Montaj Hattı İşçi Atama ve Yeniden Dengeleme Problemi için Bir Yapay Arı Kolonisi Algoritması

Aslıhan Karaş

YÜKSEK LİSANS TEZİ

Endüstri Mühendisliği Anabilim Dalı

Haziran 2020

An Artificial Bee Colony Algorithm for Assembly Line Worker Assignment and Rebalancing Problem

Aslıhan Karaş

MASTER OF SCIENCE THESIS

Department of Industrial Engineering

June 2020

Montaj Hattı İşçi Atama ve Yeniden Dengeleme Problemi için Bir Yapay Arı Kolonisi Algoritması

Aslıhan Karaş

Eskişehir Osmangazi Üniversitesi
Fen Bilimleri Enstitüsü
Lisansüstü Yönetmeliği Uyarınca
Endüstri Mühendisliği Anabilim Dalı
Üretim ve Servis Sistemleri Bilim Dalında
YÜKSEK LİSANS TEZİ
Olarak Hazırlanmıştır

Danışman: Doç. Dr. Feriştah Özçelik

Haziran 2020

ETİK BEYAN

Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna göre, Doç. Dr. Feriştah Özçelik danışmanlığında hazırlamış olduğum "Montaj Hattı İşçi Atama ve Yeniden Dengeleme Problemi için Bir Yapay Arı Kolonisi Algoritması" başlıklı YÜKSEK LİSANS tezimin özgün bir çalışma olduğunu; tez çalışmamın tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; tezimde verdiğim bilgileri, verileri akademik ve bilimsel etik ilke ve kurallara uygun olarak elde ettiğimi; tez çalışmamda yararlandığım eserlerin tümüne atıf yaptığımı ve kaynak gösterdiğimi ve bilgi, belge ve sonuçları bilimsel etik ilke ve kurallara göre sunduğumu beyan ederim. 05/06/2020

Aslıhan Karaş

İmza

ÖZET

Seri üretim sistemlerinin en önemli parçalarından olan montaj hatları, verimliliğin artırılması konusunda büyük rol oynamaktadır. Gerçek hayatta, çeşitli aksaklıklar ve değişimler hattın yeniden dengelenmesi ihtiyacını ortaya çıkarmaktadır. Literatürde bulunan montaj hattı yeniden dengeleme çalışmalarında, görev sürelerinin sabit ve işlemi gerçekleştiren işçiden bağımsız olduğu varsayılarak işçi atamaları göz ardı edilmiştir. Ancak, pratikte işçilerin görevleri gerçekleştirmek için ihtiyaç duydukları süreler çeşitli sebepler dolayısıyla değişkenlik göstermektedir. Bu nedenle, montaj hattında meydana gelen değişimler sonrasında sadece görevlerin değil işçilerin de istasyonlara yeniden atanması gerekmektedir.

Bu çalışmada, montaj işlemleri konusunda farklı performanslara sahip işçilerin bulunduğu montaj hatlarının bir veya birkaç iş istasyonu herhangi bir sebeple kullanılamaz duruma geldiğinde yeniden dengelenmesi problemi ilk kez tanımlanmıştır. Yeniden dengeleme sebebiyle çevrim süresinde ve görev-iş istasyonu atamalarında meydana gelen değişimlerin enküçüklenmesi amaçlanmıştır. Problemin çözümü için matematiksel model ve iyileştirilmiş yapay arı kolonisi algoritması önerilmiştir. Elde edilen sonuçlara göre her iki yöntem de küçük boyutlu problemlerde optimal çözümlere ulaşabilmiştir. Bu problemlerin oldukça büyük bir kısmında algoritma, en başarılı çözümlere ulaşma süresi açısından matematiksel modeli geride bırakmıştır. Büyük boyutlu test problemlerinde ise geliştirilen algoritma, matematiksel model ile verilen süre limiti içinde elde edilen çözümlere kıyasla daha başarılı çözümleri oldukça makul süreler içinde elde etmiştir.

Anahtar Kelimeler: Montaj Hatları, Yeniden Dengeleme, İşçi Atama, Matematiksel Modelleme, Yapay Arı Kolonisi Algoritması.

SUMMARY

Assembly lines which are one of the most important parts of mass production systems play a major role in increasing efficiency. In real life, various disruptions and changes bring about the need to rebalance the line. In assembly line rebalancing studies in the literature, worker assignments were ignored, assuming that task times were constant and independent from the worker performing the operation. However, in practice, the time periods that workers need to perform tasks vary for various reasons. Therefore, after changes to the assembly line, not only the tasks but also the workers should be reassigned to the stations.

In this study, the problem of rebalancing assembly lines with workers with different performances in assembly operations when one or more workstations become unusable for any reason has been identified for the first time. It was intended to minimize changes in cycle time and task-workstation assignments due to rebalancing. A mathematical model and an improved artificial bee colony algorithm were proposed for solving the problem. According to the results obtained, both methods were able to achieve optimal solutions in small-sized problems. In a large number of these problems, the algorithm outperformed the mathematical model in terms of time to reach the most successful solutions. In large-scale test problems, the algorithm developed achieved more successful solutions in fairly reasonable times compared to solutions obtained by the mathematical model within the time limit given.

Keywords: Assembly Lines, Rebalancing, Worker Assignment, Mathematical Modelling, Artificial Bee Colony Algorithm.

TEŞEKKÜR

Yüksek lisans tezimin her aşamasında bilgi ve tecrübelerini benimle paylaşan, bu süreçte büyük emekleri olan değerli danışman hocam Doç. Dr. Feriştah Özçelik'e ve tez savunması sırasında kıymetli görüşlerini aktaran Doç. Dr. Tuğba Saraç ve Dr. Öğr. Üyesi Banu Güner hocalarıma teşekkürü bir borç bilirim. Ayrıca, hayatımın her anında yanımda olan ve başarılı olacağıma inanan sevgili aileme, tüm yakınlarıma ve bu sürecin ilk günlerinden itibaren desteğini hissettiğim arkadaşım ve meslektaşım Arş. Gör. Hatice Erdoğan Akbulut'a teşekkür ederim.

İÇİNDEKİLER

Sayfa

ÖZET vi
SUMMARY vii
TEŞEKKÜR viii
İÇİNDEKİLER ix
ŞEKİLLER DİZİNİ xii
ÇİZELGELER DİZİNİ xiii
SİMGELER VE KISALTMALAR DİZİNİ xiv
1. GİRİŞ VE AMAÇ 1
2. ÜRETİM VE MONTAJ HATLARI 4
2.1. Üretim Kavramı ve Üretim Sistemleri 4
2.1.1. Siparişe göre üretim 4
2.1.2. Parti tipi üretim 5
2.1.3. Proje tipi üretim 6
2.1.4. Seri üretim 7
2.1.4.1. Transfer hattı 7
2.1.4.2. Montaj hattı 8
2.2. Montaj Hattı Dengeleme 8
2.2.1. Montaj hattı dengelemede kısıtlar 9
2.2.1.1. Temel kısıtlar 10
2.2.1.2. Yan kısıtlar 10
2.2.2. Montaj hattı dengelemede temel kavramlar 11
2.2.2.1. Görev 11
2.2.2.2. İş istasyonu 11
2.2.2.3. İş istasyonu süresi 11
2.2.2.4. Toplam iş süresi 12
2.2.2.5. Çevrim süresi 12
2.2.2.6. Gerekli en az iş istasyonu sayısı 12
2.2.2.7. Ortalama iş istasyonu süresi 13
2.2.2.8. Teknolojik öncelik diyagramı 13

İÇİNDEKİLER (devam)

Sayfa

2.2.2.9. Öncelik matrisi 14
2.2.2.10. Düzgünlük indeksi 14
2.2.2.11.Hat etkinliği 15
2.2.2.12. Kuramsal etkinlik 15
2.2.2.13. Esneklik oranı 15
2.2.2.14. Sıra kuvveti 16
2.2.2.15. Denge kaybı 16
2.2.2.16. Darboğaz 17
2.2.2.17. Güvenlik düzeyi 17
2.2.2.18. Ürün karışımı 17
2.2.3. Montaj hatlarının sınıflandırılması 17
2.2.3.1. Model sayısına göre montaj hatları 17
2.2.3.2. İstasyon yerleşimine göre montaj hatları 18
2.2.3.3. Hattın kontrol yapısına göre montaj hatları 22
2.2.3.4. Kurulum sıklığına göre montaj hatları 23
2.2.3.5. Otomasyon seviyesine göre montaj hatları 24
2.2.4. Montaj hattı dengeleme problemi 24
2.2.4.1. Varsayım özelliklerine göre MHD problemleri 27
2.2.4.2. Amaç sayısına göre MHD problemleri 28
2.2.4.3. Model sayısına göre MHD problemleri 28
2.2.4.4. İşlem sürelerinin doğasına göre MHD problemleri 29
2.2.4.5. Hat yerleşimine göre MHD problemleri 29
2.2.5. Montaj hattı işçi atama ve yeniden dengeleme problemi 29
3. LİTERATÜR ARAŞTIRMASI 32
3.1. Literatürde Montaj Hattı Yeniden Dengeleme Problemi 32
3.1.1. Stokastik görev süreli montaj hattı yeniden dengeleme problemi 32
3.1.2. Deterministik görev süreli montaj hattı yeniden dengeleme problemi 34
3.2. Literatürde Montaj Hattı İşçi Atama ve Dengeleme Problemi 41

İÇİNDEKİLER (devam)

Sayfa

4. MATERYAL VE YÖNTEM 55
4.1. Önerilen Matematiksel Model 55
4.2. Yapay Arı Kolonisi Algoritması Hakkında Genel Bilgiler 58
4.3. Geliştirilen Yapay Arı Kolonisi Algoritması 61
4.3.1. Çözümlerin temsil edilmesi 64
4.3.2. Montaj hattı işçi atama ve yeniden dengeleme problemi girdileri 64
4.3.3. Montaj hattı işçi atama ve yeniden dengeleme problemi safhaları 65
4.3.3.1. Başlangıç çözümlerin oluşturulması 65
4.3.3.2. Görevli arı safhası 67
4.3.3.3. Gözcü arı safhası 69
4.3.3.4. Kâşif arı safhası 71
4.3.3.5. Yerel arama safhası 72
5. BULGULAR VE TARTIŞMA 74
5.1. Kullanılan Test Problemleri 74
5.2. Montaj Hattı İşçi Atama ve Dengeleme Probleminin Çözülmesi 74
5.3. Montaj Hattı İşçi Atama ve Yeniden Dengeleme Probleminin Çözülmesi 75
5.3.1. Matematiksel model ile çözüm 75
5.3.2. Yapay arı kolonisi algoritması ile çözüm 76
5.4. Elde Edilen Sonuçlar 77
6. SONUÇ VE ÖNERİLER 85
KAYNAKLAR DİZİNİ 87

ŞEKİLLER DİZİNİ

Şekil **Sayfa**

2.1. Teknolojik öncelik diyagramı 13
2.2. Model sayısına göre montaj hatları: a) tek modelli montaj hattı, b) karma modelli montaj hattı, c) çok modelli montaj hattı 18
2.3. Düz bir montaj hattı 19
2.4. U-tipi bir montaj hattı 19
2.5. Çift taraflı bir montaj hattı 20
2.6. Paralel montaj hatları 21
4.1. Önerilen YAK algoritmalarının sözde-kodu 62
4.2. Örnek probleme ilişkin öncelik diyagramı 63
4.3. Çözüm vektörü 64
4.4. İlk hat dengesi çözüm vektörü 65
4.5. Başlangıç çözüm vektörü 67
4.6. Görevli arı safhasının sözde-kodu 69
4.7. Görevli arı safhasında bulunan komşu çözüm vektörü 69
4.8. Görevli arı safhasının sözde-kodu 70
4.9. Gözcü arı safhasında bulunan komşu çözüm vektörü 71
4.10. Kâşif arı safhasının sözde-kodu 72
4.11. Yerel arama safhasının sözde-kodu 73
5.1. T1 problemi için iterasyon sayısına bağlı amaç fonksiyonu değeri değişimi 77
5.2. Roszieg problem ailesi için ağırlıklı Öklid uzaklıkları 83
5.3. Heskia problem ailesi için ağırlıklı Öklid uzaklıkları 83
5.4. Tonge problem ailesi için ağırlıklı Öklid uzaklıkları 84
5.5. Wee-Mag problem ailesi için ağırlıklı Öklid uzaklıkları 84

ÇİZELGELER DİZİNİ

Çizelge **Sayfa**

2.1. Öncelik matrisi 14
3.1. MHYD literatüründe yer alan çalışmalar 39
3.2. MHİAD literatüründe yer alan çalışmalar 53
4.1. İşçi bazında işlem süreleri 63
5.1. MHİAD problemi test problemlerinin özellikleri 74
5.2. Başlangıç istasyon sayısı açısından kapanan istasyon sayıları 75
5.3. Roszieg problem ailesine ait sonuçlar 79
5.4. Heskia problem ailesine ait sonuçlar 80
5.5. Tonge problem ailesine ait sonuçlar 81
5.6. Wee-Mag problem ailesine ait sonuçlar 82

SİMGELER VE KISALTMALAR DİZİNİ

Simgeler	Açıklama
i, j	Görev indisleri
h	İşçi indisi
s	İş istasyonu indisi
N	Görev sayısı
H	İşçi sayısı
S	İş istasyonu sayısı
$ÇS$	Çevrim süresi
TS	Eldeki toplam süre
$ÜS$	Üretilecek ürün sayısı
$S_{en\ az}$	Gerekli en az istasyon sayısı
OS^*	Ortalama iş istasyonu süresi
$Dİ$	Düzgünlük indeksi
HE	Hat etkinliği
ES_s	s. iş istasyonunun etkin süresi
KE	Kuramsal etkinlik
EO	Esneklik oranı
Y	Öncelik matrisinde 0 değerine sahip eleman sayısı
SK	Sıra kuvveti
$ÖS$	Öncelik ilişkileri sayısı
DK	Denge kaybı
t_s	s. iş istasyonunun süresi
t_{enb}	En büyük istasyon süresi
t_{ih}	Görev i'nin işçi h açısından işlem süresi
P_j	Görev j'nin komşu öncül kümesi
F	Kapanan istasyonların kümesi
f	Kapanan istasyonlarda başlangıç durumdaki toplam görev sayısı
w	Çevrim süresinin alt sınırlardan kısmi sapması için ağırlık
a_{is}	Görev i'nin başlangıçta iş istasyonu s'ye atanıp atanmama durumunu gösteren parametre

SİMGELER VE KISALTMALAR DİZİNİ (devam)

Simgeler	**Açıklama**
C_0	Hattın başlangıçtaki çevrim süresi
C	Yeniden dengelenmiş hattın çevrim süresi
T	Yeniden atanan görev sayısı
x_{ish}	Görev i'nin işçi h'ye iş istasyonu s'de atanıp atanmama durumunu gösteren karar değişkeni
y_{sh}	İşçi h'nin iş istasyonu s'ye atanıp atanmama durumunu gösteren karar değişkeni
q_{is}	Görev i'nin yeni durumda farklı bir iş istasyonundan iş istasyonu s'ye atanıp atanmama durumunu gösteren karar değişkeni
M	Yeterince büyük bir sayı
P	Problem
z	Amaç fonksiyonu değeri
z_{eniyi}	En başarılı amaç fonksiyonu değeri
z_{ort}	Amaç fonksiyonu değerlerinin ortalaması
d	Öklid uzaklığı

Kısaltmalar	**Açıklama**
BMHD	Basit Montaj Hattı Dengeleme
COMSOAL	Computer Method for Sequencing Operations for Assembly Lines
ErgoMHİAD	Ergonomik Montaj Hattı İşçi Atama ve Dengeleme
GAMS	The General Algebraic Modeling System
GMHD	Genel Montaj Hattı Dengeleme
HİAMHD	Hiyerarşik İşçi Ataması ile Montaj Hattı Dengeleme
MATLAB	Matrix Laboratory
MHD	Montaj Hattı Dengeleme
MHİAD	Montaj Hattı İşçi Atama ve Dengeleme
MHİAYD	Montaj Hattı İşçi Atama ve Yeniden Dengeleme
MHİED	Montaj Hattı İşçi Entegrasyonu ve Dengeleme
MIP	Karma tamsayılı programlama
MHYD	Montaj Hattı Yeniden Dengeleme

SİMGELER VE KISALTMALAR DİZİNİ (devam)

Kısaltmalar	Açıklama
MNSGA	Modified Non-dominated Sorting Genetic Algorithm
NP	Nondeterministic Polynomial Time
NSGA	Non-dominated Sorting Genetic Algorithm
OCRA	Operational Competitiveness RAting
OİMMHİAD	Olasılıklı İşçi Mevcudiyeti ile Montaj Hattı İşçi Atama ve Dengeleme
TOPSIS	Technique for Order Preference by Similarity to Ideal Solution
YAK	Yapay Arı Kolonisi

1. GİRİŞ VE AMAÇ

Günümüzde, artan rekabet ortamında işletmelerin varlıklarını koruyabilmeleri için birim üretim maliyetlerini düşürerek müşteri talebini zamanında ve yüksek kaliteye sahip ürünlerle karşılayabilmeleri gerekir. Bu amaç doğrultusunda, kısıtlı üretim kaynaklarını rasyonel biçimde kullanmak gerekmektedir. Bu sebeple, birçok işletme üretim verimliliğini artırılabilmek için seri üretim sistemlerini benimsemiştir. Kesikli seri üretim sistemlerinin bir parçası olan montaj hatları, standartlaştırılmış ürünlerin yüksek miktarlarda üretimi açısından oldukça önemli bir bileşendir (Akyol ve Baykasoğlu, 2019a).

Montaj hatları, belirli bir yerleşim düzeni doğrultusunda sıralanmış belirli sayıda iş istasyonundan oluşan, parçaların/yarı mamullerin hat boyunca bir taşıma aracı vasıtasıyla hareket ettiği ve istasyonlarda bulunan işçilerin/robotların kendilerine atanmış olan montaj görevlerini gerçekleştirdikleri üretim sistemleridir.

İlk montaj hattı, 1913 yılında Henry Ford ve ekibi tarafından otomobil üretiminde kullanılmıştır. Ford, işçilerin etrafına yerleştirildiği yürüyen bant sistemini ilk defa bir fabrikada geliştirerek Model-T'yi üretmiştir. Yeni sistem sayesinde tek seferde bir otomobil üretmek yerine tek seferde bir parçanın montajı gerçekleştirilmiştir. O zamandan beri montaj hatları, üretim maliyetlerini düşürürken üretim yönetiminde de reforma sebep olmuştur (Küçükkoç, 2019).

Bir montaj hattının verimli olarak çalışabilmesi için toplam iş yükünün iş istasyonlarına dengeli bir şekilde dağıtılması gerekmektedir. Bu noktada, montaj hattı dengeleme (MHD) problemi ortaya çıkmıştır. MHD, literatürde ilk kez Bryton'un (1954) yüksek lisans tezinde ele alınan bir kavram iken problemin tanımı Salveson (1955) tarafından yapılmıştır. MHD problemi, bazı kısıtlar altında belirli bir performans ölçütünü eniyilecek şekilde görevlerin iş istasyonlarına mümkün olduğunca dengeli bir biçimde atanması olarak tanımlanabilir.

Montaj hatlarının dengede olmaması, maliyetlerin artmasına ve üretim hızının düşmesine sebep olacağından bu hatların işletme performansı üzerinde önemli bir etkisinin

bulunduğu söylenebilir. Buna bağlı olarak, MHD çalışmaları üretim verimliliği açısından büyük önem arz etmektedir. MHD problemi, ortaya çıkışından itibaren literatürde yaygın şekilde çalışılan bir konu olmuş ve NP (Nondeterministic Polynomial Time)-zor problem sınıfında bulunan bu problemin çözümü için çeşitli kesin, sezgisel ve meta-sezgisel çözüm yöntemleri geliştirilmiştir.

Basit MHD probleminde, her işçinin belirli bir görevi gerçekleştirmek için eşit süreye ihtiyacı olduğu varsayılmıştır. Ancak, işçiler deneyim ve yetenek gibi unsurlar sebebiyle farklı özelliklere sahip olduğundan bu varsayım gerçek hayat problemlerini tam olarak yansıtmamaktadır.

Görev sürelerinin işçi becerilerine bağlı olarak değişkenlik gösterdiğini temel alan montaj hattı işçi atama ve dengeleme (MHİAD) problemi, basit MHD probleminin bir uzantısıdır (Zacharia & Nearchou, 2016). Bu problemde, görevlerin tamamlanma süreleri yetenek farklılıkları sebebiyle işçiye özgüdür. MHİAD probleminde, istenen etkinlik ölçüsünü optimize etmek için belirli kısıtlar altında görev ve işçiler iş istasyonlarına eşzamanlı olarak atanır. Bu problem, engelli bireylerin çalıştığı korunaklı iş merkezleri için önerilmesine rağmen yüksek işçi değişim oranına sahip geleneksel montaj hatları için de oldukça önemlidir (Akyol ve Baykasoğlu, 2019a).

MHD ve MHİAD problemleri, hattın sadece kurulum sonrasında bir kez dengelenmesini ele almaktadır. Ancak, gerçek hayatta işletmelerde ve piyasada mevcut hat dengesinin bozulmasına sebep olabilecek çeşitli değişimler meydana gelebilmektedir. Buna bağlı olarak, hattın yeni duruma uyum sağlayabilmesi için yeniden dengelenmesi gerekmektedir. Ürün yapısının veya görev sürelerinin değişimi, talepte meydana gelen değişimler, iş istasyonlarının kapanması/arızalanması gibi durumlar sebebiyle başlangıç hat dengesini göz önünde bulundurarak belirli kısıtlar altında istenen performans ölçütünü optimize etmek amacıyla hattın yeniden dengelenmesi, montaj hattı yeniden dengeleme (MHYD) problemi olarak adlandırılmaktadır.

Literatürde bulunan MHYD çalışmalarında, görev sürelerinin işlemi gerçekleştiren işçiden bağımsız ve sabit olduğu varsayılarak işçi atamaları göz ardı edilmiştir. Ancak, pratikte işçilerin görevleri gerçekleştirmek için ihtiyaç duydukları süreler çeşitli sebeplere

bağlı olarak değişkenlik göstermektedir. Bu sebeple, montaj hattında meydana gelen değişimler sonrasında sadece görevlerin değil işçilerin de istasyonlara yeniden atanması gerekmektedir.

Bu çalışmada, montaj hattı işçi atama ve yeniden dengeleme (MHİAYD) problemi ilk kez tanımlanmıştır. Bu problem türünde, görev süreleri hangi görevin hangi işçiye atandığına bağlı olarak değişmekte ve görevlerin yanı sıra işçilerin de istasyonlara atanması gerçekleştirilmektedir. NP-zor problem sınıfında bulunan MHYD problemine, işgücü atama probleminin eklenmesiyle problem karmaşıklığı daha da artmaktadır.

Ele alınan problemde, bir veya daha fazla iş istasyonunun kapanması sonucunda ilk duruma göre mümkün olan en az değişkenlikle hattın yeniden dengelenmesi amaçlanmıştır. Değişkenlik açısından; ilk hat dengesi ile yeni durum arasındaki çevrim süresi ve görev-iş istasyonu atamalarındaki değişim göz önünde bulundurulmuştur.

Çalışmada, çevrim süresi ve görev-iş istasyonu atamaları açısından mümkün olan en az değişkenlikle hattın yeniden dengelenmesi için bir matematiksel model geliştirilmiştir. Büyük boyutlu problemlerin çözümü için ise bir yapay arı kolonisi (YAK) algoritması önerilmiştir. İyi çözümlere daha hızlı ulaşabilmek adına klasik YAK algoritmasının bazı safhalarında çeşitli değişiklikler yapılmıştır. Önerilen yöntemler, literatürde bulunan MHİAD test problemleri kullanılarak karşılaştırılmıştır.

Bu tez çalışması, altı bölümden oluşmaktadır. İzleyen bölümde, üretim kavramı ve üretim sistemleri hakkında bilgi verilmiş ve montaj hattı, hat dengeleme kavramları ve MHD problemi ayrıntılı şekilde açıklanmıştır. Üçüncü bölümde, MHYD ve MHİAD problemleri konusunda literatürde yapılan çalışmalar özetlenmiştir. Dördüncü bölümde, MHİAYD problemi tanımlanmış ve problemin çözümü için önerilen matematiksel model sunulmuştur. Bu bölümde, YAK algoritması hakkında genel bilgi verilmesinin ardından geliştirilen algoritmanın adımları açıklanmıştır. Beşinci bölümde, her iki çözüm yöntemi kullanılarak elde edilen sonuçlar karşılaştırılmıştır. Son bölümde ise tez kapsamında gerçekleştirilen çalışma değerlendirilmiş ve gelecekte yapılabilecek çalışmalar önerilmiştir.

2. ÜRETİM VE MONTAJ HATLARI

Bu bölümde öncelikle, üretim kavramı ve üretim sistemleri açıklanmıştır. Ardından; montaj kavramı, montaj hattı ve montaj hattı dengeleme problemine yönelik bilgiler sunulmuştur.

2.1. Üretim Kavramı ve Üretim Sistemleri

Üretim, ekonomistler tarafından fayda yaratmak şeklinde tanımlanmaktadır. Mühendisler açısından ise fiziksel bir varlık üzerinde onun değerini artırmaya yönelik yapılan bir değişiklik veya hammadde ve yarı mamulleri kullanılabilir hale dönüştürme işlemine verilen isimdir (Karaca, 1996). Kısaca, mal ve hizmetlerin ortaya çıkarılması olarak ifade edilebilir (Soba, 2008).

İşletmenin bir sistem olduğu düşünülürse, üretim sistemi bu sistemin bir alt sistemi olarak görülebilir. Üretim sistemi, elde bulunan kaynakların rasyonel bir biçimde kullanılarak girdilerin ürün veya hizmete dönüştürüldüğü süreçtir (Gökşen, 2003). Hammadde ve/veya yarı mamuller, üretim sisteminde bir dönüşüm birimi aracılığıyla ürün haline getirilir (Karaca, 1996).

Üretim sistemleri; üretim tipi, makine ve gereçlerin yerleşimi, ürün türü, sayısı ve yapısı gibi özellikler açısından farklılıklar gösterir (Tanrıtanır, 1990). Üretim sistemlerini, üretim miktarı ve ürün çeşidi açısından siparişe göre üretim, parti tipi üretim, proje tipi üretim ve seri (akış tipi) üretim olarak sınıflandırmak mümkündür.

2.1.1. Siparişe göre üretim

Üretilecek ürünün özelliğinin, miktarının ve üretim zamanının müşteri siparişine göre belirlendiği üretim sistemidir. Üretimde nitelikli işçiler görev almakta ve genel amaçlı makine ve donanımlar kullanılmaktadır. Bu üretim tipinde genellikle küçük partiler halinde üretim yapılır. Ürün çeşitliliği fazla ve esneklik yüksektir. Bu üretim tipine örnek olarak

çeşitli amaçlar için kullanılan makineler, özel elektronik cihazlar ve büyük takım tezgâhları verilebilir (Kağnıcıoğlu vd., 2012).

Siparişe göre üretim sistemi, üretimin yapıldığı sürelerin düzeni açısından üç sınıfa ayrılabilir (Karaca, 1996):

- Az sayıda ürünün bir defada üretilmesi
- Az sayıda ürünün talep geldikçe belirsiz aralıklarla üretilmesi
- Az sayıda ürünün talep geldikçe belirli aralıklarla üretilmesi

Bu üretim tipinin avantajlarından biri, çalışanların sürekli aynı görevleri yerine getirmemelerine bağlı olarak daha yüksek motivasyonla çalışabilmeleridir. Ayrıca, bir makinede meydana gelen arıza tüm üretimin durmasına sebep olmamaktadır. Kontrolün zor ve verimliliğin düşük olması ise bu üretim sisteminin en büyük dezavantajlarıdır. Ek olarak, üretim akışının düzensiz olması da değişken üretim maliyetini artırmaktadır. Üretim alanında yarı bitmiş ürünlerin bulundurulması stok maliyetlerini artırabilmekte ve uzun süre bekleyen yarı bitmiş ürünler alan ihtiyacı doğurmaktadır. Çeşitli ürünlerin makineler arasında izledikleri farklı rotalar, taşıma maliyetlerinin yüksek olmasına sebep olmaktadır (Kağnıcıoğlu vd., 2012).

Belirli ve belirsiz aralıklarla üretimi yapılan ürünler için yöntem, işlem planlanması ile kontrol faaliyetlerinin düzenlenmesi ve bunlarla ilgili bilgilerin daha sonra kullanılabilmesi için saklanması faydalıdır. Siparişe göre üretim tipinde, makine ve iş gücü kapasitesinden yararlanma oranı düşük ve siparişlerin yığılması sebebiyle kuyrukta bekleme olasılığı yüksektir. Siparişlerin gelme aralığı belirliyse, planlama ve kontrol kolaylaşır. Tekrarlar dolayısıyla yöntem geliştirme ve standart zaman tespiti çalışmalarının maliyeti düşer (Tanrıtanır, 1990).

2.1.2. Parti tipi üretim

Bir ürünün özel bir siparişi ve sürekli bir talebi karşılama amacıyla partiler halinde üretiminin yapılması parti tipi üretim olarak adlandırılmaktadır. Parti tipi üretim, endüstride en sık karşılaşılan üretim sistemidir (Tanrıtanır, 1990).

Bu üretim sistemi de siparişe göre üretimde olduğu gibi üç grupta incelenir (Karaca, 1996):

- Partinin sadece bir defa üretilmesi
- Partinin belirsiz aralıklarla üretilmesi
- Partinin sürekli talebi karşılamak amacıyla bilinen zaman aralıklarıyla üretilmesi

Parti tipi üretim sistemi, yüksek miktarlarda çok çeşitli ürünlerin üretimine yöneliktir. Ev eşyası, hazır giyim ve gıda gibi ürünler partiler halinde üretilir. Belirli bir ürünün partiler halinde üretilmesinin ardından makine ve donanım farklı bir ürünün üretimi için hazırlanır. En uygun parti büyüklüğünün belirlenmesi ve kapasite kullanım oranını en büyük yapacak üretim programlarının hazırlanması bu üretim tipinde oldukça önemlidir (Kağnıcıoğlu vd., 2012).

Parti tipi üretim sisteminin zorluğu; parti büyüklüklerinin ve parti adetlerinin belirlenmesi ile partilerin çizelgelenmesidir. Parti büyüklüğü ve üretim periyodu sıklığına bağlı olarak makine, takım ve işgücü planlaması değişir. Parti hacmi büyüdükçe, üretilen ürün çeşitliliği azaldıkça ve periyotlar belirginleştikçe üretim planlama ve kontrol faaliyetlerinin verimliliği artar (Tanrıtanır, 1990).

2.1.3. Proje tipi üretim

Proje tipi üretim, müşteri siparişine göre eşsiz bir ürünün tek defaya mahsus olmak üzere üretildiği üretim sistemidir. Ürün özelliklerini müşterinin belirlemesi ve tek ürünün üretilmesi sebebiyle siparişe göre üretim sistemine benzemektedir. Bu üretim tipinde, ürünün konumu genellikle sabittir. Makine ve çalışanlar ise ürün çevresinde veya içinde bulunur. Bu sayede, eşzamanlı olarak birden fazla faaliyet gerçekleştirilir. Bu üretim sistemi söz konusu olduğunda özel proje yönetim tekniklerinden faydalanılmaktadır. Köprü inşa edilmesi, yük gemisi ve kargo uçağı imalatı gibi büyük boyutlu ürünlerin üretimi bu üretim tipine örnektir. Üretimde, uzmanlaşmış çalışanlar görev almakta ve özel tasarlanmış aletler kullanılmaktadır. Ürünün özel olması sebebiyle maliyet oldukça yüksektir. Tek bir ürün üretildiği için ürün proje olarak görülür ve gerçekleştirilecek faaliyetler, bunların sıraları ve

süreleri belirlendikten sonra planlama yapılarak belirlenen sürede proje tamamlanmaya çalışılır. (Kağnıcıoğlu vd., 2012).

