

KIRIKKALE ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ

ENDÜSTRİ MÜHENDİSLİĞİ ANABİLİM DALI  
YÜKSEK LİSANS TEZİ

Stokastik Montaj Hattı Dengeleme Problemlerine Kuyruk Ağı Yaklaşımı

Hacı Mehmet ALAĞAŞ

HAZİRAN 2012

**Endüstri Mühendisliği Anabilim Dalında** Hacı Mehmet ALAĞAŞ tarafından hazırlanan “STOKASTİK MONTAJ HATTI DENGELEME PROBLEMLERİNE KUYRUK AĞI YAKLAŞIMI” adlı Yüksek Lisans Tezinin Anabilim Dalı standartlarına uygun olduğunu onaylarım.

Prof. Dr. Burak BİRGÖREN

Anabilim Dalı Başkanı

Bu tezi okuduğumu ve tezin **Yüksek Lisans Tezi** olarak bütün gereklilikleri yerine getirdiğini onaylarım.

Yrd. Doç. Dr. Mustafa YÜZÜKIRMIZI

Ortak Danışman

Yrd. Doç. Dr. Ahmet Kürşad TÜRKER

Danışman

*Jüri Üyeleri*

Başkan : Yrd. Doç. Dr. Süleyman ERSÖZ \_\_\_\_\_

Üye (Danışman) : Yrd. Doç. Dr. Ahmet Kürşad TÜRKER \_\_\_\_\_

Üye : Yrd. Doç. Dr. Ümit Sami SAKALLI \_\_\_\_\_

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

Bu tez ile Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu Yüksek Lisans derecesini onaylamıştır.

Doç. Dr. Erdem Kamil YILDIRIM

Fen Bilimleri Enstitüsü Müdürü

*Fedakâr Aileme*

*Sevgili Eşime*

*Canım Oğluma*

## ÖZET

### STOKASTİK MONTAJ HATTI Dengeleme Problemlerine Kuyruk Ağı Yaklaşımı

ALAĞAŞ, Hacı Mehmet

Kırıkkale Üniversitesi

Fen Bilimleri Enstitüsü

Endüstri Mühendisliği Anabilim Dalı, Yüksek Lisans tezi

Danışman: Yrd. Doç. Dr. Ahmet Kürşad TÜRKER

Ortak Danışman: Yrd. Doç. Dr. Mustafa YÜZÜKIRMIZI

Haziran 2012, 127 sayfa

Bu çalışmada, stokastik işlem süreli, düz, tek modelli ve gecikmeli montaj hatları ele alınmıştır. Ele alınan montaj hatları için belirlenen sabit istasyon sayısına bağlı olarak çevrim süresinin minimize edilmesi (Tip-2) amaçlanmıştır. Bu problemin çözümü için önerilen algoritmada, bir montaj hattı için mümkün olan bütün atama alternatifleri arasından en iyi çıktı hızını veren alternatifin bulunması amaçlanmıştır. Algoritma üç aşamadan oluşmaktadır. İlk olarak, kısıt programlama metodu kullanılarak mümkün bir görev-istasyon atama kombinasyonu bulunmuştur. Algoritmanın ikinci adımında, bu atama kombinasyonunun çıktı hızı ve istasyon doluluk oranları kapalı kuyruk ağı modeli kullanılarak hesaplanmıştır. Bütün olası görev-istasyon atama kombinasyonları için bu iki adım tekrarlanmıştır. Son adımda ise bulunan bütün atama kombinasyonları değerlendirilmiş ve en iyi çıktı hızı değerini veren atama bulunmuştur. Algoritma, literatürde stokastik tip-2 montaj hattı modelleri için deney seti bulamadığından dolayı deterministik tip-2 montaj hattı modelleri için oluşturulan deney setleri kullanılarak test edilmiştir.

**Anahtar kelimeler:** Stokastik Montaj Hattı, Tip-2 Montaj Hattı Dengeleme, Kapalı Kuyruk Ağı Modeli, Kısıt Programlama.

## ABSTRACT

### QUEUING NETWORK APPROACH FOR STOCHASTIC ASSEMBLY LINE BALANCING PROBLEMS

ALAĞAŞ, Hacı Mehmet

Kırıkkale University

Graduate School of Natural and Applied Sciences

Department of Industrial Engineering, M.Sc Thesis

Supervisor: Asst. Prof. Dr. Ahmet Kürşad TÜRKER

Co-Supervisor: Asst. Prof. Dr. Mustafa YÜZÜKIRMIZI

June 2012, 127 pages

In this study, stochastic process time, smooth, single model and delayed assembly lines were examined. For the considered assembly lines depending on determined cycle time minimizing number of fixed stations (Type-2) is aimed. The proposed algorithm for solving the problem, for an assembly line from among all possible assignment alternatives to find having the best output rate alternative is aimed. The algorithm consists of three phases. Firstly, a possible combination of a task-station the assignment was found using constraint programming method. The second step of the algorithm, the output rate and station utilizations of this assignment combination are calculated using a closed queuing network model. Task-station assignments for all possible combinations of these two steps are repeated. At the last step, having the best output rate assignment was found in all assignment combinations. Because it cannot find experimental sets for stochastic type-2 assembly line models in the literature, the algorithm was tested using deterministic type-2 assembly line models experimental sets.

**Keywords:** Stochastic Assembly Line, Type-2 Assembly Line Balancing, Closed Queuing Network Model, Constraint Programming

## TEŞEKKÜR

Tezimin hazırlanmasında tüm bilgi birikimi ile yanımda olup, çalışmalarımnda desteklerini ve kıymetli zamanını ayırıp her türlü desteğini eksik etmeyen, bilimsel açıdan gelişimim için her türlü imkânı sunan, tez yöneticisi hocam Sayın Yrd. Doç. Dr. A. Kürşad TÜRKER'e ve Sayın Yrd. Doç. Dr. Mustafa YÜZÜKIRMIZI'ya sonsuz teşekkür ederim. Bölüm hocalarım, Sayın Prof. Dr. Burak BİRGÖREN'e, Sayın Yrd. Doç. Dr. Süleyman ERSÖZ'e, Yrd. Doç. Dr. Tamer EREN'e ve Öğr. Gör. Fedai YILMAZ'a teşekkür ederim.

Tez çalışmalarımnda benden yardımlarını ve desteklerini esirgemeyen saygı değer arkadaşlarım Mehmet PINARBAŞI, Çağrı SEL ve Öznur DUMAN'a sonsuz teşekkür ederim. Desteklerini esirgemeyen ve çalışma aşamamda bana sabırla katlanan sevgili eşime teşekkür ederim.

# İÇİNDEKİLER DİZİNİ

	<u>Sayfa</u>
<b>ÖZET</b> .....	<b>i</b>
<b>ABSTRACT</b> .....	<b>ii</b>
<b>TEŞEKKÜR</b> .....	<b>iii</b>
<b>İÇİNDEKİLER DİZİNİ</b> .....	<b>iv</b>
<b>ÇİZELGELER DİZİNİ</b> .....	<b>vii</b>
<b>ŞEKİLLER DİZİNİ</b> .....	<b>xi</b>
<b>1. GİRİŞ</b> .....	<b>1</b>
1.1. Tez Çalışmasının İçeriği .....	4
<b>2. LİTERATÜR ARAŞTIRMASI</b> .....	<b>6</b>
2.1. Montaj Hattı Dengeleme Problemleri .....	6
2.2. Kuyruk Ağı Modelleri.....	13
<b>3. MONTAJ HATTI VE MONTAJ HATTI DENGELEME</b> .....	<b>17</b>
3.1. Montaj Hattı Çeşitleri .....	17
3.2. Montaj Hattı Dengeleme Problemi .....	18
3.3. Montaj Hattı Dengeleme İçin Temel Kavramlar .....	19
3.4. Montaj Hattı Dengelemenin Amaçları.....	21
3.5. Montaj Hattı Dengeleme Problemlerinin Sınıflandırılması.....	22
<b>4. KUYRUK AĞLARI İLE SİSTEM PERFORMANSININ DEĞERLENDİRİLMESİ</b> .....	<b>26</b>
4.1. Tek Sınıflı Kuyruk Ağları.....	29
4.1.1. Matematiksel Formdaki Sistemler .....	33
4.1.2. Kapalı Kuyruk Ağlarının Matematiksel Formu .....	35
4.2. Ortalama Değer Analizi (ODA).....	37
4.2.1. Tek Sınıflı Blokesiz Sistemler İçin ODA .....	37

<b>5. KISIT PROGRAMLAMA .....</b>	<b>39</b>
5.1. Kısıt Programlama Temel Kavramı .....	41
5.2. Arama Stratejileri.....	43
5.2.1. Öncelikle Derinlik Arama.....	43
5.2.2. Çok Noktalı Arama.....	43
5.2.3. Yeniden Başlatmalı Arama.....	44
<b>6. KUYRUK AĞLARININ MONTAJ HATLILARINA UYUMLULUĞU ....</b>	<b>45</b>
<b>7. TİP 2 STOKASTİK DÜZ MONTAJ HATTI İÇİN ÖNERİLEN ÇÖZÜM METODOLOJİSİ .....</b>	<b>55</b>
7.1. Problemin Tanımı .....	55
7.1.1. Kuyruk Ağı Modeli Varsayımlar ve Kabuller .....	56
7.1.2. Montaj Hattı İçin Varsayım ve Kabuller .....	57
7.2. Önerilen Çözüm Metodolojisi.....	58
7.2.1. Montaj Hattı Dengeleme Probleminin Doğrusal Programlama Modeli	59
7.2.2. Kısıt Programlama Algoritması .....	60
7.2.3. Kapalı Kuyruk Ağı Modeli ve ODA Algoritması .....	62
7.2.4. Kısıt Programlama ile Alternatif Atamaların Bulunması .....	64
<b>8. DENEYSEL ÇALIŞMALAR VE SONUÇLARI .....</b>	<b>66</b>
8.1. Öncelikle Derinlik Arama Stratejisi Deneyleri.....	66
8.1.1. 7 İş 2 İstasyonlu Mertens (1967) Deneyi.....	67
8.1.2. 7 İş 3 İstasyonlu Mertens (1967) Deneyi.....	69
8.1.3. 7 İş 5 İstasyonlu Mertens (1967) Deneyi.....	71
8.1.4. 9 İş 3 İstasyonlu Jaeschke (1964) Deneyi.....	73
8.1.5. 9 İş 4 İstasyonlu Jaeschke (1964) Deneyi.....	75
8.1.6. 9 İş 7 İstasyonlu Jaeschke (1964) Deneyi.....	77
8.1.7. 11 İş 3 İstasyonlu Jackson (1956) Deneyi .....	79
8.1.8. 11 İş 4 İstasyonlu Jackson (1956) Deneyi .....	81



8.1.9.	11 İş 5 İstasyonlu Jackson (1956) Deneyi .....	83
8.1.10.	21 İş 3 İstasyonlu Mitchell (1957) Deneyi .....	84
8.1.11.	21 İş 5 İstasyonlu Mitchell (1957) Deneyi .....	87
8.1.12.	21 İş 8 İstasyonlu Mitchell (1957) Deneyi .....	89
8.1.13.	28 İş 4 İstasyonlu Heskiaoff (1968) Deneyi .....	91
8.1.14.	28 İş 5 İstasyonlu Heskiaoff (1968) Deneyi .....	93
8.2.	Çok Noktalı Arama Stratejisi Deneyleri .....	94
8.2.1.	28 İş 4 İstasyonlu Heskiaoff (1968) Deneyi .....	95
8.2.2.	28 İş 5 İstasyonlu Heskiaoff (1968) Deneyi .....	96
8.2.3.	30 İş 5 İstasyonlu Sawyer (1970) Deneyi .....	98
8.2.4.	30 İş 8 İstasyonlu Sawyer (1970) Deneyi .....	100
8.2.5.	30 İş 13 İstasyonlu Sawyer (1970) Deneyi .....	101
8.3.	Deneysel Çalışmaların Sonuçlarının Değerlendirilmesi .....	103
8.4.	Montaj Hattının Performansının Değişik Ürün Sayılarına Göre Test Edilmesi .....	106
<b>9.</b>	<b>SONUÇLAR VE ÖNERİLER .....</b>	<b>108</b>
	<b>KAYNAKLAR .....</b>	<b>111</b>
	<b>EKLER.....</b>	<b>117</b>
	EK-1 ILOG OPL 6.3 CP OPTIMIZER KODU .....	117
	EK-2 JACKSON MONTAJ HATTI ÇÖZÜM SONUCU .....	120

## ÇİZELGELER DİZİNİ

<u>ÇİZELGE</u>	<u>Sayfa</u>
3.1 Montaj hattı dengeleme problemlerinin istasyon ve hat karakteristiklerine göre sınıflandırılması(Boysen vd., 2007) .....	23
3.2 Montaj hattı dengeleme problemlerinin öncelik diyagramı karakteristiklerine göre sınıflandırılması (Boysen vd., 2007). .....	24
3.3 Montaj hattı dengeleme problemlerinin hat dengelemede kullanılan amaçlara göre sınıflandırılması (Boysen vd., 2007) .....	25
6.1 Jackson Probleminden Testler için Oluşturulan Alternatif İş/İstasyon Atamaları ve İstasyon Süreleri.....	46
7.1 Notasyonlar ve açıklamaları.....	60
8.1 Deney Setleri.....	66
8.2 Mertens (1967) 7 işli 2 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri .....	68
8.3 Mertens (1967) 7 işli 2 istasyonlu montaj hattı için performans değerleri .....	68
8.4 Mertens (1967) 7 işli 2 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları .....	68
8.5 Mertens (1967) 7 işli 3 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri .....	70
8.6 Mertens (1967) 7 işli 3 istasyonlu montaj hattı için performans değerleri .....	70
8.7 Mertens (1967) 7 işli 5 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri .....	71
8.8 Mertens (1967) 7 işli 5 istasyonlu montaj hattı için performans değerleri .....	72
8.9 Mertens (1967) 7 işli 5 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları .....	72
8.10 Jaeschke (1964) 9 işli 3 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri .....	74

8.11 Jaeschke (1964) 9 işli 3 istasyonlu montaj hattı için performans değerleri .....	74
8.12 Jaeschke (1964) 9 işli 4 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri .....	75
8.13 Jaeschke (1964) 9 işli 4 istasyonlu montaj hattı için performans değerleri .....	76
8.14 Jaeschke (1964) 9 işli 7 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri .....	77
8.15 Jaeschke (1964) 9 işli 7 istasyonlu montaj hattı için performans değerleri .....	77
8.16 Jaeschke (1964) 9 işli 7 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları .....	78
8.17 Jackson (1956) 11 işli 3 istasyonlu montaj hattı için görev atamaları ve istasyon süreleri .....	79
8.18 Jackson (1956) 11 işli 3 istasyonlu montaj hattı için performans değerleri.....	80
8.19 Jackson (1956) 11 işli 3 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları .....	80
8.20 Jackson (1956) 11 işli 4 istasyonlu montaj hattı için görev atamaları ve istasyon süreleri .....	81
8.21 Jackson (1956) 11 işli 4 istasyonlu montaj hattı için performans değerleri.....	82
8.22 Jackson (1956) 11 işli 5 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri .....	83
8.23 Jackson (1956) 11 işli 5 istasyonlu montaj hattı için performans değerleri.....	83
8.24 Mitchell (1957) 21 işli 3 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri .....	85
8.25 Mitchell (1957) 21 işli 3 istasyonlu montaj hattı için performans değerleri.....	85
8.26 Mitchell (1957) 21 işli 3 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları .....	86
8.27 Mitchell (1957) 21 işli 5 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri .....	87
8.28 Mitchell (1957) 21 işli 5 istasyonlu montaj hattı için performans değerleri.....	87

8.29 Mitchell (1957) 21 işli 5 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları .....	88
8.30 Mitchell (1957) 21 işli 8 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri .....	89
8.31 Mitchell (1957) 21 işli 8 istasyonlu montaj hattı için performans değerleri.....	89
8.32 Mitchell (1957) 21 işli 8 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları .....	90
8.33 Heskiaoff (1968) 28 işli 4 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri .....	91
8.34 Heskiaoff (1968) 28 işli 4 istasyonlu montaj hattı için performans değerleri ...	92
8.35 Heskiaoff (1968) 28 işli 5 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süresi .....	93
8.36 Heskiaoff (1968) 28 işli 5 istasyonlu montaj hattı için performans değerleri ...	93
8.37 Heskiaoff (1968) 28 işli 4 istasyonlu montaj hattı için çoklu arama stratejisi görev atamaları ve ortalama istasyon süreleri .....	95
8.38 Heskiaoff (1968) 28 işli 4 istasyonlu montaj hattı için çoklu arama stratejisi performans değerleri.....	95
8.39 Heskiaoff (1968) 28 işli 5 istasyonlu montaj hattı için çoklu arama stratejisi görev atamaları ve ortalama istasyon süreleri .....	96
8.40 Heskiaoff (1968) 28 işli 5 istasyonlu montaj hattı için çoklu arama stratejisi performans değerleri.....	97
8.41 Sawyer (1970) 30 işli 5 istasyonlu montaj hattı için çoklu arama stratejisi görev atamaları ve ortalama istasyon sayıları.....	98
8.42 Sawyer (1970) 30 işli 5 istasyonlu montaj hattı için çoklu arama stratejisi performans değerleri.....	99
8.43 Sawyer (1970) 30 işli 8 istasyonlu montaj hattı için çoklu arama stratejisi görev atamaları ve ortalama istasyon süreleri .....	100
8.44 Sawyer (1970) 30 işli 8 istasyonlu montaj hattı için çoklu arama stratejisi performans değerleri.....	100

8.45 Sawyer (1970) 30 işli 13 istasyonlu montaj hattı için çoklu arama stratejisi görev atamaları ve ortalama istasyon süreleri .....	102
8.46 Sawyer (1970) 30 işli 13 istasyonlu montaj hattı için çoklu arama stratejisi performans değerleri.....	102
8.47 Öncelikle derinlik arama stratejisi için çözüm sonuçları .....	104
8.48 Çok noktalı arama stratejisi için çözüm sonuçları .....	105

## ŞEKİLLER DİZİNİ

<u>ŞEKİL</u>	<u>Sayfa</u>
1.1 Örnek bir montaj hattı resmi 1 .....	2
1.2 Örnek bir montaj hattı resmi 2 .....	3
1.3 Montaj hattının kuyruk terminolojisinde gösterimi .....	3
3.1 9 görevli bir montaj hattı için örnek öncelik diyagramı.....	20
4.1 Açık ve kapalı kuyruk ağları (sıra ile).....	27
4.2 Kuyruk ağlarındaki temel topolojiler .....	28
4.3 3 istasyon ve 5 müşteriye bir kapalı ağ için durum geçiş diyagramı.....	35
6.1 Jackson Montaj Hattı Şebeke Serimi .....	45
6.2 Arena 11 programında oluşturulan kapalı kuyruk ağ sistemi modeli .....	49
6.3 Sistem çıktı hızlarının karşılaştırılması .....	50
6.4 İstasyon 1 için kullanım oranları karşılaştırılması .....	50
6.5 İstasyon 2 için kullanım oranları karşılaştırılması .....	51
6.6 İstasyon 3 için kullanım oranları karşılaştırılması .....	51
6.7 İstasyon 4 için kullanım oranları karşılaştırılması .....	52
6.8 İstasyon 5 için kullanım oranları karşılaştırılması .....	52
6.9 Ortalama istasyon kullanım oranlarının karşılaştırılması .....	53
7.1 Montaj hattında ürünlerin taşınması için oluşturulmuş alanlar .....	56
7.2 Çözüm metodolojisi IDEF diyagramı .....	59
8.1 Mertens (1967) 7 işli montaj hattı serimi .....	67
8.2 Mertens (1967) 7 iş 2 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	69
8.3 Mertens (1967) 7 iş 3 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	71

8.4 Mertens (1967) 7 iş 5 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	73
8.5 Jaeschke (1964) 9 işli montaj hattı serimi.....	73
8.6 Jaeschke (1964) 9 iş 3 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	75
8.7 Jaeschke (1964) 9 iş 4 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	76
8.8 Jaeschke (1964) 9 iş 7 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	78
8.9 Jackson 11 işli montaj hattı serimi.....	79
8.10 Jackson(1956) 11 iş 3 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	81
8.11 Jackson (1956) 11 iş 4 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	82
8.12 Jackson(1956) 11 iş 5 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	84
8.13 Mitchell (1957) 21 işli montaj hattı serimi.....	84
8.14 Mitchell(1957) 21 iş 3 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	86
8.15 Mitchell (1957) 21 iş 5 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	88
8.16 Mitchell (1957) 21 iş 8 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	90
8.17 Heskiaoff (1968) 28 işli montaj hattı serimi.....	91
8.18 Heskiaoff (1968) 28 iş 4 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	92
8.19 Heskiaoff (1968) 28 iş 5 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği.....	94

8.20 Heskiaoff (1968) 28 iş 4 istasyonlu montaj hattı için çok noktalı arama stratejisine göre çıktı hızlarının değişim grafiği .....	96
8.21 Heskiaoff (1968) 28 iş 5 istasyonlu montaj hattı için çok noktalı arama stratejisine göre çıktı hızlarının değişim grafiği .....	97
8.22 Sawyer 30 işli montaj hattı serimi.....	98
8.23 Sawyer (1970) 30 iş 5 istasyonlu montaj hattı için çok noktalı arama stratejisine göre çıktı hızlarının değişim grafiği .....	99
8.24 Sawyer (1970) 30 iş 8 istasyonlu montaj hattı için çok noktalı arama stratejisine göre çıktı hızlarının değişim grafiği .....	101
8.25 Sawyer (1970) 30 iş 13 istasyonlu montaj hattı için çok noktalı arama stratejisine göre çıktı hızlarının değişim grafiği .....	103
8.26 Değişik ürün sayılarına göre Jackson (1956) montaj hattı 5 istasyonlu durumu için çıktı hızlarının değişim grafiği .....	106



## 1. GİRİŞ

Montaj hatları, ürünü oluşturan alt parçaların birleştirilmelerinin sıralı iş istasyonlarında yapıldığı üretim hatlarıdır. Montaj hatlarında istasyonlar, işler arasındaki öncelik ilişkilerine göre işlerin atanmasıyla oluşmaktadır ve her bir istasyonun süresi atanan işlerin sürelerinin toplanmasıyla bulunmaktadır. Montaj hatlarında istasyon sürelerinin aynı veya birbirine en yakın olacak şekilde görevlerin istasyonlara atanması ve böylelikle hat verimini yükseltmek amacıyla yapılan çalışmalara montaj hattı dengeleme denilmektedir.

Montaj hatlarının üretimde yaygınlaşması ile hatlarda, istasyon sürelerinin farklarından kaynaklanan dengesizlikler, hat boyunca istasyon süresi yüksek olan istasyonlarda yaşanan tıkanmalar, sürekli aynı işi yapmaktan dolayı işçilerde oluşan monotonluklar vb. birçok problem ortaya çıkmıştır. Bu problemlerin çözümü için de araştırmacılar literatürde yer almış birçok çözüm önerisi geliştirmişlerdir.

Bunlara ek olarak montaj hatlarında karşılaşılan diğer bir sorun ise, istasyona atanan işlerin her zaman aynı sürede yapılamamasından dolayı istasyon süresinin değişmesidir. Bu nedenle, istenen çevrim süresinin içinde işler tamamlanamamaktadır. İstasyon sürelerindeki bu değişkenliğin çeşitli olasılık dağılımlarına uyduğu kabul edilmiştir. Bu tip istasyon sürelerinin değişkenlik gösterdiği hatlar stokastik işlem süreli montaj hatları olarak adlandırılmıştır. Bu konuda literatürde bulunan çözüm yaklaşımlarına çalışmanın literatür taraması bölümünde yer verilmiştir.

Stokastik montaj hatlarının dengelenmesi ise, istasyon süresinin sabit olduğu (deterministik) montaj hatlarının dengelenmesine göre daha zordur. Stokastik montaj hatlarında değişkenlikten kaynaklanan çevrim süresinin aşılması ihtimali montaj hattının dengelenmesini daha zor kılmaktadır. Stokastik montaj hatları için, hattın kurulması ve istasyon sürelerinin belirlenmesinin yanında kurulumundan sonra hattın istenen performans değerlerini ve çıktıyı sağlayıp sağlamadığının belirlenmesi de önem arz etmektedir.

Şekil 1.1 ve 1.2’de örnek bir montaj hattının resimleri bulunmaktadır. Şekil 1.1 ve 1.2’de görüldüğü üzere istasyonlardan önce işlem bekleyen ürünler bulunmaktadır. İnsanın yapısından kaynaklanan yorulma, dikkatin dağılması gibi kişisel sebeplerden ve/veya araç-gereç arızası, monotonluk gibi dış etkenlerden kaynaklanan sebeplerden dolayı istasyonlar için belirlenen istasyon süresinde işlerin yapılamaması ve istasyon sürelerinde meydana gelen değişimler ürünlerin istasyonlar arasında beklemesine neden olmaktadır. Bu bekleme nedenleri diğer bir sebebi ise istasyon süreleri arasındaki farklılıklardır. Sistem performansının ölçülmesinde istasyonlar arasındaki bekleyen ürünlerde dikkate alınması gereklidir.

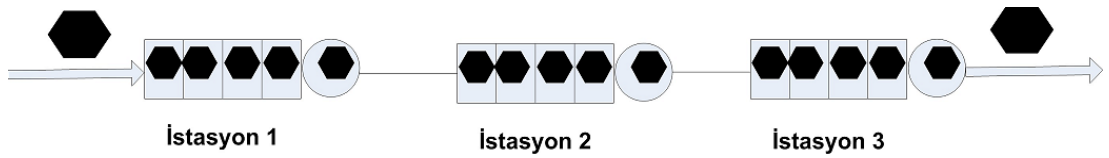


Şekil 1.1 Örnek bir montaj hattı resmi 1



**Şekil 1.2** Örnek bir montaj hattı resmi 2

Şekil 1.3'te düz bir montaj hattının kuyruk ağı terminolojisindeki şekilsel ifadesi verilmiştir. İstasyonlar, ürünlerin işlem gördüğü alanlardan ve bekleme yerlerinden oluşmaktadır. İşlem alanları dairesel şekille ifade edilmiştir. Bekleme yerleri ise dikdörtgen şekiller ile gösterilmiştir. İstasyonlarda ürüne katma değer sağlandığını göstermek amacıyla her bir istasyondaki ürün farklı bir şekille ifade edilmiştir. Hattın sonunda nihai ürün içi taralı üçgensel bir şekille gösterilmiştir.



**Şekil 1.3** Montaj hattının kuyruk terminolojisinde gösterimi

Montaj hatlarının dengelenmesi problemini ele alan arařtırmacılar genel olarak iki ana amaca gre dengeleme alıřmaları zerine yoęunlařmıřlardır. Bu amalardan birincisi ve arařtırmacıların daha ok zerinde durdukları ama olan, sabit evrim sresi kısıdı altında istasyon sayısının minimizasyonudur (tip-1). Dięer bir ama ise sabit istasyon sayısı kısıdı altında evrim sresinin minimizasyonudur (tip-2). Ancak, bu konuda yapılan alıřmalar evrim sresinin sabit kabul edildięi alıřmalara gre daha az sayıdadır. Ayrıca bu iki ama dıřında montaj hattının tipine ve sistemin gereksinimlerine gre alıřma ve kurulum maliyetlerinin minimizasyonu, hat etkinlięinin artırılması, farklı amaların birleřtirilmesi gibi deęiřik amalar dikkate alınarak arařtırmacılar tarafından hat dengeleme alıřmaları yapılmıřtır.

Bu alıřmada stokastik iřlem sreli, dz, tek modelli, gecikmeli montaj hatlarının belirlenen sabit istasyon sayısı kısıdı altında evrim sresinin minimizasyonu amalanmıřtır. Bu ama erevesinde montaj hattının ıktı hızı, istasyon kullanım oranı, istasyonda bulunan rn sayısı, bir rnn istasyonda geirdięi sre gibi performans deęerlerini hesaplamak iin kapalı kuyruk aęı modeli kullanılmıřtır.

alıřmada nerilen zm algoritması  ařamadan oluřmaktadır. Genel hatları ile řu řekilde zetlenebilir: Birinci ařamada kısıt programlama modeli kullanılarak ve ncelik iliřkileri dikkate alınarak oluřturulabilecek alternatif grev-istasyon kombinasyonları belirlenmekte ve bylelikle mevcut olabilecek btn istasyon atamaları deęerlendirilebilmektedir. İkinci ařamada, kapalı kuyruk aęı modeli ile her bir alternatifin performans deęerleri hesaplanmaktadır. nc ve son ařamada ise alternatifler arasından en iyi ıktı hızı deęerine sahip olan atama kombinasyonu belirlenmekte ve algoritma sonlandırılmaktadır. Bylece montaj hattı iin en iyi ıktı hızını veren ve hattın en verimli alıřacaęı grev istasyon ataması bulunmuřtur.

### **1.1. Tez alıřmasının İerięi**

Tez alıřması genel hatları ile řu blmlerden oluřmaktadır: Giriř blmnn ardından ikinci blmde, tip 2 ve stokastik montaj hattı tipleri iin literatr arařtırması verilmiřtir. Ayrıca, literatrde montaj hattında kuyruk aęı modelleri

kullanılan alıřmalar zetlenmiřtir. Üüncü bölümde, montaj hattı ve montaj hattı dengeleme kavramları açıklanmıřtır. Dördüncü bölümde, kuyruk modelleri ve kuyruk ađı modelleri hakkında kavramsal bilgilerden ve yöntemin genel özelliklerinden bahsedilmiřtir. Ortalama deđer analizi formülasyonu hakkında bilgi verilmiřtir. Beřinci bölümde, kısıt programlama yönteminden bahsedilmiř ve yöntemde kullanılan arama stratejileri açıklanmıřtır.

Altıncı bölümde, hattın stokastikliđi ve istasyon sürelerinin üssel dađılıma uyumu test edilmiřtir. İstasyon sürelerinin normal dađılıma uyduđu durumlar ile üssel dađılıma uyduđu durum simülasyon modeli kurularak test edilmiřtir. Yedinci bölümde, ele aldığımız stokastik tip-2 montaj hattı dengeleme probleminin tanımı yapılmıř ve sistemin kabulleri ve varsayımları verilmiřtir. Problemin özümü için oluřturulan kısıt programlama ve kapalı kuyruk ađı modeli tabanlı yeni bir özüm metodolojisi tanıtılmıřtır. Montaj hattı matematiksel modeli ve matematiksel modele dayalı olarak oluřturulan kısıt programlama yapısı verilmiřtir. Sekizinci bölümde, deneysel alıřmalar ve sonuçları verilmiřtir. Dokuzuncu bölümde ise sonuçlardan elde edilen ıkarımlar verilmiřtir ve ilerde yapılabilecek alıřmalara iliřkin önerilerde bulunulmuřtur.

## 2. LİTERATÜR ARAŞTIRMASI

Bu bölümde tip 2 ve stokastik düz montaj hatları ve kuyruk modelleri ile ilgili literatür taramalarına yer verilmiştir. Kuyruk modelleri ile ilgili literatür taramasında montaj hattı ile ilgili çalışmalara yoğunlaşmıştır.

### 2.1. Montaj Hattı Dengeleme Problemleri

Çalışmamızda ele aldığımız problemde tek modellenli gecikmeli stokastik işlem zamanlı montaj hatları için sabit istasyon sayısı kısıdı altında çevrim süresinin minimizasyonu amaçlanmıştır. Literatür çalışmamızda öncelikle istasyon sayısının sabit olması durumunda çevrim süresinin minimizasyonunu (Tip 2) amaçlayan tek modellenli hatlarla ilgili yapılan çalışmalar verilmiştir. Devamında ise stokastik işlem zamanlı montaj hattı çalışmaları özetlenmiştir.

Montaj hatlarının dengelenmesi için ilk çalışmayı Bryton yüksek lisans tezinde ortaya atmıştır. Sabit iş istasyonu sayısında görevleri istasyonlar arasında değiştirerek istasyonlar arasındaki süre dengesizliğini azaltmaya ve iş istasyonu sürelerini eşitlemeye çalışmıştır. (Bryton, 1954) Montaj hatlarının dengelenmesi ile ilgili ilk makale Salveson'un çalışmasıdır. Montaj hattı dengeleme probleminin 0-1 tam sayılı modelini ortaya koymuştur. Modelin amaç fonksiyonu istasyonlardaki boş zamanı en küçüklemektir. (Salveson, 1955)

İstasyon sayısının sabit olması durumunda çevrim süresinin minimizasyonunu ele alan çalışmalar aşağıdaki gibi özetlenebilir.

Tip 2 montaj hattı dengeleme problemlerini ele alan ilk çalışma Helgeson ve Bernie'nin yaptığı çalışmadır. Geliştirdikleri sezgisel çalışma iterasyonlara dayalı çözüm prosedürüne dayanmaktadır. Tip 2 probleminin çözümü, tip 1 probleminin seri iterasyonlarla çözülmesi ile elde edilmektedir. (Helgeson ve Bernie, 1961)

Mansoor, sıralanmış pozisyon ağırlıkları yöntemine dayanan bir algoritma geliştirmiştir. Bu yöntemde, görevin süresi ve görevden sonra gelen bütün görevlerin sürelerinin toplamı alınarak göreve pozisyon ağırlığı verilir. Görevler pozisyon ağırlıklarına göre artan sırada sıralanır. Çevrim zamanı ve öncelik diyagramları kısıtları dikkate alınarak görevler bu sıralamaya göre istasyonlara atanır. Bütün görevler atanınca algoritma durdurulur. Atamalar sonucunda belirlenen istasyon sayısınınca atama yapılamamış ise, çevrim süresi bir artırılarak işlem tekrarlanır. Ancak istenen istasyon sayısına ulaşılamaz ise istasyon sayısı bir artırılır. Bu metot tam olarak sayım metodudur. (Mansoor, 1964)

Mansoor ve Yadin, bir başka prosedür önermişlerdir. Minimum teorik çevrim süresinden başlayarak ve sırayla geçerli listeden görevlerin ataması yapılır. Metot boş bir atama seti ile başlar. Görevler her seferinde bir tane olmak üzere kendisinden sonra gelen işlerin uygun setinden eklenir. Birinci istasyona uyan bütün alt setler oluşturulunca tamamlanır. Bu prosedür diğer istasyonlar içinde tekrarlanır. Eğer oluşturulan bütün alt setler verilen çevrim zamanı için ilgili istasyonlara atanabiliyor ise, bütün optimal çözümler bulunur. Optimal sonuç bulunamamış ise çevrim süresi bir artırılarak prosedür tekrarlanır. (Mansoor ve Yadin, 1971)

Rao, pozisyon ağırlığı metoduna benzer bir çalışma yapmıştır. Teorik minimum istasyon sayısını kullanarak teorik toplam boş zamanı hesaplamıştır. Daha sonra istasyonları kurmuş ve toplam boş zamanı güncellemiştir. Bütün istasyonlar kurulana kadar bu işlemler tekrarlanır. Eğer istasyon çevrim zamanını aşmış ise son görev silinir. Eğer hiçbir uygun çözüm bulunamaz ise, çevrim zamanı bir artırılır. İşlem sürelerinin sabit ve normal dağılıma uyduğu stokastik durumlar için karşılaştırmalar yapmıştır. (Rao, 1971)

Wee ve Magazine, geliştirdikleri dal-sınır algoritması ile dört değişik arama metodunu tip 2 problemlerinin çözümü için kullanmışlardır. Metotlardan iki tanesi ayrı ayrı alt ve üst sınırlarla başlamaktadır. Diğer iki metot ise ikili arama algoritması (BISEARCH) ve ikili Fibonacci arama prosedürüdür. Bu iki algoritmada da tip 1 probleminin çözümünde elde edilen sonuçlardan yararlanılmaktadır. (Wee ve Magazine, 1981)

Hackman vd., sezgisel metotlarla çözüm ağacını küçülttükleri bir dal-sınır algoritması geliştirmişlerdir. Bu algoritma ile öncelikle tip 1 problemlerini çözmüşler ve 53 örnek problem için çözümleri karşılaştırmışlardır. Algoritma ile tip 2 problemlerinin de çözülebileceğini göstermişlerdir. (Hackman vd., 1989)

Klein ve Scholl, tip 2 problemlerinin çözümü için dal-sınır algoritması tanımlamışlardır. Önerilen çözüm yönteminde yerel alt sınır yöntemi olarak tanımladıkları bir sayım tekniği yöntemi kullanmışlardır. Bu yöntemde tamamlayıcı olarak sınır sayıları ve baskınlık kurallarını kullanmışlardır. Önerdikleri yöntem SALOME-2 olarak literatüre girmiştir. (Klein ve Scholl, 1996)

Scholl ve Vob, çalışmalarında tip 1 ve tip 2 problemlerinin çözümü için sezgisel bir metot ortaya koymuşlardır. Bu metot da, ilk çözümü bulmada öncelik tabanlı metoda dayalı bir geliştirme prosedürü kullanılmıştır. Yöntemde yeni genişletilmiş dinamik kurallara ve çift yönlü planlamaya dayalı yeni öncelik kurallarını önermişlerdir. Bunlara ek olarak, istasyonlar arasında görevlerin değişimine dayanan yeni bir geliştirme prosedürünü tarif etmişlerdir. Bu prosedürü tabu arama algoritmasının statik versiyonu ile birleştirilmişlerdir. (Scholl ve Vob, 1996)

Uğurdağ vd., tip 2 problemlerinin çözümü ve montaj hattı tasarımı için iki aşamalı bir çözüm prosedürü geliştirmişlerdir. Prosedürün ilk aşamasında sezgisel bir yaklaşımla başlangıç çözümünü elde edilmektedir. İkinci aşamasında ise başlangıç çözümünü geliştiren yeni bir yaklaşımı ortaya koymuşlardır. Çalışmada çevrim süresini minimize ederken iş yükü dağılımının değişiminin düzgünleştirilmesi de amaçlanmıştır. Bu karakteristikleri dikkate alarak simpleks algoritmasına benzeyen yeni bir yaklaşımı önermişlerdir. (Uğurdağ vd., 1997)

Rekiek vd., tip 2 problemlerinde farklı iş istasyonları için proses zamanlarının eşit olduğu kabulünü yapmışlardır. Öncelik ilişkilerine ek olarak, görevleri ayırmak ve gruplamak için bazı öncelik ilişkisi kısıtlarını algoritmaya dahil ederek genetik algoritma tabanlı bir çözüm metodu geliştirmişlerdir. (Rekiek vd., 1999)



Carnahan vd., Tip-2 probleminin çözümünde işlerde istenilen fiziksel yetenekleri dikkate almıştır. Seviyelendirme sezgiselleri, kombinatoriyal genetik algoritma ve problem uzayı genetik algoritmadan oluşan üç metot geliştirmişlerdir. Bunlar eşzamanlı olarak en büyük fiziksel ihtiyacı ile çevrim süresini en küçüklemeyi amaçlamaktadırlar. Problem uzayı genetik algoritmasının diğer yöntemlerden daha iyi sonuç verdiğiğine karar vermişlerdir. (Carnahan vd., 2001)

Liu vd., tip 2 problemi için stokastik görev zamanlı durumda bir çözüm yöntemi geliştirmişlerdir. Bunun için üç aşamalı bir yöntem uygulamışlar, birinci aşamada ileri veya geri yönlü olarak görevler iş istasyonlarına atanmış, ikinci aşamada görevler, istasyonlar arasında değiştirilerek düzgünlük sağlanmış ve son olarak da ikinci aşamadan elde edilen çevrim zamanı üst sınırı düşürülerek en düşük çevrim zamanı elde edilmiştir. (Liu vd., 2005),

Pastor ve Ferrer, çalışmalarında daha etkin yeni bir matematiksel model önermişlerdir. Bu modele göre görevlerin atanabileceği iş istasyonları, iş istasyonlarının üst sınırı veya çevrim sürelerinin üst sınırına göre belirlenmektedir. Yeni modelden elde edilen sonuçları literatürde bulunan diğer modellerle karşılaştırmışlardır. (Pastor ve Ferrer, 2009),

Wei ve Chao, çalışmalarında tip 1 ve tip 2 problemlerinin birleşiminden oluşan ve tip E olarak adlandırılan hat etkinliğini artırmayı amaçlayan problem tarzı için bir model önermişlerdir. Modelde çevrim zamanını minimize ederken istasyon boş zamanlarını azaltmayı amaçlamışlardır. Model verilen çevrim süresi üst sınırına göre minimum değişken, kısıt ve hesaplama zamanına göre optimum çözümü bulmaktadır. (Wei ve Chao, 2011)

Nourmohammadia ve Zandieh, önerdikleri metotta MODEA olarak adlandırdıkları bir çözüm yöntemini geliştirmişlerdir. Bu yöntemde montaj hattının çevrim süresinin ve düzgünlük endeksinin birlikte minimizasyonunu amaçlamışlardır. Genetik algoritma kullandıkları çalışmalarında, alternatif atamalar arasında seçim yapmak için pareto analizine ve TOPSIS'e dayalı geliştirme şemasını kullanmışlardır. Ayrıca faktörler arasında önem derecesini belirlemek için Taguchi metodunu

kullanmışlardır. Çalışmanın etkinliğini çok amaçlı dengeleme yöntemini kullanan çalışmaların sonuçları ile karşılaştırmışlardır. (Nourmohammadia ve Zandieh, 2011)

Stokastik montaj hattı dengeleme problemleri için yapılan çalışmalar ise aşağıda verilmiştir.

