540} \right]$$

$$R \ (-540, -160, 310, 690) = 0.5306 \left[\frac{1490}{1980} \right] + 0.4694 \left[1 - \frac{1340}{1980} \right] = 0.5511$$

$$R \ (-610, -215, 255, 650)$$

$$= 0.5306 \left[\frac{650 + 800}{800 + 800 - 255 + 650} \right] + (1 - 0.5306) \left[1 - \frac{800 + 610}{800 + 800 - 215 + 610} \right]$$

$$R \ (-610, -215, 255, 650) = 0.5306 \left[\frac{1450}{1995} \right] + 0.4694 \left[1 - \frac{1410}{1995} \right] = 0.5233$$

$$\begin{aligned}
& R(-800, -300, 300, 800) \\
&= 0.5306 \left[\frac{800 + 800}{800 + 800 - 300 + 800} \right] + (1 - 0.5306) \left[1 - \frac{800 + 800}{800 + 800 - 300 + 800} \right] \\
& R(-800, -300, 300, 800) = 0.5306 \left[\frac{1600}{2100} \right] + 0.4694 \left[1 - \frac{1600}{2100} \right] = 0.516
\end{aligned}$$

$$\begin{aligned}
& R(-580, -190, 280, 670) \\
&= 0.5306 \left[\frac{670 + 800}{800 + 800 - 280 + 670} \right] + (1 - 0.5306) \left[1 - \frac{800 + 580}{800 + 800 - 190 + 580} \right] \\
& R(-580, -190, 280, 670) = 0.5306 \left[\frac{1470}{1990} \right] + 0.4694 \left[1 - \frac{1380}{1990} \right] = 0.5358
\end{aligned}$$

EK-H’de Atli ve Kahraman (2012) makale çalışmamızda kullandığımız temsili bir uçak yakıt türbin motoruna ait depo seviyesi bakım faaliyetlerinin bulanık sayılardan oluştuğu ve Şekil H.1’de şebeke yapısı ve Çizelge H.1’de verileri gösterilen örnek B-KYM probleminin sonuçları gösterilmiştir.

6. DETERMİNİSTİK VE BULANIK FAALİYET SÜRELİ KAYNAK KISITLI PROJE ÇİZELGELEME PROBLEMLERİ

Bir önceki bölümde deterministik/kesin (deterministic/crisp) KYM ve bulanık KYM açıklanmıştır. KYM, genel olarak KKPC problemlerinde en temel yapı olması bakımından incelenmiştir. KYM, kaynak kısıtlarının göz ardı edilmesi nedeniyle projelerin çizelgelenmesinde yetersiz kalmaktadırlar. KYM’ler, kaynakların miktar ve kapasite kısıtlarına bağlı olduğunun göz ardı edilmesi, kapasite darboğazlarının oluşumunun dengelemeyle bile önlenememesi gibi sebeplerle kesin çözüm elde etmede yetersiz kalmışlardır. İşte bu ihtiyaçtan doğan “KKPC Modelleri”ne ait temel yaklaşımlar bu bölümde açıklanacaktır. Kaynak kısıtlarını da dikkate alan yöntem deterministik/kesin KKPC ve bulanık KKPC modelleri bu bölümde incelenmiş ve problemin çözümü için yeni ve etkin çözüm yöntemlerinin tanıtılması ve geliştirilmesi amaçlanmıştır. “Sezgisel Öncelik Kuralı (MinSlack ve ROT öncelik kuralları)” esasına göre oluşturulan bazı çözümler, başlangıç çözüm olarak kabul edilmiş ve alternatif olabilecek metasezgisel yöntemlerden TA yöntemiyle sınanmıştır; çözümler iyileştirilmiştir. Literatürde bilinen çeşitli problemler ile daha birçok PSPLIB’de verilen çeşitli projeler üzerinde Microsoft Visual C#’la geliştirilen bir program aracılığıyla denemeler yapılarak “Tabu Arama Sezgisel Çözüm Yöntemiyle” probleme güvenilir ve basit çözüm önerilerinde bulunulmuştur. Hesaplama sonuçlarıyla belirlenen performans kriterleri karşılaştırılarak, bilinen sezgisel yaklaşımların TA yöntemiyle birlikte çok daha hızlı, optimal ve optimale yakın sonuçlar verdiği belirlenmiştir. Kaynakların kısıtlı, zamanın çok değerli olduğu günümüz koşullarında, kaynakların verimli şekilde kullanımını sağlayacak bir yönetimin gerekliliği açıktır. Bu amaçla, “TA Algoritması” mantığıyla çalışan bir paket program hazırlanmış ve performansı literatürde tanımlanan örnek veri kümeleri ve problemler üzerinde test edilmiştir. Hazırlanan bu paket program ile, kısıtlı kaynak koşullarında projelerin tamamlanma süresi ve kritik yolu belirlenebilmektedir. Sonuç olarak, ortaya konulan bu paket program ile herhangi bir yüklenici firma, kendi kaynaklarını ve kısıtlarını göz önüne alarak ihalesine gireceği bir işi ne kadar sürede tamamlayacağını ve oluşan yeni duruma göre tedbirlerini

alabilecektir. TA yönteminin daha iyi ve gerçekçi planlamalar için projelere efektif etkili çözümler sağladığı düşünülmektedir. Burada incelenen “Deterministik ve Bulanık KKPC Problemleri” zaman-tabanlı amaç fonksiyonu, proje tamamlanma süresinin minizasyonunun kullanıldığını ve maliyet faktörüne yer verilmeyeceğini ifade etmekte fayda görülmektedir. Ancak bir sonraki bölümde incelenecek “Deterministik ve Bulanık ÇM-KKPC Modelleri” incelenirken maliyet unsuru da modele dahil edilecektir.

Bu bölümde; KKPC problemleri çözüm yöntemleri ve KKPC için tamsayılı doğrusal programlama (TDP) modeli verilmiştir. Ayrıca çalışma kapsamında, KKPC problemlerinin çözümünde sezgisel yaklaşımların kullanımına ilişkin çalışmalar tanıtılarak, bu yaklaşımlardan TA prosedürünün probleme uygulanması ile elde edilen deneysel sonuçlar, literatürde karşılaştırma amaçlı olarak yaygın biçimde kullanılmış; çeşitli büyüklüklerdeki problemler ve veri kümeleri üzerinde denemeler yapılmıştır. Elde edilen bulgular son bölümde sunulmuştur.

6.1 Kaynak Kısıtlı Proje Çizelgeleme

KKPC problemi, en uygun veya optimal çözümü, olurlu çözüm seçenekleri arasından sayılama yöntemi ile saptayıp amaç ölçütlerine göre değerlendiren yöntemler dizisidir. KKPCP, kısıtlı kaynaklar kullanılarak bir projeyi oluşturan faaliyetlerin, öncelik ilişkilerini ihlal etmeden amaç ve işlevlerini en iyilenecek biçimde çizelgelenmesidir. Problemin tanımına esas teşkil eden varsayımları aşağıdaki gibi sıralayabiliriz:

- i. Faaliyet süreleri deterministik/kesin veya bulanıktır.
- ii. Faaliyetlerin birim zamandaki kaynak kullanımı sabittir.
- iii. Bir faaliyete atanan kaynak, faaliyet süresince sadece o faaliyet tarafından kullanılır.
- iv. Başlatılan faaliyetler kesintisiz bitirilmek zorundadır; ara verilmesine izin verilmez.
- v. Faaliyetler iptal edilemez. Proje serimindeki her faaliyet gerçekleştirilmek zorundadır.

- vi. Projedeki faaliyetlerin gerçekleştirilmesi için yenilenebilir türde kaynaklara ihtiyaç olduğu varsayılmaktadır.
- vii. Kaynakların kullanımından doğan maliyetler toplam proje çizelgeleme maliyeti olarak modele dahil edilmiştir.
- viii. Projedeki faaliyetlerin bazılarının başlayabilmesi için öncül faaliyetlerin tamamlanması gerektiği öngörülmektedir.
- ix. Faaliyet sürelerinin yamuk bulanık sayılarla modellenmesi öngörülmektedir.

Bu varsayımların bazıları kaldırılarak daha farklı problem tanımlarına gitmek mümkündür. Örneğin, faaliyet sürelerinin rassal olmasına izin verilmesi durumunda AR-GE projeleri gibi faaliyetlerinin içeriği tam olarak öngörülemeyen projeler için, daha gerçekçi tanımlamalar yapılabilmektedir. Ayrıca belirsizlik ortamında faaliyetlerin içeriğinin tam ve kesin bir bilgiye dayandırlamaması durumunda, bulanık faaliyet süreli KKPÇ modellerinden bahsedilecektir. Bir kaynağın aynı anda birden fazla faaliyete hizmet verdiği durumlar vardır. Aynı anda birden fazla parça işleyebilen takım tezgahları gibi. KKPÇ probleminin proje yönetimi için ifade ettiği pratik değerin yanında, teorik açıdan da araştırmacılar için ayrı bir çekiciliği vardır. Proje süresinin enazlandığı KKPÇ probleminin “NP-hard in the strong sense” bir problem olmasıdır.

6.2 TM-KKPÇP’inin TDP Modeli

KKPÇ probleminde kullanılan iki grup karar değişkeni vardır: Birinci grup karar değişkenleri, faaliyetlerin hangi zaman aralığında yürütüleceğini belirleyen değişkenlerdir. İkinci grup karar değişkenleri ise, modelde kullanılan kaynakların faaliyetlere nasıl dağıtılacağını belirleyen kaynak atama değişkenleridir. Ayrıca, modelde kullanılan beş grup parametre bulunmaktadır. Bunlar sırasıyla; faaliyet süresi, kaynak kullanım maliyeti, kaynak kullanım sınırı, faaliyet kaynak ihtiyacı ve öncelik kısıtı parametreleridir.

Bu çalışmada kullanılan birinci grup karar değişkenleri; her faaliyetin başlangıç zamanını, bitiş zamanını ve hangi aralıkta devam ettiğini belirleyen faaliyet devam karar değişkenleridir. Bu değişkenler, modele ikili tamsayı değişkenleri olarak dahil edilmiştir. Kaynakların faaliyetlere dağılımını belirleyen ikinci grup kaynak atama karar değişkenleri ise; çoklu kaynak durumuna göre modele dahil edilmiştir. Her

dönemde kullanılabilecek miktarı sınırlandırılmış olan her kaynağın hangi proje faaliyetinde kaç birim olarak kullanılacağını belirleyen kaynak atama karar değişkenleri, tamsayı değişkenleri olarak modelde kullanılmaktadır. Bu bölümde incelenecek problem, tek bir projenin yukarıda yapılan varsayımlar çerçevesinde projenin tamamlanma süresinin minimizasyonu problemidir. Kaynak kullanımı tek modludur. TM-KKPC'nin bu en basit halinin matematiksel programlama gösterimi aşağıda verilmiştir.

$$\min Z = \sum_{t=EFT_j}^{LFT_j} tX_{jt} \quad (6.1)$$

$$\text{Toplam Maliyet} = \sum_{t=EFT_j}^{LFT_j} C_{jr} tX_{jt} \quad (6.2)$$

Kısıtlar

$$\sum_{t=EFT_j}^{LFT_j} X_{jt} = 1 \quad j = 1, \dots, J \quad (6.3)$$

$$\sum_{t=EFT_i}^{LFT_i} tX_{it} \leq \sum_{t=EFT_j}^{LFT_j} (t - d_j) X_{jt} \quad j = 2, \dots, J, \quad i \in P_j \quad (6.4)$$

$$\sum_{j=1}^J k_{jr} \sum_{\tau=t}^{t+d_j-1} X_{jr} \leq K_r \quad r \in R, \quad t = 1, \dots, T \quad (6.5)$$

$$X_{jt} = \begin{cases} 1, & \text{eger } j \text{ faaliyeti } t \text{ donemi sonunda bitiyorsa} \\ 0, & \text{diger durumlarda} \end{cases} \quad (6.6)$$

Kullanılan sembollerin tanımları aşağıda verilmiştir:

t Zaman indisi $t = 1, \dots, T$

j Faaliyet indisi $j = 1, \dots, J$

R Yenilenebilir kaynaklar kümesi,

d_j j Faaliyetinin süresi,

P_j j Faaliyetinin öncüllerinin kümesi,

EFT_j j Faaliyetinin en erken bitiş zamanı,

LFT_j j Faaliyetinin en geç bitiş zamanı,

k_{jr} j Faaliyetinin r kaynağından birim zaman kullanım miktarı,

K_r r Yenilenebilir kaynağının birim zaman kullanım üst sınırı.

C_{jr} j Faaliyetinin r kaynağının birim zamanda kullanım maliyeti (1 pb)

Denk.6.1. probleminde, amaç işlevi; bitiş faaliyetinin bitiş zamanı diğer bir deyişle, proje süresinin minimizasyonudur. Denk.6.2. probleminde ise, projenin toplam çizelgeleme maliyetini ifade eder. Denk.6.3. kısıt kümesi, her faaliyetin mutlaka çizelgelenmesini sağlamaktadır. Kısıt kümesi Denk.6.4. ise, j faaliyeti ile bu faaliyetin öncülü olan i faaliyeti arasındaki öncüllük ilişkisinin yerine getirilmesi gereğini ifade etmektedir. Birim zaman başına kaynak kısıtı ise, kısıt kümesi Denk.6.5. ile gösterilmiştir. Denk.6.6., $\{0,1\}$ değişkeni x_{jt} 'nin tanımıdır. Değişken adedini azaltabilmek amacı ile x_{jt} değişkeni $[EFT_j, LFT_j]$ zaman aralığında tanımlanmıştır. Bu zaman aralıklarının, optimal çözümü dışlamadan, en dar şekilde tanımı, değişken sayısının azaltılmasına katkıda bulunacaktır.

KKPÇ probleminin yapısına uygun olarak modelde kullanılan beş grup parametre mevcuttur. Bunlardan birincisi faaliyet süresi parametresidir. Modelde faaliyet süreleri, yamuk bulanık sayı üyelik fonksiyonları şeklinde ifade edilmektedir. Faaliyet çizelgelemesi yapılırken kullanılan bulanık yamuk sayılar, Yager (1981) tarafından önerilen alan karşılığı indisi kullanılarak sıralandırılmıştır. Modelde kullanılan ikinci grup parametre, kaynak kullanım maliyetlerini belirleyen parametredir. Kaynak kullanım maliyetinin dönemsel olarak değişmediği varsayımı kullanılmaktadır. Dolayısıyla, her kaynağın projenin dönemlerinde sabit brüt birim maliyetle (1 Pb) kullanılabileceği, kaynak kullanım maliyeti parametresiyle ifade edilmektedir. Üçüncü grup parametre, çoklu kaynak varsayımı altında kaynak kullanım sınırlarını belirleyen parametredir. Her kaynağın birim zamanda ve proje süresince kullanılabilecek miktarının kullanılan kaynak türüne göre sınırlı olduğu varsayılmıştır. Dördüncü grup parametre ise, faaliyetlerin gerçekleştirilmesi için gereken kaynak ihtiyacı miktarını belirleyen parametredir. Buna göre, her faaliyetin

yürütülebilmesi için mevcut kaynakların tamamının veya bir kısmının belirli miktarda kullanması gerekmektedir. Modelde kullanılan beşinci grup parametre, öncelik kısıtı parametreleridir. Proje faaliyetlerinin teknik yapısı gereği, bazı faaliyetlerin başlatılabilmesi için, öncül faaliyetlerin tamamlanması gerekmektedir. Modelde bu ilişkiler, teknik kısıtlar matrisini ifade etmektedir.

6.3 B-TM-KKPÇP'nin TDP Modeli

Proje çizelgeleme konusunda yapılan çalışmaların büyük çoğunluğunda, problemle ilgili tüm bilgilere sahip olunduğu ve problemin statik, deterministik ortamda çalıştığı varsayılmaktadır. Ancak gerçek hayatta proje faaliyetleri büyük ölçüde belirsizdir ve proje uygulanırken aşama aşama çözülmemektedir. “Bulanık Küme Teorisi” tam olarak tanımlanması zor olan sistemlerin modellenmesinde kullanılmaktadır. Bu teori, kesin bilginin olmadığı ve özneliğin bulunduğu bir modeli formüle ederek çözüm sürecine alan bir yöntemdir. B-TM-KKPÇ'nin bu en basit halinin matematiksel programlama gösterimi aşağıda verilmiştir:

$$\text{Min } \sum_{t=1}^{\tilde{H}} \tilde{t}x_{n+1,t} \quad (6.7)$$

Kısıtlar

$$\sum_{t=1}^{\tilde{H}} \tilde{t}x_{jt} + \tilde{p}_i - \sum_{t=1}^{\tilde{H}} \tilde{t}x_{it} \leq 0 \quad (6.8)$$

$$\sum_{j=1}^n \left(r_{jk}^{\rho} \sum_{u=t}^{\tilde{t}_i + \tilde{p}_i - 1} x_{ju} \right) \leq \mathfrak{R}_k^{\rho} \quad (6.9)$$

$$\sum_{t=1}^{\tilde{H}} x_{jt} = 1 \quad \text{for } j=1, \dots, n \quad (6.10)$$

$$\tilde{H} = \sum_{j=1}^v \tilde{p}_j \quad (6.11)$$

$$x_{jt} = \begin{cases} 1, & \text{If activity } j \text{ finishes at time instant } t \\ 0, & \text{otherwise} \end{cases} \quad (6.12)$$

Kullanılan sembollerin tanımları aşağıda verilmiştir:

$\tilde{t}$	Bir faaliyetin kaynaklara tahsis edilmesi için kabul anlık şimdiki zaman
A	Çizelgelenen faaliyetler kümesi;
$\tilde{\rho}_i$	$i \in A$ Faaliyetinin bulanık hazırlık süresi
$\tilde{d}_i$	$i \in A$ Faaliyetinin bulanık süresi
$\tilde{s}_i$	$i \in A$ Faaliyetinin bulanık başlangıç süresi
$\tilde{f}_i$	$i \in A$ Faaliyetinin bulanık tamamlanma süresi
$E \tilde{t}$	$\tilde{t}$ zamanında uygun olan faaliyetler kümesi. Örnek; $\tilde{t}$ zamanında hemen önceli olan faaliyet tamamlanmış ancak faaliyetler henüz çizelgelenmemiştir.
$E' \tilde{t}$	$\tilde{\rho}_i \geq \tilde{t}$ için faaliyetlerin $E \tilde{t}$ kümesinin alt kümesidir. Örnek $\tilde{t}$ zamanında hazır değildir
$S \tilde{t}$	$\tilde{s}_i \leq \tilde{t}$ için Faaliyetler kümesi Örnek, $B = \tilde{f}_i : \forall i \in S \tilde{t} \cup \tilde{\rho}_i : \forall i \in E' \tilde{t}$ kümesinden önce çizelgelenir
$\tilde{l}$	Zayıf bulanık sayı kıyaslamayı temel alan B kümesinin en küçük değeri
p_j	Projede j faaliyetinin işlem süresi
r_{jk}	j Faaliyetinde kullanılan k kaynak tipinden herbir dönemde kullanılan miktar
$\Re_k^\rho$	Sabit bir dönemde elde edilebilir sınırlı k tipi yenilenebilir kaynağının maksimum miktarı
S	p Döneminde işlem gören faaliyetlerin kümesi

Dolayısıyla belirsizlikle dolu bu problemin en uygun çözümünü bulmak için belirsizlikle uğraşabilecek çözüm tekniklerine ihtiyaç vardır. Böyle bir problemin çözümünde günümüzde kullanılabilecek en iyi yöntemlerden biri “Bulanık Küme Teorisi” yöntemidir. Özellikle proje faaliyetleriyle ilgili geçmiş bilgilere sahip

olunmadığı ürün geliştirme projelerinde, faaliyet sürelerinin proje yöneticisi tarafından uzman görüşü esas alınarak belirlenmesi önem kazanmaktadır. Bu araştırmada, NP-zor, NP-tamam kapsamında yer alan, bulanık proje çizelgeleme problemlerinin çözümlerinde kullanılan, Karınca Kolonileri, Genetik Algoritma, Yerel Arama Algoritması ve TA Algoritması metasezgisel yöntemleri incelenmiştir. Ayrıca son yıllarda kullanılan ve başarılı sonuçlar veren Paralel Kanguru Algoritması metasezgisel yöntemi ile örnek bir bulanık proje çizelgeleme problemi çözülmüştür. Araştırmanın ikinci bölümünde, proje çizelgeleme konusu açıklanmıştır. Üçüncü bölümde, bulanık proje çizelgeleme konusu izah edilmiştir. Dördüncü bölümde, bulanık proje çizelgeleme problemlerinin, kesin çözüm veren yöntemler ile kesin çözümü garanti etmeyen metasezgisel yöntemlerle çözümleri, açıklanmıştır. Sonuç kısmını oluşturan beşinci bölümde ise, metasezgisellerin bulanık proje çizelgeleme problemlerinin çözümündeki önemi üzerinde durulmuştur.

6.4 KKPÇP'lerinin Çözüm Yöntemleri

6.4.1 Kesin çözüm veren doğrusal programlama modelleri

KKPÇ probleminde yenilenebilen kaynakların tamsayı birimlerle ifade edilmesi ve seçilen zaman biriminin tamsayı değişkeni olması, problemin matematik modelinin de TDP modeli ile yapılabileceğini göstermektedir. KKPÇ problemlerinin çözümü amacıyla optimal sonuçlar veren TDP modelleri deterministik KKPÇ için, yukarıda belirtilen Denk (6.1-6.5) ile; bulanık KKPÇ için Denk (6.6-6.11)'de verilmiştir. Ancak bu yöntemler, değişken ve kısıt sayılarının az olduğu ve gerçek hayatta uygulanabilirlik olasılığının düşük olduğu problemlerde başarıyla uygulanmaktadır. İşlem sayısının aritmetik bir dizi şeklinde arttığı problemlerde, olurlu çözüm sayısı da üstel olarak artmaktadır. Bu da bu tip problemlerde modellerin oluşturulmasını, çözüm esnasında iyi bir planlama yapılmasını ve güçlü bir bilgisayar sistemini gerekli kılmaktadır ki çoğu zaman bu ihtiyacı karşılayacak güçlü bilgisayarlar mevcut değildir ve küçük boyuttaki problemlerin çözümü için bile oldukça uzun işlemci (CPU-AİB) zamanı ve çok sayıda iterasyon gerekmektedir.

Pratikte, problemleri kesin çözüm veren yöntemlerle çözmek, mevcut bilgisayarların hesaplama süresinden dolayı uygun olmasa da; bunların geliştirilmesi, bazı nedenlerden dolayı haklı gösterilebilir. Sezgisel yöntemlerin incelenmesi ve de

verimliliklerinin kıyaslanması için optimal çözümlerin bulunması oldukça yararlıdır. Ayrıca kesin çözüm veren yöntemler, belirli bir süre sonra durdurularak, sezgisel yöntem gibi kullanılabilir. En önemlisi; optimizasyon problemi, NP-zor olduğunda bunların uygulaması ile uygun bir çözümün olup olmadığı tespit edilebilmektedir.

Doğrusal Programlama (DP) problemi, karar değişkenlerinin doğrusal fonksiyonunu, doğrusal kısıtlar çerçevesinde, maksimize veya minimize etmek için optimize etmektir. Bu tanıma göre TDP, bazı değişkenlerin veya tüm değişkenlerin negatif olmayan tam sayı olduğu doğrusal programlamadır. Birçok çizelgeleme problemi, geleneksel DP veya TDP şeklinde formüle edilmektedir. Ancak önemli basitleştirmelerin yapılması gerekmektedir. Genel olarak, kesin sonuç veren yöntemler, amaç fonksiyonun karakteristiklerine bağlıdır (örneğin tek modlu faaliyetler). Ayrıca DP formülasyonları, genelde problemi tam olarak yansıtamamaktadır. Bu nedenle sadece belirli örneklerde veya küçük problemlerde kullanılabilir. KKPC ve CM-KKPC problemlerinin OPL-CPLEX editörünün kullanımıyla optimal çözümleri elde edilebilir. Url-4 sitesinde kısıtlı sürümüne ulaşılabilen program yardımıyla geliştireceğiniz C kodlamalarına dahil ederek çözümler üretilebilir. Bu konudaki çalışmalar halen devam etmektedir. KKPC için gerekli OPL-CPLEX editöründe hazırlanana psedo kod EK-I'da ve CM-KKPC için gerekli kod EK-I'da sunulmuştur.

Çok sayıda çalışmada dal-sınır (branch and bound) yöntemi kullanılmaktadır. Bir düğümdeki kısmi çizelgede devam etmekte olan faaliyetlerin en erken biteninin bitiş zamanı, o düğümden çıkacak düğümlerin çizelgeleme zamanını belirlemektedir. Yeni yaratılan düğümler henüz atanmamış faaliyetlerin olurlu alt kümeleridir. Düğüm seçimi, hiyerarşik olarak uygulanan altı seçim kuralı ile gerçekleştirilir. Dal-sınır ağacını budamak için, iki baskınlık kuralı kullanılmıştır. Demeulemeester ve Herroelen (1992), dikey inişli (depth first) bir dal-sınır yöntemi uygulaması önermişlerdir. Aday faaliyet kümesinde yer alan faaliyetler, belirli bir çizelgeleme anında öncüllük ilişkilerine göre atanabilir durumda olan faaliyetlerdir. Dallanılan düğümden, geçerli çizelgeleme anında aday faaliyet kümesindeki tüm faaliyetler atanmaktadır. Kaynak kısıtlarının ihlali durumunda, ertelenmeleri sonucu çizelgeyi kaynak kullanımı açısından olurlu kılacak en küçük faaliyet alt kümeleri saptanmaktadır. Bu alt kümeler, herhangi birisi diğer birisine baskın olmayacak

şekilde belirlenmektedir. En küçük ertelenen faaliyetler kümesi olarak nitelendirilen bu alt kümelerin herbiri, dallanılan düğüme bağlı yeni bir düğümü oluşturmaktadır. Burada ilginç olan, en küçük ertelenen faaliyetler kümesindeki bazı faaliyetlerin, dallanan düğümdaki çizelgede yer alabiliyor olmasıdır. Buna göre, bazı faaliyetler dal-sınır ağacı üzerinde dikey olarak aşağıya inerken, önce kısmi çizelgeye dahil olup sonra çıkabilmektedir. Düğüm budanması için, iki baskınlık kuralından yararlanılmaktadır. Bunlardan birincisinde, başka bir düğüm ile sola kaydırma sonucu aynı kısmi çözüme sahip olacak bir düğüm budanır. Dikey iniş için seçilecek düğüm, en küçük kritik sıralama bazlı altsınıra sahip düğüm olarak belirlenmektedir. Birden fazla böyle düğüm bulunması durumunda, seçim rassal olarak yapılır. Dikey iniş sonucu bir dal sonuna ulaşıldığında, dallanılmamış bir düğüme ilk ulaşılan kadar geri dönüş (backtracking) uygulanmaktadır.

6.4.2 Sezgisel Yöntemler

Kesin çözüm yöntemleri ile çözüm sağlanabilen problem büyüklüklerinin sınırlı olması ve ihtiyaç duyulan hesap sürelerinin büyüklüğü, araştırmacıları daha büyük proje serimleri için eniyi çözüm olmazsa da, “iyi” bir çözüme süratle ulaşan sezgisel yöntemler üzerinde çalışmaya yöneltmiştir. Sezgisel yöntemlerin sonuçları, aynı zamanda, kesin çözüm yöntemleri için bir alt veya üst sınır oluşturmaları bakımından da önem taşımaktadır. Öncelik kurallarına dayalı sezgisel yöntemlerin iki boyutu vardır: Kullanılan öncelik kuralı ve çizelgenin oluşturulma yöntemi. Kullanılan öncelik kuralına göre oluşturulan algoritmalar, tek bir çözüm oluşturmaya yöneliktir ve tüm faaliyetler, öncelik kuralları kullanılarak atandıktan ve böylece öncüllük ilişkileri ve kaynak kısıtları açısından olurlu bir çizelge oluşturulduktan sonra algoritma sonlandırılmaktadır. Literatürde çok sayıda sezgisel öncelik kuralı önerilmiştir ve bu kuralların performansına ilişkin olarak da çeşitli değerlendirme çalışmaları yapılmıştır. Literatürde sıklıkla kullanılan öncelik kurallarını şöyle sıralayabiliriz:

- i. En kısa işlem süresi (shortest processing time - SPT).
- ii. En kısa aylak süre (minimum slack - MinSlack).
- iii. En çok sayıda toplam ardıl (most total successor - MTS).
- iv. En büyük kaynak talebi (greatest resource demand - GRD).

- v. En geç bitirme zamanı (latest finish time - LFT).
- vi. En geç başlama zamanı (latest start time - LST).
- vii. Kaynak çizelgeleme metodu (resource scheduling method - RSM).
- viii. En büyük konumsal ağırlık (greatest rank positional weight - GRPW).
- ix. En kötü alternatif aylak (worst case slack – WCS)

Sezgisel öncelik kurallarının bir sınıflandırması; çizelgeleme esnasında değeri sabit kalanlar (durağan – static) ve çizelgelemenin değişik aşamalarında değişik değerler alabilenler (devingen – dynamic) şeklindedir. Örneğin, SPT durağan bir kural iken, MinSlack devingen bir kuraldır. Diğer bir sınıflandırma da, salt faaliyet bilgisini kullanan öncelik kurallarının yerel; bunun dışında daha geniş bilgi kullananların ise, global olarak sınıflandırılmasıdır. Buna göre MinSlack, yerel iken; GRPW, global bir öncelik kuralıdır. Çizelgenin oluşturulması esasına göre iki ayrı yöntem vardır: (i) Paralel çizelgeleme ve (ii) seri çizelgeleme.

6.4.2.1 Paralel çizelgeleme

En fazla faaliyet adedi (J) kadar aşamadan oluşmaktadır. Her aşamada, bir veya daha fazla faaliyet kısmi çizelgeye atanır. Uygulama esnasında faaliyetler dört küme içinde yer almaktadır. Bitmiş faaliyetler kümesi, atanmış; fakat henüz tamamlanmamış süregiden faaliyetler kümesi, kaynak kısıtlarını ihlal etmeyen ve öncülleri bitmiş faaliyetlerin oluşturduğu karar kümesi ve bunların dışında kalan faaliyetlerin oluşturduğu kalan faaliyetler kümesi. Her aşamada çizelgeleme anı, süregiden faaliyetler kümesindeki faaliyetlerden en erken biten faaliyet(ler) tarafından belirlenmektedir. Bu faaliyet(ler), bitmiş faaliyetler kümesine dahil edilir. Karar kümesi mevcut duruma göre, yeniden belirlenir. Kullanılan öncelik kuralı uygulanarak, karar kümesinden yeni atanacak faaliyet seçilir ve atanır. Atama döngüsü, karar kümesinde atanacak faaliyet kalmayana veya mevcut kaynak kullanımına göre kaynak kısıtını ihlal etmeyen faaliyet kalmayana kadar sürdürülür. Tüm faaliyetler henüz çizelgelenmemişse adı geçen kümeler güncellenerek yeni bir aşamaya geçilir ve yeni bir çizelgeleme anı saptanarak algoritmaya devam edilir. Paralel çizelgeleme yönteminde, çizelgeleme anında süregiden faaliyetler kümesinin elemanları karar kümesine dahil edilirler. Buna göre, algoritmanın uygulanmasında bir faaliyet aynı anda birden fazla kere atanabilir.

6.4.2.2 Seri çizelgeleme

Sezgisel çizelgelemede üç küme tanımlanmıştır. Bitmiş faaliyetler kümesi, öncülleri bitmiş faaliyetler kümesinde yer alan faaliyetlerin oluşturduğu karar kümesi ve bunların dışında kalan faaliyetlerin oluşturduğu kalan faaliyetler kümesi. Her aşamada bir faaliyet atanır. Buna göre algoritma faaliyet adedi (J) kadar aşamada sona erer. Her aşamada karar kümesi içinden bir faaliyet öncelik kuralı kullanılarak seçilir ve öncüllük ve kaynak kısıtları bakımından olurlu olduğu en erken anda atanır. Adı geçen üç küme güncellendikten sonra, yeni bir aşamaya geçilerek algoritma, atanacak faaliyet kalmayana kadar sürdürülür. Sezgisel yöntemlerin daha ayrıntılı irdelenmesi Kolisch (1995), Kolisch ve Hartmann (1999), Hartmann ve Kolisch (2000) tarafından verilmektedir. Paralel çizelgelemenin ertelemez çizelgeler türettiği; seri çizelgemenin ise aktif çizelgeler türettiği gösterilmiştir (Kolisch, 1996). Daha önce de belirtildiği gibi düzgün bağımlı performans ölçütleri için aktif çözümler kümesi, mutlaka eniyi çözümü içermekle birlikte; ertelemez çözümler kümesi, eniyi çözümü içermeyebilmektedir. Bu açıdan bakıldığında seri çizelgeleme yöntemi eniyi çözümü bulmaya daha yakın durmaktadır.

6.4.2.3 Öncelik kuralları

Öncelik kurallarına dayalı çizelgeleme kullanılarak ve birden çok geçiş (multi pass) uygulanarak, birden çok çizelge üretmek için çeşitli yollar denenmiştir. Bunlardan bir tanesi, türetilen her çizelge için ayrı bir öncelik kuralı uygulamaktır. Diğer bir yaklaşım ise, her aşamada kullanılacak öncelik kuralının rassal olarak kararlaştırılmasıdır (Storer vd., 1992). Bu iki yaklaşıma da dahil olmayan; ancak birinci yaklaşıma yakın olan bir sezgisel öncelik kuralı da Ulusoy ve Özdamar (1989) tarafından önerilmiştir. Ağırlıklandırılmış kaynak kullanım oranı ve ardıllık (weighted resource utilisation ratio and precedence - WRUP) olarak nitelendirilen bu sezgisel kuraldaki ağırlık parametresine [0,1] aralığında değişik değerler atanarak, çeşitli çizelgeler türetilmektedir. Bu yaklaşım çeşitli sezgisel kuralların içbükey kombinasyonunun kullanılmasına bir örnektir.

6.4.2.4 Yaklaşık çözüm (YÇ) yöntemleri

Yaklaşık çözüm yöntemlerinin geliştirilmesinde genellikle şu ilkeler gözetilmektedir:

- ◆ Öncelik kuralı yöntemlerine, kısmen sezgisel unsurlar katılarak, en iyi olmasa da, iyi bir çözüm süratli bir şekilde elde edilebilmektedir.
- ◆ Öncelik kuralı yöntemleriyle elde edilen olurlu bir çözüm üzerinde değişiklikler yapılarak daha iyi bir çözüm ortaya çıkartılabilmektedir.

YÇ yöntemleri içinde yer alan diğer çalışmalar arasında, “iterasyon yöntemleri” belirtilebilir. Bu yöntemlerle, ilk aşamada bir “ÖK” yönteminin uygulanmasıyla elde edilen başlangıç çözümüne dayanılarak işlemlerin zaman konumları, kaynak kısıtları altında belirlenebilen bolluk sınırları içinde değiştirilmekte ve bir dizi iterasyonla amaç fonksiyonu değerini iyileştiren çizelge düzeltmeleri gerçekleştirilmektedir. Sezgisel yöntemlerde ise hedef, problemin iyi bir çözümünün elde edilmesi olduğuna göre, bu yöntemlerin ve nasıl uygulanacaklarını belirleyen yaklaşımların geliştirilmesinde, öncelikle şu sorunun ele alınması gerekir: Hızlı, ancak güvenilirliği o ölçüde düşük olan sezgisel yöntemler nasıl bir yaklaşımla ve hangi koşullarla uygulanmalıdır ki, problemlerin “olurlu çözümler uzayının çok küçük bir kısmı taranarak” elde edilmesi mümkün olsun? Genel olarak iş sıralama, atölye veya akış tipi üretim çizelgeleme, montaj hattı dengeleme gibi benzer problemlerde gündemde kalan bu soruya tatminkar bir yanıt bulabilmek amacıyla, deterministik yapıda veya temin bazı probabilistik unsurlar içeren ve esas itibariyle bir yöntemin bazı parametreler değiştirilerek tekrarlı uygulanması önerilmektedir:

6.4.2.5 Rot ve MinSlack algoritması

KKPÇ probleminin büyüklüğü arttıkça, doğrudan optimizasyon yöntemiyle makul sürelerde optimal veya optimale yakın sonuçlar elde edilmesi mümkün olamamaktadır. Bu durum literatürde tespit edilmiş olup, önerilen çeşitli çözüm yöntemleri mevcuttur. Sezgisel algoritmaların kullanılması yöntemi bunlardan birisidir. Bu çalışmada, büyük boyutlu problemlerin makul sürelerde çözülebilmesini sağlamak amacıyla, iyi bir başlangıç çözümün sezgisel yöntemlerle bulunması yaklaşımı kullanılmıştır. Böylece, çok büyük olan çözüm uzayında, makul sürelerde, optimal veya optimale yakın sonuçlar elde edilmesi hedeflenmiştir. Çözüm uzayının aranmasında, iyi bir başlangıç çözümü elde edilmesi amacıyla, MinSlack ve Rot öncelik kurallarını uygulayan algoritmalar kullanılmıştır.

Elsayed, projenin tamamlanma zamanını en aza indirgeyen ve kaynakların faaliyetlere dağılımını sağlayan sezgisel bir algoritma geliştirmiştir. Her bir kaynak türünün şebekedeki dağılımı ROT (Resources Over Time) kriterine bağlıdır. ROT değeri, her bir faaliyet kaynakları toplamını, faaliyetin süresine böldükten sonra, herhangi bir yol üzerindeki şebeke aracılığıyla bir faaliyetin kontrol ettiği maksimum değerdir. Kısacası; şebekenin patikaları üzerinde, kaynaklarda zaman aşımına neden olan faaliyetlerin, maksimum değerine sahip olanının önceliklendirilmesi olarak da tanımlanabilir.

ROT ve MinSlack Algoritması aşağıda açıklanacaktır:

Adım 1. Tüm faaliyetlerin ROT değerleri bulunur ve büyükten küçüğe sıralanarak, kaynakların tahsis edilebilecekleri öncelikler belirlenmiş olur. Minslack değerleri ise, KYM ve B-KYM'den elde edilir ve küçükten büyüğe doğru sıralanarak, kaynak tahsisi için gerekli öncelikler belirlenmiş olur. Bundan sonraki adımlar, her iki yöntem için de aynıdır.

Adım 2. Bir faaliyetin kaynakları dikkate alınmadan hesaplanan en erken başlangıç zamanını gösteren “TEARL” değeri hesaplanır. Bir başka deyişle, bir faaliyetin TEARL değeri; bütün acil öncül faaliyetler için, en geç bitiş zamanıdır.

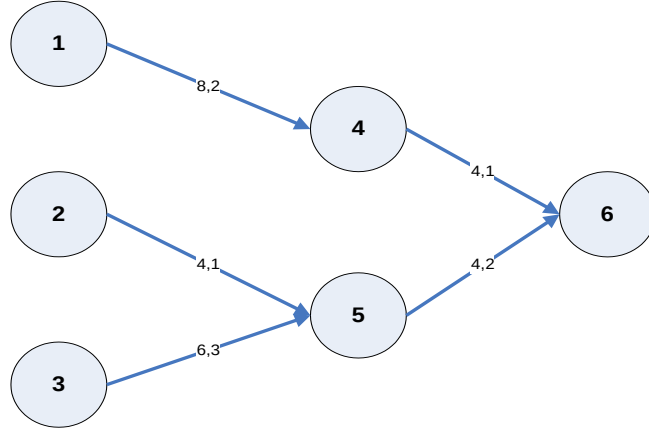
Adım 3. TSTART değeri ise; kaynaklar, zaman ve öncelik kısıtları dikkate alındığında bir faaliyetin gerçek başlangıç zamanı olarak tanımlanabilir. Eğer kaynak kısıtı dikkate alınmaz ise, TEARL=TSTART değeri olacaktır.

Adım 4. TFIN değeri ise, bir faaliyetin gerçek tamamlanma süresi olarak tanımlanabilir. TSTART değeri ile söz konusu faaliyetin işlem zamanının toplamıyla bulunur.

Adım 5. TNOW, içerisinde bulunulan zaman sayıcı olarak tanımlanabilir. O anda yapılan faaliyetlerin zamanıdır. Başlangıçta TNOW değeri, 0 olarak alınır.

6.4.2.6 ROT algoritmasının örnek problem üzerinde gösterilmesi

Şimdi de 5 işli, tek kaynaklı aşağıdaki örnek problem ele alındığında; 1'inci kaynaktan, 4 birim elde edilebilir. İşlem süreleri ve kaynak gereksinimleri ise aşağıdaki Şekil 6.1'de verilmiştir:



Şekil 6.1 : Örnek problem Şebeke Gösterimi.

Verilenlere göre problemin çözüm prosedürünü incelersek; Rot değerlerinin hesaplanması ve kaynak kısıtlarını dikkate almaksızın kritik yolu ve her bir faaliyetin en erken başlama ve tamamlanma süreleri bulunur.

$$ROT(1) = \text{Max} \{ 2/8+1/4 \} = 2/4 = 0,5$$

$$ROT(2) = \text{Max} \{ 1/4+2/4 \} = 3/4 = 0,75$$

$$ROT(3) = \text{Max} \{ 3/6+2/4 \} = 1$$

$$ROT(4) = \text{Max} \{ 1/4 \} = 0,25 \quad ROT(5) = \text{Max} \{ 2/4 \} = 0,5$$

Her bir faaliyet için ROT'un, şebeke dahilinde herhangi bir yol için verdiği maksimum değerler bulunur. Faaliyetler, ROT değerlerine göre azalan şekilde sıralanır. Çizelge 6.1' de görüldüğü gibi, en uzun süreli faaliyet 1. sıraya konulunca faaliyetler arası bağlar kopmuştur. TNOW = 0 ve eldeki kaynak miktarı 4 ile başlanır. TNOW=0'da başlayabilen faaliyetler, TEARL değerleri dikkate alınarak 3-5, 2-5 ve 1-4 olup, ROT değerleri tabloda verilmiştir.

Başlangıçta kaynaklardan ikisi, 3-5 faaliyetine ve kaynaklardan biri, 2-5 faaliyetine tahsis edilir. (her ikisi de toplam 4 kaynak gerektirmektedir.) Bu faaliyetler için gerçek başlama zamanı, TSTART olarak 0 girilir. Bu faaliyetler için TFIN, TSTART ve faaliyetin süresi toplanarak hesaplanır. Bu sırasıyla 3-5 ve 2-5 faaliyetleri için TFIN=6 ve TFIN=4' tür. Elde hiç kaynak kalmadığı için 0 başlangıç zamanında başlayabilecek olan 1-4 faaliyetine başlanamamış olur. Eldeki TFIN'in minimumu olan TNOW' un yeni bir değerine geçilir. Bu iterasyon da eldeki TFIN 6 ve 4 ise ;TNOW değeri olarak, TNOW = min {6,4}=4 olarak düzenlenir.

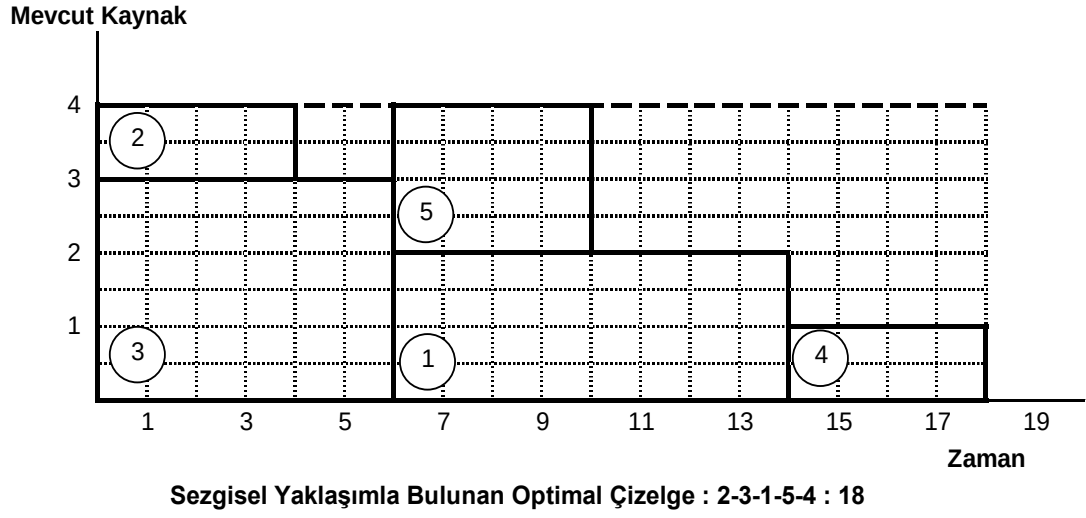
Çizelge 6.1 : Rot Algoritması Çözüm Algoritması.

Faaliyetler	3	2	1	5	4
	3 - 5	2 - 5	1 - 4	5 - 6	4 - 6
ROT	1	0,75	0,5	0,5	0,25
Süre	6	4	8	4	4
Gerekli Kaynak	3	1	2	2	1
TEARL	0	0	0	6	8
TSTART	0	0	6	6	14
TFIN = TSTART + Süre	6	4	14	10	18
TNOW	0	4	6	10	14
Mevcut Kaynaklar	4 1 0	1	4 2 0	2	4
İzin Verilen Faaliyet	3 - 5, 2 - 5, 1 - 4	1 - 4, 5 - 6	1 - 4, 5 - 6		1 - 4
İterasyon Sayısı	1	2	3	4	5

TNOW=4 iken, 2-5 faaliyetinin tamamlanmasından sonra açığa çıkan 1 kaynak mevcuttur. 2-5 faaliyetinin tamamlanmasından sonra diğer faaliyetler çizelgelenemezler; ancak geriye 1-4 faaliyeti çizelgelenmek üzere beklemekte olup TNOW= 4'te 1-4 faaliyeti de başlayamaz; çünkü elde yeterli miktarda kaynak mevcut değildir.

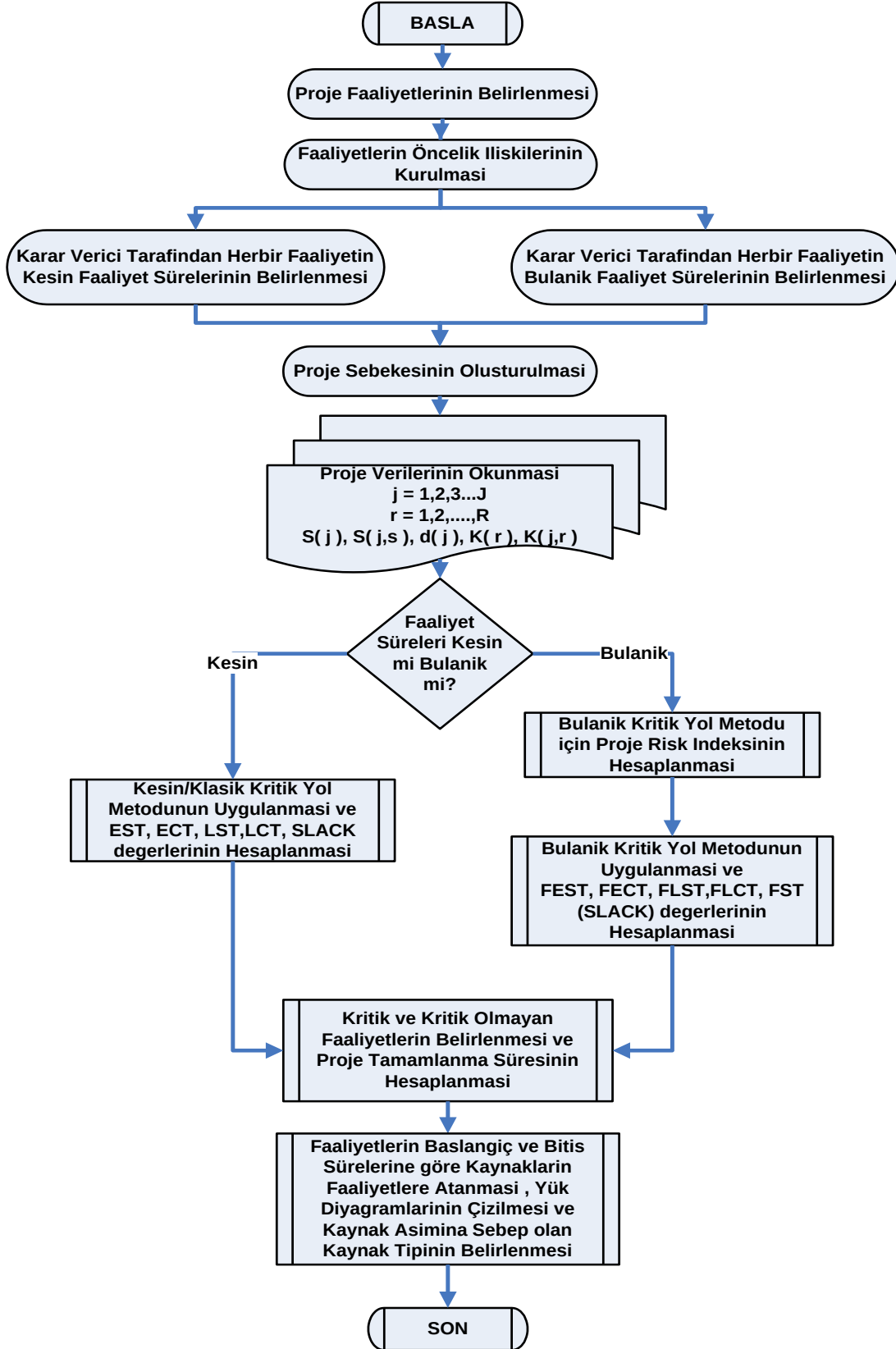
TNOW=6 ise, elde dört kaynak mevcuttur; üç birim kaynak, 3-5 işinin tamamlanması sonucu elde edilen ve 1 birim kaynak da daha önce serbest kalan kaynaktır. ROT sıralamasına, TEARL'ye ve öncelik ilişkisine bağlı olarak yapılan faaliyetlerin listesi belirlenir. Daha önce çizelgelenebilecek ilk faaliyet olan 1-4 faaliyeti için, 2 kaynak tahsis edilerek, 1-4 işinin 6 zamanda başlaması sağlanır. Geriye kalan 2 kaynak ile de, 3-5 ve 2-5 faaliyetlerinin tamamlanmasının akabinde çizelgelenebilecek faaliyet olan 5-6 faaliyeti, 6. zamanda başlatılabilir.

TNOW= min {14,10} = 10 iken, her ne kadar 5-6 faaliyetinin tamamlanması sonucu iki birim kaynak açığa çıkmış olsa da; henüz çizelgelenmemiş olan 4-6 faaliyeti, 1-4 faaliyetinin tamamlanmasından sonra başlayabileceğinden; TNOW=10 iken, 4-6 faaliyeti başlatılamaz. TNOW = 14 iken, 1-4 faaliyeti de tamamlanmış olacağından; 2 birim kaynak daha ortaya çıkarak, toplam 4 birim kaynak mevcut olacaktır. 1-4 faaliyeti tamamlanmış olacağından TNOW=14 iken, 1 birim kaynağa ihtiyaç duyan 4-6 faaliyeti başlatılır. Tüm faaliyetler çizelgeleninceye kadar iterasyon tekrar edilir. Kaynak kısıtları altında söz konusu projenin minimum tamamlanma zamanı Max TFIN değeri olarak 18 birim zamandır. Şekil 6.2'deki örnek problem için bulunan sonuç, optimal çözüm olsa da, ROT kriterine bağlı olarak faaliyetlerin kaynak dağılımı, optimalliği garantilemez. ROT kriteri kullanımı yerine, bu projeyi tamamlamak için daha az süreli bir başka kriteri kullanmak faydalı olacaktır.

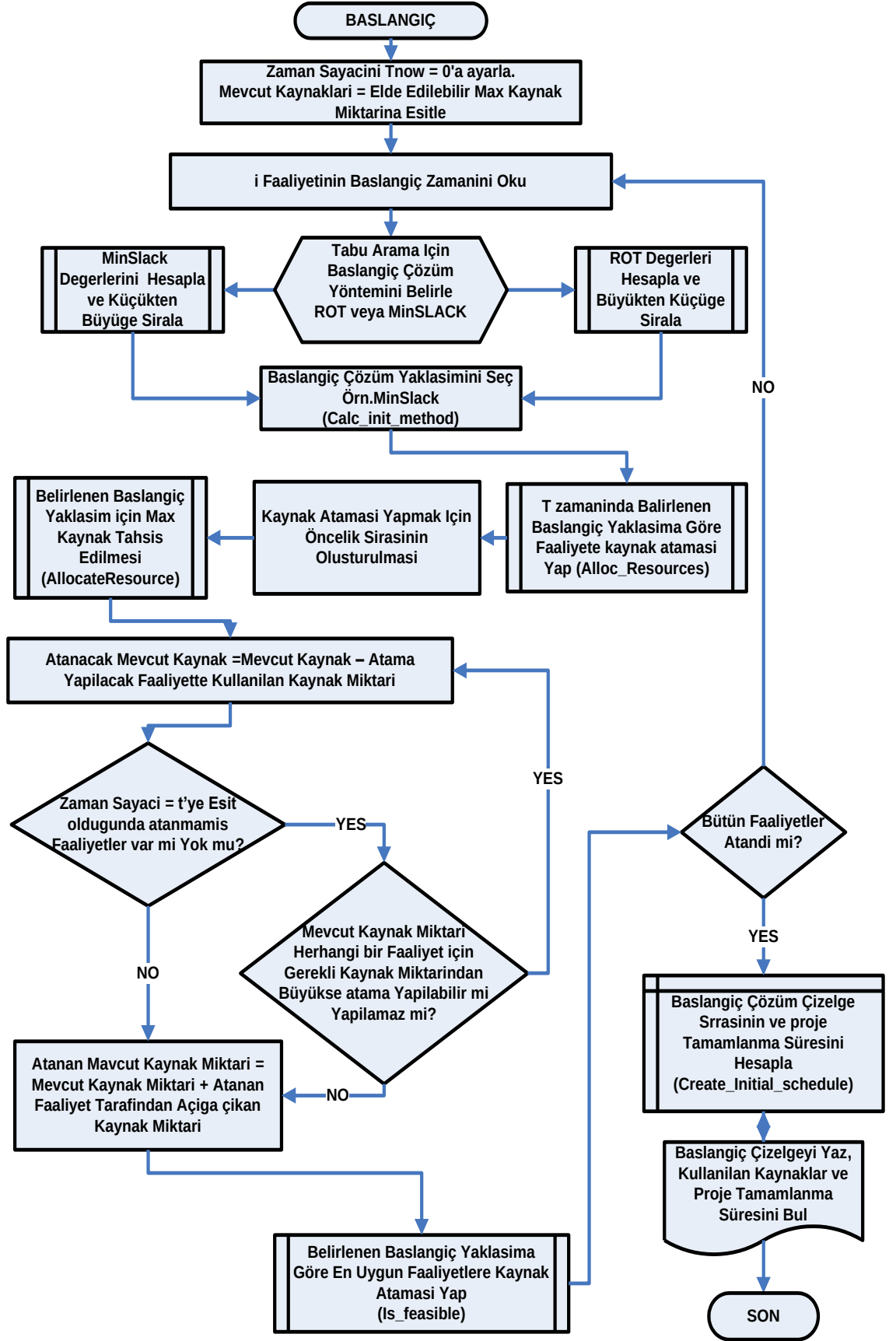


Şekil.6.2 : ROT Kriteri Faaliyet-Kaynak-Zaman Çizelgesi.

Modelde kullanılan TA için başlangıç çözümün bulunması Şekil 6.3’de, ROT ve MinSlack algoritmalarının akış diyagramı Şekil 6.4’de sunulmuştur.



Şekil 6.3 :Tabu Arama İçin Başlangıç Çözümün Bulunması.



Şekil 6.4 : ROT ve MinSlack Algoritması Akış Diyagramı.

6.4.2.7 B-TM-KKPÇP çözümü için sezgisel bir yöntem kanguru algoritması

Kanguru algoritması literatürde, “Pollard’ın Kangurusu” veya “Pollard’ın Rho Algoritması” veya “Pollard’ın Lambda Algoritması” olarak da bilinmektedir. Kanguru Algoritmasını (KA) ilk defa, 1978 yılında Pollard ortaya attı. Rho Algoritması veya Lambda Algoritması ismi; bu algoritmaların görüntülerinin Yunan harflerindeki λ ’ya olan benzerliğine işaret etmektedir. KA, “Tavlama Benzetimi”nden esinlenen; fakat çok daha farklı bir arama stratejisi olan, stokastik kökenli bir yaklaşıklık yöntemidir. Günümüzde “Kesikli Algoritma Problemleri”nin (Discrete Logarithm Problem) çözümünde bilinen en iyi yöntem, Pollard’ın Paralel Rho ve KA’larıdır. Kanguru metodu, amacı $f(u)$ gibi bir fonksiyonu minimize etmek olan, tekrarlamalı bir sürece yerleştirilerek uygulanmaktadır. Ele alınan problemdeki mevcut çözüm, u ’yu rastsal seçimle belirlediği daha iyi bir çözüm olan komşusu $N(u)$ ile değiştirmektir. Algoritma, “A” defa bu işlemi tekrarlayarak sonucu iyileştirmeye çalışır. Burada “A” algoritmanın bir parametresidir. Eğer çözümde iyileşme olmuyorsa, “zıplama” işlemiyle yerel minimumun etkisinden kaçılır. Bu aşamada, mevcut çözümde bir iyileşme olması zorunlu değildir. Bu işlem, $N'(u)$ gibi farklı bir komşuluk tanımı kullanabilir. Durma kriteri, maksimum iterasyon sayısı veya amaç fonksiyonunun alt sınırı olabilir. Tekrarlamalı süreçte karşılaşılan en iyi çözüm u^* , hafızaya alınır. İşlem sonunda u^* , algoritma tarafından bulunan “optimal” sonuçtur. Komşu $N(u)$, u ’dan i ve $i+1$ ’de bulunan işlerin permütasyonla elde edilen u' çözümlerinin bir kümesidir. Örneğin, Eğer $u=[4\ 1\ 3\ 5\ 2]$ ise, $N(u)=\{ [1\ 4\ 3\ 5\ 2], [4\ 3\ 1\ 5\ 2], [4\ 1\ 5\ 3\ 2], [4\ 1\ 3\ 2\ 5], [2\ 1\ 3\ 5\ 4] \}$ kümesini göstermektedir. Mevcut çözümde yeni bir iyileştirmenin mümkün olmadığı durumda u , “zıplama” işlemi yapılarak u' çözümüyle değiştirilir. Paralel KA’sı, başlangıç noktası $e0=gb$ ile gösterilen bir evcil kanguru olan E ve başlangıç noktası $v0=h$ ile gösterilen bir vahşi kanguru V ile çalışır. g ’nin üs olduğu durumlarda; V , bilinmeyen bir x noktasından başlarken; E , bilinen bir tamsayı ile çalışmaya başlar. Bir dizi zıplama yapan kangurular, karışık bir fonksiyon oluşturur. Zıplamalar, mutlak bir sayı olduktan sonra, evcil kanguru durur ve fonksiyonda dolaşımını bitirdiği noktaya, “EM” ile gösterilen bir tuzak kurar. Daha sonra vahşi kanguru, aynı fonksiyon üzerinde gezinmeye başlar. Bilinen bir değerden yola çıkan E ’nin tam değeri bilinir; fakat bilinmeyen x noktasından fonksiyona giren V ’nin durumu bilinmez. Bu yüzden V vahşi olarak adlandırılır. V ’nin fonksiyondaki her zıplayışından sonra tuzağa düşüp

düşmediği kontrol edilir. V 'nin tuzağa düşmesi durumunda, VN olarak kabul edilir ve $EM = VN$ ile eşitlenir. Diğer yandan; mutlak bir sayı değerinden sonra, vahşi kangurunun zıplamaları durabilir ve yeni bir vahşi kanguru, o noktaya kadar görülen en küçük zıplama aralığından yola çıkar. Eğer V , E 'nin tuzağına düşüyorsa bunun anlamı, benzer noktalardan geçen iki kangurunun, daha önceden fonksiyon üzerinde karşılaşmış olmalarıdır. Vahşi kanguru, belirli bir zıplama sayısından sonra tuzağa düşmeli ya da durmalıdır. Eğer vahşi kanguru, evcil kangurunun tuzağına düşüyorsa daha önce bu iki kangurunun aynı noktada denk geldiği sonucu anlaşılır.

