

**MALİYET TABANLI PARALEL MONTAJ HATTI
DENGEME PROBLEMLERİ: YENİ MODELLER**

TUSAN DERYA

**DOKTORA TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**EKİM 2012
ANKARA**

Tusan DERYA tarafından hazırlanan “MALİYET TABANLI PARALEL MONTAJ HATTI Dengeleme Problemleri: Yeni Modeller” adlı bu tezin Doktora tezi olarak uygun olduğunu onaylarım.

Prof. Dr. Hadi GÖKÇEN
Tez Danışmanı, Endüstri Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından oy birliği ile Endüstri Mühendisliği Anabilim Dalında Doktora tezi olarak kabul edilmiştir.

Prof. Dr. Orhan TÜRKBEY
Endüstri Mühendisliği Anabilim Dalı, G.Ü.

Prof. Dr. Hadi GÖKÇEN
Endüstri Mühendisliği Anabilim Dalı, G.Ü.

Prof. Dr. Bilal TOKLU
Endüstri Mühendisliği Anabilim Dalı, G.Ü.

Doç. Dr. Yakup KARA
Endüstri Mühendisliği Anabilim Dalı, Selçuk Üniv.

Yrd. Doç. Dr. Uğur Özcan
Endüstri Mühendisliği Anabilim Dalı, G.Ü.

Tarih: 18 / 10 / 2012

Bu tez ile G.Ü. Fen Bilimleri Enstitüsü Yönetim Kurulu Doktora derecesini onamıştır.

Prof. Dr. Şeref SAĞIROĞLU
Fen Bilimleri Enstitüsü Müdürü

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

Tusan DERYA

**MALİYET TABANLI PARALEL MONTAJ HATTI
DENGELEME PROBLEMLERİ: YENİ MODELLER**

(Doktora Tezi)

Tusan DERYA

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

Eylül 2012

ÖZET

Bu tezde, maliyet tabanlı amaç fonksiyonu ile paralel montaj hatlarının eş zamanlı dengelenmesi problemi ele alınmıştır. Bu problem, *maliyet tabanlı paralel montaj hattı dengeleme problemi* olarak adlandırılır. Maliyet tabanlı paralel montaj hattı dengeleme problemleri için karma tamsayılı programlama modelleri önerilmiştir. Ayrıca, öncelikli hedefler ile paralel montaj hatlarını dengelemek için bir hedef programlama yaklaşımı önerilmiştir. Problem NP-zor olduğu için, matematiksel model tabanlı bir sezgisel algoritma ve bir melez paralel genetik algoritma orta ve büyük boyutlu problemleri çözmek için geliştirilmiştir. Önerilen sezgisel yaklaşımlar literatürde iyi bilinen kıyaslama problemleri üzerinde test edilmiştir ve yaklaşımların performansı birbirleri ile karşılaştırılmıştır. Deneysel araştırma sonuçları melez paralel genetik algoritmanın diğer sezgisel yaklaşıma kıyasla önemli ölçüde daha iyi çözümler elde ettiğini göstermektedir.

Bilim Kodu : 906. 1.141

Anahtar Kelimeler : Maliyet tabanlı montaj hattı dengeleme, paralel montaj hatları, tamsayılı programlama, sezgisel yöntemler

Sayfa Adedi : 253

Tez Yöneticisi : Prof. Dr. Hadi GÖKÇEN

**COST-ORIENTED PARALLEL ASSEMBLY LINE BALANCING
PROBLEMS: NEW MODELS**

(Ph D. Thesis)

Tusan DERYA

**GAZİ UNIVERSITY
INSTITUTE OF SCIENCE AND TECHNOLOGY**

September 2012

ABSTRACT

In this thesis, the problem of balancing simultaneously parallel assembly lines with cost-oriented objective function is addressed. This problem is called *cost-oriented parallel assembly line balancing problem*. Several mixed integer programming models for cost-oriented parallel assembly line problems are proposed. Furthermore, a goal programming approach to balance parallel assembly lines with precise goals is proposed. Since the problem is well-known as NP-hard, a heuristic algorithm based on mathematical model and a hybrid parallel genetic algorithm are developed to solve the medium- and large-size scales problem. The proposed heuristic approaches are tested on the well-known benchmark problems in the literature, and performance of the approaches are compared with each other. The results of an experimental investigation show that the hybrid parallel genetic algorithm achieves significantly better solutions than other heuristic approach.

Science Code	: 906. 1.141
Key Words	: Cost-oriented assembly line balancing, parallel assembly lines, integer programming, heuristic methods
Page Number	: 253
Adviser	: Prof. Dr. Hadi GÖKÇEN

TEŞEKKÜR

Çalışmalarım boyunca değerli yardım ve katkılarıyla beni yönlendiren Değerli Danışman Hocam Prof. Dr. Hadi GÖKÇEN'e, araştırma süresince her türlü bilgi ve tecrübelerinden istifade ettiğim Tez İzleme Komitesi üyeleri Sayın Hocam Prof. Dr. Bilal TOKLU ve Sayın Hocam Doç. Dr. Yakup KARA'ya, yine kıymetli tecrübelerinden faydalandığım Sayın Hocam Prof. Dr. Orhan TÜRKBEY ve Sayın Hocam Yrd. Doç Dr. Uğur ÖZCAN'a, manevi destekleriyle beni hiçbir zaman yalnız bırakmayan aileme ve sevgili Hesna Özbek ÜLKÜTAŞ'a teşekkürü bir borç bilirim.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT	v
TEŞEKKÜR	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ	x
ŞEKİLLERİN LİSTESİ	xiv
SİMGELER VE KISALTMALAR	xvii
1. GİRİŞ	1
2. MONTAJ HATTI DENGEME	5
2.1. Üretim Sistemleri	5
2.2. Montaj Hattı ve Hat Dengeleme ile ilgili Temel Kavramlar	8
2.3. Montaj Hattı Dengeleme	16
2.4. Montaj Hattı Dengelemenin Amaçları	18
2.5. Montaj Hattı Dengelemenin Temel Prensipleri	19
2.6. Montaj Hattı Dengelemesini Etkileyen Temel Etmenler ve Kısıtlar	20
2.7. Montaj Hatlarının Sınıflandırılması	22
2.7.1. Model sayısına göre montaj hatları	23
2.7.2. Hattın kontrol yapısına göre montaj hatları	25
2.7.3. Otomasyon seviyesine göre montaj hatları	26
2.7.4. Yerleşim şekline göre montaj hatları	27
2.8. Montaj Hattı Dengeleme Problemlerinin Sınıflandırılması	38

Sayfa

3. MALİYET TABANLI PARALEL MONTAJ HATTI Dengeleme	43
3.1. Paralel Montaj Hattı Dengeleme Problemleri	43
3.2. Maliyet Tabanlı Montaj Hattı Dengeleme Problemi	52
3.3. Maliyet Tabanlı Paralel Montaj Hattı Dengeleme Problemi	60
3.4. Kaynak Araştırması	64
3.4.1. Paralel montaj hattı dengeleme problemi	65
3.4.2. Maliyet tabanlı montaj hattı dengeleme problemi	68
4. M-PMHD PROBLEMLERİ İÇİN YENİ MODELLER	70
4.1. M-PMHD Problemi için İstasyon Odaklı Tamsayılı Programlama Modeli	71
4.2. M-PMHD Problemi için Operatör Odaklı Tamsayılı Programlama Modeli	79
4.3. M-YZPMHD Problemi için Tamsayılı Programlama Modelleri	82
4.4. M-ÇÜPMHD Problemi için Tamsayılı Programlama Modelleri	86
4.5. Ücret Oranlarının Düzgünleştirilmesi için Tamsayılı Programlama Modeli	93
4.6. M-PMHD Problemi için Hedef Programlama Modeli	96
4.7. Toplam İşgücü Maliyeti için Alt Sınır Algoritması	111
5. M-PMHD PROBLEMİ İÇİN SEZGİSEL ALGORİTMALAR	114
5.1. Paralel Kayan Problem Penceresi Algoritması	115
5.2. Melez Çok Popülasyonlu Paralel Genetik Algoritma	132
5.2.1. Genetik algoritmalar	132
5.2.2. Çok popülasyonlu paralel genetik algoritmalar	141
5.2.3. Geliştirilen genetik algoritma	147

Sayfa

6. DENEYSEL ÇALIŞMALAR	167
6.1. Test Problemleri	168
6.2. Tamsayılı Programlama Modeli ile Elde Edilen En İyi Çözümler	170
6.3. M–MHD ve M–PMHD Problemleri Sonuçlarının Karşılaştırılması	172
6.4. PKPP Algoritmasının En Uygun Parametre Setinin Belirlenmesi	173
6.5. PKPP Algoritmasının Performansının Değerlendirilmesi	175
6.6. M–ÇPPG Algoritmasının En Uygun Parametre Setinin Belirlenmesi	179
6.7. M–ÇPPG Algoritmasının Performansının Değerlendirilmesi	186
7. SONUÇ VE ÖNERİLER	191
KAYNAKLAR	194
EKLER.....	205
EK–1 PKPP algoritmasının parametre setleri	206
EK–2 M–ÇPPG algoritmasının farklı parametre setleriyle <i>Roszieng, Hahn</i> ve <i>Lutz2</i> problemleri için elde edilen sonuçlar	209
EK–3 <i>Roszieng, Hahn</i> ve <i>Lutz2</i> problemleri için M–ÇPPG algoritmasının farklı parametre setleriyle yapılan deney sonuçları	221
EK–4 <i>Roszieng</i> problemi için varyans analizi sonuçları	224
EK–5 <i>Roszieng</i> problemi için etkileşim grafikleri	225
EK–6 <i>Hahn</i> problemi için varyans analizi sonuçları	226
EK–7 <i>Hahn</i> problemi için etkileşim grafikleri	227
EK–8 <i>Lutz2</i> problemi için varyans analizi sonuçları	228
EK–9 <i>Lutz</i> problemi için etkileşim grafikleri	229
ÖZGEÇMİŞ	230

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 2.1. Montaj hatların sınıflandırılması	23
Çizelge 2.2. BMHDP'nin sınıflandırılması	40
Çizelge 3.1. Geleneksel MHD ve PMHD problemlerinin en iyi görev atamaları	50
Çizelge 3.2. M-MHD probleminin tamsayı karar modelinde kullanılan notasyonlar	55
Çizelge 3.3. M-MHD probleminin tamsayı karar modelinde kullanılan parametreler	56
Çizelge 3.4. M-MHD probleminin tamsayı karar modelinde kullanılan değişkenler	56
Çizelge 3.5. Maliyet tabanlı MHD ve zaman tabanlı MHD problemlerinin en iyi çözümleri	59
Çizelge 4.1. M-PMHD probleminin istasyon odaklı tamsayı karar modelinde kullanılan notasyonlar	72
Çizelge 4.2. M-PMHD probleminin istasyon odaklı tamsayı karar modelinde kullanılan parametreler	72
Çizelge 4.3. M-PMHD probleminin istasyon odaklı tamsayı karar modelinde kullanılan değişkenler	72
Çizelge 4.4. Operatörlerin ücret oranları	76
Çizelge 4.5. PMHD probleminin en iyi görev atamaları	78
Çizelge 4.6. M-MHD problemlerinin en iyi görev atamaları	78
Çizelge 4.7. M-PMHD probleminin operatör odaklı tamsayı karar modelinde kullanılan değişkenler	79
Çizelge 4.8. M-YZPMHD probleminin en iyi görev atamaları	86
Çizelge 4.9. Hatlardaki montajı yapılacak ürünlere göre duyarlılık analizi	92

Çizelge	Sayfa
Çizelge 4.10. M-PMHD probleminin ücret düzgünleştirme ve toplam işgücü maliyetini enküçükleme amaçlarıyla elde edilen en iyi görev atamaları	96
Çizelge 4.11. Hedef programlama formülasyonları	98
Çizelge 4.12. Hedef programlama modelinde kullanılan parametre ve hedef değerleri	100
Çizelge 4.13. Hedef programlama modelinde kullanılan değişkenler	100
Çizelge 4.14. Üç hedef için farklı öncelik sıralamaları	103
Çizelge 4.15. Üç hedefin farklı öncelik sıralamaları için uzlaşık çözümler	105
Çizelge 4.16. Toplam üretim maliyetini enküçüklemek için önerilen tamsayılı karar modellerinde kullanılan parametreler	107
Çizelge 4.17. M-PMHD probleminin farklı sermaye maliyetlerinde elde edilen en iyi görev atamaları	111
Çizelge 4.18. Toplam işgücü maliyeti alt sınır değeri hesabı algoritmasında kullanılan notasyonlar	112
Çizelge 5.1. Paralel kayan problem penceresi algoritmasında kullanılan notasyonlar	117
Çizelge 5.2. Ürünlere ait görevlerin pozisyon ağırlıkları	122
Çizelge 5.3. PKPP algoritmasının birinci aşamasındaki birinci problem penceresinin en iyi görev atamaları	123
Çizelge 5.4. PKPP algoritmasının birinci aşamasındaki ikinci problem penceresinin en iyi görev atamaları	125
Çizelge 5.5. PKPP birinci aşaması sonucunda elde edilen çözüm	125
Çizelge 5.6. PKPP ikinci aşamasındaki birinci problem penceresinin en iyi çözümü	128
Çizelge 5.7. PKPP ikinci aşamasındaki ikinci problem penceresinin en iyi çözümü	129
Çizelge 5.8. PKPP ikinci aşamasındaki üçüncü problem penceresinin en iyi çözümü	129

Çizelge		Sayfa
Çizelge 5.9.	PKPP ikinci aşamasındaki dördüncü problem penceresinin en iyi çözümü	130
Çizelge 5.10.	PKPP ikinci aşaması sonucunda elde edilen çözüm	130
Çizelge 6.1.	M-PMHD test problemlerinin karakteristikleri	170
Çizelge 6.2.	Operatör odaklı tamsayılı programlama modeli ile elde edilen M-PMHD problemlerinin en iyi değerleri	171
Çizelge 6.3.	M-PMHD ve M-MHD problemlerinin en iyi çözümleri	172
Çizelge 6.4.	PKPP algoritması ile küçük boyutlu problemler için elde edilen sonuçlar	176
Çizelge 6.5.	PKPP algoritması ile orta boyutlu problemler için elde edilen sonuçlar	177
Çizelge 6.6.	PKPP algoritması ile büyük boyutlu problemler için elde edilen sonuçlar	178
Çizelge 6.7.	Deneyisel çalışmalarda kullanılan test problemlerinin karakteristikleri	180
Çizelge 6.8.	M-ÇPPG algoritmasının parametre değerleri	182
Çizelge 6.9.	M-ÇPPGA'sının en iyi parametre seti	185
Çizelge 6.10.	M-ÇPPG algoritması ile küçük boyutlu problemler için elde edilen sonuçlar	186
Çizelge 6.11.	M-ÇPPG algoritması ile orta boyutlu problemler için elde edilen sonuçlar	187
Çizelge 6.12.	M-ÇPPG algoritması ile büyük boyutlu problemler için elde edilen sonuçlar	188
Çizelge 6.13.	PKPP ve M-ÇPPGA karşılaştırılması	190
Çizelge 7.1.	PKPP algoritmasının birinci ve ikinci aşamasındaki parametre değerleri	206
Çizelge 8.1.	Roszieng probleminin deney sonuçları	209
Çizelge 8.2.	Hahn probleminin deney sonuçları	213

Çizelge	Sayfa
Çizelge 8.3. Lutz2 probleminin deney sonuçları	217

ŞEKİLLERİN LİSTESİ

Şekil		Sayfa
Şekil 2.1.	Genel bir üretim sistemi	5
Şekil 2.2.	Denge gecikmesi olan bir üretim hattı	11
Şekil 2.3.	10 görevli örnek öncelik diyagramı	12
Şekil 2.4.	10 görevli örnek öncelik matrisi	13
Şekil 2.5.	Örnek bir montaj hattı şeması	16
Şekil 2.6.	(a) Tek modellenli (b) Karışık modellenli (c) Çok modellenli hatlar	25
Şekil 2.7.	Fiziksel montaj hattı tipleri: (a) düz hat (b) zig-zag hat (c) değişik açılı (d) U tipi hat (e) dairesel hat	28
Şekil 2.8.	Düz hat yerleşimi	29
Şekil 2.9.	Paralel düz hat yerleşimi	30
Şekil 2.10.	U tipi hat yerleşimi	33
Şekil 2.11.	Eş zamanlı dengelenmiş U tipi montaj hatları	37
Şekil 2.12.	Montaj Hattı Dengelem Problemlerinin sınıflandırılması	39
Şekil 3.1.	Paralel konumlu 3 düz hatlı bir üretim sistemi	43
Şekil 3.2.	Normalizasyon algoritmasının adımları	47
Şekil 3.3.	(a) 10 görevli (b) 8 görevli öncelik diyagramı	48
Şekil 3.4.	(a) 10 görevli geleneksel MHD problemi için en iyi görev atamaları (b) 8 görevli geleneksel MHD problemi için en iyi görev atamaları (c) PMHD problemi için en iyi görev atamaları	49
Şekil 3.5.	8 görevli öncelik diyagramı	58
Şekil 4.1.	(a) 10 görevli (b) 8 görevli öncelik diyagramları	75
Şekil 4.2.	M-PMHD probleminin istasyon odaklı tamsayılı karar modeli ile bulunan en iyi görev atamaları	75

Şekil		Sayfa
Şekil 4.3.	M–PMHD probleminin operatör odaklı tamsayılı karar modeli ile bulunan en iyi görev atamaları	81
Şekil 4.4.	Dönüştürme işlemi sonucunda elde edilen en iyi görev atamaları	82
Şekil 4.5.	(a) 11 görevli (b) 9 görevli (c) 7 görevli öncelik diyagramları	91
Şekil 4.6.	Üç hatlı M–ÇÜPMHD probleminin en iyi görev ataması	92
Şekil 4.7.	(a) 11 görevli (b) 10 görevli öncelik diyagramları	95
Şekil 4.8.	(a) 10 görevli (b) 8 görevli öncelik diyagramları	102
Şekil 4.9.	(a) 10 görevli (b) 8 görevli öncelik diyagramları	110
Şekil 4.10.	Toplam işgücü maliyetinin alt sınır değeri hesabı için bir algoritma	113
Şekil 5.1.	PKPP algoritmasının birinci aşamasının adımları	118
Şekil 5.2.	PKPP algoritmasının ikinci aşamasının adımları	119
Şekil 5.3.	(a) 25 görevli (b) 24 görevli öncelik diyagramları	120
Şekil 5.4.	PKPP algoritmasının birinci aşamasının genel gösterimi	126
Şekil 5.5.	PKPP algoritmasının ikinci aşamasının genel gösterimi	131
Şekil 5.6.	İkili genetik gösterim için örnek bir kromozom	135
Şekil 5.7.	Örnek bir popülasyon	136
Şekil 5.8.	Örnek bir çaprazlama işlemi (tek nokta çaprazlama)	137
Şekil 5.9.	Örnek bir mutasyon işlemi	137
Şekil 5.10.	Genetik algoritmanın akış şeması	138
Şekil 5.11.	Çok popülasyonlu paralel genetik algoritmaların genel yapısı .	143
Şekil 5.12.	Örnek bir genetik gösterim	148
Şekil 5.13.	Başlangıç popülasyonu oluşturmak için kullanılan algoritma ..	150

Şekil		Sayfa
Şekil 5.14.	Çaprazlama operatörü	153
Şekil 5.15.	Adaptasyon operatörü	154
Şekil 5.16.	(a) 8 görevden (b) 10 görevden oluşan 2 ürünlü bir problem ...	155
Şekil 5.17.	Örnek bir çaprazlama gösterimi ($r = 4$)	155
Şekil 5.18.	Çaprazlama operatörü sonucunda kusurlu çıkan bir çocuk	156
Şekil 5.19.	Adaptasyon operatörü sonucunda düzeltilen çocuk	156
Şekil 5.20.	Mutasyon operatörü uygulanacak örnek bir kromozom	158
Şekil 5.21.	Mutasyona uğramış örnek bir çözüm	158
Şekil 5.22.	İstasyon sayısında azalmanın gerçekleştiği bir duruma örnek ..	159
Şekil 5.23.	Melez çok popülasyonlu paralel genetik algoritmanın genel yapısı	161
Şekil 5.24.	Benzerlik oranı hesabına bir örnek	164
Şekil 5.25.	M-ÇPPGA'nın genel adımları	166
Şekil. 6.1.	<i>Roszieng</i> problemi için yapılan 5 deneme sonunda elde edilen toplam işgücü maliyet değerlerinin ortalaması	221
Şekil. 6.2.	<i>Hahn</i> problemi için yapılan 5 deneme sonunda elde edilen toplam işgücü maliyet değerlerinin ortalaması	222
Şekil. 6.3.	<i>Lutz2</i> problemi için yapılan 5 deneme sonunda elde edilen toplam işgücü maliyet değerlerinin ortalaması	223
Şekil. 7.1.	<i>Roszieng</i> problemi için parametreler arası etkileşim grafiği	225
Şekil. 7.2.	<i>Roszieng</i> problemi için parametre – ortalama işgücü maliyeti grafiği	225
Şekil. 8.1.	<i>Hahn</i> problemi için parametreler arası etkileşim grafiği	227
Şekil. 8.2.	<i>Hahn</i> problemi için parametre – ortalama işgücü maliyeti grafiği	227
Şekil. 9.1.	<i>Lutz2</i> problemi için parametreler arası etkileşim grafiği	229

Şekil		Sayfa
Şekil. 9.2.	<i>Lutz2</i> problemi için parametre – ortalama işgücü maliyeti grafiği	229

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	Açıklama
n	toplam görev sayısı
i, j, r, s	görev numarası (indeksi), $i = 1, \dots, n$
K	en fazla istasyon sayısı
k	istasyon numarası (indeksi), $k = 1, \dots, K$
K_{min}	belirli bir çevrim zamanı için teorik en az istasyon sayısı
t_i	i görevinin tamamlanma zamanı, $i = 1, \dots, n$
C	çevrim zamanı
C_{alt}	çevrim zamanı için alt sınır
$C_{üst}$	çevrim zamanı için üst sınır
S_k	k istasyonunun istasyon zamanı, $k = 1, \dots, K$
S_{max}	en büyük istasyon zamanı
BZ_k	k istasyonunun boş zamanı, $k = 1, \dots, K$
TBZ	bir hattaki istasyonların toplam boş zamanı
D	belirlenen dönemde üretimi gerçekleştirilecek ürün miktarı
T	belirlenen dönemde aktif olarak kullanılan zaman (gün, saat vs.)
DG	denge geçikmesi
HE	hattın etkinliği
$D\dot{I}$	düzgünlük indeksi
SK	sıra kuvveti
EO	esneklik oranı
P_i	i görevinin öncül görevler kümesi, $i = 1, \dots, n$
F_i	i görevinin ardıl görevler kümesi, $i = 1, \dots, n$

Simgeler	Açıklama
$r \in P_s$	bir öncelik ilişkisi; r görevi s görevinden önce tamamlanmalı
$ F_i $	i görevinin ardıl görevler sayısı
Y	öncelik matrisindeki toplam “0” (sıfır) sayısı
$\overline{H}$	en fazla hat sayısı
h	hat numarası (indeksi), $h = 1, \dots, \overline{H}$
C_h	h hattı için çevrim zamanı, $h = 1, \dots, \overline{H}$
EKOK	tüm hatların çevrim zamanlarının en küçük ortak katı
TM	birim ürün için toplam üretim maliyet
n_h	h hattında üretilen ürüne ait toplam görev sayısı, $h = 1, \dots, \overline{H}$
$\overline{K}$	en fazla istasyon sayısı
I_k	k istasyonuna atanan görevler kümesi
k_c	bir istasyon için toplam sermaye maliyeti
v_i	i görevinin birim zaman için ücret oranı, $i = 1, \dots, n_h$
w_i	i görevinin birim zaman için maliyet oranı, $i = 1, \dots, n_h$
vs_k	k istasyonunun birim zaman için ücret oranı, $k = 1, \dots, \overline{K}$
ws_k	k istasyonunun birim zaman için maliyet oranı, $k = 1, \dots, \overline{K}$
x_{ik}	1, eğer i görevi k istasyonuna atanmış ise; 0, aksi halde
z_k	1, eğer k istasyonu kullanılmış ise; 0, aksi halde
P_{hi}	h hattında montajı yapılacak ürüne ait i görevinin öncül görevler kümesi, $h = 1, \dots, \overline{H}$, $i = 1, \dots, n_h$
F_{hi}	h hattında montajı yapılacak ürüne ait i görevinin ardıl görevler kümesi, $h = 1, \dots, \overline{H}$, $i = 1, \dots, n_h$
$(r, s) \in P_{hs}$	bir öncelik ilişkisi; h hattındaki r görevi s görevinden önce tamamlanmalıdır

Simgeler	Açıklama
t_{hi}	h hattında montajı yapılacak ürüne ait i görevinin tamamlanma zamanı, $h = 1, \dots, \overline{H}$, $i = 1, \dots, n_h$
w_{hi}	h hattında montajı yapılacak ürüne ait i görevinin birim zaman için ücret oranı, $h = 1, \dots, \overline{H}$, $i = 1, \dots, n_h$
x_{hik}	1, eğer h hattında üretilecek olan ürüne ait i görevi k istasyonuna atanmış ise; 0, aksi halde (<i>istasyon odaklı model için</i>)
u_{hk}	1, eğer h hattındaki k istasyonu kullanılmış ise; 0, aksi halde
$z_{h(h+1)k}$	1, eğer h ve $h+1$ bitişik hatlarının komşu olan k istasyonları arasında ortak işyeri kurulmuş ise; 0, aksi halde
wn_{hk}	h hattındaki k istasyonuna kurulan ayrıık işyerinde çalışan operatörün ücret oranı, $h = 1, \dots, \overline{H}$, $k = 1, \dots, \overline{K}$
$ws_{h(h+1)k}$	h ve $h+1$ hatlarının komşu olan k istasyonları arasında kurulan ortak işyerinde çalışan operatörün ücret oranı, $h = 1, \dots, \overline{H} - 1$, $k = 1, \dots, \overline{K}$
x_{hik}	1, eğer h hattında üretilecek olan ürüne ait i görevi k operatörü tarafından yapılacak ise; 0, aksi halde (<i>operatör odaklı model için</i>)
w_k	k operatörünün ücret oranı, $k = 1, \dots, \overline{K}$
d	bitişik hatlar arasındaki tek yön yürüme süresidir.
$p_{h(h+1)k}$	1, eğer h ve $h+1$ bitişik hatlarının komşu istasyonlarında k operatörü çalışıyor ise; 0, aksi halde
n_p	p ürününe ait toplam görev sayısı, $p = 1, \dots, P$
p	ürün numarası (indeksi), $p = 1, \dots, P$
t'_{pi}	p ürününe ait i görevinin tamamlanma zamanı, $p = 1, \dots, P$, $i = 1, \dots, n_p$
w'_{pi}	p ürününe ait i görevinin ücret oranı, $p = 1, \dots, P$, $i = 1, \dots, n_p$

Simgeler	Açıklama
x_{hpik}	1, eğer h hattında montajı yapılacak olan p ürününe ait i görevi k istasyonuna atanmış ise; 0, aksi halde (<i>istasyon odaklı model için</i>)
x_{hpik}	1, eğer h hattında montajı yapılacak olan p ürününe ait i görevi k operatörüne atanmış ise; 0, aksi halde (<i>operatör odaklı model için</i>)
y_{hp}	1, eğer h hattında p ürününün montajı yapılacak ise; 0, aksi halde
w_{enb}	en büyük görev ücret oranı
w_k^+	ortalama ücret oranından pozitif sapma miktarı, $k = 1, \dots, \overline{K}$
w_k^-	ortalama ücret oranından negatif sapma miktarı, $k = 1, \dots, \overline{K}$
TIM	toplam işgücü maliyeti için hedef değeri
OS	işyeri (operatör) sayısı için hedef değeri
HU	hat uzunluğu (konveyör bant uzunluğu) için hedef değeri
M	büyük bir sayı
m^+	toplam işgücü maliyeti hedefi için pozitif sapma değişkeni
m^-	toplam işgücü maliyeti hedefi için negatif sapma değişkeni
o^+	gerekli olan operatör sayısı hedefi için pozitif sapma değişkeni
o^-	gerekli olan operatör sayısı hedefi için negatif sapma değişkeni
g^+	hat uzunluğu hedefi için pozitif sapma değişkeni
g^-	hat uzunluğu hedefi için negatif sapma değişkeni
NS	paralel hatların bant uzunluğunu gösteren tamsayı değişken
s_{hk}	h hattında çalışan k operatörünün görev aldığı istasyonu gösteren tamsayı değişken
c_a	işyeri açma maliyeti
c_b	konveyör bant sisteminin birim uzunluğunun maliyeti
k_h	h hattında montajı yapılan ürün için gerekli teorik istasyon sayısı, $h = 1, \dots, \overline{H}$

Simgeler	Açıklama
k_{min}	her hatta üretilecek ürünler için gerekli olan teorik istasyon sayılarının en büyüğü
W	farklı ücret braket sayısı
w	ücret braket numarası (indeksi), $w = 1, \dots, W$
k_w	w ücret braketinin ücret oranı, $w = 1, \dots, W$
d_w	w ücret braketinde bulunan görevlerin toplam tamamlanma zamanı, $w = 1, \dots, W$
d_{kalan}	mevcut istasyondaki boş zaman
M_w	w ücret braketinde bulunan tüm görevlerin atanması için gerekli istasyon sayısı, $w = 1, \dots, W$
M_{lb}	tüm görevlerin uygun olarak atanabileceği en az istasyon sayısı
M_{toplam}	görevlerin atandığı toplam istasyon sayısı
k_{lb}	toplam işgücü maliyeti için alt sınır değeri
PW_{hi}	h hattında montajı yapılacak ürüne ait i görevinin pozisyon ağırlığı
α	sabitlenme oranı; $0 < \alpha \leq 1$
L	problem penceresindeki görev sayısı; $L \leq n_h$, $h = 1, \dots, \overline{H}$
I_{hk}	h hattındaki k istasyonuna atanan görevler kümesi (1. aşama için)
I_{hk}^*	h hattındaki k istasyonuna atanan görevler kümesi (2. aşama için)
$I_{h,pencere}$	problem penceresini oluşturan h hattında montajı yapılan ürüne ait görevler kümesi $h = 1, \dots, \overline{H}$
$I_{h,kalan}$	henüz herhangi bir istasyona atanmamış h hattında montajı yapılan ürüne ait görevler kümesi $h = 1, \dots, \overline{H}$
$K_{pencere}$	Alt problemin en iyi çözümde problem penceresini oluşturan görevlerin atandığı her hatta kurulan istasyon sayısı
K_{sabit}	sabitlenecek istasyon sayısı
K_{toplam}	toplam sabitlenen istasyon sayısı

Simgeler	Açıklama
S	istasyon sayısı
K_{ilk}	ilk istasyon indeksi
K_{son}	son istasyon indeksi
$f(p)$	problem penceresini oluşturan görevlerin mevcut atamalarına göre amaç fonksiyonu değeri
$f^*(p)$	alt problemin (problem penceresinin) en iyi değeri
imp	amaç fonksiyonunda iyileşme sağlandığını gösteren değişken
δ	kayma miktarı

Kısaltmalar	Açıklama
TZÜ	Tam Zamanında Üretim
TMD	Tek modellenli Deterministik
TMS	Tek Modelli Stokastik
ÇMD	Çok Modelli Deterministik
ÇMS	Çok Modelli Stokastik
MHD	Montaj hattı dengeleme
BMHD	Basit montaj hattı dengeleme
GMHD	Geleneksel montaj hattı dengeleme
UMHD	U tipi montaj hattı dengeleme
PMHD	Paralel montaj hattı dengeleme
M-MHD	Maliyet tabanlı montaj hattı dengeleme
M-PMHD	Maliyet tabanlı paralel montaj hattı dengeleme
M-YZPMHD	Maliyet tabanlı yürüme zamanlı paralel montaj hattı dengeleme
M-ÇÜPMHD	Maliyet tabanlı çok ürünli paralel montaj hattı dengeleme
PKPP	Paralel kayan problem penceresi
PGA	Paralel genetik algoritma
M-ÇPPGA	Melez çok popülasyonlu paralel genetik algoritma

1. GİRİŞ

Günümüzde küreselleşme süreciyle birlikte yaşanan yoğun rekabet koşulları, sanayi işletmelerinin mevcut işgücü, makine, malzeme, hammadde ve sermaye kaynaklarını etkin ve verimli bir şekilde kullanmalarını zorunlu kılmaktadır. Kaynakların verimli kullanılması sonucunda işletmelerin maliyetleri önemli ölçüde azalacak, kapasite kullanım oranları ve toplam kârlarında artışlar sağlanabilecektir. Bu amaçlara ulaşabilmek için ise işgücü, makine, malzeme ve teçhizattan oluşan üretim sistemlerinde kullanılan üretim yöntemlerinin yeniden tasarımı gerekmektedir.

Sürekli üretim sistemlerinde, üretimin birimler halinde gerçekleştirildiği ve kitle talebin olduğu durumlarda, yüksek üretim hızıyla talebi karşılamanın en makul yolu montaj hatlarının yapılandırılmasıdır. Endüstrileşme sürecinde yüksek hacimde, hızlı ve düşük maliyetli üretim yapabilmek için uygulanmaya başlanan montaj hatları, günümüzde başta otomotiv, beyaz eşya ve elektronik endüstrisi olmak üzere birçok seri üretim sistemlerinin vazgeçilmez en temel parçası haline gelmiştir.

Bir ürünü oluşturan parçaların önceden belirlenmiş bir sıraya uygun olarak birleştirilmesi işlemi *montaj* olarak adlandırılır. Sürekli ve yüksek talepli, standart nihai ürünleri elde etmek için, montaj işleminin ve ilgili operasyonların önceden belirlenmiş bir sıraya göre gerçekleştirildiği ve aralarında yarı mamulün malzeme taşıma sistemi (genelde bir konveyör) yardımıyla taşındığı ardışık iş istasyonlarına *montaj hattı* denir. Montaj işlemlerinin yapılabilmesi için gerekli görevlerin, montaj sürecindeki kısıtlara uyarak, bir veya daha fazla amacı en iyileyecek şekilde, sıralı iş istasyonlarına atanması *Montaj Hattı Dengeleme* (MHD) problemi olarak adlandırılmaktadır. MHD probleminin amacı, genellikle sabit bir çevrim zamanı için hat boyunca kullanılacak istasyon sayısını enküçükmektir [Ghosh and Gagnon, 1989]. Diğer bir ifadeyle, tüm istasyonlardaki toplam boş zamanı enküçükmektir [Baybars, 1986]. Bundan dolayı bu problemlere *zaman tabanlı montaj hattı dengeleme* denilmektedir.

Son yıllarda sanayide ortaya çıkan acımasız rekabet ile birlikte, üretim maliyetleri daha fazla anlamlı hale gelmiştir. Montaj hattına sahip başarılı üretim sistemlerinde bile, üretim maliyetlerini aşağı çekme çabası bir zorunluluk haline gelmiştir. Yatırım maliyetinin yüksek olduğu bu üretim sistemlerinde düşük kapasite kullanımı ve verimsiz çalışmanın maliyeti oldukça yüksektir. Böyle bir ekonomik bakış açısı altında, montaj hatlarının dengelenmesi için maliyet tabanlı bir amaç kullanılabilir. Montaj hatlarının maliyet tabanlı bir yaklaşımla birlikte dengelenmesine *maliyet tabanlı montaj hattı dengeleme* denilmektedir. Maliyet tabanlı montaj hattı dengeleme konusunda ayrıntılı bilgi [bölüm 3.2](#)'de verilmiştir.

Birbirine paralel olarak konumlandırılmış birden fazla düz montaj hattına sahip imalat sistemlerine sanayide oldukça yaygın olarak rastlanmaktadır. Ürünleri bir ya da daha fazla düz montaj hattında üretmede iki durum söz konusudur. *Birinci durum* talebin yeterince yüksek olması ve bir hattın kapasitesinin söz konusu talebi karşılamada yetersiz kalması ve ikinci hattın kurulması ihtiyacıdır. Diğer bir ifadeyle, aynı ürünün birden fazla özdeş hatta üretilmesidir. *İkinci durum* ise, her bir ürün talebinin hat kurmak için yeterli büyüklükte olması ve birden fazla benzer ürünün birbirinden ayrı montaj hatlarında üretilmesidir. Bu hatlarda montajı yapılan ürünlerin birbirinin aynı ya da benzeri oldukları ve çevrim zamanlarının birbirine eşit ya da yakın oldukları görülmektedir.

Montaj hattı dengeleme kaynaklarında, şu ana kadar yapılan çalışmaların çoğunda, bir montaj hattında bir ürünün veya birden fazla farklı ürünün aynı hatta üretilmesi düşünülerek, istasyon sayısını enküçükleyecek şekilde paralel hatlar birbirinden bağımsız olarak dengelenmiştir. Yani bu çalışmalarda, sadece bir montaj hattının dengelenmesi ele alınmıştır. Bir montaj hattının talebi karşılamaması durumunda ise, kaç tane özdeş paralel hattın kurulması üzerine çok kısıtlı olmakla birlikte bazı çalışmalar bulunmaktadır: [[Süer ve Dağlı, 1994](#), [Süer, 1998](#)]. Ancak bu çalışmalarda da kurulacak paralel hatların dengelenmesi bağımsız olarak ayrı ayrı yapılmaktadır. Paralel düz montaj hatlarının eş zamanlı dengelenmesi düşüncesi ilk olarak [Gökçen ve ark. \(2006\)](#) tarafından ortaya atılmıştır. Birden fazla düz hattın ortak bir şekilde dengelendiği bu problem, *Paralel Montaj Hattı Dengeleme* (PMHD) problemi olarak

adlandırılmaktadır. PMHD probleminin ana amacı, istasyon sayısını (veya operatör sayısını) enküçüklemeektir. PMHD problemi ile ilgili ayrıntılı bilgi [bölüm 3.1](#)'de verilmiştir.

Bu çalışmanın amacı; birbirine paralel olan birden fazla özdeş ya da benzer düz montaj hatların eş zamanlı dengelenmesi için maliyet tabanlı yaklaşımlar ve yeni çözüm önerileri getirmektir. Bu amaç doğrultusunda, *Maliyet tabanlı Paralel Montaj Hattı Dengeleme* (M-PMHD) probleminin farklı yönlerini dikkate alarak, en iyi çözümünü bulmak için yeni tamsayılı programlama modelleri önerilmiştir. Fakat problem boyutunun artmasıyla birlikte çözüm süresinin de üstel olarak artması sonucu, büyük boyutlu problemlerin mantıksal zaman dilimi içerisinde en iyi görev atamaları bulunamamaktadır. Büyük boyutlu problemlerin yaklaşık çözümlerini bulmak amacıyla etkin bir matematiksel model tabanlı sezgisel algoritma ve melez çok popülasyonlu paralel genetik algoritma da önerilmiştir.

Bu çalışma yedi bölümden meydana gelmektedir. İkinci bölümde, öncelikle montaj hatlarıyla ilgili kaynaklarda kullanılan temel kavramlara ve MHD problemine yer verilmiştir.

Üçüncü bölümde, bu çalışmanın kapsamını oluşturan maliyet tabanlı MHD problemi ve PMHD probleminin tanımı ve çözümü için bugüne kadar geliştirilen değişik yöntemler hakkında detaylı bir kaynak araştırması verilmiştir.

Dördüncü bölümde, M-PMHD problemi için yeni tamsayılı programlama modelleri ve çözüm yaklaşımları önerilmiştir. M-PMHD problemi için istasyon odaklı ve operatör odaklı olmak üzere birbirinden farklı iki tamsayılı programlama modeli ve hedef programlama modeli, Maliyet tabanlı Yürüme Zamanlı Paralel Montaj Hattı Dengeleme (M-YZPMHD) problemi için tamsayılı programlama modelleri, Maliyet tabanlı Çok ürünli Paralel Montaj Hattı Dengeleme (M-ÇÜPMHD) problemi için tamsayılı programlama modelleri sunulmuştur. Ayrıca, önerilen yeni modeller ve yaklaşımlar değişik örnekler üzerinde uygulanmış ve elde edilen sonuçlar zaman tabanlı montaj hattı dengeleme sonuçlarıyla karşılaştırılmıştır.

Beşinci bölümde, makul sürelerde M-PMHD probleminin yaklaşık çözümlerini bulabilmek için önerilen matematiksel model tabanlı sezgisel algoritma ve melez çok popülasyonlu paralel genetik algoritma sunulmuştur.

Altıncı bölümde, önerilen sezgisel algoritmaların etkinlikleri deney setleri üzerinde test edilmiş ve elde edilen sonuçlar önerilen tamsayılı karar modeli sonuçlarıyla (veya alt sınır değerleriyle), çözüm kalitesi ve çözüm süresi açısından karşılaştırılmıştır.

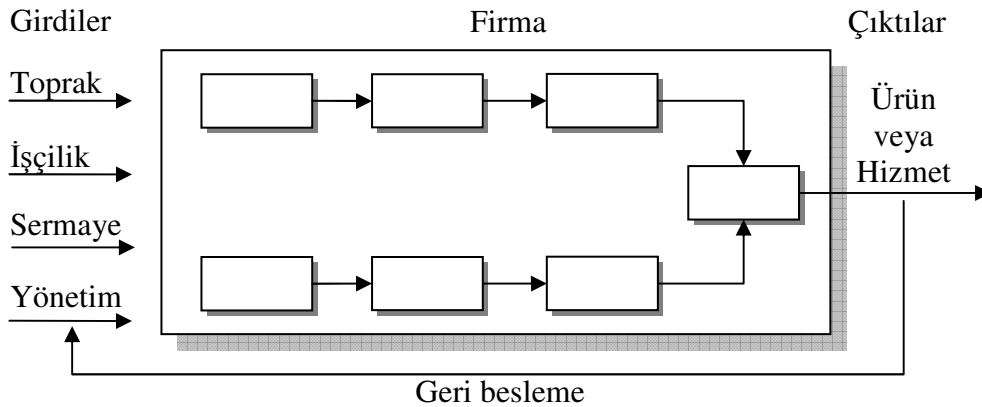
Son bölümde ise, araştırma sonuçlarının genel bir değerlendirmesi ve ileride yapılabilecek çalışmalar için öneriler yer almaktadır.

2. MONTAJ HATTI DENGELEME

2.1. Üretim ve Üretim Sistemleri

Üretim, mühendisler tarafından, bir fiziksel varlık üzerinde onun değerini arttıran bir değişiklik yapmak veya hammadde ve yarı mamulleri mamule dönüştürmek olarak tanımlanır [Buffa, 1965]. Ekonomistler ise, üretimi, bir fayda meydana getirilmesi şeklinde düşünürler. Bu iki tanımdan da anlaşılacağı gibi, üretim, içeriği çok geniş olmasına rağmen, ana amacı topluma değer yaratmak olan bir fonksiyondur. Diğer bir deyişle, üretim sistemleri sadece fiziksel üretimi kapsayan sistemlerce sınırlandırılmamıştır. Hizmet üreten sistemlerde, (eğitim, ulaşım, dağıtım vb.) girdilerin fiziksel veya kimyasal durumunda bir değişiklik söz konusu olmamakla beraber yine topluma bir değer yaratılmaktadır; dolayısıyla bu sistemler de üretim sistemi olarak tanımlanırlar [Acar, 1996].

Bütün üretim sistemlerinin ortak amaçlarına ulaşmak için uymak zorunda oldukları ortak nokta ise kaynakların verimli kullanılmasıdır. Üretim kaynakları genelde dört başlık altında toplanır: (i) toprak (veya hammadde kaynakları) (ii) işçilik (veya insangücü kaynakları) (iii) sermaye (iv) yönetim. Genellikle bir üretim sistemi, üretim kaynaklarının ürün ve/veya hizmete dönüşmesi için gereken tüm süreçleri içerir. Şekil 2.1’de genel bir üretim sistemi gösterilmektedir.



Şekil 2.1. Genel bir üretim sistemi

Üretim sistemlerini ürün çeşiti ve üretim miktarına göre genel olarak dört başlık altında sınıflandırmak mümkündür.

Proje üretimi: Belirli bir ürünün sadece bir kez üretilmesi için işgücü, makinelerin vb. girdilerin bir araya getirildiği üretim sistemleridir. Üretim esnasında, ürünün süreçler arasında hareketi söz konusu olmayıp, üretim için gerekli olan kaynakların üretimin gerçekleştirildiği yere getirilir ve bu girdi kaynaklarının sürekli hareketi söz konusudur. Başka bir ifadeyle, yerinde üretim yapılır. Bu tip üretim, esneklik ve yenilikler açısından büyük avantaj sağlamasına rağmen, üretim maliyetleri oldukça yüksektir. Uçak üretimi ve gemi yapımı proje üretimine örnek olarak verilebilir.

Siparişe göre üretim: Az sayıda ürünün belirli veya belirsiz aralıklarla üretilmesi söz konusudur. Müşteriler tarafından belirlenen miktar, kalite ve çeşitte ürünlerin üretildiği üretim sistemleridir. Geniş ürün yelpazesine sahip olan bu üretim sistemlerinde, değişik işlemleri yapabilecek özelliklerde tezgâhlar kullanılır. Bu üretim sistemlerinde sürece göre yerleşim söz konusudur. Siparişe göre üretimin dezavantajları; düşük kapasite kullanım oranları, planlama faaliyetlerinin zorluğu, işlem tekrarlılığının düşük olmasından dolayı düşük uzmanlaşma ve düşük kalitedir. Bununla birlikte, üretim esnekliği ve yeni ürünlerin üretime alınma hızı oldukça yüksektir. Siparişe göre üretim yapan sistemlere, bir otomobil yedek parçası imalatı yapan işletme örnek verilebilir.

Parti tipi üretim: Benzer veya aynı türden ürünlerin birbirini izleyen partiler halinde üretildiği ve her partide farklı bir ürün üretiminin gerçekleştirildiği üretim şeklidir. Parti üretimine göre bir partinin üretimi tamamlandıktan sonra, üretim sistemi diğer partinin üretimi için yeniden düzenlenir. Bu üretimde, bir partinin üretimi tamamlanmadan, diğerinin üretimine geçilmez. Parti tipi üretimde, siparişe göre üretime kıyasla ürün çeşitliliğinin düşük, işlem tekrarlılığının yüksek olmasından dolayı üretim maliyetleri daha düşüktür. Bu üretim tarzında genel amaçlı ekipman kullanımı ve esnek operatör kullanımı vardır. Bu üretim yöntemindeki temel planlama çalışması, parti büyüklüklerinin belirlenmesi ve üretim sıklıklarının tespitidir. Üretim planlama çalışmaları; üretim partilerinin teslim zamanlarına

yetiştirilmesi, üretimin çizelgelenmesi ve bir miktar ürünün de stoka yönlendirilmesiyle gerçekleştirilen stok yönetiminden dolayı daha büyük önem taşımaktadır. Konfeksiyon malları üreten işletmeler parti tipi üretime örnek verilebilir.

Seri üretim: Yüksek miktarlarda fakat düşük seviyede çeşitlilik gösteren birimler için uygulanan üretim şeklidir. Seri üretimin ana özelliği, ürün akışının olması (ürünün hareket halinde olması) ve tesislerin üretilecek ürüne göre tasarlanmasıdır. Seri üretimde, sistemin başlangıç noktasından başlayan hammadde, yarı mamul ve parça gibi girdiler, işlem birimlerinden geçmekte ve son ürün haline dönüşerek sistemden çıkmaktadırlar. Bu tip bir üretim, tüm üretim imkânlarının belirli bir ürüne tahsis edildiği, çok büyük miktarlarda üretimin yapılarak ürünlerin tamamının satıldığı üretim sistemleridir. Bu üretim sistemlerinde talebin sürekliliği esastır. Seri üretimde işlem tekrarlılığının yüksek olması sebebiyle, uzmanlaşma ve kalite seviyesi yüksek, üretim maliyetleri düşüktür. Bu üretimde, üretim tesisi üretilecek ürünün işlem basamaklarına uygun olarak düzenlenir.

Seri üretim, sürekli seri (akış süreçli) ve kesikli seri (akış hatlı) olmak üzere ikiye ayrılır. Sürekli seri üretimde işlenen hammadde ve ürünler doğal yapıları itibariyle kendiliğinden akarlar. Çimento, şeker, petrokimya ve gıda maddeleri üretimi bu üretimin en önemli örnekleridir. Kesikli seri üretim de aynı temel prensip üzerine gelişmiştir. Karmaşık birimler akış kuralı kullanılarak büyük miktarlarda özel akış sistemlerinin tasarımı ile üretilmektedirler. Örneğin, beyaz eşya, motorlu taşıtlar, motor silindir blokları vb.

Kesikli seri üretim sistemleri akış hatları olarak da adlandırılır. Akış hatları, transfer hatları ve montaj hatları olmak üzere ikiye ayrılmaktadır. Transfer hatları, birbirlerine otomatik taşıma araçları ile bağlanmış büyük ve karmaşık makinelerden meydana gelirler. Ürünler, bu makineler arasında otomatik malzeme taşıma sistemi aracılığıyla hareket ettirilerek işlemleri tamamlanır. Transfer hatlarında tek bir ürün üretilebileceği gibi, farklı ürünlerin partiler halinde üretimi de yapılabilir. Bu

çalışmanın temelini oluşturan montaj hatları hakkında ayrıntılı bilgi ilerleyen bölümde verilmiştir.

2.2. Montaj Hattı ve Hat Dengeleme ile ilgili Temel Kavramlar

Montaj hatları ve hat dengelenmesi ile ilgili karşımıza çıkacak terimler aşağıda kısaca açıklanmıştır:

Montaj: Ürünler genellikle birden çok parçanın bir araya gelmesi ile oluşur. Üretim sürecinde bu parçaların belli bir sıra dahilinde birleştirilmesi *montaj* işlemi olarak tanımlanır.

Görev: Bir montaj hattında yerine getirilmesi gereken toplam işin daha fazla bölünemeyen, mantıksal olarak en küçük parçasına *görev* denilmektedir. Örneğin; bilgisayar montajında hard diskin yerine takılarak vidalarının sıkıştırılması, bu montaj işleminin çok sayıdaki en küçük parçalarından birisidir ve bir görev olarak adlandırılır.

Görev zamanı: Bir görevin tamamlanabilmesi için gereken süredir. Montaj hattı dengeleme problemlerinin çözümünde bu zamanlar sabit ve belirli (deterministik) olarak alınabileceği gibi, ortalaması ve standart sapması bilinen bir dağılıma uyduğu da (stokastik) kabul edilebilmektedir. t_i , i görevinin tamamlanma zamanını ifade etmektedir.

Toplam görev zamanı: Montaj hattı üzerinde üretilecek bir ürünün montajı için gerekli olan süre veya işi oluşturan tüm görevlerin standart tamamlanma sürelerinin toplamıdır.

İş istasyon: Montaj hattı üzerinde, bir veya daha çok görevin gerçekleştirildiği hat parçasına *iş istasyonu* denir. Her istasyonda bir işçinin, bir işlem için gerekli

araçlarla çalıştığı varsayılmaktadır. Bundan sonraki bölümlerde, *iş istasyonu* yerine kısaca *istasyon* terimi kullanılacaktır.

İstasyon zamanı: Montaj hattındaki bir istasyonda tamamlanması gerekli olan görevlerin standart tamamlanma süreleri toplamıdır. Diğer bir deyişle, istasyonda yapılması gereken ilk görevin başlangıç anı ile son görevin bitiş anı arasındaki süre farkıdır. Bir istasyon süresi, o istasyona atanan en büyük görev zamanından küçük, çevrim süresinden büyük olamaz. S_k , k istasyonunu atanan görevlerin toplam tamamlanma zamanlarıdır.

Çevrim zamanı: Montaj hattında, ürünün bir istasyonda kalabileceği en büyük süre veya bir istasyondaki işçinin, o istasyonda yapılması gerekli görevleri tamamlaması için gerekli olan süre olarak tanımlanabilir. Başka bir ifadeyle, standart bir çalışma temposuyla, bir montaj hattının ucundan ardı ardına iki ürün çıkışı arasında geçen süredir. Çevrim zamanı C ile simgelenmektedir. Bir montaj hattının çevrim zamanı, o hattı oluşturan istasyonlara ait istasyon zamanlarının en büyüğüne ($S_{\max} = \max\{S_k\}$) eşittir veya daha büyüktür. Montaj işlemindeki görev zamanlarından en büyüğü ise, çevrim zamanı için alt sınırdır (C_{alt}). Belirli bir üretim döneminde ürüne olan talebin karşılanabilmesi için, çevrim zamanının belirli bir üst sınırı ($C_{üst}$) aşmaması gerekir. Bu üst sınır aşağıdaki Eş. 2.1 kullanılarak hesaplanır:

$$C_{üst} = \frac{T}{D} (\text{dakika /adet}) \quad (2.1)$$

Burada D üretilcek miktar, T ise miktar üretiminin gerçekleştirileceği zaman dilimini ifade etmektedir. Bu sınırlar aşağıdaki Eş. 2.2'deki gibi özetlenebilir:

$$\max\{t_i\} \leq \max\{S_j\} \leq C \leq C_{üst} \quad , \quad i = 1, \dots, N \quad , \quad k = 1, \dots, K \quad (2.2)$$

Görevler bölünemez olduklarından çevrim süresi, en büyük görev süresinden küçük olamaz. Çevrim süresinin tersi, o hattın *üretim oranı* değerini verir.

İstasyon boş zamanı: Çevrim süresi ile istasyon süresi arasındaki pozitif farktır. BZ_k , k istasyonunun boş zamanı ise, aşağıdaki Eş. 2.3 yardımıyla hesaplanır:

$$BZ_k = C - S_k, \quad j = 1, \dots, K \quad (2.3)$$

İstasyon boş zamanı, o istasyonun *atıl süresini* gösterir, çünkü bu süre zarfında işçi boştaadır.

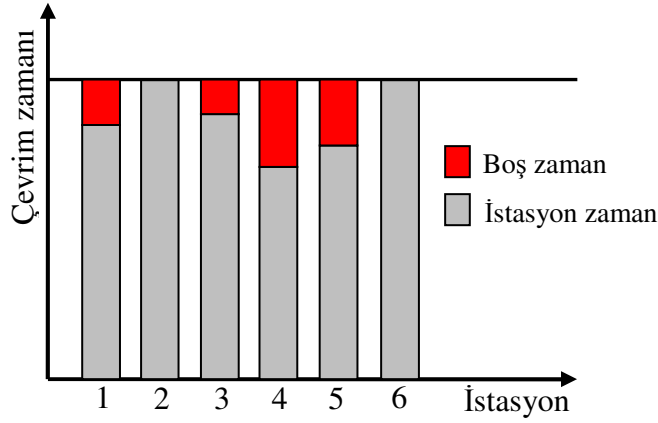
Toplam boş zaman: Montaj hattındaki istasyonların boş zamanlarının toplamı, montaj hattının *toplam boş zamanı* (TBZ) olarak ifade edilir. TBZ, aşağıdaki Eş. 2.4 yardımıyla hesaplanır:

$$TBZ = \sum_{k=1}^K (C - S_k) \quad (2.4)$$

Denge gecikmesi: Etkinliğin bir ölçüsü olan denge gecikmesi, toplam boş zamanın, ürünün hattın başından sonuna hareketi sırasında harcadığı toplam istasyonuna kötü bir şekilde dağıtılması sonucunda, bazı istasyonlardaki operatörlerin boş kaldığı süreye *denge gecikmesi* (DG) denir. Bu süre, montaj hattının dengesizliğinin de bir ölçüsü olarak kullanılır ve Eş. 2.5'e göre hesaplanır. Mükemmel dengelenmiş hatlarda denge gecikmesi sıfırdır. Şekil 2.2'de denge gecikmesi olan 6 istasyonlu bir üretim hattına ait grafik verilmiştir.

$$DG = \frac{\left(K.C - \sum_{i=1}^n t_i \right)}{K.C} \quad (2.5)$$

K , toplam istasyon sayısını, n , görev sayısını, C ise çevrim zamanını göstermektedir.



Şekil 2.2. Denge gecikmesi olan bir üretim hattı

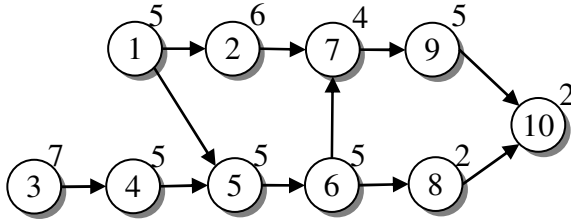
Hattın etkinliği: Verimlilik ölçütü olarak kullanılır ve Eş. 2.6'ya göre hesaplanır. Bu oran her zaman 0 ile 1 arasında değer almak durumundadır.

$$HE = \frac{\sum_{k=1}^K S_k}{K.C} \quad \text{veya} \quad HE = 1 - DG \quad (2.6)$$

Düzensizlik indeksi: Montaj hattı dengeleme problemlerinde amaç, her bir istasyonun istasyon zamanını çevrim zamanına eşit olacak şekilde düzenlemektir. Bu durumda TBZ ve doğal olarak DG sıfır olacaktır. TBZ'nin sıfıra yakın olması, beraberinde istasyonlar arasındaki iş yükü farkının da sıfıra yaklaşması anlamına gelmektedir. Fakat bu durum her zaman mümkün olmamaktadır. Düzensizlik indeksi (DI) ise, dengeleme performansının bir ölçüsü olarak kullanılır. DI , her bir istasyon zamanının en büyük istasyon zamanından ($S_{\max}$) farklarının kareleri toplamının kareköküdür. Mükemmel dengelenmiş hatlarda DI değeri sıfırdır. DI , aşağıdaki Eş. 2.7 yardımıyla hesaplanır:

$$DI = \sqrt{\sum_{k=1}^K (S_{\max} - S_k)^2} \quad (2.7)$$

Öncelik diyagramı: Bir ürünün teknik özelliklerinden dolayı montaj işlemini oluşturan görevler arasında bazı öncelik veya ardılık ilişkileri söz konusudur. Bu özelliklerin hepsi *öncelik ilişkisi* adı altında toplanır. Bu ilişkiler genellikle bir grafik ile gösterilir. Bu grafik gösterim, analitik hat dengeleme sistemlerinde öncelik ilişkilerinin belirtilmesinde çok yaygın olarak kullanılan *öncelik diyagramıdır*. Bu diyagram; bir okla birbirine bağlanmış iki görevden okun çıktığı yönde bulunanın, okun ucunda bulunan görevden daha önce işleme alınacağını gösterir. Yani, bir montaj hattındaki j görevinin yapılabilmesi için önce i görevinin yapılmış olması gerekiyorsa, i görevi j görevinin *öncülü*, j görevi ise i görevinin *ardılı*'dır denir. [Şekil 2.3](#)'de, 10 adet görevden meydana gelen bir montaj işleminin görevleri arasındaki öncelik ilişkilerini gösteren bir öncelik diyagramı görülmektedir. Dairelerin içersinde bulunan rakamlar görev numaralarını, dairelerin yanında bulunan rakamlar da bu görevlerin belirlenmiş tamamlanma zamanlarını (dakika) ifade etmektedir.



Şekil 2.3. 10 görevli örnek öncelik diyagramı

Öncelik matrisi: Bir montaj işleminde bulunan görevler arasındaki öncelik ilişkilerinin ifade edildiği, boyutu montaj işlemindeki görev sayısına eşit olan ve a_{ij} elemanlarından meydana gelen ikili (binary) bir kare matristir. Eğer öncelik diyagramındaki j görevi i görevini takip ediyorsa yani ardılı ise, öncelik matrisinin a_{ij} elemanı “1” değerini, aksi durumda “0” değerini alacaktır. Öncelik matrisi, öncelik ilişkilerinin bir diyagramı ile değil de bir matris ile gösterilmesini sağlamaktadır. [Şekil 2.3](#)'de verilen öncelik diyagramına ait öncelik matrisi aşağıdaki [şekil 2.4](#)'de görülmektedir.

	Görev									
	1	2	3	4	5	6	7	8	9	10
1	-	1	0	0	1	1	1	1	1	1
2		-	0	0	0	0	1	0	1	1
3			-	1	1	1	1	1	1	1
4				-	1	1	1	1	1	1
5					-	1	1	1	1	1
6						-	1	1	1	1
7							-	0	1	1
8								-	0	1
9									-	1
10										-

Şekil 2.4. 10 görevli örnek öncelik matrisi

Sıra kuvveti ve esneklik oranı: Bunlar, montaj hattı dengeleme problemlerinin zorluk derecesini ifade eden ölçütlerden ikisidir. Bu ölçütler, öncelik ilişkileri diyagramı ve öncelik matrisi kullanılarak, montaj hattı dengeleme problemlerindeki öncelik ilişkilerinin görelî sayısını belirler. Bu ölçütler ayrıca, problemin çözümünde elde edilebilecek uygun çözüm sayısının birer ölçüsü olup, her ikisinin de alabileceği değerler 0 ile 1 arasında değişmektedir. Esneklik Oranı (*EO*), Sıra Kuvvetinin (*SK*) tersi niteliğindedir. *EO*'nın 0'a yaklaşması durumunda *SK* değeri 1'e yaklaşacaktır. *SK* ve *EO* değerleri sırasıyla aşağıdaki Eş. 2.8 ve 2.9 kullanılarak hesaplanır:

$$SK = \frac{2 \cdot \sum_{i=1}^n |F_i|}{n \cdot (n-1)} \quad (2.8)$$

$$EO = \frac{2 \cdot Y}{n \cdot (n-1)} \quad \text{veya} \quad EO = 1 - SK \quad (2.9)$$

Eş. 2.8'deki $|F_i|$ değeri, *i* görevinin tüm ardıl görevlerinin sayısı; *n*, toplam görev sayısı; $\sum_{i=1}^n |F_i|$ ise öncelik matrisindeki “1” değerlerinin toplam sayısını ifade etmektedir. Eş. 2.9'daki *Y* değeri, öncelik matrisindeki “0” değerlerinin toplam sayısını ifade etmektedir.

Şekil 2.3'deki verilen örnek öncelik diyagramı ve aynı örneğin Şekil 2.4'deki verilen öncelik matrisi kullanılarak örnek problem için SK ve EO değerleri aşağıdaki gibi hesaplanabilir;

$$SK = \frac{2 * 36}{10 * (10 - 1)} = 0,8$$

$$EO = \frac{2 * 9}{10 * (10 - 1)} = 0,2$$

Hiçbir öncelik ilişkisi olmayan bir montaj hattı dengeleme probleminde SK değeri sıfıra, EO değeri ise 1'e eşittir. EO değerinin sıfıra yakın olması, esnekliğin az olduğunu göstermektedir. Eğer esneklik oranı sıfır ise, tüm görevler ardışık olarak sıralanmış demektir ve hatta herhangi bir esneklik sözkonusu değildir. Esneklik oranının 1'e yakın olması, problemde değerlendirilebilecek çok sayıda tasarım alternatifi oluşturacağından, problemin çözümünü de güçleştirecektir. Esneklik oranı 1 olduğunda ise görevler arasında önceliği gösteren herhangi bir ilişki yoktur, yani şebekedeki düğümler birbirlerinden kopuktur. Bu durumda hat tamamen esnektir ve montaj hattı problemi kutu paketleme (bin-packing) problemine benzer. Kutu paketleme problemi ise NP-zor sınıfında yer alan bir problemdir.

En küçük istasyon sayısı: Tüm görevlerin çevrim süresini aşmayacak şekilde istasyonlara atanması durumunda gereksinim duyulan en az istasyon sayısıdır. Hesaplamak için çeşitli yöntemler vardır. Genelde birden fazla yöntem sıra ile uygulanır, en fazla istasyon sayısını veren sonuç, gerekli en az istasyon sayısı olarak kabul edilir. Aşağıdaki Eş. 2.10 kullanılarak hesaplanabilir:

$$k_{enk} = \left\lceil \frac{\sum t_i}{C} \right\rceil \quad (2.10)$$

$\sum t_i$: Tüm görevlerin tamamlanma sürelerinin toplamı (toplam görev zamanı), $\lceil x \rceil$ ise, x 'e eşit veya x 'den büyük en küçük tamsayıyı ifade etmektedir.

Görevlerin paralelliği: Bir görevin, birden fazla istasyonda yapılmasına müsaade edilmesidir.

İstasyonların paralelliği: Hat boyunca belirli noktalarda, denk iş istasyonlarına müsaade edilmesidir. Paralellik kavramı, bir hat tasarımını çeşitli şekilde düzenlemek için kullanılır; böylece denge etkinliği geliştirilebilir ve çevrim zamanını geçen iş elamanları da rahatlatılabilir.

Bölgeleme kısıtları: Bu kısıt iki şekilde ifade edilebilir. Bunlardan ilki pozitif bölgeleme kısıtlarıdır ve burada belirli görevlerin aynı yerde veya aynı istasyonda yapılması gerekliliği vardır. İkincisi negatif bölgeleme kısıtlarıdır ki bu kısıtlar görevlerin aynı istasyonda yapılmaması istenildiğinde kullanılırlar.

İstasyon tıkanması (Blocking): Eğer istasyonda tamamlanmış birimler, yetersiz alan yüzünden stoklanamıyorsa veya bir sonraki istasyon iş yükünü tamamlamaması nedeniyle birimler sözkonusu istasyonda kalmak zorunda kalıyorsa istasyon tıkanmış demektir. Bu durumda istasyon, parçanın istasyondan çıkması için şartlar müsait olana kadar boş bekleyecektir.

İstasyon açlığı (Starving): Eğer istasyon uygunsa ve işlemi başlatmak için mevcut birime sahip değilse, istasyon açtır denilir. Bu durumda istasyon ihtiyacı olan birim gelene kadar boş bekleyecektir.

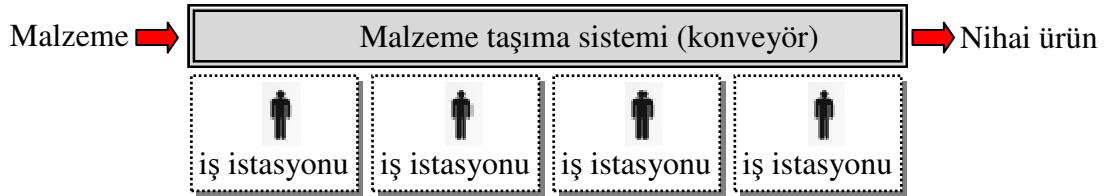
Gecikmesiz hatlar (paced): Her bir istasyona aynı zaman miktarı izin verilir. Her bir çevrim zamanından (istasyon için müsaade edilen zaman) sonra birim tamamlansın ya da tamamlanmasın bir sonraki istasyona iletilir.

Gecikmeli hatlar (unpaced): İstasyon ilgili faaliyetlerini tamamladığında, birim bir sonraki istasyona iletilir. Görev zamanlarındaki değişkenliğin çok fazla olduğu durumlarda bu hatlar tercih edilirler.

Montaj hattı dengeleme kaynaklarında sıkça kullanılmakta olan kavramlar yukarıda açıklanmıştır. Bu kavramlar ışığında aşağıda, montaj hattı dengeleme probleminin tanımlamasına ve sınıflandırılmasına yer verilmiştir.

2.3. Montaj Hattı Dengeleme

Endüstrileşme sürecinde, Taylor'un öncülüğünde, toplam işin öğelerine ayrılarak, bu parçaların ayrı ayrı işçiler tarafından yapılmasının işlemleri basitleştirdiği, üretim hızını arttırdığı ve maliyetleri düşürdüğü görüşü ortaya çıkmıştır. Buradan yola çıkılarak üretim, etrafında değişik iş istasyonlarının kurulu olduğu hareket kabiliyetine sahip bir hat üzerinden parçaların geçirilmesi yoluyla yapılır. Malzemelerin akış hattı boyunca işgücü veya otomatik malzeme taşıma sistemlerinden yararlanılarak transfer edildiği, işgücü veya robotlar sayesinde parça üzerindeki işlemlerin aralarındaki öncelik ilişkileri gibi kısıtları gözönüne alarak gerçekleştirildiği, hat boyunca sıralanmış iş istasyonlarından oluşan sisteme *montaj hattı* adı verilir. [Şekil 2.5](#)'de örnek bir montaj hattı şeması verilmiştir.



Şekil 2.5. Örnek bir montaj hattı şeması

Montaj hatları yapıları itibari ile işlemleri yapacak çalışanların veya makinelerin yer değiştirdiği hatlardan meydana gelebileceği gibi, montaj işleminin elektrikli bir motor tarafından ilerleyen bir bant üzerinde yapıldığı konveyör bantlardan da meydana gelebilir. İş istasyonunda bulunan çalışanlar veya robotlar, hareketli bant üzerinde gelen yarı mamulü gerek hat üzerinde gerekse hattın kısa bir süre alarak, kendilerine ilişkin görevleriyle ilgili bir veya birkaç işlemi ürüne uygularlar. Bu süreç hat boyunca uygulanarak adım adım tamamlanma aşamasına gelen ürün hat sonunda tamamlanmış olarak hattın çıkar. İşletmeler montaj hatlarında mamulün konveyör

sayesinde taşınmasından, taşıma esnasındaki meydana gelebilecek çizilme, kırılma, deforme olma gibi etkilerden büyük ölçüde tasarruf sağlar.

Montaj hatları üretimin yığın olarak gerçekleştirildiği işletmelerin üretim sistemlerinin önemli ve vazgeçilmez bir parçasıdır. Bir veya birkaç ürün için yapılacak montaj hattı üretimi tasarlandığında, üretim hattındaki iş istasyonlarına ilişkin işlem sürelerinin dengelenmesi problemi ortaya çıkar. Bundaki amaç, kurulan montaj hattının verimli olarak çalışabilmesi için; üretim süresi içinde her bir işçiye, montaj hattında çok az veya hiç boş süre bırakılmayacak şekilde işlemlerin istasyonlara dağıtılması, yani varolan kısıtlar altında, işlem sayısının çok ve üretim hızının yüksek olması nedeniyle, iş istasyonları arasındaki işlem süresi farkları toplamının enküçük olmasıdır. Problemin bu noktasında, sürekli üretim yapan sistemlerin yerleştirme düzeninin kurulmasında, hat dengeleme problemi ortaya çıkar.

Denge bir üretim veya montaj hattının tasarımıdaki en temel konudur. Montaj hattı tasarlandığında veya dengelenmek istendiğinde hatta yapılması gerekli olan işlerin işlem sürelerinin eşit olmadığı görülür. Bu da çalışmanın konusunu oluşturan problemin ortaya çıkmasına neden olur. Problemin çıkış noktasından da anlaşılabileceği gibi buradaki amaç farklı sürelerle sahip işleri istenilen üretim düzeyi veya imkanları dahilinde istenilen iş istasyonu sayısı çerçevesinde, iş istasyonları arasında mümkün olan en düşük süre farkı ile hattın verimli halde çalışırılığını sağlayabilmektir.

Montaj hatlarındaki verimlilik artışı, kapasiteyi artıracak ve maliyeti azaltacak için oldukça önem arz etmektedir. Montaj işlemlerinin yapılabilmesi için gerekli görevlerin, montaj sürecindeki kısıtlara uyarak (öncelik ilişkilerini ihlal etmeden ve çevrim zamanını geçmeden), bir veya daha fazla amacı en iyileyecek şekilde, sıralı iş istasyonlarına atanması *Montaj Hattı Dengeleme* (MHD) problemi olarak adlandırılmaktadır. Görüldüğü gibi hat dengeleme problemleri, üretim hatlarının tasarımında ve işletilmesinde her an ortaya çıkabilen bir konudur. Gerçekte hat dengeleme problemleri, çeşitli sızgelerde ve üretim hızlarında çalışıldığında, hatta

oluşabilecek ve istasyonların verimliliğini etkileyecek başlıca etkenlerden olan atıl kaynak kullanımını en alt düzeyde tutmak için en uygun çözümün atanmasında ortaya çıkmıştır.

Üretim hızının artırılması, sağlıklı bir tasarlama yapılması ve işletmenin ekonomik sorunlarına çözüm getirmeye yönelik olmasından dolayı, montaj hattı dengeleme konusu, endüstri dünyasında büyük önem taşır. Kaliteli ve yüksek hızlı bir üretim düzeyi tutturmak, gerçekte hat dengeleme çalışmalarının detaylı ve iyi bir şekilde yapılması ile sağlanır.

Eğer üretilen mamulün işgücü maliyeti yüksekse öncelikle işgücünün maksimum kullanımı gereklidir. Bununla beraber makinelerin oluşturduğu maliyet çok yüksekse amaç, makine kapasitelerinden en iyi şekilde yararlanmak olacaktır. İşlem için minimum zaman ve maliyet değerleri hat dengeleme ile gerçekleştirilecektir.

Montaj hatları dengelenmesi konusu statik olmayan taleplerinin etkin olduğu günümüzde işletmelerde değişen üretim hızının verimli biçimde işletmeye adapte edilmesine de olanak sağlayarak montaj hattını barındıran bir çok sektörde önem taşır. Ayrıca üretim hattı dengelenirken çalışanların yaptıkları iş yükü de dengelendiği için yüksek üretim miktarı ile birlikte ürünlerde kaliteyi de olumlu yönde etkileyecektir.

Montaj hatları boyunca bulunan iş istasyonlarına işler eşit biçimde dağıtılacağı için işgücü dengelenmesi yapılmış olur, bu da çalışanlar arasından meydana gelebilecek huzursuzlukları ve motivasyon düşüklüğünü engeller. Ayrıca işlerin sabit istasyonlarda yapılması yapılan iş üstünde uzmanlaşmayı doğuracak buda toplam üretim miktarında bir artış meydana getirecektir.

2.4. Montaj Hattı Dengelemenin Amaçları

Montaj hattı dengelemenin ana amacı, oluşan şartlar altında montaj hattının sürekli olarak çalışmasını sağlamaktır. Bu işlerin, çevrim zamanında tamamlanması ile

mümkündür. İstasyon sayısının sabit olduğu problemlerde amaç çevrim zamanını enküçükmektir. Hat dengeleme çalışmalarında ulaşmak istenen amaçlar şunlardır: [Erkut ve Baksak, 1997].

1. Düzenli bir malzeme akışını sağlamak.
2. İnsan gücü kullanımını en üst düzeye ulaştırmak.
3. Makine kapasitelerini en üst düzeyde kullanmak.
4. İşlemler için en az süreyi kullanmak.
5. İşlemler için en az miktarda malzeme kullanmak.
6. Boş zamanları veya dengeleme kayıplarını enküçükmek.
7. İş istasyonu sayısını enküçükmek.
8. Denge kayıplarını, iş istasyonları arasında düzgün şekilde dağıtmak.
9. Hat dengeleme maliyetini enküçükmek.

Hat dengeleme çalışmasında ulaşmak istenen bazı amaçlar birbirleriyle çelişebilmektedir. Bu tip durumlarda hepsini göz önünde bulundurarak en büyük faydayı sağlayacak şekilde bir dengeleme çalışması yapılmalıdır.

Montaj hattı dengelemenin amaçları birbiriyle çeliştiklerinden, hepsinin birden en üst düzeye ulaştırmak, olası olmayabilir. Dengelemede ana amaç, bu çelişkilerin gözönüne alınarak en uygun çözüme ulaşılmasıdır. Bu yapılırken, mantaj maliyetinin de enküçük olması sağlanmalıdır. Dengeleme işleminde hesaba katılan etmenler içinde, maliyet etkisi olan tek değişken, işgücüdür.

2.5. Montaj Hattı Dengelemenin Temel Prensipleri

Montaj hattının düzenlenmesi aşağıdaki prensiplerin en iyi uyumuna göre yapılır [Wild, 1972]:

İşgücü prensibi: Montaj hattında ne kadarlık bir işgücüne veya işçiye gereksinim vardır, bunlar nerelerde ve ne kadar bulundurulacaktır sorularına cevap aranır.

İş akışı prensibi: İşlemler belli bir iş akışını sağlamalıdır. Bu durum süreklilik açısından da önemlidir.

En küçük hareket uzaklığı prensibi: Montaj hattı üzerinde taşıma mesafelerinin oluşturacağı maliyet, darboğaz ve depolama açısından en küçük uzaklık olmalıdır.

Sabit yol prensibi: İşlemler son aşamaya kadar aynı yol üzerinde hareket etmelidir.

İşlemlerin eş zamanlı sürdürülmesi prensibi: Montaj işlemleri ilk istasyondan son istasyona kadar eş zamanlı yapılır.

Parçaların değişebilirliği prensibi: Montaj işlemi sırasında bazı parçalar işlem sürekliliğini bozmadan değiştirilebilmelidir.

Birim işlem prensibi: Ürün üzerinde her işlem, en küçük işlem birimlerine ayrılmıştır.

En küçük işlem süresi: Montaj işlemi, mümkün olan en kısa sürede bitirilmelidir. Burada işlem süreleri sabit olduğuna göre en küçükleme kayıp zaman üzerinde yapılır.

2.6. Montaj Hattı Dengelenmesini Etkileyen Temel Etmenler ve Kısıtlar

Temel etmenler

- Mühendislik spesifikasyonları
- İşin yapılmasında izlenen yöntem (Bazı montaj işlerinde verilen belirli bir teknolojik sıra, işin daha kolay ve daha hızlı yapılmasını sağlayabilir)
- Kullanılan aygıt ve makineler (bazı aletlerin, montaj hattının birden çok yerinde kullanılmasına gerek duyulabilir; böyle durumlarda, aynı alet setini, birden fazla işlemcinin kullanabileceği bir şekilde, bazı istasyonları ardarda yerleştirmek gerekebilir)

Montaj hatlarının dengelenmesini etkileyen kısıtları birincil ve ikincil kısıtlar olarak ikiye ayırabiliriz.

Birincil kısıtlar

Çevrim süresi: İşletmenin planlanmış yıllık, aylık, haftalık veya günlük üretim düzeyi çerçevesinde belirlenmiş değer istasyonlara atanacak olan işlemlerin toplam süresinin sınırını teşkil eder. Diğer bir deyişle, ürünün bir istasyonuna girdiği süre ile çıktığı süre arasındaki farklıdır. Eğer bu süre aşılır ise hat dengeleme sonucun elde etmek istediğimiz sonuçların dışına çıkmış oluruz. Böylelikle hedeflerden sapılmış olur. Bir istasyona atanan etkinliklerin süreleri toplamı, çevrim süresini aşamaz.

Öncelik ilişkileri: Ürünün montaj aşamasında parçalara ayrılan işi ürünün bütünlük içinde tamamlanabilmesi için belirli bir sıra dahilinde birleştirilmesi gereklidir. Dolayısıyla ile istasyonlara işlerin atanması yapılırken öncelik ilişkisi dahilinde atama yapılmalıdır.

İkincil kısıtlar

Konum kısıtı: Bu kısıt bantta işlem yapan çalışanların bant üzerindeki ürüne, ürünün yapısından ve hattın durumundan kaynaklanan işçilerin ürüne istedikleri gibi müdahale edememelerinden kaynaklanır. Bu kısıt üretim hattında büyük hacimli ürünler işlem gören montaj hatlarında ortaya çıkar. Örneğin otomobil imalatı yapan bir bantta ki çalışanın otomobilin diğer yanına müdahale etmek istediğinde şartlar gereği bunu zor gerçekleştirecektir veya gerçekleştirse bile uzun zaman kaybına uğrayacaktır. Bunun önüne geçebilmek için ise diğer tarafa da çalışan yerleştirilebilir.