Modie ve Young, görev tamamlanması için verilen belirli bir güven aralığı sağlanacak şekilde görev zamanlarının değişkenliğini göz önüne almıştır. Bu çalışmada görev zamanlarının olasılık dağılımı için normal dağılım varsayımı yapılmıştır. (Modie ve Young, 1965)

Kottas ve Lau, stokastik montaj hattı için maliyet tabanlı bir sezgisel sunmuşlardır. Çalışmada görev zamanlarının normal dağılıma uyduğu, tamamlanmama durumunda ise birimin hatta devam ederek öncelik diyagramının izin verdiği görevlerin yapıldığı, yapılamayan görevlerin ise hat dışında yapıldığı varsayılmıştır. Geliştirilen sezgisel, çözüm zamanı açısından etkin olmasına rağmen, tek bir sonuç sunmakta, farklı çözümler incelememektedir. (Kottas ve Lau, 1973)

Kottas ve Lau, önceki çalışmalarına ek olarak tüm tamamlanmama kombinasyonlarını göz önüne alarak dengelenmiş hatlar için yeni bir maliyet fonksiyonu sunmuşlardır. Bu maliyet fonksiyonu tamamlanamama durumlarının olasılıkları göz önüne alınarak tanımlanmıştır. (Kottas ve Lau, 1976)

Diğer iki çalışmasını temel alan Kottas ve Lau, rastsallık içeren ve farklı çözümleri inceleyen bir sezgisel yaklaşım kullanmış, elde edilen sonuçlar içinden geliştirdikleri maliyet fonksiyonuna göre en iyi değeri veren çözümü olarak sonuca ulaşmışlardır. (Kottas ve Lau, 1981)

Kao, tip I stokastik montaj hattı dengeleme problemi için istasyon sayısının en küçüklenmesini amaçlayan ve yönetim tarafından belirlenmiş çevrim zamanını aşma olasılığını geçmeyecek şekilde dengeleme sağlayan bir dinamik programlama (DP) algoritması geliştirmiştir. Bu çalışmada görev süreleri değişik dağılımlara uyduğu durumlar incelenmiştir. Sürelerin normal, gama, binomial ve poisson dağılıma uyduğu

durumlar ele alınmıştır. (Kao 1976) Kao, bir diğer çalışması ile bilgisayar çözümünü hızlandırabilecek tercih ve sıralama prosedürü ve buna uygun programlama stratejisi önermiştir. (Kao, 1979)

Raouf ve Tsui, değişim katsayı yanlılığına dayalı farklı olasılık dağılımlara sahip stokastik hatlar için bir sezgisel önermişlerdir. Burada işlere öncelik tanımı yapılmakta ve buna göre atama yapılması sağlanmaktadır. Böylece çevrim zamanından sapmanın en aza indirilmesini sağlamışlardır. (Raouf ve Tsui, 1982)

Silverman ve Carter, stokastik hatlarda istasyon zamanının aşılması durumunda ihtiyaç olan zaman kadar tüm hattın durdurulduğunu göz önüne alarak bir maliyet fonksiyonu geliştirilmiştir. Stokastik maliyet fonksiyonu etkin bir dengeleme algoritmasına bütünleştirilerek yaklaşık en küçük maliyetli denge oluşturulmaya çalışılmıştır. Fakat geliştirilen fonksiyonun uygulanması, içerdiği matematiksel ifadelerin karmaşıklığı nedeniyle zordur. (Silverman ve Carter, 1986)

Henig, çalışmasında çevrim zamanının en küçüklenmesi, istasyon sayısının en küçüklenmesi ve istenilen güven seviyesinin en büyüklenmesi amaçları için Dinamik Programlama (DP) modelleri geliştirmiştir. (Henig, 1986)

Shin, istasyon zamanlarına bağlı bir maliyet fonksiyonu geliştirmiş ve toplam maliyeti en küçükleyen çevrim zamanını bulmayı amaçlayan bir sezgisel önermiştir. Sezgisel aynı zamanda her bir iş istasyonunun belirli çevrim zamanı ve değişkenliği için gerekli emniyet zamanlarını da araştırmaktadır. (Shin, 1990)

Suresh ve Sahu, tavlama benzetimi kullanarak stokastik montaj hattı dengeleme problemi için hat durmasının en küçüklenmesini amaçlamışlardır. Böylece yüksek görev sayısına sahip hat problemlerine uygun çözümler üretmeye çalışmışlardır. (Suresh ve Sahu, 1994)

Nkasu ve Leung, iki asamadan oluşan bir çözüm yöntemi önermiştir. Birinci modülde COMSOAL sezgiselinin geliştirilmiş bir versiyonu ile çözüm üretilmiş, bu şekilde istasyon sayısının en küçüklenmesi, denge gecikmesinin en küçüklenmesi ve

çevrim zamanının en küçüklenmesi ya da bunların birleşimini sağlayacak dört amaç fonksiyonu ele alınmıştır. İkinci modül ile simülasyon yapılarak sistem denenmiştir. (Nkasu ve Leung, 1995)

Sarin vd., tarafından, tek modelli stokastik montaj hattı dengeleme problemi için başlangıç çözümünün DP ile bulunduğu, daha sonra alt sınır yaklaşımı yerine düğüm azaltma yöntemli bir yaklaşımı kullanan dal-sınır algoritması ile çözümün geliştirildiği bir yöntem sunulmuştur. (Sarin vd., 1999)

Gökçen ve Baykoç, stokastik görev zamanlı MHD problemlerinde tamamlanamayan işleri seyyar bir istasyon vasıtasıyla yaptırarak bu işlerin hattın etkinliğine olan etkisini azaltmıştır. Bu çalışmanın etkinliğini test etmek için simülasyon kullanılmıştır. (Gökçen ve Baykoç, 1999)

Tip 2 montaj hattı dengeleme problemleri ile ilgili literatürde bahsedilen Liu vd. (2005) diğerlerinin çalışmasında da görev süreleri stokastik alınmıştır.

Ağpak ve Gökçen, çalışmalarında düz ve u-tipi montaj hatlarının dengelenmesinde şans kısıtlı yaklaşımı önermişlerdir. Görev sürelerinin stokastikliğinden kaynaklanan sistem güvenilirliğini artırmak için amaç programlama yaklaşımını kullanmışlardır. (Ağpak ve Gökçen, 2007)

Ayazi vd., stokastik montaj hatları için çok amaçlı bir yaklaşım çalışmışlardır. Bu amaçlar çevrim süresinin minimizasyonu, istasyon sayısının minimizasyonu ve iş yüklerinin sapmalarının minimizasyonudur. Problemin matematiksel modelini ortaya koymuşlardır ve problemi çözmek için genetik algoritmayı kullanmışlardır. (Ayazi vd., 2011)

Bu çalışmalar dışında karışık model, u-tipi vb. stokastik montaj hatları ile ilgili birçok çalışma araştırmacılar tarafından literatüre kazandırılmıştır. Ancak konunun çok geniş olmasından dolayı bu çalışmalara yer verilmemiştir. Daha ayrıntılı literatür taramaları için, Boysen vd. (2007), Baybars (1986), Erel ve Sarin (1998) ve Becker ve Scholl (2006) montaj hatları konusunda yaptıkları literatür çalışmalarına

başvurulabilir. Yazarlar, çalışmalarında montaj hattının genel sınıflandırılmalarını da yapmışlardır. Ayrıca, Scholl ve Becker yaptıkları çalışmadan yola çıkarak montaj hattı konusunda çok kullanışlı ve araştırmacılara kaynak niteliğinde olan [www.assembly-line-balancing.de](http://www.assembly-line-balancing.de) sitesini kurmuşlardır.

## 2.2. Kuyruk Ağı Modelleri

Kuyruk ağı modellerini kullanarak montaj hattı konusunda yapılan çalışmaların bir özeti verilmiştir. Ayrıca sistem performansının ölçülmesinde kuyruk ağı modellerini kullanan çalışmalardan örnekler verilmiştir.

Montaj hatlarında görevlerin dağıtıldığı ve ürün üzerinde işlemlerin gerçekleştirildiği istasyonların öncesinde işlenecek ürün bulunmaması istasyonun aç beklemesine ve bir istasyon için bekleme yerlerinin tam dolu olması istasyonun bloke olmasına sebep olur. Bu yüzden bekleme yeri sayısı montaj hatları için önem kazanmaktadır. Harrison, bekleme yeri kapasitesinin 2'den büyük olduğu durumlar için montaj hattı istasyonunu ele almıştır. Bekleme yeri kapasitesinin sonsuz olmadığı durumlarda montaj hattındaki kuyruk uzunluğunun istenen sabit dengeli durumuna ulaşmadığını belirlemiştir. Herhangi bir girdi prosesinin kuyruğu sınırsız bir şekilde büyümektedir. Birçok analitik model bu problemi sınırlı bekleme yeri kapasitesi olduğunu varsayarak veya kaynağın kuyruğuna olan girişleri kontrol altında tutmaya çalışarak aşmaya çalışmışlardır. (Harrison, 1973)

Latouche and Neuts, gelişlerin Poisson ve işlem sürelerinin üssel dağılıma uyduğu durumlarda, bekleme zamanlarının sabit olduğunu göstermişlerdir. (Latouche and Neuts, 1980)

Montaj hattı için gerçek hayatta karşılaşılan durum ise bekleme yerlerinin sınırlı ve sabit olmasıdır. Ammar, montaj hatlarının deterministik işlem sürelerine ve sınırlı bekleme yeri kapasitelerine sahip olduğu durumlar için analiz etmiştir. İstasyonların değişken başarısızlıklara sahip olduğunu varsaymıştır. (Ammar, 1980)

Lipper ve Sengupta, farklı sınıflardaki müşteri gelişlerinin olduğu montaj istasyonunu ele almıştır. Bu farklı sınıftaki müşterilerin, aynı geliş oranlarına sahip bağımsız poisson prosesine sahip olduklarını ve servis zamanlarının üssel dağılıma uyduğunu kabul etmiştir. Sınıfların farklı bekleme yerlerine sahip olduğu ve bu bekleme yerlerinin kapasitelerinin eşit ve sınırlı olduğu varsayılmıştır. Performans değerleri olarak bloklama olasılığı, çıktı hızı, ortalama kuyruk uzunluğu ve sistemde geçen ortalama süre alınmıştır. Problemin çözümünde sonuç olarak uygun bir çözüme ulaşılamamışlar ve bir yaklaşım modelini önermişlerdir. (Lipper ve Sengupta, 1986)

Gershwin, bozulma ihtimali olan makinelerin ve sonlu bekleme yerlerinin olduğu montaj hatlarını ele almıştır. Üretim oranı ve ortalama envanter seviyesini içeren performans ölçütleri hesaplanmıştır. Model K adet tekil-bekleme yeri sistemli K tane bekleme yeri kapasiteli montaj hattına benzemektedir. Bu tekil-istasyon bekleme yeri sistemleri transfer hatları modeli ile analiz edilmiştir. Daha sonra bu yaklaşımı uzun transfer hatları ile birleştirmiştir. Yaklaşım algoritması M/G/1/B kuyruk modeli serilerini iterasyonlarla çözmektedir. (Gershwin, 1986)

Duenyas and Hopp, birden fazla sıralı hatlar tarafından beslenen montaj sistemini analiz etmiştir. Bu sıralı hatların deterministik işlem zamanlı bozulma riski bulunan makineler içerdiği varsayılmıştır. Bu model üssel dağılım sürelerine sahip olduğu varsayılarak genişletilmiştir. (Duenyas and Hopp, 1992)

Azaron vd., çalışmalarında çok aşamalı montaj sistemleri için teslim zamanının optimum kontrolünü sağlamak için ayrışma metodunu kullanarak çok amaçlı bir model geliştirmişlerdir. Montaj sistemleri açık kuyruk ağları ile modellenmiştir. Bu hattın istasyonlarında üretim veya montaj operasyonları yapılabilmektedir. Sisteme gelişlerin poisson prosesine uyduğu ve işlem zamanlarının üssel olduğu ve servis kapasiteleri kontrol edilebilen bir veya sonsuz sayıda işlemciye sahip olan istasyonların olduğu kabulü yapılmıştır. İstasyonlar arası taşımaların bağımsız rastgele değişkenlere sahip erlang dağılımına uyduğu kabul edilmiştir. Problemden bir periyottaki sistem maliyetlerinin minimizasyonu, teslim süresinin minimizasyonu, teslim süresinin varyansının minimizasyonu amaçlamışlardır. Problemin kontrol

matrisini ve sistemin optimum servis kapasitesini belirlemek için hedef elde etme metodunu kullanmışlardır. (Azaron vd., 2006)

Azaron ve Kianfar, Azaron vd.'nin çalışmalarında ele aldıkları probleme ek olarak bir amaç daha eklemişler ve dört amaçlı bir yaklaşımı çalışmalarında ele almışlardır. Bu amaçlar; bir periyottaki sistem maliyetlerinin minimizasyonu, teslim süresinin minimizasyonu, teslim süresinin varyansının minimizasyonu ve üretim teslim zamanının olasılığının belirli bir eşiği aşmayacak şekilde maksimizasyonudur. (Azaron ve Kianfar, 2006) Perkgoz vd. (2006) ise Azaron ve Kianfar'ın çalışmalarında ele aldığı problemi çalışmışlardır ve genetik algoritma ile çözümünü elde etmişlerdir.

Manitz, kademeli montaj hatlarının üretim prosesini çalışmıştır. Bu tip hatlar için bir istasyona gelen ürünler kendinden önce gelen istasyonlardan gelmekte ve bunların gelişleri senkronize olmamış stokastik zamanlarla olduğunu belirtmiştir ve bekleme yerlerinden dolayı kuyruk etkilerinin ortaya çıktığını belirlemiştir. Genel dağılımlı proses zamanlarının ve sonlu bekleme yeri kapasitesine sahip olduğunu varsaymıştır. Bu sistemde ürünlere yapılan işlemlerin kuyruk sistemlerine benzediğini ortaya koymuş ve bu tip montaj hatlarının çıktı oranlarını belirlemek amacıyla bir yaklaşım prosedürünü geliştirmiştir. (Manitz, 2007)

Lazaro ve Perez, gerçek bir montaj hattını bloklanma ve boş kalma olasılığı olan makinelerden ve konveyör ile oluşmuş bekleme yerlerine sahip kapalı döngü ağ olarak modellemişlerdir. Ana montaj hattı ile kapı montaj hatları için makinelerin sabit ve geçici çalışma durumları için makine ve istasyonların boş kalma veya bloklanma durumlarını analiz etmişlerdir. Her bir istasyon için ayrı ayrı durumların incelemesini yapmışlardır. (Lazaro ve Perez, 2009)

Yapılan literatür çalışması sonucunda, tip 2 montaj hattı dengelenme problemlerinin, genelde tip-1 montaj hattı dengeleme problemleri için kurulan modellerin algoritmik olarak çözülmesi ile çözüldüğü görülmüştür. Çalışmamızda ise tip 2 montaj hattı dengeleme problemlerinin direk çözümünü veren yeni bir model önerilmiştir.

Literatüre yapılan diđer bir katkı ise, montaj hatlarının dengelenmesinde kapalı kuyruk ađı modellerinin kullanılmıř olmasıdır. Esnek imalat sistemleri, bilgisayar iřletim sistemleri gibi birok alanda kullanılan kapalı kuyruk ađlarının montaj hattı dengeleme problemlerinin dengelenmesinde kullanılması literatürde yeni bir bakıř aısı kazandırmaktadır. Bunlara ek olarak, modelde önerilen kısıt programlama modeli görev atamalarında kullanılan yeni bir yöntem olarak literatüre katkı sağlanmaktadır.



### 3. MONTAJ HATTI VE MONTAJ HATTI DENGELEME

Montaj hatları birim zamanda üretim hızını artırmak amacıyla ilk olarak Henry Ford tarafından uygulanmıştır. Ürünü oluşturan parçaların belirli bir sıra ile ürünü oluşturmak için birleştirilmesi ve/veya ürün üzerindeki işlemlerin yapılması işine montaj, ürünlerin sıralı iş istasyonları boyunca ilerlemesi ile montaj işleminin yapılabilmesi için kurulan üretim hattına ise montaj hattı denir.

#### 3.1. Montaj Hattı Çeşitleri

Montaj hatları çeşitli özelliklerine göre sınıflandırılmıştır. Bu sınıflar;

a. Yapısına göre montaj hatları:

- Manüel montaj hatları: İstasyonlar arası taşımalar manüel olarak yapılmaktadır.
- Otomatik montaj hatları: İstasyonlar arası taşımalar konveyör gibi taşıyıcılar ile yapılmaktadır.

b. Üretilen model çeşitlerine göre montaj hatları:

- Tek modelli montaj hatları: Bu tip montaj hatlarında tek çeşit ürün üretilmektedir.
- Çok modelli montaj hatları: Birden fazla tip ürün üretilen hatlardır.
- Karışık modelli montaj hatları: Birden fazla tip ürünlerin karışık olarak üretildiği hatlardır.

c. Fiziksel yerleşimlerine göre montaj hatları: Düz, dairesel, u-tipi, zigzag, değişik açılı gibi yerleşim tiplerine ayrılmaktadır.

d. Gecikmeli ve gecikmesiz montaj hatları:

- Gecikmeli Montaj Hattı: Bu tip hatlarda, bir istasyonda işi biten parça eğer bir sonraki istasyonda işlenen parça var ise ara stoklarda beklemeye başlar. Bir diğer özelliği ise işi biten istasyon bekleyen

parça yok ise ve diğer istasyonda parçanın işi bitmemiş ise istasyon boş beklemektedir. Bu durumlardan dolayı istasyonlar boş bekleyebilmekte veya dar boğazlar oluşabilmektedir.

- Gecikmesiz Montaj Hattı: Bu hat tipinde ise istasyonlara belirlenen çevrim süresini aşmasına izin verilmemektedir. Çevrim süresi dolduğunda parçalar diğer istasyona geçmektedir. Tamamlanamayan görevler ise hat dışında bir istasyonda tamamlanmaktadır. İstasyonlar arasında parçaların taşınması sabit hızla hareket eden bir konveyör ile veya kesikli olarak hareket eden bir taşıma sistemi ile yapılmaktadır.

e. İşlem sürelerine göre montaj hatları:

- Deterministik işlem süreli montaj hatları: İşlem sürelerinin sabit olduğu, işlem sürelerinde değişkenliğin olmadığı montaj hatlarıdır.
- Stokastik işlem süreli montaj hatları: İşlem sürelerinin belirli bir olasılık dağılımına göre değişkenlik gösterdiği montaj hatlarıdır.

f. Çift ve tek taraflı montaj hatları: Montaj hattında istasyonlar hattın bir tarafına kurulmuş ise tek taraflı, hattın iki tarafına kurulmuş ise çift taraflı montaj hattı olarak adlandırılır.

### **3.2. Montaj Hattı Dengeleme Problemi**

Montaj hatlarında en önemli unsur yapılacak olan işlemlerin hat etkinliğini artırmak amacıyla hangi istasyonlarda yapılacağı belirlenmesidir. Hat etkinliğini artırmak amacıyla işlerin istasyonlara, işler arasındaki öncelik ilişkileri dikkate alınarak, istasyon süreleri eşit veya en yakın olacak şekilde atanmasına montaj hattı dengelemesi olarak adlandırılmıştır.

Montaj hattı dengeleme çalışmasının yapılabilmesi için öncelikle hat boyunca yapılacak işlemlerin, bu işlemlere ait işlem sürelerinin ve işlemlerin öncelik ilişkilerinin belirlenmesi gerekmektedir. Bu iş etüdü çalışmalarının tamamlanmasından sonra hattın dengelenmesinde kullanılacak amaç ve parametreler

belirlenmeli ve son olarak belirlenen amaca göre hangi işin hangi istasyonda yapılacağı bulunmalıdır.

Montaj hattı dengeleme problemi ilk olarak Bryton (1954) tarafından yapılan tez çalışmasında ortaya konulmuştur. Analitik olarak montaj hattının incelendiği ilk çalışma olarak literatüre girmiştir. 1954 yılından bu yana birçok çalışma yapılmış, montaj hattı ve montaj hattı dengeleme konuları endüstri mühendisliğinin temel çalışma konuları arasında yerini almıştır. Ele aldığımız problem türü ile ilgili yapılan çalışmalardan, çalışmanın literatür araştırması bölümünde bahsedilmiştir.

Montaj hattı dengeleme problemlerinde görevlerin istasyonlara dağıtımı için yapılan çalışmalarda görevlerin atanmasında ve istasyonların oluşturulmasında aşağıdaki kısıtların sağlanması gereklidir

- a) Bütün görevler bir istasyona atanmalıdır.
- b) Bir görev sadece bir istasyona atanabilir.
- c) Görevler bölünemezler.
- d) Her bir istasyon için toplam süre, hattın çevrim süresini geçemez.
- e) Görevlerin istasyonlara atanmasında öncelik diyagramı bozulmayacak şekilde bir atama yapılmalıdır.

### 3.3. Montaj Hattı Dengeleme İçin Temel Kavramlar

Montaj hattı dengeleme problemlerinde kullanılan temel kavramlar aşağıdaki gibidir.

*Montaj:* Bir ürünü oluşturmak için alt parçaların birleştirilmesi işlemidir.

*Operasyon/görev/işlem:* Bir montaj prosesinde, yapılacak işlerin ayrıldığı en küçük parçadır.  $i$  indisi ile gösterilir. Toplam operasyon sayısı ise  $N$  ile ifade edilir. Operasyon zamanı ise  $t_i$  ile gösterilir.

*İş istasyonu:* Operasyonların yapılması için montaj hattı üzerinde belirlenen alandır.  $k$  indisi ile gösterilir.

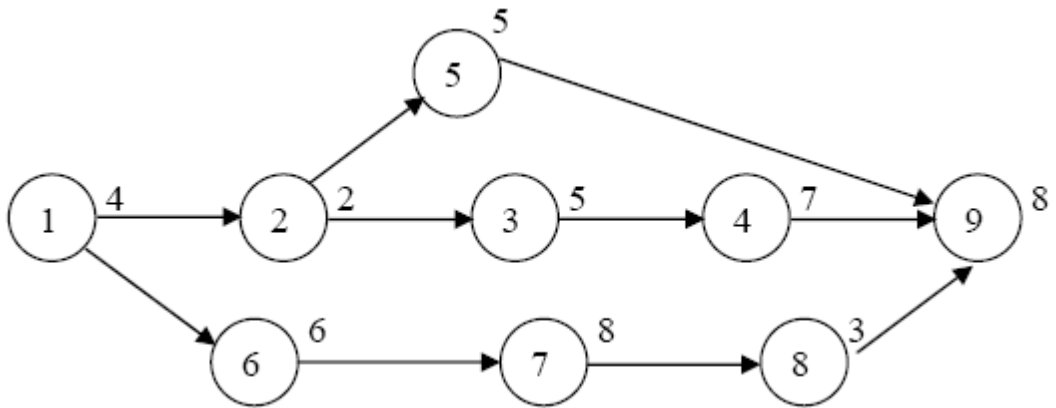
*Çevrim zamanı:* Montaj hattında, ürünün bir istasyonda geçirebileceği en büyük süre veya bir iş istasyonundaki çalışanın o istasyonda yapılması gerekli işleri tamamlaması için gerekli süre olarak tanımlanabilir. Çevrim süresini seçmekteki ana düşünce, gerek duyulan üretim hızıdır. Çevrim zamanı  $C$  ile gösterilir.

*İstasyon Zamanı:* İstasyona gelen bir parça üzerinde o istasyonda yapılması gereken ilk operasyonun başlangıç anı ile son operasyonun bitiş anı arasındaki süre farkıdır.  $j$  istasyonuna ait zaman  $S_j$  ile gösterilir ve montaj hattındaki bir istasyon zamanı, çevrim zamanından küçük veya çevrim zamanına eşit olmalıdır.

*Toplam İş Zamanı:* Montaj hattı üzerinde bir ürünün montajı için gerekli olan sürelerin veya işi oluşturan tüm operasyonların standart sürelerinin toplamıdır.  $T$  ile gösterilir.

*İstasyon Gecikme Zamanı/Boş zaman:* Çevrim süresi ile iş istasyonu zamanı arasındaki fark iş istasyonunun boş zamanını göstermektedir. İstasyondaki boş zaman  $C - S_j$  olarak hesaplanır.

*Öncelik Diyagramı:* Montajın teknik özelliklerinden dolayı, bazı operasyonların zorunlu olarak birbirini izlemesi gerekir. Bu özelliklerin tümü öncelik ilişkileri adı altında toplanır. 9 işli bir montaj hattı için öncelik diyagramı bir ağ yapısı şeklinde Şekil 3.1’de bir örneği verilmiştir.



**Şekil 3.1** 9 görevli bir montaj hattı için örnek öncelik diyagramı

*Denge kaybı:* Bir montaj hattında istasyonlarda oluşan boş zamanların ortalamasının çevrim süresine bölünmesi ile bulunur. Montaj hattında bir kaynağın ne kadar atıl olarak kaldığının göstergesidir.

$$DK = \frac{(n * C - \sum t_i)}{n * C} x 100 \quad (2.1)$$

*Hat etkinliği:* Hattın verimlilik ölçütü olarak kullanılır ve aşağıdaki gibi hesaplanır.

$$HE = \frac{\sum t_i}{n * C} \quad (2.2)$$

### **3.4. Montaj Hattı Dengelemenin Amaçları**

Montaj hattı dengeleme çalışmaları çevrim zamanını minimize etmek ve istasyon sayısını minimize etmek olmak üzere iki temel amaca göre yapılmaktadır. Bu amaçlar göz önünde bulundurularak montaj hattı dengeleme problemleri iki ana kategoriye ayrılmaktadır. Bu iki ana kategori:

- Çevrim zamanı verilen bir hat için istasyon sayısını minimize etmektir.(Tip I)
- İstasyon sayısı verilen bir hat için çevrim zamanını minimize etmektir.(Tip II)

Bunların dışında montaj hattının kurulması ve dengelenmesi ile ulaşılmak istenen bazı amaçlar şu şekilde sıralanabilir:

- Düzenli malzeme akışını sağlamak.
- İnsan gücünün optimum kullanımını sağlamak.
- İş yeri düzenini sağlayarak ürün akışının kontrolünü kolaylaştırmak.
- Ürünlerin çevrim süresinin azaltılmasını sağlamak.
- Üretim maliyetlerini düşürmek.
- Üretim planlamasını kolaylaştırmak.

- Teslim zamanlarının en doğru şekilde tahmin edilmesini sağlamak.
- Tesis yerleşimindeki karmaşıklığı gidermek.(Başkak, 1991)

### **3.5. Montaj Hattı Dengeleme Problemlerinin Sınıflandırılması**

Montaj hattı dengeleme problemleri genel montaj hattı tiplerine ve dengelemede kullanılan amaçlara göre çeşitli kategorilere ayrılmaktadır. 1954'ten günümüze sanayinin gelişimi ile birlikte çok çeşitli montaj hatları oluşturulmasına paralel olarak çok çeşitli tipte montaj hattı dengeleme problemleri ortaya çıkmıştır. Çizelge 3.1, 3.2 ve 3.3'de montaj hattı dengeleme problemlerinin sınıflandırılmasını veren geniş bir liste verilmiştir. Çizelgelerde hat dengeleme problemlerinin sınıflandırılmasında kullanılan ana kategori üst başlık olarak verilmiş ve bu başlıkların her bir alt başlığı kalın kenarlıklarla ayrılmıştır.

**Çizelge 3.1** Montaj hattı dengeleme problemlerinin istasyon ve hat karakteristiklerine göre sınıflandırılması (Boysen vd., 2007)

<b>İstasyon ve hat karakteristikleri</b>	
<ul style="list-style-type: none"> <li>❖ İş parçalarının taşınabilirliği           <ul style="list-style-type: none"> <li>➤ Gecikmeli hatlar               <ul style="list-style-type: none"> <li>▪ Çevrim zamanına bağlı olarak ortalama iş içeriği kısıdı</li> <li>▪ Her bir model çevrim süresini karşılamalıdır</li> <li>▪ Çevrim süresi verilen olasılık ile aşılabildiği</li> <li>▪ Tek global çevrim zamanı</li> <li>▪ Lokal çevrim zamanı</li> </ul> </li> <li>➤ Gecikmesiz montaj hatları               <ul style="list-style-type: none"> <li>▪ Senkronize olmamış hatlar</li> <li>▪ Senkronize hatlar</li> </ul> </li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>❖ Kaynak atamaları           <ul style="list-style-type: none"> <li>➤ Ekipman seçim problemleri</li> <li>➤ Ekipman tasarım problemleri               <ul style="list-style-type: none"> <li>▪ Bir istasyona bir kaynağı paylaşan iki görevin ataması ve kurulum maliyetinin düşmesi</li> <li>▪ Birden fazla görevin aynı kaynağı paylaştığı durumlar</li> <li>▪ Diğer bağımlılık ve sinerji durumları</li> </ul> </li> <li>➤ Kaynak atamalarının dikkate alınmadığı durumlar</li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>❖ Hat yerleşimi           <ul style="list-style-type: none"> <li>➤ Seri hatlar</li> <li>➤ U tipi hatlar               <ul style="list-style-type: none"> <li>▪ Tek U tipi hat</li> <li>▪ Çoklu U tipi hatlar</li> </ul> </li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>❖ İstasyon bağımlı zaman artışları           <ul style="list-style-type: none"> <li>➤ Üretim dışı işlemlerin dikkate alınması</li> <li>➤ İstasyon bağımlı zaman artışlarının göz önüne alınmadığı</li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>❖ Parallelleşme           <ul style="list-style-type: none"> <li>➤ Paralel hatlar</li> <li>➤ Paralel istasyonlar</li> <li>➤ Paralel görevler</li> <li>➤ İstasyon içinde paralel çalışma alanları</li> <li>➤ Parallelleşmenin göz önüne alınmaması</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>❖ Diğer konfigürasyon yaklaşımları           <ul style="list-style-type: none"> <li>➤ Bekleme yeri atamaları ve kapasitelerinin belirlenmesi</li> <li>➤ Besleme hatlarının birlikte dengelenmesi</li> <li>➤ Malzeme kutularının yerleştirilmesi ve kapasitelerinin belirlenmesi ihtiyacı</li> <li>➤ İşlemlerin pozisyon değişiklikleri için makinelere ihtiyaç duyulması</li> <li>➤ Ek hat konfigürasyonlarının olmadığı durumlar</li> </ul> </li> </ul>

**Çizelge 3.2** Montaj hattı dengeleme problemlerinin öncelik diyagramı karakteristiklerine göre sınıflandırılması (Boysen vd., 2007).

<b>Öncelik Diyagramı Karakteristikleri</b>	
❖ Ürün özellikli öncelik diyagramı	❖ Atanma kısıtları
➤ Karışık model üretim	➤ Bağlı görevlerin aynı istasyona atanmasının zorunlu olması
➤ Çok model üretim	➤ Zıt görevlerin aynı istasyona atanmaması gerekliliği
➤ Tek model üretim	➤ Görev-istasyon atanmasında birliktelik kısıdı
❖ Öncelik diyagramının yapısı	➤ Belirli görevlerin özel istasyonlara atanması kısıdı
➤ Özel öncelik diyagramı yapısında sınırlamalı	➤ Görevlerin belirli istasyonlara atanmaması kısıdı
➤ Öncelik diyagramının çevrimsel yapıya sahip olmaması	➤ Görevlerin belirli tip istasyona atanması zorunluluğu
❖ İşlem zamanları	➤ Görevler arasında minimum mesafenin sağlanması kısıdı
➤ Stokastik işlem zamanı	➤ Görevler arasındaki en yüksek mesafenin aşılmaması kısıdı
➤ Dinamik işlem zamanı (öğrenme etkisi gibi)	➤ Hiçbir atama kısıdının olmadığı durumlar
➤ Deterministik işlem zamanı	❖ Proses alternatifleri
❖ Sıra bağımlı işlem zamanı yapıları	➤ Proses zamanının ve maliyetinin değişkenliği
➤ İşlemlerinin birbirlerine direk bağımlı olduğu durumlar	➤ Öncelik kısıtlarının bütün olarak değişkenliği
➤ İşlemlerin birbirlerini takip ettiği durumlar	➤ Görevler arasında alt önceliklerin değişkenliği
➤ Sıra bağımlılığın göz önüne alınmadığı durumlar	➤ Proses alternatiflerinin olmadığı durum



**Çizelge 3.3** Montaj hattı dengeleme problemlerinin hat dengelemede kullanılan amaçlara göre sınıflandırılması (Boysen vd., 2007)

❖ Amaçlar
➤ İstasyon sayısının minimize edilmesi
➤ Çevrim zamanının minimize edilmesi
➤ Hat etkinliğinin maksimize edilmesi
➤ Maliyet minimizasyonu
➤ Kar maksimizasyonu
➤ Karışık amaçların minimize veya maksimize edilmesi
➤ Sadece uygun çözümlerin aranması

Çizelge 3.1, 3.2 ve 3.3'te de görüldüğü üzere montaj hattı dengeleme problemleri çeşitliliği bu konuda yapılan çalışmaların çok geniş bir alanı kapsadığını göstermesi açısından önemlidir. Ayrıca, bu çeşitlilik montaj hattı alanında çalışma yapacak araştırmacılar içinde çok geniş bir çalışma alanının olduğunu da göstermektedir.