6.4.2.8 Paralel kanguru algoritması adımları

1.Adım: Öncelik ilişkilerine göre seçilebilecek faaliyetleri belirle. (Öncülü olmayan veya öncülü tamamlanmış olan faaliyetler.)

2.Adım: Kaynak çakışma kontrolünü yap.

a. Seçilebilecek faaliyetlerin kaynak ihtiyacı mevcut kaynak miktarından fazla ise, kaynak çakışması var demektir. 3.Adıma git.

b. Seçilebilecek faaliyetlerin kaynak ihtiyacı mevcut kaynak miktarına eşit veya az ise, çakışma yoktur. Bu faaliyetleri seç. 4.Adıma git.

3.Adım: Seçilecek faaliyeti belirle.

a. Evcil kanguru için rastsal olarak bir faaliyet seç.

b. Vahşi kanguru için evcil kangurunun seçmediği bir faaliyeti rastsal olarak seç.

4.Adım: Seçilen faaliyetleri çizelgele.

Seçilen faaliyetlerin öncüllerinin tamamlanma zamanlarına göre başlama zamanlarını ve tamamlanma zamanlarını hesapla.

5.Adım: Algoritmayı sonlandırma kontrolünü yap.

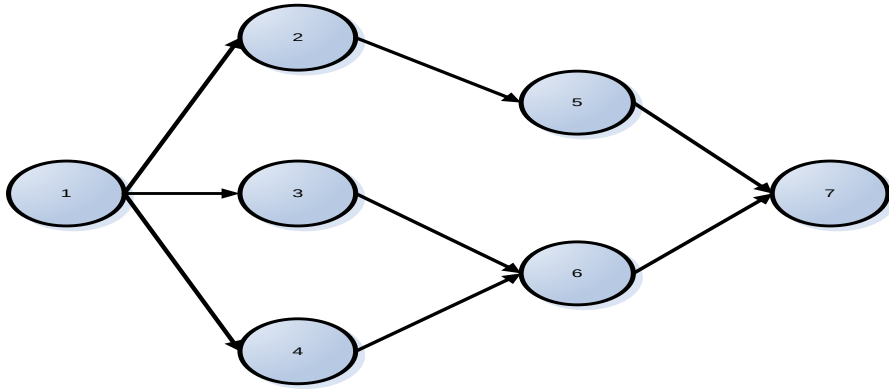
a. Eğer bütün faaliyetler çizelgelenmişse algoritmayı sonlandır.

b. Eğer bütün faaliyetler çizelgelenmemişse 1.Adıma dön.

6.4.2.9 Paralel kanguru algoritması örnek 1

Paralel kanguru algoritması ile çözülen problem, Wang (2004, 2010)'ın çalışmasından alınmıştır. Örnek proje, yedi faaliyetten oluşmaktadır. Bu faaliyetlerin

öncelik ilişkileri, süre ve kaynak ihtiyaçları, Şekil 6.5'te ve Çizelge 6.2'de sunulmuştur.



Şekil 6.5 : Örnek problemin öncelik ilişkileri diyagramı.

Projenin bulanık en erken başlama zamanı, (0, 1, 1, 1) olduğu, sadece bir çeşit kaynağın bulunduğu ve bu kaynağın bulunabilirliğinin 2 olduğu varsayılmıştır.

Çizelge 6.2 : Örnek problemin faaliyet süreleri ve kaynak ihtiyaçları

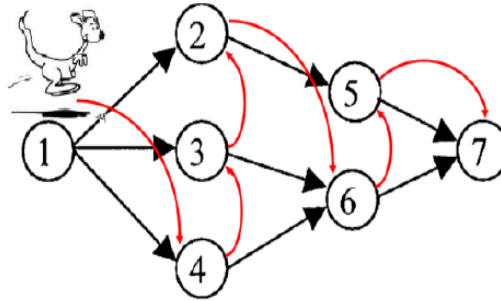
Faaliyetler	Bulanık Faaliyet Süreleri	Kaynak Gereksinimi
a_1	(5, 7, 8, 10)	1
a_2	(8, 10, 15, 18)	1
a_3	(14, 17, 20, 24)	1
a_4	(9, 12, 16, 20)	1
a_5	(3, 5, 7, 9)	1
a_6	(5, 9, 12, 15)	1
a_7	(20, 24, 28, 33)	1

Başlangıçta öncelik ilişkilerine göre seçilebilecek tek faaliyet, hiçbir öncülü olmayan “a1” faaliyetidir. 2.Adımda; kaynak ihtiyacı, (1 birim) mevcut olan kaynaktan (2 birim) daha az olduğu için, a1 faaliyeti seçilir ve 4.Adıma gidilir ve bu faaliyetin çizelgelemesi yapılarak başlama ve tamamlanma zamanları hesaplanır. 5.Adımda; bütün faaliyetlerin (7 faaliyet) çizelgelenmediği görülür ve 1.Adıma geri dönülür. 2.Aşamada; tekrar öncelik ilişkilerine göre seçilebilecek faaliyetler (a2, a3, a4) belirlenir. 2.Adımda; bu faaliyetlerin kaynak ihtiyacının (3 birim) mevcut kaynaktan (2 birim) daha fazla olduğu görülür ve 3.Adıma gidilir. 3. adımda; evcil kanguru rastsal olarak bir faaliyeti (a4) seçer; vahşi kanguru, rastsal olarak evcil kangurunun

seçmediği bir faaliyeti (a_2) seçer. 4.Adımda; seçilen faaliyetlerin öncüllerinin tamamlanma zamanlarına göre başlama zamanları ve tamamlanma zamanları hesaplanır. 5.Adımda; bütün faaliyetlerin (6 faaliyet) çizelgelenmediği görülür ve 1.Adıma geri dönülür. Bu işlem, tüm faaliyetler çizelgeleninceye kadar devam eder. Bulunan sonuçlara göre vahşi kanguru, 1-2-4-5-3-6-7 faaliyet sırasıyla çizelgeleyerek (53,70,85,103) süresinde projeyi tamamlarken (Çizelge 6.3); evcil kanguru, 1-4-3-2-6-5-7 sırasıyla Şekil 6.6'da görüldüğü gibi çizelgeleyerek (45, 59, 75, 91) süresinde projeyi tamamlamaktadır (Çizelge 6.4).

Çizelge 6.3 : Vahşi kanguru ile elde edilen çizelge.

İterasyon	Alternatifler	Seçilen Faaliyet	Kaynak İhtiyacı	Mevcut Kaynak	Başlama Zamanı	Bitiş Zamanı
1	a_1	a_1	1	2	(0 1 1 1)	(5 8 9 11)
2	a_2 a_3 a_4	a_2	3	2	(5 8 9 11)	(13 18 24 29)
3	a_3 a_4 a_5	a_4	3	1	(5 8 9 11)	(14 20 25 31)
4	a_3 a_5	a_5	2	1	(13 18 24 29)	(16 23 31 38)
5	a_3	a_3	1	1	(14 20 25 31)	(28 37 45 55)
6	a_6	a_6	1	1	(28 37 45 55)	(33 46 57 70)
7	a_7	a_7	1	2	(33 46 57 70)	(53 70 85 103)



Şekil 6.6 : Evcil kanguru ile belirlenen çizelge yolu.

Çizelge 6.4 : Evcil kanguru ile elde edilen çizelge.

İterasyon	Alternatifler	Seçilen Faaliyet	Kaynak İhtiyacı	Mevcut Kaynak	Başlama Zamanı	Bitiş Zamanı
1	a_1	a_1	1	2	(0 1 1 1)	(5 8 9 11)
2	a_2 a_3 a_4	a_4	3	2	(5 8 9 11)	(14 20 25 31)
3	a_2 a_3	a_3	2	1	(5 8 9 11)	(19 25 29 35)
4	a_2 a_6	a_2	2	1	(14 20 25 31)	(22 30 40 49)
5	a_5 a_6	a_6	1	1	(19 25 29 35)	(24 34 41 50)
6	a_6	a_5	1	1	(22 30 40 49)	(25 35 47 58)
7	a_7	a_7	1	2	(25 35 47 58)	(45 59 75 91)

6.4.3 Diğer sezgisel yöntemler

6.4.3.1 Yerel kısıt analizi

Yerel kısıt temelli analiz (LCBA) yöntemi (Özdamar, 1991), her çizelgeleme anında faaliyetlerin kısıtlarını yeniden değerlendirir. Bu kısıtlar, kaynakların sınırlı olmasından kaynaklanan kaynak kısıtları ve öncüllük ilişkilerinden kaynaklanan kısıtlardır. Her çizelgeleme anında kısıtlar, güncellenir. Bu kısıtlar çizelgelenebilir faaliyetler kümesini belirler. Çizelgelenebilir faaliyetler kümesini oluşturan faaliyetleri, atama önceliğine göre gruplandıran bir dizi temel kural geliştirilmiştir. Temel kurallar, bu kümenin elemanları arasında en yüksek atama önceliğine sahip faaliyetleri belirlerler. Bu faaliyetler, atanmamaları halinde, poje süresinin uzamasına neden olacak olan faaliyetlerdir. Temel kurallar, bu çizelgeleme anında atanmaması gereken faaliyetleri de ayrıca belirleyerek kümeden dışlar ve böylece çizelgelenebilir faaliyetler kümesini daraltır. En yüksek öncelikli gruptaki faaliyetlerin hepsinin çizelgelenmesinden sonra, hala yeterli kaynak varsa, çizelgelenebilir faaliyetler kümesinden dışlanmamış faaliyetler, çizelgelenmek üzere ele alınır.

6.4.3.2 Yerel arama (YA)

Makina çizelgeleme problemlerinin çözümünde oldukça yoğun olarak kullanılan yerel arama yöntemlerinin proje çizelgelemede bir örneği, Sampson ve Weiss (1993) tarafından verilmiştir. Uygulanacak yerel arama yöntemi; çözümün gösterimi, komşu çözüm kümesinin tanımı ve bu kümenin taranma yöntemi ile tanımlanır. Eldeki çözümün öncüllük ilişkileri açısından olurlu olması istenir; ancak çözüm, kısıtlı kaynaklar bakımından olurlu olmayabilir. Kısıtlı kaynakların ihlali, belirli bir ceza yöntemi ile yöntemin proje süresinin enazlanması olan amaç işlevine dahil edilir. Faaliyetler sıra ile ele alınarak; her faaliyet j için, yukarıda tanımlanan öteleme vektörü V 'nin j . elemanının, $V(j)$, değeri sırası ile 0, 1, 2, ..., m_j alınarak, her değer için yeni bir çözüm elde edilir. J faaliyetinin komşu çözüm kümesi tarandıktan sonra, elde edilen en iyi çözüm, mevcut çözüm olarak kabul edilerek, yeni bir V vektörü oluşturulur ve sıradaki faaliyet için bu döngü tekrarlanır. Bütün faaliyetler için bu döngünün tekrarlanması, bir iterasyon oluşturur. Önceden kararlaştırılmış bir iterasyon sayısına erişildikten sonra, algoritma sonlandırılır.

6.4.3.3 Budanmış dal-sınır yöntemi

Budanmış dal-sınır (truncated branch and bound) yönteminde, tüm olası dallanma olanakları kullanılmaz. Dallanma, sezgisel kural(lar) marifeti ile daraltılır ve böylece birerleme ağacının çok daha az sayıda düğüm içermesi ve daha kısa sürede bir sonuca ulaşılması sağlanır. Ulaşılan sonucun eniyi olması garantisi yoktur; iyi bir çözüm elde edilmeye çalışılır. Alvarez- Valdes ve Tamarit (1989) tarafından önerilen budanmış dal-sınır yönteminde her düğümden sadece bir düğüme dallanılır.

6.4.3.4 Dal sınır yöntemi

Birçok çözüm yöntemi, proje planındaki öncelik ilişkilerinden elde ettikleri karar ağaçlarında arama yapmaktadır. Ağacın kökü, ilk göreve karşılık gelmektedir. Ağacın ikinci seviyesinde ilk görev çizelgelendikten sonra, çizelgelenebilecek görevler kümesi yer alır ve böylece sonuna kadar devam eder. En sonunda oluşan ağaç, öncelik açısından uygun görev sıralarının kümesini göstermektedir. Alternatif olarak, eğer ağaç oluşturma algoritması kaynak kısıtlarını göz önünde bulunduruyorsa, görevlerin sıraları direkt olarak çizelgelenebilir. Arama işleminde ağacın kökünden yaprağına kadar sıralamalar ele alınır. En iyi kök-yaprak yolu bulunana kadar, arama devam eder. Ancak ağaç, faaliyet sayısına bağlı olarak çok hızlı bir şekilde büyümektedir. Öncelik ilişkilerine göre eklenen her yeni görev, ağaca birçok dal ekleyebilir. “Dal Sınır” yaklaşımı; ağaca ilk görevden başlayarak her bir görev için öncelik ve kaynak kısıtlarına göre çizelgelenebilecek bir düğüm ekler. Böylece ağacı oluşturur. “Dal Sınır” yöntemi geniş problemleri çözememektedir. Sınırlama tekniklerinde önemli gelişmeler yapılmış olsa da, Dal Sınır yöntemleri halen, yüzden daha az faaliyetle sınırlıdır.

6.4.3.5 Ayırkaç ayırıt yöntemi

“Ayırkaç Ayırıt” (disjunctive precedence arc) yönteminin ana fikri; öncüllük açısından olurlu olmakla birlikte, kaynak kısıtları nedeni ile birarada atanamayan faaliyetlerin en küçük kümesindeki faaliyetler arasında, gerçekte olmayan öncüllük ilişkileri yaratarak bu kümeyi parçalamaktır. Bu işlem, yaratılan öncüllük ilişkisinin proje seriminin AON gösteriminde, iki faaliyet arasına bir ayırkaç ayırıt eklenmesi ile gerçekleştirilir. Faaliyetlerin en erken atanabileceği, hem öncüllük hem de kaynak

kısıtları açısından olurlu yeni faaliyet kümelerinin oluşturulması hedeflenir. Alvarez-Valdes ve Tamarit (1989); en küçük kümeyi parçalamak için dört ayrı yol önermişlerdir. Bunların en başarılı olanı; en küçük eleman sayısına sahip kümeler arasından birinin rasgele seçilmesi ve bu kümedeki faaliyetler arasına proje süresini en az artıracak ayırkaç ayırıtın eklenmesi kuralıdır. Bu iteratif uygulama, gelişme sağlanamayana kadar devam etmektedir.

6.4.4 Sezgisel kuralların etkinliği

“Sezgisel Kuralların Etkinliği” üzerine çok sayıda çalışma yapılmıştır. Bu çalışmaların bazılarında, proje seriminin yapısal özelliklerinin de etkisi araştırılmıştır. Son yıllardaki çalışmalar, genel olarak metasezgisel yöntemlerin üstünlüğünü ortaya koymuştur. İlginç bir gözlem de, **metasezgisel algoritmaların 1000 çözüm düzeyinde görülen üstünlüğünün, 5000 çözüm düzeyinde daha da artmasıdır.** Bunun makul bir açıklaması, sezgisel karar kurallarını kullanan algoritmalarından farklı olarak, meta-sezgisel algoritmaların yeni çözüm türetirken daha önce türetmiş olduğu çözümlerin oluşturduğu bilgi birikiminden yararlanmasıdır. Meta-sezgisel algoritmaların en iyileri, sezgisel öncelik kurallarını kullanan algoritmaların en iyilerinden daha iyi sonuçlar türetiyor olmakla birlikte sezgisel öncelik kuralları yine de önemlidir. Sezgisel öncelik kuralları, çok büyük problemlerin çözümünde kullanıldıkları gibi, meta-sezgisel algoritmaların başlangıç çözümlerinin oluşturulmasında da kullanılmaktadırlar. Sezgisel öncelik kuralını kullanan algoritmalar arasında genellikle, örneklemeye dayalı algoritmalar öne çıkmaktadır. En iyi sonuçların “adaptif algoritmalar” (Schirmer, 1998; Kolisch ve Drex1, 1996b) ile alındığı görülmektedir. Sezgisel kurallardan, LFT, LST ve WCS de iyi sonuçlar verdiği ifade edilmektedir. Ancak henüz görüş birliğine varılamamıştır.

Seri ve paralel çizelgeleme uygulamaları sonucunda; seri çizelgelemenin, nispeten küçük problemlerde, (30 faaliyet gibi) daha iyi sonuçlar verdiği; paralel çizelgelemenin ise, kısmen büyük problemlerde, (120 faaliyet ve üstü gibi) daha iyi sonuçlara ulaştırdığı gözlenmiştir. Bilindiği gibi, paralel çizelgeleme ertelemesiz çizelgeler türetir ve ertelemesiz çizelgeler kümesi, eniyi çözümü içermeyebilir. Buna karşın, seri çizelgeleme, aktif çizelgeler türetir ve aktif çizelgeler kümesi eniyi çözümü içerir. Kolisch (1995), bu gözlem ve tespitten hareketle, daha küçük problemlerde paralel çizelgelemenin eniyi çözümü dışlayabileceğini; buna karşın seri

çizelgelemenin eniyi çözümü daha kolaylıkla saptayabileceğini; daha büyük problemlerde ise, paralel çizelgelemenin, daha küçük bir çözüm kümesi üzerinde çalışması sebebiyle, daha iyi sonuçlara ulaşabileceğini belirtmektedir.

6.4.5 Meta-sezgisel yöntemler

Meta-sezgisel yöntemlerin genelde, kombinatoriyal en iyilemede; özelde ise, proje ve makina çizelgelemede giderek daha yaygın bir kullanım alanı bulduğu gözlemlenmektedir. **KKPÇ problemlerinin çeşitli yerel arama, genetik algoritma, tavlama benzetimi (simulated annealing) ve TA gibi komşuluk araması temeline dayanan arama yöntemleriyle çözümünde, komşuluk yapısı çok önemli bir unsur olarak ortaya çıkar. Çünkü bu yöntemlerde kullanılan komşu çözüm üretme düzeneğinin yapısı, çözüm kalitesini etkilemektedir.** Meta-sezgisel yöntemler arasında yer alan sinir ağları (neural networks) ve karınca kolonisi (ant colony) konularına ise, bu çalışmada değinilmeyecektir. Çözüm gösteriminin çizelgeye dönüştürülmesinde, seri ve paralel çizelgeleme kullanılır. Paralel çizelgelemenin erteleme-siz çizelge türetmesi ve eniyi çözümün bunlar arasında olmaması olasılığı sebebiyle bazı paralel çizelgeleme uygulamalarında çizelgelelenebilir faaliyetlerin ertelenmesine olanak sağlanmıştır. Bu şekilde uygulanan paralel çizelgeleme, uyarlanmış paralel çizelgeleme olarak nitelendirilmektedir. Meta-sezgisel yöntemlerin bitiş kriteri olarak çeşitli seçenekler önerilmiştir: Sabit bir CPU süresi, belirli bir sayıda çizelge (komşu çözüm) türetilmesi veya amaç işlevi üzerinde tanımlanmış bir durdurma kriteri.

6.4.5.1 Genetik algoritma (GA)

Genetik algoritmalar, Holland (1975) tarafından, biyolojik evrim sürecinden hareketle, önerilmiştir. Burada tanıtılan diğer iki meta-sezgisel yöntemden farklı olarak genetik algoritma, tek bir çözüm yerine, aynı anda birden fazla sayıda çözümü değerlendirir. Bu özellik, genetik algoritmanın paralel arama özelliği olarak nitelendirilir. Problem çözümü, genlerden oluşan bir kromozom ile ifade edilir. GA, bir dizi çözümün oluşturduğu bir kuşak (generation) ile başlatılır. Güncel kuşak içindeki kromozomlara; kopyalama (reproduction), çaprazlama (crossover) ve mutasyon (mutation) işlemleri (operator) uygulanarak, yeni bir kuşak oluşturulur. Yeni kuşak oluşturulurken; uygulanan işlemlerde, “daha iyi” kromozomların bir

sonraki kuşağa aktarılması ve yeni kromozomların oluşturulmasındaki etkinliklerinin olasılıkları, uygunluk değerlerine (fitness value) doğrudan bağlıdır. Daha iyi kromozomlar, daha yüksek uygunluk değerlerine sahip olan kromozomlardır. Bu şekilde, biyolojik evrim sürecinin “güçlünün yaşamını sürdürmesi” anlayışı, genetik algoritmaya yansıtılmaktadır. Burada genetik algoritmaların ayrıntısına girilmeyecektir. Genetik algoritmalara iyi bir giriş olarak, Goldberg (1989) önerilir. GA’ların KKPCP’lerine uygulamaları, çözümün değişik gösterimlerini esas alarak, başlangıç kuşak kromozomlarının gen değerleri, rassal olarak veya öncelik kuralları kullanılarak elde edilir. Lee ve Kim (1996); önerdikleri genetik algoritmada, rassal anahtar gösterimi ile birlikte tek noktadan çaprazlama (one-point crossover) kullanmışlar ve paralel çizelgeleme uygulamışlardır. Hartmann (1998), üç gösterimi de incelemiştir. Her gösterimde de, tek noktadan çaprazlama, çift noktadan çaprazlama (two-point crossover) ve düzgün sıralı çaprazlama (uniform crossover) yöntemleri sınanmıştır. Yine her gösterimde, seri çizelgeleme kullanılmaktadır. Faaliyet listesi gösteriminde, faaliyetlerin kromozom üzerinde sıralanışı, öncüllük açısından olurlu bir sıralamadır. Kromozom üzerindeki sıraya göre; en erken olurlu ana atama yapılarak çizelge oluşturulur. Hartmann, öncelik gösteriminde LFT, LST, MTS, MSLK, WRUP ve GRPW öncelik kurallarını kullanmıştır. Rassal anahtar gösterimi uygulamasında her gen, öncelik değerine karşı gelen $[0,1]$ aralığında bir değer almaktadır.

6.4.5.2 Tavlama benzetimi

İlk defa Kirkpatrick vd. (1983) tarafından sunulan “Tavlama Benzetimi Yöntemi”, ilk karşılaşılan daha iyi çözümü kabul ederek, temelde en iyiyi arama stratejisinin değişik bir uygulama biçimidir. Her adımda daha iyi bir noktaya gitmeyi hedefleyen gradyan ile arama yönteminden farklı olarak, Tavlama Benzetimi Yöntemi, süreç içinde giderek azalan bir olasılıkla, daha kötü bir çözüme gitmeye izin verir. Tavlama Benzetimi yöntemi ile bir çözüm önerisi Boctor (1996) tarafından sunulmuştur. Önerilen algoritmada, faaliyet listesi kullanılarak seri çizelgeleme uygulanmıştır. Bu çalışmanın amacı, komşuluk araması temelinde, çağdaş optimizasyon teknikleriyle daha iyi çizelgeleme yöntemlerinin oluşturulmasına katkı sağlamaktır. Bu bölümde, tabu algortiması ile çözümün her bir adımı açıklanmaya

çalışılacaktır. Hazırlanan paket program ve uygulama sonuçları için Atli (2005, 2011)'e bakınız.

6.4.5.3 Tabu Arama (TA)

Yerel arama teknikleri, mevcut çözümün komşularının araştırılmasıyla, kompleks iş problemlerine optimum ya da yaklaşık optimum çözüm bulan bir yeteneğe sahiptir. Burada karşılaşılan en büyük problem, yerel optimuma yakalanma sorunu ve aşırı zaman kaybıdır. TA tekniği, daha önce incelenmiş belli sayıda çözümü, tabu listesi olarak adlandırılan bir listede tutarak (bir hafıza fonksiyonu oluşturarak), o çözümlere geri dönmeyi bir müddet yasaklayan ve böylece aramayı, çözüm uzayının daha iyi noktalarına yönlendiren bir arama prosedürüdür.

TA tekniğine ait ilk çalışmalar, F.Glover tarafından yapılmıştır. Buna göre tekniğin esası, zeki problem çözme eğiliminden kaynaklanmaktadır. Bununla beraber, TA, yerel optimalliğe düşmekten kaçınmak için bir hafıza fonksiyonu kullanan, global optimumu hızlı bir şekilde aramada bir veya daha çok YA yordamını hiyerarşik olarak yönlendiren zeki bir tekniktir.

Glover (1989), tarafından geliştirilen tabu tanımlayarak arama; temelde güncel çözümün çevresinde yerel eniyi çözümü arayan gradyan yönteminin (best fit strategy) bir uyarlamasıdır. **Algoritmanın bir yerel eniyi çözümden ayrıldıktan hemen sonra, tekrar aynı yerel eniyi çözüme yönelmemesi için, algoritmanın attığı son adımlar tabu ilan edilir. Tabu listesi dinamiklidir. Her yeni bir eleman tabu listesine girdiğinde, tabu listesinde en çok kalmış olan eleman, liste dışına çıkarılır. Bu şekilde algoritmaya bir hafıza kazandırılmış olunur.**

Komşuluk arama yönteminde bir başlangıç sıralamasının, ne kadar çok sayıda komşusu türetilirse, performansı da o derecede artmaktadır. Ancak komşuluk arama tekniğinin, bir yerel çözüm bulup, bu çözümü global çözüm olarak kabul etme eğilimi vardır. Bu teknikle, bir lokal optimal çözüm elde edildikten sonra, bu çözüm üzerinden arama işlemine devam edilir ve bu yüzden problemin, global optimal çözümüne ulaşma garantisi verilmez. Çünkü, problemin global çözümüne, belki de daha kötü amaç fonksiyonu değerine sahip olan bir sıralamadan ulaşılabilir. **Karar ağacı üzerinde başlangıç sıralamasından daha kötü amaç fonksiyonu**

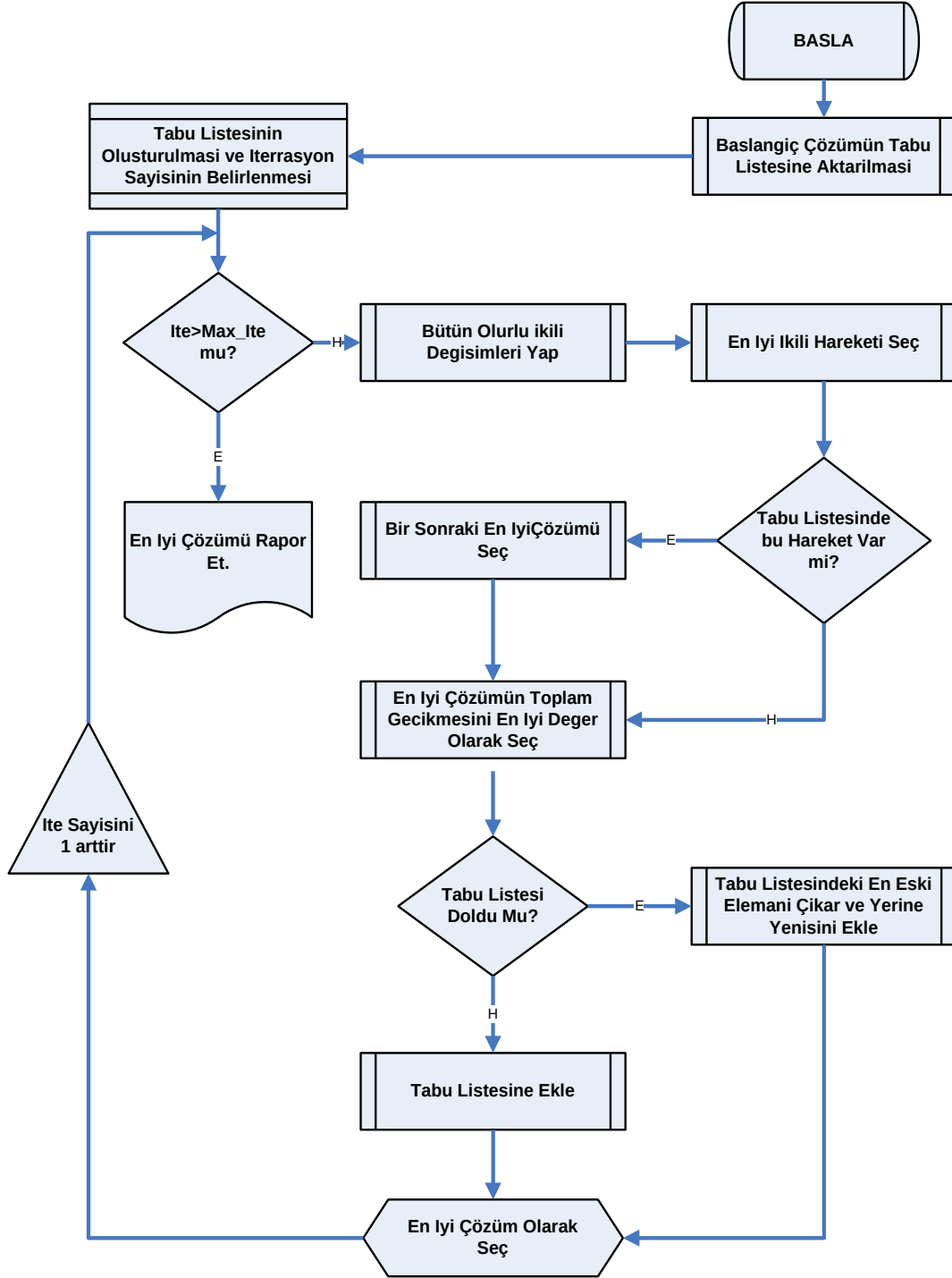
değerine sahip olan bir düğüm üzerinden arama yapma esnekliği, TA metodunda mevcuttur.

Temel olarak TA yöntemi, komşuluk arama metodunun daha geliştirilmiş bir şekli olarak düşünülmektedir. Yani her komşunun oluşturulması ve bu komşulardan (düğümleden) birinin seçimi, bir düğümden diğerine geçiş, hareket olarak adlandırılmaktadır. **TA metodu, bir yerel optimal çözümde arama işlemlerini sona erdirmek yerine, daha kötü bir değere sahip olsa da, yeni bir sıralamayı ele alır ve o sıralamadan arama işlemine devam eder.** Yeni seçilen sıralama, bir önceki sıralamadan daha kötü ise, arama işlemi sonsuz kez tekrarlanabilir. Bu durumu önlemek amacıyla, yeni seçilen düğümden bir önceki düğüme gitmek (tabu) yasaklanmaktadır. Benzer şekilde bu düğümden iki ya da üç adım önceki düğüme gitmek de yasaklanmaktadır. Yasaklanmış olan adımlar tabu listesine eklenir. TA metodunun her aşamasında, başlangıç sıralamasının komşuları arasından tabu listesinde olmayan en iyi sıralama seçilir.

TA algoritmasının işleyişi kısmen basittir, ancak belirlenmesi gereken bazı önemli stratejiler ve parametre değerleri vardır. Temel olarak **başlangıç çizelgesi oluşturma yöntemi; komşuluk yapısı, tabu listesi uzunluğu ve işletme stratejisi, aspirasyon ölçütü, seçkin çözümler listesi uzunluğu, aramayı yoğunlaştırma-çeşitlendirme stratejisi ve aramayı durdurma ya da maksimum iterasyon sayısıdır.**

TA parametre değerleri ve stratejilerinden en önemlisi komşuluk yapısıdır. Başlangıç çizelgesi ROT ve MinSlack kriterlerinden herhangi biri seçilerek oluşturulmuştur. Tabu listesi uzunluğu sabit değişken olarak seçilmiş ve liste, “İlk giren ilk çıkar.” kuralıyla işletilmiştir. Aday çözümler listesinin ve tabu listesinin işletilme biçimi aynı zamanda yoğunlaşma ve çeşitlendirme stratejilerini de şekillendirmektedir. Değerlendirme ölçütü ise şöyle kullanılmıştır: Eğer bir ikili yer değişim (alternatif komşu çözüm), tabu olsa bile, o ana kadar bulunan en iyi çözümden daha iyi ise; o ikili yer değişimin tabu durumu iptal edilerek arama, o komşudan devam ettirilir. Son olarak iterasyon yapısı şöyledir; her bir aday çözüm bulunduğunda iterasyon sıfırlanır, iyileştirmeyen ikili yer değişimlerin sayısı, seçilen değere ulaştığında; aday çözümler listesine geri dönülür ve eğer aday çözümler listesi boşaltılmış ise, algoritma durdurulur. TA algoritmasına ait genel akış şeması Şekil 6.7’de verilmiştir. Tabu aramasını sunmadan önce, çizelgeleme etkinlikleri için ve belirlenen öncelik sıralaması bakımından temel bir süreç ortaya konur. Bu sebeple birkaç sezgisel kural;

en az aylak süre (MinSlack-minimum slack first) ve ROT sezgiselleri, tabu aramasında başlangıç çözümleri sağlamak için kullanılır. Çizelgeleme süreci başlamadan önce, bütün etkinliklerin göreceli öncelik sıralaması düzenlenir. Genellikle öncelik sıralaması, bazı sezgisel kurallar ya da formüller ile hesaplanır. Etkinliklerin öncelik sıralamasına göre, kaynaklar tahsis edilir. Uygulanabilir bir etkinlik, önceki etkinliklerin tamamlanmış olmasını gerektiren bir etkinliktir. Her çizelgeleme anında, kaynak tahsisi bakımından yalnızca uygulanabilir etkinlikler göz önünde bulundurulur. t anında, kullanılabilir kaynaklar yeterli ise; en yüksek öncelikli uygulanabilir etkinlik, çizelgelenir. t zamanından sonraki uygulanabilir etkinlik için, kullanılabilir kaynakların sayısı uygun değilse; $t+1$ zamanı dikkate alınır. Etkinlik önceliği; önceden tahsis ve kısmi kaynak tahsisatlarına izin verilmemektedir.



Şekil 6.7 : TA Algoritmasının Genel Akış Şeması.

6.4.5.4 TA için notasyonlar

N , projedeki etkinliklerin sayısı olmak üzere; $I = \{ 1, 2, \dots, N \}$, bütün etkinliklerin kümesi ve Π ; I üzerinde tanımlanan permütasyon kümesi olsun. Her $\pi \in \Pi$ permütasyonu, bir N vektör $(\pi(1) \pi(2) \pi(3) \dots \pi(N))$ olarak tanımlanır. Bir π

permütasyonundaki etkinliklerin düzeni, projenin öncelik ilişkileri ile tutarlı ise; π permütasyonu, “uygulanabilir bir ardışıklık” diye adlandırılır. Bu, etkinliklerin ardışıklık ile aynı düzen içinde tamamlanabileceği anlamına gelir. Bir permütasyonun, illa uygulanabilir bir çizelge olması gerekmez. F, bütün olurlu komşuluklar kümesi olduğunda F, Π ’nin bir alt kümesidir. Bir proje; kaynak kısıtlı olduğunda, uygulanabilirlik ardışıklığındaki düzene göre çizelgelenebilir ve proje süresi böylece belirlenebilir. TA’sının amacı, optimal ya da optimale yakın çizelgeleme veren F’de uygulanabilir bir ardışıklık bulmaktır.

Bir başlangıç çözümü; sezgisel yaklaşımdan elde edilen uygulanabilir bir ardışıklıktır ve başlama çözümü, tabu araması algoritması ile iyileştirilmelidir. TA ile elde edilen çözümün niteliği, başlama çözümünden epeyce etkilendiğinden; farklı başlama çözümlerini belirlemek için MinSlack ve ROT vb. kurallar kullanılabilir. İkili bir yer değişim, uygulanabilir ardışıklıktan bir başkasına geçiştir. Bu i ve j gibi iki etkinliğin pozisyonlarının karşılıklı değişimi şeklinde yapılır. Nesnel bir fonksiyon, bir uygulanabilir ardışıklığın proje süresi olarak tanımlanır. İkili yer değişimin değeri; karşılıklı değiştirme ile üretilen uygulanabilir ardışıklığın nesnel fonksiyon değeri ile cari uygulanabilir ardışıklığın nesnel işlev değeri arasındaki farktır. Bir hareketin değeri olumsuz ise; harekete, ilerleme hareketi denir.

Bazı uygulamalarda, yeni ikili yer değişim, yalnızca bir tabu listesi kullanır. Bu tabu listesi, çözümün belli sayıda tekrarlama için yeniden ziyaret edilmesini önlemek için kullanılır. Bu aramada, aramanın farklı niteliklerine bağlı olarak iki tabu listesi uygulanabilir. Bu sebeple bir projenin etkinlikleri iki kategoriye ayrılır: kritik etkinlik ve kritik olmayan etkinlik. KYM/PERT kullanarak çizelgelenen bir projenin kritik yolu üzerinde ise; bir etkinlik, kritik etkinlik olarak sınıflanır, böyle değilse; o, kritik olmayan bir etkinliktir. Bu konu, şu bakımdan önemlidir; **İkili yer değişimler esnasında bir kritik yol üzerinde yer alan etkinliğin, o faaliyetin sahip olduğu EST’den daha küçük bir faaliyetle yer değiştirmesine izin verilmemektedir.** Yani kritik faaliyet ertelenebilir; ancak EST’sinden küçük bir zamanda başlayamaz.

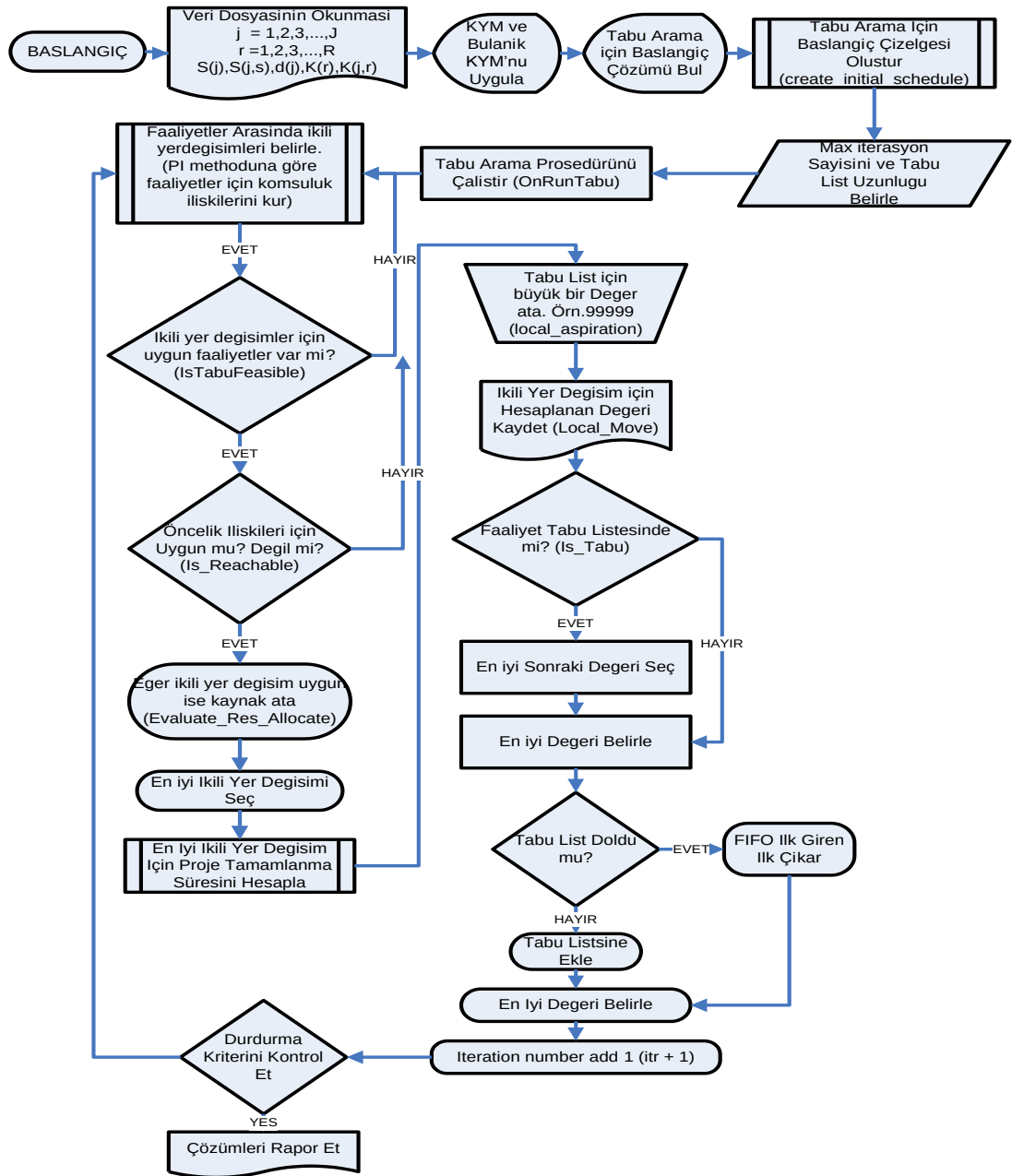
Kaynak; sınırlamaları içeriyorsa, genelde kritik bir etkinlik geciktiğinde, ertelenen etkinliğin ardılları da gecikecektir. Sonuç olarak, kısıt kaynaklı projenin tamamlanma çizelgelemesi, yüksek bir olasılıkla, gecikecektir. Kritik etkinlik, ileri doğru hareket ettirildiğinde, karşılıklı değişime bağlı olarak daha erken tamamlanmasına izin verilmeyecektir. Bütün ikili yer değişimlerin sayısı, PI (All Pairwise Interchange)

komşuluk yapısına göre her tekrarlama üretilip değerlendirilen aday ikili yer değişimlerin sayısı oluşturulmaktadır. **Aday listesindeki her aday ikili yer değişimlerin sayısı, $C(N,2)$ kadardır.** Aday listesindeki her aday ikili yer değişim, bir değerlendirme fonksiyonu ile değerlendirilir. Değerlendirme fonksiyonu, ikili yer değişimlerinin değeri ile ceza fonksiyonunun bir kombinasyonudur. Bir ceza fonksiyonu, önceden belirlenen sınırlamalar ihlal edildiğinde uygulanır. Bu çizelgeleme uygulamasında yalnızca, uygulanabilir ardışıklıklar göz önünde bulundurulduğu için, ceza fonksiyonu gerekli değildir. Bir uygulanabilir ardışıklık, tabu etkinliği içeren bir ikili yer değişim ile üretildiğinde, hareket; tabu sınırlı ikili yer değişim diye adlandırılır. Bu ikili yer değişimler üzerinde bir değerlendirme testi yapılmalıdır. Değerlendirme testi, tabu aramasındaki esnekliğin önemli bir ögesidir. Değerlendirme testine, yani başlangıç çözümüne göre, yeni bulunan proje tamamlanma süresi karşılaştırılarak, tabu statüsü iptal edilebilir; aksi taktirde ikili yer değişime izin verilmez. Bu çalışmada kullanılan değerlendirme testi aşağıdaki gibidir. Eğer yeni ardışıklığın proje süresi, şimdiye dek bulunan en iyi proje düzeyinden daha kısa ise; tabu statüsü iptal edilebilir. Kabul edilebilir bir ikili yer değişim, tabu sınırlı olmayan bir harekettir ya da değerlendirme fonksiyonu ile değerlendirildiğinde, en iyi değere sahip olan tabu sınırlı harekettir. En iyi ikili yer değişim; değerlendirme fonksiyonu ile değerlendirilirken en iyi değere sahip olan kabul edilebilir ikili yer değişimdir. En iyi ikili yer değişimi yaparak üretilen ardışıklık, sonraki tekrarlamanın başlangıç çözümü olarak hizmet eder. En iyi ardışıklığın mutlaka var olan çözümü üzerinde bir iyileşme olması gerekmez. Bu; tabu aramasının yerel minimumdan kaçınmak için kullandığı bir yöntemdir.

TA'nın performansı bir çok parametreye bağlıdır. En önemli parametrelerden biri tabu uzunluğudur. Tabu uzunluğu değeri için tercih edilen seçim, geleneksel olarak “ampirik” bir teste dayanır. Küçük bir tabu uzunluğu muhtemelen, aramada döngüsellliği başlatacaktır. Tabu uzunluğu çok yüksek ise; tabu araması, küçük bir alanla sınırlanacaktır. Çizelgeleme problemi için tabu uzunluğu değerleri tipik olarak etkinliklerin sayısının basit bir fonksiyonu olarak etkinliklerin sayısının kare kökü vb getirilebilir. Bir başka önemli parametre de, aday ikili yer değişimlerin sayısıdır. İkili yer değişimler, her tekrarlama göz önünde bulundurulur. Hem çözüm hızı hem de kalitesi; uygun aday listesi stratejilerinden, önemli ölçüde etkilenebilir.

6.4.5.5 TA'nın uygulaması

Algoritmanın akış yapısı Şekil 6.8'de sunulmuştur.

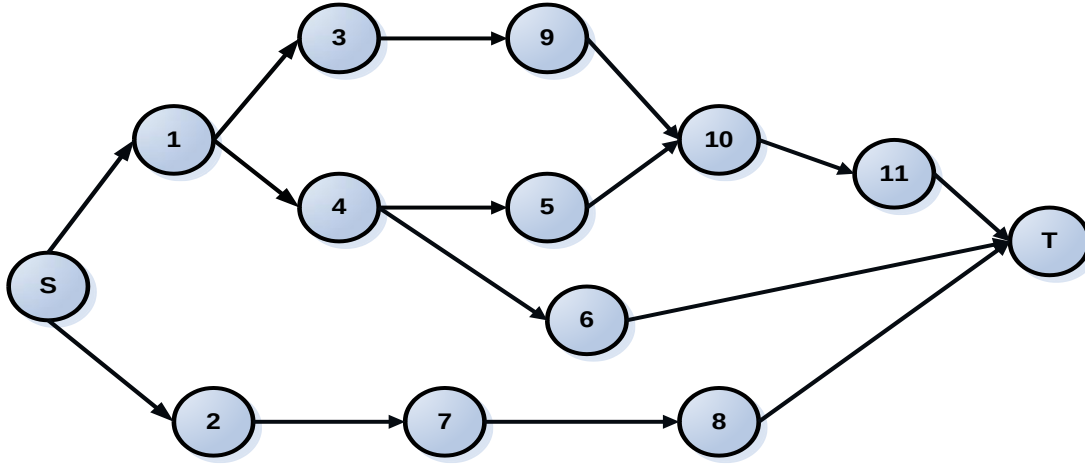


Şekil 6.8 : KKPÇP için Önerilen TA Algoritması.

Microsoft Visual C# 2008'de geliştirilen kaynak kısıtlı, deterministik ve bulanık faaliyet süreli proje çizelgeleme algoritması aşağıdaki gibi uygulanmaktadır. Bu bölümde algoritmanın daha iyi anlaşılabilmesi için sayısal bir örnek üzerinde algoritmanın her bir adımı açıklanmaya çalışılacaktır.

6.4.5.6 TA için deterministik sayısal örnek 2

Literatürde Patterson 110 problemleri olarak bilinen ve PSPLIB web sayfasında <http://129.187.106.231/psplib/> dan PAT3.rcp veri dosyası kullanılacaktır. Şekil 6.9’da şebeke yapısı ve Çizelge 6.5’de örnek probleme ait bilgiler verilmiştir.



Şekil 6.9 : Örnek Şebeke Yapısı.

Çizelge 6.5 : Sayısal Örnek Veri Yapısı.

Faaliyet Toplam Sayısı :	13	Kaynak Türü Sayısı :	3			
		1.Kaynak	2.Kaynak	3.Kaynak		
Kaynak Maks.Sayısı		6	7	6		
Birim Brüt Kaynak Kullanım Maliyeti		6 Pb	7 Pb	6 Pb		
Faaliyetler	Süreler	1.Kaynak Türü Kullanımı	2.Kaynak Türü Kullanımı	3.Kaynak Türü Kullanımı	Faaliyetin Ardıllarının Sayısı	Ardıl Faaliyetler
S	0	0	0	0	2	2 3
1	3	3	2	1	2	4 5
2	5	2	4	2	1	8
3	6	3	1	2	1	10
4	2	4	3	1	2	6 7
5	3	2	0	3	1	11
6	3	1	1	1	1	13
7	4	3	1	1	1	9
8	5	2	2	2	1	13
9	4	3	2	3	1	11
10	2	4	1	0	1	12
11	3	5	4	2	1	13
E	0	0	0	0	0	

Adım 1. KYM ve B-KYM yöntemi ile ;

✓ Şebeke yapısı ve öncelik ilişkileri esasına göre faaliyetlerin EST (En erken başlama zamanı), LST (En geç başlama zamanı), EFT (En erken tamamlanma zamanı), LFT (En geç tamamlanma zamanı) ve aylak süreleri ile bulanık faaliyetlerin ise FEST, FLST FEFT FLFT ve aylak süreleri hesaplanır.

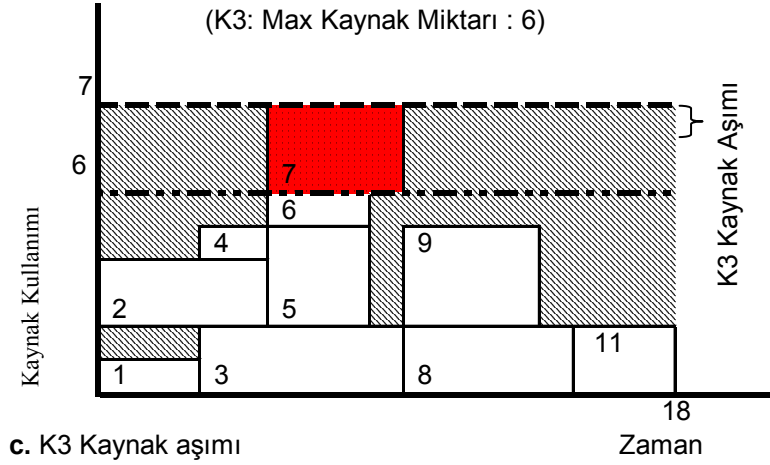
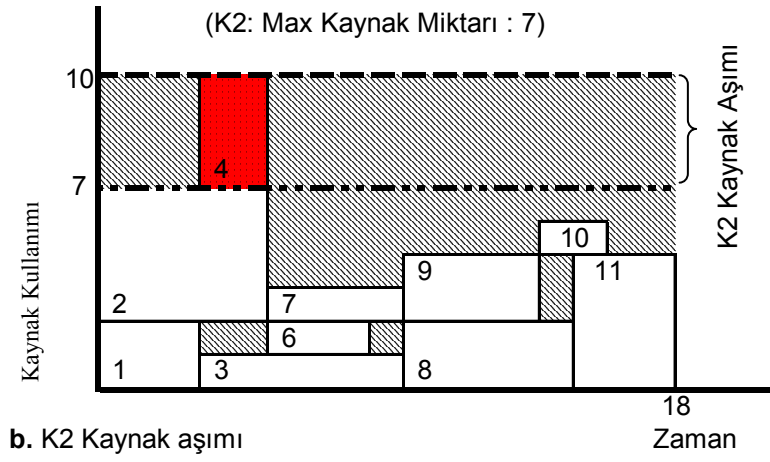
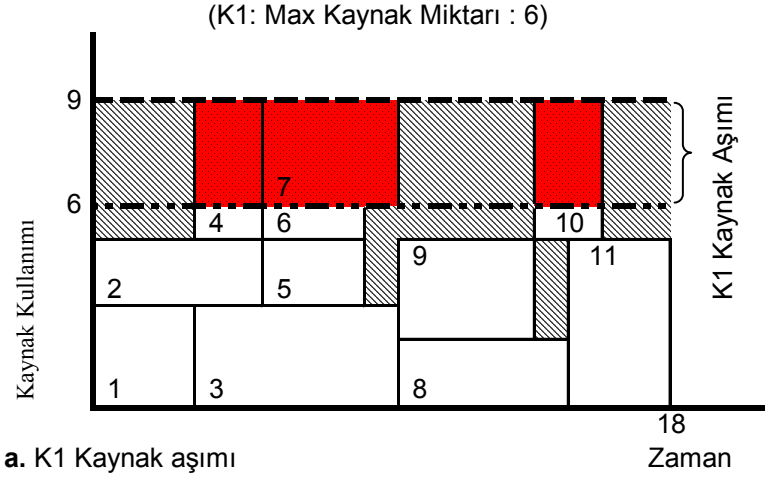
- ✓ Kritik etkinlikler ve kritik olmayan etkinlikler tanımlanır.
- ✓ Proje kaynak ihtiyaçları dikkate alınmadan çizelgelenir.
- ✓ KYM ve B-KYM'ne göre en küçük proje tamamlanma sırası ve süresi bulunur.

Çizelge 6.5'deki veriler ışığında Adım 1.'de KYM uygulanarak elde edilen sonuçlar Çizelge 6.6'da gösterilmiştir. S ve T faaliyetleri yapay faaliyetler olup projenin başlangıç ve bitiş faaliyetlerini oluşturmaktadır. Yapay faaliyetlerin işlem süreleri ve kaynak kullanım miktarları 0'dır.

Çizelge 6.6 : Adım 1 Sonunda Elde Edilen Sonuçlar.

No.	Süreler	EST	EFT	LST	LFT	Aylak Süreler (LFT-EFT)
S	0	0	0	0	0	0
1	3	0	3	0	3	0
2	5	0	5	4	9	4
3	6	3	9	3	9	0
4	2	3	5	8	10	5
5	3	5	8	10	13	5
6	3	5	8	15	18	10
7	4	5	9	9	13	4
8	5	9	14	13	18	4
9	4	9	13	9	13	0
10	2	13	15	13	15	0
11	3	15	18	15	18	0
T	0	18	18	18	18	0

Projenin kaynak kısıtları dikkate alınmaksızın yapılan KYM ile çözümü sonucu, bulunan proje tamamlanma süresi 18; Aylak süreleri 0 olan faaliyetler, kritik yolu; diğer faaliyetler ise, kritik olmayan faaliyetleri belirtir. Buna göre kritik yol, S-1-3-9-10-11-T. Kritik olmayan faaliyetler ise, 2-4-5-6-7-8 faaliyetleridir. Şekil 6.10'da; KYM'ye göre hazırlanan çizelgelerden, kaynak aşımına neden olan kaynaklar belirlenebilir.



Şekil 6.10 : KYM Çizelgesi ve Kaynak Aşımı.

Adım 2. Başlangıç çözümün bulunması:

- ✓ Başlangıç çözümü sezgisel bir kural kullanılarak belirlenebilir,
- ✓ Örneğin ROT ve MinSlack kurallarından birisi seçilir,

- ✓ Her bir faaliyetin ROT değerleri, ilk kaynak tipine göre ağ yapısı üzerinde son faaliyetten başlayarak geriye doğru hesaplanır,
- ✓ Ayrıca MinSlack değerleri de kaynak tipine bağlı olmadan hesaplanarak öncelik kuralları belirlenir.
- ✓ Faaliyetlerin ROT değerleri büyükten küçüğe ve MinSlack değerleri küçükten büyüğe doğru sıralanarak öncelik kuralları belirlenmiş olur.
- ✓ Faaliyetlerin ROT ve MinSlack değerlerinin oluşan sırasına göre kaynak ataması şu şekilde yapılır,
 - ◆ 0 başlangıç zamanından itibaren, faaliyetlerin EST veya FEST değerlerine göre çizelgelenebilecek faaliyetler belirlenir,
 - ◆ Kaynaklar, faaliyetlerin öncelik ilişkileri esas olmak üzere ROT ve MinSlack kriterlerine göre tüm kaynak tiplerinde, kaynak aşımı olmayacak şekilde atanır,
 - ◆ EST'si uygun ve yeterli miktarda kaynak mevcut olması halinde (Eğer kaynak kullanımı söz konusu değil ise bu gerçekleşir.), bir önceki faaliyetin gerçek tamamlanma zamanından önce, kaynak ataması yapılacak faaliyetin gerçek başlangıç zamanı bir önceki faaliyetin tamamlanma zamanından önce başlamasına izin verilmemiştir. Aksi takdirde optimalden küçük proje tamamlanma süreleri bulunabilir. Buna izin verilemez.
 - ◆ Atama yapılan faaliyetlerden gerçek tamamlanma zamanı en küçük (bulanık faaliyet süreli projelerde yager sıralama indeksine göre sıralama yapılmalıdır) olanı bir sonraki iterasyonun yeni zaman sayacı olarak güncellenir.
 - ◆ Tüm faaliyetlerin her bir kaynak tipinden ataması bitene kadar iterasyona devam edilir.
- ✓ ROT ve MinSlack kriterlerine göre başlangıç çizelge sırası ve proje tamamlanma süresi bulunur,
- ✓ Elde edilen çizelge sırası tabu algoritması için başlangıç çözümü olarak seçilir.
- ✓ Şimdiye kadar bulunan en iyi sıralama (S_b) uygulanabilir ardışıklık S_i olarak seçilir.

✓ Çizelge 6.7’de Minslack ayrıca Çizelge 6.8’de ROT algoritmasından elde edilen sonuçlar ve Şekil 6.11’de grafik gösterimi sunulmuştur. Buna göre; Minslack algortimasına göre bulunan çizelge sırası, S-1-2-3-7-9-8-4-5-6-10-11-T ve proje tamamlanma süresi, 23 olarak bulunur.

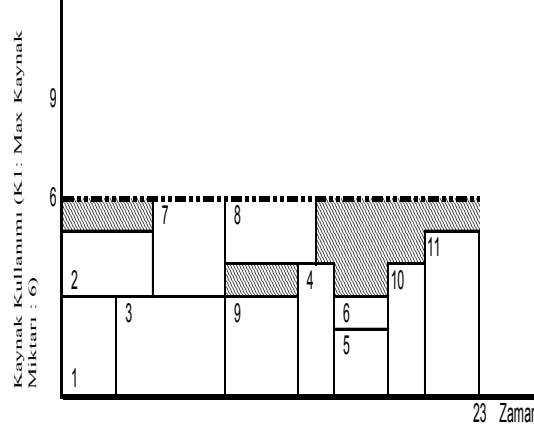
Çizelge 6.7 : Minslack Algoritmasının İşleyişi.

j	S	1	3	9	10	11	T	2	7	8	4	5	6
Minslack	0	0	0	0	0	0	0	4	4	4	5	5	10
Süre	0	3	6	4	2	3	0	5	4	5	2	3	3
Kaynak	0,0,0	3,2,1	3,1,2	3,2,3	4,1,0	5,4,2	0,0,0	2,4,2	3,1,1	2,2,2	4,3,1	2,0,3	1,1,1
EST	0	0	3	9	13	15	18	0	5	9	3	5	5
Tstart	0	0	3	9	18	20	23	0	5	9	13	15	15
TSON = Süre+Tstart	0	3	9	13	20	23	23	5	9	14	15	18	18
TNOW	0	0	3	5	9	13	14	15	18	20	23		
Kaynak Düzeyi	6,7,6	6,7,6	4,3,4	3,6,4	6,7,6	4,5,4	2,4,5	6,7,6	6,7,6	6,7,6	6,7,6		
Kaynak Kullanımı	0	3,5,5	1,2,2	0,5,3	3,5,3	0,2,3		4,7,3	2,6,6	1,3,4	6,7,6		
Kaynak Kullanımı	0	1,1,3			1,3,1			3,6,2					
Çizelgelenebilecek Faal. Aday Liste	S	1,2	3,4	4,7	9,8,4	4	-	5,6	10	11	T		
Çizelgelenen Faaliyetler	S	1,2	3	7	9,8	4	-	5,6	10	11	T		
İterasyon	1	2	3	4	5	6	7	8	9	10	11		

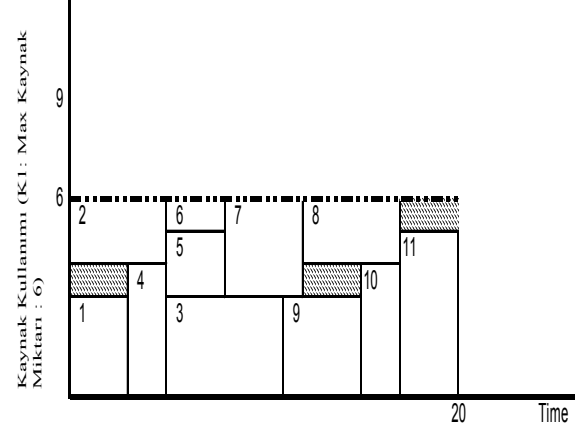
ROT algortimasına göre bulunan çizelge sırası, S-1-2-4-3-5-6-7-9-8-10-11-T ve proje tamamlanma süresi, 20 olarak bulunur.

Çizelge 6.8 : ROT Algoritmasının İşleyişi.