2.1.4. Seri üretim

Makine ve donanımların belirli bir ürünün üretimi için kullanıldığı üretim sistemidir. Ürün talebi sürekli ve yüksektir. Buna bağlı olarak, sürekli ve yüksek miktarlarda üretim yapılmaktadır. Girdiler, aralıksız olarak ürüne dönüşmektedir. Üretimde gerçekleştirilen işlemler yüksek otomasyona sahiptir ve özel donanım ve aletler kullanılır. Donanım otomasyona bağlı olduğundan işgücü gereksinimi daha azdır. Çalışanlar tarafından genellikle izleme ve kontrol görevleri yerine getirilir. Çıktı miktarı ve çeşidinin değiştirilmesi güçtür. Tüm sistem birbirine bağlı çalıştığı için girdiler durdurulduğunda tüm sistemi de durdurmak gerekir. Sistemi durdurmak maliyetli olduğundan genel bakım harici bir sebep olmadan üretim durdurulmamaktadır. Seri üretim sisteminde, sabit maliyetler yüksek iken hammadde ve malzeme maliyetleri değişkendir. Kullanım oranının en büyük olması istenir (Kağnıcıoğlu vd., 2012).

Bu üretim sistemi, sürekli seri üretim ve kesikli seri üretim olmak üzere iki gruba ayrılır. Sürekli seri üretimde, ürünler doğal özelliklerine bağlı olarak kendiliğinden akar. Çimento ve şeker üretimi bu gruba dahildir. Beyaz eşya ve motorlu taşıtlar gibi özel akış sistemi gerektiren ve büyük miktarlarda yapılan üretim ise kesikli seri üretim adını almaktadır (Ağpak 2001: Özgörmüş'ten (2007)). Kesikli seri üretim sistemi ise kendi içinde transfer hattı ve montaj hattı olmak üzere iki gruba ayrılır.

2.1.4.1. Transfer hattı

Transfer hatları, bir transfer mekanizması ve bir kontrol sistemi ile bir sisteme entegre edilmiş sıralı iş istasyonlarından meydana gelen senkronize üretim hatlarıdır. İş istasyonları, iş öğeleri üzerinde gerekli işlemlerin gerçekleştirildiği noktalardır. Bu hatlarda, makineler konveyör vasıtasıyla birbirine bağlıdır ve birimler konveyör sistemi üzerinde eşzamanlı olarak hareket eder. Bu hatlar, tüm makineler her çevrimde bir birim üretirken tek bir makine gibi çalışır. İlaç ve gıda endüstrilerindeki dolum ve paketleme hatları bu sistemlere örnek olarak gösterilebilir (Altıok, T., 1997).

2.1.4.2. Montaj hattı

Montaj hatları, bitmemiş ürünlerin ve bileşenlerin bir istasyondan sonrakine aktarılmasını sağlayan bir taşıma aracı ile birbirine bağlanmış iş istasyonlarından oluşan hatlardır. Her iş istasyonunda, çevrim süresi olarak adlandırılan süre içinde o istasyona atanmış temel görevler gerçekleştirilir. Görevler, öncelik ilişkilerine göre istasyonlara atanır (Pereira, 2018). Montaj hatlarında; otomobiller ve diğer taşıma araçları, ev aletleri, elektronik eşya, bilgisayar, motor gibi ürünlerin montajı gerçekleştirilebilir. Montaj görevleri; işçiler, robotlar veya her ikisi tarafından yapılabilir (Akyol ve Baykasoğlu, 2019a).

2.2. Montaj Hattı Dengeleme

Montaj, nihai bir ürün oluşturmak amacıyla bileşenlerin ardışık olarak birleştirildiği bir üretim sürecidir (Özcan, 2010). Başka bir ifadeyle, tasarım ve imalat işlemlerinin sonucunda parçaların birleştirilerek bitmiş ürünün elde edilmesi işlemidir. Montaj, imalat sanayiinde ürün geliştirmenin en önemli safhalarındandır (Çiçek ve Gülesin, 2007).

Montaj hatları, istasyonlar olarak adlandırılan, işlemlerin gerçekleştirildiği üretken birimlerin seri şekilde sıralandığı akış odaklı üretim sistemleridir (Boysen vd., 2006). Başka bir deyişle, montaj hatları, bir dizi iş istasyonunun bir konveyör bandı veya benzer bir mekanik malzeme taşıma sistemi boyunca sıralandığı akış tipi üretim sistemleridir. Bu hatlar, yüksek kaliteli ve düşük maliyetli standart homojen ürünler üretmek için tasarlanmıştır. Bir montaj sürecinde, toplam iş içeriğinin bölünemez bir parçasına görev denir. Bir görevi gerçekleştirmek için gerekli olan süre, görev süresi olarak adlandırılır. Görevler, öncelik ilişkilerine göre istasyonlara tahsis edilir. Her istasyonda, bir veya daha fazla görevden oluşabilen bir görev grubu, çevrim süresi olarak bilinen sınırlı bir süre içinde yerine getirilir. Bitmemiş ürünler, her çevrim sonunda sonraki istasyona aktarılır. Son istasyonda ürünün montajı tamamlanır ve ürün hattan ayrılır (Özcan, 2010).

Endüstrileşme sürecinde, toplam işin öğelere ayrılarak farklı işçiler tarafından gerçekleştirilmesiyle daha hızlı, kitlesel ve düşük maliyetli üretim yapılabileceği görüşü ortaya çıkmıştır (Erkut ve Baskak, 2003). Hızla gelişen teknoloji ve artan talep ihtiyaçlarını

karşılama noktasında, istenen kalitede üretim yapabilmek için montaj hatları yaygın şekilde kullanılmaktadır (Aksoy vd., 2014).

İşletmeler, yüksek miktarlardaki talebin mümkün olan en ekonomik ve en hızlı şekilde karşılanmasını hedeflemektedir (Doğan ve Sakallı, 2016). İş istasyonları arasındaki iş yüklerinin dengeli olması ve istenen çıktının elde edilebilmesi için görevlerin istasyonlara en uygun şekilde atanması gerekir. İstasyonlar arasındaki hız farklarının en aza indirilmesini mümkün kılan bu planlama faaliyeti, MHD olarak adlandırılır (Kağnıcıoğlu, 2012).

MHD; üretim hızının artırılması, sağlıklı bir tasarlamanın yapılması ve işletmelerin ekonomik sorunlarına çözüm sağlaması açısından oldukça önemli bir konudur. Montaj hatlarının kurulmasında tüm amaçları aynı anda en üst düzeye ulaştırmak mümkün değildir. Asıl amaç, bu amaçlar arasında bulunan çelişkileri göz önünde bulundurarak en uygun çözümü elde etmektir. Hat dengeleme amaçları aşağıdaki gibi ifade edilebilir (Erkut ve Baskak, 2003):

- Malzeme akışının düzenli olmasını sağlamak
- İnsan gücünden en üst düzeyde yararlanmak
- Makine kapasitelerini en üst düzeyde kullanmak
- İşlem sürelerini enküçüklemek
- Malzeme kullanımını enküçüklemek
- Âtıl zamanları veya denge kayıplarını enküçüklemek
- İş istasyonu sayısını enküçüklemek
- Denge kayıplarını istasyonlar arasında düzgün bir şekilde dağıtmak
- Tüm kısıtları sınırları zorlamadan sağlamak
- Montaj hattı dengeleme maliyetinin en alt düzeyde tutulmasını sağlamak

2.2.1. Montaj hattı dengelemede kısıtlar

Montaj hatlarının dengede olmaması, denge kayıplarının yüksek olmasına ve darboğazların oluşmasına sebep olacağından işletmelerin etkin, ekonomik ve hızlı üretim yapmasına engel olabilmektedir (Doğan ve Sakallı, 2016). Bu sebeple, montaj hatlarının dengelenmesi, üretim sürecinin verimliliği açısından oldukça kritiktir. Ancak, montaj hatları

dengelenirken bazı kısıtların göz önünde bulundurulması gerekmektedir. Bu kısıtlar, temel ve yan kısıtlar olmak üzere ikiye ayrılır (Erkut ve Baskak, 2003):

2.2.1.1. Temel kısıtlar

Görevlerin bölünmezliği kısıtı: Her görev sadece bir iş istasyonuna atanabilir.

Çevrim süresi kısıtı: Çevrim süresi, montajı yapılmakta olan ürünün her bir iş istasyonunda kalmasına izin verilen süredir. Bir istasyonda gerçekleştirilecek tüm görevlerin sürelerinin toplamı, yani istasyon yükü, çevrim süresini aşamaz.

Öncelik ilişkileri kısıtı: İş istasyonlarına görevler atanırken bunların arasındaki öncelik ilişkileri sağlanmalıdır. Bir işlemi gerçekleştirmeden önce yapılması gereken işlem veya işlemler ilgili işlemin öncülü iken, bu işlem gerçekleştirildikten sonra yapılabilenler ise ilgili işlemin ardıllarıdır.

2.2.1.2. Yan kısıtlar

Konum kısıtı: Özellikle büyük hacimli ürünlerin montajı sırasında bazı işlemlerin hattın belirli bir tarafında gerçekleştirilmesi gerekmektedir. Çevrim süresinin aşılmaması için hattın farklı taraflarında yapılması gereken işlemlerin aynı işçiye atanması uygun değildir. Buna bağlı olarak, hattın farklı taraflarına farklı işçiler atanır.

Sabit donanım kısıtı: Montaj hatlarında, yerlerinin değiştirilmesi mümkün olmayan bazı makine ve ekipmanlar bulunabilmekte ve bunları gerektiren işlemler sadece bu makine ve ekipmanların bulunduğu istasyonlarda gerçekleştirilebilmektedir. Bu kısıt, işlemlerin değiştirilebilirliğinin azalmasına yol açmaktadır.

İstasyon yükü kısıtı: Bazı durumlarda, belirli iş istasyonlarına atanan işlem süreleri toplamının, çevrim süresinden az olması gerekir. Böylelikle, baştaki istasyon veya istasyonlarda meydana gelen herhangi bir aksaklığın hatta bulunan diğer istasyonları etkilemesi engellenmiş olur.

Aynı istasyona atanması gereken işlemler kısıtı: Zaman tasarrufu, işçiden kaynaklı veya özel araç kullanımı gibi nedenler dolayısıyla bazı işlemlerin aynı iş istasyonuna atanması gerekebilir. Bu gibi durumlarda, işlemler ardışık olarak yapılır. Özel bir araç kullanarak gerçekleştirilen iki farklı işlemin aynı işçiye atanması zaman kaybını önler ve fazladan araç kullanımının önüne geçilir.

Aynı istasyona atanmaması gereken işlemler kısıtı: Bazı görevler, nitelikleri dolayısıyla aynı iş istasyonuna atanmaya uygun olmayabilir. Fiziksel güç gereksiniminin çok olduğu iki farklı görevin aynı istasyonda gerçekleştirilmesi aksaklıklara sebep olabilir. Benzer şekilde, teknolojik araç gereksinimi çok olan işlemler aynı istasyona atanmamalıdır. Örnek olarak, titreşimli işlemler ve hassas ölçüm gerektiren işlemler aynı iş istasyonuna atanamaz.

2.2.2. Montaj hattı dengelemede temel kavramlar

2.2.2.1. Görev

Üretim süreci içinde toplam iş içeriğinin mantıksal olarak bölünmüş bir parçasıdır (Erkut ve Baskak, 2003).

2.2.2.2. İş istasyonu

Bir iş istasyonu, belirli miktarda montaj görevinin gerçekleştirildiği alandır. Montaj hattı iş istasyonlarında genellikle bir işçi bulunur. Ayrıca, kısa süreli durumlar için bir işçinin birden fazla istasyona atanabildiği ve büyük boyutlu ürünlerin üretimi için bir istasyonda birden fazla işçinin bulunduğu montaj hatları da bulunmaktadır (Kilbridge ve Wester, 1961).

2.2.2.3. İş istasyonu süresi

Bir istasyonda tamamlanması gereken iş öğelerinin standart sürelerinin toplamıdır. Bir iş istasyonu süresi, en büyük iş öğesi süresinden daha az ve ürünün bir istasyonda kalması için izin verilen en büyük süreden, yani çevrim süresinden daha fazla olamaz (Erkut ve Baskak, 2003).

2.2.2.4. Toplam iş süresi

Toplam iş süresi, toplam iş içeriğini gerçekleştirmek için gerekli olan süredir. Bu süre, sadece doğrudan işçilik zamanını içerir. Doğrudan olmayan işler için harcanan zaman bu süreye dahil değildir (Kilbridge ve Wester, 1961).

2.2.2.5. Çevrim süresi

Hat standart hızda ilerlerken ardışık birimler arasında geçen süredir (Kilbridge ve Wester, 1961). Başka bir ifadeyle, bir iş istasyonundaki işçinin o istasyona atanmış olan görevleri tamamlaması için kullanabileceği süredir. Ürün, bir istasyonda en fazla çevrim süresi kadar kalabilir. Çevrim süresi, istasyon süresine eşit veya daha büyük olabilir. Çevrim süresi; üretken iş süresi, üretken olmayan iş süresi ve âtıl süre olmak üzere üçe ayrılabilir. Bir istasyonda görevleri yerine getirmek için kullanılan süre, üretken iş süresi olarak adlandırılır. Bir istasyonda gerçekleştirilen ardışık iki görev arasındaki bitiş-başlama geçiş süresi ise üretken olmayan iş süresidir. İstasyondaki tüm görevler tamamlanmış olmasına rağmen çevrim süresi dolmamış olabilir. Bu durumda âtıl süre ortaya çıkar. Çevrim süresi Denklem (2.1)'deki gibi hesaplanır (Erkut ve Baskak, 2003):

$$ÇS = \frac{TS}{ÜS} \tag{2.1}$$

$ÇS$ çevrim süresini, TS eldeki toplam süreyi ve $ÜS$ ise üretilecek ürün sayısını temsil etmektedir.

2.2.2.6. Gerekli en az iş istasyonu sayısı

İşlemlerin her iş istasyonuna çevrim süresini tümüyle ya da en az biri hariç tümüyle dolduracak şekilde atandığı düşünülürse gerekli en az iş istasyonu sayısının hesaplanması Denklem (2.2)'de verilmiştir (Erkut ve Baskak, 2003):

$$S_{en\ az} = \frac{\sum t_s}{ÇS} \quad s = [1,2,\dots,S] \tag{2.2}$$

Burada; $S_{en\ az}$ gerekli en az istasyon sayısı, $ÇS$ çevrim süresi ve t_s s. iş istasyonunun süresidir.

2.2.2.7. Ortalama iş istasyonu süresi

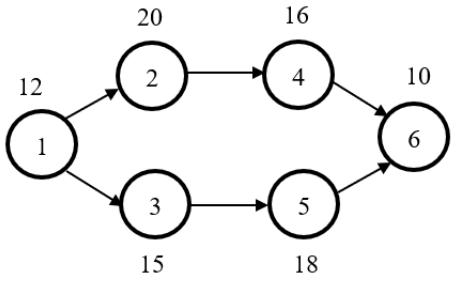
Montaj hattında bulunan iş istasyonları için ortalama süre, görevlerin işlem sürelerinin toplamının dengeleme sonunda bulunan iş istasyonu sayısına oranı şeklinde ifade edilir ve Denklem (2.3)'teki gibi hesaplanır (Erkut ve Baskak, 2003):

$$OS^* = \frac{\sum t_s}{S} \quad (S \geq S_{en\ az}\ ;\ ÇS \geq OS^*;\ \ s = [1,2,\dots,S]) \tag{2.3}$$

OS^* ortalama iş istasyonu süresi, t_s s. iş istasyonunun süresi, S dengeleme sonundaki iş istasyonu sayısı, $S_{en\ az}$ gerekli en az istasyon sayısı ve $ÇS$ çevrim süresidir.

2.2.2.8. Teknolojik öncelik diyagramı

Montajın teknik özelliklerine bağlı olarak bazı görevlerin birbirini izlemesi gerekmektedir. Bu durum, iş öğeleri arasındaki öncelik ilişkilerinin bulunmasıyla alakalıdır. Bu ilişkiler, genellikle bir grafik ile temsil edilir. Bu gösterim, öncelik diyagramı olarak adlandırılır. Bu diyagramda, aralarında öncelik ilişkisi bulunan iş öğeleri bir okla birbirine bağlı şekilde gösterilir. Okun çıktığı tarafta bulunan iş öğesi, okun uç tarafında bulunandan daha önce yapılmalıdır. Çemberlerin içinde yer alan sayılar iş öğelerinin numaralarını ve dışındakiler ise işlem sürelerini ifade etmektedir (Erkut ve Baskak, 2003). Şekil 2.1, örnek bir teknolojik öncelik diyagramını göstermektedir.



Şekil 2.1. Teknolojik öncelik diyagramı

2.2.2.9. Öncelik matrisi

Teknolojik öncelik diyagramının 0 ve 1 elemanlarından oluşan bir matris haline dönüştürülmüş halidir. Bu matriste, aralarında öncelik ilişkisi bulunan iş öğelerinden önce yapılması gereken öğenin numarasının bulunduğu satır ile sonra yapılması gereken öğenin numarasının bulunduğu sütunun kesişiminde 1 elemanı yer alır. Aralarında ilişki bulunmayan noktalarda ise 0 elemanı bulunur (Erkut ve Baskak, 2003). Çizelge 2.1'de bir öncelik matrisi verilmiştir.

Çizelge 2.1. Öncelik matrisi

	1	2	3	4	5	6
1	-	1	1	1	1	1
2		-	0	1	0	1
3			-	0	1	1
4				-	0	1
5					-	1
6						-

2.2.2.10. Düzgünlük indeksi

Montaj hattı dengesinin göreceli düzgünlüğünü gösterir. Düzgünlüğün 0'a eşit olması mükemmel dengeyi ifade eder (Kumar ve Mahto, 2013). Bu indeks, işlerin istasyonlar arasında ne kadar düzgün dağıldığını ifade eder ve Denklem (2.4) ile hesaplanır (Erkut ve Baskak, 2003):

$$D\dot{I}\ (\%) = \frac{\sqrt{\sum(t_{enb-}t_s)^2}}{S * \text{Ç}S} * 100 \quad s = [1,2,\dots,S] \tag{2.4}$$

Burada, $D\dot{I}$ düzgünlük indeksi, t_{enb} en büyük istasyon süresi, t_s $s.$ iş istasyonunun süresi, S dengeleme sonundaki iş istasyonu sayısı ve $\text{Ç}S$ çevrim süresidir.

2.2.2.11.Hat etkinliği

Montaj hattında bulunan iş istasyonlarının toplam sürelerinin ne kadarının iş öğeleri işlemlerine ayrıldığını gösteren bir performans ölçütüdür. Hat etkinliğinin %100 olduğu bir durumda, istasyonların hiçbirinde âtıl zaman ortaya çıkmamaktadır (Kılınçcı, 2004). Başka bir ifadeyle, iş istasyonlarındaki toplam etkin sürenin montaj süresine oranına hat etkinliği denir ve Denklem (2.5) ile hesaplanır (Erkut ve Baskak, 2003):

$$HE\ (\%) = \frac{\sum ES_s}{S * ÇS} * 100 \quad s = [1,2,\dots,S] \tag{2.5}$$

HE hat etkinliği, ES_s s. iş istasyonunun etkin süresi, S ise dengeleme sonundaki iş istasyonu sayısı ve $ÇS$ çevrim süresidir.

2.2.2.12. Kuramsal etkinlik

Montaj hattının belirli bir çevrim süresi için en az sayıda iş istasyonu ile çalıştırılmasıyla elde edilen etkinliktir. Kuramsal etkinlik, hat verimliliğinin üst sınırı olup iş öğelerinde düzenlemeler yapılmasıyla artırılabilir. Kuramsal etkinliğin hesaplanması Denklem (2.6)'daki gibidir (Erkut ve Baskak, 2003):

$$KE\ (\%) = \frac{\sum ES_s}{S_{en\ az} * ÇS} * 100 \quad s = [1,2,\dots,S] \tag{2.6}$$

KE kuramsal etkinlik, ES_s s. iş istasyonunun etkin süresi, $S_{en\ az}$ gerekli en az istasyon sayısı ve $ÇS$ çevrim süresidir.

2.2.2.13. Esneklik oranı

Bir problemden üretilebilecek uygun sıralama sayısını ifade eden bir ölçüttür. Bir problemin öncelik yapısı bu oran ile değerlendirilebilir. Esneklik oranı Denklem (2.7)'deki gibi hesaplanır (Erel ve Sarin, 1998).

$$EO = \frac{2 * Y}{N * (N - 1)} \quad (2.7)$$

Burada; EO esneklik oranı, Y ise öncelik matrisinde "0" değerine sahip eleman sayısı ve N görev sayısıdır.

2.2.2.14. Sıra kuvveti

Belirlenen öncelik ilişkilerinin izin verdiği farklı sıralamaların hacmini ölçer ve Denklem (2.8) ile hesaplanır (Erel ve Sarin, 1998):

$$SK = \frac{2 * \text{Ö}S}{N * (N - 1)} \quad (2.8)$$

Burada; SK sıra kuvveti, ÖS öncelik ilişkileri sayısı ve N görev sayısıdır.

2.2.2.15. Denge kaybı

Montaj hatlarında, görevlerin işçiler veya istasyonlar arasında düzgün olmayan bir şekilde dağıtılmasından ötürü âtıl zaman oluşur, buna denge kaybı ismi verilir. Hat dengelemenin amacı, verilen koşullar için mümkün olan en düşük denge kaybını sağlamaktır (Kilbridge ve Wester, 1961).

Denge kaybının 0 olması ideal durumdur. Bu değer, her istasyonda birim üretime ayrılan toplam süre ile gerekli sürenin farkının montaj süresine oranlanması ile hesaplanır ve genellikle 0'dan büyük bir değer olur. Denge kaybı Denklem (2.9) ile hesaplanabilir (Erkut ve Baskak, 2003):

$$DK\ (\%) = \frac{\text{Ç}S - OS^*}{\text{Ç}S} * 100 = \frac{S * OS^* - \sum t_s}{S * \text{Ç}S} * 100 \quad s = [1,2,\dots,S] \quad (2.9)$$

Burada; DK denge kaybını, ÇS çevrim süresini, OS^* ortalama iş istasyonu süresini, S iş istasyonu sayısını ve t_s s. iş istasyonunun süresini temsil etmektedir.

2.2.2.16. Darboğaz

Üretim hızını yavaşlatan aktarmada gecikme durumudur. Montaj hattı dengelenerek bu durumun üstesinden gelinebilir (Kumar ve Mahto, 2013).

2.2.2.17. Güvenlik düzeyi

Çevrim süresi içinde iş istasyonlarına atanmış olan işlerin tamamlanabilme olasılığının düzeyine verilen isimdir (Erkut ve Baskak, 2003).

2.2.2.18. Ürün karışımı

Çok modelli üretimin yapıldığı montaj hatlarında, modellerin üretim oranını ifade eder (Erkut ve Baskak, 2003).

2.2.3. Montaj hatlarının sınıflandırılması

Bu bölümde, montaj hatlarının; model sayısına, istasyon yerleşimine, kontrol yapısına, kurulum sıklığına ve otomasyon seviyesine göre sınıflandırılması açıklanmıştır.

2.2.3.1. Model sayısına göre montaj hatları

Tek modelli montaj hatları: Tek tip ürünün yüksek miktarda üretiminin yapıldığı montaj hatlarıdır. Kurulum süreleri ihmal edilebilir olan ve işlem süreleri açısından birbirine yakın birden fazla ürünün montajının gerçekleştirildiği hatlar da bu gruba dahil edilmektedir. Tek modelli hatlara örnek olarak kompakt disk ve meşrubat şişesi üretimi verilebilir (Boysen vd., 2008).

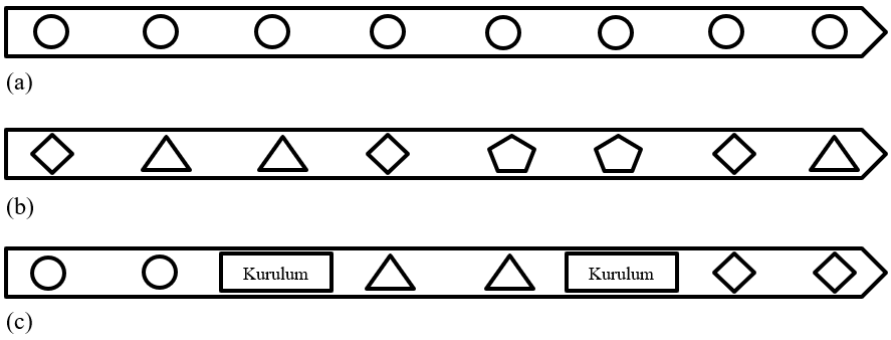
Karma modelli montaj hatları: Temel ürünün, bazı özelleştirilebilir özellikleri açısından farklı çeşitlerinin üretildiği hatlardır. Farklı seçeneklerin işlem süreleri de birbirinden farklı olduğundan istasyon süreleri büyük ölçüde montajı yapılacak modele bağlıdır. Örneğin, bir otomobil için manuel bir açılır tavan montajı yapmak, elektrikli türüne

göre farklı bir işlem süresi gerektirir (Boysen vd, 2008). Karma modelli üretimin yapıldığı hatlarda, üretim sırasının belirlenmesi oldukça önemlidir (Merengo vd., 2010).

Karma modelli hatlarda, benzer özelliklere sahip farklı modellerin montaj hattında aynı anda üretilebilmesi sayesinde müşteri isteklerini karşılamak kolaylaşır. Karma modelli hatlar, bitmiş ürün envanterini azaltma konusunda da avantajlıdır. Bununla birlikte, bu hatların bazı dezavantajları bulunmaktadır. Modellerin işlem süreleri arasındaki farklılıkların iş akışında meydana getirdiği düzensizlikler, istasyonlarda âtıl süreleri ve yarı mamul stoklarında artışı beraberinde getirmektedir (Karaca, 1996). Yatırım maliyetleri yüksek olan bu hatlarda, düşük kapasite kullanımı ve verimsiz çalışma maliyeti de oldukça yüksektir. (Yağmahan ve Emel, 2015).

Çok modelli montaj hatları: Farklı ürünlerin veya bir ürünün değişik modellerinin partiler halinde ve değişik zamanlarda montajının yapıldığı hatlardır. Bir modelin üretiminin tamamlanmasının ardından diğer ürünün montajına geçilmektedir. Farklı bir modelin üretimine başlanmadan önce montaj hattında o modelin gerektirdiği düzenlemeler yapılır (Karaca,1996).

Şekil 2.2.'de model sayısına göre montaj hatları gösterilmektedir.



Şekil 2.2. Model sayısına göre montaj hatları: a) tek modelli montaj hattı, b) karma modelli montaj hattı, c) çok modelli montaj hattı.

2.2.3.2. İstasyon yerleşimine göre montaj hatları

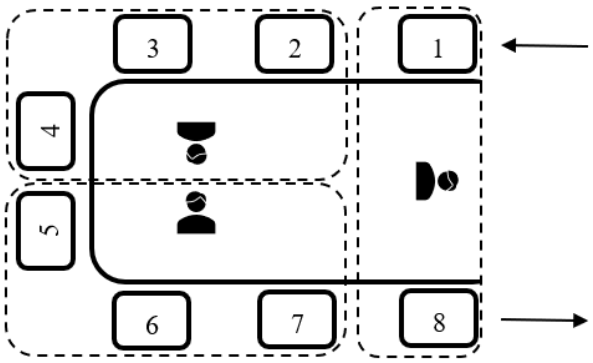
Düz montaj hatları: İş istasyonlarının montaj hattı boyunca ardı ardına seri şekilde sıralandığı hatlardır. Ürün, montaj işlemlerinin yapılması için ilk istasyondan hatta girer ve

tamamlanmış şekilde son istasyondan hattı terk eder. Bu hatlarda, iş akışı kolay ve hızlıdır. Ancak, kapladığı alan açısından dezavantajlıdır (Özgörmüş, 2007). Montaj hattı üretim sistemleri için ilk uygulama, düz hatlar üzerinde başlamıştır (Ağpak vd., 2002). Şekil 2.3, 4 iş istasyonuna sahip düz bir montaj hattını göstermektedir.



Şekil 2.3. Düz bir montaj hattı

U-tipi montaj hatları: İş istasyonlarının yerleşiminin U şeklinde olduğu ve hattın giriş ve çıkışının aynı pozisyonda bulunduğu hatlardır. Bu hatlar, Toyota'nın Tam Zamanında Üretim prensiplerini uygulamak amacıyla montaj hattını U şeklinde tasarlamasıyla ortaya çıkmıştır (Ağpak vd., 2002). Toyota firmasında, aynı tür otomobillerin farklı özelliklere sahip çeşitleri üretilmekte ve her otomobil için talep sürekli olarak dalgalanmaktadır. Buna bağlı olarak, atölyelerde iş yükü açısından değişkenlikler görülmektedir. Bu yerleşim tipi, talep değişimlerine ayak uydurabilmek adına atölyedeki işçi sayısı açısından esneklik sağlamaktadır (Özgörmüş, 2007). Şekil 2.4'te 8 istasyona sahip U-tipi bir montaj hattı verilmiştir.

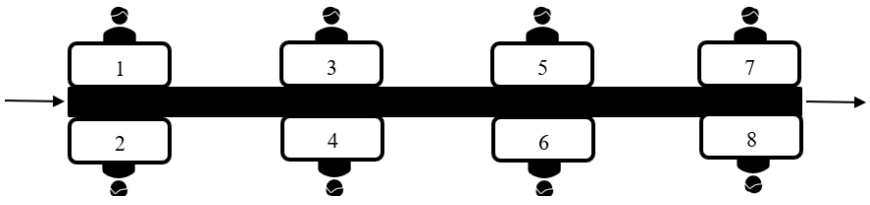


Şekil 2.4. U-tipi bir montaj hattı

U-tipi hatların düz hatlara kıyasla sağladığı avantajlar bulunmaktadır. Bunlar aşağıdaki gibi açıklanabilir (Miltenburg ve Wijngaard, 1994: Özgörmüş'ten (2007)):

- U-tipi montaj hatlarında çalışan işçiler arasındaki iletişim, düz hatlarda çalışanlara kıyasla daha güçlüdür. Buna bağlı olarak, ortaya çıkabilecek sorunlar karşısında iş birliği açısından U-tipi hatlar daha avantajlıdır.
- U-tipi hatlarda çalışan işçiler; çeşitli görevleri yerine getirebilen, birçok görev hakkında bilgi sahibi ve üretim hakkında bütünsel bilgiye sahip çok fonksiyonlu işçi niteliği kazanırlar.
- U-tipi hatlar, talep değişimlerine uyum sağlamak için işçi sayısını artırıp azaltma konusunda kolaylık sağlar.
- Belirli bir üretim hacmi için U-tipi montaj hatlarında düz hatlara göre daha az veya eşit sayıda istasyona ihtiyaç duyulur.