#### 4. KUYRUK AĞLARI İLE SİSTEM PERFORMANSININ DEĞERLENDİRİLMESİ

Kuyruk modellerinde, işlem görmesi gereken veya hizmet alacak öğeler belirli bir gelişler arası süre ile sisteme girerler ve kuyrukta bekletilirler. Sisteme gelen öğeler belirlenen kuyruk disiplinine (FIFO-ilk gelen ilk çıkar, LIFO-son gelen ilk çıkar gibi) göre işlem görürler. İşlem gören öğeler sistemden çıkar. Bu sistemin bütününe kuyruk sistemi denilir. Bu tip sistemler tek kanallı kuyruk sistemleri olarak adlandırılır.

Kuyruk sistemlerini tanımlamak için Kendall tarafından önerilmiş olan üç bölümlü A/B/C sistemi kullanılır. A gelişler arası sürenin dağılımını ifade etmektedir, B işlem sürelerinin dağılımını ifade etmektedir, C ise sistemde bulunan işlemci sayısını belirtmektedir. Sisteme gelişler arası sürelerin ve işlem sürelerinin dağılımları üssel, normal, uniform olasılık dağılımlarına uyduğu gibi sabit değerlerde olabilmektedir.

Kuyruk ağları ise, birden fazla kuyruk sisteminin birbirini takip etmesinden oluşan sistemler olarak tanımlanmaktadır. Kuyruk ağları sistemdeki ürün sayısının değişkenliğine göre açık kuyruk ağları ve kapalı kuyruk ağları (Şekil 4.1) olmak üzere ikiye ayrılmaktadır.

- *Açık kuyruk ağları;* müşterilerin/ürünlerin bir veya daha fazla yolla dışarıdan gelerek hizmet aldıkları ve sonrasında sistemi terk ettikleri modellerdir. Açık ağlarda sistemdeki müşteri sayısı random (rastgele) gelişlere ve terk edilmişlere bağlı olarak zamanla değişir.
- *Kapalı kuyruk ağları;* sistemdeki müşteri sayısının sabit olduğu ve sürekli bir şekilde sistemde dolaştıkları modellerdir. Bu ağlarda sisteme dışarıdan girişler veya sistemden dışarı çıkışlar yoktur.



**Şekil 4.1** Açık ve kapalı kuyruk ağları (sıra ile)

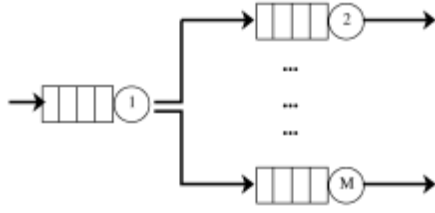
Bu tanımlamaya ek olarak kuyruk ağları sistemdeki müşteri tiplerinin sayısına göre de ayrılırlar: Tek Müşteri Sınıflı Kuyruk Ağları ve Çok Müşteri Sınıflı Kuyruk Ağları.

- *Tek Müşteri Sınıflı veya sadece Tek Sınıflı kuyruk ağları* sistemdeki müşterilerin aynı özellikte varsayıldığı sistemlerdir.
- *Çok Müşteri Sınıflı veya sadece Çok Sınıflı kuyruk ağları*, sistemdeki müşterilerin değişik özellikte olduğu, işlem sürelerinin, izlediği yolların vb niteliklerin müşterilerin tiplerine bağlı olduğu sistemlerdir.

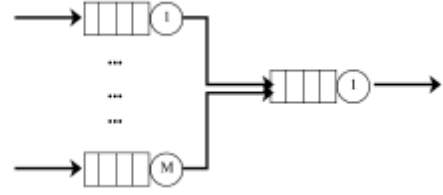
Topoloji yönünden kuyruk ağları, istasyonların üç temel düzenlemesinin kombinasyonundan oluşmaktadır: tandem (sıralı), birleşen ve ayrışan (Şekil 4.2). Tandem topolojide istasyonlar birbiri ardınca sıralanmışlardır. Eğer ağ, kapalı bir ağ ise bu tip sistemlere Döngüsel Ağ adı verilir. Ayrışan topolojilerde bir istasyon  $M(>1)$  istasyona ilintilenmiştir. İlk seviye  $i$  istasyonunda işlemi biten bir müşteri, takip eden seviyedeki bir  $j$  istasyonuna  $p_{ij}$  olasılığı ile ilerler. Birleşen topolojide ise birden çok akış bir istasyona ilintilenmiştir.



a) Tandem (Sıralı) topoloji



b) Ayrışan topoloji



c) Birleşen topoloji

Şekil 4.2 Kuyruk ağlarındaki temel topolojiler

Bunlara ek olarak kuyruk ağları blokeli ve blokesiz kuyruk ağları olarak da sınıflandırılabilir.

- *Blokesiz Kuyruk Ağları:* Ağdaki bütün kuyrukların sonsuz kapasitede olduğu sistemlerdir. Sınırsız sistemler olarak da adlandırılır.
- *Blokeli Kuyruk Ağları:* Ağdaki bazı veya tüm istasyonların bekleme yerlerinin belirli bir sayıda olduğu sistemlerdir. Sınırlı sistemler adı da verilir. Böyle bir ağda, sınırlı sayıda bekleme yeri olan kuyruğa ilerleyen müşteri eğer bekleme yerleri dolu ise bloke durumu ile karşılaşır. Bu durum üç ana bloke çeşidi ile betimlenir:
  - *İşlem Sonrası Bloke:* Daha çok üretim hatlarında görüldüğü için üretim blokesi diye de adlandırılır. Bu bloke durumunda, istasyon  $i$ 'de işlemini bitiren müşteri bir sonraki istasyon  $j$ 'ye ilerlemek ister. Eğer istasyon  $j$  dolu ise  $i$ 'inci istasyonu işgal etmeye devam eder ve bu istasyon  $i$  bloke olmuş olur.  $i$ 'deki işlemci, sırasında bekleyen başka müşterilere hizmet veremez.
  - *İşlem Öncesi Bloke:* Daha çok bilgisayar ve telekomünikasyon ağlarında görüldüğünden iletişim blokesi diye de adlandırılır. İstasyon  $i$ 'deki müşteri işleme başlamadan sonraki istasyonu  $j$ 'yi

tayin eder. Eğer o istasyon dolu ise  $i$ 'inci istasyon bloke olur.  $j$ 'den bir müşteri ayrıldığında  $i$  bloke durumundan kurtulur ve işleme başlar. Eğer işlem süresinde  $j$  tekrar dolarsa, işlem kesilir ve istasyon  $i$  bloke olur. Tayin edilen istasyon  $j$  boşalır boşalmaz  $i$ 'de işleme kaldığı yerden devam edilir.

- *Tekrarlı Bloke:*  $i$  istasyonunda işlemini tamamlayan müşteri sonraki  $j$  istasyonuna gitmek ister. Eğer istasyon  $j$  dolu ise  $i$ 'deki kuyruğun en arkasına geçer. Bu süreç müşteri  $j$  istasyonuna ilerleyene kadar devam eder.

Bu tezde ele alınan problem tek modelli stokastik montaj hatları olması sebebiyle ve diğer yandan kuyruk ağlarının sınıflandırma özellikleri de dikkate alınarak performans değerlendirmesinde tek sınıflı kuyruk ağları kullanılmıştır. Yine aynı amaçlar doğrultusunda tandem blokesiz kuyruk ağları kullanılmıştır. Bu bölümde tek sınıflı ağlar detaylı bir şekilde ele alınmıştır.

#### 4.1. Tek Sınıflı Kuyruk Ağları

Tek sınıflı kuyruk ağları algoritmalarında kullanılan parametrelere ait notasyonlar aşağıdaki gibidir.

$M$	İstasyon sayısı
$N$	Müşteri/Ürün sayısı
$n_i$	$i^{\text{inci}}$ istasyondaki müşteri sayısı ( $\sum_i^M = N$ )
$(n_1, n_2, \dots, n_M)$	Ağın anlık durumu
$\pi_i(n_1, n_2, \dots, n_M)$	$i^{\text{inci}}$ istasyonda $n_i$ müşteri olma olasılığı ( $i = 1, \dots, M$ )
$\mu_i$	$i^{\text{inci}}$ istasyonun işlem oranı ( $i = 1, \dots, M$ )
$1/\pi_i$	$i^{\text{inci}}$ istasyondaki işlerin ortalama süresi ( $i = 1, \dots, M$ )
$V_i$	$i^{\text{inci}}$ istasyona geliş oranı ( $i = 1, \dots, M$ )
$r_{ij}$	Bir müşterinin $i^{\text{inci}}$ istasyonda işlemini tamamladıktan sonra $j^{\text{inci}}$ istasyona gelme olasılığı, rota olasılığı ( $i \neq j, i =$

	$1, \dots, M, j = 1, \dots, M)$
$\pi_i(j n)$	Sistemde $n$ müşteri varken $i^{inci}$ istasyonda $j$ iş olma koşullu olasılığı ( $i = 1, \dots, M, j = 0, \dots, n, n = 0, \dots, N$ )
$K_i$	$i^{inci}$ istasyonun işlemcilerde dahil olmak üzere bekleme yeri kapasitesi ( $i = 1, \dots, M$ )
$c_i$	$i^{inci}$ istasyondaki paralel işlemcilerin sayısı ( $c_i \geq 1, i = 1, \dots, M$ )
$b_i$	$i^{inci}$ istasyondaki bekleme yeri sayısı ( $i = 1, \dots, M$ )
$\lambda$	Sistemin çıktı hızı
$\lambda_i$	$i^{inci}$ istasyondaki çıktı hızı ( $i = 1, \dots, M$ )
$\rho_i$	$i^{inci}$ istasyonun İstasyon Doluluk Oranı ( $i = 1, \dots, M$ )
$W$	Çevrim zamanı, müşterilerin sisteme giriş anından çıktıkları ana kadar geçen veya bir çevrimi tamamladıkları süre.
$W_i$	$i^{inci}$ istasyondaki ortalama süre ( $i = 1, \dots, M$ )
$Q_i$	$i^{inci}$ istasyondaki ortalama müşteri sayısı ( $i = 1, \dots, M$ )

Kuyruk ağları tek kanallı sistemlere göre daha çok istasyondan oluşmaktadır. Yalnız her iki sistemde de önemli olan karar durum olasılıklarının hesaplanmasıdır. Bu olasılıkların hesaplanması ile diğer ölçütler de hesaplanabilir. Kuyruk ağları ile sistem analizinde göz önüne alınan performans ölçütleri aşağıdaki gibidir.

- Karar Durum Olasılıkları,  $\pi_i(k)$

$$\pi_i(k) = \sum_{n_i=n} \pi(n_1, n_2, \dots, n_M) \quad (4.1)$$

Buradaki ifade de  $i=1,2, \dots, M$ 'deki istasyonlardaki  $n_i$  müşteri olasılıkları olarak okunabilir.

- İstasyon Doluluk Oranı,  $\rho_i$

$$\rho_i = \sum_{k=1}^{\infty} \pi_i(k) \quad (4.2)$$

Ayrıca  $\rho_i$  istasyonunun doluluk oranı olduğundan, tek işlemcisi olan istasyonlar için şu şekilde de hesaplanabilir.

$$\rho_i = 1 - \pi_i(0) \quad (4.3)$$

$c_i$  sayıda işlemcisi olan bir istasyon için ise:

$$\rho_i = 1 - \sum_{k=0}^{c_i-1} \frac{c_i - k}{c_i} \pi_i(k) \quad (4.4)$$

Eğer işlem oranı müşteri sayısından bağımsız ise:

$$\rho_i = \frac{\lambda_i}{c_i \mu_i} \quad (4.5)$$

- Çıktı Hızı,  $\lambda_i$  : Birim zamanda istasyonu terk eden işlerin sayısıdır.

$$\lambda_i = \sum_{k=1}^{\infty} \pi_i(k) \mu_i(k) \quad (4.6)$$

Eğer işlem oranı müşteri sayısından bağımsız ise;

$$\lambda_i = c_i \rho_i \mu_i \quad (4.7)$$

Bunlara ek olarak, denge halinde çıktı hızı (birim zamanda çıkan müşteri sayısı) ile girdi hızı (birim zamanda giren müşteri sayısı) eşittir. Eğer bekleme yerleri sınırlı ise bu durumda gelen müşterilerin bir kısmı sistemden ayrılmak zorunda kalacaktır ve çıktı hızı girdi hızından düşük olacaktır.

- Sistem Çıktı Hızı,  $\lambda_i$  : Açık ağlar için sistem çıktı hızı, birim zamanda çıkan müşterilerin sayısıdır. Denge durumunda çıktı hızı sisteme giren akışların toplamıdır. Yani;

$$\lambda = \sum_{i=1}^N \lambda_{0i} \quad (4.8)$$

Kapalı sistemler için ise müşterilerden birim zamanda bir turu tamamlayanların sayısıdır.

$$\lambda = \frac{\lambda_i}{e_i} \quad (4.9)$$

- Ortalama Müşteri Sayısı,  $Q_i$  : bir  $i$  istasyonundaki ortalama müşteri sayısı.

$$Q_i = \sum_{n=1}^{\infty} n \pi_i(k) \quad (4.10)$$

Little Teoreminden;

$$Q_i = \lambda_i W_i \quad (4.11)$$

- Kuyruktaki Ortalama Müşteri Sayısı,  $q_i$  : Bir  $i$  istasyonundaki kuyrukta bekleyen ortalama müşteri sayısı.

$$q_i = \sum_{n=c_i}^{\infty} (n - c_i) \pi_i(k) \quad (4.12)$$

Little Teoreminden;

$$q_i = \lambda_i \omega_i \quad (4.13)$$

- İstasyondaki Ortalama Süre,  $W_i$  : Bir müşterinin  $i$  istasyonda harcadığı ortalama zaman



$$W_i = \frac{Q_i}{\lambda_i} \quad (4.14)$$

- İstasyondaki Ortalama Kuyruk Süresi,  $\omega_i$  : Bir müşterinin  $i$  istasyonunda kuyrukta harcadığı ortalama zaman

$$\omega_i = W_i - \frac{1}{\mu_i} \quad (4.15)$$

#### 4.1.1. Matematiksel Formdaki Sistemler

Matematiksel-form terimi, üssel gelişler arası ve işlem zamanları olan ağlarda önce açık sistemler için Jackson (1963) ve sonrasında kapalı sistemler için Gordon ve Newell (1967) tarafından kullanılmıştır. Bu sistemlerde kuyruk disiplini First In First Out –İlk Gelen İlk İşlenir- (FIFO) olarak varsayılmıştır. Bu ağlar için, karar durum olasılıklarının her bir istasyonu oluşturan faktörlerin çarpımı olarak ifade edilebileceği gösterilmiştir.

Daha sonrasında bu çözüm yöntemi, Baskett vd. (1975) tarafından açık, kapalı veya karışık, çok sınıflı müşteri tipleri de olabilen ve değişik kuyruk disiplinleri içeren kuyruk ağlarına genişletilmiştir. Bu bölümde bu yaklaşım değişik kuyruk ağları için detaylı olarak incelenmiştir.

Bu çözüm yaklaşımlarında normalizasyon katsayısının hesaplanması gereklidir. Bu amaçla, normalizasyon katsayısını hesaplayan Convolution Algoritması Buzen (1973) tarafından önerilmiştir.

Reiser ve Lavenberg (1980) normalizasyon katsayısının hesaplanması güçlüğünü bertaraf eden yeni bir algoritma olan Ortalama Değer Analizini -ODA- (Mean Value Analysis-MVA) geliştirmişlerdir. Kapalı, çok sınıflı ve bekleme yerlerinin sınırsız olduğu ağlar için olan algoritma çevrim zamanı, çıktı hızı ve ortalama müşteri sayıları gibi ilgilenilen performans değerlerini direkt olarak hesaplayan bir

algoritmadır. Little Teoremine ve Arrival (Geliş) teoremine dayanan algoritma iteratif bir şekilde ortalama performans değerlerini hesaplar.

- Little Teoremi: Tüm düğümler veya bütün sistem için ortalama iş sayısı, çıktı hızı ve ortalama işlem zamanı arasındaki ilişkiyi tanımlamaktadır. Teorem bir sistemdeki ortalama iş sayısı ( $K^-$ ), birim zamandaki çıktı hızı ( $\lambda$ ) ve ortalama işlem zamanının ( $T^-$ ) çarpımı ile elde edilebildiğini açıklar.

$$K^- = \lambda \cdot T^- \quad (4.16)$$

Ayrıca ortalama kuyruk uzunluğu ( $Q^-$ ) ise birim zamandaki çıktı hızı ( $\lambda$ ) ile ortalama bekleme zamanının ( $W^-$ ) çarpılması ile hesaplanır.

$$\bar{Q} = \lambda \cdot \bar{W} \quad (4.17)$$

- Arrival Teoremi: Kapalı kuyruk ağı modelleri için,  $i$ 'inci düğümde  $k$  adet iş varken olasılık yoğunluk fonksiyonu, aynı düğümde bir eksik ( $k-1$ ) iş varken işlerin olasılık yoğunluk fonksiyonuna eşittir.

Başka matematiksel formdaki kuyruk ağlarını analiz edebilen algoritma Akış-Denk-İşlemci'dir (Chandy vd., 1975). Bu metot elektrik devrelerindeki Norton Teoremini kuyruk ağlarına uygulamaktadır.

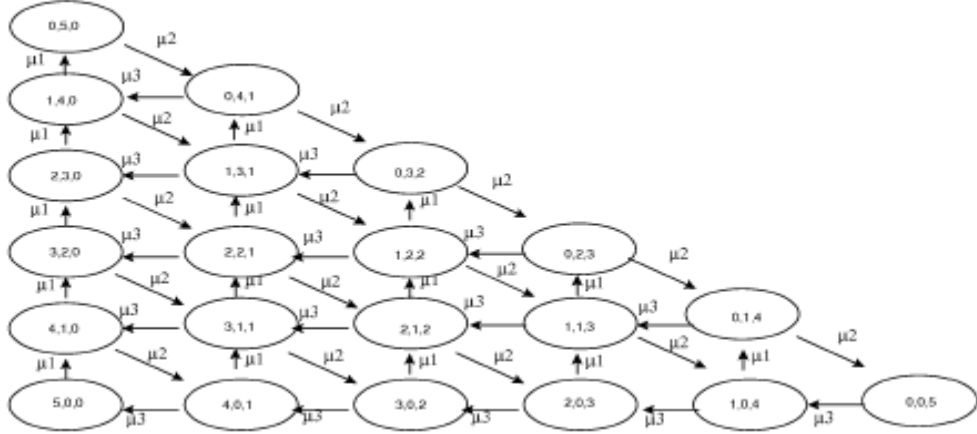
Convolution, Ortalama Değer ve Akış-denk-işlemci algoritmaları aynı zorluk değerlerine sahiptir. Fakat Ortalama Değer Analizi Algoritmasının pratikliği oldukça cezbedici olmuş, hatta algoritma bilimsel olarak yayımlanmadan eksik kaldığı yönlerde yaklaşık algoritmalar geliştirilmiştir (Bard, 1979; Schmidt, 1979). Wang (1997) ODA algoritmasını koşum zorluğu yönünden incelemiş, bu konuda sunulmuş yakınsama algoritmalarını karşılaştırmıştır.

#### 4.1.2. Kapalı Kuyruk Ağlarının Matematiksel Formu

Kapalı ağlarda matematiksel form özelliği Gordon ve Newell tarafından gösterilmiştir. Kapalı ağlarda sisteme giriş ve çıkış yoktur, sistemdeki müşteri sayısı sabittir ( $N = \sum_{i=1}^M n_i$ ). Dolayısıyla mümkün durum sayısı N sayıdaki müşterinin M istasyona dağıtılması olarak şu şekilde binomial olarak ifade edilebilir.

$$\binom{N + M - 1}{M - 1} \quad (4.18)$$

Örneğin bir 3 istasyonlu ve 5 müşterili kapalı ağ için durumlar ve durumlar arası geçiş diyagramı Şekil 4.3 deki gibidir. Aslında bu durumların karar olasılıkları Sürekli-Zamanlı Markov Zinciri yaklaşımı kullanılarak hesaplanabilir. Fakat şebekeler büyüdükçe ve ağdaki müşteri sayıları arttıkça durum sayısı üssel olarak artar.



Şekil 4.3 3 istasyon ve 5 müşterili bir kapalı ağ için durum geçiş diyagramı

Karar durum olasılıklarının hesaplanması ise Gordon ve Newell tarafından şu teoremlerle ispatlanmıştır.

**Teorem 1:** Bir kapalı ağda, ağın karar durum olasılıkları her bir istasyonun karar durum olasılıkları çarpımı olarak şu şekilde ifade edilebilir. Bu durum kuyruk ağlarında Product Form -Çarpım Formu- olarak adlandırılır.

$$\pi(n_1, n_2, \dots, n_M) = \frac{1}{G(N)} \prod_{i=1}^M F_i(n_i) \quad (4.19)$$

Burada  $G(N)$  normalizasyon katsayısı olarak tanımlanır ve şu şekildedir:

$$G(N) = \sum_{\sum_{i=1}^M n_i = N} \prod_{i=1}^M F_i(n_i) \quad (4.20)$$

$F_i(n_i)$  ise istasyonların ziyaret oranları, servis süreleri ve işlemci sayılarına bağlı olarak şu şekilde ifade edilir:

$$F_i(n_i) = \left(\frac{V_i}{\mu_i}\right) \frac{1}{\beta_i n_i} \quad (4.21)$$

$\beta_i(n_i)$  katsayısı ise;

$$\beta_i n_i = \begin{cases} n_i!, & n_i \leq c_i \\ c_i! c_i^{n_i - c_i}, & n_i > c_i \\ 1, & c_i = 1 \end{cases}$$

Normalizasyon katsayısının hesaplanarak karar durum olasılıklarının belirlenmesi kolay bir hesaplama gibi görülebilir. Fakat örneğin 5 istasyon ve 10 müşterili bir küçük ağ için 1001 durum hesaplanması gerekir. Ağ biraz daha genişletilirse yukarıdaki binomial hesabı uyarınca 10 istasyon ve 35 müşteri için 52.451.256 durum hesaplanması gerekir. Dolayısıyla daha verimli algoritmalar geliştirilmiştir.

## 4.2. Ortalama Değer Analizi (ODA)

Ortalama değer analizi kapalı kuyruk ağları için çıktı hızını, İstasyon Doluluk Oranını ve ortalama kuyruk uzunluğunu hesaplayan bir tekniktir. Sisteme bağlı olarak servis ve geliş zamanlarının istatistiksel dağılımlarının başlangıç durumları ile ilişkilidir. Ayrıca, yöntem çarpım formu olarak ifade edilebilen ağların analizinde kullanılan tam sonuç veren bir algoritmadır. İlk defa Reiser ve Lavenberg (1980) tarafından ortaya konulmuştur. Schweitzel (1979) ve Bard (1979) tarafından yöntem geliştirilmiştir. Hildebrant(1980) yöntemin çoklu sınıflar için kullanımını ortaya koymuştur.

### 4.2.1. Tek Sınıflı Blokesiz Sistemler İçin ODA

Tez çalışmasının tek sınıflı blokesiz sistemler için yapılan analizlerinde aşağıdaki ODA algoritması kullanılmıştır.

#### Algoritma 1:

Adım 1. Başlangıç değerlerini ata.  $i = 1, \dots, M$  için

$$\pi_i(0|0) = 1$$

Adım 2. İterasyon:  $n = 1, \dots, N$  tüm popülasyon kombinasyonlarını itere et

Adım 2.1. Her  $i = 1, \dots, M$  için işlem oranlarını hesapla

$$\mu_i(n) = \begin{cases} n\mu_i, & \text{eğer } n \leq c_i \\ c_i\mu_i, & \text{eğer } n > c_i \end{cases}$$

Adım 2.2. Her  $i = 1, \dots, M$  için istasyon  $i$ 'deki ortalama istasyon zamanlarını hesapla

$$W_i(n) = \sum_{j=1}^n \frac{j}{\mu_i(j)} \pi_i(j-1|n-1)$$

Adım 2.3. Sistem çıktı hızını hesapla

$$\lambda(n) = \frac{n}{\sum_{i=1}^M W_i(n) V_i}$$

Adım 2.4.  $i = 1, \dots, M$  için koşullu olasılıkları hesapla

$$\pi_i(j|n) = \frac{\lambda(n)}{\mu_i(j)} \pi_i(j-1|n-1) V_i, \quad \text{her } j = 1, \dots, n \text{ için}$$

$$\pi_i(0|n) = 1 - \sum_{j=1}^n \pi_i(j|n) \quad j = 0 \text{ için}$$

ODA algoritması ile ürünlerin her bir istasyonda geçirdikleri süre, bekleme süreleri, işlem süreleri ve toplam sistemde geçen süre algoritmik hesaplanabilmektedir. Ayrıca, istasyonda bekleyen ürün sayısı, istasyonda işlem gören ürün sayısı, her bir istasyon için istasyon doluluk oranı ve ortalama istasyon doluluk oranları hesaplanabilmektedir. Bir sistem için en önemli performans değerlerinden biri olan sistem çıktı hızı da algoritmik olarak hesaplanmaktadır.

## 5. KISIT PROGRAMLAMA

Kısıt programlama hesaplama ve mantığa dayalı tekniklerin kombinasyonuna dayanan alternatif bir programlama yaklaşımıdır. Ana kavram kısıtlardır. Kısıtlar değişkenlerin bir dizisi şeklinde tanımlanan alanlardan oluşmaktadır. Tanımlı alanlardan oluşan değişkenlerin kombinasyonu olarak tanımlanabilir. Kısıt programlamada ana mantık problemin sahip olduğu kısıtların matematiksel ve mantıksal ifadelerle oluşturulması ve uygun çözüm alanlarının tanımlanmasıdır. Kısıt sağlama problemleri ise her biri değişken dizilerinin oluşturduğu belirli sayıda kısıtları içermektedir. Bütün bu kısıtların sağlandığı çözümleri bulmakta ve bulunan çözümlerin içinden belirlenen amaca göre optimum çözümün bulunmasıdır. Kısıt sağlama problemleri bir amaca bağlı olmadan problemin bütün uygun çözümlerinin bulunması amacıyla da çözülmektedir.

Verilen bir problemi kısıt programlama ile çözmek için kısıt sağlama problemi olarak formülize edilmelidir. Bir problemi kısıt programlama olarak formüle etmek için:

- Değişkenler ve bu değişkenlerle ilişkilendirilmiş kısıtlar ve alanlar tanımlanmalıdır.
- Kısıtların ifade edilebileceği programlama dilleri seçilmelidir. (Krzysztof, 2003)

Problemin bu aşaması modelleme aşaması olarak tanımlanmaktadır. Problemi kısıt sağlama problemi olarak ifade etmek için birden fazla yöntem bulunmaktadır. Bu yöntemler:

Alan özellikli metot özel amaçlı algoritmaların uygulamaların formlarını içermektedir. Örnek verecek olursak; lineer denklem sistemlerinin çözümünü içeren programlar, lineer program paketleri gibi.

Genel metot ise arama alanını özel arama metotları ile daraltılması yollarını içermektedir. Bu arama algoritmaları kısıt yayma algoritmaları olarak adlandırılmaktadır. Bu algoritmalar eşitlikleri korurken ele alınan problemi basitleştirmektedir. Kısıtların genel tutarlılığını dikkate alarak yerel tutarlı çözüm

kümeleri oluşturulmaktadır. Arama metotlarında geleneksel geri arama ve dal sınır algoritmalarının değişik formları birleştirilmektedir.

Kısıt programlamanın amaçlarından biri etkin alan özellikli metodu (genel metot yerine) bulmaktır ve genel yapı ile ilişkilendirmektir.

Öncelikle çözeceğimiz problemi kısıt sağlama problemi olarak ifade etmemiz gereklidir. Uygulamada aşağıdakilerle ilgilenilmelidir:

- Seçilen tutarlı temsili bir çözüm olup olmadığı belirlenmelidir.
- Bir çözüm bulmak ve sırayla bütün çözümleri bulmak.
- Bazı kalite kriterlerine göre optimum çözümü bulmak ve sırayla bütün optimal çözümleri bulmak.

Bunlara ek olarak kısıt programlamanın aşağıdaki temel karakteristikleri formüle edilmelidir.

- İki aşamalı yaklaşım: Programlama prosesi iki aşamadan oluşmaktadır. Problemi temsil eden kısıtların ifade edilmesi ve bunların çözümünün bulunmasıdır. Uygulamada bu iki aşama küçük adımlardan oluşur.
- Esneklik: Problemi ifade etmekte kullanılan kısıtlar çok esnektir çünkü kısıtlar eklenebilir, silinebilir veya değiştirilebilirler.
- Çeşitli yapıların var olması: Programlamayı desteklemek için çeşitli yapısal metotlar kullanmak mümkündür. Bunlar özel kısıt çözümleri, kısıt yumuşatma algoritmaları ve arama algoritmaları ile ilgilidir. (Krzysztof, 2003)

Kısıt programlamanın bir başka yönü, problemin kısıtları arasındaki ilişkilerinin ifade edilebilmesidir. İlişkiler veri tabanı gibi bazı veri tabanı sistemleri ile benzerlikler taşımaktadır. Kısıtlar aynı zamanda veri tabanı sistemi gibi çalışmaktadır. Bazı bilgilerin bulunduğu durumlarda kullanışlıdır. Örneğin, haritanın bir bölgesinin tanımlanmasında, ihtiyaçların dolaylı temsil edilmesi gerektiğinde, gerçek kısıtların ifade edilmesinde.



İki konu arasındaki farklar veri tabanı bağlamında ele alındığında, veri tabanlarında çok esnek bir şekilde çeşitli ilişkilere göre sorgulamalar yapılabilmektedir. Bu ilişkiler bağımsız olarak açıkça tanımlanmış olabilir (örneğin tablolar yoluyla) veya dolaylı olarak (örneğin özyinelemeli veya eşitsizliklerle). Bunun karşılığında, kısıt programlamada kabul ilişkileri dolaylı olarak tanımlanır ve burada görev çözmek ve bir çözümün var olup olmadığını bulmaktır. Bu da farklı yöntemleri ve farklı teknikleri gerektirmektedir.

Kısıt programlama ile gereksinimleri genel özellikleri veya kuralları doğal olarak formüle edilebilen problemlerin çözümü en iyi şekilde elde edilebilir. Kısıt programlama ile çözülebilecek problem gruplarına örnek verecek olursak: yöneylem araştırması problemleri, moleküler biyoloji, elektrik mühendisliği, iş hayatı uygulamaları, sayısal hesaplamalar alanlarındaki problemler.

### **5.1. Kısıt Programlama Temel Kavramı**

Şimdiye kadar kısıt programlama ile ilgili verdiğimiz genel bilgiler problemin modellenmesi ile ilgili idi. Bazı problemlerin modellenmesi kolay ve açıklıkla yapılabilmektedir. Bazılarının ise modellenmesi karışık ve zordur. Bu bölümde kısıt sağlama problemlerinin temel mantığından bahsedeceğiz. Böylelikle modellemenin daha iyi anlaşılması amaçlanmaktadır.

Kısıt sağlama problemlerini daha iyi açıklamak için belirli kriterlere göre sınıflandırılabilir. Kısıtların tanımladıkları alanların özelliklerine ve kısıtları tanımlamada kullanılan ifadelere dayanan iki kategoriye ayrılabilir.

Daha önce kısıt sağlama problemlerinin temeli olan kısıt programlama açıklamıştık. Yaptığımız tanımlamaları biçimsel olarak da ifade edeceğiz. Bu ifadeyi kesin olarak tanımlamak kolaydır ancak öncelikle kısıtlama kavramı tanımlanmalıdır.

$Y$ :  $y_1, y_2, \dots, y_k$  şeklinde bir değişkenler dizisi ve bunlarla bağlantılı olan  $D_1, D_2, \dots, D_k$  şeklinde alanlar tanımlanmış olsun ( $k > 0$ ). Böylelikle her bir  $y$  değişkeni her bir  $d$

alanı ile ifade edilebilir.  $Y$  ile ifade edilen  $C$  kısıdı  $D_1 \times D_2 \times \dots \times D_k$  ifadesinin alt seti olarak tanımlanır.  $k=1$  olduğunda kısıt birli,  $k=2$  olduğunda kısıt ikilidir. Kısıt sağlama problemleri,  $X: x_1, \dots, x_n$  değişkenlerine ait sonlu dizilere karşılık gelen  $D_1, \dots, D_n$  alanları,  $C$  kısıtlarının sonlu setinden oluşur ve  $X$  değerlerinin her bir alt dizisi olarak tanımlanır. Böyle bir ifadeyi tanımlamak için  $\langle C : D \rangle$  şeklinde yazılır, bu ifadede  $D \in: x_1 \in D_1, \dots, x_n \in D_n$ , her bir  $x \in D$  formundaki yapı alan ifadeleri olarak tanımlanır. Özel  $C$  kısıt setlerinin gösterimini de basitleştirmek amacıyla “ $\{\}$ ” şeklinde küme parantezleri kullanılmaktadır.

Kısıt sağlama problemlerinin önemli notasyonlarını tanımlayacak olursak. Bir kısıt sağlama probleminin çözümü tüm kısıtları sağlayan tüm değişkenler için geçerli olan bir dizidir. Notasyonlar ile belirtilirse, kısıt sağlama problemi  $\langle C : D \rangle$ ,  $D \in: x_1 \in D_1, \dots, x_n \in D_n$  şeklinde verildiği varsayılmıştır.  $d_1, \dots, d_n \in D_1 \times \dots \times D_n$  şeklinde  $n$ -demet olsun. Bu demetler  $x_1, \dots, x_n$  ile tanımlanmış  $C$  kısıtlarını sağlıyorsa  $d_1, \dots, d_n$  problemin bir çözümüdür denilir. Bütün çözüm demetleri bulunduğunda kısıt sağlama problemi çözülmüş olacaktır. Eğer kısıt sağlama probleminin çözümü varsa tutarlı, eğer çözüm mevcut değilse tutarsız olarak adlandırılır. (Krzysztof, 2003)

Bir problem kısıt programlama ve kısıt sağlama ile çözümü için bir yazılım dili ile modellenmelidir. Bu modellemede bütün kısıtlar ve alan ifadeleri tanımlanmalıdır. Bu tanımlamada her bir kısıt ilişkili değişkenlerin Kartezyen çarpımlarının alt setleri olarak gösterilmektedir.

Kısıt programlama ile optimum çözümü bulmayı amaçlanan problemlerde bir amaç fonksiyonuna göre her bir çözüm değerlendirilir ve amaca göre maksimum ve minimum değerler bulunur. Bu problem tarzı kısıt optimizasyonu problemi olarak tanımlanır. Kısıt optimizasyon problemleri, kısıt sağlama problemlerinin amaç fonksiyonu eklenerek geliştirilmiş bir problem çözme yöntemidir.

## 5.2. Arama Stratejileri

Kısıt programlama metodunda kullanılan arama stratejilerine bu bölümde yer verilecektir. Üç tip arama stratejisi programlamada kullanılmaktadır. Bu metotlar:

- Öncelikle Derinlik arama (Depth-first search)
- Çok noktalı arama (Multi-point search)
- Yeniden başlatmalı arama (Restart search)

### 5.2.1. Öncelikle Derinlik Arama

Bu arama algoritmasında üst düğümden başlanır ve en alttaki düğüme kadar dallanma yapılır. Son düğüm için çözüm bulunur. Daha sonra bir önceki düğüme dönülür ve buradan yeniden dallanma yapılarak son düğüm için yeniden hesaplama yapılır. Tekrar geri dönüşlerde bir üst aşama için bütün dallar araştırıldığında, bu aşamadan önce gelen aşama için işlemler tekrarlanır. Böylelikle ilk düğüme ulaşılan kadar aramaya devam edilir. İlk düğüme ulaşıldığında dallanma diğer bir düğüme geçilerek tekrarlanır. Böylelikle bütün çözümler araştırılmaktadır. (ILOG CP Optimizer User's Manual, 2009)

Bu algoritma hata ayıklamada ve arama ayarlarını yaparken çok yararlıdır. Ancak, zayıf dallanmalardan kolay kurtulamaması nedeniyle yeniden başlatma ve çok noktalı arama algoritmalarına göre daha az etkili bir arama algoritmasıdır.

### 5.2.2. Çok Noktalı Arama

Çok noktalı arama algoritması, arama noktalarından oluşan bir noktalar havuzuna dayanmaktadır. Arama noktaları ise uygun ve kısmi çözümlere sahip karar değişkenlerinin atamalarının derlemesidir. Başlangıç havuzu arama noktaları yapısal arama metodu kullanılarak bulunur. Yeni oluşturulacak olan arama havuzları başlangıç havuzunda bulunan arama noktaları kullanılarak oluşturulur. Optimizasyon

problemlerinde ise algoritma öğrenme etkisi kısmi çözümleri uygun çözümlere ulaştıracak şekilde çalışmaktadır. Algoritmada çözüm havuzunda bulunan çözümlerin bir kombinasyonu ile oluşturulan yeni havuzlarda daha iyi çözümlere ulaşmak amaçlanmaktadır. (ILOG CP Optimizer User's Manual, 2009)

Algoritma optimalliği veya çözümün bulunmasını garanti etmemektedir. Buna karşılık, algoritma derinlemesine ilk ve yeniden başlatmalı aramalara göre çözümü daha çok çeşitlendirmektedir. Arama bulunan en iyi sonuç geliştirilemediği adımda sona ermektedir. Bundan dolayı arama için bir limit oluşturulması tavsiye edilir.