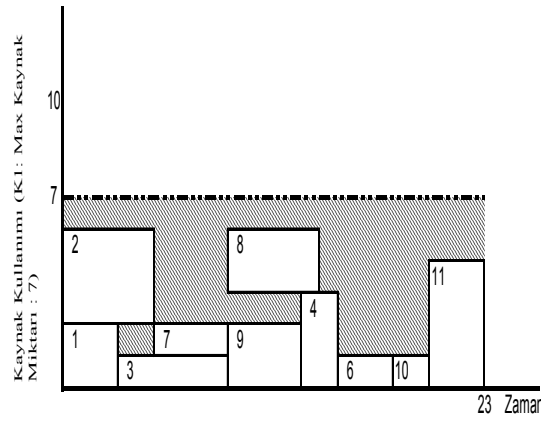
j	S	1	4	3	9	5	10	11	2	7	8	6	T
ROT(j)	7,3	7,3	6,3	4,9,2	4,4,2	4,3	3,6	1,6	1,5,5	1,1,5	0,4	0,3	0
Süre	0	3	2	6	4	3	2	3	5	4	5	3	0
Kaynak	0,0,0	3,2,1	4,3,1	3,1,2	3,2,3	2,0,3	4,1,0	5,4,2	2,4,2	3,1,1	2,2,2	1,1,1	0,0,0
EST	0	0	3	3	9	5	13	15	0	5	9	5	18
Tstart	0	0	3	5	11	5	15	17	0	8	12	5	20
TSON = Süre+Tstart	0	3	5	11	15	8	17	20	5	12	17	8	20
TNOW	0	0	3	5	8	11	12	15	17	20			
Kaynak Düzeyi	6,7,6	6,7,6	4,3,4	6,7,6	3,6,4	3,6,5	3,5,3	4,5,4	6,7,6	6,7,6			
Kaynak Kullanımı	0	3,5,5	0,0,3	3,6,4	0,5,3	0,4,2	1,3,1	0,4,4	1,3,4	6,7,6			
Kaynak Kullanımı	0	1,1,3		1,6,1									
Kaynak Kullanımı				0,5,0									
Çizelgelenebilecek Faal. Aday Liste	S	1,2	4,3	3,5,7,6	7	9	8	10	11	T			
Çizelgelenen Faaliyetler	S	1,2	4	3,5,6	7	9	8	10	11	T			
İterasyon	1	2	3	4	5	6	7	8	9	10			



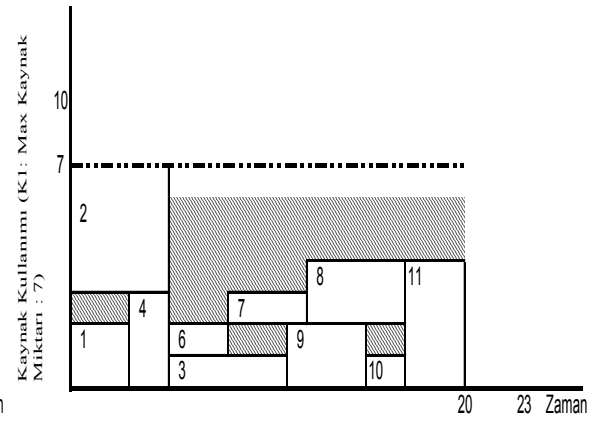
a. K1 Kaynak Tipi için MinSlack Algoritması



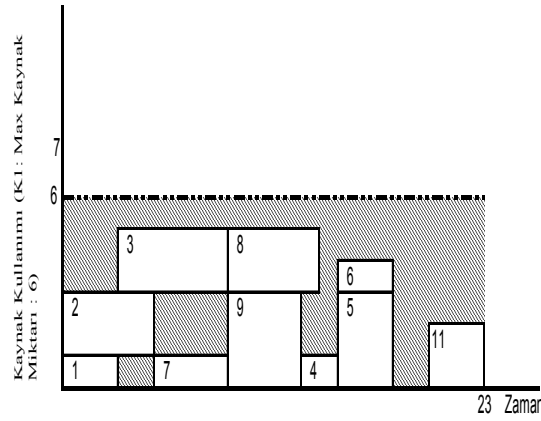
a. K1 Kaynak Tipi için ROT Algoritması



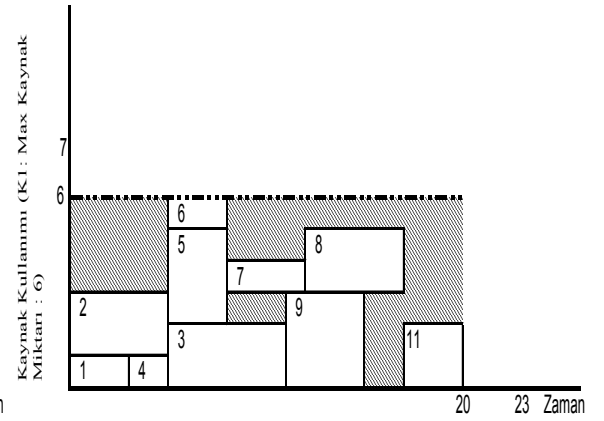
b. K2 Kaynak Tipi için MinSlack Algoritması



b. K2 Kaynak Tipi için ROT Algoritması



c. K3 Kaynak Tipi için MinSlack Algoritması



c. K3 Kaynak Tipi için ROT Algoritması

Şekil 6.11 : MinSlack ve Rot Algoritmalarının Çözümü.

Adım 3. Tabu aramasında kullanılan değişkenlere değer atanması:

✓ Tabu listesi genişliği için değer belirle. Örnek için $\sqrt{n-2} = \sqrt{13-2} \cong 3$ olarak belirlenir.

✓ Durdurma kriteri olarak maksimum iterasyon sayısı için değer belirle, geliştirilen programda default olarak iterasyon sayısı 2xn olarak belirlenmiştir. Ancak istenilen sayıda iterasyon girilebilmesine imkan tanınmıştır.

Adım 4. TA Prosedürü

✓ Başlangıç uygulanabilir ardışıklığı seç ve Si olarak göster. Si= S-1-2-3-7-9-8-4-5-6-10-11-T

✓ Başlangıç çözümden yola çıkarak, sıralı komşuluklar (PI) esasına göre oluşturulan komşuluklar belirlenir.

◆ S ve T yapay faaliyetlerinin, başlangıç ve bitiş olaylarını ifade etmesi ve öncelik ilişkileri nedeniyle, hiç bir şekilde (kendi aralarında ve diğer faaliyetlerle) ikili yer değiştirilmesine izin verilmez.

◆ Bu değerlendirmeye göre; (n-2)'nin ikili tüm kombinasyonları kadar, ikili yer değişim yapılabilir. Buna göre; $C\left(\begin{smallmatrix} 11 \\ 2 \end{smallmatrix}\right)$ kombinasyonlarının sayısı kadar (55 adet), ikili yer değişim yapılabilir. Ancak her bir ikili yer değişim, proje şebekesi ve öncelik ilişkileri nedeniyle olurlu değildir.

✓ Çizelge sırasındaki komşulukları kullanarak olurlu ikili yer değişimler uygulanır.

✓ İkili yer değişimler şu kurallara göre yapılır:

◆ Eğer, birbirine eşit EST/FEST değerine sahip faaliyetler arasında ikili yer değişimler gerçekleştirilebilir ve farklı EST/FEST değerine sahip faaliyetlerin ikili yer değişim sonucu geri gitmeyecek şekilde ikili yer değişimleri yapılacaktır.

◆ Öncelik ilişkilerini ihlal etmeyecek şekilde ardaşık ve aynı hat üzerinde yer alan ve birbirini takip eden faaliyetler arasında, ikili yer değişim yapılmasına izin verilmeyecektir.

◆ Kritik faaliyetler ve aynı hat üzerinde yer alan faaliyetler arasında ikili yer değişimler gerçekleştirilmeyecektir.

◆ Kritik faaliyetlerin EST/FEST değerleri ikili yer değişim sonucu geri gitmeyecek şekilde, ikili yer değişimler yapılacaktır. Tabu listesindeki elemanlar için faaliyetler arasında ikili yerdeğişim yapılmayacaktır.

✓ Buna göre her bir ikili yer değişim sonucu oluşan çizelge sırası esas alınmak kaydıyla, kaynak kısıtları ve öncelik ilişkileri dikkate alınarak kaynak atamaları gerçekleştirilerek, proje tamamlanma süreleri hesaplanacaktır.

✓ İkili yer değişimler arasından en iyi sonuç vereni seç.

✓ Bu ikili yer değişimi tabu listesine ekle.

Bulunan çözümü şimdiye kadar bulunan en iyi sonuç olarak kaydet. Adım 4’de ifade edilenleri Çizelge 6.9’de iteratif olarak ilerletilen sonuçlar, özetle sunulmuştur.

Çizelge 6.9 : Tabu İterasyonlarındaki Uygun (Olurlu) Komşuluklar.

İterasyon 1	İterasyon 2	İterasyon 3	İterasyon 4
<1-2>, PTZ: 23	<1-2>, PTZ: 23	<1-2>, PTZ: 24	<1-2>, PTZ: 23
<2-3>, PTZ: 23	<2-3>, PTZ: 26	<2-3>, PTZ: 27	<2-3>, PTZ: 26
<7-9>, PTZ: 23	<7-4>, PTZ: 27	<7-4>, PTZ: 28	<7-8>, PTZ: 27
<9-8>, PTZ: 23	<4-8>, PTZ: 23	<4-8>, PTZ: 22	<5-6>, PTZ: 23
<9-4>, PTZ: 22	<8-9>, PTZ: 23	<5-6>, PTZ: 24	<6-10>, PTZ: 23
<8-4>, PTZ: 23	<5-6>, PTZ: 23	<6-10>, PTZ: 24	
<5-6>, PTZ: 23	<6-10>, PTZ: 23		
<6-10>, PTZ: 23			
İterasyon Numarası			
İte 1’de Tabu Listesi : {<4-9>},			
İte 2’de Tabu Listesi : {<4-9>,<9-8>},			
İte 3’de Tabu Listesi : {<4-9>,<9-8>,<8-4>}			
İte 4’de Tabu Listesi : {<9-8>,<8-4>,<6-5>}			

Adım 5. Her bir iterasyon için tekrarla.

✓ Bir önceki adımdan iterasyonda (Tabu hareketi yapılarak oluşan) bulunan mevcut çözüm, başlangıç çözüm olarak atanır.

✓ Yeni çizelge sırasındaki komşulukları kullanarak ikili yer değişimlere uygunluk testi yap.

✓ Çizelge sırasındaki komşulukları kullanarak ikili yer değişimler uygulanır.

✓ Uygun komşuluklara göre her bir ikili yer değişim sonucu oluşan çizelge sırasına göre kaynak kısıtları ve öncelik ilişkilerini de dikkate alarak kaynak atamaları gerçekleştirilerek, proje tamamlanma süreleri hesaplanacak.

✓ İkili yer değişimler arasından en iyi sonuç verini seç,

- ✓ Hareketin tabu statüsünü denetle,
- ✓ Bulunan en iyi çözüm veren ikili yer değişim tabu listesinde ise ve şimdiye kadar bulunan mevcut en iyi çözümden daha iyi ise, en iyi çözüm olarak kaydet.
- ✓ Bu hareket, tabu sınırlamalı değilse;
- ✓ Eğer şimdiye kadar bulunan mevcut en iyi çözümden daha iyi değilse;
- ✓ İkili yer değişimler arasında tabu listesinde olmayan en iyi çözümü seç (mevcut iterasyonda bulunan en iyi çözümü bir sonraki iterasyonun başlangıç çözümü olarak ata).
- ✓ Seçilen ikili yer değişimi (hareketi), tabu listesine ekle.
- ✓ Bu çalışmada tüm iterasyonları ilerletmek mümkün olamayacağından ; Adım 5’de ifade edilenleri Çizelge 6.10’de iteratif olarak ilerletilen sonuçlar, özetle sunulmuştur.

Çizelge 6.10 : Tabu İterasyonları.

Tabu List Uzunluğu	3		
Tabu List	1	2	3
Tabu Hareket	<4-9>	<9-8>	<8-4>
Tabu Listteki Hareketin Proje Tamamlanma Süresi	23	22	22
MinSlack Çözümünden Elde Edilen Başlangıç Çizelge	S - 1 - 2 - 3 - 7 - 9 - 8 - 4 - 5 - 6 - 10 - 11 - E 23		
Tabu Listedeki Hareketten Oluşan Çizelge Sırası (İterasyon 1)	S - 1 - 2 - 3 - 7 - 4 - 8 - 9 - 5 - 6 - 10 - 11 - E 23		
Tabu Listedeki Hareketten Oluşan Çizelge Sırası (İterasyon 2)	S - 1 - 2 - 3 - 7 - 4 - 9 - 8 - 5 - 6 - 10 - 11 - E 22		
Tabu Listedeki Hareketten Oluşan Çizelge Sırası (İterasyon 3)	S - 1 - 2 - 3 - 7 - 8 - 9 - 4 - 5 - 6 - 10 - 11 - E 22		
Kritik Yol ve Kritik Faaliyetler	S	1	3 9 10 11 T
Kritik Olmayan Faaliyetler	2	4	7 8 5 6
Bir önceki iterasyondan(ite_3) Elde Edilen Çözüme ait Çizelge	S - 1 - 2 - 3 - 7 - 8 - 9 - 4 - 5 - 6 - 10 - 11 - E 22 TABU <6-5>		

Adım 6. Toplam Proje Maliyetinin Hesaplanması

Problemin elde edilebilir maksimum kaynak miktarı brüt birim kaynak kullanım maliyeti olarak kabul edilmiş olup, Toplam Proje Çizelgeleme Maliyeti = $\sum_{k=1}^K$ (Birim Maliyet x Kaynak Kullanım Miktarı x Faaliyetin Süresi) olarak Çizelge 6.11’de hesaplanmıştır.

Çizelge 6.11 : Toplam Proje Çizelgeleme Maliyeti.

Kaynaklar	Kaynak Birim Maliyet x Faaliyetin Süresi x Kaynak Miktarı													Maliyet
K1 (6 Pb)	0	54	90	108	72	72	90	36	54	54	36	54	0	720
K2 (7 Pb)	0	42	70	84	56	56	70	28	42	42	28	42	0	560
K3 (6 Pb)	0	18	30	36	24	24	30	12	18	18	12	18	0	240
Toplam Proje Çizelgeleme Maliyeti	0	114	190	228	152	152	190	76	114	114	76	114	0	1520

Adım 7. Durdurucu ölçütleri denetle

- ✓ Maksimum iterasyon sayısına ulaşıldı mı?
- ◆ Eğer maksimum iterasyon sayısına ulaşılmadı ise Adım 5’e git.
- ✓ Maksimum iterasyon sayısına ulaşıldığı anda, şimdiye kadar bulunan en iyi çözümü, sonuç çözüm olarak rapor et.

TA’sının izlediği yol, önemli ölçüde başlangıç çözümüne bağlıdır. Dolayısıyla, TA kullanarak belirlenen çözüm de, başlangıç çözümüne bağlıdır. Daha iyi çözümün bulunabileceği farklılaşmış yolları bulmak için, farklı başlangıç çözümleri ile yukarıda betimlenen süreç, defalarca tekrarlanabilir.

ROT ve MinSlack kuralı, tabu araması için başlangıç uygulanabilir ardışıklığı bulmak için kullanılır. Bu ardışıklık için proje süresi; bu bölümün başında sunulan prosedür kullanılarak belirlenir. Burada EST, ilk başlangıç zamanı; aylak süre, bir etkinliğin aylak zamanı ve kritikliğin göstergesidir. Örnek proje için, TA’sının uygulaması sonucu projenin optimal tamamlanma süresi, 20 ve optimal çizelge, S-1-2-4-3-5-6-7-9-8-10-11-T olarak bulunur. Hazırlanan proje çizelgeleme (Project Scheduling) programı ile 1000 iterasyon çalıştırılarak elde edilen çıktı sonuçları EK-B’de sunulmuştur.

Sezgisel yöntemlerle en iyi çözümün sağlanıp sağlanamadığı, bir karşılaştırma yapılamadığı sürece kesin olarak bilinemez. Ancak, bir sezgisel yöntemin etkinliği hakkında şu genel yargı geçerlidir. İyi bir sezgisel yöntem; büyük bir olasılıkla, her zaman en iyiye yakın veya önceden tespit edilmiş değerlendirme ölçütlerine göre yeterli sayılabilecek çözümleri sağlayan yöntemdir. Bu nedenle optimizasyon yöntemlerinin, en iyi çözümün mutlaka sağlanması amacıyla geliştirilmesine karşın, sezgisel yaklaşımda amaç, etkin ve güvenilirliği yüksek olan “en iyi çözüm stratejisinin” geliştirilmesidir.

YÇ yöntemlerinin geliştirilmesinde genellikle şu ilkeler gözetilmektedir:

✓ Optimizasyon yöntemlerine, kısmen sezgisel unsurlar katılarak, en iyi olmasa da iyi bir çözüm, süratli bir şekilde elde edilebilir.

✓ Öncelik kuralı yöntemlerine, kısmen sezgisel unsurlar katılarak, en iyi olmasa da, iyi bir çözüm, süratli bir şekilde elde edilebilir.

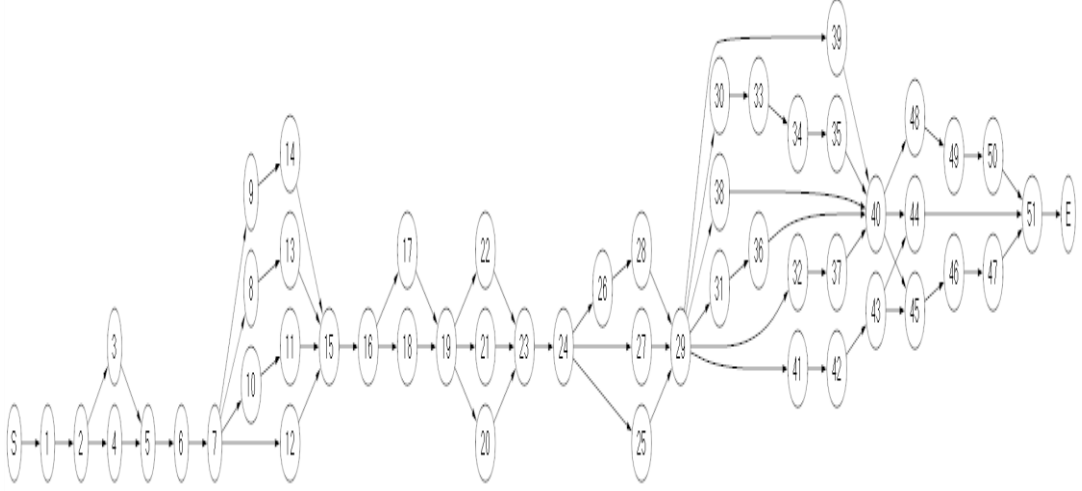
✓ Öncelik kuralı yöntemleriyle elde edilen olurlu bir çözüm üzerinde değişiklikler yapılarak, daha iyi bir çözüm ortaya çıkartılabilir.

Endüstri mühendisliği alanında son yıllarda yapılan çalışmaların önemli bir bölümünde, bulanık küme teorisini temel alan tekniklerin kullanıldığı görülmektedir. Bu konu hakkında çok sayıda çalışma yapılmış, değişik alanlarda uygulanmış ve önemli sonuçlar elde edilmiştir. Belirsizliklerin çözümünde yeni bir çıkış açan bulanık küme teorisi alanında yapılan çalışmalar, günümüzde de hız kesmeden devam etmektedir. Proje çizelgeleme alanında, projelerin doğasından kaynaklanan belirsizliklerin çözümünde, bulanıklığın kullanımıyla, “bulanık proje çizelgeleme” kavramı ortaya çıkmış ve bu konu, günümüzde artan önemi sebebiyle, birçok araştırmacının dikkatini çekmiştir. Bulanık proje çizelgelemede, projenin karmaşıklığı ve projede arıza veya bekleme oluşması gibi durumlar nedeniyle, geleneksel yöntemlerle çözümü çok zor olan problemlerin çözümde metasezgisel yöntemlerin kullanımı, kısa sürede iyi çözümlerin elde edilmesini sağlamaktadır. Gerçek hayatta, ilk defa yapılacak olan projelerde kullanılacak olan kaynaklar, belirsizdir. Bu belirsizlik; ancak bulanık mantık tekniklerinin kullanımı ile giderilebilir. Bulanık proje çizelgeleme problemlerinin çözümü ile ilgili, literatürde henüz optimum çözüm veren metasezgisel yöntemler henüz geliştirilmemiş olsa da; bu çalışma ile, bu konudaki eksiklik giderilmeye çalışılmıştır. İleriki çalışmalarda, yeni metasezgisel yöntemlerin geliştirilmesi ya da mevcut olanların melez olarak kullanımı ile optimuma yakın çözümler elde edilmesi beklenmektedir.

6.4.5.7 TA için bulanık sayısal örnek 3.

Şekil 6.12’de şebeke diyagramı gösterilen ve Wang (2004)’ın makalesinden yararlanılan orijinalinde söbü bulanık faaliyetlerden oluşan örnek yazım projesine ait verileri, yamuk bulanık sayılara dönüştürülmüş haliyle Çizelge 6.12’de sunulmuştur. Bulanık faaliyet sürelerinin dönüşümü daha önceki bölümlerde açıklanmış olup, kısaca ikinci ve beşinci sayıların silinmesiyle elde edilmiştir. Proje toplam 51 ve 2

yapay faaliyet olmak üzere 53 faaliyetten oluşmakta olup her bir faaliyetin açıklamasına ilgili makaleden ulaşılabilir. Faaliyetler icra edilebilmesi için dört farklı kaynak kullanılmış olup, maksimum elde edilebilir miktarları sırasıyla; 6, 6, 5, 4'tür.



Şekil 6.12 : Yazılım Proje Şebekesi.

Çizelge 6.12’de her bir faaliyetin yamuk bulanık faaliyet süreleri, kaynak kullanım miktarları, ardıllarının sayısı ve o faaliyeti takip eden ardıl faaliyetler gösterilmiştir.

Çizelge 6.12 : Yazılım Proje Çizelgeleme İçin Örnek Uygulama.

Yamuk Bulanık Faaliyet											Yamuk Bulanık Faaliyet																		
No	Süreler								Kaynaklar	Ardıl Sayısı		Ardıl Faaliyetler		No	Süreler								Kaynaklar	Ardıl Sayısı		Ardıl Faaliyetler			
S	0	0	0	0	0	0	0	0	0	1	2			27	6	7	7	8	0	4	0	2	1	30					
1	5	6	6	7	3	0	0	0	0	1	3			28	10	12	12	14	2	2	4	4	1	30					
2	10	12	12	13	5	3	3	2	2	4	5			29	1	2	2	4	4	2	2	2	6	31	32	33	39	40	42
3	4	5	5	6	4	2	4	2	1	6				30	9	11	11	13	4	2	3	0	1	34					
4	7	8	8	10	4	2	2	2	1	6				31	10	13	13	14	5	0	3	2	1	37					
5	2	4	4	6	5	0	0	0	1	7				32	8	10	10	11	0	0	4	2	1	38					
6	3	4	4	5	4	2	2	2	1	8				33	6	8	8	9	4	3	2	0	1	35					
7	2	3	3	4	4	3	3	2	4	9	10	11	13	34	3	4	4	5	4	2	3	3	1	36					
8	2	3	3	5	0	4	2	0	1	14				35	4	6	6	7	2	2	0	4	1	41					
9	3	4	4	6	2	1	5	3	1	15				36	1	3	3	4	0	0	2	0	1	41					
10	4	5	5	6	4	0	0	0	1	12				37	3	4	4	5	0	0	2	2	1	41					
11	7	9	9	10	5	0	0	0	1	16				38	1	2	2	3	1	0	3	1	1	41					
12	3	4	4	5	3	0	0	1	1	16				39	2	3	3	4	0	0	2	2	1	41					
13	1	2	2	3	3	2	0	1	1	16				40	1	1	1	2	4	0	4	3	3	45	46	49			
14	1	2	2	3	3	0	3	2	1	16				41	3	4	4	5	0	4	0	0	1	43					
15	1	2	2	2	4	2	2	2	1	17				42	1	2	2	3	3	3	0	0	1	44					
16	3	4	4	5	5	2	2	2	2	18	19			43	1	2	2	3	3	0	3	0	2	45	46				
17	4	5	5	6	5	2	4	2	1	20				44	3	4	4	5	2	3	4	1	1	52					
18	4	5	5	6	4	2	4	3	1	20				45	4	5	5	6	2	2	4	2	1	47					
19	5	6	6	7	5	2	3	4	3	21	22	23		46	4	5	5	6	5	2	2	3	1	48					
20	5	7	7	9	4	0	0	0	1	24				47	3	4	4	5	2	0	4	2	1	52					
21	3	5	5	7	5	1	4	4	1	24				48	5	7	7	8	2	0	2	3	1	50					
22	4	5	5	6	0	3	0	0	1	24				49	6	8	8	10	4	2	3	2	1	51					
23	1	2	2	3	5	1	1	1	1	25				50	2	3	3	5	4	2	3	3	1	52					
24	8	10	10	12	2	1	5	3	3	26	27	28		51	2	4	4	7	4	2	3	3	1	53					
25	4	5	5	6	2	1	5	3	1	30				E	0	0	0	0	0	0	0	0							
26	4	6	6	8	0	0	4	3	1	29																			

Faaliyetler; yenilenebilir ve yenilenemeyen kaynaklar olmak üzere iki kategoriye sahip kaynağa gereksinim duyar. Çizelge 6.12’de kaynaklar, programcı ve kıdemli programcılar olmak üzere iki gruptan oluşmaktadır. Bununla beraber, aynı işin kıdemli programcı tarafından daha kısa sürede yapılabilmesi de söz konusudur. Böylece çoklu uygulama modlarının varlığı, problemin çoklu doğasını ifade eder.

Proje yöneticisi, 3 farklı kriteri minimize etmekle ilgilenmektedir: Proje tamamlanma süresi, (max) günlerle ifade edilebilir. Toplam proje maliyeti, (cost) dolarla ifade edilebilir. Kaynak pürüzsüzleştirme oranı, (Rs) ortalama kaynak kullanımı kaynak birimleri tarafından ifade edilebilen ortalama fark olarak hesaplanır. İlk 2 kriter, bulanık değerlerdir; son kriter ise gerçel değerdir. Wang’ın çalışmasında örnek problem, Bulanık Pareto çizelgeleme algoritmasıyla (FPSA) çözülmüş ve kaynakların maksimum elde edilebilir miktarlarında değişiklikler yapılarak bir duyarlılık analizi gerçekleştirilmiştir. Bizde yamuk bulanık faaliyet sürelerine uyarladığımız problemi TA ile çözerek belirtilen duyarlılık analizi yaparak kıyasladık. Çizelge 6.13’de örnek probleme ait B-KYM sonucu elde edilen sonuçlar Çizelge 6.14’de TA kullanılarak elde edilen sonuçlar sunulmuş, Çizelge 6.15’de Wang’ın çalışmasıyla yapılan duyarlılık analizi sonuçları ile karşılaştırılarak, hazırlanan “proje çizelgeleme” programının geçerli olduğu test edilmiştir.

Çizelge 6.13 : Bulanık KYM Sonuçları.

No	FEST	FECT	FLST	FLCT	SLACK	No	FEST	FECT	FLST	FLCT	SLACK
S	0 0 0 0	0 0 0 0	-77 0 0 77	-77 0 0 77	-77 0 0 77	27	67 87 87 105	73 94 94 113	42 98 98 152	50 105 105 158	-63 11 11 85
1	0 0 0 0	5 6 6 7	-77 0 0 77	-70 6 6 82	-77 0 0 77	28	71 93 93 113	81 105 105 127	36 93 93 148	50 105 105 158	-77 0 0 77
2	5 6 6 7	15 18 18 20	-70 6 6 82	-57 18 18 92	-77 0 0 77	29	81 105 105 127	82 107 107 131	50 105 105 158	54 107 107 159	-77 0 0 77
3	15 18 18 20	19 23 23 26	-53 21 21 95	-47 26 26 99	-73 3 3 80	30	82 107 107 131	91 118 118 144	54 107 107 159	67 118 118 168	-77 0 0 77
4	15 18 18 20	22 26 26 30	-57 18 18 92	-47 26 26 99	-77 0 0 77	31	82 107 107 131	92 120 120 145	70 120 120 170	84 133 133 180	-61 13 13 88
5	22 26 26 30	24 30 30 36	-47 26 26 99	-41 30 30 101	-77 0 0 77	32	82 107 107 131	90 117 117 142	72 122 122 170	83 132 132 178	-59 15 15 88
6	24 30 30 36	27 34 34 41	-41 30 30 101	-36 34 34 104	-77 0 0 77	33	91 118 118 144	97 126 126 153	67 118 118 168	76 126 126 174	-77 0 0 77
7	27 34 34 41	29 37 37 45	-36 34 34 104	-32 37 37 106	-77 0 0 77	34	97 126 126 153	100 130 130 158	76 126 126 174	81 130 130 177	-77 0 0 77
8	29 37 37 45	31 40 40 50	-24 46 46 114	-19 49 49 116	-69 9 9 85	35	100 130 130 158	104 136 136 165	81 130 130 177	88 136 136 181	-77 0 0 77
9	29 37 37 45	32 41 41 51	-25 45 45 113	-19 49 49 116	-70 8 8 84	36	92 120 120 145	93 123 123 149	84 133 133 180	88 136 136 181	-61 13 13 88
10	29 37 37 45	33 42 42 51	-32 37 37 106	-26 42 42 110	-77 0 0 77	37	90 117 117 142	93 121 121 147	83 132 132 178	88 136 136 181	-59 15 15 88
11	33 42 42 51	40 51 51 61	-26 42 42 110	-16 51 51 117	-77 0 0 77	38	82 107 107 131	83 109 109 134	85 134 134 180	88 136 136 181	-46 27 27 98
12	29 37 37 45	32 41 41 50	-21 47 47 114	-16 51 51 117	-66 10 10 85	39	82 107 107 131	84 110 110 135	84 133 133 179	88 136 136 181	-47 26 26 97
13	31 40 40 50	32 42 42 53	-19 49 49 116	-16 51 51 117	-69 9 9 85	40	104 136 136 165	105 137 137 167	88 136 136 181	90 137 137 182	-77 0 0 77
14	32 41 41 51	33 43 43 54	-19 49 49 116	-16 51 51 117	-70 8 8 84	41	82 107 107 131	85 111 111 136	85 133 133 179	90 137 137 182	-46 26 26 97
15	40 51 51 61	41 53 53 63	-16 51 51 117	-14 53 53 118	-77 0 0 77	42	85 111 111 136	86 113 113 139	90 137 137 182	93 139 139 183	-46 26 26 97
16	41 53 53 63	44 57 57 68	-14 53 53 118	-9 57 57 121	-77 0 0 77	43	86 113 113 139	87 115 115 142	93 139 139 183	96 141 141 184	-46 26 26 97
17	44 57 57 68	48 62 62 74	-9 57 57 121	-3 62 62 125	-77 0 0 77	44	105 137 137 167	108 141 141 172	108 151 151 192	113 155 155 195	-59 14 14 87
18	44 57 57 68	48 62 62 74	-9 57 57 121	-3 62 62 125	-77 0 0 77	45	105 137 137 167	109 142 142 173	96 141 141 184	102 146 146 188	-71 4 4 79
19	48 62 62 74	53 68 68 81	-3 62 62 125	4 68 68 130	-77 0 0 77	46	109 142 142 173	113 147 147 179	102 146 146 188	108 151 151 192	-71 4 4 79
20	53 68 68 81	58 75 75 90	4 68 68 130	13 75 75 135	-77 0 0 77	47	113 147 147 179	116 151 151 184	108 151 151 192	113 155 155 195	-71 4 4 79
21	53 68 68 81	56 73 73 88	6 70 70 132	13 75 75 135	-75 2 2 79	48	105 137 137 167	110 144 144 175	90 137 137 182	98 144 144 187	-77 0 0 77
22	53 68 68 81	57 73 73 87	7 70 70 131	13 75 75 135	-74 2 2 78	49	110 144 144 175	116 152 152 185	98 144 144 187	108 152 152 193	-77 0 0 77
23	58 75 75 90	59 77 77 93	13 75 75 135	16 77 77 136	-77 0 0 77	50	116 152 152 185	118 155 155 190	108 152 152 193	113 155 155 195	-77 0 0 77
24	59 77 77 93	67 87 87 105	16 77 77 136	28 87 87 144	-77 0 0 77	51	118 155 155 190	120 159 159 197	113 155 155 195	120 159 159 197	-77 0 0 77
25	67 87 87 105	71 92 92 111	44 100 100 154	50 105 105 158	-61 13 13 87	E	120 159 159 197	120 159 159 197	120 159 159 197	120 159 159 197	-77 0 0 77
26	67 87 87 105	71 93 93 113	28 87 87 144	36 93 93 148	-77 0 0 77						

Çizelge 6.14 : Örnek Çalışmanın Sonuçları.

B-KYM	
Kritik Yol	S - 1 - 2 - 4 - 5 - 6 - 7 - 10 - 11 - 15 - 16 - 17 - 19 - 20 - 23 - 24 - 26 - 28 - 29 - 30 - 33 - 34 - 35 - 40 - 48 - 49 - 50 - 51 - E
Bulanık Proje Tamamlanma Süresi	(120, 159, 159, 197)
Başlangıç Çözümler (MinSlack and ROT)	
MinSlack ile bulunan En iyi Çizelge	S - 1 - 2 - 4 - 3 - 5 - 6 - 7 - 10 - 9 - 11 - 8 - 14 - 13 - 12 - 15 - 16 - 17 - 18 - 19 - 20 - 22 - 21 - 23 - 24 - 26 - 28 - 27 - 25 - 29 - 30 - 39 - 41 - 33 - 38 - 34 - 35 - 42 - 43 - 31 - 36 - 32 - 37 - 40 - 48 - 49 - 50 - 45 - 46 - 47 - 44 - 51 - E
MinSlack ile Bulunan Sonuç	(181, 240, 240, 294)
R1. Kaynak için Rot ile bulunan En iyi Çizelge	S - 1 - 2 - 3 - 4 - 5 - 6 - 7 - 9 - 10 - 8 - 14 - 12 - 13 - 11 - 15 - 16 - 17 - 18 - 19 - 21 - 22 - 20 - 23 - 24 - 25 - 26 - 28 - 27 - 29 - 41 - 30 - 39 - 42 - 38 - 33 - 34 - 31 - 35 - 36 - 43 - 32 - 37 - 40 - 44 - 48 - 49 - 50 - 45 - 46 - 47 - 51 - E
R1. Kaynak için Rot ile bulunan Sonuç	(179, 237, 237, 293)
Tabu Arama Çözümler	
Tabu MinSlack ile bulunan En iyi Çizelge	S - 1 - 2 - 4 - 3 - 5 - 6 - 7 - 10 - 9 - 8 - 11 - 14 - 12 - 13 - 15 - 16 - 17 - 18 - 19 - 22 - 20 - 21 - 23 - 24 - 26 - 27 - 28 - 25 - 29 - 34 - 32 - 31 - 36 - 38 - 33 - 42 - 43 - 30 - 35 - 41 - 39 - 37 - 40 - 48 - 49 - 45 - 50 - 46 - 47 - 44 - 51 - E
Tabu MinSlack ile Bulunan Sonuç	(169, 224, 224, 275)
R1. Kaynak için Tabu Rot ile bulunan En iyi Çizelge	S - 1 - 2 - 4 - 3 - 5 - 6 - 7 - 10 - 9 - 8 - 11 - 12 - 13 - 14 - 15 - 16 - 18 - 17 - 19 - 20 - 22 - 21 - 23 - 24 - 27 - 28 - 25 - 26 - 29 - 34 - 35 - 30 - 43 - 42 - 31 - 41 - 32 - 33 - 39 - 38 - 36 - 37 - 40 - 48 - 44 - 49 - 50 - 45 - 46 - 47 - 51 - E
R1. Kaynak için Tabu Rot ile bulunan Sonuç	(171, 227, 227, 280)

Çizelge 6.15 : Önerilen TA İle Wang çalışmasının Kıyaslanması.

Wang's Çalışması					
Maksimum Eldeki Kaynak Miktarı	(6,6,5,4)	(7,6,5,4)	(6,7,5,4)	(6,6,6,4)	(6,6,5,5)
Bulanık Faaliyet Süreleri	(175, 231, 231, 283)	(174, 229, 229, 280)	(175, 231, 231, 283)	(165, 219, 219, 270)	(169, 224, 224, 275)
Kaynak pürüzsüzleştirme oranı	0.4933	0.5270	0.4933	0.6622	0.5946
Olası Proje Süresi	257	253	257	236	245
Bu Çalışmada					
Minslack	(181, 240, 240, 294)	(181, 240, 240, 294)	(181, 240, 240, 294)	(167, 221, 221, 272)	(175, 233, 233, 286)
TabuMinSlack	(169, 224, 224, 275)	(170, 224, 224, 274)	(169, 224, 224, 275)	(160, 212, 212, 261)	(164, 217, 217, 267)
ROT -R1	(179, 237, 237, 293)	(178, 235, 235, 290)	(179, 237, 237, 293)	(165, 219, 219, 272)	(173, 230, 230, 285)
Tabu ROT-R1	(171, 227, 227, 280)	(170, 225, 224, 276)	(171, 227, 227, 280)	(158, 211, 211, 262)	(166, 220, 220, 272)
ROT -R2	(179, 238, 238, 292)	(177, 234, 234, 287)	(179, 238, 238, 292)	(167, 221, 221, 272)	(173, 230, 230, 283)
Tabu ROT-R2	(174, 230, 230, 283)	(171, 225, 225, 279)	(179, 236, 236, 291)	(164, 218, 218, 270)	(165, 220, 220, 273)
ROT -R3	(176, 233, 233, 288)	(178, 235, 235, 288)	(176, 233, 233, 288)	(169, 223, 223, 276)	(170, 228, 228, 280)
Tabu ROT-R3	(169, 225, 225, 278)	(167, 221, 221, 271)	(169, 225, 225, 278)	(160, 213, 213, 264)	(163, 216, 216, 267)
ROT -R4	(178, 236, 236, 289)	(179, 235, 235, 287)	(177, 235, 235, 289)	(173, 228, 228, 280)	(172, 229, 229, 281)
Tabu ROT-R4	(168, 223, 223, 275)	(167, 221, 221, 271)	(168, 223, 223, 275)	(162, 215, 215, 266)	(163, 216, 216, 267)

7. DETERMİNİSTİK/BULANIK FAALİYET SÜRELİ ÇOK MODLU KAYNAK KISITLI PROJE ÇİZELGELEME (B-ÇM-KKPÇP)

7.1 Problemin Tanımı ve Matematiksel Programlama Gösterimi

Bu bölümde tez çalışmasının başlığını oluşturan ÇM-KKPÇ modellerinin çözüm yaklaşımları ve yöntemleri gösterilecektir. Bu bölüme kadar ki tüm çabalar ve adı anılan yöntemler, Bulanık ÇM-KKPÇ çözümü için bir basamak oluşturması amacıyla, temel dayanak olarak kullanılmıştır. Çok modlu kaynak kullanımlı projeler üzerindeki çalışmalar, Elmaghraby (1977) ile başlamıştır. Problemin tanımında, KKPÇ problemine göre sadece kaynak kullanımında birden fazla modun varlığına izin verilmesi, bir farklılık olarak ortaya çıkmaktadır. Bu şekilde, problem tanımının daha gerçekçi olması sağlanmıştır. Aşağıdaki gösterimde, mod indisi “m” kullanılarak, değişken ve parametrelere mod boyutu eklenmiştir. Yenilenemeyen kaynakların matematiksel gösterimini örneklemek amacı ile aşağıdaki gösterime, yenilenemeyen kaynaklar da dâhil edilmiştir. Ayrıca kullanılan formülasyonda, hem toplam proje maliyetinin en aza indirilmesi öngörülmüş; hem de proje süresinin toplam maliyete etkisi göz önünde bulundurulmuştur. Böylece projenin mümkün olan en kısa sürede bitirilmesi amaçlanmıştır.

$$\min Z = \sum_{m=1}^{M_j} \sum_{t=EFT_j}^{LFT_j} tX_{jmt} \quad (7.1)$$

$$\sum_{t=EFT_j}^{LFT_j} C_{jr} tX_{jt} \quad (7.2)$$

Kısıtlar

$$\sum_{m=1}^{M_j} \sum_{t=EFT_j}^{LFT_j} X_{jmt} = 1 \quad j = 1, \dots, J \quad (7.3)$$

$$\sum_{j=1}^J \sum_{m=1}^{M_j} k_{jmr} \sum_{\tau=t}^{t+d_{jm}-1} X_{jmr\tau} \leq K_r \quad r \in R, \quad t = 1, \dots, T \quad (7.4)$$

$$\sum_{m=1}^{M_j} \sum_{t=EFT_j}^{LFT_j} tX_{jmt} \leq \sum_{m=1}^{M_j} \sum_{t=EFT_j}^{LFT_j} (t-d_{jm})X_{jmt} \quad j=2,...,J, \quad i \in P_j \quad (7.5)$$

$$\sum_{j=1}^J \sum_{m=1}^{M_j} k_{jmr} \sum_{t=EFT_j}^{LFT_j} X_{jmt} \leq K_r \quad r \in N, \quad (7.6)$$

$$X_{jmt} \in 0,1 \quad j=1,...,J; \quad m=1,...,M_j; \quad t=EFT_j,...,LFT_j. \quad (7.7)$$

Notasyon : Matematiksel modeli formüle etmek için aşağıda indisler, parametreler ve karar değişkenleri tanımlanmıştır. Kullanılan sembollerin tanımları aşağıda verilmiştir:

t Zaman indisi $t=1,...,T$

j Faaliyet indisi $j=1,...,J$

R Yenilenebilir kaynaklar kümesi,

d_j “ j ” faaliyetinin süresi,

P_j “ j ” faaliyetinin öncüllerinin kümesi,

EFT_j “ j ” faaliyetinin en erken bitiş zamanı,

LFT_j “ j ” faaliyetinin en geç bitiş zamanı,

k_{jr} “ j ” faaliyetinin “ r ” kaynağından birim zaman kullanım miktarı,

K_r “ r ” yenilenebilir kaynağının birim zaman kullanım üst sınırı.

M_j “ j ” faaliyetinin mod adedi,

m Mod indisi ($m=1,...,M_j$),

N Yenilenemeyen kaynakların kümesi.

Denk. 7.1’in, amaç işlevi, proje süresini minimize etmektir. Denk. 7.2’nin amaç işlevi, proje çizelgeleme maliyetini hesaplamak amacıyla modele dahil edilmiştir. Denk. 7.3, kısıt kümesindeki her faaliyetin çizelgelenmesini sağlamaktadır.

Denk. 7.4, yenilenebilir kaynakların birim zaman içinde kullanımlarının üst sınırını göstermektedir. Denk. 7.5 ise, “j” faaliyeti ile bu faaliyetin öncülü olan “i” faaliyeti arasındaki öncüllük ilişkisinin yerine getirilmesi gereğini ifade etmektedir. Denk. 7.6 kısıt kümesi, yenilenemeyen kaynakların birim zaman içindeki tüketimlerinin üst sınırını göstermektedir. Denk. 7.7 ise, $\{0, 1\}$ değişkeni “ x_{jmt} ”nin tanımıdır. Karar değişkeni “ x_{jmt} ”nin üç indisli olması, değişken adedini tek modlu gösterime göre büyük oranda artırmaktadır. Bu nedenle, $\{EFT_j, LFT_j\}$ zaman aralıklarının, en iyi çözümü dışlamadan en dar şekilde tanımı, bu gösterimde daha da önem kazanmaktadır. Yani bir periyot içerisinde belirli bir kaynak, sınırsız kabul edilir; ancak sürecin tamamında bu kaynak sınırlıdır. Projenin sermaye bütçesi, bu kaynak tipine örnek olarak verilebilir.

KKPÇ problemlerinin klasik modelleri, gerçek dünya problemleri için yeterli değildir. Bu sebeple klasik KKPÇ modelleri; çok kategorili kaynaklar Weglarz (1980), faaliyetlerin çoklu icraa edilen modları Patterson vd. (1990) ve çok projeli performans ölçüleri Slowinski (1981, 1989) ile bağlantılı olarak genişletilmiştir. Proje çizelgelemenin bir diğer gerçekçi görünüşü de, faaliyet zaman parametrelerinin belirsizliğidir. Belirsizlik altında KKPÇ problemlerinin çözümü için, birçok stokastik yaklaşım öne sürülmüştür. (Örneğin Loostma (1966, 1989), Elmaghraby (1967), Gaul (1981)). Ancak bir çok sayısız projedeki yeni teknik ve metodolojilerin kullanımı geçmiş deneyimlere olan ilgiyi azaltmıştır. Geçmiş veri yoksunluğunun, stokastik yaklaşımların kullanılması için elverişli olamayacağı açıktır.

Bu bölümde literatürde yapılmış en ciddi çalışmalardan Hapke (1997) ve Wang (2004) çalışmaları referans kaynaklar olarak alınmış; yapmış oldukları çalışmalar, yamuk bulanık sayılara uyarlanarak, B-ÇM-KKPÇ problemlerine çözüm aşamaları düzenlenmiştir. Bu yapılan çalışma ile, onların buldukları sonuçlardan daha iyi çözümler bulunmuştur. Yazılım projesi çizelgeleme, projeye güzel bir örnek olarak verilebilir. Uygulamada, görülmektedir ki yazılım proje yöneticileri; kendi deneyimlerini, özel mühendislerin yeteneklerini ve yeni teknikler konusunda iyi bilgileri de hesaba katarak, belirli faaliyetlerin yaklaşık tamamlanma sürelerini sıklıkla tahmin edebilirler. Bu uzman görüşü benimseyen yöneticiler, belirli bir aralık içerisinde en olası değerlerle, mümkün proje tamamlanma süresinin aralığını belirleyebilirler. Faaliyet süreleri hakkındaki bu bilgi, doğal olarak bir bulanık sayı olarak ifade edilebilir. Belirsizlik altında bulanık sayılarla modellenen çizelgeleme,

yeni bir araştırma sahasıdır. İlk makale, bulanık PERT modeliyle ilgili olarak yapılmıştır (Prade (1979); Chanas ve Kamburowski (1981); Buckley (1989); Rommelfanger (1994)). Ker ve Walker (1989), McCahon ve Lee (1992) ve Fortemps (1998), bir iş atölyesi çizelgeleme problemi için bulanık aritmetikler uygulamışlardır. Bulanık zaman parametreleriyle daha fazla genel proje çizelgeleme problemi ise, Hapke ve Slowinski tarafından araştırılmıştır. Hapke ve Slowinski, bulanık zaman parametreleriyle ilgili olarak sezgisel öncelik kurallarını genelleştirmiştir. Daha sonra Hapke, bulanık paralel çizelgeleme prosedürünün metasezgisel aramada nasıl daha etkin bir şekilde kullanılabileceğini göstermiştir. Pratik uygulamalarda KKPC problemlerinin, genel klasik modelinde, hem bulanık zaman parametrelerine göre, hem de çok amaçlı olarak düşünülmesi daha gerçekçidir. Ancak, bu çalışma kapsamında çok amaçlı modellerle ilgilenilmeyecektir. Sadece, referans olması bakımından Hapke vd. (1997) tarafından yapılan çalışma incelenerek konuya açıklık getirilmeye çalışılacaktır. Hapke vd. (1997), bulanık çok amaçlı KKPC (MOCO) problemleri için bir çözüm önermiştir. Bu çözüm iki aşamada gerçekleştirilir. Birinci aşamada, çok amaçlı bulanık KKPC problemleri için pareto kümesinin bir örneği geliştirilmiştir. Daha sonra ikinci aşamada, en uyumlu çizelgeler için, örnek üzerinde aramayı organize etmekte, interaktif bir prosedür kullanılmıştır. KKPC problemleri NP-zor olduğu için, tek amaçlı durumda bile, çok amaçlı metasezgisel bir metod kullanılabilir. Baskın olmayan çizelgelerin bir kümesini türetmek için “Pareto Benzetim Tavlaması Metodu (PBTM)”, bulanık baskın olmayan çizelgelerden ibaret, örnek B-KKPC problemleriyle ilişkili olarak uyarlanmıştır. Önerilen yaklaşımın ikinci aşaması, “Işık Hüzmesi Arama (IHA)-Light Beam Search (LBS)” olarak isimlendirilen uygulamayla ilgilidir. LBS interaktif prosedürü Jaszkiwicz ve Slowinski (1997), en iyi uyumlu çizelgenin seçilmesinde, karar vericileri destekler ve üretilen çözüm kümesinin etkileşimli olarak analiz edilmesine olanak sağlar. Bu prosedür, çok amaç ölçütlerini genellemektedir. Çizelgeleme prosedürünün, PBTM metodunun ve IHA prosedürünün genelleştirilmesi, bulanık sayıların kıyaslanması için yaklaşık kuralların olası özelliklerine ihtiyaç duymaktadır.

Bu bölümde, bulanık kümeler kullanımıyla belirsizliğin modellenmesi, bulanık sayılar üzerine bazı işlemler, bulanık sayıların zayıf ve güçlü kıyaslama kuralları ve bulanık çok amaçlı kombinasyonları ve optimizasyon problemlerinin karakteristikleri sunulmuştur. Bulanık zaman parametrelili KKPC modelleri tanıtılmış ve bulanık

çizelgeleme prosedürüyle ilgili bazı konular tartışılmıştır. Daha sonra B-KKPÇ ve PSA metodunun uyarlaması sunulmuştur. LBS kullanılarak arama prosedürüyle en uygun çizelgenin belirlenmesi tanıtılmıştır. Bir sonraki bölümde, bir yazılım proje çizelgeleme problemi için sunulan yaklaşımın uygulanması belirtilmiştir.

7.2 Belirsizlik Modelleme

Bir temel X kümesinde A kümesi, $\mu_A : X \rightarrow [0,1]$ üyelik fonksiyonuyla tanımlanabilir. Eğer $x \in A$ ise, $\mu_A(x) = 1$ ve $x \notin A$ ise, $\mu_A(x) = 0$ dır. Belirsizlik ortamında x elemanı, A kümesine ait olabilir veya olmayabilir. Yukarıdaki mode, $[0,1]$ aralığında üyelik fonksiyonu şemalarında genişletilebilir. Bu üyelik fonksiyonun yüksek bir değeri yüksek olabilirliği gösterirken, düşük bir değeri ise, zayıf olabilirliği gösterir. Bulanık bir küme için aşağıdaki şu tanım yapılabilir (Dubois and Prade (1980):

$$\tilde{A} = \{x \in X \mid \mu_{\tilde{A}}(x) > 0\} \quad (7.8)$$

Burada $\mu_{\tilde{A}}$, $\tilde{A}$ nın üyelik fonksiyonu olarak isimlendirilir.

Diğer bir yararlı tanım ise, seviye kümesi veya bir seviye kesim kavramıdır. $\tilde{A}$, X de bulanık bir küme ve $\alpha \in [0,1]$ olsun. α –seviye kümesi veya $\tilde{A}$ nın α –seviye kesimi, bir kümedir.

$$\tilde{A}_\alpha = \{x \in X \mid \mu_{\tilde{A}}(x) \geq \alpha\} \quad (7.9)$$

$\tilde{A}_\alpha$, α -kesiminin alt ve üst sınırları sırasıyla $\inf_{x \in X} \mu_{\tilde{A}}(x)$ ve $\sup_{x \in X} \mu_{\tilde{A}}(x)$ ’ya eşittir.

Bir söbü (flat) bulanık sayı ise, bulanık kümenin özel bir türüdür. Kesinlikle, bir söbü bulanık sayı R gerçel sayılar kümesini (real line), bulanık bir alt kümesinde konveksliği normalleştirilmesi olarak tanımlanır. Örneğin Denk. 7.9 formülüyle tanımlanmış olana göre, $X=R$ olarak tanımlanır. Bir bulanık sayının kesin biçimi uzman kişi, tarafından tanımlanabilir ve belirlenebilir. Bulanık verinin uygun üyelik fonksiyonlarla getirmenin pratik bir yolu Rommelfanger (1990) tarafından önerilmiştir. Rommelfanger’in önerisine göre; bir uzman bazı önemli üyelik seviyelerini α , üzerinde parametre belirsizliği hakkında uzman iyimser ve kötümser bilgilerini vurgular. Örneğin;

$\alpha = 1; \mu_x = 1$ nin anlamı x değeri kesinlikle olabilir değerler kümesine ait olduğu anlamına gelir.

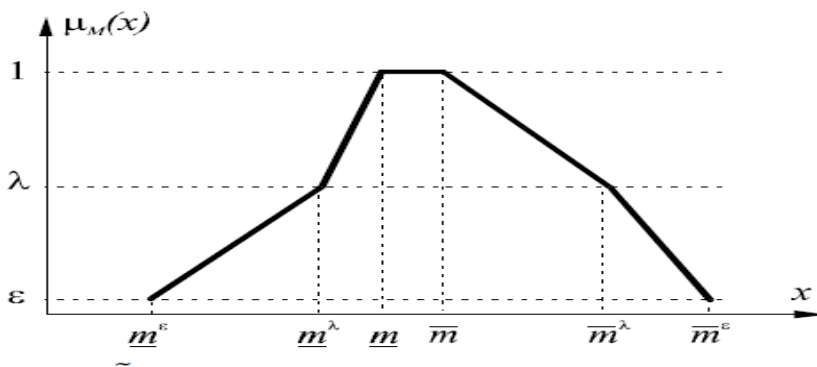
$\alpha = \lambda; \mu_x > \lambda$ 'nin anlamı uzmanın, olası(olabilir) değerler kümesine ait olduğunu iyi bir şansa sahip olarak $\mu_x \geq \lambda$ değerine sahip x değerini tahmin etmesidir.

$\alpha = \varepsilon; \mu_x < \varepsilon$ 'nin anlamı $\mu_x < \varepsilon$ 'lu x değerinin olası değerler kümesine ait olması yani çok küçük bir şansa sahip olmasıdır. Örneğin uzman, $\mu_x < \varepsilon$ 'lu x yeni tutan değerleri ihmal etmeyi arzular.

Hapke (1997) ve Wang (2004), çalışmalarında yamuk bulanık sayıların özel bir durumu olan söbü bulanık sayılarla ilgilenmişlerdir. Çalışmamızın bu kısmında, söbü bulanık sayıların yapısından kısaca bahsedilecek ve bunları yamuk bulanık sayılarla nasıl ifade edileceği gösterilecektir. Bir söbü bulanık sayı, 6 noktalı kabul edilen düzenle, 6 gerçel sayı tarafından sunulur. İdeal olarak, $\tilde{\mu}$ bir söbü bulanık sayı sembolik olarak aşağıdaki gibi listelenebilir.

$$\tilde{\mu} = \underline{\mu}^\varepsilon, \underline{\mu}^\lambda, \underline{\mu}, \bar{\mu}, \bar{\mu}^\lambda, \bar{\mu}^\varepsilon \quad (7.10)$$

$\lambda=0,6$ ve $\varepsilon=0,1$ olduğu varsayılırsa, 6 noktalı kabul edilen bir söbü bulanık sayı, Şekil 7.1 de sunulduğu gibi gösterilir.

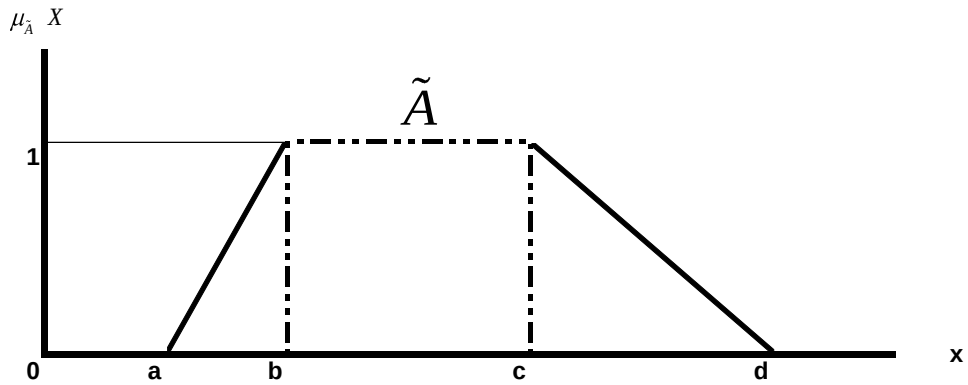


Şekil 7.1 : 6 noktalı $\tilde{M}$ söbü bulanık sayı gösterimi.

Söbü bulanık sayıları, yamuk bulanık sayıya dönüştürmek oldukça basittir. Söbü bulanık sayılarda, α -kesim seviyesi olan λ değerlerine karşılık gelen sayılarını iptal

ederek kolayca yamuk bulanık sayılar elde edilebilir. Burada $\mu_{\tilde{A}}$, $\tilde{A}$ yamuk bulanık sayısının üyelik fonksiyonudur ve şekilsel gösterimi şu şekildedir:

$$\mu_{\tilde{A}} x = \begin{cases} \frac{x-a}{b-a}, & a \leq x < b, \\ 1, & b \leq x < c, \\ \frac{d-x}{d-c}, & c \leq x < d, \\ 0, & \text{otherwise} \end{cases} \quad (7.11)$$



Şekil 7.2 : $\mu_{\tilde{A}} x$ yamuk bulanık sayı gösterimi.

7.3 Bulanık Aritmetik

$\tilde{A}$ ve $\tilde{B}$, x ve y uzayında bulanık sayılar olsun. * (asteriks) (“+”, “x”, “-“, “/”) temel aritmetik operatörleri göstermek üzere $\tilde{A} * \tilde{B}$ işlemi Zadeh’in genişletme prensibine göre karakterize edilebilir .

$$\mu_{\tilde{A} * \tilde{B}} z = \max_{x * y = z} \min \left[\mu_{\tilde{A}} x, \mu_{\tilde{B}} y \right] \quad (7.12)$$

Denk. 7.12’de sunulanan yerine bulanık sayılar üzerinde aritmetik işlemlerin durumu olduğunda;

$$\tilde{C}_\alpha = \tilde{A} * \tilde{B}_\alpha = \tilde{A}_\alpha * \tilde{B}_\alpha \text{ herhangi bir } \alpha \in [0,1] \quad (7.13)$$

Denk. 7.13’de, iki bulanık sayı üzerindeki genel bir aritmetik işlemi üzerinde α - kesimi üzerindeki aritmetik işlemlere eşit olduğunu gösterir. Hem $\tilde{A} * \tilde{B}_\alpha$, hem

$\tilde{A}_\alpha * \tilde{B}_\alpha$, klasik aralık analizinin miktar Ross (1995) kullanılarak yapılan aralık miktar işlemleridir. Böylece 4 nokta ile sunulan bulanık sayılar için aşağıdaki aritmetik işlemler tanımlanabilir. $\tilde{A} = a_1, a_2, a_3, a_4$ and $\tilde{B} = b_1, b_2, b_3, b_4$ olmak üzere;

$$\tilde{A} \oplus \tilde{B} = a_1, b_1, c_1, d_1 \oplus a_2, b_2, c_2, d_2 = a_1 + a_2, b_1 + b_2, c_1 + c_2, d_1 + d_2 \quad (7.14)$$

$$\tilde{A} \ominus \tilde{B} = a_1, b_1, c_1, d_1 \ominus a_2, b_2, c_2, d_2 = a_1 - a_2, b_1 - b_2, c_1 - c_2, d_1 - d_2 \quad (7.15)$$

$$\max \tilde{A}, \tilde{B} = [\max a_1, b_1, \max a_2, b_2, \max a_3, b_3, \max a_4, b_4] \quad (7.16)$$

$$\min \tilde{A}, \tilde{B} = [\min a_1, b_1, \min a_2, b_2, \min a_3, b_3, \min a_4, b_4] \quad (7.17)$$

$$\tilde{m}\tilde{a}\tilde{x} \tilde{A}, \tilde{B} = \max \underline{a}^\epsilon, \underline{b}^\epsilon, \max \underline{a}^\lambda, \underline{b}^\lambda, \max \underline{b}, \underline{a}, \max \bar{b}, \bar{a}, \max \bar{b}^\lambda, \bar{a}^\lambda, \max \bar{b}^\epsilon, \bar{a}^\epsilon \quad (7.18)$$

$$\tilde{m}\tilde{i}\tilde{n} \tilde{A}, \tilde{B} = \min \underline{a}^\epsilon, \underline{b}^\epsilon, \min \underline{a}^\lambda, \underline{b}^\lambda, \min \underline{b}, \underline{a}, \min \bar{b}, \bar{a}, \min \bar{b}^\lambda, \bar{a}^\lambda, \min \bar{b}^\epsilon, \bar{a}^\epsilon \quad (7.19)$$

7.4 İki Bulanık Sayının Kıyaslanması (Karşılaştırılması)

Baskın tanımlama, bir bulanık amaç uzayında kombinatorial arama, karar vericiye amaç fonksiyonları üzerinde bulanık skorları kıyaslaması gerekir. Bulanık skorları kıyaslamanın bir yolu da, bulanık sayıların anlam bilimle (semantic) ilişkilendirilmesidir. Aslında bulanık sayılar, kesin olmayan (imprecise) olasılık dağılımları gibi görülebilir. (bakınız Demspter (1967), Shafer (1976)). Bu perspektifle, iki bulanık sayının kıyaslanması; olasılık teorisinde iyi bilinen çıkarım tanımlamasıyla tutarlı olan ortalama değerin kıyaslanmasıyla belirlenebilir. Bu fikir Dubois ve Prade (1987) tarafından, bir bulanık sayı matematiksel gerçeklere dayalı olarak açıklanmıştır. Olabilirlik ölçütü, daha üst bir olasılık dağılımının yerini almaktadır. Bu gereklilik ölçütüyle bir alt olasılık dağılımı, Demspter Shafer'ın bakış açısıyla rassal değişkenlerin yerine konulabilir. Daha sonra bu şebeke kapalı aralığın sınırları olarak bir bulanık sayının ortalama değeri üst ve alt olasılık dağılımların beklenen değerleri olarak tanımlanabilir.