Montaj işlemleri esnasında bazı ağır, yüksek veya büyük hacimli nesnelerin alt veya üst kısmında işlem yapılmak istenebilir, bu durumda işlem yapılmak istenen noktalara erişmek oldukça zordur. Böyle durumlarda karşılaşıldığında işlemin yapıldığı istasyona gerekli düzenekler kurularak ya ürün ters çevrilir veya bir asansör

yardımı ile çalışan yapmak istediği noktaya çıkartılarak işlemleri yapılabilirliği sağlanır. Tahmin edilebileceği gibi her istasyona bu işlemi gerektirir işleri aynı iş istasyonlarında toplamak daha mantıklı olacaktır.

Sabit donanım kısıtı: Montaj işleminin gerçekleştirilmesinde kullanılan belirli hacme sahip makineler ve diğer ekipmanlar yapı gereği sabit olabilirler. Bu durumda hattın dengelenmesi durumunda yerleri değiştirilemez.

İstasyon yükü: Montaj hattı dizayn edilirken veya tekrar dengelenmek istenildiğine hattaki bazı istasyonların montaj sürelerinin, çevrim süresinden düşük olması istenebilir. Bu durum genelde muhtemel aksaklıkların yaşanabileceği istasyonlarda istenir. Böylece bu istasyonlarda bir aksaklık olsa da çevrim süresine erişmediği için toplam üretim miktarında bir sapma meydana getirmeyecektir.

Aynı istasyona atanması istenen görevler: Montaj hattında gerçekleştirilen görevlerden bazılarının aynı istasyonda uygulanması istenebilir. Genelde aynı alet veya aygıtın kullanıldığı görevlerde bu durum geçerli olur. Böyle bir durumda bu görevler tek bir görev olarak düşünülebilir.

Aynı istasyona atanmaması istenen görevler: İşçiye gözetmek veya işin sağlıklı yapılmasını sağlamak gibi çeşitli sebeplerden ötürü bazı görevlerin aynı istasyona atanmaları istenmeyebilir. Öncelik ilişkisi açısından bir engel olmasa bile bu görevlerin ayrı istasyonlara atanması gerekebilir. Bu durumun istenmesinin nedeni ise işler arasındaki uyumsuzluk veya işgücüne aşırı yüklenmedir. Örneğin çok ağır koşulları olan iki işi tek bir istasyona koymak buradaki çalışanı olumsuz etkileyecektir.

2.7. Montaj Hatlarının Sınıflandırılması

Montaj Hatları birçok şekilde sınıflandırılabilmektedir. Model sayısına göre; *tek modelli, çok modelli, karışık modelli*, hattın kontrol yapısına göre; *gecikmeli ve gecikmesiz*, otomasyon seviyesine göre; *manual ve otomatik* ve hat yerleşim tipine

göre; *geleneksel (düz), U tipi*. Montaj hatların sınıflandırılma şekilleri [çizelge 2.1](#)'de özetlenmiştir.

Çizelge 2.1. Montaj hatların sınıflandırılması

Sınıflandırma faktörleri	Hat çeşitleri		
Model sayısına göre	Tek modelli	Karışık modelli	Çok modelli
Kontrol yapısına göre	Gecikmeli	Gecikmesiz	
Otomasyon seviyesine göre	Manuel	Otomatik	
Yerleşim şekline göre	Düz	U tipi	Paralel

2.7.1. Model sayısına göre montaj hatları

Üretim hattında bir model ya da birden fazla ürünün üretilmesine göre montaj hatları; tek modelli, çok modelli ve karışık modelli hatlar olmak üzere üçe ayrılmaktadır. Bir montaj hattının sadece bir ürün modelinin üretimi için kullanılması sözkonusu olabilir. Bu tip montaj hatlarına *tek modelli montaj hatları* adı verilir. Bir ürün modelinin üretimi tamamlandıktan sonra, aynı montaj hattında farklı bir modelin üretimi de gerçekleştirilebilir. Bu tip montaj hatlarına ise *çok modelli montaj hatları* denir. Bununla birlikte, bir montaj hattı aynı anda birden fazla modelin üretimini gerçekleştirecek esnekliğe sahip olabilir. Bu tip montaj hatlarına ise *karışık modelli montaj hatları* adı verilir.

Tek modelli hatlar: Tek tip ürün ya da modelin sürekli olarak ve yüksek miktarlarda üretildiği hatlardır. Ürün çeşitliliğinin artması nedeni ile günümüzde tek modelli montaj hatları ile pek karşılaşmamaktadır. Ürün grupları arasında çok fazla farklılık olmayan ve partiler arasında hazırlık zamanına göreceli olarak daha az ihtiyaç duyan üretim hatları da tek modelli montaj hatlarına dahil edilmektedir. [Şekil 2.6 \(a\)](#)'da tek modelli hatlar gösterilmektedir.

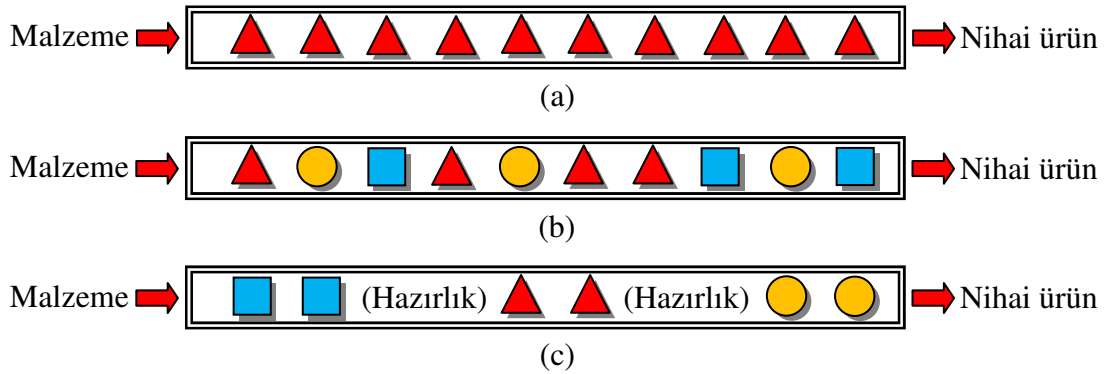
Karışık modelli hatlar: Karışık modelli hatlar aynı anda birden fazla benzer tipteki modellerin karışık olarak üretildiği hatlardır. Karışık modelli üretimin en önemli faydası, müşteri isteklerinin anında karşılanması ve bu nedenle bitmiş ürün

stoklarının daha az olmasıdır. Karışık modellenli hatlarda gözlenen temel olumsuzluk, modellerin özelliğinden kaynaklanan, ayrı iş parçalarının; eşit olmayan iş akışlarına, boş istasyon sürelerine ve daha fazla istasyon sayısına neden olmasıdır. Bu tip problemlerde dengeleme problemi oldukça zordur. Bu nedenle istasyonlarda daha fazla istasyon boş zamanları oluşmakta ve ara stoklar meydana gelebilmektedir.

Tek modellenli hatlardan farklı olarak karışık modellenli hatlarda, kısa dönem model sıralama ve çizelgeleme problemi ortaya çıkmaktadır. Burada dengeleme problemiyle aynı anda ortaya çıkan sıralama ve çizelgeleme probleminde, üretilecek modellerin kendi içerisinde sıralarını belirlerken, hattın duruş zamanını, işçilerin verimliliğini, üretim hızını ve her istasyondaki çevrim zamanı kısıtlarını da dikkate almak gerekmektedir.

Karışık modellenli montaj hatlarının kullanıldığı sektörler e örnek olarak otomotiv, beyaz eşya vb. sektörler verilebilir. Örneğın Mercedes firmasının ürettiğı C sınıfı modeli için 2²⁷ farklı spesifikasyon söz konusudur. Bu kadar değışkenliğın söz konusu olduğı modeller için öncelik diyagramlarını oluşturmak ve hattı dengelemek oldukça karmaşık bir problemdir. Karışık modellenli hatlarda dengeleme yapılırken tüm modeller için hazırlanan öncelik diyagramları birleştirilerek birleşik öncelik diyagramları oluşturulmaktadır ve bu birleşik öncelik diyagramına göre hat dengelemesi yapılmaktadır. Şekil 2.6 (b)'de karışık modellenli hatlar gösterilmektedir.

Çok modellenli hatlar: Çok modellenli hat üzerinde, birden fazla modelin üretimi gerçekleştirilir. Modellerin üretimi partiler halinde yapılır. Yani üretim yapılırken modellerin karıştırılmasına izin verilmez. Çok modellenli hatlarda dengeleme yapılırken, hazırlık zamanı ve hazırlık maliyetlerinin dikkate alınması gerekir. Bu nedenle hattın dengelenmesi öncesinde hazırlık maliyetlerini dikkate alarak parti büyüklüğünün belirlenmesi gerekmektedir. Çok modellenli montaj hatları, eğer yığınlar büyük ise tek modellenli montaj hatlarına, küçük ise karışık modellenli montaj hatlarına benzerlik gösterirler. Şekil 2.6 (c)'de çok modellenli hatlar gösterilmektedir.



Şekil 2.6. (a) Tek modelli (b) Karışık modelli (c) Çok modelli hatlar

2.7.2. Hattın kontrol yapısına göre montaj hatları

Hattın kontrol yapısına göre montaj hatları gecikmesiz ve gecikmeli hatlar olmak üzere iki sınıfa ayrılmaktadır.

Gecikmesiz hatlar: Gecikmesiz hatta, her istasyona her ürün birimini üretmek için eşit zaman miktarı verilir. Bu çevrim zamanı sonunda sistem otomatik olarak yeni istasyona geçildiğini varsayar. Gecikmesiz hatlarda, belli bir çevrim süresi verilerek tüm istasyonların bu kısıta uyacak şekilde görev zamanlarını ayarlanması istenmektedir. Konveyörler ya da hareketli bantlar gibi malzeme taşıma ekipmanları istasyonları birbirine bağlar. İş parçaları bu bantlar üzerinde bir istasyondan diğerine sabit hızda hareket eder veya işlendikten sonra transfer edilirler. Her iki durumda da her istasyona görevlerini yerine getirmesi için aynı miktarda süre verilir. İş parçası istasyondan geçerken operatör de onunla birlikte hareket eder, işlemi gerçekleştirir ve istasyonun başlangıç noktasına geri döner.

Gerçek hayatta karşımıza çıkan problemler, insan faktöründen dolayı genellikle stokastik işlem zamanlı olduğu için, dengenin tam anlamıyla sağlanabilmesinde gecikmeli hatlar tasarlamak daha doğru olmaktadır. İş parçası bir istasyondan diğerine geçirilirken, gecikmeleri önlemek her zaman mümkün olmadığından, görevlerin zamanında tamamlanamama olasılığı da dikkate alınmalıdır.

Gecikmeli hatlar: Gecikmeli hatlarda, gerekli olduğunda iş istasyonlarının sürelerinin çevrim süresini aşmasına izin verilmektedir. Bu durumda yavaş çalışan iş istasyonlarının önlerinde ara stoklar bulunmaktadır. Tampon, takip eden istasyonda bir önceki işlem devam ederken iş parçasının tutulduğu yer olarak tanımlanır. Tampon kapasiteleri kısıtlı olduğundan, eğer takip eden tampon tam doluysa istasyon tıkanır. Bu durumda istasyon, tamponda alan müsait olana kadar boş bekleyecektir. Başka bir olumsuz durum ise istasyon açlığıdır. Bu durumda da istasyon işleme hazır olduğu halde işlem yapacak parçanın bulunmaması yani tampona parça girene kadar istasyonun boş beklemesidir.

2.7.3. Otomasyon seviyesine göre montaj hatları

Otomasyon seviyesine göre montaj hatları, *manuel hatlar* ve *otomatik hatlar* olarak üzere ikiye ayrılmaktadır.

Manuel hatlar: Günümüzde gelişen teknolojinin etkisiyle otomasyon ve robotların montaj hatlarında kullanılması artsa da, hala pek çok işletme temelde işgücüne dayalı montaj hatlarını kullanmaktadır. Manuel hatlar, her istasyonda toplam iş yükünün bir bölümünün bir veya daha çok işçi tarafından yapıldığı hatlardır. Manuel hatlar özellikle kırılgan ya da hassas parça üreten firmalar tarafından tercih edilmektedir. Ayrıca, işgücü maliyetlerinin makine maliyetlerine göre daha düşük olduğu durumlarda, manuel hatlar tercih edilmektedir.

Manuel hatlarda montaj hattı dengeleme çalışmasına giderken dikkat edilmesi gereken temel husus, görev zamanlarının işgücü performansındaki değişimleri dikkate alarak stokastik olmasıdır. Motivasyon, iş çevresindeki koşullar, fiziksel ya da zihinsel stres çalışanın performansını etkileyebilmekte, aynı görevi yapma zamanı değişkenlik gösterebilmektedir. Özellikle sık tekrar eden görevler, çalışanda bir bıkkınlık hissi uyandırmakta, bu da manuel hatlar için büyük bir dezavantaj oluşturmaktadır.

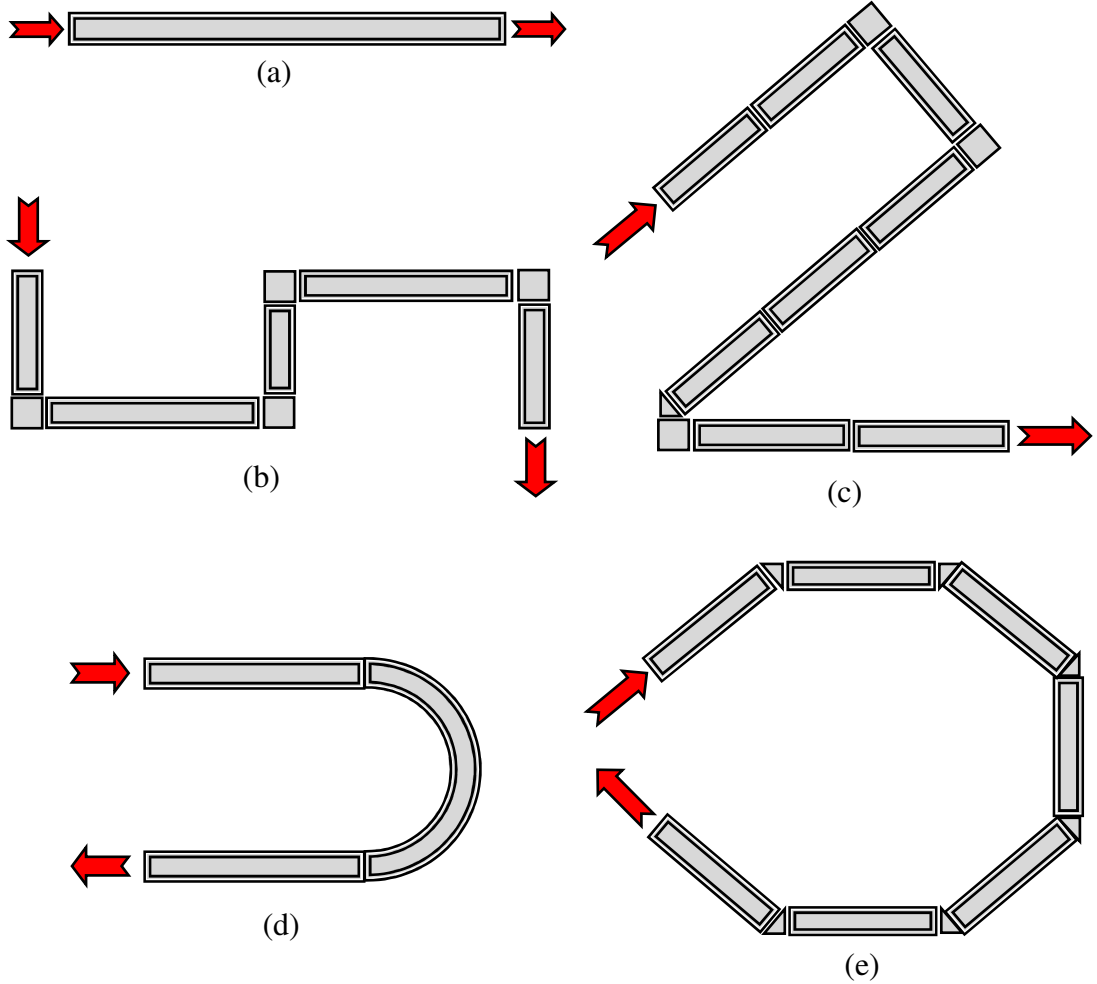
Manuel hatlarda işgücünü etkileyen diğer bir faktör de çalışanın deneyimidir. Çalışanlar zaman içinde deneyimlerini arttırarak, işin yapılışında pratiklik kazanmaktadır. Böylece, görev zamanları azalacaktır. Burada, hat dengelemesi yapılırken öğrenme etkisini de dikkate alarak, görev zamanları dinamik kabul edilen çalışmalar yapılmalıdır.

Otomatik hatlar: İstasyonlardaki görevler ve istasyonlar arası transferler otomatik olarak yapılmaktadır. İstasyonlar arası malzeme transferinin mekanik ve mekanik olmayan hatlar şeklinde iki yolu vardır. Mekanik olmayan hatlarda, parçalar bir istasyondan diğerine elle geçirilir iken, mekanik hatlarda ise hareketli konveyörler ve benzeri malzeme taşıma sistemleri kullanılmaktadır. Çalışma koşullarının çalışanlar üzerinde ciddi sağlık sorunlarına yol açabileceği iş çevreleri için tasarlanan otomasyon hatları, önceleri otomotiv sektöründe kullanılmıştır. Daha sonra pek çok ürünün üretiminde otomatik hatlar kullanılmıştır. Özellikle işgücü maliyetinin yüksek olduğu durumlarda iş sahipleri, tesis tasarım aşamasında otomasyon hatlarını ve robotları kullanmayı tercih etmektedir. Otomasyonun montaj hattı dengeleme problemine sağladığı en büyük avantaj, görev zamanlarındaki değişkenliğin çok az olmasıdır. Bu nedenle manuel hatlarda yaşanan gecikme ve boş bekleme zamanları, bu hatlarda daha az oluşmaktadır.

2.7.4. Yerleşim şekline göre montaj hatları

Montajı yapılacak ürüne bağlı olarak montaj hattının şekli ve buna bağlı iş istasyonlarının yerleşim biçimi farklılık gösterir. Bunun yanında üretimin yapılacağı alan, kurulum maliyeti ve montaj yapacak makinelerin konumu montaj hatlarının yerleşim şekillerinde önemli bir role sahiptir. Örneğin işletmenin kapalı alanının yetersiz olduğu durumlarda üretim hattının fabrika içindeki boş yerlere doğru kıvrımlar çizdirilerek ilerletilmesi sağlanır. Kaynaklarda da hattın ilerleyiş yolunun çizmiş olduğu eğri ve doğruluğuna göre isimler verilerek montaj hatlarına biçimsel birçok isimler verilmiştir: düz, dairesel, değişik açılı, U-şekilli ve zig-zag vb. [Baskak, 2005]. Fiziksel montaj hattı tipleri [şekil 2.7](#)'de gösterilmiştir. Montaj hattı

dengeleme kaynaklarında konu olmuş ve en çok kullanılan montaj hattı yerleşim şekilleri hakkında ilerleyen bölümlerde ayrıntılı bilgi verilmiştir.

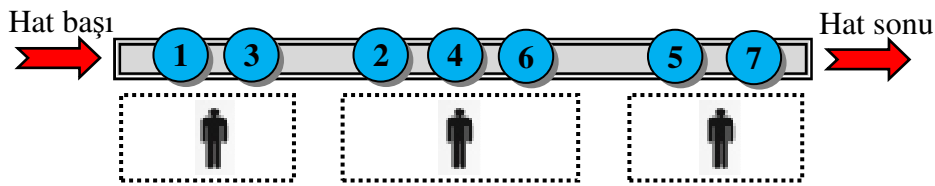


Şekil 2.7. Fiziksel montaj hattı tipleri: (a) düz hat (b) zig-zag hat (c) değişik açılı (d) U tipi hat (e) dairesel hat

Düz montaj hatları

Düz montaj hatları, bir doğru boyunca birbiri ardına sıralanmış iş istasyonlarından oluşan montaj hatlarıdır. [Şekil 2.8](#)'de düz hat yerleşimi gösterilmektedir. Üretim sistemlerinde montaj hatları ilk olarak düz montaj hatları olarak ortaya çıkmıştır. Montajı yapılacak ürün ilk istasyondan girer ve son istasyondan montajı bitmiş bir şekilde çıkar. Düz hatların birtakım üstün yanları olduğu gibi sakıncaları da vardır. İş

akışının daha hızlı ve kolay olması, basit ve sistematik olması, kolayca yerleşim yapılabilmesi, servis verme olanaklarının kolaylıkla sağlanabilmesi, konveyör sistemlerinin uygulanabilirliğinin artması ve maliyetlerinin düşmesi, ayrıca köşelerde meydana gelebilecek transfer zorluklarının ortadan kalkması gibi nedenlerle, montaj ve üretim hatlarının yerleşiminde düz hatlar tercih edilir. Bunların yanında, çok yer kaplaması ve işçilerin birbirleriyle olan iletişim olanağının az olması ise düz hatların sakıncaları arasında gösterilebilir.



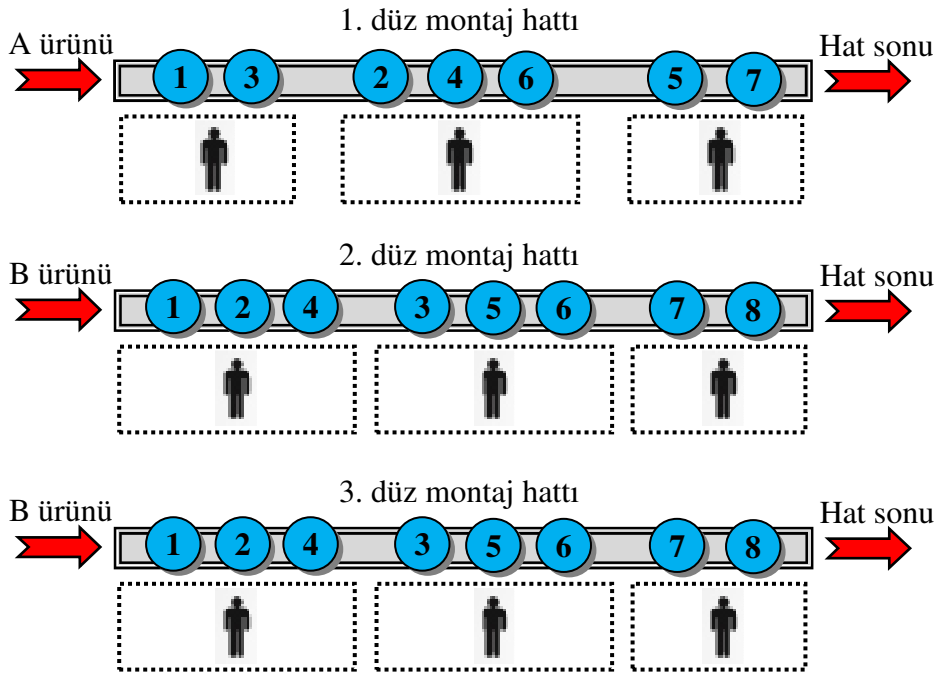
Şekil 2.8. Düz hat yerleşimi

Üretimde sağladığı avantajlar ve basitliği sayesinde üretim bantlarının doğrusal olması istenir. Fakat yukarıda da belirtilen bazı nedenlerden dolayı hatların şekillerinin doğrudan farklı şekillerde olmasının nedenlerini söyle sıralayabiliriz: Eğer üretim yapılacak mamul birçok parçadan oluşuyorsa buna bağlı olarak montaj hattının boyuda uzun olacaktır. Dolayısıyla böyle bir hattın kapalı bir fabrika altında doğru biçiminde yerleştirmek imkansız olacaktır. Bu yüzden hattın bazı noktalarında dönüşlerin olması kaçınılmaz olur. Diğer yandan fabrikaya düz bir hat konulduğunda fabrikanın diğer yerlerinde aşırı boşluklar kalıyorsa, düz bir montaj hattı başarılı bir yerleşim olmaz. Fabrikanın sahip olduğu makinelerin verimli bir şekilde kullanılmak istenmesi de montaj hattının yerleşim şeklini etkiler. Doğrusal bir hatta makineden sadece bir noktadan faydalanırken, u-tipi bir tasarımda makinenin yerleşiminden yararlanılarak makine daha verimli bir biçimde kullanılır.

Paralel düz montaj hatları

Sanayide çoğu üretim sistemleri bir ya da daha fazla düz montaj hatlarından oluşmaktadır. Bu düz montaj hatları fabrika içinde birbirine paralel olarak yerleştirilmektedir. Ürünleri bir ya da daha fazla düz montaj hattında üretmede iki

durum söz konusudur. Birinci durum talebin yeterince yüksek olması ve bir hattın kapasitesinin söz konusu talebi karşılamada yetersiz kalması ve ikinci bir hattın kurulması ihtiyacıdır. Yani, aynı ürünün birden fazla özdeş hatta üretilmesidir. İkincisi durum ise her bir ürün talebinin, hat kurmak için yeterli büyüklükte olması ve birden fazla benzer ürünü birbirinden ayrı montaj hatlarında üretilmesidir. Şekil 2.9'da birbirine paralel olarak konumlandırılmış 3 tane düz montaj hattına sahip üretim sistemi görülmektedir. Bu paralel hatların aynı yada farklı ürünleri aynı zamanda bağımsızca üretmeleri çok sık rastlanan bir olaydır.



Şekil 2.9. Paralel düz hat yerleşimi

Paralel düz montaj hatları, birbirinin aynısı olabileceği gibi, farklı da olabilirler. Eğer yukarıda anlatılan birinci durum söz konusu ise, yani talebi karşılamada bir hat yetersiz kalıyorsa, kurulacak ikinci veya üçüncü düz hatların aynı ürünün montaj işlemini yapacağı için birbirinin aynısı olması gerekir. Diğer taraftan ikinci durum söz konusu ise, farklı ürünlerin montajı yapılacağı için hatların farklı olması doğaldır. Şekil 2.9'da görüldüğü gibi, birbirine paralel olan 3 düz montaj hattında A ve B ürünleri üretilmektedir. Birinci düz montaj hattında, A ürünü üretilirken, ikinci ve üçüncü düz montaj hatlarında B ürünü üretilmektedir. İkinci ve üçüncü hatlarda

aynı ürünlerin montajı yapıldığı için, üretim şeklide aynıdır. Başka bir ifadeyle, dengeleme çalışması sonucunda istasyonlara atanan görevler ve görevlerin yapılış şekilleri aynıdır.

Paralel montaj hatlarının geleneksel düz montaj hattı ile kıyaslandığında çeşitli avantajları mevcuttur: (i) aynı ürün veya benzer ürünlerin birbirine paralel bitişik hatlarda üretilmesini sağlar, (ii) ortak kaynak kullanımına izin vererek boş zamanları azaltarak hatların verimliliğini yükseltir, (iii) her bir montaj hattı için değişik çevrim süreleriyle üretim olanağı sağlar, (iv) operatörler arasındaki iletişim ve gözlenebilirlik olanaklarını artırır, (v) operatör gereksinimlerini azaltır, (vi) alet ve ekipman maliyetlerini azaltırlar.

Montaj hattı kaynaklarında, bu tür durumlar için, her bir hat ile ilgili bağımsız dengeler oluşturulmakta ve üretim gerçekleştirilmektedir. [Gökçen ve ark. \(2006\)](#), birden fazla özdeş ya da benzer hattın bulunması durumunda, paralel hatların birlikte dengelenmesi için yeni bir felsefe önermişlerdir. Montaj hatların eş zamanlı olarak dengelenmesi konusunda daha ayrıntılı bilgi [bölüm 3.1](#)'de verilmiştir. Bu tezde, paralel hatların eş zamanlı dengelenmesi için işgücü maliyetlerini dikkate alan yeni bir felsefe ortaya konulmakta ve bu doğrultuda yeni prosedürler önerilmektedir.

U tipi montaj hatları

Küreselleşme ve beraberindeki değişimler üretim sistemlerini de etkilemiştir. Günümüz üretim sistemleri, olabildiğince esnek yapıya ihtiyaç duymaktadır. Bunun nedenleri olarak ürün çeşitliliğinin artması, kitlesel talebin azalması, rekabetin artması, verimliliğin rekabette önemli bir faktör haline gelmesi sayılabilir. Son yıllarda, işletmelerin tam zamanında üretim ilkesini uygulamaya başlaması ile birlikte, montaj hatları günümüz üretim ihtiyaçlarını karşılayacak şekilde değişime uğramış ve sonuç olarak geleneksel düz hatların yerini U-tipi hatlar almaya başlamıştır.

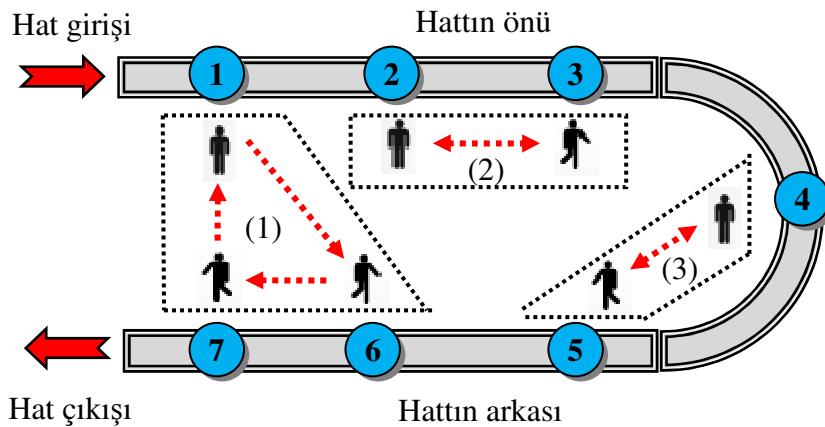
Yalın üretim yaklaşımında bir fabrika veya atölyenin işleyişinde olabilecek israflar, çalışanların bir yerden bir yere gitme, makinelerin çalışmasını kontrol etme ya da makine başında makinenin işleminin bitmesini bekleme gibi üretime değer katmayan eylemlerden doğan zaman kayıplarıdır. Makinelerin doğru çalışıp çalışmadığının kontrolü, makineye parça yerleştirme, işlenmiş parçayı alma gibi eylemleri mekanikleştirerek ve otomatikleştirerek, kazanılan zaman her işçinin birden fazla makineyi çalıştırması şeklinde değerlendirilebilir. Böylece bir yandan aynı işi çok daha az sayıda işçiyle gerçekleştirmek mümkün olabilir, diğer yandan da talep yükselme ve düşme durumlarında sadece işçi sayısı ile oynanarak üretim verimini talepteki esnekliğe adapte etme olanağı elde edilebilir. Bu düşünceden hareketle, bir işçinin birden fazla makineden sorumlu olması ilkesi, tek parça akışı (parçanın süreçler arasında beklemeden teker teker aktarılması) ve süreç bazlı hat (bir parçanın son halini alması için gereken tüm makinelerin, parçaların işlenme akışına dayanarak birbiri ardı sıra yerleştirilmeleri) anlayışıyla da birleşince ortaya çıkan yerleşim düzeni U hatları olmuştur.

U tipi montaj hatları, ilk kez Toyota fabrikasında uygulanmıştır. Firma farklı özelliklere sahip aynı türden çeşitli otomobiller üretmektedir ve her bir araba için talep, sürekli dalgalanma göstermektedir. Bu nedenle, tesiste bulunan her bir atölyedeki iş yükü sürekli olarak değişkenlik göstermektedir. Talepte meydana gelen değişimlere uyum sağlamak için atölyedeki işçi sayısı esnekliğinin sağlanması, U tipi yerleşim ile mümkün olmuştur. Bu yerleşimde, her bir işçinin sorumlu olduğu iş sırası kolaylıkla genişletilip daraltılabilmektedir.

Tam Zamanında Üretim (TZÜ) sistemleri, ürün kalitesini arttırmak ve maliyetleri düşürmek için üretim sürecindeki varolan israfları ortadan kaldırmayı amaçlar. U Tipi montaj hatları, günümüzde Tam Zamanında Üretim sistemlerinin vazgeçilmez bir parçasıdır. U Tipi hatta, makinalar hattın çevresine konuşlandırılır. İşçiler ise hattın iç tarafında çalışırlar. U tipi yerleşimde hattın giriş ve çıkışı aynı pozisyonudadır. U Tipi hatlarda çalışan işçilerin yüksek nitelikli olmaları, yani birden çok süreç içerisinde çalışabilecek durumda olmaları, U tipi montaj hatlarının esnekliğinin dayandığı noktadır. Eğer hattın giriş ve çıkış yerleri birbirine yakın

olursa bu yerleşim şekli U tipi yerleşim olarak, eğer bu yerleşim içerisinde aynı işçi birden fazla makinada çalışıyorsa, bu durumda da Montaj Hattı *U tipi* üretim hattı olarak adlandırılır. Genelde hattın girişi ve çıkışından bir işçi sorumludur. Bu sayede hattın bir ürün çıkmadan, yeni bir ürünün hatta girmesine izin verilmez. Bu sayede hatta bir sorun olduğunda, hattın hemen durdurulması olanaklı olur. U tipi hatlarda, ürünler ve işçiler saat yönünde veya tersi yönde hareket edebilir. Bu aynı zamanda işçilerin ayakta ve yürüyerek çalışmalarını gerektirir.

Şekil 2.10’de yedi görevin yapıldığı bir U tipi montaj hattı gösterilmektedir. U tipi montaj hatlarının, parçanın hatta girdiği tarafa *hattın önü*, parçanın hattan çıktığı tarafa ise *hattın arkası* adı verilir. Bu hatlarda, hattın önü ve arkası aynı uzunluktadır. İşçiler, U tipi montaj hattının içinde çalışırlar. U tipi montaj hattı uygulamalarında karşılaşılan en önemli problem; işçilerin yapacakları görevlerin belirli bir çevrim zamanına göre belirlenmesidir. Yani; istasyonlara atanacak görevlerin belirlenmesidir. Bu problem kaynaklarda U tipi MHD problemi olarak adlandırılır. Şekil 2.10’de görülen U tipi montaj hattı üç istasyondan meydana gelmektedir. Bugüne kadar yapılan çalışmalar U tipi montaj hatlarında ihtiyaç duyulan işçi sayısının (istasyon sayısı) aynı üretim miktarı için düzgün montaj hatlarında ihtiyaç duyulan işçi sayısına eşit veya bundan az olduğunu göstermektedir.



Şekil 2.10. U tipi hat yerleşimi

Bir U tipi montaj hattında, istasyonlara atanan görevlerin hat üzerindeki yerlerine bağlı olarak iki farklı istasyon ortaya çıkmaktadır. Bunlar;

- Görevleri hat üzerinde sıralı bir biçimde yerleştirilmiş olan *klasik* istasyonlar. [Şekil 2.10](#)'daki (2) ve (3) numaralı istasyonlar klasik istasyonlardır. Bir klasik istasyonda çalışan işçi, iki farklı türde mesafeyi yürür. Bunlardan ilki: bir görevin tamamlanabilmesi için gerekli olan *görev mesafesi*, diğeri ise, çevrim zamanı tamamlandıktan sonra istasyonun başlangıcına dönmek için gerekli olan *geri dönüş mesafesi*'dir.
- Hattın hem önünden hem de arkasından görevler içeren *karşıya geçişli* istasyonlar. [Şekil 2.10](#)'daki (1) numaralı istasyon karşıya geçişli istasyondur. Karşıya geçişli istasyonlarda, klasik istasyonlardan farklı olarak bir de *karşıya geçiş mesafesi* vardır. Karşıya geçiş mesafesi, işçinin sorumlu olduğu bir görevi yerine getirebilmesi için hattın önünden hattın arkasına geçerken yürüdüğü mesafedir.

U tipi montaj hatlarında, bir istasyonun yükü, istasyona atanan görevlerin zamanları toplamının yanı sıra, işçinin istasyondaki yürüme zamanına bağlıdır. U tipi montaj hattı dengeleme problemleri çözülürken yürüme zamanları istasyon yüküne dahil edilebileceği gibi, göz ardı edilmeleri de çok fazla sorun yaratmayacaktır.

Düz montaj hattı sistemlerini U tipi montaj hattı sistemlerine dönüştürme kararı, büyük bir yerleşim tasarımı değişikliği ve montaj işlemleri için yatırım içerir. Yalın üretim destekçileri ve TZÜ felsefesi, U tipi montaj hattı sistemlerinin geleneksel düz hat yerleşiminden daha fazla işgücü verimliliği gelişimini içeren fayda sunduğunu ileri sürer. Bu faydalar firmaların montaj hatlarını neden geleneksel düz hattan, U tipine dönüştürmeyi düşündüklerinin esas sebebini ortaya koyar [[Aase ve ark, 2003](#)].

Dünya kalitesinde üretim ya da yalın üretim çalışmaları dış rekabet ortamındaki değişimlerle başa çıkmak için esnek üretim sistemlerini gerektirir. Tam zamanında üretimin faydalarından biri firmayı dış çevredeki değişimlere uyarlama yeteneğidir [[Miltenburg, 1998](#)]. Araştırmalar tam zamanında üretimin dış çevredeki değişimlerle

başa çıkabilmesinin, çok fonksiyonlu işçiler ve etkin tesis yerleşimlerini içeren pek çok sebebi olduğunu ileri sürüyor [Scholl ve Klein, 1999].

Çok fonksiyonlu işçilerden yararlanma çabasında olan firmalar genellikle U tipi montaj hattı yerleşiminin kullanımını benimserler. Geleneksel düz montaj hatlarını, U tipi yerleşim biçiminde düzenleme yoluyla, işçiler düz bir hat yerleşiminde mümkün olmayan görev birleştirmelerini yerine getirebilmek için U hattının iki bacağı arasında hareket edebilirler. Bu esneklik TZÜ firmalarına, tesislerinde toplam işçi sayılarını muhtemelen azaltma ve bu şekilde daha etkin bir tesis yerleşimi yaratma imkanı verir. U tipi hatların sağladığı yararlar aşağıdaki gibi listelenmiştir [Cheng ve ark., 2000].

- TZÜ ortamında bir hattın üretim hızı sık sık değişir. Böyle bir ortamda, miktar esnekliğinden dolayı U tipi hat düz hatta tercih edilir. Hattaki operatör sayısını artırma ya da azaltma yoluyla, şirket gereksinim duyduğunda üretim hızını ayarlayabilir. Miktar esnekliğini bu ölçüde düz bir hatla elde etmek zordur.
- U tipi hatta yürüme mesafesi düz hattan daha kısa olduğu için, bir operatörün pek çok iş merkezini izlemesi daha kolaydır.
- U tipi hatta gereksinim duyulan iş istasyonu sayısı düz hatta gereksinim duyulandan asla daha fazla değildir, bazen daha azdır. Bu U tipi hattaki iş istasyonlarına görev gruplamada daha fazla imkan olduğundandır.
- U tipi hatlar konveyörler ve özel malzeme taşıma operatörleri gibi özel malzeme taşıma ekipmanlarına ihtiyacı ortadan kaldırır. Onun yerine üretim operatörleri ürünleri makineden makineye taşır.
- Küçük boyutlu bir U hattı görülebilirlik ve iletişimi geliştirir. Bu ekip çalışmasını geliştirir, daha büyük bir ait olma duygusu verir ve operatörlerin uzun bir hat boyunca birbirlerinden ayrı hareket ettikleri ve belki stok duvarlarıyla ayrıldıkları düz bir hattan daha çok sorumluluğu ve sahipliği artırır.

- Toplam Kalite Yönetiminin bir prensibi, hatalı bir ürün üretildiği istasyona geri döndükten sonra ilk fırsatta kalite problemlerini düzeltmeyi gerektiren kaynak kalitesidir. Bir U hattında hatalı ürünün geri dönmesi için mesafe, bu prensibi izlemeyi daha kolaylaştıracak kadar kısadır. Bu hatalı ürünü ayrı bir yeniden işleme bölgesine gönderen geleneksel politikayla çelişir.

Geleneksel düz hatların yerine U hatların tercih edilmesinin çeşitli yararları vardır. Bunlar;

- Yerleşim nedeniyle çalışanların birbirleriyle daha rahat görsel temas ve iletişim kurmaları sağlanır. Bu durum bir kalite problemi meydana geldiğinde daha çabuk belirlenmesini ve karar verilmesini sağlar [[Miltenburg ve Wijngaard, 1994](#)].
- Çalışanlar U hatlarında birçok işi yerine getirebilecek veya birçok farklı makineyi kullanabilecek yüksek beceriye sahip hale gelirler. Bu durum çevrim süresinin veya çıktı hızının değiştirilmesinin gerektiği zamanlarda işçilerin herhangi bir makineye hemen adapte olmalarını sağlar. Ayrıca çalışanlar birden fazla işi yerine getirebilme düzeyine ulaştıklarında işler arasındaki ilişkileri anlamaktadırlar. Böylece çalışanların gelişime daha iyi katkıda bulunmaları sağlanmaktadır [[Miltenburg ve Wijngaard, 1994](#)].
- Tam zamanında üretim uygulamasının sonucunda talepteki değişikliklere uyum sağlanabilmesi kolaylaşmıştır. U hattın çıktı hızı hatta yeni işçi ilave edilmesi veya işçi eksiltilmesi ile düzenlenebilmektedir. Talep düşük olduğunda hatta az işçi çalıştırılmakta ve bu durumda her bir işçi oldukça yüksek sayıda makineden sorumlu olduğundan hattın üretim hızı düşmektedir. Sonuçta üretim adetleri de talebe uygun hale gelmektedir. Hattaki ürüne talep arttığı zaman ise, hatta işçi takviyesi yapılmaktadır. Bu durumda her bir işçi daha az sayıda makineden sorumlu olduğundan üretim hızı ve üretim miktarı talebe uyumlu hale gelecektir. Geleneksel düz hatlarda ise işçilerin farklı işlere ve farklı makinelere uyum sağlamaları güç olduğundan bu hatlarda tekrar dengeleme çalışması yapılması çok zordur. Bu tip hatlarda çevrim süresi genellikle değişmediğinden talepteki

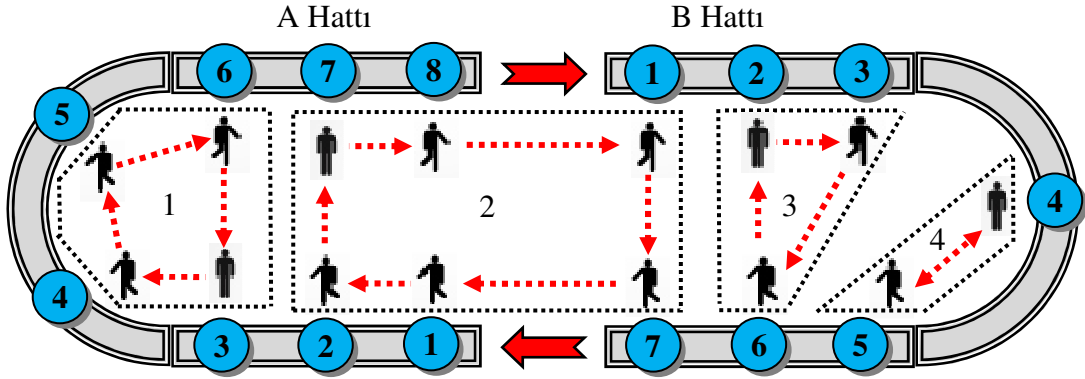
değişim hattın çalışma süresinin düzenlenmesi ile karşılanmaya çalışılır. Artan talebin, düşük nitelikli işgücü nedeniyle çevrim süresini değiştiremeyip, üretim saatini arttırarak karşılandığı doğrusal hatlara kıyasla, U Tipi hatlardaki bu esneklik sayesinde talep değişiklikleri, hat üzerindeki çalışan sayısının değiştirilmesi ile kolayca çözülür.

- U hatlarda ihtiyaç duyulan istasyon sayısı geleneksel hatlarda ihtiyaç duyulandan daha azdır. Çünkü U hatlarda işlerin istasyonlarda gruplandırılması için birçok olasılık vardır [Miltenburg ve Wijngaard, 1994].

Stokların azaltılması, malzeme taşımalarının basitleştirilmesi, üretim planlaması ve kontrolünün kolaylaştırılması, takım çalışması ve problem çözmeye elverişli olması, daha iyi kalite kontrol sağlanması gibi birçok sebepten dolayı U hatlar tercih edilmektedir.

Çoklu U tipi montaj hatları

Bir U tipi montaj hattı, bir TZÜ sisteminin en temel taşıdır. Bir TZÜ sistemi ise, çok sayıda U tipi montaj hattından meydana gelebilmektedir. Toyota üretim sisteminde, çok fonksiyonlu işgücünün kullanılması sayesinde bir işçinin birden fazla sayıda U tipi montaj hattında çalışabilmesi mümkün olabilecektir. Bunun sonucunda, talebin düşük olduğu dönemlerde ihtiyaç duyulan işçi sayısı önemli ölçüde düşecektir. Bu durumun söz konusu olabilmesi için, birbirine yakın bir şekilde yerleştirilmiş birden fazla sayıda U tipi montaj hattının eş zamanlı olarak dengelenmesi gerekir. Eş zamanlı olarak dengelenmiş çok sayıda U tipi montaj hattı, klasik ve karşıya geçişli istasyonların yanında bir de çok hatlı istasyonlara sahip olacaktır. Çok hatlı istasyonlar, birden fazla sayıda U tipi montaj hattındaki görevleri gerektiren istasyonlardır. Aşağıda [şekil 2.11](#)'de eş zamanlı dengelenmiş iki U tipi montaj hattı görülmektedir.



Şekil 2.11. Eş zamanlı dengelenmiş U tipi montaj hatları

Yukarıdaki [şekil 2.11](#)'da görülen (2) numaralı istasyon çok hatlı bir istasyondur. Görüldüğü gibi, bu istasyon hem A hattında hem de B hattında yer alan görevlere sahiptir. Çok hatlı istasyonlara atanan görevler hatların ucunda yer almak zorundadır. Örneğin; B hattında yer alan (4) numaralı görevin çok hatlı istasyona atanması söz konusu değildir. Çünkü; bu durumda (2) numaralı çok hatlı istasyondaki işçi ile (3) numaralı istasyondaki işçi arasında bir kesişme olacaktır. Bu durum, işçilerin görevlerini zamanında tamamlamalarına engel olabilecektir [[Kara, 2004](#)].

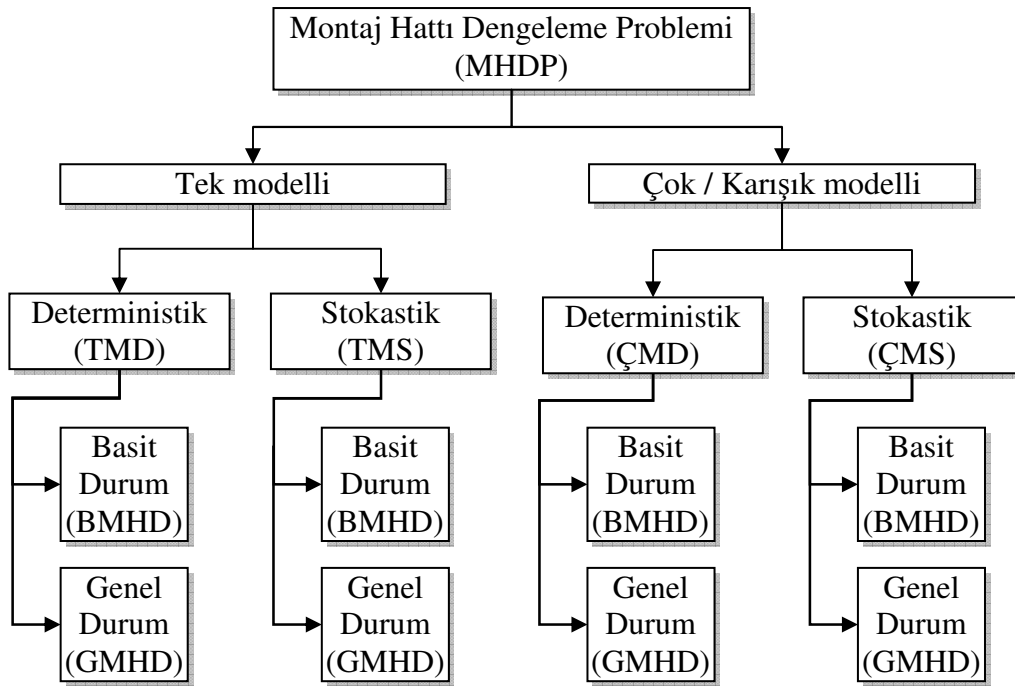
2.8. Montaj Hattı Dengeleme Problemlerinin Sınıflandırılması

MHD problemlerinin sınıflandırılmasına yönelik kaynaklarda farklı çalışmalara rastlamak mümkündür. MHD problemlerinin sınıflandırılması hakkında ayrıntılı bilgi için şu çalışmalar incelenebilir: [[Buxey ve ark., 1973](#), [Baybars, 1986](#), [Shtub ve Dar-El, 1989](#), [Ghosh ve Gagnon, 1989](#), [Erel ve Sarin, 1998](#), [Scholl, 1999](#), [Rekiek ve ark., 2002](#), [Scholl ve Becker, 2006](#), [Becker ve Scholl, 2006](#), [Boysen ve ark., 2007](#)].

Baybars, deterministik MHD problemlerini, Basit Montaj Hattı Dengeleme (BMHD) problemi ve Genel/Geleneksel Montaj Hattı Dengeleme (GMHD) problemi olmak üzere iki gruba ayırmıştır [[Baybars, 1986](#)]. MHD probleminin en basit hali olan BMHD problemi, sadece görevler arasındaki öncelik ilişkisi kısıtlarını kapsayan, görev zamanlarının sabit ve belirli olduğu, üretim akışına yönelik istasyonların düz olarak sıralandığı ve tek bir ürünün (modelin) montajının yapıldığı problemidir.

Kaynaklarda, BMHD problemlerinin bazı varsayımlarını genelleştiren ya da kaldıran tüm problem tipleri GMHD problemleri olarak adlandırılır. Bu sınıfta yer alan problemler teçhizat seçimi, proses alternatifleri, bölgeleme kısıtları vb. pratik uygulamaları kapsayan çok geniş bir uygulama alanına sahiptir. Bir başka ifadeyle, GMHD problemi, BMHD probleminin bir uzantısıdır.

Kaynaklardaki bir başka sınıflandırmaya göre, MHD problemleri, 4 gruba ayrılır: *Tek Modelli Deterministik* (TMD), *Tek Modelli Stokastik* (TMS), *Çok/Karışık Modelli Deterministik* (ÇMD), *Çok/Karışık Modelli Stokastik* (ÇMS) dengeleme problemleri [Ghosh ve Gagnon, 1989]. MHD problemlerinin sınıflandırılması [şekil 2.12](#)'de verilmiştir.



Şekil 2.12. Montaj Hattı Dengelem Poblemlerinin sınıflandırılması

TMD problemi, MHD problemlerinde hatta yalnız bir modelin üretildiği ve görev zamanlarının bir birimden diğerine herhangi bir değişiklik göstermediği yani sabit ve belirli olduğu varsayılır. Bu varsayım, montaj hattında makinelerin ve robotların kullanıldığı hatlarda geçerli olmakla birlikte insanların çalıştığı hatlarda çok geçerli olmamaktadır. Buna göre görevler, öncelik ilişkileri sağlanacak ve belirli bir

performans ölçüsü eniyilenecek şekilde iş istasyonlarına atanırlar. TMD, yukarıda da belirtildiği gibi MHD probleminin orjinal ve en basit şekli olan BMHD problemidir.

BMHD problemi (BMHDP) için temel varsayımlar şunlardır [Gökçen, 1994]:

- Montaj hattında tek bir ürünün montajı büyük miktarlarda gerçekleştirilir.
- Görev zamanları belirli ve sabittir.
- Problemin öncelik diyagramı bilinmektedir.
- Bir görev iki ya da daha fazla iş istasyonu arasında bölüştürülemez.
- Bir görev kendisinden önce gelen görevler tamamlanmadan başlayamaz.
- Bir istasyona atanan görevlerin tamamlanma zamanlarının toplamı, çevrim zamanına eşit yada ondan küçük olmalıdır.

BMHD problemlerinin çözümü için çeşitli performans ölçüleri kullanılmıştır. BMHDP'nin versiyonları çizelge 2.2'de verilmiştir.

Çizelge 2.2. BMHDP'nin sınıflandırılması

BMHD problemi versiyonları		Çevrim Zamanı (C)	
		Verilmiş	Enküçükle
İstasyon sayısı	Verilmiş	BMHDP-F	BMHDP-1
	Enküçükle	BMHDP-1	BMHDP-E

BMHDP – F: Çevrim zamanı ve istasyon sayısı verilir, uygunluk kontrolü yapılır.

BMHDP – 1: Çevrim zamanı verilir, istasyon sayısı enküçüklenir.

BMHDP – 2: İstasyon sayısı verilir, çevrim zamanı enküçüklenir.

BMHDP – E: Çevrim zamanı ve istasyon sayısı birer değişken iken, hat etkinliği enbüyüklenir.

Literatürde yer alan pek çok çalışmada BMHDP ele alınmıştır. BMHDP'de tüm parametreler kesin olarak bilinmektedir ve yukarıdaki varsayımlar sağlanmaktadır. BMHD problemine farklı kısıtların ya da faktörlerin (paralel istasyonlar, paralel

görev, bölgelendirme kısıtları vb.) ilave edilmesi ile problemin yapısı GMHD problemine dönüşür.

Tek Modelli Stokastik (TMS) problemi tek tip ürün ve modelin üretilmesi için gerekli olan görevlerin tamamlanma zamanlarının özellikle işgücü performansının değişkenliğinden kaynaklanan farklılaşmaların meydana geldiği durumları içerir. Ayrıca, işçilerin işlem zamanlarının nadiren de olsa sabit olduğunu ve manuel montaj hattı için daha gerçekçi olmak üzere görev zamanlarının değişkenliğini belirtir. Görev zamanındaki değişiklik, *insan* ve/veya *süreçten* kaynaklanmaktadır. Söz konusu değişkenliğin sebepleri arasında; yorulma, dikkatin dağılması, yetersiz nitelikteki işgücü, iş tatminsizliği, hatalı girdiler, araç-gereç bozulmaları vb. faktörler sayılabilir. Bu durum, istasyonlara atanan görevlerin toplam süresinin çevrim zamanını aşmasına ve dolayısıyla bazı görevlerin tamamlanamamasına sebep olabilmektedir. Özellikle görevler arasındaki öncelik ilişkileri dikkate alındığında, bazı görevlere başlanamadığı da görülebilmektedir. Hatta tamamlanamayan görevler, ya en kısa sürede tamamlanmak üzere hattın dışına alınırlar ve orada tamamlanırlar, ya da bitmemiş ürün olarak ayrılırlar. Bu şekilde yarım kalan veya hiç başlanmayan görevlerde, görev zamanlarının stokastik olduğunun kabul edilmesi, montaj hatlarını iyileştirme politikalarından birisidir. Stokastik MHDP’de ise, görev zamanlarının kesin olarak bilinmediği varsayılmakta fakat görev zamanları belirli bir olasılık dağılımı ile modellenebilmektedir. Buna göre problem, öncelik ilişkileri dikkate alınarak ve belirli bir performans ölçüsü eniyilenecek şekilde görevlerin uygun iş istasyonlarına atanmasıdır. Manuel montaj hattı için daha gerçekçi bir yaklaşımdır. Bu tip problemlerde, amaç toplam işgücü ve toplam beklenen tamamlanmama maliyetinden oluşan toplam sistem maliyetinin enküçüklenmesidir [[Gökçen, 1994](#)].

Tüketici sayılarının ve bilinç düzeylerinin artması ile birlikte diğer imalatçı kuruluşların rekabeti, tek tip ürünün seri üretimini engellemektedir. Bu durumda çok modelli ve karışık modelli hat tasarımları gündeme gelmektedir. Çok/Karışık Modelli Deterministik (ÇMD), görev zamanlarının deterministik (sabit ve kesin belirli) olduğunu kabul eden ve bir montaj hattında farklı ürünlerin veya aynı ürünün farklı modellerinin üretildiği MHD problemidir. Çok modelli üretim, iki veya daha

fazla ürünün büyük partiler halinde ayrı bir şekilde üretilmesidir. Çok modellenli üretim akışı, bir partinin üretiminin tamamlanmasından sonra diğer partinin üretimine geçilmesiyle sağlanır. Karışık modellenli montaj hatlarında ise, ürünlerin farklı modellerinin çok düşük hacimlerde (genellikle bir adet olarak ifade edilen) herhangi bir sırada karışık olarak üretilmesi söz konusudur. Her iki durumda da benzer üretim süreçleri söz konusudur. Tek modellenli montaj hatlarından farklı olarak model seçimi, model sıralaması, model parti büyüklüklerinin belirlenmesi vb. problemlerin ele alındığı çok/karışık modellenli hatlar, tek modellenli hatlara göre daha karmaşık ve zor problemlere yol açarlar. Bu tip üretimin en önemli faydası, müşteri isteğini karşılamak üzere değişik modellerin sürekli olarak üretilmesi ve büyük mamul stoklarına gidilmemesidir. Modellerin değişik işlem zamanlarından doğan dezavantajı ise, iş akışının düzenli olmaması, dolayısıyla da daha fazla istasyon boş zamanı ve yarı bitmiş mamüllerden oluşan yığınlardır.

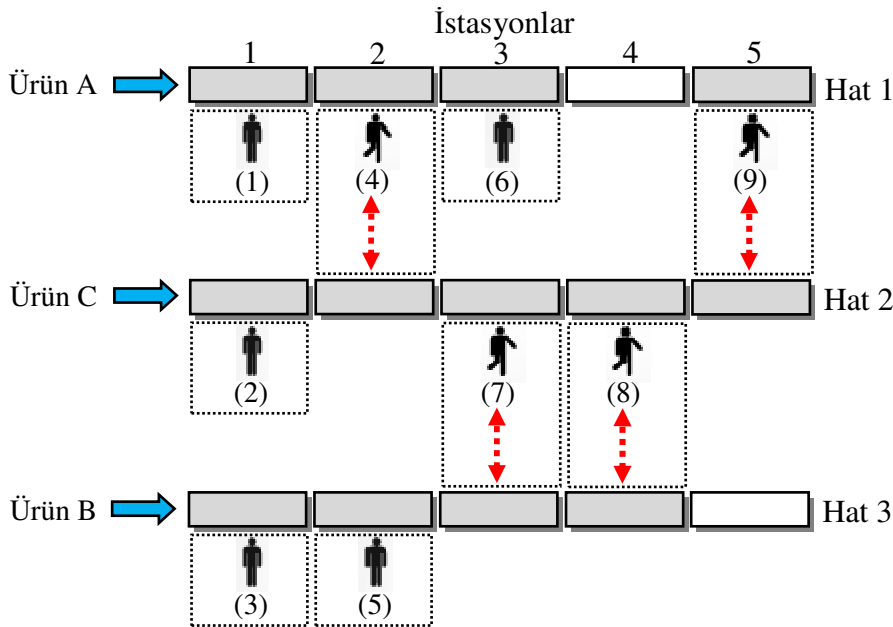
Çok/Karışık Modellenli Stokastik (ÇMS) problem perspektifinin yukarıdaki problem yapısından tek farkı, görev zamanlarının stokastik olmasıdır. Ayrıca, bu problem yapısında model sıralama problemi daha karmaşıktır. Bu karmaşıklığın sebebi, öğrenme etkileri, operatör yetenek seviyesi, iş tasarımı ve operatör görev zamanı değişkenliği vb. faktörlerin hattın sıklıkla montajı yapılan her bir ürün için yeniden dengelenmesini zorunlu kılması, problemin analiz edilmesini daha da güçleştirmektedir.

Çok modellenli hatlarda yapılan dengeleme işlemleri karışık modellenli hatlarda yapılan işlemlerden daha kolaydır. Çünkü bu hatlarda üretimlerde benzer imalat gereksinimleri olduğundan hattın bir model için dengelenmesi (tasarlanması) ve elde edilen sonuçlardan yararlanılarak hat üzerinde gerekli düzenlemelerin yapılmasıyla diğer modeller için de uygun bir hale getirilmesi mümkündür. Çok modellenli ve karışık modellenli hatların dengelenmesi probleminin çözümünde yaşanan zorluklar nedeniyle bu tip problemler için en iyi çözümü veren kesin çözüm yöntemleri yerine sezgisel yöntemlerden yararlanılır.

3. MALİYET TABANLI PARALEL MONTAJ HATTI Dengeleme

3.1. Paralel Montaj Hattı Dengeleme Problemleri

Sanayide, birbirine *paralel* olarak konumlandırılmış *birden fazla özdeş* ya da *benzer düz* montaj hatlarına (Bkz. [Şekil 3.1](#)) sahip üretim sistemlerine oldukça yaygın olarak rastlanmaktadır. Bu hatlarda, aynı zamanda birbirinden bağımsız olarak, *aynı* ya da *benzer* ürünlerin montaj işlemi yapılmaktadır. Diğer bir ifadeyle, her düz hatta sadece belirli bir ürünün montajı yapılmaktadır. [Gökçen ve ark. \(2006\)](#), paralel montaj hatlarından oluşan üretim sisteminin verimliliğinin ve kaynak kullanımının artırılması amacıyla, bu paralel hatların bazı komşu istasyonlarının iş yüklerinin birleştirilmesini önermişlerdir. Birden fazla düz hattın eş zamanlı olarak dengelendiği bu problem, *Paralel Montaj Hattı Dengeleme* (PMHD) problemi olarak isimlendirilmiştir. PMHD probleminin amacı, hatlarda kullanılan toplam istasyon sayısını (işyeri sayısını veya operatör sayısını) enküçükmektir.



Şekil 3.1. Paralel konumlu 3 düz hatlı bir üretim sistemi

İlk defa [Gökçen ve ark. \(2006\)](#) tarafından sunulan paralel hatların ortak bir şekilde dengeleme felsefesini daha iyi anlayabilmek için şematik olarak ifade etmeye

çalışalım. [Şekil 3.1](#)'de görüldüğü gibi paralel olarak tasarlanmış yan yana üç düz hattın olduğunu varsayalım. Her bir düz montaj hattı ardışık olarak sıralanmış beş istasyondan oluşmaktadır. Her hat için ortak olan çevrim süresini aşmayacak şekilde, her hatta belirli bir ürünün montaj işlemi yapılmaktadır. Birinci hatta ürün A, ikinci hatta ürün C ve üçüncü hatta ürün B üretilmektedir.

Geleneksel olarak, her ürün için öncelik diyagramı ve görevlerin tamamlanma zamanları verildiğinde, öncelik ilişkileri ve çevrim zamanı kısıtlarını sağlayarak, toplam istasyon sayısını enküçükmek amacıyla, her hat birbirinden bağımsız olarak dengelenir. Bu probleme, Basit Montaj Hattı Dengeleme problemi denilmektedir. Bununla birlikte, paralel montaj hatların eş zamanlı dengelenmesinde, yan yana olan hatların komşu (dikey olarak aynı konumda olan) istasyonlarının iş yüklerinin birleştirilmesi söz konusudur. Başka bir deyişle, yan yana olan hatların komşu istasyonlarına atanan görevler bir çevrim zamanı içerisinde bir operatör tarafından yapılmaktadır. Örneğin, (4) numaralı operatör, hat 1'in 2 numaralı istasyonunda ürün A ile ilgili görevleri tamamladıktan sonra, ürün C ile ilgili görevleri yapmak üzere hat 2'ye geçer. Aynı çevrim zamanı içerisinde, hat 2'nin 2 numaralı istasyonunda ürün C ile ilgili görevleri de tamamladıktan sonra tekrar hat 1'e geri döner. PMHD probleminde hatlar arasında operatörlerin yürüme zamanı ihmal edilir.

Paralel montaj hattı dengelemenin faydasını anlayabilmek için, kullanılan terimleri birbirine karıştırmamak gerekir. Malzeme taşıma sistemiyle birbirine bağlı hat parçasına *istasyon* denirken, bir veya daha çok görevin gerçekleştirildiği montaj hattı üzerinde istasyonlara kurulan alana *işyeri* denir. Sadece bir hattaki ürünle ilgili görevlerin yapılabilmesi için bir istasyona kurulan alana *ayrık işyeri*, yan yana olan iki hatta üretilecek ürünlerle ilgili görevlerin yapılabilmesi için komşu istasyonların arasında kurulan alana ise *ortak işyeri* denir. Ortak işyerinde çalışan operatör, iki ayrı ürün üzerinde aynı çevrim zamanı içerisinde sırayla çalışabilmektedir. Ayrık işyerinde çalışan operatör ise, bir çevrim zamanı içerisinde sadece bir ürünle ilgili görevleri yerine getirir ve çalışma konumu sabittir. [Şekil 3.1](#)'deki (4), (7), (8) ve (9) numaralı operatörler kurulan ortak işyerlerinde çalışırken; (1), (2), (3), (5) ve (6) numaralı operatörler ayrık işyerlerinde çalışırlar ve başka bir hatta geçiş yapmazlar.

Ayrıca, birinci hattın 4 numaralı istasyonuna herhangi bir ayırık veya ortak işyeri kurulmamıştır. Bunun gibi hat parçalarına *boş istasyon* denilmektedir ve bu istasyonlarda herhangi bir ürünle ilgili işlem yapılmamaktadır. Boş istasyonlar, yarı mamulü bitişikte bulunan istasyona taşıyan malzeme taşıma sistemini simgelemektedir.

Hatların bazı istasyonlarında yapılan görevlerin birleştirilebilmesi için hatların yan yana ve birbirlerine yakın olması gerekir. Örneğin [şekil 3.1](#)'deki paralel hatların yerleşimi düşünüldüğünde, hat 1 - hat 2 veya hat 2 – hat 3 arasında operatörler ortak kullanılabilirken, hat 1 – hat 3 arasında kullanılamaz. Çünkü hat 1'deki ilgili görevlerini tamamlayan operatör hat 3'e gitmek için gereksiz zaman harcıyacaktır. Bunun sonucunda, çalışma zamanının büyük bir kısmını hatlar arasında yürüyerek geçireceği için, bu durum verimsizliğe neden olur. Bu yüzden yan yana olan hatlarda dahi, operatörlerin her iki hatta ortak olarak çalışabilmesi için hatların birbirine yürüme mesafesinde yakın olması gerekir. Ayrıca, operatörlerin yan yana olan hatlarda bile sadece dikey konumda komşu olan istasyonlarında çalışmalarına izin verilmektedir. Diğer bir deyişle, birbirine yakın olan komşu hatların aynı istasyonlarının iş yükleri birleştirilebilir. Bu durumda, hat 1'in (2) numaralı istasyonunda çalışan bir operatör, hat 2'nin sadece (2) numaralı istasyonunu atanan görevleri yapabilir, hat 2'nin diğer istasyonlarına atanan görevleri yapmasına izin verilmez.

Paralel montaj hattı dengelemede, herhangi bir işyerine (ayırık işyeri veya ortak işyeri) atanan görevler sadece bir operatör tarafından yapılabilir varsayımı vardır. Bu nedenle, $h-1$ ve h hattının k . istasyonlarına kurulan ortak işyerine atanan görevler bir operatör tarafından yapılıyor iken, aynı zamanda h ve $h+1$ hattının k . istasyonlarına ortak işyeri kurarak, bu işyerine atanan görevler başka bir operatör tarafından yapılamaz. Bu şartlarda, h hattın k . istasyonunda 2 operatör çalışıyor olur ki, bu durum her işyerinde sadece bir operatör çalışabilir varsayımıyla çelişir. Örneğin, [şekil 3.1](#)'deki (4) numaralı operatör, hat 1 ve hat 2'nin 2 numaralı istasyonlarına kurulmuş olan ortak işyerinde çalışmaktadır. Bununla birlikte, hat 2 ve hat 3'ün 2

numaralı istasyonlarına başka bir ortak işyeri açılırsa, bu durumda hat 2'nin 2 numaralı istasyonunda iki operatörün çalışması demektir.

Yukarıda anlatılan sadece yan yana olan hatlarda ve sadece dikey konumunda komşu olan istasyonlarda çalışabilme kuralını özetliyecek olursak; h hattının k . istasyonunda çalışan bir operatörün, h hattının yanında olan $h-1$ veya $h+1$ hatlarından sadece birinin sadece k . istasyonunda çalışabilmesine izin verilir. Paralel hatların sadece dikey konumundaki komşu olan istasyonlarının birleştirilmesinin nedeni, operatörlerin yürüme mesafelerinin mümkün olduğunca kısa tutmak ve operatörlerin çalışma alanlarının birbiriyle çakışmasını engellemektir. Eğer bir operatör, h hattının k . istasyonu ile $h+1$ hattının $k+1$. istasyonunda ortak çalışıyor iken, diğer bir operatör, h hattının $k+1$. istasyonu ile $h+1$ hattının k . istasyonunda ortak çalışıyor ise, operatörlerin çalışma alanları birbirine girdiği için üretimde karmaşık bir yapı oluşacak ve operatörler hatlar arasında daha fazla yürüyecekleri için üretimde aksamalara neden olacaktır.

PMHD probleminde ele alınan paralel hatlardan oluşan üretim sisteminin bir parçası olan her düz hat, diğer hatlardaki ürünlerin montaj işlemini yapabilecek esnekliğe sahip değildir. Diğer bir ifadeyle, paralel olarak konumlandırılmış düz hatlardan her biri, sadece belirli bir ürünün montaj işlemi için gerekli olan özel alet ve teçhizatlarla donatılmıştır. Paralel montaj hattı dengelemenin, geleneksel montaj hattı dengelemeden farkı, hatlarda montajı yapılan ürünler aynı kalmakla birlikte, yan yana olan hatların komşu istasyonları arasında ortak işyerleri kurarak, hatlardaki toplam işyeri sayısını dolayısıyla da toplam operatör sayısını enküçükmeye çalışmasıdır. PMHD probleminin çözümünde, hangi ürünün hangi hatta üretilmesi gerektiğine cevap aranmaz. Bu durumun neticesinde, paralel hatların eş zamanlı dengelenmesi sırasında, herhangi bir hatta montajı yapılan ürüne ait bir görev başka bir hattın bir işyerine atanamaz.

Paralel hatların üretim esnekliğinin olmamasının yanında, ortak işyerlerinde çalışan operatörler farklı ürünlerin montajıyla ilgili görevleri yapmak zorunda olduklarından dolayı, bu operatörlerin her görevi yapabilecek bilgi ve beceriye sahip olmaları

gerekmektedir. Özetle, paralel montaj hattı dengelemede hatların üretim esnekliği değil, operatörlerin esnekliği esastır.

Yan yana olan hatların dikey komşuluğunda bulunan istasyonlarında kullanılan genel amaçlı alet ve teçhizatlar varsa, bu durumda alet ve teçhizatlar her iki hatta da ortak olarak kullanılabilir. Bu şekilde yerleştirilebilirse, kurulan ortak işyerleri sayesinde gerekli olan alet-teçhizat sayısından da tasarruf sağlanabilir. Bu durum sadece genel amaçlı alet ve teçhizatlar için geçerlidir. Her farklı ürüne özel araç ve teçhizatlar bu kapsamda değerlendirilemez.

Paralel hatlarda montajı yapılan ürünlerin çevrim zamanlarının birbirine eşit ya da yakın olduğu gibi, birbirinden farklıda olabilir. Bu da her hattın farklı bir çevrim zamanı için çalışması anlamına gelmektedir. Bu durumda, paralel hatları eş zamanlı dengelenmesinden söz etmek mümkün değildir. Çevrim zamanlarının birbirinden farklı olduğu durumlarda, her hat için ortak bir çevrim zamanı belirleyerek, hatların eş zamanlı dengelenmesini yapabilmek için Gökçen ve ark. (2006) normalizasyon yöntemi önermişlerdir. Gökçen ve ark. (2006) önerdiği normalizasyon yönteminin adımları aşağıda Şekil 3.2’de verilmiştir:

Algoritma: Normalizasyon yöntemi

C_h : h hattı için çevrim zamanı $h = 1, \dots, \overline{H}$

Adım 0:

t_{hi} : h hattında üretilen ürüne ait i görevinin tamamlanma zamanı

Adım 1: Tüm hatların çevrim zamanının en küçük ortak katını bul: $EKOK$

Adım 2: Her hat için bir çarpan değeri bul: $\zeta D_h = \frac{EKOK}{C_h}$, $h = 1, \dots, \overline{H}$

Adım 3: Her hatta üretilecek ürüne ait görev zamanlarını, o hatta ait çarpanla

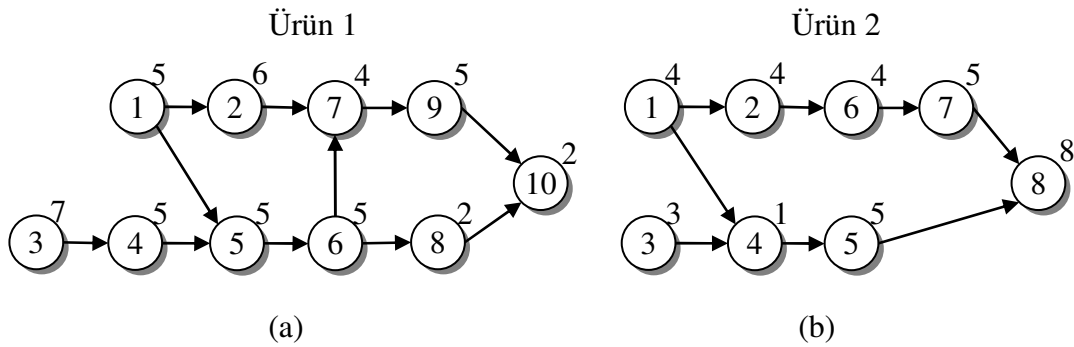
çarp: $t'_{hi} = t_{hi} \cdot \zeta D_h$, $h = 1, \dots, \overline{H}$

Adım 4: Problemin çözümde yeni görev tamamlanma zamanlarını kullan: t'_{hi}

Çevrim zamanını $EKOK$ değeri olarak al.

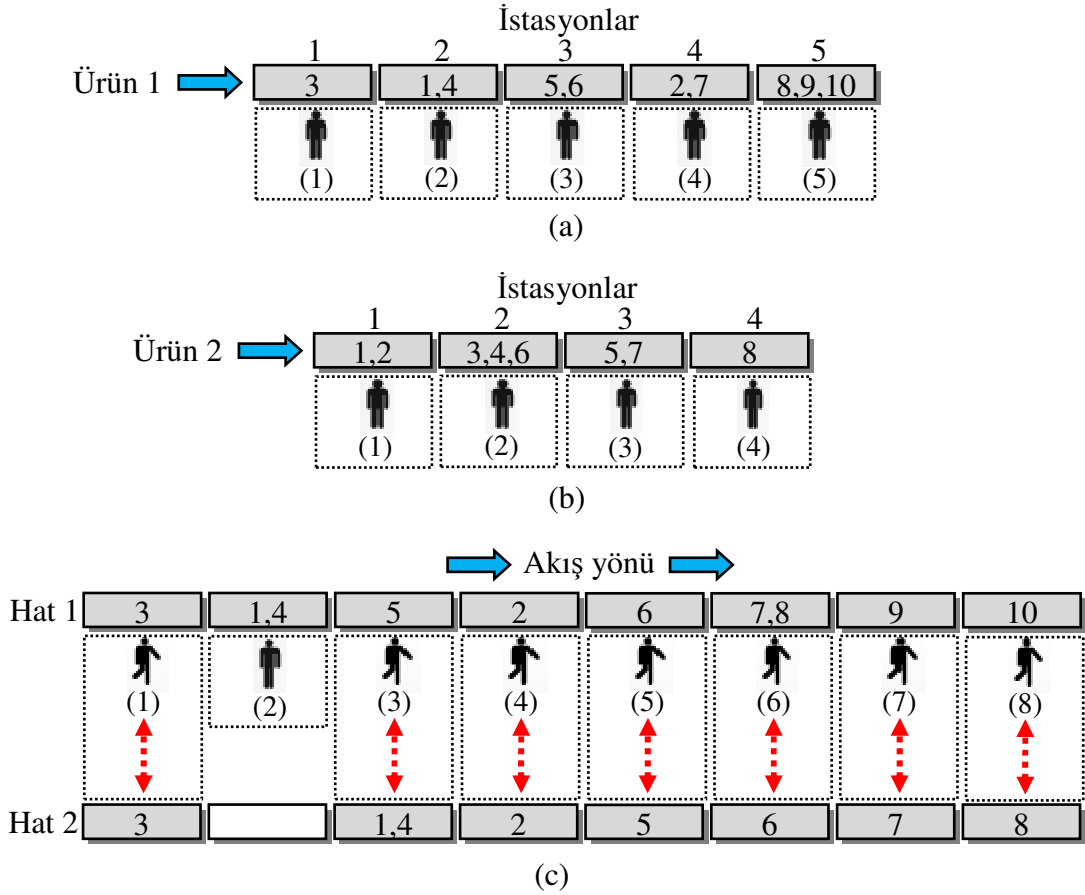
Şekil 3.2. Normalizasyon algoritmasının adımları

Yukarıda anlatılanlar ışığında, PMHD yaklaşımının dengeleme sonuçlarını nasıl değiştirdiğini görmek amacıyla örnek sayısal bir problem çözdürülmüştür. Paralel olarak yerleştirilmiş iki düz hatta birbirinden farklı olan iki ürünün montajı yapılmaktadır. Birinci hatta [şekil 3.3\(a\)](#)'da öncelik diyagramı verilen ürün 1'in, ikinci hatta ise, [şekil 3.3\(b\)](#)'de öncelik diyagramı verilen ürün 2'nin montajı yapılmaktadır. Düğümlerin içerisindeki sayılar görev numaralarını, düğümleri birbirine bağlayan oklar öncelik ilişkilerini, düğümlerin üstündeki sayılar ise görev tamamlanma zamanlarını göstermektedir. Ürünlere ait öncelik diyagramında gösterilen görev numaraları aynı olsa bile, birbirinden farklı görevlerdir. Örneğin ürün 1'in üretilmesi için gerekli olan 1 numaralı görev ile ürün 2 için gerekli olan 1 numaralı görev aynı değildir. Bu örnek problem için çevrim zamanı 10 olarak belirlenmiştir.



Şekil 3.3. (a) 10 görevli (b) 8 görevli öncelik diyagramı

[Şekil 3.3](#)'de verilen ürünlere ait öncelik diyagramlarına göre, iki düz montaj hattı birbirinden bağımsız olarak dengelenmiştir ve en iyi görev atamaları [şekil 3.4](#)'de verilmiştir. Paralel hatların geleneksel bir yaklaşımla birbirinden bağımsız olarak dengelenmesindeki kasıt; bir hattın kaynaklarını, diğer hattan bağımsız kullanmaktır. Başka bir ifadeyle, herhangi bir istasyonda çalışan bir operatörün yapması gereken görev(ler)i bitirdiğinde, boş zamanının olması halinde bile, diğer bir ürüne ait olan görevleri yapmak amacıyla başka bir hatta geçememesi durumunu ifade etmektedir. [Şekil 3.4\(a\)](#) ve [3.4\(b\)](#)'deki bağımsız dengeleme sonuçlarına göre, ürün 1'in üretildiği hat 1'de 5 istasyon (işyeri veya operatör), ürün 2'nin üretildiği hat 2'de ise 4 istasyon kullanılmıştır.