Çift taraflı montaj hatları: Montaj hattının hem sağ hem sol tarafının kullanılarak montaj işlemlerinin gerçekleştirildiği hatlardır. Çift taraflı montaj hatlarında, iş istasyonları hattın her iki tarafına paralel şekilde yerleştirilir. Otobüs, kamyon, otomobil gibi büyük boyutlu ürünlerin montajı bu hatlarda gerçekleştirilebilir. Bu gibi ürünlerin montaj işlemlerinden bazıları ürünün sadece solunda veya sağında yapılabilir. Örneğin, bir otomobilin sağ kapısının montajı sağ taraftan yapılmaya uygundur. Çift taraflı hatlar, ürünün montajının hattın her iki tarafında çalışan işçiler tarafından eşzamanlı bir şekilde gerçekleştirmesini sağlar (Mete ve Ağpak, 2013). Çift taraflı montaj hatlarında, birbirine bakan istasyonlar eş istasyon olarak adlandırılır ve bu istasyonlardan her birine ise parça istasyon denir (Kim vd., 2009). Şekil 2.5'te çift taraflı bir montaj hattı verilmiştir.

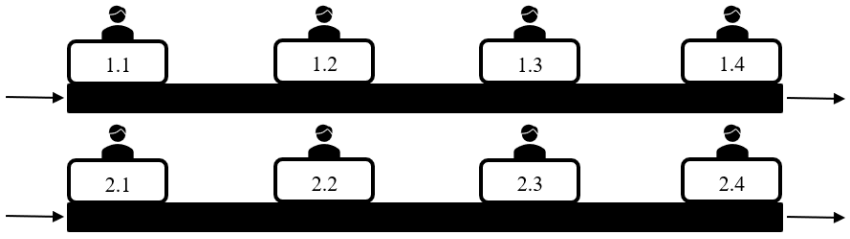


Şekil 2.5. Çift taraflı bir montaj hattı

Çift taraflı montaj hatları aşağıdaki avantajları sağlar (Bartholdi, 1993: Kim vd.'den (2009)):

- Montaj hattını kısaltır.
- Montaj hatlarında alan kazandırır.
- Alet ve demirbaş maliyetini azaltır.
- İş yapma süresini kısaltır.
- Malzeme elleçlemeyi azaltır.

Paralel montaj hatları: Paralel montaj hatlarında, bir veya daha fazla ürün çeşidinin montajı birbirine paralel konumlandırılmış iki veya daha fazla hat üzerinde gerçekleştirilir (Özcan vd., 2010). Talebin yüksek olduğu durumlarda, üretim ortamlarında paralel şekilde birden fazla montaj hattının bulunması yaygın şekilde görülmektedir. Birbirine paralel hatların bulunması, hat uzunluğunun kısalması açısından avantajlıdır. Bir istasyonda sorun çıkması durumunda sadece bir hat çalışmayı durdurur, buna paralel diğer hatlar ise çalışmaya devam eder. Ancak, paralel hatlar daha çok alet ve ekipman gerektirebilir (Gökçen vd., 2006). Şekil 2.6 paralel iki montaj hattını temsil etmektedir.



Şekil 2.6. Paralel montaj hatları

Paralel hatlar kullanılmasının avantajları aşağıdaki gibidir (Özcan vd., 2010):

- Aynı ürünün farklı modellerinin veya benzer ürünlerin paralel komşu hatlarda üretilmesini sağlar.
- Âtıl zamanı azaltır ve montaj hatlarının verimliliğini artırır.
- Her hat farklı bir çevrim süresi ile çalışabilir.
- Operatörler arasındaki gözlenebilirlik ve iletişim becerilerini geliştirir.
- İşçi gereksinimini azaltır.

2.2.3.3. Hattın kontrol yapısına göre montaj hatları

Gecikmesiz montaj hatları: Tüm istasyonların aynı çevrim süresine sahip olduğu hatlardır. Her çevrim süresi sonunda, istasyonlar görevlerini tamamladıktan sonra montajı yapılmakta olan ürün bir sonraki istasyona aktarılır. Çevrim süresi dolmadan görevlerini tamamlayan istasyonlar olsa da parçanın bir sonraki istasyona geçmesi için bu sürenin dolması beklenir. Bu hatlarda, parçaların istasyonlar arası hareketi genellikle periyodik veya sürekli olarak hareket eden bir konveyör bant ile gerçekleştirilir. Hat, konveyörün sürekli hareket etmesi durumunda çevrim süresine göre belirlenen sabit bir hıza sahip olur. Çevrim süresinin en küçük değeri iş istasyonlarının en büyük iş yüküne göre belirlenir. Gecikmesiz hatlarda, çevrim süresi dolmadan veya konveyör bir istasyondan diğerine hareket etmeden önce her bir istasyondaki görevler tamamlanmaya çalışılır. Herhangi bir istasyonun iş yükü ve çevrim süresi arasındaki fark, istasyonun görev süresinde meydana gelen rassal değişimleri telafi edebilme derecesini gösterir. Bu türde bir montaj hattının üretim hızı, hattın çevrim süresine bağlıdır. Bu hatlarda, istasyonlar arasında tampon stoklar kullanılmamaktadır (Saif vd., 2014).

Gecikmeli montaj hatları: Gecikmeli montaj hatlarında, istasyonlardaki işlemlerin bitmesinin ardından montajı yapılan parçalar sonraki istasyona aktarılır. Kısacası, görevlerini tamamlamış istasyonların önceden belirlenmiş süre kadar beklemesi gerekmez. Bu hatlar, işlem sürelerinin rassal değişimlerden etkilenmesi durumunda sıkça kullanılmaktadır (Boysen vd., 2006).

Gecikmeli hatlar, parçaların istasyonlar arası hareketleri bakımından senkronize ve asenkronize gecikmeli montaj hatları olmak üzere iki gruba ayrılır. Senkronize gecikmeli hatlarda, belirlenmiş bir sürenin ardından montajı yapılmış parçalar tüm istasyonlardan eşzamanlı olarak taşınır ve buna bağlı olarak tampon stoklar kullanılmaz. Senkronize gecikmeli hatlar, işlem sürelerinin deterministik olduğu durumlarda gecikmesiz hatlarla benzerlik göstermektedir.

Asenkronize gecikmeli montaj hatlarında ise her istasyon kendi iş yüküne eşit bir çevrim süresine sahiptir. Dolayısıyla, her istasyonun çevrim süresi birbirinden farklıdır. Bu hatlarda, bir istasyonda görevler tamamlanmış iken bir sonrakinde henüz tamamlanmamış

olabilir. Böyle bir durumda, görevleri tamamlanmış olan istasyon diğerini beklemek durumunda kalır. Bu durumun tam tersi de mümkündür. Bir istasyon, kendisine atanan görevleri bitirse dahi bir önceki iş istasyonunda işlemler devam ediyor olabilir. Bu durumda ise işlemlere başlayabilmek için önceki istasyondan gelecek parçaların beklenmesi gerekir. İstasyonların âtıl zamanlarının azaltılması amacıyla genellikle tampon stoklar kullanılır (Saif vd., 2014).

2.2.3.4. Kurulum sıklığına göre montaj hatları

İlk kez dengeleme: Bir montaj hattı kurulacağı zaman, üretim süreci kesinleşmemiştir veya tam anlamıyla doğrulanmamış haldedir. Sadece üretimi yapılacak ürünün özellikleri ve bunun çeşitleri belirlenmiştir. Hattın kurulumu, ürün göz önünde bulundurularak yapılır. Bu amaçla, ürüne ait görevler arasındaki öncelik ilişkileri belirlenir ve alternatif üretim seçenekleri incelenir. Farklı makine ve işçilerin görev gerçekleştirme performansları ve maliyetleri göz önünde bulundurularak maliyetin en az olmasını sağlayacak şekilde hattın kurulumu gerçekleştirilir (Boysen vd., 2008).

Yeniden dengeleme: İşletmelerin üretim programında meydana gelen birtakım önemli değişimler dolayısıyla montaj hatlarının yeni duruma uyum sağlayacak şekilde yeniden dengelenmesi gerekebilir. İş istasyonları önceden mevcut olduğundan iş istasyonu sayısını enküçüklemenin önemi azdır. Hattın çevrim süresi de genellikle talep tahminlerine göre belirlendiğinden verilen çevrim süresi ve iş istasyonu sayısı için olurlu bir çözüm elde etmek ve hattı yeniden dengelemek yeterli olmaktadır. Ayrıca, işlemlerin istasyonlar arasında mümkün olabildiğince eşit dağıtılması da ilave bir amaç olarak benimsenebilir (Boysen vd., 2008).

Üretim ortamlarında çeşitli sebeplerden ötürü yeri değiştirilemeyen makineler bulunabilir. Bu gibi durumlarda, taşınamayan makineleri gerektiren görevlerin farklı istasyonlara atanamaması gibi bazı özel kısıtlar görülebilmektedir (Purnomo vd., 2013).

2.2.3.5. Otomasyon seviyesine göre montaj hatları

Manuel montaj hatları: İşlemlerin, istasyonlara atanmış olan işçiler tarafından gerçekleştirildiği hatlardır. Manuel işlemler, işçiler tarafından gerçekleştirildiğinden işlem süreleri sabit değil, rassal değişkenlerdir. Gecikmesiz veya hareketli hatlarda, bir işçi belirlenen süre içinde kendisine atanan görevi tamamlayamazsa o görevi bitirmek için tekrar üzerinde çalışması gerekir. Buna bağlı olarak, üretim maliyeti yükselir. Gecikmeli hatlarda ise üretim kapasitesini etkileyen tıkanma veya boşta kalma durumları görülebilir. Hatlar, bu sorunların ortaya çıkma olasılığını en aza indirecek şekilde tasarlanmalıdır (Merengo vd., 2010).

Manuel hatlarda, bazı işçiler belirli işlemler konusunda uzman ve diğer bazı işlemleri gerçekleştirmede yetersiz olabilir (Saif vd., 2014). İşlem sürelerinin rassal sapmalar göstermesinin diğer bir sebebi ise motivasyon, çalışma ortamı, stres gibi etmenler sebebiyle işçi performansının etkilenmesidir (Tempelmeier, 2003; Boysen vd.'den, (2008)).

Otomatik montaj hatları: Parçaların istasyonlar arası hareketinin otomatik olarak gerçekleştirildiği hatlardır. İstasyonlar arası parça aktarımı, mekanik veya mekanik olmayan olmak üzere iki şekilde yapılabilir. Mekanik hatlarda, hareketli konveyörler veya benzer malzeme aktarma sistemleri kullanılır. Mekanik olmayan hatlarda ise parçalar bir istasyondan diğerine elle geçer. Otomatik hatlar, çalışma ortamının insan sağlığı açısından tehlikeli olduğu durumlar için tasarlanmıştır. Ayrıca, işgücü maliyetlerinin yüksek olduğu durumlarda da otomatik hatlar ve robotlar kullanılmaktadır. Bu hatlarda işlem süreleri arasındaki değişkenlik az olduğundan manuel hatlara nazaran gecikmeler ve âtıl zamanlar açısından daha avantajlıdır (Özgörmüş, 2007).

2.2.4. Montaj hattı dengeleme problemi

Montaj hatları, genellikle üretim sürecinin son aşamasıdır ve sürecin genel performansını önemli derecede etkilemektedir. Malzeme akışı hattın ihtiyacına göre ayarlandığından, montaj hatlarının işletmenin performansı üzerinde kritik etkisinin bulunduğu söylenebilir. Hattın tasarımında ve talep değişkenliğine göre üretim hızının

yeniden ayarlanması noktasında MHD problemi ortaya çıkmaktadır (Bryton, 1954; Aksoy vd.'den, (2014)).

Montaj hatlarının dengelenmesi; üretim hızının arttırılması, planlamanın sağlıklı bir şekilde yapılması ve işletmelerin ekonomik sorunlarına çözüm getirmesi açısından oldukça önemlidir (Alp vd., 2001; Aksoy vd.'den, (2014)). Bir montaj hattının tam dengelendiği bir durumda, üretim akışı gecikmesiz olurken istasyonlar eşit miktarda görev gerçekleştirir (Aksoy vd., 2014).

MHD problemleri, NP-zor yapıda problemlerdir. Buna bağlı olarak, görev sayısı veya görevler arası öncelik ilişkilerinde meydana gelebilecek her artış, çözüm uzayını üstel olarak artırmakta ve dolayısıyla optimum hat dengesini bulmayı zorlaştırmaktadır. Bununla birlikte, günümüz bilgisayarlarının saniyedeki işlem yapabilme kapasitelerinin oldukça yüksek olması sayesinde sadeleştirilmiş bir model ile gerçek hayattaki birçok MHD problemini makul süreler içerisinde optimum olarak çözmek mümkündür (Doğan ve Sakallı, 2016).

MHD, literatürde ilk kez Bryton'un (1954) yüksek lisans tezinde ele alınan bir kavramdır. Bu çalışmada, belirli iş istasyonu sayısı için çevrim süresini enküçüklemeyi amaçlayan bir sezgisel sunulmuştur. MHD probleminin tanımı ise Salveson (1955) tarafından yapılmıştır. Salveson (1955), problemi çözmek için tamsayılı programlama modeli geliştirmiştir. Bu modelde, belirli çevrim süresi için iş istasyonu sayısı ve istasyonlar arasındaki âtıl zamanların enküçüklenmesi amaçlanmıştır.

1956-1961 yılları arasında, Jackson (1956), Bowman (1960), Supnick ve Solinger (1960), White (1961) ve Hu (1961) gibi çalışmalar, MHD'nin bir araştırma alanı olması konusunda temel katkıları sağlamıştır (Pearce, 2015).

Jackson (1956), MHD problemi için aşamalı sıralamayla çözüm yöntemini önermiştir. Çalışmada, her bir istasyona atanmış olan görevlerin toplam işlem sürelerinin çevrim süresini aşmaması ve görevlerin öncelik ilişkilerine uygun sırada gerçekleştirilmesi olmak üzere iki kısıta yer verilmiştir.

Bowman (1960), MHD probleminin çözümü için tamsayılı programlama yaklaşımını önererek iki farklı doğrusal model sunmuştur. İstasyon sayısının enküçüklenmesinin amaçlandığı bu çalışmada, görevlerin iş istasyonları arasında bölünmesini engelleyen görevlerin bölünmezliği kısıtı dikkate alınmıştır.

Supnick ve Solinger (1960), çok sayıda benzer ürünün üretildiği ve görevlerin gruplara ayrılarak her bir görev grubundaki tüm işçilerin aynı görevleri tekrarlı olarak gerçekleştirdiği bir üretim ortamını ele almıştır. Aynı gruptaki işçilerin görevleri aynı anda gerçekleştirdiği ve ardışık iş gruplarının birbiri arasında belirli süreler geçmesinin ardından çalışmaya başladığı varsayılmıştır. Çalışmada, süreç içi stokların enküçüklenmesi için hat çalışmaya başladıktan sonra görev gruplarının hangi zamanlarda çalışmaya başlayacağının belirlenmesi problemi formüle edilmiştir.

White (1961), MHD probleminin çözümü için Bowman (1960) çalışmasında önerilen ilk tamsayılı programlama modelini modifiye ederek 0-1 karar değişkenlerini tanımlamıştır.

Hu (1961), sıralama kısıtları bulunan montaj görevlerinin eşit yeteneğe sahip ve tüm görevleri gerçekleştirebilen işçiler tarafından yapıldığı bir üretim ortamı için paralel görev çizelgelemesini yönlü çevrimsiz grafik çizelgeleme problemi olarak formüle ederek öncelik grafiklerinin paralel işlenmesini incelemiştir. Çalışmada, belirli bir süre içinde tüm görevlerin gerçekleştirilebilmesi için en az işçi sayısını gerektiren çizelgenin ve belirli sayıda işçi uygun olduğunda tüm görevlerin en kısa sürede tamamlanmasını sağlayan bir çizelgenin oluşturulması ele alınmış ve optimal bir çizelgeleme algoritması önerilmiştir.

MHD probleminin ilk ele alındığı dönemlere kıyasla teknolojik alanda meydana gelen gelişmeler yeni MHD problemi türlerinin ortaya çıkmasını sağlamış ve MHD problemleri araştırmacılar tarafından farklı şekillerde sınıflandırılmıştır.

Baybars (1986), Becker ve Scholl (2006), Scholl ve Becker (2006), Boysen vd. (2006) ve Boysen vd. (2008) çalışmalarında MHD problemlerini basit MHD (BMHD) ve genel MHD (GMHD) problemleri olmak üzere iki gruba ayırmıştır.

MHD problemleri için yapılan başka bir sınıflandırma Ghosh ve Gangnon'a (1989) aittir. Araştırmacılar; MHD problemini model sayısı açısından tek-çok/karışık modelli, işlem sürelerinin doğası açısından deterministik ve stokastik, varsayım özellikleri açısından ise basit ve genel olarak sınıflandırmıştır. Benzer bir sınıflandırma, Erel ve Sarin (1998) tarafından tek-karışık/çok modelli ve deterministik-stokastik MHD problemleri şeklinde yapılmıştır. Sivasankaran ve Shahabudeen (2014) ise Erel ve Sarin (1998) tarafından yapılan sınıflandırmaya düz-U-tipi hatlar olmak üzere farklı bir kategori eklemiştir.

MHD problemleri, literatürde yapılan sınıflandırılmaları temel alarak varsayım özellikleri, amaç sayısı, model sayısı, işlem sürelerinin doğası ve hat yerleşimi gibi çeşitli faktörler göz önünde bulundurularak gruplandırılabilir.

2.2.4.1. Varsayım özelliklerine göre MHD problemleri

Varsayım özellikleri açısından MHD problemleri, basit ve genel olmak üzere ikiye ayrılır. BMHD, çözümü daha basit hale getiren bazı teknolojik kısıtlar ve ilişkiler içeren MHD problemi türüdür. Problemin çözümünü kolaylaştıran bazı temel varsayımlar aşağıdaki gibidir (Baybars (1986); Çavdur vd.'dan (2018)):

- Tüm girdi parametreleri bilinmektedir.
- Bir görev, sadece bir iş istasyonunda gerçekleştirilebilir.
- Görevler, teknolojik ve organizasyonel kısıtlar sebebiyle keyfi bir sırada yapılamaz.
- Tüm görevler gerçekleştirilmelidir.
- Tüm iş istasyonlarında her görevi gerçekleştirebilecek ekipman ve donanım bulunmaktadır.
- Her görev herhangi bir istasyonda gerçekleştirilebilir.
- Tüm hat, besleyici veya paralel alt montaj hatlı olmayacak şekilde seri olarak düzenlenmelidir.
- Çevrim süresi verilmiştir ve sabittir (BMHD tip-1 için).
- İş istasyonu sayısı verilmiştir ve sabittir (BMHD tip-2 için).

BMHD problemi, amaç fonksiyonları açısından tip1-, tip-2, tip-E ve tip-F olmak üzere dörde ayrılmaktadır (Scholl ve Becker, 2006):

- BMHD tip-1: Verilen çevrim süresi için iş istasyonu sayısının enküçüklenmesidir.
- BMHD tip-2: Verilen iş istasyonu sayısı için çevrim süresinin enküçüklenmesidir.
- BMHD tip-E: En genel problem türüdür. Hat etkinliğinin enbüyüklenmesi, yani çevrim süresi ve iş istasyonu sayısının enküçüklenmesidir.
- BMHD tip-F: Verilen çevrim süresi ve iş istasyonu sayısı kombinasyonu için olurlu bir hat dengesinin bulunup bulunmadığıyla ilgili uygunluk problemidir.

GMHD problemi, BMHD probleminin bazı varsayımlarının kaldırılması ve genelleştirilmesiyle ortaya çıkar. Bu sınıfta yer alan problemler; ekipman seçimi, işleme alternatifleri ve atama kısıtları gibi pratikte görülen oldukça geniş çaptaki problem uzantılarını içerir (Kriengkorakot ve Pianthong, 2007).

GMHD probleminin BMHD problemine göre daha az varsayım içermesi, GMHD probleminin gerçek hayatta uygulanabilir olmasını mümkün kılmaktadır. Örneğin, montaj hattının düz yerine U-tipi, paralel veya çift taraflı olması, talep ve işlem süreleri açısından belirsizlik bulunması veya hatta karma veya çok modelli üretimin yapılması gibi bazı değişikliklerin yapılmasıyla GMHD sınıfına giren yeni problem türleri ortaya çıkmıştır (Saif vd., 2014).

2.2.4.2. Amaç sayısına göre MHD problemleri

MHD problemleri, optimize edilmek istenen performans ölçütü sayısı açısından tek amaçlı ve çok amaçlı olmak üzere ikiye ayrılmaktadır. Çok amaçlı problemler, aynı anda birden fazla amacın eniyilenmeye çalışılması sebebiyle tek amaçlı problemlere nazaran daha karmaşık yapıdadır.

2.2.4.3. Model sayısına göre MHD problemleri

Model sayısı açısından montaj hatlarında; tek modelli, karma modelli ve çok modelli olmak üzere üç farklı şekilde üretim yapılabilir. Tek modelli hatlarda sadece bir modelin

üretimi gerçekleştirilir. Karma modelli hatlarda, bazı özellikler bakımından farklılıklar gösteren temel bir ürünün birkaç farklı çeşidi üretilir. Bu hatlarda, modeller arasında kurulum yapmak gerekmez. Aksine, çok modelli hatlarda modellerin üretimi partiler halinde gerçekleştirildiğinden bir modelin üretimi tamamlanmadan diğerine geçilmez ve modeller arası geçişlerde kurulum yapmak gerekir.

2.2.4.4. İşlem sürelerinin doğasına göre MHD problemleri

İşlem sürelerinin doğası bakımından MHD problemleri; deterministik, stokastik ve bulanık olmak üzere üç gruba ayrılabilir. İşlem sürelerinin deterministik olması, bu sürelerin önceden bilindiğini ve herhangi bir değişkenlik görülmediğini ifade eder. Stokastik MHD probleminde ise bu süreler belirli bir olasılık dağılımına göre değişkenlik göstermektedir. Bulanık MHD probleminde, bir montaj hattında işlem sürelerinin deterministik olarak ifade edilemediği, uzun tecrübeler sonucunda bazı işlemlerin sistem dışı etkenler sebebiyle standartlaştırılamadığı ve genellikle üçgensel sayılar çerçevesinde gerçekleştiği değerlendirilmektedir (Doğan ve Sakallı, 2016).

2.2.4.5. Hat yerleşimine göre MHD problemleri

MHD problemleri, montaj hattının tesis içindeki yerleşim şekline göre düz, U-tipi, paralel ve çift taraflı MHD problemleri olarak sınıflandırılabilir. I-tipi, iş istasyonlarının düz bir sırayla dizildiği hatları ifade eder. U-tipi hatlarda, istasyonlar "U" şeklinde sıralanmış haldedir. Paralel hatlar, birden fazla montaj hattının birbirinden bağımsız olarak paralel şekilde dizildiği türdür. Çift taraflı hatlarda ise hattın her iki tarafında karşılıklı olarak işlemlerin gerçekleştirildiği eş istasyonlar bulunmaktadır. Bu hatlar, genellikle büyük boyutlu ürünlerin montajında kullanılır.

2.2.5. Montaj hattı işçi atama ve yeniden dengeleme problemi

MHD problemleri, hattın kurulum aşamasında ilk kez dengelenmesini ele almaktadır. Ancak, gerçek üretim ortamlarında iş istasyonlarında meydana gelen arıza veya kapanmalar, yeni talep gelişleri ve teknolojik şartnamede değişiklikler gibi durumlarla karşılaşılmaktadır. Bu tür durumlar, ilk dengeleme planının uygulanmasını zorlaştırabilir ve hattı yeniden

dengeleme ihtiyacı doğurabilir (Sancı ve Azizoğlu, 2017). Buna bağlı olarak, yeni kurulmuş bir üretim sisteminde montaj hatlarının ilk defa dengelenmesinden ziyade mevcut hatların çeşitli sebeplerden ötürü yeniden dengelenmesi durumuyla daha sık karşılaşılmaktadır (Falkenauer, 2005).

Hattın çeşitli sebeplere bağlı olarak yeniden dengelenmesi faaliyeti MHYD olarak ifade edilmektedir. Literatürde MHYD problemi konusunda bulunan çalışmalar, işlem sürelerinin sabit ve işçiden bağımsız olduğu varsayımına dayanmaktadır. Bu sebeple, sadece görevlerin iş istasyonlarına yeniden atanması gerçekleştirilir. Ancak, işçiler arasındaki performans farklılıklarının göz önünde bulundurulması, gerçek hayat problemlerine daha gerçekçi bir yaklaşım sunmaktadır.

MHD alanında, işçi atamalarının da probleme dahil edildiği problem türü MHİAD problemidir. Fakat, bu problemde yeni kurulan ve daha önce dengelenmemiş montaj hatlarının ilk kez dengelenmesi ele alınmaktadır.

Bu çalışmada, montaj işlemleri konusunda farklı performanslara sahip işçilerin bulunduğu montaj hatlarında bir veya daha fazla iş istasyonunun kapanması durumunda görev atamalarının yanı sıra işçi atamalarının da gerçekleştirildiği MHİAYD problemi tanımlanmıştır. Bilindiği kadarıyla literatürde, iş istasyonu sayısının değişimine bağlı olarak işçi yeteneklerini göz önünde bulunduracak şekilde hattın yeniden dengelendiği bir çalışma bulunmamaktadır.

İzleyen bölümde, MHYD ve MHİAD problemlerinin birleşimi olan MHİAYD probleminin daha iyi anlaşılabilmesi için MHYD ve MHİAD konularında literatürde yer alan çalışmalar özetlenmiştir.

3. LİTERATÜR ARAŞTIRMASI

3.1. Literatürde Montaj Hattı Yeniden Dengeleme Problemi

Montaj hatlarının yeniden dengelenmesi; ürün taleplerinin veya iş istasyonlarının açılması/kapanması veya aletlerin yer değiştirmesi gibi durumlara bağlı olarak hat dengesinin değişimi ile ilgilidir (Grangeon vd., 2011). Meydana gelen bazı değişimler dolayısıyla mevcut hattın yeniden dengelenmesi, kullanılan kaynakların yeniden tahsis edilmesini ve bileşen yönetiminde değişiklik yapılmasını gerektirmektedir (Makssoud vd., 2015).

Manuel bir montaj hattı yeniden dengelendiğinde, işçilerin verimlilik kaybı ve kalite sorunlarına karşın kendilerine atanan yeni görevler için tekrar eğitilmesi gerekir. Genel olarak eğitim maliyeti, yeniden atanan görev sayısı ve atanan görevlerin zorluk derecesine bağlıdır (Yang vd., 2013). Bunun yanında, hat dengesinde değişiklik meydana geldiğinde kalite güvencesi, ekipman değiştirme ve taşıma gibi maliyetler de ortaya çıkmaktadır (Gamberini vd., 2009).

Literatürde MHYD problemi konusunda yapılan çalışmalar, stokastik ve deterministik görev süreli olmak üzere iki sınıfa ayrılmaktadır.

3.1.1. Stokastik görev süreli montaj hattı yeniden dengeleme problemi

Stokastik MHYD problemini çok amaçlı olarak ele alan Gamberini vd. (2006) ve Gamberini vd. (2009) çalışmalarında, toplam işçilik ve beklenen tamamlanmama maliyetlerinin ve istasyonu değişen görev sayısının enküçüklenmesini amaçlanmıştır. Bu çalışmalarda, girdi parametrelerinde (ürün özellikleri ve çevrim süresi) meydana gelen değişimlere bağlı olarak hat yeniden dengelenmiştir.

Gamberini vd. (2006), problemin çözümü için çok kriterli karar verme yöntemi olan TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) yöntemine ve sezgisel bir yaklaşıma dayanan yeni bir sezgisel yöntem geliştirmiştir. İlk dengeleme ve yeniden dengeleme arasındaki benzerliğin ölçülmesi için 0 ve 1 arasında değer alan

benzerlik faktörü isimli bir indeks tanımlanmıştır. Hattın eski ve yeni hali arasındaki benzerlik arttıkça benzerlik faktörü 1 değerine yaklaşmakta, eski ve yeni durumun aynı olması durumunda ise 1'e eşit olmaktadır. Sonuçlar, önerilen yaklaşımın maliyette azalma ve benzerlik faktöründe artış meydana getirdiğini göstermiştir.

Gamberini vd. (2009), problemin Pareto sınırını bulmak için çoklu tek geçiş sezgiseli önermiştir. Önerilen algoritmada dört farklı sezgisel yöntem kullanılmıştır. Ayrıca, çok amaçlı bir genetik algoritma geliştirilmiştir. Yapılan deneyler sonucunda, genetik algoritmanın benzerlik optimizasyonu açısından tutarsız sonuçlar ortaya koyduğu ve önerilen çoklu tek geçiş sezgiselinin daha üstün olduğu görülmüştür. Çoklu tek geçiş sezgiseli bir vaka çalışmasında kullanılmış ve iki amaç fonksiyonu arasında iyi bir ödünleşme sağlanmıştır.

Çelik vd. (2014), stokastik görev süreli U-tipi bir montaj hattında yeniden dengeleme çalışması gerçekleştirmiştir. Görev taşıma, iş istasyonu açma/kapama ve iş istasyonu işletme maliyetlerinden oluşan toplam yeniden dengeleme maliyetinin enküçüklenmesi amacıyla karınca kolonisi algoritması önerilmiştir. Çalışmada, test problemleri oluşturulmuş ve farklı faktörler açısından yeniden dengeleme maliyetleri karşılaştırılmıştır.

Li (2017), işlemlerin robotlar tarafından gerçekleştirildiği ve hattın izlenerek robotların öğrenme yeteneklerinin birbirine aktarılabildiği ve stokastik işlem süresi iyileşmelerinin görüldüğü bir üretim ortamı düşünerek ilk dinamik MHYD problemi yaklaşımını sunmuştur. İşlem sürelerinde görülen stokastik azalmalar sayesinde hattın gerçek zamanlı olarak yeniden dengelendiği bu çalışmada, çevrim süresinin enküçüklenmesi amaçlanmış ve çözüm için dal-sınır algoritmasına dayalı çevrimiçi bir algoritma önerilmiştir. İşlem sürelerinde iyileşmeler meydana geldiğinde yeni optimal çözümler üreten bu algoritma, literatürde bilinen diğer yöntemlerle karşılaştırılmış ve önerilen yaklaşımın daha yüksek performans gösterdiği sonucuna ulaşılmıştır.

Serin vd. (2019), stokastik işlem süreli U-tipi MHYD problemini ele alarak toplam açık/kapalı iş istasyonu maliyeti, iş istasyonu işletme maliyeti ve görev taşıma maliyetini enküçüklemek için genetik algoritma önermiştir. Önerilen yaklaşım test problemleri

kullanılarak sınanmış ve karınca kolonisi optimizasyon algoritması ile karşılaştırıldığında performansının başarılı olduğu görülmüştür.