### **5.2.3. Yeniden Başlatmalı Arama**

Yeniden başlatma algoritması ILOG CP Optimizer tarafında varsayılan arama algoritması olarak kullanılmaktadır. Bu algoritmada arama uzayının ulaşılmamış bölgelerini de aramaya dahil etmeyi amaçlayan bir yapısal arama kullanılmaktadır. Yapısal aramanın arama strajesi, bir arama ağacının çaprazlaması sonucunda elde edilecek yeni çözümleri bulmak şeklindedir. Çözüm için aramanın başlangıç noktası arama ağacının köküdür. Ağacın kökünden itibaren azalarak devam eden dallanma aramada alternatif çözümleri temsil etmektedir. Arama uzayındaki değerlerin her bir kombinasyonu arama ağacının yapraklarını temsil etmektedir. Yani çözüm noktalarını içeren değerler arama ağacının yapraklarıdır ve bu noktalar uygun çözüm alanı kısıtlarını sağlamalıdır.

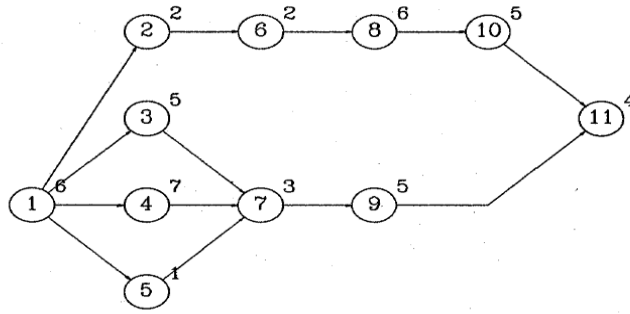
Yeniden başlatmalı algoritma yapısal algoritma tabanlı çalışmaktadır. Derinlemesine araması ise belirli bir hatalı dallanmaya ulaşıldıktan sonra yeniden başlamaktadır. IBM ILOG CP Optimizer yeniden başlatma için *Restart Growth Factor* parametresini kullanmaktadır. Yeniden başladıktan sonra hata değeri  $f$  olursa, bir sonraki koşulda, yeni hata limiti  $f$  adım daha bu parametrenin değeri olacaktır. Başlangıç hata değeri *RestartFailLimit* parametresi ile kontrol edilir. (ILOG CP Optimizer User's Manual, 2009)

## 6. KUYRUK AĞLARININ MONTAJ HATTILARINA UYUMLULUĞU

Stokastik montaj hatlarında süreler genel olarak normal dağılıma uyduğu kabulü yapılmış ve çalışmalar bu konuda yoğunlaşmıştır. Diğer dağılımları kullanan çalışmalarda literatürde mevcuttur. Stokastik montaj hatları ile ilgili literatür çalışmamızda bu çalışmalara ayrıntılı olarak değinilmiştir.

Çalışmamızda istasyon sürelerinin üssel dağılıma uyduğu sistem kabulleri arasında verilmişti. Bu kabulü doğrulamak amacıyla simülasyon modeli kurulmuştur. Simülasyon çalışmasında sürelerin normal dağılıma uyduğu durumlar ile üssel dağılıma uyduğu durum karşılaştırılmıştır.

Karşılaştırmaları yapmak amacıyla literatürde yer alan test problemlerinden Jackson montaj hattı (Şekil 6.1) ele alınmıştır. Montaj hattı için 5 istasyon kurulmuş ve çıktı hızı ve istasyon kullanım oranları alternatif atama kombinasyonları belirlenerek karşılaştırılmıştır.



**Şekil 6.1** Jackson Montaj Hattı Şebeke Serimi

Ele aldığımız montaj hattındaki işler için öncelik kısıdı altında 40 adet olurlu atama belirlenmiştir (Çizelge 6.1). Bu atama alternatiflerinden en iyi çıktı hızına sahip alternatifini belirlemek için tek sınıflı kapalı kuyruk ağları için kullanılan ODA algoritması ile hesaplamalar yapılmıştır. Çıktı hızı ve istasyon kullanım oranları hesaplanmıştır. Bu algorithmada istasyon süreleri üssel kabul edilmektedir.

**Çizelge 6.1** Jackson Probleminden Testler için Oluşturulan Alternatif İş/İstasyon Atamaları ve İstasyon Süreleri

		İstasyon 1	İstasyon 2	İstasyon 3	İstasyon 4	İstasyon 5
Alternatif 1	Görev Atamaları	1	2-3-4	5-6-7	8-9	10-11
	İstasyon süresi	6	14	6	11	9
Alternatif 2	Görev Atamaları	1	3-4-5	2-6-7	8-9	10-11
	İstasyon süresi	6	13	7	11	9
Alternatif 3	Görev Atamaları	1-3	2-4	5-7-9	6-8	10-11
	İstasyon süresi	11	9	9	8	9
Alternatif 4	Görev Atamaları	1-2-6	5-8	3-10	4-7	9-11
	İstasyon süresi	10	7	10	10	9
Alternatif 5	Görev Atamaları	1-2	3-4	5-6-7	8-9	10-11
	İstasyon süresi	8	12	6	11	9
Alternatif 6	Görev Atamaları	1-2	3-5	4-6-7	8-9	10-11
	İstasyon süresi	8	6	12	11	9
Alternatif 7	Görev Atamaları	1-2	5-4	3-6-7	8-9	10-11
	İstasyon süresi	8	8	10	11	9
Alternatif 8	Görev Atamaları	1-2	6-4	3-5-7	8-9	10-11
	İstasyon süresi	8	9	9	11	9
Alternatif 9	Görev Atamaları	1-3	2-4-5	6-8	7-9	10-11
	İstasyon süresi	11	10	8	8	9
Alternatif 10	Görev Atamaları	1-2	5-4	3-6-7	8-10	9-11
	İstasyon süresi	8	8	10	11	9
Alternatif 11	Görev Atamaları	1-2	5-4	3-6-7	8-10	9-11
	İstasyon süresi	8	8	10	11	9
Alternatif 12	Görev Atamaları	1-3	2-4-5	6-7	8-9	10-11
	İstasyon süresi	11	10	5	11	9
Alternatif 13	Görev Atamaları	1-3	7-4-5	6-2	8-9	10-11
	İstasyon süresi	11	11	4	11	9
Alternatif 14	Görev Atamaları	1-3	2-4-6	5-7	8-9	10-11
	İstasyon süresi	11	11	4	11	9
Alternatif 15	Görev Atamaları	1-3	7-6-5	4-2	8-9	10-11
	İstasyon süresi	11	6	9	11	9
Alternatif 16	Görev Atamaları	1-3	2-4-5	6-7	8-10	9-11
	İstasyon süresi	11	10	5	11	9
Alternatif 17	Görev Atamaları	1-3	2-4-5	6-7	8-10	9-11
	İstasyon süresi	11	10	5	11	9
Alternatif 18	Görev Atamaları	1-4	2-3-5	6-7	8-9	10-11
	İstasyon süresi	13	8	5	11	9

**Çizelge 6.1 (devam) Jackson Probleminden Testler için Oluşturulan Alternatif İş/İstasyon Atamaları ve İstasyon Süreleri**

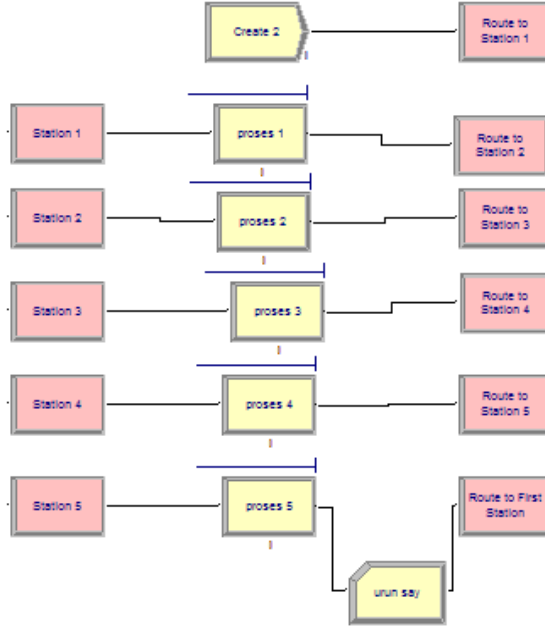
		İstasyon 1	İstasyon 2	İstasyon 3	İstasyon 4	İstasyon 5
Alternatif 19	Görev Atamaları	1-4	7-3-5	6-2	8-9	10-11
	İstasyon süresi	13	9	4	11	9
Alternatif 20	Görev Atamaları	1-4	2-3-6	5-7	8-9	10-11
	İstasyon süresi	13	9	4	11	9
Alternatif 21	Görev Atamaları	1-4	7-6-5	3-2	8-9	10-11
	İstasyon süresi	13	6	7	11	9
Alternatif 22	Görev Atamaları	1-4	2-3-5	6-7	8-10	9-11
	İstasyon süresi	13	8	5	11	9
Alternatif 23	Görev Atamaları	1-4	3-5	6-2-7	8-9	10-11
	İstasyon süresi	13	6	7	11	9
Alternatif 24	Görev Atamaları	1-4	2-3-5	6-7-8	10	9-11
	İstasyon süresi	13	8	11	5	9
Alternatif 25	Görev Atamaları	1-4	7-3-5	6-2-9	8	10-11
	İstasyon süresi	13	9	9	6	9
Alternatif 26	Görev Atamaları	1-4	7-3-5	6-2-9	8-10	11
	İstasyon süresi	13	9	9	11	4
Alternatif 27	Görev Atamaları	1-3	7-4-5	6-2-9	8	10-11
	İstasyon süresi	11	11	9	6	9
Alternatif 28	Görev Atamaları	1-5	7-4-3	6-2-9	8	10-11
	İstasyon süresi	7	15	9	6	9
Alternatif 29	Görev Atamaları	1-5	2-3-6	7-4-9	8	10-11
	İstasyon süresi	7	9	15	6	9
Alternatif 30	Görev Atamaları	1-5	2-4-6	7-3-9	8	10-11
	İstasyon süresi	7	11	13	6	9
Alternatif 31	Görev Atamaları	1-4	7-5-3	6-2-9	8	10-11
	İstasyon süresi	13	9	9	6	9
Alternatif 32	Görev Atamaları	1-4	2-3-6	7-5-9	8	10-11
	İstasyon süresi	13	9	9	6	9
Alternatif 33	Görev Atamaları	1-4	2-5-6	7-3-9	8	10-11
	İstasyon süresi	13	5	13	6	9
Alternatif 34	Görev Atamaları	1-3	7-5-4	6-2-9	8	10-11
	İstasyon süresi	11	11	9	6	9
Alternatif 35	Görev Atamaları	1-3	2-4-6	7-5-9	8	10-11
	İstasyon süresi	11	11	9	6	9

**Çizelge 6.1 (devam)** Jackson Probleminden Testler için Oluşturulan Alternatif İş/İstasyon Atamaları ve İstasyon Süreleri

		İstasyon 1	İstasyon 2	İstasyon 3	İstasyon 4	İstasyon 5
Alternatif 36	Görev Atamaları	1-3	2-5-6	7-4-9	8	10-11
	İstasyon süresi	11	5	15	6	9
Alternatif 37	Görev Atamaları	1-2	3-5-6	7-4-9	8	10-11
	İstasyon süresi	8	8	15	6	9
Alternatif 38	Görev Atamaları	1-2	4-5-6	7-3-9	8	10-11
	İstasyon süresi	8	10	13	6	9
Alternatif 39	Görev Atamaları	1-2	3-4-6	7-5-9	8	10-11
	İstasyon süresi	8	14	9	6	9
Alternatif 40	Görev Atamaları	1-2	3-4	7-5-9	6-8	10-11
	İstasyon süresi	8	12	9	8	9

Kuyruk ağı modelinden elde edilen sistem performans değerlerini karşılaştırmak amacıyla sistem Arena 11 programı ile simüle edilmiştir. Bir sistemin performansının ölçülmesinde çok kullanılan araçlardan biri simülasyondur. Simülasyon çalışmaları sistemin çalışma koşullarının bire bir kopyasının bilgisayar ortamında modellenmesi ile yapılmaktadır ve sistemin çalışma performansına yakın performans çıktılarını vermektedir. Simülasyon yardımı ile kaynakların İstasyon Doluluk Oranı, sistemden sağlanan çıktı değerleri, istasyonlar arasında bekleyen ürün sayısı ve ürünlerin bekleme süresi gibi sistemi açıklayıcı veriler elde edilmektedir.



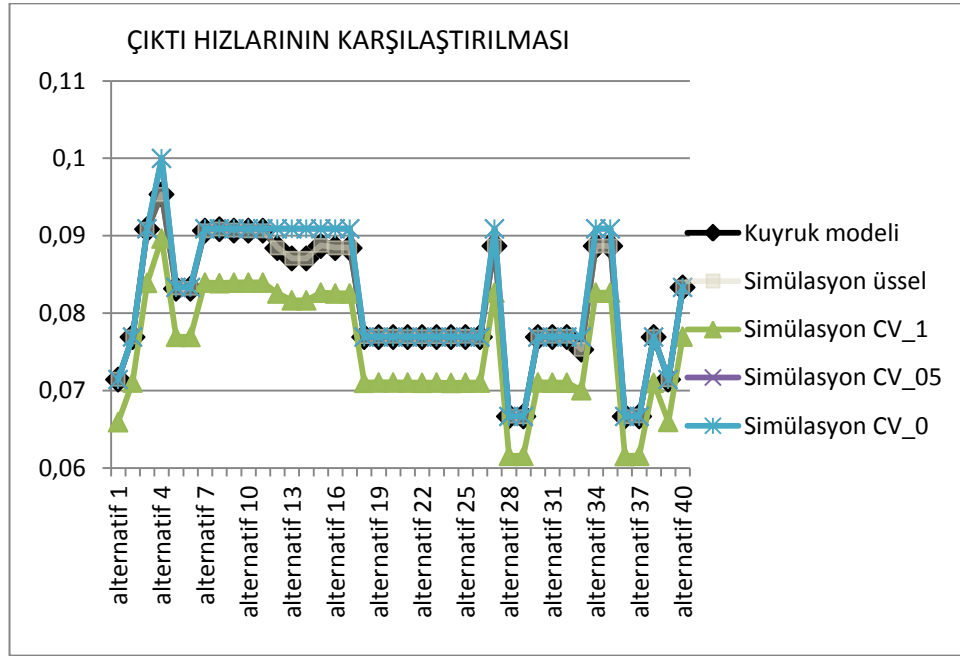


**Şekil 6.2** Arena 11 programında oluşturulan kapalı kuyruk ağı sistemi modeli

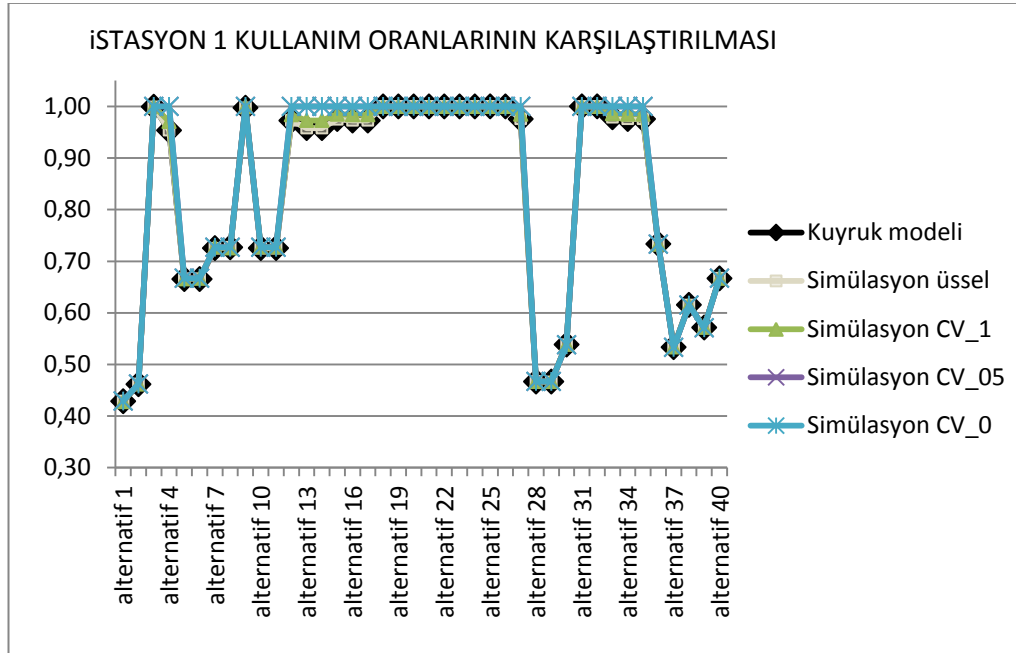
Montaj hattının stokastik işlem süresine sahip olduğu kabulünü sağlamak amacıyla 3 farklı senaryo oluşturulmuş ve simüle edilmiştir. Bu senaryolar; istasyon sürelerinin üssel olması, istasyon sürelerinin değişim katsayısı (coefficient of variance(CV))=1 olan normal dağılıma uyduğu ve istasyon sürelerinin CV=0,5 olduğu normal dağılıma uyduğudur. Ayrıca sürelerin deterministik olması durumu da simüle edilmiştir.

Çalışmamızda belirlediğimiz her bir senaryoda her bir alternatif için simülasyon modeli 10000 dk ısınma süresi olmak üzere 100000 dk 10 tekrarlı olarak çalıştırılmıştır. Sistemden çıkan ürün sayısı ve istasyonların İstasyon Doluluk Oranı değerleri çıktı olarak alınmıştır.

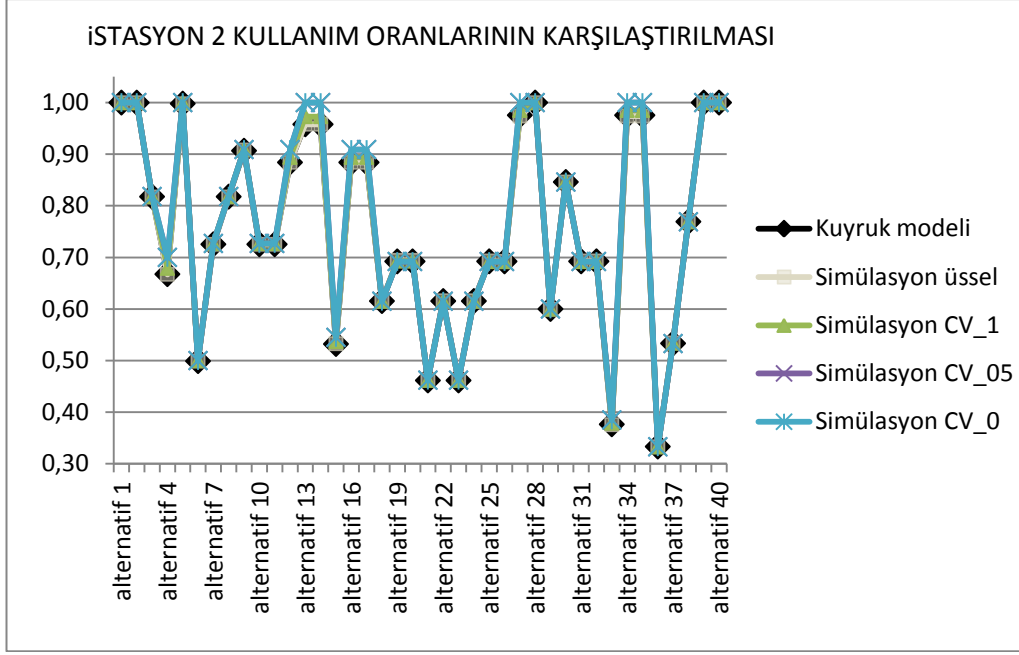
Yapılan deneysel çalışmalar sonucunda kuyruk ağı modeli ile simülasyon modelinden elde edilen sonuçlar karşılaştırılmıştır. Çıktı hızı ve istasyon kullanım oranları için grafiksel karşılaştırmalar yapılmıştır. İlgili grafikler Şekil 6.3, 6.4, 6.5, 6.6, 6.7, 6.8 ve 6.9’da verilmiştir.



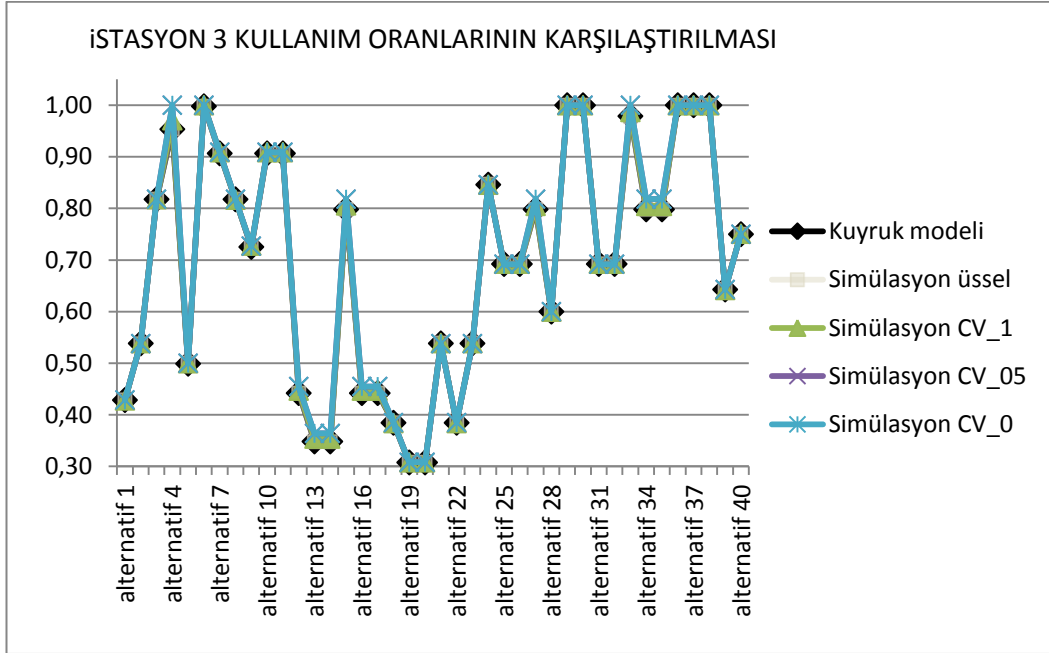
Şekil 6.3 Sistem çıktı hızlarının karşılaştırılması



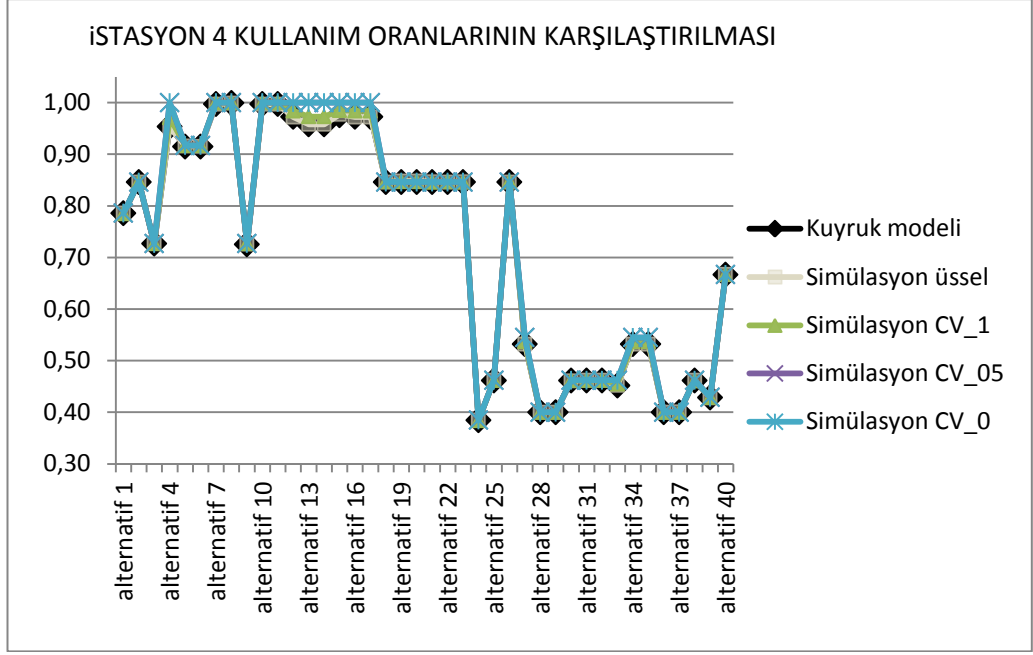
Şekil 6.4 İstasyon 1 için kullanım oranları karşılaştırılması



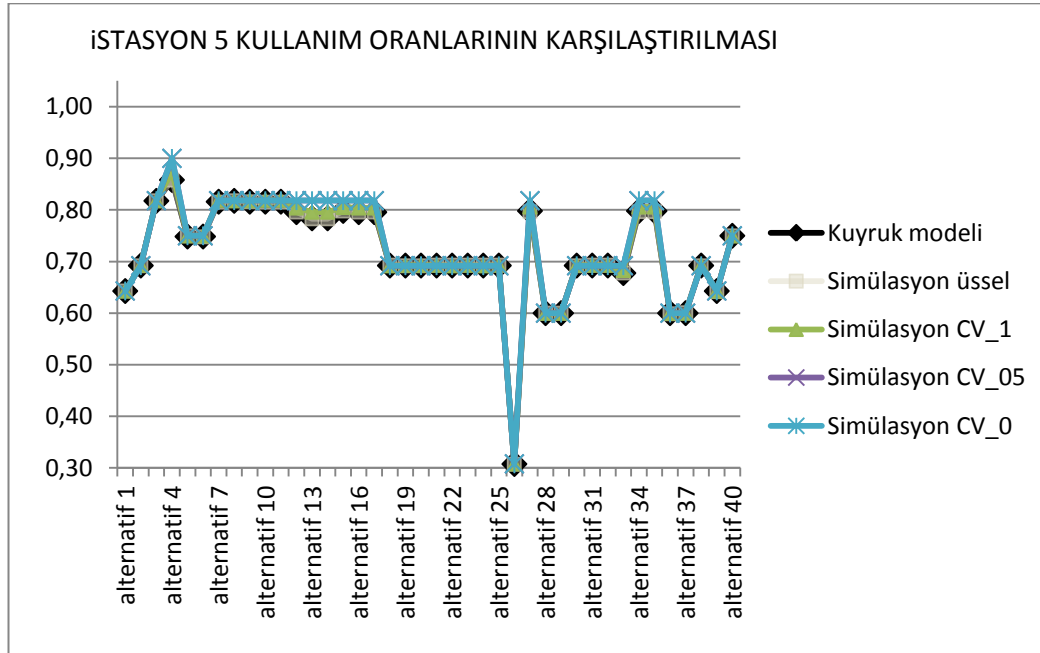
**Şekil 6.5** İstasyon 2 için kullanım oranları karşılaştırılması



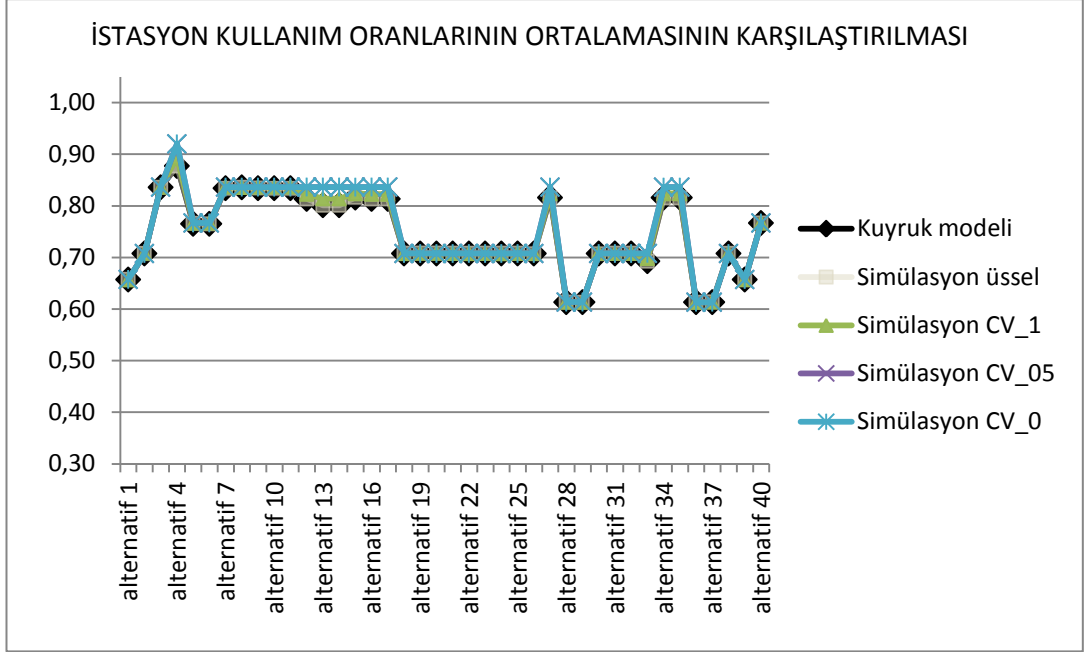
**Şekil 6.6** İstasyon 3 için kullanım oranları karşılaştırılması



Şekil 6.7 İstasyon 4 için kullanım oranları karşılaştırılması



Şekil 6.8 İstasyon 5 için kullanım oranları karşılaştırılması



**Şekil 6.9** Ortalama istasyon kullanım oranlarının karşılaştırılması

Yaptığımız çalışma sonucunda simülasyon modelinden elde edilen verilerle, kuyruk ağı modelinden elde ettiğimiz sonuçlar karşılaştırıldığında şu sonuçlar elde edilmiştir.

- Montaj hatlarının alternatif atamalarında simülasyon senaryolarından elde edilen sonuçlar ile kapalı kuyruk ağı modeli ile elde edilen sonuçlar birbirine benzer bir eğilim içerisindedir. Bazı değerlerde az bir fark olsa da sonuç olarak grafiklerde de görüldüğü üzere verilerin eğilimleri birbirine çok benzerdir.
- Çıktı hızları için kuyruk ağı ve simülasyon modellerine ilişkin grafik göz önüne alındığında simülasyon senaryoları için en iyi alternatifin aynı zamanda kapalı kuyruk ağı modeli için de en iyi alternatif olduğu görülmektedir.
- Diğer bir performans değeri olan ortalama istasyon kullanım oranları için yapılan karşılaştırmada da simülasyon sonuçları ve kapalı kuyruk ağı modelinden elde edilen sonuçların, çıktı hızına göre en iyi alternatifin, en iyi İstasyon Doluluk Oranına sahip olduğunu göstermiştir.

Bu sonuçları deęerlendirdiđimizde kapalı kuyruk ađı modelinin montaj hatlarının dengelenmesinde kullanılabileceđini gstermiřtir. Algoritmanın ok hızlı bir řekilde deęerleme yapması avantajı olarak gsterilebilir.

## 7. TİP 2 STOKASTİK DÜZ MONTAJ HATTI İÇİN ÖNERİLEN ÇÖZÜM METODOLOJİSİ

Tez çalışmasının bu bölümünde öncelikle ele aldığımız problem tanımlanmış ve ardından çözüm için önerilen algoritma adım adım açıklanmıştır.

### 7.1. Problemin Tanımı

Ele aldığımız problemde, stokastik işlem süreli, gecikmeli, düz ve tek modelli montaj hatları için verilen sabit istasyon sayısı kısıdı altında çevrim süresinin minimize edilmesi amaçlanmıştır.

Montaj hattı sıralı iş istasyonlarından oluşması dolayısıyla, bir istasyonun performansı kendisini takip eden diğer bir istasyonun performansını etkilemektedir. Bu sebeple, istasyonlardan birinde meydana gelen performans düşüklüğü hattın performansını da etkilemektedir. Hattın performansını etkileyen diğer bir etken ise sürelerin stokastik olmasıdır. Stokastik işlem süreli sistemlerin performansının ölçülmesi, işlem sürelerinin sabit olduğu sistemlere göre daha zordur. Ayrıca, montaj hatlarında işlem sürelerinin stokastikliğinden dolayı istasyon sürelerinde meydana gelen farklılıklar ürünlerin istasyonlar öncesinde beklemelerine veya ilgili istasyondan sonra gelen istasyonun boş beklemesine sebep olmaktadır. Bu tür değişkenliklerin olduğu sistemlerin performansını ölçmek için sıralı istasyonların birbiri üzerindeki etkilerini ve stokastiklik durumları dikkate alan bir yöntem olan kuyruk ağı modelleri kullanılmaktadır.

Bu çalışmada ele aldığımız montaj hattının çevrim süresini, istasyon doluluk oranlarını, istasyonlar arasında bekleyen ürün sayıları ve ürünlerin istasyonda geçirdiği süreler gibi performans değerleri kuyruk ağı modeli kullanılarak hesaplanmıştır.

Şekil 7.1’de montaj hattında ürünlerin hat boyunca taşınırken bulunabilecekleri alanlar gösterilmiştir. Montaj hattında ürünlerin bulunabileceği alanların kısıtlı sayıda olmasından dolayı, hat üzerinde belirli sayıda ürünün bulunmasına izin verildiği ve ürün sayısının sabit olduğu sonucuna ulaşılmaktadır. Montaj hattının bu yapısı değerlendirildiğinde, montaj hattının ürün sayısının sabit kabul edildiği kapalı kuyruk ağı modeli ile modellenebileceği sonucuna ulaşılmaktadır.



**Şekil 7.1** Montaj hattında ürünlerin taşınması için oluşturulmuş alanlar

Ele aldığımız problem için, kuyruk ağı modeli ve montaj hattı için yaptığımız kabuller ve varsayımlar aşağıda verilmiştir.

#### **7.1.1. Kuyruk Ağı Modeli Varsayımlar ve Kabuller**

Kuyruk ağı modelleri için yaptığımız varsayımlar ve kabuller şu şekildedir:

- Ele aldığımız montaj hattı düz montaj hattı olması dolayısıyla sıralı (tandem) kuyruk ağına uymaktadır.



- Hatta tek tip ürün üretildiği varsayılmıştır ve buna bağlı olarak ürün çeşitliliğine göre tek sınıflı kuyruk ağı modeli kullanılmıştır.
- Montaj hatlarında alan kısıdından dolayı hatta belirli sayıda ürün bulunabilmektedir. Bu sebeple ürün sayısının sabit olduğu kapalı kuyruk ağı modeli kullanılmıştır.
- İşlem süreleri stokastiktir ve üssel dağılıma uymaktadır.
- Sistemin blokesiz olduğu varsayılmıştır.
- İstasyon arası bekleme yerlerinin sınırsız olduğu varsayılmıştır.
- Sistemde bulunan ürün sayısının sabit ve belirli bir sayıda olduğu kabul edilmiştir.

### 7.1.2. Montaj Hattı İçin Varsayım ve Kabuller

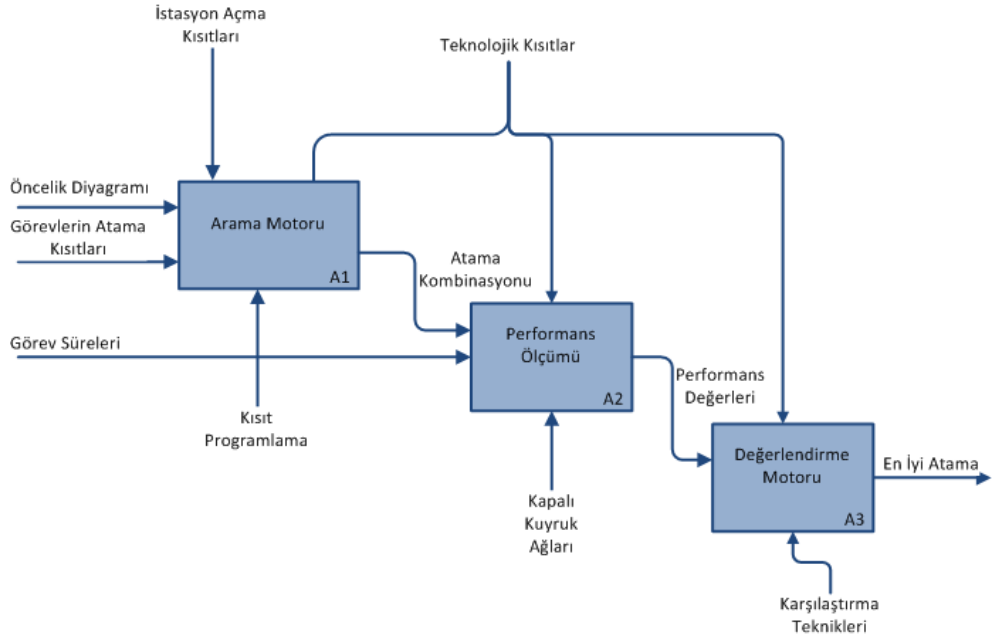
Montaj hattı için yapılan varsayımlar ve kabuller şu şekildedir:

- Her görev mutlaka bir istasyona atanmalıdır.
- Düz montaj hattı olduğu varsayılmıştır.
- İşlem sürelerinin stokastik olduğu ve üssel dağılıma uyduğu kabul edilmiştir.
- İstasyon sayısının sabit olduğu kabul edilmiştir. Çevrim süresinin minimizasyonu amaçlanmıştır.(Tip-2)
- Montaj hattında tek tip ürün üretildiği kabul edilmiştir.
- Görevlerin öncelik kısıtları bellidir ve görev istasyon atamalarında bozulmamalıdır.
- Hazırlık zamanları görev zamanlarına dahil edilmiştir.
- İşlem zamanları görev sıralarından bağımsızdır.
- Her istasyonda en az bir görev bulunmalıdır.
- Kuyruk ağlarında çevrim zamanı olarak ürünün sisteme girişinden, sistemi terk edene kadar geçen zaman alınır. Montaj hatlarından ise çevrim süresi ürünün bir istasyonda kalabileceği en büyük süre olarak tanımlanmaktadır. Çalışmamızda çevrim süresi olarak montaj hatları için kullanılan tanımlama kabul edilmiştir.