Şekil 3.4. (a) 10 görevli geleneksel MHD problemi için en iyi görev atamaları (b) 8 görevli geleneksel MHD problemi için en iyi görev atamaları (c) PMHD problemi için en iyi görev atamaları

Paralel hatların eş zamanlı dengelenmesi sonucunda elde edilen en iyi görev atamaları [şekil 3.4\(c\)](#) verilmiştir. Hatların ortak bir şekilde dengelenmesiyle birlikte, gerekli olan operatör sayısı 9'dan 8'e düşmüştür. (2) numaralı operatör haricindeki tüm operatörler aynı çevrim zamanı içerisinde, her iki hattaki ürünlerle ilgili görevleri yapmaktadır. Paralel hatların ortak bir şekilde dengelenmesi sırasında komşu istasyonlarda yapılan görevleri birleştirmedeki amaç, bir operatörün bir hatta yapılması gereken görevleri tamamladıktan sonra eğer boş zamanı kalmış ise, bu zamanı değerlendirerek kaynak kullanımını arttırmaktır. Dolayısıyla kaynak kullanımındaki artış neticesinde, hem işgücünden tasarruf sağlanır hem de verimlilik düzeyi artar. Hatlar arasında ortak işyerleri kurmak, hatlarda bazı boşlukların olmasını gerektirmektedir. Örneğin hat 2'nin 2. istasyonu boş bir istasyondur. Bu

istasyonda ürün üzerinde herhangi bir işlem yapılmamaktadır. Hat 2'deki boş istasyon, (1) numaralı operatörün hat 2'deki görevi bittiğinde, yarı mamulü (3) numaralı operatörün çalıştığı ortak işyerine transfer eden malzeme taşıma sistemini göstermektedir.

Şekil 3.4'te verilen hatların bağımsız ve eş zamanlı olarak dengelerine göre, operatörlerin boş zamanları hesaplanmış ve çizelge 3.1'de verilmiştir. Çizelge 3.1'deki sonuçlara göre, hatların bağımsız olarak dengelendiği durumda, her iki hattaki operatörlerin toplam boş zamanı 10 (dakika)'dır. Oysaki, paralel hatların komşu istasyonları arasında kurulan ortak işyerinde çalışan operatörlerin toplam boş zamanı sıfır olduğu görülmektedir. Bu sonuçtan açık bir şekilde anlaşıyor ki, paralel hatların eş zamanlı dengelenmesi, kaynak kullanımını ve sistemin verimliliğini arttırmaktadır. Ayrıca, operatör (veya işyeri) sayısında tasarruflar sağlanmaktadır.

Çizelge 3.1. Geleneksel MHD ve PMHD problemlerinin en iyi görev atamaları

Operatör (k)	Geleneksel MHDP (Bağımsız)				PMHDP (Eş zamanlı)		
	Hat 1	Boş zaman	Hat 2	Boş zaman	Hat 1	Hat 2	Boş zaman
1	{3}	3	{1,2}	2	{3}	{3}	0
2	{1,4}	0	{3,4,6}	2	{1,4}	-	0
3	{5,6}	0	{5,7}	0	{5}	{1,4}	0
4	{2,7}	0	{8}	2	{2}	{2}	0
5	{8,9,10}	1	-	-	{6}	{5}	0
6	-	-	-	-	{7,8}	{6}	0
7	-	-	-	-	{9}	{7}	0
8	-	-	-	-	{10}	{8}	0
Operatör sayısı (Hat 1)				5			
Operatör sayısı (Hat 2)				4			
Toplam operatör sayısı				9	8		
Toplam boş zaman				10	0		

PMHD problemini özetliyecek olursak;

- operatör yürümelerine uygun, birbirine *paralel* ve *yakın konumda* olan *birden fazla özdeş* ya da *benzer düz montaj hattının eş zamanlı dengelemesi* amacıyla,
- montaj sürecindeki kısıtlara (çevrim zamanı, öncelik ilişkileri, atama kısıtları) uyarak,

- her hatta montajı yapılan *belirli* ürünlere ait olan görevlerin, *ayrık* veya *ortak işyerlerine* atanmasını kapsamaktadır.

PMHD probleminin amacı, istasyon sayısını enküçükmek değil, işyeri (operatör) sayısını enküçükmektir.

PMHD problemi için aşağıdaki varsayımlar yapılmıştır:

- Her bir montaj hattında sadece bir çeşit ürün üretilmektedir ve her hatta üretilcek ürünler daha önceden belirlenmiştir.
- Her bir ürünün öncelik diyagramları bilinmektedir.
- Her bir ürün için görev süreleri belirli ve sabittir.
- Her hattın her istasyonunda çalışan operatörler her göreve yapabilecek bilgi ve beceriye sahip olması.
- Hatların her iki tarafında da çalışılabilmektedir.
- Hatlar arasındaki yürüme mesafenin yakın olduğu ve ortak işyerlerinde çalışan operatörler için hatlar arasındaki yürüme zamanının sıfır olduğu kabul edilmektedir.

PMHD probleminde, her hatta montajı yapılacak olan ürün önceden belirlidir. Eğer hatların her birisi her ürünün montajını yapabilecek esneklikte olurda, ürünlerin hatlara atanması problemide dikkate alınırsa, bu problem *Çok Ürünlü Paralel Montaj Hattı Dengeleme* (ÇÜPMHD) problemi olarak isimlendirilir. ÇÜPMHD problemi, iki alt problemi içermektedir: (i) ürünlerin hatlara atanması (ii) paralel hatların eş zamanlı olarak dengelenmesi.

PMHD probleminde, operatörlerin hatlar arasında yürüme zamanları ihmal edilmektedir. Yani sıfır olarak alınmaktadır. Oysaki, operatörlerin hatlar arasındaki yürüme zamanları hatların eş zamanlı dengelenmesinde çok önemli bir konudur. Yürüme zamanlarının dikkate alındığı PMHD problemi, *Yürüme Zamanlı Paralel Montaj Hattı Dengeleme* (YZPMHD) problemi olarak isimlendirilmiştir.

3.2. Maliyet Tabanlı Montaj Hattı Dengeleme Problemi

Montaj hattı dengeleme ile ilgili kaynaklardaki çalışmalar incelendiğinde, geleneksel MHD probleminin amacı olarak, genellikle sabit bir çevrim zamanı için hat boyunca kullanılacak istasyon sayısını enküçükleme kullanılmıştır [Ghosh ve Gagnon, 1989]. Diğer bir ifadeyle, tüm istasyonlardaki toplam boş zaman enküçülenmeye çalışılmıştır [Baybars, 1986]. Bu nedenle, bu problemlere *zaman tabanlı montaj hattı dengeleme* denilmektedir. Montaj hattı dengeleme çalışmalarında, bölüm 2.4'te verilen diğer amaçlar da kullanılmıştır [Ghosh ve Gagnon, 1989, Scholl, 1999].

Küreselleşmenin etkisiyle sanayide ortaya çıkan acımasız rekabet ile birlikte, üretim maliyetleri daha fazla anlamlı hale gelmiştir. Montaj hattına sahip başarılı üretim sistemlerinde bile, üretim maliyetlerini aşağı çekme çabası bir zorunluluk haline gelmiştir. Yatırım maliyetinin yüksek olduğu bu üretim sistemlerinde düşük kapasite kullanımı ve verimsiz çalışmanın maliyeti oldukça yüksektir. Yukarıda belirtilen ekonomik bakış açısına göre, montaj hatlarının dengelenmesi için kullanılan geleneksel dengeleme yaklaşımları yerine, maliyet tabanlı bir amaç kullanılabilir. Montaj hatlarının maliyet tabanlı bir yaklaşımla birlikte dengelenmesine *maliyet tabanlı montaj hattı dengeleme* denilmektedir. Montaj işlemlerinin yapılabilmesi için gerekli görevlerin, montaj sürecindeki kısıtlara uyarak, üretim maliyetini enküçükleyecek şekilde, sıralı iş istasyonlarına atanması *Maliyet tabanlı Montaj Hattı Dengeleme* (M-MHD) problemi olarak adlandırılmaktadır. M-MHD probleminin amacı, *birim ürün başına (düşen) toplam üretim maliyetini enküçükleme*dir [Rosenberg ve Ziegler 1992, Amen 1997, 2000a, 2000b, 2001].

Bir montaj hattında üretilen birime dair toplam maliyet, işgücü maliyetleri, sermaye maliyetleri, malzeme maliyetleri vb. unsurları içermektedir [Scholl, 1999]. İşçi sayısı hariç, ücret oranları, araç-gereç sayısı vb. diğer maliyet etkenleri zaman tabanlı montaj hattı dengeleme yaklaşımlarında dikkate alınmamıştır. Bundan dolayı, maliyet tabanlı montaj hattı dengeleme çalışmasında, bu maliyet etkenlerinin ayrıntılı bir şekilde analiz edilmesi gerekir.

Otomotil, ev aletleri, kamera gibi ürünlerin üretildiği sektörlerde, nihai ürünü oluşturan parçalarının montajı (son montaj), genellikle otomasyonun fazla olmadığı emek yoğun üretim ortamlarında yapılmaktadır. Emek yoğun üretim ortamlarında, üretim maliyetinin büyük bir kısmını *işgücü maliyeti* oluşturur. Bundan dolayı, bu montaj hatlarının dengeleme çalışmasında ilk olarak, *ücret politikaları* ve *işgücü maliyeti* üzerinde ayrıntılı bir araştırma yapılması gerekir.

Çoğu gelişmiş sanayi ülkelerinde (örn. Almanya endüstrisi) sendikalar ve çalışanlar arasındaki neredeyse tüm ortaklaşa anlaşmalar, bir ücret farklılaşmasını içermektedir. Bir operatörün ücret oranı yaptığı görevlerin zorluk seviyesine (görev değerine) göre değişiklik göstermektedir [Steffen, 1977, Rosenberg ve Ziegler, 1992, Amen, 2000a, b, 2001, 2006, Scholl ve Becker, 2005]. Montaj hattında, bir operasyonu oluşturan farklı *görev değerli* (az ya da çok zorlu) münferit görevler olabilir. Eğer *i* görevi münferit olarak alınırsa, onun görev değerini, birim zaman için bir maliyet oranıyla doğrudan ilişkilendirmek mümkün olabilecektir. Bu değere *görevin ücret oranı* denilmektedir. Son montaj hattındaki görevler zorluklarına göre farklılaştıklarından, ücret oranlarında belirli farklılıklar söz konusu olacaktır. Bir görevin yapılmasındaki zorluk ne kadar fazlaysa, görevin ücret oranının değeri de o kadar yüksek olacaktır. Bir istasyonda çalışan operatörün birim zaman için ücret oranı, o istasyona atanan görevlerin en büyük ücret oranının değerine bağlıdır [Rosenberg ve Ziegler, 1992]. Bir operatöre boş kaldığı zamanlarda dahil olmak üzere, bir çevrim zamanı için ödeme yapılır. Dolayısıyla, bir operatöre ödenen ücreti bulmak için, operatörün ücret oranı ile çevrim zamanı çarpılır. *Toplam işgücü maliyeti* ise, montaj hattında çalışan operatör sayısı dikkate alınarak, her operatöre ödenen ücretlerin toplamıdır.

Montaj hattı üretim sistemlerinde katlanılan diğer bir maliyet ise *malzeme taşıma sisteminin maliyetidir*. Malzeme taşıma maliyeti, direkt olarak montaj hattının uzunluğuna (istasyon sayısına) bağlıdır. Montaj hattında her istasyonun uzunluğunun aynı olduğu varsayılmaktadır.

Diğer bir başka maliyet, *araç-gereç ve makine satın alma maliyetidir*. Montaj hattındaki bir görevi yerine getirmek için *genel* veya *özel amaçlı* ekipmanlara ve

makinelere ihtiyaç olabilir. Özel amaçlı araç-gereçlerin birbirinden farklı olduğu ve her özel amaçlı ekipmanın sadece belirli bir görev için kullanıldığı düşünüldüğünde, bu ekipmanların sayıları ve maliyetleri sabittir. Başka bir ifadeyle, montaj hattında özel amaçlı ekipman ihtiyacı olan her görev için ayrı bir araç-gereç kullanıldığı düşünülürse, bu görevlerin istasyonlara atanmasıyla satın alma maliyetleri değişmez. Bunun yanında, genel amaçlı araç-gereçler ise, her görev için aynı ve ortaktır. Diğer bir ifadeyle, her türlü görevi yaparken kullanılan ekipmanlardır ve bu ekipmanların her istasyonda bulunması gerekir. Dolayısıyla, genel amaçlı ekipmanların sayısı hat uzunluğuna (istasyon sayısına) bağlıdır [[Amen 2000a, b, 2001, 2006](#)].

Son olarak *hammadde maliyeti* ise, görevlerin istasyonlara atanmasına bağlı değildir [[Steffen, 1977](#)].

Yukarıdaki bilgiler doğrultusunda, birim ürün için toplam üretim maliyeti aşağıdaki [Eş. 3.1](#) kullanılarak hesaplanmaktadır:

$$TM = \bar{K} \cdot k_c + \sum_{k=1}^{\bar{K}} c \cdot v s_k \quad , \quad v s_k = \max\{v_i \mid i \in I_k\} \quad , \quad k=1, \dots, \bar{K} \quad (3.1)$$

$\bar{K}$, istasyon sayısını; k_c , bir istasyonun bir çevrim zamanı için toplam sermaye maliyetini, başka bir ifadeyle bir istasyon uzunluğu kadar malzeme taşıma sistemi ve bu istasyonda kullanılacak *genel amaçlı* araç-gereçlerin bir çevrim zamanı için amortisman ve faiz bedelleri toplamını; c , çevrim zamanını; $v s_k$, k istasyonunun birim zaman için ücret oranını; v_i , i görevinin birim zaman için ücret oranını; I_k , k istasyonuna atanan görevler kümesini göstermektedir. Birim ürünün toplam üretim maliyetini daha basit şekilde ifade etmek amacıyla, amaç fonksiyonun ilk parçası $(\bar{K} \cdot k_c)$ yeniden düzenlenirse, $\bar{K} \cdot c \cdot (k_c/c)$ şeklinde yazılabilir. Birim zaman için sermaye maliyeti (k_c/c) ile i görevinin ücret oranı (v_i) birleştirilerek [Eş. 3.2](#)'de görüldüğü gibi yeniden yazılabilir.

$$w_i = v_i + \frac{k_c}{c} \quad , \quad i = 1, \dots, n \quad (3.2)$$

w_i , i görevinin maliyet oranı demektir ve içerisinde hem görev ücret oranını hem de sermaye maliyetini barındırmaktadır. Eş. 3.2'deki görevin maliyet oranı (w_i), görevin ücret oranı (v_i) ve sermaye maliyeti (k_c/c) değerleri birim zaman içindir. n , toplam görev sayısını göstermektedir. Eş. 3.2'de verilen görev maliyet oranı (w_i) dikkate alınarak, bir istasyon için ilgili maliyetler Eş. 3.3'de verildiği gibi yazılabilir:

$$ws_k = \max\{v_i \mid i \in I_k\} + \frac{k_c}{c} = \max\{w_i \mid i \in I_k\} \quad , \quad k = 1, \dots, \bar{K} \quad (3.3)$$

ws_k , k istasyonunun maliyet oranını göstermektedir. Yapılan ek işlemler sonucunda, Eş. 3.1 yeniden düzenlenirse, birim ürün için toplam üretim maliyeti Eş. 3.4'de verildiği gibi hesaplanabilir:

$$TM = \sum_{k=1}^{\bar{K}} c \cdot ws_k \quad , \quad ws_k = \max\{w_i \mid i \in I_k\} \quad , \quad k = 1, \dots, \bar{K} \quad (3.4)$$

Tek modellenli M-MHD probleminin çözümü için Amen (2006) tarafından önerilen karma tamsayılı doğrusal programlama modeli aşağıda verilmiştir. Tamsayılı karar modelinde kullanılan notasyonlar, parametreler ve karar değişkenleri çizelge 3.2 – çizelge 3.4'de açıklanmıştır.

Çizelge 3.2. M-MHD probleminin tamsayılı karar modelinde kullanılan notasyonlar

Notasyon	Açıklama
k	istasyon numarası (indeksi), $k = 1, \dots, \bar{K}$, $\bar{K}$: en fazla istasyon sayısı
i, r, s	görev numarası (indeksi), $i = 1, \dots, n$, n : toplam görev sayısı
P_s	s görevinin öncül görevler kümesi
$r \in P_s$	bir öncelik ilişkisi; r görevi s görevinden önce tamamlanmak zorundadır

Çizelge 3.3. M–MHD probleminin tamsayı karar modelinde kullanılan parametreler

Parametre	Açıklama
c	çevrim zamanı
t_i	i görevinin tamamlanma zamanı
w_i	i görevinin birim zaman için maliyet oranı

Çizelge 3.4. M–MHD probleminin tamsayı karar modelinde kullanılan değişkenler

Değişken	Açıklama
x_{ik}	1, eğer i görevi k istasyonuna atanmış ise; 0, aksi halde
z_k	1, eğer k istasyonu kullanılmış ise; 0, aksi halde
ws_k	k istasyonunun maliyet oranı

$$\text{Enküçükleme } TM = c \cdot \sum_{k=1}^{\bar{K}} ws_k \quad (3.5)$$

$$\sum_{k=1}^{\bar{K}} x_{ik} = 1 \quad , \quad i = 1, \dots, n \quad (3.6)$$

$$\sum_{i=1}^n t_i x_{ik} \leq c z_k \quad , \quad k = 1, \dots, \bar{K} \quad (3.7)$$

$$\sum_{i=1}^n x_{ik} \geq z_k \quad , \quad k = 1, \dots, \bar{K} \quad (3.8)$$

$$z_k \geq z_{k+1} \quad , \quad k = 1, \dots, \bar{K} - 1 \quad (3.9)$$

$$\sum_{k=1}^{\bar{K}} k x_{rk} \leq \sum_{k=1}^{\bar{K}} k x_{sk} \quad , \quad s = 1, \dots, n \quad , \quad r \in P_s \quad (3.10)$$

$$ws_k \geq w_i x_{ik} \quad , \quad i = 1, \dots, n \quad , \quad k = 1, \dots, \bar{K} \quad (3.11)$$

$$ws_k \geq 0 \quad , \quad k = 1, \dots, \bar{K} \quad (3.12)$$

$$x_{ik}, z_k \in \{0, 1\} \quad , \quad i = 1, \dots, n \quad , \quad k = 1, \dots, \bar{K} \quad (3.13)$$

M–MHD probleminin çözümü için kullanılan karma tamsayı doğrusal programlama modelinin amacı (Eş. 3.5), *birim ürünün toplam üretim maliyetini* enküçükmektir. Üretim maliyeti, istasyonlarda kullanılan genel amaçlı araç-gereçlerin, makinelerin amortisman ve faiz değerlerinden (sermaye maliyeti) ve işgücü maliyetinden

oluşmaktadır. Bundan dolayı, Eş. 3.5’de görüldüğü gibi birim ürün için toplam üretim maliyeti, zaman birimi başına işgücü ücret oranı ve zaman birimi başına sermaye maliyetinin toplamından oluşan *istasyon maliyet oranlarının* çevrim zamanıyla çarpımlarının toplamı şeklinde yazılmaktadır. Ayrıca, istasyonlardaki boş zamanlar da düşük kapasite kullanımı nedeniyle maliyet oluşturmaktadır. İstasyon maliyet oranlarının, istasyon zamanlarıyla değil de, boş zamanları da içerecek şekilde çevrim zamanlarıyla çarpılması, amaç fonksiyonunun boş zamanların oluşturduğu maliyetleri de içermesini sağlamaktadır.

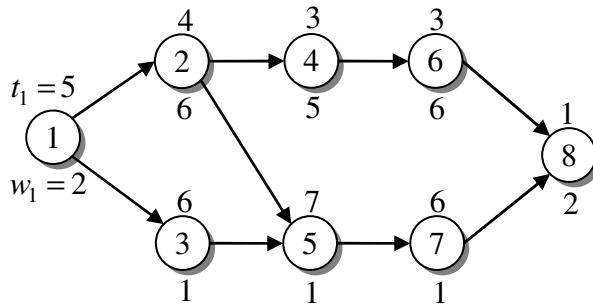
Eş. 3.6’da gösterilen kısıt, her görevin mutlaka ve sadece bir istasyona atanmasını sağlamaktadır. Bu eşitliklere *atama kısıtları* denir. Eş. 3.7’deki kısıt, montaj hattında kullanılan herhangi bir istasyonun toplam iş yükünün, çevrim zamanını aşmamasını sağlamaktadır. Bu eşitsizliklere *çevrim zamanı kısıtları* denir. Eş. 3.7 ve 3.8’de gösterilen kısıt grupları, hangi istasyonların açılması (kullanılması) gerektiğine karar vermektedir.

Eş. 3.9’da verilen kısıt, eğer üretim hattında görev atanmayan (boş) istasyonlar olursa, bu boş istasyonların hattın en sonunda olmalarını sağlamaktadır. Diğer bir anlatımla, en fazla istasyon sayısının $\bar{K}$ olduğu durumda, problemin en iyi çözümünde görevlerin atandığı istasyon sayısı $\bar{K} - 1$ ise, bir istasyon boş olacak demektir. Bu boş istasyon, $[1 - \bar{K}]$ aralığında numaralandırılmış ardışık istasyonlardan herhangi birisi olabilir. Fakat gerçek hayattaki görev atamalarında aradaki bir istasyonun boş olması görülür bir durum olmadığı için, Eş. 3.9’daki kısıt bu boş istasyonun hattın en sonunda oluşmasını sağlamaktadır.

Eş. 3.10’da gösterilen kısıt, ürünün montaj işleminin yapılabilmesi için gerekli olan görevler arasındaki öncelik ilişkilerinin ihlal edilmemesini sağlamaktadır. Eş. 3.11’deki kısıt, istasyonların maliyet oranlarını belirlemektedir. Bir istasyonun maliyet oranı, o istasyona atanan görevlerin en yüksek maliyet oranı değeri düzeyinde belirlenmektedir. Eş. 3.12, karar değişkenlerin negatif değer almamasını

sağlar. Bu kısıtlara *işaret kısıtları* da denir. Eş. 3.13’de belirtilen kısıt, karar değişkenlerinin 0 veya 1 değerini almalarını sağlamaktadır.

Örnek 1: Şekil 3.4’te bir ürüne ait 8 görevli öncelik diyagramı verilmiştir. Öncelik diyagramındaki düğümler görevleri, düğümleri birbirine bağlayan oklar öncelik ilişkilerini, düğümlerin içindeki sayılar görev numaralarını, düğümlerin üzerindeki sayılar görev tamamlanma zamanlarını, düğümlerin altındaki sayılar ise, görevlerin maliyet oranlarını belirtmektedir. Bu örnek problem için çevrim zamanı 10 olarak belirlenmiştir. M–MHD yaklaşımını daha iyi anlamak amacıyla şekil 3.5’te verilen sayısal örnek problem çözdürülmüştür ve maliyet tabanlı yaklaşımla elde edilen çözüm ile zaman tabanlı yaklaşımla elde edilen çözüm, istasyon sayısı ve toplam üretim maliyeti açısından karşılaştırılmıştır.



Şekil 3.5. 8 görevli öncelik diyagramı

Yukarıdaki *tek modelli* M–MHD problemine Amen (2006) tarafından önerilen karma tamsayı doğrusal programlama modeli uygulanarak, örnek problemin en iyi çözümü IBM ILOG CPLEX 12.4 yazılımı yardımıyla bulunmuştur. Çizelge 3.5’in sağ tarafında yer alan M–MHD probleminin en iyi görev atamalarına göre, birim ürünün toplam üretim maliyeti 120 pb ve tüm görevlerin atandığı istasyon sayısı ise 5 olarak bulunmuştur. Bunun yanında, görevlerin maliyet oranları ihmal edilerek aynı örnek problem, istasyon sayısını enküçükleyecek şekilde BMHDP–1 problemi olarak da çözülmüştür. BMHDP–1 probleminin elde edilen en iyi çözümü çizelge 3.5’in sol tarafında verilmiştir. BMHDP–1 probleminin en iyi görev atamalarına göre, kullanılan istasyon sayısı 4 iken, birim ürünün toplam üretim maliyeti 190 pb bulunmuştur.

da içerecek şekilde çevrim zamanıyla çarpılır. Dolayısıyla, istasyonlardaki boş zamanlar nedeniyle oluşan maliyetler de maliyet tabanlı yaklaşımda dikkate alınmıştır.

Özetle; montaj hattının kurulacağı atölyede alan kısıtı yoksa, 5 istasyon kullanılarak toplam üretim maliyetini en düşük seviyeye indirilebileceği görülmüştür. Fakat atölyede ardışık olarak sıralanacak 5 istasyon için yeterli alan yoksa, yukarıdaki modele aşağıdaki Eş. 3.14'te gösterilen yeni kısıt eklenerek çözdürülmelidir.

$$\sum_{k=1}^{\bar{K}} z_k \leq 4 \quad (3.14)$$

Bu kısıtın eklenmesiyle birlikte, tüm görevlerin atandığı istasyon sayısı en fazla 4 olan çözümler arasından en düşük üretim maliyetli görev atamaları bulunur. Bu durumda, zaman tabanlı MHD probleminin 4 istasyonun gerekli olduğu en iyi çözümü bulunmaz. Zaman tabanlı MHD yaklaşımında, üretim maliyetlerinin hiçbiri dikkate alınmadığı için, sadece en az istasyon sayılı çözüm bulunmaktadır. İstasyon sayısı azaldıkça açılış maliyetleri azalacağından, toplam üretim maliyetinin azalması beklenirken, yukarıdaki örneğin en iyi çözümü bunun doğru olmadığını gösterir niteliktedir. Diğer bir deyişle, en düşük maliyetli çözüm, istasyon sayısının en az olduğu çözüm demek değildir. Ayrıca, zaman tabanlı MHD probleminin en iyi çözümüne göre, en az istasyon sayısı 4'tür. Fakat, gerekli istasyon sayısının 4 olduğu birden fazla alternatif en iyi çözümler olabilir. Bu alternatif en iyi çözümlerin görev atamalarının farklı olmasından dolayı, birim ürünün toplam üretim maliyetleri de farklılık gösterir. Zaman tabanlı yaklaşımla, aynı istasyon sayısına sahip alternatif en iyi çözümler arasından toplam üretim maliyetini en küçükleyen çözüm bulunamazken, maliyet tabanlı yaklaşımla birlikte bu en iyi çözümü bulmak mümkündür.

3.3. Maliyet Tabanlı Paralel Montaj Hattı Dengeleme Problemi

Küreselleşmeyle beraber, ülkeler arasındaki ticari sınırlar ortadan kalkmıştır. Bu durum, işletmelerin hem ulusal rakipleriyle hem de uluslararası rakipleriyle sürekli rekabet içinde olmalarına zemin hazırlamıştır. Dolayısıyla işletmeler günümüzde daha sıkı ve zorlu bir rekabet ortamı içinde gelişimlerini ve devamlılıklarını sürdürmek zorundadırlar. İşletmeler kârlarını, pazar paylarını arttırmak ve devamlılıklarını sürdürebilmek için artan tüketici talepleri karşısında, daha hızlı ve daha düşük maliyetli üretim yapabilecekleri üretim sistemlerini geliştirmeleri gerekir.

Yüksek hacimlerdeki standart ürünlerin üretilmesi amacıyla ortaya çıkan kesikli seri üretim sistemlerinden olan montaj hatları, günümüz üretim ihtiyaçlarını karşılayacak şekilde değişime uğramış ve sonuç olarak U tipi montaj hatları, iki taraflı montaj hatları, paralel montaj hatları olmak üzere farklı tasarım alternatifleri ortaya çıkmıştır. Değişen tasarımlar yeni problemleri de beraberinde getirmiştir. [Gökçen ve ark. \(2006\)](#) tarafından ortaya konulan *paralel montaj hattı dengeleme problemi* de bu yeni problemlerden birisi olarak kaynaklarda yerini almıştır.

Montaj hatlarındaki verimlilik artışı, kapasiteyi artıracak ve üretim maliyetlerini azaltacak için, montaj hatlarının düzgün bir şekilde dengelenmesi günümüz endüstrisinde oldukça önem arz etmektedir. PMHD probleminin amacı, işyeri sayısını (veya operatör sayısını) enküçükmektir [[Gökçen ve ark., 2006](#), [Scholl ve Boysen, 2009](#)].

Son yıllarda, araştırmacılar montaj hattı dengeleme problemlerini daha genel ele almaya ve gerçek hayatı daha iyi yansıtan modellemeler geliştirmeye çalışmaktadırlar. Kaynaklardaki yeni çalışmalara bakıldığında, yeni karakteristiklerin ve yeni dengeleme yaklaşımlarının kullanıldığı görülmektedir. Bu tezde, [bölüm 2.7](#)'de anlatılan paralel düz hatların eş zamanlı dengelenmesi için, [bölüm 3.1](#)'de anlatılan zaman tabanlı yaklaşım yerine, [bölüm 3.2](#)'de anlatılan maliyet tabanlı bir yaklaşım kullanılması amaçlanmıştır. Paralel düz montaj hatların eş zamanlı olarak maliyet tabanlı bir yaklaşımla dengelendiği bu problem, *Maliyet tabanlı Paralel*

Montaj Hattı Dengeleme (M-PMHD) problemi olarak isimlendirilmiştir. M-PMHD probleminin temel amacı; her montaj hattında üretilen her birim ürünün toplam üretim maliyetini enküçüklenmesidir.

M-PMHD probleminin kısıtları ve varsayımları aşağıda verilmiştir:

- Her bir montaj hattında sadece bir çeşit ürün üretilmektedir.
- Her bir montaj hattında üretilecek ürün belirlidir.
- Belirli bir montaj hattında üretilecek belirli bir ürüne ait görevler, başka hattın istasyonuna atanamaz. Diğer bir ifadeyle, belirli bir ürüne ait görevler, sadece montajı yapılan hattın istasyonlarına atanabilir.
- Bir ürüne ait görevlerin her biri sadece bir istasyona atanabilir.
- Her bir ürünün öncelik diyagramları bilinmektedir.
- Herhangi bir ürüne ait bir görev, ancak kendisinden önceki aynı ürüne ait tüm öncül görevler tamamlandıktan sonra başlayabilmektedir.
- Her bir ürün için görev tamamlanma zamanları (deterministik) belirli ve sabittir.
- Tüm görevler tam zamanında tamamlanmaktadır.
- Her bir ürüne ait görevlerin maliyet oranları belirli ve sabittir.
- Her bir görevin maliyet oranı, görev ücret oranını ve birim zaman için ilgili hattın sermaye maliyetini içermektedir.
- Her bir görevin maliyet oranı birim zaman içindir.
- Her montaj hattı için çevrim zamanı belirli ve tüm hatlar için aynıdır. Eğer, paralel hatlarda eş zamanlı olarak üretilen ürünler için çevrim zamanları birbirinden farklı olursa, [bölüm 3.1](#)'de anlatılan [Gökçen ve ark. \(2006\)](#) tarafından önerilen normalizasyon yöntemi ile tüm hatlar için ortak bir çevrim zamanı bulunur.
- Herhangi bir hattın herhangi bir istasyonunda çalışan bir operatörün iş yükü çevrim zamanını geçemez.
- Her hattın her istasyonunda sadece bir operatör çalışmaktadır.
- Paralel hatlarda çalışan tüm operatörler her türlü görevi yapabilecek bilgi ve beceriye sahiptir (operatörler esnektir).
- Birbirine paralel konumda olan düz hatların her iki tarafı da çalışmaya uygundur.

- Paralel hatların birbirine yakın konumda oldukları varsayıldığı için, hatlar arasındaki operatör yürümleri dikkate alınmamaktadır.
- Sadece yan yana olan hatlar arasında ortak işyerleri kurulabilir.
- Her bir operatör aynı çevrim zamanı içerisinde en fazla iki hatta çalışabilir.
- h hattının k . istasyonunda çalışan bir operatörün, h hattının yanında olan $h-1$ veya $h+1$ hatlarından sadece birinin sadece k . istasyonunda çalışabilmesine izin verilir.
- Hatların güvenilirliği tamdır.
- İstasyonlar arası stok tutulmasına izin verilmemektedir.

Eğer görevlerin maliyet oranları aynı alırsa, bu durumda M-PMHD problemi PMHD problemine dönüşmektedir. Bu nedenle, *maliyet tabanlı* PMHD problemi, *zaman tabanlı* PMHD probleminin genelleştirilmiş halidir. PMHD, NP-zor sınıfında olan bir kombinatoriyal optimizasyon problemidir [Gökçen ve ark., 2006, Scholl ve Boysen, 2009]. Dolayısıyla, M-PMHD problemi de NP-zor bir problemidir.

M-PMHD probleminde, her hatta montajı yapılacak olan ürün önceden belirlidir. Başka bir ifadeyle, paralel düz hatlar her ürünün montajını yapabilecek esnekliğe sahip değildir. Bu nedenle, M-PMHD probleminde, ürünlerin hatlara atanması problemi ele alınmaz. Eğer hatların her biri her ürünün montajını yapabilecek esneklikte olursa, ürünlerin hatlara atanması problemi de ele alınmalıdır. Ürünlerin hatlara atanmasını da dikkate alarak, toplam üretim maliyetini enküçükleyecek şekilde birden fazla düz hattın eş zamanlı olarak dengelendiği bu problem, *Maliyet tabanlı Çok Ürünlü Paralel Montaj Hattı Dengeleme* (M-ÇÜPMHD) problemi olarak isimlendirilmiştir. Ayrıca, operatörlerin paralel hatlar arasındaki yürüme zamanları dikkate alındığında, toplam üretim maliyetinin nasıl etkilendiği incelenmiştir. Yürüme zamanlarının dikkate alındığı M-PMHD problemi, *Maliyet tabanlı Yürüme Zamanlı Paralel Montaj Hattı Dengeleme* (M-YZPMHD) problemi olarak isimlendirilmiştir.

3.4. Kaynak Araştırması

Montaj hatlarının dengelenmesi düşüncesi ilk kez Bryton'ın yüksek lisans tezinde yer almıştır [Bryton, 1954]. Yayınlanmış ilk bilimsel çalışma ise Salveson'a aittir [Salveson, 1955]. Salveson'un çalışmasından sonra günümüze kadar bu konuda pek çok çalışma yapılmıştır. MHD probleminin en basit hali, düz hatlı ve tek modelli montaj hattı dengeleme (BMHD) problemidir. Kaynaklarda, BMHD problemi için kesin çözüm yöntemleri, sezgisel ve meta-sezgisel yöntemleri içeren çok sayıda çalışma vardır. Konuyla ilgili daha geniş bilgi için kaynak tarama çalışmalarına bakılabilir: [Baybars, 1986, Ghosh ve Gagnon, 1989, Erel ve Sarin, 1998, Scholl ve Becker, 2006, Becker ve Scholl, 2006].

Tek modelli U tipi MHD problemi konusunda yapılan çalışmalar, BMHD problemiyle ilgili çalışmalara göre daha az sayıdadır. U tipi montaj hatları ilk defa Miltenburg ve Wijngaard tarafından ele alınmıştır [Miltenburg ve Wijngaard, 1994]. Tek modelli U tipi MHD problemi için tamsayı programlama modeli ilk kez Urban tarafından önerilmiştir [Urban, 1998]. Bugüne kadar U tipi MHD problemi için çeşitli çözüm teknikleri geliştirilmiştir: [Nakade ve ark., 1997, Ajenblit ve Wainwright, 1998, Miltenburg, 1998, Sparling, 1998, Scholl ve Klein, 1999, Miltenburg, 2000, Erel ve ark., 2001, Aase ve ark., 2003, Gökçen ve ark., 2005, Gökçen ve Ağpak, 2006, Toklu ve Özcan, 2008, Sabuncuoğlu ve ark., 2009, Kara ve ark., 2009]. Ayrıca, tek modelli U tipi MHD probleminin çözümü için geliştirilen kesin çözüm yöntemleri ve sezgisel yöntemler konusunda ayrıntılı kaynak taraması için Becker ve Scholl'in çalışmasına bakılabilir [Becker ve Scholl, 2006].

Karışık modelli düz hatlı MHD problemi konusunda şu çalışmalara bakılabilir: [Thomopoulos, 1967, 1970, Macaskill, 1972, Dar-El ve Cother, 1975, McMullen ve Frazier, 1997, 1998, Gökçen ve Erel, 1997, 1998, Erel ve Gökçen, 1999, Jin ve Wu, 2002, Vilarinho ve Simaria, 2002, 2006, Simaria ve Vilarinho, 2004, Bukchin ve Rabinowitch, 2006]. Karışık modelli U tipi MHD problemi ilk defa Sparling and Miltenburg tarafından ele alınmıştır [Sparling and Miltenburg, 1998]. Kaynaklardaki diğer çalışmalar: [Kim ve ark., 2000, Miltenburg, 2002, Kim ve ark., 2006, Kara ve

ark., 2007a, 2007b, Hwang ve ark., 2008, Rabbani ve ark., 2012]. Ayrıntılı kaynak taraması için şu çalışmalara bakılabilir: [Becker ve Scholl, 2006, Lusa, 2008 ve Boysen ve ark., 2009].

Bu tez kapsamında ortaya konulan M-PMHD problemleri ile ilgili kaynaklarda herhangi bir çalışma bulunmamaktadır. M-PMHD problemi, temelde kaynaklarda yer alan M-MHD ve PMHD problemlerinin birlikte ele alınmış hali olduğu için, kaynaklardaki M-MHD ve PMHD problemleri ile ilgili çalışmalar ayrıntılı olarak incelenmiştir.

3.4.1. Paralel montaj hattı dengeleme problemi

Montaj hattı dengeleme kaynaklarındaki şu ana kadar yapılan çalışmaların çoğu, kurulan bir montaj hattında bir ürünün ya da birden fazla farklı ürünün aynı hatta üretilmesi ve bu durum temel olmakla birlikte bununla ilgili değişik durumları yansıtmaktadır. Diğer bir ifadeyle, bu çalışmaların tamamına yakınında bir montaj hattının dengelenmesi ele alınmıştır. Kaynaklarda çalışmanın konusu ile isim olarak benzerlik gösteren paralel hatlar konusunda yapılan çalışma sayısı oldukça azdır. Bu çalışmalarda, bir montaj hattının talebi karşılayamaması durumunda, kaç tane özdeş paralel hattın kurulması gerektiği ele alınmıştır. Ancak kurulacak paralel hatların dengesi bağımsız olarak ayrı ayrı yapılmaktadır.

Süer ve Dağlı, paralel montaj hattında sezgisel bir prosedür tasarlamış ve dinamik olarak hat sayısını belirleyen bir algoritma geliştirmişlerdir [Süer ve Dağlı, 1994]. Aynı zamanda Süer, tek ürün için alternatif montaj hattı tasarım stratejileri üzerine de çalışmıştır. Bu çalışmasındaki amacı, en küçük toplam işgücü ve montaj hattı sayısını belirlemektir. Bunun için, 3 fazlı bir metodoloji önermiştir: (1) montaj hattının dengesi, (2) paralel istasyonların belirlenmesi, (3) paralel hatların belirlenmesi [Süer, 1998].

Montaj hattın etkinliğinin artırılması için görevler ya da istasyonların paralel hale getirilmesi (paralellenmesi) durumuna şu çalışmalarda rastlanabilir: [Freeman ve

Jucker, 1967, Reiter, 1969, Buxey, 1974, Pinto ve ark., 1975, Akagi ve ark., 1983, Johnson, 1983, Sarker ve Shantikumar, 1983, Bard, 1989, Askin ve Zhou, 1997, McMullen ve Frazier, 1997, Bukchin ve Tzur, 2000, Simaria ve Vilarinho, 2001, Bukchin ve Rubinovitz, 2002, Vilarinho ve Simaria, 2002]. Bunların bir kısmı, dengeleme esnasında paralel yapılacak görev ya da istasyonlardan kaynaklanan görev ve ekipman maliyetlerini de dikkate alarak modellerini geliştirmişlerdir. Şu ana kadar paralellikle ilgili yapılan çalışmaların tamamı, dengeleme esnasında görev ya da istasyonların paralel yapılmasıyla ilgili çalışmalardır. Paralel hatlar üzerindeki bu çalışmalar Gökçen ve ark. (2006) yaklaşımından mantıksal olarak farklıdır. Birden fazla hattın eş zamanlı dengelenmesi düşüncesi ilk defa Gökçen ve ark. tarafından ortaya atılmıştır. Bu çalışmada, üretim sisteminin verimliliğinin ve kaynak kullanımının artırılması için, paralel bitişik hatların bazı istasyonlarının iş yüklerinin birleştirilmesi ele alınmıştır. Ayrıca, PMHD probleminin en iyi çözümü için 0-1 tamsayılı programlama modeli ve büyük boyutlu problemlerin yaklaşık çözümleri için iki tane sezgisel algoritma önerilmiştir [Gökçen ve ark., 2006].

Benzer ve ark., PMHD problemi için en kısa yol formülasyonunu kullanan bir network modeli önermiştir. Bu çalışma içerisinde küçük boyutlu PMHD problemi için en iyi çözüm veren bir ağ çözümü yaklaşımı ortaya konulmuştur [Benzer ve ark., 2007].

Çerçioğlu ve ark. çalışmasında, PMHD problemi için tavlama benzetimi (TB) tabanlı bir yaklaşım geliştirilmiştir. Önerilen yaklaşım, sayısal bir örnekle açıklanmış ve bu yaklaşımın performansı Gökçen ve ark. (2006) çalışmasındaki test verileri üzerinde denenmiştir [Çerçioğlu ve ark., 2009].

Scholl ve Boysen çalışmasında, PMHD probleminin ayrıntılı tanımını vererek, bu probleminin çözümü için 0-1 tamsayılı doğrusal programlama modeli önermişlerdir. BMHD problemi için geliştirilen SALOME isimindeki dal-sınır algoritması PMHD problemine uyarlanmıştır [Scholl ve Boysen, 2009].

Kara ve ark., öncelikli ve bulanık hedefleri ile paralel montaj hatlarını dengelemek için iki farklı hedef programlama modeli önermişlerdir. Hedef programlama yaklaşımlarında birbiriyle çelişen 3 tane amaç (*istasyon sayısı, çevrim zamanı, bir istasyona atanan iş sayısı*) için uzlaşık çözümler aranmıştır. Önerilen yaklaşımlar, sayısal bir örnek ve duyarlılık analiziyle açıklanmıştır. Bu önerilen yaklaşımlar karar vericiye karar verme anında esneklik sağlamaktadır [Kara ve ark., 2009].

Özcan ve ark. çalışmasında, birbirine paralel olarak yerleştirilmiş birden fazla çift taraflı montaj hatları ele alınmıştır. Paralel çift taraflı montaj hatlarının eşzamanlı dengelenmesi için Tabu Arama meta-sezgisel algoritması önerilmiştir. Önerilen algoritmanın etkinliği değerlendirmek için kaynaklardaki test problemleri çözülmüştür [Özcan ve ark., 2009a].

Özcan ve ark., birbirine paralel olan her düz hatta birbirinden farklı birden fazla ürünün üretilmesi ve ürünlerin sıralanması ele alınmıştır. Paralel hatların eş zamanlı olarak dengelendiği bu problem, paralel karışık modellenmiş montaj hattı dengeleme ve model sıralama problemi olarak isimlendirilmiştir. Ele alınan problemin çözümü için Tavlama Benzetimi algoritması önerilmiştir [Özcan ve ark., 2009b].

Hat verimliliği enbüyüklenme ve istasyon iş yüklerinin değişimini enküçüklenme amacı ile PMHD probleminin çözümü için Özcan ve ark. tarafından çok amaçlı tabu arama algoritması geliştirilmiştir. Algoritmanın etkinliğini ölçmek için kaynaklarda kullanılan test problemleri üzerinde deneyler yapmışlardır [Özcan ve ark., 2009c].

Ozbakır ve ark., istasyonlardaki boş zamanları enküçüklenme ve istasyonların etkinliğini enbüyüklenme amaçları ile paralel hatların eş zamanlı dengelenmesi için yeni çok kolonili karınca algoritması önermişlerdir. Önerilen algoritma ile kaynaklardaki karşılaştırma amacıyla kullanılan test problemlerini çözülürken, kaynaklardaki diğer algoritmalar ile performans karşılaştırması yapmışlardır [Ozbakır ve ark., 2011].

3.4.2. Maliyet tabanlı montaj hattı dengeleme problemi

[Steffen \(1977\)](#) çalışmasında, bir istasyonda çalışan işçinin saatlik ücret oranının, istasyonuna atanan görevlerin ücret oranlarına farklı şekillerde bağlı olabileceğini belirtmiştir. Bir işçinin ücret oranının, o istasyona atanan görevlerin ücret oranlarının *ortalamasına, ağırlıklı ortalamasına, en büyük ücret oranı değerine veya ortanca değere* bağlı olarak değişebileceğini belirtmiştir [[Steffen, 1977](#)].

Rosenberg ve Ziegler, maliyet tabanlı MHD probleminin tanımını yaparak, bir işçinin ücret oranının, o istasyona atanan görevlerin en büyük ücret oranı değerine göre belirleneceğini belirtmiştir. Maliyet tabanlı MHD problemi için iki farklı sezgisel algoritma önermiştir. Kaynaklarda, MHD probleminin çözümü için kullanılan basit sezgisel algoritmalar ile önerdikleri sezgisel algoritmaların performanslarının test problemleri üzerinde karşılaştırmalarını yapmışlardır [[Rosenberg ve Ziegler, 1992](#)].

Askin and Zhou, görevler ile ilişkili takım gereksinimleri dikkate alınarak paralel istasyonların açılmasına izin verilen bir matematiksel model geliştirmişlerdir. Dikkate aldıkları amaç ise, gerekli ekipmanların amortisman değerlerini ve istasyonların sabit açılma maliyetlerini enküçükmektir [[Askin and Zhou, 1997](#)].

Amen, maliyet tabanlı MHD probleminin tanımını yaparak, karakteristik özelliklerinden bahsetmiştir. [Rosenberg ve Ziegler \(1992\)](#)'in çalışmasında ele alınan maliyetlere sermaye maliyetini de eklemiştir. Maliyet tabanlı MHD problemini, *maksimum doluluk istasyon baskın kuralına* göre çözüldüğünde, problemin en iyi çözümünü bulunamayacağını belirterek, bu kriter yerine *iki istasyon kuralına* göre çözülmesi gerektiğini belirtmiştir. Maliyet tabanlı MHD probleminin en iyi çözümünü bulabilmek için 7 tane baskın kural tanımlamıştır. Ayrıca, tanımladığı 7 baskın kuralı kullanan dal-sınır algoritması da önermiştir [[Amen, 2000a](#)].

Amen'nin diğer çalışmasında, MHD probleminin çözümü için kullanılan kaynaklardaki sezgisel algoritmaları sınıflandırarak, maliyet tabanlı MHD problemini çözmek için yeni sezgisel algoritmalar önermiştir. Sezgisel algoritmaların kullandığı öncelik kurallarına yeni bir tanesini eklemiştir [[Amen, 2000b](#)].

[Amen \(2001\)](#) çalışması, daha önceki [Amen \(2000b\)](#) çalışmasının devamı niteliğindedir. [Amen \(2000b\)](#) çalışmasında sınıflandırmasını yaptığı tüm sezgisel algoritmalar ile önerdiği algoritmanın test problemleri üzerinde çözüm süresi ve çözüm kalitesine göre karşılaştırmalarını yapmıştır [[Amen, 2001](#)].

Scholl ve Becker çalışmasında, [Amen \(2000a\)](#) tarafından M-MHD probleminin çözümü için önerilen Dal-Sınır algoritmasının kullandığı baskın kurallardan bir tanesinin yanlış olduğunu belirterek, düzeltmiştir. Baskın kurallara yeni bir tanesini eklemiştir [[Scholl ve Becker, 2005](#)].

[Amen \(2006\)](#) çalışmasında, M-MHD probleminin çözümü için karma tamsayılı programlama model geliştirmiştir. Bu modelin kısıtlarında bazı geliştirmeler önermiştir. Ayrıca, M-MHD probleminin çözümünün, zaman tabanlı MHD problemine göre daha zor olduğunu kanıtlamıştır [[Amen, 2006](#)].

Padrón ve ark. çalışmasında, görev yeri ve bölgesel kısıtlar ile birlikte ele alınan M-MHD problemini çözmek için sezgisel ve kesin çözüm yöntemini birleştiren bir algoritma önermiştir [[Padrón ve ark., 2009](#)].

Yazdanparast ve ark., hazırlık zamanını dikkate alan genel MHD probleminin maliyet tabanlı bir amaç fonksiyonuyla dengelenmesi için tamsayılı programlama modeli önermişlerdir. Önerilen matematiksel model açıklayıcı bir örnek üzerinde gösterilmiştir [[Yazdanparast ve ark., 2011](#)].

Roshani ve ark., iki taraflı montaj hatlarının maliyet tabanlı bir şekilde dengelenmesi için karma tamsayılı programlama modeli ve yaklaşık çözümleri makul sürelerde bulabilmek için Tavlama Benzetimi sezgisel algoritması önermişlerdir. [[Roshani ve ark., 2012](#)].

4. M-PMHD PROBLEMLERİ İÇİN YENİ MODELLER

PMHD problemine özgü farklı özellikler ve paralel hatların eş zamanlı dengelenmesi için maliyet tabanlı yaklaşım dikkate alınarak, montaj hattı dengeleme literatürüne yeni problemlerin kazandırılması düşünülmüştür. Bu yeni problemler [bölüm 3.3](#)'de ayrıntılı olarak anlatılan M-PMHD, M-YZPMHD ve M-ÇÜPMHD problemleridir. Bu bölümde, ele alınan yeni problemlerin en iyi çözümlerinin bulunabilmesi için birbirinden farklı yapıya sahip yeni tamsayılı programlama modelleri sunulmuştur. Önerilen tamsayılı programlama modelleri kullanılan karar değişkenlerinin anlamlarına göre *istasyon odaklı* ve *operatör odaklı* olarak adlandırılmıştır. *İstasyon odaklı yaklaşımda*, paralel hatların her birinde belirli bir ürünün üretilebilmesi için gerekli olan görevlerin ilgili hattın istasyonlarına ataması yapılırken, *operatör odaklı yaklaşımda* ise, bu görevlerin paralel hatlarda kurulan ayrık veya ortak işyerlerinde çalışan operatörlere ataması yapılmaktadır. Geliştirilen karma tamsayılı doğrusal programlama modelleri aşağıda listelenmiştir:

- M-PMHD problemi için *istasyon odaklı* ve *operatör odaklı* tamsayılı programlama modelleri
- M-YZPMHD problemi için *istasyon odaklı* ve *operatör odaklı* tamsayılı programlama modelleri
- M-ÇÜPMHD problemi için *istasyon odaklı* ve *operatör odaklı* tamsayılı programlama modelleri
- Operatör ücret oranlarının düzgünleştirilmesi için tamsayılı programlama modeli
- PMHD problemi için birbiriyle çelişen amaçlar için uzlaşık çözümler bulan bir hedef programlama modeli

Ayrıca, bu tamsayılı karar modellerdeki bazı kısıtlar üzerinde modifikasyonlar yapılarak yeni kısıtlar ve toplam işgücü maliyeti için alt sınır değeri bulan bir algoritma önerilmiştir.

Paralel olarak konumlandırılmış düz montaj hatlarının her birinde belirli bir çeşit ürünün montaj işlemi yapılmaktadır. Dolayısıyla, M-PMHD problemlerinin çözümü için önerilen tamsayılı programlama modellerinin temel amacı, *her montaj hattında üretilen her birim ürünün toplam üretim maliyetini enküçükleme*dir. [Bölüm 3.2](#)'de açıklandığı üzere toplam üretim maliyeti, operatörlerin ücret oranları ve sermaye maliyetinden oluşmaktadır. Toplam üretim maliyetini oluşturan her maliyet unsurunun (toplam işgücü maliyeti, malzeme taşıma sistemi maliyeti, genel amaçlı araç-gereç ve makine maliyeti) aralarındaki ilişkiler hedef programlama modelinin sunulduğu bölümde anlatılmıştır. Bu ayrıntılı analizin yapılabilmesi için her maliyet unsuru için eniyileme çalışması yapılmıştır. Bu amaçla tamsayılı karar modellerinde kullanılan amaç, her maliyet unsurunu teker teker eniyileyecek şekilde ele alınarak, değişik çözüm desenleri elde edilmeye çalışılmıştır. Bu bölümün en sonunda toplam üretim maliyetinin enküçüklenmesi için kullanılan amaç fonksiyonu bütün halde verilmiştir. Öncelikle toplam üretim maliyeti içerisinde büyük bir paya sahip olan *toplam işgücü maliyetinin* enküçüklenmesi ele alınmıştır.

Önerilen tamsayılı karar modelleri kaynaklarda sıkça kullanılan açıklayıcı örnek problemler üzerinde uygulanarak, IBM ILOG CPLEX 12.4 yazılımı yardımıyla çözülmüştür. Elde edilen en iyi çözümler aynı problemin *zaman tabanlı yaklaşımla* elde edilen en iyi çözümleri ile *işyeri sayısı (operatör sayısı)* ve *toplam işgücü maliyeti* açısından karşılaştırılmıştır.

4.1. M-PMHD Problemi için İstasyon Odaklı Tamsayılı Programlama Modeli

Bu çalışmada bahsedilen M-PMHD problemi, her düz hatta montajı yapılacak olan ürünün belirli olduğu durumda, çevrim zamanı ve öncelik ilişkileri kısıtlarını ihlal etmeden, toplam işgücü maliyetini enküçükleyecek şekilde, belirli ürünlere ait görevlerin ilgili hattın sıralı istasyonlarına atanması olarak modellenmiştir. Önerilen istasyon odaklı tamsayılı programlama modelinde kullanılan notasyonlar, parametreler ve karar değişkenleri [çizelge 4.1 – çizelge 4.3](#)'te açıklanmıştır.

Çizelge 4.1. M-PMHD probleminin istasyon odaklı tamsayılı karar modelinde kullanılan notasyonlar

Notasyon	Açıklama
h	hat numarası (indeksi), $h = 1, \dots, \bar{H}$, $\bar{H}$: en fazla hat sayısı
k	istasyon numarası (indeksi), $k = 1, \dots, \bar{K}$, $\bar{K}$: en fazla istasyon sayısı
n_h	h hattında montajı yapılacak ürüne ait görev sayısı
i, j, r, s	görev numarası (indeksi), $i = 1, \dots, n_h$
P_{hi}	h hattında montajı yapılacak ürüne ait i görevinin öncül görevler kümesi
$(r, s) \in P_{hs}$	bir öncelik ilişkisi; h hattındaki r görevi s görevinden önce tamamlanmalıdır

Çizelge 4.2. M-PMHD probleminin istasyon odaklı tamsayılı karar modelinde kullanılan parametreler

Parametre	Açıklama
c	çevrim zamanı
t_{hi}	h hattında montajı yapılacak ürüne ait i görevinin tamamlanma zamanı
w_{hi}	h hattında montajı yapılacak ürüne ait i görevinin birim zaman için ücret oranı

Çizelge 4.3. M-PMHD probleminin istasyon odaklı tamsayılı karar modelinde kullanılan değişkenler

Değişken	Açıklama
x_{hik}	1, eğer h hattında üretilecek olan ürüne ait i görevi k istasyonuna atanmış ise; 0, aksi halde
u_{hk}	1, eğer h hattındaki k istasyonu kullanılmış ise; 0, aksi halde
$z_{h(h+1)k}$	1, eğer h ve $h+1$ hatlarının komşu olan k istasyonları arasında ortak işyeri kurulmuş ise; 0, aksi halde
wn_{hk}	h hattındaki k istasyonuna kurulan ayrık işyerinde çalışan operatörün ücret oranı
$ws_{h(h+1)k}$	h ve $h+1$ hatlarının komşu olan k istasyonları arasında kurulan ortak işyerinde çalışan operatörün ücret oranı

M-PMHD problemi için önerilen istasyon odaklı tamsayılı doğrusal programlama modeli aşağıdaki gibidir.

$$\text{Enküçüklem} \quad TM = c. \sum_{h=1}^{\bar{H}} \sum_{k=1}^{\bar{K}} w n_{hk} + c. \sum_{h=1}^{\bar{H}-1} \sum_{k=1}^{\bar{K}} w s_{h(h+1)k} \quad (4.1)$$

$$\sum_{k=1}^{\bar{K}} x_{hik} = 1 \quad , \quad i = 1, \dots, n_h \quad , \quad h = 1, \dots, \bar{H} \quad (4.2)$$

$$\sum_{i=1}^{n_h} t_{hi} x_{hik} + \sum_{i=1}^{n_{h+1}} t_{(h+1)i} x_{(h+1)ik} \leq c. (u_{hk} + u_{(h+1)k} - z_{h(h+1)k}) \quad , \quad k = 1, \dots, \bar{K} \quad , \quad h = 1, \dots, \bar{H} - 1 \quad (4.3)$$

$$\sum_{i=1}^{n_h} t_{hi} x_{hik} \leq c. u_{hk} \quad , \quad k = 1, \dots, \bar{K} \quad , \quad h = 1, \dots, \bar{H} \quad (4.4)$$

$$\sum_{i=1}^{n_h} x_{hik} \geq u_{hk} \quad , \quad k = 1, \dots, \bar{K} \quad , \quad h = 1, \dots, \bar{H} \quad (4.5)$$

$$z_{h(h+1)k} + z_{(h+1)(h+2)k} \leq 1 \quad , \quad \bar{H} \geq 3 \quad , \quad h = 1, \dots, \bar{H} - 2 \quad , \quad k = 1, \dots, \bar{K} \quad (4.6)$$

$$\sum_{k=1}^{\bar{K}} (\bar{K} - k + 1) (x_{hrk} - x_{hsk}) \geq 0 \quad , \quad \forall (r, s) \in P_{hs} \quad , \quad h = 1, \dots, \bar{H} \quad (4.7)$$

$$w n_{hk} \geq \begin{cases} w_{hi} (x_{hik} - z_{h(h+1)k}) & , h = 1, i = 1, \dots, n_h, k = 1, \dots, \bar{K} \\ w_{hi} (x_{hik} - z_{(h-1)hk} - z_{h(h+1)k}) & , 1 < h < \bar{H}, i = 1, \dots, n_h, k = 1, \dots, \bar{K} \\ w_{hi} (x_{hik} - z_{(h-1)hk}) & , h = \bar{H}, i = 1, \dots, n_h, k = 1, \dots, \bar{K} \end{cases} \quad (4.8)$$

$$w s_{h(h+1)k} \geq w_{hi} (x_{hik} - z_{h(h+1)k} - u_{hk}) \quad , \quad i = 1, \dots, n_h \quad , \quad k = 1, \dots, \bar{K} \quad , \quad h = 1, \dots, \bar{H} - 1 \quad (4.9)$$

$$w n_{hk}, w s_{h(h+1)k} \geq 0 \quad , \quad k = 1, \dots, \bar{K} \quad , \quad h = 1, \dots, \bar{H} \quad (4.10)$$

$$x_{hik}, z_{h(h+1)k}, u_{hk} \in \{0, 1\} \quad , \quad i = 1, \dots, n_h \quad , \quad k = 1, \dots, \bar{K} \quad , \quad h = 1, \dots, \bar{H} \quad (4.11)$$

Amaç fonksiyonu (Eş. 4.1), toplam işgücü maliyetini enküçüklemektir. Herhangi bir (ayrık veya ortak) işyerinde çalışan operatörün ücret oranı, o işyerine atanan görevlerden en yüksek ücret oranı değerine göre belirlenmektedir. Dolayısıyla, amaç fonksiyonu ayrık ve ortak işyerlerinde çalışan operatörlerin ücret oranlarının çevrim zamanıyla çarpımlarının toplamı şeklinde yazılmaktadır.

Eş. 4.2’de gösterilen kısıt, belirli bir ürüne ait görevlerin her birinin, kesinlikle montaj işleminin yapıldığı hattaki sadece bir istasyona atanmasını sağlamaktadır. Her görev mutlaka bir istasyona atanmalıdır.

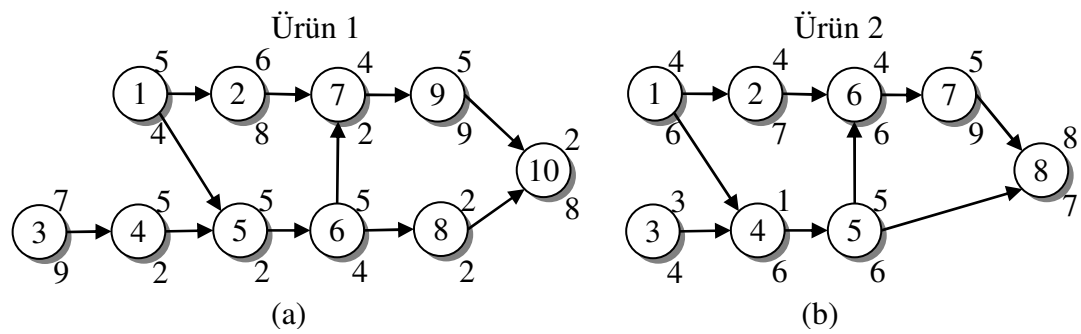
Eş. 4.3 ve 4.4'deki grup kısıtlar, herhangi bir istasyonun iş yükünün, daha önceden talebe bağlı olarak belirlenen çevrim zamanını aşmamasını sağlamaktadır. Eş. 4.3'deki kısıt, yan yana olan hatlardaki komşu istasyonların iş yükleri birleştirilmişse, ortak olarak kullanılan bu istasyonların toplam iş yükünün çevrim zamanını aşmamasını sağlarken, Eş. 4.4'deki kısıt ise, ayırık istasyonların her birinin iş yükünün çevrim zamanını aşmamasını sağlamaktadır. Eş. 4.5'deki kısıt, hatlardaki hangi istasyonların kullanılması gerektiğine karar vermektedir.

Eş. 4.6'daki kısıt, h hattının k istasyonunda çalışan bir operatörün, aynı çevrim zamanı içerisinde h hattının yanında bulunan ($h-1$ veya $h+1$) hatlarından sadece bir tanesinin k istasyonunda da çalışmasına izin verir. Başka bir ifadeyle, h ve $h-1$ hatlarının k istasyonlarında bir operatör çalışırken, aynı zamanda h ve $h+1$ hatlarının k istasyonlarında da başka bir operatör çalışamaz. Bu durumda h hattının k istasyonunda iki operatör çalışıyor olur ki, bu durum her istasyonda bir operatör çalışabilir varsayımıyla çelişir. Her hatta üretilen ürünle ilgili görevler arasındaki öncelik ilişkileri, Eş. 4.7'deki kısıt ile sağlamaktadır. Her hatta üretilen ürün birbirinden farklı olabileceği için, bu öncelik ilişkilerini sağlayan kısıtlar her hattaki görevler için birbirinden bağımsız olarak yazılmaktadır.

Eş. 4.8 ve 4.9'daki grup kısıtlar, operatörlerin ücret oranlarını belirlemektedir. Eş. 4.8'deki kısıt, hatlardaki ayırık işyerlerinde çalışan operatörlerin ücret oranlarını belirlerken, Eş. 4.9'daki kısıt, ortak işyerlerinde çalışan operatörlerin ücret oranlarını belirlemektedir. Bir operatörün ücret oranı, çalıştığı istasyona atanan görevlerin en yüksek ücret oranı değeri düzeyinde belirlenir. Dolayısıyla, bir hattın ayırık istasyonunda çalışan bir operatörün ücret oranı belirlenirken, sadece o istasyona atanan görevlerin ücret oranları dikkate alınır. Bunun yanında bitişik hatların komşu istasyonlarında ortak olarak çalışan bir operatörün ücret oranı, birbirine komşu olan her iki istasyona atanan görevlerin ücret oranları dikkate alınarak belirlenir.

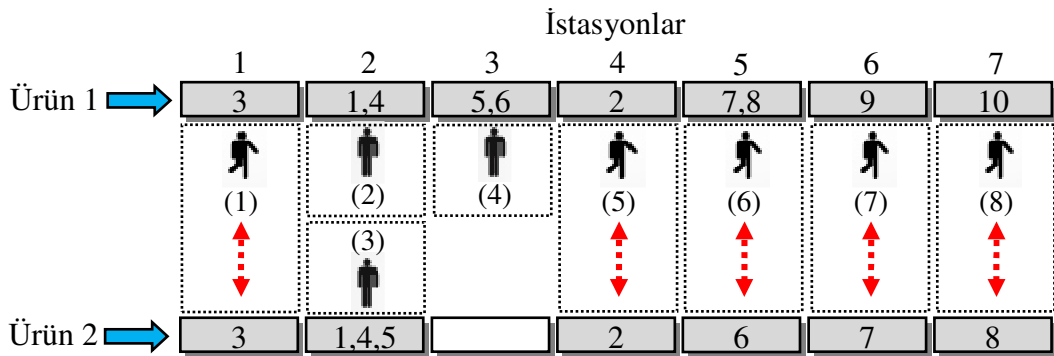
Eş. 4.10, karar değişkenlerin negatif değer almamasını sağlarken, Eş. 4.11 ise, karar değişkenlerinin 0 veya 1 değerini almalarını sağlamaktadır.

Örnek 2: Birbirine paralel olan iki düz hatlı bir üretim sisteminde iki farklı ürünün montaj işlemleri yapılmaktadır. 10 görevden oluşan ürün 1 ve 8 görevden oluşan ürün 2'nin montaj işlemleri sırasıyla hat 1 ve hat 2'de yapılmaktadır. Ürünlerin öncelik diyagramları [şekil 4.1\(a\)](#) ve [şekil 4.1\(b\)](#)'de verilmiştir. Düğümlerin üstündeki sayılar görev tamamlanma zamanlarını; düğümlerin altındaki sayılar ise görevlerin ücret oranlarını göstermektedir. Bu örnek problem için çevrim zamanı 10 olarak belirlenmiştir.



Şekil 4.1. (a) 10 görevli (b) 8 görevli öncelik diyagramları

[Şekil 4.1](#)'deki M-PMHD probleminin en iyi çözümü önerilen istasyon odaklı tamsayılı karar modeli ile bulunmuştur. [Şekil 4.2](#)'deki en iyi görev atamalarına bakıldığında, 2 ve 4 numaralı operatörler sadece hat 1'de montaj işlemi yapılan ürün 1'e ait olan görevleri yaparken; 3 numaralı operatör ise sadece hat 2'de montaj işlemi yapılan ürün 2'ye ait olan görevleri yapmaktadır. Bu operatörler ilgili hattın ayrık işyerlerinde sabit konumda çalışırlar ve bitişik hatta geçmezler.



Şekil 4.2. M-PMHD probleminin istasyon odaklı tamsayılı karar modeli ile bulunan en iyi görev atamaları

Bununla birlikte, 1, 5, 6, 7 ve 8 numaralı operatörler sırasıyla, iki hattın 1, 4, 5, 6 ve 7 numaralı istasyonları arasında kurulan ortak işyerlerinde çalışmaktadırlar. Bu ortak işyerlerinde çalışan operatörler, iki hatta montajı yapılan farklı iki ürüne ait olan görevleri aynı çevrim zamanı içerisinde yerine getirirler. Örneğin, her iki hattın 6 numaralı istasyonları arasında kurulan ortak işyerinde çalışan 7 numaralı operatör, önce ürün 1'e ait 5 zb tamamlanma zamanı olan 9 numaralı görevi yerine getirdikten sonra, hat 2'ye geçmektedir. Aynı çevrim zamanı içerisinde hat 2'ye geçen 7 numaralı operatör, ürün 2'ye ait 5 zb tamamlanma zamanı olan 7 numaralı görevi de yaptıktan sonra, aynı görevleri aynı sırayla yapmak üzere tekrar hat 1'e geri döner. Hatlar arasında yürüme zamanı sıfır olarak varsayıldığı için, 7 numaralı operatörün işyükü 10 zb'dir. Diğer bir ifadeyle boş zamanı sıfırdır. Hat 2'deki 3 numaralı istasyonda herhangi bir işyeri kurulmamıştır ve bu istasyonda hiçbir görev yapılmamaktadır. Bu boş istasyon, yarı mamulü bitişikteki istasyona transfer eden malzeme taşıma sistemini simgelemektedir. Şekil 4.2'deki en iyi görev atamalarına göre, operatörlerin ücret oranları Çizelge 4.4'de verilmiştir.

Çizelge 4.4. Operatörlerin ücret oranları

Operatör (k)	Hat 1	Hat 2	Boş zaman	Ücret oranı
1	{3}	{3}	0	9
2	{1,4}	–	0	4
3	–	{1,4,5}	0	6
4	{5,6}	–	0	4
5	{2}	{2}	0	8
6	{7,8}	{6}	0	6
7	{9}	{7}	0	9
8	{10}	{8}	0	8
Toplam ücret oranı				54
Toplam işgücü maliyeti (54 * 10 =)				540
Toplam boş zaman				0
Toplam ortak işyeri sayısı				5
Toplam ayrı işyeri sayısı				3
Toplam operatör (işyeri) sayısı				8

Bir operatörün ücret oranı, çalıştıkları ayrık veya ortak işyerinde yerine getirdiği görevlerin en yüksek ücret oranı değeri düzeyinde belirlenir. Örneğin 4 numaralı

operatör, ürün 1'e ait olan 5 ve 6 numaralı görevleri yapmaktadır. 5 ve 6 numaralı görevlerin ücret oranları sırasıyla 2 ve 4 pb/zb'dir. Dolayısıyla 4 numaralı operatörün ücret oranı birim zaman için 4 pb'dir.

Aynı çevrim zamanı içerisinde, her iki hattın komşu istasyonlarında çalışan operatörün ücret oranı, bu komşu istasyonlarda yerine getirdiği görevlerin ücret oranlarına göre aynı şekilde hesaplanmaktadır. Örneğin 5 numaralı operatör, ürün 1'e ait 8 pb/zb ücret oranı olan 2 numaralı görevi ve ürün 2'ye ait 7 pb/zb ücret oranı olan 2 numaralı görevi yerine getirmektedir. Bundan dolayı, 5 numaralı operatörün ücret oranı birim zaman için 8 pb'dir. Geriye kalan operatörlerin ücret oranları yukarıda anlatılan mantık çerçevesinde hesaplanarak [çizelge 4.4](#)'te verilmiştir. [Çizelge 4.4](#)'te verilen 8 operatörün ücret oranlarına göre, bir çevrim zamanı için toplam işgücü maliyeti 540 pb'dir.

Görev ücret oranları birbirinin aynı alındığında veya ihmal edildiğinde, M-PMHD problemi PMHD problemine indirgenir. PMHD probleminin amacı işyeri (operatör) sayısını enküçükmektir. Paralel montaj hatların eş zamanlı dengelenmesinde zaman tabanlı yaklaşım ile maliyet tabanlı yaklaşımın çözümleri nasıl değiştirdiğini incelemek amacıyla, örnek 2'deki problem görev ücret oranları ihmal edilerek yeniden çözdürülmüştür. PMHD probleminin en iyi çözümünü bulmak için [Gökçen ve ark. \(2006\)](#) tarafından önerilen tamsayılı programlama modeli kullanılmıştır. PMHD probleminin bulunan en iyi görev atamaları ve bu görev atamalarına göre operatörlerin ücret oranları ve toplam işgücü maliyeti [çizelge 4.5](#)'de verilmiştir.

PMHD probleminin en iyi çözümü incelendiğinde; operatör sayısının 8, toplam işgücü maliyetinin 560 pb olduğu görülmektedir. M-PMHD probleminin en iyi çözümündeki operatör sayısının da 8 olduğu görülmektedir. Dolayısıyla M-PMHD probleminin en iyi çözümü, aynı zamanda PMHD probleminin de en iyi çözümüdür. Ayrıca, M-PMHD ile toplam işgücü maliyeti 560 pb'inden 540 pb'ine düşürülerek 20 pb tasarruf sağlanmıştır.

Çizelge 4.5. PMHD probleminin en iyi görev atamaları

Operatör (k)	Hat 1	Hat 2	Boş zaman	Ücret oranı
1	{3}	{3}	0	9
2	{4}	{1,4}	0	6
3	{1}	{5}	0	6
4	{2}	{2}	0	8
5	{5,6}	-	0	4
6	{7,8}	{6}	0	6
7	{9}	{7}	0	9
8	{10}	{8}	0	8
Toplam ücret oranı				56
Toplam işgücü maliyeti ($56 * 10 =$)				560
Toplam boş zaman				0
Toplam ortak işyeri sayısı				7
Toplam ayrı işyeri sayısı				1
Toplam operatör (işyeri) sayısı				8

Gökçen ve ark. (2006) paralel hatların eş zamanlı dengelenmesi sonucunda işyeri (operatör) sayısında azaltma sağlanabileceğini göstermişlerdir. Maliyet tabanlı yaklaşımla paralel hatlarının eş zamanlı dengelenmesinin maliyet üzerindeki etkisini incelemek amacıyla, hatların her birini birbirinden bağımsız M-MHD problemi olarak ele alınmıştır. Her hattı birbirinden bağımsız olarak dengelemek amacıyla, Amen (2006)'nin tek hatlı tek modelli M-MHD problemi için önerdiği tamsayılı programlama modeli kullanılmıştır.

Çizelge 4.6. M-MHD problemlerinin en iyi görev atamaları

Operatör (k)	Hat 1	Ücret oranı	Hat 2	Ücret oranı
1	{3}	9	{1,3}	6
2	{1,4}	4	{2,4,5}	7
3	{5,6}	4	{6,7}	9
4	{2,7}	8	{8}	7
5	{8,9,10}	9		
Hat 1 için toplam ücret oranı		34		-
Hat 2 için toplam ücret oranı		-		29
Toplam işgücü maliyeti		($63*10 =$)		630
Hat 1 için operatör sayısı		5		-
Hat 2 için operatör sayısı		-		4
Toplam operatör sayısı		($5+4 =$)		9

Çizelge 4.4 ve çizelge 4.6 birlikte incelendiğinde görülüyor ki, paralel hatların ortak bir şekilde dengelenmesi toplam işgücü maliyetinde 90 pb tasarruf sağlamaktadır. Ayrıca kullanılan operatör sayısı da 9'dan 8'e azalmaktadır. Maliyet tabanlı yaklaşımla paralel hatların eş zamanlı dengelenmesinin hem operatör sayısı hem de toplam işgücü maliyeti açısından faydası vardır.

4.2. M-PMHD Problemi için Operatör Odaklı Tamsayılı Programlama Modeli

Bu bölümde operatör odaklı yaklaşımla M-PMHD problemi, her düz hatta montajı yapılacak olan ürünün belirli olduğu durumda, çevrim zamanı ve öncelik ilişkileri kısıtlarını ihlal etmeden, toplam işgücü maliyetini enküçükleyecek şekilde, belirli ürünlere ait görevlerin operatörlere atanması olarak modellenmiştir. Önerilen operatör odaklı tamsayılı programlama modelinde kullanılan karar değişkenleri çizelge 4.7'de açıklanmıştır. Bölüm 4.1'de anlatılan istasyon odaklı tamsayılı karar modelinde kullanılan ortak notasyon ve karar değişkenleri burada açıklanmamıştır.

Çizelge 4.7. M-PMHD probleminin operatör odaklı tamsayılı karar modelinde kullanılan değişkenler

Değişken	Açıklama
x_{hik}	1, eğer h hattında üretilecek olan ürüne ait i görevi k operatörü tarafından yapılacak ise; 0, aksi halde
u_{hk}	1, eğer h hattındaki k istasyonu kullanılmış ise; 0, aksi halde
z_k	1, eğer k operatörü çalışıyor ise; 0, aksi halde
w_k	k operatörünün ücret oranı

M-PMHD problemi için önerilen operatör odaklı tamsayılı doğrusal programlama modeli aşağıdaki gibidir.

$$\text{Enküçükleme } TM = c \cdot \sum_{k=1}^{\bar{K}} w_k \quad (4.12)$$

$$\sum_{k=1}^{\bar{K}} x_{hik} = 1 \quad , \quad i = 1, \dots, n_h \quad , \quad h = 1, \dots, \bar{H} \quad (4.13)$$

$$\sum_{h=1}^{\bar{H}} \sum_{i=1}^{n_h} t_{hi} x_{hik} \leq c.z_k, \quad k=1, \dots, \bar{K} \quad (4.14)$$

$$\sum_{i=1}^{n_h} x_{hik} \leq n_h.u_{hk}, \quad k=1, \dots, \bar{K}, \quad h=1, \dots, \bar{H} \quad (4.15)$$

$$\sum_{i=1}^{n_h} x_{hik} \geq u_{hk}, \quad k=1, \dots, \bar{K}, \quad h=1, \dots, \bar{H} \quad (4.16)$$

$$u_{hk} + u_{(h+a)k} \leq 1, \quad k=1, \dots, \bar{K}, \quad \bar{H} \geq 3, \quad h=1, \dots, \bar{H}-2, \quad a=2, \dots, \bar{H}-h \quad (4.17)$$

$$\sum_{k=1}^{\bar{K}} (\bar{K} - k + 1)(x_{hrk} - x_{hsk}) \geq 0, \quad \forall (r,s) \in P_{hs}, \quad h=1, \dots, \bar{H} \quad (4.18)$$

$$w_k \geq w_{hi} x_{hik}, \quad h=1, \dots, \bar{H}, \quad i=1, \dots, n_h, \quad k=1, \dots, \bar{K} \quad (4.19)$$

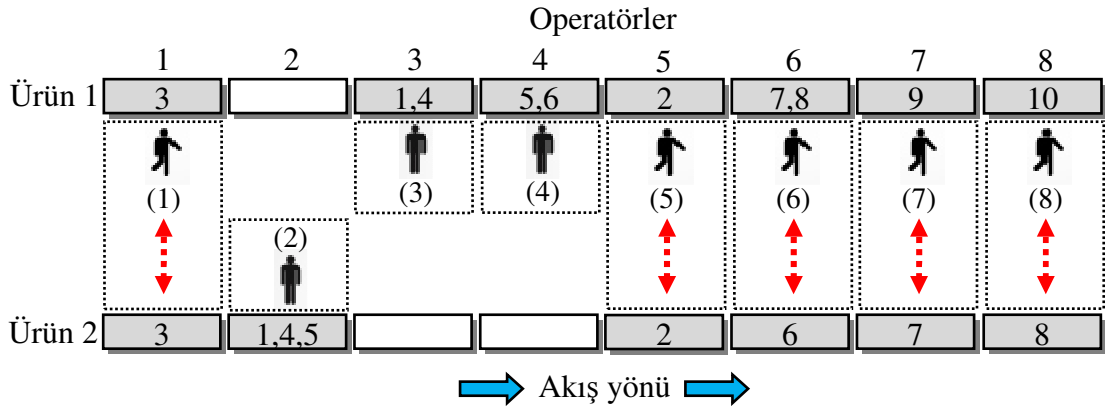
$$w_k \geq 0, \quad k=1, \dots, \bar{K} \quad (4.20)$$

$$x_{hik}, z_k, u_{hk} \in \{0, 1\}, \quad i=1, \dots, n_h, \quad k=1, \dots, \bar{K}, \quad h=1, \dots, \bar{H} \quad (4.21)$$

M-PMHD probleminin amacı *toplam işgücü maliyetini* enküçükmektir. Eş. 4.12’de görüldüğü gibi, toplam işgücü maliyeti operatörlerin ücret oranlarının çevrim zamanıyla çarpımlarının toplamı şeklinde yazılmaktadır. Eş. 4.13’de gösterilen kısıt, herhangi bir hatta montaj işlemi yapılan ürüne ait görevlerin her birini ilgili hattın kesinlikle bir istasyonuna atanmasını sağlamaktadır. Eş. 4.14’deki kısıt, herhangi bir operatörün iş yükünün daha önceden talebe bağlı olarak belirlenen çevrim zamanını aşmamasını sağlamaktadır. Eş. 4.15 ve 4.16’daki grup kısıtlar, hatlardaki hangi istasyonların kullanılması gerektiğine karar vermektedir.