3.1.2. Deterministik görev süreli montaj hattı yeniden dengeleme problemi

Yılmaz ve Erol (2005), deterministik görev süreli MHYD problemi üzerinde çalışmıştır. Bu çalışmada, görev sürelerinde meydana gelen değişikliklerin ardından denge gecikmesinin belirlenen değeri aşması durumunda, taşınmaya müsait görevlerin istasyonlarının değişmesiyle hat yeniden dengelenmektedir. Çalışmada, görevlerin farklı istasyonlara taşınmasıyla meydana gelen taşıma maliyetini enküçüklemek amaçlanmıştır. Problemin çözümü için tamsayılı programlama modeli ve dal-sınır algoritması önerilmiştir. Matematiksel modelde; çevrim süresi, öncelik ilişkileri, denge gecikmesi, bölgeleme, bölünmeme ve taşıma kısıtları bulunmaktadır. Önerilen dal-sınır algoritması, gereksiz dallanmaları önleyerek çözümlere daha hızlı bir şekilde ulaşılmasını sağlayan kesme kurallarını içermektedir. Algoritma, farklı boyutlara sahip problemler kullanılarak sınanmış ve kesme kuralları sayesinde optimal çözümlere hızlı bir şekilde ulaşılmıştır.

Corominas vd. (2008), işçilerin tecrübeli ve tecrübesiz olarak gruplandırıldığı bir motosiklet montaj hattında yılın bazı zamanlarında talebin artması nedeniyle hattın yeniden dengelenmesi üzerine bir çalışma gerçekleştirmiştir. Tecrübesiz işçiler, talebin sezona bağlı olarak değişmesinden dolayı geçici olarak işe alınan ve kalıcı işçilere göre görevleri daha uzun sürede gerçekleştiren gruptur ve tecrübesiz işçilerin yanında en az bir tecrübeli işçi çalışmaktadır. Geçici işçiler, her görevi yapamamakta ve her görev her istasyona atanamamaktadır. Aynı zamanda, kirli ve temiz el işleri olarak ifade edilen ve aynı istasyona atanamayan uyumsuz iş grupları bulunmaktadır. Çalışmada, verilen çevrim süresi ve kalıcı işçi sayısı için geçici işçi sayısının enküçüklenmesi amacıyla matematiksel bir model önerilmiştir. Elde edilen sonuçlar, yeni durumda montaj hattının daha az sayıda geçici işçi ile dengelenebileceğini göstermiştir.

Ağpak (2010), çevrim süresi değişikliğine bağlı bir yeniden dengeleme çalışması gerçekleştirmiştir. Çalışmada, farklı çevrim süresi değerleri için görev ve ekipman/demirbaşların yerleri değişmeyecek şekilde iş istasyonu sayısını enküçüklemeyi sağlayan ortak bir görev sırası bulmak amaçlanmıştır. Farklı çevrim süreleri için ortak bir

görev sırası bulmak istenmesinin sebebi, bazı makine/demirbaşların yerlerinin değiştirilememesi ve yer değiştirme maliyetinin çok yüksek olmasıdır. Problemin çözümü için iki aşamalı ve Arcus (1965) tarafından önerilen COMSOAL (Computer Method for Sequencing Operations for Assembly Lines) algoritmasına dayalı bir sezgisel önerilmiştir. Sezgisel, yeniden dengeleme esnekliği açısından düz ve U-tipi hatların karşılaştırılmasında kullanılmıştır. Sonuçlar, farklı çevrim süreleri için ortak bir görev sırasıyla optimal iş istasyonu sayısına erişilebildiğini göstermiştir. Makine ve ekipmanların yerlerini değiştirmeden hatların iyi bir şekilde dengelenebileceği vurgulanmıştır.

Grangeon vd. (2011), Fransız bir otomobil firmasında karma modelli bir montaj hattının yeniden dengelenmesi üzerine çalışmıştır. Ele alınan montaj hattında, temel montaj hattı dengeleme kısıtlarının yanında, çalışma yüksekliği ve depolama alanı uzunluğu gibi özel kısıtlar da bulunmaktadır. Bu işletmede, her ay sonraki ayda hangi modelden kaç adet üretileceği belirlenmekte ve iş istasyonlarına görev atamaları buna göre yapılmaktadır. Herhangi bir ay için yapılan görev atamaları, bazı aylar için kısıtları sağlayamamaktadır. Ayrıca, yeni araçların üretimi için yeni işlemler gerekebilmektedir. Bu durumlarda, hattın yeniden dengelenmesi ihtiyacı doğmaktadır. Çalışmada, mümkün olan en az görevin taşınmasını sağlayarak açık iş istasyonu sayısını ve verilen açık istasyon sayısı için istasyonların iş yükü düzgünlüğünü birbirinden bağımsız olarak optimize etmek amaçlanmıştır. Bunun için ilk aşamada optimal bir çözüm elde eden, ikinci aşamada iş istasyonu sayısını enküçükleyen ve son aşamada ise iş yükünü düzgünleştiren üç aşamalı bir sezgisel çözüm yöntemi önerilmiş ve beş endüstriyel problem üzerinde uygulanmıştır.

Oliveria vd. (2012), karma modelli U-tipi MHYD problemi üzerinde çalışarak 0-1 tamsayılı programlama yaklaşımı ve sezgisel bir yöntem önermiştir. Çalışmada, performans ölçütü olarak iş istasyonu sayısı, ortalama iş yükü ve dengesizlik seviyesi kullanılmıştır. Önerilen algoritma bir otomobil montaj hattı vaka çalışmasında uygulanmış ve önemli iyileştirmeler sağlanmıştır.

Yang vd. (2013), talebin sezona bağlı olarak değişmesi sebebiyle yeniden dengelenmesi gereken karma modelli bir montaj hattını ele alan çok amaçlı bir çalışma gerçekleştirmiştir. İş istasyonu sayısını, farklı modeller için her istasyondaki iş yükü değişkenliğini ve yeniden dengeleme maliyetini enküçüklemek için matematiksel bir model

sunulmuştur. Yeniden dengeleme maliyeti olarak iş istasyonu değişen görevlerin toplam süresi alınmıştır. Bu problemi çözmek için çok amaçlı bir genetik algoritma geliştirilmiştir. Algoritma 23 test problemi kullanılarak sınanmış ve çözüm kalitesi açısından iyi sonuçlar elde edilmiştir.

Zha ve Yu (2014), U-tipi MHYD problemini ele alarak makine taşıma ve işçilik maliyetlerinin ve işçilerin yürüme süresinin enküçüklenmesi için çok amaçlı bir matematiksel model formüle etmiştir. Ayrıca, karınca kolonisi algoritması ve filtrelenmiş ışın aramasına dayanan melez bir yöntem geliştirilmiştir. Deneysel sonuçlar, önerilen algoritmanın oldukça etkili olduğunu göstermiştir.

Makssoud vd. (2015), üretim içeriği değiştiğinde hattın yeniden dengelenmesinin gerektiği durumlar için istasyonu değişen görev sayısını ve istasyon sayısını tek amaç fonksiyonunda enküçüklemeyi amaçlamış ve kesin bir çözüm yöntemi önermiştir. Deneysel sonuçlar, matematiksel modelin 34 göreve kadar olan problemleri çok kısa sürelerde çözebildiğini göstermiştir.

Samadhi ve Sumihartati (2016), talep dalgalanmaları nedeniyle kapasitenin talebi karşılamak için yetersiz kaldığı durumlarda, yeni makinelerin satın alınması ve dış kaynak kullanımının ortaya çıktığı bir giyim firmasında bulunan montaj hattını yeniden dengelemiştir. Makine, ek kapasite, gelecekteki âtıl kapasite ve dış kaynak maliyetlerinden oluşan toplam maliyetin enküçüklenmesini amaçlayan bir tamsayılı programlama modeli geliştirilmiştir. Matematiksel model, 41 makine ve 1 yıllık talep miktarlarını içeren bir veri seti kullanılarak test edilmiştir.

Sancı ve Azizoğlu (2017), bir veya daha fazla iş istasyonunda arıza meydana gelmesi nedeniyle kapatılması durumunda hattın yeniden dengelenmesi üzerine çalışmıştır. Çevrim süresi ve iş istasyonu değişen görev sayısı arasında ödünleşmenin bulunmaya çalışıldığı bu çalışmada, tüm bastırılmamış çözümlerin bulunması amacıyla biri karmaşık tamsayılı programlamaya, diğeri ise dal-sınır algoritmasına dayalı iki optimizasyon algoritması önerilmiştir. Deneysel sonuçlar, iki algoritmanın da başarılı sonuçlar elde ettiğini fakat dal-sınır algoritmasının daha üstün olduğunu göstermiştir.

Belassiria vd. (2018), ürün talepleri değiştiğinde hattın yeniden dengelenmesi için bütünleşik bir matematiksel model sunmuştur. Önerilen model, otomobil kablosu üreten bir firmada sınanmıştır. Çalışmada, hat etkinliği ve istasyonlar arasındaki iş yükü dengesini optimize etmek amaçlanmıştır. Temel kısıtlar haricinde, farklı bir iş istasyonuna taşımak için çok ağır olan makineler ile ilgili atama kısıtları da modele dahil edilmiştir. Problemin çözümü için öncelik kurallarına dayanan sezgisel bir yöntem ile melezlenmiş bir genetik algoritma önerilmiştir. Önerilen algoritma, test problemleri kullanılarak ve bir vaka çalışmasında sınanmıştır. Elde edilen sonuçlar, önerilen yaklaşımın etkili ve rekabetçi bir yöntem olduğunu kanıtlamıştır.

Zhang vd. (2018a), çok amaçlı çift taraflı MHYD problemi üzerinde çalışarak inşaat makinesi üretimi yapan Çinli bir firmada; üretim talebi, hat yapısı ve üretim sürecinde meydana gelen değişiklikler sebebiyle hattı yeniden dengelemiştir. Çalışmada; bekleme süresi ile ilişkili yeni çevrim süresi, âtıl zaman ve iş istasyonlarının ve bir istasyonun her iki tarafının enbüyük ve enküçük tamamlanma süreleri arasındaki farkı kontrol eden denge kısıtları gibi özel kısıtlar göz önünde bulundurulmuştur. Çalışmada, çevrim süresi ve yeniden dengeleme maliyetinin enküçüklenmesi amaçlanmıştır. Problemin çözümü için MNSGA (Modified Non-dominated Sorting Genetic Algorithm)-II algoritması önerilmiştir. İlk hat dengesi ve yeni durum karşılaştırıldığında, algoritma ile elde edilen Pareto optimal çözümlerin çok daha iyi olduğu sonucuna ulaşılmıştır. Ayrıca, geliştirilen yöntem hız, yakınsama ve yayılım açısından başlangıç NSGA (Non-dominated Sorting Genetic Algorithm)-II algoritmasının iki çeşidi ile karşılaştırılmış ve önerilen algoritmanın çoğu durumda daha üstün olduğu görülmüştür.

Zhang vd. (2018b), çok amaçlı çift taraflı MHYD problemini ele alarak çevrim süresi ve yeniden dengeleme maliyetlerini enküçüklemek amacıyla matematiksel bir model geliştirmiştir. Modeli optimize etmek için kısıt programlama modeli uygulanmış ve iş istasyonu odaklı bir sezgisel önerilmiştir. Algoritma, bir problem üzerinde sınanmış ve önerilen yöntemin etkin olduğu sonucuna ulaşılmıştır.

Zhang vd. (2018c), değişen piyasa koşulları sebebiyle montaj hatlarının sıklıkla yeniden dengelenmesi gerektiğini belirterek çok amaçlı çift taraflı MHYD problemi üzerinde çalışmıştır. Çalışmada, çevrim süresi ve yeniden dengeleme maliyeti arasında

ödünleşmenin bulunması için sırasıyla görev kaldırılacak istasyonun seçimi, görev eklenecek istasyonun seçimi ve taşınacak görevin seçimi olmak üzere üç sezgisel modülden oluşan itme kuralı temelli bir sezgisel geliştirilmiştir. Önerilen yöntem, bir test problemi kullanılarak doğrulanmıştır.

Zhang vd. (2019a), literatürde ilk kez alan ve kaynak kısıtlı çok amaçlı çift taraflı MHYD problemi için hat etkinliği, yeniden dengeleme maliyeti ve iş yükü düzgünlüğü arasında bir ödünleşme bulmayı amaçlamıştır. Çalışmada, çift taraflı hatlardaki kaçınılmaz bekleme süresi nedeniyle bir istasyonun görev tamamlanma süresinin her zaman iş yüküne eşit olmadığı belirtilmiş ve düzgünlüğü ölçmek için tamamlanma süresinin ortalama mutlak sapması kullanılmıştır. Problemi çözmek için matematiksel bir model ve iyileştirilmiş bir emperyalist rekabetçi algoritma önerilmiştir. Önerilen yöntem, olurlu başlangıç çözümler üretmek için klasik sezgisel kurallara dayanan yeni bir başlatma sezgiseli kullanmaktadır. Ayrıca, yerel arama için yeni bir sezgisel asimilasyon yöntemi ve ek kısıtlarla görevlerin yeniden atamasını gerçekleştirmek için grup bazlı bir kod çözme sezgiseli geliştirilmiştir. Algoritmanın performansı, öncelikle alan ve kaynak kısıtları bulunmayan çok amaçlı çift taraflı MHYD problemi üzerinde test problemlerini kullanarak mevcut bazı algoritmalarla kıyaslanmıştır. Ardından, algoritmanın alan ve kaynak kısıtları ile çift taraflı MHYD problemini çözme konusundaki performansı, NSGA-II algoritması ve orijinal emperyalist rekabetçi algoritma ile karşılaştırılmıştır. Sonuçlar, önerilen algoritmanın başarısını göstermiştir. Bununla birlikte, algoritmayı sınamak için bir vaka çalışması gerçekleştirilmiştir.

MHYD literatüründe yapılan çalışmalar Çizelge 3.1'de özetlenmiştir. bU çalışmaların tamamında görevlerin istasyonlara atanması dikkate alınmış, işçi atamaları ise göz ardı edilmiştir.

Çizelge 3.1. MHYD literatüründe yer alan çalışmalar

Çalışma	Problem türü	Performans ölçütü	Çözüm yöntemi
Li (2017)	stokastik MHYD	çevrim süresi	dal-sınır algoritmasına dayalı çevrimiçi algoritma
Gamberini vd. (2006)	çok amaçlı stokastik MHYD	toplam işçilik ve tamamlanmama maliyeti yeniden atanan görev sayısı	sezgisel
Gamberini vd. (2009)	çok amaçlı stokastik MHYD	toplam işçilik ve tamamlanmama maliyeti yeniden atanan görev sayısı	sezgisel ve çok amaçlı genetik algoritma
Çelik vd. (2014)	stokastik U-tipi MHYD	toplam iş istasyonu açma/kapama ve işletme maliyeti	karınca kolonisi optimizasyon algoritması
Serin vd. (2019)	stokastik U-tipi MHYD	toplam açık/kapalı iş istasyonu, iş istasyonu işletme ve görev taşıma maliyeti	genetik algoritma
Yılmaz ve Erol (2005)	deterministik MHYD	taşıma maliyeti	matematiksel model ve dal-sınır algoritması
Corominas vd. (2008)	deterministik MHYD	geçici işçi sayısı	matematiksel model
Makssoud vd. (2015)	deterministik MHYD	yeniden atanan görev sayısı ve iş istasyonu sayısı	matematiksel model
Samadhi ve Sumihartati (2016)	deterministik MHYD	toplam makine, ek kapasite, gelecekteki âtıl kapasite ve dış kaynak maliyeti	matematiksel model
Belassiria vd. (2018)	deterministik MHYD	hat etkinliği ve istasyonlar arası iş yükü dengesi	öncelik kurallarına dayalı sezgisel ile melezlenmiş genetik algoritma
Sancı ve Azizoğlu (2017)	çok amaçlı deterministik MHYD	çevrim süresi yeniden atanan görev sayısı	matematiksel model ve dal-sınır algoritması
Grangeon vd. (2011)	çok amaçlı karma modelli deterministik MHYD	taşınan görev sayısı açık iş istasyonu sayısı iş yükü düzgünlüğü	sezgisel
Yang vd. (2013)	çok amaçlı karma modelli deterministik MHYD	iş istasyonu sayısı farklı modeler için her istasyondaki iş yükü değişkenliği yeniden dengeleme maliyeti	sezgisel ve çok amaçlı genetik algoritma

Çizelge 3.1. MHYD literatüründe yer alan çalışmalar (devam)

Çalışma	Problem türü	Performans ölçütü	Çözüm yöntemi
Ağpak (2010)	deterministik MHYD ve U-tipi MHYD	iş istasyonu sayısı	sezgisel
Zha ve Yu (2014)	çok amaçlı deterministik U-tipi MHYD	makine taşıma ve işçilik maliyeti işçilerin yürüme süresi	matematiksel model karınca kolonisi optimizasyon algoritması ve filtrelenmiş ışın aramasına dayalı melez algoritma
Oliveria vd. (2012)	çok amaçlı karma modelli deterministik U-tipi MHYD	iş istasyonu sayısı ortalama iş yükü dengesizlik seviyesi	matematiksel model ve sezgisel
Zhang vd. (2018a)	çok amaçlı deterministik çift taraflı MHYD	çevrim süresi yeniden dengeleme maliyeti	MNSGA-II algoritması
Zhang vd. (2018b)	çok amaçlı deterministik çift taraflı MHYD	çevrim süresi yeniden dengeleme maliyeti	matematiksel model ve sezgisel
Zhang vd. (2018c)	çok amaçlı deterministik çift taraflı MHYD	çevrim süresi yeniden dengeleme maliyeti	sezgisel
Zhang vd. (2019a)	çok amaçlı deterministik çift taraflı MHYD	hat etkinliği yeniden dengeleme maliyeti iş yükü düzgünlüğü	matematiksel model ve iyileştirilmiş emperyalist rekabetçi algoritma

3.2. Literatürde Montaj Hattı İşçi Atama ve Dengeleme Problemi

Geleneksel montaj hattı dengeleme probleminde, görev sürelerinin işlemi yapan kişiden bağımsız ve sabit olduğu varsayılmıştır. Ancak, gerçek hayatta işçilerin görevleri gerçekleştirmek için ihtiyaç duydukları süreler; tecrübe, yetenek, bazı işçilerin engelli bireyler olması gibi sebeplerden ötürü değişkenlik göstermektedir.

Miralles vd. (2007), gerçek hayat problemlerine daha gerçekçi yaklaşımlar sunabilmek ve korunaklı iş merkezlerinde çalışan engelli bireyleri hatta entegre edebilmek amacıyla MHİAD problemini tanımlanmıştır.

MHİAD probleminde, farklı yeteneklere sahip belirli sayıda işçi bulunduğundan görev süreleri hangi görevin hangi işçiye atandığına bağlı olarak değişir. Bu problem türünde, görevlerin yanı sıra işçilerin de istasyonlara atanması gerçekleştirilmektedir. NP-zor problem sınıfında bulunan MHD problemine, işgücü atama probleminin eklenmesiyle problem karmaşıklığı daha da artmaktadır.

Miralles vd. (2008), engelli işçilerin çalıştığı korunaklı iş merkezlerine yönelik bir uygulama gerçekleştirmiştir. Çalışma kapsamında, küçük boyutlu problemler için çevrim süresini enküçüklemek amacıyla tamsayılı bir programlama modeli ve bir dal-sınır algoritması geliştirilmiştir. Sunulan algoritma, üç farklı arama stratejisi ve farklı parametreler ile denenmiştir. Ardından, büyük boyutlu problemler için dal-sınır temelli bir sezgisel önerilmiştir. Bu sezgisel ile elde edilen çözümlerin kalitesi makul değerler göstermiştir.

Belirli sayıda iş istasyonun ve montaj görevlerini yerine getirmek için farklı işlem sürelerine ihtiyaç duyan belirli sayıda işçinin bulunduğu düz bir montaj hattında, çevrim süresini enküçükleyecek şekilde görev ve işçilerin istasyonlara eşzamanlı olarak dengeli bir şekilde atanmasının amaçlandığı problem, MHİAD tip-2 problemidir.

MHİAD tip-2 problemi için Miralles vd. (2008) tarafından yapılan varsayımlar aşağıda verilmiştir:

- Görev süreleri ve görevler arasındaki öncelik ilişkileri deterministik olarak bilinmektedir.
- Montaj hattında tek tip ürünün montajı gerçekleştirilmektedir.
- Tampon stokların bulunmadığı, düz bir montaj hattı düşünülmüştür.
- Her görevin işlem süresi, bu görevi gerçekleştiren işçinin yeteneğine bağlı olarak değişmektedir.
- Bazı görevler bazı işçiler tarafından gerçekleştirilemeyebilir, bazı işçiler farklı görevleri yapma konusunda yetenekli olabilir.
- Her işçi sadece bir istasyona atanabilir.
- Her görev, öncelik ilişkilerini sağlayacak ve sadece o görevi gerçekleştirebilecek bir işçinin bulunduğu tek bir istasyona atanabilir.

Notasyonlar

i, j: görev indisleri

h: işçi indisi

s: iş istasyonu indisi

N: görev sayısı

H: işçi sayısı

S: iş istasyonu sayısı

P_j: görev j'nin komşu öncül kümesi $P_j \subset N$

t_{ih}: görev i'nin işçi h açısından işlem süresi

Karar değişkenleri

$ÇS$: çevrim süresi

x_{ish} = 1, eğer görev i işçi h'ye iş istasyonu s'de atanmış ise; 0, diğer durumlar

y_{sh} = 1, eğer işçi h iş istasyonu s'ye atanmış ise; 0, diğer durumlar

Matematiksel model

$$\sum_{s \in S} \sum_{h \in H} x_{ish} = 1 \quad \forall i \tag{3.1}$$

$$\sum_{h \in H} y_{sh} \leq 1 \quad \forall h \tag{3.2}$$

$$\sum_{s \in S} y_{sh} \leq 1 \quad \forall s \tag{3.3}$$

$$\sum_{s \in S} \sum_{h \in H} s.x_{ish} \leq \sum_{s \in S} \sum_{h \in H} s.x_{jsh} \quad \forall j, i \in P_j \tag{3.4}$$

$$\sum_{i \in N} t_{ih}.x_{ish} \leq ÇS \quad \forall s, h \tag{3.5}$$

$$\sum_{i \in N} x_{ish} \leq M.y_{sh} \quad \forall s, h \tag{3.6}$$

$$y_{sh}, x_{ish} \in [0,1] \quad \forall i, s, h \tag{3.7}$$

kısıtları altında

$$enk\ Z = ÇS \tag{3.8}$$

Kısıt (3.1), her görevin sadece bir iş istasyonu ve bir işçiye atanmasını sağlar. Kısıt (3.2), her işçinin sadece bir iş istasyonuna atanmasını kontrol eder. Kısıt (3.3), her iş istasyonuna sadece bir işçi atanmasını garanti eder. Kısıt (3.4), görevler arasındaki öncelik ilişkilerinin yerine getirilmesini sağlar. Kısıt (3.5) ve (3.6), bir s istasyonuna atanan h işçisinin çevrim süresi aşılmadıkça birden fazla görevi yerine getirebileceğini gösterir. Burada M, yeterince büyük bir sayıdır ($\sum_{h \in H} \sum_{i \in N} t_{hi} < M$). Kısıt (3.7), karar değişkenlerine ait işaret kısıtlarıdır. Modelin amaç fonksiyonu (3.8), çevrim süresinin enküçüklenmesidir.

Miralles vd. (2007) tarafından MHİAD probleminin tanımlanasının ardından araştırmacılar problemin çözümü için farklı çözüm yöntemleri geliştirmiştir. Chaves vd. (2007), bu problemi çözmek için kümeleyerek arama yaklaşımını önermiştir. Chaves vd. (2009) ise Chaves vd. (2007) tarafından önerilen kümeleyerek arama yaklaşımını tekrarlı yerel arama ile melezlemiştir. Yöntemin çözüm kalitesi ve gücü açısından karşılaştırıldığında, melez yaklaşımın Chaves vd. (2007) tarafından geliştirilen kümeleyerek arama yaklaşımından daha iyi olduğu görülmüştür.

Moreira ve Costa (2009), MHİAD tip-2 problemindeki bazı kısıtları güncellemiş ve çözümlerin daha basit, esnek, doğru ve hızlı şekilde elde edilebilmesi için olurlu olmayan çözümleri cezalandırma mantığına dayanan bir yasaklı arama algoritması geliştirmiştir.

Costa ve Miralles (2009), korunaklı iş merkezlerinde iş rotasyonu stratejilerinin uygulanması üzerine çalışmıştır. Çalışmada, tam bir rotasyon periyodunda her bir işçi tarafından gerçekleştirilen farklı görev sayısının enbüyüklenmesi amaçlanmıştır. Bu amaç doğrultusunda, bir karmaşık tamsayılı programlama modeli geliştirilmiştir. Ayrıca problemi alt problemlere ayırmak amacıyla sezgisel bir ayrıştırma yöntemi önerilmiştir. Elde edilen sonuçlara göre küçük boyutlu problemler için makul süreler içinde kaliteli çözümlere ulaşılabilmiştir. Ancak, kompleks problemlerin ayrıştırılmasına dayandığından, yöntemin büyük boyutlu problemler açısından sınırlı olduğu sonucuna ulaşılmıştır.

Blum ve Miralles (2011), MHİAD tip-2 problemini çözmek için ışın aramasına dayalı bir algoritma önermiştir. Deneysel sonuçlar, sunulan algoritmanın MHİAD'nin bu türü için gelişmiş bir yöntem olduğunu göstermiştir. Literatürdeki diğer çalışmalarda elde edilen sonuçlara kıyasla, bu algoritma ile tüm durumlarda daha iyi veya eşit sonuçlar elde edilmiştir. Algoritmanın farklı özelliklere sahip problemlerde oldukça başarılı olduğu görülmüştür.

Moreira vd. (2012), MHİAD probleminde görevlerin ve işçilerin iş istasyonlarına atanma sıralarını tanımlayan görev ve işçi önceliği kurallarına dayalı sezgisel bir yaklaşım geliştirmiştir. Bu kurallar, bir genetik algoritma ile melezlenmiştir. Geliştirilen öncelik kuralları; tek başına bir yöntem olarak, meta-sezgiseller için başlangıç çözüm üretici olarak ve melez genetik algoritmada kod çözücü olarak kullanılmaları açısından üç farklı durumda karşılaştırılmıştır. Elde edilen sonuçlar, sezgisellerin hızlı ve tek başına bir yöntem olarak iyi sonuç verdiğini göstermiştir. Ayrıca, daha karmaşık yaklaşımlar içinde bir başlangıç çözüm üretici veya kod çözücü olarak kullanıldığında verimli oldukları sonucuna ulaşılmıştır. Sunulan melez genetik algoritma, literatürde bilinen en iyi yöntemlerle karşılaştırılabilir sonuçlar elde etmiştir.

Araújo vd. (2012), heterojen işçilerin bulunduğu bir montaj hattında, hattın belirli bir noktasında birden fazla istasyonun aynı görev kümesini gerçekleştirebildiği ve birden fazla

işçinin aynı istasyonda aynı ürün üzerinde birlikte çalışabildiği iki farklı MHİAD problemi uzantısı sunmuştur. Bu iki yeni yaklaşım, hat tasarımı ve atama süreci açısından daha karmaşık olsa da sınırlı yeteneğe sahip işçilerin geleneksel hatlarda çalışması açısından daha yüksek esneklik sağlamaktadır. Bu iki problem için doğrusal modeller ve sezgiseller sunulmuştur. Önerilen yöntemlerin geçerliliği ve uygulanabilirliği yapılan testlerle değerlendirildiğinde, bu yeni uzantıların getirdiği esnekliğin hattın verimliliğini önemli ölçüde artırabileceği görülmüştür. Sonuçlar, önerilen yaklaşımların işçilerin görev süreleri birbirinden çok farklı olduğunda da faydalı olabileceğini de göstermiştir.

Castellucci ve Costa (2012), az oranda dengesiz montaj hatlarının mükemmel şekilde dengelenmiş hatlardan daha yüksek verimliliğe sahip olduğunu savunmuştur. Çanak olgusu olarak adlandırılan bu etki, ortadaki istasyonların uç taraftakilerden daha az yüklü olmasıyla ortaya çıkmaktadır. Bu çalışmada, farklı yeteneklere sahip işçilerin bulunduğu montaj hatlarında bu olgunun varlığı incelenmiştir. İlgili durumu temsil edecek şekilde mevcut MHİAD tip-2 matematiksel modeli güncellenmiştir. Bu yapılar, stokastik bir benzetim modelinde uygulanmış ve elde edilen sonuçlar olgunun varlığını doğrulamıştır.

Mutlu vd. (2013), MHİAD tip-2 problemini çözmek için tekrarlı genetik algoritma geliştirmiştir. Algoritmada, arama çeşitliliği ve verimliliği sağlamak için değiştirilmiş ikili arama, genetik algoritma ve tekrarlı yerel arama olmak üzere üç arama yaklaşımı benimsenmiştir. Algoritmanın performansı, test problemleri kullanılarak sezgisel ve meta-sezgisel yaklaşımlarla karşılaştırılmıştır. Chaves vd. (2007) tarafından sunulan 32 test problemi grubundan 26'sı için yeni en iyi sonuçlara veya bilinen en iyi sonuçlara ulaşılmıştır. Dahası, yaklaşım büyük boyutlu problemler için çok daha düşük hesaplama sürelerine ulaşmayı başarmıştır.

Moreira ve Costa (2013), MHİAD problemini iş rotasyonu çizelgeleme ile birlikte ele almıştır. Çalışmada, çevrim süresini enküçüklemek amacıyla sezgisel olarak elde edilen çözüm havuzundan uygun çizelgeleri seçmek için MIP (karma tamsayılı programlama) kullanan melez bir algoritma geliştirilmiştir. Ayrıca, post-optimizasyon yöntemi olarak MIP komşuluğa dayanan yerel bir arama kullanılmıştır. Deneysel sonuçlar, önerilen yöntemin literatürdeki diğer yöntemlere kıyasla hızlı, esnek ve kesin çözümler ürettiğini göstermiştir.

Vilà ve Pereira (2014), MHİAD tip-2 problemi ve literatürdeki diğer bazı problemler arasındaki ilişkiyi araştırmış ve yeni alt sınırlar türetmiştir. Çalışmada, problemi çözmek için dal-sınır ve hatırlama algoritmasına dayanan kesin bir çözüm yöntemi önerilmiştir. Algoritma, test problemleri kullanılarak sınanmış ve sonuçlar, algoritmanın çözüm kalitesi açısından literatürdeki en iyi performans gösteren yöntemleri geride bıraktığını ve mevcut diğer kesin yöntemlerden daha fazla optimal çözümü doğruladığını göstermiştir.

Borba ve Ritt (2014), MHİAD tip-2 problemini çözmek için yeni bir karmaşık tamsayılı model önermiş ve modele süreklilik kısıtları olarak adlandırılan iki kısıt daha dahil ederek farklı bir model oluşturmuştur. Bu problem için doğrusal gevşetmeler ile alt sınırlar elde edilmiştir. Çalışmada ayrıca, ışın aramasına dayalı yeni bir sezgisel algoritma ve yeni indirgeme kuralları ve alt sınırlar kullanan görev odaklı bir dal-sınır yöntemi önerilmiştir. Geniş bir problem seti kullanılarak yapılan kapsamlı testler sonucunda, önerilen yöntemlerin etkili olduğu ve mevcut yöntemleri çözüm süresi ve kalitesi açısından geride bıraktığı görülmüştür.