## 7.2. Önerilen Çözüm Metodolojisi

Montaj hatlarında görevlerin istasyonlara atanması için çeşitli algoritmalar kullanılmaktadır. Genel olarak bu algoritmalarda belirli çevrim sürelerinde istasyonların boş zamanını minimize edecek şekilde atamalar yapılmaktadır. Ancak, çalışmamızda ele aldığımız problem tarzında sabit istasyon sayısı kısıdı altında çevrim süresi minimize etmek amaçlanmıştır. İstasyon sayısının sabit olmasından dolayı yapılacak her bir atama için çevrim süresi değişmektedir. Bu bağlamda ele aldığımız problem Tip 2 montaj hattı dengeleme problemi için çevrim süresinin değişkenliğinden dolayı literatürde çevrim süresine bağlı olarak atama yapan yöntemler kullanılamamaktadır. Görevlerin istasyonlara atanmasında kısıt programlama yöntemi kullanılarak yeni bir atama algoritması oluşturulmuştur. Kısıt programlama yöntemi ile atanan işlerin performans değerleri kapalı kuyruk ağı ile ölçülmüştür ve en iyi atama kombinasyonu belirlenmiştir.

Bu bölümde öncelikle tip 2 düz montaj hattı için doğrusal programlama modeli verilmiştir ve doğrusal modele bağlı kalarak istasyonlara görev atamaları için oluşturulan kısıt programlama modeli açıklanmıştır. Son olarak, kapalı kuyruk ağı modeli için kullanılan ODA algoritması modele dâhil edilmiştir. Şekil 7.2 'de algoritmanın Integration Definition for Function Modeling(IDEF) diyagramı verilmiştir.



Şekil 7.2 Çözüm metodolojisi IDEF diyagramı

### 7.2.1. Montaj Hattı Dengeleme Probleminin Doğrusal Programlama Modeli

Öncelikle tip 2 montaj hattının matematiksel modeli verilecektir. Bu model üzerinden kısıt programlama algoritmasında nasıl modellendiği açıklanacaktır.

$$\text{Min } z = ct \quad (7.1)$$

Şu kısıtlara göre:

$$\sum_{k=1}^{|W|} x_{ik} = 1 \quad \forall t_i \in T \quad (7.2)$$

$$\sum_{k=1}^{|W|} k * x_{ik} \leq \sum_{k=1}^{|W|} k * x_{jk} \quad \forall \{t_i \rightarrow t_j | t_i, t_j \in T\} \quad (7.3)$$

$$ct \geq \sum_{i=1}^{|T|} \tau_i * x_{ik} \quad \forall w_k \in W \quad (7.4)$$

$$x_{ik} \in \{0,1\} \quad \forall (t_i, w_j) \in (T \times W) \quad (7.5)$$

**Çizelge 7.1** Notasyonlar ve açıklamaları

T	Görev seti
W	İş istasyonları seti
$t_i$	T seti içindeki i. Görev
$w_i$	W seti içindeki i. İstasyon
$\tau_i$	$t_i$ 'nin proses zamanı
$\omega_i$	$W_j$ istasyonundaki iş yükü
$\bar{\omega}$	İstasyonlardaki iş yüklerinin ortalaması
Ct	Çevrim zamanı
$t_i \rightarrow t_j$	Öncelik kısıtı, $t_i$ görevi mutlaka $t_j$ görevin önce yapılmalıdır.
$x_{ik}$	=1 eğer $t_i$ görevi $w_k$ istasyonuna atanmış ise =0 diğer durumlarda

Burada denklem 7.1'de amacımız olan çevrim süresinin minimizasyonudur, Denklem 7.2 her görevin mutlaka bir istasyona atanmasını garanti etmektedir, 7.3 görevlerin istasyonlara atanmasında öncelik ilişkilerinin bozulmaması sağlayan kısıttır, 7.4 ise çevrim süresinin en büyük istasyon süresine eşit olmasını sağlamaktadır. 7.5 ise değişkenlerin ikili değerler almasını sağlamaktadır. Bu modelde süreler stokastik alındığında stokastik optimizasyon problemine dönüşmektedir.

### 7.2.2. Kısıt Programlama Algoritması

Kısıt programlama modeli için ILOG 6.3 CP Optimizer ile programladığımız kısıtların montaj hattı doğrusal programlama formülasyonunda hangi kısıtlara denk geldiği bu bölümde açıklanacaktır.

Görev atamaları için oluşturduğumuz kısıt programlama algoritmasında değişken olarak görevlerin aldığı istasyon numaraları tanımlanmıştır. Algoritma da bu değişken  $istno[i]$  olarak adlandırılmıştır. Bu değişken için eğer  $t_i$  görevi  $w_j$  istasyonuna atanmış ise, değişken j değerini almaktadır. Buna göre:

7.2 kısıdı her görevin mutlaka bir istasyona atanması için oluşturulmuştur. Kısıt programlama algoritmasında bu kısıdı sağlamak için tanımladığımız değişkenin bir istasyon numarası değeri alması ile sağlanmıştır.

Görevlerin atanabileceği ilk ve son istasyonları (algoritmamızda değişkenlerin alabileceği değerlere karşılık gelmektedir) bulmak için Klein ve Scholl (1996)'un kullandıkları 7.6 ve 7.7 numaralı denklemlerde verilen formülasyon kullanılmıştır.

$$E_j(ct) \leq \left\lceil \frac{(t_j + \sum_{h \in P_j^*} t_h)}{ct} \right\rceil \quad (7.6)$$

$$L_j(ct) \leq m + 1 - \left\lfloor \frac{(t_j + \sum_{h \in F_j^*} t_h)}{ct} \right\rfloor \quad (7.7)$$

Bu formülasyonda  $ct$  çevrim süresini belirtmektedir. Amacımız çevrim süresini minimize etmek olduğu için çevrim süresi olarak teorik çevrim süresi alınmış ve hesaplamalar buna göre yapılmıştır. Teorik çevrim süresinin hesaplanması denklem 7.8'de verilmiştir.  $P_j^*$ ,  $j$  görevinden önce gelen bütün görevlerin kümesidir.  $F_j^*$ ,  $j$  görevinden sonra gelen görevlerin kümesidir.

$$ct = \frac{\sum_{i=1}^{|T|} \tau_i}{|W|} \quad (7.8)$$

Bu hesaplamalar sonucunda görevlerin atanabileceği alternatif istasyonlar aşağıdaki eşitsizlikler ile verilmiştir.  $istno[i]$  değişkeninin alabileceği değerler,  $E_i$  değerinden büyük veya eşit olmalı ve  $L_i$  değerinden küçük veya eşit olmalıdır.

$$istno[i] \geq E_i \quad (7.9)$$

$$istno[i] \leq L_i \quad (7.10)$$

7.3 kısıdını sağlamak için öncelik ilişkilerine göre eğer  $t_i$  görevi  $t_j$  görevinden önce gelmesi gerekiyor ise  $istno[i]$  değeri  $istno[j]$  değerinden küçük veya eşit olması kısıdı eklenerek sağlanmıştır.

$$istno[i] \leq istno[j] \quad (7.11)$$

Montaj hattında bir istasyonun kurulması için o istasyona bir görevin atanması gerekmektedir. Bu kısıt da, kısıt programlama algoritmasına dâhil edilmelidir. Algoritmaya, her istasyona en az bir görevin atanması için aşağıdaki kısıt modele eklenmiştir. Burada, en az bir tane  $istno[i]$  değişkeninin  $j$  istasyon numarasını alması sağlanmıştır.

```
forall (j in istasyon){
count(all(i in gorev) istno[i], j) >= 1;
}
```

Bu iki kısıt yapısı dikkate alınarak görevlerin istasyona atanması sağlanmış ve alternatif atamalar bulunmuştur. Bu alternatif atamalar kuyruk ağı modeli ile değerlendirilmiş ve en iyi çıktı değerini veren atama kombinasyonu optimum değer olarak elde edilmiştir.

Doğrusal programlama da değişkenlerin ikili değer almasını sağlayan 7.5 kısıdı ve çevrim süresinin maksimum istasyon süresinden küçük olmasını engelleyen kısıtlar kısıt programlama algoritmasında dikkate alınmamıştır. İkili kısıt yapısı değişkenlerin istasyon numarası değerini alması ile sağlanmıştır. Çevrim süresi kısıdı kısıt programlama algoritmasına dâhil edilmemiştir. Bu kısıdın yerine alternatif atamalar kuyruk ağı modeli ile değerlendirilmiştir.

### 7.2.3. Kapalı Kuyruk Ağı Modeli ve ODA Algoritması

Montaj hattının dengelenmesinde 6. bölümde bahsedildiği gibi kapalı kuyruk ağı modeli kullanılmıştır. Bu modelin performansını ölçmek için kullanılan ODA

algoritması kısıt programlama ile bulunan alternatiflerin çıktı hızlarını belirlemek için kullanılmıştır.

Kısıt programlama ile yaptığımız atamalarda değişkenlerin istasyon numarası değerleri alması sağlanmıştı. Ancak, ODA algoritmasında performans hesaplamalarında işlem süreleri (montaj hattı için istasyon süreleri) kullanılmaktadır. Tanımladığımız değişkenler kullanılarak istasyon süreleri hesaplanmıştır. Bu hesaplama ILOG 6.3'te şu şekilde kodlanmıştır.

```
execute istasyonsuresihesaplama {
    for (var i in istasyon){
        istsur[i]=0
    };
    for (i in thisOplModel.istasyon){
        for(var j in thisOplModel.gorev)
            if (thisOplModel.istno[j]==i){
                istsur[i]= istsur[i]+gorsur[j]
            };
    };
};
```

Bu kodlamada j istasyonuna atanan görevlerin sürelerinin toplamı alınarak istasyon süreleri hesaplanmıştır. İstasyon süreleri kuyruk algoritmasında sistem performansını ölçmek için kullanılmıştır.

Modelde sıralı kapalı kuyruk ağının özelliği de dikkate alınarak istasyonlara ürünlerin ziyaret olasılık (visit ratios) değerleri  $VR=1$  olarak atanmıştır. Sistemin çalışmaya başlama anında istasyonlarda ürün olmamasını sağlamak için istasyonda hiç ürün olmaması olasılığı  $pi[i][0][0]=1$  olarak atanmıştır. Ayrıca, ODA algoritmasında kullanılan istasyon çıktı hızları her bir istasyon için  $mu[i]=1/istsur[i]$  ile hesaplanmıştır. Bu hesaplamalar başlangıç değer atamaları bloğu ile kodlanmıştır.

```
execute basdegatamasi {
    for (var i in thisOplModel.istasyon){
        mu[i]= 1/istsur[i];
        pi[i][0][0]=1;
        vr[i]=1;
    };
};
```

Kapalı kuyruk ağı için kullanılan ODA algoritması modelde aşağıdaki kod bloğu ile kodlanmıştır. Böylelikle her bir alternatif için çıktı değerleri hesaplanmıştır.

```
execute kuyrukalgoritmasi {
    var tom=thisOplModel;
    for (var k in mustsay){
        Tg=0;
        for (var i in istasyon){
            for(var j=1; j<=k; j++){
                Tg+=Tg + (j/mu[i])*pi[i][j-1][k-1];
            };
            T[i][k]=Tg;
        };

        for (i in istasyon){
            topsur[k]=T[i][k]*vr[i];
        };
        lambda[k]=k/topsur[k];
        lambdaobj=lambda[k];

        for (i in istasyon){
            toppi=0;
            for(j=1; j<=k; j++){
                pi[i][j][k]=(lambda[k]/mu[i])*pi[i][j-1][k-1]*vr[i];
                toppi=toppi+pi[i][j][k];
            };
            pi[i][0][k]=1-toppi;
        };
    };
};
```

Kapalı kuyruk ağı algoritması sonucunda belirlenen atama için hesaplanan performans değeri (*labmdaobj*) alternatif atamaların değerleri ile karşılaştırmak üzere bir dosyaya kaydedilmektedir.

#### 7.2.4. Kısıt Programlama ile Alternatif Atamaların Bulunması

ILOG 6.3 CP Optimizer ile alternatif atamalar startNewSearch() fonksiyonu ile yeniden arama yaptırılmak suretiyle elde edilmiştir. Bu fonksiyon kısıt programlamada geri arama yapılmasını ve yeni alternatiflerin bulunmasını sağlamaktadır. Aşağıda, modelde yeniden üretim için tanımlanan *main* fonksiyonu verilmiştir.

```
main {
    var status=1;
    thisOplModel.generate();
}
```



```
cp.startNewSearch();  
while (cp.next()) {  
    status ++;  
    thisOplModel.postProcess();  
}
```

Durdurma kriteri sağlanana kadar elde edilen alternatif atamaların çıktı hızları karşılaştırılıp en iyi atama kombinasyonu elde edilmiştir. Böylelikle montaj hattının kurulmasında hangi görevin hangi istasyona atanması gerektiği ve bu atamanın sonucunda sistemin performans değerlerinin neler olduğu belirlenmiştir.

## 8. DENEYSEL ÇALIŞMALAR VE SONUÇLARI

Önerilen yöntemi test etmek amacı ile deney setleri oluşturulmuştur. Uğurdağ vd.(1997) çalışmasında yer alan deney setlerinden 17 adet deney seti seçilmiştir. Çizelge 8.1’de kullanılan deney setleri verilmiştir.

**Çizelge 8.1** Deney Setleri

Set Adı	Görev Sayısı	İstasyon Sayısı
Mertens	7	2
Mertens	7	3
Mertens	7	5
Jaeschke	9	3
Jaeschke	9	4
Jaeschke	9	7
Jackson	11	3
Jackson	11	4
Jackson	11	5
Mitchell	21	3
Mitchell	21	5
Mitchell	21	8
Heskiaoff	28	4
Heskiaoff	28	5
Sawyer	30	5
Sawyer	30	8
Sawyer	30	13

8.1’de verilen deney setleri, öncelikle derinlik ve çok noktalı arama stratejileri kullanılarak kısıt programlama algoritması ile çözülmüştür.

### 8.1. Öncelikle Derinlik Arama Stratejisi Deneyleri

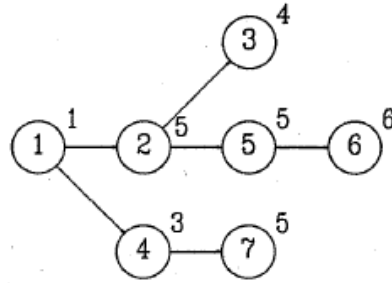
Bölüm 5.2’de kısıt programlamada kullanılan arama stratejilerinden bahsedilmişti. Arama stratejileri arasında öncelikle derinlik arama stratejisi bütün olası çözümlerin

bulunmasını sağlamaktadır ve çözümlerin tekrarlanmasına izin vermemektedir. Bu özelliklerinden dolayı öncelikle derinlemesine arama stratejisi ile problemin bütün olası çözümlerinin değerlendirilmesi amaçlanmıştır. Deney setleri, ILOG 6.3 Cp Optimizer ile modellenerek 7. bölümde adımları açıklanan çözüm metodolojisi ile çözümlenmiştir. Ek-1’de algoritmanın kodu verilmiştir. Ek-2’de ise Jackson deney setinin 3 istasyonlu durumu için elde edilen bütün alternatif atamalar örnek olarak verilmiştir.

Çözüm algoritması kullanılan bilgisayarın özellikleri de dikkate alınarak en fazla 250000 atama alternatifi için çalıştırılmıştır. Elde edilen sonuçlar bu bölümde her bir deney için ayrı ayrı verilmiştir.

### 8.1.1. 7 İş 2 İstasyonlu Mertens (1967) Deneyi

Mertens (1967) montaj hattı için öncelik diyagramı şekil 8.1’de verilmiştir.



**Şekil 8.1** Mertens (1967) 7 işli montaj hattı serimi

İstasyon sayısının 2 olduğu durum için en iyi atama kombinasyonu ve performans değerleri çizelge 8.2 ve 8.3’te verilmiştir.

**Çizelge 8.2** Mertens (1967) 7 işli 2 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri

<b>Mertens (1967) 7 iş 2 istasyon</b>		
	<b>İstasyon 1</b>	<b>İstasyon 2</b>
<b>En İyi Atama Kombinasyonu</b>	1,4,5,6	2,3,7
<b>Ortalama İstasyon Süresi</b>	14	14

**Çizelge 8.3** Mertens (1967) 7 işli 2 istasyonlu montaj hattı için performans değerleri

<b>Mertens (1967) 7 iş 2 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	0,9804	0,98	14,280	14,2644	-0,11%
<b>İstasyon 2</b>	0,9804	0,98			

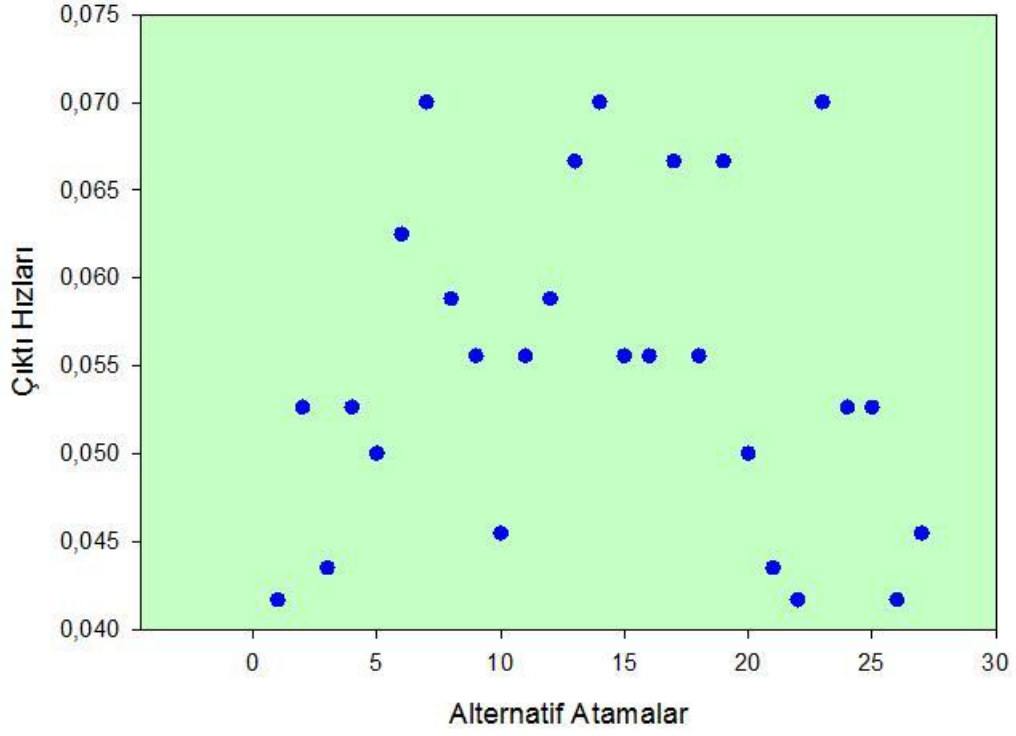
Problemin çözdürülmesi sonucunda alternatif en iyi atama kombinasyonları elde edilmiştir. Bu atama kombinasyonları çizelge 8.4'te verilmiştir.

**Çizelge 8.4** Mertens (1967) 7 işli 2 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları

<b>Mertens (1967) 7 iş 2 istasyon</b>		
	<b>istasyon 1</b>	<b>istasyon 2</b>
<b>Alternatif Atama Kombinasyonları</b>	1,2,3,5	4,6,7
	1,2,4,7	3,5,6

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.2'de verilmiştir.

Mertens\_7\_2



**Şekil 8.2** Mertens (1967) 7 iş 2 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

### 8.1.2. 7 İş 3 İstasyonlu Mertens (1967) Deneyi

Mertens montaj hattı için istasyon sayısının 3 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.5 ve 8.6'da verilmiştir.

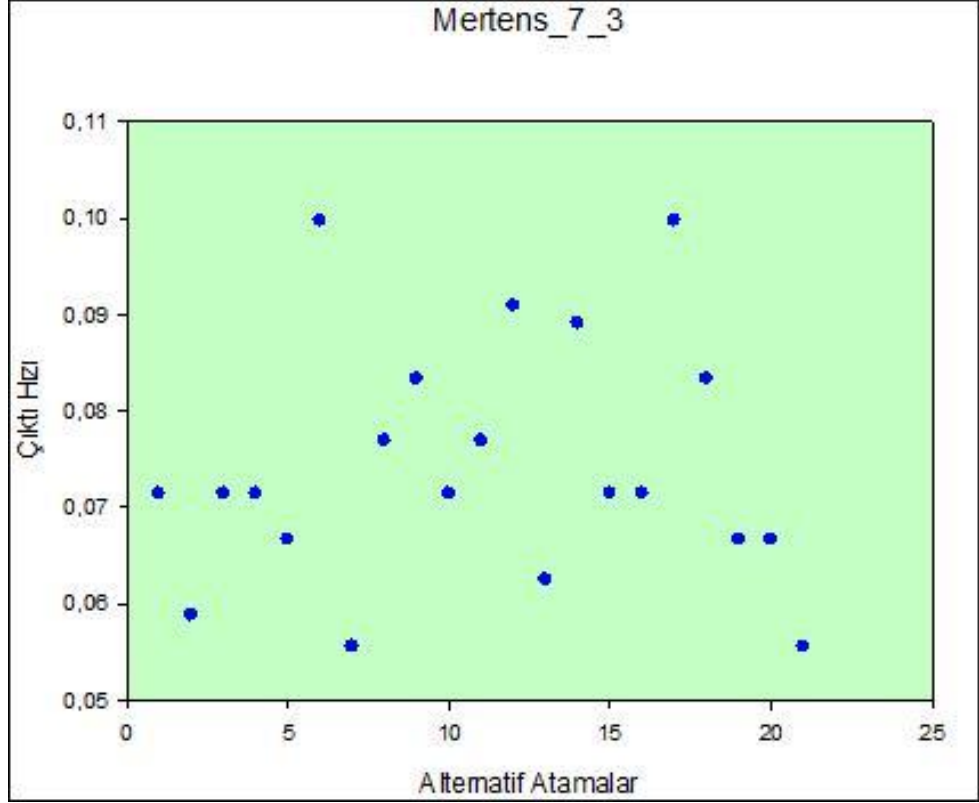
**Çizelge 8.5** Mertens (1967) 7 işli 3 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri

<b>Mertens (1967) 7 iş 3 istasyon</b>			
	<b>istasyon 1</b>	<b>istasyon 2</b>	<b>istasyon 3</b>
<b>En İyi Atama Kombinasyonu</b>	1,2,4	5,7	3,6
<b>Ortalama İstasyon Süresi</b>	9	9	10

**Çizelge 8.6** Mertens (1967) 7 işli 3 istasyonlu montaj hattı için performans değerleri

<b>Mertens (1967) 7 iş 3 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	0,898	0,899	10,027	10,021	0,06%
<b>İstasyon 2</b>	0,898	0,897			
<b>İstasyon 3</b>	0,997	0,997			

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.3'te verilmiştir.



Şekil 8.3 Mertens (1967) 7 iş 3 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

### 8.1.3. 7 İş 5 İstasyonlu Mertens (1967) Deneyi

Mertens montaj hattı için istasyon sayısının 5 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.7 ve 8.8’de verilmiştir.

Çizelge 8.7 Mertens (1967) 7 işli 5 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri

Mertens (1967) 7 iş 5 istasyon					
	istasyon 1	istasyon 2	istasyon 3	istasyon 4	istasyon 5
<b>En İyi Atama Kombinasyonu</b>	1,2	4,5	6	3	7
<b>Ortalama İstasyon Süresi</b>	6	7	6	4	5

**Çizelge 8.8** Mertens (1967) 7 işli 5 istasyonlu montaj hattı için performans değerleri

<b>Mertens (1967) 7 iş 5 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	0,856	0,856	7,006	7,01	-0,06%
<b>İstasyon 2</b>	0,999	0,999			
<b>İstasyon 3</b>	0,856	0,856			
<b>İstasyon 4</b>	0,571	0,57			
<b>İstasyon 5</b>	0,714	0,713			

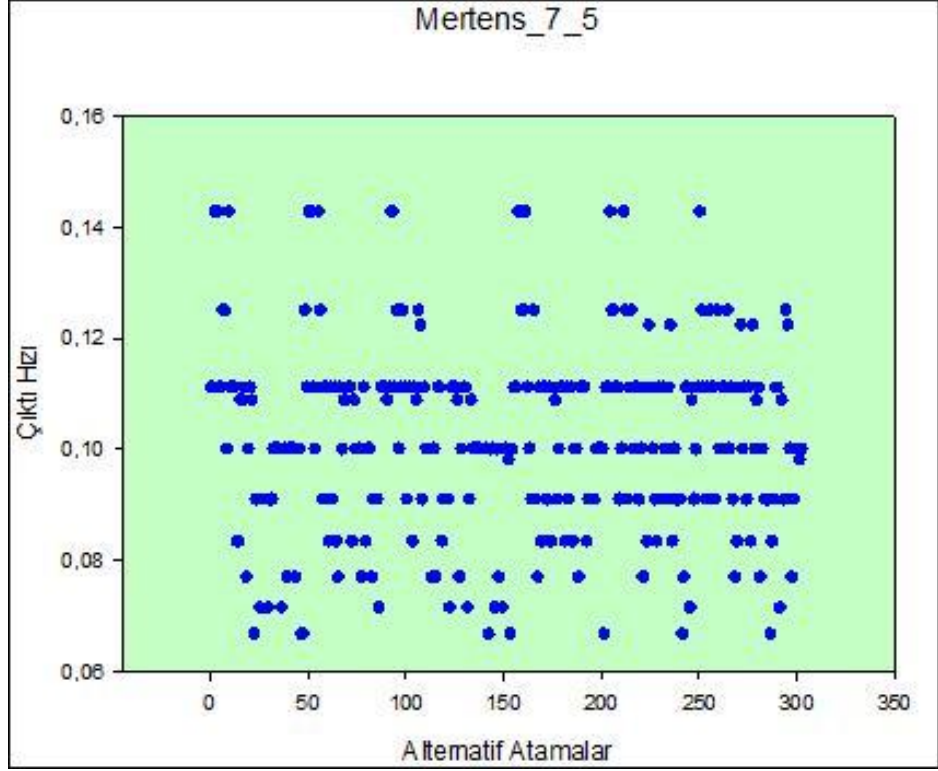
Problemin çözdürülmesi sonucunda alternatif en iyi atama kombinasyonları elde edilmiştir. Bu atama kombinasyonları çizelge 8.9’da verilmiştir.

**Çizelge 8.9** Mertens (1967) 7 işli 5 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları

<b>Mertens (1967) 7 iş 5 istasyon</b>					
	<b>istasyon 1</b>	<b>istasyon 2</b>	<b>istasyon 3</b>	<b>istasyon 4</b>	<b>istasyon 5</b>
<b>Alternatif Atama Kombinasyonları</b>	1,2	4,5	7	6	3
	1,2	4,5	6	7	3
	1,2	5	6	3,4	7
	1,2	4,5	3	6	7
	1,2	5	3,4	6	7

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.4’te verilmiştir.

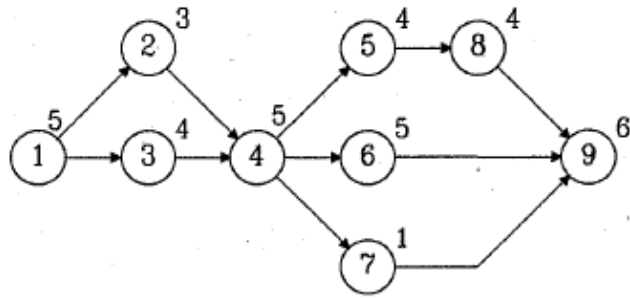




Şekil 8.4 Mertens (1967) 7 iş 5 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

#### 8.1.4. 9 İş 3 İstasyonlu Jaeschke (1964) Deneyi

Jaeschke (1964) montaj hattı için öncelik diyagramı şekil 8.5'te verilmiştir.



Şekil 8.5 Jaeschke (1964) 9 işli montaj hattı serimi

İstasyon sayısının 3 olduğu durum için en iyi atama kombinasyonu ve performans değerleri çizelge 8.10 ve 8.11’de verilmiştir.

**Çizelge 8.10** Jaeschke (1964) 9 işli 3 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri

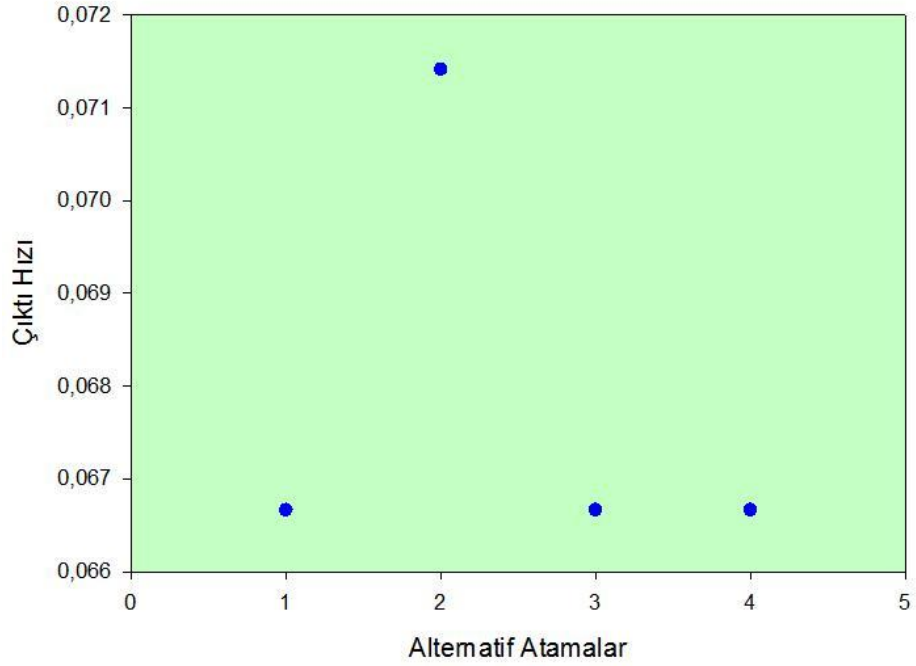
<b>Jaeschke (1964) 9 iş 3 istasyon</b>			
	<b>istasyon 1</b>	<b>istasyon 2</b>	<b>istasyon 3</b>
<b>En İyi Atama Kombinasyonu</b>	1,2,3	4,5,6	7,8,9
<b>Ortalama İstasyon Süresi</b>	12	14	11

**Çizelge 8.11** Jaeschke (1964) 9 işli 3 istasyonlu montaj hattı için performans değerleri

<b>Jaeschke (1964) 9 iş 3 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	0,857	0,858	14,002	14,007	-0,04%
<b>İstasyon 2</b>	0,9998	1,000			
<b>İstasyon 3</b>	0,786	0,783			

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.6’da verilmiştir.

Jaeschke\_9\_3



Şekil 8.6 Jaeschke (1964) 9 iş 3 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

### 8.1.5. 9 İş 4 İstasyonlu Jaeschke (1964) Deneyi

Jaeschke montaj hattı için istasyon sayısının 4 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.12 ve 8.13'te verilmiştir.

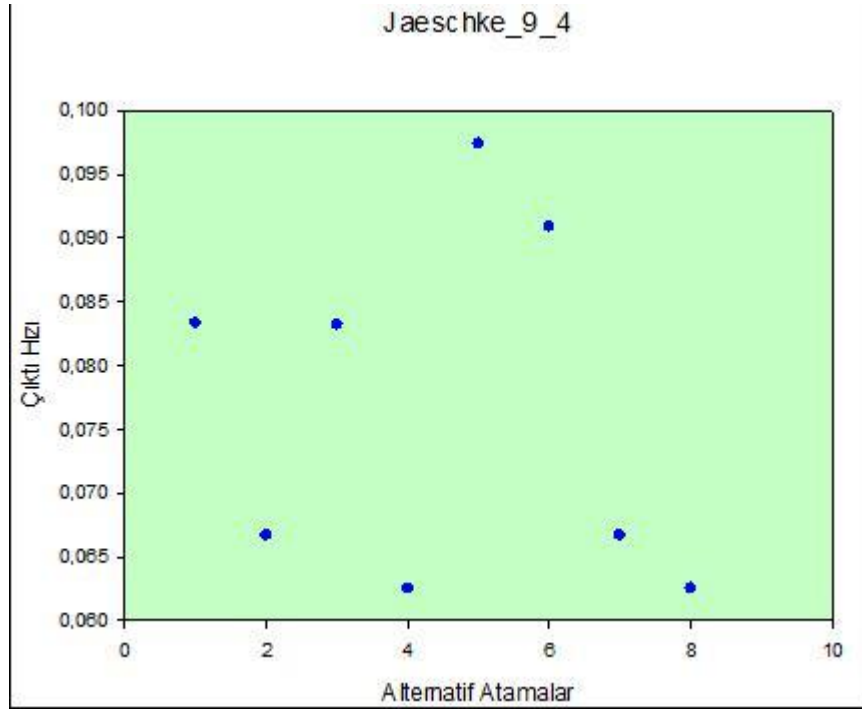
Çizelge 8.12 Jaeschke (1964) 9 işli 4 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri

Jaeschke (1964) 9 iş 4 istasyon				
	istasyon 1	istasyon 2	istasyon 3	istasyon 4
En İyi Atama Kombinasyonu	1,2,3	4	5,6,7	8,9
Ortalama İstasyon Süresi	9	8	10	10

**Çizelge 8.13** Jaeschke (1964) 9 işli 4 istasyonlu montaj hattı için performans değerleri

<b>Jaeschke (1964) 9 iş 4 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	0,8766	0,877	14,002	14,007	-0,04%
<b>İstasyon 2</b>	0,7792	0,780			
<b>İstasyon 3</b>	0,9740	0,974			
<b>İstasyon 4</b>	0,9740	0,973			

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.7’de verilmiştir.



**Şekil 8.7** Jaeschke (1964) 9 iş 4 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

### 8.1.6. 9 İş 7 İstasyonlu Jaeschke (1964) Deneyi

Jaeschke montaj hattı için istasyon sayısının 7 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.14 ve 8.15’te verilmiştir.

**Çizelge 8.14** Jaeschke (1964) 9 işli 7 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri

Jaeschke (1964) 9 iş 7 istasyon							
	istasyo n 1	istasyo n 2	istasyo n 3	istasyo n 4	istasyo n 5	istasyo n 6	istasyo n 7
<b>En İyi Atama Kombinasyonu</b>	1	2,3	4	5,7	8	6	9
<b>Ortalama İstasyon Süresi</b>	5	7	5	5	4	5	6

**Çizelge 8.15** Jaeschke (1964) 9 işli 7 istasyonlu montaj hattı için performans değerleri

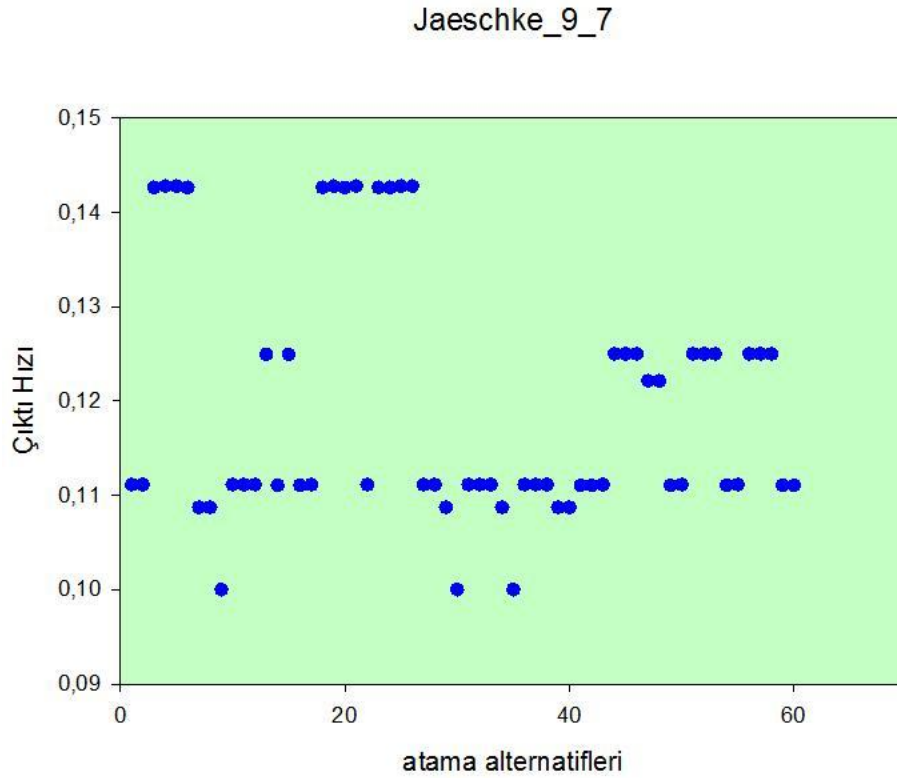
Jaeschke (1964) 9 iş 7 istasyon					
	İstasyon Doluluk Oranı		Çevrim Süresi		
	Önerilen Algoritma	Simülasyon	Önerilen Algoritma	Simülasyon	% Fark
<b>İstasyon 1</b>	0,714	0,714	7,005	7,000	0,07%
<b>İstasyon 2</b>	0,999	0,999			
<b>İstasyon 3</b>	0,714	0,714			
<b>İstasyon 4</b>	0,714	0,714			
<b>İstasyon 5</b>	0,571	0,571			
<b>İstasyon 6</b>	0,714	0,715			
<b>İstasyon 7</b>	0,857	0,857			

Problemin çözdürülmesi sonucunda alternatif en iyi atama kombinasyonları elde edilmiştir. Bu atama kombinasyonları çizelge 8.16’da verilmiştir.