Eş. 4.17’deki kısıt, h ve $h-1$ hatlarının k istasyonlarında bir operatör çalışırken, aynı zamanda h ve $h+1$ hatlarının k istasyonlarında da başka bir operatörün çalışmasını engeller. Aksi durumda, h hattının k istasyonunda iki operatör çalışıyor olur ki, bu durum her istasyonda bir operatör çalışabilir varsayımıyla çelişir. Her hatta üretilen ürünle ilgili görevler arasındaki öncelik ilişkileri, Eş. 4.18’deki kısıt ile sağlamaktadır. Eş. 4.19’deki kısıt, operatörlerin ücret oranlarını belirlemektedir. Eş. 4.20, karar değişkenlerin negatif değer almamasını sağlarken, Eş. 4.21, karar değişkenlerinin 0 veya 1 değerini almalarını sağlamaktadır.

Bölüm 4.1’de istasyon odaklı tamsayıli programlama modelinin uygulandıđı örnek 2’nin verilerinde hiç bir deđişiklik yapılmadan, operatör odaklı tamsayıli programlama modeli ile de çözülmüştür. Operatör odaklı model ile elde edilen en iyi görev atamaları [şekil 4.3](#)’de gösterilmiştir.

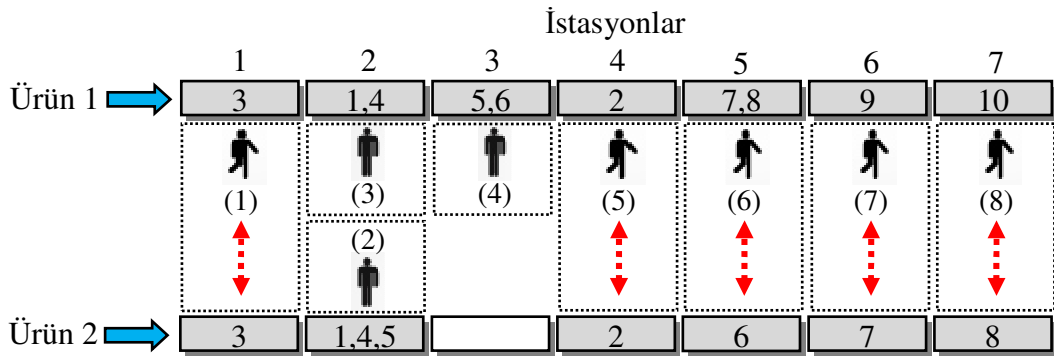


Şekil 4.3. M-PMHD probleminin operatör odaklı tamsayıli karar modeli ile bulunan en iyi görev atamaları

M-PMHD probleminin [şekil 4.3](#)’deki en iyi çözümü incelendiğinde, görevlerin operatörlere atandığı görülmektedir. Örneğin 3 numaralı operatöre bakıldığında sadece hat 1’deki görevleri yerine getirdiği görülmektedir. 3 numaralı operatörün hat 2’de yapması gereken hiçbir görev olmadığı için ilgili alan boş bırakılmıştır. Bu boş alan, istasyon odaklı yaklaşımda olduğu gibi malzeme taşıma sistemini göstermez. Bundan başka, 2 numaralı operatör hat 2’de montaj işlemi yapılan ürün 2 ile ilgili görevleri yaparken; 4 numaralı operatör ise hat 1’de montajı yapılan ürün 1 ile ilgili görevleri yapmaktadır. Ayrıca, 1, 5, 6, 7 ve 8 numaralı operatörler aynı çevrim zamanında her iki hattaki görevleri de yerine getirmektedirler.

Operatörlerin ücret oranı hesabında hiçbir deđişiklik yoktur. Fakat, [şekil 4.3](#)’deki en iyi görev atamalarından, bu operatörlerin hatların hangi bölümünde (hangi istasyonda) çalışacağı bilgisi çıkartılamaz. Bu bilginin elde edilebilmesi için en iyi çözüm üzerinde ek bir dönüştürme işleminin yapılması gereklidir. Bu dönüştürme ise, operatörlerin yapacakları görevleri deđiştirmeden mümkün olabildiğince sola doğru kaydırılması işlemidir. Örneğin 3 numaralı operatörün hat 1’de yaptığı 1 ve 4

numaralı görevleri bir birim sol tarafa doğru kaydırabiliriz. Fakat 2 numaralı operatörün yaptığı hat 2'ye ait olan 1, 4 ve 5 numaralı görevleri sol tarafa doğru kaydırılamaz. Bu mantıkla tüm görevler boşluklar mümkün olduğunca dolana kadar sol tarafa doğru kaydırılır. Görevlerin sola kaydırma işlemi tamamlandığında [şekil 4.4](#)'de gösterilen görev atamaları elde edilir. Dönüştürme işlemi sonucunda toplam işgücü maliyeti ve operatör sayısı değişmez.



Şekil 4.4. Dönüştürme işlemi sonucunda elde edilen en iyi görev atamaları

Dönüştürme işlemi sonucunda elde edilen [şekil 4.4](#) üzerinde operatörlerin hatların hangi bölgelerinde (istasyonlarında) çalıştıkları bilgisi elde edilebilmektedir. [Şekil 4.4](#)'deki görevlerin atamasına göre, 2 numaralı operatör ile 3 numaralı operatör birbirinden bağımsız olarak farklı hatların komşu olan 2 numaralı istasyonlarında aynı anda çalışmakta iken, 1, 5, 6, 7 ve 8 numaralı operatörler her iki hatta üretilen ürünlerin görevlerini bir çevrim zamanı içinde yapmaktadırlar. Operatör odaklı tamsayılı programlama modeli ile bulunan en iyi çözüme (Bkz. [Şekil 4.4](#)) dönüştürme işlemi uygulandığında elde edilen çözüm ile istasyon odaklı tamsayılı programlama modeli ile bulunan en iyi çözümün (Bkz. [Şekil 4.2](#)) aynı olduğu görülmektedir.

4.3. M-YZPMHD Problemi için Tamsayılı Programlama Modelleri

Paralel montaj hatlarının birbirinden bağımsız olarak veya ortak bir şekilde dengelenmesinin arasındaki en önemli farklılık, montaj hatları üzerinde açılan işyerlerinin biçimidir. Paralel hatların bağımsız olarak dengelenmesinde düz montaj

hatlarında sadece ayırık işyerleri bulunurken; eş zamanlı dengelenmesinde ise ayırık işyerlerinin yanı sıra, ortak işyerleri de bulunmaktadır. Paralel montaj hatlarının eş zamanlı dengelenmesiyle operatör sayısında azalmayı ve operatörlerin etkinliğini sağlayan en önemli etken, ortak işyerlerinin kullanılmasıdır. Kurulan bu ortak işyerlerinde çalışan operatörler, bitişik iki hattın komşu istasyonlarında bir çevrim zamanını aşmadan çalışmaktadırlar. Paralel montaj hatlarında kurulan bu ortak işyerlerinin iş yükü, iki temel bileşenden meydana gelmektedir. Bunlar:

1. Bitişik hatların komşu istasyonlarına atanan görevlerin tamamlanma zamanı
2. Operatörün komşu istasyonlara atanmış görevleri yaparken gerçekleştirdiği hatlar arası yürümeler için harcadığı zaman

MHD problemlerinin temel varsayımlarından birisi, bir görevin tamamlanma zamanının, bu görevin atandığı istasyondan bağımsız olmasıdır. Yani; bir görev hangi istasyona atanırsa atansın, tamamlanma zamanı değişmemektedir. Dolayısıyla, bir görev düz montaj hattındaki ayırık bir istasyona (işyerine) da atansa, bitişik montaj hatlarının komşu istasyonlarına (ortak işyerine) da atansa, bu görevin tamamlanma zamanı aynı olacaktır. Aynı görevlerin hem ayırık işyerine hem de ortak işyerine atandığını düşünürsek, bu iki işyerinin yükleri arasındaki farkı belirleyen unsur, işyerlerinde çalışan operatörlerin yürüme zamanlarıdır.

Paralel montaj hatlarının ortak işyerlerinde çalışan operatörler iki farklı mesafeyi yürürler. Bunlar: (i) karşıya geçiş mesafesi (ii) geri dönüş mesafesi. Karşıya geçiş mesafesi, ortak işyerinde çalışan bir operatörün çalıştığı ilk hattaki görevlerini tamamladıktan sonra, çalıştığı diğer hattaki görevlerini tamamlamak üzere karşıya geçerken yürüdüğü mesafedir. Geri dönüş mesafesi, çevrimini tamamlamak üzere olan operatörün yeni bir çevrime başlamak üzere çalıştığı ilk hatta dönerken yürüdüğü mesafedir. Birbirine paralel olarak yerleştirilmiş düz montaj hatlarında, hatlar arası uzaklık her noktada aynı olduğu varsayıldığından, karşıya geçiş mesafesi ve geri dönüş mesafesi birbirine eşit olarak alınır. Ayrıca karşı hatta yürüme zamanları operatörlere göre değişiklik göstermediği varsayılmaktadır. Ortak işyerlerinde çalışan operatörler, ayırık işyerlerinde çalışan operatörlerden farklı olarak

karşıya geçiş ve geri dönüş mesafesini yürümek zorundadırlar. Karşıya geçiş ve geri dönüş mesafeleri aynı olduğu için aynı mesafeyi 2 kere yürümek zorunda kalırlar.

PMHD probleminin çözümü için önerilen yaklaşımların hepsinde karşıya geçiş ve geri dönüş zamanları ihmal edilmiştir. Gökçen ve ark. (2006) tarafından önerilen tamsayı programlama modelinde ortak işyerlerinde çalışan operatörlerin yürüme mesafeleri ihmal edilmiştir. PMHD problemindeki görevlerin tamamlanma zamanlarına, yürüme zamanlarının iki katı kadar zaman ilave edilerek dengeleme yapıldığında, ortak işyerlerine atanan görevlerin çevrim zamanı içerisinde tamamlanamama olasılığı da artabilmektedir. Bu olumsuz durumun önüne geçebilmek için, ortak işyerlerinde çalışan operatörlere karar verici tarafından belirlenen miktarda yürüme zamanı farkı ayrılmalıdır.

Yukarıda verilen açıklamalar ışığında, M-PMHD probleminin ortak işyerleri için yürüme zamanlarını dikkate alacak şekilde ele alınması gerekmektedir. Bu amaçla, M-PMHD probleminin çözümü için ortak işyerlerinde çalışan operatörler için yürüme sürelerini dikkate alan yeni istasyon odaklı ve operatör odaklı tamsayı programlama modelleri geliştirilmiştir. Yeni karar modellerinde kullanılan yeni parametre ve karar değişkeni aşağıda açıklanmıştır:

d : bitişik hatlar arasındaki tek yön yürüme süresidir. Tüm bitişik hatların arasındaki yürüme süresi aynıdır. Yani; birbirine paralel olan üç hatlı bir üretim sisteminde, hat 1 – hat 2 ve hat 2 – hat 3 arasındaki her noktada yürüme süresi aynıdır.

$p_{h(h+1)k}$: 1, eğer h ve $h+1$ bitişik hatlarının komşu istasyonlarında k operatörü çalışıyor ise; 0, aksi halde

M-YZPMHD probleminin çözümü için önerilen istasyon odaklı tamsayı programlama modeli aşağıdaki gibidir.

Amaç fonksiyonu (4.1)

Eş. (4.2), (4.4), (4.5), (4.6), (4.7), (4.8), (4.9), (4.10), (4.11)

$$\sum_{i=1}^{n_h} t_{hi} x_{hik} + \sum_{i=1}^{n_{h+1}} t_{(h+1)i} x_{(h+1)ik} + 2d \cdot z_{h(h+1)k} \leq c \cdot (u_{hk} + u_{(h+1)k} - z_{h(h+1)k}) \quad (4.22)$$

$, k=1, \dots, \bar{K}, \quad h=1, \dots, \bar{H}-1$

Ayrık işyerlerinde çalışan operatörler karşı hatta geçmedikleri için, yürüme süreleri sadece ortak işyerleri için yazılmış çevrim zamanı kısıtını (Eş. 4.3) etkilemektedir. Ortak işyerlerinde çalışan operatörler hatlar arasındaki aynı yolu iki kere katedeceği için, ortak işyerinin iş yüküne yürüme süresinin iki katı kadar süre ilave edilmektedir. Dolayısıyla, M-PMHD probleminin çözümü için önerilen istasyon odaklı tamsayılı karar modelinde, Eş. 4.3 yerine yürüme sürelerinin de eklendiği Eş. 4.22 kullanılmıştır.

İstasyon odaklı tamsayılı karar modelinin çevrim zamanı kısıtı üzerinde yapılan aynı değişiklik operatör odaklı tamsayılı karar modelinde de yapılmıştır. Sonuç olarak, M-YZPMHD probleminin çözümü için önerilen operatör odaklı tamsayılı programlama modeli aşağıdaki gibidir.

Amaç fonksiyonu (4.12)

Eş. (4.13), (4.15), (4.16), (4.17), (4.18), (4.19), (4.20), (4.21)

$$\sum_{h=1}^{\bar{H}} \sum_{i=1}^{n_h} t_{hi} x_{hik} + 2d \cdot \sum_{h=1}^{\bar{H}-1} p_{h(h+1)k} \leq c \cdot z_k \quad , k=1, \dots, \bar{K} \quad (4.23)$$

$$p_{h(h+1)k} \geq u_{hk} + u_{(h+1)k} - 1 \quad , k=1, \dots, \bar{K} \quad , h=1, \dots, \bar{H}-1 \quad (4.24)$$

$$p_{h(h+1)k} \leq u_{hk} \quad , k=1, \dots, \bar{K} \quad , h=1, \dots, \bar{H}-1 \quad (4.25)$$

$$p_{h(h+1)k} \leq u_{(h+1)k} \quad , k=1, \dots, \bar{K} \quad , h=1, \dots, \bar{H}-1 \quad (4.26)$$

$$p_{h(h+1)k} \in \{0, 1\} \quad , k=1, \dots, \bar{K} \quad , h=1, \dots, \bar{H}-1 \quad (4.27)$$

M-YZPMHD probleminin çözümü için önerilen istasyon odaklı tamsayılı karar modeli ile bölüm 4.1'deki örnek 2, yürüme süresi sırasıyla 0,5 ve 1 zb alınarak yeniden çözdürülmüştür. Örnek problemin 0,5 ve 1 zb yürüme süreleri için en iyi

görev atamaları [çizelge 4.8](#)'de verilmiştir. Farklı yürüme süreleri için M–YZPMHD probleminin [çizelge 4.8](#)'deki en iyi çözümleri incelendiğinde, hatlar arası yürüme süreleri arttıkça kurulan ortak işyeri sayısının azaldığı görülmektedir. Dolayısıyla buna bağlı olarak toplam işgücü maliyeti de artmaktadır. Yürüme süreleri belirli değere yükselene kadar ortak işyerlerinin sayısı azalırken, belirli değerden sonra hatlar tamamen bağımsız dengeleme sonuçlarıyla aynı olmaktadır. Yani; paralel hatlar arasındaki yürüme süreleri arttıkça her hattın istasyonlarında sadece ayrıık işyerleri açılmakta ve bitişik hatların komşu istasyonlarının işyükleri birleştirilmemektedir. Bu durumda işgücü maliyetlerinin artmasına neden olmaktadır.

Çizelge 4.8. M–YZPMHD probleminin en iyi görev atamaları

Opr (k)	Yürüme zamanı = 0,5				Yürüme zamanı = 1			
	Hat 1	Hat 2	Boş zaman	Ücret oranı	Hat 1	Hat 2	Boş zaman	Ücret oranı
1	{1}	{1}	0	6	{3}	-	3	9
2	{2}	{3}	0	8	-	{1,3}	3	6
3	{3}	{4}	1	9	{1,4}	-	0	4
4	{4,5}	-	0	2	-	{2,4,5}	0	7
5	-	{2,5}	1	7	{5,6}	-	0	4
6	{6,7}	-	1	4	-	{6,7}	1	9
7	-	{6,7}	1	9	{2,7}	-	0	8
8	{8,9,10}	-	1	9	-	{8}	2	7
9	-	{8}	2	7	{8,9,10}	-	1	9
Toplam ücret oranı				61	63			
Toplam işgücü maliyeti			61*10=610		63*10=630			
Toplam boş zaman				7	10			
Toplam ortak işyeri sayısı				3	0			
Toplam ayrı işyeri sayısı				6	9			
Toplam operatör (işyeri) sayısı				9	9			

4.4. M–ÇÜPMHD Problemi için Tamsayı Programlama Modelleri

Birden fazla düz montaj hattına sahip üretim sistemlerinde, her hatta sadece bir ürünün montajı yapılmaktadır. PMHD probleminde, her hatta montajı yapılacak ürün önceden belirli iken; ÇÜPMHD probleminde ise hangi hatta hangi ürünün montajının yapılacağına da karar verilmektedir. Dolayısıyla ÇÜPMHD problemi, iki alt

problemi içermektedir: (i) ürünlerin hatlara atanması (ii) paralel hatların eş zamanlı olarak dengelenmesi. Gökçen ve ark. (2006) tarafından PMHD probleminin çözümü için önerdikleri tamsayı programlama modelinde, ürünlerin hatlara atanması problemi dikkate alınmamıştır.

M-PMHD probleminde hatlarda üretilen ürünler değiştirildiğinde toplam işgücü maliyeti de değişmektedir. Bundan dolayı, hangi ürünün montajının hangi hatta yapılacağı maliyet açısından önemli bir karardır. Bu başlık altında, paralel hatların maliyet tabanlı yaklaşımla eş zamanlı dengelenmesi problemi (M-PMHD problemi), ürünlerin hatlara atanması problemini de ilave ederek ele alınmıştır. Bu problem, *Maliyet tabanlı Çok Ürünlü Paralel Montaj Hattı Dengeleme* (M-ÇÜPMHD) problemi olarak adlandırılmıştır. Toplam işgücü maliyetini enküçükleyecek M-ÇÜPMHD probleminin en iyi görev atamalarını bulmak amacıyla karma tamsayı doğrusal karar modelleri önerilmiştir. Bu tamsayı karar modellerinin önceki bölümlerde sunulan karar modellerinden farklı olan parametreleri ve karar değişkenleri aşağıda açıklanmıştır.

t'_{pi} : p ürününe ait i görevinin tamamlanma zamanı

w'_{pi} : p ürününe ait i görevinin ücret oranı

x_{hpik} : 1, eğer h hattında montajı yapılacak olan p ürününe ait i görevi k istasyonuna atanmış ise; 0, aksi halde (istasyon odaklı karar modeli için)

x_{hpik} : 1, eğer h hattında montajı yapılacak olan p ürününe ait i görevi k operatörüne atanmış ise; 0, aksi halde (operatör odaklı karar modeli için)

y_{hp} : 1, eğer h hattında p ürününün montajı yapılacak ise; 0, aksi halde

M-ÇÜPMHD probleminin çözümü için önerilen istasyon odaklı tamsayı programlama modeli aşağıdaki gibidir.

$$\text{Enküçükleme } TM = c \cdot \sum_{h=1}^{\bar{H}} \sum_{k=1}^{\bar{K}} w n_{hk} + c \cdot \sum_{h=1}^{\bar{H}-1} \sum_{k=1}^{\bar{K}} w s_{h(h+1)k} \quad (4.28)$$

$$\sum_{h=1}^{\bar{H}} \sum_{k=1}^{\bar{K}} x_{hpik} = 1 \quad , \quad i = 1, \dots, n_p \quad , \quad p = 1, \dots, \bar{P} \quad (4.29)$$

$$\sum_{p=1}^{\bar{P}} \sum_{i=1}^{n_p} t'_{pi} \cdot x_{hpik} + \sum_{p=1}^{\bar{P}} \sum_{i=1}^{n_p} t'_{pi} \cdot x_{(h+1)pik} \leq c \cdot (u_{hk} + u_{(h+1)k} - z_{h(h+1)k}) \quad , \quad \forall k \quad , \quad h = 1, \dots, \bar{H} - 1 \quad (4.30)$$

$$\sum_{p=1}^{\bar{P}} \sum_{i=1}^{n_p} t'_{pi} \cdot x_{hpik} \leq c \cdot u_{hk} \quad , \quad k = 1, \dots, \bar{K} \quad , \quad h = 1, \dots, \bar{H} \quad (4.31)$$

$$\sum_{p=1}^{\bar{P}} \sum_{i=1}^{n_p} x_{hpik} \geq u_{hk} \quad , \quad k = 1, \dots, \bar{K} \quad , \quad h = 1, \dots, \bar{H} \quad (4.32)$$

$$z_{h(h+1)k} + z_{(h+1)(h+2)k} \leq 1 \quad , \quad \bar{H} \geq 3 \quad , \quad h = 1, \dots, \bar{H} - 2 \quad , \quad k = 1, \dots, \bar{K} \quad (4.33)$$

$$\sum_{h=1}^{\bar{H}} y_{hp} = 1 \quad , \quad p = 1, \dots, \bar{P} \quad (4.34)$$

$$\sum_{p=1}^{\bar{P}} y_{hp} = 1 \quad , \quad h = 1, \dots, \bar{H} \quad (4.35)$$

$$\sum_{k=1}^{\bar{K}} x_{hpik} \geq y_{hp} \quad , \quad i = 1, \dots, n_h \quad , \quad p = 1, \dots, \bar{P} \quad , \quad h = 1, \dots, \bar{H} \quad (4.36)$$

$$\sum_{h=1}^{\bar{H}} \sum_{k=1}^{\bar{K}} k \cdot x_{hprk} \leq \sum_{h=1}^{\bar{H}} \sum_{k=1}^{\bar{K}} k \cdot x_{hpsk} \quad , \quad \forall (r,s) \in P_{hs} \quad , \quad p = 1, \dots, \bar{P} \quad (4.37)$$

$$wn_{hk} \geq \begin{cases} w'_{pi}(x_{hpik} - z_{h(h+1)k}) & , h = 1, i = 1, \dots, n_p \quad , \quad \forall k, p \\ w'_{pi}(x_{hpik} - z_{(h-1)hk} - z_{h(h+1)k}) & , 1 < h < \bar{H} \quad , \quad i = 1, \dots, n_p \quad , \quad \forall k, p \\ w'_{pi}(x_{hpik} - z_{(h-1)hk}) & , h = \bar{H} \quad , \quad i = 1, \dots, n_p \quad , \quad \forall k, p \end{cases} \quad (4.38)$$

$$ws_{h(h+1)k} \geq w'_{pi}(x_{hpik} - z_{h(h+1)k} - u_{hk}) \quad , \quad i = 1, \dots, n_p \quad , \quad h = 1, \dots, \bar{H} - 1 \quad , \quad \forall k, p \quad (4.39)$$

$$wn_{hk} \quad , \quad ws_{h(h+1)k} \geq 0 \quad , \quad k = 1, \dots, \bar{K} \quad , \quad h = 1, \dots, \bar{H} \quad (4.40)$$

$$x_{hpik} \quad , \quad z_{h(h+1)k} \quad , \quad u_{hk} \in \{0, 1\} \quad , \quad i = 1, \dots, n_h \quad , \quad h = 1, \dots, \bar{H} \quad , \quad \forall k, p \quad (4.41)$$

Amaç fonksiyonu (Eş. 4.28), paralel montaj hatlarında kurulan ayrık ve ortak işyerlerinde çalışan operatörlere bir çevrim zamanı için ödenen ücretlerin toplamını enküçükmektir. Eş. 4.29'daki kısıt, ürünlere ait görevlerin her birinin sadece bir hattaki sadece bir istasyona atanmasını sağlamaktadır. Herhangi bir ürüne ait olan bir görev aynı anda farklı hatlardaki istasyonlara atanamaz.

Eş. 4.30'daki kısıt, paralel hatların komşu istasyonlarına atanan iki ürüne ait görevlerden oluşan toplam iş yükünün çevrim zamanını aşmamasını sağlar. Eş. 4.31'deki kısıt, her hattaki ayrık istasyonlara atanan sadece bir ürüne ait görevlerden oluşan toplam iş yükünün çevrim zamanını aşmamasını sağlar. Eş. 4.32'deki kısıt, hatlardaki hangi istasyonların kullanılması gerektiğine karar vermektedir. Eş. 4.33'deki kısıt, h hattının k istasyonunda çalışan bir operatörün çevrim zamanı aşılması şartıyla, h hattının bitişik komşuluğunda bulunan $h-1$ veya $h+1$ hatlarından sadece birinin k istasyonunda çalışabilmesine izin verir.

Eş. 4.34'deki kısıt, her ürünün mutlaka ve sadece hatlardan bir tanesine atanmasını sağlar. Eş. 4.35'deki kısıt ise her hatta mutlaka ve sadece bir ürünün montajının yapılmasını sağlamaktadır. Eş. 4.36'daki kısıt, bir ürünle ilgili tüm görevlerin Eş. 4.34'deki kısıt ile belirlenen hatta atanmasını sağlamaktadır.

Eş. 4.37'deki kısıt, her hatta üretilecek olan ürüne ait olan görevler arasındaki öncelik ilişkilerinin ihlal edilmemesini sağlamaktadır. Eş. 4.38'deki kısıt, tek ürünle ilgili görevlerin yapıldığı ayrık istasyonlarda çalışan operatörlerin ücret oranlarını belirlerken; Eş. 4.39'daki kısıt ise, Eş. 4.35'deki kısıt ile belirlenen iki ürünle ilgili görevlerin yapıldığı ve bitişik hatların komşu her iki istasyonda çalışan operatörlerin ücret oranlarını belirlemektedir. Eş. 4.40'daki kısıt, operatörlerin ücret oranlarının negatif değer almamasını sağlar. Eş. 4.41'de gösterilen kısıt, tamsayı karar modelindeki karar değişkenlerinin 0 veya 1 değerini almalarını sağlamaktadır.

M-ÇÜPMHD probleminin çözümü için önerilen operatör odaklı tamsayı programlama modeli aşağıdaki gibidir.

$$\text{Enküçükleme } TM = c \cdot \sum_{k=1}^{\bar{K}} w_k \quad (4.42)$$

$$\sum_{h=1}^{\bar{H}} \sum_{k=1}^{\bar{K}} x_{hpik} = 1 \quad , \quad i = 1, \dots, n_p \quad , \quad p = 1, \dots, \bar{P} \quad (4.43)$$

$$\sum_{h=1}^{\bar{H}} \sum_{p=1}^{\bar{P}} \sum_{i=1}^{n_p} t'_{pi} x_{hpik} \leq C.z_k \quad , \quad k=1, \dots, \bar{K} \quad (4.44)$$

$$\sum_{p=1}^{\bar{P}} \sum_{i=1}^{n_p} x_{hpik} \leq n_{\max} u_{hk} \quad , \quad k=1, \dots, \bar{K} \quad , \quad h=1, \dots, \bar{H} \quad , \quad n_{\max} = \max\{n_1, n_2, \dots, n_{\bar{P}}\} \quad (4.45)$$

$$\sum_{p=1}^{\bar{P}} \sum_{i=1}^{n_p} x_{hpik} \geq u_{hk} \quad , \quad k=1, \dots, \bar{K} \quad , \quad h=1, \dots, \bar{H} \quad (4.46)$$

$$u_{hk} + u_{(h+a)k} \leq 1 \quad , \quad k=1, \dots, \bar{K} \quad , \quad \bar{H} \geq 3 \quad , \quad h=1, \dots, \bar{H}-2 \quad , \quad a=2, \dots, \bar{H}-h \quad (4.47)$$

$$\sum_{h=1}^{\bar{H}} y_{hp} = 1 \quad , \quad p=1, \dots, \bar{P} \quad (4.48)$$

$$\sum_{p=1}^{\bar{P}} y_{hp} = 1 \quad , \quad h=1, \dots, \bar{H} \quad (4.49)$$

$$\sum_{k=1}^{\bar{K}} x_{hpik} \geq y_{hp} \quad , \quad i=1, \dots, n_h \quad , \quad p=1, \dots, \bar{P} \quad , \quad h=1, \dots, \bar{H} \quad (4.50)$$

$$\sum_{h=1}^{\bar{H}} \sum_{k=1}^{\bar{K}} k.x_{hprk} \leq \sum_{h=1}^{\bar{H}} \sum_{k=1}^{\bar{K}} k.x_{hpsk} \quad , \quad \forall (r,s) \in P_{hs} \quad , \quad p=1, \dots, \bar{P} \quad (4.51)$$

$$w_k \geq w'_{pi} x_{hpik} \quad , \quad h=1, \dots, \bar{H} \quad , \quad i=1, \dots, n_h \quad , \quad k=1, \dots, \bar{K} \quad , \quad p=1, \dots, \bar{P} \quad (4.52)$$

$$w_k \geq 0 \quad , \quad k=1, \dots, \bar{K} \quad (4.53)$$

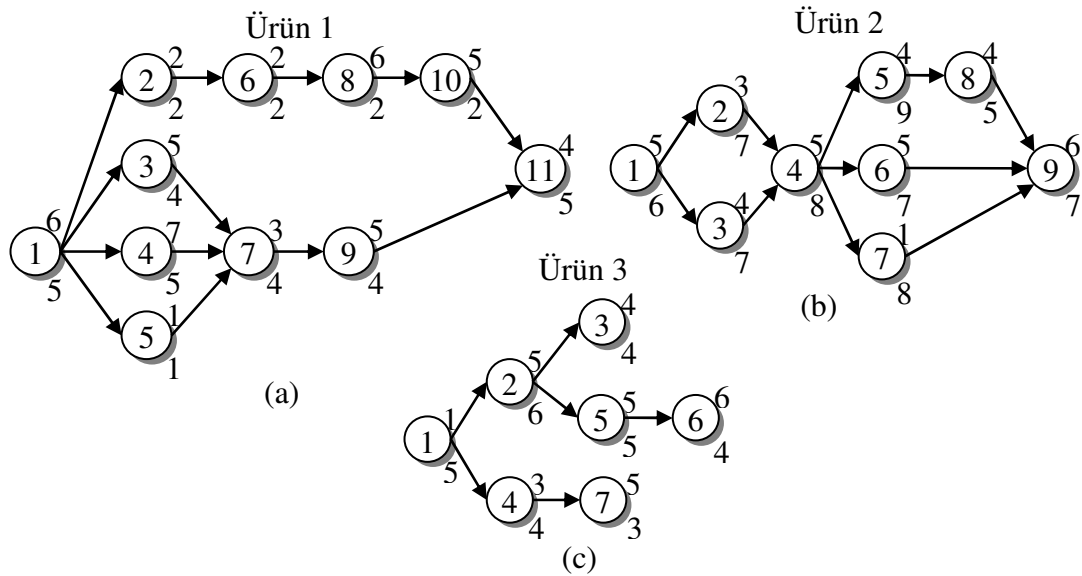
$$x_{hpik}, z_k, u_{hk} \in \{0, 1\} \quad , \quad i=1, \dots, n_h \quad , \quad k=1, \dots, \bar{K} \quad , \quad h=1, \dots, \bar{H} \quad (4.54)$$

Amaç fonksiyonu toplam işgücü maliyetini enküçükmektir. Eş. 4.43'deki kısıt, her ürüne ait görevlerin kesinlikle sadece bir hattaki bir istasyona atanmasını sağlamaktadır. Eş. 4.44'deki kısıt, bir istasyonda çalışan operatörün toplam iş yükünün çevrim zamanını aşmamasını sağlamaktadır. Eş. 4.45 ve 4.46'daki kısıtlar, paralel hatlardaki hangi istasyonların açılması gerektiğine karar vermektedir. Eş. 4.47'deki kısıt, h hattının k istasyonunda çalışan bir operatörün, bitişik komşu hatlardan sadece birinin k istasyonuna atanan görevleri yapabilmesini sağlar. Eş. 4.48'deki kısıt, her ürünün kesinlikle sadece hatlardan bir tanesine atanmasını sağlamaktadır. Eş. 4.49'deki kısıt, her hatta kesinlikle sadece bir ürünün montajının yapılmasını sağlamaktadır. Eş. 4.50'deki kısıt, her hatta montajı yapılacak ürünle

ilgili görevlerin atanmasını sağlamaktadır. Eş. 4.51'deki kısıt, ürünlere ait olan görevler arasındaki öncelik ilişkilerinin ihlal edilmemesini sağlamaktadır. Eş. 4.52'deki kısıt, operatörlerin yaptıkları görevlere göre ücret oranlarını belirlemektedir. Eş. 4.53'deki kısıt, karar değişkenlerin negatif değer almamasını sağlar. Eş. 4.54'de gösterilen kısıt, tamsayıli modeldeki karar değişkenlerinin 0 veya 1 değerini almalarını sağlamaktadır.

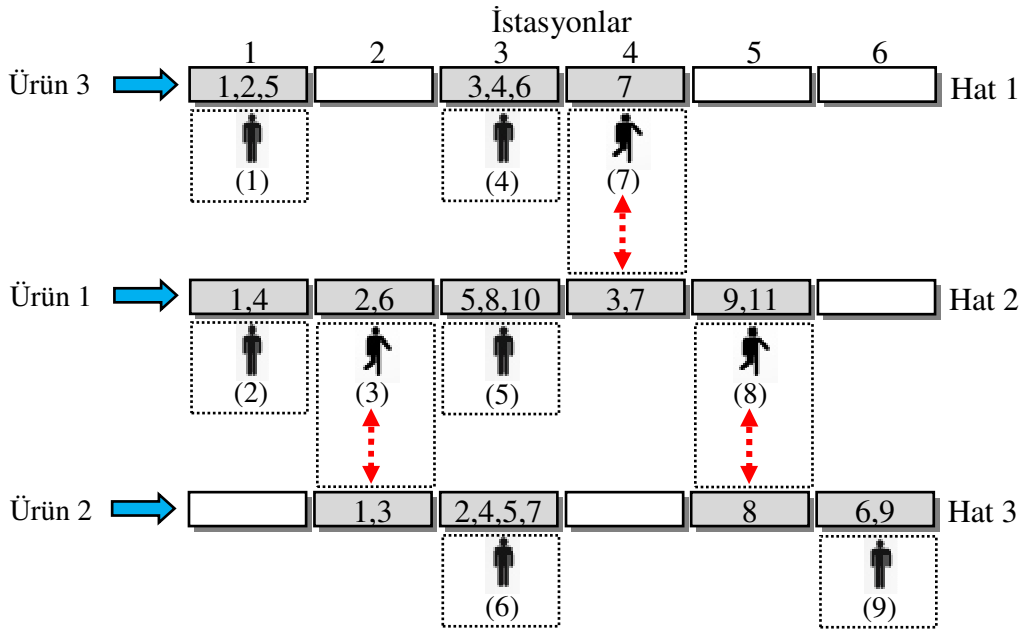
Birbirine paralel konumda olan iki paralel hat için M-ÇÜPMHD probleminin sonuçları ile M-PMHD probleminin sonuçları arasında fark yoktur. Çünkü her iki hatta atanan ürünler yer değiştirirse bile problemin yapısı değişmez. Bundan dolayı yeni tanıtilan problemin farkını tam anlamıyla ortaya koyabilmek amacıyla 3 hatlı bir problem kullanılmıştır.

Örnek 3: 11, 9 ve 7 adet görevden oluşan birbirinden farklı 3 ürünün 3 hatlı bir üretim sisteminde montajı yapılacaktır. Ürünlerin öncelik diyagramları, görev tamamlanma zamanları ve görev ücret oranları [şekil 4.5](#)'de verilmiştir. Düğümlerin üstündeki sayılar görev tamamlanma zamanlarını; düğümlerin altındaki sayılar ise görev ücret oranlarını göstermektedir. Çevrim zamanı 13 olarak belirlenmiştir.



Şekil 4.5. (a) 11 görevli (b) 9 görevli (c) 7 görevli öncelik diyagramları

Bu örnek problemde ele alınan M-ÇÜPMHD probleminin en iyi çözümü, önerilen istasyon odaklı tamsayılı programlama modeli kullanılarak IBM ILOG CPLEX 12.4 yazılımı yardımıyla bulunmuştur. Elde edilen en iyi görev atamaları ve hangi ürünlerin hangi hatlarda üretileceği [şekil 4.6](#)'da verilmiştir.



Şekil 4.6. Üç hatlı M-ÇÜPMHD probleminin en iyi görev ataması

Hangi ürünün hangi istasyonda üretilmesi ile ilgili $3!$ kombinasyon vardır. Tüm bu kombinasyonlar [çizelge 4.9](#)'de verilmiştir. Fakat 1-2-3 atamasıyla 3-2-1 atamasının aynı sonuçlar vereceği düşünüldüğünde, kombinasyonların yarısı aynı çözümleri göstermektedir. Dolayısıyla gerçekte bakılması gereken $3!/2$ sayıda kombinasyon vardır.

Çizelge 4.9. Hatlardaki montajı yapılacak ürünlere göre duyarlılık analizi

Senaryo	Hat 1	Hat 2	Hat 3	Toplam işgücü maliyeti
1	Ürün 1	Ürün 2	Ürün 3	650
2	Ürün 1	Ürün 3	Ürün 2	650
3	Ürün 2	Ürün 1	Ürün 3	637*
4	Ürün 2	Ürün 3	Ürün 1	650
5	Ürün 3	Ürün 1	Ürün 2	637*
6	Ürün 3	Ürün 2	Ürün 1	650

M-ÇÜPMHD probleminin amacı, tüm bu kombinasyonların içinden toplam işgücü maliyetini enküçük yapan ürünlerin hatlara atanmasını bulmaktır. Şekil 4.6’da görüldüğü gibi ürün 1, ürün 2 ve ürün 3’nin montajları sırasıyla hat 2, hat 3 ve hat 1’de yapılır ise, toplam işçilik maliyeti 637 para birimi ile en küçük değerini alır. Diğer kombinasyonların toplam işgücü maliyet değerleri çizelge 4.9’de verilmiştir.

4.5. Ücret Oranlarının Düzgünleştirilmesi için Tamsayılı Programlama Modeli

M-PMHD probleminde toplam işgücü maliyetini enküçükleyecek görev atamaları bulunurken, aynı zamanda operatörlerin ücret oranları da belirlenmektedir. Paralel hatlarda montajı yapılacak ürünlere ait görevlerin ücret oranları birbirinden farklı olduğu için, aynı veya bitişik hatta çalışan operatörlerin ücret oranları da birbirine göre farklılık göstermektedir. Gelişmiş sanayi ülkelerinin çoğunda, sendikalar ve çalışanlar arasındaki ortaklaşa anlaşmalar bir ücret farklılaşmasını içermesine rağmen, bazı firmaların ücret politikaları gereği montaj hattında çalışan operatörlerin mümkün olduğunca birbiriyle aynı ücreti alması istenebilir. Bir esasa dayanmayan ve tamamen şahsî görüşlere göre yapılan ücretlendirme haliyle bazı dengesizliklere sebep olmakta, bu da personel arasında huzursuzluk yaratmaktadır.

Bu gibi durumlara çözüm önerisi getirmek için, M-PMHD probleminin amacı toplam işgücü maliyetini enküçüklemek yerine, operatörlerin ücret oranlarındaki farkın enküçüklenmesi ele alınmıştır. M-PMHD probleminde ücret oranları farkını enküçükleyecek görev atamalarını bulan tamsayılı doğrusal programlama modeli geliştirilmiştir. Önerilen tamsayılı karar modelinde kullanılan yeni parametre ve değişkenler aşağıda açıklanmıştır:

w_{enb} : en büyük görev ücret oranı

w_k^+ : ortalama ücret oranından pozitif sapma miktarı

w_k^- : ortalama ücret oranından negatif sapma miktarı

Operatör ücret oranlarının düzgünleştirilmesi için önerilen tamsayı doğrusal karar modeli, M-PMHD problemi için [bölüm 4.2](#)'de anlatılan operatör odaklı tamsayı programlama modelinin güncellenmiş halidir. Ücret oranlarının düzgünleştirilmesi için önerilen tamsayı doğrusal karar modelini aşağıda verilmiştir.

$$Enküçükleme \quad ORTW = \sum_{k=1}^{\bar{K}} (w_k^+ + w_k^-) \quad (4.55)$$

$$\sum_{k=1}^{\bar{K}} x_{hik} = 1 \quad , \quad i=1, \dots, n_h \quad , \quad h=1, \dots, \bar{H} \quad (4.56)$$

$$\sum_{h=1}^{\bar{H}} \sum_{i=1}^{n_h} t_{hi} x_{hik} \leq c.z_k \quad , \quad k=1, \dots, \bar{K} \quad (4.57)$$

$$\sum_{i=1}^{n_h} x_{hik} \leq n_h . u_{hk} \quad , \quad k=1, \dots, \bar{K} \quad , \quad h=1, \dots, \bar{H} \quad (4.58)$$

$$\sum_{i=1}^{n_h} x_{hik} \geq u_{hk} \quad , \quad k=1, \dots, \bar{K} \quad , \quad h=1, \dots, \bar{H} \quad (4.59)$$

$$u_{hk} + u_{(h+a)k} \leq 1 \quad , \quad k=1, \dots, \bar{K} \quad , \quad \bar{H} \geq 3 \quad , \quad h=1, \dots, \bar{H}-2 \quad , \quad a=2, \dots, \bar{H}-h \quad (4.60)$$

$$\sum_{k=1}^{\bar{K}} (\bar{K} - k + 1)(x_{hrk} - x_{hsk}) \geq 0 \quad , \quad \forall (r,s) \in P_{hs} \quad , \quad h=1, \dots, \bar{H} \quad (4.61)$$

$$w_k \geq w_{hi} x_{hik} \quad , \quad h=1, \dots, \bar{H} \quad , \quad i=1, \dots, n_h \quad , \quad k=1, \dots, \bar{K} \quad (4.62)$$

$$w_k + w_k^- - w_k^+ = \frac{1}{\bar{K}} \sum_{k=1}^{\bar{K}} w_k \quad , \quad k=1, \dots, \bar{K} \quad (4.63)$$

$$w_k \leq w_{hi} x_{hik} + w_{enb}(1 - x_{hik}) + \sum_{i' \in \{j | w_{hj} > w_{hi}\}} w_{hi'} x_{hi'k} \quad , \quad h=1, \dots, \bar{H} \quad , \quad i=1, \dots, n_h \quad , \quad k=1, \dots, \bar{K} \quad (4.64)$$

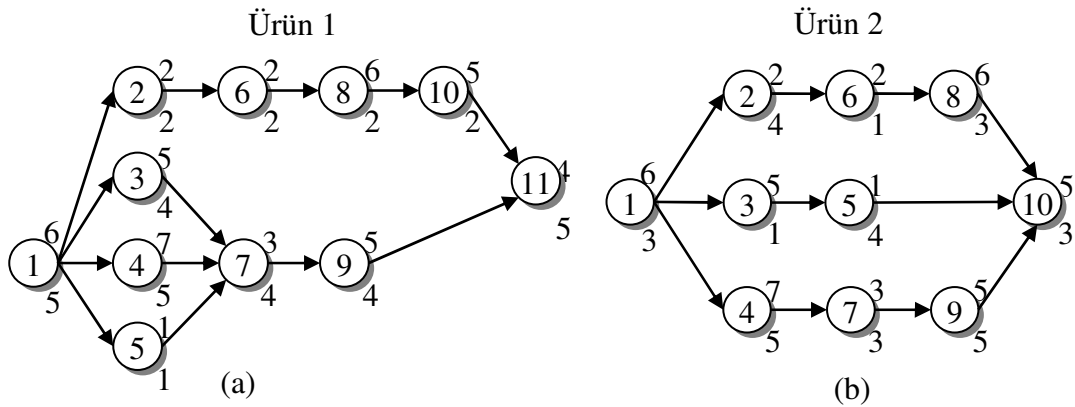
$$w_k \geq 0 \quad , \quad k=1, \dots, \bar{K} \quad (4.65)$$

$$x_{hik} , z_k , u_{hk} \in \{0,1\} \quad , \quad i=1, \dots, n_h \quad , \quad k=1, \dots, \bar{K} \quad , \quad h=1, \dots, \bar{H} \quad (4.66)$$

M-PMHD problemi için önerilen operatör odaklı tamsayı programlama modelinin amaç fonksiyonu [Eş. 4.55](#) ile değiştirilerek, ayrıca [Eş. 4.63](#) ve [4.64](#)'de gösterilen yeni kısıtlar da modele ilave edilmiştir. Dolayısıyla diğer kısıtlar operatör odaklı model ile aynı olduğu için burada tekrar açıklanmayacaktır (Bkz. [Bölüm 4.2](#)). Yeni eklenen [Eş.](#)

4.63’de gösterilen kısıt, operatörlerin ortalama ücret oranlarını hesaplayarak, bu değerden pozitif ve negatif sapmaları bulmaktadır. Eş. 4.62 ve 4.64’de gösterilen grup kısıtlar ise, her operatörün ücret oranını belirlemektedir. M-PMHD probleminin amacı toplam işgücü maliyetini enküçüklemek olduğu için, önerilen operatör odaklı tamsayılı karar modelinin amaç fonksiyonu (Eş. 4.12) operatör ücret oranlarını üstten sınırlandırmaktadır. Fakat ücret oranlarının düzgünleştirilmesinde bu amaç fonksiyonu değiştirildiği için, ücret oranlarını Eş. 4.64’de gösterilen kısıt yardımıyla üstten sınırlandırılmaktadır.

Örnek 4: Ücret oranlarının düzgünleştirilmesinin, toplam işgücü maliyeti üzerindeki etkisini görmek için örnek bir problem üzerinde önerilen tamsayılı doğrusal programlama modeli uygulanmıştır. Paralel konumda kurulmuş olan iki düz hatta birbirinden farklı olan iki ürünün montajı yapılmaktadır. Birinci hatta [şekil 4.7\(a\)](#)’da öncelik diyagramı verilen 11 görevden oluşan ürün 1’in montajı yapılırken, ikinci hatta [şekil 4.7\(b\)](#)’de öncelik diyagramı verilen 10 görevden oluşan ürün 2’nin montajı yapılmaktadır. Bu örnek problem için çevrim zamanı 19 olarak belirlenmiştir.



Şekil 4.7. (a) 11 görevli (b) 10 görevli öncelik diyagramları

Ücret oranlarını düzgünleştirmek amacıyla önerilen tamsayılı programlama modeli örnek probleme uygulanarak, IBM ILOG CPLEX 12.4 yazılımı yardımıyla yardımıyla problemin en iyi görev atamaları araştırılmıştır. Örnek problemin [çizelge 4.10](#)’daki en iyi görev atamalarına göre, paralel hatlarda çalışan tüm operatörlerin

Çok amaçlı optimizasyon problemleri matematiksel olarak aşağıdaki gibi ifade edilir:

$$\text{Enküçükleme } f(x) = \{f_1(x), f_2(x), \dots, f_n(x)\} \quad (4.67)$$

$$g_j(x) \leq 0 \quad , \quad j = 1, \dots, J \quad (4.68)$$

$$h_k(x) = 0 \quad , \quad k = 1, \dots, K \quad (4.69)$$

Burada; f , amaç fonksiyonları vektörü; x , karar değişkenleri vektörü; g_j, j . eşitsizlik kısıtı; h_k, k . eşitlik kısıtı; n , amaç fonksiyonu sayısı; J , eşitsizlik kısıtları sayısı ve K , eşitlik kısıtları sayısıdır. Yukarıda verilen çok amaçlı optimizasyon probleminin tek amaçlı optimizasyon problemlerinden tek farkı, amaç fonksiyonunun sadece bir amaçtan değil, bir amaç vektöründen meydana gelmesidir.

Çok amaçlı karar problemlerinde, çelişen amaçları eniyi kılan tek bir çözüm bulmak olanaksız olabilir. Bunun yerine, her amacın önem derecesini temel alan *uzlaşık çözümler* bulunabilir. Çok amaçlı karar problemlerinin çözümü için geliştirilmiş çok sayıda yöntem bulunmakla birlikte, en yaygın olarak kullanılan çok amaçlı karar verme tekniği *hedef programlama*'dır. Hedef programlamadaki temel düşünce, çok amaçlı karar probleminin tek amaçlı karar problemine dönüştürülmesidir. Kaynaklarda çok amaçlı bir karar probleminin tek amaçlı bir karar problemine dönüştürülmesi amacıyla iki farklı yöntem kullanılmaktadır. Bunlar:

1. Ağırlıklı hedef programlama
2. Öncelikli hedef programlama

Her iki yöntem de, eşitsizlik veya eşitlik durumundaki hedef denklemlerine pozitif ve negatif sapma değişkenlerinin eklenmesini ve eşitsizliklerin yönüne bağlı olarak bu sapma değişkenlerinin bazılarının enküçüklenmesini esas alır. Farklı hedef tipleri için hedef programlama formülasyonları ve enküçüklenmesi gereken sapma değişkenleri [çizelge 4.11](#)'de görüldüğü gibi özetlenebilir:

Çizelge 4.11. Hedef programlama formülasyonları

Hedef tipi	Formülasyon	Enküçüklenmesi gereken değişken
$f_i(x) \leq b_i$	$f_i(x) - d_i^+ + d_i^- = b_i$	d_i^+
$f_i(x) \geq b_i$	$f_i(x) - d_i^+ + d_i^- = b_i$	d_i^-
$f_i(x) = b_i$	$f_i(x) - d_i^+ + d_i^- = b_i$	$d_i^+ + d_i^-$

İki yöntem arasındaki farkı açıklamak amacıyla f_1 ve f_2 gibi iki amaç fonksiyonuna sahip çok amaçlı optimizasyon problemi düşünelim. Birinci amaç, f_1 'in b_1 'den küçük veya b_1 'e eşit; ikinci amaç f_2 'nin ise b_2 'ye eşit olması gerektiğini varsayalım. Yukarıdaki çok amaçlı problem, ağırlıklı hedef programlama yöntemi kullanılarak aşağıdaki gibi tek amaçlı problem haline dönüştürülür:

$$\text{Enküçükleme } (w_1^+ d_1^+ + w_2^+ d_2^+ + w_2^- d_2^-) \quad (4.70)$$

$$f_1(x) - d_1^+ + d_1^- = b_1 \quad (4.71)$$

$$f_2(x) - d_2^+ + d_2^- = b_2 \quad (4.72)$$

$$g_j(x) \leq 0 \quad , \quad j = 1, \dots, J \quad (4.68)$$

$$h_k(x) = 0 \quad , \quad k = 1, \dots, K \quad (4.69)$$

Yukarıdaki eşitliklerdeki w değerleri hedeflerin önem derecelerini ifade eden ağırlıkları, d değerleri ise hedeflerden sapmaları ifade eden sapma değişkenlerini ifade etmektedir. Örneğin; d_1^+ değişkeninin enküçüklenmesi f_1 hedefinin b_1 değerini aşma miktarının enküçüklenmesi anlamına gelir.

Aynı çok amaçlı problem öncelikli hedef programlama yöntemi kullanılarak aşağıdaki gibi tek amaçlı bir probleme dönüştürülebilir:

$$\text{Enküçükleme } \{d_1^+, (d_2^+ + d_2^-)\} \quad (4.73)$$

$$f_1(x) - d_1^+ + d_1^- = b_1 \quad (4.71)$$

$$f_2(x) - d_2^+ + d_2^- = b_2 \quad (4.72)$$

$$g_j(x) \leq 0 \quad , \quad j = 1, \dots, J \quad (4.68)$$

$$h_k(x) = 0 \quad , \quad k = 1, \dots, K \quad (4.69)$$

Öncelikli hedef programlama, karar vericinin çok sayıda hedefe önem derecelerine göre öncelikler vermesi ile işleyen bir yöntemdir. Problem öncelikle en yüksek önceliğe sahip hedef için en iyi çözüm bulunur ve elde edilen sonuç sabitlenir. Problem daha düşük önceliklerdeki hedefler için de eniyisi bulunmak suretiyle bütün hedefler tamamlanıncaya kadar çözülmeye devam edilir. Yukarıdaki çok amaçlı problem için birinci öncelik f_1 hedefine, ikinci öncelik ise f_2 hedefine aittir. Bu problemde öncelikle d_1^+ sapma değişkeni enküçüklenerek uygun çözüm alanı daraltılır, arkasından $(d_2^+ + d_2^-)$ değeri enküçüklenir.

Öncelikli hedef programlamadaki temel mantık, karar verici açısından üst seviyede önceliğe sahip hedeflerin daha alt seviyedeki önceliğe sahip hedeflerden önce tatmin edilmesini sağlamaktır. Belirli bir problem için bu hedefler ve öncelikleri karar vericilere göre farklılık gösterebilir.

M-PMHD problemi için birbiriyle çelişen üç hedef ele alınmıştır:

Hedef 1: toplam işgücü maliyetini enküçüklemek

Hedef 2: İşyeri (operatör) sayısını enküçüklemek

Hedef 3: hat uzunluğunu (malzeme taşıma sisteminin uzunluğunu) enküçüklemek

Önerilen hedef programlama modeli, M-PMHD probleminin çözümü için önerilen operatör odaklı tamsayı programlama modelinin geliştirilmiş halidir. Bu nedenle, operatör odaklı tamsayı karar modelinin yer alan parametre ve karar değişkenleri hedef programlama modeli için de geçerlidir. Hedef programlama modelinde farklı olarak kullanılan yeni parametrelerin ve hedef kısıtlarındaki sapma değişkenlerinin açıklamaları aşağıdaki [çizelge 4.12](#) ve [çizelge 4.13](#)'te verilmiştir.

Çizelge 4.12. Hedef programlama modelinde kullanılan parametre ve hedef değerleri

Parametre	Açıklama
TIM	toplam işgücü maliyeti için hedef değeri
OS	işyeri (operatör) sayısı için hedef değeri
HU	hat uzunluğu (konveyör bant uzunluğu) için hedef değeri
M	büyük bir sayı

Çizelge 4.13. Hedef programlama modelinde kullanılan değişkenler

Değişken	Açıklama
m^+	toplam işgücü maliyeti hedefi için pozitif sapma değişkeni
m^-	toplam işgücü maliyeti hedefi için negatif sapma değişkeni
o^+	gerekli olan operatör sayısı hedefi için pozitif sapma değişkeni
o^-	gerekli olan operatör sayısı hedefi için negatif sapma değişkeni
g^+	hat uzunluğu hedefi için pozitif sapma değişkeni
g^-	hat uzunluğu hedefi için negatif sapma değişkeni
NS	paralel hatların bant uzunluğunu gösteren tamsayılı değişken
s_{hk}	h hattında çalışan k operatörünün görev aldığı istasyonu gösteren tamsayılı değişken

M-PMHD probleminin yukarıdaki hedefler doğrultusunda uzlaşık çözümler bulmak amacıyla önerilen hedef programlama modeli aşağıdaki gibidir.

$$Enküçükleme \{m^+, o^+, g^+\} \quad (4.74)$$

$$\sum_{k=1}^{\bar{K}} w_k + m^- - m^+ = TIM \quad (4.75)$$

$$\sum_{k=1}^{\bar{K}} z_k + o^- - o^+ = OS \quad (4.76)$$

$$NS + g^- - g^+ = HU \quad (4.77)$$

$$\sum_{k=1}^{\bar{K}} x_{hik} = 1 \quad , \quad i = 1, \dots, n_h \quad , \quad h = 1, \dots, \bar{H} \quad (4.78)$$

$$\sum_{h=1}^{\bar{H}} \sum_{i=1}^{n_h} t_{hi} x_{hik} \leq c.z_k \quad , \quad k = 1, \dots, \bar{K} \quad (4.79)$$

$$\sum_{i=1}^{n_h} x_{hik} \leq n_h u_{hk} \quad , \quad k=1, \dots, \bar{K} \quad , \quad h=1, \dots, \bar{H} \quad (4.80)$$

$$\sum_{i=1}^{n_h} x_{hik} \geq u_{hk} \quad , \quad k=1, \dots, \bar{K} \quad , \quad h=1, \dots, \bar{H} \quad (4.81)$$

$$u_{hk} + u_{(h+a)k} \leq 1 \quad , \quad k=1, \dots, \bar{K} \quad , \quad \bar{H} \geq 3 \quad , \quad h=1, \dots, \bar{H}-2 \quad , \quad a=2, \dots, \bar{H}-h \quad (4.82)$$

$$\sum_{k=1}^{\bar{K}} (\bar{K}-k+1)(x_{hrk} - x_{hsk}) \geq 0 \quad , \quad \forall (r,s) \in P_{hs} \quad , \quad h=1, \dots, \bar{H} \quad (4.83)$$

$$w_k \geq w_{hi} x_{hik} \quad , \quad h=1, \dots, \bar{H} \quad , \quad i=1, \dots, n_h \quad , \quad k=1, \dots, \bar{K} \quad (4.84)$$

$$s_{hk} \geq \begin{cases} u_{hk} & , \quad k=1 & , \quad h=1, \dots, \bar{H} \\ s_{h(k-1)} + u_{hk} & , \quad k=2, \dots, \bar{K} & , \quad h=1, \dots, \bar{H} \end{cases} \quad (4.85)$$

$$s_{hk} \geq \begin{cases} s_{(h+1)k} + M(u_{hk} + u_{(h+1)k} - z_k - 1) & , \quad h=1, \quad k=2, \dots, \bar{K} \\ s_{(h-1)k} + M(u_{(h-1)k} + u_{hk} - z_k - 1) & , \quad 1 < h < \bar{H}, \quad k=2, \dots, \bar{K} \\ s_{(h+1)k} + M(u_{hk} + u_{(h+1)k} - z_k - 1) & , \quad 1 < h < \bar{H}, \quad k=2, \dots, \bar{K} \\ s_{(h-1)k} + M(u_{(h-1)k} + u_{hk} - z_k - 1) & , \quad h=\bar{H}, \quad k=2, \dots, \bar{K} \end{cases} \quad (4.86)$$

$$NS \geq s_{hk} \quad , \quad h=1, \dots, \bar{H} \quad , \quad k=\bar{K} \quad (4.87)$$

$$z_k \geq z_{k+1} \quad , \quad h=1, \dots, \bar{H}-1 \quad , \quad k=1, \dots, \bar{K} \quad (4.88)$$

$$w_k \geq 0 \quad , \quad s_{hk} \geq 0 \quad , \quad NS \geq 0 \quad , \quad k=1, \dots, \bar{K} \quad , \quad h=1, \dots, \bar{H} \quad (4.89)$$

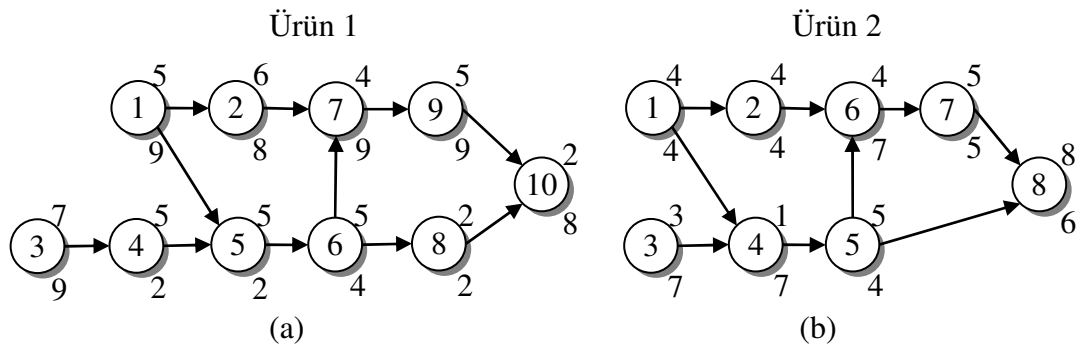
$$x_{hik} , z_k , u_{hk} \in \{0,1\} \quad , \quad i=1, \dots, n_h \quad , \quad k=1, \dots, \bar{K} \quad , \quad h=1, \dots, \bar{H} \quad (4.90)$$

Hedef programlama modelinin amaç fonksiyonu (Eş. 4.74), hedeflenen toplam işgücü maliyetinden pozitif sapmayı, hedeflenen işyeri (operatör) sayısından pozitif sapmayı ve hedeflenen hat uzunluğundan pozitif sapmayı enküçüklemektir. Eş. 4.75’de gösterilen kısıt, hedeflenen toplam işgücü maliyetinden pozitif ve negatif sapmaları; Eş. 4.76’da gösterilen kısıt, hedeflenen operatör sayısından pozitif ve negatif sapmaları; Eş. 4.77’de gösterilen kısıt ise hedeflenen hat uzunluğundan pozitif ve negatif sapmaları belirlemektedir. Eş. 4.78, 4.79, 4.80, 4.81, 4.82, 4.83, 4.84 ile gösterilen kısıtlar daha önceki bölümlerde ayrıntılı olarak açıklandığı için burada açıklanmayacaktır.

Eş. 4.85, 4.86, 4.87’de gösterilen kısıtlar hat uzunluğunu hesaplamaktadır. Eş. 4.88’de verilen kısıt, eğer üretim hattında görev atanmayan (boş) istasyonlar olursa, bu boş istasyonların hattın en sonunda olmalarını sağlamaktadır. Diğer bir anlatımla, en fazla istasyon sayısının $\bar{K}$ olduğu durumda, problemin en iyi çözümünde görevlerin atandığı istasyon sayısı $\bar{K} - 1$ ise, bir istasyon boş olacak demektir. Bu boş istasyon, $[1 - \bar{K}]$ aralığında numaralandırılmış ardışık istasyonlardan herhangi biri olabilir. En iyi çözümde herhangi bir değişiklik yapmayan ama gerçek hayatta da karşımıza çıkmayacak bu gibi durumları engellemek için Eş. 4.88’de verilen kısıt kullanılmaktadır.

Eş. 4.89’deki kısıt, karar değişkenlerin negatif değer almamasını sağlar. NS ve s_{hk} değişkenleri tamsayılı değişkenler olmasına rağmen, yer aldıkları kısıtlardaki diğer değişkenlerinin aldıkları değerler tamsayılı olduğu için bu değişkenler de tamsayılı değerler alırlar. Dolayısıyla bu değişkenler modelde özellikle tamsayılı değişken olarak tanıtılmamıştır. Eş. 4.90’de gösterilen kısıt, tamsayılı modeldeki karar değişkenlerinin 0 veya 1 değerini almalarını sağlamaktadır.

Örnek 5: Önerilen hedef programlama modeli [şekil 4.8](#)’de verilen örnek üzerinde uygulanarak, geçerliliği test edilmiştir. Birbirine paralel olan iki düz hatlı bir üretim sisteminde iki farklı ürünün üretilmektedir. 10 görevden oluşan ürün 1 ve 8 görevden oluşan ürün 2’nin montaj işlemleri sırasıyla hat 1 ve hat 2’de yapılmaktadır.



Şekil 4.8. (a) 10 görevli (b) 8 görevli öncelik diyagramları

Ürünlerin öncelik diyagramları, görevlerin tamamlanma zamanı ve görevlerin ücret oranları [şekil 4.8\(a\)](#) ve [şekil 4.8\(b\)](#)'de verilmiştir. Paralel hatlar arasındaki yürüme süreleri dikkate alınmamıştır. Bu örnek problem için çevrim zamanı 10 olarak belirlenmiştir. Ele alınan problemde yukarıda belirtilen üç farklı hedef dikkate alınmıştır.

Hedef 1: Toplam işgücü maliyeti hedefi: Toplam işgücü maliyetinin 560 pb'ni geçmemelidir.

Hedef 2: İşyeri (operatör) sayısı hedefi: Paralel hatlarda kullanılan işyeri (operatör) sayısı 8'i geçmemelidir.

Hedef 3: Paralel hatların uzunluğu hedefi: Atölyedeki alan kısıtı yüzünden hatlarda malzeme taşıma amaçlı kullanılan bant uzunluğu 5 birimi geçmemelidir.

Yukarıdaki üç hedefin her biri üç farklı önceliğe sahip olabilir. Hedeflerin farklı önceliklerde sıralanmaları sonucu $3!=6$ farklı öncelik sıralamasına göre çözüm elde edilebilir. Bu altı farklı öncelik sıralaması [çizelge 4.14](#)'te verilmiştir.

Çizelge 4.14. Üç hedef için farklı öncelik sıralamaları

Senaryo	Öncelik 1	Öncelik 2	Öncelik 3
1	Toplam işgücü maliyet	Operatör sayısı	Hat uzunluğu
2	Toplam işgücü maliyet	Hat uzunluğu	Operatör sayısı
3	Operatör sayısı	Toplam işgücü maliyet	Hat uzunluğu
4	Operatör sayısı	Hat uzunluğu	Toplam işgücü maliyet
5	Hat uzunluğu	Toplam işgücü maliyet	Operatör sayısı
6	Hat uzunluğu	Operatör sayısı	Toplam işgücü maliyet

Yukarıda verilen altı farklı öncelik sırası dikkate alınarak, belirlenen hedefler doğrultusunda açıklayıcı örnek problem önerilen hedef programlama modeli kullanılarak çözülmüştür. Uzlaşık çözümler, üç farklı hedefin sırasına bağlı olarak değişmektedir. Tüm çözümlerin açıklaması uzun süreceğinden dolayı sadece senaryo 1 için elde edilen çözümler açıklanmıştır. Senaryo 1 için ilk olarak en yüksek önceliğe sahip toplam işgücü maliyeti hedefi ele alınmıştır. Toplam işgücü

maliyetinin enküçüklenmesi için kullanılan hedef programlama modeli aşağıda verilmiştir:

Enküçüklemeye $\{m^+\}$

$$\sum_{k=1}^{\bar{K}} w_k + m^- - m^+ = 560 \quad (4.75)$$

Eş. (4.78), (4.79), (4.80), (4.81), (4.82), (4.83), (4.84), (4.85), (4.86), (4.87), (4.88), (4.89), (4.90)

Bu modelin en iyi çözümü bulunduğunda m^+ değerinin sıfır çıktığı görülmektedir. Yani; hedeflenen 560 pb toplam işgücü maliyeti hedefi sağlanmıştır. Birinci aşamada bulunan toplam işgücü maliyeti aşağıdaki güncellenmiş modeldeki Eş. 4.75'te görüldüğü gibi 560 pb olarak sabitlenmiştir. Ayrıca, operatör sayısı hedefindeki sapmaları belirleyecek olan Eş. 4.76'da gösterilen kısıt modele ilave edilmiştir. İkinci aşamada, $m^+ = 0$ değeri sabitlenerek işyeri (operatör) sayısını enküçükleyecek aşağıdaki hedef programlama modeli tekrar çözdürülür.

Enküçüklemeye $\{o^+\}$

$$\sum_{k=1}^{\bar{K}} w_k = 560 \quad (4.75)$$

$$\sum_{k=1}^{\bar{K}} z_k + o^- - o^+ = 8 \quad (4.76)$$

Eş. (4.78), (4.79), (4.80), (4.81), (4.82), (4.83), (4.84), (4.85), (4.86), (4.87), (4.88), (4.89), (4.90)

Hedef programlama modeli yeniden çözdürüldüğünde; o^+ değişkeninin en iyi çözümdeki değerinin 1 olduğu görülmektedir. Bunun anlamı toplam işgücü maliyeti 560 pb olan çözümler arasından operatör sayısı en az olan aranmış ve 9 olarak bulunmuştur. Son aşamada, o^+ değişkenin değeri de 9'a sabitlenerek (Bkz. Eş. 4.76)

ve Eş. 4.77’de gösterilen kısıt modele ilave edilerek, hat uzunluğunu enküçükleyecek aşağıdaki hedef programlama modeli son kez çözdürülmüştür.

Enküçükleme $\{g^+\}$

$$\sum_{k=1}^{\bar{K}} w_k = 560 \quad (4.75)$$

$$\sum_{k=1}^{\bar{K}} z_k = 9 \quad (4.76)$$

$$NS + g^- - g^+ = 5 \quad (4.77)$$

Eş. (4.78), (4.79), (4.80), (4.81), (4.82), (4.83), (4.84), (4.85), (4.86), (4.87), (4.88), (4.89), (4.90)

Modelin en iyi çözümünde g^+ değerinin 3 olduğu görülmektedir. Bunun anlamı 560 pb toplam işgücü maliyeti ve 9 operatörlü çözümler arasından en kısa hat uzunluğu 8 birimdir. Bu sonuçlar, örnek problem için toplam işgücü maliyeti hedefinin karşılandığını, fakat işyeri (operatör) sayısı ve hat uzunluğu hedeflerinin sağlanamadığı anlamına gelmektedir. Her öncelik sıralaması için bulunan toplam işgücü maliyeti, işyeri (operatör) sayısı ve hat uzunluğu değerleri çizelge 4.15’te verilmiştir.

Çizelge 4.15. Üç hedefin farklı öncelik sıralamaları için uzlaşık çözümler

Senaryo	m^+	o^+	g^+	Toplam işgücü maliyet	Operatör sayısı	Hat uzunluğu
1	0	1	3	560	9	8
2	0	1	3	560	9	8
3	10	0	3	570	8	8
4	30	0	2	590	8	7
5	60	1	0	620	9	5
6	60	1	0	620	9	5

Hedef programlama yaklaşımı, karar vericilerin hassasiyetleri ve öncelikleri doğrultusunda alternatif çözümler geliştirebilmektedir. Yukarıda ele alınan örnek

problemde, belirlenen üç hedefe de aynı anda ulaşmak mümkün olmamıştır. Karar verici tarafından belirlenen 3 hedef düzeyi düşünüldüğünde, [çizelge 4.15](#)'te verilen çözümlerden her biri, diğer çözümlere karşı baskın olamayan pareto çözüm kümesinin elemanlarıdır. Dolayısıyla istenen 3 hedef düzeyi düşünülerek 6 farklı çözüm içinden en iyisini belirlemek söz konusu değildir.

Karar vericiler, yukarıdaki [çizelge 4.15](#) doğrultusunda bir maliyet analizi yaparak kendileri için en uygun şartları tercih etmek durumundadırlar. Örneğin senaryo 1 ve senaryo 6 arasında bir seçim yapmak gerekirse şöyle bir maliyet analizi yapılabilir: Her iki çözümün operatör sayıları birbirine eşittir. Senaryo 1 ile nitelendirilen çözüm incelendiğinde, toplam işgücü maliyeti 560 pb ve hat uzunluğu 8 birim iken; senaryo 6 ile nitelendirilen çözümde, toplam işgücü maliyeti 620 pb ve hat uzunluğu ise 5 birimdir. Hat uzunluğunun bir birim artması demek, her iki hatta da malzeme taşıma sisteminin bir birim artması demektir. Diğer bir ifadeyle; hat uzunluğundaki her birim artış için, birbirine paralel olan iki düz hattın olmasından dolayı iki birimlik malzeme taşıma sistemine ihtiyaç vardır. Her iki senaryo ile elde edilen çözümlerdeki hat uzunlukları arasındaki fark 3 birim olduğu için 6 birimlik malzeme taşıma sistemi gerekmektedir. Bunun yanında, toplam işgücü maliyetleri arasındaki fark ise 60 pb'dir. Sonuç olarak; eğer bir çevrim zamanı için bir birim malzeme taşıma sisteminin sermaye maliyeti 10 pb olursa, toplam üretim maliyetleri birbirine eşit olacağı için her iki senaryo ile elde edilen görev atamaları seçilebilir. Bir çevrim zamanı için sermaye maliyeti 10 pb'nden küçük ise senaryo 1; 10 pb'den büyük ise senaryo 6 ile elde edilen görev atamaları seçilebilir.

Hedef programlama yaklaşımdaki hedefler dikkate alınmazsa ve paralel hatların eş zamanlı dengelenmesindeki amaç toplam üretim maliyetinin enküçüklenmesi ise, aşağıda verilen istasyon odaklı ve operatör odaklı tamsayılı karar modelleri kullanılabilir. Toplam üretim maliyeti, işgücü maliyeti ve sermaye maliyetinden oluşmaktadır. Sermaye maliyeti, kullanılan genel amaçlı araç-gereçlerin maliyetleri ve malzeme taşıma sisteminin maliyetinden oluşmaktadır. Toplam üretim maliyetini enküçüklemek amacıyla önerilen tamsayılı programlama modellerinde kullanılan yeni parametre ve karar değişkenleri [çizelge 4.16](#) ve [çizelge 4.17](#)'de açıklanmıştır.

Çizelge 4.16. Toplam üretim maliyetini enküçükleme için önerilen tamsayıli karar modellerinde kullanılan parametreler

Parametre	Açıklama
c_a	işyeri açılma maliyeti; ayrık veya ortak işyerlerinde ürünlerin montajı yapılırken kullanılan genel amaçlı araç-gereç ve makinelerin sermaye maliyeti; bu maliyet bir çevrim zamanı içindir. Ortak işyerlerinde çalışan operatörlerin genel amaçlı ekipmanları bitişik hatların komşu istasyonlarının her ikisinde kullanabileceği şekilde yerleştirileceği varsayılmaktadır.
c_b	Malzeme ve yarı mamulü bitişik istasyona taşımak için kullanılan konveyör bant sisteminin birim uzunluğunun (bir istasyon boyu) maliyeti; bu maliyet bir çevrim zamanı içindir.
k_h	h hattında montajı yapılan ürün için gerekli teorik istasyon sayısı; $k_h = \left\lceil \frac{\sum_{i=1}^{n_h} t_{hi}}{c} \right\rceil, \quad h=1, \dots, \bar{H}$
$k_{\min}$	Her hatta üretilecek ürünler için gerekli olan teorik istasyon sayılarının en büyüğü; , $k_{\min} = \max\{k_1, k_2, \dots, k_{\bar{H}}\}$

M-PMHD probleminin *her montaj hattında üretilen her birim ürünün toplam üretim maliyetini enküçükleyecek* en iyi görev atamalarını bulmak amacıyla önerilen istasyon odaklı ve operatör odaklı tamsayıli doğrusal karar modelleri aşağıdaki gibidir.

İstasyon odaklı tamsayıli karar modeli:

$$\begin{aligned}
 \text{Enküçükleme } TM = & c \cdot \sum_{h=1}^{\bar{H}} \sum_{k=1}^{\bar{K}} w n_{hk} + c \cdot \sum_{h=1}^{\bar{H}-1} \sum_{k=1}^{\bar{K}} w s_{h(h+1)k} \\
 & + c_a \left(\sum_{h=1}^{\bar{H}} \sum_{k=1}^{\bar{K}} u_{hk} - \sum_{h=1}^{\bar{H}-1} \sum_{k=1}^{\bar{K}} z_{h(h+1)k} \right) + c_b \cdot \bar{H} \cdot NS
 \end{aligned} \tag{4.91}$$

$$NS \geq k \cdot u_{hk} \quad , \quad k = k_{\min}, \dots, \bar{K} \quad , \quad h=1, \dots, \bar{H} \tag{4.92}$$

$$NS \geq 0 \tag{4.93}$$

Eş. (4.2), (4.3), (4.4), (4.5), (4.6), (4.7), (4.8), (4.9), (4.10), (4.11)

Amaç fonksiyonunun (Eş. 4.91) ilk parçası, ayrıık işyerlerinde çalışan operatörlerin ücret oranlarını; ikinci parçası, ortak işyerinde çalışan operatörlerin ücret oranlarını; üçüncü parçası, her işyerinde olan genel amaçlı ekipman ve makinelerin sermaye maliyetlerini (amortisman ve faiz değerleri); dördüncü parçası ise konveyör bant sisteminin sermaye maliyetini göstermektedir. Sermaye maliyetleri bir çevrim zamanı için amortisman ve faiz değerleridir.

Görevler paralel montaj hatlarında kurulan (ayrıık veya ortak) işyerlerinde yapılır. Bu işyerlerinde bir operatörün sorumlu olduğu görevleri yapabilmesi için gerekli araç-gereç ve makineler mevcuttur. Her ayrıık işyerinde aynı miktarda genel amaçlı ekipmanlar ve makineler vardır. Ayrıık işyerinde çalışan bir operatör sadece bir hattın sadece bir istasyonuna atanan görevleri yapmaktadır. Fakat ortak işyerinde çalışan operatör bitişik hatların komşu olan her iki istasyonuna atanan görevleri yapmakla sorumlu olduğu için, ekipman ve makinelerin yerleşiminde sıkıntı olabilir. Genel amaçlı ekipmanlar her türlü görevi yapmak için kullanıldığı için, bitişik hatların komşu istasyonlarında ortak olarak kullanılacak ekipmanlar hatlar arasında uygun bir yere yerleştirilebilir ve bu ekipmanlar her iki hattın istasyonunda sorunsuz şekilde kullanılabilecekse ekipman maliyetinde azalma sağlanabilir. Dolayısıyla ekipman ve makine maliyeti tamamen operatör sayısına bağlı olur. Eş. 4.91'deki amaç fonksiyonunun üçüncü parçası olan genel amaçlı ekipman maliyeti operatör sayısına bağımlı olarak yazılmıştır. Eğer genel amaçlı ekipmanlar her istasyonda olmak zorundaysa veya makineler her iki hattın komşu istasyonlarında ortak şekilde kullanılabilecek şekilde yerleştirilemezse, bu durumda görev atanmayan istasyonlar hariç her istasyona genel amaçlı ekipman ve makineler yerleştirmek gerekir. Her istasyona genel amaçlı araç-gereç ve makine yerleştirilmesi durumunda toplam üretim maliyeti aşağıdaki Eş. 4.94 yardımıyla hesaplanmaktadır:

$$Enküçüklem\ TM = c. \sum_{h=1}^{\bar{H}} \sum_{k=1}^{\bar{K}} wn_{hk} + c. \sum_{h=1}^{\bar{H}-1} \sum_{k=1}^{\bar{K}} wS_{h(h+1)k} + c_a \sum_{h=1}^{\bar{H}} \sum_{k=1}^{\bar{K}} u_{hk} + c_b \cdot \bar{H} \cdot NS \quad (4.94)$$

Paralel hatların eş zamanlı dengelenmesi kurulan ortak işyerleri neticesinde, bazı hatlarda boş istasyonlar meydana gelmektedir. Boş istasyonlarda herhangi bir görev yapılmazken, yarı mamulü ardışık istasyona taşınması için konveyör bant sistemine ihtiyaç vardır. Bu durumda boş istasyonlara görevlerin yapılabilmesi için gerekli araç-gereçler konmazken, malzemenin taşınması için bant sistemi kurulmak zorundadır. Dolayısıyla boş istasyonlar sadece bir istasyon boyu kadar bant maliyeti oluşturmaktadır. Her istasyonun boyunun aynı olduğu ve paralel hatların her birinin uzunluğunun eşit olduğu varsayılmaktadır.