Moreira, Miralles ve Costa (2015), yeni bir montaj hattı dengeleme problemi olan montaj hattı işçi entegrasyonu ve dengeleme (MHİED) problemini tanıtmıştır. Bu probleme konu olan hatlarda geleneksel ve engelli işçiler çalışmaktadır. Problemin amacı, ek iş istasyonu sayısını enküçükleyerek yüksek verimlilik sağlamak ve bu sırada montaj hattında bir dizi engelli çalışanla bütünleşmektedir. Bu problem, MHİED tip-1 problemi olarak adlandırılmaktadır. Çalışmada, problemi çözmek için tüm engelli çalışanların montaj hattında bulunmasını sağlayan ve istasyon sayısını enküçükleyen doğrusal tamsayılı model geliştirilmiştir. Ayrıca, bu modelin engelli işçilerin atandığı istasyonların âtıl zamanını azaltan bir çeşidi önerilmiştir. Basit bir montaj hattı dengeleme durumuyla başlayan ve ek istasyonları azaltırken uygun engelli işçileri atayan bir sezgisel yaklaşım uygulanmıştır. Çözümlerin geliştirilmesi için MIP komşuluğa dayanan iki post-optimizasyon prosedürü tasarlanmıştır. Son olarak, sezgiseli sınamak için orijinal görev düzenini değiştirmeden geleneksel bir montaj hattında engelli işçileri yerleştirmeye çalışan basit bir yerine koyma sezgiseli uygulanmıştır. Çok sayıda test problemi kullanılarak yapılan deneysel çalışmanın sonuçları önerilen sezgisel yöntemin verimliliğini göstermiştir. Engelli bireylerin montaj hatlarına çok az üretkenlik kaybıyla dahil edilip, aynı zamanda diğer planlama hedeflerinin de dikkate alınabileceği görülmüştür.

Araújo vd. (2015), paralel hatlarda birbirinden bağımsız çalışan görev ekiplerinin bulunduğu paralel MHİAD tip-2 problemini tanımlamıştır. Çalışmada, önerilen problem için doğrusal karmaşık tamsayılı model sunulmuştur. Ayrıca, yasaklı arama algoritması ve yanlı rassal sayılı genetik algoritma olmak üzere iki çözüm yöntemi önerilmiştir. Test problemleri kullanılarak yapılan testler, alternatif hat düzenlerinin avantajını göstermiştir.

Ramezanian ve Ezzatpanah (2015), karma modelli MHİAD problemi için çevrim süresinin ve işçilerle ilgili işletme maliyetlerinin enküçüklenmesini amaçlanmıştır. Bu çok amaçlı problemi çözmek için bir hedef programlama yaklaşımı kullanılmış ve problemin çok karmaşık olması sebebiyle sosyo-politik tabanlı ve asimilasyon, devrim ve emperyalist rekabet gibi emperyalist prosedürleri içeren emperyalist rekabetçi algoritma adlı bir evrimsel algoritma geliştirilmiştir. Önerilen algoritmanın performansının kayda değer olduğu görülmüştür. Elde edilen sonuçlar, algoritmanın performansının yüksek ve hesaplama süresinin düşük olduğunu göstermiştir.

Moreira vd. (2015), MHİED probleminin robust türü üzerinde çalışarak görev sürelerinin heterojen işçilere bağlı ve olası değer aralıklarında bulunduğu bir montaj hattı düşünmüştür. Çalışmada, engelli bir dizi işçiyi hatta entegre ederken gerekli ekstra iş istasyonu sayısını enküçüklemek amaçlanmıştır. Öncelikle, MHİAD tip-1 probleminin robust türü için iki karmaşık tamsayılı model önerilmiş ve bunların MHİED tip-1 problemine nasıl uyarlanacağı açıklanmıştır. Ayrıca, kurma temelli bir sezgisel geliştirilmiştir. Deneysel sonuçlar, algoritmanın oldukça kısa süreler içinde kaliteli çözümlere ulaştığını ve robust problemin deterministik olana kıyasla çok daha iyi sonuçlar elde ettiğini göstermiştir.

Sungur ve Yavuz (2015), hiyerarşik işçi ataması ile montaj hattı dengeleme (HİAMHD) problemi isimli yeni bir montaj hattı dengeleme problemini tanıtmıştır. Bu problem türünde, görevler yeterlilik gereksinimlerine göre farklılık göstermekte ve işçilerin yeterlilik düzeyleri hiyerarşik olarak sıralanmaktadır. Hiyerarşik işgücü yapısında, daha yüksek yeterliliğe sahip bir işçi, daha yüksek maliyetle daha düşük yeterlilikte olanların yerine geçebilir fakat bunun tersi geçerli değildir. Daha yüksek yeterlilikte bir işçi daha yüksek maliyet gerektirdiğinden dengeleme kararı toplam maliyeti etkilemektedir. Belirli bir çevrim süresi için toplam maliyeti enküçükleyerek istasyonlara işçilerin ve görevlerin en uygun şekilde atanmasını sağlayan doğrusal tamsayılı programlama modeli önerilmiştir.

Zaman ve maliyet faktörlerinin, toplam maliyete ve istasyonlara atanan toplam işçi türü sayısına olan etkilerini değerlendirmek için sayısal bir deney yapılmıştır. Maliyet faktörünün düzeyi arttıkça, en yüksek yeterliliğe sahip işçi sayısı ve toplam maliyet artmış, daha düşük yeterlilikte işçi sayısı azalmıştır. Zaman faktörü aynı etkiye sahip olmasına rağmen, maliyet faktörünün analiz edilen tüm değişkenler üzerinde daha büyük bir etkiye sahip olduğu görülmüştür.

Polat vd. (2016), MHİAD tip-2 problemini çözmek için iki aşamalı bir değişken komşu arama algoritması sunmuştur. İlk aşamada değişken komşu arama yaklaşımı ile görevler istasyonlara, ikinci aşamada ise değişken komşu iniş yöntemi ile işçiler istasyonlara atanmaktadır. Sunulan algoritmanın performansı test problemleri kullanılarak değerlendirilmiştir. Sonuçlar, algoritmanın verimlilik ve sağlamlık açısından literatürdeki diğer yöntemlerden üstün olduğunu göstermiştir. Ayrıca, LCD-TV üretimi yapan bir işletmede bir uygulama gerçekleştirilmiştir. Bu sayede, işletmede son montaj hattının performansının önemli ölçüde artması sağlanmıştır.

Heterojen işgücünün olduğu montaj hatlarında, hattın dengelenmesini zorlaştıran devamsızlık problemiyle sıklıkla karşı karşıya kalınmasından hareketle Ritt vd. (2016), işçi mevcudiyetinin belirsiz olduğu durumda çevrim süresini enküçüklemek için yeni bir MHİAD problemi uzantısı tanıtmış ve bu probleme olasılıklı işçi mevcudiyeti ile montaj hattı işçi atama ve dengeleme (OİMMHİAD) problemi adını vermiştir. Problem, ilk aşaması iş istasyonlarına görev atanması, ikinci aşaması ise iş istasyonlarına işçi atanması olan iki aşamalı karmaşık tamsayılı stokastik bir problem olarak modellenmiştir. Sabit görev atama ve esnek görev atama durumları için iki matematiksel model sunulmuştur. Sabit görevler modelinde, tüm görevler istasyonlara sabitlenmektedir. Esnek görevler modeli ise her iş gününün başında bazı görevlerin yeniden atanabileceği şekilde ilk modelin genelleştirilmiş halidir. Çalışmada, tavlama benzetimine dayalı sezgisel bir çözüm yöntemi önerilmiştir. Problemler üzerinde yapılan değerlendirmeler sayesinde, modellerin çözümlerinin küçük boyutlu problemlerle sınırlı olduğu ve önerilen sezgiselin ise büyüklüğü 8 işçiye kadar olan problemler için çok daha kısa sürede iyi çözümler bulabilen etkili bir algoritma olduğu görülmüştür. Ayrıca, önerilen modellerin devamsızlık oranının yüksek olduğu diğer üretim ortamlarında ve işçilerin farklı becerilere sahip olduğu durumlarda da uygulanabileceği belirtilmiştir.

Zacharia ve Nearchou (2016), iki amaçlı MHİAD problemi için Pareto-optimal çözüm kümesini belirlemek amacıyla çok amaçlı bir evrimsel algoritma sunmuştur. Çalışmada, çevrim süresinin ve iş yükü düzgünlüğünün optimize edilmesi amaçlanmıştır. Ele alınan amaçların birleştirilmesinde beş farklı yöntem kullanılmıştır. Pareto çözümlerinin kalitesi ile ilgili bir performans analizi yapılmış ve sonuçların çözüm kalitesi açısından tatmin edici olduğu görülmüştür.

Moreira vd. (2017) çok amaçlı MHİED tip-2 problemini sunmuştur. Çalışmada, işçilerin kümelenmesini önlemek ve daha yüksek bir entegrasyon düzeyi sağlamak amaçlanmıştır. Özel bir dizi işçinin (engelli işçiler veya monitör görevlileri) dengeli dağıtımı ve hat verimliliğinin optimizasyonu için sırasıyla Miltenburg'un düzenlilik kriteri ve çevrim süresi ölçüt olarak kullanılmıştır. İşçilerin birbirinden çok farklı görev sürelerinin bulunduğu bu problem için matematiksel modeller formüle edilmiş ve sezgiseller geliştirilmiştir. Önerilen algoritmalar iki aşamalı olup, ilk aşamada olası işçi dağılımları elde edilmiş ve ikinci aşamada ise bunlar işçi ve görev atamalarını yapmak için kullanılmıştır. Kapsamlı bir dizi deney ile elde edilen sonuçlar, heterojen işçilerin büyük verimlilik kayıpları olmaksızın hatta entegre edilebilmelerinin yanı sıra düzgün dağıtım amacının kolaylıkla sağlanabileceğini göstermiştir.

Öksüz vd. (2017), U-tipi montaj hatlarında işçi performanslarının göz önünde bulundurulduğu ilk tip-E problemini sunmuştur. Problemin çözümü için doğrusal olmayan bir model sunulmuş ve bu model parçalı doğrusallaştırma tekniği kullanılarak doğrusallaştırılmıştır. Çözüm için yapay arı kolonisi algoritması ve genetik algoritma önerilmiştir. Literatürde bulunan veri setleri bu model ve meta-sezgiseller yardımı ile çözülmüştür. Geliştirilen meta-sezgiseller optimum çözümlere ulaşabilmiştir. Test problemleri işçi performansları göz önünde bulundurulmadan tekrar çözülerek sonuçlar karşılaştırılmıştır. Görevleri gerçekleştirmek için gerekli standart süreler ve her işçi için görevler açısından performans çarpanlarının bulunduğu bu çalışmada, bir test probleminde performans çarpanları değiştirilmiş ve işçi performansının hat verimliliği üzerine etkisini göstermek amacıyla karşılaştırmalar yapılmıştır. Sonuçlar, işçi performansına bağlı olarak hattın daha az sayıda iş istasyonu ile dengelenebileceğini göstermiştir. Bununla birlikte,

işçilerin performansının iyi olmadığı durumlarda daha fazla istasyonun gerekli olduğu görülmüştür.

Janardhanan vd. (2018), çift taraflı MHİAD problemini tanımlayarak çevrim süresinin optimize edilmesi amacıyla üç çeşit yapay arı kolonisi algoritması önermiştir. İlk tür, orijinal yapay arı kolonisi algoritması ve diğer ikisi ise bunun geliştirilmiş halleridir. Önerilen algoritmalar, test problemleri kullanılarak sınanmıştır. Önerilen üç algoritma ile uyarlanmış test problemleri üzerinde elde edilen sonuçlar, literatürdeki iyi bilinen meta-sezgisel algoritmalar ile elde edilen sonuçlar ile karşılaştırılmıştır. Geliştirilen iki yapay arı kolonisi algoritmasının en iyi performansa sahip algoritmalar olduğu görülmüştür.

Efe vd. (2018), görev atamalarında sadece görev sürelerini dikkate almanın bazı işçiler üzerinde aşırı iş yüküne sebep olduğunu ve bu durumun hattın performansını olumsuz yönde etkilediğini savunarak işçilerin yaş ve cinsiyetine bağlı verimli iş yükü farklılıklarının bulunduğu MHİAD tip-2 problemini ele almıştır. Çalışmada, aşırı çalışma koşullarının olumsuz etkilerinden kaçınmak için çevrim süresini ya da yaş ve cinsiyete bağlı fiziksel iş yükü kapasitesini aşmadan görevleri iş istasyonlarına atamak amaçlanmıştır. Yaş ve cinsiyete bağlı iş yükü kapasitesini belirlemek için bir regresyon modeli sunulmuş ve yaş ve cinsiyetin iş yükü kapasitesi üzerindeki etkisini görmek için altı farklı yaş kategorisi incelenmiştir. Problemin çözümü için doğrusal ikili tamsayılı model geliştirilmiştir. Model, test problemleri kullanılarak ve bir tekstil firmasında uygulama gerçekleştirerek değerlendirilmiştir. Çalışma sayesinde, tekstil firmasındaki tüm iş istasyonları fiziksel iş yükü ve görev süreleri açısından dengelenmiştir.

Pereira (2018), MHİAD tip-2 probleminin verilerin belirli aralıklarda olduğu robust (enkenb pişmanlık) MHİAD türünü sunmuştur. Bu problemde, görev sürelerinin alt ve üst sınırlarının bilindiği varsayılarak olası tüm senaryolar arasındaki mutlak en büyük pişmanlığı enküçükleyecek görev ve işçi atamalarını bulmak amaçlanmıştır. Çalışmada, yeni bir pişmanlık değerlendirme yöntemi önerilmiş ve problemin çözümü için yeni ve uyarlanmış çözüm yöntemleri sunulmuştur. Sonuçlar, yeni sezgiselin uygulanabilir bir çözüm yöntemi olduğunu ve orta büyüklükteki test problemlerini makul çalışma süreleri içinde optimal olarak çözebildiğini göstermiştir.

Akyol ve Baykasoğlu (2019a), MHİAD tip-2 problemini çözmek için kural tabanlı kurma temelli rassallaştırılmış arama algoritması önermiştir. Çalışmada, 39 görev önceliği kuralı ve 4 işçi önceliği kuralı tanımlanmıştır. Önerilen çözüm yöntemi, test problemleri kullanılarak sınanmıştır. Sonuçlar, algoritmanın, çözüm kalitesi ve süresi açısından ilgili literatürdeki en iyi performansa sahip yöntemleri geride bıraktığını göstermiştir.

Akyol ve Baykasoğlu (2019b), klasik yaklaşımın aksine aynı istasyon sürelerine sahip iki istasyonun eşit olmadığını, görevlerin yorucu olma düzeylerinin de hesaba katılması gerektiğini savunmuştur. Özellikle işgücü yoğun montaj hatlarında tekrarlı görevlerin işçiler üzerindeki birikimli etkisi ve çalışma koşullarının ergonomik olmasının engelli bireyler için hayati önem taşıması sebebiyle ergonomik riskleri ele alan yeni bir MHİAD problemi türü olan ErgoMHİAD problemini tanıtmıştır.

Önerilen problem, Akyol ve Baykasoğlu (2019a) çalışmasında sunulan kural tabanlı kurma temelli rassallaştırılmış arama algoritması ile çözülmüştür. Ergonomik risk değerlendirmesi için ise OCRA (Operational Competitiveness RAting) indeksi kullanılmıştır. Birincil amaç, çevrim süresini enküçüklemektir. Ek olarak, ergonomik koşullar ile ilgili dört amaç daha ele alınmıştır: Ergonomik açıdan tehlikeli anlamına gelen kırmızı istasyon sayısını enküçükleme, kırmızı istasyonların OCRA indeksleri ortalamasını enküçükleme, tüm istasyonların OCRA indekslerinin ortalama mutlak sapmasını enküçükleme ve tüm istasyonların OCRA indeksleri ortalamasını enküçükleme. Önerilen problemin çok amaçlı doğasının üstesinden gelmek için öncelikli hedef programlama yaklaşımı geliştirilmiştir.

Bu problem, ilgili literatürde öncü olduğundan anlamlı bir karşılaştırma yapmak için iki adımlı bir çözüm yöntemi sunulmuştur. İlk olarak, MHİAD tip-2 problemi çözülmüş ve ilk adımın bir sonucu olarak, her test problemi için en küçük çevrim süresi değerine sahip bir çözüm elde edilmiştir. Daha sonra, ergonomik risk faktörlerini dikkate alarak aynı problem için önleyici hedef programlama ile en iyi ergonomik koşullara sahip bir çözüm bulunmuştur. Sonuç olarak, daha ergonomik bir ortamın makul bir çevrim süresi artışı ile elde edilebileceği görülmüştür. Bu sayede, çalışma ortamındaki ergonomik risk faktörlerini göz önünde bulundurarak ergonomik koşulların önemli ölçüde geliştirilebileceği kanıtlanmıştır.

Janardhanan vd. (2019), çift taraflı MHİAD tip-2 problemini çözmek için karmaşık tamsayılı model ve yerel eniyiye takılmayı önleyen yeniden başlama mekanizması içeren göçmen kuşlar algoritması önermiştir. Test problemleri kullanılarak yapılan deneyler ile önerilen yöntemin diğer yöntemlere üstünlük sağladığı sonucuna ulaşılmıştır.

Zhang vd. (2019b), U-tipi hatlarda MHİAD tip-2 probleminin çözümü için göçmen kuşlar algoritmasının geliştirilmiş bir türünü önermiştir. Algoritmada, çözümlerin iyileştirilmesi için iki komşuluk yapısı ve popülasyonda çeşitlilik sağlamak ve yerel eniyiye takılmayı önlemek için bir sıcaklık kabul kriteri kullanılmıştır. Ayrıca, en çok umut vadeden kuşun hattın en başında bulunma olasılığını artırmak için bir rekabet mekanizması tanımlanmıştır. Önerilen yöntem karşılaştırma problemleri üzerinde denenmiş ve literatürdeki diğer yöntemleri geride bıraktığı sonucuna ulaşılmıştır.

Çizelge 3.2'de MHİAYD problemi konusunda yapılan çalışmalar özetlenmiştir. Literatürde yer alan çalışmaların çoğunda performans ölçütü olarak işletmeler için oldukça önemli olan çevrim süresi alınmıştır. Bu çalışmalarda, işlem sürelerinin tüm işçiler için sabit olduğu varsayımının kaldırılmasıyla gerçek hayat problemleri için daha uygulanabilir bir yaklaşım sunulmasına karşın hatların pratikte sıklıkla karşılaşılan yeniden dengelenme problemini değil, kurulum aşamasında sadece bir kez dengelenmesi ele alınmaktadır.

Bu çalışmada, MHYD ve MHİAYD literatürlerindeki boşluğu doldurmak amacıyla görevlerin işçiler açısından işlem sürelerinin birbirinden farklı olduğu göz önünde bulundurularak görev ve işçilerin iş istasyonlarına yeniden atandığı MHİAYD problemi ilk kez tanımlanmıştır. Çevrim süresinde ve görev-iş istasyonu atamalarında meydana gelen değişkenliğin optimize edilmesi amacıyla matematiksel model ve YAK algoritması önerilmiştir.

Çizelge 3.2. MHİAD literatüründe yer alan çalışmalar

Çalışma	Problem türü	Performans ölçütü	Çözüm yöntemi
Miralles vd. (2007)	MHİAD	çevrim süresi	matematiksel model
Chaves vd. (2007)	MHİAD	çevrim süresi	kümeleyerek arama
Miralles vd. (2008)	MHİAD	çevrim süresi	matematiksel model, dal-sınır algoritması ve sezgisel
Moreira ve Costa (2009)	MHİAD	çevrim süresi	yasaklı arama
Chaves vd. (2009)	MHİAD	çevrim süresi	melez kümeleyerek arama
Blum ve Miralles (2011)	MHİAD	çevrim süresi	ışın aramasına dayalı algoritma
Moreira vd. (2012)	MHİAD	çevrim süresi	melez genetik algoritma
Castellucci ve Costa (2012)	MHİAD	çevrim süresi	benzetim modeli
Mutlu vd. (2013)	MHİAD	çevrim süresi	tekrarlı genetik algoritma
Vilà ve Pereira (2014)	MHİAD	çevrim süresi	dal-sınır ve hatırlama algoritmasına dayalı yöntem
Borba ve Ritt (2014)	MHİAD	çevrim süresi	matematiksel model, ışın aramasına dayalı sezgisel ve dal-sınır algoritması
Polat vd. (2016)	MHİAD	çevrim süresi	değişken komşu arama algoritması
Akyol ve Baykasoğlu (2019a)	MHİAD	çevrim süresi	kural tabanlı kurma temelli rassallaştırılmış arama algoritması
Efe vd. (2018)	MHİAD	çevrim süresi	matematiksel model
Zacharia ve Nearchou (2016)	çok amaçlı MHİAD	çevrim süresi iş yükü düzgünlüğü	çok amaçlı evrimsel algoritma
Ramezanian ve Ezzatpanah (2015)	çok amaçlı karma modelli MHİAD	çevrim süresi işçiler ile ilgili toplam işletme maliyeti	matematiksel model ve emperyalist rekabetçi algoritma
Öksüz vd. (2017)	U-tipi MHİAD	hat etkinliği	matematiksel model, yapay arı kolonisi algoritması ve genetic algoritma
Zhang vd. (2019b)	U-tipi MHİAD	çevrim süresi	göçmen kuşlar algoritması
Janardhanan vd. (2018)	çift taraflı MHİAD	çevrim süresi	yapay arı kolonisi algoritması

Çizelge 3.2. MHİAD literatüründe yer alan çalışmalar (devam)

Çalışma	Problem türü	Performans ölçütü	Çözüm yöntemi
Janardhanan vd. (2019)	çift taraflı MHİAD	çevrim süresi	matematiksel model ve göçmen kuşlar algoritması
Araújo vd. (2012)	paralel istasyonlar ve işbirliğine dayalı MHİAD	çevrim süresi	matematiksel model ve sezgisel
Araújo vd. (2015)	paralel MHİAD	çevrim süresi	matematiksel model, yasaklı arama algoritması ve yanlı rassal sayılı genetik algoritma
Ritt vd. (2016)	stokastik MHİAD	beklenen çevrim süresi	matematiksel modeller ve yerel arama sezgiselleri
Pereira (2018)	robust MHİAD	mutlak en büyük pişmanlık	kesin ve sezgisel çözüm yöntemleri
Costa ve Miralles (2009)	iş rotasyonlu MHİAYD	tam bir rotasyon periyodunda her işçi tarafından gerçekleştirilen farklı görev sayısı	matematiksel model ve sezgisel ayrıştırma yöntemi
Moreira ve Costa (2013)	iş rotasyonlu MHİAYD	çevrim süresi	MIP temelli melez algoritma ve post-optimizasyon yöntemi
Moreira, Miralles ve Costa (2015)	MHİED	iş istasyonu sayısı	matematiksel model ve sezgisel
Moreira vd. (2015)	robust MHİED	ekstra iş istasyonu sayısı	matematiksel model ve sezgisel
Moreira vd. (2017)	çok amaçlı MHİED	çevrim süresi heterojen işçilerin düzgün dağıtımı	matematiksel modeller ve sezgiseller
Sungur and Yavuz (2015)	HİAMHD	toplam maliyet	matematiksel model
Akyol ve Baykasoğlu (2019b)	çok amaçlı ErgoMHİAD	çevrim süresi kırmızı istasyon sayısı ortalama kırmızı OCRA endeksi OCRA endekslerinin ortalama mutlak sapması Tüm istasyonların OCRA enkdeksleri ortalaması	kural tabanlı kurma temelli rassallaştırılmış arama algoritması

4. MATERYAL VE YÖNTEM

Literatürde yer alan MHYD çalışmalarında, montaj görev sürelerinin işlemi yapan işçiden bağımsız ve sabit olduğu varsayılmıştır. Bu çalışmada, bir veya daha fazla iş istasyonunun kapanması sonucunda ilk duruma göre mümkün olan en az değişkenlikle hattın yeniden dengelenmesi amaçlanmıştır. Değişkenlik açısından; ilk hat dengesi ile yeni durum arasındaki çevrim süresi değişimi ve görev-iş istasyonu atamalarındaki değişimler göz önünde bulundurulmuş, görev-işçi atamalarındaki değişimler direkt olarak dikkate alınmamıştır. Bunun sebebi, görev-işçi atamalarında meydana gelen değişimlerin çevrim süresini etkilemesi ve böylelikle bu değişimlerin etkisinin amaç fonksiyonuna yansımasıdır.

Montaj görevlerinin başlangıç hat dengesinde belirli sayıda iş istasyonuna atandığı bir durum düşünülmüştür. Bir veya daha fazla istasyonun kapanması sebebiyle tüm görevlerin kalan istasyonlara kısıtları sağlayacak şekilde dağıtılması gerekmektedir. Yeni durumda, işçi sayısı açık istasyon sayısından fazla olmaktadır.

4.1. Önerilen Matematiksel Model

MHİAYD problemi için yapılan varsayımlar aşağıdaki gibidir:

- Görev süreleri deterministik olarak bilinmektedir.
- Görevler arasındaki öncelik ilişkileri bilinmektedir.
- Montaj hattında tek tip ürünün montajı gerçekleştirilmektedir.
- Tampon stokların bulunmadığı, düz bir montaj hattı düşünülmüştür.
- Her görevin süresi, bu görevi gerçekleştiren işçiye bağlıdır.
- Bazı görevler bazı işçiler tarafından gerçekleştirilemeyebilir, bazı işçiler farklı görevleri gerçekleştirme konusunda yetenekli olabilir.
- Her işçi tek bir istasyona atanabilir.
- Her görev, öncelik ilişkilerini sağlayacak ve sadece o görevi gerçekleştirebilecek bir işçinin bulunduğu tek bir istasyona atanabilir.
- Montaj hattı başlangıç durumda dengelenmiş haldedir.

Notasyonlar

i, j: görev indisleri

h: işçi indisi

s: iş istasyonu indisi

N: görev sayısı

H: işçi sayısı

S: başlangıç durumdaki iş istasyonu sayısı

P_j: görev j'nin komşu öncül kümesi $P_j \subset N$

F: kapanan istasyonların kümesi

C_0: hattın başlangıçtaki çevrim süresi

t_{ih}: görev i'nin işçi h açısından işlem süresi

$a_{is} = 1$, eğer görev i eski durumda iş istasyonu s'ye atanmış ise; 0, diğer durumlar

f: kapanan istasyonlardaki başlangıç durumdaki toplam görev sayısı

$$f = \sum_{i} \sum_{s \in F} a_{is} \tag{4.1}$$

w: amaç fonksiyonunda çevrim süresinin alt sınırlardan kısmi sapması için ağırlık değeri

Karar değişkenleri

C: yeniden dengelenmiş hattın çevrim süresi

x_{ish} = 1, eğer görev i yeniden dengelenmiş hatta işçi h'ye iş istasyonu s'de atanmış ise; 0, diğer durumlar

y_{sh} = 1, eğer işçi h yeniden dengelenmiş hatta iş istasyonu s'ye atanmış ise; 0, diğer durumlar

q_{is} = 1, eğer görev i yeni durumda farklı bir iş istasyonundan iş istasyonu s'ye taşınmışsa ($a_{is} \neq \sum_h x_{ish}$); 0, diğer durumlar

T: yeniden atanan görev sayısı (yeni ve eski hat dengesinde istasyonu değişen görev sayısı)

$$T = \sum_{i} \sum_{s \,|\, a_{is}=0} q_{is} \tag{4.2}$$

Matematiksel model

$$\sum_{s \in S} \sum_{h \in H} x_{ish} = 1 \quad \forall i \tag{4.3}$$

$$\sum_{h \in H} y_{sh} \leq 1 \quad \forall h \tag{4.4}$$

$$\sum_{s \in S} y_{sh} \leq 1 \quad \forall s \tag{4.5}$$

$$\sum_{s \in S} \sum_{h \in H} s.x_{ish} \leq \sum_{s \in S} \sum_{h \in H} s.x_{jsh} \quad \forall j, i \in P_j \tag{4.6}$$

$$\sum_{i \in N} t_{ih}.x_{ish} \leq C \quad \forall s, h \tag{4.7}$$

$$\sum_{i \in N} x_{ish} \leq M.y_{sh} \quad \forall s, h \tag{4.8}$$

$$\sum_{i \in N} \sum_{h \in H} x_{ish} = 0 \quad \forall s \in F \tag{4.9}$$

$$q_{is} \geq \sum_{h} x_{ish} - a_{is} \quad \forall i, s \tag{4.10}$$

$$q_{is} \geq a_{is} - \sum_{h} x_{ish} \quad \forall i, s \tag{4.11}$$

$$x_{ish}, y_{sh}, q_{is} \in [0,1] \quad \forall i, s, h \tag{4.12}$$

$$C \geq 0 \tag{4.13}$$

kısıtları altında

$$enk\ Z = w\frac{C - C_0}{C_0} + (1 - w)\frac{T - f}{f} \tag{4.14}$$

Kısıt (4.3), her görevin sadece bir iş istasyonu ve bir işçiye atanmasını sağlar. Kısıt (4.4), her işçinin sadece bir iş istasyonuna atanmasını kontrol eder. Kısıt (4.5), her iş istasyonuna sadece bir işçi atanmasını garanti eder. Kısıt (4.6), görevler arasındaki öncelik ilişkilerinin yerine getirilmesini sağlar. Kısıt (4.7) ve (4.8), istasyon s'ye atanan işçi h'nin çevrim süresi aşılmadıkça birden fazla görevi yerine getirebileceğini gösterir. Burada M, yeterince büyük bir sayıdır ($\sum_{i \in N} \sum_{h \in H} t_{ih} < M$). Kısıt (4.9), kapanan istasyonlara görev atanmamasını sağlar. Kısıt (4.10) ve (4.11), q_{is} değerini hesaplar. Kısıt (4.12) ve (4.13), karar değişkenlerine ait işaret kısıtlarıdır. Modelin amaç fonksiyonu (4.14), dikkate alınan amaçların alt sınırlardan kısmi sapmalarının ağırlıklı toplamının enküçüklenmesidir. Çevrim

süresinin alt sınırı, ilk hat dengesinin çevrim süresidir (C_0). Yeniden atanan görev sayısı için alt sınır ise kapanan istasyonlarda ilk hat dengesinde bulunan görev sayısıdır (f). Kısmi sapma, gerçek sapmanın en kötü sapmaya oranını ifade eder.

4.2. Yapay Arı Kolonisi Algoritması Hakkında Genel Bilgiler

Bilim ve teknolojinin artan bir ivmeyle gelişmeye devam etmesi, gerçek hayat problemlerinin boyutunu ve dolayısıyla karmaşıklığını ve çözüm süresini artırmaktadır. Buna bağlı olarak, klasik yöntemlerin optimizasyon problemlerini çözme konusunda yetersiz kalması, sezgisel yöntemlerin geliştirilmesine yol açmıştır (Erdoğmuş, 2016).

Sezgisel algoritmalar, eniyi çözümü garanti etmeksizin, kısa sürede eniyiye yakın çözümler elde etmeyi hedefler (Karaboğa, 2011). Sezgisel optimizasyon algoritmaları; biyoloji tabanlı, fizik tabanlı, sürü tabanlı, sosyal tabanlı, müzik tabanlı ve kimya tabanlı olmak üzere altı grupta incelenmektedir. Ayrıca, bunların birleşimiyle ortaya çıkan melez yöntemler de bulunmaktadır (Alataş, 2007: (Akyol ve Alataş'tan (2012)).

Günümüzde, sürü zekâsı optimizasyon algoritmaları en etkili algoritmalar arasındadır. Bu algoritmalar; balık sürüleri, kuş sürüleri, karınca kolonileri, termit kolonileri, arı kolonileri gibi canlıların davranışlarından ilham alınarak geliştirilmiştir (Alqattan ve Abdullah, 2015).