İki bulanık sayının kıyaslanması, bu sınırların aritmetik ortalamasının kıyaslanması (boil down) olur. İki bulanık sayının üyelik fonksiyonlarının karşılaştırılması,

belirlenen alan takasını (area compensation) temel alan diğer kıyaslama prensibi için kıyaslamalı olarak eşittir. **Alan takası metodunun ana avantajı, sadece iki bulanık sayıdan birinin daha büyük olduğunun değil, aynı zamanda bulanık sayılardan hangisinin diğerinden daha büyük olduğunun ve derecesinin ne olduğunun cevaplanmasını sağlar.** Bulanık çizelgeleme problemleriyle çalışan yazarlar (Hapke vd.), öncelik ve kaynak kısıtları ve güçlü kıyaslama kuralı (SCR) olarak isimlendirilen düşünciyi belirlemek için kullanılan çok daha sınırlı kıyaslama metodlarından ayırt edilebilsin diye zayıf kıyaslama kuralı (WCR) ismi altında bu kıyaslama metodunu kullanmışlardır.

7.4.1 Zayıf kıyaslama kuralı (WCR)

$\tilde{A}$ ve $\tilde{B}$ iki bulanık sayı olmak üzere, $S_L \tilde{A} \geq \tilde{B}$, $S_R \tilde{A} \geq \tilde{B}$, $S_L \tilde{A} \leq \tilde{B}$, $S_R \tilde{A} \leq \tilde{B}$ dir.

$\tilde{A}$ ve $\tilde{B}$ alanları Şekil 7.3’de gösterilen diğer üyelik fonksiyonları tarafından belirlenir.

$$S_L \tilde{A} \geq \tilde{B} = \int_{U_{A,B}} \inf_{x \in R} A_\alpha - \inf_{x \in R} B_\alpha d\alpha \quad (7.20)$$

$$S_R \tilde{A} \geq \tilde{B} = \int_{V_{A,B}} \sup_{x \in R} A_\alpha - \sup_{x \in R} B_\alpha d\alpha \quad (7.21)$$

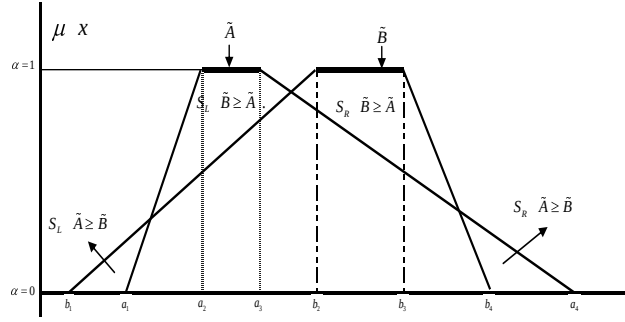
Burada

$$U_{\tilde{A}, \tilde{B}} = \alpha | \inf_{x \in R} A_\alpha \geq \inf_{x \in R} B_\alpha, \varepsilon \leq \alpha \leq 1 \quad (7.22)$$

$$V_{\tilde{A}, \tilde{B}} = \alpha | \sup_{x \in R} A_\alpha \geq \sup_{x \in R} B_\alpha, \varepsilon \leq \alpha \leq 1 \quad (7.23)$$

$S_L \tilde{A} \geq \tilde{B}$ ve $S_R \tilde{A} \geq \tilde{B}$, $S_L \tilde{A} \leq \tilde{B}$ ve $S_R \tilde{A} \leq \tilde{B}$ benzer şekilde tanımlanır.

$\tilde{A}$ ve $\tilde{B}$ ’nin potansiyel gerçekleşme (gerçekleştirme) olduğu olabirliliğiyle ilişkili alanlardır. $\tilde{A} \geq \tilde{B}$ hangisine $C(\tilde{A} \geq \tilde{B})$ derecesi olduğu $A^* \geq B^*$ ’deki alanların toplamından $A^* \leq B^*$ ’daki alanların toplamından farkı olarak hesaplanır.



Şekil 7.3 : Alan Takasını Temel Alan İki Bulanık Sayının Kıyaslanması.

Zayıf kıyaslama kuralına (WCR)'ye göre, $\tilde{A}$ bulanık sayısının $\tilde{B}$ 'ye eşit ve daha büyük olabilmesi için C derecesi, şu şekilde tanımlanır.

$$C \tilde{A} \geq \tilde{B} = \frac{1}{2} S_L \tilde{A} \geq \tilde{B} + S_R \tilde{A} \geq \tilde{B} - S_L \tilde{B} \geq \tilde{A} - S_R \tilde{B} \geq \tilde{A} \quad (7.24)$$

WCR kullanımıyla $\tilde{A}$ ve $\tilde{B}$ arasındaki 3 ilişki tanımlanabilir. Bulanık sayılar arasında, sırasıyla zayıf eşitsizlik, tam eşitsizlik ve eşitlik benzetilir.

$$\tilde{A} \geq \tilde{B} \quad \text{Eğer } C \tilde{A} \geq \tilde{B} \geq 0 \quad (7.25)$$

$$\tilde{A} > \tilde{B} \quad \text{Eğer } C \tilde{A} \geq \tilde{B} > 0 \quad (7.26)$$

$$\tilde{A} \approx \tilde{B} \quad \text{Eğer } C \tilde{A} \geq \tilde{B} = C \tilde{B} \geq \tilde{A} = 0 \quad (7.27)$$

Kolayca fark edilebilir ki $C \tilde{A} \geq \tilde{B} = -C \tilde{B} \geq \tilde{A}$ dır. Önerilen WCR bulanık sayıların orta-ağırlık merkezi değerlerinin kıyaslanmasıyla ilgilidir. (Chanas ve Forstems (1998) (Dubois ve Prade (1987)'de tanımlandığı şekliyle Demspter, “Shafer Teorisi”nin genel çerçevesini tanımlar. $\tilde{A}$ gibi bulanık bir sayının orta değeri, $[0, 1]$ aralığındadır. Burada kümülatif olabilirlik dağılımıyla (sol dağılım) ve kümülatif gerekseme gerekli dağılım sırasıyla sağ ve sol dağılımla ilgili ortalama değerlerdir.

$$\mathfrak{S} \tilde{A} = \frac{E_* \tilde{A} + E^* \tilde{A}}{2} \quad (7.28)$$

Fonksiyonu ve WCR fonksiyonu aşağıdaki özelliklerle birlikte yararlı olacaktır.

Özellik 1. Yamuk bulanık sayıların 4 noktalı sunumunda, $\tilde{A}$ bulanık sayısı için fonksiyon şu şekilde hesaplanabilir. Bir aralığın durulaştırılmasının doğal bir yolu, bu limitlerin aritmetik ortalamasını hesaplamaktır. Durulaştırma fonksiyonu aşağıdaki gibi tanımlanır (Chanas 1987, 1998).

$$\Im \tilde{A} = \frac{1}{4} \frac{1}{1-\varepsilon} \left[\lambda - \varepsilon \quad a_1 + a_4 + 1 - \lambda \quad a_2 + a_3 \right] \quad (7.29)$$

Özellik 2. Forstemp (1997) $\tilde{A}$ 'nın $\tilde{B}$ ye eşit ve daha büyük olduğu durum için $C \tilde{A} \geq \tilde{B}$ derecesi WCR'ye göre hesaplanır.

$$C \tilde{A} \geq \tilde{B} = \Im \tilde{A} - \Im \tilde{B} = \frac{E_* \tilde{A} + E^* \tilde{A}}{2} - \frac{E_* \tilde{B} + E^* \tilde{B}}{2} \quad (7.30)$$

Özellik 1 ve 2'den doğrudan doğruya $\tilde{A}$ ve $\tilde{B}$ 'nin 4 noktalı yamuk bulanık sayılardan oluşan $C \tilde{A} \geq \tilde{B}$ için formül takip eder. WCR'nin uygulanması, farklı bağımsız bulanık skorlar kıyaslandığında örnek olarak bulanık optimizasyon sonuçları için oldukça uygundur. Ancak, çizelgeleme prosedürüne uygulandığında Şekil 7.4'de görüldüğü şekilde uygun olmayan (infeasible) çizelgeler oluşabilir. Öncelik ve kaynak kısıtlamalı çizelgeleme prosedürünün bu halinden sakınmak için güçlü kıyaslama kuralları kullanılacaktır.

Tanım 2. Güçlü Kıyaslama kuralına göre (SCR)

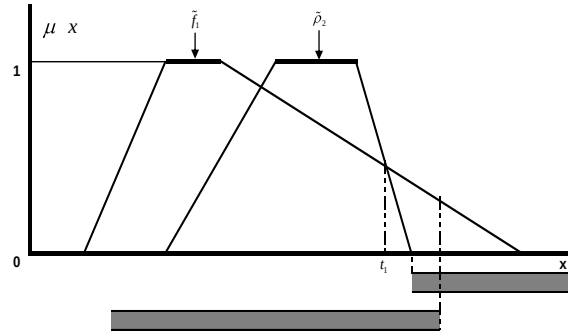
$$\tilde{A} \gg \tilde{B} \Leftrightarrow \max \tilde{A}, \tilde{B} = \tilde{A} \quad (7.31)$$

SCR'nin WCR'yi içerdiği açıktır. SCR'nin bir uygulaması gelecek bölümde gösterilecektir.

7.4.2 Güçlü Kıyaslama Kuralı (SCR)

Şekil 7.4'de zayıf kıyaslama kuralının sebep olduğu uygun olmayan bulanık bir çizelgeye örnek gösterilmiştir. Çizelgeleme prosedürünün $\tilde{\rho}_2$ hazırlık zamanında faaliyet 2 için bir kaynak ataması yapılacağı düşünüldüğü varsayılmıştır. Faaliyet 2, 1 faaliyetinin tek bir öncelidir ve $\tilde{f}_1$ de tamamlanmıştır. Zayıf kıyaslama kuralını kontrol etmek için 1 faaliyeti $\tilde{\rho}_2 \leq \tilde{f}_1$ arasındaki ilişki doğru olduğunda $\tilde{\rho}_2$ hazırlık

zamanında tamamlanmıştır. 1 faaliyetinin süresinin oldukça iyimser bir yansımasıdır. (t_1 zamanın ilerisinde ötesinde) sonuç çizelge uygun değildir çünkü 2 faaliyeti öncelik kısıtları gereği 1 faaliyeti tamamlanmadan önce başlamış olur. Sonuç olarak faaliyet 2, $\tilde{\rho}_2$ 'de çizelgelenemez (Şekil 7.4'e bakınız). Güçlü kıyaslama kuralı bu durumda $\tilde{A} \gg \max \tilde{A}, \tilde{B} = \tilde{A}$ olarak belirlenmiş olur (Hapke and Slowinski, 2000).



Şekil 7.4 : Olurlu Çözümü olmayan Bulanık Çizelge Örneği.

Yeni faaliyetler için elde edilebilir yenilebilir veya çift yönden kısıtlı kaynaklarda zaman anları nasıl belirlenir ve bulanık zaman anında nasıl tamamlanabilir? Bizim amacımız bulanık zaman parametreleriyle her bir gerçekleştirme için uygun çizelgeler veren bir prosedür kurmaktır. Aşağıdaki örnek, uygun olmayan çizelgelere önderlik edeni çözmek için WCR'nin bir uygulamasını gösterir. Farzedilen çizelgeleme prosedürü az hazırlık zamanında Z2 faaliyeti için R kaynağının bir uygulamasıdır. Z2 faaliyeti için R kaynağının bir tahsisi olduğu düşünülür. Z2 faaliyeti sadece Z1 öncülüne sahiptir, fZ1 de tamamlanır ise, aZ2 ilişkisi doğru olacaktır. Ancak fark edilecektir ki Z1 (t_1 üstünde) tamamlanma süresinin iyimser gerçeklemesi için sonuç çizelge uygun olmayabilir; çünkü Z2 faaliyeti Z1 öncülü tamamlanmadan önce başlamaktadır. Z2 faaliyeti, Az2 de bu sebeple çizelgelenmiş olamaz. Bu, öncelik kısıtının ihlalini yöneten (ilerletebilen) ilk sorunun çözümü için WCR'ın kullanımını gösterir. Benzer örnek kaynak kısıtlarının ihlalini ilerleten iki sorunun çözümü için WCR'nin kullanılabileceğini göstermektedir. Öncelik ve kaynak kısıtlarının ihlalinden sakınmak için WCR, SCR tarafından vekil tayin edilmek zorundadır. NP – zor proje çizelgeleme problemlerinde, bir paralel çizelgeleme prosedürü en etkin sezgisel bulunur. En önemli özelliği, ardışık düzende olası faaliyetler düşünüldüğünde kaynak atamaktır. Faaliyetler, düşünülen

kaynakların tahsis edilip edilmediğini doğruluyan bir prosedürdür. Bulanık zaman parametreleri üzerine paralel bir prosedür genelleştirmesi Hapke ve Slowinski (1996) tarafından önerilmiştir. Bu genelleme iki sayının çözümüyle ilgilidir.

7.5 ÇM-KKPCP'leri İçin Faaliyet Öncelik Kuralları

Önceki bölümlerde bahsedildiği için burada açıklanılmayacak olsa da bu kapsamda kullanılan öncelik kuralları MinSlack ve ROT kurallarıdır. Sürece başlamadan evvel öncelikle bazı gereksiz modların ve kaynakların elimine edilmesi gerekir. Bir j faaliyetinin bir $m \in M_j$ modunda gereksinim duyulan kaynaklar herhangi bir uygun çizelgede uygulanmıyor ise o faaliyet o modda işletilemiyor demektir. Mesela yenilenebilen bir $k \in K^p$ kaynağı için $r_{jkm}^p > R_k^p$ ise veya yenilenemeyen bir $k \in K^v$ kaynağı için $r_{jkm}^v + \sum_{j=1}^n \min_{\mu \in M_j} r_{jk\mu}^v > R_k^v$ olduğu durumdur. Bir j faaliyetinin bir $m \in M_j$ modu yetersiz olarak isimlendirilir eğer aynı faaliyetin diğer bir $\mu \in M_j$ modunda, m modunda daha büyük faaliyet süresine sahip olmayan ve herbirinde kaynaklardan daha çok kaynak birim gereksinimi olmayan bir mod vardır. Mesela $p_{j\mu} \leq p_{jm}$. Her bir $k \in K^p$ kaynağı için $r_{jk\mu}^p \leq k_{jkm}^p$ ve her bir yenilenemeyen $k \in K^v$ kaynağı için $r_{jk\mu}^v \leq r_{jkm}^v$ dir. Sonuç olarak bir yenilenemeyen k kaynağı $\sum_{j=1}^n \min_{\mu \in M_j} r_{jk\mu}^v \leq R_k^v$ ise gereksiz olarak isimlendirilir. Mesela k kaynağı atanan veya seçilen modun bağımsız olarak elde edilebiliyor olması yeterlidir. Kesinlikle, yetersiz kaynaklar kadar işletilemeyen ve gereksiz modlar optimal çözümü değiştirmeksizin çok modlu içerikten silinebilir.

7.6 Sayısal Örnek 1

Her biri $m=1,2$ olan iki modlu olarak icra edilebilen $n=4$ faaliyetli bir çok modlu proje düşünelim. Üç kaynak verilmiştir. Birinci tip kaynak $R_1^p=4$ kapasiteli yenilenemeyen bir kaynak ve ikinci ve üçüncü tip kaynak $R_2^v=13$ ve $R_3^v=14$ kapasiteli iki yenilenemeyen kaynak olsun. Bundan başka $m=1,2$ modunda, faaliyetler $i=1,...,4$ aşağıdaki Çizelge 7.1'de faaliyet süreleri ve kaynak kullanım miktarları gösterilmiştir (Brucker ve Knust 2005).

Çizelge 7.1 : Sayısal Örnek Verileri.

(i,m)	(1,1)	(1,2)	(2,1)	(2,2)	(3,1)	(3,2)	(4,1)	(4,2)
p_{im}	2	4	3	5	2	3	3	4
r_{ilm}^ρ	5	2	3	1	2	1	2	2
r_{i2m}^v	2	4	3	2	8	2	3	1
r_{i3m}^v	1	1	3	4	3	3	2	7

1 faaliyeti 1 modunda $r_{111}^\rho = 5 > R_1^\rho = 4$ olduğu için yenilenebilir 1 kaynağında işletilemez ve veri setinden silinmelidir. Bu durum faaliyet 3'ün 1 modunda $r_{321}^v + \sum_{j \neq 3, \mu \in M_j}^n r_{j2\mu}^v = 8+4+2+1=15 > R_2^v = 13$ olduğu için yenilemeyen 2 kaynağında da meydana gelmektedir. Bu modun silinmesinden sonra 2'nci kaynak geçersiz olur ve silinebilir. Daha sonra 4 faaliyetinin 2. modu yetersiz olduğundan ve bu modda kaynak 3'ün yetersiz olması nedeniyle elimine edilmelidir. Böylece verilen veriler aşağıdaki Çizelge 7.2'deki gibi yenilenebilen tek kaynaklı $R_1^\rho = 4$ ve hiç bir yenilenemeyen kaynak durumuna azaltılmış olur. Buna problem veri seti bu moda göre düzenlendiğinde;

Çizelge7.2 : Sayısal Örnek Verileri.

(i,m)	(1,1)	(2,1)	(2,2)	(3,1)	(4,1)
p_{im}	2	3	5	2	3
r_{ilm}^ρ	5	3	1	2	2

Bu örnekle işletilemeyen veya yetersiz modun sebep olduğu gereksiz kaynakların ve silinen gereksiz kaynakların neden olduğu kaynakların yeni yetersiz modlara sebep olduğu gösterilmiştir. Böylece elimine etme adımları iteratif olarak aşağıda gösterilmiştir.

Adım 1. Tüm yenilenemeyen modların silinmesi

Adım 2. Yetersiz kaynakların silinmesi.

Adım 3. Tüm geçersiz modların silinmesi. Eğer bir mod silinmiş ise Adım 2'ye git.

Her bir i faaliyetinin ÇM'lu durumundan klasik TM'lu duruma indirgendiğinde, minimum faaliyet süresi ve minimum kaynak gereksinimini şu şekilde tanımlayabiliriz; $p_i := \min_{m \in M_i} p_{im}$, $r_{ik}^\rho := \min_{m \in M_i} r_{ikm}^\rho \quad \forall k \in K^\rho$, $r_{ik}^\nu := \min_{m \in M_i} r_{ikm}^\nu \quad \forall k \in K^\nu$.

7.7 ÇM-KKPÇP'leri İçin Mod Seçim Kuralları

1. Modun Rassal olarak Seçilmesi : 1 den M_{ij} 'ye kadar düzgün dağılımdan seçilen m modu ile bir faaliyetin gerçekleştirilmesi.

2. MinDM (Minimum Faaliyet Süreli Mod Seçim Kuralı- Min duration mode) : Modlar arasında minimum faaliyet süresine sahip olanının seçilmesi.

$\min_m d_{ijm}, m = 1, \dots, M_{ij}$ değerine sahip modun belirlenmesi.

3. MaxDM (Maksimum Faaliyet Süreli Mod Seçim Kuralı- Max duration mode) : Modlar arasında maksimum faaliyet süresine sahip olanın seçilmesi.

$\max_m d_{ijm}, m = 1, \dots, M_{ij}$ değerine sahip modun belirlenmesi.

4. MinRDM (Minimum Kaynak Talebi Mod Seçim Kuralı- Minimum Resource Demand mode) Kuralı : Minimum kaynak gereksinime sahip faaliyetlere ait modun seçilmesi.

$\min_m \sum_{r=1}^R q_{ijm} d_{ijm}, m = 1, \dots, M_{ij}$ değerine sahip modun belirlenmesi.

5. MaxRDM (Maksimum Kaynak Talebi Mod Seçim Kuralı- Maximum Resource Demand mode) Kuralı : Maksimum kaynak gereksinime sahip faaliyetlere ait modun seçilmesi.

$\max_m \sum_{r=1}^R q_{ijm} d_{ijm}, m = 1, \dots, M_{ij}$ değerine sahip modun belirlenmesi.

6. MinRUM (Minimum Kaynak Kullanımı olan Modun Seçim Kuralı- Minimum Resource Usage mode) Kuralı : Minimum kaynak gereksinime sahip faaliyetlere ait modun seçilmesi.

$\min_m q_{ijm} \quad m = 1, \dots, M_{ij}$ değerine sahip modun belirlenmesi.

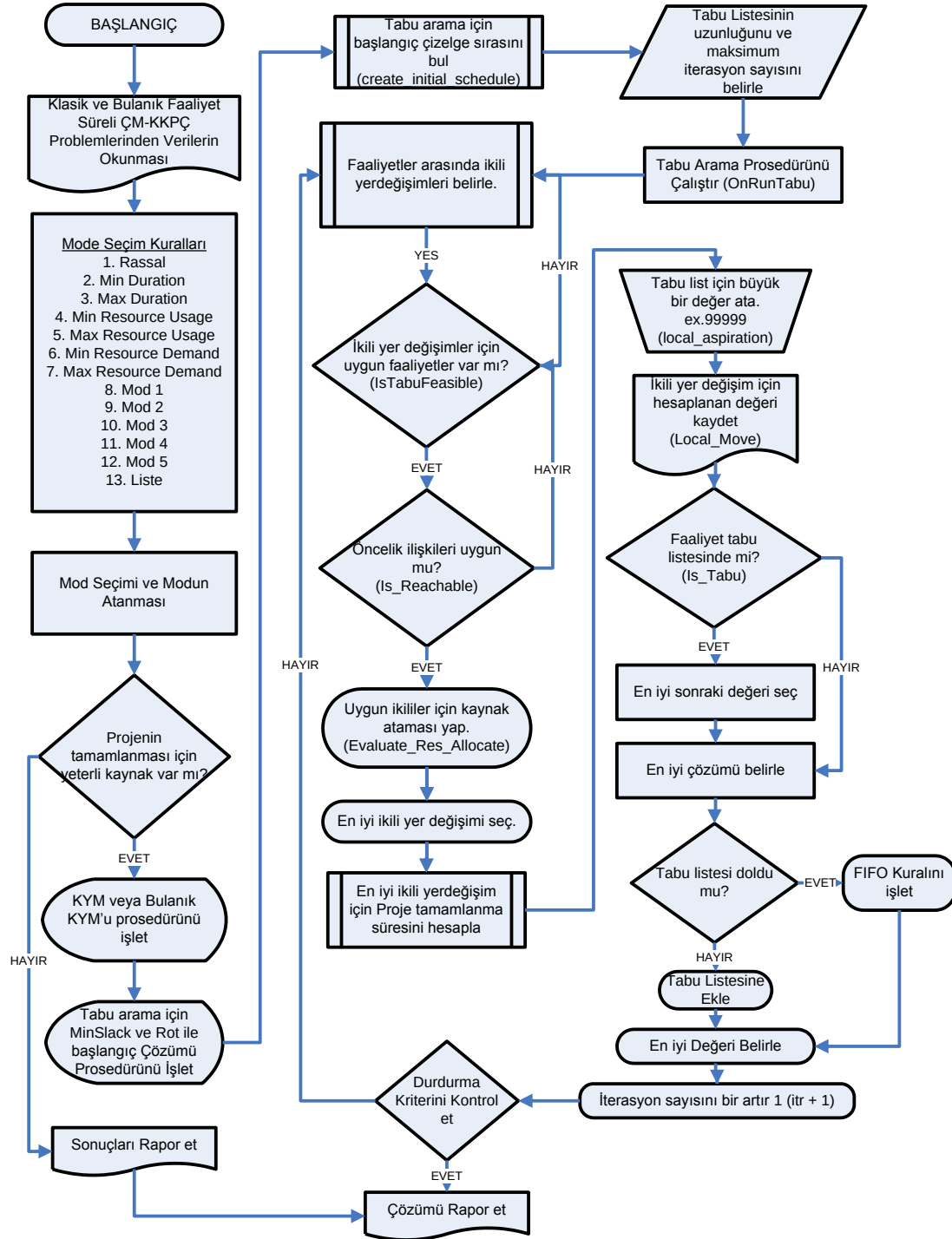
7. MaxRUM (Maksimum Kaynak Kullanımı olan Modun Seçim Kuralı- Maximum Resource Usage Mode) Kuralı: Maksimum kaynak gereksinime sahip faaliyetlere ait modun seçilmesi.

$\max_m q_{ijm} \quad m = 1, \dots, M_{ij}$ değerine sahip modun belirlenmesi.

8. Mode Seçimi : Modların isteğe bağlı olarak seçilmesi durumudur.

7.8 Deterministik/Bulanık ÇM-KKPÇM Uygulaması

Visual C#’da geliştirilen kaynak kısıtlı, bulanık etkinlik süreli proje çizelgeleme algoritması aşağıdaki gibi uygulanmaktadır. Şekil 7.5’de sunulan akış şemasının daha iyi anlaşılabilmesi için algoritmanın her bir adımı açıklanmaya çalışılacaktır.



Şekil 7.5 : K/B-ÇM-KKPÇM için Önerilen TA Algoritması.

7.9 Çok Modlu Durum için Sezgisel Yöntemler

ÇM-KKPÇ problemlerini sezgisel olarak çözmek için, sıklıkla iki aşamalı bir prosedür uygulanmaktadır. Birinci aşamada mod atamalarının belirlenmesi, diğer aşama ise ilgili bir çizelge hesaplamaktır. Bu hiyerarşik bir şekilde yapıldığında, ikincisi ise klasik TM-KKPÇP'nin çözülmesidir.

7.10 ÇM-KKPÇM İçin Sayısal Örnek 2

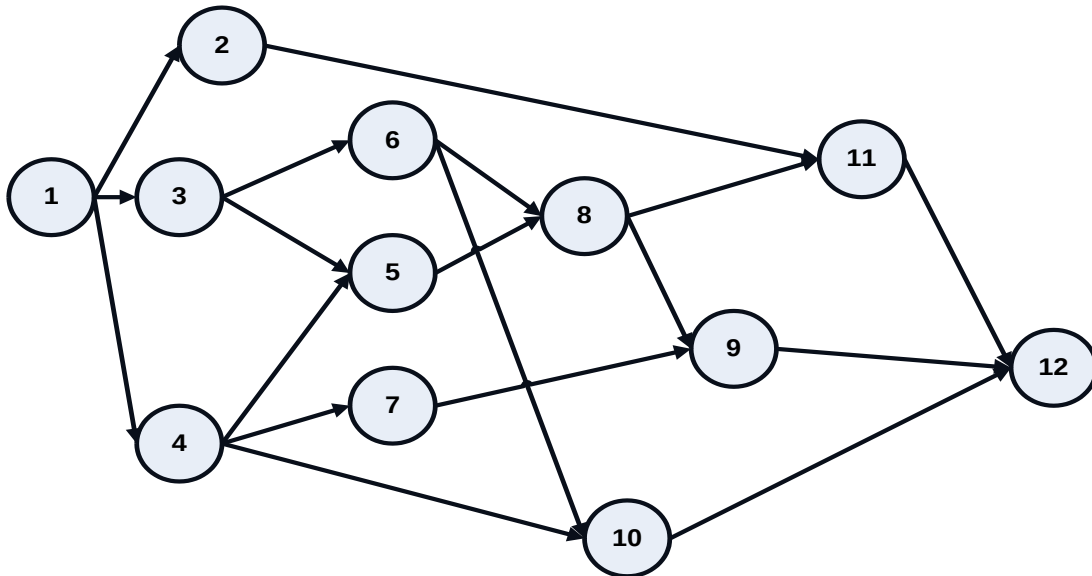
Literatürde PSPLIB web sayfasında <http://129.187.106.231/psplib/> dan J104_1.mm veri dosyası kullanılacaktır. Çizelge 7.3'de örnek probleme ait bilgiler ve Şekil 7.6'da şebeke yapısı verilmiştir. Yapay faaliyetlerle birlikte toplam 12 faaliyetten oluşan örnek problemde dört farklı kaynak kullanılmış olup, herbir kaynağın maksimum elde edilebilir miktarları sırasıyla; 9, 7, 59, 52'dir. Herbir faaliyet üç modda icra edilmektedir, her bir modda faaliyet süresi ve kaynak kullanım miktarları Çizelge 7.3'de gösterilmiştir.

Çizelge 7.3 : J104_1.mm veri dosyası

Faaliyet Toplam Sayısı :		12	4						
Kaynak Maks.Sayısı			R1 9	R2 7	NR1 59	NR2 52			
MODE 1	Süreler	1.Kaynak Türü Kullanımı	2.Kaynak Türü Kullanımı	3.Kaynak Türü Kullanımı	4.Kaynak Türü Kullanımı	Faaliyetlerin Ardıllarının Sayısı	Ardıl Faaliyetler		
1	0	0	0	0	0	3	2	3	4
2	4	9	0	6	0	1	11		
3	6	0	1	0	9	2	5	6	
4	4	4	0	8	0	3	5	7	10
5	7	0	6	8	0	1	8		
6	7	0	3	0	6	2	8	10	
7	2	5	0	8	0	1	9		
8	4	6	0	0	6	2	9	11	
9	2	7	0	0	10	1	12		
10	5	0	9	0	5	1	12		
11	5	9	0	8	0	1	12		
12	0	0	0	0	0	0			
		40	19	38	36				

MODE 2	Süreler	1.Kaynak Türü Kullanımı	2.Kaynak Türü Kullanımı	3.Kaynak Türü Kullanımı	4.Kaynak Türü Kullanımı
1	0	0	0	0	0
2	7	6	0	6	0
3	7	7	0	0	8
4	5	3	0	8	0
5	7	0	5	0	4
6	8	6	0	6	0
7	8	3	0	6	0
8	7	0	8	2	0
9	2	0	7	0	9
10	5	9	0	4	0
11	5	0	10	0	7
12	0	0	0	0	0
		34	30	32	28

MODE 3	Süreler	1.Kaynak Türü Kullanımı	2.Kaynak Türü Kullanımı	3.Kaynak Türü Kullanımı	4.Kaynak Türü Kullanımı
1	0	0	0	0	0
2	8	5	0	6	0
3	9	0	1	9	0
4	9	3	0	0	5
5	8	0	4	8	0
6	9	5	0	2	0
7	8	0	4	6	0
8	10	1	0	0	5
9	8	0	5	0	4
10	10	0	8	2	0
11	10	9	0	0	5
12	0	0	0	0	0
		23	22	33	19



Şekil 7.6 : J104_1. mm örnek verisine ait Şebeke Diyagramı.

7.11 Deterministik/Bulanık ÇM-KKPC Algortiması

Adım 1. Mod seçimi yapılması. Mod seçimi için; Mod seçim kurallarından biri belirlenerek mod seçimi yapılır.

1. Modun rassal olarak seçilmesi. Ancak uygun mod için yenilenebilir ve yenilenemeyen kaynakların yeterli olup olmadığına bakmak gerekir. Bir faaliyet için seçilen mod, yeterli kaynak olanlar arasından yapılmalıdır. Kaynakların yeterli olup olmadığına Adım 2'deki gibi karar verilir.
2. Minimum süre kuralı: Bir faaliyet için 3 mod var. 3 mod için de yeterli kaynak varsa, minimum faaliyet süresine sahip mod seçilir. Eğer modlarda aynı süreye sahip faaliyetler varsa, seçilen kaynak tipine göre minimum kaynak kullanıma sahip faaliyetli olan mod seçilir.
3. Maksimum süre: Maksimum faaliyet süresi olan modun seçimi. Eşit süreli mod varsa, minimum kaynak kullanımı olan mod seçilir.
4. Minimum kaynak kullanımı olan modun seçilmesi : Seçilen kaynak tipinde minimum kaynak ihtiyacı olan modun seçimi. Eğer eşit kaynak kullanımı olan faaliyetler varsa, minimum süreye sahip olan faaliyetin mod'u seçilir.
5. Maksimum kaynak kullanımı olan modun seçilmesi: Seçilen kaynak tipinde maksimum kaynak ihtiyacı olanın seçimi. Eğer eşit kaynak kullanımı olan faaliyetler varsa, maksimum süreye sahip olan faaliyetin mod'u seçilir.
6. Mod'un isteğe bağlı olarak seçilmesi: Tüm faaliyetler, mode 1 seçimine göre yapılırsa, mode 2 seçiminde tüm faaliyetler mode 2'de, mode 3 seçiminde tüm faaliyetler mode 3'te yapılmalıdır. Eğer seçilen mod'da bir faaliyet icra edilemiyorsa faaliyetin diğer mod'larından min süre ve min kaynak kullanımlı faaliyetlerin olduğu mod seçilir.
7. Minimum kaynak talebi olan modun seçimi : Bir faaliyetin ihtiyaç duyduğu kaynak miktarının toplamı ile faaliyet süresinin çarpımı sonucu oluşan değer minimum olan mod seçimi yapılır.
8. Maksimum kaynak talebi olan modun seçimi : Bir faaliyetin ihtiyaç duyduğu kaynak miktarının toplamı ile faaliyet süresinin sonucu oluşan değer maksimum olan mod seçimi yapılır.

Çizelge 7.4 : Örnek problemin Modları.

Mod 1				Mod 2				Mod 3			
1.Kaynak Türü	2.Kaynak Türü	3.Kaynak Türü	4.Kaynak Türü	1.Kaynak Türü	2.Kaynak Türü	3.Kaynak Türü	4.Kaynak Türü	1.Kaynak Türü	2.Kaynak Türü	3.Kaynak Türü	4.Kaynak Türü
0	0	0	0	0	0	0	0	0	0	0	0
9	0	6	0	6	0	6	0	5	0	6	0
0	1	0	9	7	0	0	8	0	1	9	0
4	0	8	0	3	0	8	0	3	0	0	5
0	6	8	0	0	5	0	4	0	4	8	0
0	3	0	6	6	0	6	0	5	0	2	0
5	0	8	0	3	0	6	0	0	4	6	0
6	0	0	6	0	8	2	0	1	0	0	5
7	0	0	10	0	7	0	9	0	5	0	4
0	9	0	5	9	0	4	0	0	8	2	0
9	0	8	0	0	10	0	7	9	0	0	5
0	0	0	0	0	0	0	0	0	0	0	0
40	19	38	36	34	30	32	28	23	22	33	19

Çizelge7.5 : Minimum Faaliyet Süreli Mod Seçimi.

Faaliyetler	MODE 1	Süreler	MODE 2	Süreler	MODE 3	Süreler	Mode	Min Süreler
1	1	0	2	0	3	0	1	0
2	1	4	2	7	3	8	1	4
3	1	6	2	7	3	9	1	6
4	1	4	2	5	3	9	1	4
5	1	7	2	7	3	8	2	7
6	1	7	2	8	3	9	1	7
7	1	2	2	8	3	8	1	2
8	1	4			3	10	1	4
9	1	2	2	2	3	8	2	2
10			2	5			2	5
11	1	5			3	10	1	5
12	1	0	2	0	3	0	1	0

Çizelge 7.6 : Maksimum Faaliyet Süreli Mod Seçimi.

Faaliyetler	MODE 1	Süreler	MODE 2	Süreler	MODE 3	Süreler	Mode	Max Süreler
1	1	0	2	0	3	0	3	0
2	1	4	2	7	3	8	3	8
3	1	6	2	7	3	9	3	9
4	1	4	2	5	3	9	3	9
5	1	7	2	7	3	8	3	8
6	1	7	2	8	3	9	3	9
7	1	2	2	8	3	8	3	8
8	1	4			3	10	3	10
9	1	2	2	2	3	8	3	8
10			2	5			2	5
11	1	5			3	10	3	10
12	1	0	2	0	3	0	3	0

Adım 2. Kaynakların yeterli olup olmadığına bakılır. Kaynak kullanımları kontrol edilir.

◆ Yenilebilir kaynaklar için Kontrol: $R1$ ve $R2$ maksimum elde edilebilir kaynak miktarı, $r1$ ve $r2$: Faaliyette kullanılan yenilebilir kaynak miktarı olmak üzere,

◆ Yenilenemeyen kaynaklar için Kontrol $NR1$ ve $NR2$ maksimum elde edilebilir kaynak miktarı, $nr1$ ve $nr2$: faaliyette kullanılan yenilenemeyen kaynaklar miktarı olsun.

1. Faaaliyetler 3 mod'dan birisiyle yapılmalı, bu sebeple de en az 1 mod için yeterli kaynak olmalıdır. Eğer hiç bir modda faaliyeti yapmak için yeterli kaynak yok ise, o proje tamamlanamaz yani infeasible'dir.
2. Bazı faaliyetler 3 moddan sadece birisiyle veya ikisiyle yapılabilir. Bu durumda geçerli mod, uygun olan modlarla yapılmalı yeterli kaynak miktarı olmayan modun veya modların seçimi yapılmamalıdır. Bizim programımızda eğer yeterli kaynak miktarı olmayan mod seçilmiş ise yeterli kaynak olmadığı rapor edilir.
3. $R1$ maksimum kaynak miktarı için, modlarda kullanılan $r1$ tipi kaynak kullanımı eşit veya küçük olmalıdır. Değilse faaliyet o modda yapılamaz. $R2$ içinde her bir faaliyetin modlarında kaynak kullanımları kontrol edilmelidir. Yani $r1$, $R1$ 'den fazla olamaz. ($R1 \geq r1$) ise, kabul değilse o modda yapılamaz.
4. Bir faaliyetin seçilen mode için ihtiyaç duyulan $NR1$ ve $NR2$ miktarları, ihtiyaç duyulan her bir faaliyetin $nr1$ ve $nr2$ toplamlarından az olmamalıdır.
5. Her faaliyet yalnızca ve yalnızca seçilen moda ait olarak işletilmelidir.

Örnek verilen J104_1.mm bulanık çok modlu KKPC problemi için bu algoritmanın tüm adımlarının bir döngü içerisinde çalıştırılarak sonuçlandırılması gerekirdi. Ancak bu çalışma kapsamında mod seçiminin etkilerini gözlemlemek istediğimiz için, mod seçimlerine göre sonuçları gruplandırarak programı kodlandırmayı daha uygun bulduk. Hangi modda hangi faaliyetlere göre işlem yapıldığı görülebilir. Uygun mod seçimi yapılarak belirlenen problem, daha önceki bölümde anlatılan bulanık/deterministik KKPC problemine dönüştürülmüş olur.

Adım 3. Seçilen mod'a göre öncelikle deterministik/bulanık KYM/BKYM'yi işletilir.

Konuya tekrar dönmek adına burada bir önceki bölümde ifade edilen adımları tekrar açıklamaya gerek görülmemiştir. Ancak açıklamalarda bulunulmuştur. Her bir mod değişiminde faaliyet süreleri de değiştiği için, KYM/BKYM'yi her defasında hesaplatmak gerekir. Uygun mod seçimi yapıldıktan sonra, her defasında KYM/BKYM çalıştırılmalıdır. KYM/BKYM yöntemi ile şebeke yapısı ve öncelik ilişkileri esasına göre faaliyetlerin deterministik ve bulanık ortamda EST/FEST (En erken başlama zamanı), LST/FLST (En geç başlama zamanı), EFT/FEFT (En erken tamamlanma zamanı), LFT/FLFT (En geç tamamlanma zamanı) ve aylak süreleri hesaplanır, Kritik etkinlikler ve kritik olmayan etkinlikler tanımlanır, proje kaynak ihtiyaçları dikkate alınmadan çizelgelenir ve KYM/FKYM'ye göre en küçük proje tamamlanma sırası ve süresi bulunur. (Çizelge 7.7 ve 7.8)

Çizelge 7.7 : Minimum Faaliyet Süreli Mod Seçimine göre KYM Sonuçları.

No.	Mod	Süreler	EST	ECT	LST	LCT	MinSlack
1	1	0	0	0	0	0	0
2	1	4	0	4	13	17	13
3	1	6	0	6	0	6	0
4	1	4	0	4	2	6	2
5	1	7	6	13	6	13	0
6	1	7	6	13	6	13	0
7	1	2	4	6	18	20	14
8	1	4	13	17	13	17	0
9	1	2	17	19	20	22	3
10	2	5	13	18	17	22	4
11	1	5	17	22	17	22	0
12	1	0	22	22	22	22	0

Çizelge7.8 : Maksimum Faaliyet Süreli Mod Seçimine göre KYM Sonuçları.

No.	Mod	Süreler	EST	ECT	LST	LCT	MinSlack
1	3	0	0	0	0	0	0
2	3	8	0	8	20	28	20
3	3	9	0	9	0	9	0
4	3	9	0	9	1	10	1
5	3	8	9	17	10	18	1
6	3	9	9	18	9	18	0
7	3	8	9	17	22	30	13
8	3	10	18	28	18	28	0
9	3	8	28	36	30	38	2
10	2	5	18	23	33	38	15
11	3	10	28	38	28	38	0
12	3	0	38	38	38	38	0

Adım 4. Başlangıç çözümü bulunması: Başlangıç çözümü, sezgisel bir kural kullanılarak belirlenebilir. Örneğin ROT ve MINSLACK kurallarından birisi seçilir. Elde edilen çizelge sırası, tabu algoritması için başlangıç çözümü olarak seçilir. Şimdiye kadar bulunan en iyi sıralama (Sb), uygulanabilir ardışıklık Si olarak seçilir. (Çizelge 7.9)

Çizelge 7.9 : Minimum Faaliyet Süreli Mod Seçimine göre MinSlack Sonuçları.

j	1	3	5	6	8	11	12	4	9	10	2	7
Minslack	0	0	0	0	0	0	0	2	3	4	13	14
Süre	0	6	7	7	4	5	0	4	2	5	4	2
Kaynak	0 0 0 0	0 1 0 9 0 5 0	4 0 3 0 6 6 0 0	6 9 0 8 0 0 0 0	0 4 0 8 0 0 7 0	9 9 0 4 0 9 0 6 0 5 0 8 0						
EST	0	0	6	6	13	17	22	0	17	13	0	4
Tstart	0	0	6	13	20	24	34	0	24	29	4	8
TSON = Süre+Tstart	0	6	13	20	24	29	34	4	26	34	8	10
TNOW	0	0	4	6	8	10	13	20	24	26	29	34
Kaynak Düzeyi	9 7 59 52	9 7 59 52	9 6 51 43 0 7 45 43 9 2 45 39 9	2 37 39 9	7 37 39 9 7 37 33 9 7 37 27 0	7 29 18 9 7 29 18 9 7 25 18						
Kaynak Kullanımı	0 0 0 0	9 6 59 43 0 6 45 43 0 2 45 39 4 2 37 39					9 4 37 33 3 7 37 27 0 7 29 27				0 7 25 18	
Kaynak Kullanımı		5 6 51 43						0 0 29 18				
Kaynak Kullanımı												
Çizelgelenebilecek Faal. Aday Liste	1	3,4,2	2,7	5,6,7	6,7	6	6	8, 10	11,9,10	10	10	12
Çizelgelenen Faaliyetler	1	3,4	2	5	7	-	6	8	11,9	-	10	12
İterasyon	1	2	3	4	5	6	7	8	9	10	11	12

Adım 5. Tabu aramasında kullanılan değişkenlere değer atanır. Tabu listesi genişliği için değer belirlenir. Örnek için, $\sqrt{n-2} = \sqrt{12-2} \cong 3$ olarak belirlenir. Durdurma kriteri olarak, maksimum iterasyon sayısı için değer belirlenir.

Adım 6. TA prosedürü işletilir.

Adım 7. Durdurucu ölçütler, denetlenir. Maksimum iterasyon sayısına ulaşıldı mı? Eğer maksimum iterasyon sayısına ulaşılmadıysa, Adım 6'ya gidilir. Maksimum iterasyon sayısına ulaşıldığı anda, şimdiye kadar bulunan en iyi çözüm, sonuç çözümü olarak rapor edilir.

Adım 8. Toplam proje maliyeti bulunur. Son olarak, her bir kaynak tipi için brüt birim maliyeti belirlenir. Kaç kaynak tipi varsa, o kadar değer girilmelidir. Daha sonra girilen maliyetlerle projenin toplam tamamlanma maliyeti ve proje tamamlanma zamanı hesaplanarak, bulunan sonuçlar rapor edilir. Bir faaliyetin toplam proje maliyeti, seçilen modda; (faaliyetin süresi x kaynak kullanım miktarı x birim brüt kaynak kullanım maliyeti) çarpımı ile hesaplanarak ve tüm faaliyetlerin

maliyetleri toplanarak bulunacaktır. Mod deęiřiminden kaynaklanan toplam proje maliyetlerinin deęiřimi grlebilir.

Adım 9. Adım 1’e dnlr. Her bir mod seim kuralı iin belirtilen algoritma tekrar edilmelidir. Bylece minimum proje tamamlanma sresi ve toplam proje maliyeti bulunarak en uygun czm belirlenmelidir.

TA’sının izledięi yol, nemli lde bařlangı czmne baęlıdır. Dolayısıyla, TA’sı kullanarak belirlenen czm de yine bařlangı czmyle iliřkilidir. Daha iyi czmn bulunacaęı farklı yolları bulmak iin, farklı bařlangı czmleri ile yukarıda aıklanan sre defalarca tekrarlanabilir. Sezgisel yntemlerle en iyi czmn saęlanıp saęlanamadıęı, bir karřılařtırma yapılamadıęı srece kesin olarak bilinemez. Ancak, bir sezgisel yntemin etkinlięi hakkında řu genel yargı geerlidir: İyi bir sezgisel yntem, byk bir olasılıkla her zaman en iyiye yakın veya nceden tespit edilmiř deęerlendirme ltlerine gre yeterli sayılabilecek czmleri saęlayan yntemdir. Bu nedenle optimizasyon yntemlerinin, en iyi czmn saęlanması amacıyla geliřtirilmesine karřın, sezgisel yaklařımda ama, etkinlik ve gvenilirlięi yksek olan “en iyi czm stratejisinin” geliřtirilmesidir.

Bu blmde kaynak kısıtları altında ok modlu proje izelgeleme problemi ve deterministik, bulanık sayılar tarafından modellenen faaliyetlerin belirsiz zaman parametreleri proje izelgeleme maliyeti ile birlikte ok amalı olarak czm prosedr sunulmuřtur. Birinci ařamada, TA metasezgisel prosedr, yaklařık olarak WCR baskın olmayan izelgelerin kmesinin temsil edildięi bir rnek retmekte kullanılmıřtır. Uygun izelgeler, (czmler) bulanık zaman parametreleri kullanmak iin MinSlack ve ROT sezgisellerini genelleřtirmek iin kullanılan yapıdadır. Bu; genelleřtirilen bulanık sayılar zerine bazı aritmetik iřlemlerin ve bulanık sayılar iin zel kıyaslama kurallarının kullanımıyla ilgilidir. zellikle de, WCR baskın olmayan izelgelerin tasarımı, nerilen kıyaslama kurallarından biri kullanılarak tanımlanmıřtır. Etkileřimli prosedr WCR baskın olmayan izelgelerin rnekleri zerine bulanık amalarla arasındaki rekabetin en iyi uzlařıyı belirleyen izelgeyi aramak iin organize edilmiřtir. Bunu mmkn kılmak iin, bir bařarım byklę olan fonksiyon bulanık ama uzayı geniřletir. Bu yaklařım, hipotetik sanal bir yazılım proje izelgeleme problemine uygulanmıřtır. Sunulan metedoloji, her ynyle geneldir ve herhangi bir bulanık ok amalı kombinatorial optimizasyon problemi czmek iin uygulanabilir.

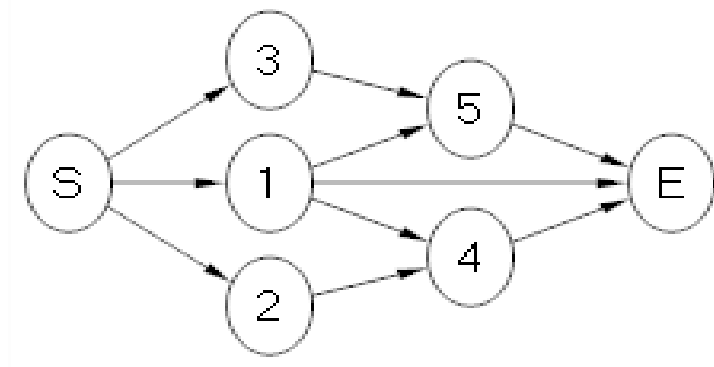
7.12 Bulanık Sayısal Örnek 3

Bir ÇM-KKPC modelinde, öncelik ilişkileri olan $i=1,...,e,...,I$ faaliyetlerinden oluşan tek bir proje olduğunu düşünüyoruz ve (e, i) faaliyetleri arasında e, i faaliyetinin önceli olacak şekilde düzenlenmiştir. Herbir i faaliyeti m_i modundan birisiyle icra edilmelidir. Her bir faaliyet mod kombinasyonu sabit bir faaliyet süresine ve R tipi yenilenebilir kaynak tipinden bir veya daha fazla sabit bir miktara gereksinim duyularak icra edilir.

Çizelge 7.10'da ÇM-KKPC modeli için veri kümesi sunulmuştur. Başlangıç ve bitiş yapay faaliyetlerinden oluşan 7 faaliyetten oluşmuştur. Modlar, faaliyet süreleri kıyaslama yapılabilmesi amacıyla klasik faaliyet süreleri, klasik faaliyet sürelerinin tekrarlanması ve uyarlanan bağımsız yamuk bulanık sayılardan oluşan faaliyet süreleri a, b ve c olmak üzere üç grupta temsil edilmiş, üç tip kaynak kullanımı (işgücü, maliyet, malzeme) ve herbir faaliyetin ardılı çizelgede sunulmuştur. Bu verileri kullanarak ve öncelikler gereği oluşturulan proje şebekesi Şekil 7.7'de olup problemin çözümleri takip eden çizelgelerde sunulmuştur.

Çizelge 7.10 : Deterministik/Bulanık Bir MM-RCPS modelin Örnek Veri Kümesi.

Faaliyetler	Modlar	(a)	(b)	(c)	Kaynak Kullanımı		
		Klasik Faaliyet Süreleri	Tekrarlı Bulanık Faaliyet Süreleri	Rassal Bulanık Faaliyet Süreleri	Yenilenebilir Kaynak R1	Yenilenemeyen Kaynak NR1	Yenilenemeyen Kaynak NR2
S	1	0	0 0 0 0	0 0 0 0	0	0	0
1	1	12	12 12 12 12	12 14 14 16	3	3	5
	2	15	15 15 15 15	15 16 16 17	2	4	4
	3	18	18 18 18 18	18 19 19 20	4	3	3
2	1	5	5 5 5 5	5 6 6 7	3	2	5
	2	11	11 11 11 11	11 12 12 13	4	5	2
	3	13	13 13 13 13	13 14 15 16	2	4	3
3	1	5	5 5 5 5	5 7 7 8	3	4	2
	2	14	14 14 14 14	14 15 15 16	2	5	4
4	1	15	15 15 15 15	15 16 17 18	4	2	3
	2	12	12 12 12 12	12 13 13 14	3	5	2
	3	8	8 8 8 8	8 9 9 9	2	4	3
5	1	13	13 13 13 13	13 14 14 15	3	5	3
	2	12	12 12 12 12	12 13 13 14	2	6	4
	3	15	15 15 15 15	15 16 16 18	3	2	3
E	1	0	0 0 0 0	0 0 0 0	0	0	0



Şekil 7.7 : Bir MM-RCPS modelinin şebeke diyagramı.

Çizelge 7.11 : Örnek Proje KYM/B-KYM Çözümü.

No	Süre	EST	ECT	LST	LCT	SLACK
S	0	0	0	0	0	0
1	12	0	12	0	12	0
2	5	0	5	11	16	11
3	5	0	5	7	12	7
4	8	12	20	16	24	4
5	12	12	24	12	24	0
E	0	24	24	24	24	0

No	Süre	FEST	FECT	FLST	FLCT	SLACK
S	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
1	12 12 12 12	0 0 0 0	12 12 12 12	0 0 0 0	12 12 12 12	0 0 0 0
2	5 5 5 5	0 0 0 0	5 5 5 5	11 11 11 11	16 16 16 16	11 11 11 11
3	5 5 5 5	0 0 0 0	5 5 5 5	7 7 7 7	12 12 12 12	7 7 7 7
4	8 8 8 8	12 12 12 12	20 20 20 20	16 16 16 16	24 24 24 24	4 4 4 4
5	12 12 12 12	12 12 12 12	24 24 24 24	12 12 12 12	24 24 24 24	0 0 0 0
E	0 0 0 0	24 24 24 24	24 24 24 24	24 24 24 24	24 24 24 24	0 0 0 0

No	Süre	FEST	FECT	FLST	FLCT	SLACK
S	0 0 0 0	0 0 0 0	0 0 0 0	-6 0 0 6	-6 0 0 6	-6 0 0 6
1	12 14 14 16	0 0 0 0	12 14 14 16	-6 0 0 6	10 14 14 18	-6 0 0 6
2	5 6 6 7	0 0 0 0	5 6 6 7	8 12 12 17	15 18 18 22	8 12 12 17
3	5 7 7 8	0 0 0 0	5 7 7 8	2 7 7 13	10 14 14 18	2 7 7 13
4	8 9 9 9	12 14 14 16	20 23 23 25	15 18 18 22	24 27 27 30	-1 4 4 10
5	12 13 13 14	12 14 14 16	24 27 27 30	10 14 14 18	24 27 27 30	-6 0 0 6
E	0 0 0 0	24 27 27 30	24 27 27 30	24 27 27 30	24 27 27 30	-6 0 0 6

Çizelge 7.12 : Örnek Proje MinDur Kuralına göre MinSlack Çözümü.

No	Süre	Başlangıç Zamanı	Bitiş Zamanı	Öncelleri	Kaynak Kullanımı	Kalan Kaynak Miktarı
0	0	0	0	-	0 0 0	8 25 21
1	12	0	12	0	3 3 5	5 22 16
3	5	0	5	0	3 4 2	2 18 14
2	5	5	10	0	3 2 5	2 16 9
5	12	12	24	1	2 6 4	6 10 5
4	8	12	20	1	2 4 3	4 6 2
6	0	24	24	5	0 0 0	8 6 2

No	Süre	Başlangıç Zamanı	Bitiş Zamanı	Öncelleri	Kaynak Kullanımı	Kalan Kaynak Miktarı
0	0 0 0 0	0 0 0 0	0 0 0 0	-	0 0 0	8 25 21
1	12 12 12 12	0 0 0 0	12 12 12 12	0	3 3 5	5 22 16
3	5 5 5 5	0 0 0 0	5 5 5 5	0	3 4 2	2 18 14
2	5 5 5 5	5 5 5 5	10 10 10 10	0	3 2 5	2 16 9
5	12 12 12 12	12 12 12 12	24 24 24 24	1	2 6 4	6 10 5
4	8 8 8 8	12 12 12 12	20 20 20 20	1	2 4 3	4 6 2
6	0 0 0 0	24 24 24 24	24 24 24 24	5	0 0 0	8 6 2

No	Süre	Başlangıç Zamanı	Bitiş Zamanı	Öncelleri	Kaynak Kullanımı	Kalan Kaynak Miktarı
0	0 0 0 0	0 0 0 0	0 0 0 0	-	0 0 0	8 25 21
1	12 14 14 16	0 0 0 0	12 14 14 16	0	3 3 5	5 22 16
3	5 7 7 8	0 0 0 0	5 7 7 8	0	3 4 2	2 18 14
2	5 6 6 7	5 7 7 8	10 13 13 15	0	3 2 5	2 16 9
5	12 13 13 14	12 14 14 16	24 27 27 30	1	2 6 4	6 10 5
4	8 9 9 9	12 14 14 16	20 23 23 25	1	2 4 3	4 6 2
6	0 0 0 0	24 27 27 30	24 27 27 30	5	0 0 0	8 6 2

Çizelge 7.13 : Örnek Proje Min/Maks Kaynak Talebi Kuralına göre Mod Seçimi.

Faaliyetler	Modlar	Klasik Faaliyet Süreleri	Tekrarlı Bulanık Faaliyet Süreleri	Rassal Bulanık Faaliyet Süreleri	Kaynak Talebi	Bulanık Kaynak Talebi	Maks. Kaynak Talebi	Min Kaynak Talebi
S	1	0	0 0 0 0	0 0 0 0	0	0,0,0,0	1	1
1	1	12	12 12 12 12	12 14 14 16	132	132,154,154,176		1
	2	15	15 15 15 15	15 16 16 17	150	150,160,160,170		
	3	18	18 18 18 18	18 19 19 20	180	180,190,190,200	3	
2	1	5	5 5 5 5	5 6 6 7	50	50,60,60,70		1
	2	11	11 11 11 11	11 12 12 13	121	121,132,132,143	2	
	3	13	13 13 13 13	13 14 15 16	117	117,126,135,144		
3	1	5	5 5 5 5	5 7 7 8	45	45,63,63,72		1
	2	14	14 14 14 14	14 15 15 16	154	154,165,165,176	2	
4	1	15	15 15 15 15	15 16 17 18	135	135,144,153,162	1	
	2	12	12 12 12 12	12 13 13 14	120	120,130,130,140		
	3	8	8 8 8 8	8 9 9 9	72	72,81,81,81		3
5	1	13	13 13 13 13	13 14 14 15	143	143,154,154,165		
	2	12	12 12 12 12	12 13 13 14	144	144,156,156,168	2	
	3	15	15 15 15 15	15 16 16 18	120	120,128,128,144		3
E	1	0	0 0 0 0	0 0 0 0	0	0,0,0,0	1	1

Çizelge 7.14 : Örnek Proje Mod Seçim Kuralları.

Mod Seçim Kuralları Faaliyetler	S	1	2	3	4	5	E	Proje Çizelgeleme Maliyeti				Durulaştırma
Random	1	3	3	2	3	3	1	643	690	699	745	745
Min_dur	1	1	1	1	3	2	1	443	514	514	567	567
Max_dur	1	3	3	2	1	3	1	706	753	771	826	826
Min_Res_R1	1	2	3	2	3	2	1	637	688	697	739	739
Max_Res_R1	1	3	2	1	1	1	1	624	683	692	742	742
Min_Res_Dem	1	1	1	1	3	3	1	419	486	486	543	543
Max_Res_Dem	1	3	2	2	1	2	1	734	787	796	849	849
1	1	1	1	1	1	1	1	505	575	584	645	645
2	1	2	2	2	2	2	1	689	743	743	797	797
3	1	3	3	1	3	3	1	534	588	597	641	745
Min_Res_NR1	1	1	1	1	1	3	1	482	549	558	624	624
Max_Res_NR1	1	2	2	2	2	2	1	689	743	743	797	797
Min_Res_NR2	1	3	2	1	2	1	1	609	669	669	720	720
Max_Res_NR2	1	1	1	2	1	2	1	615	679	688	752	752

Çizelge 7.15 : Bulanık ÇM-KKPÇ Probleminin Çözümü.