Operatör odaklı tamsayılı karar modeli:

$$\text{Enküçükleme } TM = c \cdot \sum_{k=1}^{\bar{K}} w_k + c_a \sum_{k=1}^{\bar{K}} z_k + c_b \cdot \bar{H} \cdot NS \quad (4.95)$$

$$s_{hk} \geq \begin{cases} u_{hk} & , k = 1 \\ s_{h(k-1)} + u_{hk} & , k = 2, \dots, \bar{K} \end{cases} , h = 1, \dots, \bar{H} \quad (4.96)$$

$$s_{hk} \geq \begin{cases} s_{(h+1)k} + M(u_{hk} + u_{(h+1)k} - z_k - 1) & , h = 1, k = 2, \dots, \bar{K} \\ s_{(h-1)k} + M(u_{(h-1)k} + u_{hk} - z_k - 1) & , 1 < h < \bar{H}, k = 2, \dots, \bar{K} \\ s_{(h+1)k} + M(u_{hk} + u_{(h+1)k} - z_k - 1) & , 1 < h < \bar{H}, k = 2, \dots, \bar{K} \\ s_{(h-1)k} + M(u_{(h-1)k} + u_{hk} - z_k - 1) & , h = \bar{H}, k = 2, \dots, \bar{K} \end{cases} \quad (4.97)$$

$$NS \geq s_{hk} \quad , h = 1, \dots, \bar{H} \quad , k = \bar{K} \quad (4.98)$$

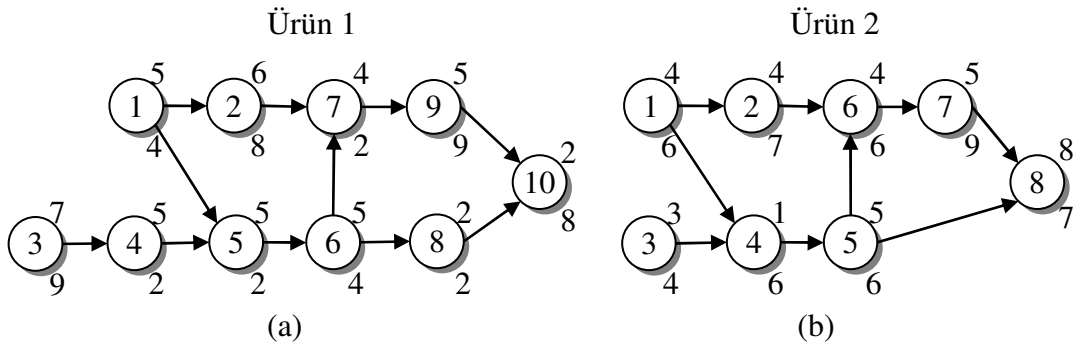
$$z_k \geq z_{k+1} \quad , h = 1, \dots, \bar{H} - 1 \quad , k = 1, \dots, \bar{K} \quad (4.99)$$

$$s_{hk} \geq 0 \quad , k = 1, \dots, \bar{K} \quad , h = 1, \dots, \bar{H} \quad (4.100)$$

Eş. (4.13), (4.14), (4.15), (4.16), (4.17), (4.18), (4.19), (4.20), (4.21)

Amaç fonksiyonunun (Eş. 4.95) ilk parçası işgücü maliyetini; ikinci parçası, her operatörün kullanması gereken genel amaçlı ekipman ve makine sermaye maliyetini; üçüncü parçası ise konveyör bant sistemi maliyetini göstermektedir. Operatör odaklı tamsayılı karar modelinin kısıtları daha önceki bölümlerde (Bkz. Bölüm 4.6) anlatıldığı için burada açıklanmamıştır. Yukarıda sunulan istasyon odaklı tamsayılı doğrusal karar modeli aşağıdaki açıklayıcı bir örnek üzerinde uygulanmıştır.

Örnek 6: Yan yana paralel bir şekilde konumlandırılmış iki düz montaj hattından oluşan bir üretim sisteminde iki farklı ürün üretilmektedir. Ürün 1, 10 görevden; ürün 2 ise 8 görevden oluşmaktadır. Hat 1’de montaj işlemi yapılacak olan ürün 1’in öncelik diyagramı [şekil 4.9\(a\)](#); hat 2’de montaj işlemi yapılacak olan ürün 2’nin öncelik diyagramı da [şekil 4.9\(a\)](#)’da gösterilmiştir. Paralel hatlar arasındaki yürüme süreleri sıfır olarak kabul edilmiştir. Bir çevrim zamanı için genel amaçlı ekipman maliyeti sırasıyla 10, 20 pb; bir istasyon uzunluğundaki konveyörün maliyeti sırasıyla 25, 50 pb alınmıştır. Düğümlerin üstündeki sayılar görev tamamlanma zamanlarını; düğümlerin altındaki sayılar ise görevlerin ücret oranlarını göstermektedir. Bu örnek problem için çevrim zamanı 10 olarak belirlenmiştir.



Şekil 4.9. (a) 10 görevli (b) 8 görevli öncelik diyagramları

Farklı sermaye maliyetlerinden oluşan M-PMHD problemlerinin toplam üretim maliyetini enküçükleyecek en iyi görev atamaları [çizelge 4.17](#)’de verilmiştir. Aynı problemi sadece toplam işgücü maliyetini enküçükleyecek şekilde çözdüğümüzde, hattın boyu 7 birim, operatör sayısı 8 ve toplam işgücü maliyeti 540 pb çıkmaktadır. Birim bant uzunluğunun maliyeti yükseldiğinde hattın uzunluğunun kısalması kaçınılmaz bir sonuçtur. Bunun yanında, hattın boyunun kısalması ortak işyeri sayısındaki azalmalara neden olmakta, bu durumda toplam işgücü maliyetini yükseltmektedir. Buradan çıkarılacak sonuç; bant uzunluğu azaldıkça toplam işgücü maliyeti artmakta; bant uzunluğu belirli değere kadar arttıkça toplam işgücü maliyeti azalmaktadır. Örnek problemin çözümüne göre, bant uzunluğu kısaldığı halde işyeri (operatör) sayısını aynı kalmıştır. Bu durumda, ekipman sermaye maliyetindeki

değişiklikler ihmal edilirse, tüm hat için katlanılan toplam ekipman maliyeti aynı kalmaktadır.

Çizelge 4.17. M–PMHD probleminin farklı sermaye maliyetlerinde elde edilen en iyi görev atamaları

Opr (<i>k</i>)	Genel amaçlı ekipman maliyeti: 10 Birim bant uzunluğu maliyeti: 25			Genel amaçlı ekipman maliyeti: 20 Birim bant uzunluğu maliyeti: 50		
	Hat 1	Hat 2	Ücret oranı	Hat 1	Hat 2	Ücret oranı
1	{1}	{3}	4	{3}	{3}	9
2	{3}	–	9	{1,4}	–	4
3	–	{1,4,5}	6	–	{1,4,5}	6
4	{2}	{2}	8	{5,6}	–	4
5	{4,5}	–	2	–	{2}	7
6	–	{6,7}	9	{2,7}	–	8
7	{6,7}	–	4	–	{6,7}	9
8	–	{8}	7	{8,9,10}	–	9
9	{8,9,10}	–	9	–	{8}	7
İşçilik maliyeti	58*10=580			63*10=630		
Operatör (işyeri) sayısı	9			9		
Ekipman maliyeti	9*10=90			9*20=180		
Bant uzunluğu	6			5		
Bant maliyeti	6*2*25=300			5*2*50=500		
Toplam üretim maliyeti	970			1310		
Toplam ortak işyeri sayısı	2			1		
Toplam ayırık işyeri sayısı	7			8		

4.7. Toplam İşgücü Maliyeti için Alt Sınır Algoritması

M–PMHD problemlerinin NP-zor yapısı nedeniyle, önerilen tamsayılı programlama modelleri ile makul sürelerde büyük boyutlu M–PMHD problemlerinin en iyi çözümleri elde edilememiştir. Bölüm 5’te anlatılan paralel kayan problem penceresi algoritması ile melez yapıda olan çok popülasyonlu paralel genetik algoritmanın performans karşılaştırmalarının yapılabilmesi için toplam işgücü maliyetinin bir alt sınır değerine ihtiyaç duyulmuştur. Alt sınır değeri hesabı için bir algoritma önerilmiştir ve algoritmanın adımları [şekil 4.10](#)’da verilmiştir. Önerilen algoritma görevlerin ücret oranlarının büyükten küçüğe doğru sıralanmasını ve

gruplandırılmasını gerektirmektedir. Algoritmada kullanılan notasyonlar [çizelge 4.18](#)'de tanıtılmıştır.

Çizelge 4.18. Toplam işgücü maliyeti alt sınır değeri hesabı algoritmasında kullanılan notasyonlar

Parametre	Açıklama
W	farklı ücret braketleri sayısı
C	çevrim zamanı
w	ücret braketleri numarası (indeksi), $w = 1, \dots, W$
w_{hi}	i görevinin ücret oranı , $h = 1, \dots, \overline{H}$
k_w	w ücret braketinin ücret oranı
t_{hi}	i görevinin tamamlanma zamanı , $h = 1, \dots, \overline{H}$
d_w	w ücret braketinde bulunan görevlerin toplam tamamlanma zamanı
d_{kalan}	mevcut istasyondaki boş zaman
M_w	w ücret braketinde bulunan tüm görevlerin atanması için gerekli istasyon sayısı
M_{lb}	tüm görevlerin uygun olarak atanabileceği en az istasyon sayısı
M_{toplam}	görevlerin atandığı toplam istasyon sayısı
k_{lb}	toplam işgücü maliyeti için alt sınır değeri

Algoritmanın ilk adımında her ücret braket değeri (k_w) birbirinden farklı olması şartıyla, tüm görevlerin ücret oran değerleri (w_{hi}) dikkate alınarak aynı olanlar aynı ücret braketinde yer alacak şekilde gruplandırılmaktadır. Oluşturulan her ücret braketinde bulunan tüm görevler, görevlerin bölünebilir olmaları varsayımına göre toplu halde istasyonlara atanmaya çalışılır. Tüm görevler ücret braketleri sırasına göre istasyonlara atandıktan sonra, bulunan istasyon sayısı teorik istasyon sayısıdır. Diğer bir ifadeyle, görevler bölünerek istasyonlara atandığı ve öncelik ilişkileri dikkate alınmadığı için bulunan istasyon sayısı teorik istasyon sayısına eşittir. Fakat, problemi tüm kısıtlarıyla birlikte ele aldığımızda istasyon sayısında ($M_{lb} - M_{toplam}$) kadar fark çıkmaktadır. M_{lb} değeri [Gökçen ve ark. \(2006\)](#) çalışmasındaki en küçük istasyon sayı değerleridir. Bu fark en küçük ücret braket değerini ve çevrim zamanını düşünerek toplam işgücü maliyetine ilave edilir.

Algoritma: *Alt sınır hesabı*

- Adım 0: $w_{hi}, t_{hi}, i = 1, \dots, n_h, , h = 1, \dots, H, k_w, w = 1, \dots, W, C$
- Adım 1: $d_w, w = 1, \dots, W$ değerlerini hesapla
- Adım 2: $k_{lb} = 0, M_w = 0, d_{kalan} = 0, w = W$
- Adım 3: Eğer $d_w > d_{kalan}$ ise $M_w = \left\lceil \frac{d_w - d_{kalan}}{C} \right\rceil$; Değilse $M_w = 0$
- Adım 4: $k_{lb} = k_{lb} + M_w \cdot k_w \cdot C$
- Adım 5: $d_{kalan} = d_{kalan} + M_w \cdot C - d_w$
- Adım 6: $M_{toplama} \leftarrow M_{toplama} + M^l$
- Adım 6: Eğer $w > 1$ ise $w = w - 1$ ve Adım 3'e dön.
- Adım 7: $k_{lb} = k_{lb} + (M_{lb} - M_{toplama}) \cdot k_1 \cdot C$ ve DUR
-

Şekil 4.10. Toplam işgücü maliyetinin alt sınır değeri hesabı için bir algoritma

5. M-PMHD PROBLEMLERİ İÇİN SEZGİSEL ALGORİTMALAR

MHD problemleri için geliştirilen çözüm yöntemleri *kesin* ve *sezgisel algoritmalar* olmak üzere ikiye ayrılır. Kesin algoritmalar, her boyuttaki problemin en iyi çözümünü garanti eden yöntemlerdir. MHD probleminin en iyi çözümünü bulmak amacıyla geliştirilen kesin algoritmalarda, çözüm uzayındaki her noktanın taranması gerektiği için problem boyutunun artmasıyla birlikte üstel olarak artan hesaplama süresine ihtiyaç duyulmaktadır. Bu durum çözüm süresinin uzamasına neden olmaktadır. Problemin en iyi çözümü için kullanılan algoritmaların, sonuca kısa sürede ulaşması esastır.

MHD problemi, NP-zor bir kombinatorial optimizasyon problemidir [Gutjahr ve Nemhauser, 1964, Wee ve Magazine, 1982]. Bu da gerçek hayattaki büyük boyutlu bir problemin gelişmiş bilgisayarlarla dahi analitik yöntemler kullanılarak çözülmesi oldukça uzun çözüm sürelerine yol açmaktadır. Dolayısıyla, kesin yöntemler küçük boyutlu problemlerin en iyi çözümlerini bulmasına rağmen, problem boyutunun artmasıyla birlikte gerekli bilgisayar hesaplama zamanının da üstel olarak artması sonucu bu yöntemleri uygulamak mümkün olmamaktadır. Bu sebeple, MHD problemlerinin en iyi değerine yakın sonuçlar veren analitik yöntemlere oranla çok daha hızlı olan sezgisel yöntemler önemli bir konuma gelmişlerdir.

Sezgisel algoritmalar, büyük boyutlu problemlerde daha az işlem ve hesaplama süresiyle en iyi değere yakın, yüksek kalitede çözümler üreten yöntemlerdir. MHD problemi için en etkili sonuçlar veren yöntemler *meta-sezgisel algoritmalar*dır. Bu yöntemlerin en önemli özelliği sadece belli bir tip problem için değil, tüm kombinatorial problemlere uygulanabilir esneklikte olmasıdır. Meta-sezgisel yöntemler kaynaklarda son zamanlarda oldukça yaygın uygulama alanı bulmuş ve başarılı sonuçlar üretmiş tekniklerdir.

Bu tezde, büyük boyutlu M-PMHD problemlerin yaklaşık çözümlerini bulmak amacıyla *etkin bir matematiksel model tabanlı sezgisel algoritma* ve *melez yapıda çok popülasyonlu paralel genetik algoritma* önerilmiştir.

5.1. Paralel Kayan Problem Penceresi Algoritması

Her hatta sadece bir çeşit ürünün montaj işleminin yapıldığı paralel montaj hatlarında eş zamanlı olarak üretilen aynı veya farklı ürünlere ait görev sayılarının az olduğu (küçük boyutlu) M-PMHD problemlerinin en iyi çözümleri [bölüm 4](#)'te sunulan tamsayılı programlama modelleri kullanılarak makul sürelerde bulunabilmektedir. Fakat, M-PMHD probleminin NP-zor yapısı nedeniyle hat sayısının ve bu hatlarda montajı yapılan ürünlere ait görev sayılarının artmasıyla birlikte (büyük boyutlu) M-PMHD problemlerinin en iyi çözümlerini tamsayılı programlama modeli ile makul sürelerde bulmak imkansız hale gelmektedir. Bu başlık altında, makul sürelerde en iyi çözümü bulunamayan büyük boyutlu M-PMHD problemlerinin yüksek kalitede çözümlerini kısa sürede bulan matematiksel model tabanlı bir sezgisel algoritma sunulmuştur.

MHD probleminin öncül ilişkileri kısıtı düşünüldüğünde, bir ürünün üretilmesi için gerekli olan görevler belirli sıraya göre yapılmak zorundadır. Önerilen sezgisel algoritmanın temel mantığı, büyük boyutlu bir M-PMHD problemini her hatta üretilen ürüne ait belirli sayıda *topolojik* olarak sıralanmış görevlerden oluşan küçük boyutlu alt problemlere ayırarak, bu küçük boyutlu problemlerin en iyi çözümlerini önerilen tamsayılı programlama modeli yardımıyla bulmaktadır. Küçük boyutlu alt problemlerin elde edilen en iyi çözümlerinin bir kısmı belirli sıraya göre birleştirildiğinde, asıl problemin en iyi değerine yakın yüksek kalitede bir çözüm elde edilebilmektedir. Her ürüne ait belirli sayıda topolojik sıralanmış görevlerden oluşan alt probleme *problem penceresi* denilmektedir. Bir problem penceresinde gösterilen bir alt probleminin en iyi çözümü, diğer alt problemlerden bağımsız olarak bulunmaktadır. Fakat, bir problem penceresini oluşturan görevlerin belirlenmesi kendinden önce bulunan diğer alt problemin en iyi çözümüne bağlıdır. Her alt probleminin en iyi çözümü araştırılırken dikkate alınan amaç, toplam işgücü maliyetini enküçükmektir. Problem penceresi denilen alt problemlerin en iyi çözümlerini tamsayılı programlama modeli ile bularak ve bu çözümlerin belirli kısımlarını belirli sıraya göre birleştirildiği bu yöntem, *paralel kayan problem penceresi* (PKPP) olarak isimlendirilmiştir.

Matematiksel model tabanlı PKPP algoritması iki aşamadan oluşmaktadır. Birinci aşamada, her ürüne ait belirli sayıda topolojik olarak sıralanmış görevlerden oluşan problem pencerelerinin en iyi çözümleri arka arkaya bulunmaktadır. Problem pencerelerinin en iyi görev atamalarının belirli kısmı hiç değiştirilmeden kabul edilmektedir. Diğer kalan kısımdaki görevler ise bir sonraki problem penceresinin elemanı haline getirilmektedir. Dolayısıyla, bir problem penceresinde yer alan görevlerin belirlenmesi, bir önceki adımda oluşturulan problem penceresinin en iyi çözümüne bağlıdır. Birinci aşamanın sonunda, ardışık olarak oluşturulan problem pencereleri için bulunan en iyi görev atamalarının belirli kısımları birleştirilerek, M-PMHD problemi için uygun görev atamaları bulunmaktadır. Bu aşamada elde edilen uygun görev atamaları, ikinci aşamanın başlangıç çözümünü oluşturmaktadır.

PKPP algoritmasının ikinci aşamasında, başlangıç çözümünde belirli istasyonlara atanan görevler alınarak yeni problem pencereleri oluşturulmaktadır. Diğer bir ifadeyle, birinci aşamada oluşturulan problem pencerelerinin en iyi çözümlerinin belirli kısımları arka arkaya eklenerek sabitlenmektedir. İkinci aşamada ise, birinci aşamadaki arka arkaya sabitlenmiş görevler birlikte ele alınarak yeni problem pencereleri oluşturulmaktadır. Yeni oluşturulan problem pencerelerinin en iyi çözümleri bulunarak asıl problemin amaç fonksiyonu iyileştirilmeye çalışılmaktadır.

M-PMHD probleminin çözümü için önerilen iki aşamalı PKPP algoritması, [Amen \(2000\)](#) tarafından M-MHD probleminin çözümü için geliştirdiği algoritmanın geliştirilmiş halidir. [Amen \(2000\)](#)'nin tek modelli tek hatlı montaj hattının maliyet tabanlı yaklaşımla dengelenmesi için önerdiği algoritma, bu tezde önerilen PKPP algoritmasının sadece birinci aşamasını kapsamaktadır. Ayrıca [Amen \(2000\)](#) çalışmasında, alt problemlerinin en iyi çözümlerinin bulunması için dal-sınır algoritması kullanmıştır.

Önerilen matematiksel model tabanlı PKPP algoritmasının birinci ve ikinci aşamasında kullanılan notasyonlar aşağıdaki [çizelge 5.1](#)'de açıklanmıştır.

Çizelge 5.1. Paralel kayan problem penceresi algoritmasında kullanılan notasyonlar

Notasyon	Açıklama
h	hat numarası (indeksi), $h = 1, \dots, \overline{H}$, $\overline{H}$: en fazla hat sayısı
k	istasyon numarası (indeksi), $k = 1, \dots, \overline{K}$, $\overline{K}$: en fazla istasyon sayısı
n_h	h hattında montajı yapılan ürüne ait görev sayısı
i, j	görev numarası (indeksi), $i = 1, \dots, n_h$
t_{hi}	h hattında montajı yapılan ürüne ait i görevinin tamamlanma zamanı
P_{hi}	h hattında montajı yapılacak ürüne ait i görevinin öncül görevler kümesi
F_{hi}	h hattında montajı yapılacak ürüne ait i görevinin ardıl görevler kümesi
PW_{hi}	h hattında montajı yapılacak ürüne ait i görevinin pozisyon ağırlığı; $PW_{hi} = t_{hi} + \sum_{j \in F_{hi}} t_{hj}$
α	sabitlenme oranı; $0 < \alpha \leq 1$
L	problem penceresindeki görev sayısı; $L \leq n_h$, $h = 1, \dots, \overline{H}$
I_{hk}	h hattındaki k istasyonuna atanan görevler kümesi (1. aşama için)
I_{hk}^*	h hattındaki k istasyonuna atanan görevler kümesi (2. aşama için)
$I_{h,pencere}$	problem penceresini oluşturan h hattında montajı yapılan ürüne ait görevler kümesi $h = 1, \dots, \overline{H}$
$I_{h,kalan}$	henüz herhangi bir istasyona atanmamış h hattında montajı yapılan ürüne ait görevler kümesi $h = 1, \dots, \overline{H}$
$K_{pencere}$	Alt problemin en iyi çözümde problem penceresini oluşturan görevlerin atandığı her hatta kurulan istasyon sayısı
K_{sabit}	sabitlenecek istasyon sayısı; $K_{sabit} = \max\{1, \lfloor \alpha \cdot K_{pencere} \rfloor\}$
$K_{toplamlam}$	toplam sabitlenen istasyon sayısı; $K_{toplamlam} = K_{toplamlam} + K_{sabit}$
S	istasyon sayısı; $2 \leq S < K_{toplamlam}$
K_{ilk}	ilk istasyon indeksi
K_{son}	son istasyon indeksi
$f(p)$	problem penceresini oluşturan görevlerin mevcut atamalarına göre amaç fonksiyonu değeri
$f^*(p)$	alt problemin (problem penceresinin) en iyi değeri
imp	amaç fonksiyonunda iyileşme sağlandığını gösteren değişken
δ	kayma miktarı

PKPP algoritmasının birinci ve ikinci aşamasının adımları sırasıyla [şekil 5.1](#) ve [şekil 5.2](#)'de verilmiştir.

Algoritma: Paralel kayan problem penceresi algoritmasının kurucu sezgiseli

Adım 1: *Sıralı pozisyon ağırlık yöntemine* göre her bir hatta montajı yapılan ürüne ait görevlerin, diğer hatlardaki görevlerden bağımsız olarak *pozisyon ağırlıklarını* hesapla:

$$PW_{hi} = t_{hi} + \sum_{j \in F_{hi}} t_{hj} \quad , \quad h = 1, \dots, \overline{H} \quad , \quad i = 1, \dots, n_h$$

Her hat için topolojik olarak sıralanmış görev kümelerini belirle:

$$I_{h,kalan} = \{ 1, \dots, i, \dots, j, \dots, n_h \} \quad , \quad h = 1, \dots, \overline{H} \quad , \quad i < j \quad , \quad w_{hi} > w_{hj}$$

L değerini belirle

Adım 2: $I_{h,kalan}$ kümelerindeki herhangi bir istasyona atanmamış ilk L görevi problem penceresi olarak belirle: $I_{h,pencere} = \{ 1, \dots, L \} \quad , \quad h = 1, \dots, \overline{H}$

$$K_{toplamlam} = 0$$

Adım 3: $I_{h,pencere}$ problem penceresinin en iyi çözümünü önerilen tamsayıli karar modeli yardımıyla bul:

her hatta kurulan istasyon sayısı : $K_{pencere}$

istasyonların iş yükü: $I_{hk} \quad , \quad k = K_{toplamlam} + 1, \dots, K_{toplamlam} + K_{pencere} \quad , \quad h = 1, \dots, \overline{H}$

Adım 4: Elde edilen en iyi çözümdeki $K_{pencere}$ istasyonun ilk K_{sabit} istasyonunu atanan görevleriyle birlikte sabitle:

$$K_{sabit} = \max \{ 1, \lfloor \alpha \cdot K_{pencere} \rfloor \} \quad ; \quad I_{hk} \quad , \quad k = K_{toplamlam} + 1, \dots, K_{toplamlam} + K_{sabit}$$

$$\text{Adım 5: } I_{h,kalan} = I_{h,kalan} - \bigcup_{k=K_{toplamlam}+1}^{K_{toplamlam}+K_{sabit}} I_{hk} \quad , \quad h = 1, \dots, \overline{H}$$

$$\text{Adım 6: } K_{toplamlam} = K_{toplamlam} + K_{sabit}$$

Adım 7: Eğer $|I_{h,kalan}| > L \quad , \quad h = 1, \dots, \overline{H}$ ise,

$I_{h,kalan}$ kümelerinin ilk L görevini problem penceresi olarak belirle.
ve Adım 3'e dön

Adım 8: $I_{h,kalan}$ son problem penceresinin en iyi çözümünü bul:

$$K_{pencere}; \quad I_{hk} \quad , \quad k = K_{toplamlam} + 1, \dots, K_{toplamlam} + K_{pencere}$$

Adım 9: Elde edilen en iyi çözümdeki $K_{pencere}$ istasyonlarının tümünü atanan görevleriyle birlikte sabitle: $K_{sabit} = K_{pencere}$

$$I_{hk} \quad , \quad k = K_{toplamlam} + 1, \dots, K_{toplamlam} + K_{pencere}$$

$$K_{toplamlam} = K_{toplamlam} + K_{pencere}$$

İkinci aşamaya geç.

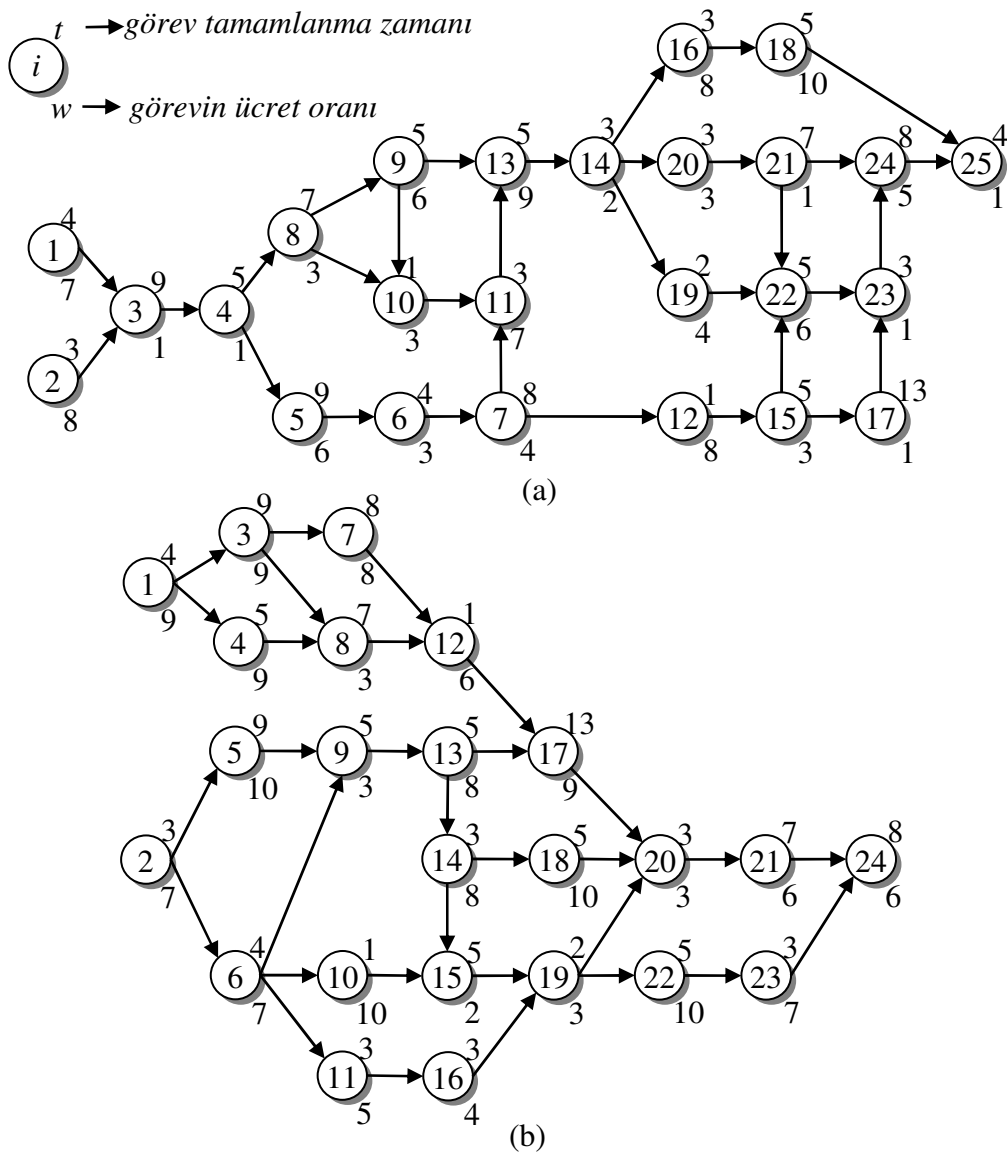
Şekil 5.1. PKPP algoritmasının birinci aşamasının adımları

Algoritma: Paralel kayan problem penceresi algoritmasının iyileştirme sezgiseli

- Adım 1: Birinci aşamanın sonunda elde edilen uygun çözüm:
 $I_{hk}, h=1, \dots, \overline{H}, k=1, \dots, K_{toplaml}$
İstasyon sayısını belirle: S ; kayma miktarını belirle: δ
- Adım 2: $K_{ilk} = 1, K_{son} = S; imp = 0$
- Adım 3: I_{hk} görev atamalarına göre $[K_{ilk} - K_{son}]$ aralığındaki S istasyona atanan görevlerden oluşan problem penceresini belirle ve $f(p)$ değerini hesapla
 $I_{h,pencere} \in I_{hk}, h=1, \dots, \overline{H}, k=K_{ilk}, \dots, K_{son}$
- Adım 4: Problem penceresinin en iyi çözümünü tamsayıli karar modeli ile bul:
 $K_{pencere}; I_{hk}^*, h=1, \dots, \overline{H}, k=K_{ilk}, \dots, K_{son}; f^*(p)$
- Adım 5: Eğer $f^*(p) < f(p)$ ise, en iyi çözümü mevcut çözümle değiştir: $imp = 1$
 $I_{hk} \in I_{hk}^*, h=1, \dots, \overline{H}, k=K_{ilk}, \dots, K_{son}$
Değilse; $f^*(p) = f(p)$ ise, [Adım 7](#)'ye git.
- Adım 6: Eğer $K_{pencere} < S$ ise,
Boş istasyonları ortadan kaldıracak şekilde sola kaydırma yap
 $K_{toplaml} \leftarrow K_{toplaml} - (S - K_{pencere})$
- Adım 7: $K_{ilk} = K_{ilk} + \delta, K_{son} = K_{son} + \delta$
- Adım 8: Eğer $K_{son} \leq K_{toplaml}$ ise, [Adım 3](#)'e dön.
Değilse; $K_{son} > K_{toplaml}$ ise, [Adım 9](#)'e git
- Adım 9: I_{hk} görev atamalarına göre $[K_{ilk} - K_{toplaml}]$ aralığındaki $K_{toplaml} - K_{ilk}$ istasyona atanan görevlerden oluşan son problem penceresini belirle:
 $I_{h,pencere} \in I_{hk}, h=1, \dots, \overline{H}, k=K_{ilk}, \dots, K_{toplaml}; f(p)$ değerini hesapla
- Adım 10: Son problem penceresinin en iyi çözümünü bul:
 $K_{pencere}; I_{hk}^*, h=1, \dots, \overline{H}, k=K_{ilk}, \dots, K_{toplaml}; f^*(p)$
- Adım 11: Eğer $f^*(p) < f(p)$ ise, en iyi çözümü mevcut çözümle değiştir: $imp = 1$
 $I_{hk} \in I_{hk}^*, h=1, \dots, \overline{H}, k=K_{ilk}, \dots, K_{toplaml}$
- Adım 12: Eğer $K_{pencere} < S$ ise,
Boş istasyonları ortadan kaldıracak şekilde sola kaydırma yap
 $K_{toplaml} \leftarrow K_{toplaml} - (S - K_{pencere})$
- Adım 13: Eğer $imp = 1$ ise,
[Adım 2](#)'ye dön.
Değilse, [DUR](#)
-

Şekil 5.2. PKPP algoritmasının ikinci aşamasının adımları

Şekil 5.1 ve 5.2’de verilen PKPP algoritmasının adımları ayrıntılı olarak aşağıdaki bir örnek üzerinde açıklanmıştır. Önerilen sezgisel algoritmanın başarısını açık bir şekilde görebilmek için, PKPP algoritması bölüm 4’te verilen örneklere göre daha büyük boyutlu bir problem üzerinde uygulanmıştır. Ele alınan örnek problemde birbirine paralel iki düz hatta iki farklı ürün üretilmektedir. Birinci hatta montajı yapılan ürün 1 ve ikinci hatta montajı yapılan ürün 2’nin öncelik diyagramları sırasıyla şekil 5.3(a) ve 5.3(b)’de gösterilmektedir.



Şekil 5.3. (a) 25 görevli (b) 24 görevli öncelik diyagramları

Her iki ürün için ortak olan çevrim zamanı 30 zb olarak belirlenmiştir. Alt problemlerdeki görev sayısı (L), 15; sabitleme oranı (α), 0.7; algortimanın ikinci aşamasında ele alınacak istasyon sayısı (S), 6; kayma miktarı (δ) ise 1 olarak alınmıştır.

PKPP algoritması paralel hatlarda montajı yapılan ürünlere ait topolojik olarak sıralanmış görevlerin üzerinden çalışmaktadır. Bu nedenle, M-PMHD problemini çözmek amacıyla önerilen PKPP sezgisel algoritmasında [Helgeson ve Birnie \(1961\)](#) tarafından geliştirilmiş *sıralı pozisyon ağırlık yöntemi* kullanılarak görev atamaları gerçekleştirilmiştir. Sıralı pozisyon ağırlık yöntemi, kendisinden sonra gelen ardıl görevlerin toplam tamamlanma zamanı fazla olan görevleri öncelikli olarak atanması prensibine dayanmaktadır. Bu prensibe göre, her bir görev için bir pozisyon ağırlığı hesaplanmakta ve bu hesaplanan ağırlık değerlerine göre görevler büyükten küçüğe doğru sıralanarak istasyonlara atanmaktadır. Bir görevin *pozisyon ağırlığı*; görevin kendi tamamlanma zamanı ile kendisinden sonraki (ardıl) görevlerin tamamlanma zamanlarının toplanmasıyla [Eş. 5.1](#)'de görüldüğü gibi hesaplanmaktadır:

$$PW_{hi} = t_{hi} + \sum_{j \in F_{hi}} t_{hj} \quad , \quad h = 1, \dots, \overline{H} \quad , \quad i = 1, \dots, n_h \quad (5.1)$$

[Eş. 5.1](#)'de kullanılan notasyonlar [çizelge 5.1](#)'de açıklanmıştır. Her ürüne ait görevlerin pozisyon ağırlık değerleri, diğer ürün(ler)e ait görevlerden bağımsız olarak hesaplanmaktadır. [Şekil 5.3\(a\)](#) ve [5.3\(b\)](#)'de verilen iki farklı ürüne ait görevlerin öncelik ilişkilerine göre [Eş. 5.1](#)'deki denklem kullanılarak hesaplanan her görevin pozisyon ağırlığı [çizelge 5.2](#)'de verilmiştir. Görevleri topolojik olarak sıralamak amacıyla, büyükten küçüğe doğru sıralanan pozisyon ağırlık değerlerine göre [Eş. 5.2](#)'de gösterildiği gibi görevler yeniden numaralandırılmıştır:

$$I_h = \{ 1, \dots, i, \dots, j, \dots, n_h \} \quad , \quad h = 1, \dots, \overline{H} \quad , \quad i < j \quad , \quad w_{hi} > w_{hj} \quad (5.2)$$

I_h , h hattında montajı yapılan ürüne ait görevlerin kümesini göstermektedir. Görevlerin yeni numaraları [çizelge 5.2](#)'nin üçüncü ve beşinci sütunlarında verilmiştir.

Çizelge 5.2. Ürünlere ait görevlerin pozisyon ağırlıkları

Görev no	Ürün 1		Ürün 2	
	Pozisyon ağırlıkları	Sıra no	Pozisyon ağırlıkları	Sıra no
1	122	1	65	3
2	121	2	79	1
3	118	3	56	4
4	109	4	44	6
5	91	5	50	5
6	82	6	67	2
7	78	7	40	9
8	64	8	39	10
9	57	9	41	7
10	52	10	34	12
11	51	11	34	13
12	39	14	32	15
13	48	12	36	11
14	43	13	41	8
15	38	15	33	14
16	12	22	31	16
17	28	17	31	17
18	9	24	23	19
19	22	19	28	18
20	30	16	18	20
21	27	18	15	22
22	20	20	16	21
23	15	21	11	23
24	12	23	8	24
25	4	25		

PKPP algoritmasının her iterasyonunda oluşturulan alt problemlerin en iyi çözümleri [bölüm 4.2](#)'de M-PMHD probleminin en iyi çözümünü bulmak için önerilen tamsayı doğrusal karar modeli kullanılarak, IBM ILOG CPLEX 12.4 yazılımı yardımıyla bulunacaktır. Tamsayı karar modelindeki bütün kısıtlar, özellikle de çevrim zamanını kısıtı sadece problem penceresinde tanımlı olan görevler için yazılır. Alt problemlerin en iyi çözümlerinin bulunabilmesi için kesin çözüm yöntemlerinden dal-sınır algoritması da kullanılabilir. Burada önemli olan alt

problemlerin en iyi çözümlerinin bulunmasıdır. Fakat, tercih edilen çözüm yönteminin PKPP algoritmasının çalışma süresini etkileyeceği de unutulmamalıdır. Her ürünün topolojik olarak sıralanmış görevlerinden ilk L adet ($L=15$) görev ilk problem penceresini oluşturmak üzere seçilir:

Ürün 1 için: $I_{1,pencere}=\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15\}$

Ürün 2 için: $I_{2,pencere}=\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15\}$

$I_{1,pencere}$ ve $I_{2,pencere}$ kümelerindeki görevlerden oluşan alt problemin en iyi çözümü Çizelge 5.3'te verilmiştir.

Çizelge 5.3. PKPP algoritmasının birinci aşamasındaki birinci problem penceresinin en iyi görev atamaları

İstasyon	Hat 1	Hat 2	İş yükü	Ücret oranı
1	{1,2}	{1,2,3,5,12}	28	10
2	{3,4,8}	{7}	26	3
3	{5,6,7,9,10}	{13}	30	6
4	{11,12,14,15}	{4,6,10}	30	9
5	{13}	{8,9,11,14,15}	30	8

Çizelge 5.3'te gösterilen alt problemin en iyi görev atamalarına göre, paralel hatların komşu istasyonlarına atanan görevler 5 operatör tarafından bir çevrim zamanı içerisinde yapılmaktadır. Elde edilen bu çözüm, alt problemin en iyi çözümü olmasına rağmen, asıl problemin en iyi görev atamalarını yansıtmayabilir. Örneğin ürün 1'e ait 13 numaralı görev belkide asıl problemin en iyi çözümünde hat 1'in 6 numaralı istasyonuna atanmaktadır. Bu nedenle her alt problemin en iyi çözümünün tümünü olduğu gibi kabul etmek yerine sadece belirli kısmını kabul ederek, sonlardaki istasyonlara atanan görevleri tekrar atanması için serbest bırakılmaktadır. Alt problemin elde edilen en iyi çözümündeki görev atamalarının ne kadarlık bir kısmının kabul edileceği sabitleme oranı ($\alpha = 0,7$) değerine bağlıdır. Sabitleme oranı (0-1] aralığında bir rasyonel sayı ile belirtilir. Çizelge 5.3'te gösterilen birinci problem penceresinin en iyi görev atamalarına göre, her hatta kurulan istasyon sayısı 5'tir ($K_{pencere} = 5$). Bu istasyonlardan kaçının sabitleneceği aşağıda hesaplanmıştır:

$$K_{sabit} = \max\{1, \lfloor \alpha \cdot K_{pencere} \rfloor\} = \max\{1, \lfloor (0,7) \cdot 5 \rfloor\} = 3$$

Yukarıdaki işlemler sonucunda, her hatta sabitlenecek istasyon sayısı 3 olarak belirlenmiştir. Diğer bir ifadeyle, alt problemin en iyi çözümündeki her hattın ilk 3 istasyonu görevleriyle birlikte sabitlenir ve 4, 5 numaralı istasyonlara atanan görevler ise tekrar atanmak üzere serbest bırakılır. Sabitlenen görevler ilgili hattın $I_{h,kalan}$ kümesinden çıkartılır:

$$I_{1,kalan} = \{11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25\}$$

$$I_{2,kalan} = \{4, 6, 8, 9, 10, 11, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24\}$$

$|I_{1,kalan}| = 15 = L$ ve $|I_{2,kalan}| = 17 > L$ koşullarına göre, geriye kalan tüm görevler ikinci problem penceresini oluşturmak üzere seçilir:

$$\text{Ürün 1 için: } I_{1,pencere} = \{11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25\}$$

$$\text{Ürün 2 için: } I_{2,pencere} = \{4, 6, 8, 9, 10, 11, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24\}$$

Eğer $|I_{1,kalan}| > L$ ve $|I_{2,kalan}| > L$ koşulları sağlansaydı, bu durumda ikinci problem penceresini oluşturmak için $I_{1,kalan}$ ve $I_{2,kalan}$ kümelerinin ilk L adet görevi seçilecektir. Bunun yanında, herhangi bir ürünün atanmamış görev sayısı, önceden belirlenen görev sayısına (L) eşit veya küçük ($|I_{1,kalan}| \leq L$ veya $|I_{2,kalan}| \leq L$) ise her iki ürünün geri kalan tüm atanmamış görevleri son problem penceresini oluşturur.

İkinci problem penceresindeki görevlerden oluşan alt problemin en iyi çözümü [çizelge 5.4](#)'te verilmiştir. İkinci problem penceresinin aynı zamanda son problem penceresi olmasından dolayı en iyi çözümdeki tüm istasyonlar sabitlenir ($K_{sabit} = K_{pencere} = 6$).

Çizelge 5.4. PKPP algoritmasının birinci aşamasındaki ikinci problem penceresinin en iyi görev atamaları

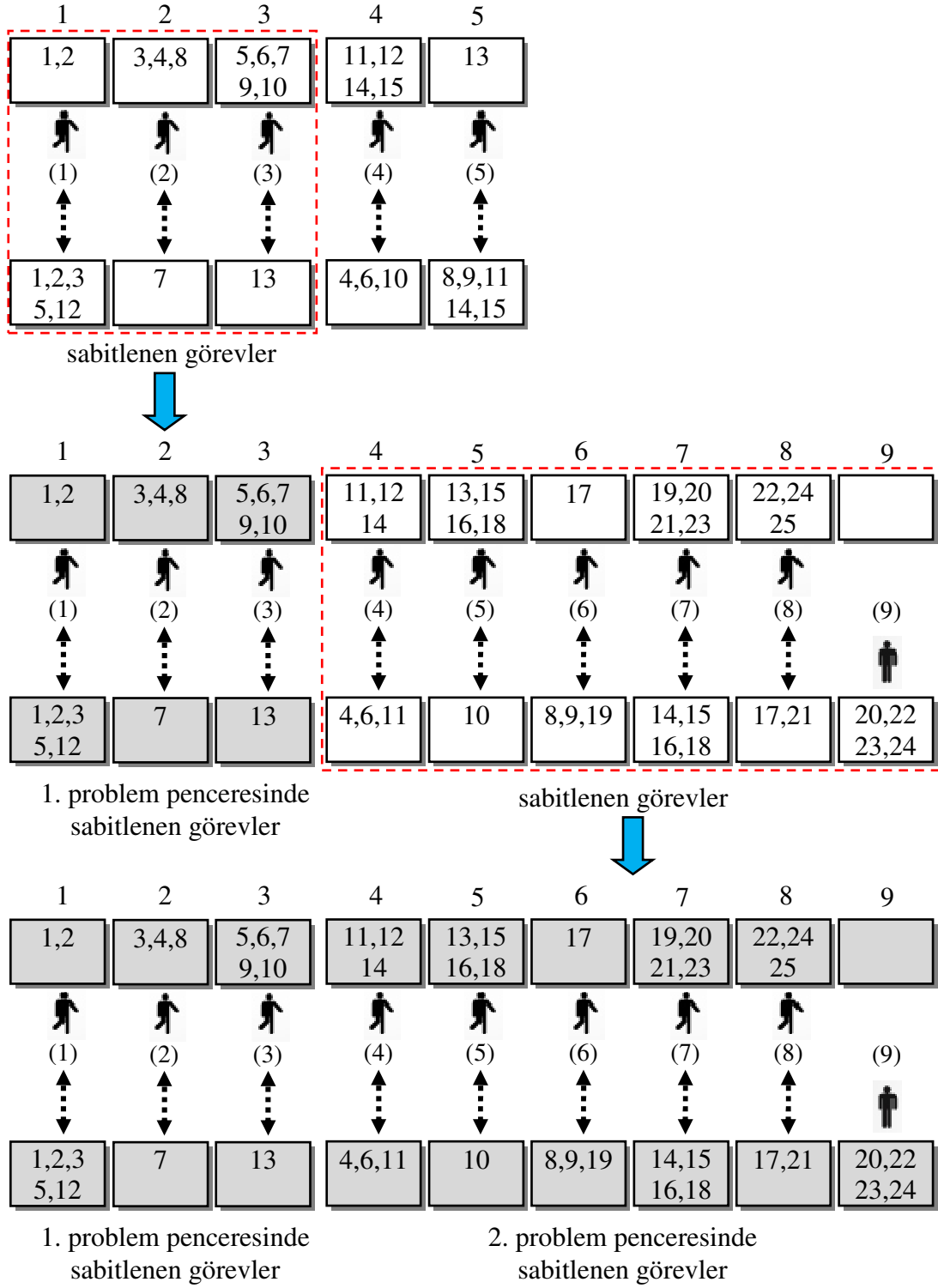
İstasyon (<i>k</i>)	Hat 1	Hat 2	İş yükü	Ücret oranı
1	{11,12,14}	{4,6,11}	28	9
2	{13,15,16,18}	{10}	25	3
3	{17}	{8,9,19}	29	10
4	{19,20,21,23}	{14,15,16,18}	29	6
5	{22,24,25}	{17,21}	30	10
6	–	{20,22,23,24}	21	7

Birinci aşamanın sonunda elde edilen en iyi çözümdeki görev numaraları, görevleri topolojik olarak sıralamak amacıyla yeniden düzenlenen numaralardır. PKPP algoritmasının ikinci aşaması örnek problemde verilen orjinal görev numaralarıyla yürütüldüğü için, görev numaraları [çizelge 5.2](#) kullanılarak eski haline dönüştürülmelidir. Örneğin; ürün 2'ye ait 9 numaralı görev pozisyon ağırlıklarına göre yeniden düzenlendiğinde görev numarası 7 yapılmıştır. Bu durumda birinci aşamanın sonunda bulunan çözümdeki 7 numaralı görev yerine 9 yazarak görev numaralarını eski haline getirebiliriz. Bu yöntemi birinci aşamanın sonunda elde edilen çözümdeki tüm görevlere uygulandığında, elde edilen görev atamaları [çizelge 5.5](#)'te verilmiştir. [Çizelge 5.5](#)'te gösterilen görev atamalarına göre, 9 operatörün kullanıldığı çözümde toplam işgücü maliyeti 1920 pb bulunmuştur. Ayrıca beşinci sütunda yer alan sayılar ise, istasyonlara atanan görevlerin hangi problem penceresinde ele alındığını göstermektedir.

Çizelge 5.5. PKPP birinci aşaması sonucunda elde edilen çözüm

İstasyon (<i>k</i>)	Hat 1	Hat 2	Ücret oranı	Pencere no
1	{1,2}	{1,2,5,6,10}	10	1
2	{3,4,8}	{9}	3	1
3	{5,6,7,9,10}	{11}	6	1
4	{11,12,13}	{3,4,13}	9	2
5	{14,15,20,21}	{8}	3	2
6	{17}	{7,14,18}	10	2
7	{19,22,23,24}	{12,15,16,19}	6	2
8	{16,18,25}	{17,22}	10	2
9	–	{20,21,23,24}	7	2
Toplam işgücü maliyeti (64 * 30 =)			1920	
Toplam operatör sayısı			9	

PKPP algoritmasının birinci aşaması şematik olarak [şekil 5.4](#)'te gösterilmiştir.



Şekil 5.4. PKPP algoritmasının birinci aşamasının genel gösterimi

PKPP algoritmasının birinci aşamasında problem pencereleri *görev tabanlı* oluşturulurken, ikinci aşamasında *istasyon tabanlı* oluşturulmaktadır. Diğer bir anlatımla, birinci aşamada problem pencerelerini belirlerken her ürünün topolojik olarak sıralanmış ilk L görev seçilirken; ikinci aşamada ise birinci aşama sonucunda elde edilen çözümdeki istasyon iş yüklerine göre, her hattaki ilk S istasyonlara atanan görevler seçilmektedir. PKPP algoritmasının birinci aşamasında sabitlenen istasyonlar ardışık olarak birbiri arkasına eklendiği için, bazı görevlerin en iyi çözümde olması gerekenden farklı istasyonlara atanması doğaldır. PKPP algoritmasının ikinci aşamasında, arka arkaya ele alınan problem pencereleri birlikte değerlendirilerek, görev atamalarındaki yanlışlıklar düzeltilmeye çalışılmıştır.

Ele alınan örnekte, PKPP algoritmasının ikinci aşamasında problem pencereleri oluşturulurken dikkate alınacak istasyon sayısı 6 olarak belirlenmiştir. Dolayısıyla, ikinci aşamanın birinci problem penceresi, birinci aşamanın sonunda elde edilen çözümdeki her hattın ilk 6 istasyonuna atanan görevlerinden oluşmaktadır. [Çizelge 5.5](#)'te gösterilen birinci aşamanın sonundaki çözüme göre, ikinci aşamanın birinci problem penceresi aşağıda verilmiştir:

$$I_{1,pencere} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 17, 20, 21\}$$

$$I_{2,pencere} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 14, 18\}$$

Bu problem penceresinin hangi görevlerden oluştuğuna bakılacak olursa; PKPP algoritmasının birinci aşamasında oluşturulan birinci ve ikinci problem pencerelerini oluşturan bazı görevlerin karması olduğu görülmektedir. Böylelikle birinci aşamada ele alınamayan problem pencereleri oluşturularak, birinci aşamanın sonunda bulunan uygun çözüm iyileştirilmeye çalışılmaktadır. İkinci aşamadaki problem pencereleri birinci aşamanın sonunda bulunan uygun çözümden oluşturulduğu için, ikinci aşamada ele alınan yeni problem penceresi çözümlerinin toplam işgücü maliyeti değerini daha da arttırması söz konusu değildir. Bu şartlar altında algoritmanın ikinci aşamasında ya aynı toplam işgücü maliyeti değerini veren bir alternatif çözüm ya da toplam işgücü maliyetinde iyileşme sağlayan bir çözüm bulunabilir. Bu iki durumda

ikinci aşamada ele alınan yeni problem pencerelerinin çözümlerinde görülmektedir. Yukarıda verilen ikinci aşamanın birinci problem penceresinin en iyi çözümü bulunarak, en iyi görev atamaları [çizelge 5.6](#)'da verilmiştir.

Çizelge 5.6. PKPP ikinci aşamasındaki birinci problem penceresinin en iyi çözümü

İst (k)	Mevcut atama			Yeni atama		
	Hat 1	Hat 2	Ücret	Hat 1	Hat 2	Ücret
1	{1,2}	{1,2,5,6,10}	10	{1,2}	{1,2,5,6,14}	10
2	{3,4,8}	{9}	3	{3,4,8}	{9}	3
3	{5,6,7,9,10}	{11}	6	{5,6,7,9,10}	{11}	6
4	{11,12,13}	{3,4,13}	9	{12,15,17}	{10,13,18}	10
5	{14,15,20,21}	{8}	3	{11,13}	{3,4,7}	9
6	{17}	{7,14,18}	10	{14,20,21}	{8}	3
Toplam işgücü maliyeti (41 * 30 =)			1230	(41 * 30 =) 1230		

[Çizelge 5.6](#)'nın sol tarafında birinci aşama sonunda bulunan çözüm; sağ tarafında ise ikinci aşamada bulunan çözüm yer almaktadır. Her iki aşamada bulunan çözüm incelendiğinde, aynı toplam işgücü maliyetine sahip iki farklı çözüm görülmektedir. Bu durumda mevcut çözümle, yeni bulunan çözüm değiştirilmemektedir. Yeni bulunan çözümün mevcut çözümle yer değiştirmesi için toplam işgücü maliyetinde iyileşme (azalma) sağlanmış olması gerekmektedir. Toplam işgücü maliyet değeri her iki çözümde de aynı olduğu için, istasyon sayısının sabit kalma şartıyla, istasyon aralığı belirlenen kayma miktarına göre istenen miktarda sağa doğru kaydırılmaktadır. Diğer bir ifadeyle, kayma miktarı 1 olarak belirlendiği için ikinci problem penceresi her hattaki [2–7] aralığındaki istasyonlara atanan görevlerden oluşmaktadır.

$$I_{1,pencere}=\{3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 17, 19, 20, 21, 22, 23, 24\}$$

$$I_{2,pencere}=\{3, 4, 7, 8, 9, 11, 12, 13, 14, 15, 16, 18, 19\}$$

İkinci problem penceresinin en iyi çözümü [çizelge 5.7](#)'de verilmiştir. Elde edilen en iyi çözüm incelendiğinde, toplam işgücü maliyetinde bir iyileşme sağlanamadığı görülmektedir.

Çizelge 5.7. PKPP ikinci aşamasındaki ikinci problem penceresinin en iyi çözümü

İst (k)	Mevcut atama			Yeni atama		
	Hat 1	Hat 2	Ücret	Hat 1	Hat 2	Ücret
2	{3,4,8}	{9}	3	{3,4,8}	{9}	3
3	{5,6,7,9,10}	{11}	6	{5,6,7,9,10}	–	6
4	{11,12,13}	{3,4,13}	9	{11,13}	{3,7,13}	9
5	{14,15,20,21}	{8}	3	{12,15,17}	{4,10,18}	10
6	{17}	{7,14,18}	10	{14,20,21}	{8,15}	3
7	{19,22,23,24}	{12,15,16,19}	6	{19,22,23,24}	{11,12,16,19}	6
Toplam işgücü maliyeti (37 * 30 =)			1110	(37 * 30 =) 1110		

Üçüncü problem penceresini [3–7] aralığındaki istasyonlara atanan görevlerden oluşturulmuştur ve problem penceresinde olan görevler aşağıdaki gibidir:

$$I_{1,pencere}=\{5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25\}$$

$$I_{2,pencere}=\{3, 4, 7, 8, 11, 12, 13, 14, 15, 16, 17, 18, 19, 22\}$$

Üçüncü problem penceresinin en iyi çözümü çizelge 5.8’de gösterilmektedir. Ele alınan bu problem penceresinde de iyileşme sağlanamadığı için birinci aşamada bulunan mevcut çözüm olduğu gibi kalmıştır.

Çizelge 5.8. PKPP ikinci aşamasındaki üçüncü problem penceresinin en iyi çözümü

İst (k)	Mevcut atama			Yeni atama		
	Hat 1	Hat 2	Ücret	Hat 1	Hat 2	Ücret
3	{5,6,7,9,10}	{11}	6	{5,6,7,9,10}	{11}	6
4	{11,12,13}	{3,4,13}	9	{11}	{3,4,7,13}	9
5	{14,15,20,21}	{8}	3	{12,13,15,17}	{10,18}	10
6	{17}	{7,14,18}	10	{14,20,21}	{8,15}	3
7	{19,22,23,24}	{12,15,16,19}	6	{19,22,23,24}	{12,16,19}	6
8	{16,18,25}	{17,22}	10	{16,18,25}	{17,22}	10
Toplam işgücü maliyeti (44 * 30 =)			1320	(44 * 30 =) 1320		

Her hattaki [4-9] aralığındaki istasyonlara atanan görevler aşağıda verilmiştir:

$$I_{1,pencere}=\{11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25\}$$

$$I_{2,pencere}=\{3, 4, 7, 8, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24\}$$

Bu görevlerden oluşmuş dördüncü problem penceresinin en iyi çözümü [çizelge 5.9](#)'da verilmiştir. Birinci aşamada bulunan görev atamalarına göre toplam işgücü maliyeti 1350 pb iken, yeni bulunan görev atamalarına göre 1320 pb'dir. Toplam işgücü maliyetinde 30 pb tasarruf sağlayacak yeni atama mevcut atamayla değiştirilir.

Çizelge 5.9. PKPP ikinci aşamasındaki dördüncü problem penceresinin en iyi çözümü

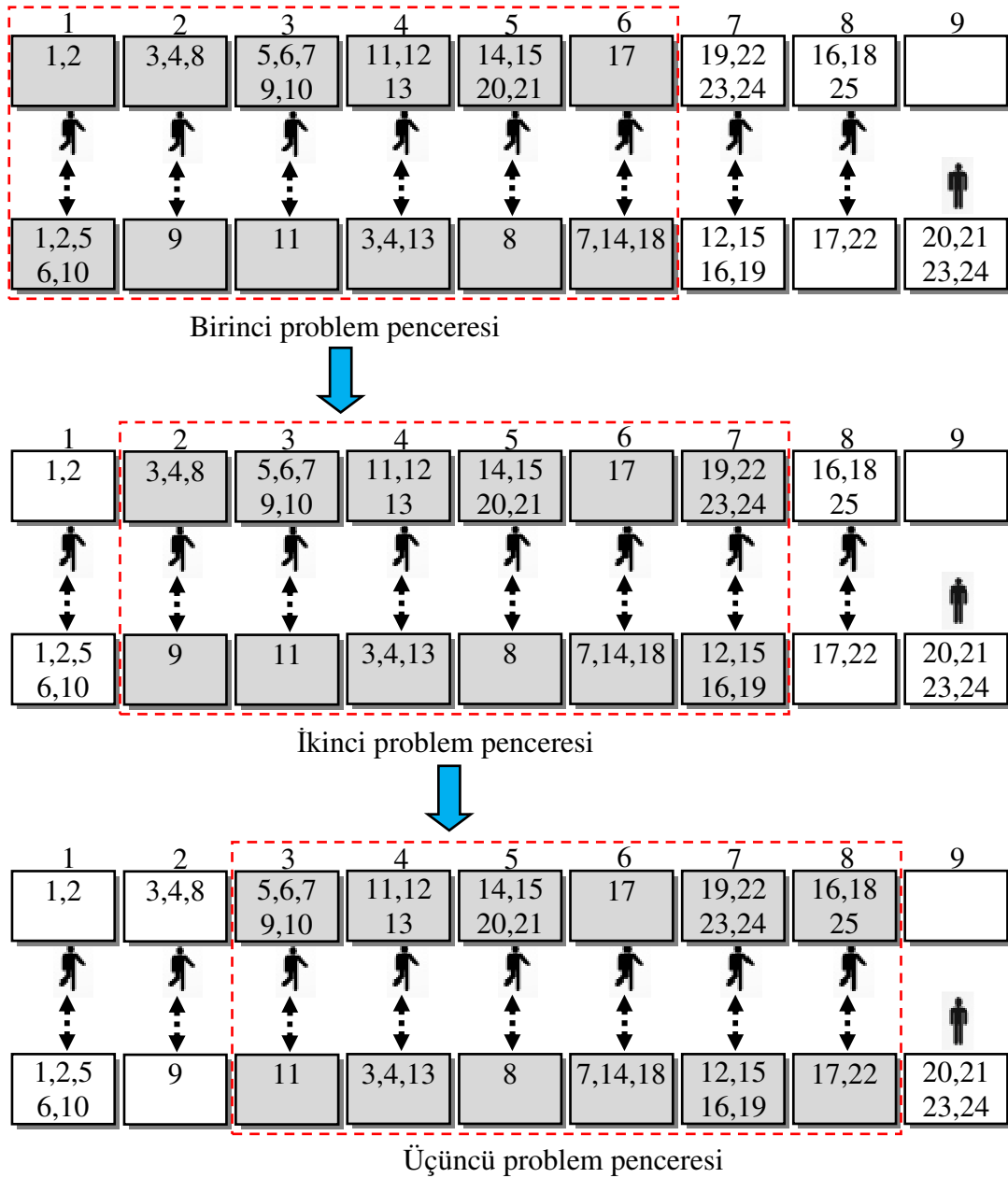
İst (<i>k</i>)	Mevcut atama			Yeni atama		
	Hat 1	Hat 2	Ücret	Hat 1	Hat 2	Ücret
4	{11,12,13}	{3,4,13}	9	{11,12,13}	{3,4,13}	9
5	{14,15,20,21}	{8}	3	{14,15,20,21}	{8}	3
6	{17}	{7,14,18}	10	{16,17}	{7,10,18}	10
7	{19,22,23,24}	{12,15,16,19}	6	{19,22,23,24}	{12,15,16,19}	6
8	{16,18,25}	{17,22}	10	{18,25}	{17,22,23}	10
9	–	{20,21,23,24}	7	–	{20,21,24}	6
Toplam işgücü maliyeti (45 * 30 =)			1350	(44 * 30 =) 1320		

İkinci aşamanın son problem penceresinin en iyi çözümündeki görev atamaları, birinci aşama sonunda bulunan çözümdeki [4–9] istasyonlarına atanan görevlerle değiştirilmiştir. İkinci aşamanın en sonunda elde edilen çözüm [çizelge 5.10](#)'da özetlenmiştir. Bu çözüm aynı zamanda asıl problemin en iyi çözümüdür.

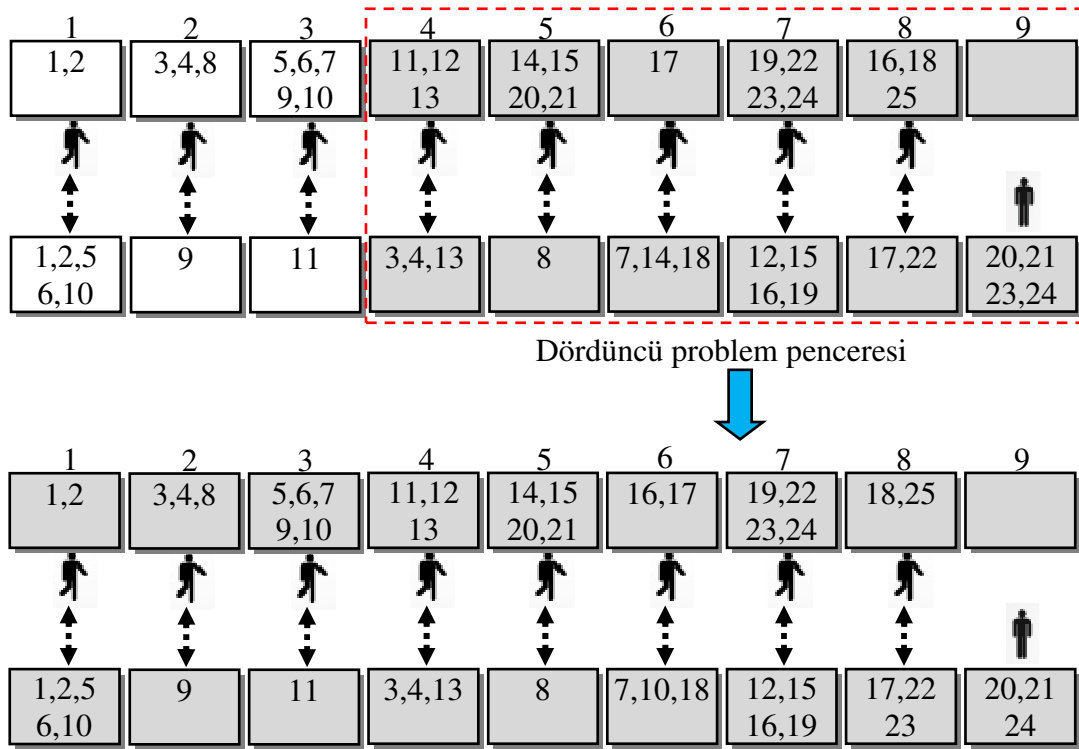
Çizelge 5.10. PKPP ikinci aşaması sonucunda elde edilen çözüm

İstasyon (k)	Hat 1	Hat 2	Ücret oranı
1	{1,2}	{1,2,5,6,10}	10
2	{3,4,8}	{9}	3
3	{5,6,7,9,10}	{11}	6
4	{11,12,13}	{3,4,13}	9
5	{14,15,20,21}	{8}	3
6	{16,17}	{7,10,18}	10
7	{19,22,23,24}	{12,15,16,19}	6
8	{18,25}	{17,22,23}	10
9	–	{20,21,24}	6
Toplam işgücü maliyeti (63 * 30 =)			1890
Toplam operatör sayısı			9

Bu bölümde, ele alınan M-PMHD probleminin en iyi çözümü IBM ILOG CPLEX 12.4 yazılımı ile 980 sn'de bulunurken, aynı problemin en iyi çözümü PKPP algoritmasıyla alt problemlere ayırarak çözüldüğünde 47 sn'de bulunmuştur. Önerilen PKPP algoritması her zaman en iyi değeri garanti etmez, fakat büyük boyutlu problemlerin yüksek kalitede çözümlerini kısa sürelerde elde edebilmektedir. PKPP algoritmasının ikinci aşaması şematik olarak [şekil 5.5](#)'te gösterilmiştir.



Şekil 5.5. PKPP algoritmasının ikinci aşamasının genel gösterimi



Şekil 5.5. (Devam) PKPP algoritmasının ikinci aşamasının genel gösterimi

5.2. Geliştirilen Melez Çok Popülasyonlu Paralel Genetik Algoritma

5.2.1. Genetik algoritmalar

Genetik algoritma (GA) doğadaki biyolojik evrimin işleyiş biçimini taklit eden, doğal seleksiyon mekanizmasına dayanan olasılıklı bir sezgisel arama yöntemidir. Diğer bir tanımla; GA, Darwin'in *en iyi olan yaşar* (*survival of the fittest*) prensibine dayalı evrim ve genetik teorilerinden esinlenen evrimsel bir algoritmadır. GA, arama amaçlı kullanılmaya başlanmadan önce, bir çok biyoloğun biyolojik sistemlerin simülasyonu için bilgisayar kullanmaya başladığı 1950'li yılların başlarına dayanmaktadır [Michalewicz, 1999]. Bu çalışmalara Barricelli (1957, 1963), Fraser (1957), Martin ve Cockerham (1960) örnek olarak verilebilir. Özellikle Frase'in çalışması modern genetik algoritma yapısına yaklaşmıştır.

Evrimsel algoritmalar arama tekniklerine rassallık katarak popülasyonların doğal evrim süreçlerini taklit etmektedir. Evrimsel algoritmalar ilk olarak Fogel (1966) tarafından *evrimsel programlama*; Rechenberg (1971) ve Schwefel (1974) tarafından *evrimsel stratejiler* olarak ortaya atılmıştır. *Genetik algoritma* ifadesi ilk defa Bagley (1967) çalışmasında kullanılmıştır. Holland (1975) çalışmasında yeni çocuklar oluşturmak üzere ebeveynlerden gelen bilgi parçacıklarını farklı kombinasyonlarda birleştirip *genetik algoritma* tekniğini geliştirmiştir. Çalışmalarının sonucunu açıkladığı “*Adaptation in Natural and Artificial Systems*” adındaki kitabının 1975’te yayınlanmasından sonra geliştirdiği yöntemin adı *Genetik Algoritmalar* olarak anılmaya başlanmıştır [Goldberg, 1989]. Holland’ın öğrencisi olan David E. Goldberg, 1985 yılında bitirdiği doktora çalışması ve 1989’da yayınladığı “*Genetic Algorithms in Search, Optimization, and Machine Learning*” adındaki kitabı ile genetik algoritmanın pratikteki yararlarına dikkat çekmiştir. Bu çalışmaların ardından GA çalışmaları hızlı bir artış göstermiştir. GA kullanarak çeşitli problemlere çözüm üretmesini sağlayan programların geliştirilmesine John Koza tarafından *Genetik Programlama* adı verilmiştir [Koza, 1992].

Darwin’in evrim teorisinden esinlenilerek oluşturulan GA, yapay zekanın giderek daha da önem kazanan bir kolu olup, evrimsel hesaplama işlemlerine dayanmaktadır. GA evrim sürecinden etkilenecek, canlılarda yaşanan genetik sürecin bilgisayar ortamında gerçekleştirilmesi işlemidir. Herhangi bir problemin GA ile çözümü, problemi sanal olarak evrimden geçirmek suretiyle yapılmaktadır. Geleneksel optimizasyon yöntemlerine göre farklılıkları olan genetik algoritmalar, parametre kümesini değil, kodlanmış biçimlerini kullanırlar. Olasılık kurallarına göre çalışan genetik algoritmalar, yalnızca amaç fonksiyonuna gereksinim duyarlar. Çözüm uzayının tamamını değil, belirli bir kısmını tararlar. Arama uzayında aynı anda birçok noktada çalıştırıldığı için küresel çözüme ulaşma olasılığı fazladır. Böylece, etkin arama yaparak çok daha kısa bir sürede çözüme ulaşırlar [Goldberg, 1989].

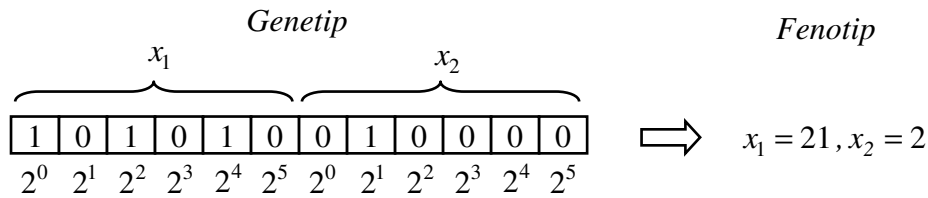
Diğer bir önemli üstünlükleri ise, çözümlerden oluşan popülasyonu eş zamanlı incelemeleri ve böylelikle yerel en iyi çözümlere takılmamalarıdır. Doğadaki evrim sürecinde yer alan bir çok mekanizma, genetik algoritmalarda probleme ait bir grup

çözümün evrimleştirilmesi ve böylece en iyi veya en iyiye yakın çözümlerin elde edilmesi için kullanılmaktadır. GA, sezgisel bir yöntem olduğundan dolayı verilen bir problem için en iyi çözümü bulamayabilir, ancak bilinen yöntemlerle çözülemeyen veya çözüm zamanı problemin büyüklüğü ile üstel artan problemlerde en iyi değere çok yakın sonuçlar vermektedir. Başlangıçta doğrusal olmayan optimizasyon problemlerine uygulanan GA, sonraları gezgin satıcı, karesel atama, yerleşim, atölye tipi çizelgeleme, ders/sınav programı hazırlanması gibi kombinatorial optimizasyon problemlerine başarıyla uygulanmıştır [Liepins ve ark., 1990].

Genetik algoritmalar doğayı taklit etme düşüncesi ile geliştirildiklerinden, genetik algoritmalarda kullanılan terminoloji büyük ölçüde genetik biliminin terminolojisi ile ilişkilidir. Bu nedenle genetik algoritmalara daha fazla değinmeden önce genetik bilimi ile ilgili bazı terimlerden söz etmekte fayda vardır. Bütün yaşayan canlılar hücrelerden oluşurlar ve her bir hücre bir veya daha fazla *kromozomdan* oluşur. Kromozomlar ise belirli sayıda *genden* oluşurlar. Genler DNA'nın fonksiyonel yapılarıdır ve her biri belirli bir proteini kodlar. Yani, genler göz rengi gibi belirli özellikleri kodlayan yapılardır. Bir özelliği oluşturabilecek mümkün farklı yapılar *allel* adını alır. Diğer bir ifadeyle, her bir gen mümkün farklı allellerden birini içerir. Bir genin alleli o genin almış olduğu değeri göstermektedir. Bir özelliğin kromozom tarafından kodlanmış haline *genotip*, bu özelliğe ise *fenotip* denilmektedir. Canlıların üremeleri sonucunda ortaya çıkan yavruların kromozomları ebeveynlerinin kromozomlarının bir karmasıdır. Dolayısıyla yavruların kromozomlarının bir kısmı anneden bir kısmı da babadan gelmektedir. Bu işleme *çaprazlama* adı verilir. Ayrıca evrim teorisine göre doğada meydana gelen bazı yıkıcı etkiler veya genetik süreçteki çeşitli hatalar nedeniyle kromozom üzerinde yer alan bazı genlerin allelleri değişmektedir. Genlerdeki bu değişim ise *mutasyon* olarak adlandırılır [Koç, 2007].

Genetik algoritmalarda kullanılan terminoloji genetik biliminin terminolojisi ile bağdaştırılarak açıklanabilir. Genetik algoritmalar kromozomların oluşturduğu bir *popülasyon* üzerinde çalışmaktadır. Bir *kromozom*, ele alınan probleme ait karar değişkenlerinin değerlerinin *genetik gösterim* ismi verilen özel bir yapı üzerinde

kodlanması ile oluşturulmuş bir vektördür. Dolayısıyla her bir kromozom probleme ilişkin bir çözümü kodlamaktadır. Karar değişkenlerinin kodlanması ile oluşan kromozoma *genotip*, kromozomun kodladığı karar değişkeni değerlerine *fenotip* denilmektedir. Kromozom üzerinde yer alan çeşitli *gen* dizileri problemin karar değişkenlerini kodlar ve her bir genin almış olduğu değerlere *allel* denir. İki adet karar değişkenine (x_1 ve x_2) sahip bir problem için ikili genetik gösterim için oluşturulmuş örnek bir kromozom Şekil 5.6’da verilmiştir. Örneğin, şekil 5.6’da x_1 ve x_2 karar değişkenleri altışar adet gen kullanılarak kodlanmıştır. x_1 karar değişkenini kodlayan ilk genin alleli “1”, ikinci genin alleli “0” dır.



Şekil 5.6. İkili genetik gösterim için örnek bir kromozom

Popülasyon çözüm uzayındaki belirli sayıda noktanın oluşturduğu çözüm kümesidir ve belirli sayıda kromozomdan oluşur. Bir başka ifadeyle, popülasyon ele alınan problemin çeşitli çözümlerinin kodlandığı bir küme olarak tanımlanabilir. GA’nın başında bu çözümler rassal olarak belirlenmektedir ve genellikle bu kümenin büyüklüğü algoritma boyunca sabit tutulmaktadır. Algoritmanın başında belirlenmesi gereken en önemli nokta bu popülasyonun büyüklüğünün ne kadar olacağıdır. Söz konusu çözüm kümesinin büyüklüğü algoritmanın etkili ve verimli bir şekilde çalışmasına yönelik olarak belirlenmelidir. Küçük boyutta popülasyon GA’nın aynı anda çözüm uzayının farklı yerlerini araması özelliğini etkin kılmayacak, çok büyük olması ise algoritmanın hızlı çalışmasını engelleyerek verimliliğini azaltacaktır [Reeves, 2003]. Dört kromozomdan oluşan örnek bir popülasyon şekil 5.7’de gösterilmiştir.

Her bir kromozom kodladığı çözümün kalitesini gösteren bir uygunluk değerine sahiptir. Problemin amaç fonksiyonunun $enbf(x_1, x_2) = x_1 + x_2$ olduğu varsayılacak

olursa, kromozomların kodladığı karar değişkenlerinin değerlerinin amaç fonksiyonu değeri kromozomların uygunluk değerleri olarak düşünülebilir. Her kromozomun uygunluk değeri [şekil 5.7](#)'de gösterilmiştir.

x_1	x_2	$enbf(x_1, x_2) = x_1 + x_2$
1 0 1 0 1 0 0 1 0 0 0 0	21 2	23
0 0 0 0 0 0 0 0 0 0 0 0	0 0	0
1 1 1 1 1 1 0 1 0 0 0 0	63 2	65
1 0 1 1 1 0 0 1 1 1 0 1	29 46	75

Şekil 5.7. Örnek bir popülasyon

Genetik algoritmaların temel çalışma mantığı, tıpkı evrim sürecinde olduğu gibi uygunluk değerleri daha iyi olan kromozomların çoğalarak artması üzerine kuruludur. Böylece daha iyi uygunluk değerine sahip kromozomlar çoğalarak aramaya yön verecektir. Genetik algoritmaların her bir iterasyonuna *jenerasyon* denilmektedir. Her bir jenerasyonda, *genetik operatör* olarak isimlendirilen çeşitli prosedürler popülasyonda yer alan kromozomlar üzerinde uygulanır. Genetik operatörler kullanılarak yeni çözümleri kodlayan kromozomlardan oluşan yeni bir popülasyon elde edilmektedir. Her bir genetik operatör doğada yer alan evrim sürecinin farklı bir mekanizmasını kromozomlar üzerinde taklit etmektedir. Genetik algoritmalarda üç genetik operatör kullanılmaktadır: *seçim*, *çaprazlama* ve *mutasyon*.

Seçim operatörünün görevi, bir önceki jenerasyonun popülasyonunda seçim yaparak yeni bir popülasyon oluşturmaktır. Kromozomların seçilme olasılıkları uygunluk değerleri ile orantılıdır. Bu nedenle uygunluk değeri yüksek olan kromozomların yeni popülasyondaki beklenen sayıları daha çok iken, uygunluk değeri düşük olanların beklenen sayıları daha azdır.