Belirli sayıda temsilcinin bulunduğu bir sürü, istenen hedefe ulaşmak amacıyla birlikte çalışır. Temsilciler arasında sık tekrarlanan davranışlar sebebiyle sürüde kolektif zekâ doğar. Sürüdeki bireyler, faaliyetlerini gerçekleştirmek için basit bireysel kurallar kullanır ve diğer sürü üyeleriyle etkileşime geçer. Sürü temsilcileri, en iyi bireyin veya diğer bireylerin davranışlarından ve kendi deneyimlerinden faydalanarak yorumlar yapar. Elde ettikleri bilgileri gelecekte karşılarına çıkacak problemlerin çözümü için bir araç olarak kullanır (Akyol ve Alataş, 2012).

Bu çalışmada önerilen yöntem olan YAK algoritması, arı sürülerinin iş birliği içinde besin arama davranışlarından esinlenerek Karaboğa (2005) tarafından geliştirilen bir sürü zekâsı optimizasyon algoritmasıdır.

YAK algoritmasına sosyal davranışları ile ilham kaynağı olan bal arıları, oldukça kalabalık bir kolonide yaşar ve ayrıntılı bir sosyal organizasyon sürdürür. Bu arılar; besin kaynaklarını keşfeden, karmaşık danslarla iletişim kurabilen ve besin üretebilen canlılardır. Bir arı kolonisi, temsilcileri arılar olan bir sürüdür. Arılar arasındaki bilgi alışverişi, kolektif bilginin oluşumunu sağlar (Abraham vd., 2012).

Doğal bir arı kolonisinde, görevler o iş için özelleşmiş arılar tarafından gerçekleştirilir. Arılar, görevlere göre aralarında iş bölümü yapar ve kendi kendilerine organize olur. İş bölümü yapabilme ve kendi kendine organize olabilme, sürü zekâsının iki önemli bileşenidir. Arıların minimal besin arama modeli; besin kaynakları, görevli işçi arılar ve görevsiz işçi arılar olmak üzere temel üç bileşenden oluşur. Bu bileşenler şu şekilde açıklanabilir (Karaboğa, 2011):

Besin kaynakları: Arıların nektar, polen ya da bal elde etmek için gittiği kaynaklardır. Bir besin kaynağının değeri; çeşidi, yuvaya yakınlığı, nektar konsantrasyonu ve nektarı çıkarmanın kolaylığı gibi çeşitli faktörlere bağlıdır. Ancak, kaynak zenginliği tek ölçüt olarak alınabilir.

Görevli işçi arılar: Bu arıların görevi, önceden keşfedilmiş besin kaynaklarına ait nektarı kovana getirmektir. Görevli arılar, besin kaynağının yeri ve kalitesi ile ilgili bilgiyi kovanda bekleyen arılara iletmekten sorumludur.

Görevsiz işçi arılar: Faydalanılacak besin kaynağı arayan arılardır. Görevi belirsiz işçi arılar kendi aralarında gözcüler ve kâşifler olmak üzere iki gruba ayrılır. Kâşifler, rastgele besin kaynağı arar. Gözcüler ise yuvada bekler ve görevli arılardan aldıkları bilgiler doğrultusunda yeni besin kaynaklarına yönelir. Besin kalitesi ve yeri ile ilgili bilgi paylaşımı dans alanında gerçekleşir. Gözcü arılar, bu alanda birkaç dansı izledikten sonra hangi besin kaynağını seçeceklerine karar verir.

Algoritmada, görevli arı sayısı besin kaynağı sayısına eşittir ve her bir besin kaynağından sadece bir görevli arı faydalanabilir. Sürüdeki arıların yarısını görevli arılar, diğer yarısını ise gözcü arılar oluşturur. Bu yüzden, gözcü arı sayısı da besin kaynağı

sayısına eşit olur. Bir besin kaynağı tükendiğinde, o besin kaynağına karşılık gelen görevli arı, yeni besin kaynağı aramak için kâşif arıya dönüşür (Karaboğa ve Akay, 2011).

YAK algoritmasının temel adımları aşağıdaki gibi özetlenebilir (Kong vd., 2013):

Adım 1: Başla
REPEAT
Adım 2: Görevli arı safhası: Görevli arıları hafızada bulunan besin kaynaklarına gönder.
Adım 3: Gözcü arı safhası: Gözcü arıları hafızada bulunan besin kaynaklarına gönder.
Adım 4: Kâşif arı safhası: Kâşif arıları yeni besin kaynakları keşfetmeleri için arama alanına gönder.
UNTIL (Durma koşulu)

Bu algoritmada, besin kaynakları olurlu çözümleri temsil eder. Her bir besin kaynağının sahip olduğu nektar miktarı ise o besin kaynağının kalitesini gösterir. Görevli arılar, besin kaynaklarından faydalanarak bu kaynaklar hakkında edindikleri bilgileri gözcü arılara iletir. Gözcü arılar, görevli arılardan aldıkları bilgiler ışığında daha iyi olan besin kaynaklarından faydalanma görevini gerçekleştirir. Bir besin kaynağı tükendiğinde, yani belirli sayıda denemenin ardından kaynağı kalitesinin daha fazla geliştirilemediği durumlarda kaynak terk edilir. Terk edilen besin kaynağına karşılık gelen görevli arı, kâşif arı haline gelir ve rastgele besin kaynağı aramaya başlar. Yeni bir besin kaynağı bulduğunda tekrar görevli arıya dönüşür. (Szeto vd., 2011).

YAK algoritmasında kullanılan kontrol parametreleri; besin kaynağı sayısı, limit ve enbüyük iterasyon sayısıdır.

Besin kaynağı sayısı: Çözüm popülasyonunun büyüklüğünü temsil eder.

Limit: Bir besin kaynağının (çözümün) kalitesinin geliştirilebilmesi için izin verilen deneme sayıdır. Bir çözümü geliştirebilmek için art arda yapılan deneme sayısı limit değerini aştıysa, o çözüm terk edilir ve yeni bir çözüm bulunur.

Enbüyük iterasyon sayısı: Algoritmanın durana kadar gerçekleştireceği iterasyon sayısıdır.

YAK algoritmasının temel özellikleri Karaboğa (2011) tarafından aşağıdaki gibi özetlenmiştir:

- Oldukça esnek ve basit bir algoritmadır.
- Gerçek besin arayıcı arıların davranışlarını oldukça yakın bir şekilde simüle eder.
- Sürü zekâsına dayanır.
- Nümerik problemler için geliştirilmiş bir algoritmadır. Ancak, kesikli problemler için de kullanılabilir.
- Kontrol parametresi sayısı oldukça azdır.
- Kâşif arılar tarafından gerçekleştirilen bütünsel arama, görevli ve gözcü arılar tarafından gerçekleştirilen yerel arama yeteneğine sahiptir. Algoritmada her iki arama da dengeli bir şekilde yürütülmektedir.

4.3. Geliştirilen Yapay Arı Kolonisi Algoritması

MHİAYD probleminde, hattın dengede olduğu varsayılmış ve ilk denge durumunda bir veya daha fazla istasyonun kapanmasına bağlı olarak hattın yeniden dengelenmesi amaçlanmıştır. Bu sebeple, ilk hat dengesini elde etmek amacıyla MHİAD problemini çözmek için bir YAK algoritması kullanılmıştır. İlk hat dengesini bulan YAK algoritmasında; girdi olarak görev sayısı, iş istasyonu sayısı, işçi sayısı, görevler arası öncelik ilişkileri ve tüm işçiler açısından görev süreleri alınmaktadır.

İlk hat dengesi elde edildikten sonra ilk algoritmaya oldukça benzer diğer bir YAK algoritması kullanılmakta ve MHİAYD problemi çözülmektedir. Bu algoritmada, ilk algoritma girdilerine ek olarak ilk dengeleme çözümü ve kapanan istasyon/istasyonların hangileri olduğu alınmakta ve hat yeniden dengelenmektedir.

Çalışma kapsamında geliştirilen iki YAK algoritması, işleyiş açısından aynı adımlara sahip iken girdileri, amaç fonksiyonları ve başlangıç çözümlerin sıfırdan veya bir ilk çözüme bağlı oluşturulması açısından farklıdır. İlk algoritmada, çevrim süresini enküçükleyen bir

hat dengesi bulmak, ikinci algoritmada ise bir veya daha fazla istasyonun kapanmasının ardından çevrim süresinde ve görev-istasyon atamalarında meydana gelen değişkenliği enküçüklemek amaçlanmıştır.

MHİAD ve MHİAYD problemlerinin çözümü için geliştirilen iki YAK algoritması da Şekil 4.1'de verilen sözde-koda uygun şekilde işlemektedir. Her ikisinde de algoritma belirlenen besin kaynağı sayısı kadar başlangıç çözümün rassal olarak türetilmesi ile başlar ve ardından görevli ve gözcü arı safhalarında komşu besin kaynakları bulunur. Limiti dolan besin kaynakları terkedilerek yerine yeni kaynaklar aranır. Yerel aramanın gerçekleştirilmesiyle bir iterasyonu tamamlanır ve bulunan en başarılı çözüm hafızada tutulur. İlk iterasyonun ardından ikinci iterasyon başlar ve görevli arı safhasından itibaren aynı adımlar tekrarlanır.

Bu çalışmada, klasik YAK algoritmasında kullanılan enbüyük iterasyon sayısı parametresi yerine enbüyük iterasyon gelişmeme sayısı kullanılmıştır. Buna göre bir iterasyon sonunda elde edilen en başarılı çözüm, geçen tüm iterasyonlar boyunca elde edilen en başarılı çözüm ile kıyaslanır ve daha başarılı bir çözüm bulunamadıysa iterasyon gelişmeme sayacı 1 birim artırılır. Aksi durumda, iterasyon gelişmeme sayacı sıfırlanır. Benzer şekilde, bir sonraki iterasyona geçilir ve enbüyük iterasyon gelişmeme sayısı aşılana kadar algoritma çalışmaya devam eder. Enbüyük iterasyon gelişmeme sayısı aşıldığında algoritma sonlanır ve tüm adımlar bir tekrara karşılık gelir. Algoritma sonunda elde edilen en başarılı çözüm, ilgili tekrar için bulunan çözümdür.

```
Parametreleri ayarla
Kâşif arıları rastgele yeni besin kaynaklarını keşfetmeleri için gönder
Kâşif arılar tarafından bulunan besin kaynaklarının kalitesini değerlendir
While İterasyon gelişmeme sayacı <= Enbüyük iterasyon gelişmeme sayısı
    Görevli arı safhası
    Gözcü arı safhası
    Kâşif arı safhası
    Yerel arama safhası
End While
```

Şekil 4.1. Önerilen YAK algoritmalarının sözde-kodu

İki YAK algoritması birbirine oldukça benzer olduğundan sadece bu çalışmanın asıl konusu olan MHİAYD problemini çözmek amacıyla geliştirilen algoritmanın safhaları, Akyol ve Baykasoğlu'nun (2019a) çalışmasında sunulan küçük boyutlu bir örnek problem

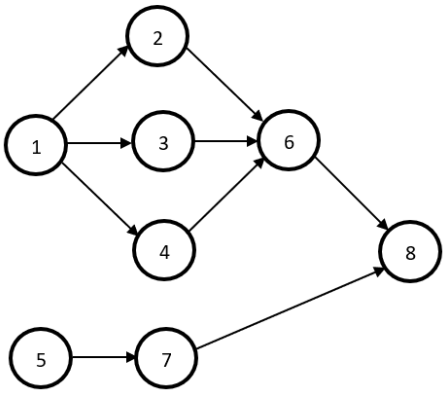
üzerinde açıklanmıştır. Örnek problem üzerinden yapılan açıklamalarda, amaç fonksiyonu değeri hesaplamalarında w değeri 0.95 olarak alınmış ve sonuçlar buna göre elde edilmiştir.

Örnek problem, 8 görevden oluşmakta ve görevlerin gerçekleştirilmesi için 3 işçi ve 3 istasyon bulunmaktadır. Çizelge 4.1'de, her bir işçinin görevleri gerçekleştirme süreleri verilmiştir. Örnek problemde 3 işçi bulunduğundan her bir görev için işçiler açısından 3 ayrı işlem süresi bulunmaktadır. X, ilgili görevin bu işçi tarafından yapılamadığını ifade etmektedir. Buna bağlı olarak, işlem süresinin X ile gösterildiği durumlarda ilgili görev o işçiye atanamamaktadır.

Çizelge 4.1. İşçi bazında işlem süreleri (Akyol ve Baykasoğlu, 2019a)

Görevler	İşçi-1	İşçi-2	İşçi-3
1	8	6	10
2	X	20	22
3	10	X	30
4	X	20	25
5	8	6	X
6	22	32	42
7	25	20	30
8	30	25	15

Görevler arası öncelik diyagramı Şekil 4.2'de verilmiştir. Bu diyagrama göre 6 numaralı görevi gerçekleştirebilmek için öncelikle öncülü olan 2, 3 ve 4 numaralı görevler yapılmalıdır. 1 ve 5 numaralı görevlerin öncülü yoktur.



Şekil 4.2. Örnek probleme ilişkin öncelik diyagramı (Akyol ve Baykasoğlu, 2019a)

4.3.1. Çözümlerin temsil edilmesi

Çözüm vektörünün elemanları tamsayılardan oluşmakta ve bu elemanlar, ilgili görev veya işçinin atandığı iş istasyonunu temsil etmektedir. Çözüm vektörü, görev-iş istasyonu atamaları ve işçi-iş istasyonu atamaları olmak üzere iki kısımdan oluşmaktadır. İlk bölümün uzunluğu görev sayısına ve ikinci bölümün uzunluğu ise işçi sayısına eşittir. 8 göreve, 3 istasyona ve 3 işçiye sahip bir örnek problem için çözüm vektörü Şekil 4.3'te verilmiştir. Bu vektörde G1, G2 ve G4 görevleri ile İ2 işçisinin iş istasyonu 1'e; G3 ve G5 görevleri ile İ1 işçisinin iş istasyonu 2'ye; G6, G7 ve G8 görevleri ile İ3 işçisinin iş istasyonu 3'e atandığı görülmektedir.

Görev atamaları								İşçi atamaları		
1	1	2	1	2	3	3	3	2	1	3
G1	G2	G3	G4	G5	G6	G7	G8	İ1	İ2	İ3

Şekil 4.3. Çözüm vektörü

Kapanan istasyona/istasyonlara görev atanamayacağı için çözüm vektörünün ilk kısmında kapalı olan istasyonların numaraları yer alamaz. İlk dengeleme sonucunda kapalı istasyonlara atanmış olan işçi ve görevler ise boşa çıkar. Bir işçinin boşa çıkması, yeniden dengeleme sonucunda da hiçbir istasyona atanmayacağı anlamına gelmez. Algoritmanın çözüm uzayında yaptığı aramalar sonucunda elde ettiği yeni hat dengesinde boştaki işçi veya işçiler açık bir istasyona atanabilir, farklı bir işçi boşta kalabilir. Görevler için böyle bir durum söz konusu değildir. Hattın yeniden dengelenmiş halinde, ilk hat dengesinde kapatılan istasyonda bulunan görevler de dahil olmak üzere her görev bir istasyona atanmış olmalıdır. Bu sebeple, her görevin ve işçinin numarası çözüm vektöründe yer almaktadır.

4.3.2. Montaj hattı işçi atama ve yeniden dengeleme problemi girdileri

Önerilen MHİAYD probleminde, mümkün olan en az değişkenlikle hattın yeniden dengelenmesi amaçlandığından problemin çözümü ilk hat dengesine bağlı gerçekleştirilir. Bunun için ilk dengeleme çözümüne ait atamalar MHİAD tip-2 probleminin ilk YAK algoritması ile çözülmesiyle elde edilir.

Örnek probleme ilişkin ilk hat dengesi çözüm vektörü Şekil 4.4'te verilmiştir. Bu çözümde G1, G2 ve G4 görevlerinin iş istasyonu 1'e; G3, G5 ve G6 görevlerinin iş istasyonu 2'ye; G7 ve G8 görevlerinin ise iş istasyonu 3'e atandığı görülmektedir. Çözüm vektörünün ikinci kısmı; İ1 işçisinin iş istasyonu 2'ye, İ2 işçisinin iş istasyonu 1'e ve İ3 işçisinin iş istasyonu 3'e atandığını göstermektedir.

1	1	2	1	2	2	3	3	2	1	3

Şekil 4.4. İlk hat dengesi çözüm vektörü

Bir istasyonun yükü, ilgili istasyondaki görevlerin o istasyona atanan işçi tarafından gerçekleştirilme sürelerinin toplanmasıyla elde edilmektedir. Çizelge 4.1'de verilen işlem sürelerine bağlı olarak her istasyonun yükü hesaplanır. İlk istasyondaki İ2 işçisinin G1, G2 ve G4 görevlerini gerçekleştirme süreleri toplamı 46 (6+20+20) saniye olarak bulunur. Benzer şekilde, 2. ve 3. istasyonlar için de İ1 ve İ3 işçileri açısından hesaplamalar yapıldığında bu istasyonların toplam süreleri sırasıyla 40 (10+8+22) saniye ve 45 (30+15) saniye olarak elde edilir. En büyük istasyon yükü çevrim süresini verdiğinden bu çözümün çevrim süresi 46 saniye olur.

Çevrim süresinin enküçüklendiği ilk hat dengesinin bulunmasının ardından kapatılan istasyon/istasyonlar rassal şekilde belirlenmektedir. Görev sayısı, iş istasyonu sayısı, işçi sayısı, görevlerin her işçi açısından işlem süreleri, görevler arası öncelik ilişkileri, ilk hat dengesi görev-iş istasyonu ve işçi-iş istasyonu atamalarına ek olarak kapatılan istasyonun/istasyonların numarası MHİAYD probleminin çözümü için geliştirilen algoritmada girdi olarak kullanılmaktadır.

4.3.3. Montaj hattı işçi atama ve yeniden dengeleme problemi safhaları

4.3.3.1. Başlangıç çözümlerin oluşturulması

Algoritma, besin kaynağı sayısı kadar olurlu çözümün rassal olarak türetilmesi ile başlar. Bu çalışmada, başlangıç çözümleri türetirken iki farklı yol kullanılmıştır. İlk yolda, ilk hat dengesine ait görev-iş istasyonu ve işçi-iş istasyonu atamaları alınarak kapanan istasyondaki görevlerin açık olanlara dağıtılması yoluna gidilmiştir. Kapanan

istasyonlardaki görevlerin açık olanlara taşınmasında, görevler arası öncelik ilişkilerinin sağlanmasına ve istasyonu değişen görevin ilgili istasyondaki işçi tarafından yapılabilir olmasına dikkat edilmektedir. Böylece, her zaman olurlu çözümlerin türetilmesi sağlanmaktadır. Bunun için öncelikle ilk durumda atandıkları istasyon kapanan görevler için öncelik ilişkilerine uygun bir atama sırası rassal şekilde oluşturulur. Ardından, bu sıra göz önünde bulundurularak her bir görev kısıtlar açısından uygun olan bir iş istasyonuna atanır. Tüm boşa çıkmış görevlerin yeni istasyonlarına atanması ile bir başlangıç çözüm oluşturulmuş olur.

Diğer yolda ise ilk hat dengesinin sadece görev-iş istasyonu atamaları korunmaktadır. Burada öncelikle, ilk denge durumundaki işçi-iş istasyonu atamaları göz ardı edilmekte ve kapalı istasyon/istasyonlardaki görevler, sadece öncelik ilişkileri dikkate alınarak uygun olan açık istasyonlara dağıtılmaktadır. Bunun için ilk yolda olduğu gibi öncelik ilişkilerine olurlu bir görev atama sırası rassal şekilde oluşturulur. Ardından, işçi atamaları gerçekleştirilir. Olurlu bir çözüm elde etmek için her bir işçi, o istasyona atanan görevlerin tümünü yapabileceği bir istasyona atanmalıdır. İşçilerin istasyonlara atanmasında her bir istasyona, ilgili istasyona atanan tüm görevleri yapabilecek işçi sayısına göre öncelik verilmektedir. En az sayıda işçinin atanmasının mümkün olduğu istasyon en büyük önceliğe sahip olur ve öncelikle bu istasyona işçi ataması gerçekleştirilir. Her atamadan sonra öncelik sırası güncellenir. Bu şekilde, olurlu çözüm oluşturma olasılığı artırılmış olur.

Popülasyonda çeşitlilik sağlamak amacıyla her bir çözümün hangi yolla oluşturulacağı rassal olarak seçilir. Olurlu bir çözümün bulunmasının ardından bir sonraki çözümü türetmek için aynı adımlar tekrarlanır. Besin kaynağı sayısı kadar olurlu çözümün elde edilmesiyle başlangıç çözüm oluşturma aşaması tamamlanır.

İlk dengeleme atamaları Şekil 4.4'te verilmiş olan örnek problem için iş istasyonu 2'nin kapatıldığı durumda türetilen bir başlangıç çözüm Şekil 4.5'te verilmiştir. Burada, 2. istasyonun kapatılmasının ardından G1, G2, G4 ve G5 görevleri iş istasyonu 1'e ve kalan tüm görevlerin ise iş istasyonu 3'e atandığı görülmektedir.

1	1	3	1	1	3	3	3	2	1	3

Şekil 4.5. Başlangıç çözüm vektörü

Yeni durumda, ilk denge durumunda kapanan istasyonda bulunan görevlerden G5 iş istasyonu 1'e, G3 ve G6 ise iş istasyonu 3'e taşınmıştır. Şekil 4.5'te görüldüğü üzere işçi atamaları ilk denge durumuna göre değişmemiş ve iş istasyonu 2'de yer alan İ1 işçisi boşa çıkmıştır. Bu çözümde, istasyon yükleri 1. ve 3. istasyonlar için sırasıyla 52 saniye ve 117 saniye olduğundan çevrim süresi 117 saniye olarak bulunur.

İstasyon 2'nin kapanması ile toplam 3 görev boşa çıkmıştır. Şekil 4.4 ve Şekil 4.5 karşılaştırıldığında sadece istasyonu kapanan görevlerin atamalarında değişim olduğu, diğer görevlerin istasyonlarının değişmediği görülmektedir. Kısaca, kapanan istasyonlarda ilk hat dengesinde bulunan görev sayısı ve yeniden atanan görev sayısı eşittir. Amaç fonksiyonu değeri, Denklem (4.12)'de ilgili değerlerin yerine yazılmasıyla 1.47 olarak bulunur.

4.3.3.2. Görevli arı safhası

Başlangıç çözümler, olurlu ancak rassal şekilde oluşturuldukları için genellikle kötü amaç fonksiyonu değerine sahip çözümlerdir. Bu sebeple, bu çözümleri geliştirmek ve daha iyi çözümler elde etmek gerekir. Çözümleri geliştirmek için YAK algoritmasındaki ilk aşama, sırayla tüm çözümler için komşu çözümlerin bulunduğu görevli arı safhasıdır. Komşu çözümler, mevcut çözüm üzerinde çeşitli değişiklikler yapılarak elde edilir.

Bu çalışmada komşu çözüm arama yapısı olarak; bir görevin bir istasyondan alınarak diğerine atanması, iki görevin karşılıklı istasyonlarının değiştirilmesi ve iki işçinin atandıkları istasyonların karşılıklı değiştirilmesi olmak üzere 3 durumun farklı çeşitlerinden oluşan 7 komşuluk yapısı kullanılmıştır. Kullanılan komşuluk yapıları aşağıdaki gibi açıklanabilir:

- En büyük iş yüküne sahip istasyondan bir görev alınarak uygun olan en az yüklü istasyona atanması (Bu komşuluk yapısında, en yüklü istasyondan bir görevin rassal olarak seçilmesinin ardından en az yüke sahip iş istasyonundan başlayarak o görevin atanabileceği ilk istasyon seçilir ve görev taşınır.)

- En büyük iş yüküne sahip istasyondan bir görev alınarak uygun olan farklı bir istasyona atanması
- En büyük iş yüküne sahip istasyondan ilk denge durumunda mevcut durumda kapalı olan istasyonda bulunan görevlerden birinin alınarak uygun olan farklı bir istasyona atanması
- Bir görevin atandığı istasyondan alınarak uygun olan farklı bir istasyona atanması
- En büyük iş yüküne sahip istasyon ile farklı bir istasyon arasında karşılıklı görev değişimi yapılması
- Farklı iki istasyon arasında karşılıklı görev değişimi yapılması
- İki işçinin atandığı istasyonların karşılıklı değiştirilmesi

İlgili çözüm için hangi komşuluk yapısı ile komşu çözüm bulunacağı rassal şekilde belirlenir. Tüm komşuluk yapıları ile öncelik ilişkileri ve görev-işçi uygunlukları dikkate alınarak komşu çözümler oluşturulur. Kapalı istasyonlara herhangi bir görev atanmaz, ancak mevcut durumda kapalı istasyonda bulunan yani boşta kalmış olan bir işçinin herhangi bir işçi ile yer değiştirerek açık bir iş istasyonuna atanması ve diğerinin boşa çıkması mümkündür.

Oluşturulan komşu çözümün amaç fonksiyonu değeri, mevcut çözümünkinden daha iyiyse komşu çözüm mevcut çözümün yerini alır ve ilgili çözümün üst üste kaç denemedir iyileşmediğini ifade eden çözüm gelişmeme sayacı sıfırlanır. Bu algoritmada, klasik YAK algoritmasından farklı olarak mevcut çözümden belirli bir oranda daha kötü komşu çözümler de mevcut çözümün yerine geçmesi için kabul edilmektedir. Bunun sebebi, mevcut çözümden az oranda kötü komşuların daha sonraki iterasyonlarda mevcut olandan daha fazla iyileştirilebilme olasılığının bulunmasıdır. Belirlenen orandan daha büyük kötüleşmelerde ise mevcut çözüm korunur ve o çözüm için çözüm gelişmeme sayacı 1 birim artırılır. Popülasyondaki tüm çözümler için aynı adımlar tekrarlandıktan sonra bu safha sonlanır. Görevli arı safhasının sözde kodu Şekil 4.6’da verilmiştir.

```
For her görevli arı
        Görevli arıyı bir sonraki besin kaynağına gönder
        Besin kaynağına komşuluk yapılarından birini uygula
        Görevli arı tarafından bulunan besin kaynağının kalitesini değerlendir
        If Komşu besin kaynağı mevcut besin kaynağından daha kaliteliyse
                Besin kaynağını güncelle
                Çözüm gelişmeme sayacı (besin kaynağı) = 0
                If Yeni besin kaynağı en iyi besin kaynağından daha kaliteliyse
                        En iyi besin kaynağını güncelle
                End If
        Else If Komşu besin kaynağı mevcut besin kaynağından az oranda daha kalitesizse
                Besin kaynağını güncelle
                Çözüm gelişmeme sayacı (besin kaynağı) = 0
        Else
                Çözüm gelişmeme sayacı (besin kaynağı) = Çözüm gelişmeme sayacı (besin kaynağı)+1
        End If
End For
```

Şekil 4.6. Görevli arı safhasının sözde-kodu

Şekil 4.5'te verilen başlangıç çözüm için görevli arı safhasında bulunan bir komşu çözüm Şekil 4.7'de görülmektedir. Şekil 4.5'te ve Şekil 4.6'de verilen atamalar karşılaştırıldığında, G7 görevinin 3. istasyondan 1. istasyona taşındığı görülmektedir. İşçi atamaları ise birbiriyle aynıdır.

1	1	3	1	1	3	1	3	2	1	3

Şekil 4.7. Görevli arı safhasında bulunan komşu çözüm vektörü

Bulunan komşu çözümde 1 ve 3 numaralı istasyonların yükleri sırasıyla 72 saniye ve 87 saniye olup çevrim süresi 87 saniye olarak bulunur. İlk hat dengesiyle karşılaştırıldığında istasyonu değişen görev sayısı 4 ve ilk çevrim süresi 46 saniye olduğundan amaç fonksiyonu değeri Denklem (4.12) ile hesaplanarak 0.85 olarak bulunur. Amaç fonksiyonu değeri 1.47 olan mevcut çözümden daha iyi bir çözüm bulunduğundan komşu çözüm kabul edilerek mevcut olanın yerini alır ve bu çözüme ait gelişmeme sayacı sıfırlanır.

4.3.3.3. Gözcü arı safhası

Gözcü arı safhasında görevli arı safhasından farklı olarak tüm çözümler için değil, turnuva seçimi ile belirlenen çözümler için komşu çözümler aranmaktadır. Kullanılan komşu çözüm arama yapıları, görevli arı safhasındakiler ile aynıdır. Komşuları aranacak çözümlerin belirlenmesinde ikili turnuva seçim yöntemi kullanılmıştır. Bu yönteme göre popülasyondan rastgele iki çözüm belirlenir ve daha iyi amaç fonksiyonu değerine sahip olan çözüm

seçilerek komşusu aranır. Turnuva seçimi rassal bir işlem olduğundan bir çözümün birden fazla kez komşu çözüm aramak için seçilmesi veya hiç seçilmemesi mümkündür.

Bulunan komşu çözüm, görevli arı safhasında olduğu gibi mevcut çözüm ile karşılaştırılır. Komşu çözümün amaç fonksiyonu değeri mevcut çözümünkinden daha iyiyse komşu çözüm mevcut olanın yerini alır ve çözüme ait gelişmeme sayacı sıfırlanır. Amaç fonksiyonu değerinde en fazla belirli bir yüzdeye kadar olan kötüleşmeye sahip komşu çözümler de kabul edilerek mevcut çözümün yerini alır ve gelişmeme sayacı sıfırlanır. Mevcut çözümden belirlenen yüzdeden daha fazla kötüleşmeye sebep olan çözümler reddedilir, mevcut çözüm korunur ve o çözüm için sayaç 1 birim artırılır. Aynı adımlar besin kaynağı sayısı kadar tekrarlanır. Gözcü arı safhasının sözde-kodu Şekil 4.8'de verilmiştir.

```
For her gözcü arı
        Rassal olarak iki besin kaynağı seç
        İkili turnuva seçimi yöntemi uygula
        Gözcü arıyı seçilen besin kaynağına gönder
        Besin kaynağına komşuluk yapılarından birini uygula
        Gözcü arı tarafından bulunan besin kaynağının kalitesini değerlendir
        If Komşu besin kaynağı mevcut besin kaynağından daha kaliteliyse
                Besin kaynağını güncelle
                Çözüm gelişmeme sayacı (besin kaynağı) = 0
                If Yeni besin kaynağı en iyi besin kaynağından daha kaliteliyse
                        En iyi besin kaynağını güncelle
                End If
        Else If the Komşu besin kaynağı mevcut besin kaynağından az oranda daha kalitesizse
                Besin kaynağını güncelle
                Çözüm gelişmeme sayacı (besin kaynağı) = 0
        Else
                Çözüm gelişmeme sayacı (besin kaynağı) = Çözüm gelişmeme sayacı (besin kaynağı)+1
        End If
End For
```

Şekil 4.8. Görevli arı safhasının sözde-kodu

Şekil 4.7'de verilen çözüm için gözcü arı safhasında bulunan bir komşu çözüm Şekil 4.9'da gösterilmiştir. Şekil 4.7 ve Şekil 4.9'da verilen atamalar karşılaştırıldığında görev-iş istasyonu atamalarının aynı olduğu görülmektedir. Ancak, işçi atamaları farklıdır. Yeni durumda, İ1 ve İ3 işçilerinin yer değiştirdiği görülmektedir. Böylelikle, İ1 işçisi 3. istasyona atanırken İ3 işçisi boşa çıkmıştır.