**Çizelge 8.16** Jaeschke (1964) 9 işli 7 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları

Jaeschke (1964) 9 iş 7 istasyon							
	istasyon 1	istasyon 2	istasyon 3	istasyon 4	istasyon 5	istasyon 6	istasyon 7
Alternatif Atama Kombinasyonları	1	2,3	4	5	6	7,8	9
	1	2,3	4	5,7	6	8	9
	1	2,3	4	5	6	7,8	9
	1	2,3	4	6	5,7	8	9
	1	2,3	4	6	5	7,8	9

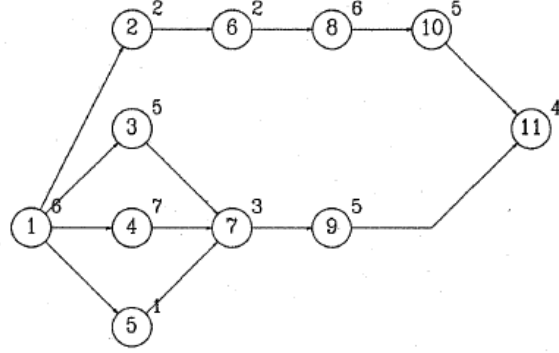
Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.8’de verilmiştir.



**Şekil 8.8** Jaeschke (1964) 9 iş 7 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

### 8.1.7. 11 İş 3 İstasyonlu Jackson (1956) Deneyi

Jackson (1956) montaj hattı için öncelik diyagramı şekil 8.9'da verilmiştir.



Şekil 8.9 Jackson 11 işli montaj hattı serimi

Jackson montaj hattı için istasyon sayısının 3 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.17 ve 8.18'de verilmiştir.

Çizelge 8.17 Jackson (1956) 11 işli 3 istasyonlu montaj hattı için görev atamaları ve istasyon süreleri

Jackson (1956) 11 iş 3 istasyon			
	istasyon 1	istasyon 2	istasyon 3
En İyi Atama Kombinasyonu	1,2,3,5,6	4,7,9	10,11
Ortalama İstasyon Süresi	15	15	15

**Çizelge 8.18** Jackson (1956) 11 işli 3 istasyonlu montaj hattı için performans değerleri

<b>Jackson (1956) 11 iş 3 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	0,962	0,961	15,6	15,617	-0,11%
<b>İstasyon 2</b>	0,962	0,961			
<b>İstasyon 3</b>	0,962	0,96			

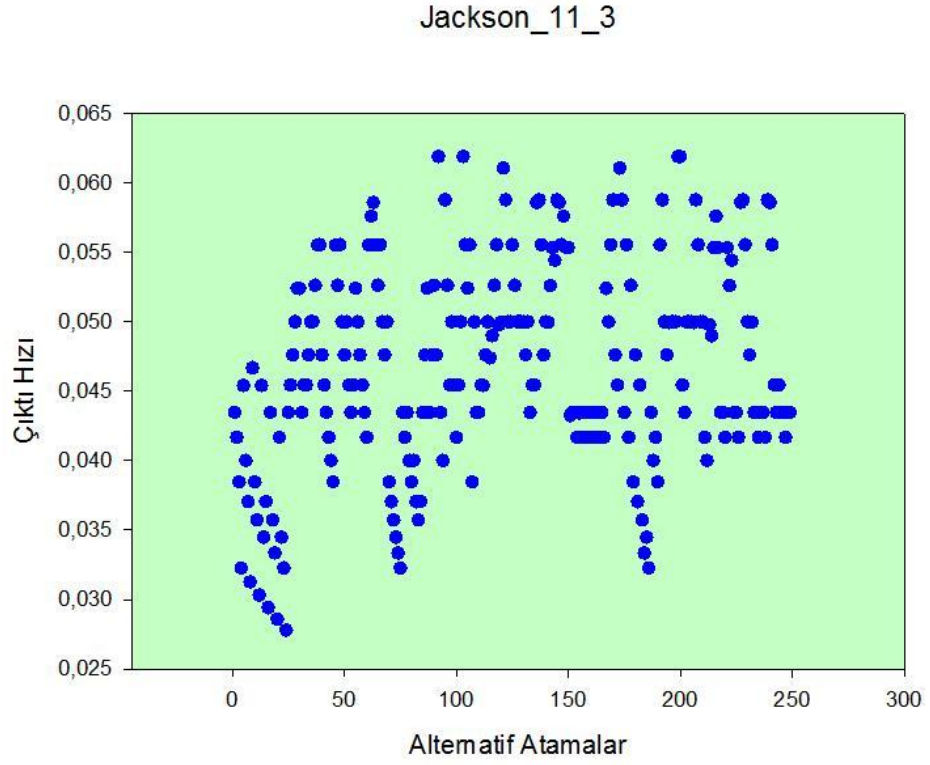
Problemin çözdürülmesi sonucunda alternatif en iyi atama kombinasyonları elde edilmiştir. Bu atama kombinasyonları çizelge 8.19’da verilmiştir.

**Çizelge 8.19** Jackson (1956) 11 işli 3 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları

<b>Jackson (1956) 11 iş 3 istasyon</b>			
	<b>istasyon 1</b>	<b>istasyon 2</b>	<b>istasyon 3</b>
<b>Alternatif Atama Kombinasyonları</b>	1,2,3,6	4,5,7,9	8,10,11
	1,2,4,5	3,6,7,9	8,10,11
	1,2,4	3,5,6,9	8,10,11

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.10’da verilmiştir.





**Şekil 8.10** Jackson(1956) 11 iş 3 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

### 8.1.8. 11 İş 4 İstasyonlu Jackson (1956) Deneyi

Jackson montaj hattı için istasyon sayısının 4 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.20 ve 8.21’de verilmiştir.

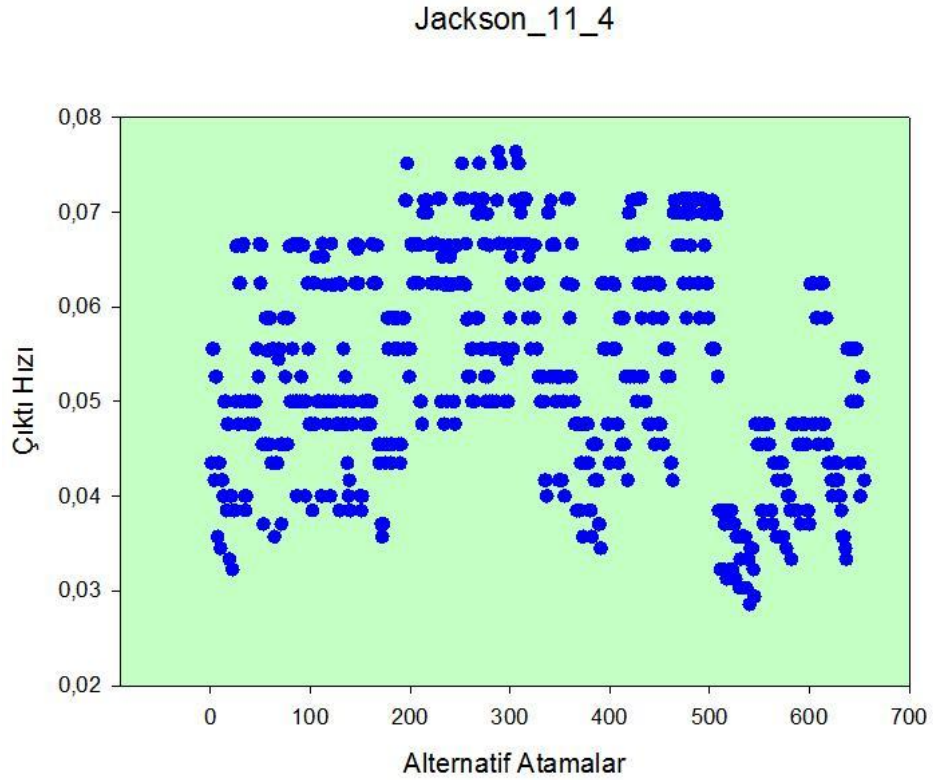
**Çizelge 8.20** Jackson (1956) 11 işli 4 istasyonlu montaj hattı için görev atamaları ve istasyon süreleri

<b>Jackson (1956) 11 iş 4 istasyon</b>				
	<b>istasyon 1</b>	<b>istasyon 2</b>	<b>istasyon 3</b>	<b>istasyon 4</b>
<b>En İyi Atama Kombinasyonu</b>	1,3,5	2,4,7	6,8,10	9,11
<b>Ortalama İstasyon Süresi</b>	11	12	13	9

**Çizelge 8.21** Jackson (1956) 11 işli 4 istasyonlu montaj hattı için performans değerleri

Jackson (1956) 11 iş 4 istasyon					
	İstasyon Doluluk Oranı		Çevrim Süresi		
	Önerilen Algoritma	Simülasyon	Önerilen Algoritma	Simülasyon	% Fark
İstasyon 1	0,843	0,843	13,093	13,053	0,31%
İstasyon 2	0,92	0,919			
İstasyon 3	0,997	0,997			
İstasyon 4	0,69	0,689			

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.11’de verilmiştir.



**Şekil 8.11** Jackson (1956) 11 iş 4 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

### 8.1.9. 11 İş 5 İstasyonlu Jackson (1956) Deneyi

Jackson montaj hattı için istasyon sayısının 5 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.22 ve 8.23'te verilmiştir.

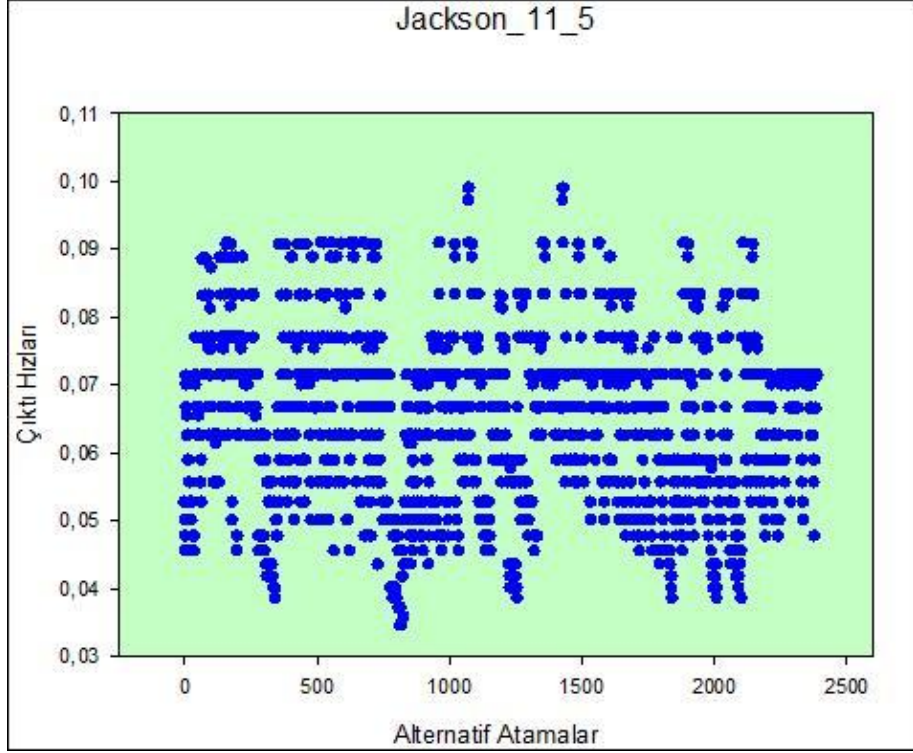
**Çizelge 8.22** Jackson (1956) 11 işli 5 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri

<b>Jackson (1956) 11 iş 5 istasyon</b>					
	<b>istasyon 1</b>	<b>istasyon 2</b>	<b>istasyon 3</b>	<b>istasyon 4</b>	<b>istasyon 5</b>
<b>En İyi Atama Kombinasyonu</b>	1,2,5	4,6	3,8	7,9	10,11
<b>Ortalama İstasyon Süresi</b>	9	8	9	10	9

**Çizelge 8.23** Jackson (1956) 11 işli 5 istasyonlu montaj hattı için performans değerleri

<b>Jackson (1956) 11 iş 5 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	0,891	0,892	10,105	10,092	0,13%
<b>İstasyon 2</b>	0,791	0,794			
<b>İstasyon 3</b>	0,891	0,892			
<b>İstasyon 4</b>	0,989	0,99			
<b>İstasyon 5</b>	0,891	0,891			

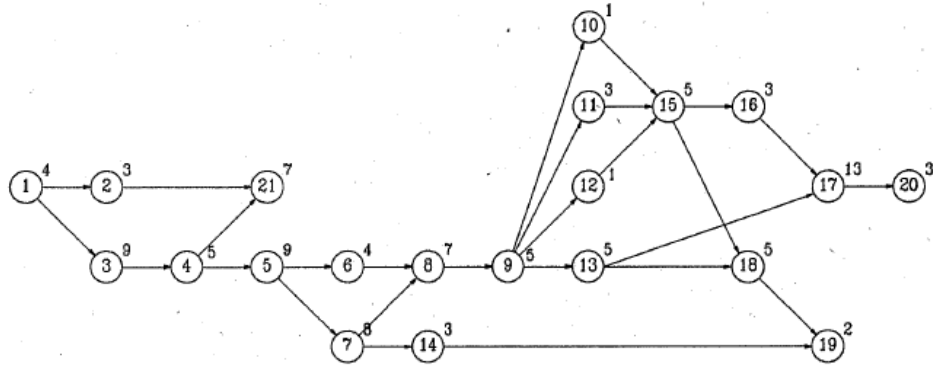
Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.12'de verilmiştir.



Şekil 8.12 Jackson(1956) 11 iş 5 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

### 8.1.10. 21 İş 3 İstasyonlu Mitchell (1957) Deneyi

Mitchell (1957) montaj hattı için öncelik diyagramı şekil 8.13'te verilmiştir.



Şekil 8.13 Mitchell (1957) 21 işli montaj hattı serimi

Mitchell montaj hattı için istasyon sayısının 3 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.24 ve 8.25’te verilmiştir.

**Çizelge 8.24** Mitchell (1957) 21 işli 3 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri

<b>Mitchell (1957) 21 iş 3 istasyon</b>			
	<b>istasyon 1</b>	<b>istasyon 2</b>	<b>istasyon 3</b>
<b>En İyi Atama Kombinasyonu</b>	1,3,4,5,7	2,6,8,9,10,11, 12,14,15,16	13,17,18, 19,20,21
<b>Ortalama İstasyon Süresi</b>	35	35	35

**Çizelge 8.25** Mitchell (1957) 21 işli 3 istasyonlu montaj hattı için performans değerleri

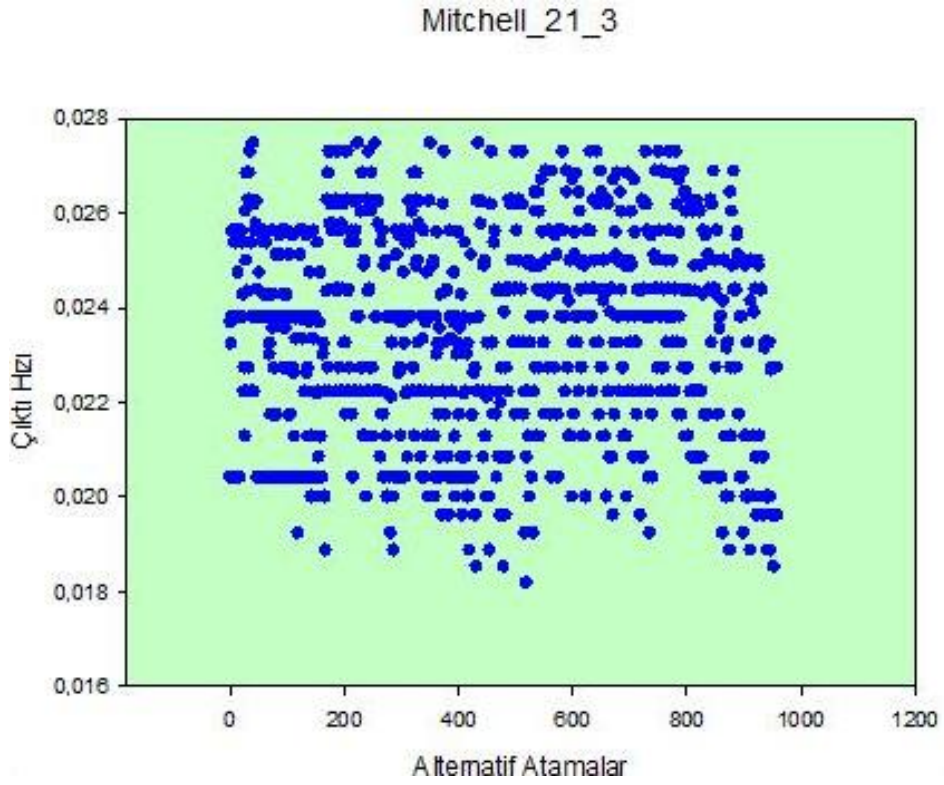
<b>Mitchell (1957) 21 iş 3 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	0,962	0,962	36,400	36,440	-0,11%
<b>İstasyon 2</b>	0,962	0,964			
<b>İstasyon 3</b>	0,962	0,959			

Problemin çözdürülmesi sonucunda alternatif en iyi atama kombinasyonları elde edilmiştir. Bu atama kombinasyonları çizelge 8.26’da verilmiştir.

**Çizelge 8.26** Mitchell (1957) 21 işli 3 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları

Mitchell (1957) 21 iş 3 istasyon			
	istasyon 1	istasyon 2	istasyon 3
Alternatif Atama Kombinasyonları	1,3,4,5,7	2,6,8,9,10,11,13,21	12,14,15,16,17,18,19,20
	1,3,4,5,7	2,6,8,9,10,11,13,21	12,14,15,16,17,18,19,20
	1,3,4,5,7	2,6,8,9,11,13,21	10,12,14,15,16,17,18,19,20
	1,3,4,5,7	2,6,8,9,12,13,14,21	10,11,15,16,17,18,19,20

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.14'te verilmiştir.



**Şekil 8.14** Mitchell(1957) 21 iş 3 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

### 8.1.11. 21 İş 5 İstasyonlu Mitchell (1957) Deneyi

Mitchell montaj hattı için istasyon sayısının 5 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.27 ve 8.28’de verilmiştir.

**Çizelge 8.27** Mitchell (1957) 21 işli 5 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri

<b>Mitchell (1957) 21 iş 5 istasyon</b>					
	<b>istasyon 1</b>	<b>istasyon 2</b>	<b>istasyon 3</b>	<b>istasyon 4</b>	<b>istasyon 5</b>
<b>En İyi Atama Kombinasyonu</b>	1,2,3,4	5,6,7	8,9,10, 12,21	11,13,15, 16,18	14,17, 19,20
<b>Ortalama İstasyon Süresi</b>	21	21	21	21	21

**Çizelge 8.28** Mitchell (1957) 21 işli 5 istasyonlu montaj hattı için performans değerleri

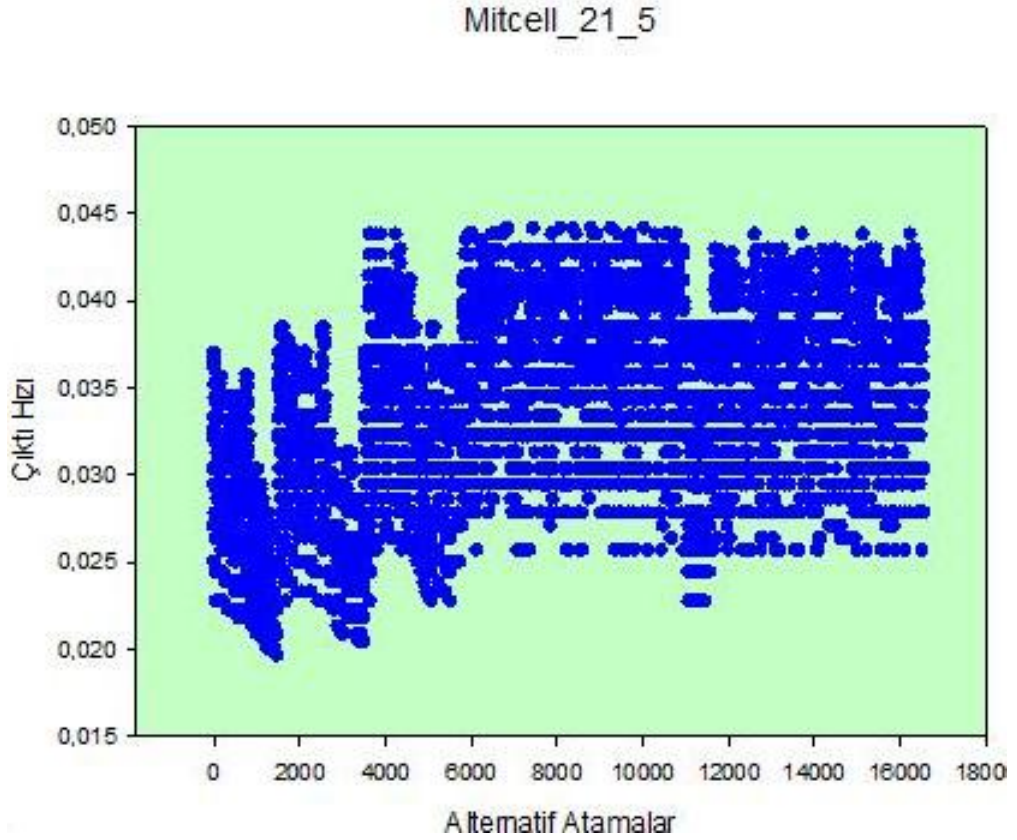
<b>Mitchell (1957) 21 iş 5 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	0,926	0,926	22,680	22,681	0,00%
<b>İstasyon 2</b>	0,926	0,926			
<b>İstasyon 3</b>	0,926	0,925			
<b>İstasyon 4</b>	0,926	0,925			
<b>İstasyon 5</b>	0,926	0,925			

Problemin çözdürülmesi sonucunda alternatif en iyi atama kombinasyonları elde edilmiştir. Bu atama kombinasyonları çizelge 8.29’da verilmiştir.

**Çizelge 8.29** Mitchell (1957) 21 işli 5 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları

Mitchell (1957) 21 iş 5 istasyon					
	istasyon 1	istasyon 2	istasyon 3	istasyon 4	istasyon 5
Alternatif Atama Kombinasyonları	1,2,3,4	5,6,7	8,9,10,12,21	11,13,14,15,18	16,17,19,20
	1,2,3,4	5,6,7	8,9,11,12,13	10,15,16,18,21	14,17,19,20
	1,2,3,4	5,6,7	8,9,11,12,13	10,14,15,18,21	16,17,19,20
	1,2,3,4	5,6,7	8,9,12,13,14	10,11,15,18,21	16,17,19,20
	1,2,3,4	5,6,7	8,9,10,11,13	12,15,16,18,21	14,17,19,20

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.15'te verilmiştir.



**Şekil 8.15** Mitchell (1957) 21 iş 5 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği



### 8.1.12. 21 İş 8 İstasyonlu Mitchell (1957) Deneyi

Mitchell montaj hattı için istasyon sayısının 8 olması durumunda en iyi atama kombinasyonu ve performans değerleri 8.30 ve 8.31’de verilmiştir.

**Çizelge 8.30** Mitchell (1957) 21 işli 8 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri

Mitchell (1957) 21 iş 8 istasyon								
	istasyon 1	istasyon 2	istasyon 3	istasyon 4	istasyon 5	istasyon 6	istasyon 7	istasyon 8
<b>En İyi Atama Kombinasyonu</b>	1,3	2,4	5,6	7,8,12	9,10,11,15	13,16,18	17, 20	14,19,21
<b>Ortalama İstasyon Süresi</b>	13	8	13	16	14	13	16	12

**Çizelge 8.31** Mitchell (1957) 21 işli 8 istasyonlu montaj hattı için performans değerleri

Mitchell (1957) 21 iş 8 istasyon					
	İstasyon Doluluk Oranı		Çevrim Süresi		
	Önerilen Algoritma	Simülasyon	Önerilen Algoritma	Simülasyon	% Fark
<b>İstasyon 1</b>	0,784	0.783	16,007	16,524	-3,23%
<b>İstasyon 2</b>	0,482	0.483			
<b>İstasyon 3</b>	0,784	0.785			
<b>İstasyon 4</b>	0,964	0.964			
<b>İstasyon 5</b>	0,844	0.844			
<b>İstasyon 6</b>	0,784	0.783			
<b>İstasyon 7</b>	0,964	0.963			
<b>İstasyon 8</b>	0,723	0.725			

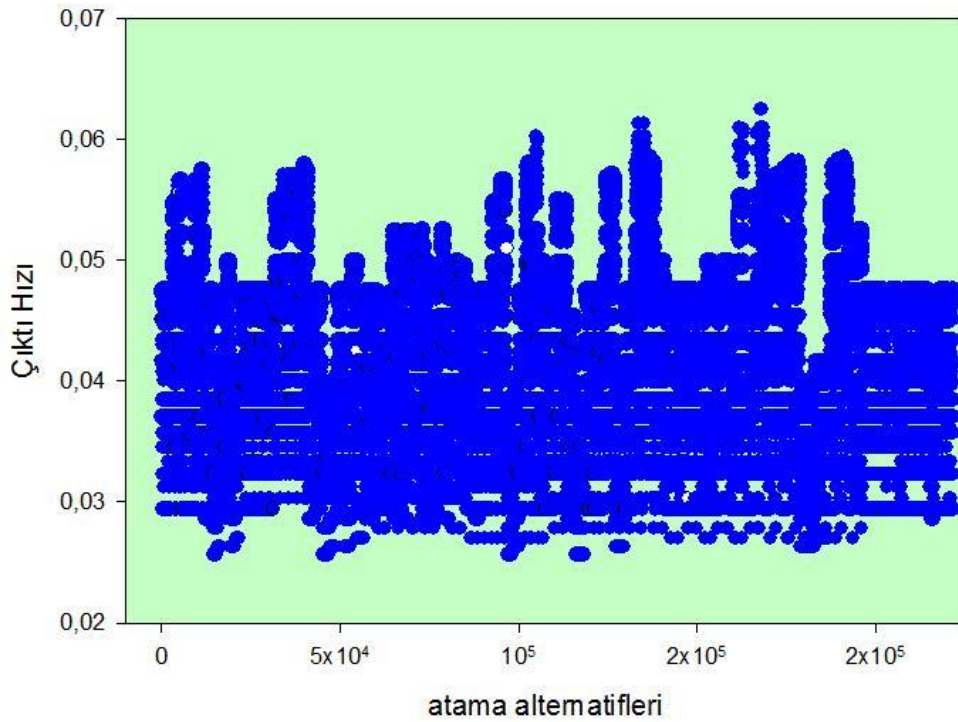
Problemin çözdürülmesi sonucunda alternatif en iyi atama kombinasyonları elde edilmiştir. Bu atama kombinasyonları çizelge 8.32’de verilmiştir.

**Çizelge 8.32** Mitchell (1957) 21 işli 8 istasyonlu montaj hattı alternatif en iyi atama kombinasyonları

Mitchell (1957) 21 iş 8 istasyon								
	istasyo n 1	istasyo n 2	istasyo n 3	istasyo n 4	istasyo n 5	istasyo n 6	istasyo n 7	istasyo n 8
Alternatif Atama Kombinasyo nları	1,3	2,4	5,6	7,8,12	9,10, 11,15	13,14, 16,18	17	19,20, 21
	1,3	2,4	5,6	7,8,12	9,10, 11,15	13,16, 18	14,17	19,20, ,21
	1,3	2,4	5,6	7,8,12	9,10, 11,15	13,14, 18	16,17	19,20, 21
	1,3	2,4	5,6	7,8,12	9,10, 11,15	13,21	16,17	14,18, 19,20

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.16'da verilmiştir.

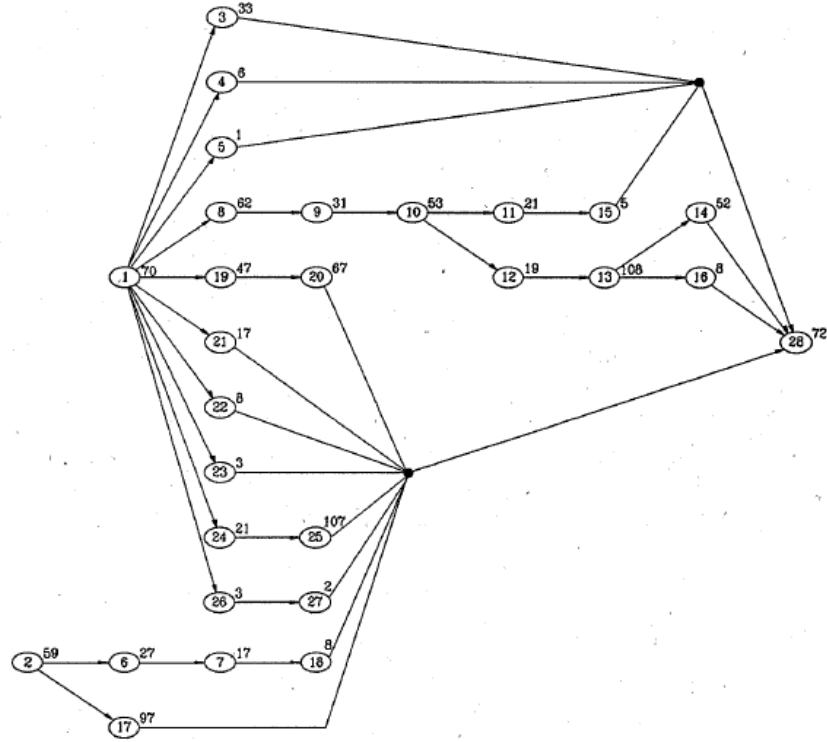
Mitchell\_21\_8



**Şekil 8.16** Mitchell (1957) 21 iş 8 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

### 8.1.13. 28 İş 4 İstasyonlu Heskiaoff (1968) Deneyi

Heskiaoff (1968) montaj hattı için öncelik diyagramı şekil 8.17’de verilmiştir.



Şekil 8.17 Heskiaoff (1968) 28 işli montaj hattı serimi

Heskiaoff montaj hattı için istasyon sayısının 4 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.33 ve 8.34’te verilmiştir.

Çizelge 8.33 Heskiaoff (1968) 28 işli 4 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süreleri

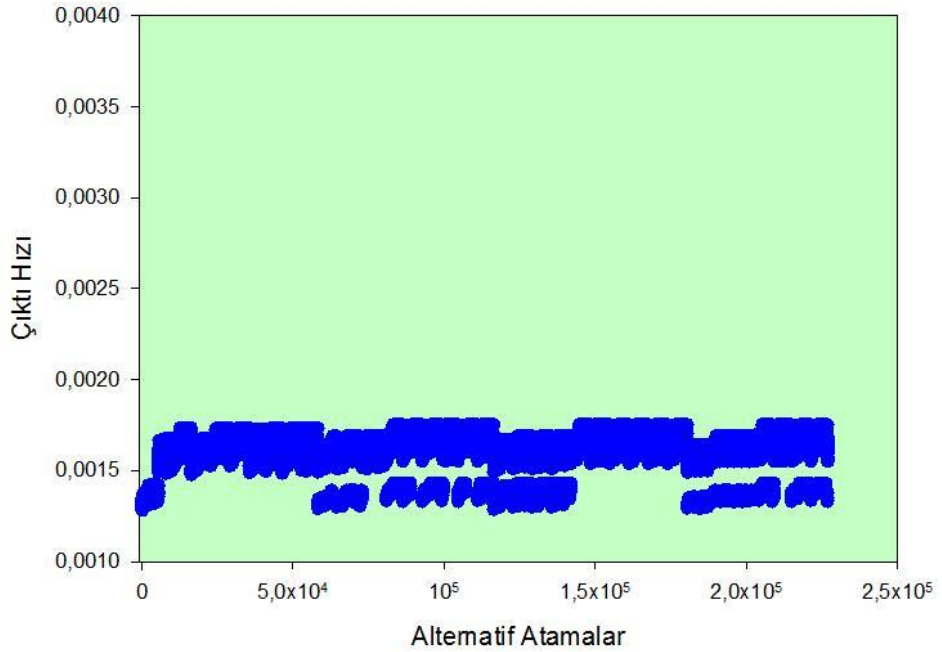
Heskiaoff (1968) 28 iş 4 istasyon				
	istasyon 1	istasyon 2	istasyon 3	istasyon 4
<b>En İyi Atama Kombinasyonu</b>	1,2,3,5,6,8,9, 10, 12,17,19,20,26,27	4,7,11,13,14,15,16 , 18, 21,23,24,25	22,27	28
<b>Ortalama İstasyon Süresi</b>	571	373	8	72

**Çizelge 8.34** Heskiaoff (1968) 28 işli 4 istasyonlu montaj hattı için performans değerleri

<b>Heskiaoff (1968) 28 iş 4 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	1,000	1,000	571	564	1,23%
<b>İstasyon 2</b>	0,653	0,646			
<b>İstasyon 3</b>	0,014	0,014			
<b>İstasyon 4</b>	0,126	0,128			

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.18’de verilmiştir.

Heskiaoff\_28\_4 depht-first search



**Şekil 8.18** Heskiaoff (1968) 28 iş 4 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

#### 8.1.14. 28 İş 5 İstasyonlu Heskiaoff (1968) Deneyi

Heskiaoff montaj hattı için istasyon sayısının 5 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.35 ve 8.36'da verilmiştir.

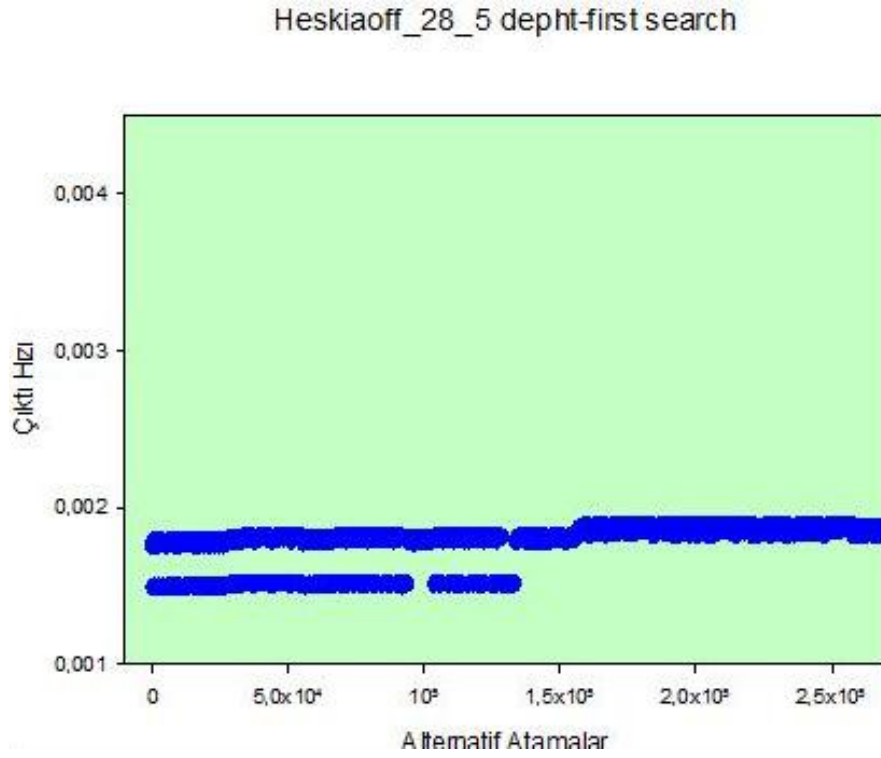
**Çizelge 8.35** Heskiaoff (1968) 28 işli 5 istasyonlu montaj hattı için görev atamaları ve ortalama istasyon süresi

<b>Heskiaoff (1968) 28 iş 5 istasyon</b>					
	<b>istasyon 1</b>	<b>istasyon 2</b>	<b>istasyon 3</b>	<b>istasyon 4</b>	<b>istasyon 5</b>
<b>En İyi Atama Kombinasyonu</b>	1,2,3,6,7,8,9,17,19,20,21	5,10,11,12,13,14,15,16,24,25,26,27	4,18,23	22	28
<b>Ortalama İstasyon Süresi</b>	527	400	17	8	72

**Çizelge 8.36** Heskiaoff (1968) 28 işli 5 istasyonlu montaj hattı için performans değerleri

<b>Heskiaoff (1968) 28 iş 5 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	1,000	1,000	527	527,334	-0,06%
<b>İstasyon 2</b>	0,759	0,755			
<b>İstasyon 3</b>	0,032	0,032			
<b>İstasyon 4</b>	0,015	0,015			
<b>İstasyon 5</b>	0,137	0,136			

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.19'da verilmiştir.