Çaprazlama operatörü seçim operatörüyle seçilmiş yeni popülasyon üzerinde çalışmaktadır. Çaprazlama operatörünün görevi, kromozomlar arasında genetik bilgi

paylaşımı yaparak, farklı genetik bilgilerin bir araya getirilmesini sağlamaktır. Çaprazlama operatörü genellikle popülasyonda yer alan iki kromozomun ebeveyn olarak seçilmesi, bu iki kromozoma çaprazlama işleminin uygulanması ve iki yeni yavru kromozomun elde edilmesi yoluyla çalışır. Örneğin, [şekil 5.7](#)'de verilen popülasyonda yer alan üçüncü ve dördüncü kromozomlar ebeveyn olarak seçildiğinde, çaprazlama (tek nokta çaprazlama; Holland, 1975) işleminin uygulanışı ve çaprazlama sonrası elde edilen iki yeni yavru kromozom [şekil 5.8](#)'de gösterilmiştir.

	x_1	x_2	x_1	x_2	$enbf(x_1, x_2)$
Ebeveyn 1:	1 0 1 1 1 0 0 1	1 1 0 1	29	46	75
Ebeveyn 2:	1 1 1 1 1 1 0 1	0 0 0 0	63	2	65
Yavru 1:	1 0 1 1 1 0 0 1	0 0 0 0	29	2	31
Yavru 2:	1 1 1 1 1 1 0 1	1 1 1 0 1	63	46	109

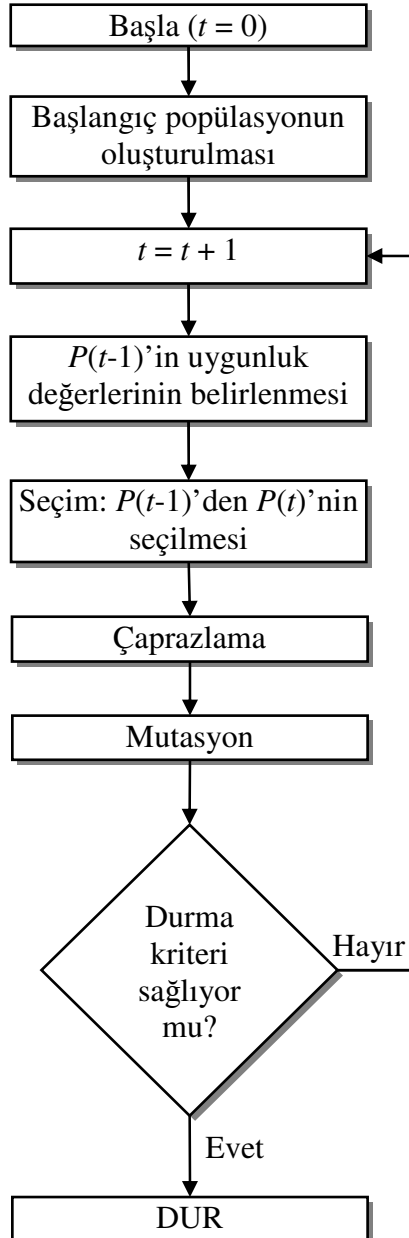
Şekil 5.8. Örnek bir çaprazlama işlemi (tek nokta çaprazlama)

Mutasyon operatörü ise kromozomlar üzerinde çeşitli rassal değişiklikler yapmak yoluyla farklı çözümleri kodlayan kromozomlar oluşturmaktadır. Mutasyon operatörünün görevi genlerin allellerini rassal olarak değiştirmek yoluyla popülasyondaki genetik bilgi zenginliğini arttırmak ve GA'nın yerel en iyi çözümlere takılmasını engellemektir. Örneğin [şekil 5.8](#)'de elde edilen ikinci yavru kromozomun yedinci geni mutasyon için seçildiğinde mutasyon işlemi sonrası elde edilen yeni kromozom [şekil 5.9](#)'da verilmiştir.

Mutasyon öncesi:	1 1 1 1 1 1 0 1 1 1 1 0 1
Mutasyon sonrası:	1 1 1 1 1 1 1 1 1 1 1 0 1

Şekil 5.9. Örnek bir mutasyon işlemi

Genetik algoritmalar seçim, çaprazlama ve mutasyon operatörlerini bir popülasyon üzerinde kullanarak, her jenerasyonda yeni bir popülasyon elde etmek yoluyla bir problemin en iyi çözümünü ararlar. Bir t jenerasyonundaki popülasyon $P(t)$ ile gösterilmek üzere genetik algoritmaların akış şeması [şekil 5.10](#)'da verilmiştir.



Şekil 5.10. Genetik algoritmanın akış şeması

Genetik algoritmaların yeni kromozomlar elde etmek için kullandığı genetik operatörler, çaprazlama ve mutasyondur operatörleridir. Her iki operatör kullandıkları yönteme bağlı olarak kromozomun belirli bir komşuluğundaki farklı kromozomları elde edebilme becerisine sahiptir. Bir genetik gösterim aracılığıyla oluşturulabilir tüm kromozomlar kromozom uzayını oluşturmak üzere, her iki operatör popülasyonda yer alan kromozomların belirli bir komşuluğunda örnekleme yapar. Seçim operatörü ise bir önceki jenerasyonun popülasyonunda seçim yaparak yeni bir popülasyon oluşturmaktadır. Çaprazlama ve mutasyon operatörleri bu yeni popülasyon üzerinde çalışmaktadır. Dolayısıyla, seçim operatörü GA'nın kromozom uzayının hangi bölgelerinde örnekleme yapacağını belirler, çaprazlama ve mutasyon operatörleri ise bu bölgelerde örnekleme yapar. Bu nedenle seçim, çaprazlama ve mutasyon operatörleri bir GA'nın kromozom uzayını örnekleme yöntemini oluştururlar.

Kromozom uzayında yer alan bir K kromozomu problemin çözüm uzayında yer alan bir $\mathcal{C}$ çözümünü kodlamaktadır. Ancak K kromozomunun kromozom uzayında yakın komşuluğunda yer alan kromozomlar, çözüm uzayında $\mathcal{C}$ çözümünün yakın komşuluğunda yer almak zorunda değildir. Çaprazlama ve mutasyon operatörleri popülasyondaki kromozomların belirli bir komşuluğunda örnekleme yaparken, aslında çözüm uzayının bambaşka bölgelerinde dolaylı bir örnekleme yapmaktadırlar. Bu nedenle, genetik algoritmalar çözüm uzayındaki yerel en iyi çözümlerden kurtulabilme becerisine sahiptir. Bu özellik genetik algoritmaların *gürbüz* (*robust*) teknikler olmasını sağlar.

Bir GA'nın kromozom uzayında ve dolaylı olarak çözüm uzayında örnekleme yapabilme becerisi popülasyonda yer alan kromozomların çeşitliliği ile sınırlı olduğundan, popülasyondaki kromozomlar ne kadar çeşitli olursa GA'nın kromozom uzayının farklı bölgelerini örnekleyebilme kabiliyeti de o kadar fazla olacaktır. Buna karşın seçim operatörü bir önceki popülasyonda bir seçim yaparak, uygunluk değeri yüksek olan kromozomların yeni popülasyona artan sayıda geçmesini sağlamaktadır. Bunun anlamı önceki popülasyonda yer alan bazı kromozomların yeni popülasyonda yer alamaması ve bazılarının yeni popülasyonda birden fazla sayıda yer almasıdır.

Dolayısıyla seçim operatörünün uygunluk değeri yüksek kromozomlara tanıdığı öncelik ne kadar artarsa, yeni popülasyonda kromozom çeşitliliği o kadar düşecektir. Bu durum GA'nın kromozom uzayının farklı bölgelerinde örnekleme yapabilme kabiliyetini kısıtlayacaktır. Bu nedenle uygunluk değeri yüksek olan kromozomlara tanınan ayrıcalığın derecesi ile popülasyondaki kromozomların çeşitliliği arasında uygun bir denge kurmak gerekmektedir. Bu durum kaynaklarda farklı terimlerle ele alınmaktadır. Bazı çalışmalar arama uzayının keşfedilmesi ile elde edilen kaliteli çözümlerin aramaya yön vermesi arasındaki dengeden söz ederken [Holland, 1975, Goldberg, 1999], bazı çalışmalar *popülasyon çeşitliliği* (*population diversity*) ve *seçim baskısı* (*selective pressure*) arasındaki dengeden söz etmektedir [Whitley, 1989, Back, 1995, Miller ve Goldberg, 1996, Thierens, 1997]. Arama uzayının keşfedilmesi ile kaliteli çözümlerin aramaya yön vermesi arasındaki denge bir çok sezgisel teknik için (örneğin: tavlama benzetimi, tabu arama, karınca kolonileri) kurulması gereken bir dengedir. Genetik algoritmalar da sezgisel algoritmalar sınıfında yer aldığı için kaynaklarda yer alan çalışmaların bir kısmının bu bakış açısıyla yaklaşıyor olması doğaldır. Ancak genetik algoritmaları diğer sezgisel yöntemlerden ayıran en önemli özellik popülasyon tabanlı olmalarıdır. Bu nedenle genetik algoritmalar için *popülasyon çeşitliliği* ve *seçim baskısı* daha uygun terimlerdir.

Popülasyon çeşitliliği ile seçim baskısı arasındaki denge, genetik algoritmaların temel çalışma ilkelerini açıklayan *şema teoremi* [Holland, 1975] ve *yapı taşı hipotezi* [Goldberg, 1989] ile daha fazla anlam kazanmaktadır. Şema kromozomlar arasındaki benzerlikleri tarifleyen özel bir yapıdır. *Yapı taşı hipotezi* ise, bu özel şemalara yapı taşı ismini vermekte ve GA'nın yapı taşlarını birleştirmek yoluyla problemin en iyi çözümünü aradığını ileri sürmektedir. Yapı taşlarını birleştirmek bir GA'nın başarısının arkasında yatan en önemli unsurdur. Yapı taşlarının önemini daha iyi anlatmak için yapı taşları şöyle tanımlanabilir: Uygunluk değeri yüksek olan bazı kromozomların, diğer kromozomlara göre daha yüksek uygunluk değerine sahip olmalarını sağlayan bazı özel genlerinin almış olduğu değerler bir yapı taşıdır. Bu nedenle bir GA'nın performansı farklı kromozomlarda yer alan farklı yapı taşlarını birleştirebilme kabiliyetine bağlıdır [Goldberg, 1989].

Eğer popülasyondaki kromozomlar farklı yapı taşlarına sahip olmazlarsa, bunların birleştirilmesi de söz konusu olamaz. Bu nedenle popülasyon çeşitliliğini korumak önemlidir. Bunun yanı sıra, eğer uygunluk değeri yüksek olan kromozomlara ayrıcalık tanınmazsa, bu durumda da yapı taşlarının ilerleyen jenerasyonlarda çoğalarak artması (şema teoremi) söz konusu olamaz. Bu nedenle seçim baskısının var olması da önemlidir. Bu iki sebepten dolayı popülasyonlar arası çeşitlilik ile seçim baskısı arasında uygun bir denge kurabilmek bir GA'nın performansı üzerinde etkilidir. Özetle, bir GA'nın performansının yüksek olabilmesi için şu dört görevi başarılı bir şekilde yerine getirmesi gerekir: (i) farklı yapı taşlarını üretebilmek, (ii) popülasyondaki yapı taşı çeşitliliğini yüksek tutabilmek (*popülasyon çeşitliliği*), (iii) Yapı taşlarının çoğalarak artmasını sağlayabilmek (*şema teoremi, seçim baskısı*), (iv) Farklı yapı taşlarını birleştirerek daha iyi çözümlerin oluşturulmasını sağlayabilmek (*yapı taşı hipotezi*).

Bunu başarmanın ise üç yolu vardır [[Beasley ve ark., 1993](#)]: Uygun çaprazlama operatörleri geliştirmek, uygun genetik gösterimler kullanmak veya genler arasındaki ilişkiyi öğrenerek GA'nın aramasına yön vermek.

5.2.2. Çok popülasyonlu paralel genetik algoritmalar

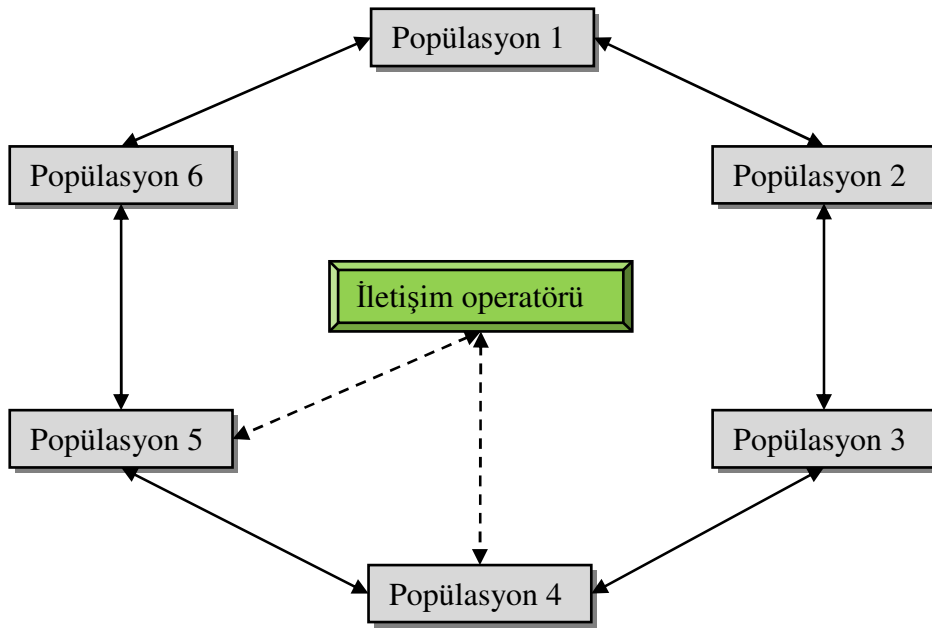
Paralel genetik algoritma çalışmalarının çoğundaki düşünce, yapılması gereken çok sayıda işi daha küçük işler kümelerine bölerek ve bu iş kümelerinin her birini bir işlemciye atayarak eş zamanlı çözümlerini sağlamaktır. [Cantu-Paz \(2000\)](#) bu bakış açısı ile birden fazla işlemci üzerinde paralelleştirilen GA yaklaşımlarını *paralel genetik algoritmalar* (PGA) başlığı altında toplamıştır. Paralel genetik algoritmalar dört ana sınıfta incelenmektedir [[Cantu-Paz, 2000](#)]:

- Yönetici-köle PGA,
- Çok popülasyonlu PGA,
- İnce-taneli PGA,
- Hiyerarşik melez PGA

Bazı paralel genetik algoritma yaklaşımları genetik algoritmanın birden çok işlemci üzerinde hızlandırılmasına dayanırken, bazıları da genetik algoritmanın çalışma mekanizmasında önemli değişiklikler yapmaktadır. Çok popülasyonlu PGA, GA'nın çalışma mekanizması üzerinde değişiklik yapan yaklaşımlardan birisidir. Bu tez kapsamında, paralel genetik algoritmalarla ilgilenilmesinin sebebi, yapılacak olan işlerin birden fazla işlemci üzerinde paralelleştirilmesinden ziyade, çok popülasyonlu PGA'nın, GA'nın çalışma yöntemine getirdiği faydalı yeniliklerdir. Bu nedenle, diğer paralel genetik algoritmalara yer verilmemiştir. Diğer paralel genetik algoritmalar hakkında detaylı bilgi için şu çalışmalara bakılabilir: [Cantu-Paz, 2000, Koç, 2007].

Çok popülasyonlu PGA'nın diğer genetik algoritma yaklaşımlarından farklı iki önemli bileşene sahiptir: *birden fazla popülasyon* ve *popülasyonlar arası bilgi alışverişine izin veren bir iletişim operatörü*. Çok popülasyonlu genetik algoritmalarda temel mantık, bir tek büyük popülasyon üzerinde çalışmak yerine boyutları küçük olan çok sayıda popülasyon üzerinde çalışmaktır. Her bir popülasyonda bir genetik algoritma çalışmaktadır. GA'nın her bir popülasyon için kromozom uzayının farklı bölgelerini örnekleme (keşfetmesi) beklenir.

Popülasyonlar birbirinden tamamen izole olduğunda, her bir popülasyonun farklı bir yerel en iyi çözüme yakınsama ve buradan kurtulamaması riski vardır. Bu nedenle popülasyonlar arasında belirli sıklıkla iletişim gerçekleştirilmelidir. Popülasyonlar arası iletişimi gerçekleştirmek için kaynaklarda yaygın olarak kullanılan *iletişim operatörü*, topoloji adı verilen ve hangi popülasyonların birbirleriyle iletişim kurabileceklerini tarifleyen bir yapıdan faydalanmaktadır. Bir iletişim operatörü, topolojiye uygun iki popülasyon arasında bilgi alışverişi sağlamaktadır. Belirli sıklıklarla iletişime izin verilen popülasyonlar arasında bir veya daha çok sayıda kromozom aktarımı yapılmaktadır. İletişim operatörüyle popülasyonlar arasında bilgi alışverişini sağlayarak, farklı popülasyonlardan elde edilen bilgilerin birleştirilmesi yoluyla daha kaliteli çözümleri kodlayan kromozomların elde edilmesi sağlanmaktadır. Çok popülasyonlu PGA'nın genel yapısı [şekil 5.11](#)'de gösterilmiştir.



Şekil 5.11. Çok popülasyonlu paralel genetik algoritmaların genel yapısı

Çok popülasyonlu PGA, *ada modeli genetik algoritmalar* olarak da bilinmektedir. Çünkü ada modeli birbirinden izole bir şekilde yaşayan ve ara sıra kendi aralarında iletişimin olduğu çok sayıdaki popülasyonu tariflemektedir. Çok popülasyonlu PGA bu bakış açısı ile genetik algoritmaların esinlenme kaynağı olan genetik bilimine daha çok uymaktadır. Nitekim, Elredge ve Gould çalışmasındaki *punctuated equilibria* teorisi çok popülasyonlu PGA ile benzerlik taşımaktadır. Bu teori, uzun süreler bir popülasyonda önemli değişiklikler olmasa da (popülasyon dengede iken), bazı olayların popülasyonda hızlı değişimlere sebep olabileceğini öne sürmektedir [Elredge ve Gould, 1972]. Cohoon ve ark. bu teorinin çok popülasyonlu PGAlarda popülasyonlar arası iletişimin popülasyonlar üzerindeki etkisi ile benzerlik taşıdığını göstermiştir [Cohoon ve ark., 1987, 1991].

Bossert, kaynaklarda çok popülasyonlu evrimsel algoritmalarla ilişkin bulunabilen ilk çalışmadır. Bossert'in çalışmasında popülasyonlar yaşamlarına devam etmek için birbirleri ile rekabet etmektedirler. Bossert, en kötü popülasyonu rassal olarak belirlenen periyotlarla yok etmiş, bunun yerine yeni bir popülasyonun rassal olarak oluşturulması yoluna gitmiştir [Bossert, 1967]. Çok popülasyonlu paralel genetik algoritmalarla ilişkin bulunan ilk çalışma ise Grefenstette'ye aittir. [Grefenstette,

1981]. Grefenstette, popülasyonlar arası bilgi alışverişini sağlamak için her bir popülasyondaki en iyi kromozomu her jenerasyonda diğer tüm popülasyonlara aktarma yoluna gitmiştir. Bu yaklaşımda popülasyonlar arasında oldukça yüksek bir iletişim miktarı söz konusudur ve popülasyonların hızla birbirlerine yakınsaması kaçınılmazdır. Nitekim, Grefenstette bunun bir sonucu olarak iletişim sıklığı, aktarılan kromozomların hangi yönde aktarılabilceği (topoloji) ve iletişim operatörünün erken yakınsama üzerindeki etkisi gibi konular üzerinde çalışılması gerektiğini vurgulamıştır [Grefenstette, 1981].

Grosso, alt popülasyonlara bölünmüş bir yaklaşım kullanarak uygunluk değerlerindeki artış oranının tek popülasyonlu yapıya göre daha yüksek olduğunu gözlemlemiştir. Aynı çalışmada, çok popülasyonlu yapıda popülasyonların birbirlerinden tamamen izole edilmesi durumunda elde edilen çözümlerin, toplamda aynı kromozom sayısına sahip tek popülasyonlu yapıda elde edilen çözümlerden daha kötü olduğu belirtilmiştir. Bu sonuç beklenen bir sonuçtur. Çünkü, çok popülasyonlu yapıda küçük alt popülasyonların her birinin popülasyon çeşitliliği büyük bir tek popülasyonlu GA'ya göre daha düşük olacaktır. Bunun anlamı, her bir küçük popülasyonun yerel en iyi çözümlere yakınsaması olasılığının daha fazla olmasıdır. Aralarında bilgi alışverişine izin verilmeden çalışan M adet küçük popülasyonlu bir GA'nın ortalama performansının bir tek küçük popülasyona sahip bir GA'nın M kere çalıştırılmasından elde edilecek ortalama performansa eşit olması beklenir. Dolayısıyla popülasyon çeşitliliği daha yüksek olan bir tek büyük popülasyonlu bir GA'nın elde edeceği çözüm kalitesinin, birbirinden tamamen izole çalışan M adet küçük popülasyonun elde edeceği çözüm kalitesinden daha yüksek olması beklenen bir durumdur [Grosso, 1985].

Grosso, iletişim sıklığı parametresinin çeşitli düzeylerini incelemiş ve iletişim sıklığı parametresinin ortalama düzeyleri için bir tek büyük popülasyona sahip tek popülasyonlu yapı ve birden fazla küçük popülasyona sahip çok popülasyonlu yapı kullanılarak elde edilen çözümlerin aynı kalitede olduğunu göstermiştir. Grosso, yaptığı deneylerin ilk kısmında her jenerasyonda iletişime izin vermiş ve hem göç edecek kromozomları hem de hangi popülasyona göç edeceklerini rassal olarak

belirlemiştir. Grosso deneylerinin ikinci kısmında, *geciktirilmiş iletişim* operatörünü kullanmıştır. Bu yaklaşımda popülasyonların kendi içlerindeki yakınsamalarını tamamlamaları beklenmekte ve daha sonra yüksek bir iletişim oranı ile iletişimde bulunmalarına izin verilmektedir. Grosso'nun analizlerinin ikinci kısmında elde edilen çözümlerin kalitelerinin, analizlerin ilk kısmında elde edilen çözüm kaliteleri ile aynı olduğu belirtilmiştir.

Munetomo ve ark., Grosso'nun (1985) önerdiği geciktirilmiş iletişim operatörünü farklı bir şekilde kullanarak popülasyon içerisindeki kromozomların çeşitliliği belirli bir değerin altına düştüğünde popülasyonlar arasında iletişim kurulmasına izin vermiştir. Bu çalışmadaki diğer bir farklı yaklaşım da, kullanılan dinamik topoloji yapısıdır. Bu topolojiye göre iletişim işlemi, en düşük popülasyon çeşitliliğine sahip popülasyona yapılmaktadır [Munetomo ve ark., 1993].

Starkweather ve ark., kısmi bir şekilde izole çalışan popülasyonların arama uzayının farklı bölgelerini örneklediğini gözlemlemiş, daha sonra bu popülasyonlar arasında iletişim operatörü ile bilgi alışverişine izin verildiğinde farklı popülasyonlardan elde edilen farklı kromozomların birleştirilmesi yoluyla daha iyi çözümlerin elde edildiğini belirtmiştir. Ayrıca bu durumun amaç fonksiyonu ayrılabilir olduğunda (kısmi çözümlerin birleştirilmesi yoluyla daha iyi çözümlerin elde edilebilir olduğu durumlarda) söz konusu olduğunu, aksi halde tek popülasyonlu yapının kullanılmasının daha mantıklı olduğunu belirtmişlerdir [Starkweather ve ark., 1991]. Bu bulgular daha sonra Whitley ve ark. tarafından da onaylanmıştır [Whitley ve ark., 1999]. Benzer şekilde Rudolph'un çok popülasyonlu evrimsel algoritmalar üzerinde yaptığı deneysel çalışmaların sonuçları da Starkweather ve ark. (1991) ile tutarlılık göstermektedir [Rudolph, 1991].

Lin ve ark., erken yakınsamayı engellemek için çeşitli iletişim operatörleri, topolojiler ve çok popülasyonlu PGA yaklaşımları üzerinde çalışmışlardır. Geliştirdikleri yaklaşımlar arasında en ilgi çeken yaklaşım "Injection-Island Genetic Algorithms" (iiGA) yaklaşımıdır. iiGA yaklaşımı her biri farklı genetik gösterim çözünürlüğüne sahip M popülasyon ile çalışmaktadır ve iletişim operatörü sadece

düşük çözünürlüklü genetik gösterim kullanan popülasyondan yüksek çözünürlük kullanan popülasyon yönünde gerçekleşmektedir. Düşük çözünürlüğe sahip popülasyon ile genel bazı yapı taşları üretip, daha sonra bu yapı taşlarının daha yüksek çözünürlükte bir popülasyon içerisinde işlenmesi amaçlanmaktadır [Lin ve ark., 1994]. Bu yaklaşım daha sonra Goodman ve ark. tarafından bir tasarım problemi üzerinde başarılı bir şekilde kullanılmıştır [Goodman ve ark., 1997].

Kapanoglu, hiyerarşik olarak çözülebilen bir dizi planlama ve çizelgeleme problemini eş zamanlı olarak çözecek, ancak üst düzey planlama kararlarının operasyonel düzeydeki kararları baskılamasına izin vermeyecek bir yaklaşım önermiştir. Yaklaşım, farklı uygunluk değerlerinin farklı yapı taşlarını yani farklı düzeydeki kararları barındıracağı varsayımını içermekte olup, alt popülasyonlar arası iletişim, daha düşük uyum değerlerini içeren alt popülasyonlardan üst popülasyonlara en iyi değerlerin göçüne olanak vermektedir. Farklı düzey kararlar tek bir uygunluk fonksiyonu içine gömülmüş olup, kararların ağırlıkları ve uygunluk değerine katkıları karar vericinin değerlendirmesine bırakılmıştır. Yaklaşım, hiyerarşik problemlerin eş zamanlı çözümlerinin çözüm uzayını daha gürbüz ve fırsatçı keşfetmesini hedeflemektedir [Kapanoglu, 1997]. Hindistan'daki kast sistemine öykündüğü için *kast-tabanlı paralel genetik algoritma* olarak adlandırılmıştır [Kapanoglu ve ark., 2001].

Cantu-Paz, çok popülasyonlu PGA'da popülasyon büyüklüğünün, popülasyon sayısının, göç sıklığının ve uygun topolojilerin belirlenmesi üzerinde çalışmıştır. Cantu-Paz'ın analizleri sonucunda ortaya koyduğu önemli bulgulardan birisi, topolojide hangi popülasyonlar arasında bilgi alışverişine izin verildiğinden çok, bir popülasyonun topoloji yoluyla bilgi alışverişinde bulunmasına izin verilen popülasyon sayısının önemli olduğudur [Cantu-Paz, 2000].

Xu ve ark., popülasyonlar arası bilgi alışverişi için yeni bir yöntem önermiştir: şema göçü (schema migration). Şema göçü operatörü ikili genetik gösterim için geliştirilmiştir ve iki safhadan oluşmaktadır: Şemaların belirlenmesi ve şemaların dağıtılması. Şemaların belirlenmesi safhasında, pozisyon tabanlı olarak bir

popülasyonda her bir pozisyonda yer alan ortak allellere göre kısa tanım uzunluklu, düşük dereceli şemalar (yapı taşları) ve bu şemaların uygunluk değerleri belirlenmektedir. Daha sonra uygunluk değeri yüksek olan bir şema, diğer tüm popülasyonlara aktarılarak iletişim işlemi tamamlanmaktadır [Xu ve ark., 2002].

5.2.3. Geliştirilen genetik algoritma

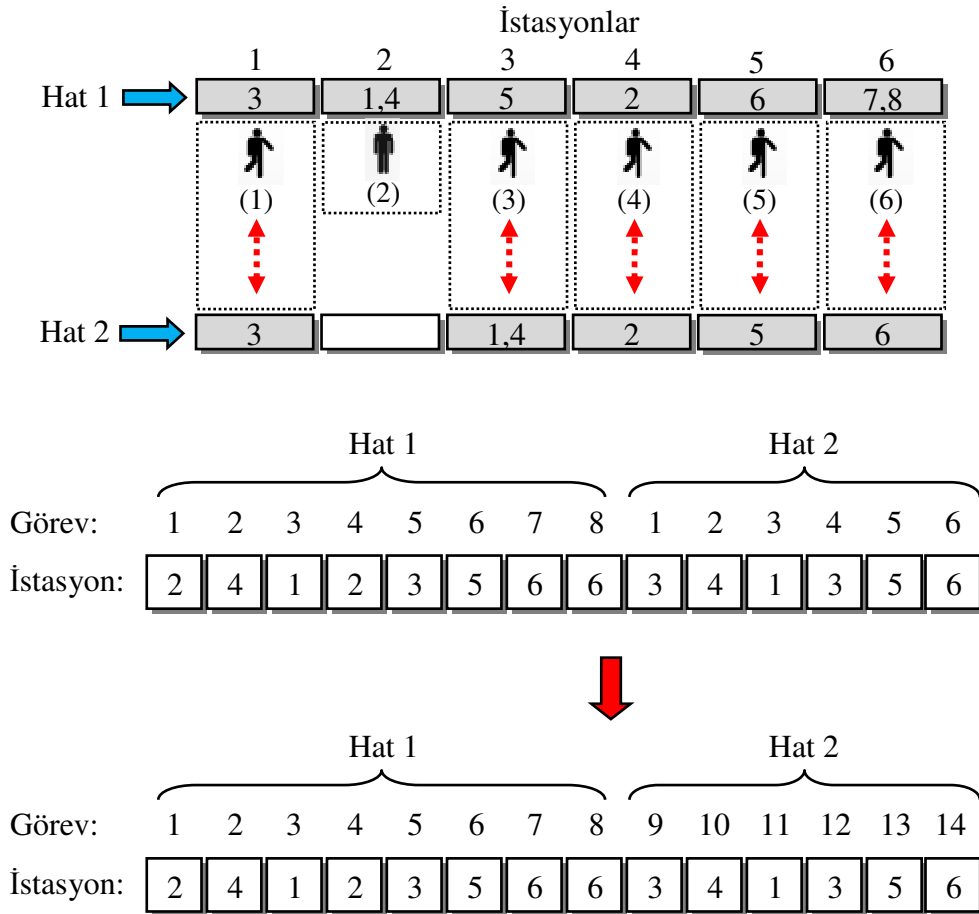
Bu bölümde, M-PMHD probleminin çözümü için önerilen melez yapıda olan çok popülasyonlu paralel genetik (M-ÇPPG) algoritma sunulmuştur. Önerilen melez algoritmanın aşağıdaki bileşenlerden oluşmaktadır.

- Genetik gösterim
- Başlangıç popülasyonunun oluşturulması
- Uygunluk fonksiyonu
- Seçim operatörü
- Çaprazlama operatörü
- Mutasyon operatörü
- İletişim operatörü
- Benzerlik operatörü
- Yönetici popülasyon
- Köle popülasyonlar
- Açgözlü algoritma

Yukarıdaki bileşenler sırayla açıklanacaktır. Bu bölümün en sonunda önerilen algoritmanın genel adımları verilmiştir.

Genetik gösterim: Genetik algoritmayı diğer arama yöntemlerinden ayıran en önemli özellik, parametrenin kendisi yerine parametreleri temsil eden kromozomların kullanılmasıdır. Bu nedenle, ilk adım ele alınan problem için arama uzayını en iyi temsil eden uygun bir kodlama yapısının seçimidir. Kaynaklarda en yaygın kullanılan gösterim ikili düzende kodlama olmasına rağmen; gezgin satıcı, çizelgeleme, araç rotalama, tesis yerleşimi gibi kombinatoriyal optimizasyon

problemlerinde ikili düzende kodlama arama uzayını tam olarak temsil edememektedir. Bu nedenle kaynaklarda, gerçel sayı kodlama ve permütasyon kodlama türleri de çok sık kullanılmaktadır. Bu çalışmada önerilen M-ÇPPG algoritmada uygun çözümler, istasyon numaralarının permütasyonu (permütasyon kodlama) olarak gösterilmiştir. Şekil 5.12’de M-PMHD probleminin bir çözümünü temsil eden genetik gösterim verilmiştir. Her hatta üretilen ürünlere ait görevlerin toplam sayısı $(n_1 + n_2)$, kromozomun uzunluğunu belirtmektedir. Kromozomda $[1 - n_1]$ arasındaki genler birinci hatta üretilen görevlerin hangi istasyonlara atandığını gösterirken; $[(n_1 + 1) - (n_1 + n_2)]$ arasındaki genler ikinci hatta üretilen görevlerin hangi istasyonlara atandığını göstermektedir. Örneğin kromozomun beşinci geni, birinci hatta üretilen ürüne ait 5 numaralı görevin (birinci hattın) 3 numaralı istasyona atandığını göstermektedir.



Şekil 5.12. Örnek bir genetik gösterim

Başlangıç popülasyonunu oluşturulması: Genetik algoritmayı diğer sezgisel arama algoritmalarından ayıran diğer bir önemli özellik ise noktadan noktaya değil, noktaların oluşturduğu bir popülasyon içinde aramayı gerçekleştirmesidir [Goldberg, 1989]. Bu nedenle, GA'nın ilk adımı başlangıç popülasyonunun oluşturulmasıdır. Kaynaklarda, başlangıç popülasyonu en basit şekilde rassal olarak oluşturulmaktadır. Ancak, özellikle kısıtlı en iyileme problemlerinde başlangıç popülasyonunun rassal oluşturulması sonucunda uygun olmayan (kısıtları sağlamayan) çözümler ortaya çıkabilmektedir. Bu durumu ortadan kaldırmak amacıyla ilgilenilen problem için geliştirilmiş sezgisel metotlardan faydalanmaktadır. Kaynaklarda, başlangıç popülasyonunun oluşturulmasında sezgisel metotlardan yararlanan çeşitli çalışmalar vardır. Örneğin, gezgin satıcı problemi için açgözlü sezgisellerden yararlanırken [Grefenstette, 1986], çizelgeleme problemi için Campbell-Dudek-Smith ve Dannenbrig sezgisellerinden yararlanarak başlangıç popülasyonunu oluşturulmuştur [Chen ve ark., 1995].

Genetik algoritmaların başarısı başlangıç popülasyonundaki çözümlerin kalitesine de bağlıdır. Bu nedenle, önerilen M-ÇPPG algoritmada başlangıç popülasyonunu oluşturmak için yüksek kalitede çözümler üretebilen iki farklı algoritma kullanılmıştır. Bunlardan ilki, bölüm 5.1'de anlatılan PKPP algoritmasının birinci aşamasıdır. PKPP algoritması, matematiksel model ile alt problemlerin en iyi çözümlerini bulduğu ve algoritma adımları arasında olasılıklı bir yapı bulunmadığı için, problem penceresinde olacak görev sayısı (L) değeri değişmediği sürece her çalıştırıldığında aynı çözümü verecektir. Dolayısıyla farklı çözümler almak için L değeri belirli aralıklarda değiştirilmiştir. Buna rağmen PKPP algoritmasıyla elde edilebilecek çözüm sayısı sınırlıdır. Başlangıç popülasyonundaki geriye kalan çözümler ise şekil 5.13'te verilen algoritma ile bulunmaktadır. Önerilen algoritma atanabilir görevler kümesi içinden rasgele bir görev seçmek üzerine kurulmuştur. Öncelik ilişkilerini ihlal etmeyen ve ele alınan istasyona çevrim zamanını aşmadan atanabilecek olan tüm görevler, atanabilir görevler kümesinin elemanıdır. Bu kümeden rasgele bir görev seçilir ve küme tekrar güncellenmektedir. Eğer istasyon tamamen dolu ise, yani atanabilir görev kümesi boş ise, yeni bir istasyon açılır ve atanabilir görevler kümesi yeni boş istasyona göre tekrar hazırlanmaktadır.

Algoritma: *Başlangıç popülasyonu oluşturma*

Adım 0:	P: Popülasyon büyüklüğü p : popülasyondaki kromozom sayısı , $p = 0$ I : tüm ürünlere ait görevler kümesi K : atanmayan görevler kümesi A : atanabilir görevler kümesi j : istasyon sayısı $S(j)$: j . istasyon zamanı $idle$: boş zaman C : çevrim zamanı
Adım 1:	$K = I$, $idle = C$, $j = 1$, $S(j) = 0$
Adım 2:	K kümesinden atanabilir görevler kümesi oluştur: A .
Adım 3:	Eğer $idle < \min\{t_i\}$, $i \in A$ ise $j = j + 1$, $S(j) = 0$, $idle = C$ ve Adım 2'e dön. Eğer $idle \geq \min\{t_i\}$, $i \in A$ ise, A kümesinden rasgele bir görev seç. i^*
Adım 4:	Eğer $t_{i^*} > idle$ ise, A kümesinden i^* görevini çıkart: $A = A \setminus i^*$ ve Adım 3'e dön. Eğer $t_{i^*} \leq idle$ ise, i^* görevini j . istasyona ata. $S(j) = S(j) + t_{i^*}$, $idle = C - S(j)$
Adım 5:	Eğer $K = \{\}$ ise $p = p + 1$ Değilse Adım 2'e dön.
Adım 6:	Eğer $p \leq P$ ise, Adım 1'e dön. Değilse DUR.

Şekil 5.13. Başlangıç popülasyonu oluşturmak için kullanılan algoritma

Uygunluk değerinin hesaplanması: Her jenerasyonda, popülasyonda bulunan kromozomların (çözümlerin) bir değerlendirme fonksiyonu yardımıyla uygunluk değerleri hesaplanır. Uygunluk değeri, bir sonraki popülasyonu oluşturacak yeni aday çözümlerin elde edilmesi için mevcut popülasyondan hangi aday çözümlerin kullanılacağını belirlenmesinde rol oynamaktadır. GA'da kullanılan değerlendirme fonksiyonu problemin amaç fonksiyonudur. Bölüm 4'te değişik birkaç farklı amaç fonksiyonu için çözümler alınmıştır. Önerilen M-ÇPPGA'da kullanılacak amaç fonksiyonu olarak Eş. 4.12'de gösterilen toplam işgücü maliyetini enküçükleme alınmıştır.

En küçükleme problemleri için orantılı seçim mekanizmaları kullanıldığında, uygunluk değerinin hesaplanması için amaç fonksiyonunun doğrudan kullanılması

mümkün değildir. Çünkü, bu seçim mekanizmalarında uygunluğun en büyükleme ile ilgilenilmektedir. En küçükleme problemini, en büyükleme problemine dönüştürmek için kullanılan bir yöntem; en küçükleme probleminin amaç fonksiyonunun -1 ile çarpılmasıdır. Ancak, GA'da uygunluk değerlerinin pozitif olma koşulundan dolayı bu yöntem kullanılamamaktadır. GA ile çözülen en küçükleme problemi, başlangıçta belirlenen büyük bir değerden problemin amaç fonksiyonu değeri çıkarılarak en büyükleme problemine dönüştürülür [Goldberg, 1989].

Seçim mekanizması: Başlangıç popülasyonu oluşturulduktan sonra algoritmanın her tekrarında (jenerasyonda), yeni popülasyonun kromozomları bir olasılıklı seçim süreci ile mevcut popülasyonun kromozomları arasından seçilir. Seçim mekanizması, mevcut popülasyondan bir sonraki kusağa hangi aday çözümlerin (kromozom) aktarılacağını belirler. Bu işlem ile özel genetik yapıların bir sonraki popülasyona taşınması sağlanır. Yüksek uygunluk değerine sahip çözümler (kromozomlar), yeni çözümlerin elde edilmesinde yüksek olasılığa sahiptir. Bu operatör, doğal seçimi yapay olarak gerçekleştirmektedir. Kaynaklarda, yeniden üretim operatöründe kullanılan çeşitli seçim mekanizmaları vardır. Bunlardan en çok kullanılanlar *rulet çemberi yöntemi*, *sıralama (rank) yöntemi*, *turnuva yöntemi*, *denge durumu yöntemi* ve *elitizm*'dir. Bu çalışmada, turnuva yöntemi, denge durumu yöntemi ve elitizm yöntemleri seçim mekanizması olarak kullanılmıştır.

Turnuva seçim mekanizması: Bu mekanizmada, popülasyondan rassal olarak seçilen bir grup kromozom (yerine koyarak / yerine koymadan) seçilir. Bu grup içindeki en iyi uygunluk değerine sahip kromozom yeni popülasyona kopyalanır. Popülasyon genişliğine (her popülasyonda olması istenen kromozom sayısı) ulaşınca kadar bu işleme devam edilir. Genellikle grup genişliği ikidir. Ancak bu sayının artırılması da mümkündür.

Denge durumu seçim mekanizması: Bu mekanizmanın işleyişi diğerlerinden biraz farklıdır. Anlatılan mekanizmalarda öncelikle mevcut popülasyondan kromozomların seçimi ile yeni popülasyon oluşturulur. Oluşturulan bu yeni popülasyona genetik

operatörler uygulanarak yeni kromozmlar elde edilir. Elde edilen yeni kromozomlar, kendilerini oluşturmakta kullanılan kromozomlar ile yer değiştirerek yeni popülasyona alınır. Bu mekanizmada ise, öncelikle doğrusal sıralı seçim mekanizması kullanılarak seçilen bir ya da iki bireye genetik operatörler uygulanır. Elde edilen yeni kromozoma mevcut popülasyondaki uygunluk değeri en düşük kromozomlar ile yer değiştirerek yeni popülasyon oluşturulur.

Elitist seçim mekanizması: Diğer seçim mekanizmaları ile birlikte kaynaklarda çok sık kullanılmaktadır. Bu mekanizmada, öncelikle orantılı seçim mekanizmalarından birisi kullanılarak yeni popülasyon elde edilir ve mevcut popülasyondaki en iyi uygunluk değerine sahip bir kromozom (veya belirli bir yüzdeye giren birkaç kromozm) yeni popülasyonda da korunur. Amaç, elde edilen en iyi uygunluk değerine sahip kromozomun örnekleme hatası veya genetik operatörler kullanımı sonucunda kaybolmasını önlemek ve GA'nın çalışma süresince her tekrarda popülasyonun en iyisini veya iyilerini korumaktır.

Çaprazlama operatörü: Çaprazlama, iki kromozomun birbirleri arasında gen alışverişinde bulunup iki yeni kromozom oluşturmaktır. Çaprazlama operatörü, popülasyondan rassal olarak seçilen iki kromozomun belirli bölümlerini karşılıklı olarak değiştirerek, arama uzayında yeni noktaları verecek yeni iki farklı kromozom elde etmektedir. Yukarıda anlatılan çaprazlama işleminin uygulanması için seçilen kromozomlara uygulanan bu işlem ile arama uzayında farklı noktaların çözüm adayı olarak değerlendirilmesi ve büyük adımlar ile çözüm uzayının taranması gerçekleştirilir. Çaprazlama toplumda çeşitliliği sağlarken, iyi özelliklerin bir araya gelmesini kolaylaştırarak en iyiye yaklaşmayı amaçlamaktadır. Permütasyon türü kodlama kullanan genetik algoritmalar için kaynaklarda pek çok çaprazlama operatörü önerilmiştir (PMX, OX, MPX vb.). Bu çalışmada yeni bir çaprazlama operatörü kullanılmıştır. Önerilen çaprazlama operatörünün adımları [şekil 5.14](#)'te verilmiştir.

Algoritma: Çaprazlama	
Adım 0:	n : istasyon sayısı , C : çevrim zamanı
Adım 1:	$[1, \lfloor (n-1)/2 \rfloor]$ aralığında rasgele bir sayı üret: r $\lfloor x \rfloor$: x 'e eşit veya küçük en büyük tamsayı.
Adım 2:	$[1, 2r]$ aralığında olan istasyon sayılarına sahip Ebeveyn 1'in genleri aynen çocuk 1'e aktarılır. $[2r, n]$ aralığında olan istasyon sayılarına sahip Ebeveyn 2'in genleri aynen çocuk 1'e aktarılır.
Adım 3:	Geriye kalan tüm boş genler * simgesiyle işaretlenir.
Adım 4:	Tüm istasyonların doluluk oranları hesaplanır: $S(i)$ Eğer $S(i) \leq C/2$ ise, i . istasyona atanan tüm görevlerin bulunduğu genler * simgesi ile işaretlenir.
Adım 5:	İşaretli olan genlere (*) karşılık gelen görevler adaptasyon süreci ile yeniden atanır.

Şekil 5.14. Çaprazlama operatörü

Adaptasyon operatörü: Seçilen iki kromozom üzerinde çaprazlama operatörü uyguladıktan sonra elde edilen çocuklarda bazı genler kusurlu oluşmaktadır. Bu kusurlu olan genler “*” simgesiyle gösterilmektedir. Bu kusurlu genlere (değeri atanmamış genler) önerilen adaptasyon operatörü kullanılarak uygun çözümler elde etmek için gerekli değerler atılmaktadır. Bu adaptasyon operatörü aynı zamanda mutasyon operatörüyle de birlikte kullanılmaktadır. Adaptasyon operatörünün adımları [şekil 5.15](#)'te verilmiştir. Adaptasyon operatöründe kullanılan notasyonlar aşağıda açıklanmıştır:

I : tüm görevler kümesi

R : atanacak görevler kümesi

$NR(i)$: i . görevin R kümesindeki öncül görev sayısı

$E(i)$: i . görevin $I - R$ kümesindeki öncül görevler kümesi

$L(i)$: i . görevin $I - R$ kümesindeki tüm ardıl görevler kümesi

T_j : j . istasyon zamanı

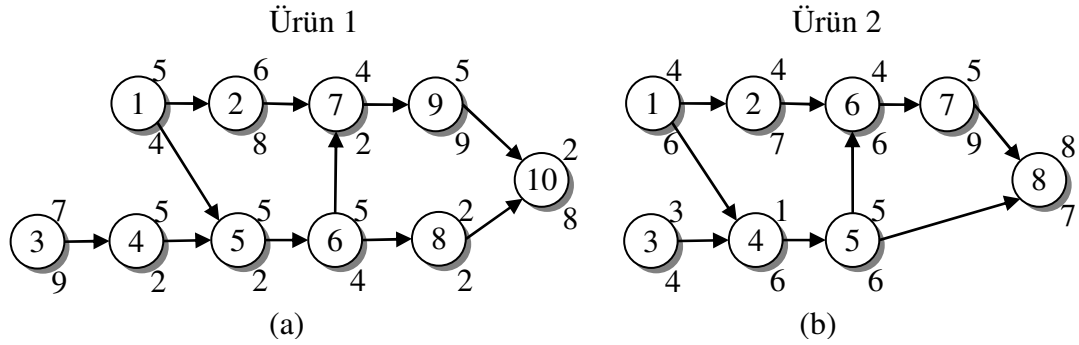
w_i : i . görevin ücret oranı

ws_j : j . istasyonun ücret oranı

Algoritma: Adaptasyon	
Adım 1:	İşaretlenmiş (*) görevler kümesi belirle: R ve T_j , $j=1, 2, \dots, n$ değerlerini hesapla
Adım 2:	i . görevin R kümesindeki öncül görev sayısını hesapla: $NR(i)$, $i \in R$
Adım 3:	$NR(i) = 0$, $i \in R$ olan i görevini seç: i^* Eğer birden fazla $NR(i) = 0$, $i \in R$ değerine sahip görev varsa, en büyük görev zamanı olan görevi seç. Eşitlik devam ediyorsa rasgele bir görev seç.
Adım 4:	$E(i^*)$ ve $L(i^*)$ kümelerini belirle $E(i^*)$ kümesindeki görevlerin en son atanmış olan istasyonu bul Eğer $E(i^*)$ kümesi boş ise, ilk istasyonu seç. $L(i^*)$ kümesindeki görevlerin ilk atanmış olan istasyonu bul Eğer $L(i^*)$ kümesi boş ise, son istasyonu seç. (Belirlenen bu istasyon aralığı i^* görevinin öncül ilişkisini bozmadan atanabileceği aralıktır)
Adım 5:	i^* . görevin atanabileceği $[E(i^*), L(i^*)]$ aralığında bir istasyon seç: j^* Eğer $T_j + t_{i^*} \leq C$ koşulunu sağlayan istasyon yoksa uygun olmayan çözüm. DUR Eğer $T_j + t_{i^*} \leq C$ koşulunu sağlayan istasyonlar varsa, $j^* = \min\{w_{i^*} - ws_j\}$ Eğer eşitlik varsa en dolu olan ($\max T_j$) istasyonu seç Eğer eşitlik varsa en boş olan ($\min T_j$) istasyonu seç. Eğer eşitlik devam ediyorsa mümkün olan istasyonlardan en küçük numaralı olanı seç. i^* görevini j^* istasyona ata.
Adım 6:	i^* görevini R kümesinden kaldır. Eğer R kümesi boş ise Adım 7'e geç. Değil ise $T_{j^*} = T_{j^*} + t_{i^*}$ ve Adım 2'ye git.
Adım 7:	Eğer hiç görev atanmamış istasyon varsa atanana görevleri sola kaydır ve DUR. Eğer tüm istasyonlara görev atanmış ise DUR

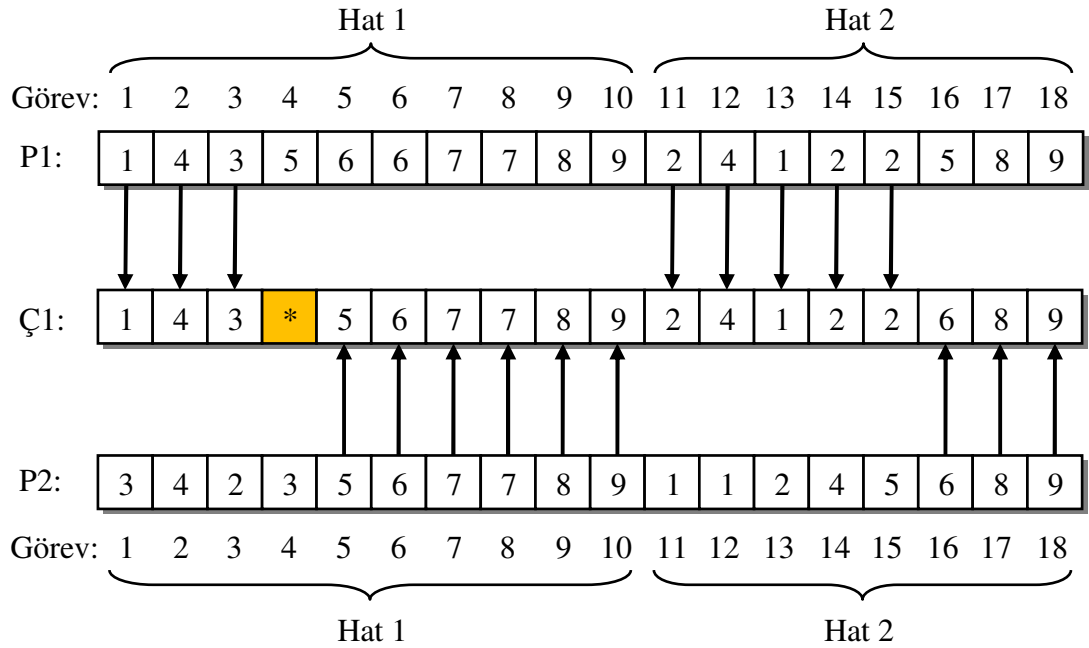
Şekil 5.15. Adaptasyon operatörü

Şekil 5.14 ve Şekil 5.15'te verilen çaprazlama ve adaptasyon operatörlerinin daha iyi anlaşılması için Şekil 5.16'da gösterilen açıklayıcı bir örnek üzerinde gösterilmiştir.



Şekil 5.16. (a) 8 görevden (b) 10 görevden oluşan 2 ürünlü bir problem

Yukarıda verilen M-PMHD probleminin uygun iki çözümü [şekil 5.17](#)'de gösterildiği gibidir. Bu iki çözüme önerilen çaprazlama operatörü uygulanmıştır. Çaprazlama operatörü uygulanırken $r = 4$ olarak belirlenmiştir.



Şekil 5.17. Örnek bir çaprazlama gösterimi ($r = 4$)

Çaprazlama işlemi sonucunda oluşan çocuğun bozuk bir geni bulunmaktadır (Bkz. [Şekil 5.18](#)). Bu gen için adaptasyon süreci başlatılacaktır. Adaptasyon sürecinin işletilen adımları aşağıda ayrıntılarıyla verilmiştir.

	Hat 1										Hat 2							
Görev:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Ç1:	1	4	3	*	5	6	7	7	8	9	2	4	1	2	2	6	8	9

Şekil 5.18. Çaprazlama operatörü sonucunda kusurlu çıkan bir çocuk

- Adım 1: $R = \{4\}$
 $T_1 = 8, T_2 = 10, T_3 = 7, T_4 = 10, T_5 = 5, T_6 = 9, T_7 = 6, T_8 = 10, T_9 = 10$
- Adım 2: $NR(4) = 0$
- Adım 3: $NR(4) = 0$ olan 4. görev seçilir: $i^* = 4$
- Adım 4: $E(4) = \{3\}$ ve $L(4) = \{5, 6, 7, 8, 9, 10\}$
 $E(4)$ kümesindeki 3. görev 3 nolu istasyona atanmıştır.
 $L(4)$ kümesindeki 5., 6., 7., 8., 9. ve 10. görevler sırasıyla 5., 6., 7., 7., 8. ve 9. istasyonlara atanmıştır. $L(4)$ kümesindeki görevlerin içinden en erken atanan 5. görev 5. istasyona atanmıştır. Dolayısıyla öncül ilişkilerinin bozulmaması için 4. görevin [3 - 5] nolu istasyonlardan birisine atanması gerekmektedir.
- Adım 5: $T_3 + t_4 = 7 + 5 > C, T_4 + t_4 = 10 + 5 > C, T_5 + t_4 = 5 + 5 \leq C, T_j + t_{i^*} \leq C, j=5$ koşulunu sağlayan tek istasyon olduğu için 4. görev 5. istasyona atanır.

P1 ve P2 ebeveyn'nin amaç fonksiyonu değeri sırasıyla 560 ve 590'dır. Çaprazlama ve adaptasyon işlemi sonucunda oluşan [şekil 5.19](#)'daki çocuk 1'in amaç fonksiyon değeri 540'tır. Bu sonuç aynı zamanda problemin en iyi değeridir.

	Hat 1										Hat 2							
Görev:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Ç1:	1	4	3	5	5	6	7	7	8	9	2	4	1	2	2	6	8	9

Şekil 5.19. Adaptasyon operatörü sonucunda düzeltilen çocuk

Mutasyon operatörü: Mutasyon operatörü kromozom üzerinde gen bazlı değişiklikler yapmak yoluyla kromozomun belirli bir komşuluğunda örnekleme yapmaktadırlar. Bu komşuluk kullanılan mutasyon operatörüne göre dar olabileceği gibi, oldukça geniş de olabilir. Mutasyon operatörlerinin bu komşulukta yaptıkları örnekleme tamamen rassal bir örneklemedir. Ancak çok popülasyonlu PGA'nın bir parçası olarak tasarlanan aç gözlü GA'nın görevi, kromozomların nispeten dar ve kaliteli çözümleri kodlayan komşuluklarını hızla örnekleyerek, her bir popülasyon için farklı yerel en iyi çözümleri hızlı bir şekilde elde etmektir. Bu nedenle bu çalışmada geliştirilen mutasyon operatörü, kromozomun nispeten kaliteli çözümleri kodlayan dar bir komşuluğunda örnekleme yapacak şekilde tasarlanmıştır.

Kaynaklarda kullanılan mutasyon operatörleri genellikle ekleme ve yerdeğiştirme mutasyon operatörleridir. Bu çalışmada, iki operatörün yaptığını sağlayan tek bir operatör kullanılmıştır. Mutasyon operatörü rasgele belirlenen 2 istasyona ait olan kromozom üzerinde yer alan genleri P_m olasılığıyla mutasyona uğratmak üzere seçmektedir. Mutasyon operatörü kromozom üzerinde mutasyona uğratılmak üzere seçilmiş olan 2 istasyona ait olan tüm genleri kromozomdan çıkartır ve çıkartılan genler '*' simgesiyle işaretlenir ve daha sonra yukarıda anlatılan adaptasyon süreci başlatılır. Aşağıdaki [şekil 5.20](#)'de çaprazlama operatörü uygulandıktan sonra meydana gelen çözümünü gösteren kromozom bulunmaktadır.

	Hat 1										Hat 2							
Görev:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Ç:	2	4	1	2	5	6	7	7	8	9	3	4	1	3	3	7	8	9

Şekil 5.20. Mutasyon operatörü uygulanacak örnek bir kromozom

Mutasyon operatöründe öncelikle hangi iki istasyonun yeniden dengelenmesi gerektiğine rasgele karar verilir. Örneğin 6 ve 8 numaralı istasyonlar yeniden düzenleneceğini varsayalım. 6 ve 8 numaralı istasyonlara ait olan tüm genler '*' simgesi ile [şekil 5.21](#)'de görüldüğü gibi işaretlenir ve adaptasyon süreci başlatılır.

	Hat 1										Hat 2							
Görev:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Ç:	2	4	1	2	5	*	7	7	*	9	3	4	1	3	3	7	*	9

Şekil 5.21. Mutasyona uğramış örnek bir çözüm

Adaptasyon sürecinin adımları aşağıda ayrıntılı olarak verilmiştir:

- Adım 1: $R = \{6,9,17\}$
 $T_1 = 10, T_2 = 10, T_3 = 10, T_4 = 10, T_5 = 5, T_6 = 0, T_7 = 10, T_8 = 0, T_9 = 10$
- Adım 2: $NR(6) = 0, NR(9) = 1, NR(17) = 0$
- Adım 3: $NR(6) = 0$ ve $NR(17) = 0$ olduğu için en büyük görev süresi olanı seç.
 $t_6 = 5$ ve $t_{17} = 5$ olduğu için rasgele olarak 6. görev seçilmiştir: $i^* = 6$
- Adım 4: $E(6) = \{5\}$ ve $L(6) = \{7,8,10\}$
 $E(6)$ kümesindeki 5. görev 5 nolu istasyona atanmıştır.
 $L(6)$ kümesindeki 7., 8. ve 10. görevler sırasıyla 7., 7. ve 9. istasyonlara atanmıştır. $L(6)$ kümesindeki görevlerin içinden en erken atanan 7. görev 7. istasyona atanmıştır. Dolayısıyla öncül ilişkilerinin bozulmaması için 6. görevin [5 - 7] nolu istasyonlardan birisine atanması gerekmektedir.
- Adım 5: $T_5 + t_6 = 5 + 5 \leq C, T_6 + t_6 = 0 + 5 \leq C, T_7 + t_6 = 10 + 5 > C$
 $T_j + t_{i^*} \leq C, j=5, 6$ koşulunu sağlayan iki istasyon olduğu için
 $j^* = \min\{w_6 - ws_j\}, j=5, 6$ hesaplanır.
 $j^* = \min\{w_6 - ws_5, w_6 - ws_6\} = \min\{4 - 2, 4 - 0\} = 2, j^* = 5$
6. görev 5. istasyona atanır
- Adım 6: $i^* = 6$ görevini R kümesinden kaldır. $R = \{9,17\}$
 $T_5 = T_5 + t_6 = 5 + 5 = 10$
- Adım 2: $NR(9) = 0, NR(17) = 0$
- Adım 3: $NR(9) = 0, NR(17) = 0$ olduğu için en büyük görev süresi olanı seç.
 $t_9 = 5$ ve $t_{17} = 5$ olduğu için rasgele olarak 9. görev seçilmiştir: $i^* = 9$
- Adım 4: $E(9) = \{7\}$ ve $L(9) = \{10\}$
 $E(9)$ kümesindeki 7. görev 7 nolu istasyona atanmıştır.
 $L(9)$ kümesindeki 10. görev 9. istasyonlara atanmıştır.
Dolayısıyla öncül ilişkilerinin bozulmaması için 9. görevin [7 - 9] nolu istasyonlardan birisine atanması gerekmektedir.
- Adım 5: $T_7 + t_9 = 10 + 5 > C, T_8 + t_9 = 0 + 5 \leq C, T_9 + t_9 = 10 + 5 > C$

- $T_j + t_{i^*} \leq C$, $j=8$ koşulunu sağlayan tek istasyon olduğu için 9. görev 8. istasyona atanır.
- Adım 6: $i^* = 9$ görevini R kümesinden kaldır. $R = \{17\}$
 $T_8 = T_8 + t_9 = 0 + 5 = 5$
- Adım 2: $NR(17) = 0$
- Adım 3: $i^* = 17$
- Adım 4: $E(17) = \{16\}$ ve $L(17) = \{18\}$
 $E(17)$ kümesindeki 16. görev 7 nolu istasyona atanmıştır.
 $L(17)$ kümesindeki 18. görev 9. istasyonlara atanmıştır.
 Dolayısıyla öncül ilişkilerinin bozulmaması için 17. görevin [7 - 9] nolu istasyonlardan birisine atanması gerekmektedir.
- Adım 5: $T_7 + t_{17} = 10 + 5 > C$, $T_8 + t_{17} = 5 + 5 \leq C$, $T_9 + t_{17} = 10 + 5 > C$
 $T_j + t_{i^*} \leq C$, $j=8$ koşulunu sağlayan tek istasyon olduğu için 17. görev 8. istasyona atanır.
- Adım 6: $i^* = 17$ görevini R kümesinden kaldır. $R = \{\}$
 $T_8 = T_8 + t_{17} = 5 + 5 = 10$
- Adım 7: 6. istasyona hiç görev atanmamıştır. [7 - 9] istasyonuna atanan görevleri birer sola kaydır: istasyon azalması var. DUR

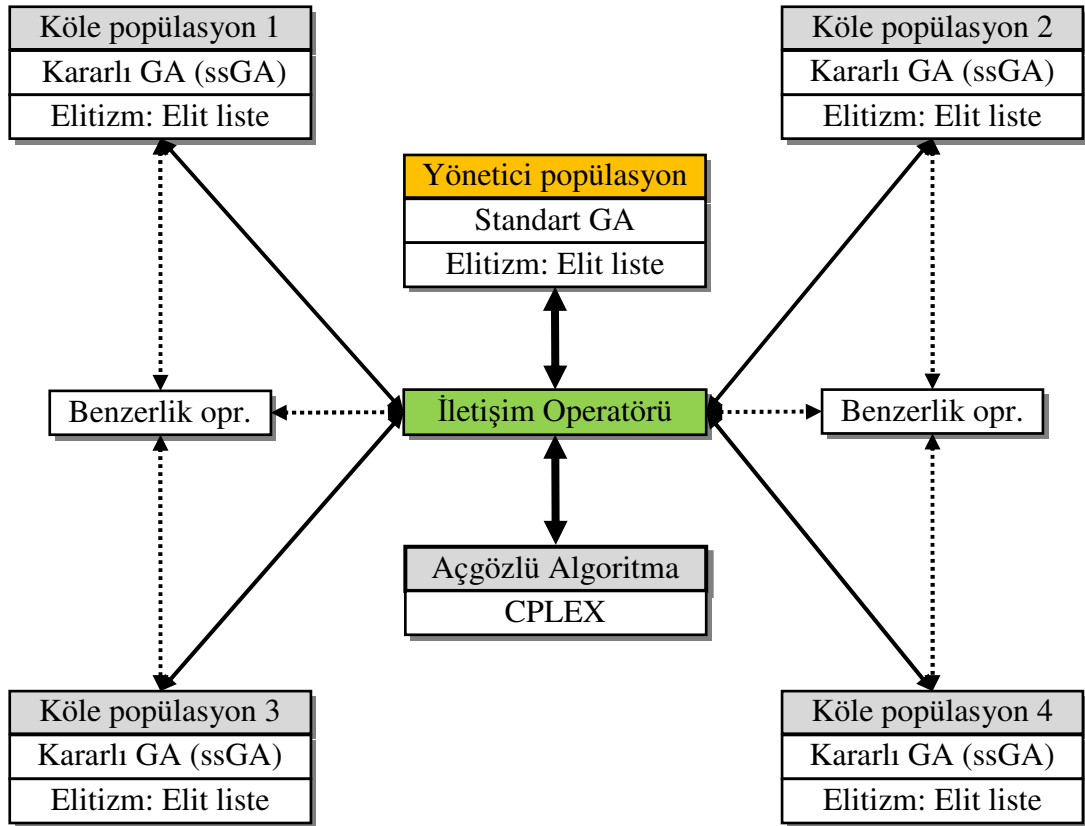
Mutasyon operatörü uygulanırken bazen istasyon sayısını azaltacak özel durumlar oluşabilmektedir. Bu duruma bir örnek [şekil 5.22](#)'de gösterilmiştir.



Şekil 5.22. İstasyon sayısında azalmanın gerçekleştiği bir duruma örnek

Önerilen genetik algoritmanın genel yapısı: Zaman tabanlı MHD probleminin amacı, istasyon sayısını enküçükmek iken; M-MHD probleminin amacı ise toplam üretim maliyetini enküçükmektir. Bir istasyona atanan görevlerin en yüksek maliyet oranı değeri, o istasyonun maliyet oranı olarak belirlendiği için, kullanılan istasyon sayısının azalması her zaman toplam üretim maliyetini azaltmayabilir. Bu konuyla ilgili ilk çalışmayı yapan [Rosenberg ve Ziegler \(1992\)](#), tek hatlı M-MHD probleminin en iyi çözümündeki istasyon sayısının, zaman tabanlı MHD probleminin en iyi çözümündeki istasyon sayısından fazla olmasına rağmen, maliyet tabanlı yaklaşımla birlikte toplam üretim maliyetinin daha düşük bulunduğunu göstermiştir. Ayrıca, bu durumu belirten başka örnekler [Amen \(2000a\)](#), [Amen \(2006\)](#), [Scholl ve Becker \(2005\)](#) çalışmalarında da gösterilmiştir. Buradan çıkarılacak sonuç ise, toplam üretim maliyetinin enküçük olduğu çözümdeki istasyon sayısı, zaman tabanlı yaklaşımla bulunan en iyi istasyon sayısından fazla olabilir. Bu durumda, toplam üretim maliyetini enküçükleyen en iyi çözümdeki istasyon sayısını önceden tahmin edemeyeceğimiz için tüm alternatif istasyon sayısına sahip çözümleri incelemek gerekir. Bu bahsedilen durum, M-PMHD problemi içinde geçerlidir. Önerilen M-ÇPPGA ile en iyi çözümü yakalayabilmek için her istasyon sayısına sahip çözümlere bakılmaktadır.

Başlangıç popülasyonunu oluşturmak amacıyla kullanılan algoritmalar ile değişik istasyon sayısına sahip çözümler elde edilebilir. Fakat, çaprazlama operatörü sonucunda meydana gelen her çocuk çözümün uygun olmamasından dolayı, popülasyondaki çözümlerde belirli istasyon sayısına doğru yakınsamaktadır. Popülasyondaki tüm çözümlerin istasyon sayılarının istenmeyen değere doğru yakınsaması sonucunda, en iyi çözümün elde edilmesi mümkün değildir. Bu sorunu çözmek amacıyla farklı istasyon sayısına sahip çözümlerin farklı popülasyonlarda; aynı istasyon sayısına sahip çözümlerin ise aynı popülasyonda yer aldığı çok popülasyonlu bir yapı kullanılmıştır. Önerilen M-ÇPPGA bir yönetici popülasyon ve n tane köle popülasyon üzerinden M-PMHD probleminin çözümünü araştırır. M-ÇPPGA'nın genel yapısı [şekil 5.23](#)'te gösterilmektedir.



Şekil 5.23. Melez çok popülasyonlu paralel genetik algoritmanın genel yapısı

Şekil 5.23’de genel yapısı gösterilen M-ÇPPGA’nın iki farklı türden popülasyona sahip olduğu görülmektedir: *Yönetici* ve *köle popülasyonlar*. Yönetici popülasyon, başlangıç popülasyonundaki tüm çözümleri içermektedir. Yönetici popülasyon sadece bir tane olup, her istasyon sayısına sahip olan karma çözümlerden oluşmaktadır. Karma çözümlerden oluştuğu için, bu popülasyonda bulunan iki farklı kromozomun çaprazlanması sonucunda çözüm uzayının yeni bölgeleri keşfedilebilir. Köle popülasyonlarda ise aynı istasyon sayısına sahip çözümler bulunmaktadır. Fakat köle popülasyonların her birinde birbirinden farklı istasyon sayılarına sahip çözümler tutulmaktadır. Dolayısıyla kaç farklı istasyon sayılı çözüm varsa, o kadar köle popülasyon var demektir. Yönetici popülasyon için sabit bir popülasyon büyüklüğü varken, köle popülasyonlar için bu değer dinamik yapıdadır. Her köle popülasyonun içerisindeki çözüm sayısı diğer köle popülasyonlarına göre değişiklik göstermektedir. Fakat, tüm köle popülasyonlardaki çözüm sayısı için bir üst sınır değeri vardır.

Popülasyonlar üzerinde iki farklı yapıda genetik algoritma çalışmaktadır: *Standart* ve *denge durumlu (kararlı durum)* genetik algoritma. Yönetici popülasyon üzerinde standart genetik algoritma çalışırken; köle popülasyonların her biri üzerinde denge durumlu genetik algoritma çalışmaktadır. Her iki genetik algoritmada uygulanan *seçim, çaprazlama, mutasyon operatörleri* aynı yapıdadır. Fakat standart genetik algoritma her popülasyon oluştururken P_c olasılıkla popülasyon büyüklüğü kadar çaprazlama operatörü uygularken; denge durumlu genetik algoritmada ise P_c olasılıkla bir kez çaprazlama uygulanmaktadır.

Her popülasyonun kendine ait en iyi iki çözümün saklandığı bir elit listesi vardır. Tüm popülasyonlarda çaprazlama operatörünün uygulanacağı iki kromozomun biri elit listeden, diğeri ise popülasyondan turnuva seçim yöntemine göre belirlenir. Çaprazlama operatörünün uygulanacağı kromozomlardan birisini elit listeden seçmenin amacı, iyi olan çözümlere yeni genler ilave ederek daha iyi hale getirmektir.

Köle popülasyonunda yer alan çözümlere kendi aralarında çaprazlama ve mutasyon operatörü uygulandığı için çözümlerdeki çeşitlilik kaybolmaz. Başka bir ifadeyle, her köle popülasyonu içerisindeki aynı istasyon sayısına sahip çözümler arasında çaprazlama operatörü uygulanacağından her istasyon sayılı çözüm elde tutulmaktadır. Bu yapı aynı zamanda çözüm uzayının farklı bölgelerinde eş zamanlı arama yapmayı da sağlamaktadır. Fakat aynı durum yönetici popülasyonu için geçerli değildir. Yönetici popülasyonunda bulunan çözümler, birbirinden farklı sayıda istasyon sayısına sahip çözümlerdir. Dolayısıyla aralarındaki çaprazlama işlemi sonunda bazılarının yok olma riski vardır. Hatta sadece belirli sayıda istasyon sayısına sahip olan çözümler bile kalarak, diğerlerinin nesli tükenebilir.

M-ÇPPGA'nın ilerleyen adımlarında (jenerasyonlarında) kromozomların üzerinde çaprazlama operatörü uygulamanın sonucunda elde edilen çözümlerin bazılarının uygun olmayan çözüm çıkması sonucu, yönetici popülasyonundaki farklı istasyon sayısına sahip çözümler açısından çeşitlilik azalacaktır. Köle popülasyonlarda ise denge durumlu genetik algoritma kullanıldığı için yerel eniyiye hızla yakınsama ve o

bölgede takılma sorunu yaşanacaktır. Kaynaklardaki çok popülasyonlu genetik algoritma çalışmalarına göre, popülasyonların arasında iletişimin olmaması halinde elde edilen çözümlerin daha kötü olduğu belirtilmektedir. Bu nedenle, popülasyonlar arasında belirli aralıklarla çözüm göçüne izin veren bir *iletişim operatörü* kullanılmıştır. Köle popülasyonların her birinde belirli istasyon sayısına sahip çözümler olduğu için köle popülasyonların arasında bir çözüm aktarımı söz konusu olamaz. İletişim operatörü sadece yönetici popülasyon – köle popülasyonlar arasında çift yönlü çözüm aktarma görevini üstlenmiştir.

Yönetici popülasyonun ve her köle popülasyonlarının kendine ait bir elit listesi bulunmaktadır. Bu listelerde popülasyonların en iyi iki çözümü tutulmaktadır. Yönetici popülasyondaki farklı sayıda istasyon sayısına sahip olan çözümleri belirli oranda tutmak için her köle popülasyonun elit listesinde bulunan en iyi iki çözüm yönetici popülasyona aktarılır. Bu aktarma işlemi iletişim operatörünün kontrolüyle belirli sayıda jenerasyon geçtikçe (örn her 1000 iterasyonda bir) yapılmaktadır. En iyi çözümleri koruma altına alma işlemi standart genetik algoritmalarda da olan bir süreçtir. Standart genetik algoritmalarda en iyi çözümleri kaybetmemek için elit listesinde bulunan çözümler hiçbir çaprazlama veya mutasyon sürecinden geçmeden olduğu gibi yeni jenerasyona aktarılır. Önerilen genetik algoritmada bu aktarma işlemi popülasyonlar arasında yapılmaktadır. Köle popülasyonlardan gelen en iyi çözümler sayesinde hem yönetici popülasyon yakınsamamış olacak hemde yeni gelen çözümlerle çözüm uzayının farklı bölgeleri araştırılabilecektir.

Köle popülasyonlarının yakınsama sorunu içinde benzer bir yapı kullanılmıştır. Köle popülasyondaki çözümlerin yerel eniyiye yakınsaması iletişim operatörü tarafından kontrol edilmektedir. Köle popülasyonu üzerinde denge durumlu genetik algoritma kullanıldığı için popülasyondaki çözümler her geçen jenerasyon birbirine daha çok benzeyecektir. Hatta önerilen melez paralel genetik algoritma uzun iterasyonlar çalıştırıldığında köle popülasyondaki çözümlerin birbirinin aynısı olduğu görülebilir. Popülasyondaki çözümlerin birbirine benzemesi arama yaptığı yerel bölgeyi tam anlamıyla araştırmamasına neden olur. Bu sorunu çözmek için her köle popülasyondaki çözümlerin benzerlik oranları hesaplanmıştır. Popülasyondaki her iki

kromozomun genleri ikili olarak karşılaştırılır. Aynı olanlara 1, farklı olanlara 0 değeri verilir. Benzerlik oranı [şekil 5.24](#)'te gösterildiği gibi hesaplanır:

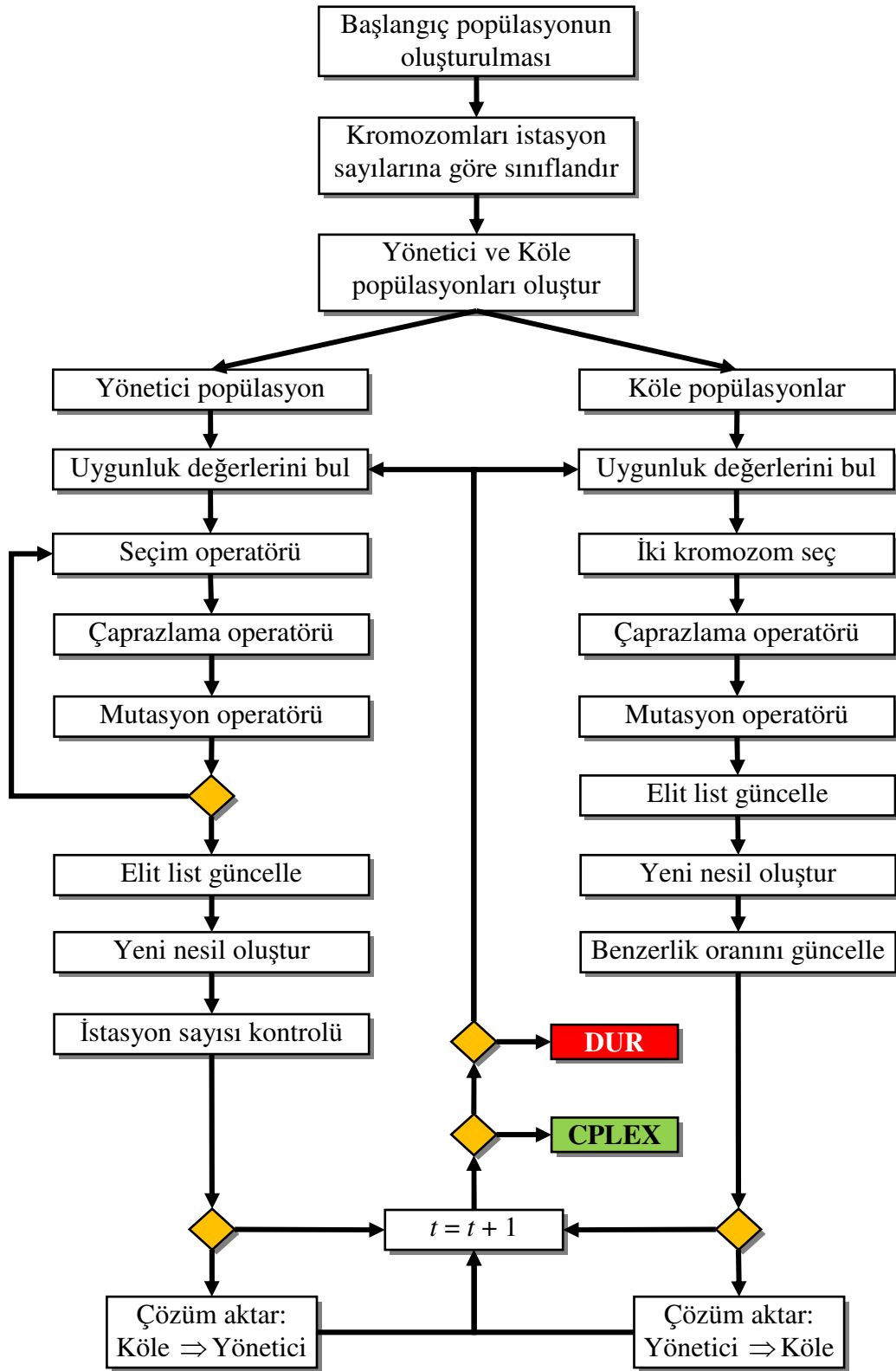
	Hat 1										Hat 2							
Görev:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
K1:	1	4	3	5	6	6	7	7	8	9	2	4	1	2	2	5	8	9
K2:	3	4	2	3	5	6	7	7	8	9	1	1	2	4	5	6	8	9
	0	1	0	0	0	1	1	1	1	1	0	0	0	0	0	0	1	1

$$\text{K1 ve K2 kromozomları arasındaki benzerlik oranı} = \frac{8}{18} = \%44,4$$

Şekil 5.24. Benzerlik oranı hesabına bir örnek

Tüm popülasyondaki kromozomların ikili olarak karşılaştırması sonucunda elde edilen benzerlik oranlarının ortalaması alınır. Bu değer popülasyonun genel benzerlik oranıdır. Eğer hesaplanan popülasyon benzerlik oranı önceden belirlenen benzerlik oranı üst limitine ulaşırsa, popülasyon yakınsamış demektir. Bu popülasyonlara *körelmiş* popülasyonlarda denilmektedir. Kaynaklardaki benzer çalışmalarda körelen popülasyonlar kapatılıp, yeniden üretim operatörü ile yeni çözümlerle doldurulur. Bu çalışmada yakınsayan (körelen) köle popülasyonunda bulunan çözümleri çeşitlendirmek için, yönetici popülasyonda bulunan aynı istasyon sayısına sahip (en iyi) çözümlerden aktarma yapılmaktadır. Yukarıda bahsedildiği gibi köle popülasyonlarının belirli bir popülasyon büyüklüğü yoktur. Sadece bir popülasyonda olmasına izin verilen en fazla kromozom sayısı limiti vardır. P popülasyon büyüklüğü üst limiti, p popülasyondaki kromozom sayısı ve n yakınsamış olan köle popülasyonuna eklenecek olan yeni kromozom sayısı ise, $p + n - P$ kadar köle popülasyonunun en kötü kromozomları popülasyondan çıkarılır. Böylece popülasyon büyüklüğü üst limit değerini geçmemiş olur. Yönetici popülasyon çözüm çeşitliliğini sağlarken, köle popülasyonlar ile çözüm uzayının araştırılan bölgesindeki yerel en iyi noktaya hızlı bir şekilde yakınsaması sağlanmıştır.