1	1	3	1	1	3	1	3	3	1	2

Şekil 4.9. Gözcü arı safhasında bulunan komşu çözüm vektörü

Bulunan komşu çözümde 1 ve 3 numaralı istasyonların yükleri sırasıyla 72 saniye ve 62 saniye olup çevrim süresi 72 saniye olarak bulunur. İlk hat dengesiyle karşılaştırıldığında istasyonu değişen görev sayısı 4 ve ilk çevrim süresi 46 saniye olduğundan amaç fonksiyonu değeri Denklem (4.12) ile hesaplanarak 0.54 olarak bulunur. Amaç fonksiyonu değeri 0.85 olan mevcut çözümden daha iyi bir çözüm bulunduğundan komşu çözüm kabul edilerek mevcut olanın yerini alır ve bu çözüme ait gelişmeme sayacı sıfırlanır.

4.3.3.4. Kâşif arı safhası

Klasik YAK algoritmasının kâşif arı safhasında, çözüm gelişmeme sayacı limit değerini aşan çözümler terkedilir ve terkedilen çözümlerin yerine arama uzayının farklı bölgelerinde rassal şekilde yeni çözümler aranarak çözüm popülasyonuna dahil edilir.

Rassal şekilde türetilen olurlu çözümlerin amaç fonksiyonu, genellikle çok iyi bir değere sahip olmamaktadır. Bu algoritmada, iyi çözümlere daha hızlı bir şekilde ulaşabilmek adına kâşif arı safhasında, terkedilen bir çözüm yerine belirli bir olasılıkla rassal ve olurlu bir çözüm oluşturma veya popülasyondaki en başarılı çözümün bir komşusunu arayarak bu çözümün yerine koyma stratejisi uygulanmaktadır.

Rassal çözüm oluşturma stratejisinin seçilmesi durumunda, terkedilen çözümün yerine başlangıç çözüm oluşturma aşamasında olduğu gibi yeni bir çözüm türetilir. Diğer stratejide ise görevli ve gözcü arı safhalarında kullanılan komşuluk yapıları ile popülasyondaki en başarılı çözümün bir komşusu bulunur ve bu çözüm mevcut çözümün yerini alır.

Bu safhada, klasik YAK algoritmasından farklı olarak o zamana kadar bulunmuş en başarılı değere sahip olan çözümlerin sayaçları limit değerini aşsa dahi kendisinden daha iyi komşu çözümlerinin bulunma olasılığının devam ettirilebilmesi için terkedilmemekte ve çözüm gelişmeme sayaçları sıfırlanmaktadır. Bu değişiklikler sayesinde iyi çözümlere

ulaşabilme hızı ve olasılığı artmaktadır. Kâşif arı safhasının sözde kodu Şekil 4.10'da verilmiştir.

```
For her besin kaynağı
        If Çözüm gelişmeme sayacı (besin kaynağı)>Limit
                If besin kaynağı = en iyi besin kaynağı
                        Çözüm gelişmeme sayacı (besin kaynağı) = 0
                Else
                        Besin kaynağını terket
                        r = (0-1 arasında rassal bir değer)
                        If r<=Belirlenen oran
                                Kaşif arıyı yeni bir besin kaynağı araması için gönder
                                Besin kaynağı = Yeni besin kaynağı
                                        Çözüm gelişmeme sayacı (besin kaynağı) = 0
                                If Yeni besin kaynağı en iyi besin kaynağından daha kaliteli
                                        En iyi besin kaynağını güncelle
                                End If
                        Else
                                En iyi besin kaynağına komşuluk yapılarından birini uygula
                                Besin kaynağı = En iyi besin kaynağının bir komşusu
                                Çözüm gelişmeme sayacı (besin kaynağı) = 0
                                        If Yeni besin kaynağı en iyi besin kaynağından daha kaliteli
                                                En iyi besin kaynağını güncelle
                                         End If
                        End If
                End If
        End If
End For
```

Şekil 4.10. Kâşif arı safhasının sözde-kodu

4.3.3.5. Yerel arama safhası

İyi çözümlere daha hızlı ulaşabilmek amacıyla bu algoritmaya klasik YAK algoritmasında bulunmayan yerel arama safhası eklenmiştir. Yerel arama, popülasyondaki her bir çözüme belirli bir olasılıkla uygulanır. Yerel aramada, mevcut çözümde en yüklü istasyondan olurlu tüm görev taşımaları ve en yüklü istasyon ile diğer tüm istasyonlar arasındaki işçi-iş istasyonu atama değişimleri birbirinden bağımsız olarak kontrol edilir ve amaç fonksiyonunda en fazla iyileşmeyi sağlayan durum mevcut çözüm olarak alınır. Görevli arı ve gözcü arı safhalarının aksine, yerel aramada mevcut çözümde herhangi bir kötüleşme kabul edilmez. Yerel arama safhasının sözde kodu Şekil 4.11'de verilmiştir.

Yerel aramanın sona ermesinin ardından bir iterasyon tamamlanmış olur. İlgili iterasyonda elde edilen en başarılı çözüm, yerel en başarılı çözüm olarak adlandırılır. O zamana kadar geçen tüm iterasyonlar boyunca elde edilen en başarılı çözüm ise bütünsel en

başarılı çözümdür. Her iterasyon sonunda, yerel ve bütünsel en başarılı çözümler birbiri ile kıyaslanır. Bir iterasyon sonunda, bütünsel en başarılı çözüm geliştirilemediğinde o iterasyonda bir iyileşme sağlanamadığını gösteren iterasyon gelişmeme sayacı 1 birim artırılır. Aksi takdirde, bu sayaç sıfırlanır. İterasyon gelişmeme sayacı, bütünsel en başarılı çözümün ardı ardına kaç iterasyondur iyileştirilemediğini gösterir ve sayaç değeri belirlenen enbüyük iterasyon gelişmeme sayısını aştığında algoritma sonlanır. Eğer belirlenen değer henüz aşılmadıysa, görevli arı safhasından itibaren yeni bir iterasyon başlar.

```
For her besin kaynağı
p = (0-1 arasında rassal bir değer)
        If p>=Yerel arama olasılığı
                Besin kaynağına yerel arama uygula
                If Yerel arama ile bulunan besin kaynağı mevcut besin kaynağından daha kaliteli
                        Besin kaynağını güncelle
                        If Yeni besin kaynağı en iyi besin kaynağından daha kaliteli
                                En iyi besin kaynağını güncelle
                        End If
                End If
        End If
End For
```

Şekil 4.11. Yerel arama safhasının sözde-kodu

5. BULGULAR VE TARTIŞMA

5.1. Kullanılan Test Problemleri

Bu çalışmada ele alınan problemin çözümü geliştirilen yöntemler, Intel Core i5 1.80GHz işlemciye ve 8 GB RAM'e sahip bir bilgisayarda sınanmıştır. Modelin ve algoritmaların performansını karşılaştırabilmek için Chaves vd. (2007) tarafından sunulan MHİAD test problemleri kullanılmıştır. Bu test problemleri; Roszieg, Heskia, Tonge ve Wee-Mag olmak üzere 4 problem ailesinden oluşmaktadır. Her problem ailesinde 8 grup bulunmakta ve her grup ise 10 test problemi içermektedir. Bu problem ailelerine ait görev sayısı, işçi sayısı ve test problemleri için bir karmaşıklık indeksi olan sıra kuvveti Çizelge 5.1'de verilmiştir.

Sunulan çözüm yöntemlerini karşılaştırmak amacıyla problem ailelerinin tüm gruplarının ilk 3 test problemi alınarak toplam 96 test problemi üzerinde sonuçlar elde edilmiştir.

Çizelge 5.1. MHİAD problemi test problemlerinin özellikleri (Chaves vd., 2007)

Aile	Görev sayısı	İşçi sayısı	Sıra kuvveti
Roszieg	25	4 (Grup 1-4) ve 6 (Grup 5-8)	71.67
Heskia	28	4 (Grup 1-4) ve 7 (Grup 5-8)	22.49
Tonge	70	10 (Grup 1-4) ve 17 (Grup 5-8)	59.42
Wee-Mag	75	11 (Grup 1-4) ve 19 (Grup 5-8)	22.67

5.2. Montaj Hattı İşçi Atama ve Dengeleme Probleminin Çözülmesi

MHİAYD problemini çözmek için ilk hat dengesi çözümü gerektiğinden, öncelikle 96 test problemi için ilk YAK algoritması kullanılarak MHİAD tip-2 problemi çözülmüş ve çevrim süresi değeri literatürde bulunan en başarılı çözümlere yakın ilk dengeleme çözümleri elde edilmiştir.

Geliştirilen ilk YAK algoritması, MATLAB (Matrix Laboratory) 2018b programlama dilinde kodlanmıştır. Her test probleminin çözümü için 20 tekrar yapılmıştır. Besin kaynağı sayısı 50 ve enbüyük iterasyon gelişmeme sayısı ise 2000 olarak belirlenmiştir. Limit değeri 50 olarak alınmıştır. Görevli ve gözcü arı safhalarında, en fazla

%5 oranında kötüleşmeye sebep olan komşu çözümler mevcut olanın yerine geçmesi için kabul edilmiştir. Kâşif arı safhasında terkedilen çözümlerin yerine eşit olasılıkla rassal bir çözüm veya popülasyondan rassal şekilde seçilen bir çözümün bir komşusu üretilmiştir. Kâşif arı safhasının ardından her çözüme %30 olasılıkla yerel arama uygulanmıştır.

5.3. Montaj Hattı İşçi Atama ve Yeniden Dengeleme Probleminin Çözülmesi

Çalışma kapsamında kullanılan test problemleri için ilk dengeleme çözümleri elde edildikten sonra her bir örnek problem için belirlenen sayıda istasyon rassal şekilde seçilerek kapatılmıştır. Ardından, geliştirilen matematiksel model ve YAK algoritması kullanılarak çevrim süresinde ve görev atamalarında meydana gelen değişkenliği enküçüklemek amaçlanmıştır.

Kapatılan istasyon sayıları, problemlerde bulunan başlangıç istasyon sayısına göre belirlenmiştir. Bazı test problemlerinde, kapatılan istasyon sayısı açısından 2 farklı duruma ait çözümler elde edilmiştir. Böylelikle, toplamda 120 MHİAYD problemi çözülmüştür. Problemlerde bulunan istasyon sayısı açısından kapatılan istasyon sayıları Çizelge 5.2'de verilmiştir.

Çizelge 5.2. Başlangıç istasyon sayısı açısından kapanan istasyon sayıları

Başlangıç istasyon sayısı	Kapatılan istasyon sayısı
4	1
6	1
7	2
10	2
11	2
17	2 ve 3
19	2 ve 3

5.3.1. Matematiksel model ile çözüm

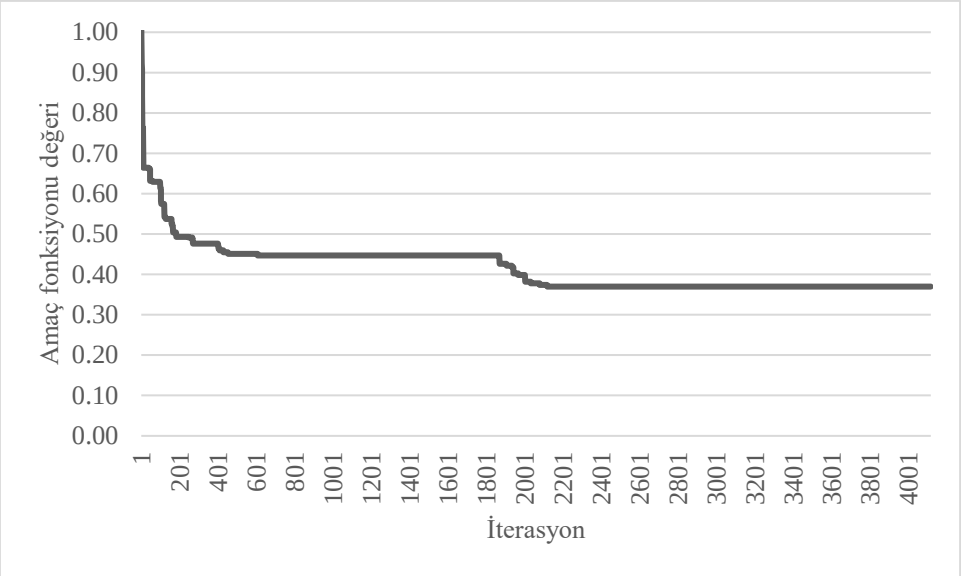
Problemin çözümü için sunulan matematiksel model, GAMS (The General Algebraic Modeling System) programı ile kodlanmış ve çözücü olarak CPLEX kullanılmıştır. Geliştirilen matematiksel model ile her bir test probleminin çözülmesi için 3600 saniye süre limiti verilmiştir.

5.3.2. Yapay arı kolonisi algoritması ile çözüm

MHİAYD probleminin çözümü için geliştirilen YAK algoritması MATLAB 2018b programlama dilinde kodlanmıştır. Her test probleminin çözümü için 20 tekrar yapılmıştır. Besin kaynağı sayısı 50 ve enbüyük iterasyon gelişmeme sayısı ise 2000 olarak belirlenmiştir. Limit değeri 50 olarak alınmıştır. Görevli ve gözcü arı safhalarında, yapılan toplam iterasyon sayısının 2000'den küçük ve 2000'e eşit olduğu durumlarda mevcut çözümde en fazla %5 oranında kötüleşmeye sebep olan komşu çözümler mevcut olanın yerine geçmesi için kabul edilmiştir. Yapılan toplam iterasyonun 2000'den fazla olduğu durumlarda ise en fazla %2 oranındaki kötüleşmelere izin verilmiştir. Kâşif arı safhasında terkedilen çözümlerin yerine %20 olasılıkla rassal bir çözüm, %80 olasılıkla popülasyondan rassal şekilde seçilen bir çözümün bir komşusu türetilmiştir. Kâşif arı safhasının ardından her çözüme %30 olasılıkla yerel arama uygulanmıştır. Çevrim süresi, görevlerin iş istasyonu değişiminden daha önemli bir performans ölçütü olarak görüldüğünden amaç fonksiyonunda ağırlık değeri (w), 0.95 olarak alınmıştır.

Örnek olması amacıyla T1 probleminin YAK algoritması ile çözümü sırasında yapılan bir tekrarda geçen iterasyon sayısına bağlı olarak amaç fonksiyonu değerinin değişimi Şekil 5.1'de verilmiştir.

T1 problemi için ilk hat dengesinin çevrim süresi 87 saniyedir. 2 ve 5 numaralı istasyonların kapanmasının ardından 12 görev boşa çıkmıştır. Türetilen başlangıç çözümlerden en başarılı olanın çevrim süresi 179 saniye ve iş istasyonu değişen görev sayısı 12 olup amaç fonksiyonu değeri ise 1.01 olarak bulunmuştur. Algoritmanın durma koşulu olan üst üste 2000 iterasyon boyunca iyileştirilememe durumu sağlandığında toplam 4118 iterasyon yapılmıştır. Sonuç olarak çevrim süresi 114 saniye ve iş istasyonu değişen görev sayısı 30 olan bir çözüm elde edilmiştir. Bu çözümün amaç fonksiyonu değeri 0.37'dir.



Şekil 5.1. T1 problemi için iterasyon sayısına bağlı amaç fonksiyonu değeri değişimi

5.4. Elde Edilen Sonuçlar

Her bir problem ailesinin test problemlerine ait ilk dengeleme ve yeniden dengeleme sonuçları Çizelge 5.3-5.6'da verilmiştir. İlk 4 sütun sırasıyla; problem adını (P), kapanan istasyon/istasyonların numarasını (F), kapanan istasyon/istasyonlarda ilk durumda bulunan toplam görev sayısını (f) ve ilk dengeleme çözümüne ait çevrim süresi değerini (C_0) vermektedir. 5-9. sütunlar matematiksel modele ve 10-15. sütunlar ise YAK algoritmasına ilişkin sonuçları göstermektedir. C yeniden dengeleme çözümüne ilişkin çevrim süresidir. T istasyonu değişen görev sayısını ifade etmektedir, z ise amaç fonksiyonu değeridir. Her bir test problemi için koyu renk ile işaretlenen değerler, ilgili problem için elde edilen en başarılı amaç fonksiyonu değerini temsil etmektedir.

Matematiksel model çözümüne ait süre değerleri 3600 saniyenin altında olan sonuçlar, ilgili problemin optimal çözüldüğünü göstermektedir. Optimal çözümün bulunamadığı problemler için ise süre limiti içinde bulunan en başarılı çözüme ait değerler verilmiştir.

YAK algoritmalarına ait z_{eniyi} değeri, her bir test problemi için yapılan 20 tekrar içinde bulunan en başarılı amaç fonksiyonu değerini göstermektedir. z_{ort} ise ilgili test problemi için yapılan tüm tekrarlarda elde edilen amaç fonksiyonu değerlerinin ortalamasını

verir. Son sütunda, yapılan tekrarlar içinde en başarılı çözümün elde edildiği en iyi süre değerleri yer almaktadır.

Matematiksel model ve YAK algoritmasından elde edilen sonuçlar, alt sınır ve çözüm arasındaki ağırlıklı Öklid uzaklığı ile karşılaştırılmıştır. Bu uzaklık, Denklem (5.1) ile hesaplanmaktadır. Burada, d öklid uzaklığını göstermektedir.

$$d = \sqrt{w(C - C_0)^2 + (1 - w)(T - f)^2} \tag{5.1}$$

Matematiksel model, küçük boyutlu problem aileleri olan Roszieg ve Heskia'ya ait tüm test problemlerini optimal çözmeyi başarmıştır. YAK algoritması, bu problemlerin tamamında matematiksel model ile elde edilen optimal çözümlere ulaşmıştır. En başarılı çözümlere ulaşma süresi açısından 46 problemde YAK algoritması, 2 problemde ise matematiksel model üstünlük sağlamıştır.

Model, Tonge ailesine ait 36 test probleminden yalnızca 2 tanesini (T11 ve T23) verilen süre limiti içinde optimal çözebilmiştir. YAK algoritması, bu problemler için aynı çözümleri çok daha kısa süreler içinde elde etmiştir. YAK algoritması, bu problem ailesine ait kalan 34 problem için matematiksel model ile verilen süre limiti içinde elde edilenlerden daha başarılı sonuçlara ulaşmıştır.

En büyük boyutlu problem ailesi olan Wee-Mag ailesi için matematiksel model verilen süre limiti içinde hiçbir problemi optimal çözememiştir. YAK algoritması, tüm Wee-Mag problemleri için matematiksel model ile süre limiti içinde elde edilenler ile karşılaştırıldığında daha başarılı çözümlere oldukça makul süreler içinde ulaşmıştır.

Elde edilen sonuçlara göre; 120 test probleminin 70 tanesinde YAK algoritması daha başarılı çözümler elde ederken, 50 tanesinde ise her iki yöntem aynı çözümlere ulaşmıştır. YAK algoritmasının, en başarılı çözümleri elde etme süresi açısından test problemlerinin 118'inde daha başarılı olduğu görülmüştür.

Çizelge 5.3. Roszieg problem ailesine ait sonuçlar

				Model				YAK					Süre	
P	F	f	C_0	C	T	z	d	C	T	z_{eniyi}	z_{ort}	d	Model	YAK
R1	2	7	20	28	9	**0.39***	7.81	28	9	**0.39**	0.39	7.81	0.55	0.05
R2	4	7	22	30	9	**0.36***	7.81	30	9	**0.36**	0.36	7.81	1.14	0.02
R3	3	4	18	26	5	**0.43***	7.80	26	5	**0.43**	0.43	7.80	0.69	0.02
R11	2	3	30	35	3	**0.16***	4.87	35	3	**0.16**	0.16	4.87	0.64	0.02
R12	3	8	32	45	12	**0.41***	12.70	45	12	**0.41**	0.76	12.70	0.64	13.54
R13	2	7	83	83	7	**0.00***	0.00	83	7	**0.00**	0.00	0.00	0.61	0.02
R21	4	7	28	39	12	**0.41***	10.78	39	12	**0.41**	0.41	10.78	0.75	0.06
R22	1	5	30	42	11	**0.44***	11.77	42	11	**0.44**	0.44	11.77	0.94	0.06
R23	4	7	26	35	13	**0.37***	8.87	35	13	**0.37**	0.37	8.87	0.59	0.06
R31	1	5	31	41	13	**0.39***	9.91	41	13	**0.39**	0.39	9.91	0.80	0.12
R32	2	8	29	38	15	**0.34***	8.91	38	15	**0.34**	0.34	8.91	0.74	0.36
R33	4	7	32	43	12	**0.36***	10.78	43	12	**0.36**	0.36	10.78	0.72	0.06
R41	4	7	10	11	11	**0.12***	1.32	11	11	**0.12**	0.12	1.32	3.05	0.06
R42	1	3	10	11	14	**0.28***	2.65	11	14	**0.28**	0.28	2.65	3.09	0.21
R43	6	4	10	12	9	**0.25***	2.25	12	9	**0.25**	0.25	2.25	2.23	0.06
R51	3	6	11	12	11	**0.13***	1.48	12	11	**0.13**	0.13	1.48	1.84	0.15
R52	3	6	10	12	7	**0.20***	1.96	12	7	**0.20**	0.20	1.96	1.75	0.06
R53	4	4	10	15	7	**0.51***	4.92	15	7	**0.51**	0.51	4.92	2.38	0.04
R61	6	4	16	19	8	**0.23***	3.06	19	8	**0.23**	0.23	3.06	3.13	0.05
R62	4	5	13	15	6	**0.16***	1.96	15	6	**0.16**	0.16	1.96	2.94	0.02
R63	4	5	19	24	10	**0.30***	5.00	24	10	**0.30**	0.30	5.00	3.75	0.10
R71	5	5	15	18	10	**0.24***	3.13	18	10	**0.24**	0.24	3.13	2.38	0.14
R72	1	3	16	18	11	**0.25***	2.65	18	11	**0.25**	0.25	2.65	2.81	0.12
R73	2	3	16	19	13	**0.34***	3.68	19	13	**0.34**	0.34	3.68	2.69	0.58

*optimal çözüm

Çizelge 5.4. Heskia problem ailesine ait sonuçlar

				Model				YAK					Süre	
P	F	f	C_0	C	T	z	d	C	T	z_{eniyi}	z_{ort}	d	Model	YAK
H1	2	6	94	156	10	**0.66***	60.44	156	10	**0.66**	0.66	60.44	0.47	0.05
H2	2	9	95	111	15	**0.19***	15.65	111	15	**0.19**	0.19	15.65	0.61	0.08
H3	1	10	102	124	23	**0.27***	21.64	124	23	**0.27**	0.27	21.64	0.72	0.44
H11	2	7	169	250	12	**0.49***	78.96	250	12	**0.49**	0.49	78.96	0.72	0.13
H12	3	3	107	167	7	**0.60***	58.49	167	7	**0.60**	0.60	58.49	0.61	0.08
H13	2	1	108	156	4	**0.57***	46.79	156	4	**0.57**	0.57	46.79	0.30	0.01
H21	1	8	216	270	20	**0.31***	52.70	270	20	**0.31**	0.32	52.70	0.58	1.34
H22	4	3	147	226	7	**0.58***	77.00	226	7	**0.58**	0.58	77.00	0.61	0.07
H23	2	8	186	245	10	**0.31***	57.51	245	10	**0.31**	0.31	57.51	0.86	0.06
H31	3	6	204	281	9	**0.38***	75.05	281	9	**0.38**	0.38	75.05	0.75	0.12
H32	3	5	147	212	8	**0.45***	63.36	212	8	**0.45**	0.45	63.36	0.63	0.02
H33	3	6	211	281	9	**0.34***	68.23	281	9	**0.34**	0.34	68.23	0.72	0.06
H41	5	5	35	37	8	**0.08***	2.06	37	8	**0.08**	0.08	2.06	4.92	0.05
H42	7	7	40	42	13	**0.09***	2.37	42	13	**0.09**	0.09	2.37	3.94	0.25
H43	2	1	35	49	1	**0.38***	13.65	49	1	**0.38**	0.38	13.65	8.64	0.02
H51	4	3	51	54	6	**0.11***	3.00	54	6	**0.11**	0.11	3.00	10.80	0.05
H52	4	3	50	51	7	**0.09***	1.32	51	7	**0.09**	0.09	1.32	8.36	0.04
H53	2	5	52	52	7	**0.02***	0.45	52	7	**0.02**	0.02	0.45	10.83	0.03
H61	3	2	66	79	6	**0.29***	12.70	79	6	**0.29**	0.29	12.70	6.83	0.03
H62	6	2	56	72	8	**0.42***	15.65	72	8	**0.42**	0.42	15.65	11.14	0.14
H63	5	3	69	72	4	**0.06***	2.93	72	4	**0.06**	0.06	2.93	1.27	0.05
H71	1	2	91	92	6	**0.11***	1.32	92	6	**0.11**	0.11	1.32	3.75	0.04
H72	1	1	65	94	7	**0.72***	28.30	94	7	**0.72**	0.72	28.30	11.75	1.04
H73	6	6	73	83	14	**0.20***	9.91	83	14	**0.20**	0.20	9.91	21.55	0.49

*optimal çözüm

Çizelge 5.5. Tonge problem ailesine ait sonuçlar

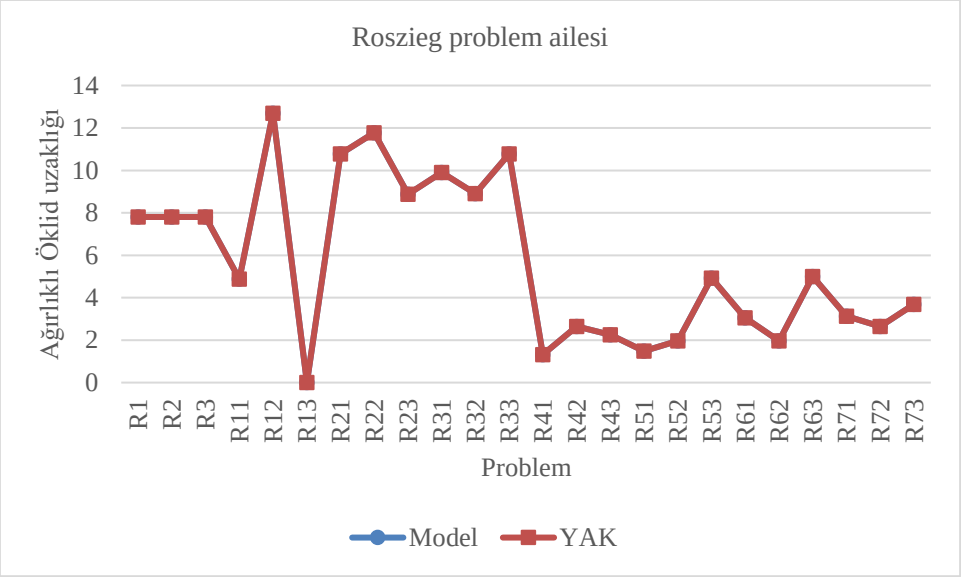
				Model				YAK					Süre	
P	F	f	C_0	C	T	z	d	C	T	z_{eniyi}	z_{ort}	d	Model	YAK
T1	2, 8	12	87	125	37	0.52	37.46	114	30	**0.37**	0.38	26.62	3600.00	9.99
T2	5, 7	19	85	111	50	0.38	26.33	111	48	**0.37**	0.44	26.21	3600.00	20.49
T3	8, 9	18	100	142	38	0.45	41.18	135	37	**0.39**	0.40	34.38	3600.00	6.68
T11	6, 10	13	112	142	15	**0.26***	29.24	142	15	**0.26**	0.26	29.24	575.77	0.07
T12	1, 2	8	108	143	39	0.50	34.81	139	36	**0.45**	0.56	30.86	3600.00	27.68
T13	2, 10	13	100	141	40	0.49	40.42	137	33	**0.43**	0.46	36.34	3600.00	18.18
T21	6, 8	10	158	237	36	0.61	77.22	214	34	**0.46**	0.50	54.85	3600.00	9.06
T22	7, 9	17	160	226	51	0.49	64.78	224	51	**0.48**	0.52	62.84	3600.00	58.87
T23	9, 10	16	144	172	51	**0.29***	28.39	172	51	**0.29**	0.36	28.39	3069.58	8.94
T31	3, 5	9	171	239	18	0.43	66.31	219	34	**0.41**	0.42	47.12	3600.00	17.92
T32	1, 5	14	180	252	29	0.43	70.26	241	27	**0.37**	0.37	59.53	3600.00	13.68
T33	1, 2	10	153	225	28	0.54	70.29	209	30	**0.45**	0.49	54.76	3600.00	12.10
T41	1, 10	8	31	49	25	0.66	17.95	36	24	**0.25**	0.34	6.05	3600.00	64.30
T41	2, 6, 15	10	31	40	20	0.33	9.05	37	23	**0.25**	0.31	6.53	3600.00	71.00
T42	12, 14	6	33	40	15	0.28	7.11	37	18	**0.22**	0.25	4.73	3600.00	33.86
T42	3, 6, 8	8	33	51	21	0.60	17.78	49	27	**0.58**	0.62	16.16	3600.00	99.63
T43	9, 12	14	34	46	32	0.40	12.37	34	27	**0.05**	0.17	2.91	3600.00	25.40
T43	5, 8, 10	18	34	70	32	1.04	35.23	42	47	**0.30**	0.40	10.14	3600.00	61.23
T51	4, 12	8	40	59	13	0.48	18.55	43	20	**0.15**	0.29	3.97	3600.00	22.23
T51	5, 9, 16	15	40	59	36	0.52	19.10	45	39	**0.20**	0.31	7.25	3600.00	45.19
T52	6, 15	8	44	64	15	0.48	19.56	49	26	**0.22**	0.34	6.32	3600.00	43.60
T52	2, 7, 10	10	44	50	24	0.20	6.63	50	22	**0.19**	0.28	6.43	3600.00	37.11
T53	4, 11	9	36	51	23	0.47	14.95	41	29	**0.24**	0.43	6.61	3600.00	123.34
T53	6, 13, 14	16	36	51	24	0.42	14.73	45	26	**0.27**	0.31	9.05	3600.00	14.17
T61	4, 5	8	71	89	26	0.35	18.00	77	28	**0.21**	0.38	7.36	3600.00	50.79
T61	2, 5, 12	12	71	104	19	0.47	32.20	77	32	**0.16**	0.31	7.36	3600.00	70.70
T62	3, 13	11	73	97	29	0.39	23.74	81	29	**0.19**	0.24	8.77	3600.00	41.73
T62	6, 13, 15	19	73	106	38	0.48	32.44	87	45	**0.25**	0.37	14.83	3600.00	62.80
T63	8, 11	9	68	100	24	0.53	31.37	73	27	**0.17**	0.34	6.32	3600.00	84.94
T63	3, 4, 8	13	68	76	18	0.13	7.88	76	17	**0.13**	0.13	7.85	3600.00	6.78
T71	2, 16	10	57	75	32	0.41	18.22	67	33	**0.28**	0.41	11.02	3600.00	51.92
T71	11, 12, 14	16	57	98	31	0.73	40.10	72	37	**0.32**	0.49	15.36	3600.00	133.60
T72	8, 12	7	65	98	18	0.56	32.26	80	15	**0.28**	0.37	14.73	3600.00	38.07
T72	5, 10, 13	13	65	99	20	0.52	33.18	76	33	**0.24**	0.35	11.62	3600.00	21.66
T73	9, 14	8	72	94	27	0.41	21.86	79	33	**0.25**	0.40	8.82	3600.00	39.78
T73	3, 7, 15	13	72	121	36	0.73	48.04	88	37	**0.30**	0.46	16.49	3600.00	65.67