Faaliyet Sayısı : 7		Kaynak Türü : 3		Kaynak Miktarı (8,25,21)		Proje Kompleksliği		1,42857 Proje Risk İndeksi		0,5333			
Çizelge Maliyeti	CPM	Min Slack	ROT	Tabu(Min Slack)	Tabu(ROT)	Mod Seçim Kuralı							
(643,690,699,745)	(33 35 35 38)	(33 35 35 38)	(33 35 35 38)	(33 35 35 38)	(33 35 35 38)	Random	0 -> 1	1 -> 3	2 -> 3	3 -> 2	4 -> 3	5 -> 3	6 -> 1
(443,514,514,567)	(24 27 27 30)	(24 27 27 30)	(24 27 27 30)	(24 27 27 30)	(24 27 27 30)	MinDur	0 -> 1	1 -> 1	2 -> 1	3 -> 1	4 -> 3	5 -> 2	6 -> 1
(706,753,771,826)	(33 35 36 38)	(33 35 36 38)	(33 35 36 38)	(33 35 36 38)	(33 35 36 38)	MaxDur	0 -> 1	1 -> 3	2 -> 3	3 -> 2	4 -> 1	5 -> 3	6 -> 1
(637,688,697,739)	(27 29 29 31)	(27 29 29 31)	(27 29 29 31)	(27 29 29 31)	(27 29 29 31)	MinResUsageR1	0 -> 1	1 -> 2	2 -> 3	3 -> 2	4 -> 3	5 -> 2	6 -> 1
(624,683,692,742)	(33 35 36 38)	(33 35 36 38)	(31 35 36 39)	(33 35 36 38)	(33 35 36 38)	MaxResUsageR1	0 -> 1	1 -> 3	2 -> 2	3 -> 1	4 -> 5	5 -> 1	6 -> 1
(419,486,486,543)	(27 30 30 34)	(27 30 30 34)	(27 30 30 34)	(27 30 30 34)	(27 30 30 34)	MinResDem	0 -> 1	1 -> 1	2 -> 1	3 -> 1	4 -> 3	5 -> 3	6 -> 1
(734,787,796,849)	(33 35 36 38)	(37 40 40 43)	(40 43 44 47)	(37 40 40 43)	(40 43 44 47)	MaxResDem	0 -> 1	1 -> 3	2 -> 2	3 -> 2	4 -> 1	5 -> 2	6 -> 1
(505,575,584,645)	(27 30 31 34)	(27 30 31 34)	(27 30 31 34)	(27 30 31 34)	(27 30 31 34)	Mod 1	0 -> 1	1 -> 1	2 -> 1	3 -> 1	4 -> 1	5 -> 1	6 -> 1
(689,743,743,797)	(27 29 29 31)	(27 29 29 31)	(27 29 29 31)	(27 29 29 31)	(27 29 29 31)	Mod 2	0 -> 1	1 -> 2	2 -> 2	3 -> 2	4 -> 2	5 -> 2	6 -> 1
(534,588,597,641)	(33 35 35 38)	(33 35 35 38)	(33 35 35 38)	(33 35 35 38)	(33 35 35 38)	Mod 3	0 -> 1	1 -> 3	2 -> 3	3 -> 2	4 -> 3	5 -> 3	6 -> 1
(482,549,558,624)	(27 30 31 34)	(27 30 31 34)	(27 30 31 34)	(27 30 31 34)	(27 30 31 34)	MinResUsageN1	0 -> 1	1 -> 1	2 -> 1	3 -> 1	4 -> 1	5 -> 3	6 -> 1
(689,743,743,797)	(27 29 29 31)	(27 29 29 31)	(27 29 29 31)	(27 29 29 31)	(27 29 29 31)	MaxResUsageN1	0 -> 1	1 -> 2	2 -> 2	3 -> 2	4 -> 2	5 -> 2	6 -> 1
(609,669,669,720)	(31 33 33 35)	(29 33 33 36)	(29 33 33 36)	(31 33 33 35)	(29 33 33 36)	MinResUsageN2	0 -> 1	1 -> 3	2 -> 2	3 -> 1	4 -> 2	5 -> 1	6 -> 1
(615,679,688,752)	(27 30 31 34)	(27 30 31 34)	(29 31 32 34)	(27 30 31 34)	(27 30 31 34)	MaxResUsageN2	0 -> 1	1 -> 1	2 -> 1	3 -> 2	4 -> 1	5 -> 2	6 -> 1

8. UYGULAMA SONUÇLARI VE DEĞERLENDİRME

Bu bölümde tez çalışmasına konu olan deterministik (kesin)/bulanık faaliyet süreli ve genel ifadesiyle KKPC problemiyle ilgili olarak yapılan modelleme çalışmalarının sonuçları en kapsamlı bir biçimde ele alınmıştır. Çalışmada kullanılan örnek modellerin oluşturulması sayısal analizi ve geliştirilmesi ve hazırlanan programın literatürdeki veri kümelerinin üzerinde yapılan çalışmaların sonuçları sunulacak ve çeşitli değerlendirmelerde bulunulacaktır. Önceki bölümlerde geliştirilen örnek çalışmalar yardımıyla, literatürde sıklıkla kullanılan kaynak <http://129.187.106.231/psplib/> proje çizelgeleme veri kütüphanesinden elde edilen KKPC ve ÇM-KKPC problemlerinin hemen hemen hepsi ve daha başka kaynaklardan elde edilebilen problemler üzerinde çeşitli çözümler gerçekleştirilmiştir. Bu problem tiplerinin özellikleri ve parametreleri, aşağıda ayrıntılı olarak ifade edilmiştir. İlk önce öncelik kuralına dayalı arama algoritmalarıyla yapılan arama sonucunda KKPC problemleri ile ilgili başlangıç çözümler elde edilmiştir. Daha sonra aynı problem setleri geliştirilen TA algoritması ile çözülmüştür. Algoritmaya ilişkin sonuçlar kıyaslanarak, algoritmanın deterministik ve bulanık ortamda benzerlik ve farklılıkları ve literatürde elde edilebilen en iyi çözümlerle kıyaslanmıştır. Daha sonra ise bu çözüm değerlerine istatistiki testler yapılmıştır. Çözüm algoritmalarının temel özelliği olurlu çözüm bölgesinde kalınarak arama yapılmıştır. Uygulama sonuçları ve değerlendirmelerine geçilmeden önce literatürde yapılan kıyaslamaların etkinlikleri ve genel kıyaslama parametreleri hakkında bilgilerle konuya açıklık getirilmeye çalışılacaktır.

8.1 Veri Kümelerinin Kıyaslanmasında Uygulanan Yaklaşımlar

8.1.1 Etkinlik araştırması

Geliştirilen TA'ların etkin olup olmadığını belirlemek amacıyla literatürde bir takım örnek problemler ve yöntemler kullanılmaktadır. Konuyla ilgili literatür araştırmasında görüldüğü üzere, geliştirilen algoritmaların performansının ölçümü ve

değerlendirilmesinde, literatürde kullanılan yöntemler temel olarak şu başlıklar altında toplanabilmektedir:

1. Optimal sonucu bilinen bir problemle kıyaslamak. En iyi çözümü bilinen bir ya da birkaç problemle test edilmesi ve sonuçların karşılaştırılması.
2. Optimali bilinmeyen ancak en iyi sonucu bilinen problemlerle kıyaslamak.
3. Yapay yöntemlerle karmaşık bir şekilde oluşturulan problemler kümesidir.
4. Değişik parametreler ile test edilerek sonuçların karşılaştırılması.
5. Elde edilen çözüm için yapılan çabanın dikkate alınması.
6. Karmaşık problem kümesi üzerinde test edilerek sonuçların istatistiksel analize tabii tutulması.

Araştırmacılar geliştirdikleri algortimaların performansını, bu alternatiflerden birini kullanarak test etmektedirler. Öncelik kuralı yöntemleriyle sağlanan sonuçların değişkenliği, yürütülen araştırmaların etkin ve baskın sezgisel kuralların ortaya çıkartılması yönünde yoğunlaşmasına yol açmıştır. Yukarıda ifade edilen yöntemlerden özellikle (1)'den (5)'e nolu yöntemler, TA'nın proje çizelgeleme konusunda yapılan başlangıç çalışmalarında çok yoğun olarak kullanılmıştır. (6) nolu performans ölçüm yöntemi, bilgisayar teknolojisindeki gelişmeler nedeniyle, değişik bilgisayarlarda (işlemci) yapılan test sonuçlarını karşılaştırmayı imkânsız kılabilir. Çünkü bu çaba, genellikle, CPU-saniye (işlem süresi) olarak kullanılmaktaydı. Bunun yerine, belli iterasyon sonucunda elde edilen çözüm kalitesini karşılaştırmak, daha etkin sonuç vereceğinden daha yaygın kullanılmaktadır (Ulusoy, 2006).

Sezgisel yöntemlerde etkinlik araştırmaları iki şekilde yapılabilir:

1. Farklı ortam ve koşullarda yapılan gerçek projeler ele alınarak, yöntemler bunlar üzerinde denenir.
2. Projeler, laboratuvar koşullarında benzetim yoluyla oluşturularak, bulanık faaliyet süresine sahip problemler deneysel olarak oluşturulmuş olup, yöntemlerin etkinliği bu problemler üzerinde araştırılır.

Gerçek projeler üzerinde yapılan araştırmaların, doğal olarak diğerlerine göre daha sıhhatli sonuçlar vereceği düşünülebilir. Ancak, çok sayıda gerçek projeye ilişkin

verilerin toplanmasındaki güçlükler, şirketlerin bu konuda genellikle gizlilik ilkesi gütmesi v.b. gibi nedenlerle, bu çalışmaların yapılmasının, özel uygulamalar dışında mümkün olmadığını göstermektedir.

Kesin çözümleri sağlanabilir büyüklükteki problemler üzerinde, sezgisel yöntemlerle elde edilen çözümlerin optimal çözümlerle karşılaştırılması ve değerlendirilmesi gereklidir. Bu durumda, incelenen problemler küçük projelerle sınırlı kalmaktadır. Optimizasyon yöntemleriyle çözümlenemeyen büyüklükteki problemlerde farklı sezgisel yöntemlerin denenmesi ve sonuçlar karşılaştırılarak bunların göreceli etkinliklerinin belirlenmesi gerekmektedir. Bu seçeneklerden birincisinin uygun gerçekçi bir değerlendirme ortamı yaratacağı düşünülebilir. Ancak, sağlanabilen küçük çaplı problemlerde sezgisel yöntemlerin verdiği sonuçlardaki değişim de azalmakta, aralarındaki farklılığın ortaya çıkartılması güçleşmektedir. Bu nedenle, ikinci seçeneğin, etkin sezgisellerin göreceli karşılaştırmalar yapılarak belirlenmesi açısından daha gerçekçi ve sağlıklı bir yaklaşım sağlayacağı açıktır. Bu çalışmada geliştirilen sezgisel yöntemler üzerinde yürütülen etkinlik araştırması, yukarıda belirtilen ikinci seçeneğe dayandırılmıştır.

8.1.2 Kullanılan yapay KKPÇP'leri ve özellikleri

Sezgisel yöntemlerin denenmesi amacıyla bir dizi projenin benzetim yoluyla oluşturulmasında, bu projelerin hangi özellikleri taşıması gerektiği sorunu ortaya çıkmaktadır. Örneğin, KKPÇP / TP / ÇK gibi kodlarla belirtilen ana model özellikleri dışında, şebekenin yapısı, süreler, KYM ile bulunan zaman ve bolluk değerleri, kaynaklar v.b. gibi özellikler “proje karakteristikleri” adı altında toplanan bir dizi ölçütün tanımlanmasına yol açmıştır.

Proje karakteristiklerini belirleyen üç temel özellik şunlardır:

- 1- Proje şebekesinin yapısı ve boyutları,
- 2- Süreler, kritik yol analizi ile bulunan zaman ve bolluk değerleri,
- 3- Kaynak kapasitesi ve kaynak kullanımına ilişkin özellikler.

Çeşitli araştırmacılar tarafından geliştirilen ölçütlerin ayrıntılı birer dökümü, tek proje (TP) modelleri için Davis (1975), çok proje (ÇP) modelleri için Patterson

(1976) tarafından çıkartılmıştır. Bunlardan temel nitelik taşıyan birkaçı aşağıda açıklanmaktadır;

1- Şebeke yapısına ilişkin karakteristikler

- ◆ Topolojik kademelendirmenin yapılmış olduğu bir şebekede bir kademedeki en büyük düğüm sayısı,
- ◆ Bir yol üzerindeki ardışık düğümlerin en büyük sayısı,
- ◆ Proje şebeke karmaşıklığı: $C = \text{Serim Sayısı} / \text{düğüm sayısı}$.

2- Süre ve Zaman Karakteristikleri

- ◆ İşlemlerin toplam süresi
- ◆ Ortalama işlem süresi
- ◆ Serbest bollukların toplamı
- ◆ Kritik yol uzunluğu,
- ◆ Bolluk yoğunluğu ölçütü .

3- Kaynaklara ilişkin karakteristikler:

- ◆ Bir kaynak türüne yaratılan toplam iş hacmi W_k
- ◆ Bir kaynak türünün kısıtlılık parametresi (RF)

RF parametresi, KYM sonucu, k kaynak türü için en erken başlama zamanlarına bağlı olarak elde edilen yük diyagramında, kaynak talebinin RF kullanılabilir kaynak miktarını RF aşan en büyük değerinin iş hacmine oranını belirleyerek kaynağın ne ölçüde kısıtlı olduğunun global bir değerini verir. Yük dağılım parametresi; yük dağılım parametresinin değeri, KYM ile en erken (veya en geç) başlama zamanına göre oluşturulan yük diyagramının, zaman ekseninde kritik yol süresinin yarısı ile belirlenen orta noktası esas alınır, yük değerlerinin bu noktaya göre momentleri hesaplanarak bulunur. Bu şekilde, kapasite yüklenmesinin programın ilk yarısında mı, yoksa ikinci yarısında mı daha yoğun olduğu belirlenir.

Algoritmanın test edilmesinde seçilen problemlerin özellikleri; karmaşıklığı, büyüklüğü ve kullanılan kaynakların kıtlığı/bolluğu gibi faktörler, çoğu zaman KKPC problemlerini oldukça fazla etkileyen faktörlerdir. **Bazı geliştirilen algoritmalar, karmaşık olmayan problemlerde etkin sonuç verirken, karmaşık**

problemlerde en iyi sonucu verememektedir. Bu nedenle, deęişik problemlerin test edilmesinde, problemin özellikleri ve kullanılan parametreler dikkate alınarak yapılmaktadır.

8.1.3 Parametre Kıyaslama

8.1.3.1 Proje şebekesinin karmaşıklığı

Proje aęının büyüklüęü, faaliyet sayısına baęlıdır. Çoęu zamanda aęın karmaşıklığını doğurmaktadır. Aę karmaşıklığı arttıkça basit aęlarda etkin çözüm üretebilen algoritmalar, karmaşık aęlarda etkin/uygun çözüm verememektedirler (Hindi vd., 2002). $C = \text{Serim Sayısı} / \text{düęüm sayısı}$.

Proje kaynak kullanım durumuna göre; projede kullanılan kaynakların kapasitesi, sayısı ve tipleri gibi özellikler karmaşıklığı etkileyen önemli faktörlerdir. Kaynakların projeye etkisi iki ana başlık altında incelenmektedir.

8.1.3.2 Kaynak faktörü

Kaynak faktörü (RF); projedeki her bir kaynak için, faaliyet başına düşen ortalama kullanım oranını ifade etmektedir. $J = \text{Faaliyet sayısı}$ $K = \text{Projede kullanılan tüm kaynak türleri}$ $k=1 \dots K$ dır. $r_{jk} = j$ faaliyetinin, k tipi kaynaktan kullandığı miktar olmak üzere;

$$RF_k = \frac{1}{JK} \sum_{j=1}^J \sum_{k \in K} x \quad (8.1)$$

şeklinde formüle edilmektedir. Denk. 8.1.'de $r_{jk} > 0$ ise $x=1$, aksi halde $x=0$ olarak değerlendirilecektir. Böylece tüm kaynaklar için kaynak kullanım durumu belirlenebilmektedir. Eğer k . kaynak için, $RF=1$ ise, her faaliyet bu kaynağı kullanmaktadır. $RF=0$ ise, hiçbir faaliyet bu kaynağı kullanmamaktadır. Projedeki tüm kaynaklar için kaynak faktörü sıfır olduęu durumda; KKPÇP olarak verilen problemin esasında, kaynak kısıtı olmayan bir problem olduęu açığa çıkacaktır (Hindi vd., 2002; Kolisch vd.,1995). Kaynak kullanım faktörü, yenilenebilir ve yenilenemeyen kaynak türleri için geçerli olan bir ölçümdür ve ($k \in R, N$ dir). Yani

k kaynağı yenilenebilir (R) ve yenilemeyen (N) kaynak türündedir. Kaynak faktörü, [0,1] aralığında değerler alabilmektedir.

8.1.3.3 Kaynak gücü

Kaynak gücü (RS); faaliyetler için gereken kaynak miktarının, kullanılabilir kaynak ile ilişkisini belirlemek amacıyla kullanılmaktadır. Kaynakların bol ya da kıt olma durumunu belirlemede kullanılan RS; bir kaynağın kullanılabilir üst sınırının, bu kaynağın ortalama kullanım miktarına bölünmesi ile elde edilmektedir.

$$RS_k = \frac{R_k}{\frac{1}{J} \sum_{j=1}^J r_{jk}} \quad (8.2)$$

Denk (8.2)'de; .

R_k : k kaynağının kapasitesini

r_{jk} : j. faaliyetinin k kaynağından kullanım miktarını

J : proje faaliyetlerinin sayısını temsil etmektedir.

Böylece RS_k 'nın sıfıra (0) çok yakın olması, k. kaynağın kapasite açısından yetersiz kalacağını göstermektedir. Aksi durumda, projenin tamamlanması açısından k. kaynağın bir sıkıntı yaratmayacağı vurgulanmaktadır. RS_k 'nın bire (1) yakın olması, kaynak gücünün yüksek düzeyde olduğunu göstermektedir. Yüksek kaynak gücüne sahip bir kaynağın da, projeyi tamamlayacak yeterlikte olduğu söylenebilmektedir. Burada da ($k \in R, N$) dir. Yani k kaynağı yenilenebilir ve yenilemeyen kaynak türündedir.

Tüm kaynak türlerinde kaynak kıtlığını ya da bolluğunu ölçen;

$$R_k = R_k^{\min} + RS_k (R_k^{\max} - R_k^{\min}) \quad (8.3)$$

şeklindeki formülde, RS_k bir ölçüm parametresi olarak kullanılmaktadır. Burada; .

R_k : k kaynağının kapasitesini

$R_k^{\min}$: projede k kaynağına yapılan taleplerin en azını

$R_k^{\max}$: projede k kaynağına KYM ile yapılan taleplerin en fazlasını (peak yapılan kaynak değerinin belirlenmesi gerekmektedir) temsil etmektedir ve ($k \in N$)’dir. Bu değerin hesaplanmasında hesaplama güçlüğü söz konusudur.

Denk 8.3.’den elde edilen RS_k ;

$$RS_k = \frac{R_k - R_k^{\min}}{R_k^{\max} - R_k^{\min}} \quad (8.4)$$

Denk 8.4. şeklindedir. Denk 8.4.’den elde edilen $RS_k=0$ değeri, kullanılan k. kaynak miktarının, mümkün olan en düşük seviyesinde olduğunu göstermektedir. Yine aynı formülden, $RS_k=1$ elde edilirse, kullanılan k. kaynak miktarının, projeyi tamamlayabilmek için gereken en üst kaynak talebini karşılayabilecek seviyede olduğu anlamını taşımaktadır. Böylece kaynak kıtlığı olmadığı söylenebilmektedir (Kolisch vd. 1995).

Yüksek kaynak faktörü ve/ya da düşük kaynak gücü; çizelgelenen problemin “zor” problem olduğunu gösterirken, düşük kaynak faktörü ve/ya da yüksek kaynak gücü; çizelgelenen problemin “zor olmayan” bir problem olduğunu göstermektedir (Kolish, 1996). Bu çalışmada kullanılan problemler; ağ büyüklüğü, kaynak faktörü ve kaynak gücü açılarından karmaşık ve zor olduğu bilinen problemlerdir; bilimsel literatürde de yaygın olarak kullanılmaktadır. R Kolisch, A. Sprecher ve A. Drexler, Patterson’un örnek problemlerini yeterince dikkatli hazırlanmadığı düşüncesini taşımaktadırlar. Üretilen problemlerin; proje ağının karmaşıklığı (network complexity), kaynak faktörü (resource factor) ve kaynak gücü (resource strength) adıyla anılan üç faktörle kontrol edilmesi gerektiğini savunmuşlardır.

8.1.4 En iyi çözümü bilinen problemler ile test edilmesi

Geliştirilen algoritma, optimal çözümü bilinen bir problem ile test edilerek, algoritmanın sonucu ile en iyi bilinen çözüm karşılaştırılmaktadır. Yöntemin kullanılmasında amaç, geliştirilen algoritmanın elle çözülen problemlere ve çözümlerine bir yakınsama yapıp yapmadığını irdelemek, geliştirilen TA’nın bir çözüm alternatifi olup olmadığını araştırmaktır. Bu nedenle TA ile problem çözme

alışmalarının bařlangı dönemlerinde, arařtırmacıların ok sık kullandığı bir yöntem olmuřtur. Ancak günümüzde TA'larla yapılan alışmalarda amaçlar geliřerek deėiřmiř, kapsamlı ve ortak niteliklere sahip problemlerin özümünde bir yöntem olarak kullanılması söz konusu edilmeye bařlamıřtır. Optimum özümü bilinen karmařık problemlerin özümünü üreten tek ve kesin bir algoritma henüz mevcut olmadığından, yöntemin karmařık problemlerde uygulanabilmesine yönelik alışmalar devam etmektedir. Dolayısıyla bu tarz problemleri temel alan algoritma, sadece test konusu edilen basit problemlerde etkin sonuçlar üretebilmekte ve özümü bilinmeyen karmařık izelgeleme problemleri için kesin özüm alternatifi olamamaktadır.

Geliřtirilen algoritmanın bařlangı testlerinde, özümü bilinen problemler üzerinde test edilmesi halen kullanılan bir yöntemdir. Ancak algoritmayı deėerlendirmek aısından yeterli bir yöntem deėildir. ünkü yapılan alışmalar; basit ve bilinen problemlerde optimum sonuç veren algoritmaların, karmařık problemlerde optimum özümünden oldukça uzak deėerler verebildiėini göstermektedir.

Literatür incelendiėinde, geliřtirilen algoritmaların özümü bilinen örneklerle test edildiėine sıka rastlanmaktadır. **Ancak, alışmamız kapsamında da görülmüřtür ki, bir problemde en iyi özüm bulan bir algoritma, diėer problemlerde en iyi özümü bulamayabilmektedir.** Bu nedenle, özümü bilinen ya da rahatlıkla özülebilen problemler algoritmanın performansının bařarılı bir řekilde deėerlendirilmesinde yeterli olamayacaktır. Sonuç olarak, algoritmanın testi ařamasında; öncelikle bilinen problemlerle test edilmesi, algoritmanın bařarılı bir řekilde izlenmesi aısından bir gereklilik olmaktadır. Ancak yeterli olduėu söylenememektedir. Dolayısıyla TA'ların performans deėerlendirilmesinde tek bařına kullanılması yanıltıcı sonuçlar verebilecektir.

8.1.5 İstatistiksel Analizlerin Kullanılması

İstatistiksel analizlerin kullanıldığı yöntemde algoritma, uygulamaya konu edilen problemler üzerinde test edilerek elde edilen sonuç ya da elde edilen en iyi özümler, istatistiksel analizlere tabi tutulmaktadır. İstatistiksel analizlerin sonuçları doėrultusunda algoritmanın performansı deėerlendirilmeye alışılmaktadır. Bu amaçla elde edilen sonuçlar Minitab 16 paket programı aracılığıyla istatistiksel

analizler gerçekleştirilmiştir. İstatistiksel analizlerde, aritmetik ortalamalardan başlayarak değişik analizler yapılabilmektedir. Literatürde yaygın olarak kullanılan yöntemlerden biri de “en iyi çözümden ortalama sapma” ve bunun “standart sapması” yöntemidir. Yani en iyi proje tamamlama süresinden ne kadar sapma gösterdiği bilgisi elde edilir. Bu yöntem, en iyi çözümün bilindiği problemlerde, algoritmanın performansını değerlendirmede kullanılan etkin bir yöntemdir. En iyi çözümün bilinmediği durumlarda, problemin alt sınırı ya da üst sınırı ile karşılaştırma yapılmaktadır (Ulusoy, 2006). Algoritma performansını etkin bir şekilde ölçebilmek için, kullanılan alt/üst sınırların çözüme mümkün olduğunca yakın olması beklenmektedir. Ancak eldeki olanaklar, test için kullanılan problemlerin alt sınırı olarak, yine problemlerde verilen alt sınırları kullanmaya yöneliktir. Bu tez çalışması kapsamında geliştirilen algoritmaya ilave edilen yordam ile bu alt sınırların, birçok problemde, problemin kritik yol metodu ile çözüm sonucu olduğu görülmektedir. Algoritmanın sadece bir probleme bağlı kalınmadan ve daha gerçekçi bir şekilde performansının değerlendirilmesi amacıyla, istatistiksel analizler çok yoğun kullanılmaktadır. Özellikle TA’ların, karmaşık optimizasyon problemlerinin çözümünde bir yöntem olarak kullanılması fikrinin yaygınlaşması sonucunda, performans ölçümünde daha fazla önem kazanmıştır. Çalışmamızda da bu yöntem kullanılarak, literatürde yayınlanan diğer yöntemler ile geliştirilen TA karşılaştırılmaktadır.

8.2 Uygulama Sonuçları ve Değerlendirme

Önceki bölümlerde sayısal örnek projeler üzerinde sınanan TA kullanımı gösterilmiştir. Geliştirilen algoritma yaklaşımı esas alınarak belirtilen uygulamalar, Microsoft Visual C# bilgisayar dili kullanılarak modüler yapıda, çeşitli alt programlardan oluşan bir program sistemi geliştirilmiştir.

KKPÇ problemlerinin benzetimle oluşturulması için EK-C’de belirtilen akış diyagramındaki çalışmalar yapılmıştır: Özetle;

- Proje karakteristiklerinin tespit edilmesi ve problemlerin gruplandırılması.
- Problemlerin önceki bölümlerde anlatılan TDP, bulanık küme teorisi, başlangıç çözümün bulunması için önerilen öncelik kuralı tabanlı algoritma

ve TA algoritmalarının geliştirilmesi, bilgisayar programlarının hazırlanması, denenmesi ve problemlerin oluşturulması.

- Çeşitli problemler üzerinde yapılan uygulamalar olarak özetlenebilir.
- Özet çalışma adım adım uygulamaya konulmuş olup EK-A'ya uygun olarak aşağıdaki gibi tasarlanmıştır.
 - İlk önce KKPC problemlerinin temel yapı taşını oluşturan KYM ve BKYM uygulamaları kodlanmıştır. (Adım 1 ve 2)
 - KYM kaynak kısıtlarını dikkate almadığı için kaynak kısıtları göz önünde bulunduran ve kaynak ataması yapılabilmesine olanak tanıyan ve TA optimizasyon yöntemi için başlangıç çizelgeler oluşturabilecek Minslack ve Rot öncelik tabanlı algortimaların hem deterministik hemde bulanık sayılar için çözümüne yönelik uygulama kodlanmıştır. (Adım 3-4-5-6)
 - Adım 7'de Bulanık KKPC problemlerinin paralel KA ile çözümü yapılmış ancak uygulamaya yönelik kodlama çalışması yapılmamıştır.
 - Adım 8 ve 9'da deterministik ve bulanık KKPC problemleri için TA algortimasına yönelik uygulama yapılmıştır.
 - Adım 10 ve 11'de ise deterministik ve bulanık ÇM-KKPC problemlerine yönelik uygulama yapılmıştır.

Çizelgeleme problemlerinde, öncül/ardıl olarak farklı iki şekilde verilen faaliyet öncelikleri göz önünde bulundurularak, algoritmanın farklı problem tipleri için çalışması sağlanmaktadır. Böylece, problem ne şekilde verilirse verilsin, kullanıcı tarafından herhangi bir dönüşüm işlemi yapılmaksızın, algortmada test edilebilecektir. Tez çalışması kapsamında kullanılan problemlere ait veri yapıları ve örnek problemler EK-C'de sunulmuştur. Ancak bazı test veri kümeleri için zaman serisi diyagramı kullanarak ulaşılan sonuçlara EK-(Ç-E) arasında sunulmuştur. Problemlerin her biri, farklı iterasyonlarda olmak üzere test edilebilir, varsayılan olarak **problem büyüklükleri temel alınmış olup** ele alınan problemin faaliyet sayısının iki katı TA için maksimum iterasyon sayısı olarak belirlenmiş ve problemler çözülmüştür. Bunun yanı sıra iterasyon sayının etkisi araştırılması

bakımından herbir problem belirlenen sayıda (100, 1000, 5000 iterasyon ve istenilen iterasyon kadar) çalıştırılabilir.

Algortimanın performansını ölçmek ve değerlendirmek amacıyla; her bir veri kümesinin ortalama optimalden sapması hesaplanarak, literatürde yayınlanmış olan diğer çalışmalarla karşılaştırılmaktadır. Test edilen problemler neticesinde; elde edilen proje bitirme sürelerinin tatmin edilebilir düzeyde olduğu gözlenmiştir.

Geliştirilen algoritma, basit problemlerde evrensel (global) sonuçlar verirken, karmaşık problemlerde genellikle yerel sonuçlar üretebilmektedir. Bu çalışma kapsamında kullanılan ve literatürde yer alan veri kümelerine ait bilgiler Çizelge 8.1’de verilmiştir. Bahse konu veri kümelerine ait özellikler ve parametrelerde gösterilmiştir. Veri kümelerinde ait; veri kümesi adı, problem tipi, faaliyet sayıları, kaynak sayıları, kaynak tipleri ve toplam test problemlerinin sayısından oluşmaktadır.

Çizelge 8.1 : Literatürdeki Bazı Veri Kümeleri

KKPÇ Problemleri		ÇM-KKPÇ Problemleri			
Dosya Adı	Dosya Sayısı	Dosya Adı	Dosya Sayısı	Dosya Adı	Dosya Sayısı
Patterson 110	110	c15.mm	551	n1.mm	637
J30.RCP	478	c21.mm	551	n3.mm	601
J60.RCP	480	J10.mm	536	r1.mm	553
J90.RCP	480	J12.mm	548	r3.mm	557
J120.RCP	600	J14.mm	551	r4.mm	553
MV.RCP	1800	J16.mm	550	r5.mm	546
RG300.RCP	480	J18.mm	551	Jall.mm	3240
set 1	900	J20.mm	554	boctor50.mm	120
set 2	182	J30.mm	640	boctor100.mm	120
set 3	242	m1.mm	640	mm_j10.mm	273
set 4	242	m2.mm	481	MMLIB50.mm	540
set 5	242	m4.mm	555	MMLIB100.mm	540
we.RCP	1441	m5.mm	558	n0.mm	470

Bu veri kümeleri üzerinde geliştirilen algoritma test edilmiş ve sonuçları ilerleyen bölümlerde detaylarıyla açıklanmaya çalışılacaktır. Algoritma; bulanıklaştırıcı kullanılarak deterministik problemler bulanık yapıya dönüştürülerekde çözelebileceği gibi Rangen 1-2 proje üretici program kullanılarak oluşturulan tüm problem veri kümeleri üzerinde oluşturulan text tipi dosyaların başında yer alan boşluklar silinerek formatının düzenlenmesiyle de çalıştırılabilir. Dolayısıyla

özöllecek veya test edilecek mevcut problem sayıları ok fazla olup hepsi iin problemlerin özölmlerini bu alıřma kapsamında sunabilmek mölmlölne deęildir.

Gerekleřtirilen bu alıřmada, literatölrde proje izelgeleme problemlerinde test amalı olarak pek ok örnekte kullanıldıęı görölne ve PSPLIB (<ftp://ftp.bwl.unikel.de/pub/operations-research/psplib>)’de yer alan yapay problemler kullanılmaktadır. Ancak deneysel amalı olarak, en iyi sonucu bilinen problemlerin test edilmesi de alıřmamızda mölmlölne kılınmaktadır.

Algoritma Performansının Deęerlendirilmesi; Geliřtirilen algoritmayı literatölrde yer alan daha önce yapılmıř alıřmalarla saęlıklı bir řekilde kıyaslayabilmek amacıyla, izelgede gösterilen problemler üzerinde test edilmektedir. Böllece, toplam 27 farklı problem tipi ve toplamda yaklařık 27000 adet izelgeleme ve bunların bulanıklařtırılmasıyla problemler test edilerek elde edilen bilgiler deęerlendirilme kapsamına alınmaktadır. Her bir veri kümesindeki tüm problemler iin en uygun özölme arayıřı, mevcut haliyle problemin faaliyet sayısının iki katı kadar olarak belirlenmiř ancak ihtiya duyulması halinde belirlenen maksimum iterasyon sayısı kullanılarak test edilmiřtir. Böllece, geliřtirilen algoritmanın literatölrde yer alan aynı iterasyon sayıları kullanılarak test edilen dięer TA sonuçları ile kıyaslanması mölmlölne olabilmektedir. Farklı iterasyon sayıları ile test etmenin bir dięer amacı da, iterasyon sayısı olarak belirlenen TA parametresinin, geliřtirilen algoritma ile elde edilen özölmler üzerindeki etkilerini izleyebilmektir. Geliřtirilen algoritmamızdan elde edilen sonuçlar ise bir sonraki kısımda özetlenecektir. Veri kümelerindeki KKP problemlerin test sonuçları, izelge 8.2’de sunulmuřtur.

Literatölrde TA’ların performansını test etmek amacıyla, özel olarak geliřtirilmiř bilgisayar programları ile üretilen farklı KKP’ler bulunmaktadır. Örnekler dört adet yenilenebilir kaynak kısıtına sahiptir (Kolisch vd.,1995). Örnek olarak faaliyet sayısına göre; patterson (110), j30 (480 örnekli), j60 (480 örnekli), j90 (480 örnekli), j120 (600 örnekli) ve MV (1800) olarak gruplanan veri kümelerinden örnek KKP’ler hazırlayarak literatölre sunmuřlardır. (Kolisch vd. 1995). PSPLIB’deki yapay problemler, j30, j60, j90 ve j120 olarak sınıflandırılmaktadır ve literatölrde “özölme zor” olarak bilinen problemlerdir (Ulusoy, 2006). Problemlerin özellięi; projede faaliyet sayısının fazlalıęı, faaliyetler arasında karmařık iliřkilerin varlıęı ve farklı kaynak türlerinin birok faaliyet tarafından kullanılmasıdır.

Çizelge 8.2 : KKPC Veri Kümeleri Test Sonuçları.

Dosya Adı	Proje Kompleksliği	İterasyon Sayısı	Problem Sayısı	Tabu-MinSlack		Tabu-ROT		Kaynak Tipi (Yenilenebilir: R)	Kaynak Sayısı	Veri Yapısı Uygun mu?
				Optimal Bulunan Sonuçlar	Optimalden Yüzde Fark	Optimal Bulunan Sonuçlar	Optimalden Yüzde Fark			
Patterson		1000	110	58	0,043242	58	0,043242			
J30.RCP	1,5	1000	158		0,06776			R	4	U
	1,8125		160		0,07601			R	4	U
	2,125		160		0,05871			R	4	U
Toplam	Toplam		478	353	0,018206				4	
J60.RCP	1,5	1000	160		0,05404			R	4	U
	1,80645		160		0,05355			R	4	U
	2,1129		160		0,051			R	4	U
Toplam			480	257	0,052862					
J90.RCP	1,5	100	160		0,0662			R	4	U
	1,80435		160		0,076			R	4	U
	2,1087		160		0,0767			R	4	U
Toplam			480	243	0,072956					
J120.RCP	1,5	250	200		0,1383		0,15279	R	4	U
	1,80328		200		0,14813		0,14281	R	4	U
	2,10656		200		0,15746		0,14195	R	4	U
			600		0,14796		0,14585		4	U
MV10.RCP		1000	360	316	0,01414	326	0,00927	R	4	U
MV20.RCP			359	226	0,05217	248	0,04553	R	4	U
MV30.RCP			361	152	0,0557	168	0,04035	R	4	U
MV40.RCP			360	109	0,0772	115	0,05672	R	4	U
MV50.RCP			360	100	0,07259	91	0,05629	R	4	U
Toplam			1800	998	0,05447	1084	0,04163			

Belirtilen özellikler problemi oldukça karmaşık hale getirmektedir. Dolayısıyla geliştirilen algoritmanın, PSPLIB'deki problemler üzerinde test edilmesi, algoritmanın güvenilirliğini artırmakta, literatürde geliştirilen TA'larda test problemleri olarak yoğun kullanıldığı görülmektedir. J30 problemleri; optimum çözümü bilenen 30 faaliyetli problemlerden oluşmaktadır ve toplam 480 adet farklı problem mevcuttur. Her birinde yine 480 adet farklı problem olan j60 ve j90 problemlerinin optimum çözümü halen bilinmeyen, ancak araştırmacılar tarafından üzerinde çalışılan problemlerdir. J120'de, 120 faaliyetli 600 adet problem içermektedir. Literatürde yapılan araştırmalar sonucunda, geliştirilen yeni TA'lar ile elde edilen, "üst sınır" ve "alt sınır" olarak belirlenen sonuçlar yine aynı ortamda yayınlanmaktadır. Böylece, proje çizelgeleme ile çalışanlara, geliştirdikleri algoritmaların performansını karşılaştırmalı olarak değerlendirme imkânı tanınmaktadır.

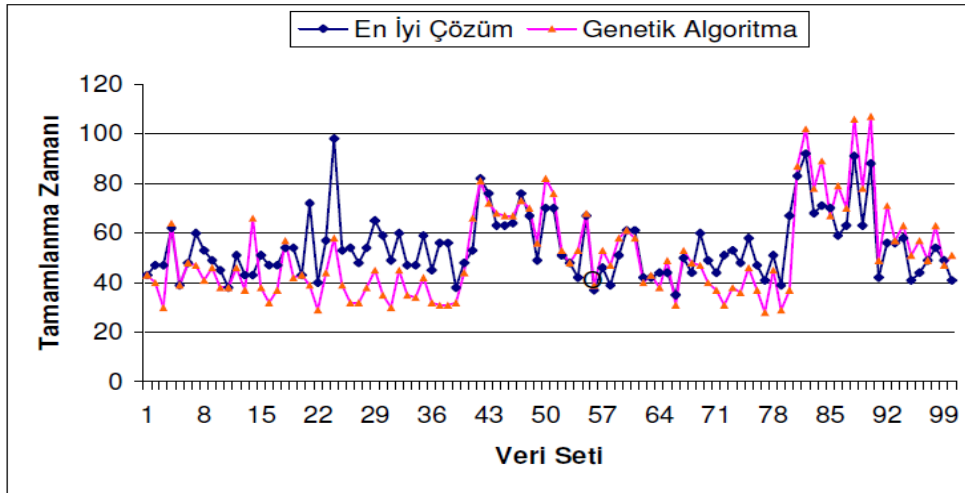
8.3 Problem Büyüklüklerine Göre Sonuçların Değerlendirilmesi

Bu kısımda literatüre konuya ilişkin yapılan çalışmalardan elde edilen sonuçlar referans kaynaklar belirtilmek koşuluyla bu kısımda özet tablolar olarak değerlendirmenin uygun olacağı kanaatindeyiz.

8.4 KKPÇ Problemlerinin Kıyaslanması

8.4.1 J30 KKPÇP veri kümesi

Çizelge 8.2’de J30 veri kümesindeki problemlerin sonuçları kullanılarak, geliştirilen algoritmanın optimal çözümden sapmaları kaynaklar belirtilmek koşuluyla analiz edilebilmektedir. Veri kümesindeki her bir proje çizelgeleme probleminin optimal çözüm değerleri ve geliştirilen algoritma ile 1000 iterasyon sonucunda elde edilen değerler zaman serileri şeklinde EK-Ç’de gösterilmektedir. Aşağıda yer alan çözüm değerleri 30 faaliyetli veri setlerinin genetik algoritma ile çözümü sonucunda elde edilmiştir. Bu çözüm değerlerine bakıldığında genetik algoritma ile yapılan çözümler literatürde bilinen en iyi çözümlerden daha iyi çözüm değerlerine ulaşabildiği ifade edilmektedir. Şekil 8.1 ve Çizelge 8.3’de J30 Veri kümesi için 30 faaliyetten oluşan veri setinin 1000 iterasyonlu Genetik Algoritma (GA) çözüm değerleri ile en iyi çözüm değerlerinin kıyaslanması gösterilmiştir (T.Sarı 2008).



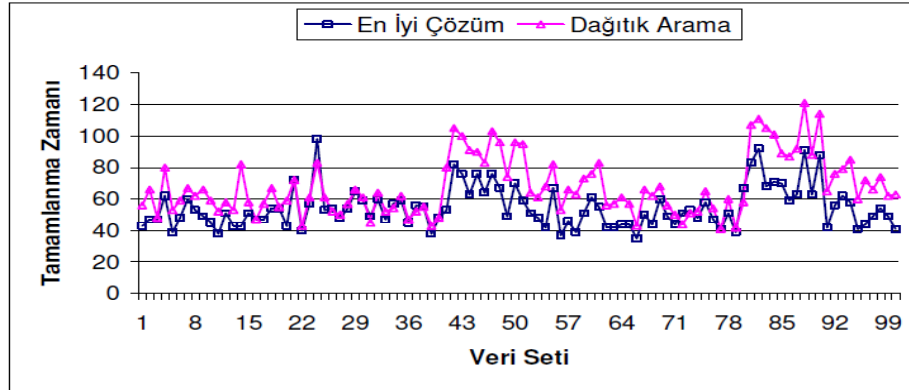
Şekil 8.1: J30 Veri Kümesi için 1000 iterasyonlu GA Yöntemi ile Elde Edilen Sonuçlar.

Çizelge 8.3 : J30 Veri Kümesi için Sezgisel Yöntemlerin Performansları.

Yıl	Araştırmacılar	Çizelge Yöntemi	Sezgisel yöntemler	Optimum çözümden ortalama sapma %	
				500	1000
2003	P.Tormos ve A.Lova	P/S	Pişmanlığa dayalı örnekleme (çok geçişli)	0.32	0.23
2001	P.Tormos ve A.Lova	P/S	Pişmanlığa dayalı örnekleme	0.35	0.25
2000	S.Hartmann	P/S	GA	-	0.37
1998	K.Bouleimen ve H.Lecocq	S	Benzetim tavlama-	-	0.38
1998	S.Hartmann	S	Faaliyet listesi	-	0.54
1998	A.Schirmer	P/S	Adapte edilmiş	-	0.65
1998	T.Baar, P.Brucker ve S.Knust	-	Tabu arama	-	0.86
1998	S.Hartmann	S	Rastgele anahtar	-	1.03
1998	S.Hartmann	S	Öncelik kuralı	-	1.38
1996	R.Kolisch ve A.Drexl	P/S	Adapte edilmiş	-	0.74
1996	R.Kolisch	S	Min LFT	-	0.83
1996	R.Kolisch	P	Min LFT	-	1.40
1996	R.Kolisch	P	WCS	-	1.40
1995	R.Kolisch	S	Rastgele	-	1.44
1995	R.Kolisch	P	Rastgele	-	1.77
1995	V.Leon ve B.Ramamoorty	-	GA	-	2.08
2007	Geliştirilen algoritma	S	Rastgele anahtar ve öncelik kuralı	25.87	25.36

Kaynak : Tormos ve Lova,2003,s.1082

Benzer şekilde 30 faaliyetten oluşan 100 adet veri seti dağıtık arama algoritması ile çözüldüğünde aşağıdaki sonuçlar elde edilmiştir. Dağıtık Algoritmasıyla (DA) en iyi çözüm değerlerine yakın değerler elde edildiği ifade edilmiştir. Şekil 8.2’de J30 Veri kümesi için 30 faaliyetten oluşan veri setinin 1000 iterasyonlu çözüm DA değerleri ile en iyi çözüm değerlerinin kıyaslanması gösterilmiştir (T.Sarı 2008).

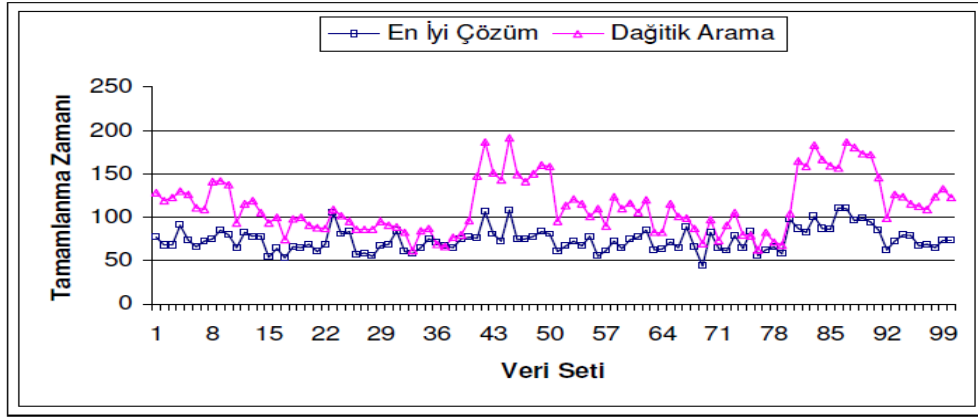


Şekil 8.2 : J30.rcp için 1000 iterasyonlu DA Yöntemi ile Elde Edilen Sonuçlar.

8.4.2 J60 KKPÇP veri kümesi

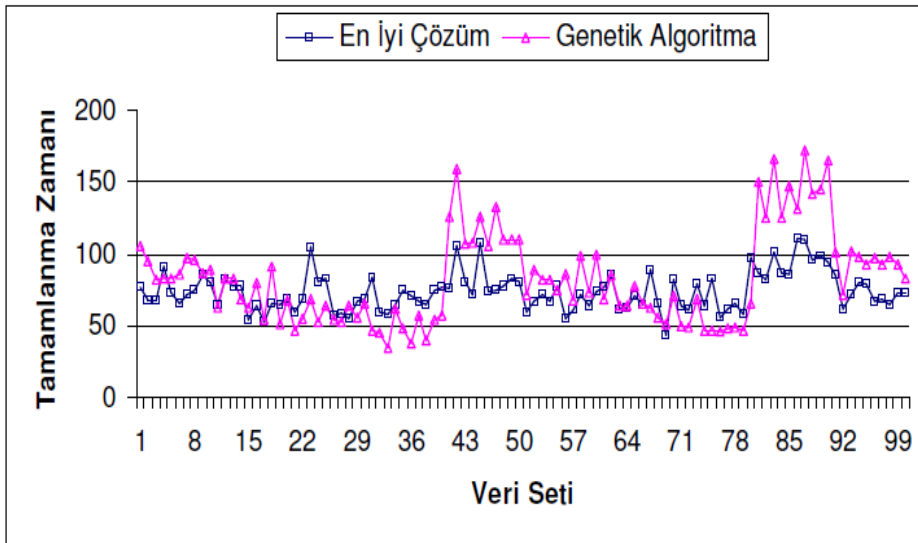
Aynı algoritmaların daha büyük veri setleri ile çalışıldığında ne şekilde çözüm elde edeceğine ilişkin değerlendirme yapabilmek için 60 faaliyetten oluşan veri setinin 1000 iterasyonlu genetik algoritma çözüm değerleri ile veri setinin alt sınır değerleri

kıyaslaması ve 1000 iterasyonlu DA ile çözüm değerleri Şekil 8.3'deki grafiklerde yapılmıştır.



Şekil 8.3 : J60.rcp için 1000 iterasyonlu DA Yöntemi ile Elde Edilen Sonuçlar.

Bu kıyaslamaya göre genetik algoritma ile yapılan çözümler alt sınır değerlerinden daha iyi sonuçlara ulaşabildiği ifade edilmiştir. Şekil 8.4'de J60 Veri kümesi için 60 faaliyetten oluşan veri setinin 1000 iterasyonlu Genetik Algoritma (GA) çözüm değerleri ile en iyi çözüm değerlerinin kıyaslanması gösterilmiştir. Ancak DA ile elde edilen çözüm değerleri GA ile elde edilen çözüm değerleri alt sınır değerlerine yaklaşık çözümler bulmakla birlikte GA ile elde edilen çözüm değerlerinden genel olarak daha iyi olmadığı ifade edilmiştir. (T.Sarı 2008).



Şekil 8.4 : J60.rcp için 1000 iterasyonlu GA Yöntemi ile Elde Edilen Sonuçlar.

J60.rcp sonuçların değerlendirildiğinde ; J60 veri kümesinin özelliği, 60 faaliyetli ve 4 kaynak kullanan çizelgeleme problemlerini içermesidir. İnternet’te kullanıma sunulan ve PSPLIB’den alınan bu problemlerin optimum çözümleri hala mevcut olmamakta ve araştırma konusu olmaya devam etmektedir. Araştırmacılar tarafından geliştirilmiş sezgisel algoritmalar ile elde edilen, proje tamamlama sürelerinin alt ve üst sınırları mevcuttur. Bu alt ve üst sınırlar, şimdiye kadar yapılan çalışmalardan elde edilen sonuçlardan oluşturulmuş sınırlardır. Ancak yine de her problemin alt ve üst sınırları belirlenmemiştir. Belirlenmeyen sınırlar için kritik yol metoduyla elde edilen proje tamamlama süresi kullanılmaktadır. J60 kümesindeki problemler test amaçlı kullanıldığında, geliştirilen algoritmanın konu edilen bu sınır değerlerinden sapmaları analiz edilebilmektedir.

Çizelge 8.4’da J60 Veri kümesi için 60 faaliyetten oluşan veri setinin 1000 iterasyonlu çözüm değerleri ile en iyi çözüm değerlerinin kıyaslanması gösterilmiştir.

Çizelge 8.4 : J60.rcp için Sezgisel Yöntemlerin Performansları.

Yıl	Araştırmacılar	Çizelge Yöntemi	Sezgisel yöntemler	Üst sınırdan ortalama sapmalar		Kritik yol temelli alt sınırdan ortalama sapmalar	
				500	1000	500	1000
2003	P.Tormos ve A.Lova	P/S	Pişmanlığa dayalı örnekleme-çok geçişli	0.59	0.50	12.16	12.04
2001	P.Tormos ve A.Lova	P/S	Pişmanlığa dayalı örnekleme	0.67	0.55	12.27	12.11
2000	S.Hartmann	P/S	GA	-	0.71	-	12.26
1998	K.Bouleimen ve H.Lecocq	S	Benzetim tavlama-	-	1.05	-	12.75
1998	S.Hartmann	S	Faaliyet listesi	-	0.88	-	12.68
1998	A.Schirmer	P/S	Adapte edilmiş	-	1.09	-	12.94
1998	T.Baar, P.Brucker ve S.Knust	-	Tabu arama	-	1.68	-	13.80
1998	S.Hartmann	S	Rastgele anahtar	-	2.36	-	14.68
1998	S.Hartmann	S	Öncelik kuralı	-	1.32	-	13.30
1996	R.Kolisch ve A.Drexl	P/S	Adapte edilmiş	-	1.48	-	13.51
1996	R.Kolisch	S	Min LFT	-	1.76	-	13.96
1996	R.Kolisch	P	Min LFT	-	1.76	-	13.59
1996	R.Kolisch	P	WCS	-	1.76	-	13.66
1995	R.Kolisch	P	Rastgele	-	2.71	-	14.89
1995	R.Kolisch	S	Rastgele	-	3.34	-	15.94
1995	V.Leon ve B.Ramamoorthy	-	GA	-	2.36	-	14.33
2007	Geliştirilen algoritma	S	Rastgele anahtar ve öncelik kuralı	36.99	36.53	39.48	38.98

Kaynak : Tormos ve Lova,2003,s.1082

8.4.3 J90 KKPÇP veri kümesi

J90 veri kümesinin sonuçları değerlendirildiğinde; 90 faaliyetli ve yenilenebilen 4 kaynak kullanan çizelgeleme problemlerini içermesidir. PSPLIB’den alınan bu problemlerin optimum çözümleri hala mevcut bulunmamaktadır. Optimum

çözümleri yerine, araştırmacıların geliştirdikleri algoritmalarla elde edilen en iyi proje tamamlama sürelerinin alt ve üst sınırları bulunmaktadır. Geliştirilen algoritmanın bu veri kümesinden elde edilen test sonuçları ve diğer veri kümesinden 1000 iterasyon ile elde edilenlerin en iyi sonuçları, Çizelge 8.5’de literatürde en iyi sonuç elde edilen algoritmalarla karşılaştırılarak gösterilmektedir.

Çizelge 8.5 : J90 Veri Kümesi için Sezgisel Yöntemlerin Performansları.

Araştırmacılar	J30		J60		J90	
	Sapma%	Rank	Sapma%	Rank	Sapma%	Rank
S.Hartmann (1998)	0.25	5	11.89	4	-	-
Alcaraz ve Maroto (2001)	0.12	2	11.86	3	-	-
S.Hartmann (2002)	0.22	3	11.70	2	-	-
Bouleimen ve Lecocq (2003)	0.23	4	11.90	5	-	-
Debels vd. (2003)	0.23	1	11.35	1	10.93	1
Çalışmada geliştirilen algoritma	25.36		36.53		41.35	

Kaynak : Debels vd.,2003,s.29

8.4.4 J120 KKPÇP veri kümesi

J120.rcp sonuçları değerlendirildiğinde; J120 veri kümesinin özelliği, 120 faaliyetli ve yenilenebilen 4 kaynak kullanan çizelgeleme problemlerini içermesidir. İnternet’te kullanıma sunulan ve PSPLIB’den alınan bu problemlerin optimum çözümleri hala mevcut bulunmamaktadır. Optimum çözümleri yerine, araştırmacıların geliştirdikleri algoritmalarla elde edilen en iyi proje tamamlama sürelerinin alt ve üst sınırları bulunmaktadır.

Çizelge 8.6 : J120 Veri Kümesi için Sezgisel Yöntemlerin Performansları.

Yıl	Araştırmacılar	Çizelge Yöntemi	Sezgisel yöntemler	Ust sınırdan ortalama sapmalar			Kritik yol temelli alt sınırdan ortalama sapmalar	
				500	1000	Rank	500	1000
2003	P.Tormos ve A.Lova	P/S	Pişmanlığa dayalı örnekleme-çok geçişli	0.45	0.23	1	36.34	35.98
2001	P.Tormos ve A.Lova	P/S	Pişmanlığa dayalı örnekleme	0.66	0.44	2	36.65	36.32
2000	S.Hartmann	P/S	GA	-	1.29	3	-	37.17
1998	S.Hartmann	S	Faaliyet listesi	-	2.36	4	-	39.37
1998	K.Bouleimen ve H.Lecocq	S	Benzetim tavlama-activity list	-	5.50	12	-	42.81
1998	S.Hartmann	S	Öncelik kuralı	-	2.58	5	-	39.93
1998	S.Hartmann	S	Rastgele anahtar	-	6.85	14	-	45.82
1998	A.Schirmer	P/S	Adapte edilmiş	-	2.87	8	-	39.93
1996	R.Kolisch	P	Min LFT	-	2.69	6	-	39.60
1996	R.Kolisch	P	WCS	-	2.71	7	-	39.65
1996	R.Kolisch ve A.Drexl	P/S	Adapte edilmiş	-	3.71	9	-	41.37
1996	R.Kolisch	S	Min LFT	-	4.55	10	-	42.84
1995	Leon ve Ramamoorthy	P	GA	-	5.09	11	-	42.91
1995	R.Kolisch	P	Rastgele anahtar	-	6.21	13	-	44.46
1995	R.Kolisch	S	Rastgele anahtar	-	9.39	15	-	49.25
2007	Geliştirilen algoritma	S	Rastgele anahtar ve öncelik kuralı	48.37	47.65		54.84	54.10

Kaynak : Tormos ve Lova,2003,s.1083

J120 veri kümesindeki problemlerin optimum çözümleri de hala araştırma konusu olmaya devam etmektedir. Çizelge 8.6'da literatürde en iyi bilinen sonuçlar ile elde edilen algoritmalarla karşılaştırılarak gösterilmektedir.

8.4.5 Patterson 110 KKPÇP veri kümesi

Bununla birlikte Depuy ve Whitehouse (2000) yaptıkları çalışmada 110 Patterson problemleri için elde edilen sonuçlara göre TA yaklaşımı, COMSOAL sonuçlarının ve aynı problem grubu için diğer sezgisel yaklaşımların sonuçları aşağıdaki Çizelge 8.6'da gösterilmiştir. Patterson veri kümesi; 7 ila 50 faaliyetten oluşan 4 adedi en az bir, 3 adedi iki ve 103 adedi en çok üç tip kaynak kullanan test problemlerinden oluşmaktadır. Biri, J.H. Patterson verileri diğeri de PSPLIB (Project Scheduling Problems Library) verileridir. Patterson verilerinde, her bir problem, 7–51 arasında faaliyet sayısına ve üç adet yenilenebilir kaynak kısıtına sahiptir. Veriler üç kümeye ayrılmıştır. Her bir veri kümesi 110 örnek problem içermektedir. İlk veri kümesi temel alınarak, ikinci 107 veri kümesi kaynak ihtiyacı açısından birincinin iki katı, üçüncü küme ise kaynak ihtiyacı açısından birincinin üç katı olacak şekilde hazırlanmıştır (Hindi vd. 2002). Depuy ve vd (2000) tarafından elde edilen sonuçlar Çizelge 8.7'de gösterilmiştir.

Çizelge 8.7 : 110 Patterson Problemleri Optimal Değerlerin diğer Sezgisel Yaklaşımlarıyla Bulunan Sonuçların Karşılaştırılması.

Sezgisel Yaklaşımlar	Optimal Sonuca Ulaşmış Denemelerin %'si	Optimal değerden ortalama Sapma %'si	Optimal değerden maksimum Sapma Farkı %'si
ACTIM	30,90	5	23,7
LFT	26,40	6,1	23,3
ROT	10,00	11,4	36,4
ACTRES	27,30	5,2	25
ACTIM+LFT	38,20	3,6	19,5
ACTIM+LFT+ROT	40,90	3,1	15,8
ACTIM+LFT+ROT+ACTRES	41,80	2,8	13,9
COMSOAL 20 İterasyon	42,18	2,34	11,3
COMSOAL 50 İterasyon	52,18	1,87	11,3
Kaynak : Depuy ve ark. 2000			

Thomas ve Salhi (1998)'nin yaptıkları çalışmada başlangıç çözümü olarak MinSlack, MINLFT, WRUP, WACRU, PSTSM gibi öncelik esaslı kurallar en iyi çözüm olarak alınarak TA prosedürü FORTRAN 77 bilgisayar programında kodlanmıştır. Bu

program aracılığıyla 110 Patterson problemleri üzerinde test edilmiştir. Bulunan sonuçlar Çizelge 8.8’de gösterilmiştir.

Çizelge 8.8 : Patterson 110 Problemleri için Bulunan Sonuçlar.

Kurallar		MINSLACK	MINLFT	WRUP	WACRU	PSTSM
Optimalin üzerinde elde edilen ortalama Sapma		5,53	6,41	6,32	5,24	2,3
Standart sapma (%)	$\pm \square$	5,66	5,89	5,4	4,7	2,77
Optimal bulunan sonuçların sayısı		28	21	22	26	51
Ortalama Bilgisayar Zamanı (s)		Patersson = 218,675				
Kaynak : Thomas ve Salhi (1998)						

Bu analiz sonuçlarından yola çıkarak, Thomas ve Salhi (1998)’nin yaptıkları çalışmada MinSlack kuralı ile bulunan sonuçlar dikkate değerdir. Bu çalışmada MinSlack kuralı ile elde edilen optimalden ortalama sapma yüzdesi % 11,6 TA prosedürüyle sağlanan değer % 2,97 olduğu görülmektedir. Buna karşın Thomas ve Salhi (1998)’nin TA prosedürüyle ortalama 218,675 saniyede elde ettiği PSTSM değeri % 2,3 ve bulunan optimal sonuç sayısı 51’dir. Bu çalışma sonuçlarıyla karşılaştırıldığında en iyi değer olarak 383 ms (~3,8 saniyede) 67 problemde optimal sonuç, % 1,38 optimalden ortalama sapma ile gerçekleştirilmiştir.