**Şekil 8.19** Heskiaoff (1968) 28 iş 5 istasyonlu montaj hattı için çıktı hızlarının değişim grafiği

## 8.2. Çok Noktalı Arama Stratejisi Deneyleri

Öncelikle derinlik arama stratejisinde karşılaşılan dallarda aramanın sıkışması problemini aşmak amacıyla, arama alanının değişik noktalarında aramayı sağlayan çok noktalı arama stratejisi kullanılmıştır. Kullanılan arama stratejisi ile çözüme yaklaşılamayan deney setleri için çözümler elde edilmiştir. Algoritmanın çalıştırıldığı bilgisayarın donanımsal özelliklerinden dolayı en fazla 250000 atama alternatifi için performans değerleri hesaplanmıştır.

Öncelikle derinlik arama stratejisinde optimum sonuca yaklaşamayan Heskiaoff (1968) deney setleri ve Sawyer (1970) deney setleri için çok noktalı arama stratejisi ile algoritma çalıştırılmıştır. Elde edilen sonuçlar bu bölümde her bir deney için ayrı ayrı verilmiştir.

### 8.2.1. 28 İş 4 İstasyonlu Heskiaoff (1968) Deneyi

Heskiaoff montaj hattı için istasyon sayısının 4 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.37 ve 8.38’de verilmiştir.

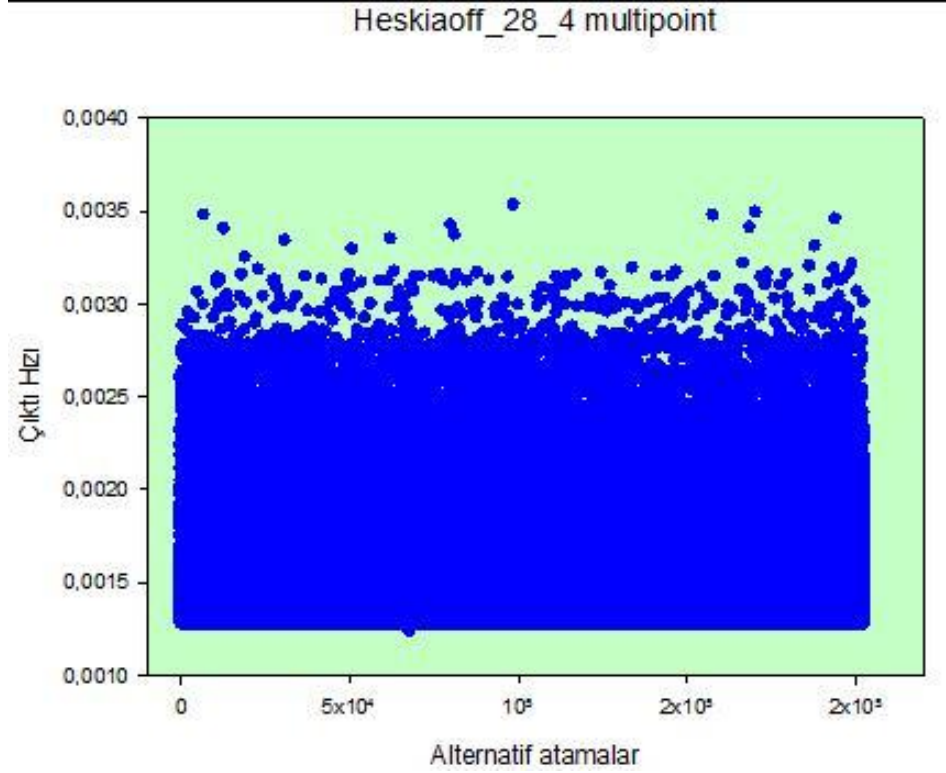
**Çizelge 8.37** Heskiaoff (1968) 28 işli 4 istasyonlu montaj hattı için çoklu arama stratejisi görev atamaları ve ortalama istasyon süreleri

<b>Heskiaoff (1968) 28 iş 4 istasyon</b>				
	<b>istasyon 1</b>	<b>istasyon 2</b>	<b>istasyon 3</b>	<b>istasyon 4</b>
<b>En İyi Atama Kombinasyonu</b>	1,3,4,5,19,20, 21,22,23,26,27	8,9,24,25	10,11,12, 13,14,15,16	2,6,7,17,18, 28
<b>Ortalama İstasyon Süresi</b>	257	221	266	280

**Çizelge 8.38** Heskiaoff (1968) 28 işli 4 istasyonlu montaj hattı için çoklu arama stratejisi performans değerleri

<b>Heskiaoff (1968) 28 iş 4 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	0,907	0,908	283,23	284,60	-0,48%
<b>İstasyon 2</b>	0,780	0,780			
<b>İstasyon 3</b>	0,939	0,942			
<b>İstasyon 4</b>	0,989	0,985			

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.20’de verilmiştir.



**Şekil 8.20** Heskiaoff (1968) 28 iş 4 istasyonlu montaj hattı için çok noktalı arama stratejisine göre çıktı hızlarının değişim grafiği

### 8.2.2. 28 İş 5 İstasyonlu Heskiaoff (1968) Deneyi

Heskiaoff montaj hattı için istasyon sayısının 5 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.39 ve 8.40'ta verilmiştir.

**Çizelge 8.39** Heskiaoff (1968) 28 işli 5 istasyonlu montaj hattı için çoklu arama stratejisi görev atamaları ve ortalama istasyon süreleri

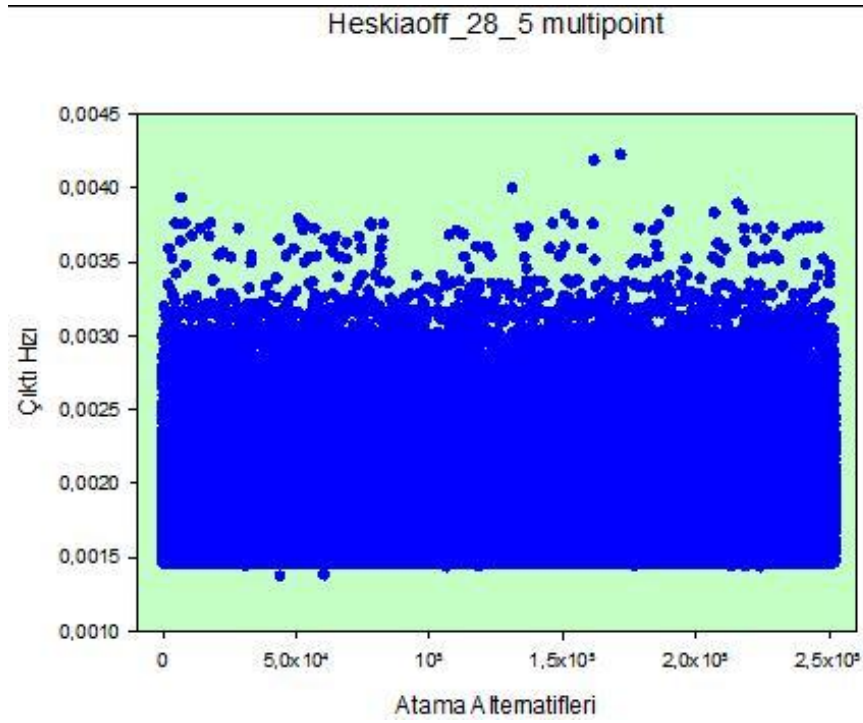
<b>Heskiaoff (1968) 28 iş 5 istasyon</b>					
	<b>İstasyon 1</b>	<b>İstasyon 2</b>	<b>İstasyon 3</b>	<b>İstasyon 4</b>	<b>İstasyon 5</b>
<b>En İyi Atama Kombinasyonu</b>	1,3,4,5,8,9, 21,22,23,26,27	10,11,15, 19,20	2,6,7,17, 18	12,13,14	16,24,25, 28
<b>Ortalama İstasyon Süresi</b>	236	193	208	179	208



**Çizelge 8.40** Heskiaoff (1968) 28 işli 5 istasyonlu montaj hattı için çoklu arama stratejisi performans değerleri

Heskiaoff (1968) 28 iş 5 istasyon					
	İstasyon Doluluk Oranı		Çevrim Süresi		
	Önerilen Algoritma	Simülasyon	Önerilen Algoritma	Simülasyon	% Fark
İstasyon 1	0,996	0,997	236,89	236,34	0,23%
İstasyon 2	0,815	0,818			
İstasyon 3	0,878	0,878			
İstasyon 4	0,756	0,753			
İstasyon 5	0,878	0,885			

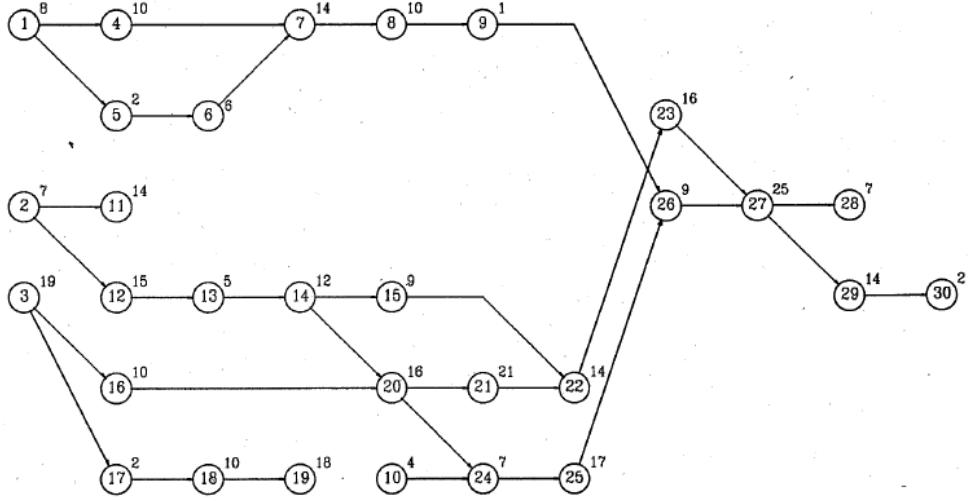
Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.21’de verilmiştir.



**Şekil 8.21** Heskiaoff (1968) 28 iş 5 istasyonlu montaj hattı için çok noktalı arama stratejisine göre çıktı hızlarının değişim grafiği

### 8.2.3. 30 İş 5 İstasyonlu Sawyer (1970) Deneyi

Sawyer (1970) montaj hattı için öncelik diyagramı şekil 8.22’de verilmiştir.



Şekil 8.22 Sawyer 30 işli montaj hattı serimi

Sawyer montaj hattı için istasyon sayısının 5 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.41 ve 8.42’de verilmiştir.

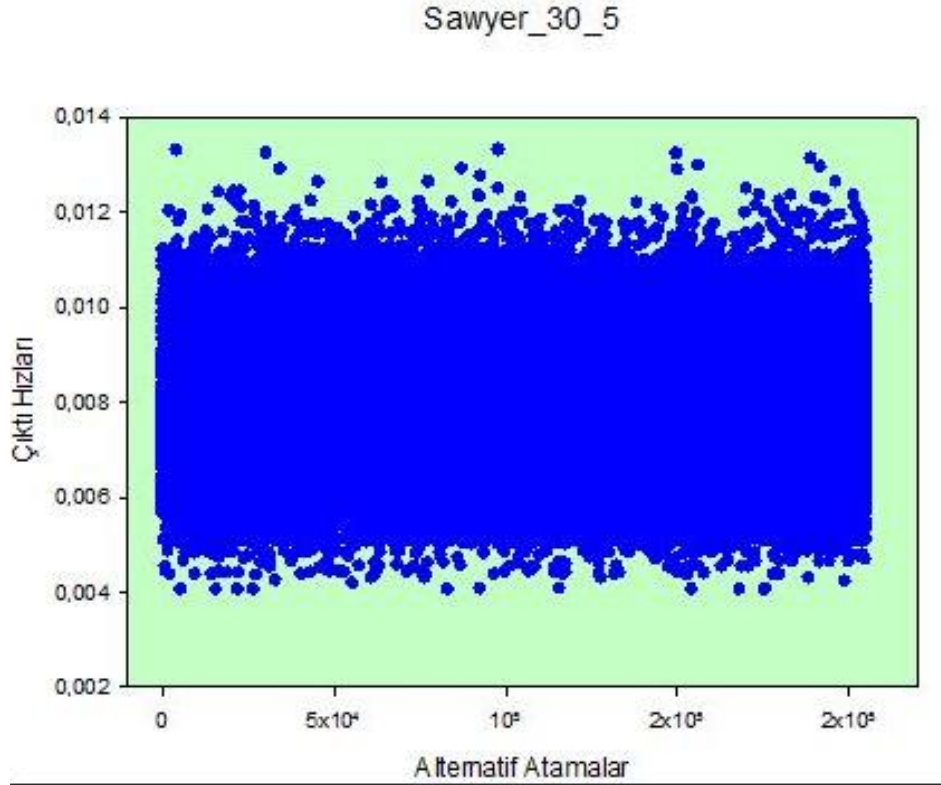
Çizelge 8.41 Sawyer (1970) 30 işli 5 istasyonlu montaj hattı için çoklu arama stratejisi görev atamaları ve ortalama istasyon sayıları

Sawyer (1970) 30 iş 5 istasyon					
	İstasyon 1	İstasyon 2	İstasyon 3	İstasyon 4	İstasyon 5
En İyi Atama Kombinasyonu	3,10,16, 17,18,19	2,11,12,13 ,14,15	20,21,22,24 ,25	1,4,5,6,7,8,9 ,26	23,27,28,29 ,30
Ortalama İstasyon Süresi	63	62	75	54	64

**Çizelge 8.42** Sawyer (1970) 30 işli 5 istasyonlu montaj hattı için çoklu arama stratejisi performans değerleri

Sawyer (1970) 30 iş 5 istasyon					
	İstasyon Doluluk Oranı		Çevrim Süresi		
	Önerilen Algoritma	Simülasyon	Önerilen Algoritma	Simülasyon	% Fark
İstasyon 1	0,839	0,843	75,15	75,72	-
İstasyon 2	0,825	0,833			
İstasyon 3	0,999	0,998			
İstasyon 4	0,719	0,723			
İstasyon 5	0,852	0,856			

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.23'te verilmiştir.



**Şekil 8.23** Sawyer (1970) 30 iş 5 istasyonlu montaj hattı için çok noktalı arama stratejisine göre çıktı hızlarının değişim grafiği

#### 8.2.4. 30 İş 8 İstasyonlu Sawyer (1970) Deneyi

Sawyer montaj hattı için istasyon sayısının 8 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.43 ve 8.44’te verilmiştir.

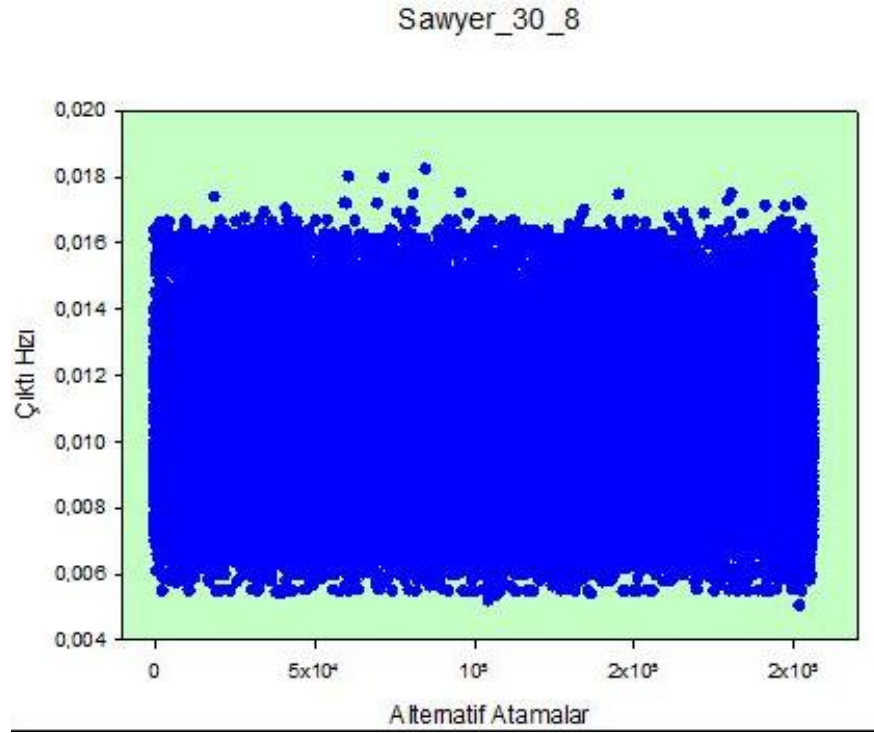
**Çizelge 8.43** Sawyer (1970) 30 işli 8 istasyonlu montaj hattı için çoklu arama stratejisi görev atamaları ve ortalama istasyon süreleri

Sawyer (1970) 30 iş 8 istasyon								
	istasyon 1	istasyon 2	istasyon 3	istasyon 4	istasyon 5	istasyon 6	istasyon 7	istasyon 8
<b>En İyi Atama Kombinasyonu</b>	1,2,4,10,12,13	3,14,15,16	20,21	22,23	17,18,19,24,25	5,6,7,8,9,26	27,28,29,30	11
<b>Ortalama İstasyon Süresi</b>	49	50	37	30	54	36	48	14

**Çizelge 8.44** Sawyer (1970) 30 işli 8 istasyonlu montaj hattı için çoklu arama stratejisi performans değerleri

Sawyer (1970) 30 iş 8 istasyon					
	İstasyon Doluluk Oranı		Çevrim Süresi		
	Önerilen Algoritma	Simülasyon	Önerilen Algoritma	Simülasyon	% Fark
<b>İstasyon 1</b>	0,891	0,895	54,88	54,72	0,30%
<b>İstasyon 2</b>	0,717	0,912			
<b>İstasyon 3</b>	0,769	0,677			
<b>İstasyon 4</b>	0,821	0,549			
<b>İstasyon 5</b>	0,315	0,986			
<b>İstasyon 6</b>	0,786	0,660			
<b>İstasyon 7</b>	0,996	0,874			
<b>İstasyon 8</b>	0,245	0,256			

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.24’te verilmiştir.



**Şekil 8.24** Sawyer (1970) 30 iş 8 istasyonlu montaj hattı için çok noktalı arama stratejisine göre çıktı hızlarının değişim grafiği

### 8.2.5. 30 İş 13 İstasyonlu Sawyer (1970) Deneyi

Sawyer montaj hattı için istasyon sayısının 13 olması durumunda en iyi atama kombinasyonu ve performans değerleri çizelge 8.45 ve 8.46'da verilmiştir.

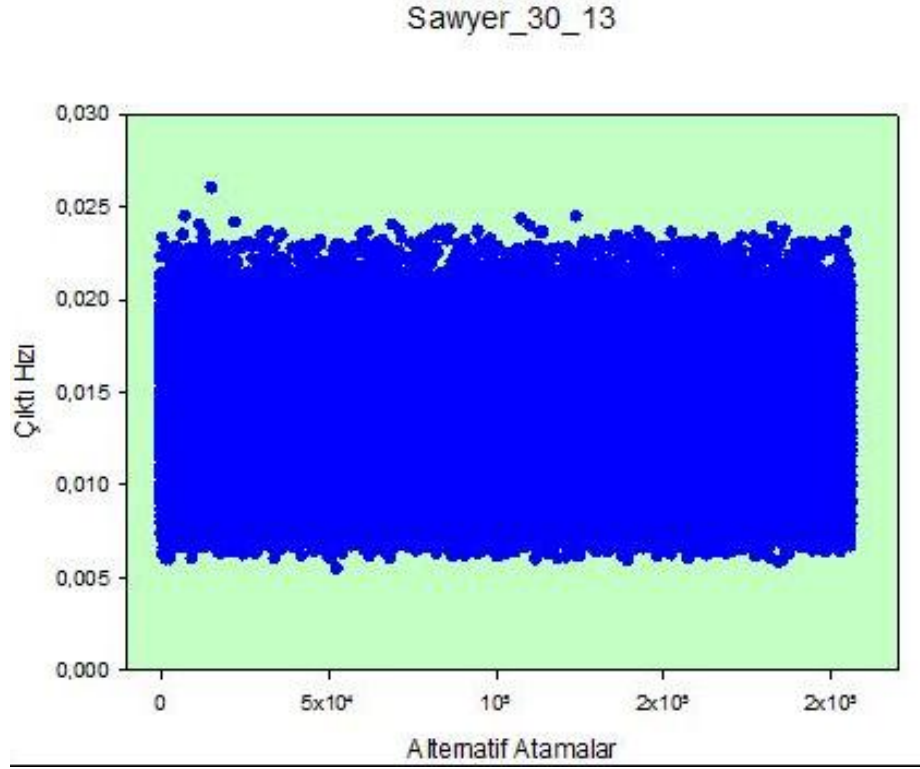
**Çizelge 8.45** Sawyer (1970) 30 işli 13 istasyonlu montaj hattı için çoklu arama stratejisi görev atamaları ve ortalama istasyon süreleri

<b>Sawyer (1970) 30 iş 13 istasyon</b>							
	İstasyon 1	İstasyon 2	İstasyon 3	İstasyon 4	İstasyon 5	İstasyon 6	İstasyon 7
<b>En İyi Atama Kombinasyonu</b>	2,10,12	13,14,15	3,16,17	20,24	21,25	22,23	1,4,5,6
<b>Ortalama İstasyon Süresi</b>	26	26	31	23	38	30	26
	İstasyon 8	İstasyon 9	İstasyon 10	İstasyon 11	İstasyon 12	İstasyon 13	
<b>En İyi Atama Kombinasyonu</b>	18	19	7,8,9,26	27,28	11	29,30	
<b>Ortalama İstasyon Süresi</b>	10	18	28	32	14	16	

**Çizelge 8.46** Sawyer (1970) 30 işli 13 istasyonlu montaj hattı için çoklu arama stratejisi performans değerleri

<b>Sawyer (1970) 30 iş 13 istasyon</b>					
	<b>İstasyon Doluluk Oranı</b>		<b>Çevrim Süresi</b>		
	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>Önerilen Algoritma</b>	<b>Simülasyon</b>	<b>% Fark</b>
<b>İstasyon 1</b>	0,681	0,678	38,42	38,22	0,52%
<b>İstasyon 2</b>	0,681	0,680			
<b>İstasyon 3</b>	0,812	0,808			
<b>İstasyon 4</b>	0,603	0,602			
<b>İstasyon 5</b>	0,996	0,995			
<b>İstasyon 6</b>	0,786	0,785			
<b>İstasyon 7</b>	0,681	0,680			
<b>İstasyon 8</b>	0,262	0,262			
<b>İstasyon 9</b>	0,472	0,473			
<b>İstasyon 10</b>	0,734	0,734			
<b>İstasyon 11</b>	0,838	0,837			
<b>İstasyon 12</b>	0,367	0,365			
<b>İstasyon 13</b>	0,419	0,419			

Çözüm sonuçlarına ilişkin atama alternatiflerinin aldığı çıktı hızlarının değişimini gösteren grafik şekil 8.25'te verilmiştir.



**Şekil 8.25** Sawyer (1970) 30 iş 13 istasyonlu montaj hattı için çok noktalı arama stratejisine göre çıktı hızlarının değişim grafiği

### 8.3. Deneysel Çalışmaların Sonuçlarının Değerlendirilmesi

Deneysel çalışmalar öncelikle derinlik ve çoklu arama olmak üzere iki arama stratejisine göre yapılmıştır. Öncelikle derinlik arama stratejisinde, deneysel çalışmalar karşılaştırıldığında görev sayısı arttığında öncelikle derinliğine arama stratejisi yetersiz kalmakta ve optimum çözüme ulaşamamaktadır. Optimum çözüme yakın bir değer de elde edilememiştir. Çözüm dallanmada alt dallarda sıkışmakta ve çözüm iyileşmemektedir.

Elde edilen sonuçlar ve Uğurdağ vd. (1997)'nin çalışmalarında elde ettikleri sonuçlarla karşılaştırılması çizelge 8.47'de verilmiştir. Küçük boyutlu problemlerde optimum sonuçlar elde edilmiştir. Uğurdağ vd. (1997)'nin çalışmalarında elde

ettikleri sonuçlarla yaklaşık değerler elde edilmiştir. 11 deney seti için karşılaştırılan çalışmaya çok yakın veya daha iyi değerler elde edilmiştir. 1 deney seti için ise çözüme yaklaşılmıştır. Ancak, 2 deney seti için önceki paragrafta bahsedildiği üzere dallanmada alt dallarda algoritma sıkışmış ve donanım yetersizliğinde dolayı bütün çözümlere ulaşılamadan sonlandırılmıştır. Bu çözümlerin karşılaştırılmaları 8.28’de verilmiştir.

**Çizelge 8.47** Öncelikle derinlik arama stratejisi için çözüm sonuçları

<b>Depth-First Arama için sistem çıktı hızı değerleri</b>						
<b>Deney Seti_görev sayısı_istasyon sayısı</b>	<b>En büyük çıktı hızı değeri</b>	<b>Ortalama çıktı hızı değeri</b>	<b>En Küçük çıktı hızı değeri</b>	<b>Çevrim süresi</b>	<b>Alternatif Çözüm Sayısı</b>	<b>Uğurdağ vd. (1997) çevrim süresi</b>
Mertens_7_2	0,0700	0,0549	0,0417	14,280	28	15
Mertens_7_3	0,0997	0,0786	0,0625	10,027	21	11
Mertens_7_5	0,1427	0,1018	0,0667	7,006	304	7
Jaeschke_9_3	0,0714	0,0679	0,0667	14,002	4	14
Jaeschke_9_4	0,0974	0,0766	0,0625	10,267	8	10
Jaeschke_9_7	0,1428	0,1197	0,1000	7,005	60	7
Jackson_11_3	0,0641	0,0480	0,0286	15,600	250	16
Jackson_11_4	0,0764	0,0532	0,0286	13,093	656	13
Jackson_11_5	0,0990	0,0650	0,0345	10,105	3810	11
Mitchell_21_3	0,0275	0,0234	0,0182	36,400	960	36
Mitchell_21_5	0,0441	0,0337	0,0196	22,680	16578	23
Mitchell_21_8	0,0625	0,0422	0,0256	16,007	221000	14
Heskiaoff_28_4	0,0018	0,0016	0,0013	571,000	226000	258
Heskiaoff_28_5	0,0019	0,0018	0,0015	527,000	268000	210

Çoklu arama stratejisi ile yapılan deneylerde ise, öncelikle derinlik arama stratejisi ile yapılan deneyler sonucunda alt dallarda sıkışan çözüm algoritmasının uygun çözüm alanının farklı bölgelerinde arama yapabilmesini sağlamak ve böylelikle optimum sonuca ulaşmak veya yakın değerler elde etmek amaçlanmıştır.



Yapılan 5 deney sonucunda optimuma yakın değerler elde edilmiştir, ancak optimum sonuca ulaşamamıştır. Deneysel çalışmaların sonuçları ve Uğurdağ vd.(1997)'nin çalışmaları ile karşılaştırılması çizelge 8.48'de verilmiştir.

**Çizelge 8.48** Çok noktalı arama stratejisi için çözüm sonuçları

<b>Çok noktalı arama stratejisi için sistem çıktı hızı değerleri</b>						
<b>Deney Seti_görev sayısı_istasyon sayısı</b>	<b>En büyük çıktı hızı değeri</b>	<b>Ortalama çıktı hızı değeri</b>	<b>En Küçük çıktı hızı değeri</b>	<b>Çevrim süresi</b>	<b>Alternatif Çözüm Sayısı</b>	<b>Uğurdağ vd. (1997) çevrim süresi</b>
Heskiaoff_28_4	0.0035	0.0016	0.0012	283.235	250000	258
Heskiaoff_28_5	0.0042	0.0018	0.0014	236.888	250000	210
Sawyer_30_5	0.0133	0.0072	0.0040	75.149	206000	66
Sawyer_30_8	0.0182	0.0097	0.0050	54.884	206000	43
Sawyer_30_13	0.0260	0.0149	0.0055	38.422	206000	28

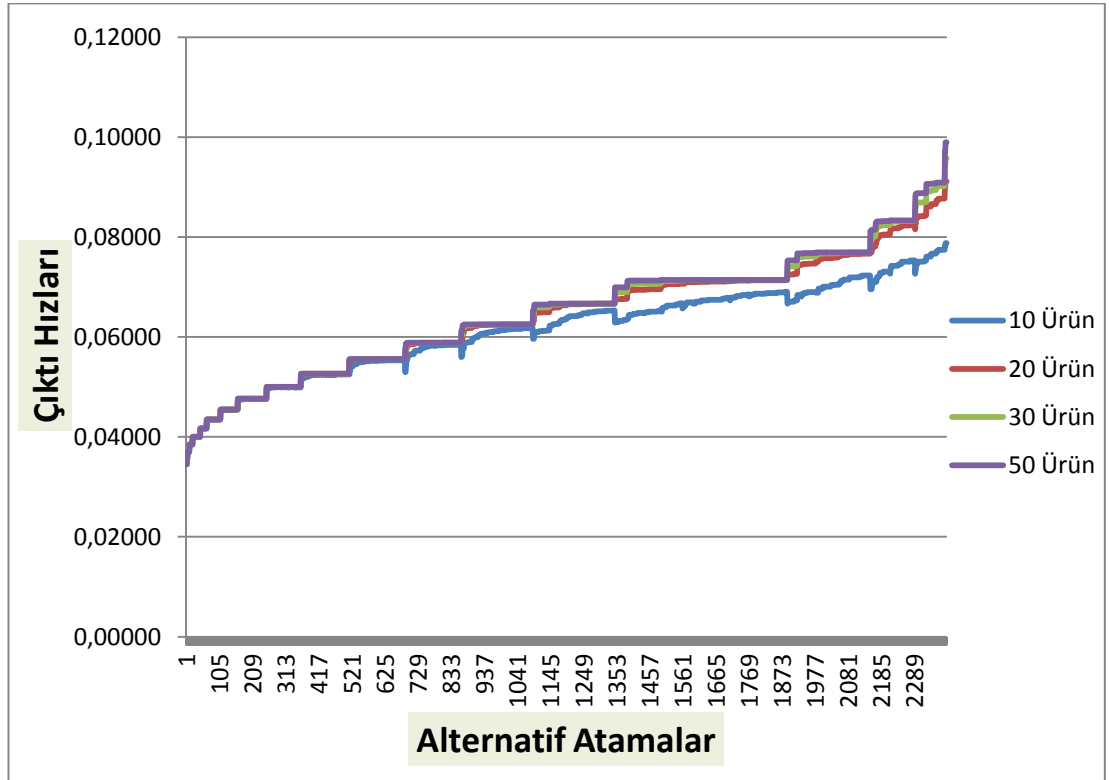
Heskiaoff deney setleri hem öncelikle derinlik hem de çok noktalı arama stratejileri kullanılarak çözülmüştür. Heskiaoff deney setleri için öncelikle derinlik arama stratejisi ile yapılan deneylerde, görev sayısının yüksek olması ve atama kombinasyonu sayısının çok olmasından dolayı, bütün atama alternatifleri değerlendirilememiştir. Şekil 8.18 ve 8.19'da dallanmaların bir bölgede sıkıştığı ve iyileşmenin sağlanamadığı görülmektedir.

Şekil 8.20 ve 8.21'de de görüldüğü üzere Heskiaoff deney setlerinin çözüm sonuçlarında, çok noktalı arama stratejisinde uygun çözüm alanının değişik noktalarında da aramalar yapılması sonucunda daha iyi performans değerleri veren çözümler bulunmuştur.

#### 8.4. Montaj Hattının Performansının Değişik Ürün Sayılarına Göre Test Edilmesi

Yapılan deneysel çalışmalarda montaj hatlarında bulunan ürün sayısının sabit ve 50 adet olduğu kabul edilmiştir. Kapalı kuyruk ağlarının özelliğinden dolayı ele aldığımız montaj hatlarında bulunan ürün sayısı sabit olduğu kabulü yapılmıştır. Ancak, bütün montaj hatlarında ürün sayısının 50 adet olması şart değildir. Önemli olan montaj hattının performans değerlerini maksimum yapacak ürün sayısının belirlenmesidir.

Montaj hattındaki müşteri sayılarının sistem çıktı hızına etkisinin test edilmesi amacıyla Jackson (1956) montaj hattının 5 istasyonlu durumu için 10, 20, 30 ve 50 ürün bulunduğu durumlarda algoritma çalıştırılmıştır. Bu dört durum için çıktı hızlarının değişim grafiği şekil 8.26'da verilmiştir.



Şekil 8.26 Değişik ürün sayılarına göre Jackson (1956) montaj hattı 5 istasyonlu durumu için çıktı hızlarının değişim grafiği

Şekil 8.26'daki grafikte görüldüğü üzere sistemdeki müşteri sayısı 10 adetten 20 adete çıkarıldığında sistem çıktı hızında % 15,7 artış olmaktadır. Ancak, 20 ürün adetinden 30 ürün adetine artırıldığında % 5 artış olmaktadır. 30 müşteri adetinden 50 müşteri adetine çıkarıldığında ise % 3,3 artış olmaktadır. Yapılan deneyler sonucunda sistemde bulunan ürün sayısının sürekli artırıldığında sistem çıktı hızını belirli bir değerden sonra fazla etkilemediği gözlenmiştir.

## 9. SONUÇLAR VE ÖNERİLER

Bu çalışmada tek modelli düz stokastik montaj hatlarının sabit istasyon sayısı kısıdı altında dengelenmesi çalışılmıştır. Montaj hattında istasyon sürelerinin üssel dağılıma uyduğu kabul edilmiştir. Üssel dağılıma uygunluğunun test edilmesi amacıyla bir simülasyon modeli kurulmuştur. İstasyon sürelerinin normal dağılıma ve üssel dağılıma uyduğu durumlar karşılaştırılmıştır. Karşılaştırmalar sonucunda istasyon sürelerinin üssel dağılıma uyduğu durumlar ile normal dağılıma uyduğu durumların benzer eğilimde olduğu bulunmuştur. Diğer atamalara göre bir atama değeri, normal dağılım için iyi bir çıktı hızı veya istasyon doluluk oranı veriyor ise üssel dağılımda da iyi sonuçlar vermektedir.

Montaj hattını dengelenmek için yeni bir metodoloji geliştirilmiştir. Bu metodolojide, montaj hattında istasyonlara görev atanmasında kısıt programlamadan yararlanılmıştır. Kısıt programlama ile belirlenen atama kombinasyonlarının performansı kapalı kuyruk ağı modeli ile ölçülmüştür ve en iyi performans değerine sahip olan atama belirlenmiştir.

Öncelikli olarak bütün atama kombinasyonlarının elde edilmesi ve performansının ölçülmesi hedeflenmiştir. Bunun için arama stratejisi olarak öncelikle derinlik arama stratejisi kullanılmıştır. Yapılan deneyler sonucunda büyük boyutlu problemlerde bu arama stratejisinin yetersiz kaldığı ve bilgisayar hafızasının yetersizliğinden dolayı bütün sonuçların elde edilemediği görülmüştür. Bu problemi aşmak amacıyla ve çözüm uzayının farklı bölgelerinde arama yapmak için çok noktalı arama stratejisi kullanılmıştır.

Önerilen metodolojide performans kriteri olarak sistemin çıktı hızı alınmıştır. Ayrıca, kapalı kuyruk ağı modeli ile her bir istasyonun doluluk oranı, bir ürünün istasyonda harcadığı süre, bir istasyondaki ürün sayısı hesaplanmış ve sonuçları her bir deney için 8. bölümde verilmiştir. Bu performans kriterleri dışında, kapalı kuyruk ağı modeli ile sistemde geçen süre, toplam bekleme süresi, istasyonların her biri için bekleyen ürün sayıları gibi sistem performans değerleri de hesaplanabilmektedir.

Geliştirilen metodolojiyi test etmek amacıyla literatürde bulunan örnek montaj hattı modelleri kullanılmıştır. Bu modellerden 14 adet deney seti için öncelikle derinlik arama stratejisi ve 5 adet deney seti için çoklu arama stratejisi kullanılarak metodoloji test edilmiştir. Deneyler, ILOG Opl CP Optimizer 6.3 programı 3,00 Ghz işlemci ve 4 Gb RAM belleğe sahip bir bilgisayarda büyük problemler için yaklaşık 14 saat çalıştırılarak yapılmıştır. Test sonuçları literatürdeki aynı örneklerden elde edilen sonuçlar ile karşılaştırılmıştır. Bu karşılaştırma sonuçlarına göre:

- Küçük boyutlu problemlerde metodolojinin etkin bir şekilde çalıştığı görülmüştür. Küçük boyutlu problemlerde bütün sonuçların değerlendirilmiş olması metodolojinin etkinliğini artırmaktadır.
- Öncelikle derinlik arama stratejisi kullanılarak yapılan 14 deneyin, 11'i için optimum çözüme ulaşılmıştır, 1 deney için optimum çözüme yaklaşılmış, 2 deney için optimum çözüme yaklaşılamamıştır.(Çizelge 8.2)
- Çok noktalı arama stratejisi ile yapılan 5 deneyde optimum çözüme yaklaşılmıştır. Ancak, arama stratejisinin bütün çözüm uzayını değerlendirme imkanı vermemesinden dolayı optimum çözüme ulaşamamıştır.
- Her iki arama stratejisi için alternatiflerin değerlendirmesi sonucunda en iyi performans değerine sahip birden fazla çözüm bulunabilmektedir. Böylelikle montaj hattının kurulumunda planlamacılara alternatif kurulum imkanı vermektedir.
- Deney sürelerinin uzun sürmesi metodolojinin kullanımı açısından bir yetersizliği olarak karşımıza çıkmaktadır. Bu problemin aşılması için metodolojinin çeşitli algoritmalarla desteklenmesi gerekmektedir.
- Literatür araştırmasında TİP-2 stokastik montaj hattı dengeleme problemleri ile ilgili bir çalışmaya rastlanmamıştır. Bu çalışmada Stokastik TİP-2 montaj hattının çalışılmış olması literatüre katkı sağlamaktadır.
- Ayrıca, çalışmamızda kullandığımız kapalı kuyruk ağı modellenli montaj hattı dengeleme aşamasında kullanılmış olması çalışmaya özgünlük katmaktadır.
- Bunlara ek olarak, görev atamalarında Kısıt Programlama modelinin kullanılması çalışmaya özgünlük katmaktadır.