Son yıllarda kaynaklardaki genetik algoritma çalışmalarının çoğunda, genetik algoritmalar bir yerel aramaya dayalı algoritmayla desteklenmiştir. Önerilen M-ÇPPGA'nın içerisinde [bölüm 5.1](#)'de anlatılan PKPP algoritmasının ikinci aşaması yerel aramaya dayalı algoritma olarak kullanılmıştır. Önceden belirlenen iterasyon aralıklarıyla devreye girerek istenen popülasyonun sadece elit listesinde bulunan çözümlerini iyileştirmeye çalışmaktadır. Önerilen M-ÇPPGA'nın tüm adımları [şekil 5.25](#)'te gösterilmiştir.



Şekil 5.25. M-CPPGA'nın genel adımları

6. DENEYSEL ÇALIŞMALAR

Bu bölümde, M-PMHD problem setlerinin üretilmesi, M-PMHD problemlerin önerilen tamsayı programlama modeli yardımıyla en iyi çözümlerinin araştırılması, en iyi çözümlerin mantıksal zaman dilimi içerisinde bulunmadığı durumlarda kullanılmak amacıyla geliştirilen PKPP ve M-ÇPPG algoritmalarıyla yüksek kalitede çözümlerin araştırılmasına yer verilmiştir. M-PMHD problemi ilk defa bu tezde ele alındığı için, kaynaklarda bu probleme ait herhangi bir deney seti bulunmamaktadır. Önerilen tamsayı programlama modellerinin, PKPP ve M-ÇPPG algoritmalarının etkinliğini belirleyebilmek için test problemlerine ihtiyaç vardır. PKPP ve M-ÇPPG algoritmalarının performans karşılaştırmalarının yapılabilmesi için, aynı zamanda literatürde M-PMHD problemi üzerinde çalışmak isteyen araştırmacılara kaynak olması amacıyla test problemleri üretilmiştir. Test problemlerinin üretilmesinde dikkate alınan hususlar ileriki bölümde ele alınacaktır.

M-PMHD probleminin çözümü için önerilen tamsayı programlama modeli küçük boyutlu M-PMHD problemleri üzerinde uygulanarak, en iyi çözümleri araştırılmıştır. Aynı problemler ele alınarak her hattın birbirinden bağımsız olarak maliyet tabanlı yaklaşımla dengelenmesi sağlanmıştır. Paralel hatların birbirinden bağımsız veya eş zamanlı olarak dengelenmesinin üretim maliyetleri üzerindeki etkisi incelenmiştir.

Orta ve büyük boyutlu M-PMHD problemlerinin çözümü için önerilen PKPP ve M-ÇPPG algoritmalarının en iyi parametre setlerinin belirlenmesi amacıyla yapılan deneysel çalışmalar ele alınmıştır. Elde edilen deney sonuçlarına göre belirlenen en iyi parametre setlerini kullanarak, küçük, orta ve büyük boyutlu M-PMHD test problemleri üzerinde PKPP ve M-ÇPPG algoritmalarının performansları karşılaştırılmıştır.

M-YZPMHD ve M-ÇÜPMHD problemleri, M-PMHD probleminin uzantılarıdır. Dolayısıyla önerilen PKPP ve M-ÇPPG algoritmaları aynı zamanda M-YZPMHD ve M-ÇÜPMHD problemlerini çözmek amaçlı kullanılabilir. Bu tez kapsamında,

PKPP ve M-ÇPPG algoritmalarının performans karşılaştırmaları sadece M-PMHD problemleri üzerinde yapılmıştır.

6.1. Test Problemleri

M-PMHD problemi ilk defa bu tezde ele alındığı için kaynaklarda bu problem ile ilgili herhangi bir test problemi bulunmamaktadır. Bu nedenle geliştirilen tamsayılı programlama modellerinin, PKPP ve M-ÇPPG algoritmalarının performans karşılaştırmalarının yapılabilmesi için gerekli olan test problemleri üretilmiştir. Literatüre bu test problemlerini kazandırabilmek amacıyla, [Amen \(2001\)](#) çalışmasında M-MHD test problemleri üretilirken görev ücret oranları ile ilgili belirtilen özellikler dikkate alınarak, [Gökçen ve ark. \(2006\)](#) çalışmasında kullanılan 14 grup PMHD problem setlerine görev ücret oranları ilave edilmiştir. PMHD test problemleri www.assembly-line-balancing.de web adresinden temin edilebilir.

Önerilen sezgisel algoritmaların performans karşılaştırmalarının yapılabilmesi amacıyla, her test probleminin farklı çevrim zamanlarını da dikkate alırsak, toplamda 95 tane test problemi kullanılmıştır. Tüm problemlerde birbirine paralel olan iki hatta iki farklı ürün üretilmektedir. Birinci ürünlere ait öncelik ilişkilerinden bazıları çıkartılarak, ikinci ürünler oluşturulmuştur. Test problemleri görev sayılarına göre üç gruba ayrılmıştır: iki hatta üretilen ürünlerin toplam görev sayısı 60 göreve kadar olanlar *küçük boyutlu*, 150 göreve kadar olanlar *orta boyutlu* ve 150'den daha fazla görev sayısı olanlar ise *büyük boyutlu* M-PMHD problemi olarak belirtilmiştir.

PMHD problemini, M-PMHD problemine dönüştürmek için her görevin ücret oranlarının belirlenmesi gerekmektedir. [Rosenberg ve Ziegler \(1992\)](#) çalışmasında, emek yoğun hatlarda en yüksek ücret oranı değerinin en düşük ücret oranı değerine oranının $[1,25 - 5,00]$ arasında olabileceğini belirtmiştir. Fakat, [Amen \(2001\)](#) çalışmasında bu oranın deneyimlerine göre 5 olamayacağını belirterek, en fazla 2 olarak kabul etmiştir. Bu oran 1'e eşit olduğunda M-PMHD problemi, PMHD problemine dönüşmektedir. Dolayısıyla bu oranın 1'den büyük bir değer olması gerektiği açıktır. Bu oran ne kadar fazla ise M-PMHD probleminin çözümü o kadar

zorlaşmaktadır. Bu tezde kullanılan problem setleri üretilirken, bir ürüne ait görevlerin ücret oranlarından en büyüğünün en küçüğüne oranının en fazla 2 olabileceği varsayılmıştır. En küçük görev ücret oranının 2 ve en yüksek görev ücret oranının 4 olduğu düşünüldüğünde, bu oran 2 iken; en küçük görev ücret oranının 10 ve en yüksek görev ücret oranının 20 olduğu durumda da bu oran 2’dir. Dolayısıyla görev ücret oranlarının üretilmesi için sadece oran bilgisi yetersiz kalmaktadır. En küçük ücret oranı değeri küçüldükçe bir ürüne ait görevlerin ücret oranları arasındaki farklılık azalmaktadır. Bir ürüne ait görevlerin ücret oranlarının farklılıklarını arttırmak ve M-PMHD probleminin çözümünü zorlaştırmak amacıyla, her görevin ücret oranı $[n_h/2, n_h]$ aralığında düzgün dağılıma uygun şekilde rassal olarak belirlenmiştir. Bir görevin olabilecek en düşük ücret oranı ve en yüksek ücret oranı değerleri düşünüldüğünde, oranlarının en fazla 2 olabileceği açıktır.

Sıra kuvveti değeri, bir ürüne ait görevlerin arasındaki öncelik ilişkilerinin sıklığını göstermektedir. Bu değer $[0 - 1]$ arasında rasyonel bir sayıdır. Sıra kuvveti, 1’e yaklaştıkça problemin çözüm alanı daraldığı için problemin çözümü kolaylaşırken; bu değer 0’a yaklaştıkça ise problemin çözümü zorlaşmaktadır. Sıra kuvveti değeri 1’e eşit olunca problemin bir çözümü vardır. Sıra kuvveti değerleri aşağıdaki [Eş. 6.1](#) kullanılarak hesaplanmıştır:

$$SK_h = \frac{2 \cdot \sum_{i=1}^{n_h} |F_{hi}|}{n_h(n_h - 1)} \quad (6.1)$$

SK_h , h hattında üretilecek olan ürünün öncelik diyagramına göre hesaplanan sıra kuvveti değeri; n_h , h hattında montajı yapılan ürüne ait görev sayısı; F_{hi} , h hattında montajı yapılan ürüne ait i görevinin tüm ardıl görevleri kümesidir. PMHD problemindeki iki hatta üretilen ürüne ait öncelik diyagramları dikkate alınarak SK değerleri hesaplanmıştır. Kullanılan problem setlerinin toplam görev sayıları, görev ücret oranlarının aralıkları, sıra kuvveti değerleri, en yüksek ücret değerinin en düşük ücret değerine oranı vb. karakteristik özellikleri [çizelge 6.1](#)’de verilmiştir.

Çizelge 6.1. M-PMHD test problemlerinin karakteristikleri

Problem	Hat 1				Hat 2			
	n_1	w_{1i}	SK	$w_{\max}/w_{\min}$	n_2	w_{2i}	SK	$w_{\max}/w_{\min}$
Merten	7	[3 – 7]	0,52	2,0	6	[3 – 7]	0,60	1,25
Jaeschke	9	[4 – 9]	0,83	1,8	8	[4 – 9]	0,78	2,0
Jackson	11	[5 – 11]	0,58	2,2	10	[5 – 11]	0,49	1,4
Roszieng	25	[12 – 25]	0,72	2,1	24	[12 – 25]	0,71	2,0
Sawyer	30	[15 – 30]	0,50	2,0	28	[15 – 30]	0,37	1,8
Kilbridge	45	[22 – 45]	0,45	2,0	43	[22 – 45]	0,38	2,0
Hahn	53	[26 – 53]	0,84	2,0	51	[26 – 53]	0,83	2,0
Tonge	70	[35 – 70]	0,59	2,0	66	[35 – 70]	0,65	1,8
Wee-mag	75	[37 – 75]	0,23	2,0	71	[37 – 75]	0,23	1,9
Arcus1	83	[41 – 83]	0,59	2,0	79	[41 – 83]	0,55	1,9
Lutz2	89	[44 – 89]	0,77	2,0	85	[44 – 89]	0,76	1,8
Lutz3	89	[44 – 89]	0,77	2,0	85	[44 – 89]	0,76	1,8
Mukherje	94	[47 – 94]	0,45	2,0	90	[47 – 94]	0,40	1,8
Arcus	111	[55 – 111]	0,40	2,0	107	[55 – 111]	0,37	1,9

6.2. Tamsayılı Programlama Modeli ile Elde Edilen En İyi Çözümler

Çizelge 6.1’deki tüm M-PMHD problemlerine önerilen operatör odaklı tamsayılı programlama modeli uygulanarak, IBM ILOG CPLEX 12.4 yazılımı yardımıyla problemlerin en iyi çözümleri araştırılmıştır. Her problemin çözümü için 86 400 sn (24 saat) zaman sınırı verilmiştir. M-PMHD problemlerinin elde edilen en iyi değerleri çizelge 6.2’de verilmiştir. Çizelge 6.2’nin dördüncü sütunundaki sayılar, paralal montaj hatlarında çalışan operatör sayısını göstermekte iken; beşinci sütunundaki sayılar ise problemin amaç fonksiyonu olan toplam işgücü maliyetinin 86 400 sn sonunda elde edilen en küçük değeridir. Beşinci sütundaki sayıların yanında parantez içerisinde “(opt)” yazılmış ise, bu çözümün en iyi çözüm olduğunu göstermektedir.

Çizelge 6.2. Operatör odaklı tamsayı programlama modeli ile elde edilen M-PMHD problemlerinin en iyi değerleri

Problem	C	Operatör sayısı (K)	Toplam işgücü maliyeti (TİM)	Çözüm süresi (sn)
Merten	9	7	279 (opt)	0,14
	11	5	253 (opt)	0,16
	13	5	286 (opt)	0,16
	17	4	306 (opt)	0,08
Jaeschke	9	8	486 (opt)	0,73
	11	7	506 (opt)	0,66
	13	6	507 (opt)	0,36
	15	5	525 (opt)	0,25
	17	4	510 (opt)	0,30
Jackson	8	13	848 (opt)	3,12
	10	9	840 (opt)	2,42
	13	7	819 (opt)	2,29
	15	6	810 (opt)	1,95
	19	5	836 (opt)	0,89
Roszieng	14	18	4 970	86 400
	16	16	4 944 (opt)	27 219
	17	15	4 947 (opt)	9 290
	22	12	5 170 (opt)	6 222
	30	9	5 340 (opt)	230
Sawyer	25	26	15 250	86 400
	27	25	15 363	86 400
	30	22	15 210	86 400
	36	18	15 300	86 400
	41	16	15 293	86 400
	54	12	15 498	86 400
Kilbridge	57	20	39 501	86 400
	79	14	39 342	86 400
	92	12	39 744	86 400
	110	10	41 030	86 400
	138	8	40 848	86 400
	184	6	41 216 (opt)	738
Hahn	2 004	14	1 238 472 (opt)	66 350
	2 338	12	1 248 492 (opt)	34 292
	2 806	10	1 257 088 (opt)	3 445
	3 507	8	1 297 590 (opt)	2 046
	4 676	6	1 342 012 (opt)	108

6.3. M-MHD ve M-PMHD Problemleri Sonuçlarının Karşılaştırılması

Gökçen ve ark. (2006) çalışmasında, paralel hatların birbirinden bağımsız olarak dengelenmesi yerine, bu paralel hatların eş zamanlı bir şekilde dengelenmesiyle işyeri (operatör) sayısında azalma sağlanabileceği gösterilmiştir. Bu bölümde, birbirine paralel olan hatların maliyet tabanlı bir amaç fonksiyonuyla eş zamanlı olarak dengelenmesinin toplam işgücü maliyeti ve operatör sayısı üzerindeki etkisi araştırılmıştır. Bu durumu incelemek amacıyla, M-PMHD sonuçları M-MHD sonuçlarıyla toplam işgücü maliyeti ve operatör sayısı açısından karşılaştırılmıştır. M-PMHD sonuçlarını elde etmek için bölüm 4.1’de sunulan operatör odaklı tamsayılı programlama modeli kullanılırken, M-MHD sonuçlarını elde etmek için Amen (2006) tarafından önerilen tamsayılı programlama modeli kullanılmıştır. Küçük boyutlu M-PMHD problemlerine tamsayılı programlama modelleri uygulanarak elde edilen en iyi çözümler çizelge 6.3’te verilmiştir.

Çizelge 6.3. M-PMHD ve M-MHD problemlerinin en iyi çözümleri

Problem	C	M-MHDP (Hat 1)		M-MHDP (Hat 2)		K1+K2	İM1+İM2	M-PMHDP (Eş zamanlı)	
		K1	İM1	K2	İM2			K	TİM
Merten (7 - 6)	9	4	162	3	126	7	288	7	279
	11	3	154	3	154	6	308	5	253
	13	3	169	2	130	5	299	5	286
	17	2	170	2	170	4	340	4	306
Jaeschke (9 - 8)	9	5	324	4	234	9	558	8	486
	11	4	341	4	264	8	605	7	506
	13	3	312	3	247	6	559	6	507
	15	3	345	3	285	6	630	5	525
	17	3	374	2	255	5	629	4	510
Jackson (11 - 10)	8	7	456	6	432	13	888	13	848
	10	5	460	5	450	10	910	9	840
	13	4	455	4	468	8	923	7	819
	15	4	495	3	435	7	930	6	810
	19	3	513	3	513	6	1 026	5	836
Roszieng (25 - 24)	14	10	3 052	10	2 478	20	5 530	18	4 970
	16	8	3 024	8	2 384	16	5 408	16	4 944
	17	8	2 924	8	2 482	16	5 406	15	4 947
	22	6	3 102	6	2 574	12	5 676	12	5 170
	30	5	3 210	5	2 850	10	6 060	9	5 340

Çizelge 6.3'ün üçüncü ve dördüncü sütunları sırasıyla, birinci hattın diğer hattan bağımsız olarak maliyet tabanlı bir amaç fonksiyonuyla dengelendiğinde en iyi çözümündeki işyeri (operatör) sayısı ve toplam işgücü maliyeti göstermektedir. Beşinci ve altıncı sütunlar, aynı şekilde ikinci hattın bağımsız olarak dengelenmesi sonucunda elde edilen işyeri (operatör) sayısı ve toplam işgücü maliyetini göstermektedir. Yedinci ve sekizinci sütunlar ise, her hattın birbirinden bağımsız olarak dengelendiğinde en iyi çözümlerindeki işyeri (operatör) sayılarının ve işgücü maliyetlerinin toplamını göstermektedir. Son olarak, dokuzuncu ve onuncu sütunlar, birbirine paralel olan iki hattın eş zamanlı dengelenmesi neticesinde bulunan en iyi çözümdeki işyeri (operatör) sayısı ve toplam işgücü maliyeti değerlerini göstermektedir.

Çizelge 6.3'teki M-PMHD ve M-MHD problemlerinin en iyi çözümleri incelendiğinde, paralel hatların ortak bir şekilde dengelenmesi toplam işgücü maliyetlerini ve işyeri (operatör) sayılarını azaltmaktadır. Örneğin *Roszieng* isimli problem setinin çevrim zamanı 14 olan ilk probleminin çözümü incelenirse, işyeri (operatör) sayısını 20'den 18'e düşerken; toplam işgücü maliyetinin ise 5530'dan 4970'e düştüğü görülmektedir. Bu sonuç, geriye kalan tüm problemlerin çözümlerinde görülmüştür. Dolayısıyla paralel hatların maliyet tabanlı bir amaç fonksiyonuyla eş zamanlı olarak dengelenmesinin hem işyeri (operatör) sayısı hem de toplam işgücü maliyeti açısından kazançları vardır.

6.4. PKPP Algoritmasının En Uygun Parametre Setinin Belirlenmesi

Kombinatorial optimizasyon problemlerinin yaklaşık çözümlerini elde etmek amacıyla kaynaklarda kullanılan meta-sezgisel algoritmalar aynı parametre setleri ile her çalıştırıldığında ele alınan problem için farklı çözümler verebilmektedir. Bu gibi durumlar için deney tasarımı yaparak, algoritmanın yüksek kalitede çözümler bulduğu parametre setleri belirlenmelidir. Bunun yanında, PKPP algoritması matematiksel model tabanlı iki aşamadan oluşan bir sezgisel algoritmadır. Önerilen PKPP sezgisel algoritmanın içerisinde olasılıklı hiçbir yapı bulunmamaktadır.

Bundan dolayı, PKPP algoritması bir problem için aynı parametre setleri kullanılarak her çalıştırıldığında aynı sonucu vermektedir.

PKPP algoritmasının elde ettiği çözümlerin kalitesi ve çözüm süresi parametre değerlerine bağlıdır. Algoritmanın birinci aşamasında belirlenmesi gereken L ve α değerleri, ikinci aşamasında belirlenmesi gereken S ve δ değerleri PKPP algoritmasının parametre setlerini oluşturmaktadır. PKPP algoritmasının parametre setlerinin çözüm kalitesi ve çözüm süresi üzerinde etkileri vardır. Problem penceresindeki görev sayısını belirleyen L parametresinin değeri arttıkça, çözüm kalitesi artarken çözüm süresi de artar. L parametresinin alabileceği en yüksek değer asıl problemde bulunan toplam görev sayısı ile sınırlıdır. L parametresinin değeri aynı zamanda asıl problemi kaç farklı alt probleme bölüneceğini de belirlemiş olur. Kaynaklarda kullanılan meta-sezgisel algoritmalarda olduğu gibi en uygun parametre seti belirleme işlemi PKPP algoritmasında gereksizdir. Eğer PKPP algoritması ile daha yüksek kalitede çözümler elde etmek istenirse, L değerinin artırılması gerekmektedir. Bu durumda çözüm süresi artmaktadır.

PKPP algoritmasının birinci aşamasında oluşturulan her alt problemin en iyi çözümü bulunduğunda, elde edilen en iyi görev atamalarının ne kadarının sabitleneceğini α parametresinin değeri belirlemektedir. Eğer α değeri 1 olursa, her alt problemin en iyi çözümünü hiç bozmadan aynen kabul edilir. α değeri küçüldükçe son istasyonlara atanan görevler tekrar atanmak üzere serbest bırakılır. α azaldıkça yeniden oluşturulacak problem penceresi sayısı artacağından dolayı çözüm süresi artar. Bunun yanında α değeri yükseldikçe çözüm süresi azalırken, görevlerin yanlış istasyonlara atanması aynen kabul edildiği için çözüm kalitesi düşebilmektedir.

PKPP algoritmasının ikinci aşamasındaki S ve δ parametre değerleri de çözüm kalitesi ve çözüm süresi üzerinde etkilidir. S değeri arttıkça aynı anda ele alınan problem penceresi sayısı artacağından dolayı çözüm kalitesinde iyileşmeler sağlanabilmektedir. δ değeri ele alınan istasyon aralığını belirlediği için mümkün

olduğunca küçük belirlenirse, birinci aşama sonunda elde edilen çözüm iyileştirilebilir.

Özetle, PKPP algoritmasının parametre değerleri probleme özgü olarak değişmektedir. Örneğin 30 görevli bir problemde $L=15$ yüksek kalitede çözümler bulunurken, 100 görevli bir problem için düşük kalitede çözümlerin bulunmasına neden olur. Bundan dolayı her problem için ayrı bir parametre seti belirlenmiştir. *Roszieng* probleminden daha küçük boyutlu problemlerin en iyi çözümleri bir saniyeden az zamanda bulunabildiği için, geriye kalan her problemin belirlenen parametre setleri [Ek-1](#)'de verilmiştir.

6.5. PKPP Algoritmasının Performansının Değerlendirilmesi

Önerilen tamsayılı programlama modeli ile küçük boyutlu test problemlerinin en iyi çözümleri makul sürelerde bulunmuştur. Fakat, M-PMHD probleminin NP-zor yapısı nedeniyle, büyük boyutlu test problemlerinin çoğunda verilen süresi içerisinde herhangi bir uygun çözüm bulunamamıştır. PKPP algoritmasının performansının değerlendirilmesi amacıyla, küçük boyutlu test problemlerinin tamsayılı karar modeli ile bulunan en iyi değerleri kullanılırken, orta ve büyük boyutlu test problemlerinin toplam işgücü maliyeti için bulunan alt sınır değeri kullanılmıştır. Bütün M-PMHD problemleri PKPP algoritması ile çözülmüştür. Elde edilen çözümler [çizelge 6.4 – çizelge 6.6](#)'de verilmiştir. Çizelgelerin üçüncü sütununda [bölüm 4.7](#)'de sunulan algoritmana kullanılarak bulunan toplam işgücü maliyetinin alt sınır değeri verilmiştir. Dördüncü sütunda M-PMHD probleminin elde edilen çözümündeki işyeri (operatör) sayısı, beşinci sütunda ise toplam işgücü maliyeti değeri verilmektedir. Yedinci sütunda ise, PKPP algoritması ile elde edilen toplam işgücü maliyeti ile alt sınır değeri arasındaki farkın alt sınıra oranı verilmektedir. Bu değer alt sınır değerinin en iyi değer olduğu varsayımıyla, PKPP algoritmasının bulduğu sonucun en iyi değere ne kadar yakınsadığını göstermektedir. Tabiki alt sınır değeri hesaplanırken görevlerin aralarındaki öncelik ilişkileri ve görevlerin bölünemezlik ilkesi ihlal edildiği için, problemin en iyi değerinin alt sınır değerinden daha yüksek olacağı açıktır.

Çizelge 6.4. PKPP algoritması ile küçük boyutlu problemler için elde edilen sonuçlar

Problem	C	Alt sınır / En iyi değer	PKPP Algoritması			
			K	TİM	Süre (sn)	%
Merten (7 - 6)	9	279	7	279	0,19	0
	11	253	5	253	0,11	0
	13	286	5	286	0,11	0
	17	306	4	306	0,06	0
Jaeschke (9 - 8)	9	486	8	486	0,27	0
	11	506	7	506	0,36	0
	13	507	6	507	0,34	0
	15	525	5	525	0,17	0
	17	510	4	510	0,14	0
Jackson (11 - 10)	8	848	13	848	5,30	0
	10	840	9	840	4,18	0
	13	819	7	819	2,70	0
	15	810	6	810	2,62	0
	19	836	5	836	1,79	0
Roszieng (25 - 24)	14	4 897	18	4 970	352	1,4
	16	4 944	16	4 976	360	0,6
	17	4 947	15	4 964	411	0,3
	22	5 170	12	5 170	241	0
	30	5 340	9	5 340	124	0
Sawyer (30 - 28)	25	14 600	26	15 025	1 438	2,9
	27	14 742	25	15 444	376	4,7
	30	14 850	22	15 180	276	2,2
	36	14 868	18	15 084	276	1,5
	41	14 842	16	15 211	797	2,5
	54	14 850	12	15 336	726	3,3

PKPP algoritmasının performansını belirlemek amacıyla [çizelge 6.4](#) ve [çizelge 6.5](#)'de gösterilen M-PMHD problemlerinin elde edilen çözümleri incelendiğinde, 25 tane küçük boyutlu problemin 16 tanesinin; 27 tane orta büyüklükteki problemin ise 6 tanesinin en iyi çözümlerinin bulunduğu görülmektedir. En iyi çözümleri bulunamayan M-PMHD problemlerin ise, PKPP algoritmasıyla alt sınır değerine çok yakın bir çözümün bulunduğu görülmektedir. Yukarıda belirtilen nedenlerden dolayı problemin en iyi değerinin alt sınır değerinden daha yüksek bir değer olması beklendiğine göre, PKPP algoritmasıyla elde edilen çözümlerin en iyi çözüm olma ihtimali vardır. PKPP algoritmasıyla elde edilen çözümler en iyi çözüm olmasa bile en iyi değere çok yakın sonuçlardır.

Çizelge 6.5. PKPP algoritması ile orta boyutlu problemler için elde edilen sonuçlar

Problem	C	Alt sınır / En iyi değer	PKPP Algoritması			
			K	TİM	Süre (sn)	%
Kilbridge (45 - 43)	57	37 791	20	39 216	828	3,8
	79	37 446	14	39 026	408	4,2
	92	37 352	12	39 560	1 163	5,9
	110	37 290	10	40 040	2 063	7,4
	138	37 674	8	40 434	457	7,3
	184	41 216	6	41 216	628	0
Hahn (53 - 51)	2 004	1 238 472	14	1 238 472	4 029	0
	2 338	1 248 492	12	1 248 492	1 038	0
	2 806	1 257 088	10	1 257 088	1 152	0
	3 507	1 297 590	8	1 297 590	613	0
	4 676	1 342 012	6	1 342 012	33	0
Tonge (70 - 66)	160	359 840	45	380 160	809	5,6
	168	358 680	44	386 568	3 522	7,8
	207	354 177	35	383 778	658	8,4
	234	363 168	31	384 228	3 622	5,8
	270	359 100	27	387 450	1 930	7,9
	293	357 753	25	392 620	345	9,7
Wee-mag (75 - 71)	28	179 424	123	188 272	1 775	4,9
	29	183 947	123	194 822	1 146	5,9
	31	192 293	121	205 375	1 243	6,8
	33	203 907	120	216 051	1 180	5,9
	34	208 624	119	220 660	1 107	5,8
	41	241 859	116	259 694	624	7,4
	42	234 864	107	244 608	656	4,1
	43	219 386	99	230 953	786	5,3
	49	164 150	63	170 422	4 286	3,8
	54	166 428	61	180 414	7 080	8,4

PKPP algoritmasıyla elde edilen sonuçlara göre, alt sınır değerinden en fazla sapma oranı %13,8 olarak belirlenmiştir. PKPP algoritması çözüm süresi bakımında incelendiğinde, meta-sezgisel algoritmalarla göre daha uzun süre çalışmaktadır. Fakat, büyük boyutlu M-PMHD problemleri için elde edilen yüksek kalitede çözümler düşünüldüğünde, PKPP algoritmasının çözüm süresinin katlanılabılır bir süre olduğu görülmektedir. Orta ve büyük boyutlu M-PMHD problemlerin 86 400 sn içerisinde önerilen tamsayı karar modeli ile bir uygun çözümü bile bulunamazken, PKPP algoritması yüksek kalitede çözümler bulabilmektedir.

Çizelge 6.6. PKPP algoritması ile büyük boyutlu problemler için elde edilen sonuçlar

Problem	C	Alt sınır / En iyi değer	PKPP Algoritması			
			K	TİM	Süre (sn)	%
Arcus1 (83 - 79)	3 786	9 177 264	40	9 699 732	2 998	5,7
	3 985	9 105 725	38	9 775 205	4 557	7,4
	4 206	9 173 286	37	9 964 014	738	8,6
	4 454	9 233 142	35	10 021 500	7 437	8,5
	4 732	9 175 348	32	9 818 900	630	7,0
	5 853	9 195 063	26	9 996 924	645	8,7
	6 842	9 428 276	22	10 016 688	1 803	6,2
	7 571	9 448 608	20	10 220 850	962	8,2
	8 412	9 522 384	18	10 170 108	1 161	6,8
	10 816	9 388 288	14	10 372 544	665	10,5
Lutz2 (89 - 85)	11	65 054	89	70 323	8 606	8,1
	12	64 992	82	70 632	3 302	8,7
	13	64 597	74	70 330	2 749	8,9
	14	65 142	70	69 916	902	7,3
	15	64 395	65	69 630	811	8,1
	16	64 128	60	69 408	1 853	8,2
	17	64 770	57	69 360	4 850	7,1
	19	64 296	51	69 711	3 222	8,4
	20	65 960	50	71 820	1 411	8,9
Lutz3 (89 - 85)	75	224 475	44	238 050	5 204	6,0
	79	223 886	43	243 557	2 119	8,8
	83	225 096	40	241 862	6 709	7,4
	87	223 764	39	246 123	1 991	10,0
	92	223 468	37	246 100	2 779	10,1
Mukherje (94 - 90)	176	550 176	49	613 008	534	11,4
	183	559 797	47	610 122	5 959	9,0
	192	563 328	44	605 184	1 372	7,4
	201	556 167	42	609 432	630	9,6
	211	551 132	41	611 478	1 475	10,9
	222	555 222	39	619 158	757	11,5
	234	564 408	37	616 590	2 818	9,2
	248	557 256	35	620 496	1 393	11,3
	263	567 817	32	612 527	3 290	7,9
	281	569 306	31	620 448	9 690	9,0
	301	578 221	28	614 341	1 300	6,2
	324	584 496	27	635 688	127	8,8
	351	583 362	25	629 694	4 041	7,9

Çizelge 6.6. (Devam) PKPP algoritması ile büyük boyutlu problemler için elde edilen sonuçlar

Problem	C	Alt sınır / En iyi değer	PKPP Algoritması			
			K	TİM	Süre (sn)	%
Arcus (111 - 107)	5 785	25 378 795	54	26 952 315	8 003	6,2
	6 016	25 561 984	53	27 505 152	4 063	7,6
	6 267	25 481 622	52	27 919 485	1 128	9,6
	6 540	24 976 260	48	27 435 300	910	9,8
	6 837	25 194 345	48	28 236 810	1 013	12,1
	7 162	24 400 934	45	27 767 074	7 635	13,8

Ayrıca, PKPP algoritmasıyla önerilen tamsayılı programlama modeli ile IBM ILOG CPLEX 12.4 yazılımı yardımıyla büyük boyutlu M-PMHD problemlerin 86 400 sn gibi büyük bir süre zarfında elde edilen çözümlerden daha yüksek kalitede çözümleri daha kısa zamanda elde edilebildiği görülmüştür.

6.6. M-ÇPPG Algoritmasının En Uygun Parametre Setinin Belirlenmesi

Bu bölümde, M-PMHD probleminin çözümü için geliştirilen M-ÇPPG algoritmasında kullanılan parametrelerin en uygun kombinasyonunu ve bu parametreleri belirlemek için yapılan deneysel çalışmalar ve elde edilen sonuçlar incelenmektedir. Bir GA'nın performansı kullanılan *seçim mekanizması* ve *genetik operatörler* ile birlikte *popülasyon büyüklüğü*, *çaprazlama oranı*, *mutasyon oranı* ve *durdurma kriteri* gibi kontrol parametrelerine bağlıdır. Bu nedenle, bir problemin çözümü için GA kullanılmadan önce bu parametrelerin en iyi kombinasyonunun belirlenmesi gerekir.

Önerilen M-ÇPPG algoritmasının parametre setlerini ve etkinliğini belirlemek amacıyla deneysel çalışmalar yapılmıştır. Geliştirilen melez paralel genetik algoritmanın etkinliği en iyi çözümü bilinen test problemleri üzerinde buldukları çözümlerin, bu problemlerin en iyi çözümünden sapma oranına göre değerlendirilmiştir. En iyi değerden sapma oranı [Eş. 6.2](#) ile hesaplanmaktadır:

$$SO = \frac{GAUD - OPT}{OPT} * 100 \quad (6.2)$$

GAUD: Melez genetik algoritma ile elde edilen çözümün uygunluk değeri

OPT: Ele alınan problemin en iyi değeri

SO: En iyi değerden sapma oranı (%)

Önerilen melez paralel genetik algoritmanın en iyi parametre setinin belirlenebilmesi için yapılan deneysel çalışmalarda kullanılan test problemlerinin ana kütle temsil etmesi önemlidir. Bu amaçla, kaynaklarda iyi bilinen küçük, orta ve büyük boyutlu problem setlerinden birer tane test problemi seçerek, deneysel çalışmalar bu seçilen problemler üzerinde yapılmıştır. Deneysel çalışmalarda, küçük boyutlu *Roszieng*, orta boyutlu *Hahn* ve büyük boyutlu *Lutz2* isimli test problemleri kullanılmıştır. Problemlere ait karakteristikler ve bölüm 4.2’de anlatılan tamsayılı karar modeli ile elde edilen en iyi çözümleri (alt sınır değerleri) çizelge 6.7’de ayrıntılı olarak verilmiştir. *Roszieng* ve *Hahn* isimli test problemlerin en iyi değerleri önerilen tamsayılı karar modeli ile bulunabilirken, *Lutz2* isimli test probleminin en iyi çözümü elde edilememiştir. Bundan dolayı, en iyi parametre setinin belirlenmesi aşamasında sapma oranları (SO) hesaplanırken, *Lutz2* isimli test problemi için bölüm 4.7’de anlatılan toplam işgücü maliyeti için alt sınır değeri bulan algoritma ile elde edilen değer kullanılmıştır.

Çizelge 6.7. Deneysel çalışmalarda kullanılan test problemlerinin karakteristikleri

Problem	C	Görev sayısı (hat 1 – hat 2)	En iyi değer / Alt sınır
Roszieng	30	25 – 24	5 340
Hahn	2 004	53 – 51	1 238 472
Lutz2	11	89 – 85	65 054

Önerilen M-ÇPPG algoritmasında iki farklı popülasyon çeşiti ve bu popülasyonlar üzerinde çalışan iki farklı genetik algoritma (*standart genetik algoritma*, *denge durumlu genetik algoritma*) kullanılmaktadır. Önerilen melez paralel genetik

algoritma ile bir problemin çözüm uzayındaki iyi kaliteli çözümler araştırılırken, aramaya asıl yön veren yönetici popülasyon olduğu için, deneysel çalışmalar bu popülasyon üzerinde çalışan standart genetik algoritma için yapılmıştır. Standart genetik algoritmanın parametre değerleri ile köle popülasyonlar üzerinde çalışan denge durumlu genetik algoritmaların parametre değerleri aynı kabul edilmiştir.

M-ÇPPG algoritmasının, yönetici popülasyon üzerinde çalışan standart genetik algoritmanın, *popülasyon büyüklüğü*, *çaprazlama oranı*, *mutasyon oranı* ve iletişim operatörünün kontrol ettiği *benzerlik oranı* olmak üzere 4 farklı parametresi mevcuttur. Parametre setleri arasında en iyisinin seçilmesi için yapılan deneysel çalışmalarda, M-ÇPPG algoritmasının diğer tüm faktörleri sabit alınmıştır. Örneğin başlangıç popülasyonunun oluşturulması için PKPP algoritmasının birinci aşaması ile birlikte [Şekil 5.13](#)'te verilen algoritma kullanılmaktadır. [Şekil 5.13](#)'te verilen algoritmanın adımları arasında atanabilir görevler kümesinden herhangi bir görev rasgele seçildiği için farklı başlangıç popülasyonları oluşturulabilir. Fakat tüm alternatif parametre setlerinin değerlendirilmesi sürecinde, önerilen melez paralel genetik algoritma koşturulurken deney ortamının aynı koşullarda olması gerekmektedir. Farklı başlangıç popülasyonlarının en iyi parametre seçimini etkilememesi için, başlangıç popülasyonu bir kere üretilerek her parametre setinin değerlendirilmesi için aynı başlangıç popülasyonu kullanılmıştır.

Mevcut yönetici popülasyondan bir sonraki jenerasyona ait popülasyonunun üretilmesi amacıyla seçim operatörü olarak *elitizm* ve *turnuva seçim yöntemi* birlikte kullanılmıştır. Her yeni popülasyon oluşturulurken, yönetici popülasyonun en iyi çözümlerinin saklandığı elit listesinde bulunan iki çözüm hiçbir çaprazlama ve mutasyon operatörüyle değiştirilmeden aynen yeni nesile aktarılır. Buradaki amaç, eldeki iyi çözümlerin korunmasıdır. Deneysel çalışmalarda, elit listesinde bulunan çözüm sayısı 2 olarak alınmıştır ve tüm popülasyonlar için aynı ve sabittir. Yeni popülasyonu oluşturacak diğer yeni kromozomlar (çözümler) ise, elit listedeki herhangi bir çözüm ile turnuva seçim yöntemiyle rasgele seçilen 4 kromozom (çözüm) içerisinden en iyisinin seçilmesiyle birlikte çaprazlama ve mutasyon operatörlerinin uygulanması sonucu elde edilir. Diğer bir ifadeyle, çaprazlama

operatörünün uygulanacağı iki kromozomun birisi elit listedeki çözümlerden alınırken, diğeri ise geriye kalan çözümlerden turnuva seçim yöntemiyle belirlenir. Seçilen kromozomlara uygulanacak çaprazlama ve mutasyon operatörleri aynı olmakla birlikte, çaprazlama ve mutasyon oranları deneysel çalışmalarda farklı alınmıştır.

İletişim operatörünün görevi popülasyonlar arasında çözüm aktarma işlemini sağlamaktır. Bu görevi yerine getirirken köle popülasyonların *benzerlik oranlarını* dikkate alır. İletişim operatörü, herhangi bir köle popülasyonunun ortalama benzerlik oranı belirli eşik değerinin üzerine çıktığı zaman popülasyonlar arasında çözüm aktarması yapar. Bu nedenle, benzerlik eşik değeri önemli bir parametredir ve deneysel çalışmalarda farklı değerleri için ele alınmıştır.

PKPP algoritmasının ikinci aşaması ile birlikte CPLEX yazılımı kullanılarak belirli aralıklarla her popülasyona ait elit listesinde bulunan çözümlerin iyileştirilmesi sağlanmaktadır. Deneysel çalışmaların yapılmasındaki amaç, önerilen melez paralel genetik algoritmanın daha iyi kalitede sonuçlar elde edebilmesi için en iyi parametre setinin belirlenmesidir. PKPP algoritmasının elde ettiği başarılı sonuçlar en iyi parametre setinin belirlenmesi sürecini olumsuz yönde etkileyebilir veya yanlış parametre setinin belirlenmesine neden olabilir. Bu yüzden, deneysel çalışmalarda önerilen M-ÇPPG algoritmasında PKPP algoritmasının ikinci aşamasını çalıştıran modülü kapatılmıştır. Deneysel çalışmaya dahil edilen parametreler ve bunların düzeyleri [çizelge 6.8](#)'de verilmektedir.

Çizelge 6.8. M-ÇPPG algoritmasının parametre değerleri

Parametreler (Faktörler)		Düzeyler	
Popülasyon büyüklüğü ($ P $)	50	100	200
Çaprazlama oranı (P_c)	0,6	0,7	0,8
Mutasyon oranı (P_m)	0,1	0,2	0,3
Benzerlik oranı ($\bar{B}$)	0,4	0,6	0,8

En iyi parametre setinin belirlenmesi için iki alternatif test yaklaşımı kullanılabilir. Birinci yaklaşımda, bir parametre sürekli değiştirilirken, diğer parametreler sabit tutulmaktadır. Her parametre diğer parametrelerin sabitliğinde değiştirilerek denenmektedir. Bu yöntemle, her parametrenin en iyi değerine ulaşılması amaçlanmaktadır. İkinci yaklaşım ise, parametrelerin farklı değerlerinin birbirleriyle farklı kombinasyonlarda birleşmesiyle ortaya çıkan sonuçların karşılaştırılmasına dayalıdır. Genetik algoritmalarda bir parametredeki değişikliğin diğer parametrelerin performansı üzerinde de etkili olabileceği için, birinci yaklaşımın daha az gerçekçi sonuçlar ortaya çıkaracağı düşünülmüştür. Bu sebeple bu çalışmada ikinci yaklaşımın kullanılması tercih edilmiştir.

Algoritmanın en iyi performans göstereceği parametre setini belirlemek için *tam faktöryel tasarım* kullanılmıştır. Çizelge 6.8’de görülen parametrelerin alt seviyeleriyle birlikte tüm kombinasyonları birer kere yapabilmek için $3^4=81$ adet deneye gerek vardır. Her parametre setinin rassallık içeren deneylerde farklı performans sergileyebilmesi nedeniyle, parametre setleri birden fazla sayıda denenmeli ve ortaya çıkan sonuçların ortalaması dikkate alınmalıdır. Bu nedenle, M-ÇPPG algoritması parametrelerinin oluşturduğu her kombinasyon düzeyinde diğer tüm faktörler sabit alınarak ve aynı başlangıç popülasyonu üzerinde toplamda 3 farklı boyuttaki problemler için 5 deneme yapılmıştır. Buna göre algoritmanın en iyi parametre setinin belirlenmesi için $1\ 215 (= 3 \times 3^4 \times 5)$ deneme gerçekleştirilmesi amaçlanmıştır.

Geliştirilen M-ÇPPG algoritması DevC++ derleyicisi kullanılarak C++ programlama dilinde kodlanmış ve en iyi parametre setini belirlemek amacıyla yapılan deneysel çalışmalar Intel Pentium 4 3.0 GHz, 3.04 Ghz iki işlemcili ve 4 GB Ram özelliğine sahip bir bilgisayar sisteminde yapılmıştır. Bu denemelerde, performans ölçütü olarak en iyi değerden sapma oranı (SO) kullanılmıştır. M-ÇPPG algoritması tüm olası parametre setleri için ayrı ayrı seçilen test problemleri üzerinde 10 000 iterasyon çalıştırılarak en iyi parametre seti belirlenmiştir.

Ele alınan 3 farklı problemin deneysel çalışmalarından elde edilen sonuçlar [Ek-2](#)'de yer alan [çizelge 8.1](#) – [çizelge 8.3](#)'te verilmiştir. Seçilen faktörlerin algoritma performansı üzerinde etkin olup olmadığını belirlemek için varyans analizi (ANOVA) kullanılmıştır. *Roszieng*, *Hahn* ve *Lutz2* problemlerin deney sonuçlarına MINITAB 15 yazılımı yardımıyla uygulanan varyans analizi (ANOVA) sonuçları sırasıyla [Ek-4](#), [Ek-6](#) ve [Ek-8](#)'de verilmiştir. ANOVA tablolarındaki P değerleri incelendiğinde, $\alpha = 0.05$ anlamlılık düzeyinde *Roszieng* problemi için sadece *popülasyon büyüklüğü*, *Hahn* problemi için sadece *popülasyon büyüklüğü* ve son olarak *Lutz2* problemi için ise *popülasyon büyüklüğü* ve *mutasyon oranı* faktörlerinin algoritmanın performansı üzerinde etkin olduğu görülmektedir.

Varyans analizinden elde edilen sonuçlara göre algoritmanın parametreleri arasında karmaşık bir etkileşim vardır. Üçlü ve dörtlü ortak etkilerden hiçbiri anlamlı değil iken, ikili olan etkilerden sadece *Lutz2* probleminde *popülasyon büyüklüğü* – *çaprazlama oranı* ve *popülasyon büyüklüğü* – *benzerlik oranı* arasındaki ortak etkiler anlamlı bulunmuştur. *Lutz2* probleminin *popülasyon büyüklüğü* ve *mutasyon oranı* parametreleri kendi başlarına anlamlı iken, birlikte bir anlam ifade etmemektedir. Bunun yanında, *benzerlik oranı* kendi başına bir anlam ifade etmez iken, *popülasyon büyüklüğü* ile birlikte anlamlıdır. Dolayısıyla, en iyi sonuç veren parametre setini belirlemek için, yukarıdaki testlere ek olarak, tüm parametre kombinasyonlarının ortalama sapma üzerindeki etkisi incelenmiştir.

Her problem ve her parametre seti için yapılan 5 deneme sonucunda elde edilen *ortalama toplam işgücü maliyet* değerleri ve *en iyi değerden ortalama sapma oranları* [Ek-2](#)'deki [çizelge 8.1](#) – [çizelge 8.3](#)'de verilmiştir. Bununla birlikte, her parametre seti için hesaplanan ortalama toplam işgücü maliyeti değerlerinin nasıl değiştiğini gösteren grafik [Ek-3](#)'de yer alan [şekil 6.1](#) – [6.3](#)'de verilmiştir (Denemeler senaryo sırasına göre listelenmiştir). Ayrıca, *Roszieng*, *Hahn* ve *Lutz2* problemleri için MINITAB 15 yazılımı kullanılarak, parametrelerin her seviyesinin ortalama toplam işgücü maliyeti üzerindeki etkilerini gösteren grafikler çizdirilmiş ve [Ek-5](#)'te verilmiştir.

Ek-2’de verilen çizelge 8.1 – çizelge 8.3’deki deney sonuçları, bu sonuçlardan elde edilen ortalama toplam işgücü maliyeti değerlerini gösteren grafikler (Bkz. Ek-3’de yer alan şekil 6.1 – 6.3) ve MINITAB yazılımından elde edilen parametrelerin etkileşim grafikleri incelendiğinde (Bkz. Ek-5, Ek-7 ve Ek-9), en küçük en iyi değerden sapma değeri 53 nolu parametre setinde elde edilmiştir. Elde edilen deney sonuçlarına göre M-PMHD problemlerinin çözümünde kullanılacak olan M-ÇPPG algoritmasının en iyi parametre seti çizelge 6.9’da verilmiştir. En iyi sonuçların elde edildiği parametre değerleri şu şekildedir: popülasyon büyüklüğü (100), çaprazlama oranı (0,8), mutasyon oranları (0,3), benzerlik oranı (0,6). Ayrıca, 71 numaralı parametre setinde de (popülasyon büyüklüğü: 200; çaprazlama oranı: 0,8; mutasyon oranları: 0,3; benzerlik oranı: 0,6) yaklaşık olarak aynı sonuçları elde edilmesine rağmen, algoritmanın çalışma zamanının uzamaması açısından, 53 numaralı parametre seti bundan sonra çözülecek olan tüm problemlerde kullanılacaktır.

Çizelge 6.9. M-ÇPPGA’sının en iyi parametre seti

Parametreler	Değerler
Popülasyon büyüklüğü	Dinamik; üst sınır = 100
Popülasyon sayısı	Dinamik
Seçim yöntemi	Turnuva yöntemi; uzunluk = 4
Çaprazlama oranı (Yönetici popülasyon)	0,8
Çaprazlama oranı (Köle popülasyonlar)	0,8
Mutasyon oranı (Yönetici popülasyon)	0,3
Mutasyon oranı (Köle popülasyonlar)	0,3
Elit list uzunluğu (Yönetici popülasyon)	2
Elit list uzunluğu (Köle popülasyonlar)	2
İterasyon sayısı	10 000
CPLEX kullanım sıklığı	2 000
Benzerlik oranı	0,6

6.7. M-ÇPPG Algoritmasının Performansının Değerlendirilmesi

Bölüm 6.6’da yapılan deneysel çalışmalar ile belirlenen en iyi parametre setini (Bkz. Çizelge 6.9) kullanarak, önerilen M-ÇPPG algoritması ile PKPP algoritmasıyla çözülen M-PMHD problemleri çözülmüştür. Elde edilen M-PMHD problemlerinin çözümleri çizelge 6.10 – çizelge 6.12’de verilmiştir. Çizelgelerin üçüncü sütunlarında bölüm 4.7’de anlatılan algoritma ile hesaplanan toplam işgücü maliyeti için bir alt sınır değeri verilmiştir. Önerilen tamsayılı karar modeli ile verilen zaman dilimi içerisinde (86 400 sn) problemin en iyi değeri bulunamadığında, önerilen M-ÇPPG algoritmasının elde ettiği çözümler alt sınır değerleriyle kıyaslanmaktadır.

Çizelge 6.10. M-ÇPPG algoritması ile küçük boyutlu problemler için elde edilen sonuçlar

Problem	C	Alt sınır / En iyi değer	M-ÇPPG Algoritması			
			K	TİM	Süre (sn)	SO (%)
Merten (7 - 6)	9	279	7	279	0,1	0
	11	253	5	253	0,1	0
	13	286	5	286	0,1	0
	17	306	4	306	0,1	0
Jaeschke (9 - 8)	9	486	8	486	0,1	0
	11	506	7	506	0,1	0
	13	507	6	507	2	0
	15	525	5	525	0,1	0
	17	510	4	510	4	0
Jackson (11 - 10)	8	848	13	848	1	0
	10	840	9	840	0,1	0
	13	819	7	819	0,1	0
	15	810	6	810	0,1	0
	19	836	5	836	1	0
Roszieng (25 - 24)	14	4 897	18	4 970	53	1,4
	16	4 944	16	4 944	58	0
	17	4 947	15	4 964	65	0,3
	22	5 170	12	5 170	69	0
	30	5 340	9	5 340	74	0
Sawyer (30 - 28)	25	14 600	26	15 075	431	3,3
	27	14 742	25	15 471	320	4,9
	30	14 850	22	15 420	296	3,8
	36	14 868	19	15 228	242	2,4
	41	14 842	16	15 334	335	3,3
	54	14 850	12	15 498	427	4,4

M-ÇPPG algoritması ile en iyi değeri bilinen 25 küçük boyutlu problemin 17'sinin ve 27 tane orta boyutlu problemin ise 6 tanesinin en iyi çözümlerinin bulunduğu görülmektedir. En iyi çözümü bulunamayan büyük boyutlu problemlerin elde edilen çözümleri incelendiğinde, alt sınır değerine kıyasla en az % 4.3 en fazla % 10,7 hata payı olduğu görülmektedir. M-PMHD problemi için [bölüm 4.7](#)'de anlatılan alt sınır algoritması ile elde edilen toplam işgücü maliyeti değeri en iyi değere çok yakın bir değer değildir. Problemin en iyi değerinin alt sınır değerinin üstünde olduğu bilindiğine göre, bu hata payı daha küçüktür.

Çizelge 6.11. M-ÇPPG algoritması ile orta boyutlu problemler için elde edilen sonuçlar

Problem	C	Alt sınır / En iyi değer	M-ÇPPG Algoritması			
			K	TİM	Süre (sn)	SO (%)
Kilbridge (45 - 43)	57	37 791	20	39 102	866	3,5
	79	37 446	14	38 789	876	3,6
	92	37 352	12	39 284	573	5,2
	110	37 290	10	39 600	993	6,2
	138	37 674	8	40 572	1 068	7,7
	184	41 216	6	41 216	376	0
Hahn (53 - 51)	2 004	1 238 472	14	1 238 472	1 246	0
	2 338	1 248 492	12	1 248 492	624	0
	2 806	1 257 088	10	1 257 088	764	0
	3 507	1 297 590	8	1 297 590	154	0
	4 676	1 342 012	6	1 342 012	49	0
Tonge (70 - 66)	160	359 840	45	377 600	551	4,9
	168	358 680	44	386 064	356	7,6
	207	354 177	35	383 985	453	8,4
	234	363 168	31	387 036	539	6,6
	270	359 100	27	383 940	910	6,9
	293	357 753	24	380 900	708	6,5
Wee-mag (75 - 71)	28	179 424	123	188 048	323	4,8
	29	183 947	123	194 590	307	5,7
	31	192 293	121	205 282	313	6,7
	33	203 907	119	214 698	303	5,3
	34	208 624	119	220 660	317	5,8
	41	241 859	116	259 612	757	7,3
	42	234 864	107	244 608	900	4,1
	43	219 386	98	227 943	251	3,9
	49	164 150	63	169 638	170	3,3
	54	166 428	60	178 362	223	7,1

Çizelge 6.12. M-ÇPPG algoritması ile büyük boyutlu problemler için elde edilen sonuçlar

Problem	C	Alt sınır / En iyi değer	M-ÇPPG Algoritması			
			K	TİM	Süre (sn)	SO (%)
Arcus1 (83 - 79)	3 786	9 177 264	40	9 741 378	1 579	6,1
	3 985	9 105 725	38	9 838 965	1 002	8,1
	4 206	9 173 286	36	9 875 688	5 051	7,7
	4 454	9 233 142	34	9 968 052	5 366	8,0
	4 732	9 175 348	32	9 951 396	397	8,5
	5 853	9 195 063	26	9 955 953	2 020	8,3
	6 842	9 428 276	22	9 982 478	1 016	5,9
	7 571	9 448 608	20	9 986 149	870	5,7
	8 412	9 522 384	18	10 161 696	621	6,7
	10 816	9 388 288	14	10 394 176	936	10,7
Lutz2 (89 - 85)	11	65 054	89	70 389	2 600	8,2
	12	64 992	82	69 888	2 541	7,5
	13	64 597	75	69 706	4 389	7,9
	14	65 142	70	69 622	3 638	6,8
	15	64 395	65	70 125	2 952	8,8
	16	64 128	61	69 824	4 864	8,8
	17	64 770	57	69 836	2 000	7,8
	19	64 296	52	69 521	4 545	8,1
	20	65 960	49	70 620	1 413	7,1
Lutz3 (89 - 85)	75	224 475	45	239 325	2 071	6,6
	79	223 886	42	239 449	2 594	7,0
	83	225 096	40	242 609	3 082	7,8
	87	223 764	38	241 512	4 474	7,9
	92	223 468	36	240 304	2 937	7,5
Mukherje (94 - 90)	176	550 176	48	595 408	4 352	8,2
	183	559 797	46	600 789	1 830	7,3
	192	563 328	44	597 120	1 934	6,0
	201	556 167	42	602 799	2 024	8,4
	211	551 132	40	603 671	1 441	9,5
	222	555 222	38	605 616	4 413	9,1
	234	564 408	36	605 826	6 091	7,3
	248	557 256	34	605 120	4 202	8,6
	263	567 817	32	610 949	1 437	7,6
	281	569 306	30	611 456	309	7,4
	301	578 221	28	609 525	4 684	5,4
	324	584 496	26	609 444	3 725	4,3
	351	583 362	24	614 601	1 807	5,4

Çizelge 6.12. (Devam) M-ÇPPG algoritması ile büyük boyutlu problemler için elde edilen sonuçlar

Problem	C	Alt sınır / En iyi değer	M-ÇPPG Algoritması			
			K	TİM	Süre (sn)	SO (%)
Arcus (111 - 107)	5 785	25 378 795	54	27 131 650	4 572	6,9
	6 016	25 561 984	53	27 096 064	3 793	6,0
	6 267	25 481 622	50	27 280 251	2 344	7,1
	6 540	24 976 260	48	27 023 280	1 090	8,2
	6 837	25 194 345	46	27 334 326	1 311	8,5
	7 162	24 400 934	43	26 957 768	3 766	10,5

Büyük boyutlu problemlerin matematiksel model ile 86 400 sn süresinde bir uygun çözüm bile bulanamadığı düşünülürse, önerilen M-ÇPPG algoritması kısa sürede yüksek kalitede çözümler bulan etkin bir sezgiseldir.

M-PMHD probleminin çözümü için önerilen M-ÇPPG ve PKPP sezgisel algoritmalarına ait genel değerlendirme [çizelge 6.13](#)'de sayısal verilerle özetlenmiştir. Genel olarak yorumlamak gerekirse, ele alınan problemlerin zorlukları ve görev sayılarının fazla olması dikkate alındığında, matematiksel modeller ile en iyi çözümlerini bulmak bir tarafa verilen süre içerisinde uygun bir çözüm dahi bulunamazken, önerilen her iki sezgisel algoritma kısa sürelerde yüksek kalitede çözümler vermektedir. [Çizelge 6.13](#)'deki sayısal verilere göre, M-ÇPPG algoritması ile toplamda 23 problemin en iyi çözümü elde edilirken, PKPP algoritmasına ile 22 problemin en iyi çözümü bulunmuştur. En iyi çözümü bulunan problemlerin içerisinde hiç büyük boyutlu M-PMHD problemi yoktur. En iyi çözümü bulunamayan problemler incelendiğinde; küçük boyutlu problemlerde PKPP algoritması alt sınır değerine daha yakın değerler elde ederken, orta ve büyük boyutlu problemlerde ise M-ÇPPG algoritması daha iyi değerler bulmaktadır. Diğer bir ifadeyle, M-ÇPPG algoritması ile PKPP algoritmasının performansları karşılaştırıldığında küçük boyutlu problemlerde PKPP alt sınır değere daha yakın çözümler bulmuştur. Küçük boyutlu problemlerde PKPP algoritmasının baskınlığı söz konusudur. Fakat, büyük boyutlu problemlerde M-ÇPPG algoritması hem çözüm kalitesi hem de çözüm süresi açısından daha iyi sonuçlar vermektedir.

Çizelge 6.13. PKPP ve M-ÇPPGA karşılaştırılması

	PKPP	M-ÇPPGA
En iyi çözümü bulunan problem sayısı		
<i>Küçük boyutlu problemler (25 problem)</i>	16	17
<i>Orta boyutlu problemler (27 problem)</i>	6	6
<i>Büyük boyutlu problemler (43 problem)</i>	0	0
En iyi çözümü bulunamamış ve alt sınıra daha yakın bulunan çözüm sayısı		
<i>Küçük boyutlu problemler (25 problem)</i>	6	1
<i>Orta boyutlu problemler (27 problem)</i>	2	16
<i>Büyük boyutlu problemler (43 problem)</i>	11	32
<i>Küçük boyutlu problemler için</i>		
En küçük sapma (SO%)	0,3	0,3
En büyük sapma (SO%)	4,7	4,9
Ortalama sapma (SO%)	2,2	2,6
<i>Orta boyutlu problemler için</i>		
En küçük sapma (SO%)	3,8	3,3
En büyük sapma (SO%)	9,7	8,4
Ortalama sapma (SO%)	6,3	5,8
<i>Büyük boyutlu problemler için</i>		
En küçük sapma (SO%)	5,7	4,3
En büyük sapma (SO%)	13,8	10,7
Ortalama sapma (SO%)	8,7	7,6

En iyi değerden (alt sınır değerinden) sapma oranları incelendiğinde görülmüştür ki, özellikle problem boyutu arttıkça M-ÇPPG algoritması PKPP algoritmasına göre daha iyi sonuçlar elde etmektedir. Ayrıca M-ÇPPG algoritması ile birlikte M-PMHD problemlerin en iyi çözümlerine çok yakın yüksek kalitede çözümleri PKPP algoritmasına göre daha kısa sürede bulunmuştur. M-ÇPPG algoritması çözüm kalitesi ve çözüm süresi bakımından daha iyi iken, bu algoritmanın en iyi parametre setinin belirlenmesi için deneysel çalışmaların yapılmak zorunda olunması ayrıca bir dezavantajdır.

7. SONUÇLAR VE ÖNERİLER

Bu tezde, [Gökçen ve ark. \(2006\)](#) tarafından ilk defa ele alınan PMHD problemi ile ilgili kaynaklara ve mevcut bakış açısına göre, paralel montaj hatlarının farklı özellikleri ve paralel montaj hatlarının eş zamanlı dengelenmesinde maliyet tabanlı bir bakış açısı dikkate alınarak, bu yeni problemler için yeni çözüm yöntemleri geliştirilmiştir. Bu bağlamda, MHD problemi kaynaklarına yeni problemler kazandırılmıştır ve bu yeni problemlerin temelini oluşturan *Maliyet tabanlı Paralel Montaj Hattı Dengeleme* (M-PMHD) problemi ilk defa bu tezde ayrıntılı olarak ele alınmıştır. M-PMHD probleminde, paralel hatların eş zamanlı dengelenmesinde dikkate alınan amaç, işyeri (operatör) sayısını enküçüklemek yerine toplam işgücü maliyetini enküçüklemek ele alınmıştır. M-PMHD probleminin en iyi çözümünü bulmak amacıyla *istasyon odaklı* ve *operatör odaklı* olmak üzere iki farklı tamsayılı programlama modeli önerilmiştir.

Paralel hatların her birinde montajı yapılacak olan ürünler belirlidir. Fakat, montajı yapılacak olan ürünlerin belirlenmiş olan hatlar dışında üretildiklerinde toplam işgücü maliyeti bu yeni atamalardan etkilenmektedir. Bu nedenle, PMHD probleminin temelinde ürünlerin hatlara atanması problemi de aynı zamanda ele alınmalıdır. Bu problem *Maliyet tabanlı Çok Ürünlü Paralel Montaj Hattı Dengeleme* (M-ÇÜPMHD) problemi olarak isimlendirilmiştir. M-ÇÜPMHD probleminin en iyi çözümünü bulmak amacıyla *istasyon odaklı* ve *operatör odaklı* olmak üzere iki farklı modelleme yaklaşımıyla birlikte tamsayılı programlama modelleri önerilmiştir.

Paralel hatların komşu istasyonları arasında açılan ortak işyerlerinde çalışan operatörlerin iş yüklerine karar vericiler tarafından belirlenen miktarda yürüme zamanı farkı ayrılmasına olanak tanıyan bir tamsayılı programlama modeli önerilmiştir. Bu ilave zaman sadece operatörlerin yürümek için harcadıkları zaman olarak algılanmamalıdır. Bu ilave zaman ayrıca, paralel montaj hatlarının avantajlarından birisi olan operatörler arasındaki iş birliğinin ve iletişim gücünün artırılması için, paralel hatlar arasında görevlerin yapılmasında ve yürümede

harcanan zamanın dışında operatörler tarafından ihtiyaç duyulan zamandır. Yürüme zamanlarının dikkate alındığı bu problem *Maliyet tabanlı Yürüme Zamanlı Paralel Montaj Hattı Dengeleme* (M–YZPMHD) problemi olarak adlandırılmıştır ve çözümü için iki farklı tamsayılı programlama modeli önerilmiştir.

M–PMHD probleminin en iyi görev atamalarına göre, paralel hatlarda çalışan operatörlerin ücret oranları da belirlenmektedir. Her görevin ücret oranının birbirinden farklı olması ve bir operatörün ücret oranını çalıştığı istasyona atanan görevlerin en büyük ücret oranı değerine göre belirlenmesi sonucunda, yan yana çalışan operatörlerin ücret oranları birbirine göre farklılık göstermektedir. Ücret oranlarındaki farklılıklar çalışanları husursuz edebileceği düşüncesiyle, bir arada çalışan operatörlerin ücret oranları arasındaki farkı enküçükleyecek tamsayılı doğrusal programlama model de geliştirilmiştir.

Bu çalışmada ayrıca, paralel hatların eş zamanlı dengelenmesi için önerilen tamsayılı programlama modelini temel alan bir *hedef programlama modeli* önerilmiştir. Bu model, paralel montaj hatlarının eş zamanlı dengelenmesinde sadece bir amacı eniylemek yerine, çok sayıda amaca uzlaşık çözümler üretmektedir.

M–PMHD problemlerinin NP-zor yapısı nedeniyle, bu problemlerin en iyi çözümlerini kısa sürelerde önerilen tamsayılı programlama yaklaşımlarıyla elde etmek neredeyse imkansızdır. Bu yüzden, M–PMHD problemi için *matematiksel model tabanlı etkin bir sezgisel algoritma* (PKPP algoritması) geliştirilmiştir. Bu sezgisel algoritma bir problemi belirli görev sayısından oluşan küçük alt problemlere bölerek, her alt problemin en iyi görev atamalarını önerilen tamsayılı programlama modeli ile bulmaktadır. PKPP algoritması yapısı gereği alt problemlerin en iyi çözümlerini kesin olarak bulduğu için açgözlü bir algoritma gibi davranmaktadır. PKPP algoritmasından başka, M–PMHD probleminin yüksek kaliteli çözümlerini bulmak amacıyla *melez çok popülasyonlu paralel genetik* (M–ÇPPG) *algoritma* da geliştirilmiştir. Önerilen sezgisel algoritmaların performansları kaynaklarda iyi bilinen PMHD problemleri üzerinde test edilmiştir. M–PMHD problemindeki görev

sayıları arttıkça M-ÇPPG algoritmasıyla, PKPP algoritmasına göre daha iyi çözümler elde edilmektedir.

Bu çalışma kapsamında önerilen her tamsayılı programlama modeli ve sezgisel algoritmalar bir örnek problem üzerinde uygulanmıştır. Elde edilen tüm M-PMHD problemlerinin en iyi çözümleri, [Gökçen ve ark. \(2006\)](#) tarafından PMHD problemi için önerdikleri tamsayılı programlama modelinin buldukları çözümlerle *toplam işgücü maliyeti* ve gerekli *operatör sayısı* açısından karşılaştırılmıştır. Ayrıca, M-PMHD problemlerindeki her hatta üretilen ürün için (M-MHD problemi) [Amen \(2006\)](#)'nin önerdiği tamsayılı programlama modeli uygulanarak, her hat birbirinden bağımsız şekilde maliyet tabanlı olarak dengelenmiştir. Her hattın bağımsız olarak dengelenmesi sonucunda elde edilen çözümler ile M-PMHD problemlerinin en iyi çözümleri karşılaştırılmıştır. Sonuç olarak, paralel montaj hatların maliyet tabanlı bir yaklaşımla eş zamanlı olarak dengelenmesinin, toplam işgücü maliyeti açısından büyük faydalar sağladığı görülmüştür.

Bu tez kapsamında, M-PMHD problemleri kapsamında dikkate değer birçok konu başlığı ele alınmıştır. M-PMHD problemlerinde her istasyonda çalışan operatörlerin her türlü görevi yapabilecek bilgi ve beceriye sahip oldukları varsayılmaktadır. Gerçek hayatta bu durum böyle olmadığı için, ileride modellemeyi daha genel ele alabilmek ve gerçek hayatı daha iyi yansıtabilmek amacıyla operatörlerin becerilerinin de dikkate alındığı bir modelleme çalışması yapılabilir. Ayrıca M-PMHD problemlerin çözümleri için kesin çözüm yöntemlerinden biri olan dal-sınır algoritması geliştirilebilir.

KAYNAKLAR

Aase, G. R., Schniederjans, M. J., Olson, J. R., "U-OPT: An Analysis of Exact U-shaped Line Balancing Procedures", *International Journal of Production Research*, 41 (17), 4185 – 4210 (2003).

Acar, N., "Üretim planlaması yöntem ve uygulamaları", *MPM Yayınları*, Ankara, 9 – 15 (1996).

Ajenblit, D. A., Wainwright, R. L., "Applying genetic algorithms to the U-shaped assembly line balancing problem", In: *Proceedings of the 1998 IEEE International Conference on Evolutionary Computation*, Anchorage, AK, 96 – 101 (1998).

Akagi, F., Osaki, H., Kikuchi, S., "A method for assembly line balancing with more than one worker in each station", *International Journal of Production Research*, 21: 755 – 770 (1983).

Amen, M., "An exact method for cost-oriented assembly line balancing", *International Journal of Production Economics*, 64: 187 – 195 (2000a).

Amen, M., "Heuristic methods for cost-oriented assembly line balancing: A survey", *International Journal of Production Economics*, 68: 1 – 14 (2000b).

Amen, M., "Heuristic methods for cost-oriented assembly line balancing: A comparison on solution quality and computing time", *International Journal of Production Economics*, 69: 255 – 264 (2001).

Amen, M., "Cost-oriented assembly line balancing: model formulations, solution difficulty, upper and lower bounds", *European Journal of Operational Research*, 168: 747 – 770 (2006).

Askin, R. G., Standridge, C. R., "Modelling and analysis of manufacturing systems", *John Wiley and Sons Inc.*, Singapore 10 – 15 (1993).

Askin, R. G., Zhou, M., "Parallel station heuristic for the mixed-model production line balancing problem", *International Journal of Production Research*, 35 (11): 3095 – 3105 (1997).

Back, T., "Generalized convergence models for tournament and (M,h) selection", In L. Eschelmann (Ed.), *Proceedings of the Sixth International Conference on Genetic Algorithms*, San Francisco: Morgan Kaufmann Publishers, 2 – 8 (1995).

Bagley, J. D., "The behavior of adaptive systems which employ genetic and correlation algorithms", Doktora tezi, *University of Michigan*, 30 – 35 (1967).

Bard, J. F., "Assembly line balancing with parallel workstations and dead time", *International Journal of Production Research*, 27 (6): 1005 – 1018 (1989).

Barricelli, N. A., "Symbiogenetic evolution processes realized by artificial methods", *Methodos*, 143 – 182 (1957).

Barricelli, N. A. "Numerical testing of evolution theories. Part II. Preliminary tests of performance, symbiogenesis and terrestrial life", *Acta Biotheoretica*, 16(3–4): 99 – 126 (1963).

Baybars, I., "A survey of exact algorithms for the simple assembly line balancing problem", *Management Science*, 32 (8): 909 – 932 (1986).

Beasley, D., Bull, D. R., Martin, R. R., "Reducing epistasis in combinatorial problems by expensive coding", In S. Forrest (Ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms*, San Mateo, CA: Morgan Kaufmann Publishers, 400 – 407 (1993).

Becker, C., Scholl, A., "A survey on problems and methods in generalized assembly line balancing", *European Journal of Operational Research*, 168 (3): 694 – 715 (2006).

Benzer, R., Gökçen, H., Çetinyokuş, T., Çerçioğlu, H., "A network model for parallel line balancing problem", *Mathematical Problems in Engineering*, Art. No. 10106, 1 – 12 (2007).

Bossert, W., "Mathematical optimization: are there abstract limits on natural selection?", In Moorehead, P. S., and Kaplan, M. M. (Eds.), *Mathematical Challenges to the Neo-Darwinian Interpretation of Evolution*, 35 – 46 (1967).

Boysen, N., Fliedner, M., Scholl, A., "Sequencing mixed-model assembly lines: survey, classification and model critique", *European Journal of Operational Research*, 192 (2): 349 – 373 (2009).

Bryton, B., "Balancing of a continuous production line", Yüksek Lisans tezi, *Northwestern University*, Evanston, IL., 5 – 16 (1954).

Buffa E., "Modern production management", *John Wiley and Sons Inc.*, 25 – 36 (1965).

Bukchin, J., Tzur, M., "Design of flexible assembly line to minimize equipment cost", *IIE Transactions*, 32 (7): 585 – 598 (2000).

Bukchin, J., Rubinovitz, J., "A weighted approach for assembly line design with station paralleling and equipment selection", *IIE Transactions*, 35 (1): 73 – 85 (2002).

Bukchin, Y., Rabinowitch, I., "A branch-and-bound based solution approach for the mixed-model assembly line-balancing problem for minimizing stations and task

duplication costs”, *European Journal of Operational Research*, 174(1): 492 – 508 (2006).

Buxey, G. M., “Assembly line balancing with multiple stations”, *Management Science*, 20 (6): 1010 – 1021 (1974).

Cantu-Paz, E., “Efficient and accurate parallel genetic algorithms”, Kluwer Academic Publishers, 58 – 75 (2000).

Chen, C. L., Vempatiz, V. S., Aljaber, N., “An application of gas for flow shop problems”, *European Journal of Operational Research*, 80: 389 – 396 (1995).

Cheng, C.H., Miltenburg, J., Motwani, J., “The effect of straight- and U-shaped lines on quality”, *IEEE Transactions on Engineering Management*, 47 (3):321 – 333. (2000).

Cohon, J. P., Hegde, S. U., Martin, W. N., Richards, D., “Punctuated equilibria: a parallel genetic algorithm”, In J. J. Grefenstette (Ed.), *Genetic Algorithms and Their Applications: Proceedings of the Second International Conference on Genetic Algorithms*, Lawrence Erlbaum Associates, 148 – 154 (1987).

Cohon, J. P., Martin, W. N., Richards, D., “Genetic algorithms and punctuated equilibria in VLSI”, In Schwefel, H-P., Manner, R. (Eds.), *Parallel Problem Solving from Nature*, Springer Verlag, 134 – 144 (1991).

Çerçioğlu, H., Özcan, U., Gökçen, H., Toklu, B., “Paralel Montaj Hattı Dengeleme Problemleri için Bir Tavlama Benzetimi Yaklaşımı”, *J. Fac. Eng. Arch. Gazi Univ*, 24 (2): 331 – 341 (2009).

Dar-El, E. M., Cothor, R. F., “Assembly line sequencing for model mix”, *International Journal of Production Research*, 13 (5): 463 – 477 (1975).

Elredge, N., Gould, S., “Punctuated equilibria: an alternative to phyletic gradualism”, In Schopf, T. J. M. (Ed.), *Models in Paleobiology*, 82 – 115 (1972).

Erel, E., Sarin, S. C., “A survey of the assembly line balancing procedures”, *Production Planning and Control*, 9 (5): 414 – 434 (1998).

Erel, E., Gökçen, H., “Shortest route formulation of mixed-model assembly line balancing problem”, *European Journal of Operational Research*, 116 (1): 194 – 204 (1999).

Erel, E., Sabuncuoglu, I., Aksu, B.A., “Balancing of U-type assembly systems using simulated annealing”, *International Journal of Production Research*, 39 (13): 3003 – 3015 (2001).

Fogel, L. J., Owens, A. J., Walsh, M. J., "Artificial intelligence through simulated evolution", *New York: John Wiley*, 15 – 38 (1966)

Fraser, A. S., "Simulation of genetic systems by automatic digital computers. I. Introduction", *Australian Journal of Biological Sciences*, 10: 484 – 491 (1957).

Freeman, D. R., Jucker, J. V., "The line balancing problem", *Journal of Industrial Engineering*, 18: 361 – 364 (1967).

Ghosh, S., Gagnon, R. J. "A comprehensive literature review and analysis of the design, balancing and scheduling of assembly systems", *International Journal of Production Research*, 27 (4): 637 – 670 (1989).

Goldberg, D. E., "Genetic Algorithms in Search Optimization and Machine Learning", *Addison Wesley*, 30 – 78 (1989).

Goldberg, D., "Genetic Algorithms in search, optimization and machine learning", *Addison-Wesley*, Reading, MA, 24 – 56 (1989).

Goldberg, D. E., "The race, the hurdle, and the sweet spot: Lessons from genetic algorithms for the automation of design innovation and creativity", In P. Bentley (Ed.), *Evolutionary Design by Computers*. San Mateo, CA: *Morgan Kaufmann Publishers*, 105 – 118 (1999).

Goodman, E. D., Averill, R. C., Punch, W. F., Eby, D. J., "Parallel genetic algorithms in the optimization of composite structures", East Lansing, MI, *Michigan State University* (1997).

Gökçen, H., Erel, E., "A goal programming approach to mixed-model assembly line balancing problem", *International Journal of Production Economics*, 48 (2): 177 – 185 (1997).

Gökçen, H., Erel, E., "Binary integer formulation for mixed-model assembly line balancing problem", *Computers and Industrial Engineering*, 34 (2–4): 451 – 461 (1998).

Gökçen, H., Ağpak, K., Gencer, C., Kizilkaya, E., "A shortest route formulation of simple U-type assembly line balancing problem", *Applied Mathematical Modelling*, 29 (4): 373 – 380 (2005).

Gökçen, H., Ağpak, K., "A goal programming approach to simple U-line balancing problem", *European Journal of Operational Research*, 171 (2): 577 – 585 (2006).

Gökçen, H., Ağpak, K., Benzer, R., "Balancing of parallel assembly lines", *International Journal of Production Economics*, 103: 600 – 609 (2006).

Grefenstette, J. J., "Parallel adaptive algorithms for function optimization". Tech. Rep. No. CS-81-19, *Vanderbilt University, Computer Science Department*, Nashville, TN, 35 – 46 (1981).

Grefenstette, J. J., "Optimization of control parameters for genetic algorithms", *IEEE Transactions on Systems, Man and Cybernetics*, 161 (1), 122 – 128 (1986).

Grosso, P. B., "Computer simulations of genetic adaptation: Parallel sub-component interaction in a multilocus model", Doktora tezi, *The University of Michigan* (1985).

Helgeson, W. P., Birnie, D.P., "Assembly line balancing using the ranked positional weight technique", *Journal of Industrial Engineering*, 12 (6): 384 – 398, (1961).

Holland, J. H. "Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence", *University of Michigan Press*, 5 – 26 (1975).

Hwang R. K, Katayama H., Gen M. "U-shaped assembly line balancing problem with genetic algorithm", *International Journal of Production Research*, 46: 4637 – 4649 (2008).

Jin, M., Wu, S. D., "A new heuristic method for mixed model assembly line balancing problem", *Computers and Industrial Engineering*, 44 (1): 159 – 169 (2002).

Johnson, R. V., "A branch and bound algorithm for assembly line balancing with formulation irregularities", *Management Science*, 29 (11): 1309 – 1324 (1983).

Kapanoglu, M., "Genetics-based concurrent planning and scheduling for flexible and modular manufacturing systems", Doktora tezi, *Department of Industrial and Management Systems Engineering, College of Engineering, University of South Florida*, Tampa, Florida, 57 – 85 (1997).

Kapanoglu, M., Miller, W.A., Sunol, A.K., "Concurrent planning and scheduling of flexible manufacturing systems by using parallel genetic algorithms", In *Proceedings of the ICC&IE/IEMS, 2001 Joint Meeting*, Cocoa Beach, Florida, 780 – 785 (2001).

Kara, Y., Özcan, U., Peker, A., "An approach for balancing and sequencing mixed-model JIT U-lines", *International Journal of Advanced Manufacturing Technology*, 32 (11–12): 1218 – 1231 (2007a).

Kara, Y., Özcan, U., Peker, A., "Balancing and sequencing mixed-model JIT U-lines with multiple objectives", *Applied Mathematics and Computation*, 184 (2): 566 – 588 (2007b).

Kara, Y., Gökçen, H., Atasagun, Y., “Balancing parallel assembly lines with precise and fuzzy goals”, *International Journal of Production Research*, 48 (6): 1685 – 1703 (2009).

Kara, Y., Paksoy, T., Chang, C., “Binary fuzzy goal programming approach to single model straight and U-shaped assembly line balancing”, *European Journal of Operational Research*, 195: 335 – 347 (2009).

Kim, Y. K., Kim, S. J., Kim, J. Y., “Balancing and sequencing mixed-model U-lines with a co-evolutionary algorithm”, *Production Planning and Control*, 11 (8): 754 – 764 (2000).