8.5 ÇM-KKPC Problemlerinin Kıyaslanması

PSPLib’de yer ÇM-KKPC veri kümeleri üzerinde yapılan incelemelerde problemler faaliyet sayılarına, yenilenebilir ve yenilenemeyen kaynak sayılarına, mod sayılarına göre gruplandırılmıştır. Damak ve vd. (2009)’a göre literatürdeki bu problemlerin optimal sonuçlarının 20 faaliyete kadar bilindiği, 30 faaliyetli veri kümelerinin optimal sonuçlarının bilinmediği ifade edilmektedir. Bizim çalışmamızda Damak ve vd. (2009) yaptıkları kıyaslamayla karşılaştırdık ve veri kümeleri üzerindeki çözümlerimi ayrıca ilave ettik. Bizim çalışmamızda mod seçim kurallarının işlevselliği test edilmiş olup, MinDur ve MinResDem mod seçim kurallarının ÇM-KKPC problemleri için uygun çözümler olduğu optimal ve optmale yakın en iyi sonuçların verdiği tespit edilmiştir. Diğer kurallarla elde edilen sonuçlara bakıldığında problemin KYM zamanın optimal sonuçların çok üzerinde değerler alması gibi bir durumla karşılaşmıştır ki, maksimum faaliyet sürelerine sahip faaliyetlerin KYM zamanında mod seçimine göre seçilen faaliyetlerin sürelerindeki değişimden kaynaklanmaktadır. Bu da çözüm yönteminin doğru olarak hazırlandığını

açıklık getirmektedir. Önceki bölümde Çizelge 8.1’de içerikleri ifade edilen ÇM-KKPC problemleri incelendiğinde; J10, J12, J14, J16, J18, J20, J30.mm, boctor50, boctor 100, c15, c21 MMLib50, MMLib100 veri kümeleri yanlarında ifade edilen sayı kadar faaliyet sayına; r1, r3, r4, r5 yenilenebilir kaynak sayısına, n0, n1, n3 yenilenemeyen kaynak sayısına, m1, m2, m4, m5 mod sayısına bağlı olarak, jall.mm veri kümesi çeşitli sayıda faaliyet sayılarına bağlı olarak düzenlenmiştir. Örneğin j102–2.mm örnek problemi; j10 olarak anılan 10 faaliyetli örnek problemlerin listesi olan EK-F’de, parametre sütununda 1 ve örnekler sütununda 2 değerini içeren satırdaki problemi temsil etmektedir. Benzer şekilde j102–2 örnek problemi, aynı listede parametre sütunu 2 ve örnekler sütununda 2 değerini içeren satırdaki problemi temsil etmektedir. Çizelge 8.9’da J10, J12, J14, J16, J18, J20.mm veri kümeleri üzerinde yapılan test sonuçlarımız ile Damak ve vd. (2009)’un çalışmasında yapılan çözümlerle kıyaslanması gösterilmiştir.

Çizelge 8.9 : J10, 12, 14, 16, 18, 20.mm için Algortimanın Performansı.

Kıyaslama Problemi	Pb. Sayısı	Tabu Arama		Benzetim Tavlama		Kuş Sürüsü Optimizasyonu		Evrin Algoritması	
		Optimal		Optimal		Optimal		Optimal	
		Ortalamadan bulunların Sapma	Ortalamadan bulunların Yüzdesi	Ortalamadan bulunların Sapma	Ortalamadan bulunların Yüzdesi	Ortalamadan bulunların Sapma	Ortalamadan bulunların Yüzdesi	Ortalamadan bulunların Sapma	Ortalamadan bulunların Yüzdesi
10	536	0,046891	95,3	0.21	96.30	0.06	99.25	0.09	99.30
12	547	0,034031	96,6	0.19	91.20	0.14	98.47	0.29	99.30
14	551	0,034199	96,58	0.92	82.60	0.44	91.11	0.63	97.60
16	550	0,033299	96,67	1.43	72.80	0.59	85.91	0.83	96.38
18	552	0,032611	96,74	1.85	69.40	0.99	79.89	1.58	94.43
20	554	0,033367	96,66	2.10	66.90	1.21	74.19	1.91	92.75

Kaynak : Damak ve vd. (2009)

Bu sonuçlar değerlendirildiği geliştirilen TA’nın benzetim tavlama, kuş sürüsü optimizasyonu ve evrensel algoritma sonuçlarına göre daha etkin olduğu, geliştirilen algoritmanın ÇM-KKPC problemlerinin çözümünde başarıyla uygulanabileceği değerlendirilmektedir.

Sadece ortalama değerlere bakarak geliştirilen TA yönteminin literatürde yapılan diğer çalışmalardan üstün olduğunu söyleyebilmek mümkün değildir. Bu sebeple, bu kısımda literatürde bilinen en iyi çözüm değeri bulunan SA, PSO ve DE yöntemleri ile geliştirilen TA arasındaki ilişkiyi ortaya koyabilmek amacıyla varyans analizi yapılmasına ihtiyaç duyulmuştur. Varyans analizi için Çizelge 8.9’daki veri kendi içinde 4 gruba ayrılmıştır. Birinci grup veri geliştirilen TA’ına ait değerleri ifade etmektedir. İkinci grup veri benzetim tavlama (SA) sonucu bulunan değerleri, üçüncü grup veri, Kuş Sürüsü Optimizasyonu (PSO) sonucu bulunan değerleri ve

dördüncü grup veri, Evrim Algoritması (DE) sonucu bulunan değerleri, J10,12,14,16,18 ve 20 faaliyetten oluşan ÇM-KKPÇP veri setlerinin çözümlerini ifade etmektedir. Buna göre, geliştirilen TA'sı en iyi çözüm değerleri grup ortalaması 0,03573, benzetim tavlama (SA) grup ortalaması 1,117, Kuş Sürüsü Optimizasyonu (PSO) grup ortalaması 0,572 ve Evrim Algoritması (DE) grup ortalaması 0,888 olarak bulunmuştur. En iyi çözüm ortalama sapma değerlerine göre geliştirilen TA sonuçlarının diğer yöntemlere göre daha iyi olduğu Çizelge 8.10'da görülmektedir.

Çizelge 8.10 : Varyans Analizi Tanımlayıcı İstatistik

N	Ortalama	Std.Sapma	Varyans	Standart Varyans için			
				95% Güvenlik Aralığı			
				Alt Sınır	Üst Sınır	Minimum	Maksimum
TA	6	0,03573	0,0055	0,00003	(0,000012; 0,000182)	0,03261	0,04689
SA	6	1,117	0,815	0,664	(0,259; 3,997)	0,19	2,1
PSO	6	0,572	0,458	0,21	(0,082; 1,261)	0,06	1,21
DE	6	0,888	0,719	0,518	(0,202; 3,113)	0,09	1,91
Toplam	24	0,653	0,689	0,589	(0,362; 0,944)	0,033	2,1

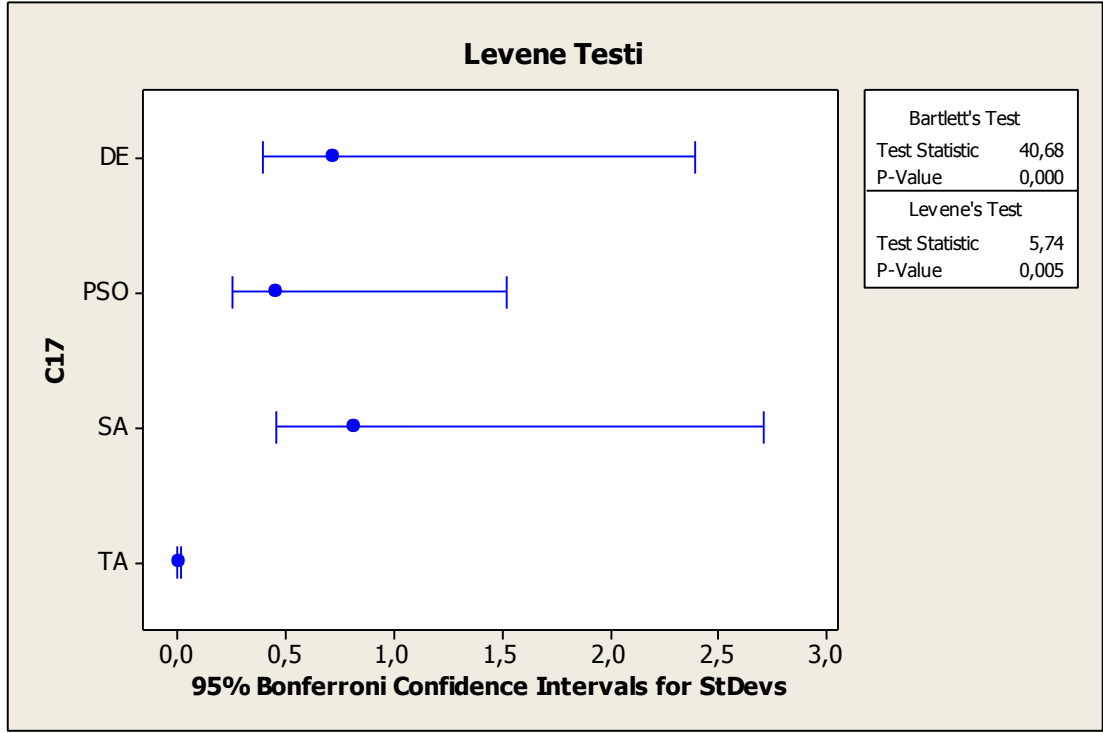
Varyansların homojen olup olmadığını anlamak için Levene testi yapılmıştır*. Levene testinde anlamlılık $0,000 < 0,05$ olduğundan grupların varyansları homojen değildir. Levene testi farklı örnek kitlelerin varyanslarının eşit olup olmadığını değerlendiren bir test olup, varyans analizinde kullanılır. Eğer Levene testi anlamlılık düzeyi $< 0,05$ ise sonraki analizler varyansların eşitsizliğine dayanarak yapılır. Levene testine ait sonuçlar Çizelge 8.11 ve Şekil 8.5'de gösterilmiştir.

Çizelge 8.11 : Varyansların Homojenliği

	N	Alt Sınır	Standart Sapma	Üst Sınır
DE	6	0,399452	0,7194	2,39059
PSO	6	0,254243	0,457883	1,52156
SA	6	0,452617	0,815148	2,70877
TA	6	0,003052	0,005496	0,01826

Levene Testi

Test İstatistiği :	5,74	p-değeri :	0,005



Şekil 8.5 : Levene Testi Sonuçları

$H_0: \mu_1 = \mu_2 = \mu_3$

H_1 : En az iki μ birbirinden farklıdır.

Çizelge 'deki varyans analizi tablosuna bakılırsa buradan gruplar arası anlamlı bir fark olduğu görülmektedir. $0,000 < 0,05$ 'tir. Dolayısıyla, gruplar arası karşılaştırmada en az bir grup diğerinden farklıdır.

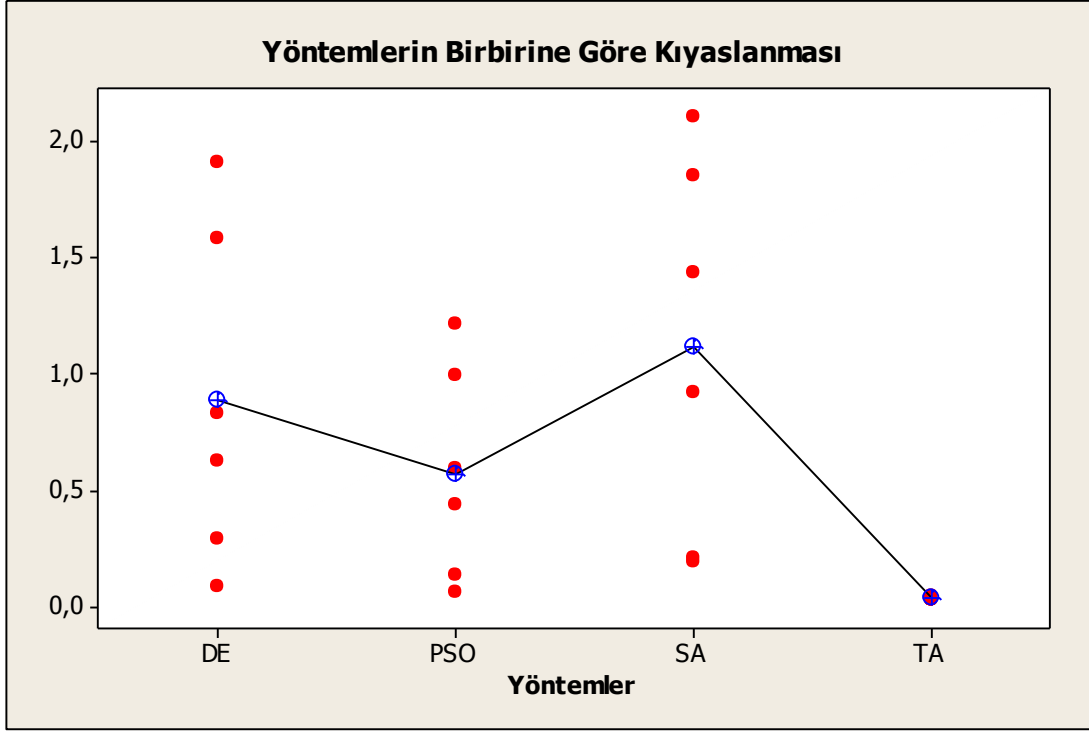
Çizelge 8.12 : ANOVA Analizi Sonuçları

	DF	SS	MS	F	p-değeri
Gruplar arası	3	3,948	1,316	3,78	0,027
Grup İçi	20	6,958	0,348		
Toplam	23	10,906			

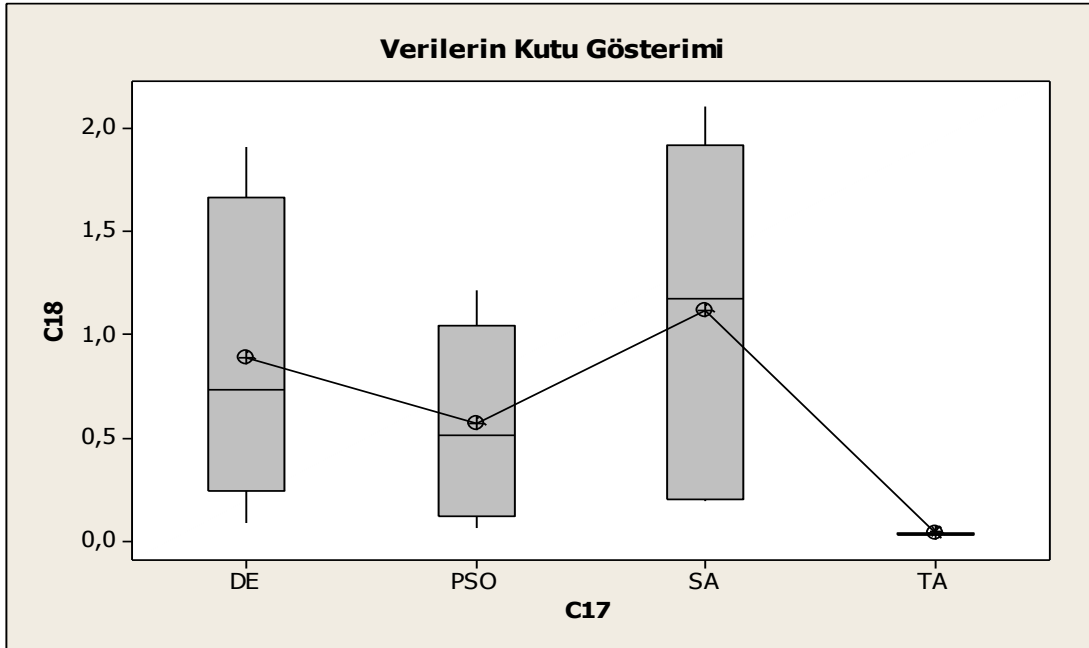
$S = 0,5898$ $R-Sq = 36,20\%$ $R-Sq(adj) = 26,63\%$

Ancak varyans analizinin önşartlarından biri olan varyansların homojenliği kabulü burada $p < 0,05$ ile geçerli olduğu Çizelge den elde edilmiştir. Dolayısıyla ANOVA testi ile aynı grup veri için geçerli olduğundan gruplar arası (yöntemler arasında) fark vardır. Hangi grup ya da grupların birbirinden farklı olduğuna çoklu karşılaştırma testleriyle karar verilecektir. Ortalamaların grafiğine bakıldığında en iyi çözüm değerleri olarak bilinen çözümlerin ortalamasının geliştirilen TA için en düşük ortalama sahip olduğu, PSO ve DE yönteminin ortalama değerlerinin en iyi

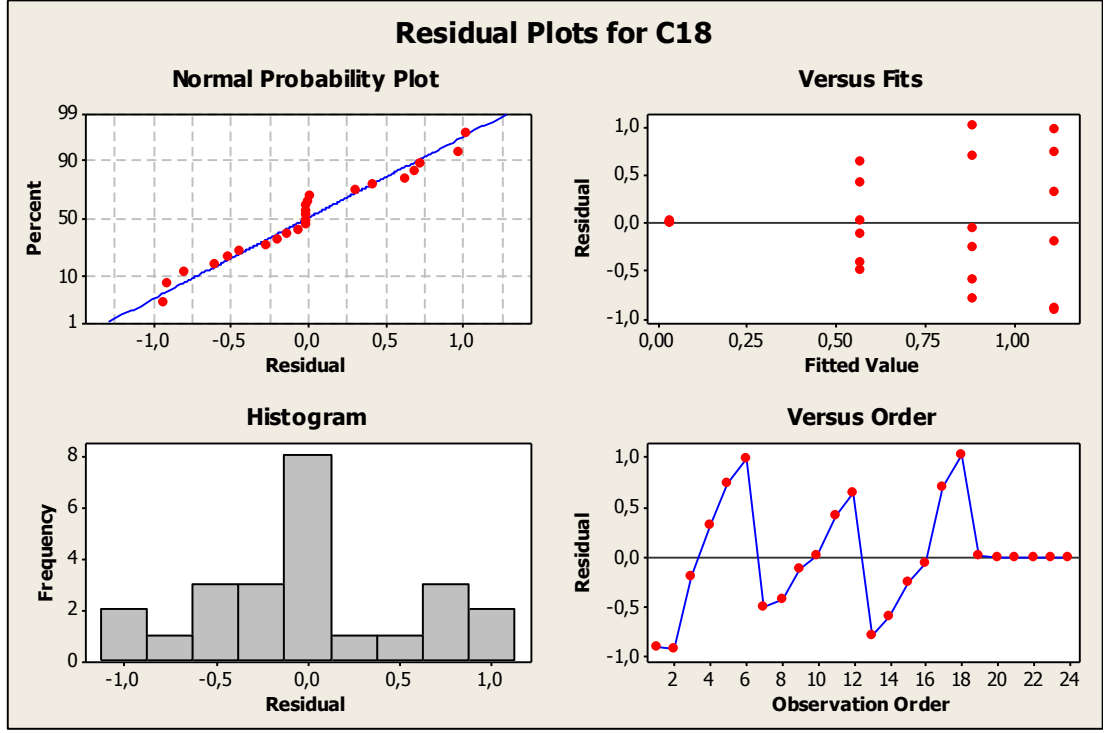
çözümlere yakın çözümler bulduğu, son olarak da SA yöntemiyle bulunan çözüm ortalaması diğerlerine göre en yüksek değere sahip olduğu ortaya çıkmıştır.



Şekil 8.6 : ANOVA Analizi Sonucu Yöntemler Arasındaki Fark



Şekil 8.7 : ANOVA Analizi Sonucu Yöntemler Arasındaki Farkın Kutu Gösterimi



Şekil 8.8 : ANOVA Analizi Grafik Sonuçlar

Her bir yöntem için ANOVA testi sonuçları incelendiğinde ortalamalar arasında anlamlı bir fark olduğu ve bu grupların ortalamalarının birbirinden anlamlı düzeyde farklı olduğu belirlenmiştir. Bu durum açıklanmak istenirse metasezgisel yöntemler her çalıştırmada farklı iyi çözümler bulabilen yöntemlerdir. Dolayısıyla grupların birbirinden farklı olması doğal bir durumdur. Bu farklılığın bir diğer sebebi ise çözüm algoritmalarının farklı çalışmasından kaynaklanmaktadır. Algoritmalar kendi içinde yerel arama prosedürleri de kullanabilmektedir. Bu durum ise çözüm değerlerinde farklılaşmaya yol açmaktadır. Her yöntem incelendiğinde ortalamalar bakımından ve en iyi çözüm değerleri bakımında birbirine yakın olduğu gözlenmiştir. Bu benzerlik anlamlılık değerlerinde de görülebilmektedir.

8.6 Uygulama

Bölüm 4, 5, 6 ve 7’de belirtilen algoritma yaklaşımı esas alınarak belirtilen uygulamalar, Microsoft Visual C# bilgisayar dili kullanılarak modüler yapıda, çeşitli alt programlardan oluşan bir program sistemi geliştirilmiştir. Test projesi için tabu araması tarafından sağlanan ortalama çözümler bölüm 8.4 ve 8.5’de özet olarak gösterilmiştir olması nedeniyle bu bölümde sonuçlar Çizelge 8.13, Çizelge 8.14, Çizelge 8.15’de değerlendirilecektir.

Çizelge 8.13 : J10.mm için Algoritmanın Performansı

Mod Seçim	Sufficient	Insufficient	M-O	R-O	TM-O	TR-O
Kuralı	Pb's	Pb's	Avg	Avg	Avg	Avg
Random	71	465	0,83482	0,15495	0,62616	0,092009
MinDur	185	351	0,09081	0,09518	0,049849	0,046891
MaxDur	459	77	1,1583	1,1659	0,89533	0,89297
MinResR3	297	239	0,85998	0,87105	0,5675	0,57401
MinResR4	303	233	0,83149	0,83844	0,56759	0,56313
MaxResR1	208	328	6,2067	0,50521	0,25421	0,24767
MaxResR2	207	329	0,3849	0,38909	0,24954	0,24129
Mode1	185	351	0,09081	0,09585	0,048954	0,047112
Mode2	302	234	0,75189	0,7643	0,48886	0,4971
Mode3	453	83	1,1575	1,1573	0,88943	0,88434
MinResDem	417	119				
MaxResDem	186	350				

J10.mm

ROT ve MinSlack tarafından sağlanan başlangıç uygulanabilir çözümleriyle birlikte, 100, 1000 veya istenilen sayıda iterasyonlu TA algoritmasının çalıştırılması sonucu optimal ve optimale çok yakın sonuçlar elde edilmiştir. Çözümler çok kısa bir ortalama hesaplama zamanında ve nihai proje süreleri optimale yakındır.

Çizelge 8.14 : J12.mm için Algoritmanın Performansı

Mod Seçim	Sufficient	Insufficient	M-O	R-O	TM-O	TR-O
Kuralı	Pb's	Pb's	Avg	Avg	Avg	Avg
Random	71	476				
MinDur	195	352	0,07661	0,08134	0,035446	0,034031
MaxDur	464	83	1,1337	1,1365	0,87366	0,87616
MinResR1	350	197	0,85825	0,86404	0,57634	0,57189
Mode1	252	295	0,76213	0,81158	0,54707	0,59816
Mode2	254	293	0,68914	0,81712	0,51084	0,59265
Mode3	195	352	0,07173	0,08134	0,036512	0,034321

J12

Bulunan çalışma sonuçları, iyi bilinen birkaç öncelik kuralları esasına göre oluşturulan ROT ve MinSlack bu yaklaşımların elde edilen sonuçlarla karşılaştırıldığında, başlangıç çözüm olarak seçilen yaklaşımların doğru çözümler verdiği ve algoritmanın doğru bir şekilde çalıştığı görülmektedir. Bu başlangıç çözümleriyle elde edilen sonuçlarla TA prosedürü işletilerek iyileştirilmiş ve önerilen

yaklaşımlarla bulunan çözümlerin daha iyi sonuçlar verdiği bulunmuştur. TA prosedürünü daha az iterasyonla daha iyi sonuçlar verdiği ifade edilebilir.

Çizelge 8.15 : J14.mm için Algoritmanın Performansı

Mod Seçim Kuralı	Sufficient Pb's	Insufficient Pb's	M-O Avg	R-O Avg	TM-O Avg	TR-O Avg
MinDur	216	336	0,08924	0,09584	0,042471	0,034199
MaxDur	468	84	1,2005	1,2057	0,92323	0,92748

J14

C15

18 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 551'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir. Proje kompleksliği 1,5 olarak tanımlanmıştır.

C21

18 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 552'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir. Proje kompleksliği 2,111111 olarak tanımlanmıştır.

M1

18 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 640'i çözülebilmektedir. Faaliyetler 1 modla icra edilmektedir. Proje kompleksliği 1,8333 olarak tanımlanmıştır.

M2

18 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 640'i çözülebilmektedir. Faaliyetler 2 modla icra edilmektedir. Proje kompleksliği 1,8333 olarak tanımlanmıştır.

M4

18 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 555'i çözülebilmektedir. Faaliyetler 4 modla icra edilmektedir Proje kompleksliği 1,8333 olarak tanımlanmıştır.

M5

18 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 558'i çözülebilmektedir. Faaliyetler 5 modla icra edilmektedir Proje kompleksliği 1,8333 olarak tanımlanmıştır.

N0

12-22 Faaliyetten oluşan 2 yenilenebilir ve 0 yenilenemeyen olmak üzere toplam 2 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 558'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,8333 << 1,85714285714286 olarak tanımlanmıştır.

N1

18 Faaliyetten oluşan 2 yenilenebilir ve 1 yenilenemeyen olmak üzere toplam 3 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 637'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,8333 olarak tanımlanmıştır.

N3

18 Faaliyetten oluşan 2 yenilenebilir ve 3 yenilenemeyen olmak üzere toplam 5 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 600'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,8333 olarak tanımlanmıştır.

R1

18 Faaliyetten oluşan 1 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 3 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 551'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,8333 olarak tanımlanmıştır.

R3

18 Faaliyetten oluşan 3 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 5 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 557'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,8333 olarak tanımlanmıştır.

R4

18 Faaliyetten oluşan 4 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 6 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 552'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,8333 olarak tanımlanmıştır.

R5

18 Faaliyetten oluşan 5 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 7 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 546'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,8333 olarak tanımlanmıştır.

J10

10 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 536'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,5 olarak tanımlanmıştır.

J12

12 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 547'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,85714285714286 olarak tanımlanmıştır.

J14

14 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemten oluşmaktadır. 640 problemin 551'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,8125 olarak tanımlanmıştır.

J16

16 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemden oluşmaktadır. 640 problemin 550'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,8333333333333333 olarak tanımlanmıştır.

J18

18 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemden oluşmaktadır. 640 problemin 552'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,8 olarak tanımlanmıştır.

J20

20 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemden oluşmaktadır. 640 problemin 554'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,81818181818182 olarak tanımlanmıştır.

J30

30 Faaliyetten oluşan 2 yenilenebilir ve 2 yenilenemeyen olmak üzere toplam 4 farklı kaynak tipinden oluşan 640 problemden oluşmaktadır. 640 problemin 552'i çözülebilmektedir. Faaliyetler 3 modla icra edilmektedir Proje kompleksliği 1,8125 olarak tanımlanmıştır.

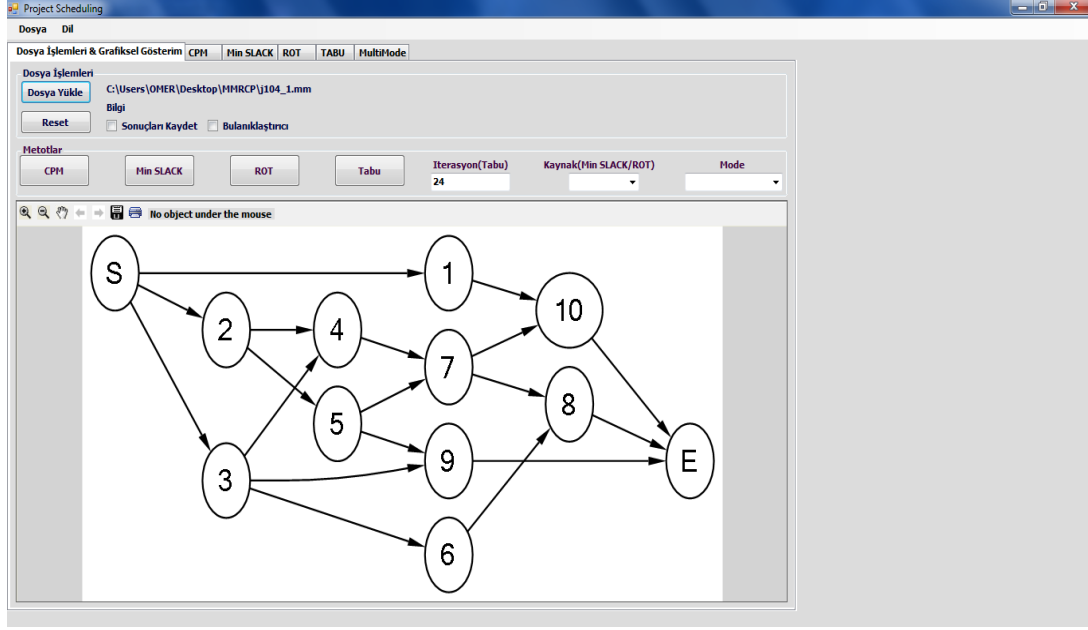
Çalışmada yenilenemeyen kaynakların proje süresince elde edilebilir miktarı kadar var olduğu varsayımıyla PSPLIB'den elde edilen yukarıdaki ÇM-KKPC problemlerinin deterministik çözümleri geliştirilen algoritma ile performansı denemiş ve sonuçları Çizelge 8.16'da gösterilmiştir. Pb ile problem ifade edilmek istenmiş ve veri kümesinin adınına atıfta bulunulmuştur. Mod seçim kuralı, mm problemlerinin hangi mod seçim kuralı işletilerek sonuçların bulunduğu, sufficeint Pb's ile yenilenemeyen kaynak için proje süresince elde edilebilir maksimum kaynak miktarı yeterli kaç problem bulunduğu, M-O Avg ile Minslack başlangıç çözümüyle optimal veya bilinen en iyi çözümden sapmanın ortamasını, R-O Avg ile Rot başlangıç çözümüyle optimal veya bilinen en iyi çözümden sapmanın ortamasını, TM-O Avg ile Minslack başlangıç çözümü ve tabu arama yöntemiyle bulunan sonucun optimal

veya bilinen en iyi çözümden sapmanın ortalamasını, TR-O Avg ile Rot başlangıç çözümü ve tabu arama yöntemiyle bulunan sonucun optimal veya bilinen en iyi çözümden sapmanın ortalaması ifade edilmektedir.

Çizelge 8.16 : PSPLIB mm için Algortimanın Performansı

Pb	Mod Seçim Kuralı	Sufficient Pb's	M-O Avg	R-O Avg	TM-O Avg	TR-O Avg
C15	MinDur	139	0,37299	0,3851	0,22733	0,22289
C15	MinResDem	182	0,2747	0,29568	0,1151	0,11856
C21	MinDur	131	0,38906	0,40343	0,25624	0,23329
C21	MinResDem	145	0,29218	0,31847	0,14945	0,14046
M1	MinDur	24	0,52619	0,52849	0,37493	0,33298
M1	MinResDem	24	0,52619	0,52849	0,37302	0,33561
M2	MinDur	69	0,40161	0,42249	0,24553	0,24926
M2	MinResDem	106	0,26909	0,28967	0,11726	0,12061
M4	MinDur	140	0,29771	0,30946	0,17375	0,15482
M4	MinResDem	153	0,25239	0,27442	0,12201	0,11472
M5	MinDur	141	0,25072	0,26539	0,13397	0,1236
M5	MinResDem	154	0,27342	0,29478	0,11753	0,11687
N0	MinDur	292	0,33459	0,34578	0,1973	0,20545
N0	MinResDem	292	0,34735	0,36845	0,15748	0,17266
N1	MinDur	281	0,30123	0,3115	0,17145	0,16342
N1	MinResDem	296	0,31991	0,34507	0,14719	0,14983
N3	MinDur	104	0,38	0,38809	0,23498	0,21238
N3	MinResDem	124	0,30211	0,32765	0,13223	0,1505
R1	MinDur	297	0,33063	0,34345	0,19866	0,18729
R1	MinResDem	286	0,26636	0,28285	0,1036	0,10188
R3	MinDur	105	0,39855	0,40893	0,24573	0,2207
R3	MinResDem	132	0,31275	0,33093	0,13869	0,1444
R4	MinDur	108	0,401	0,41651	0,26915	0,24601
R4	MinResDem	137	0,30983	0,32577	0,16231	0,14695
R5	MinDur	113	0,35916	0,37687	0,23501	0,2124
R5	MinResDem	135	0,2792	0,29947	0,12612	0,12193

Hazırlanan paket programın ana arayüz görünümü Şekil 8.9'da gösterilmiş olup, programın kullanıma ilişkin bilgiler EK-G'de programın alt bileşenlerine ait görünümü EK-Ğ'de gösterilmiştir.



Şekil 8.9 : Ana Arayüz Görünümü.

8.7 Yapılan Çalışmalarda Sağlanan Sonuçlar

Yapılan çalışmalarda şu soruların yanıtları aranmıştır.

- ◆ Çeşitli çizelgeleme yöntemlerinden hangisi daha üstündür?
- ◆ Hangi öncelik kuralı diğerlerine göre daha üstündür?
- ◆ Proje karakteristikleri ile sezgisel kurallar arasında bir ilişkinin varlığı kanıtlanabilir mi?
- ◆ Bir projenin karakteristiklerine bakılarak, hangi kuralın uygulanması gerektiği önceden belirlenebilir mi?
- ◆ Belirsizlik ortamında bulanık yapıda hazırlanan projelerde faaliyetlerin içerdiği belirsizlik başlangıçtan sona kadar veri kaybına uğramadan taşınabilmekte midir?
- ◆ ÇM-KKPC problemlerinin tüm olası uygun çözümleri başlangıç sentez aşamasında belirlenebilir mi?

Seri-paralel ve statik-dinamik sınıflandırma özelliklerini taşıyan çizelgeleme stratejilerinin etkinliği geniş ve kapsamlı bir çalışmanın konusu olamamıştır. Ancak, aynı öncelik kuralının hem seri hem de paralel yöntemlerde kullanıldığı bazı

arařtırmalarda, paralel yöntemlerin daha etkin olduđu belirtilmektedir. Dinamik yöntemlerin statik yöntemlere göre çoğunlukla daha iyi çözümler sağlamaktadır. Buna karřın hesap hacmini arttırarak çözüm sürecini uzatmaktadır.

Literatürde, özellikle ikinci sorunun yanıtının bulunmasına yönelik çalışmaların ağırlık kazandığı görölmektedir. Bu çalışmalarda, tek proje modellerinde amaç, genellikle “proje süresinin minimizasyonu”, çok proje modellerinde ise, “projelerde süre uzamalarının toplamının minimizasyonu” veya en büyük proje süresinin minimizasyonu şeklinde belirlenmiştir. Modellerde, işlemlerin bölünemez olduğu ve kaynak tahsisi yoğunluğunun işlem süresi boyunca sabit kaldığı varsayımları yapılmıştır. KKPÇP / TP / TK probleminde, basit sezgisel kurallarla elde edilen sonuçlar, optimal çözümlerle karşılaştırılmış ve “en erken başlama zamanı” kuralının hem statik hem de dinamik yöntemlerde diğerlerine göre daha iyi sonuç verdiği belirlenmiştir.

Yukarıda belirtilen çalışmalarda elde edilen sonuçların ışığında KKPÇ problemlerinde uygulanan öncelik kuralı yöntemlerine ilişkin řu özet değerlendirmeler yapılabilir:

- ◆ Yapılan çalışmalardan elde edilen sonuçlar, genel olarak, incelenen proje modellerine göre bazı öncelik kurallarının uygulanmasına ilişkin tavsiyeler dışında kesin yargılar içermemektedir.
- ◆ Çok ölçütlü, karmařık öncelik kuralları, tek ölçütlü basit öncelik kurallarına göre daha etkindir savı hiçbir araştırma sonucunda kanıtlanamamıştır.

Genel olarak yukarıda belirtilen üçüncü sorunun da yanıtları henüz bulunamamış, hangi programlama stratejisinin ve hangi öncelik kuralının uygulanabileceğı hakkında genel geçerliliğı olan somut bilgi ve veriler oluşturulamamıştır.

8.8 Tartışma ve Öneriler

Uygun kombinasyonun bulunması karakterli KKPÇ problemlerinin çözümü amacıyla son 40 yıl içinde geliştirilen çok sayıda yöntem yetersiz kalmıştır. Yapılan literatür taraması, geliştirilen programlama yöntemleriyle karar ağacı esasına dayanan sayılama yöntemlerinin problemin kesin optimal çözümlerini sağlamada, “çözüm hızı” açısından yetersiz kaldıklarını, programlamada öncelik kurallarının uygulanması veya yaklaşık alt optimal çözüm oluşturma esasına dayanan sezgisel

yöntemlerinde “çözüm kalitesi” açısından güvenilirlikten yoksun olduklarını ortaya koymaktadır.

TM-KKPÇ koduyla belirtilen 1. grupta yapılan temel varsayım, işlemlerin hangi kaynak kullanılarak yapılacağı ve dolayısıyla işlem süreleri ile maliyetlerinin belirlenmiş ve sabit olduğudur. ÇM-KKPÇ modellerinde ise, “gerçek” koşullar da dikkate alınarak işlemlerin farklı kaynak tahsis özelliklerine bağlı olarak farklı süre ve maliyetlerle gerçekleştirilebilecekleri dikkate alınmaktadır.

TM-KKPÇ probleminin geçmişte yoğun araştırmalara konu olmasına karşın, ÇM-KKPÇ probleminin çözümü amacıyla son yıllarda gerçekleştirilen az sayıda araştırma dışında çalışmalar yapılmamış olması bu çalışmanın temel dayanağını oluşturmuş ve en kapsamlı şekilde KKPÇ problemleri ele alınmıştır.

Sonlu sayıda olurlu çözüm içeren çok aşamalı bir karar destek problemi niteliği taşıyan KKPÇ probleminin çözümü amacıyla geliştirilen optimizasyon yöntemlerinin yetersizliği, sezgisel yöntemlerin güvenilirlikten yoksun oluşu, esas itibarıyla problemin olurlu çözümler uzayının deterministik ve probabilistik karakterli yöntemlerle taranmasının yanı sıra bulanık karakterli bulanık küme teorisini temel alan yaklaşımlarla bir dizi çözüm seçeneği oluşturularak en iyisinin seçilmesini yaklaşımları önerilmiştir.

Proje çizelgeleme, yeni araştırmalar için uygulayıcıların dikkate almaları gereken önemli çizelgeleme alanları mevcuttur. Çok önemli deterministik proje çizelgeleme problemleri kaynak kısıtını gözetmeksizin çok kolay şekilde ve iyi bir şekilde çözümlenebilmektedir. Kaynak kısıtlı problemler, kaynak kısıtının dikkate alınmadığı problemlere nazaran çok daha zordur. KKPÇ problemi çizelgeleme literatüründe önemli bir problemdir ve iyi bilinen pek çok problemin özel bir durumudur. Bu problemin çok önemli özel durumlarından bir tanesi de iş atölyesi çizelgeleme problemidir. Proje çizelgeleme probleminde ki kaynaklar, iş atölyesi problemlerinde makinelere eşdeğerdir.

İşlem süreleri rassal olduğunda problemler oldukça zorlaşır. Kaynak kısıtları olmaksızın problemleri çözmek zordur. PERT prosedürü her zaman memnuniyet verici çözümler vermez. Rassal işlem sürekli stokastik yapılı ve KKPÇ problemleri literatürde incelenmesi gereken konulardan olup üzerinde bir çok araştırmacının dikkatini çekmektedir. Çizelgelemenin bu alanı çok daha fazla dikkate layık olduğu

açıktır. Stokastik örneklemeye dayanan çözüm yaklaşımlarının KKPÇ problemlerine uygulanması ve bu amaçla etkin strateji ve yöntemlerin geliştirilmesi konusunda oldukça az sayıda çalışma gerçekleştirilmiştir.

Ancak, optimizasyon yöntemlerinin uygulamada yarattığı karamsarlık, sezgisel yöntemlerin bir türlü yükseltilemeyen güvenilirlikleri, fakat buna karşın bilgisayarların giderek güçlenmesi, yaygınlaşması ve ucuzlaması, uygulamada örnekleme yaklaşımlarının öneminin giderek artacağına işaret etmektedir. Ayrıca, karar vericiye bir dizi farklı çözüm seçeneğinin sunulmasını sağlayan bu yaklaşımların uygulamada esneklik kazandıracağı görüşü de giderek benimsenmektedir.

Bu gelişmelerin bir yansıması da model yapısında görülmektedir. Projelerin planlanması ve kontrolünde yapılması kaçınılmaz olan maliyet ve nakit akışı analizleri yanında, fiziki kaynakların ve işgücünün işlemlere tahsisinde karar vericiye esneklik kazandıracak farklı seçeneklerin modellerin yapısına girmesi ve böylece problemin çözümünde süre, maliyet ve kaynak kullanımına ilişkin unsurların birlikte ele alınması yönünde sürdürülen çabalar, yeni modellerin geliştirilmesinde bu özelliklerin de dikkate alınmasını gerektirmektedir.

Genel olarak, bir projenin en düşük toplam proje maliyeti yaratacak şekilde gerçekleştirilmesi amaçlanır. Projelerin planlanmasında, “kaynak” ve “maliyet” unsurları gözetilerek işlemlerin başlama zamanlarının tespit edilmesine dayanan proje çizelgeleme bu amacın sağlanması için çözülmesi gereken en karmaşık ve zor probemlerden birisidir.

Bu çalışmada projelerin kaynak kısıtları altında çizelgelenmesi problemi incelenmiş ve problemin çözümü için yeni ve etkin çözüm yöntemlerinin tanıtılması ve geliştirilmesi hedeflenmiştir. Optimal çözümü sağlayan modeller incelenerek ne derece uygulanabilir olduğunu araştırılmış ve sonucunda örnek problemler yardımıyla uygulama zorluklarını ortaya çıkarılmıştır. Ayrıca kullanılan yönteme bazı sezgisel öncelik kuralı esasına göre oluşturulan çözümler, başlangıç çözüm olarak kabul edilmiş ve alternatif olabilecek TA yöntemiyle sınanmıştır. Bu yöntemler arasında bir karşılaştırma yapılmıştır. Bunun yanı sıra literatürde yer alan çeşitli problemler üzerinde denemeler yapılarak TA sezgisel çözüm yöntemiyle probleme, güvenilir basit çözüm önerilerinde bulunulmuştur.

Proje yönetimi alanında, projelerin doğasından kaynaklanan belirsizliklerin çözümünde, bulanıklığın kullanımıyla “bulanık proje çizelgeleme” kavramı ortaya çıkmış ve bu konu, günümüzde artan önemi sebebiyle, birçok araştırmacının dikkatini çekmiştir. Bulanık proje çizelgelemede, projenin karmaşıklığı ve projede arıza veya bekleme oluşması gibi durumlar nedeniyle geleneksel yöntemlerle çözümü çok zor olan problemlerin çözümde meta sezgisel yöntemlerin kullanımı kısa sürede iyi çözümlerin elde edilmesini sağlamaktadır. Gerçek hayatta ilk defa yapılacak olan projelerde kullanılacak olan kaynaklar belirsizdir. Bu belirsizlik ancak bulanık mantık tekniklerinin kullanımı ile giderilebilir. Bulanık proje çizelgeleme problemlerinin çözümü ile ilgili literatürde henüz optimum çözüm veren meta sezgisel yöntemler geliştirilememiştir. İleriki çalışmalarda, yeni meta sezgisel yöntemlerin geliştirilmesi yada mevcut olanların melez olarak kullanımı ile optimuma yakın çözümler elde edileceği beklenmektedir. Henüz üzerinde fazla çalışılmamış meta sezgisel yöntemlerden **oryantirik** ve bu çalışma kapsamında da önerilen **paralel kanguru algoritmalarının** KKPC problemlerine uygulanmasını öneriyoruz. Anılan yöntemlerin probleme yenilik getireceği düşünülmektedir.

Probleme maliyet faktörlerinin ilave edilmesiyle genel bir bulanık çok amaçlı kombinatorial optimizasyon (fuzzy MOCO) problemlerine genişletilir. **Bulanık altında proje çizelgeleme çok amaçlı bulanık proje çizelgeleme problemi** Proje performans ölçütleri (amaçlar, kriterler) C (kriterler) kümesi ve amaçlar kümesi öncelik kısıtlı faaliyetler arasında tahsis edebilmek için (Weglarz 1980, Slowinski 1981) R kaynaklar kümesi tarafından karakterize edilebilir. R kümesi 3 kategoriye ait kaynaklardan oluşur. Yenilenebilir kaynaklar (kullanımı sınırlı), Tüketimi sınırlı yenilenemeyen kaynaklar, Tüketimi ve kullanımı sınırlı çift kısıtlı kaynaklar. Faaliyetler farklı kaynak gereksinimlerine sahiptir. Bunlar birkaç moduyla icraa edilebilir. Bir faaliyetin her bir icraa modunda farklı çoklu kategorilerde kaynak gereksinimli ve faaliyet süreleri tarafından tanımlanabilir. Bundan başka ayrıca her bir faaliyet, hazırlık zamanı bilinir. Faaliyetlerin tüm zaman parametrelerini olduğu farzedilir. Genellikle belirsizlik bulanık sayılarla modellenir ve tilda ile gösterilir. Faaliyetlerin çoklu icraa edilen modları problemin (zaman-maliyet trade-off) çoklu kriter doğasını gösterir. Proje çizelgeleme probleminde takip eden icraa ölçütleri genelde şu şekilde olduğu düşünülebilir; proje makespanı, kaynak kullanımının pürüzsüzleştirilmesi, maksimum gecikme, ortalama akış zamanı, net şimdiki değer,

proje maliyeti. Bu durumda fark ederiz ki faaliyetlerin belirsiz zaman parametreleri nedeniyle C kümesinde zaman ve zaman- maliyet kriterlerinin değerleride belirsizdir. Böylece bu değerler bulanık sayılardır. NP – hard proje çizelgeleme problemlerinde bir paralel çizelgeleme prosedürü en etkin sezgisel bulunur. En önemli özelliği ardışık düzende olası faaliyetlere düşünüldüğünde kaynak tahsis (atama) etmektir. Faaliyetler düşünülen kaynakları tahsis edilip edilemediğini doğrulayan bir prosedürdür. Bulanık zaman parametreleri üzerine paralel bir prosedür genelleştirmesi hapke ve slowinski (1996) tarafından önerilmiştir. Bu genelleme 2 sayının çözümüyle alakalıdır ilgilidir. Yeni faaliyetler için elde edilebilir yenilebilir veya çifli kısıtlı kaynaklarda zaman anları nasıl belirlenir (faaliyet güncel (o an ki) bulanık zaman anında tamamlanabilir. Bizim amacımız bulanık zaman parametreleriyle her bir gerçekleştirme için feasible uygun çizelgeler veren bir prosedür kurmaktır. Bu yönleriyle **çok amaçlı ve çok kriterli optimizasyon problemleri** yenilikçi uygulamalar olarak yeni araştırmacılar için önerilmektedir.

Bu problemin çözümünde kesin çözüm veren yöntemlerin kullanılması hem gerçekçilik açısından hem de hesaplanması için gerekli süre açısından uygun olmamaktadır. Belirsizliğin çözümünde en doğal yöntem, bulanık küme teorisidir. Bu yöntemin, proje çizelgelemede kullanılmasıyla daha etkili ve verimli çizelgeler oluşturmak mümkün olmaktadır. Bu çalışmada bulanık proje çizelgeleme problemlerinin meta sezgisel yöntemlerle çözümü araştırılmıştır. Gün geçtikçe artan rekabet ve belirsizlik ortamında, işletmeler ayakta kalabilmek ve rakiplerine karşı bir fark oluşturabilmek için daha çok çalışmaktadırlar. Böyle bir ortamda, projelerin başarılı bir şekilde yönetilmesi işletmeler için son derecede önemlidir. Projelerin çizelgelenmesi, proje yönetiminin en önemli parçalarından biridir. Ancak günümüzde sürekli olarak ürün çeşitliliğinin artması ve ömürlerinin kısalmasıyla her seferinde farklı faaliyetler ve süreçlerle uğraşılmasını ve bu da belirsizliklerle dolu bir sürecin planlanmasını ve dolayısıyla da çizelgelenmesini gerektirmektedir. Ürünlerin zamanında teslimi, üretim sürelerinin kısaltılması, stok maliyetlerinin azaltılması, vb. hedefler ancak doğru çizelgeleme teknikleriyle mümkün olabilir. Çizelgeleme, üretim ve hizmet sektörlerinin çoğunda temel olarak kullanılan bir karar verme sürecidir. Çizelgeleme, verilen zaman periyotlarında kaynakların görevlere atanması işlemini yapar ve hedefi, bir ya da birkaç amacı optimize etmektir. Çizelgeleme problemleri spesifik üretim görevlerine göre çok geniş bir alana

yayılmıştır ve birçoğu NP (Nondeterministic Polynomial-time)-zor problemlerdir. Gerçek hayatta çizelgeleme problemleri, iş çizelgeleme ve proje çizelgeleme problemleri olarak sınıflandırılır. Faaliyetlerin beklenen zamanda yapılması hem finansal (gerekli mali kaynağın temini) hem de operasyonel (gerekli kaynakların temini) açıdan çok önemlidir. Proje faaliyetlerinin çizelgelenmesi günlük çalışma saatleri, haftalık çalışma günleri, tatiller vb. yani çalışma takviminin belirlenmesiyle başlar. Her faaliyetin ne kadar süreceği tahmin edilir ve faaliyetler arasındaki öncelik ilişkilerine ve her faaliyetin tahmini süresine göre çalışma takvimi yardımıyla proje çizelgesi oluşturulur.

Teoride ve uygulamada yer alan **öncelik kuralı sezgisel çizelgeleme kurallarındaki eksiklikler** ise şu şekilde ortaya konulabilir:

Sezgisel esaslı kaynak tahsis problemleri arkasındaki temel fikir, bir veya daha fazla öncelik kurallarına dayanan faaliyetleri sınıflandırmak, daha sonra da faaliyetleri çizelgelemek için bir modelleme prosesi uygulamaktır. Öncelik politikasının seçimi her zaman sorgulanmıştır. Kısıt kaynaklı şebeke esaslı proje çizelgeleme problemleri üzerinde araştırmalar halen sürmektedir ve objektif bir şekilde uygun bir sezgiselin seçilebileceği tarzda kısıt kaynaklı projeleri sınıflandırmak henüz mümkün olmamıştır.

Bu çalışmada TA prosedürü, kısıt kaynaklı proje çizelgeleme problemleri için geçerli bir çözüm yöntemi olduğu ve oldukça iyi uygulamalar sunan, çözüm kalitesi bakımından diğer sezgisel yaklaşımlarla ve literatürde konuyla ilgili yapılan diğer çalışmalardan elde edilen sonuçlarla karşılaştırıldığında çok kısa bir sürede oldukça yüksek kaliteli çözümler üretebilmektedir.

Bu çalışmada KKPÇ probleminin tüm çalışma alanları kapsanmaya çalışılmıştır. Kaynakçada yer alan kaynak sayısı, bu alandaki yoğun araştırmaların bir göstergesidir. Ancak buna rağmen araştırmacılar henüz bir çok konuya yeterince el alamamışlardır. Böyle henüz yeterince işlenmemiş bir konu, proje ortamında belirsizliktir. Belirsizlik sadece KKPÇ için değil, proje yönetiminin tüm unsurları ve aşamaları için de kritik bir olgudur (Chapman ve Ward, 1997). Nitekim proje yönetimi yayınları arasında risk yönetimine ilişkin yayınların hızla arttığını görüyoruz. Bu nedenle, yönetim alanında risk yönetimi ve çizelgeleme alanında stokastik modelleme iki ilginç araştırma alanıdır. Stokastik modellemenin temel

katkısı gerçek durumları daha iyi yansıtmaya yönelik yeni modellerin tasarımıdır. Nakit akışlarındaki belirsizlik; kaynakların kullanımındaki belirsizlik; bir faaliyetin istenen standartlara uygun biçimde gerçekleştirilememesi nedeni ile kısmen veya tümü ile tekrarı; belirli faaliyetlerin sonuçlarına göre projenin topolojisinin değişmesi gibi hususların tatminkar bir biçimde modele dahil edilebilmesi ve matematiksel programlama yöntemleri ile çözüm getirilmesi teorik çalışmaların pratiğe yansıtılması ve pratikte kullanılabilir sonuçların türetilmesi bakımından önemli katkılar olacaktır.

Üretim planlama ve kontrolü araştırmalarında; malzeme yönetimi geniş bir yer tutar. Proje yönetimi araştırmalarında ise malzeme yönetimine aynı önemin verildiği söylenemez. Bu konuda görece olarak az sayıda araştırma yayımlanmıştır. Projenin ve gerektirdiği malzemenin müştereken planlanması ilginç bir araştırma alanıdır. Bir önemli uygulama-araştırma alanı da proje planlama yazılımlarının geliştirilmesidir. Proje planlama yazılımlarının, uzun uğraşlar sonucu elde edilen kuramsal katkıları içermesinin sağlanması, bu kuramsal uğraşların katma değerini artırması bakımından da değerlendirilmelidir. Ayrıca, bu tür bir ilişkinin, kuramsal çalışmaların yönünün belirlenmesine olan etkisi ile yönlendirici bir rolü de vardır.

Değişik zamanlarda, araştırmacılar, bazı makina çizelgeleme problemlerinin KKPÇ problemi şeklinde nasıl ifade edilebileceği ve iki ayrı yaklaşımın ilişkisi üzerinde çalışmış ve yollar önermişlerdir. Böyle bir ilişki kurulmasının hemen akla gelen pratik yansımaları ise, makina çizelgeleme ile proje çizelgeleme üzerinde çalışan araştırmacıların birbirlerinin araştırmalarından öğrenmeleri ve bazen iki alan için de geçerli yöntemler geliştirerek araştırmanın katma değerini artırmalarıdır.

Bilindiği gibi atölye çizelgeleme problemi NP-complete in the strong sense'dir (Garey vd., 1976). Atölye çizelgeleme problemi ve KKPÇP (P1) arasındaki ilişki Schrage (1970) ve daha sonra başka araştırmacılar tarafından irdelenmiştir. Sprecher (1994) atölye çizelgeleme probleminin KKPÇP'nin bir özel hali olduğunu göstermiştir.

Montaj hattı dengeleme problemi üretim yönetiminin eski problemlerinden birisidir. Montaj hattı dengeleme probleminde amaç, operasyonların atandığı istasyon adedinin enazlanması veya hattın çevrim süresinin enazlanmasıdır. İstasyon adedinin enazlanması problemi için Elmaghraby (1977), KKPÇP (P1) ile montaj hattı

dengeleme problemi arasındaki ilişkiyi göstermiştir. De Reyck ve Herroelen (1995) ise bu amaçla (SS) öncüllük ilişkilerini kullanmışlardır. Sprecher (1999) faaliyetin uygulaması esnasında değişken kaynak talebi kullanarak montaj hattı dengeleme problemi ile KKPCP arasındaki ilişkiyi yansıtmıştır.

Tek boyutlu kutu doldurma problemi ile KKPCP (P1) arasındaki ilişki Garey vd. (1976) tarafından irdelenmiştir. İki boyutlu kutu doldurma problemi için Schrage (1970) kutunun bir boyutunun KKPCP'nin zaman boyutu; ikinci boyutunun ise KKPCP'nin yenilenebilir kaynak boyutu ile ifade edildiği bir ilişki kurmuştur. Üç boyutlu kutu doldurma problemi için bir dönüşüm önerisi Hartmann (1999) tarafından verilmiştir. Hartmann, ayrıca, sırt çantası doldurma (knapsack packing) probleminin bir, iki ve üç boyutlu gösterimleri için dönüşüm önerileri geliştirmiştir.

Demeulemeester ve Herroelen (1996) üretim planlamanın hazırlık süreleri, katile üretimi gibi unsurlarını, KKPCP'nin en az bekleme süresi, termin, faaliyetin uygulanması esnasında değişken kaynak talebi gibi unsurlarını kullanarak ifade edebilmenin yollarını irdelemişlerdir. Yukarıda makina ve proje çizelgeleme arasındaki ilişkinin irdelendiği bazı çalışmalardan örnekler verilmiştir. Teknik yazın incelendiğinde bu konuda yayınların giderek arttığı gözlenmektedir. Bu alan da verimli bir araştırma alanı olarak gözükmektedir.

Oluşturulan matematiksel modelin kapsamını genişletmek üzere yapılacak bir diğer çalışma alanı olarak; modelde proje faaliyet sürelerinde mevcut bulunan belirsizliğin modellenmesi amacıyla bulanık dağılımlar kullanılmıştır. Modelin geliştirilmesi amacıyla kaynak kullanım maliyetlerindeki ve kaynak kullanım miktarlarındaki belirsizliklerin de bulanık dağılımlar kullanılarak modele dahil edilmesi mümkündür. Ayrıca, bu çalışmada ele alınan tek proje durumu genişletilerek çoklu proje durumu üzerinde çalışma yapılabilecektir. Açıkça belirtmek gerekir ki, bu yaklaşım yamuk bulanık faaliyet süreleriyle sınırlandırılmaz. Diğer tip bulanık sayılar içeren L-R tipi, üçgensel tip, vb. uygulanabilir. Ayrıca KKPCP problemlerine de uyarlanabilir.

Aynı konfigürasyona sahip bir bilgisayarda iki algoritma ile çözüm yapılmış olması her iki model arasında objektif bir değerlendirme imkanı da sunmuştur. Birçok araştırmacı tarafından yapılan çalışmalar farklı konfigürasyona sahip bilgisayarlar aracılığıyla yapıldığından çözümler ancak bilinen en iyi çözüm değerleriyle

kıyaslanmaktadır. Özellikle algoritmaların hızları bakımından yapılan değerlendirmeler yanıltıcı olabilmektedir.

Çalışmanın ticari programlardan en temel farkı, ticari amaçlı paket programların sezgisel yöntem kullanarak bir tek çözüm elde etmesidir. Oysa, bu çalışmada algoritmaları her bir çalıştırmada, farklı bir alternatif çözüm üretme imkanı sözkonusudur. Dolayısıyla paket programların aksine, birden fazla alternatif iyileştirilmiş çözüm bulma imkanı sağlanmış olmaktadır.

Çalışma KKPC problemlerinin farklı versiyonlarına uyarlanabilir niteliktedir. Bu çalışmada çok modlu KKPC problemleri, öncelik çizelgeleme metoduna göre kaynak kısıtlı çizelgeleme problemleri ele alınmıştır. Sonraki çalışmalarda kaynak yatırım problemleri, net bugünkü değer problemleri, çok modlu kaynak kısıtlı öncelik çizelgeleme problemlerine de uyarlanarak çözümler geliştirilebilecektir.

Bu çalışmada ele alınan B-ÇM-KKPC problemi ile ilgili olarak bundan sonraki dönemde yapılabilecek çalışmalar üç alanda toplanmaktadır. Bu alanlardan birincisi, ele alınan proje ağında mevcut bulunan belirsizliklerin modellenmesinde kullanılacak yöntemle ilgili yapılabilecek çalışmaları kapsamaktadır. İkinci alandaki çalışmalar matematiksel modelin kapsamının genişletilmesini konu olabilecektir. Çalışma yapılabilecek üçüncü alan ise oluşturulacak matematiksel modelin çözüm yönteminin geliştirilmesi alanıdır.

Çalışma yapılabilecek birinci konu matematiksel modele dahil edilen proje faaliyetlerindeki belirsizliklerin modellenmesinde uygun bulanık sayı üyelik fonksiyonlarının belirlenmesidir. Proje uzmanlarından alınan bilginin modellenmesinde kayıp yaşanmaması ve aynı zamanda model sonuçlarında iyi sonuçlar elde edilebilmesi amacıyla uygun bulanık sayı üyelik fonksiyonlarının araştırılması büyük önem taşımaktadır. Ayrıca, uygun bulanık sayı üyelik fonksiyonlarının belirlenmesinde ele alınan projede kullanılacak faaliyetlerin özelliklerinin göz önünde bulundurulması gerekmektedir.

Oluşturulan matematiksel modelin kapsamını genişletmek üzere yapılacak çalışmalar ikinci alanı oluşturmaktadır. Modelde proje faaliyet sürelerinde mevcut bulunan belirsizliğin modellenmesi amacıyla bulanık dağılımlar kullanılmıştır. Modelin geliştirilmesi amacıyla kaynak kullanım maliyetlerindeki ve kaynak kullanım miktarlarındaki belirsizliklerin de bulanık dağılımlar kullanılarak modele dahil

edilmesi mümkündür. Ayrıca, bu çalışmada ele alınan tek proje durumu genişletilerek çoklu proje durumu üzerinde çalışma yapılabilir.