- Tip-2 montaj hattı probleminin çözümünde literatürde yapılan çalışmalarda, tip-1 problemlerinin çözümünde kullanılan yöntemlerin tip-2 problemini çözecek şekilde değiştirildiği ve böylelikle çözümlerin elde edildiği görülmektedir. Genel olarak tip-1 problemlerinin çözüm algoritması kullanılarak, alt sınır değeri elde edilip çevrim süresi birer artırılarak veya üst sınır değeri belirlenip çevrim süresi birer azaltılarak istenen istasyon sayısına ulaşıldığında algoritma durdurulması sonucu çözüme ulaşılmıştır. Literatürde tip-2 problemlerinin çözümü, tip-1 problemlerini çözen algoritmalar ile iterasyonel olarak çözülmesi sonucu elde edilmiştir. Önerilen metodoloji, direk olarak tip-2 problemlerini çözecek şekilde geliştirilmesi açısından literatüre ayrıca katkı sağlamaktadır.

İleriki çalışmalarda aşağıda verilen konular ele alınabilir.

- Montaj hattının sürelerinin normal dağılıma uyduğu durumlar için kapalı kuyruk ağı modeli kullanılabilir.
- Montaj hattının sürelerinin farklı bir dağılıma (normal ve üssel dağılım dışında) uyduğu durumlar göz önüne alınarak yeniden modellenebilir.
- Kısıt programlama algoritması için daha etkin bir arama stratejisinin belirlenmesi ve modellenmesi.
- Kısıt programlama modeli ile birlikte bir sezgisel arama algoritması kullanılması.
- Modelin Tip-1 montaj hattına uygulanacak şekilde değiştirilmesi.
- Modelde kapalı kuyruk ağı yerine açık kuyruk ağlarının kullanılmasının araştırılması.

## KAYNAKLAR

- Ađpak, K., & Gökçen, H. A chance-constrained approach to stochastic line balancing. *European Journal Of Operational Research* , 180(3): 1098–1115, 2007.
- Ammar , M. H. (1980) Modelling and analysis of unreliable manufacturing assembly networks with finite storages. Massachusetts Institute of Technology, Massachusetts
- Apt, K. R. (2003). *Principles of Constraint Programming*. Amsterdam, The Netherlands: Cambridge University Press.
- Ayazi, S., Hajizadehb, A. E., Nooshabadic, M. R., Jalaiea, H., & Moradi, Y. Multi-objective assembly line balancing using genetic algorithm. *International Journal of Industrial Engineering Computations* , 2: 863–872, 2011.
- Azaron, A., Katagari, H., Kato, K., Sakawa, M., Modelling complex assemblies as a queueing network for lead time control, *European Journal of Operational Research* 174: 150–168, 2006.
- Azaron, A. ve Kianfar, F., A multi-objective lead time control problem in multistage assembly systems using an interactive method. *Applied Mathematics and Computation* 176: 609–620, 2006.
- Bard, Y. (1979). Some Extensions to Multiclass Queueing Network Analysis. in *Performance of Computer systems*, M. Arato, A. Butrimcnko, and E. Gelenbe, eds. Amsterdam: Holland.
- Baskett, F., Chandy, K. M., Muntz, R. R., & Palacios-Gomez, F. Open, closed and mixed networks for queues with different classes of customers. *Journal of A.C.M* , 22(2): 248-260, 1975.
- Başkak, M. (1991). *Montaj Hatlarının Dengelenmesinde Çok Amaçlı Bir Yaklaşım*. İstanbul Teknik Üniversitesi, İstanbul.
- Baybars, İ. A Survey of Exact Algoritihms for the Simple Assembly Line Balancing Problem. *Management Science* , 32(8): 909-932, 1986.
- Baybars, İ. An Efficient Heuristic Method for the Simple Assembly Line Balancing Problem. *Int. Jor. of Pro. Res* , 24(1): 149-166, 1986.

- Becker, C., & Scholl, A. A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research*, 168: 694–715, 2006.
- Boysen, N., Fliedner, M., & Scholl, A. A classification of assembly line balancing problems. *European Journal of Operational Research*, 183: 674–693, 2007.
- Bryton, B. (1954). *Balancing of a Continuous Production Line*, Yüksek Lisans tezi. Northwestern University, Northwestern.
- Buzen, J. Computational algorithms for closed queueing networks with exponential servers. *Communications of A.C.M.* , 16 (9): 527-531, 1973.
- Carnahan, B. J., Norman, B. A., & Redfern, M. S. Incorporating Physical Demand Criteria Into Assembly Line Balancing. *IIE Transactions* , 33(10): 875–887, 2001.
- Chandy, K., Herzog, U., & Woo, L. Parametric analysis of queueing networks. *IBM Journal of Research and Development* , 19 (1): 36-42, 1975.
- Duenyas, I., Hopp, W. J., Estimating the throughput of an exponential CONWIP assembly system. *Queueing Systems* 14: 135-157, 1993.
- Erel, E., & Sarin, S. A survey of the assembly line balancing procedures. *Production Planning & Control* , 9(5): 414- 434, 1998.
- Gershwin, S. B., *Assembly/Disassembly Systems: An efficient Decomposition Algorithm for Tree-Structured Networks*, MIT Laboratory for Information and Decision Systems Report LIDS-P- 1579, July, 1986
- Gordon, W. J., & Newell, G. F. Cyclic Queueing Systems with Restricted Length Queues. *Operations Research* , 15 (2): 266-277, 1967.
- Gordon, W., & Newell, G. Closed queueing systems with exponential servers. *Operations Research* , 15 (2): 254-265, 1967.
- Govil, M. K., Fu, M. C., Queueing theory in manufacturing. A survey. *Journal of Manufacturing Systems*, 18(3): 214-240, 1999.
- Gökçen, H., & Baykoç, Ö. A new line remedial policy for the paced lines with stochastic task times. *International Journal of Production Economics*, 58(2): 191-197, 1999.



- Hackman, S., Magazine, M. J., & Wee, T. Fast, effective algorithms for simple assembly line balancing problems. *Operations Research*, 37(6): 916-924, 1989.
- Harrison, J. M., Assembly like queues, *Journal of Applied Probability*, 10: 354-367, 1973.
- Helgeson, W., & Birnie, D. "Assembly Line Balancing Using The Ranked Positional Weight Technique". *Journal of Industrial Engineering* , 12(6): 384-389, 1961.
- Henig, M. I. Extensions of the dynamic programming method in the deterministic and stochastic assembly line balancing problems. *Computers and Operations Research* , 13 (4): 443-449, 1986.
- Hildebrant, R. Scheduling Flexible Manufacturing System Using Mean Value Analysis. *Proc. of IEEE Cont: on Decision and Control*, 701-706. Albuquerque, NM, 1980.
- IBM. ILOG CP Optimizer User's Manual. 01 30, 2012 tarihinde [http://publib.boulder.ibm.com/infocenter/cosinfoc/v12r3/index.jsp?topic=%2Flog.odms.cpo.help%2FContent%2FOptimization%2FDocumentation%2FOptimization\\_Studio%2F\\_pubskel%2Fusrcoptimizer2773.html](http://publib.boulder.ibm.com/infocenter/cosinfoc/v12r3/index.jsp?topic=%2Flog.odms.cpo.help%2FContent%2FOptimization%2FDocumentation%2FOptimization_Studio%2F_pubskel%2Fusrcoptimizer2773.html) adresinden alındı
- Jackman, J., Johnson, E., The Role of Queueing Network Models in Performance Evaluation of Manufacturing Systems. *J. Opl Res. Soc.*, 44(8): 797-807, 1993.
- Jackson, J. Jobshop-like queueing systems. *Management Science* , 10(1): 131-142, 1963.
- Kao, E. A preference order dynamic program for stochastic assembly line balancing. *Management Science* , 22(10): 1097-1104, 1976.
- Kao, E.P.C. Computational Experience with a Stochastic Assembly Line Balancing Algorithm. *Computers and Operations Research*, 6: 79-86, 1979.
- Klein, R., & Scholl, A. Maximizing the production rate in simple assembly line balancing -- A branch and bound procedure. *European Journal of Operational Research* , 91(2): 367-385, 1996.
- Kottas, J., & Lau, H. A cost oriented approach of stochastic line balancing. *AIIE Transactions* , 5(2): 164-171, 1973.
- Kottas, J.F., & Lau, H. A stochastic line balancing procedure. *International Journal of Production Research* , 19(2): 177-193, 1981.

- Kottas, J.F., & Lau, H. A total operating cost model for paced lines with stochastic task times. *AIIE Transactions* , 8(2): 234-240, 1976.
- Latouche G, Neuts MF, Efficient algorithm solutions to exponential tandem queue with blocking. *SIAM J Algebra Disc Method* 1(1): 93–106, 1980.
- Lazaro A. R., ve perez, C. J. L., Dynamic analysis of an automobile assembly line considering starving and blocking. *Robotics and Computer-Integrated Manufacturing* 25: 271–279, 2009.
- Lipper, E. H., Sengupta B., Assembly-like queues with finite capacity: bounds, asymptotics and approximations. *Queueing systems: Theory and Applications* 1: 67, 1986
- Little, J. D. A proof for the queueing formula  $L=\lambda*W$ . *Operations Research* , 9(3): 383-387, 1961.
- Liu, S., Ong, H., & Huang, H. A Bi-directional Heuristic for Stochastic Assembly Line Balancing Problems. *Int J Adv Manuf Technol* , 25: 71-77, 2005.
- Manitz, M., Queueing-model based analysis of assembly lines with finite buffers and general service times, *Computers & Operations Research* 35, 2520 – 2536, 2007.
- Mansoor, E. M. Assembly Line Balancing-An Improvement on the Ranked Positional Weight Technique. *J. Industrial Engineering* , 15; 73-78, 1964.
- Mansoor, E.M., & Yadin, M. (1971). On the Problem of Assembly Line Balancing. B. Avi-Itzhak içinde, *Developments in Operations Research* (s. 361-375). New York: Gordon and Breach.
- Modie, C. L., Young, H. H., A heuristic method of assembly line balancing for assumptions of constant or variable work element times. *Journal of Industrial Engineering*, 16: 23-29, 1965.
- Nkasu, M. M., & Leung, K. H. A stochastic approach to assembly line balancing. *International Journal of Production Research* , 33(4): 975–991, 1995.
- Nourmohammadia, A., & Zandieh, M. Assembly line balancing by a new multi-objective differential evolution algorithm based on TOPSIS. *International Journal of Production Research* , 49(10): 2833–2855, 2011.

- Pastor, R., & Ferrer, L. An improved mathematical program to solve the simple assembly line balancing problem. *International Journal of Production Research*, 47(11): 2943–2959, 2009.
- Pınarbaşı, M. (2010). *Esnek İmalat Sistemlerinin Tasarım Süreci ve Performans Analizinde Kuyruk Ağı Modelleri*. Kırıkkale Üniversitesi, Kırıkkale
- Rao, D. (1971). *Single and Mixed Model Assembly Line Balancing Methods for Both Deterministic and Normally Distributed Work Element Times*. Yüksek Lisans Tezi, Oregon University, Oregon.
- Raouf, A., & Tsui, C. A new method for assembly line balancing having stochastic work elements. *Computers and Industrial Engineering*, 6(2): 131-148, 1982.
- Reiser, M., & Lavenberg, S. Mean value analysis for closed multi-chain queueing networks. *Journal of A.C.M.*, 27 (2): 313-322, 1980.
- Rekiek, B., De Lit, P., Pellichero, F., Falkenauer, E., & Delchambre, A. (1999). Applying the Equal Piles Problem to Balance Assembly Lines. *Proceedings of the IEEE International Symposium on Assembly and Task Planning*.
- Salveson, M. E. The Assembly Line Balancing Problem. *Journal of Industrial Engineering*, 6(3), 6: 18-25, 1955.
- Sarin, S. C., Erel, E., & Dar-El, M. A methodology solving single-model, stochastic assembly line balancing problem. *International Journal of Management Science*, 27(5): 525-535, 1999.
- Schmidt, R. An approximate mva algorithm for exponential, class dependent multiple servers. *Performance Evaluation*, 29(4): 245-254, 1979.
- Scholl, A., & Vob, S. Simple assembly line balancing-heuristic approaches. *Journal of Heuristics*, 2(3): 217-244, 1996.
- Shin, D. An efficient heuristic for solving stochastic assembly line balancing problems. *Computers and Industrial Engineering*, 18(3): 285-295, 1990.
- Silverman, F., & Carter, J. A cost-based methodology for stochastic line balancing with intermittent line stoppages. *Management Science*, 32(4): 455-463, 1986.
- Suresh, G., & Sahu, S. Stochastic assembly line balancing using simulated annealing. *International Journal of Production Research*, 32: 1801-1810, 1994.

- Ugurdag, H., Rachamadugu, R., & Papachristou, C. A. Designing paced assembly lines with fixed number of stations. *European Journal of Operational Research* , 102: 488-501, 1997.
- Wang, H. 1997. Approximate MVA algorithms for solving queueing network models. Yüksek Lisans Tezi University of Toronto.
- Wee, T. S., & Magazine, M. An Efficient Branch and Bound Algorithm for Assembly Line Balancing-Part 2: Maximize the Production Rate. Working Paper, University of Waterloo, 1981.
- Wei, N., & Chao, M. A solution procedure for type E simple assembly line balancing problem. *Computers & Industrial Engineering* , 61(3): 824–830, 2011.
- Yüzükırmızı, M. Performance evaluation of closed queueing networks with limited capacities. *Turkish J. Eng. Env. Sci.*, 30 :269-283, 2006.

## EKLER

### EK-1 ILOG OPL 6.3 CP OPTIMIZER KODU

Jackson Montaj hattının 5 istasyon sayısına sahip olduğu durum için örnek ILOG Opl 6.3 CP Optimizer kodu aşağıda vermiştir.

```
/******  
 * OPL 6.3 Model  
 * Author: HACI MEHMET  
 * Creation Date: 31.Eki.2011 at 10:05:27  
*****/  
using CP;  
int maxgorsay=11;  
range gorev = 1..maxgorsay;  
int maxmustsay=50;  
int maxistsay=5;  
float toppi;  
range istasyon = 1..maxistsay;  
range mustsay=1..maxmustsay;  
int gorsur [gorev] = [6, 2, 4, 7, 1, 2, 3, 6, 5, 5, 4];  
float mu[istasyon];  
float pi[istasyon][0..maxmustsay][0..maxmustsay];  
int vr[istasyon];  
float T[istasyon][mustsay];  
float Tg;  
int istsur[istasyon];  
float topsur[mustsay];  
float lambda[mustsay];  
float lambdaobj;  
  
dvar int+ istno[gorev];  
  
subject to  
  
{  
  
istno[1]>=1;  
istno[2]>=1;  
istno[3]>=2;  
istno[4]>=2;  
istno[5]>=1;  
istno[6]>=2;  
istno[7]>=3;  
istno[8]>=2;  
istno[9]>=3;  
istno[10]>=3;  
istno[11]>=5;  
  
istno[1]<=1;
```

```

istno[2]<=3;
istno[3]<=4;
istno[4]<=3;
istno[5]<=4;
istno[6]<=4;
istno[7]<=4;
istno[8]<=4;
istno[9]<=5;
istno[10]<=5;
istno[11]<=5;

istno[1]<=istno[2];
istno[1]<=istno[4];
istno[1]<=istno[5];
istno[2]<=istno[6];
istno[3]<=istno[7];
istno[4]<=istno[7];
istno[5]<=istno[7];
istno[6]<=istno[8];
istno[7]<=istno[9];
istno[8]<=istno[10];
istno[9]<=istno[11];
istno[10]<=istno[11];

    forall (j in istasyon){
        count(all(i in gorev) istno[i], j) >= 1;
    }
}

execute {
    for (var i in istasyon){
        istsur[i]=0
    };
    for (i in thisOplModel.istasyon){
        for(var j in thisOplModel.gorev)
            if (thisOplModel.istno[j]==i){
                istsur[i]= istsur[i]+gorsur[j]
            };
    };
}

execute basdegatamasi {
    for (var i in thisOplModel.istasyon){
        mu[i]= 1/istsur[i];
        pi[i][0][0]=1;
        vr[i]=1;
    };
};

execute kuyrukalgoritmasi {
    var tom=thisOplModel;
    for (var k in mustsay){
        Tg=0;
        for (var i in istasyon){
            for(var j=1; j<=k; j++){
                Tg+=Tg + (j/mu[i])*pi[i][j-1][k-1];
            };
            T[i][k]=Tg;
        };
    };
};

```

```

        for (i in istasyon){
            topsur[k]=T[i][k]*vr[i]
        };
        lambda[k]=k/topsur[k];
        lambdaobj=lambda[k];

        for (i in istasyon){
            toppi=0;
            for(j=1; j<=k; j++){
                pi[i][j][k]=(lambda[k]/mu[i])*pi[i][j-1][k-1]*vr[i];
                toppi=toppi+pi[i][j][k]
            };
            pi[i][0][k]=1-toppi;
        };
    };
};

main {
    var status=1;
    thisOplModel.generate();
    var ofile = new IloOplOutputFile("istasyonkombinasyonu.txt");
    cp.startNewSearch();
    while (cp.next()) {
        ofile.writeln(" ");
        ofile.write(status);
        for (var i in thisOplModel.gorev){
            ofile.write(" ",thisOplModel.istno[i]);
        }
        ofile.write(" ",status-1," ",thisOplModel.lambdaobj," ");
        status ++;
        thisOplModel.postProcess();
    }

    ofile.close();

}

```

## EK-2 JACKSON MONTAJ HATTI ÇÖZÜM SONUCU

Çizelge 2.1 Jackson montaj hattı 3 istasyonlu durum için elde edilen çözümler.

Sıra No	Görev-istasyon atamaları											Çıktı Hızı
	1	2	3	4	5	6	7	8	9	10	11	
1	1	1	1	1	1	1	2	2	2	2	3	0,04348
2	1	1	2	1	1	1	2	2	2	2	3	0,04167
3	1	1	1	2	1	1	2	2	2	2	3	0,03846
4	1	1	2	2	1	1	2	2	2	2	3	0,03226
5	1	1	1	1	2	1	2	2	2	2	3	0,04542
6	1	1	2	1	2	1	2	2	2	2	3	0,04000
7	1	1	1	2	2	1	2	2	2	2	3	0,03704
8	1	1	2	2	2	1	2	2	2	2	3	0,03125
9	1	1	1	1	1	2	2	2	2	2	3	0,04668
10	1	1	2	1	1	2	2	2	2	2	3	0,03846
11	1	1	1	2	1	2	2	2	2	2	3	0,03571
12	1	1	2	2	1	2	2	2	2	2	3	0,03030
13	1	1	1	1	2	2	2	2	2	2	3	0,04542
14	1	1	1	2	2	2	2	2	2	2	3	0,03448
15	1	1	2	1	2	2	2	2	2	2	3	0,03704
16	1	1	2	2	2	2	2	2	2	2	3	0,02941
17	1	2	1	1	1	2	2	2	2	2	3	0,04348
18	1	2	2	1	1	2	2	2	2	2	3	0,03571
19	1	2	1	2	1	2	2	2	2	2	3	0,03333
20	1	2	2	2	1	2	2	2	2	2	3	0,02857
21	1	2	1	1	2	2	2	2	2	2	3	0,04167
22	1	2	2	1	2	2	2	2	2	2	3	0,03448
23	1	2	1	2	2	2	2	2	2	2	3	0,03226
24	1	2	2	2	2	2	2	2	2	2	3	0,02778
25	1	1	1	1	1	1	2	2	3	2	3	0,04348
26	1	1	1	1	2	1	2	2	3	2	3	0,04545
27	1	1	1	1	1	2	2	2	3	2	3	0,04762
28	1	1	1	1	2	2	2	2	3	2	3	0,05000
29	1	2	1	1	1	2	2	2	3	2	3	0,05242
30	1	2	1	1	2	2	2	2	3	2	3	0,05242
31	1	1	1	1	1	1	3	2	3	2	3	0,04348
32	1	1	1	1	2	1	3	2	3	2	3	0,04545
33	1	1	1	1	3	1	3	2	3	2	3	0,04545
34	1	1	1	1	1	2	3	2	3	2	3	0,04762
35	1	1	1	1	2	2	3	2	3	2	3	0,05000



**Çizelge 2.1 (Devam)** Jackson montaj hattı 3 istasyonlu durum için elde edilen çözümler.

Sıra No	Görev-istasyon atamaları											Çıktı Hızı
	1	2	3	4	5	6	7	8	9	10	11	
36	1	1	1	1	3	2	3	2	3	2	3	0,05000
37	1	2	1	1	1	2	3	2	3	2	3	0,05263
38	1	2	1	1	2	2	3	2	3	2	3	0,05553
39	1	2	1	1	3	2	3	2	3	2	3	0,05555
40	1	1	1	2	1	1	2	2	3	2	3	0,04762
41	1	1	1	2	2	1	2	2	3	2	3	0,04545
42	1	1	1	2	1	2	2	2	3	2	3	0,04348
43	1	1	1	2	2	2	2	2	3	2	3	0,04167
44	1	2	1	2	1	2	2	2	3	2	3	0,04000
45	1	2	1	2	2	2	2	2	3	2	3	0,03846
46	1	1	1	2	1	1	3	2	3	2	3	0,05553
47	1	1	1	2	2	1	3	2	3	2	3	0,05263
48	1	1	1	2	3	1	3	2	3	2	3	0,05555
49	1	1	1	2	1	2	3	2	3	2	3	0,05000
50	1	1	1	2	2	2	3	2	3	2	3	0,04762
51	1	1	1	2	3	2	3	2	3	2	3	0,05000
52	1	2	1	2	1	2	3	2	3	2	3	0,04545
53	1	2	1	2	2	2	3	2	3	2	3	0,04348
54	1	2	1	2	3	2	3	2	3	2	3	0,04545
55	1	1	2	1	1	1	2	2	3	2	3	0,05242
56	1	1	2	1	2	1	2	2	3	2	3	0,05000
57	1	1	2	1	1	2	2	2	3	2	3	0,04762
58	1	1	2	1	2	2	2	2	3	2	3	0,04545
59	1	2	2	1	1	2	2	2	3	2	3	0,04348
60	1	2	2	1	2	2	2	2	3	2	3	0,04167
61	1	1	2	1	1	1	3	2	3	2	3	0,05553
62	1	1	2	1	2	1	3	2	3	2	3	0,05761
63	1	1	2	1	3	1	3	2	3	2	3	0,05860
64	1	1	2	1	1	2	3	2	3	2	3	0,05553
65	1	1	2	1	2	2	3	2	3	2	3	0,05263
66	1	1	2	1	3	2	3	2	3	2	3	0,05555
67	1	2	2	1	1	2	3	2	3	2	3	0,05000
68	1	2	2	1	2	2	3	2	3	2	3	0,04762
69	1	2	2	1	3	2	3	2	3	2	3	0,05000
70	1	1	2	2	1	1	2	2	3	2	3	0,03846

**Çizelge 2.1 (Devam)** Jackson montaj hattı 3 istasyonlu durum için elde edilen çözümler.

Sıra No	Görev-istasyon atamaları											Çıktı Hızı
	1	2	3	4	5	6	7	8	9	10	11	
71	1	1	2	2	2	1	2	2	3	2	3	0,03704
72	1	1	2	2	1	2	2	2	3	2	3	0,03571
73	1	1	2	2	2	2	2	2	3	2	3	0,03448
74	1	2	2	2	1	2	2	2	3	2	3	0,03333
75	1	2	2	2	2	2	2	2	3	2	3	0,03226
76	1	1	2	2	1	1	3	2	3	2	3	0,04348
77	1	1	2	2	2	1	3	2	3	2	3	0,04167
78	1	1	2	2	3	1	3	2	3	2	3	0,04348
79	1	1	2	2	1	2	3	2	3	2	3	0,04000
80	1	1	2	2	2	2	3	2	3	2	3	0,03846
81	1	1	2	2	3	2	3	2	3	2	3	0,04000
82	1	2	2	2	1	2	3	2	3	2	3	0,03704
83	1	2	2	2	2	2	3	2	3	2	3	0,03571
84	1	2	2	2	3	2	3	2	3	2	3	0,03704
85	1	1	1	1	1	1	2	2	2	3	3	0,04348
86	1	1	1	1	1	2	2	2	2	3	3	0,04762
87	1	2	1	1	1	2	2	2	2	3	3	0,05242
88	1	1	1	1	1	1	2	3	2	3	3	0,04348
89	1	1	1	1	1	2	2	3	2	3	3	0,04762
90	1	2	1	1	1	2	2	3	2	3	3	0,05263
91	1	1	1	2	1	1	2	2	2	3	3	0,04762
92	1	1	1	2	1	1	2	3	2	3	3	0,06191
93	1	1	1	2	1	2	2	2	2	3	3	0,04348
94	1	2	1	2	1	2	2	2	2	3	3	0,04000
95	1	1	1	2	1	2	2	3	2	3	3	0,05879
96	1	2	1	2	1	2	2	3	2	3	3	0,05263
97	1	1	1	1	2	1	2	2	2	3	3	0,04545
98	1	1	1	1	2	2	2	2	2	3	3	0,05000
99	1	1	1	2	2	1	2	2	2	3	3	0,04545
100	1	1	1	2	2	2	2	2	2	3	3	0,04167
101	1	1	1	1	2	1	2	3	2	3	3	0,04545
102	1	1	1	1	2	2	2	3	2	3	3	0,05000
103	1	1	1	2	2	1	2	3	2	3	3	0,06191
104	1	1	1	2	2	2	2	3	2	3	3	0,05555
105	1	2	1	1	2	2	2	2	2	3	3	0,05242

**Çizelge 2.1 (Devam)** Jackson montaj hattı 3 istasyonlu durum için elde edilen çözümler.

Sıra No	Görev-istasyon atamaları											Çıktı Hızı
	1	2	3	4	5	6	7	8	9	10	11	
106	1	2	1	1	2	2	2	3	2	3	3	0,05555
107	1	2	1	2	2	2	2	2	2	3	3	0,03846
108	1	2	1	2	2	2	2	3	2	3	3	0,05000
109	1	1	1	1	1	1	2	2	3	3	3	0,04348
110	1	1	1	1	1	1	2	3	3	3	3	0,04347
111	1	1	1	1	2	1	2	2	3	3	3	0,04545
112	1	1	1	1	2	1	2	3	3	3	3	0,04542
113	1	1	1	1	1	2	2	2	3	3	3	0,04762
114	1	1	1	1	2	2	2	2	3	3	3	0,05000
115	1	1	1	1	1	2	2	3	3	3	3	0,04740
116	1	1	1	1	2	2	2	3	3	3	3	0,04901
117	1	2	1	1	1	2	2	2	3	3	3	0,05263
118	1	2	1	1	2	2	2	2	3	3	3	0,05556
119	1	2	1	1	1	2	2	3	3	3	3	0,04979
120	1	2	1	1	2	2	2	3	3	3	3	0,04997
121	1	1	1	2	1	1	2	2	3	3	3	0,06108
122	1	1	1	2	2	1	2	2	3	3	3	0,05879
123	1	1	1	2	1	1	2	3	3	3	3	0,05000
124	1	1	1	2	2	1	2	3	3	3	3	0,05000
125	1	1	1	2	1	2	2	2	3	3	3	0,05556
126	1	1	1	2	2	2	2	2	3	3	3	0,05263
127	1	1	1	2	1	2	2	3	3	3	3	0,05000
128	1	1	1	2	2	2	2	3	3	3	3	0,05000
129	1	2	1	2	1	2	2	2	3	3	3	0,05000
130	1	2	1	2	1	2	2	3	3	3	3	0,05000
131	1	2	1	2	2	2	2	2	3	3	3	0,04762
132	1	2	1	2	2	2	2	3	3	3	3	0,05000
133	1	1	1	1	1	1	3	2	3	3	3	0,04348
134	1	1	1	1	2	1	3	2	3	3	3	0,04545
135	1	1	1	1	3	1	3	2	3	3	3	0,04545
136	1	1	1	2	1	1	3	2	3	3	3	0,05860
137	1	1	1	2	2	1	3	2	3	3	3	0,05879
138	1	1	1	2	3	1	3	2	3	3	3	0,05555
139	1	1	1	1	1	2	3	2	3	3	3	0,04762
140	1	1	1	1	2	2	3	2	3	3	3	0,05000

**Çizelge 2.1 (Devam)** Jackson montaj hattı 3 istasyonlu durum için elde edilen çözümler.

Sıra No	Görev-istasyon atamaları											Çıktı Hızı
	1	2	3	4	5	6	7	8	9	10	11	
141	1	1	1	1	3	2	3	2	3	3	3	0,04997
142	1	2	1	1	1	2	3	2	3	3	3	0,05261
143	1	2	1	1	2	2	3	2	3	3	3	0,05535
144	1	2	1	1	3	2	3	2	3	3	3	0,05444
145	1	1	1	2	1	2	3	2	3	3	3	0,05879
146	1	1	1	2	2	2	3	2	3	3	3	0,05860
147	1	1	1	2	3	2	3	2	3	3	3	0,05555
148	1	2	1	2	1	2	3	2	3	3	3	0,05761
149	1	2	1	2	2	2	3	2	3	3	3	0,05535
150	1	2	1	2	3	2	3	2	3	3	3	0,05535
151	1	1	1	1	2	1	3	3	3	3	3	0,04325
152	1	1	1	2	1	1	3	3	3	3	3	0,04348
153	1	1	1	2	2	1	3	3	3	3	3	0,04348
154	1	1	1	2	3	1	3	3	3	3	3	0,04167
155	1	1	1	1	1	2	3	3	3	3	3	0,04344
156	1	1	1	1	2	2	3	3	3	3	3	0,04347
157	1	1	1	1	3	2	3	3	3	3	3	0,04167
158	1	1	1	2	1	2	3	3	3	3	3	0,04348
159	1	1	1	2	2	2	3	3	3	3	3	0,04348
160	1	1	1	2	3	2	3	3	3	3	3	0,04167
161	1	2	1	1	1	2	3	3	3	3	3	0,04348
162	1	2	1	1	2	2	3	3	3	3	3	0,04348
163	1	2	1	1	3	2	3	3	3	3	3	0,04167
164	1	2	1	2	1	2	3	3	3	3	3	0,04348
165	1	2	1	2	2	2	3	3	3	3	3	0,04348
166	1	2	1	2	3	2	3	3	3	3	3	0,04167
167	1	1	2	1	1	1	2	2	2	3	3	0,05242
168	1	1	2	1	2	1	2	2	2	3	3	0,05000
169	1	1	2	1	1	1	2	2	3	3	3	0,05556
170	1	1	2	1	2	1	2	2	3	3	3	0,05879
171	1	1	2	1	1	2	2	2	2	3	3	0,04762
172	1	1	2	1	2	2	2	2	2	3	3	0,04545
173	1	1	2	1	1	2	2	2	3	3	3	0,06108
174	1	1	2	1	2	2	2	2	3	3	3	0,05879
175	1	2	2	1	1	2	2	2	2	3	3	0,04348

**Çizelge 2.1 (Devam)** Jackson montaj hattı 3 istasyonlu durum için elde edilen çözümler.

Sıra No	Görev-istasyon atamaları											Çıktı Hızı
	1	2	3	4	5	6	7	8	9	10	11	
176	1	2	2	1	1	2	2	2	3	3	3	0,05556
177	1	2	2	1	2	2	2	2	2	3	3	0,04167
178	1	2	2	1	2	2	2	2	3	3	3	0,05263
179	1	1	2	2	1	1	2	2	2	3	3	0,03846
180	1	1	2	2	1	1	2	2	3	3	3	0,04762
181	1	1	2	2	2	1	2	2	2	3	3	0,03704
182	1	1	2	2	2	1	2	2	3	3	3	0,04545
183	1	1	2	2	1	2	2	2	2	3	3	0,03571
184	1	2	2	2	1	2	2	2	2	3	3	0,03333
185	1	1	2	2	2	2	2	2	2	3	3	0,03448
186	1	2	2	2	2	2	2	2	2	3	3	0,03226
187	1	1	2	2	1	2	2	2	3	3	3	0,04348
188	1	2	2	2	1	2	2	2	3	3	3	0,04000
189	1	1	2	2	2	2	2	2	3	3	3	0,04167
190	1	2	2	2	2	2	2	2	3	3	3	0,03846
191	1	1	2	1	1	1	2	3	2	3	3	0,05555
192	1	1	2	1	2	1	2	3	2	3	3	0,05879
193	1	1	2	2	1	1	2	3	2	3	3	0,05000
194	1	1	2	2	2	1	2	3	2	3	3	0,04762
195	1	1	2	1	1	1	2	3	3	3	3	0,04997
196	1	1	2	1	2	1	2	3	3	3	3	0,05000
197	1	1	2	2	1	1	2	3	3	3	3	0,05000
198	1	1	2	2	2	1	2	3	3	3	3	0,05000
199	1	1	2	1	1	2	2	3	2	3	3	0,06191
200	1	1	2	1	2	2	2	3	2	3	3	0,06191
201	1	1	2	2	1	2	2	3	2	3	3	0,04545
202	1	1	2	2	2	2	2	3	2	3	3	0,04348
203	1	1	2	1	1	2	2	3	3	3	3	0,05000
204	1	1	2	2	1	2	2	3	3	3	3	0,05000
205	1	1	2	1	2	2	2	3	3	3	3	0,05000
206	1	1	2	2	2	2	2	3	3	3	3	0,04997
207	1	2	2	1	1	2	2	3	2	3	3	0,05879
208	1	2	2	1	2	2	2	3	2	3	3	0,05555
209	1	2	2	1	1	2	2	3	3	3	3	0,05000
210	1	2	2	1	2	2	2	3	3	3	3	0,05000

**Çizelge 2.1 (Devam)** Jackson montaj hattı 3 istasyonlu durum için elde edilen çözümler.

Sıra No	Görev-istasyon atamaları											Çıktı Hızı
	1	2	3	4	5	6	7	8	9	10	11	
211	1	2	2	2	1	2	2	3	2	3	3	0,04167
212	1	2	2	2	2	2	2	3	2	3	3	0,04000
213	1	2	2	2	1	2	2	3	3	3	3	0,04979
214	1	2	2	2	2	2	2	3	3	3	3	0,04901
215	1	1	2	1	1	1	3	2	3	3	3	0,05535
216	1	1	2	1	2	1	3	2	3	3	3	0,05761
217	1	1	2	1	3	1	3	2	3	3	3	0,05535
218	1	1	2	1	1	1	3	3	3	3	3	0,04348
219	1	1	2	1	2	1	3	3	3	3	3	0,04348
220	1	1	2	1	3	1	3	3	3	3	3	0,04167
221	1	1	2	2	1	1	3	2	3	3	3	0,05535
222	1	1	2	2	2	1	3	2	3	3	3	0,05261
223	1	1	2	2	3	1	3	2	3	3	3	0,05444
224	1	1	2	2	1	1	3	3	3	3	3	0,04348
225	1	1	2	2	2	1	3	3	3	3	3	0,04348
226	1	1	2	2	3	1	3	3	3	3	3	0,04167
227	1	1	2	1	1	2	3	2	3	3	3	0,05860
228	1	1	2	1	2	2	3	2	3	3	3	0,05879
229	1	1	2	1	3	2	3	2	3	3	3	0,05555
230	1	1	2	2	1	2	3	2	3	3	3	0,05000
231	1	1	2	2	2	2	3	2	3	3	3	0,04762
232	1	1	2	2	3	2	3	2	3	3	3	0,04997
233	1	1	2	1	1	2	3	3	3	3	3	0,04348
234	1	1	2	1	2	2	3	3	3	3	3	0,04348
235	1	1	2	1	3	2	3	3	3	3	3	0,04167
236	1	1	2	2	1	2	3	3	3	3	3	0,04348
237	1	1	2	2	2	2	3	3	3	3	3	0,04348
238	1	1	2	2	3	2	3	3	3	3	3	0,04167
239	1	2	2	1	1	2	3	2	3	3	3	0,05879
240	1	2	2	1	2	2	3	2	3	3	3	0,05860
241	1	2	2	1	3	2	3	2	3	3	3	0,05555
242	1	2	2	2	1	2	3	2	3	3	3	0,04545
243	1	2	2	2	2	2	3	2	3	3	3	0,04348
244	1	2	2	2	3	2	3	2	3	3	3	0,04545
245	1	2	2	1	1	2	3	3	3	3	3	0,04348

**Çizelge 2.1 (Devam)** Jackson montaj hattı 3 istasyonlu durum için elde edilen çözümler.

Sıra No	Görev-istasyon atamaları											Çıktı Hızı
	1	2	3	4	5	6	7	8	9	10	11	
246	1	2	2	1	2	2	3	3	3	3	3	0,04348
247	1	2	2	1	3	2	3	3	3	3	3	0,04167
248	1	2	2	2	1	2	3	3	3	3	3	0,04348
249	1	2	2	2	2	2	3	3	3	3	3	0,04348