Kim, Y. K., Kim, J. Y., Kim, Y., “An endosymbiotic evolutionary algorithm for the integration of balancing and sequencing in mixed-model U-lines”, *European Journal of Operational Research*, 168 (3): 838 – 852 (2006).

Koç, İ. O., “Gezgin satıcı problemi için çok popülasyonlu paralel bir genetik algoritma tasarımı, geliştirilmesi ve analizi”, Doktora tezi, *Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü*, 15 – 70 (2007)

Koza, J., “Genetic Programming: On the programming of computers by means of natural selection”, *MIT Press*, 34 – 46 (1992).

Liepins, G. E., Hilliard, M. R., Richardson, J., Palmer, M., “Genetic algorithms applications to set covering and travelling salesman problems”, *Operations Research and Artificial Intelligence*, (Editors: Brown, D., White, C.), Kluwer Academic Publishers, 29 – 57 (1990).

Lin, S., Punch , W. F., Goodman, E., “Coarse-grain parallel genetic algorithms: categorization and new approach”, In *Sixth IEEE Symposium on Parallel and Distributed Processing*, Los Alamitos, CA: IEEE Computer Society Press, 24 – 37 (1994).

Lusa, A., “A survey of the literature on the multiple or parallel assembly line balancing problem”, *European Journal of Industrial Engineering*, 2 (1): 50 – 72 (2008).

Macaskill, J. L. C., “Production–line balances for mixed model lines”, *Management Science*, 19 (4): 423 – 434 (1972).

Martin, F. G., Cockerham, C. C. “High speed selection studies”. In O. Kempthorne, ed., *Biometrical Genetics*, Pergamon, 28 – 65 (1960).

McMullen, P. R., Frazier, G. V., “A heuristic for solving mixed-model line balancing problems with stochastic task durations and parallel stations”, *International Journal of Production Economics*, 51 (3): 177 – 190 (1997).

McMullen, P. R., Frazier, G. V., “Using simulated annealing to solve a multi objective assembly line balancing problem with parallel workstations”, *International Journal of Production Research*, 36 (10): 2717 – 2741 (1998).

Michalewicz, Z., “Genetic Algorithms + Data Structures = Evolution Programs”, Third Edition, *Springer-Verlag*, 19 – 67 (1999).

Miller, B. L., and Goldberg, D. E., “Genetic Algorithms, Selection Schemes and the Varying Effects of Noise”, *Evolutionary Computation*, 4 (2): 113 – 131 (1996).

Miltenburg, G.J., Wijngaard, J., The U-line balancing problem, *Management Science*, 40 (10): 1378 – 1388 (1994).

Miltenburg, J., “Balancing U-lines in a multiple U-line facility”, *European Journal of Operational Research*, 109: 1 – 23 (1998).

Miltenburg, J., “The effect of breakdowns on U-shaped production lines”, *International Journal of Production Research*, 38: 353 – 364 (2000).

Miltenburg, J., “Balancing and scheduling mixed-model U-shaped production lines”, *International Journal of Flexible Manufacturing Systems*, 14 (2): 119 – 151 (2002).

Munetemo, M., Takai, Y., Sato, Y., “An efficient migration scheme for subpopulation-based asynchronously parallel genetic algorithms”, In Forrest, S. (Ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms*, 649 – 649 (1993).

Nakade, K., Ohno, K., Shanthikumar, J.G., “Bounds and approximations for cycle times of a U-shaped production line”, *Operations Research Letters*, 21: 191 – 200 (1997).

Ozbakır, L., Baykasoglu, A., Gorkemli, B., Gorkemli, L., “Multiple-colony ant algorithm for parallel assembly line balancing problem”, *Applied Soft Computing*, 11: 3186 – 3198 (2011).

Özcan, U., Gökçen, H., Toklu, B., “Balancing parallel two-sided assembly lines”, *International Journal of Production Research*, 48 (16): 4767 – 4784 (2009a).

Özcan, U., Çerçioğlu, H., Gökçen, H., Toklu, B., “Balancing and sequencing of parallel mixed-model assembly lines”, *International Journal of Production Research*, 48 (17): 5089 – 5113 (2009b).

Özcan, U., Çercioglu, H., Gökçen, H., Toklu, B., “A tabu search algorithm for the parallel assembly line balancing problem”, *Gazi University Journal of Science*, 22 (4): 313 – 323 (2009c).

- Padrón, M., Resto, P., Mejía, H.P., “A methodology for cost-oriented assembly line balancing problems”, *Journal of Manufacturing Technology Management*, 20 (8): 1147 – 1165 (2009).
- Pinto, P., Dannenbring, D. G., Khumawala, B. M., “A branch and bound algorithm for assembly line balancing with parallelling”, *International Journal of Production Research*, 13: 183 – 196 (1975).
- Rabbani, M., Kazemi, S. M., Manavizadeh, N., “Mixed model U-line balancing type-1 problem: A new approach”, *Journal of Manufacturing Systems*, 31: 131 – 138 (2012).
- Rechenberg, I., “Evolutionsstrategie – Optimierung technischer Systeme nach Prinzipien der biologischen Evolution”, Doktora tezi, *Fromman-Holzboog*, 21 – 37 (1971).
- Reeves, C. R., Rowe J. E., “Genetic Algorithms-Principles and Perspectives A Guide to GA Theory”, *Kluwer Academic Publishers*, 19 – 47 (2003).
- Reeves, C., “Genetic Algorithms in Handbook of Metaheuristics”, Glover, F., Kochenberger, G. A., (editors) 55 – 82 (2003).
- Reiter, P., “On assembly line balancing problem”, *Operation Research*, 17 (4): 685 – 700 (1969).
- Rosenberg, O., Ziegler, H., “A comparison of heuristic algorithms for cost-oriented assembly line balancing”, *Zeitschrift für Operations Research*, 36: 477 – 495 (1992).
- Roshani, A., Fattahi, P., Roshani, A. , Salehi, M., Roshani, A., “Cost-oriented two-sided assembly line balancing problem: A simulated annealing approach”, *International Journal of Computer Integrated Manufacturing*, 1 – 27 (2012).
- Rudolph, G., “Global optimization by means of distributed evolution strategies”, In Schwefel, H-P., Manner, R. (Eds.), *Parallel Problem Solving from Nature*, Springer Verlag, 209 – 213 (1991).
- Sabuncuoglu I., Erel, E., Alp, A., “Ant colony optimization for the single model U-type assembly line balancing problem”, *International Journal of Production Economics*, 120: 287 – 300 (2009).
- Salveson, M. E., “The assembly line balancing problem”, *Journal of Industrial Engineering*, 6 (3): 18 – 25 (1955).
- Sarker, B. R., Shanthikumar, J. G., “A generalized approach for serial or paralel line balancing”, *International Journal of Production Research*, 21 (1): 109 – 133 (1983).

Scholl, A., Klein, R., "ULINO: Optimally balancing U-shaped JIT assembly lines", *International Journal of Production Research*, 37 (4): 721 – 736 (1999).

Scholl, A., "Balancing and Sequencing of Assembly Lines". 2 ed., Heidelberg: Physica-Verlag, 5 – 72 (1999).

Scholl, A., Klein, R., "ULINO: optimally balancing U-shaped JIT assembly lines", *International Journal of Production Research*, 37 (4): 721 – 736 (1999).

Scholl, A., Becker, C., "A note on an exact method for cost oriented assembly line balancing", *International Journal of Production Economics*, 97: 343 – 352 (2005).

Scholl, A., Becker, C., "State-of-the-art exact and heuristic solution procedures for simple assembly line balancing", *European Journal of Operational Research*, 168 (3): 666 – 693 (2006).

Scholl, A., Boysen, N., "Designing parallel assembly lines with split workplaces: model and optimization procedure", *International Journal of Production Economics*, 119: 90 – 100 (2009).

Schwefel, H. P., "Numerische Optimierung von Computer-Modellen", Doktora tezi, *Birkhäuser*, 8 – 22 (1974).

Simaria, A. S., Vilarinho, P. M., "The simple assembly line balancing problem with parallel workstations - A simulated annealing approach", *International Journal of Industrial Engineering: Theory Applications and Practice*, 8 (3): 230 – 240 (2001).

Simaria, A. S., Vilarinho, P. M., "A genetic algorithm based approach to the mixed-model assembly line balancing problem of type II", *Computers and Industrial Engineering*, 47 (4): 391 – 407 (2004).

Sparling, D., "Balancing JIT production units: The N U-line balancing problem", *Information Systems and Operational Research*, 36: 215 – 237 (1998).

Sparling, D., Miltenburg, J., "The mixed-model U-line balancing problem", *International Journal of Production Research*, 36 (2): 485 – 501 (1998).

Süer, G.A., Dagli, C., "A knowledge-based system for selection of resource allocation rules and algorithms". In: Mital, A., Anand, S. (Eds.), *Handbook of Expert System Applications in Manufacturing; Structures and Rules*. Chapman & Hall, London, 108 – 147 (1994).

Süer, G.A., "Designing parallel assembly lines". *Computers and Industrial Engineering*, 35 (3–4): 467 – 470 (1998).

Starkweather, T., Whitley, D., Mathias, K., “Optimization using distributed genetic algorithms”, In Schwefel, H-P., Manner, R. (Eds.), *Parallel Problem Solving from Nature*, Springer Verlag, 176 – 185 (1991).

Steffen, R., “Produktionsplanung Bei Fließbandfertigung”, Gabler, Wiesbaden, 15 – 22 (1977).

Thierens, D., “Selection Schemes, Elitist Recombination and Selection Intensity”, In T. Back (Ed.), *Proceedings of the Seventh International Conference on Genetic Algorithms*, San Francisco, 152 – 159 (1997).

Thomopoulos, N. T., “Line balancing–sequencing for mixed-model assembly”, *Management Science*, 14 (2): 59 – 75 (1967).

Thomopoulos, N. T., “Mixed model line balancing with smoothed station assignments”, *Management Science*, 16 (9): 593 – 603 (1970).

Toklu, B., Özcan, U., “A fuzzy goal programming model for the simple U-line balancing problem with multiple objectives”, *Engineering Optimization*, 40 (3): 191 – 204 (2008).

Urban, T.L., “Note. Optimal balancing of U-shaped assembly lines”, *Management Science*, 44 (5): 738 – 741 (1998).

Vilarinho, P. M., Simaria, A. S., “A two-stage heuristic method for balancing mixed-model assembly lines with parallel workstations”, *International Journal of Production Research*, 40 (6): 1405 – 1420 (2002).

Vilarinho, P. M., Simaria, A. S., “ANTBAL: an ant colony optimization algorithm for balancing mixed-model assembly lines with parallel workstations”, *International Journal of Production Research*, 44 (2): 291 – 303 (2006).

Whitley, D., “The genitor algorithm and selection pressure”, In J. D. Schaffer (Ed.), *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann Publishers, 116 – 121 (1989).

Whitley, D., Rana, S., Heckendorn, R. B., “Exploiting separability in search: The island model genetic algorithm”, *Journal of Computing and Information Technology*, 7 (1): 33 – 47 (1999).

Xu, B., Guan, Y., Chen, Z., Leung, K.R.P.H., “Parallel genetic algorithms with schema migration”, In *Proceedings of the 26th International Computer Software and Applications Conference on Prolonging Software Life: Development and Redevelopment*, 879 – 886 (2002).

Yazdanparast, V., Hajihosseini, H., Bahalke, A., “Cost Oriented Assembly Line Balancing Problem with Sequence Dependent Setup Times”, *Australian Journal of Basic and Applied Sciences*, 5 (9): 878 – 884 (2011).

EKLER

EK-1. PKPP algoritmasının parametre setleri

Çizelge 7.1. PKPP algoritmasının birinci ve ikinci aşamasındaki parametre değerleri

Problem	C	PKPP algoritmasının parametreleri			
		L	α	S	δ
Roszieng (25 - 24)	14	18	0,7	10	2
	16	18	0,7	10	2
	17	18	0,7	10	2
	22	15	0,7	8	1
	30	15	0,7	8	1
Sawyer (30 - 28)	25	12	0,7	10	3
	27	12	0,7	10	3
	30	12	0,7	10	3
	36	15	0,7	8	2
	41	15	0,7	8	2
	54	15	0,7	8	2
Kilbridge (45 - 43)	57	12	1,0	7	2
	79	20	1,0	7	2
	92	20	1,0	7	2
	110	20	1,0	6	1
	138	25	1,0	6	1
	184	30	1,0	6	1
Hahn (53 - 51)	2004	20	1,0	10	1
	2338	20	1,0	8	1
	2806	20	1,0	8	1
	3507	30	1,0	8	1
	4676	40	1,0	6	1
Tonge (70 - 66)	160	12	0,7	8	3
	168	12	0,7	8	3
	207	12	0,7	8	3
	234	15	0,7	8	3
	270	15	0,7	8	3
	293	15	0,7	8	3
Wee-mag (75 - 71)	28	10	0,7	20	5
	29	10	0,7	20	5
	31	10	0,7	20	5
	33	10	0,7	20	5
	34	10	0,7	20	5
	41	10	0,7	20	5
	42	10	0,7	15	5
	43	8	0,7	10	5
	49	8	0,7	10	5
	54	8	0,7	10	5

EK-1. (Devam) PKPP algoritmasının parametre setleri

Çizelge 7.1. (Devam) PKPP algoritmasının birinci ve ikinci aşamasındaki parametre değerleri

Problem	C	PKPP algoritmasının parametreleri			
		L	α	S	δ
Arcus1 (83 - 79)	3786	12	0,7	5	3
	3985	12	0,7	5	3
	4206	12	0,7	5	3
	4454	15	0,7	5	3
	4732	15	0,7	5	3
	5853	18	0,7	5	3
	6842	25	0,7	5	3
	7571	25	0,7	5	3
	8412	25	0,7	5	3
	10816	30	0,7	5	3
Lutz2 (89 - 85)	11	12	0,7	15	5
	12	12	0,7	15	5
	13	12	0,7	10	5
	14	12	0,7	10	5
	15	12	0,7	8	3
	16	12	0,7	8	3
	17	12	0,7	8	3
	19	12	0,7	8	3
	20	12	0,7	8	3
Lutz3 (89 - 85)	75	20	0,7	7	3
	79	20	0,7	7	3
	83	20	0,7	7	3
	87	20	0,7	7	3
	92	20	0,7	7	3
Mukherje (94 - 90)	176	12	0,7	5	3
	183	12	0,7	5	3
	192	12	0,7	5	3
	201	12	0,7	5	3
	211	12	0,7	5	3
	222	12	0,7	5	3
	234	12	0,7	5	3
	248	12	0,7	5	3
	263	12	0,7	5	2
	281	12	0,7	5	2
	301	12	0,7	5	2
	324	12	0,7	5	2
	351	12	0,7	5	2

EK-1. (Devam) PKPP algoritmasının parametre setleri

Çizelge 7.1. (Devam) PKPP algoritmasının birinci ve ikinci aşamasındaki parametre değerleri

Problem	C	PKPP algoritmasının parametreleri			
		L	α	S	δ
Arcus (111 - 107)	5785	12	0,7	8	3
	6016	12	0,7	8	3
	6267	12	0,7	8	3
	6540	12	0,7	8	3
	6837	10	0,7	6	3
	7162	10	0,7	6	3

EK-2. M-ÇPPG algoritmasının farklı parametre setleriyle *Roszieng*, *Hahn* ve *Lutz2* problemleri için elde edilen sonuçlar

Çizelge 8.1. Roszieng probleminin deney sonuçları

Set	$ P $	P_ϵ	P_m	$\bar{B}$	Denemeler					ORT	SO (%)
					1	2	3	4	5		
1	50	0,6	0,1	0,4	5 550	5 550	5 550	5 550	5 550	5 550	3,9
2	50	0,6	0,1	0,6	5 640	5 580	5 640	5 580	5 580	5 604	4,9
3	50	0,6	0,1	0,8	5 670	5 670	5 610	5 520	5 610	5 616	5,2
4	50	0,6	0,2	0,4	5 550	5 550	5 550	5 670	5 580	5 580	4,5
5	50	0,6	0,2	0,6	5 520	5 550	5 580	5 670	5 580	5 580	4,5
6	50	0,6	0,2	0,8	5 490	5 490	5 580	5 550	5 550	5 532	3,6
7	50	0,6	0,3	0,4	5 610	5 580	5 580	5 670	5 610	5 610	5,1
8	50	0,6	0,3	0,6	5 520	5 430	5 520	5 550	5 430	5 490	2,8
9	50	0,6	0,3	0,8	5 520	5 610	5 490	5 550	5 670	5 568	4,3
10	50	0,7	0,1	0,4	5 580	5 580	5 520	5 520	5 580	5 556	4,0
11	50	0,7	0,1	0,6	5 580	5 610	5 640	5 520	5 520	5 574	4,4
12	50	0,7	0,1	0,8	5 520	5 580	5 580	5 460	5 520	5 532	3,6
13	50	0,7	0,2	0,4	5 670	5 610	5 520	5 580	5 610	5 598	4,8
14	50	0,7	0,2	0,6	5 520	5 520	5 520	5 520	5 550	5 526	3,5
15	50	0,7	0,2	0,8	5 520	5 490	5 460	5 460	5 460	5 478	2,6
16	50	0,7	0,3	0,4	5 640	5 550	5 520	5 520	5 490	5 544	3,8
17	50	0,7	0,3	0,6	5 670	5 670	5 670	5 460	5 670	5 628	5,4
18	50	0,7	0,3	0,8	5 460	5 670	5 460	5 670	5 460	5 540	3,8
19	50	0,8	0,1	0,4	5 550	5 550	5 580	5 520	5 550	5 550	3,9
20	50	0,8	0,1	0,6	5 640	5 640	5 580	5 610	5 490	5 592	4,7
21	50	0,8	0,1	0,8	5 580	5 580	5 580	5 670	5 640	5 610	5,1
22	50	0,8	0,2	0,4	5 520	5 520	5 520	5 550	5 460	5 514	3,3

EK-2. (Devam) M-ÇPPG algoritmasının farklı parametre setleriyle *Roszieng*, *Hahn* ve *Lutz2* problemleri için elde edilen sonuçlar

Çizelge 8.1. (Devam) Roszieng probleminin deney sonuçları

Set	$ P $	P_ϵ	P_m	$\bar{B}$	Denemeler					ORT	SO (%)
					1	2	3	4	5		
23	50	0,8	0,2	0,6	5 610	5 580	5 610	5 460	5 580	5 568	4,3
24	50	0,8	0,2	0,8	5 550	5 550	5 580	5 640	5 640	5 592	4,7
25	50	0,8	0,3	0,4	5 520	5 550	5 550	5 520	5 520	5 532	3,6
26	50	0,8	0,3	0,6	5 520	5 550	5 430	5 520	5 460	5 496	2,9
27	50	0,8	0,3	0,8	5 610	5 520	5 610	5 580	5 580	5 580	4,5
28	100	0,6	0,1	0,4	5 580	5 580	5 580	5 580	5 550	5 574	4,4
29	100	0,6	0,1	0,6	5 580	5 580	5 520	5 550	5 580	5 562	4,2
30	100	0,6	0,1	0,8	5 460	5 490	5 460	5 490	5 490	5 478	2,6
31	100	0,6	0,2	0,4	5 580	5 580	5 580	5 550	5 430	5 544	3,8
32	100	0,6	0,2	0,6	5 490	5 490	5 490	5 490	5 490	5 490	2,8
33	100	0,6	0,2	0,8	5 610	5 430	5 460	5 430	5 640	5 514	3,3
34	100	0,6	0,3	0,4	5 520	5 550	5 520	5 460	5 460	5 502	3,0
35	100	0,6	0,3	0,6	5 550	5 550	5 550	5 550	5 550	5 550	3,9
36	100	0,6	0,3	0,8	5 580	5 550	5 520	5 550	5 520	5 544	3,8
37	100	0,7	0,1	0,4	5 640	5 550	5 580	5 610	5 460	5 568	4,3
38	100	0,7	0,1	0,6	5 550	5 550	5 550	5 550	5 550	5 550	3,9
39	100	0,7	0,1	0,8	5 580	5 640	5 430	5 430	5 640	5 544	3,8
40	100	0,7	0,2	0,4	5 550	5 460	5 550	5 520	5 550	5 526	3,5
41	100	0,7	0,2	0,6	5 550	5 400	5 550	5 490	5 520	5 502	3,0
42	100	0,7	0,2	0,8	5 490	5 490	5 430	5 580	5 490	5 496	2,9
43	100	0,7	0,3	0,4	5 610	5 550	5 490	5 400	5 400	5 490	2,8
44	100	0,7	0,3	0,6	5 430	5 430	5 520	5 520	5 520	5 484	2,7

EK-2. (Devam) M-ÇPPG algoritmasının farklı parametre setleriyle *Roszieng*, *Hahn* ve *Lutz2* problemleri için elde edilen sonuçlar

Çizelge 8.1. (Devam) Roszieng probleminin deney sonuçları

Set	$ P $	P_{ϵ}	P_m	$\bar{B}$	Denemeler					ORT	SO (%)
					1	2	3	4	5		
45	100	0,7	0,3	0,8	5 580	5 520	5 460	5 460	5 460	5 496	2,9
46	100	0,8	0,1	0,4	5 520	5 520	5 520	5 460	5 430	5 490	2,8
47	100	0,8	0,1	0,6	5 460	5 460	5 460	5 460	5 460	5 460	2,2
48	100	0,8	0,1	0,8	5 520	5 580	5 550	5 550	5 580	5 556	4,0
49	100	0,8	0,2	0,4	5 460	5 460	5 460	5 460	5 550	5 478	2,6
50	100	0,8	0,2	0,6	5 580	5 640	5 580	5 580	5 580	5 592	4,7
51	100	0,8	0,2	0,8	5 550	5 550	5 550	5 460	5 580	5 538	3,7
52	100	0,8	0,3	0,4	5 580	5 490	5 550	5 490	5 520	5 526	3,5
53	100	0,8	0,3	0,6	5 400	5 400	5 490	5 460	5 520	5 454	2,1
54	100	0,8	0,3	0,8	5 580	5 610	5 580	5 520	5 460	5 550	3,9
55	200	0,6	0,1	0,4	5 550	5 460	5 460	5 460	5 430	5 472	2,5
56	200	0,6	0,1	0,6	5 520	5 520	5 520	5 550	5 520	5 526	3,5
57	200	0,6	0,1	0,8	5 520	5 580	5 490	5 490	5 580	5 532	3,6
58	200	0,6	0,2	0,4	5 550	5 490	5 520	5 520	5 520	5 520	3,4
59	200	0,6	0,2	0,6	5 550	5 550	5 550	5 490	5 550	5 538	3,7
60	200	0,6	0,2	0,8	5 490	5 490	5 490	5 460	5 490	5 484	2,7
61	200	0,6	0,3	0,4	5 430	5 490	5 430	5 550	5 550	5 490	2,8
62	200	0,6	0,3	0,6	5 520	5 520	5 520	5 580	5 580	5 544	3,8
63	200	0,6	0,3	0,8	5 550	5 490	5 550	5 550	5 550	5 538	3,7
64	200	0,7	0,1	0,4	5 400	5 580	5 580	5 580	5 640	5 556	4,0
65	200	0,7	0,1	0,6	5 520	5 580	5 520	5 550	5 610	5 556	4,0
66	200	0,7	0,1	0,8	5 460	5 490	5 580	5 430	5 550	5 502	3,0

EK-2. (Devam) M-ÇPPG algoritmasının farklı parametre setleriyle *Roszieng*, *Hahn* ve *Lutz2* problemleri için elde edilen sonuçlar

Çizelge 8.1. (Devam) Roszieng probleminin deney sonuçları

Set	$ P $	P_ϵ	P_m	$\bar{B}$	Denemeler					ORT	SO (%)
					1	2	3	4	5		
67	200	0,7	0,2	0,4	5 490	5 580	5 550	5 550	5 580	5 550	3,9
68	200	0,7	0,2	0,6	5 520	5 430	5 460	5 460	5 580	5 490	2,8
69	200	0,7	0,2	0,8	5 520	5 550	5 490	5 580	5 550	5 538	3,7
70	200	0,7	0,3	0,4	5 430	5 550	5 490	5 460	5 460	5 478	2,6
71	200	0,7	0,3	0,6	5 430	5 460	5 550	5 520	5 520	5 496	2,9
72	200	0,7	0,3	0,8	5 490	5 490	5 580	5 550	5 550	5 532	3,6
73	200	0,8	0,1	0,4	5 550	5 640	5 490	5 550	5 550	5 556	4,0
74	200	0,8	0,1	0,6	5 490	5 550	5 520	5 400	5 520	5 496	2,9
75	200	0,8	0,1	0,8	5 550	5 460	5 460	5 460	5 460	5 478	2,6
76	200	0,8	0,2	0,4	5 580	5 490	5 490	5 580	5 520	5 532	3,6
77	200	0,8	0,2	0,6	5 580	5 550	5 550	5 430	5 460	5 514	3,3
78	200	0,8	0,2	0,8	5 580	5 490	5 520	5 520	5 490	5 520	3,4
79	200	0,8	0,3	0,4	5 580	5 430	5 580	5 520	5 490	5 520	3,4
80	200	0,8	0,3	0,6	5 430	5 430	5 460	5 430	5 430	5 436	1,8
81	200	0,8	0,3	0,8	5 490	5 400	5 490	5 430	5 520	5 466	2,4

EK–2. (Devam) M–ÇPPG algoritmasının farklı parametre setleriyle *Roszieng*, *Hahn* ve *Lutz2* problemleri için elde edilen sonuçlar

Çizelge 8.2. Hahn probleminin deney sonuçları

Set	$ P $	P_ϵ	P_m	$\bar{B}$	Denemeler					ORT	SO (%)
					1	2	3	4	5		
1	50	0,6	0,1	0,4	1 274 544	1 282 560	1 282 560	1 268 532	1 254 504	1 272 540	2,8
2	50	0,6	0,1	0,6	1 266 528	1 254 504	1 268 532	1 262 520	1 262 520	1 262 921	2,0
3	50	0,6	0,1	0,8	1 272 540	1 264 524	1 272 540	1 270 536	1 260 516	1 268 131	2,4
4	50	0,6	0,2	0,4	1 270 536	1 274 544	1 266 528	1 266 528	1 266 528	1 268 933	2,5
5	50	0,6	0,2	0,6	1 266 528	1 254 504	1 266 528	1 270 536	1 270 536	1 265 726	2,2
6	50	0,6	0,2	0,8	1 268 532	1 270 536	1 266 528	1 266 528	1 258 512	1 266 127	2,2
7	50	0,6	0,3	0,4	1 266 528	1 264 524	1 266 528	1 266 528	1 270 536	1 266 929	2,3
8	50	0,6	0,3	0,6	1 258 512	1 258 512	1 258 512	1 256 508	1 258 512	1 258 111	1,6
9	50	0,6	0,3	0,8	1 258 512	1 262 520	1 268 532	1 264 524	1 266 528	1 264 123	2,1
10	50	0,7	0,1	0,4	1 268 532	1 268 532	1 272 540	1 270 536	1 258 512	1 267 730	2,4
11	50	0,7	0,1	0,6	1 262 520	1 254 504	1 254 504	1 266 528	1 258 512	1 259 314	1,7
12	50	0,7	0,1	0,8	1 274 544	1 276 548	1 276 548	1 276 548	1 272 540	1 275 346	3,0
13	50	0,7	0,2	0,4	1 266 528	1 266 528	1 262 520	1 272 540	1 266 528	1 266 929	2,3
14	50	0,7	0,2	0,6	1 270 536	1 262 520	1 270 536	1 254 504	1 264 524	1 264 524	2,1
15	50	0,7	0,2	0,8	1 280 556	1 266 528	1 266 528	1 258 512	1 288 512	1 272 127	2,7
16	50	0,7	0,3	0,4	1 272 540	1 268 532	1 266 528	1 268 532	1 264 524	1 268 131	2,4
17	50	0,7	0,3	0,6	1 258 512	1 264 524	1 254 504	1 262 520	1 274 544	1 262 921	2,0
18	50	0,7	0,3	0,8	1 258 512	1 266 528	1 254 504	1 266 528	1 254 504	1 260 115	1,7
19	50	0,8	0,1	0,4	1 272 540	1 266 528	1 264 524	1 266 528	1 272 540	1 268 532	2,4
20	50	0,8	0,1	0,6	1 274 544	1 270 536	1 282 560	1 282 560	1 274 544	1 276 949	3,1
21	50	0,8	0,1	0,8	1 266 528	1 270 536	1 268 532	1 266 528	1 266 528	1 267 730	2,4
22	50	0,8	0,2	0,4	1 258 512	1 258 512	1 258 512	1 258 512	1 264 524	1 259 714	1,7

EK-2. (Devam) M-ÇPPG algoritmasının farklı parametre setleriyle *Roszieng*, *Hahn* ve *Lutz2* problemleri için elde edilen sonuçlar

Çizelge 8.2. (Devam) Hahn probleminin deney sonuçları

Set	$ P $	P_ϵ	P_m	$\bar{B}$	Denemeler					ORT	SO (%)
					1	2	3	4	5		
23	50	0,8	0,2	0,6	1 266528	1 270 536	1 266 528	1 266 528	1 266 528	1 267 330	2,3
24	50	0,8	0,2	0,8	1 270536	1 274 544	1 276 548	1 272 540	1 266 528	1 272 139	2,7
25	50	0,8	0,3	0,4	1 264524	1 270 536	1 258 512	1 268 532	1 272 540	1 266 929	2,3
26	50	0,8	0,3	0,6	1 274544	1 268 532	1 268 532	1 268 532	1 266 528	1 269 334	2,5
27	50	0,8	0,3	0,8	1 270536	1 270 536	1 258 512	1 272 540	1 274 544	1 269 334	2,5
28	100	0,6	0,1	0,4	1 266528	1 266 528	1 266 528	1 266 528	1 266 528	1 266 528	2,3
29	100	0,6	0,1	0,6	1 274544	1 274 544	1 264 524	1 272 540	1 258 512	1 268 933	2,5
30	100	0,6	0,1	0,8	1 274544	1 274 544	1 274 544	1 274 544	1 274 544	1 274 544	2,9
31	100	0,6	0,2	0,4	1 264524	1 258 512	1 264 524	1 264 524	1 270 536	1 264 524	2,1
32	100	0,6	0,2	0,6	1 266528	1 268 532	1 268 532	1 266 528	1 268 532	1 267 730	2,4
33	100	0,6	0,2	0,8	1 272540	1 264 524	1 264 524	1 264 524	1 266 528	1 266 528	2,3
34	100	0,6	0,3	0,4	1 266528	1 266 528	1 272 540	1 264 524	1 266 528	1 267 330	2,3
35	100	0,6	0,3	0,6	1 264524	1 270 536	1 264 524	1 260 516	1 260 516	1 264 123	2,1
36	100	0,6	0,3	0,8	1 266528	1 254 504	1 254 504	1 254 504	1 268 532	1 259 714	1,7
37	100	0,7	0,1	0,4	1 266528	1 266 528	1 270 536	1 272 540	1 258 512	1 266 929	2,3
38	100	0,7	0,1	0,6	1 266528	1 272 540	1 272 540	1 272 540	1 266 528	1 270 135	2,6
39	100	0,7	0,1	0,8	1 258512	1 256 528	1 258 512	1 254 504	1 258 512	1 257 314	1,5
40	100	0,7	0,2	0,4	1 254504	1 254 504	1 258 512	1 262 520	1 266 528	1 259 314	1,7
41	100	0,7	0,2	0,6	1 268532	1 266 528	1 266 528	1 270 536	1 270 536	1 268 532	2,4
42	100	0,7	0,2	0,8	1 264524	1 264 524	1 258 512	1 266 528	1 268 532	1 264 524	2,1
43	100	0,7	0,3	0,4	1 272540	1 272 540	1 272 540	1 268 532	1 272 540	1 271 738	2,7
44	100	0,7	0,3	0,6	1 254504	1 254 504	1 266 528	1 266 528	1 258 512	1 260 115	1,7

EK-2. (Devam) M-ÇPPG algoritmasının farklı parametre setleriyle *Roszieng*, *Hahn* ve *Lutz2* problemleri için elde edilen sonuçlar

Çizelge 8.2. (Devam) Hahn probleminin deney sonuçları

Set	$ P $	P_ϵ	P_m	$\bar{B}$	Denemeler					ORT	SO (%)
					1	2	3	4	5		
45	100	0,7	0,3	0,8	1 258 512	1 268 532	1 272 540	1 272 540	1 268 532	1 268 131	2,4
46	100	0,8	0,1	0,4	1 266 528	1 258 512	1 258 512	1 266 528	1 266 528	1 263 322	2,0
47	100	0,8	0,1	0,6	1 270 536	1 264 524	1 258 512	1 260 516	1 258 512	1 262 520	1,9
48	100	0,8	0,1	0,8	1 272 540	1 272 540	1 272 540	1 266 528	1 272 540	1 271 338	2,7
49	100	0,8	0,2	0,4	1 272 540	1 266 528	1 268 532	1 272 540	1 258 512	1 267 730	2,4
50	100	0,8	0,2	0,6	1 272 540	1 268 532	1 258 512	1 268 532	1 254 504	1 264 524	2,1
51	100	0,8	0,2	0,8	1 254 504	1 258 512	1 254 504	1 254 504	1 268 532	1 258 111	1,6
52	100	0,8	0,3	0,4	1 266 528	1 254 504	1 264 524	1 254 504	1 260 516	1 260 115	1,7
53	100	0,8	0,3	0,6	1 254 504	1 254 504	1 254 504	1 254 504	1 262 540	1 256 111	1,4
54	100	0,8	0,3	0,8	1 264 524	1 258 512	1 274 544	1 258 512	1 272 540	1 265 726	2,2
55	200	0,6	0,1	0,4	1 266 528	1 272 540	1 270 536	1 266 528	1 270 536	1 269 334	2,5
56	200	0,6	0,1	0,6	1 254 504	1 258 512	1 254 504	1 264 524	1 266 528	1 259 714	1,7
57	200	0,6	0,1	0,8	1 258 512	1 266 528	1 258 512	1 266 528	1 272 540	1 264 524	2,1
58	200	0,6	0,2	0,4	1 254 504	1 254 504	1 266 528	1 264 524	1 254 504	1 258 913	1,7
59	200	0,6	0,2	0,6	1 268 532	1 254 504	1 258 512	1 270 536	1 254 504	1 261 318	1,8
60	200	0,6	0,2	0,8	1 258 512	1 264 524	1 264 524	1 266 528	1 264 524	1 263 722	2,0
61	200	0,6	0,3	0,4	1 260 516	1 268 532	1 254 504	1 254 504	1 254 504	1 258 512	1,6
62	200	0,6	0,3	0,6	1 254 504	1 254 504	1 254 504	1 268 532	1 268 532	1 260 115	1,7
63	200	0,6	0,3	0,8	1 272 540	1 258 512	1 272 540	1 266 528	1 272 540	1 268 532	2,4
64	200	0,7	0,1	0,4	1 268 532	1 254 504	1 268 532	1 264 524	1 264 524	1 264 123	2,1
65	200	0,7	0,1	0,6	1 264 524	1 258 512	1 268 532	1 268 532	1 272 540	1 266 528	2,3
66	200	0,7	0,1	0,8	1 262 520	1 258 512	1 264 524	1 264 524	1 266 528	1 263 322	2,0

EK-2. (Devam) M-ÇPPG algoritmasının farklı parametre setleriyle *Roszieng*, *Hahn* ve *Lutz2* problemleri için elde edilen sonuçlar

Çizelge 8.2. (Devam) Hahn probleminin deney sonuçları

Set	$ P $	P_ϵ	P_m	$\bar{B}$	Denemeler					ORT	SO (%)
					1	2	3	4	5		
67	200	0,7	0,2	0,4	1 254 504	1 266 528	1 254 504	1 270 536	1 266 528	1 262 520	1,9
68	200	0,7	0,2	0,6	1 254 504	1 268 532	1 266 528	1 272 540	1 272 540	1 266 929	2,3
69	200	0,7	0,2	0,8	1 266 528	1 254 504	1 270 536	1 266 528	1 268 532	1 265 326	2,2
70	200	0,7	0,3	0,4	1 266 528	1 266 528	1 264 524	1 268 532	1 272 540	1 267 730	2,4
71	200	0,7	0,3	0,6	1 254 504	1 252 500	1 258 512	1 258 512	1 258 512	1 256 508	1,5
72	200	0,7	0,3	0,8	1 254 504	1 264 524	1 264 524	1 254 504	1 254 504	1 258 512	1,6
73	200	0,8	0,1	0,4	1 280 556	1 254 504	1 254 504	1 254 504	1 266 528	1 262 119	1,9
74	200	0,8	0,1	0,6	1 272 540	1 252 500	1 252 500	1 254 504	1 254 504	1 257 310	1,5
75	200	0,8	0,1	0,8	1 268 532	1 266 528	1 274 544	1 272 540	1 268 532	1 270 135	2,6
76	200	0,8	0,2	0,4	1 264 524	1 264 524	1 268 532	1 264 524	1 258 512	1 264 123	2,1
77	200	0,8	0,2	0,6	1 268 532	1 266 528	1 266 528	1 266 528	1 266 528	1 266 929	2,3
78	200	0,8	0,2	0,8	1 254 504	1 254 504	1 254 504	1 266 528	1 258 512	1 257 710	1,6
79	200	0,8	0,3	0,4	1 274 544	1 274 544	1 262 520	1 266 528	1 274 544	1 270 536	2,6
80	200	0,8	0,3	0,6	1 256 508	1 256 508	1 252 500	1 254 504	1 264 524	1 256 909	1,5
81	200	0,8	0,3	0,8	1 258 512	1 258 512	1 258 512	1 258 512	1 264 524	1 259 714	1,7

EK–2. (Devam) M–ÇPPG algoritmasının farklı parametre setleriyle *Roszieng*, *Hahn* ve *Lutz2* problemleri için elde edilen sonuçlar

Çizelge 8.3. Lutz2 probleminin deney sonuçları

Set	$ P $	P_ϵ	P_m	$\bar{B}$	Denemeler					ORT	SO (%)
					1	2	3	4	5		
1	50	0,6	0,1	0,4	71 775	71 907	71 544	72 116	72 589	71 986	10,7
2	50	0,6	0,1	0,6	72 468	71 896	71 841	73 040	72 050	72 259	11,1
3	50	0,6	0,1	0,8	71 632	72 193	71 995	71 423	71 786	71 806	10,4
4	50	0,6	0,2	0,4	72 391	72 600	72 644	72 809	72 699	72 629	11,6
5	50	0,6	0,2	0,6	71 808	72 567	71 555	72 985	72 435	72 270	11,1
6	50	0,6	0,2	0,8	72 237	72 886	71 731	73 392	72 820	72 613	11,6
7	50	0,6	0,3	0,4	73 106	73 062	71 610	73 062	72 050	72 578	11,6
8	50	0,6	0,3	0,6	73 381	72 226	72 204	73 326	72 314	72 690	11,7
9	50	0,6	0,3	0,8	71 929	71 632	71 918	71 962	72 237	71 936	10,6
10	50	0,7	0,1	0,4	71 500	73 040	72 688	72 006	71 412	72 129	10,9
11	50	0,7	0,1	0,6	72 798	72 897	72 534	73 293	72 369	72 778	11,9
12	50	0,7	0,1	0,8	72 842	72 193	72 512	73 337	73 073	72 791	11,9
13	50	0,7	0,2	0,4	72 435	72 336	72 281	72 688	71 588	72 266	11,1
14	50	0,7	0,2	0,6	72 886	72 941	72 886	72 424	72 149	72 657	11,7
15	50	0,7	0,2	0,8	73 095	72 193	72 622	73 359	72 919	72 838	12,0
16	50	0,7	0,3	0,4	71 764	71 170	71 731	72 072	72 325	71 812	10,4
17	50	0,7	0,3	0,6	72 589	73 018	72 083	72 083	71 709	72 296	11,1
18	50	0,7	0,3	0,8	72 149	71 665	71 632	72 424	72 424	72 059	10,8
19	50	0,8	0,1	0,4	72 721	72 083	72 490	72 160	72 930	72 477	11,4
20	50	0,8	0,1	0,6	72 622	72 358	73 722	72 116	72 215	72 607	11,6
21	50	0,8	0,1	0,8	71 830	72 182	72 710	72 776	73 084	72 516	11,5
22	50	0,8	0,2	0,4	71 929	72 413	72 391	72 765	73 095	72 519	11,5

EK–2. (Devam) M–ÇPPG algoritmasının farklı parametre setleriyle *Roszieng*, *Hahn* ve *Lutz2* problemleri için elde edilen sonuçlar

Çizelge 8.3. (Devam) Lutz2 probleminin deney sonuçları

Set	$ P $	P_ϵ	P_m	$\bar{B}$	Denemeler					ORT	SO (%)
					1	2	3	4	5		
23	50	0,8	0,2	0,6	72 886	71 841	72 303	73 040	72 732	72 560	11,5
24	50	0,8	0,2	0,8	71 401	71 643	72 721	72 006	72 578	72 070	10,8
25	50	0,8	0,3	0,4	72 787	72 578	71 808	73 194	72 281	72 530	11,5
26	50	0,8	0,3	0,6	73 480	72 050	72 281	72 182	72 633	72 525	11,5
27	50	0,8	0,3	0,8	72 589	72 435	72 622	72 160	72 358	72 433	11,3
28	100	0,6	0,1	0,4	72 050	72 358	73 073	72 314	72 809	72 521	11,5
29	100	0,6	0,1	0,6	72 842	72 831	73 700	72 006	72 248	72 725	11,8
30	100	0,6	0,1	0,8	72 677	73 348	72 831	72 776	72 952	72 917	12,1
31	100	0,6	0,2	0,4	72 006	72 787	72 754	71 764	71 808	72 224	11,0
32	100	0,6	0,2	0,6	72 248	71 808	72 248	71 973	72 567	72 169	10,9
33	100	0,6	0,2	0,8	72 699	73 106	72 798	72 798	72 886	72 857	12,0
34	100	0,6	0,3	0,4	72 864	72 545	72 435	72 160	71 764	72 354	11,2
35	100	0,6	0,3	0,6	72 281	72 226	71 731	73 216	71 720	72 235	11,0
36	100	0,6	0,3	0,8	72 567	72 468	72 160	72 391	72 281	72 373	11,3
37	100	0,7	0,1	0,4	72 413	71 764	72 028	71 720	72 336	72 052	10,8
38	100	0,7	0,1	0,6	72 215	71 522	71 896	71 808	71 544	71 797	10,4
39	100	0,7	0,1	0,8	73 194	73 117	72 105	73 073	72 864	72 871	12,0
40	100	0,7	0,2	0,4	72 402	72 072	71 874	72 270	72 292	72 182	11,0
41	100	0,7	0,2	0,6	72 666	72 160	72 875	71 467	72 061	72 246	11,1
42	100	0,7	0,2	0,8	71 852	72 501	72 776	72 050	72 325	72 301	11,1
43	100	0,7	0,3	0,4	72 215	72 479	71 797	72 369	72 138	72 200	11,0
44	100	0,7	0,3	0,6	71 676	71 280	70 994	71 588	71 544	71 416	9,8

EK-2. (Devam) M-ÇPPG algoritmasının farklı parametre setleriyle *Roszieng*, *Hahn* ve *Lutz2* problemleri için elde edilen sonuçlar

Çizelge 8.3. (Devam) Lutz2 probleminin deney sonuçları

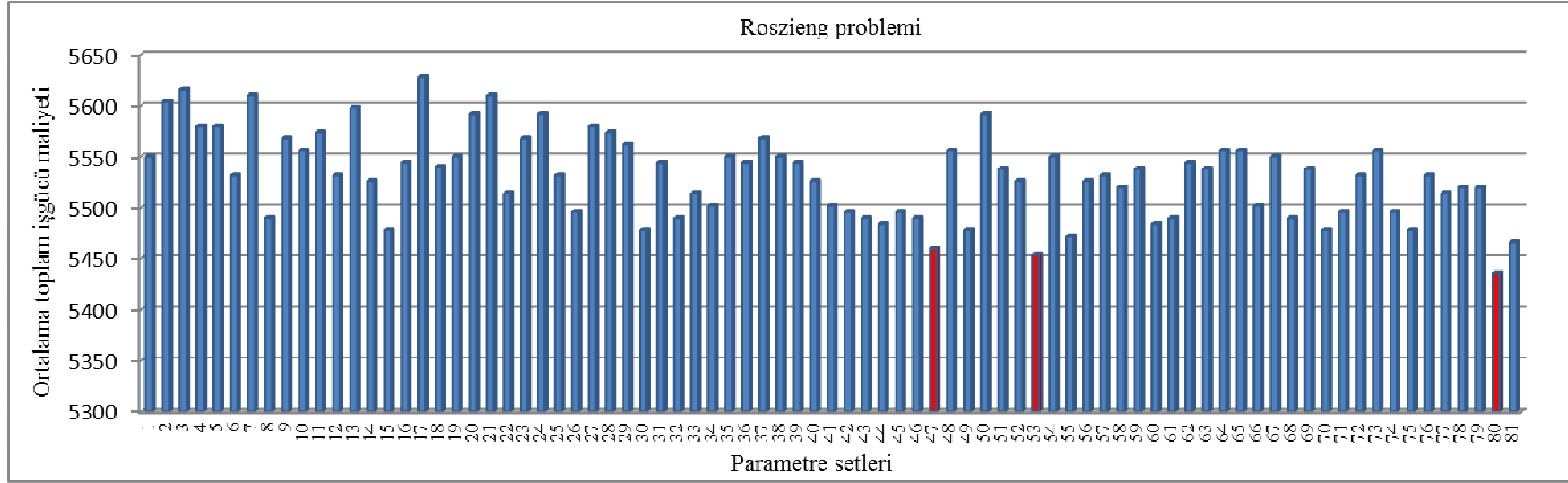
Set	$ P $	P_{ϵ}	P_m	$\bar{B}$	Denemeler					ORT	SO (%)
					1	2	3	4	5		
45	100	0,7	0,3	0,8	71 973	71 841	72 468	71 269	72 358	71 982	10,6
46	100	0,8	0,1	0,4	73 205	72 215	72 512	72 314	72 402	72 530	11,5
47	100	0,8	0,1	0,6	72 358	72 358	72 501	71 720	72 369	72 261	11,1
48	100	0,8	0,1	0,8	72 369	72 391	71 610	72 171	72 974	72 303	11,1
49	100	0,8	0,2	0,4	71 973	72 556	72 809	72 655	71 896	72 378	11,3
50	100	0,8	0,2	0,6	72 138	72 149	72 149	72 028	72 622	72 217	11,0
51	100	0,8	0,2	0,8	72 996	71 412	71 808	71 170	72 105	71 898	10,5
52	100	0,8	0,3	0,4	72 171	72 116	72 116	72 006	72 006	72 083	10,8
53	100	0,8	0,3	0,6	71 797	71 676	71 577	71 577	71 753	71 676	10,2
54	100	0,8	0,3	0,8	72 501	72 776	72 127	71 434	72 083	72 184	11,0
55	200	0,6	0,1	0,4	72 677	72 314	71 533	71 148	73 304	72 195	11,0
56	200	0,6	0,1	0,6	72 303	71 544	72 149	71 896	72 732	72 125	10,9
57	200	0,6	0,1	0,8	72 666	72 589	72 556	72 435	71 874	72 424	11,3
58	200	0,6	0,2	0,4	71 247	71 929	71 962	72 358	72 039	71 907	10,5
59	200	0,6	0,2	0,6	71 566	72 105	71 940	70 807	71 907	71 665	10,2
60	200	0,6	0,2	0,8	71 588	72 303	72 193	72 116	72 534	72 147	10,9
61	200	0,6	0,3	0,4	72 028	71 665	71 599	72 017	71 874	71 837	10,4
62	200	0,6	0,3	0,6	71 786	71 940	72 567	72 358	71 302	71 991	10,7
63	200	0,6	0,3	0,8	71 786	71 775	71 918	71 665	71 566	71 742	10,3
64	200	0,7	0,1	0,4	71 731	72 182	72 468	72 039	72 468	72 178	11,0
65	200	0,7	0,1	0,6	71 588	72 600	73 051	72 193	71 951	72 277	11,1
66	200	0,7	0,1	0,8	72 413	72 853	72 083	72 358	72 292	72 400	11,3

EK–2. (Devam) M–ÇPPG algoritmasının farklı parametre setleriyle *Roszieng*, *Hahn* ve *Lutz2* problemleri için elde edilen sonuçlar

Çizelge 8.3. (Devam) Lutz2 probleminin deney sonuçları

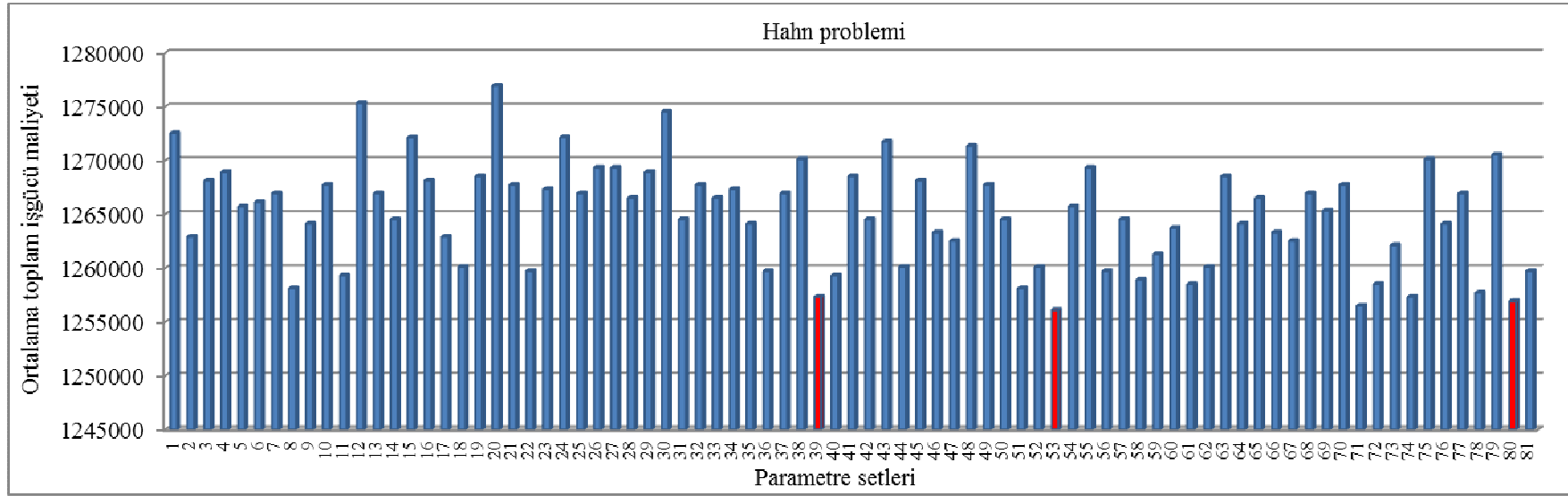
Set	$ P $	P_ζ	P_m	$\bar{B}$	Denemeler					ORT	SO (%)
					1	2	3	4	5		
67	200	0,7	0,2	0,4	72 864	73 194	72 061	72 380	72 523	72 604	11,6
68	200	0,7	0,2	0,6	71 907	71 907	72 754	72 028	72 270	72 173	10,9
69	200	0,7	0,2	0,8	72 721	72 193	71 566	72 116	72 501	72 219	11,0
70	200	0,7	0,3	0,4	71 918	71 753	71 819	72 787	71 588	71 973	10,6
71	200	0,7	0,3	0,6	72 314	71797	71 665	72 061	72 314	72 030	10,7
72	200	0,7	0,3	0,8	71 940	71 819	71 203	71 698	72 127	71 757	10,3
73	200	0,8	0,1	0,4	71 907	72 413	71 863	71 467	72 270	71 984	10,7
74	200	0,8	0,1	0,6	71 797	72 391	71 874	71 863	72 600	72 105	10,8
75	200	0,8	0,1	0,8	72 270	72 325	72 831	71 533	71 357	72 063	10,8
76	200	0,8	0,2	0,4	71 819	72 061	73 128	72 787	72 545	72 468	11,4
77	200	0,8	0,2	0,6	71 632	70 928	72 050	71 599	72 589	71 760	10,3
78	200	0,8	0,2	0,8	71 852	72 039	71 797	71 962	72 303	71 991	10,7
79	200	0,8	0,3	0,4	72 556	71 907	72 523	72 358	72 413	72 351	11,2
80	200	0,8	0,3	0,6	71 929	71 390	71 423	71 511	71 445	71 540	10,0
81	200	0,8	0,3	0,8	71 357	72 600	73 348	73 249	72 754	72 662	11,7

EK-3. *Roszieng*, *Hahn* ve *Lutz2* problemleri için M-ÇPPG algoritmasının farklı parametre setleriyle yapılan deney sonuçları



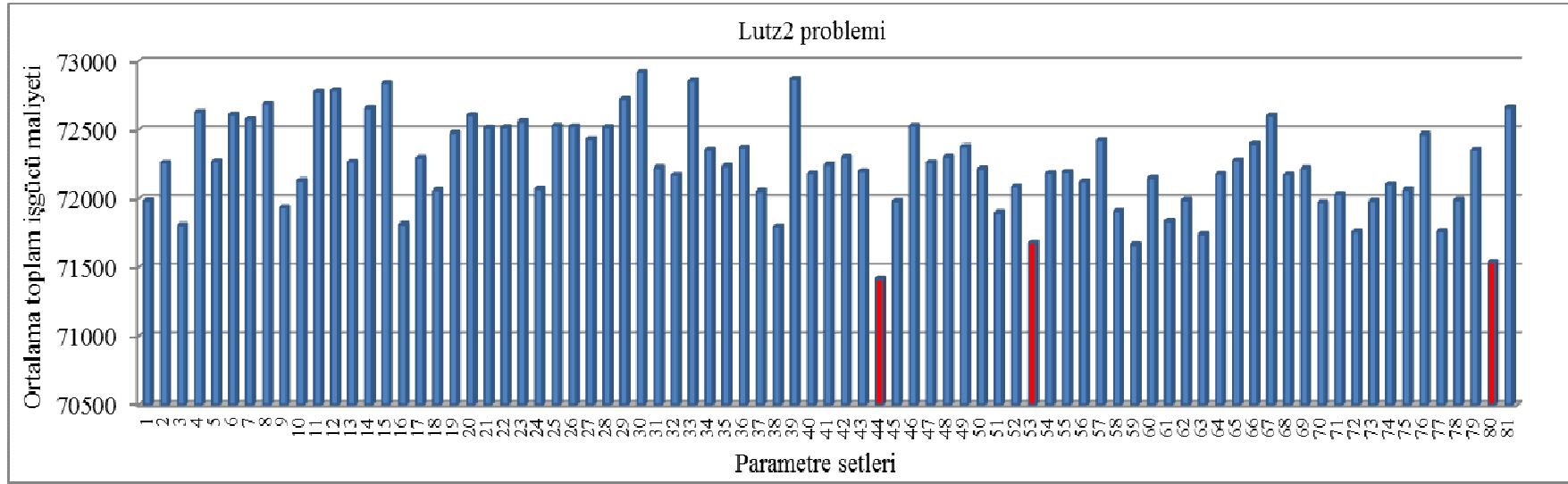
Şekil. 6.1. *Roszieng* problemi için yapılan 5 deneme sonunda elde edilen toplam işgücü maliyet değerlerinin ortalaması

EK-3. (Devam) *Roszieng*, *Hahn* ve *Lutz2* problemleri için M-ÇPPG algoritmasının farklı parametre setleriyle yapılan deney sonuçları



Şekil. 6.2. *Hahn* problemi için yapılan 5 deneme sonunda elde edilen toplam işgücü maliyet değerlerinin ortalaması

EK-3. (Devam) *Roszieng*, *Hahn* ve *Lutz2* problemleri için M-ÇPPG algoritmasının farklı parametre setleriyle yapılan deney sonuçları



Şekil. 6.3. *Lutz2* problemi için yapılan 5 deneme sonunda elde edilen toplam işgücü maliyet değerlerinin ortalaması

EK-4. Roszieng problemi için varyans analizi sonuçları

General Linear Model: ORT versus POPULASYON BUYUK; ÇAPRAZLAMA ORANI; ...

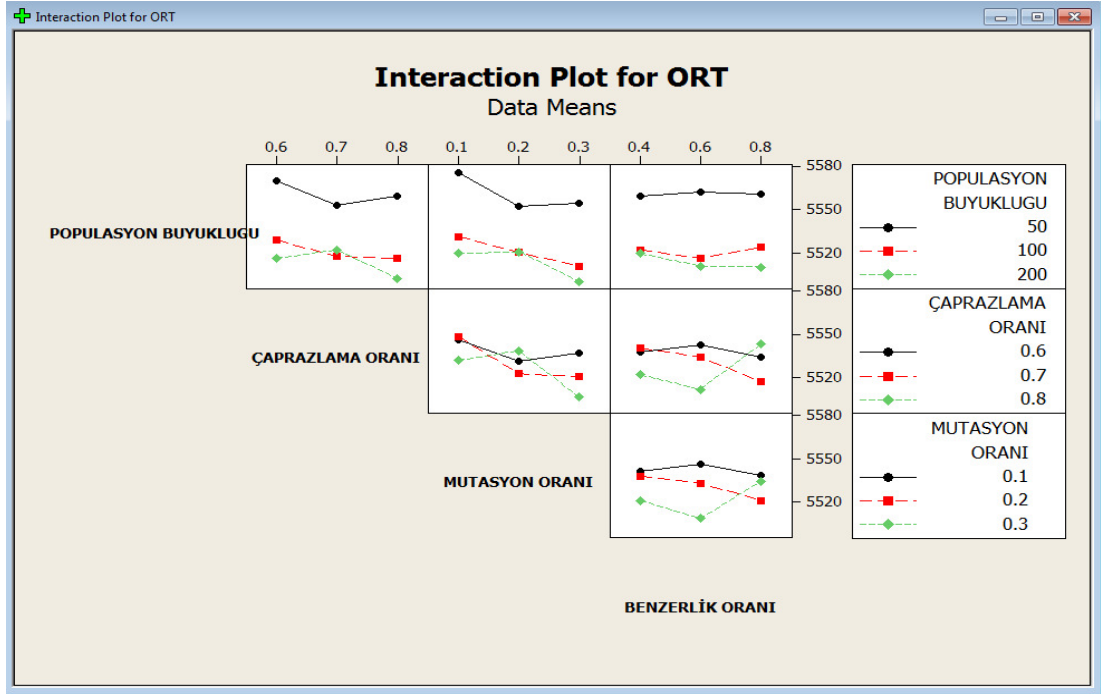
Factor	Type	Levels	Values
POPULASYON BUYUKLUGU	fixed	3	50; 100; 200
ÇAPRAZLAMA ORANI	fixed	3	0.6; 0.7; 0.8
MUTASYON ORANI	fixed	3	0.1; 0.2; 0.3
BENZERLİK ORANI	fixed	3	0.4; 0.6; 0.8

Analysis of Variance for ORT, using Adjusted SS for Tests

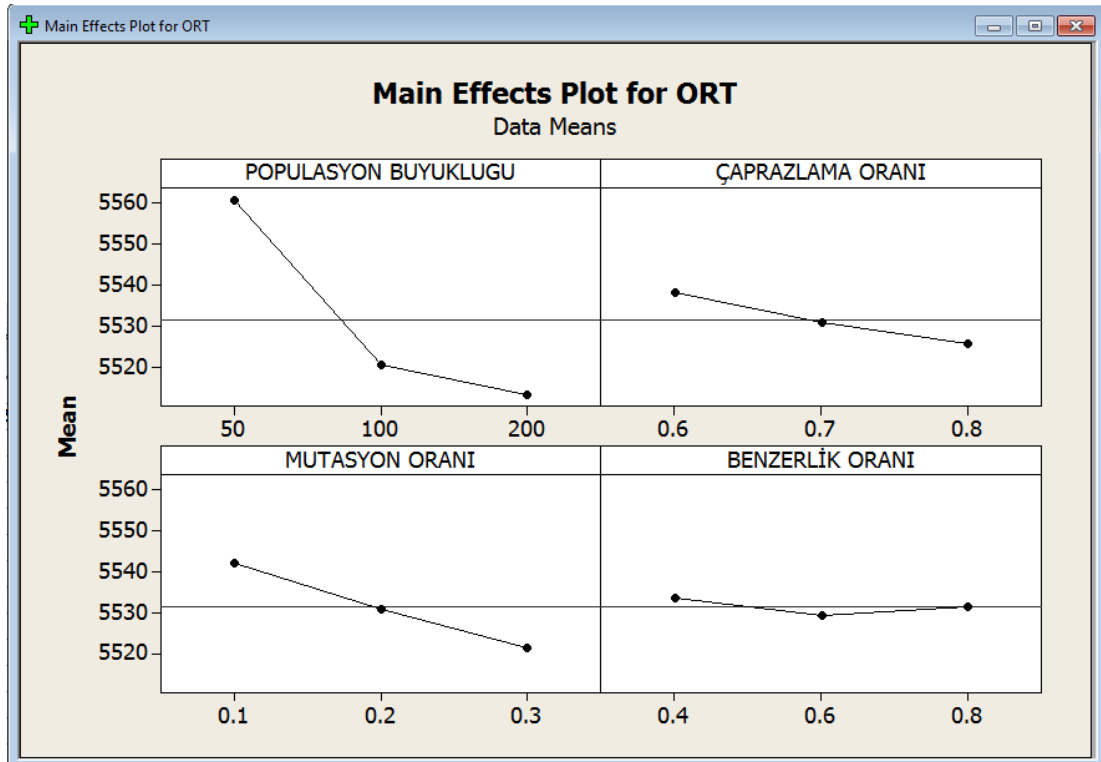
Source	DF	Seq SS	Adj SS	Adj MS	F	P
POPULASYON BUYUKLUGU	2	35165	35165	17582	11.32	0.000
ÇAPRAZLAMA ORANI	2	2119	2119	1060	0.68	0.510
MUTASYON ORANI	2	5744	5744	2872	1.85	0.168
BENZERLİK ORANI	2	216	216	108	0.07	0.933
POPULASYON BUYUKLUGU* ÇAPRAZLAMA ORANI	4	1993	1993	498	0.32	0.863
POPULASYON BUYUKLUGU*MUTASYON ORANI	4	1758	1758	439	0.28	0.888
POPULASYON BUYUKLUGU*BENZERLİK ORANI	4	616	616	154	0.10	0.982
ÇAPRAZLAMA ORANI*MUTASYON ORANI	4	4718	4718	1179	0.76	0.557
ÇAPRAZLAMA ORANI*BENZERLİK ORANI	4	7280	7280	1820	1.17	0.335
MUTASYON ORANI*BENZERLİK ORANI	4	4509	4509	1127	0.73	0.579
Error	48	74569	74569	1554		
Total	80	138686				

S = 39.4146 R-Sq = 46.23% R-Sq(adj) = 10.39%

EK-5. Roszieng problemi için etkileşim grafikleri



Şekil. 7.1. Roszieng problemi için parametreler arası etkileşim grafiği



Şekil. 7.2. Roszieng problemi için parametre – ortalama işgücü maliyeti grafiği

EK-6. Hahn problemi için varyans analizi sonuçları

General Linear Model: ORT versus POPULASYON BUYUK; ÇAPRAZLAMA ORANI; ...

Factor	Type	Levels	Values
POPULASYON BUYUKLUGU	fixed	3	50; 100; 200
ÇAPRAZLAMA ORANI	fixed	3	0.6; 0.7; 0.8
MUTASYON ORANI	fixed	3	0.1; 0.2; 0.3
BENZERLİK ORANI	fixed	3	0.4; 0.6; 0.8

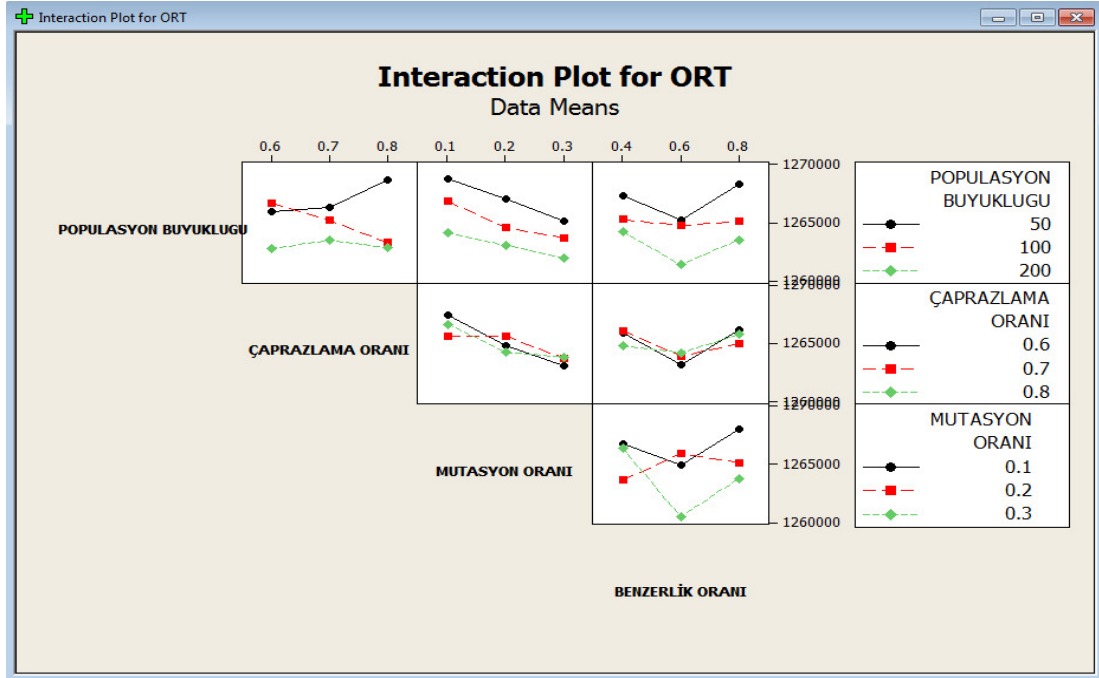
Analysis of Variance for ORT, using Adjusted SS for Tests

Source	DF	Seq SS	Adj SS	Adj MS
POPULASYON BUYUKLUGU	2	212051882	212051882	106025941
ÇAPRAZLAMA ORANI	2	502976	502976	251488
MUTASYON ORANI	2	124295476	124295476	62147738
BENZERLİK ORANI	2	61962433	61962433	30981217
POPULASYON BUYUKLUGU* ÇAPRAZLAMA ORANI	4	93139497	93139497	23284874
POPULASYON BUYUKLUGU*MUTASYON ORANI	4	7085651	7085651	1771413
POPULASYON BUYUKLUGU*BENZERLİK ORANI	4	24667466	24667466	6166867
ÇAPRAZLAMA ORANI*MUTASYON ORANI	4	26677843	26677843	6669461
ÇAPRAZLAMA ORANI*BENZERLİK ORANI	4	21252288	21252288	5313072
MUTASYON ORANI*BENZERLİK ORANI	4	168054577	168054577	42013644
Error	48	1093020871	1093020871	22771268
Total	80	1832710959		

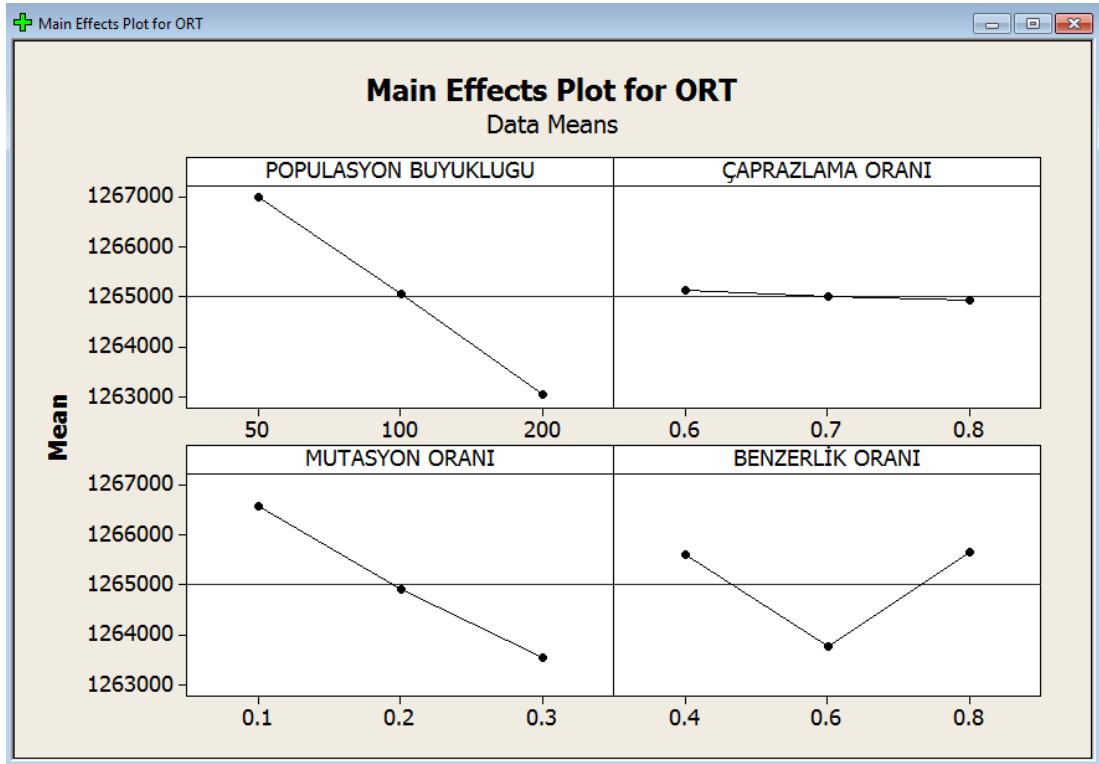
Source	F	P
POPULASYON BUYUKLUGU	4.66	0.014
ÇAPRAZLAMA ORANI	0.01	0.989
MUTASYON ORANI	2.73	0.075
BENZERLİK ORANI	1.36	0.266
POPULASYON BUYUKLUGU* ÇAPRAZLAMA ORANI	1.02	0.405
POPULASYON BUYUKLUGU*MUTASYON ORANI	0.08	0.989
POPULASYON BUYUKLUGU*BENZERLİK ORANI	0.27	0.895
ÇAPRAZLAMA ORANI*MUTASYON ORANI	0.29	0.881
ÇAPRAZLAMA ORANI*BENZERLİK ORANI	0.23	0.918
MUTASYON ORANI*BENZERLİK ORANI	1.85	0.136
Error		
Total		

S = 4771.92 R-Sq = 40.36% R-Sq(adj) = 0.60%

EK-7. Hahn problemi için etkileşim grafikleri



Şekil. 8.1. Hahn problemi için parametreler arası etkileşim grafiği



Şekil. 8.2. Hahn problemi için parametre – ortalama işgücü maliyeti grafiği

EK-8. Lutz2 problemi için varyans analizi sonuçları

General Linear Model: ORT versus POPULASYON BUYUK; ÇAPRAZLAMA ORANI; ...

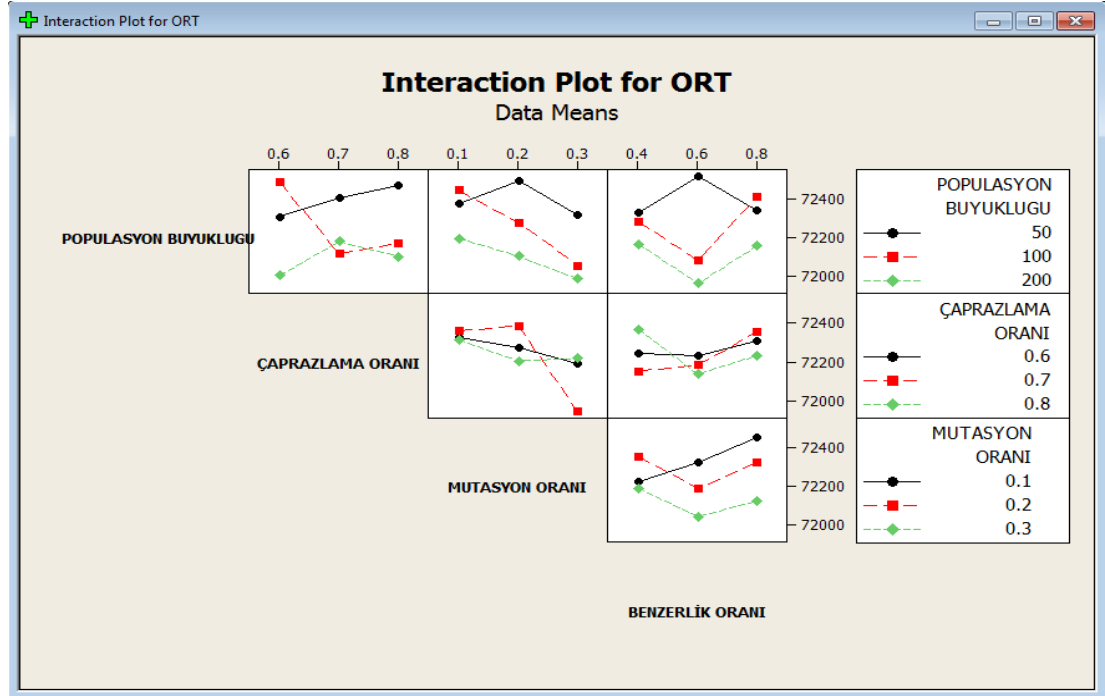
Factor	Type	Levels	Values
POPULASYON BUYUKLUGU	fixed	3	50; 100; 200
ÇAPRAZLAMA ORANI	fixed	3	0.6; 0.7; 0.8
MUTASYON ORANI	fixed	3	0.1; 0.2; 0.3
BENZERLİK ORANI	fixed	3	0.4; 0.6; 0.8

Analysis of Variance for ORT, using Adjusted SS for Tests

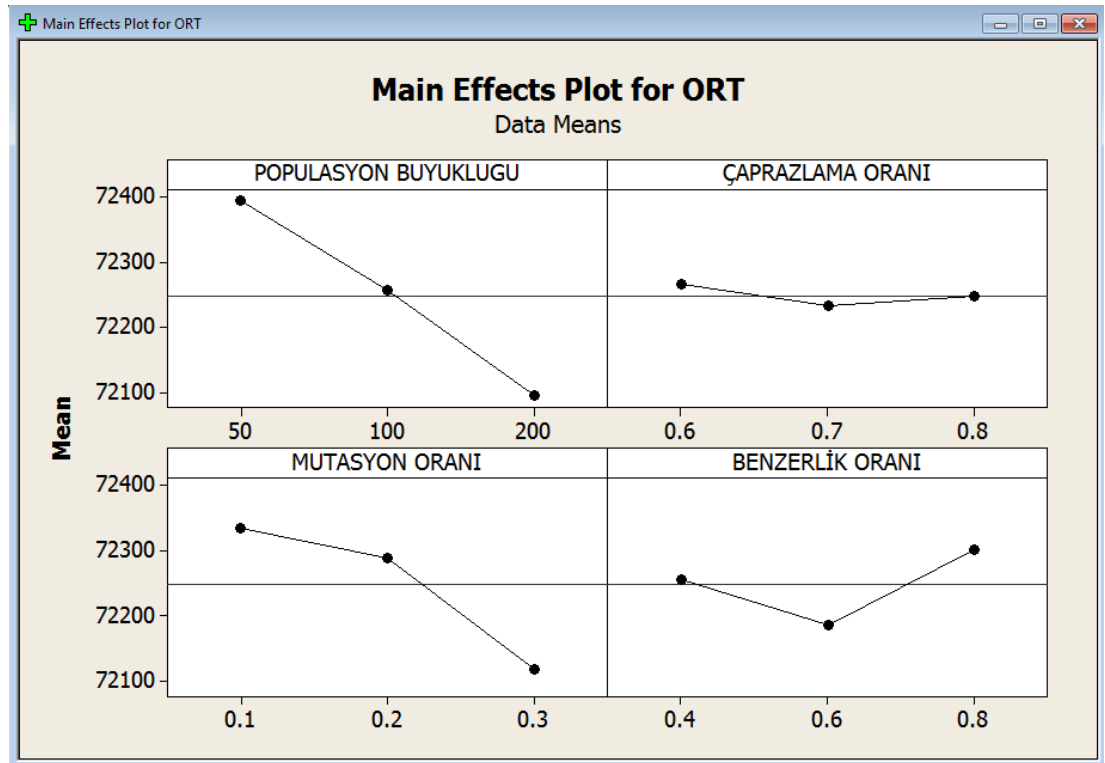
Source	DF	Seq SS	Adj SS	Adj MS	F	P
POPULASYON BUYUKLUGU	2	1206704	1206704	603352	8.30	0.001
ÇAPRAZLAMA ORANI	2	14738	14738	7369	0.10	0.904
MUTASYON ORANI	2	698471	698471	349235	4.80	0.013
BENZERLİK ORANI	2	181255	181255	90627	1.25	0.297
POPULASYON BUYUKLUGU* ÇAPRAZLAMA ORANI	4	964185	964185	241046	3.31	0.018
POPULASYON BUYUKLUGU*MUTASYON ORANI	4	312884	312884	78221	1.08	0.379
POPULASYON BUYUKLUGU*BENZERLİK ORANI	4	746064	746064	186516	2.56	0.050
ÇAPRAZLAMA ORANI*MUTASYON ORANI	4	552886	552886	138222	1.90	0.126
ÇAPRAZLAMA ORANI*BENZERLİK ORANI	4	303524	303524	75881	1.04	0.395
MUTASYON ORANI*BENZERLİK ORANI	4	284120	284120	71030	0.98	0.429
Error	48	3490688	3490688	72723		
Total	80	8755519				

S = 269.671 R-Sq = 60.13% R-Sq(adj) = 33.55%

EK-9. Lutz problemi için etkileşim grafikleri



Şekil. 9.1. Lutz2 problemi için parametreler arası etkileşim grafiği



Şekil. 9.2. Lutz2 problemi için parametre – ortalama işgücü maliyeti grafiği

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, Adı : DERYA, Tusan
 Uyuğu : T.C.
 Doğum tarihi ve yeri : 16.07.1980 Kdz.Ereğli
 Medeni hali : Bekar
 Telefon : 0 (312) 246 66 66 / 1235
 Faks : 0 (312) 246 66 60
 e-mail : tderya@baskent.edu.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Yüksek lisans	Başkent Ün. / Endüstri Müh. Bölümü	2008
Lisans	Başkent Ün. / Endüstri Müh. Bölümü	2005
Lise	TED Kdz. Ereğli Koleji	1999

İş Deneyimi

Yıl	Yer	Görev
2008 –	Başkent Üniversitesi Endüstri Mühendisliği Bölümü	Araştırma Görevlisi
2007 – 2008	Başkent Üniversitesi Endüstri Mühendisliği Bölümü	Endüstri Mühendisi
2005 – 2007	Başkent Üniversitesi Endüstri Mühendisliği Bölümü	Lisanüstü Bursiyer

Yabancı Dil

İngilizce

Hobiler

Bilgisayar teknolojileri, Turistik Gezi, Bowling