Konu üzerinde çalışma yapılabilir üçüncü alan ise çözüm yönteminin geliştirilmesi konusudur. Literatürde KKPC problemi hesaplama karmaşıklığı olan bir problem türü olarak tanımlanmıştır. Problemin çözümünde kullanılan başlıca yöntemler arasında dal-sınır algoritmaları, sezgisel algoritmalar ve metasezgisel yöntemler yer almaktadır. Başlangıç çözümlerinin bulunması ve çözüm uzayının sınırlandırılması yöntemleri problemin optimal veya optimale yakın çözümünde büyük önem taşımakta olup iyi başlangıç çözümlerinin bulunması ve aranacak çözüm uzayını iyi bir şekilde sınırlayan yöntemler üzerinde çalışma yapılabilir. Ayrıca, başlangıç çözümü bulunmuş ve çözüm uzayı sınırlandırılmış problemin çözümünde özellikle genetik algoritma, benzetilmiş tavlama, TA, kanguru algortması, oryantirik, arı kolonisi algoritması, kuş sürüsü algoritması, nöral network algortması, kurbağa zıplama algortması, renk algortması, demster shaffer algortması vb metasezgisel arama yöntemleri kullanılarak problemin çözülmesi üzerinde çalışmalar yapılması mümkün bulunmaktadır.

Çok mod'lu olarak anılan; alternatif faaliyet süreleri ve/ya da alternatif kaynak türü ve miktarları kullanılabilmesi de algoritmaların işlevselliğini daha da artıracaktır. Kaynak kısıtlı problemlerde, bir faaliyetin her bir süre - kaynak tanımı tek bir mod olarak tanımlanmaktadır. Faaliyetlere alternatif süre ve/ya da kaynak tanımı yapılması halinde çok mod (multi-mode) anlamını taşımaktadır. Faaliyetlere atanacak her bir süre-kaynak şeklinde atanan alternatifler farklı bir mod olarak değerlendirilmektedir. (Erenguç, 2001). Faaliyetlere mod olarak atanan değerler, değişen ya da değişebilecek şartlardan ya da olası gelişmelerden kaynaklanabilmektedir. Çok mod'un tanımından da anlaşıldığı üzere, proje çizelgeleme aşamasında, öngörülebilir olası alternatiflerin değerlendirmeye katılması ile yapılan en iyi çözüm arayışı, çizelge çözümünün daha gerçekçi olmasını sağlayacaktır. Bu nedenle, değişebilecek şartlara hazırlıklı olunması amacıyla alternatif çözüm olarak, algoritmalarda değerlendirilmeye alınabilir bir özelliktir.

Çok mod'un uygulandığı algoritmalarda, mod değerlerinin seçimi rastgele ya da içlerinden en iyi değeri temsil eden durum seçilerek yapılabilmektedir. Faaliyet süresi ya da kullanılan kaynakların seçiminden sonraki işlemler, tek mod'lu algoritmalarda olduğu gibi devam etmektedir. Böylece öngörülen alternatiflerde

değerlendirmeye katılarak, değişen koşullara en uygun çözümler aranabilecektir. Yukarıda altı madde halinde belirtilen bu öneriler doğrultusunda isteğe bağlı (opsiyonel) olarak seçilebilir bir yapıda hazırlanmalıdır. Böylece algoritma, birçok çizelgeleme projelerinde gereksinim duyulabilecek amaç ve işlevleri destekler hale gelecektir. İsteğe bağlı seçimlerin tek ya da çoklu yapılabilmesi ise algoritmanın etkinliğini daha da artıracaktır. Bu şekilde çeşitlendirilmiş arama işlemleri neticesinde üretilen farklı çözümleri bulma olasılığı artarken, en iyi ya da en iyiye çok yakın proje çizelgelerinin elde edilmesi söz konusu olabilecektir.

Alınan sonuçlara bakıldığı zaman, alternatif tabu algoritmalar arasında permütasyon temelli algoritma, öncelik değerini temel alan algoritmaya üstün gelmiştir. Bu sonuçlar TA algoritmanın gerçek dünya problemlerine uygulanmasının mümkün olduğunu göstermektedir. Bu araştırmadaki hesaplamalar sonucu permütasyon temelli tabu algoritmanın her çeşit kısıtlı kaynaklı proje çizelgeleme probleminde hızlı ve optimale yakın sonuçlar verdiği görülmüştür.

KKPÇ problemleriyle ilgili en önemli problem bilgisayarların sınırlamalarından kaynaklanmaktadır. Standart donanımlı bir bilgisayarla matlab bilgisayar programla dili kullanıldığında 9 faktöriyel işlemi hesaplayabilmekte 10 faktöriyel hesaplayamamaktadır. 10 faaliyetten sonra her bir faaliyet 10 kat çözüm uzayı büyüklüğü arttırmaktadır. Buradaki problem matematiksel olarak bilgisayarların bu hesabı gerçekleştirememesidir. Konuyla ilgili matematiksel bir engel bulunduğunu söylemek en doğrusu olacaktır. 1000 faaliyetten oluşan bir projede 1000 faktöriyel hesaplamakta ki güçlüğü giderebilmek için matematiksel olarak yeni yöntemler geliştirilmelidir.

KAYNAKLAR

- Altaş, İ. H.** (1999). Bulanık Mantık : Bulanıklılık Kavramı, Enerji, Elektrik, Elektromekanik-3e, S. **62**, S.:80-85, *Bilesim yayıncılık A.Ş.*, İstanbul,
- Altaş, İ. H.** (1999). Bulanık Mantık : Bulanıklılık Denetim, Enerji, Elektrik, Elektromekanik-3e, S. **64**, S.:76-81, *Bilesim yayıncılık A.Ş.*, İstanbul,
- Atlı, Ö.** (2005). Kısıtlı Kaynaklar Altında Proje Yönetimi ve Bir Uygulama, *Yüksek Lisans Tezi*. S.124.
- Barutçugil, İ.** (1998). Üretim Sistemleri ve Yönetim Teknikleri, *Beta Yayınevi*, Bursa.
- Buckley, J.J.** (1989) Fuzzy PERT. In Applications of Fuzzy Set Methodologies in Industrial Engineering, *Elsevier*, Amsterdam, p.103-114.
- Brucker,P., Drexl,A., Möhring, R. Neumann,K. ve Pesch.E.** (1999). Resources-Constrained Project Scheduling: Notation, Classification, Model and Methods. *European Journal of Operational Research*, **112**, p. 3-41.
- Brucker,P.,Knust S.** (2006). Complex Scheduling, *GOR Publication. Springer* p. 1-188.
- Chen, S. Hsueh, Y.**, (2008). A simple approach to fuzzy critical path analysis in project Networks, *Applied Mathematical Modelling* **32** 1289–1297.
- Chanas, S. and Kamburowski, J.** (1981) The use of fuzzy variables in PERT. *Fuzzy Sets and Systems* **5** 11-19.
- Chanas, S.** (1982) Fuzzy sets in few classical operational research problems. In Approximate Reasoning in Decision Analysis (M. M. Gupta and E. Sanchez, Eds.), North Holland, Amsterdam, 351-363.
- Chanas, S. and Zielinski, P.** (2001) Critical Path Analysis in the Network with Fuzzy Activity Times, *Fuzzy Sets and Systems*, Vol. **122**, 195-204.
- Çubuk. R.** (2008) Proje Yönetiminde Zaman Ve Maliyet Risklerinin Çizelgeleme Yöntemiyle Minimize Edilmesi. *Doktora Tezi*, Adana, Çukurova Üniversitesi, p. 158.
- Damak N, Jarboui B, Siarry P, Loukil T.** (2009). Differential evolution for solving multi-mode resource-constrained project scheduling problems. *Computers Operations Research*. **36**, p. 2653-59.
- Drexl,A., Nissen R., Patterson J. ve Salewski F.** (2000). PROGEN/ΠX – An Instance Generator For Resources-Constrained Project Scheduling Problems With Partially Renewable Resources And Further Extensions. *European Journal of Operational Research*, **125**, p. 59-72.
- Dubois, D. and Prade, H.** (1988) In Possibility Theory: An Approach to Computerized Processing of Uncertainty, *Plenum*, New York, 62-63.

- Dubois, D. and Prade, H.** (1980). Fuzzy Sets and Systems: Theory and Applications. *Academic Press*.
- Dubois, D. and Prade, H.** (1979). Decision making under fuzziness. In *Advances in Fuzzy Set Theory*, North-Holland, Amsterdam, 279-302.
- Ecer, F.** (2007). Üyelik fonksiyonu olarak üçgen bulanık sayılar mı yamuk bulanık sayılar mı? *Gazi Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi* 9/2, p. 161-180
- Elsayed,E.A.** (1982). Algorithm for Project Scheduling with Resource Constraints. *International Journal of Production Research*, Vol.20, p. 95-103
- Erdal, M.** (2007). Kısıtlı Kaynak Koşullarında Yapı Projelerinin Genetik Algoritma ile Programlanması. *Doktora Tezi*, Ankara, Gazi Üniversitesi. P. 122.
- Erenguç S.S. Ahn T.** (1998). The resource constrained project scheduling problem with multiple crashable modes: A heuristic procedure. *European Journal of Operational Research* 107, 250-259.
- Fortemps, P. Roubens, M.** (1996). Ranking and defuzzification methods based on area compensation, *Fuzzy Sets and Systems*, 82 319–330.
- Gazdik, I.** (1983). Fuzzy network planning. *IEEE Trans, Reliability R-32* August 304-313.
- Gen M, Cheng R, Lin L.** (2008). Network models and optimization multi objective genetic algorithm approach *Springer*. 420-475.
- Glover,F.** (1989). Tabu Search-Part I, ORSA. *Journal on Computing*, V.1 No.3, p. 190-206.
- Glover F.** (1990). Tabu Search-Part II, ORSA. *Journal on Computing*, Vol.2 No.1 p.10-32.
- Hapke, M. and Slowinski, R.** (1996). Fuzzy Priority Heuristics for Project Scheduling, *Fuzzy Sets and Systems*, Vol.83, 291-299.
- Hapke, M. Jaskiewicz, A. and Slowinski, R.** (1998). Interactive Analysis of Multiple-criteria Project Scheduling Problems, *European Journal of Operational Research*, Vol.107, 315-324
- Hapke, M., Jaskiewicz, A., Slowinski, R.** (2000). Pareto Simulated Annealing For Fuzzy Multi-Objective Combinatorial Optimization. *Journal Of Heuristics*, 6: 329-345.
- Hartmann,S. ve Kolisch R.** (2000). Experimental Evaluation of State-of-The-Art Heuristics for the Resource-Constrained Project Scheduling Problem. *European Journal of Operational Research*, 127,p. 394-407.
- Hekimoğlu, Ö.** (2007). Sınırlı Kaynak Kullanımlı Proje Çizelgelemesi Problemlerinde Proje Yönetimi Yazılım Paketlerinin Kaynak Atama Kabiliyetlerinin Karşılaştırılması. *Yüksek Lisans Tezi*. Ortadoğu Teknik Üniversitesi, Ankara. P. 178.
- Hsiau; H. J. and Chun Lin, Wei R.** (2009). A fuzzy pert approach to evaluate plant construction project scheduling risk under uncertain resources capacity *Journal of Industrial Engineering and Management*, Vol 2, p 31-47.

- İçmeli, O. ve Rom W.** (1995). Solving The Resource Constrained Project Scheduling With Optimization Subroutine Library. *International Journal of Production Research*, Vol.23, No.8, p. 801-817
- Jaszkiewicz A.** (2001). Multiple Objective Metaheuristic Algorithms for Combinatorial Optimization. *Politechnika Poznanska*, Poznan.
- Khamooshi, H.** (1999). Dynamic Priority-Dynamic Programming Scheduling Method (DP)2SM : a dynamic approach to resources constraint project scheduling. *International Journal of Production Management*, Vol. 17, No. 6, p. 383-391.
- Kalender F.Y, Yılmaz M.M., Türkbey O.** (2008). Montaj Hattı Dengeleme Problemine Bulanık Bir Yaklaşım. *Gazi Üniversitesi Müh.Mim.Fak.Dergisi*. Cilt 23, no1.129-138,
- Kavlak, N.** (2005). Client-Contractor Bargaining Problem In The Context Of Multimode Project Scheduling With Limited Resources, *Yüksek Lisans Tezi. Sabancı Üniversitesi*. İstanbul. P.78.
- Kılıç, M.** (2003). Risk Altında Proje Çizelgeleme Problemine Genetik Algoritma Çözüm Yaklaşımları, *Sabancı Üniversitesi*, İstanbul. p. 112.
- Kim K.W, Gen M, and Yamazaki G.** (2003). Hybrid Genetic Algorithm with fuzzy logic for Resource-constrained project scheduling. *Applied Soft Computing Elsevier* 2/3 F. 174-188.
- Kökçam, A.H., Engin O.** (2010). Solving the fuzzy project scheduling problems with metaheuristic methods. *YTÜ Sigma Dergisi*. P.86-101.
- Liang t.F.** (2006). Project Management decisions using fuzzy linear programming, *International Journal Of Sytems Science*, vol. 37, no,15, p1141-1152.
- Liu, S.-T.** (2003). Fuzzy Activity Times in Critical Path and Project Crashing Problems. *International Journal Cybernetics and Systems*., 34: 161-172.
- Liu B.** (2008). Theory and Practice of Uncertain Programming. *Studies in Fuzziness and Soft Computing*. Vol.239. Springer p. 139-146.
- Lova A, Tormos P, Barber F.** (2006). Multi-mode Resource constrained project scheduling, *Intelligencia Artifical*. İspanya.
- Mamdani, E.H. ve Assilian, S.** (1975). An experiment in linguistic synthesis with a fuzzy logic controller, *Int. J. Man-Machine Studies*, 7, 1-13,.
- Mahmoudoff A.** (2006). Nonlinear Optimization For Project Scheduling And Resource Allocation Under Uncertainty *Doktora Tezi*. P. 156.
- Mares, M.** (1989). Network analysis of fuzzy technologies. In Applications of Fuzzy Set Methodologies in Industrial Engineering (G. W. Evans, W. Karwowski, and M. R. Wilhelm, Eds.), *Elsevier*, Amsterdam, 115-125.
- McCahon C. S. and Lee, E. S.** (1988) Project network analysis with fuzzy activity times. *Comput. Math. Appl.* 15(10) 829-838.

- Milca M, Waligoro G, Weglarz J.** (2008). Tabu search for multi mode resource constrained project scheduling with schedule-dependant setup time. *European Journal of operational Research*.**187**, 1238-50.
- Mori M, Tseng C.C.** (1997). A genetic algorithm for multi mode resource constrained project scheduling problem. *European Journal of operational Research*. **100**, 134-141.
- Nasution, S.H.** (1994). Fuzzy Critical Path Method, *IEEE Transportation Systems Man Cybernet*, Vol. **24**, 48-57.
- Nasution, S. H.** (1993). Fuzzy durations in critical path method. In *Second IEEE International Conference on Fuzzy Systems*, San Francisco, 1069-1073.
- Özdamar, L., Alanya, E.** (2000). Uncertainty modelling in software development projects (with case study). *Annals of Operations Research*, **102** 157–178.
- Özdamar, L., Ulusoy, G.** (1995). A survey on the resource-constrained project scheduling problem. *IIE Transactions*, **27**, 574-586.
- Özdemir G.** (2006). Kısıtlı Kaynaklarla Proje Çizelgelemesi Prpbblemlerinde Kullanılan Genetik Algoritma Metodları ve Bunların Karşılaştırılması. *Yüksek Lisans Tezi*, Ankara Üniversitesi Ankara p. 122.
- Özen S. Ç.** (2005). Geliştirilmiş Ağırlıklı Konumlandırma Yöntemiyle Kaynak Kısıtlı Birden Fazla Projenin Çizelgelendirilmesi, *Yüksek Lisans Tezi*, Marmara Üniversitesi, p. 154.
- Paksoy, S.** (2007). Genetik Algoritma İle Proje Çizelgeleme. *Çukurova Üniversitesi*. Adana, p. 218.
- Pan,H. and Yeh, C.** (2003) Fuzzy Project Scheduling. *Monash University*.Australia
- Pan,H. Willis R.J, Yeh, C.**Resources-constrained Project Scheduling with fuzziness.
- Pape, L.C.** (1994). Implementation of Resource constraints in ILOG schedule. *Intelligent Systems Engineering*. **3(2)**, 55-66.
- Patterson.H.J.** (1984). A Comparison of Exact Approaches For Solving The Multiple Constrained Resource, Project Scheduling Problem. *Management Science*, Vol.**30** No.7, p. 854-867.
- Pinedo.M.** (1995). Scheduling Theory, Algorithms and Sysytems, *Prentice- Hall International*. USA.
- Sarı, T.** (2008). Metasezgisel Yöntemlerle Proje Çizelgeleme Optimizasyonu. *Doktora Tezi*, Marmara Üniversitesi, İstanbul. P. 161.
- Sharafi M., Jolai F., Iranmanesh H., Hatefi S.M.** (2008). A Model for Project Scheduling with Fuzzy Precedence Links., *Australian Journal of Basic and Applied Sciences*, **2(4)** 1356-1361.
- Slyeptsov, A.I. and Tyshchuk, T.A.** (2003). A Method of Computation of Characteristics of Operations in a Problem of Fuzzy Network, *Cybernetics and Planning and Management System Analysis*,Vol. **39**, No. 3, 367-378.

- Soltani A., Haji R.** (2007). A Project Scheduling Method Based on Fuzzy Theory, *Journal of Industrial and Systems Engineering*, Vol.1, No.1, pp 77-80.
- Stein, A. and Teske, E.** The Parallelized Pollard Kangaroo Method in Real Quadratic Function Fields, *Mathematics of Computation*, Vol. 71, No. 238, 793.
- Shipley F.** A fuzzy logic approach for determining expected values; a project management applications. *JSTOR*.
- Şimşir F.** (2008). Kaynak Kısıtlı Bakım Çizelgeleme Problemine Bir Hibrid Çözüm Yaklaşımı, *Doktora Tezi*. Sakarya. P.147.
- Taha,H.** (1987). Operations Research An Introduction. *Prentice-Hall International, Inc*, Sixth Edition, USA.
- Teske, E.** On Random Walks For Pollard's Rho Method, *Mathematics of Computation*, Vol. 70, No. 234, Pp 809.
- Thomas,P.R ve Salhi.S.** (1998). A Tabu Search Approach For The Resource Constrained Project Scheduling Problem. *Journal of Heuristics*, 4, p.123-139.
- Tsai.Y. ve Gemmill D.** (1998). Using Tabu Search To Scheduling Activities Of Stochastic Resource-Constrained Projects. *European Journal of Operational Research*, 111, p. 129-141.
- Tseng, C.C.** (2008). Two heuristic Algorithms for a multi mode resource constrained multi project scheduling problem, *Journal of science and engineering Technology*. V.4, N.2. 63-74.
- Tsujimura, Y. Chang, I. S. and Gen M.** (1993). An effective method for large scale project planning with multiple fuzzy activity times. In *Second IEEE International Conference on Fuzzy Systems*, San Francisco, 1009-1015.
- Uğur,İ.** (1988). Kaynak Kısıtlı Proje Programlama Problemlerinin Çözümü İçin Yeni Yöntem ve Algoritmalar. *Doktora Tezi*, İstanbul Teknik Üniv.
- Wang, J.** (2002). A fuzzy project scheduling approach to minimize risk for product development. *Fuzzy Sets and Systems*, 127, 99-116.
- Wang, J.** (2004). A Fuzzy Robust Project Scheduling For Product Development Projects. *European Journal Of Operational Research*, 152, 180-194.
- Wang X., Huang W.** (2010). Fuzzy Resource-Constrained Project Scheduling Problem for Software Development, *Journal of Natural Sciences*. Vol.15 no1 p,25-30,
- Weglarz J.** (2007). Scheduling Under Resource Constraints, *International Handbooks of Information Systems*, Springer p.425-475.
- Yager, R.R.** (1981) A procedure for ordering fuzzy subsets of the unit interval, *Information Sciences*, 24, 143–161.
- Yousefli, A., Ghazanfari, Shahanaghi, & Heydari, M.** (2008). A New Heuristic Model for Fully Fuzzy Project Scheduling, *Journal of Uncertain Systems*, 2, 75-80.
- Zadeh, L.A.** (1965). Fuzzy sets, *Information and Control*, 8, 338-353,.

Zadeh, L.A. (1973). Outline of a new approach to the analysis of complex systems and decision processes, *IEEE Transactions on Systems, man, and Cybernetics*, Vol. **SMC-3**, No. **1**, 28-44.

Zadeh, L.A. (1975). The concept of linguistic variable and its application to approximate reasoning, Part 1, 2, and 3. *Information Sciences*, **8**: 199-249, 8: 301-357-, 9: 43-80,

Zimmerman, H. J. (1996). Fuzzy Set Theory and Its Applications, **Kluwer Academic Publishers**.

Url-1 <<http://129.187.106.231/psplib/>>, alındığı tarih: 25.05.2012.

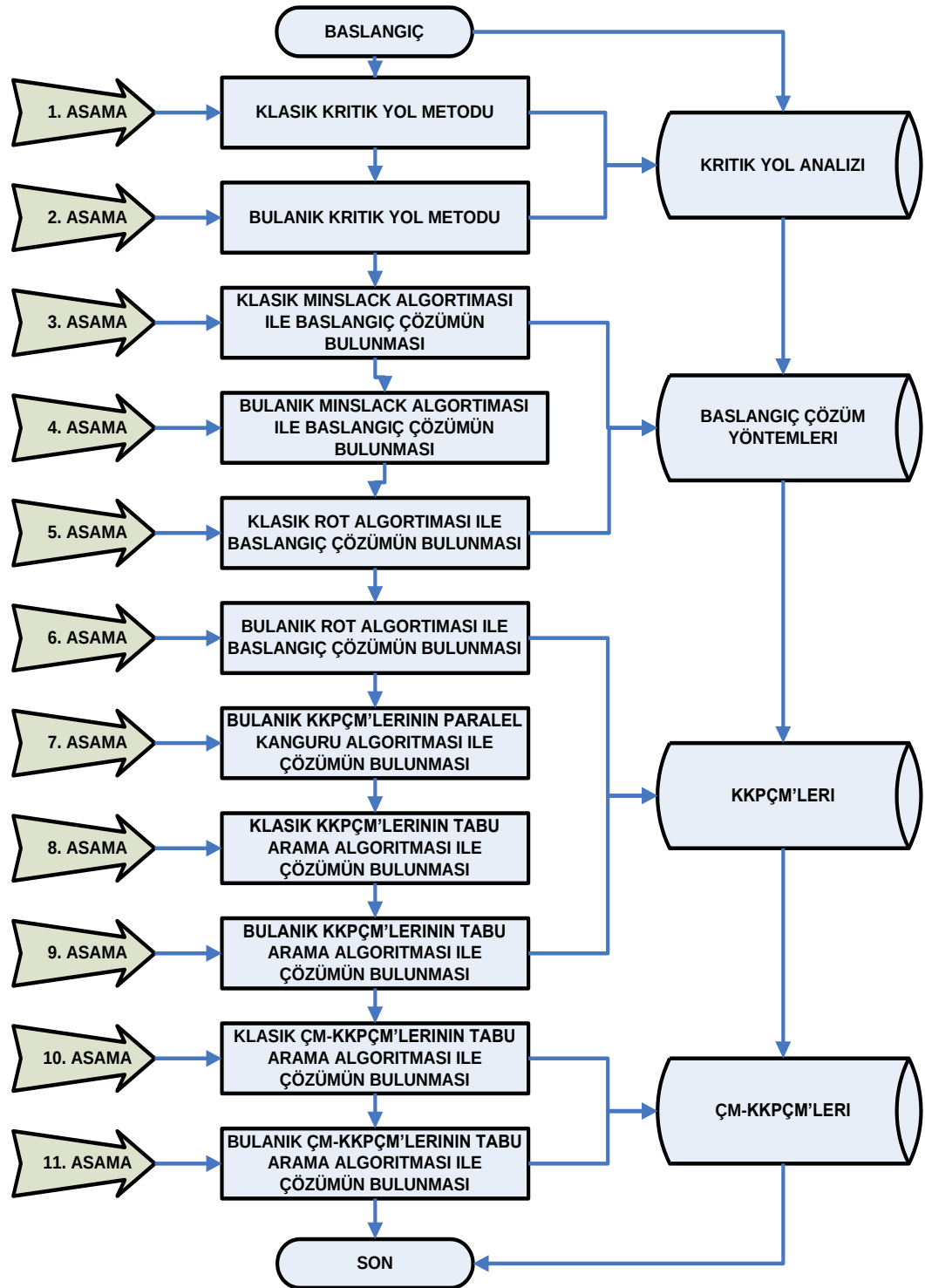
Url-2 <<http://www.cs.put.poznan.pl/~and-j/psa.html>>, alındığı tarih: 25.05.2012.

Url-3 <<http://www.projectmanagement.ugent.be/downloads.html>>, alındığı tarih: 25.05.2012.

Url-4 < <http://www-01.ibm.com/software/websphere/products/optimization/cplex-studio-preview-edition/>>, alındığı tarih: 29.07.2012.

EKLER

EK-A	Tezin Aşamaları	245
EK-B	Pat3.rcp Program Çıktısı.....	247
EK-C	TM ve ÇM-KKPÇP için Deterministik/Crisp Veri Kümelerinin Yapısı	249
EK-Ç	J30 Veri Kümesi için 1000 iterasyonlu TA ile Elde Edilen Sonuçlar	253
EK-D	J60 Veri Kümesi için 1000 iterasyonlu TA ile Elde Edilen Sonuçlar	257
EK-E	J90 Veri Kümesi için 1000 iterasyonlu TA ile Elde Edilen Sonuçlar	261
EK-F	J10.MM Veri Kümesi için 1000 İterasyon TA ile Elde Edilen Sonuçlar	265
EK-G	Microsoft Visual C# 2008 Programı Geliştirilen KKPÇ Programı Project Scheduling	269
EK-Ğ	Project Scheduling Programının Ekran Çıktıları	271
EK-H	Bulanık B-KYM için Örnek Problem	275
EK-I	KKPÇP'leri için OPL-CPLEX Kodu	277
EK-İ	ÇM-KKPÇP'leri için OPL-CPLEX Kodu	279



Şekil A.1 : Tezin Aşamaları

EK-B

No	Süre	EST	ECT	LST	LCT	SLACK
S	0	0	0	0	0	0
1	3	0	3	0	3	0
2	5	0	5	4	9	4
3	6	3	9	3	9	0
4	2	3	5	8	10	5
5	3	5	8	10	13	5
6	3	5	8	15	18	10
7	4	5	9	9	13	4
8	5	9	14	13	18	4
9	4	9	13	9	13	0
10	2	13	15	13	15	0
11	3	15	18	15	18	0
E	0	18	18	18	18	0

KRÝTÝK YOL

Yol: S - 1 - 3 - 9 - 10 - 11 - E

Süre: 18

MIN SLACK

NO	-	SÜRELER	-	BAĞLANGIÇ	-	BÝTÝP	-	ÖNCEL	-
	KAYNAK	KULLANIMI		KALAN	KAYNAK	MÝKTARI			
0	0	0	0	-	-	0 0 0	-	6 7 6	
1	3	0	3	-	0	3 2 1	-	3 5 5	
2	5	0	5	-	0	2 4 2	-	1 1 3	
3	6	3	9	-	1	3 1 2	-	1 2 2	
7	4	5	9	-	2	3 1 1	-	0 5 3	
9	4	9	13	-	3	3 2 3	-	3 5 3	
8	5	9	14	-	7	2 2 2	-	1 3 1	
4	2	13	15	-	1	4 3 1	-	0 2 3	
5	3	15	18	-	4	2 0 3	-	4 7 3	
6	3	15	18	-	4	1 1 1	-	3 6 2	
10	2	18	20	-	5	4 1 0	-	2 6 6	
11	3	20	23	-	10	5 4 2	-	1 3 4	
12	0	23	23	-	11	0 0 0	-	6 7 6	

Bağlangıç Çözüm: S - 1 - 2 - 3 - 7 - 9 - 8 - 4 - 5 - 6 - 10 - 11 - E

Proje tamamlanma süresi: 23

TABU

Maksimum Iterasyon Sayısı: 100

Tabu Liste Uzunluğu: 3

-----1 nolu Iterasyon-----

<1-2>, PTZ: 23

<2-3>, PTZ: 23

<7-9>, PTZ: 23

<9-8>, PTZ: 23

<9-4>, PTZ: 22

<8-4>, PTZ: 23

<5-6>, PTZ: 23

<6-10>, PTZ: 23

Bu Iterasyondaki En İyi Hareket: 4 - 9

Bu Iterasyondaki maliyet: 71
 Bu Iterasyondaki Tabu Listesi: {<4-9>,
 Genel Çözümde En İyi Hareket: 4 - 9
 0 Nolu Iterasyondaki En İyi Hareketin Ardýndan Olupán
 Baþlangýç Çözümü:
 0 - 1 - 2 - 3 - 7 - 4 - 8 - 9 - 5 - 6 - 10 - 11 - 12 -
 -----2 nolu Iterasyon-----
 <1-2>, PTZ: 23
 <2-3>, PTZ: 26
 <7-4>, PTZ: 27
 <4-8>, PTZ: 23
 <8-9>, PTZ: 23
 <5-6>, PTZ: 23
 <6-10>, PTZ: 23
 Bu Iterasyondaki En İyi Hareket: 9 - 8
 Bu Iterasyondaki maliyet: 71
 Bu Iterasyondaki Tabu Listesi: {<4-9>,<9-8>,
 1 Nolu Iterasyondaki En İyi Hareketin Ardýndan Olupán
 Baþlangýç Çözümü:
 0 - 1 - 2 - 3 - 7 - 4 - 9 - 8 - 5 - 6 - 10 - 11 - 12 -
 -----3 nolu Iterasyon-----
 <1-2>, PTZ: 24
 <2-3>, PTZ: 27
 <7-4>, PTZ: 28
 <4-8>, PTZ: 22
 <5-6>, PTZ: 24
 <6-10>, PTZ: 24
 Bu Iterasyondaki En İyi Hareket: 8 - 4
 Bu Iterasyondaki maliyet: 71
 Bu Iterasyondaki Tabu Listesi: {<4-9>,<9-8>,<8-4>,
 2 Nolu Iterasyondaki En İyi Hareketin Ardýndan Olupán
 Baþlangýç Çözümü:
 0 - 1 - 2 - 3 - 7 - 8 - 9 - 4 - 5 - 6 - 10 - 11 - 12 -
 -----4 nolu Iterasyon-----
 <1-2>, PTZ: 23
 <2-3>, PTZ: 26
 <7-8>, PTZ: 27
 <5-6>, PTZ: 23
 <6-10>, PTZ: 23
 Bu Iterasyondaki En İyi Hareket: 6 - 5
 Bu Iterasyondaki maliyet: 71
 Bu Iterasyondaki Tabu Listesi: {<9-8>,<8-4>,<6-5>,
 3 Nolu Iterasyondaki En İyi Hareketin Ardýndan Olupán
 Baþlangýç Çözümü:
 0 - 1 - 2 - 3 - 7 - 8 - 9 - 4 - 6 - 5 - 10 - 11 - 12 -
 -----5 nolu Iterasyon-----
 -----6 nolu Iterasyon-----
 -----7 nolu Iterasyon-----
 -----8 nolu Iterasyon-----
 -----9 nolu Iterasyon-----

 -----999 nolu Iterasyon-----
 -----1000 nolu Iterasyon-----

EK-C

Çizelge C.1 : TM-KKPÇP için Deterministik/Crisp Veri Kümelerinin Yapısı

J	R									
K(1)	K(2)	...	...	...	K (R)					
d(1)	k(1,1)	...	...	...	k(1,R)	S(1)	S(1,1)	...	...	S(1,S(1))
...	...	...	...	...	...	...	...	...	...	...
d(J)	k(J,1)				k(J,R)	S(J)	0			

$j = 1,2, \dots, J$: Faaliyetler
 $r = 1,2, \dots, R$: Kaynak tipleri
 $S(j)$: j faaliyetinin ardıl işlemlerinin sayısı
 $S(j, s)$: j faaliyetinin ardıl işlemlerinden s nci olan faaliyeti
 $d(j)$: j işinin süresi
 $K(r)$: herbir periyotta r kaynak tipinden maksimum elde edilebilir miktar
 $K(j,r)$: r kaynak tipinden j faaliyeti için gereksinim duyulan miktar

Çizelge C.2 : ÇM-KKPÇP için Deterministik/Crisp Veri Kümelerinin Yapısı

J	R	NR								
RESOURCES										
- renewable			: 2	R						
- nonrenewable			: 2	N						
- doubly constrained			: 0	D						
PRECEDENCE RELATIONS:										
jobnr.	#modes		#successors	successors						
j	m		S(1)	S(1,1) ..	S(1,S(1))					
..	..	..	..	...	...	..	..	..	..	..
J	M		S(J)	S(J)	0					
REQUESTS/DURATIONS:										
jobnr.	mode	duration		R					NR	
j	m	d(1)		k(1,1)	K(1,R))				k(1,1)	K(1,NR)
..	..	..	..	..	...	..	..	..	..	..
J	M	d(J)		k(J)	0					
RESOURCE AVAILABILITIES:										
R				NR						
K(1)	K(2)	..	K (R)	K(1)	K(2)	..	K (NR)			

m : Mod indisi
 M : Maksimum Mod Sayısı
 $K(r,m)$: m modunda herbir periyotta r kaynak tipinden maksimum elde edilebilir miktar
 $K(j,R)$: R yenilenebilir kaynak tipinden j faaliyeti için gereksinim duyulan miktar
 $K(j,NR)$: NR yenilenemeyen kaynak tipinden j faaliyeti için gereksinim duyulan miktar

Çizelge C.3 : B-KKPÇP için Bulanık Veri Kümelerinin Yapısı

J	R	F							
K(1)	K(2)	...	...	...	K (R)				
d(1;a1,a2,a3,a4)	k(1,1)	...	...	...	k(1,R)	S(1)	S(1,1)	...	...
...	...	...	...	...	...	...	...	...	...
d(J;a1,a2,a3,a4)	k(J,1)				k(J,R)	S(J)	0		
F	: Bulanık Sayı İndisi								
d(j; a1,a2,a3,a4)	: j işinin bulanık faaliyet süresi								

Çizelge C.4 : B-ÇM-KKPÇP için Bulanık Veri Kümelerinin Yapısı

J	R	NR	F						
RESOURCES									
- renewable		: 2		R					
- nonrenewable		: 2		N					
- doubly constrained		: 0		D					
PRECEDENCE RELATIONS:									
jobnr.		#modes		#successors		successors			
j		m		S(1)		S(1,1) ..		S(1,S(1))	
..	..	..	..	..	...	...	..	..	..
J		M		S(J)		S(J) 0			
REQUESTS/DURATIONS:									
jobnr.		mode		duration		R		NR	
j		m		d(1;a1,a2,a3,a4)		k(1,1,m)		..k(1,R).....K(1,NR)	
..	..	..	..	..	...	..	..	..	..
J		M		d(J;a1,a2,a3,a4)		k(J,NR,M)		S(J) 0	
RESOURCEAVAILABILITIES:									
R						NR			
K(1)	K(2)	..	K (R)	K(1)	K(2)	..	K (NR)		

Çizelge C.5 : TM-KKPC Örnek Veri Yapısı

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

Çizelge C. 6 : ÇM-KKPC Örnek Veri Yapısı

```

*****
file with basedata          : mm2_.bas
initial value random generator: 1424959589
*****
projects                    : 1
jobs (incl. supersource/sink ): 12
horizon                     : 86
RESOURCES
- renewable                 : 2   R
- nonrenewable              : 2   N
- doubly constrained        : 0   D
*****
PROJECT INFORMATION:
pronr.  #jobs rel.date duedate tardcost  MPM-Time
      1    10      0      13         3      13
*****
PRECEDENCE RELATIONS:
jobnr.   #modes  #successors  successors
      1         1         3         2   3   4
      2         3         2         5   6
      3         3         2        10  11
      4         3         1         9
      5         3         2         7   8
      6         3         2        10  11
      7         3         2         9  10
      8         3         1         9
      9         3         1        12
     10         3         1        12

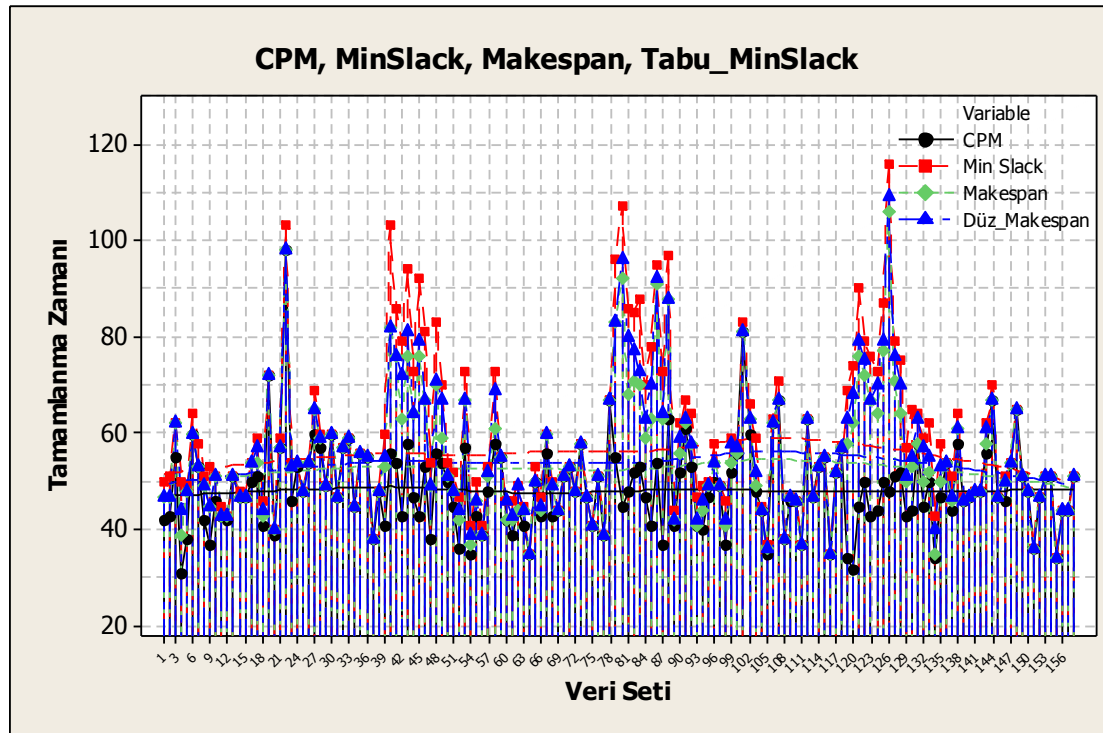
```

11	3	1	12
12	1	0	

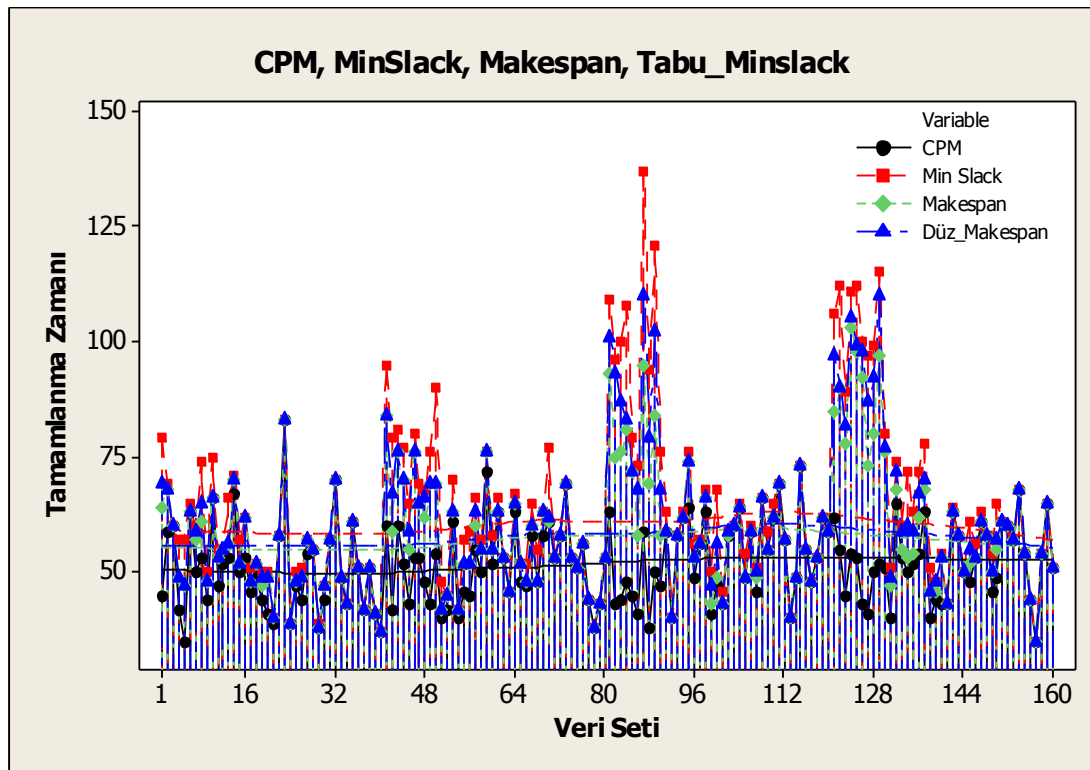
REQUESTS/DURATIONS:			
jobnr.	mode	duration	R 1 R 2 N 1 N 2

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

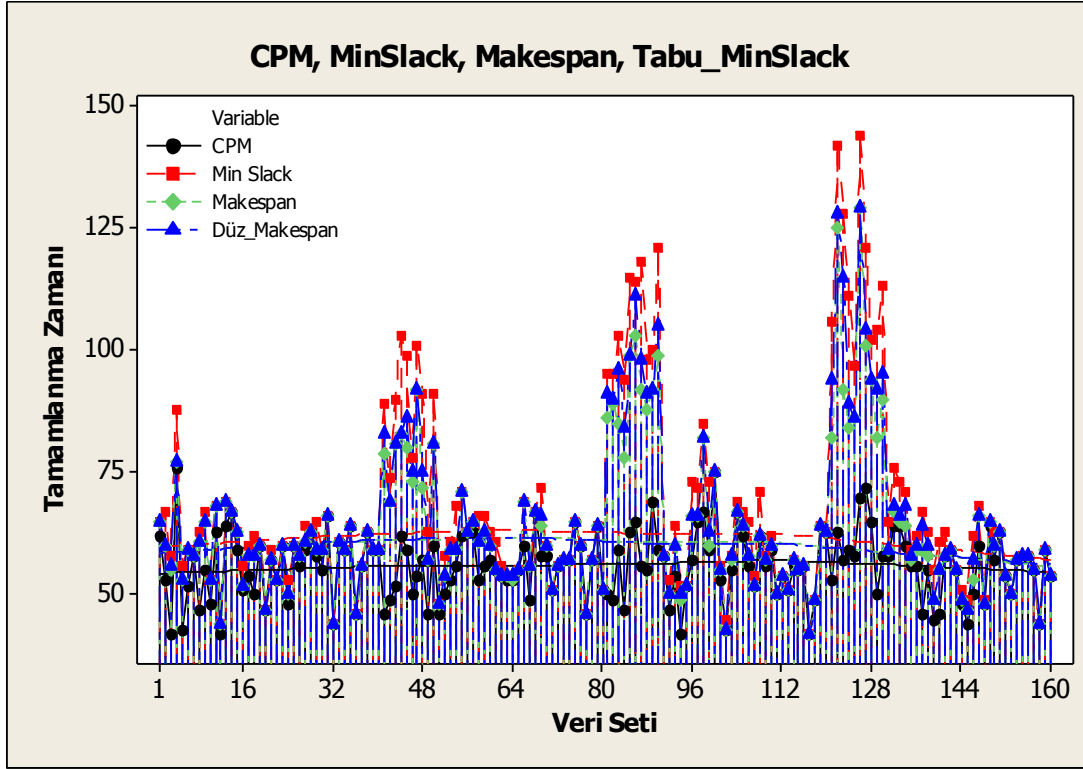
RESOURCEAVAILABILITIES:			
R 1	R 2	N 1	N 2
9	4	29	40



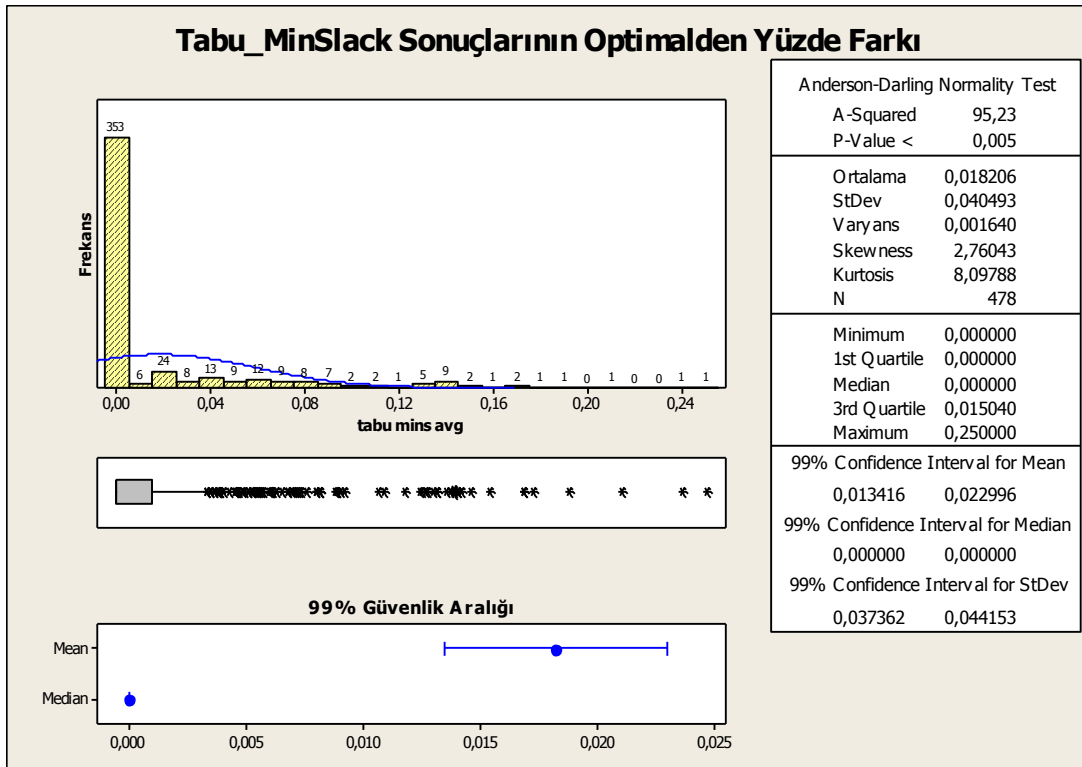
Şekil Ç.1 : J30 Veri Seti için Optimal Sonuçlar Grafiği (1000 İterasyon)



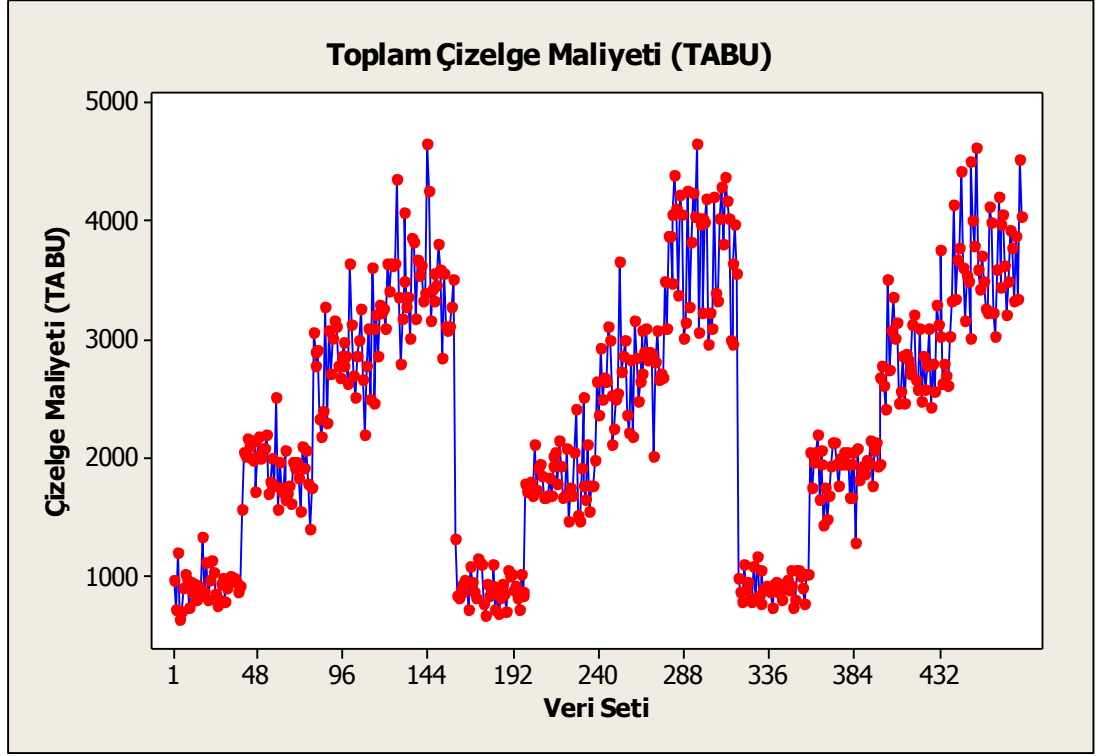
Şekil Ç.2 : J30 Veri Seti için Optimal Sonuçlar Grafiği (1000 İterasyon)



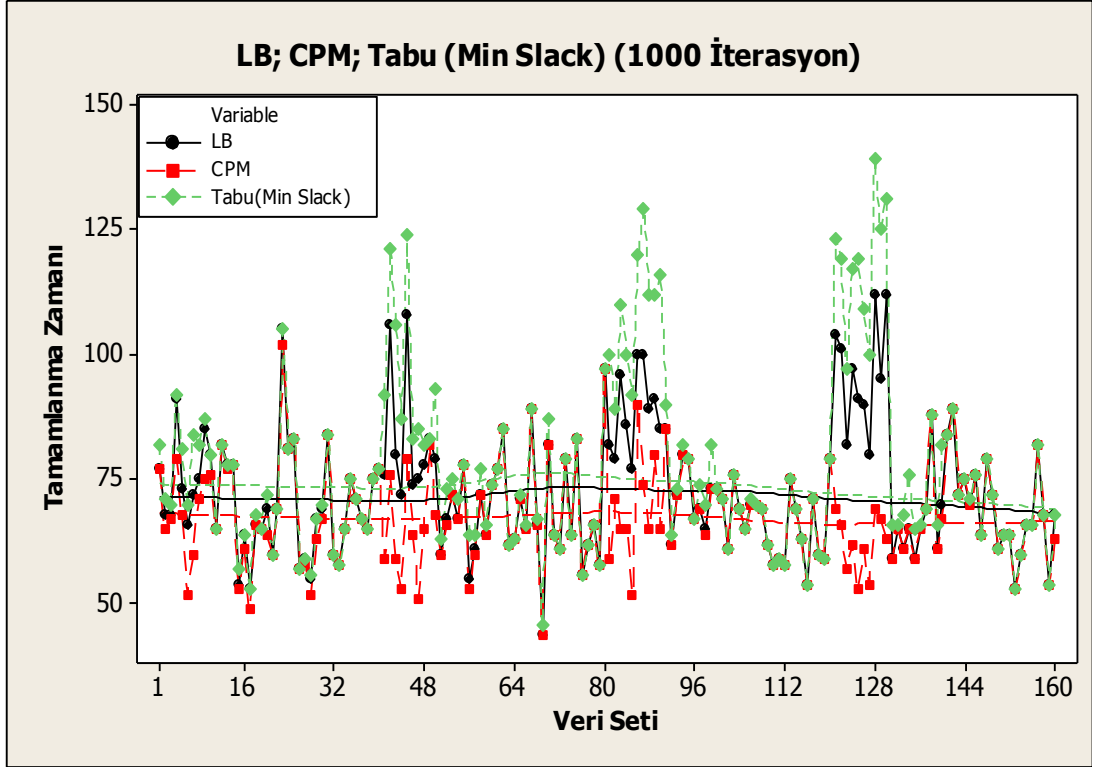
Şekil Ç. 3 : J30 Veri Seti için Optimal Sonuçlar Grafiği (1000 İterasyon)



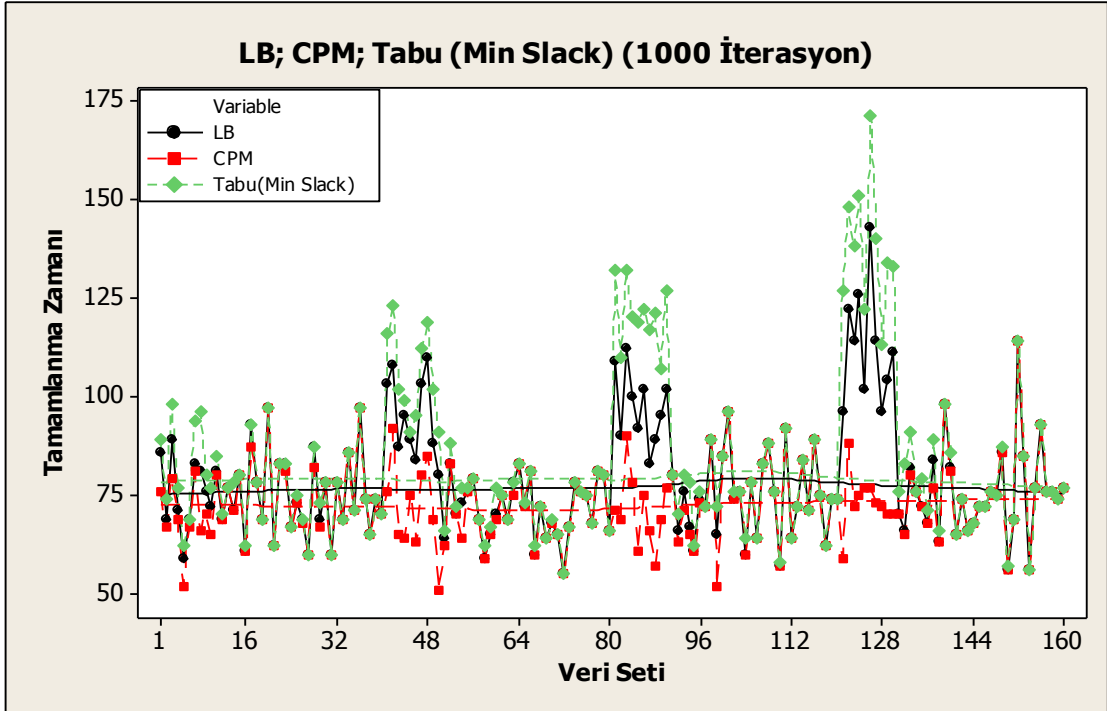
Şekil Ç.4 : J30 Veri Seti için Optimalden Yüzde Farkı Temel İstatistik Grafiği (1000 İterasyon)



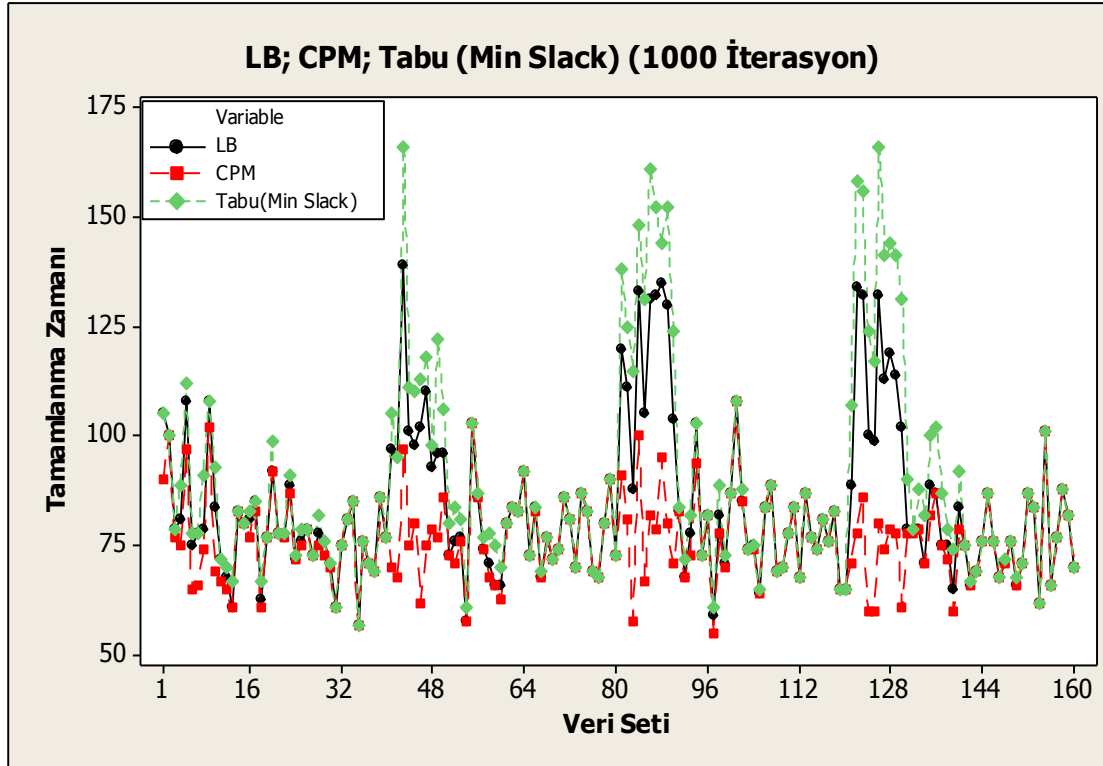
Şekil Ç.5 : J30 Veri Seti için Toplam Çizelge Maliyeti Grafiği (1000 İterasyon)



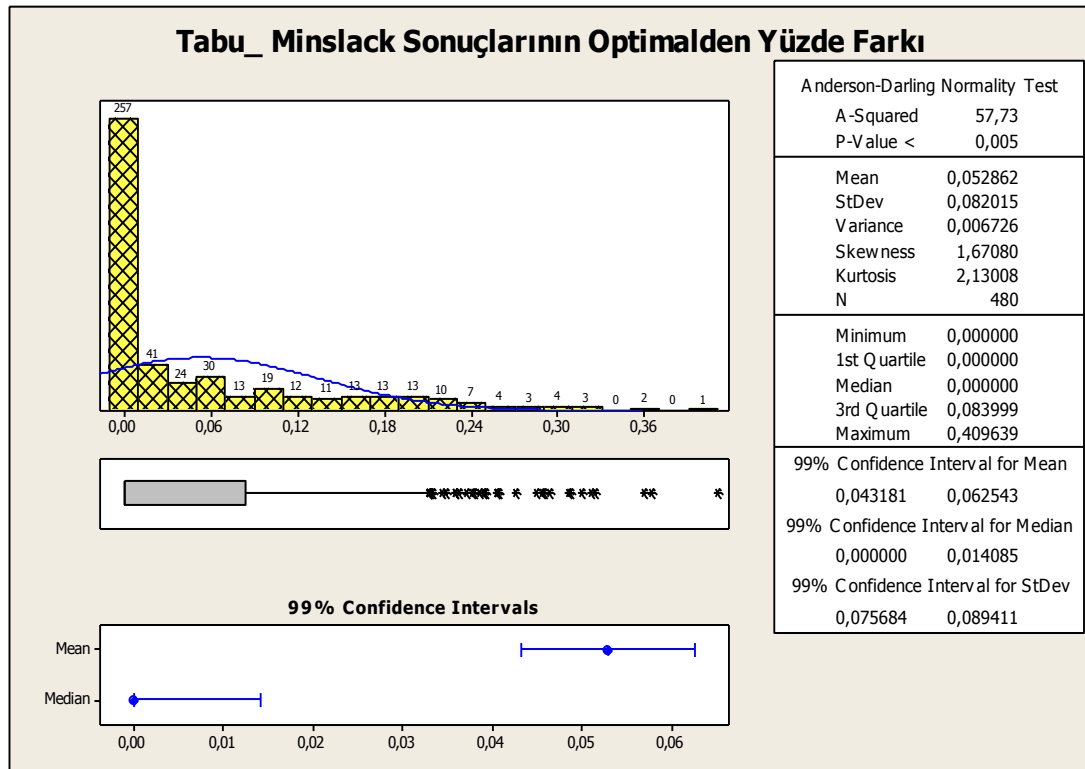
Şekil D.1 : J60 Veri Seti için Optimal Sonuçlar Grafiği



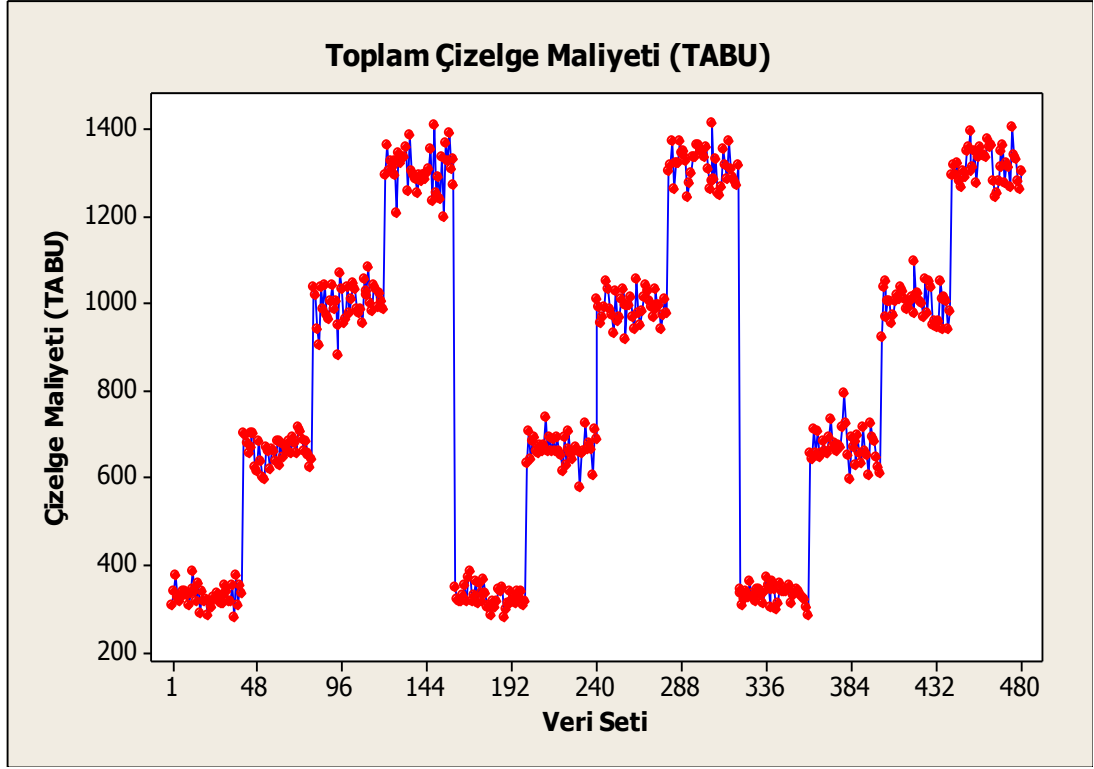
Şekil D.2 : J60 Veri Seti için Optimal Sonuçlar Grafiği



Şekil D.3 : J60 Veri Seti için Optimal Sonuçlar Grafiği

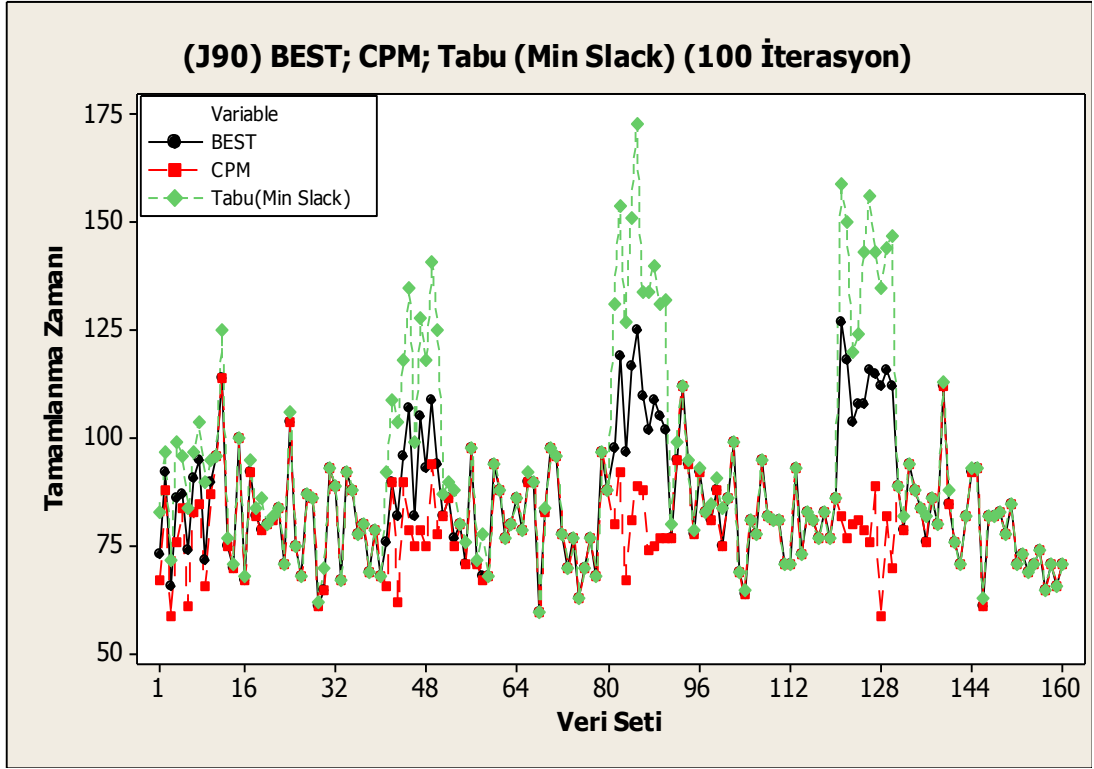


Şekil D.4 : J60 Veri Seti için Optimalden Yüzde Farkı Temel İstatistik Grafiği

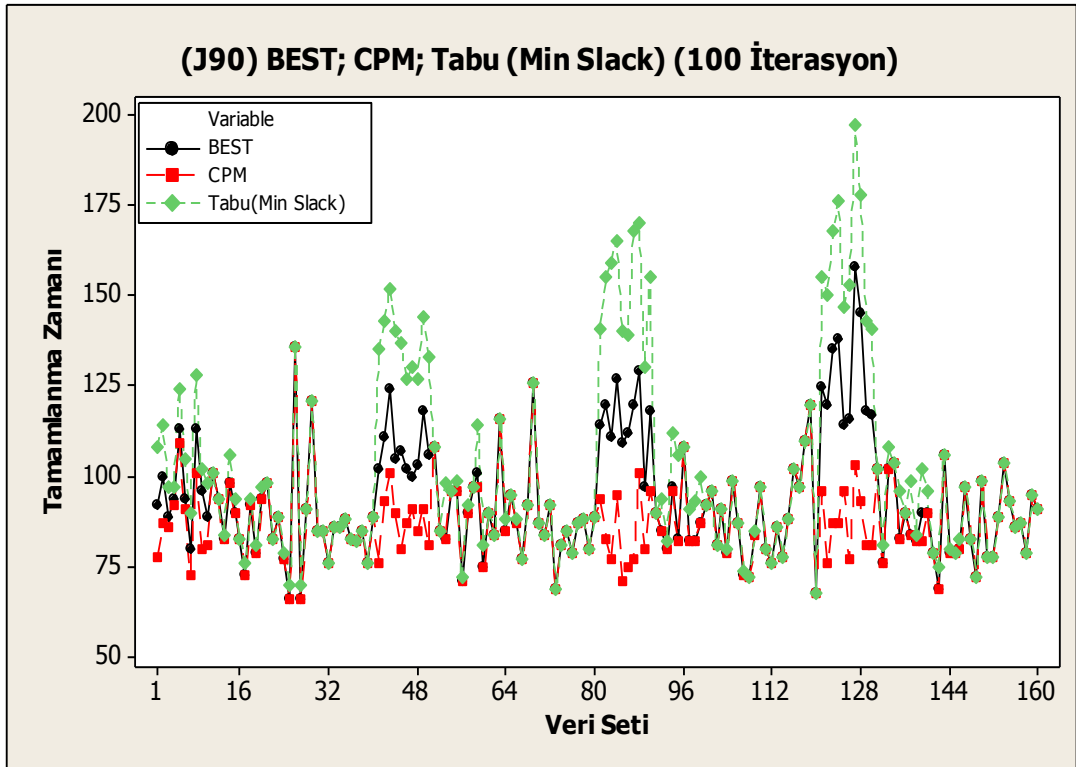


Şekil D.5 : J60 Veri Seti için Toplam Çizelge Maliyeti Grafiği (1000 İterasyon)

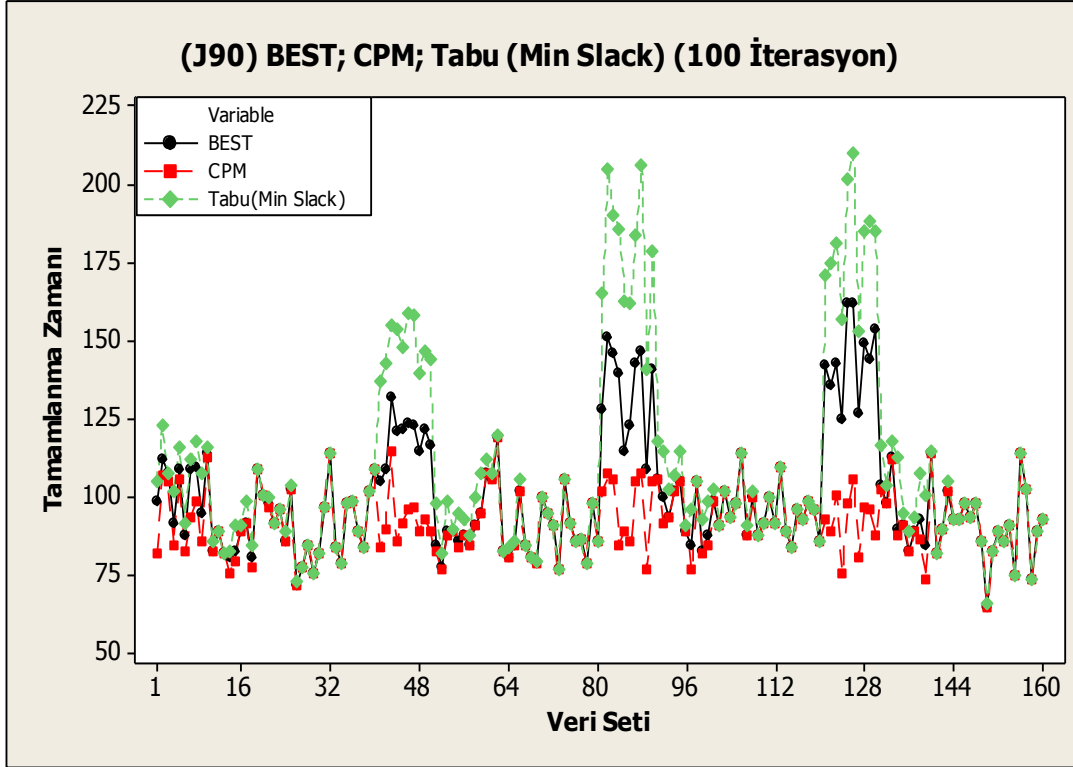
EK-E



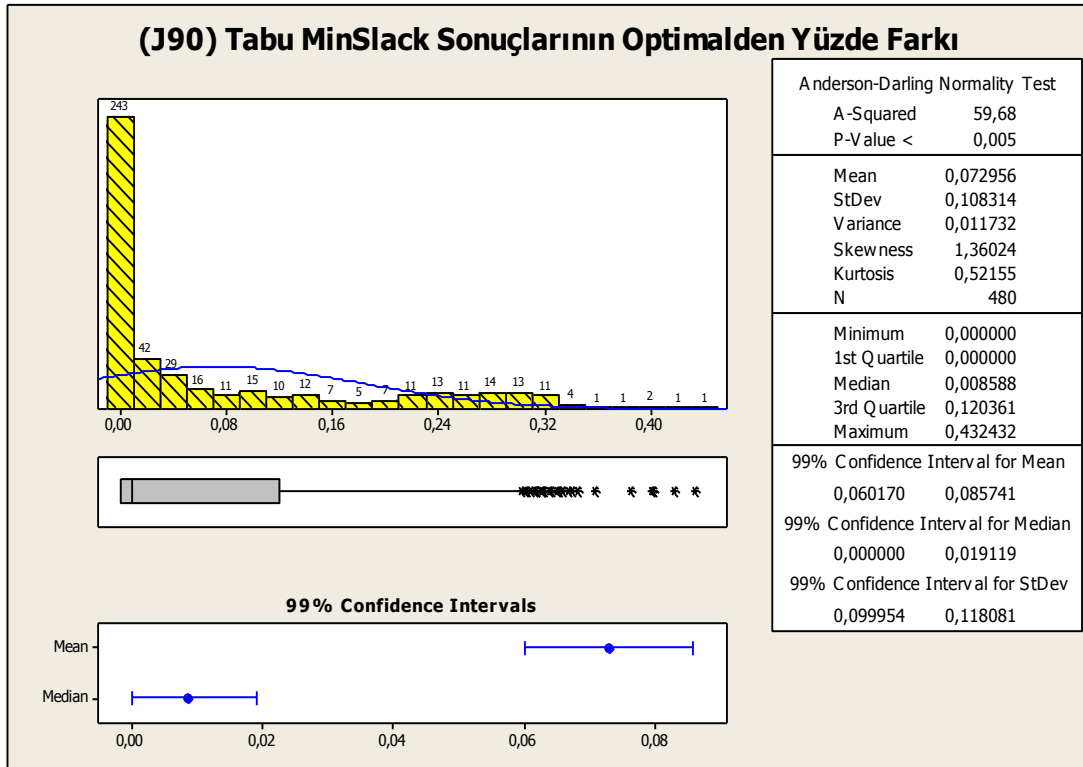
Şekil E.1 : J90 Veri Seti için Optimal Sonuçlar Grafiği



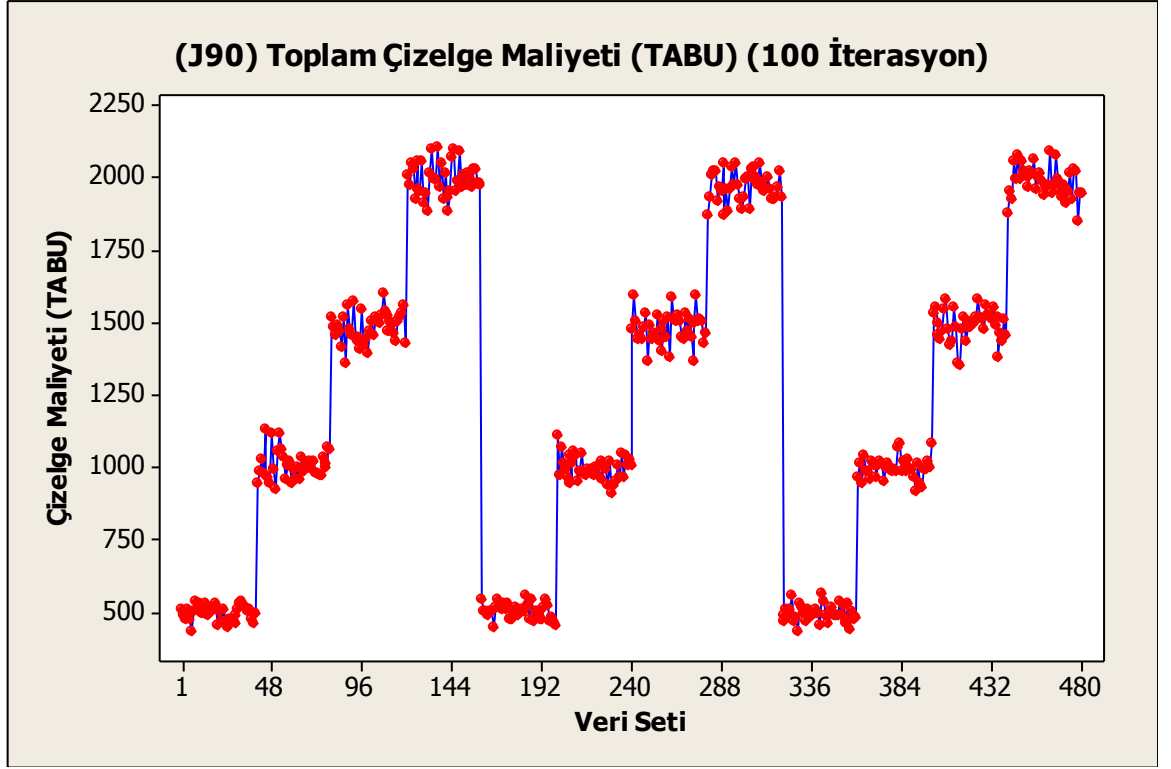
Şekil E.2 : J90 Veri Seti için Optimal Sonuçlar Grafiği



Şekil E.3 : J90 Veri Seti için Optimal Sonuçlar Grafiği

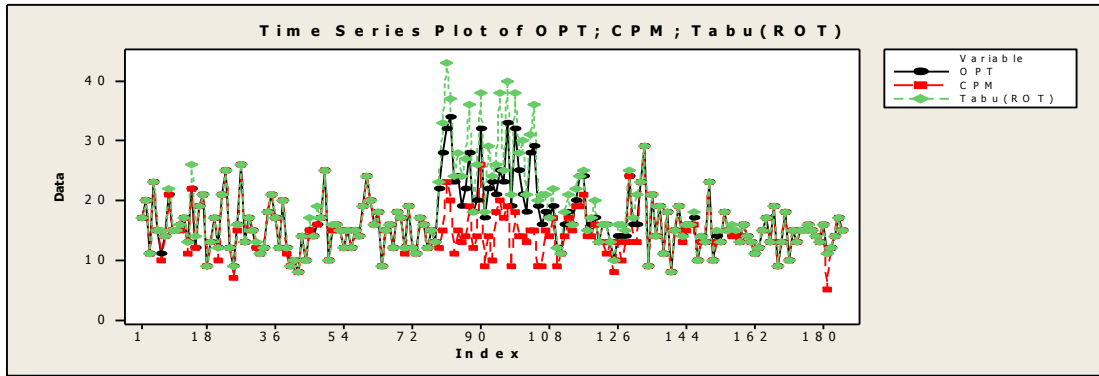


Şekil E.4 : J90 Veri Seti için Optimalden Yüzde Farkı Temel İstatistik Grafiği

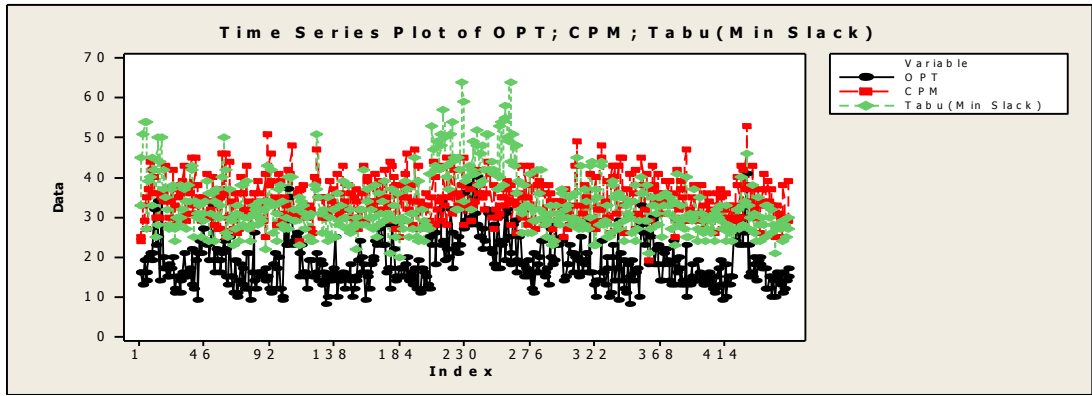


Şekil E.5 : J90 Veri Seti için Toplam Çizelge Maliyeti Grafiği (100 İterasyon)

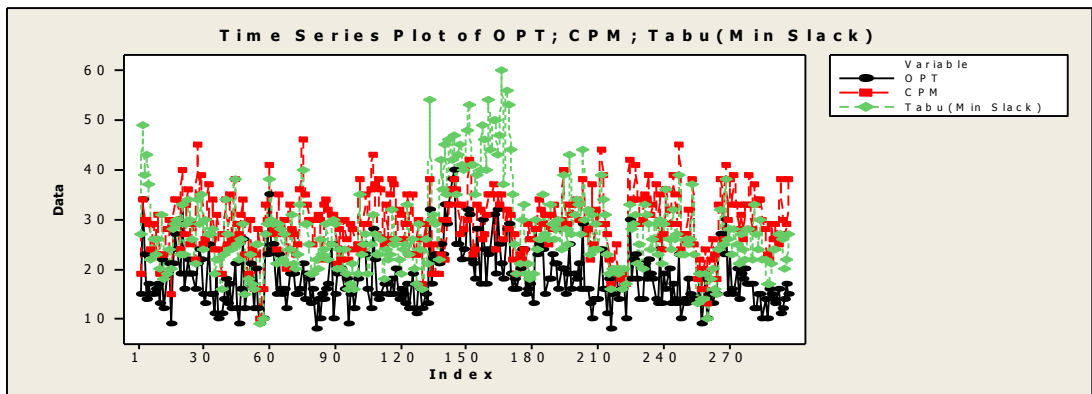
EK-F



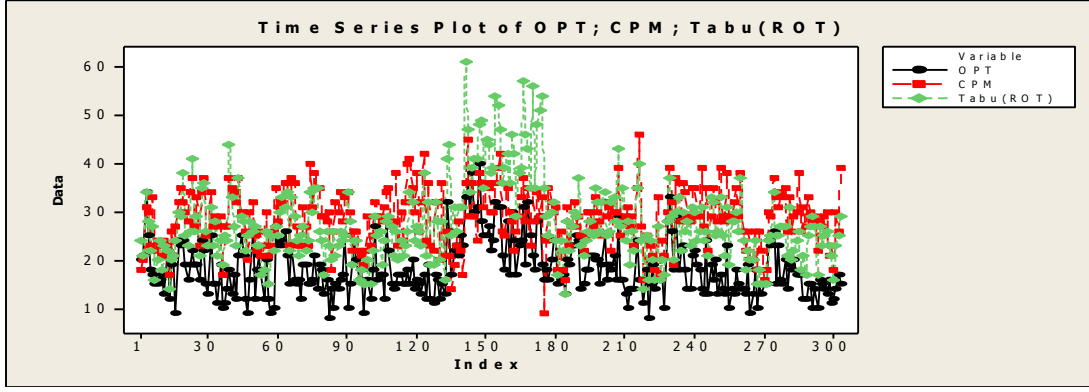
Şekil F. 1 : Mindur Mod Seçim Kuralına göre J10.mm sonuçları



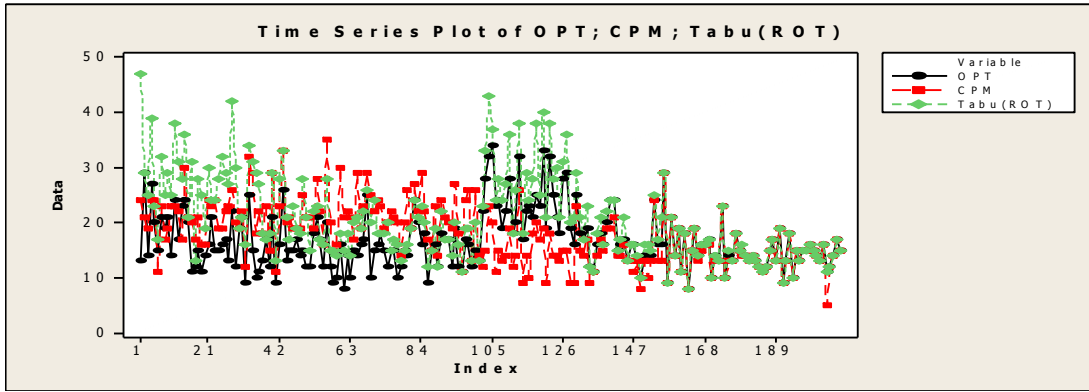
Şekil F. 2 : Maxdur Mod Seçim Kuralına göre J10.mm sonuçları



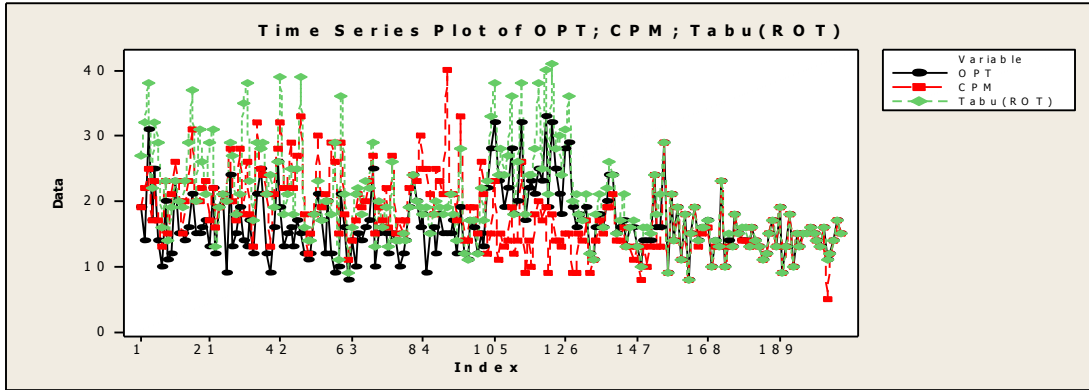
Şekil F. 3 : MinResUsage R3 Mod Seçim Kuralına göre J10.mm sonuçları



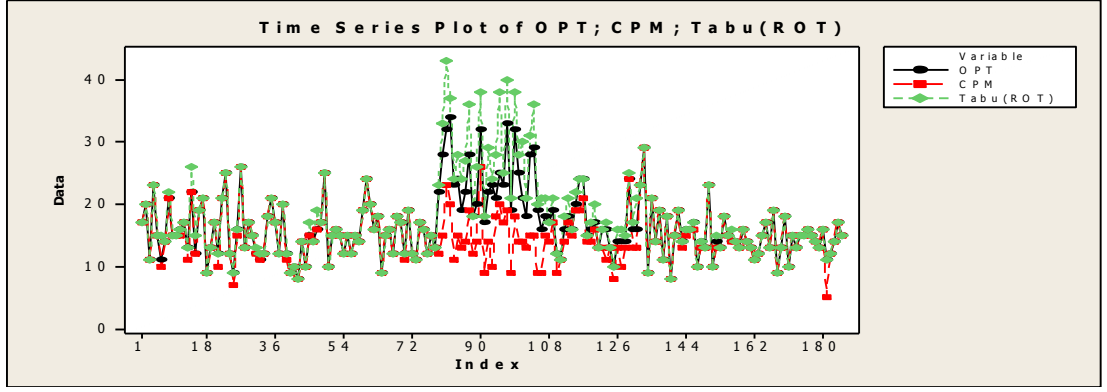
Şekil F. 4 : MinResUsage R4 Mod Seçim Kuralına göre J10.mm sonuçları



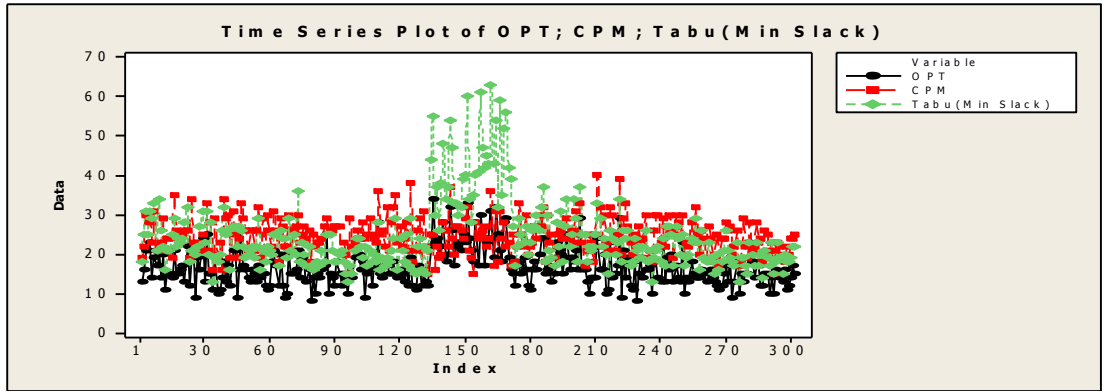
Şekil F. 5 : MaxResUsage R1 Mod Seçim Kuralına göre J10.mm sonuçları



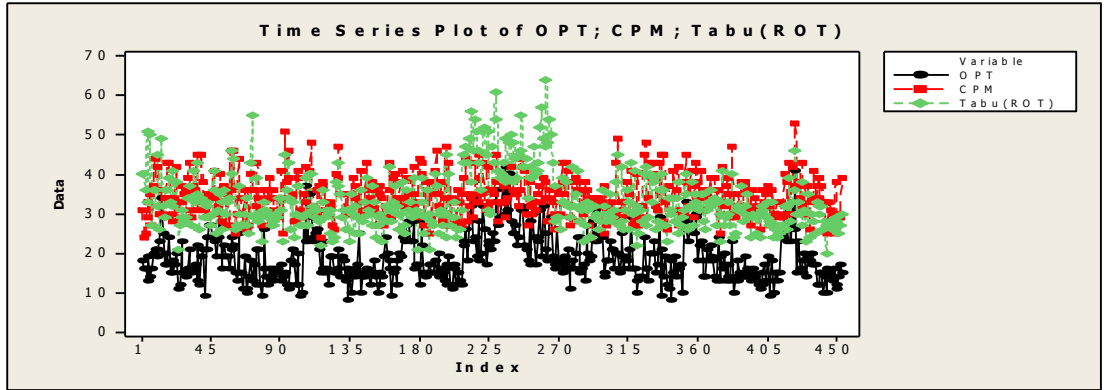
Şekil F. 6 : MaxResUsage R2 Mod Seçim Kuralına göre J10.mm sonuçları



Şekil F. 7 : Mod 1 Seçim Kuralına göre J10.mm sonuçları



Şekil F. 8 : Mod 2 Seçim Kuralına göre J10.mm sonuçları



Şekil F. 9 : Mod 3 Seçim Kuralına göre J10.mm sonuçları

Microsoft Visual C# 2008 Programı ile Geliştirilen Project Scheduling Programı

Bu çalışma kapsamında hazırlanan Microsoft Visual C# 2008’de kodlanan, KKPC, B-KKPC, ÇM-KKPC ve B-ÇM-KKPC problemlerinin çözülebilmeye amacıyla geliştirilen “Project Scheduling” bilgisayar programının çalıştırılma aşamaları açıklanacaktır.

CPM_Project.exe dosyası çalıştırılır.Ekrana Şekil Ğ.1’deki arayüz görünümü açılır.

Ara yüz görünüşü çeşitli menülerden, alt tablardan ve işlem yapılmak istenen butonlardan oluşmaktadır.

Dosya Menüsü : deterministik ve bulanık (.RCP ve .FZY) uzantılı yeni problemler oluşturabilmenize olanak tanıyacak şekilde tasarlanmıştır.

Dil Menüsü : ise ingilizce türkçe dil seçeneği mevcuttur.

Bulanıklaştırıcı Tıkı : Yüklenen deterministik problemini bulanık hale dönüştürür.

Sonuçları Kaydet Tıkı : Dosyayı yüklemeyen önce yükleyeceğiniz problemin sonuçlarını text dosya tipinde görmek isterseniz yapmanız gereken bir tuştur. Dosyayı kaydet tuşu ile problemin sonuçlarının izlenebilmesi amacıyla programın sonuçlarının kaydedilmesi sağlanmaktadır.

Dosya Yükle : Çalıştırmak istediğiniz dosyanın text tipi dosyasını açmak için kullanılır. Eğer birden fazla dosya seçimi yapıldığı takdirde her bir problem otomatik olarak çözüm sürecine alınacak ve sonuçları text olarak ve excel formatında kaydedilecektir. Şu dosya uzantılı dosyalar olmalıdır. (.RCP, .FZY ve .MM) uzantılı dosyalar çalıştırılabilir. Ve bu text dosyaların formatı daha önce açıklanan biçimde olmalıdır. Bunun olmaması durumunda hata verecektir. .FZY uzantılı bir dosyanın açılması durumunda Şekil Ğ.2’deki gibi ekran görünümü gelecektir. Bu ekranda herhangi bir işlem yapılmamaktadır. Sadece problemin grafiği ve temel bilgileri gösterilmesi sağlanmaktadır. Kaç faaliyetin oluştuğu, kaç kaynaktan oluştuğu, maksimum kaynak miktarı ekranda görülebilir.

CPM tuşuna basılarak; projenin kaynak kısıtları dikkate alınmaksızın proje çizelge sırası ile proje tamamlanma süresi bulunur. Bu işlem yapıldığında gerekli bilgiler CPM tabında görülebileceği gibi şebeke diyagramı üzerinde faaliyet düğümleri üzerine mouse ile gelindiği takdirde o faaliyete ait hesaplanan değerler şebeke diyagramının hemen üst kısmında belirecektir.

Min Slack ve ROT Tuşları: Başlangıç çözümü bulmadan önce;

Rot değerleri için, hangi kaynak tipine göre hesaplama yapılacağı seçilmelidir.

Tabu için Maksimum iterasyon sayısı yazılmalıdır.

Hangi öncelik kuralına göre hesaplama yaptırılacağı seçilmelidir. Rot, veya Minslack öncelik kuralından biri seçilecektir.

Seçilen başlangıç çözümünden sonra sonuçlar ve çizelge MinSlack ve ROT tablaları ile sonuç dosyasına yazdırılacaktır.

Başlangıç Çözümü atama tuşuna basılarak, seçilen sezgisel yaklaşıma göre kaynak tahsisi gerçekleştirilir.

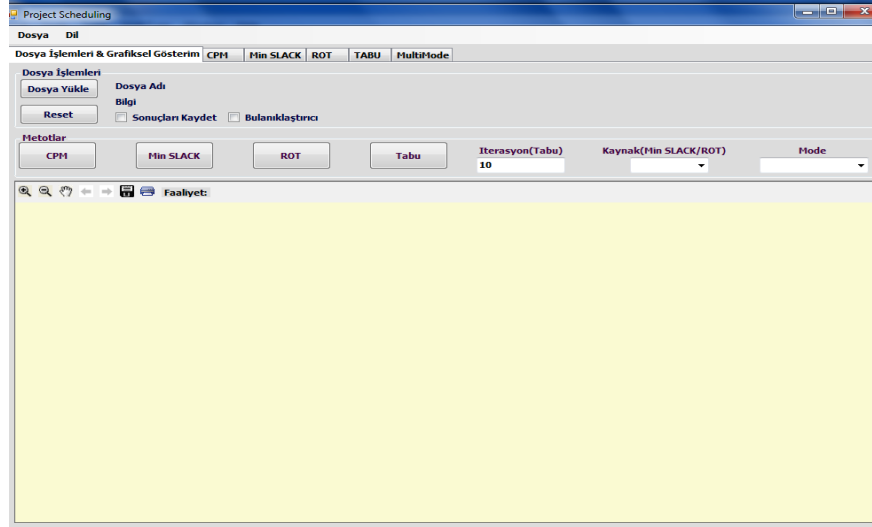
Seçim kriterlerine göre başlangıç çözüm için Rot, Actim ve Minslack değerleri, uygun çizelge sırası, proje tamamlanma süresi ve zaman ufku boyunca kullanılan kaynak miktarları bulunur.

Tabu Tuşu ile seçilen başlangıç çözüme göre ikili yer değişimlerin yapıldığı algoritma sonucu optimal veya optimale yakın proje çizelge sırası ve proje tamamlanma süresi bulunacaktır.

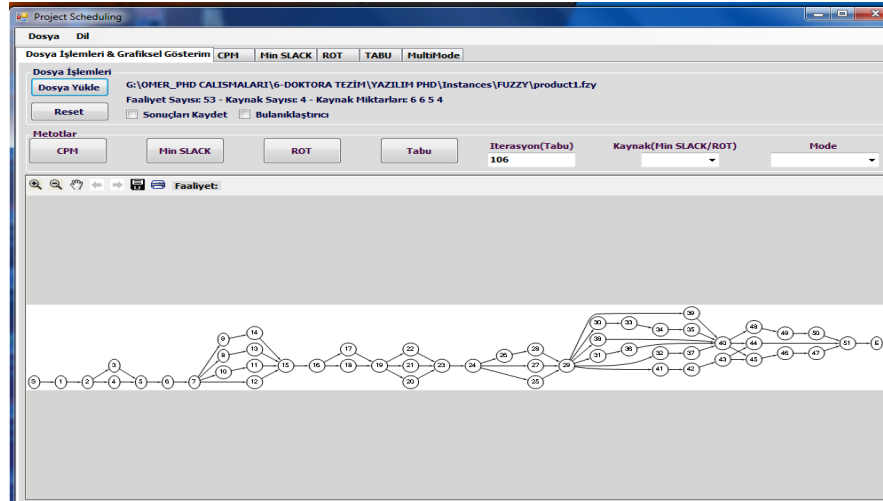
Mode Kombo boxı: Eğer dosya yükle ile seçilen .MM uzantılı dosya ise ÇM-KKPÇ problemi yüklenmiş olur. Yukarıdaki işlemlerin yapılabilmesi için öncelikle Mode kombo box tuşuyla bir kural seçilmelidir. Eğer Min Res ve Max Res kuralları seçilmek istenirse öncelikle bir kaynak tipi seçilmelidir. Mode Seçim kuralı seçildikten sonra programın dipar fonksiyonlarının herbiri sırasıyla yapılabilir.

Programın herbir tabına ait ekran görünümüleri EK-8’de sırasıyla gösterilecektir.

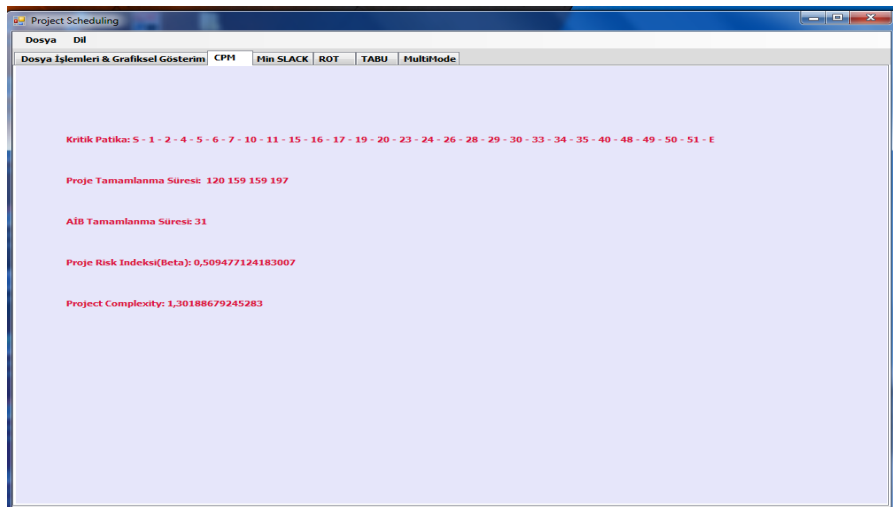
EK-Ğ



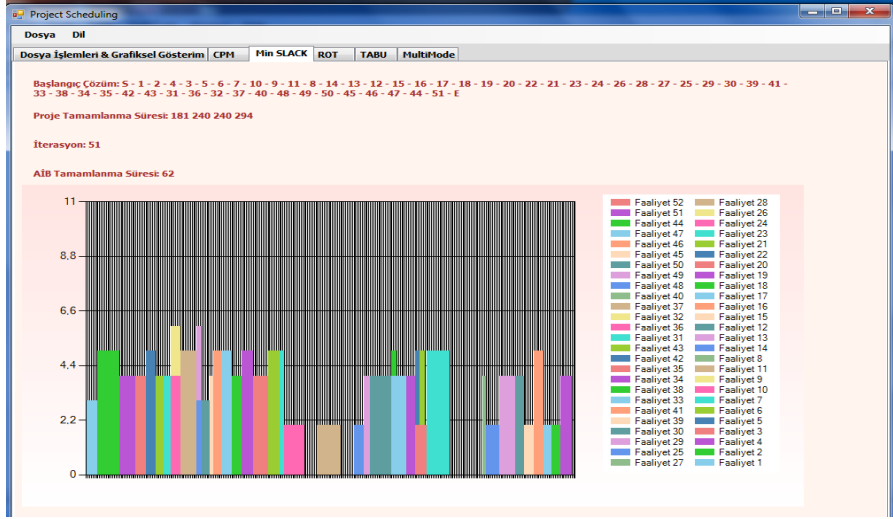
Şekil Ğ. 1 : Project Scheduling Progamının açılış Görünümü



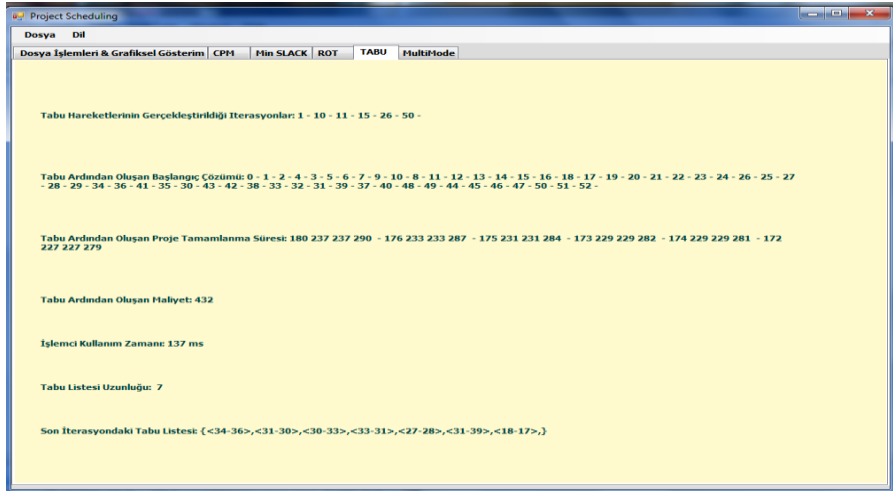
Şekil Ğ. 2 : Dosya Yükleme Görünümü



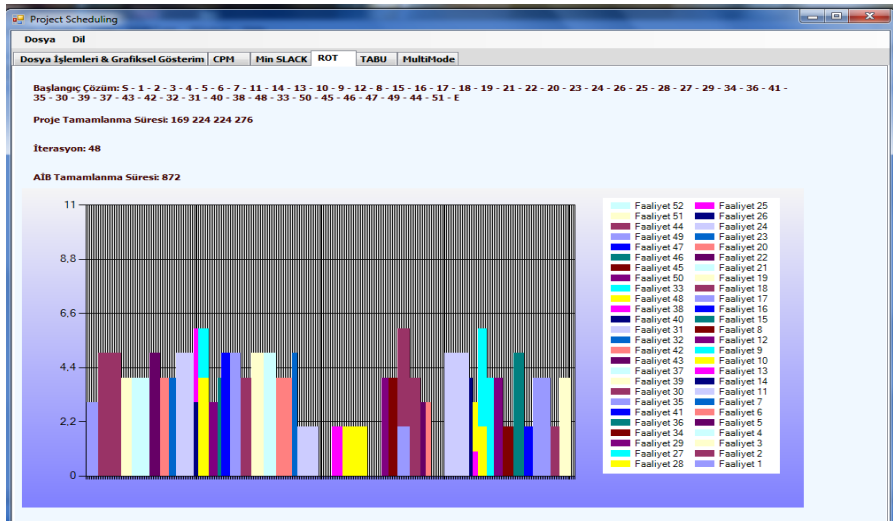
Şekil Ğ. 3 : KYM Tabı Görünümü



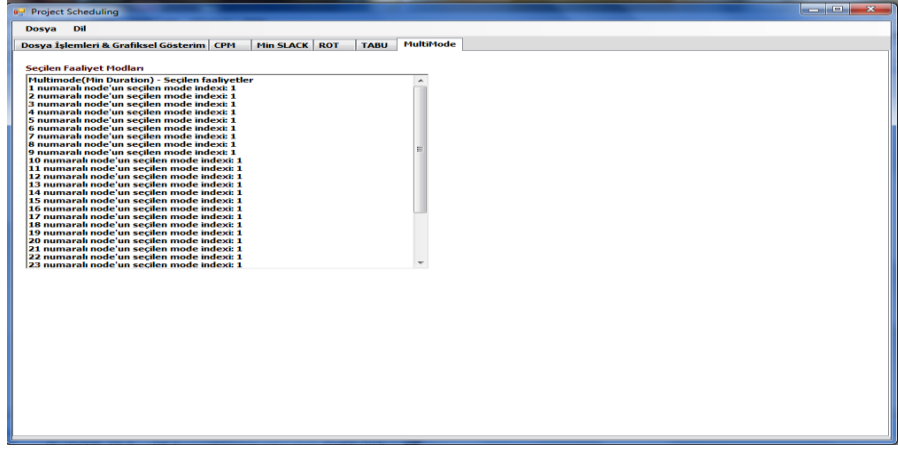
Şekil Ğ. 4 : Min Slack Tabı Görünümü



Şekil Ğ. 5 : Tabu Tabı Görünümü



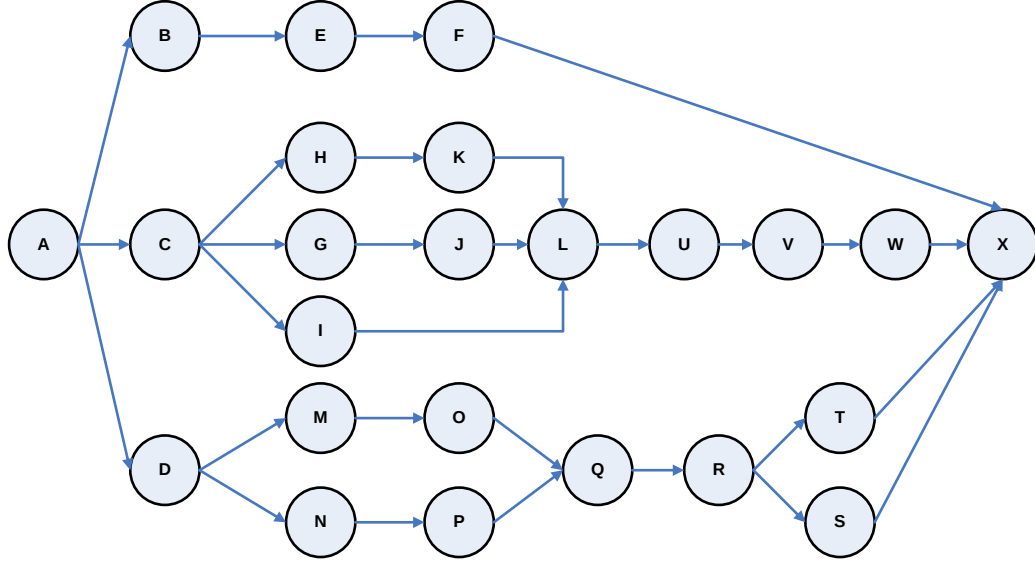
Şekil Ğ. 6 : ROT Tabı Görünümü



Şekil 7 : MultiMode Tabı Görünümü

EK-H

Bulanık B-KYM için Örnek Problem



Şekil H.1 : Uçak Yakıt Gaz Turbin Motoru Bakım Şebeke Diyagramı.

Çizelge H.1 : Uçak Yakıt Gaz Turbin Motoru Bakım Faaliyetleri Verileri

Faaliyet Adı	Faaliyetlerin Tanımı	Crisp Duration Man-hours	Faaliyetlerin Önceli	Bulanık Faaliyet Süreleri (Man-Hours)	FAT
A	Check stand by pump	32	-	Around 32 man-hours	(28,32,32,36)
B	Calibrate all ganges	80	A	Around 80 man-hours	(72,80,80,88)
C	Dismantle pump cover and remove rotor	8	A	Around 8 man-hours	(6,8,8,10)
D	Dismantle turbine cover and remove rotor	16	A	Around 16 man-hours	(12,16,16,20)
E	Clean all ganges and line	32	B	Around 32 man-hours	(28,32,32,36)
F	Replace ganges	16	E	Around 16 man-hours	(12,16,16,20)
G	Repair lubrication system	32	C	Around between 28 and 40 man-hours	(28,32,36,40)
H	Rebuild impeller	288	C	Around between 256 and 352 man-hours	(256,288,320,352)
I	Clean pump casting	32	C	Around 32 man-hours	(28,32,32,36)
J	Fix pump bearings	16	G	Around 16 man-hours	(12,16,16,20)
K	Balance Impeller	32	H	Around 32 man-hours	(28,32,32,36)
L	Reinstall Impeller	16	I,J,K	Around 16 man-hours	(12,16,16,20)
M	Rebuild turbine rotor	320	D	Around between 260 and 400 man-hours	(280,320,360,400)
N	Check turbine bearings	16	D	Around 16 man-hours	(14,16,16,18)
O	Balance turbine rotor	64	M	Around 64 man-hours	(56,64,64,72)
P	Fix turbine bearings	16	N	Around 16 man-hours	(12,16,16,20)
Q	Fix turbine rotor	24	O,P	Around between 16 and 28 man-hours	(16,20,24,28)
R	Fix turbine cover	24	Q	Around between 20 and 32 man-hours	(20,24,28,32)
S	Test components	16	R	Around between 12 and 18 man-hours	(12,14,16,18)
T	Check clearance	16	R	Around 16 man-hours	(14,16,16,18)
U	Fix pump bearings	16	L	Around 16 man-hours	(12,16,16,20)
V	Fix pump cover	24	U	Around between 18 and 24 man-hours	(18,20,22,24)
W	Install shaft packing	16	V	Around 16 man-hours	(14,16,16,18)
X	Final test	64	S,T,W,F	Around between 48 and 72 man-hours	(48,56,64,72)

Çizelge H.2 : Uçak Yakıt Gaz Turbin Motoru Bakım Projesi B-KYM

Faaliyet No.	Faaliyet Adı	FEST	FECT	FLST	FLCT	FTS(LCT-ECT)
1	A	(0 0 0 0)	(28 32 32 36)	(-72 -24 80 244)	(-36 8 112 272)	(-72 -24 80 244)
2	B	(28 32 32 36)	(100 112 112 124)	(222 340 396 494)	(310 420 476 566)	(186 308 364 466)
3	C	(28 32 32 36)	(34 40 40 46)	(-36 110 200 326)	(-26 118 208 332)	(-72 78 168 298)
4	D	(28 32 32 36)	(40 48 48 56)	(-20 24 128 288)	(0 40 144 300)	(-56 -8 96 260)
5	E	(100 112 112 124)	(128 144 144 160)	(362 468 524 610)	(398 500 556 638)	(238 356 412 510)
6	F	(128 144 144 160)	(140 160 160 180)	(414 516 572 654)	(434 532 588 666)	(254 372 428 526)
7	G	(34 40 40 46)	(62 72 76 86)	(272 390 456 554)	(312 426 488 582)	(226 350 416 520)
8	H	(34 40 40 46)	(290 328 360 398)	(-368 -194 -72 82)	(-16 126 216 338)	(-414 -234 -112 48)
9	I	(34 40 40 46)	(58 72 72 82)	(300 414 472 574)	(336 446 504 598)	(254 374 432 540)
10	J	(62 72 76 86)	(74 88 92 106)	(332 446 504 598)	(352 462 520 610)	(246 370 432 536)
11	K	(290 328 360 398)	(318 360 392 434)	(300 414 472 566)	(336 446 504 594)	(-98 -54 144 276)
12	L	(318 360 392 434)	(330 376 408 454)	(352 462 520 610)	(372 478 536 622)	(-82 70 160 292)
13	M	(40 48 48 56)	(300 368 408 456)	(-496 -304 -160 52)	(-96 56 160 312)	(-552 -352 -208 12)
14	N	(40 48 48 56)	(50 64 64 74)	(320 432 496 596)	(338 448 512 606)	(264 384 448 556)
15	O	(300 368 408 456)	(356 432 472 528)	(232 352 416 516)	(304 416 480 572)	(-224 -56 48 216)
16	P	(50 64 64 74)	(62 80 80 94)	(336 448 512 604)	(356 464 528 616)	(262 384 448 554)
17	Q	(356 432 472 528)	(372 452 496 556)	(348 456 524 612)	(376 480 544 628)	(-180 -16 92 256)
18	R	(372 452 496 556)	(392 476 524 588)	(372 476 540 624)	(404 504 564 644)	(-184 -20 88 252)
19	S	(392 476 524 588)	(404 490 540 606)	(418 516 576 654)	(436 532 590 666)	(-170 -8 100 262)
20	T	(392 476 524 588)	(406 492 540 606)	(418 516 572 650)	(436 532 588 664)	(-170 -8 96 258)
21	U	(330 376 408 454)	(342 392 424 474)	(372 478 536 622)	(392 494 552 634)	(-82 70 160 292)
22	V	(342 392 424 474)	(360 412 446 498)	(388 488 548 628)	(412 510 568 646)	(-86 64 156 286)
23	W	(360 412 446 498)	(374 428 462 516)	(418 516 572 650)	(436 532 588 664)	(-80 70 160 290)
24	X	(406 492 540 606)	(454 548 604 678)	(382 484 548 630)	(454 548 604 678)	(-224 -56 56 224)

Çizelge H.3 : Uçak Yakıt Gaz Turbin Motoru Bakım Projesi Kritik Faaliyetleri.

Sıra No	Proje Yolları	RFCPM	RFCPM	Total FTS				Sıra
1	A-B-E-F-X	$R_{FCPM} P_1$	0.5992	258	860	1276	1878	7
2	A-C-H-K-L-U-V-W-X	$R_{FCPM} P_2$	0.4857	-1206	42	952	2200	3
3	A-C-G-J-L-U-V-W-X	$R_{FCPM} P_3$	0.6315	-550	642	1496	2688	8
4	A-C-I-L-U-V-W-X	$R_{FCPM} P_4$	0.5678	-740	340	1120	2200	6
5	A-D-M-O-Q-R-T-X	$R_{FCPM} P_5$	0.3864	-1632	-448	448	1632	1
6	A-D-M-O-Q-R-S-X	$R_{FCPM} P_6$	0.3867	-1632	-448	450	1634	2
7	A-D-N-P-Q-R-T-X	$R_{FCPM} P_7$	0.5757	-764	336	1152	2252	4
8	A-D-N-P-Q-R-S-X	$R_{FCPM} P_8$	0.5760	-764	336	1154	2254	5

EK-I

Çizelge I.1 : KKPC Problemleri için OPL-CPLEX Kodu

```
using CP;
int NbTasks = ...;
int NbRsrcs = ...;
range RsrcIds = 0..NbRsrcs-1;
int Capacity[r in RsrcIds] = ...;
tuple Task {
key int id;
int pt;
int dmds[RsrcIds];
{int} succs;
}
{Task} Tasks = ...;
dvar interval itvs[t in Tasks] size t.pt;
cumulFunction rsrcUsage[r in RsrcIds] =
sum (t in Tasks: t.dmds[r]>0) pulse(itvs[t], t.dmds[r]);
execute {
cp.param.FailLimit = 10000;
}
minimize max(t in Tasks) endOf(itvs[t]);
subject to {
forall (r in RsrcIds)
rsrcUsage[r] <= Capacity[r];
forall (t1 in Tasks, t2id in t1.succs)
endBeforeStart(itvs[t1], itvs[<t2id>]);
}
execute {
for (var t in Tasks) {
writeln("Task " + t.id + " starts at " + itvs[t].start);
}
}
```


Çizelge İ.1 : ÇM-KKPC Problemleri için OPL-CPLEX Kodu

```

using CP;
int NbTasks = ...;
int NbRenewableRsrcs = ...;
int NbNonRenewableRsrcs = ...;
range RenewableRsrcIds = 0..NbRenewableRsrcs-1;
range NonRenewableRsrcIds = 0..NbNonRenewableRsrcs-1;
int CapRenewableRsrc [RenewableRsrcIds] = ...;
int CapNonRenewableRsrc [NonRenewableRsrcIds] = ...;
tuple Task {
key int id;
int nmodes;
{int} succs;
}
{Task} Tasks = ...;
tuple Mode {
int taskId;
int id;
int pt;
int dmdRenewable [RenewableRsrcIds];
int dmdNonRenewable[NonRenewableRsrcIds];
}
{Mode} Modes = ...;
dvar interval task[t in Tasks];
dvar interval mode[m in Modes] optional size m.pt;
cumulFunction renewableRsrcUsage[r in RenewableRsrcIds] =
sum (m in Modes: m.dmdRenewable[r]>0) pulse(mode[m],
m.dmdRenewable[r]);
dexpr int nonRenewableRsrcUsage[r in NonRenewableRsrcIds] =
sum (m in Modes: m.dmdNonRenewable[r]>0) m.dmdNonRenewable[r] *
presenceOf(mode[m]);
execute {
cp.param.FailLimit = 10000;
}
minimize max(t in Tasks) endOf(task[t]);
subject to {
forall (t in Tasks)
alternative(task[t], all(m in Modes: m.taskId==t.id) mode[m]);
forall (r in RenewableRsrcIds)
renewableRsrcUsage[r] <= CapRenewableRsrc[r];
forall (r in NonRenewableRsrcIds)
nonRenewableRsrcUsage[r] <= CapNonRenewableRsrc[r];
forall (t1 in Tasks, t2id in t1.succs)
endBeforeStart(task[t1], task[<t2id>]);
}
execute {
for (var m in Modes) {
if (mode[m].present)
writeln("Task " + m.taskId + " with mode " + m.id + " starting at "
+
mode[m].start);
}
}
}

```


ÖZGEÇMİŞ

Ad Soyad: Ömer ATLI
Doğum Yeri ve Tarihi: Bandırma 24.07.1978
Adres: Hava Kuvvetleri Komutanlığı
E-Posta: atliomer@gmail.com
Lisans: Uludağ Üniversitesi Endüstri Mühendisliği Bölümü
Yüksek Lisans: Uludağ Üniversitesi Endüstri Mühendisliği Bölümü

Mesleki Deneyim ve Ödüller:

İlk, orta ve lise öğrenimi Balıkesir ilinin Edremit ilçesinde tamamladı. Lisans eğitimini takiben Bursa organize pilot sanayiide Makine İmalat sektörünün önder kuruluşlarından olan özel bir firmada sistem kurucusu ve üretim programcısı olarak, 2001 Aralık ayında Hava Kuvvetleri Komutanlığında teğmen olarak nasbedilinceye kadar görev yaptı. 2002 yılında Eskişehir 1’inci HİBM K.lığında ilk görevine atandı. 2010 yılı genel atamalarıyla birlikte Hava Kuvvetleri Komutanlığı Lojistik Başkanlığına atandı. Halen Hv.K.K.lığı karargahındaki görevine devam etmekte olup, evlidir.

Yayın ve Patent Listesi:

TEZDEN TÜRETİLEN YAYINLAR/SUNUMLAR

- Atlı Ö.** (2011). Tabu Search and an Exact Algorithm for the Solutions of Resource-Constrained Project Scheduling Problems, *International Journal of Computational Intelligence Systems* Vol 4, Issue 2, P. 255 – 267.
- Atlı Ö., Kahraman C.** (basımda). The Minslack and Kangaroo Algorithm Heuristic for Fuzzy Resource-Constrained Project Scheduling Problems, *Journal of Multiple-Valued Logic and Soft Computing*.
- Atlı Ö., Kahraman C.** (basımda). Bulanık Kritik Yol Analizi, *Sigma Dergisi*.
- Atlı Ö., Kahraman C.** (2012). Aircraft Maintenance Planning Using Fuzzy Critical Path Analysis” *International Journal of Computational Intelligence Systems*, Vol.5 , No. 3, P. 553–567.
- Atlı Ö., Kahraman C.** (basımda). Fuzzy Resource Constrained Project Scheduling Using Taboo Search Algorithm *International Journal of Intelligent Systems*.