*optimal çözüm

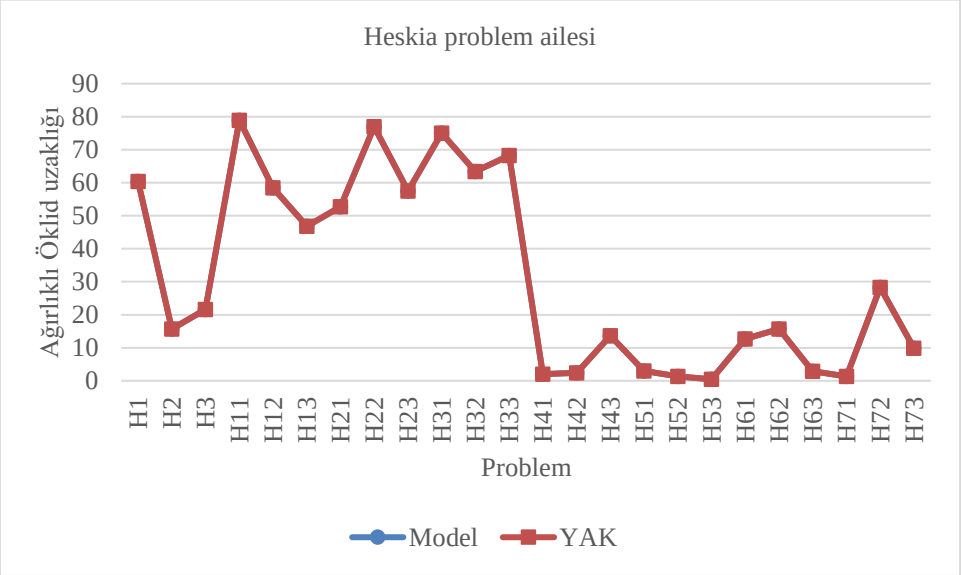
Çizelge 5.6. Wee-Mag problem ailesine ait sonuçlar

				Model				YAK					Süre	
P	F	f	C_0	C	T	z	d	C	T	z_{eniyi}	z_{ort}	d	Model	YAK
W1	5, 8	15	27	39	54	0.55	14.59	34	32	**0.30**	0.39	7.81	3600.00	14.05
W2	1, 9	13	26	38	30	0.50	12.30	31	27	**0.24**	0.33	5.79	3600.00	20.56
W3	3, 7	18	26	37	59	0.52	14.11	31	42	**0.25**	0.30	7.25	3600.00	68.00
W11	6, 7	17	30	44	45	0.53	15.01	41	31	**0.39**	0.43	11.17	3600.00	65.65
W12	2, 5	16	33	43	47	0.38	11.96	41	34	**0.29**	0.35	8.77	3600.00	75.18
W13	3, 6	16	34	43	36	0.31	9.85	41	33	**0.25**	0.26	7.81	3600.00	48.42
W21	8, 9	13	48	72	33	0.55	23.82	67	35	**0.46**	0.54	19.16	3600.00	34.88
W22	2, 11	12	44	68	48	0.67	24.74	61	35	**0.46**	0.50	17.35	3600.00	8.29
W23	6, 8	17	48	71	35	0.51	22.78	66	40	**0.42**	0.49	18.28	3600.00	30.86
W31	4, 10	10	50	70	37	0.52	20.41	67	24	**0.39**	0.44	16.86	3600.00	17.22
W32	3, 9	14	50	66	48	0.43	17.35	58	35	**0.23**	0.28	9.10	3600.00	13.67
W33	8, 11	18	51	77	44	0.56	26.00	66	35	**0.33**	0.34	15.11	3600.00	16.91
W41	5, 10	8	11	11	20	0.08	2.68	11	15	**0.04**	0.11	1.57	3600.00	47.59
W41	9, 12, 16	11	11	13	29	0.25	4.47	12	27	**0.16**	0.22	3.71	3600.00	18.20
W42	6, 8	7	10	14	19	0.47	4.73	12	18	**0.27**	0.32	3.14	3600.00	59.00
W42	9, 18, 19	16	10	22	40	1.22	12.87	13	31	**0.33**	0.44	4.45	3600.00	61.13
W43	3, 14	10	12	18	17	0.51	6.05	13	18	**0.12**	0.23	2.04	3600.00	37.18
W43	8, 16, 18	14	12	19	38	0.64	8.68	13	26	**0.12**	0.20	2.85	3600.00	31.06
W51	2, 15	7	14	33	33	1.48	19.41	15	12	**0.10**	0.19	1.48	3600.00	6.79
W51	12, 14, 15	8	14	26	23	0.91	12.17	16	29	**0.27**	0.37	5.08	3600.00	127.03
W52	9, 15	9	12	17	16	0.43	5.12	13	21	**0.15**	0.21	2.85	3600.00	29.56
W52	1, 3, 17	10	12	16	30	0.42	5.93	13	28	**0.17**	0.25	4.14	3600.00	56.27
W53	16, 17	7	13	22	33	0.84	10.52	14	41	**0.32**	0.41	7.66	3600.00	67.03
W53	7, 9, 18	12	13	21	36	0.68	9.47	14	25	**0.13**	0.23	3.07	3600.00	11.89
W61	7, 12	6	19	22	20	0.27	4.28	20	21	**0.18**	0.23	3.49	3600.00	23.77
W61	3, 16, 17	12	19	25	30	0.38	7.10	21	42	**0.23**	0.28	6.99	3600.00	140.59
W62	3, 5	10	20	30	13	0.49	9.77	21	24	**0.12**	0.18	3.28	3600.00	41.77
W62	1, 12, 15	8	20	29	26	0.54	9.65	23	33	**0.30**	0.41	6.31	3600.00	65.71
W63	16, 17	7	22	33	13	0.52	10.81	27	17	**0.29**	0.33	5.36	3600.00	31.31
W63	3, 16, 17	8	22	30	17	0.40	8.05	27	20	**0.29**	0.34	5.56	3600.00	10.95
W71	12, 13	8	21	26	22	0.31	5.79	22	22	**0.13**	0.18	3.28	3600.00	20.05
W71	6, 11, 19	10	21	24	18	0.18	3.43	22	25	**0.12**	0.21	3.49	3600.00	44.20
W72	2, 18	10	22	26	39	0.32	7.57	22	20	**0.05**	0.08	2.24	3600.00	61.78
W72	5, 10, 13	8	22	26	31	0.32	6.45	23	23	**0.14**	0.20	3.49	3600.00	45.89
W73	6, 14	11	21	24	32	0.23	5.53	21	22	**0.05**	0.16	2.46	3600.00	42.30
W73	10, 15, 19	14	21	36	31	0.74	15.11	25	30	**0.24**	0.29	5.29	3600.00	134.16

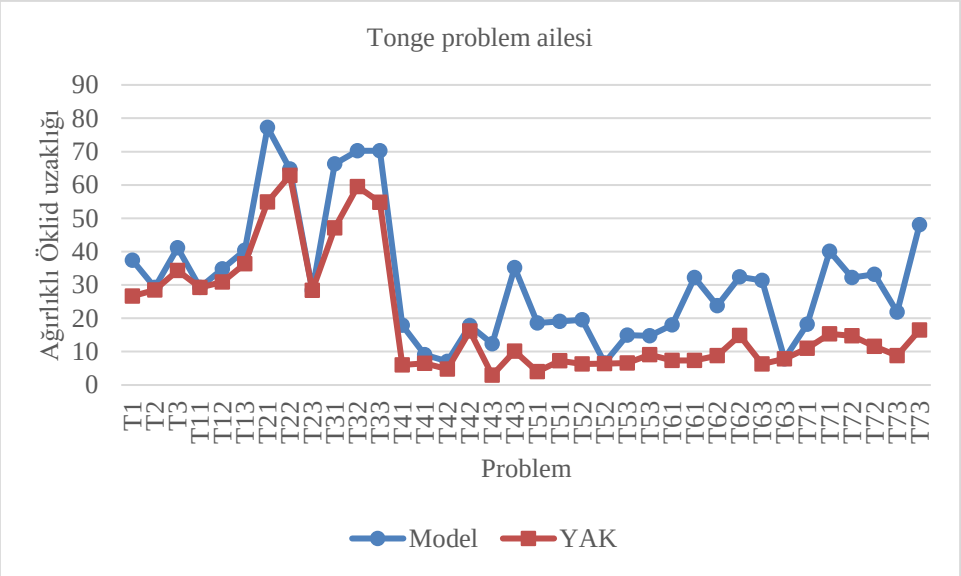
Şekil 5.2-5.5 her problem ailesi için önerilen yöntemler ile elde edilen çözümlerin alt sınırdan ağırlıklı Öklid uzaklıklarını göstermektedir. Bu değerler, Roszieg ve Heskia problem ailelerinde her iki çözüm yöntemi için eşit iken YAK algoritması Tonge ve Wee-Mag problemlerinde daha başarılı sonuçlar elde etmiştir.



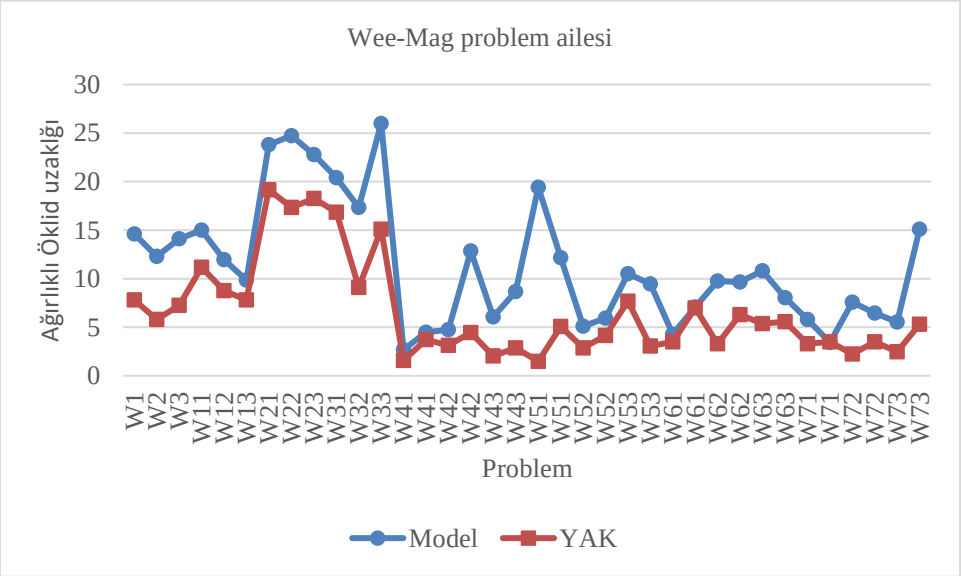
Şekil 5.2. Roszieg problem ailesi için ağırlıklı Öklid uzaklıkları



Şekil 5.3. Heskia problem ailesi için ağırlıklı Öklid uzaklıkları



Şekil 5.4. Tonge problem ailesi için ağırlıklı Öklid uzaklıkları



Şekil 5.5. Wee-Mag problem ailesi için ağırlıklı Öklid uzaklıkları

6. SONUÇ VE ÖNERİLER

İşletmelerin rekabetçi piyasada ayakta durmaları ve kar elde etmeleri için kaynakları verimli kullanmaları ve değişimlere hızlı bir şekilde cevap verebilmeleri gerekmektedir. Seri üretim sistemlerinin vazgeçilmez bileşenlerinden olan montaj hatları, üretim verimliliğinin artırılması konusunda büyük öneme sahiptir. Manuel montaj hatlarında görevleri gerçekleştiren işçiler ise bu hatların başlıca kaynaklarındandır.

Literatürde, MHYD problemi konusunda yapılan çalışmaların tümünde hatta meydana gelen değişiklik sonrasında sadece görevlerin iş istasyonlarına yeniden atanması dikkate alınmış, görev sürelerinin işlemi yapan işçiden bağımsız ve sabit olduğu varsayılarak işçi atamaları göz ardı edilmiştir. Bununla birlikte, MHD literatüründe işçi performanslarının göz önünde bulundurulduğu tek problem türü MHİAD problemidir ve bu problemde hattın ilk kez dengelenmesi ele alınmaktadır.

Bu çalışmada, başlangıç durumda dengede olan montaj hatlarının, bir veya daha fazla iş istasyonunun kapanmasına bağlı olarak işçi performansları göz önünde bulundurulacak şekilde yeniden dengelenmesi amaçlanmıştır. MHYD ve MHİAD problemlerinin bütünleşmesini sağlayan ve literatürde ilk kez tanımlanan bu problem, "Montaj Hattı İşçi Atama ve Yeniden Dengeleme Problemi" olarak isimlendirilmiştir.

NP-zor problem sınıfında bulunan bu problemin çözümü için matematiksel model ve meta-sezgisel bir yöntem olan YAK algoritması önerilmiştir. Model ve algoritmaya; görev sayısı, işçi sayısı, iş istasyonu sayısı, işlem süreleri, görevler arasındaki öncelik ilişkileri, ilk hat dengesine ait atamalar, ilk çevrim süresi değeri ve kapanan istasyon/istasyonların numarası girdi olarak verilerek çevrim süresi ve görev-iş istasyonu atamaları açısından ilk ve yeni hat dengesi arasındaki değişkenliği enküçüklemek amaçlanmıştır.

Geliştirilen iki çözüm yöntemini karşılaştırmak amacıyla literatürde bulunan test MHİAD test problemleri kullanılmıştır. Elde edilen sonuçlara göre küçük boyutlu problemlerde her iki yöntemin de optimal çözümlere ulaşabildiği görülmüştür. YAK

algoritması, en başarılı çözümlere ulaşma süresi açısından matematiksel modeli bu problemlerin çok büyük bir kısmında geride bırakmıştır.

YAK algoritması, büyük boyutlu test problemlerinde, matematiksel model ile verilen süre limiti içinde elde edilen çözümlere kıyasla daha başarılı çözümlere oldukça kısa süreler içinde ulaşmıştır.

Gelecek çalışma önerileri olarak; problemin karma modelli ve farklı yerleşim düzenine sahip hatlar için uyarlanması, işçilerin öğrenme etkisinin göz önünde bulundurulması, görev sürelerinin stokastik olduğu bir üretim ortamının ele alınması, ergonomik kısıtların eklenmesi, farklı amaç fonksiyonlarının benimsenmesi verilebilir. Ayrıca problem, farklı sezgisel ve meta-sezgisel yöntemlerle çözülerek geliştirilen yöntemler arasında kıyaslamalar yapılabilir.

KAYNAKLAR DİZİNİ

Abraham, A., Jatoth, J.V., Rajasekhar, A., 2012, Hybrid Differential Artificial Bee Colony Algorithm, Journal of Computational and Theoretical Nanoscience, 9, 2, 1-9.

Ağpak, K., 2010, An approach to find task sequence for re-balancing of assembly lines, Assembly Automation 30, 4, 378–387.

Ağpak, K., 2001, Tek modelli basit U tipi montaj hattı dengeleme problemleri için bir sezgisel metot, Yüksek lisans tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü, 96 s. (yayımlanmamış).

Ağpak, K., Gökçen, H., Saray, N.N., Özel, S., 2002, Stokastik görev zamanlı tek modelli U tipi montaj hattı dengeleme problemleri için bir sezgisel, Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, 17, 4, 115-124.

Aksoy, S., Yıldız, M.S., Altınova, S., 2014, Pozisyon ağırlığı metodu ile tek model U tipi montaj hattı dengeleme, Afyon Kocatepe Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 16, 2, 83-89.

Akyol, S., Alataş, B., 2012, Güncel sürü zekâsı optimizasyon algoritmaları, Nevşehir Üniversitesi Fen Bilimleri Enstitü Dergisi, 1, 1, 36-50.

Akyol, Ş.D., Baykasoğlu, A., 2019a, A multiple-rule based constructive randomized search algorithm, Journal of Intelligent Manufacturing, 30, 2, 557-573.

Akyol, Ş.D., Baykasoğlu, A., 2019b, ErgoALWABP: a multiple-rule based constructive randomized search algorithm for solving assembly line worker assignment and balancing problem under ergonomic risk factors, Journal of Intelligent Manufacturing, 30, 1, 291-302.

Alataş, B., 2007, Kaotik Haritalı Parçacık Sürü Optimizasyon Algoritmaları Geliştirme, Doktora Tezi, Fırat Üniversitesi Fen Bilimleri Enstitüsü, 157 s.

Alqattan, Z.N.M., Abdullah, R., 2015, A hybrid artificial bee colony algorithm for numerical function optimization, International Journal of Modern Physics C, 26, 10, 1-17.

Alp A., Çerçioğlu, H., Tokaylı M.A., Dengiz, B., 2001, Stokastik montaj hattı dengeleme: Bir tavlama benzetimi algoritması, Endüstri Mühendisliği Dergisi, 12, 3-4, 32-51.

Altıok, T., 1997, Performance Analysis of Manufacturing Systems, e-kitap, 185, https://link.springer.com/chapter/10.1007/978-1-4612-1924-8_5, erişim tarihi:13.02.2020.

Araújo, F.F.B., Costa, A.M., Miralles, C., 2012, Two extensions for the ALWABP: Parallel stations and collaborative approach, International Journal of Production Economics, 140, 1, 483-495.

KAYNAKLAR DİZİNİ (devam)

Araújo, F.F.B., Costa, A.M., Miralles, C., 2015, Balancing parallel assembly lines with disabled workers, European Journal of Industrial Engineering, 9, 3, 344–365.

Arcus, A. L., 1965, COMSOAL: A computer method of sequencing operations for assembly lines, International Journal of Production Research, 4, 4, 259-277.

Belassiria I., Mazouzi, M., ELfezazi, S., Cherrafi A., ELMaskaoui Z., 2018, An integrated model for assembly line re-balancing problem, International Journal of Production Research 56, 16, 5324–5344.

Bartholdi, J.J., 1993, Balancing two-sided assembly lines: a case study, International Journal of Production Research, 31, 10, 2447–2461.

Baybars, I., 1986, A survey of exact algorithms for the simple assembly line balancing problem, Management Science, 32, 8, 909-932.

Becker, C., Scholl, A., 2006, A survey on problems and methods in generalized assembly line balancing, European Journal of Operational Research, 168, 694–715.

Blum, C., Miralles, C., 2011, On solving the assembly line worker assignment and balancing problem via beam search, Computers & Operations Research, 38, 1, 328-339.

Borba, L., Ritt, M., 2014, A heuristic and a branch-and-bound algorithm for the assembly line worker assignment and balancing problem, Computers & Operations Research, 45, 87-96.

Bowman, E.H., 1960, Assembly-line balancing by linear programming, Operations Research, 8, 3, 385-389.

Boysen, N., Fliedner, M., Scholl, A., 2006, A classification of assembly line balancing problems, European Journal of Operational Research, 183, 2, 674-693.

Boysen, N., Fliedner, M., Scholl, A., 2008, Assembly line balancing: Which model to use when, International Journal Of Production Economics, 111, 2, 509-528.

Bryton, B., 1954, Balancing of a continuous production line, M.Sc. Thesis, Northwestern University, 132 p. (unpublished).

Castellucci, P.B., Costa, A.M., 2012, Simulation model for assembly lines with heterogeneous workers, Pesquisa Operacional, 35, 1, 57–72.

KAYNAKLAR DİZİNİ (devam)

Chaves, A.A., Lorena, L.A.N., Miralles, C., 2009, Hybrid metaheuristic for the assembly line worker assignment and balancing problem, Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 5818 LNCS, 1–14.

Chaves, A.A., Miralles, C., Lorena, L.A.N., 2007, Clustering search approach for the assembly line worker assignment and balancing problem, 37th International Conference on Computers and Industrial Engineering, 1151–1160.

Corominas, A., Pastor, R., Plans, J., 2008, Balancing assembly line with skilled and unskilled workers, Omega, 36, 6, 1126-1132.

Costa, A.M., Miralles, C., 2009, Job rotation in assembly lines employing disabled workers, International Journal of Production Economics 120, 2, 625–632.

Çavdur, F., Kaymaz, E., Sebatlı, A., 2018, Montaj hatlarında yeniden işleme istasyon pozisyonunun optimizasyonu için bir karışık tamsayılı programlama modeli, Uludağ Üniversitesi Mühendislik Fakültesi Dergisi, 23, 3, 273 – 288.

Çelik, E., Kara, Y., Atasagun, Y., 2014, A new approach for rebalancing of U-lines with stochastic task times using ant colony optimization algorithm, International Journal of Production Research, 52, 24, 7262–7275.

Çiçek, A., Gülesin, M., 2007, Montaj planlama yaklaşımları üzerine bir araştırma, Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 22, 2, 255-268.

Doğan, A., Sakallı, Ü.M., 2016, Bulanık işlem zamanlı geleneksel montaj hattı dengeleme problemi için yeni bir yaklaşım: Savunma sanayii uygulaması, Uluslararası Mühendislik Araştırma ve Geliştirme Dergisi, 8, 1, 31-50.

Efe, B., Kremer G.E.O. Kurt, M., 2018, Age and gender-based workload constraint for assembly line worker assignment and balancing problem in a textile firm, International Journal of Industrial Engineering, 25, 1, 1-17.

Erdoğmuş, P., 2016, Doğadan esinlenen optimizasyon algoritmaları ve optimizasyon algoritmalarının optimizasyonu, Düzce Üniversitesi Bilim ve Teknoloji Dergisi, 4, 1, 293-304.

Erel, E., Sarin, S., 1998, A survey of the assembly line balancing procedures, Production Planning and Control: The Management of Operations, 9, 5, 414- 434.

Erkut, H., Baskak, M., 2003, Stratejiden Uygulamaya Tesis Tasarımı, İrfan Yayımcılık, s. 353-367.

KAYNAKLAR DİZİNİ (devam)

Falkenauer, E., 2005, Line balancing in the real world, Proceedings of the International Conference on Product Lifecycle Management PLM, 5, 360–370. http://www.optimaldesign.com/Download/OptiLine/FalkenauerPLM05.pdf

Gamberini, R., Gebennini, E., Grassi, A., Regattieri, A., 2009, A multiple single-pass heuristic algorithm solving the stochastic assembly line rebalancing problem, International Journal of Production Research, 47, 8, 2141–2164.

Gamberini, R., Grassi, A., Rimini, B., 2006, A new multi-objective heuristic algorithm for solving the stochastic assembly line re-balancing problem, International Journal of Production Economics, 102, 2, 226–243.

Ghosh, S., Gagnon, R. J., 1989, A comprehensive literature review and analysis of the design, balancing and scheduling of assembly systems, International Journal of Production Research, 27, 4, 637-670.

Gökçen, H., Ağpak, K., Benzer, R., 2006, Balancing of parallel assembly lines, International Journal of Production Economics, 103, 2, 600-609.

Gökşen, Y., 2003, Geleneksel üretimden esnek üretime: Karşılaştırmalı bir inceleme, Dokuz Eylül Üniversitesi Sosyal Bilimler Enstitüsü Dergisi, 5, 4, 32-48.

Grangeon, N., Leclaire, P., Norre, S., 2011, Heuristics for the re-balancing of a vehicle assembly line, International Journal of Production Research, 49, 22, 6609–6628.

Hu, T.C., 1961, Parallel sequencing and assembly line problems, Operations Research, 9, 6, 841- 848.

Jackson, J., R., 1956, A computing procedure for a line balancing problem, Management Science, 2, 3, 261-271.

Janardhanan, M.N., Li Z., Nielsen P., Tang Q., 2018, Artificial bee colony algorithms for two-sided assembly line worker assignment and balancing problem, Advances in Intelligent Systems and Computing, 620, 11–18.

Janardhanan, M.N., Li, Z., Nielsen, P., 2019, Model and migrating birds optimization algorithm for two-sided assembly line worker assignment and balancing problem, Soft Computing, 23, 21, 11263-11276.

Kağnıcıoğlu, H., Aydın, S., Hasgül, S., Anagün, S., Üretim Yönetimi, Anadolu Üniversitesi Açıköğretim Fakültesi Yayınları, s.8-95.

Karaboğa, D., 2005, An idea based on honey bee swarm for numerical optimization, Technical Report-TR06, Erciyes University Engineering Faculty Computer Engineering Department, 10 p.

KAYNAKLAR DİZİNİ (devam)

Karaboğa D., 2011, Yapay Zekâ Optimizasyon Algoritmaları, Nobel Yayın Dağıtım, s.201-213.

Karaboğa, D., Akay, B., 2011, A modified artificial bees colony (ABC) algorithm for constrained optimization, Applied Soft Computing, 11, 3, 3021-3031.

Karaca, M.K., 1996, Montaj hatları, Süleyman Demirel Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 1, 1, 117-128.

Kılınççı, Ö., 2004, Basit montaj hattı dengeleme problemi çözümü için bir Petri ağı yaklaşımı, Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi, 6, 2, 1-15.

Kilbridge, M., Wester, L., 1961, The balance delay problem, Management Science, 8, 69-84.

Kim Y.K., Song W.S., Kim J.H., 2009, A mathematical model and a genetic algorithm for two-sided assembly line balancing, Computers & Operations Research, 36, 3, 853–865.

Kong, X., Liu, S., Wang, Z., 2013, An improved artificial bee colony algorithm and its application. International Journal of Signal Processing, Image Processing and Pattern Recognition, 6, 6, 259-274.

Kriengkorakot, N., Pianthong, N., 2007, The assembly line balancing problem: Review articles, KKU Engineering Journal, 34, 2, 133-140.

Kumar, N., Mahto, D., 2013, Assembly line balancing: A review of developments and trends in approach to ındustrial application, Global Journal of Research In Engineering, 13, 2, 29-50.

Küçükkoç, İ., 2019, Montaj Hattı Tasarımı ve Analizi Ders Notları, http://ikucukkoc.baun.edu.tr/lectures/EMM4208/EMM4208MontajHattiBirlesikNotlar.pdf, erişim tarihi: 21.08.2019.

Li, Y., 2017, The type-II assembly line rebalancing problem considering stochastic task learning, International Journal of Production Research, 55, 24, 7334-7355.

Makssoud, F., Battaïa, O., Dolgui, A., 2015, Re-balancing problem for assembly lines: new mathematical model and exact solution method, Assembly Automation 35, 1, 16–21.

Merengo, C., Nava, F., Pozetti, A., 2010, Balancing and sequencing manual mixed-model assembly lines, International Journal of Production Research, 37, 12, 2835-2860.

KAYNAKLAR DİZİNİ (devam)

Mete, S., Ağpak, K., 2013, Çok amaçlı genelleştirilmiş kaynak kısıtlı çift taraflı montaj hattı dengeleme problemi ve hesaplama analizi, Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 28, 3, 567-576.

Miltenburg, J., Wijngaard J., 1994, The U-line line balancing problem, Management Science, 40, 10, 1378-1388.

Miralles, C., Garcia-Sabater, J.P., Andres, C., Cardós, M., 2007, Advantages of assembly lines in sheltered work centres for disabled. A case study, International Journal of Production Economics, 110, 1-2, 187-197.

Miralles, C., García-Sabater, J.P., Andrés, C., Cardós, M., 2008, Branch and bound procedures for solving the assembly line worker assignment and balancing problem: Application to sheltered work centres for disabled, Discrete Applied Mathematics, 156, 3, 352-367.

Moreira, M.C.O., Costa, A.M., 2009, A minimalist yet efficient tabu search algorithm for balancing assembly lines with disabled workers, Anais Do XLI Simpósio Brasileiro de Pesquisa Operacional, 660–671.

Moreira, M.C.O., Costa, A.M., 2013, Hybrid heuristics for planning job rotation schedules in assembly lines with heterogeneous workers, International Journal of Production Economics 141, 2, 552–560.

Moreira, M.C.O., Miralles, C., Costa, A.M., 2015, Model and heuristics for the assembly line worker integration and balancing problem, Computers & Operations Research, 54, 64-73.

Moreira, M.C.O., Cordeau, J.F., Costa, A.M., Laporte, G., 2015, Robust assembly line balancing with heterogeneous workers, Computers & Industrial Engineering, 88, 254–263.

Moreira, M.C.O., Pastor, R., Costa, A.M., Miralles, C., 2017, The multi objective assembly line worker integration and balancing problem of type-2. Computers & Operations Research, 82, 114-125.

Moreira, M.C.O., Ritt, M., Costa, A.M., Chaves, A.A., 2012, Simple heuristics for the assembly line worker assignment and balancing problem, Journal of Heuristics, 18, 3, 505-524.

Mutlu, Ö., Polat, O., Supciller, A.A., 2013, An iterative genetic algorithm for the assembly line worker assignment and balancing problem of type-II, Computers & Operations Research, 40,1, 418-426.

KAYNAKLAR DİZİNİ (devam)

Oliveria, F.S., Vittori, K., Rusell, R.M.O., Travassos, X.L., 2012, Mixed assembly line re-balancing: A binary integer approach applied to real world problems in the automotive industry, International Journal of Automotive Technology, 13, 6, 933-940, doi: 10.1007/s12239−012−0094−4

Öksüz, M.K., Büyükozkan, K., Satoğlu, Ş.I., 2017, U-shaped assembly line worker assignment and balancing problem: A mathematical model and two meta-heuristics. Computers & Industrial Engineering, 112, 246–263.

Özcan, U., 2010, Balancing stochastic two-sided assembly lines: A chance-constrained, piecewise-linear, mixed integer program and a simulated annealing algorithm, European Journal of Operational Research, 205, 1, 81-97.

Özcan U, Gökçen H, Toklu B (2010) Balancing parallel two-sided assembly lines. International Journal of Production Research 48, 16, 4767–4784.

Özgörmüş, E., 2007, Ergonomik koşullar altında montaj hattı dengeleme, Yüksek lisans tezi, Pamukkale Üniversitesi Fen Bilimleri Enstitüsü, 87 s.

Pearce, B., 2015, A study on general assembly lıne balancıng modelıng methods and technıques, Ph.D. thesis, Clemson University, 239 p.

Pereira, J., 2018, The robust (minmax regret) assembly line worker assignment and balancing problem, Computers & Operations Research, 93, 27-40.

Polat, O., Kalaycı, C.B., Mutlu, Ö., Gupta, S.M., 2016, A two-phase variable neighbourhood search algorithm for assembly line worker assignment and balancing problem type-II: An industrial case study, International Journal of Production Research, 54, 3, 722-741.

Purnomo, H.D., Wee, H.M., Rau, H., 2013, Two-sided assembly lines balancing with assignment restrictions, Mathematical and Computer Modelling, 57, 1-2, 189-199.

Ramezanian, R., Ezzatpanah, A., 2015, Modeling and solving multi-objective mixed-model assembly line balancing and worker assignment problem. Computers & Industrial Engineering, 87, 74-80.

Ritt, M., Costa, A.M., Miralles, C., 2016, The assembly line worker assignment and balancing problem with stochastic worker availability, International Journal of Production Research, 54, 3, 907-922.

Saif, U., Guan, Z., Jahanzaib, M., 2014, Survey of assembly lines and its types, Frontiers of Mechanical Engineering, 9, 2, 95-105.

KAYNAKLAR DİZİNİ (devam)

Salveson, M.E., 1955, The assembly line balancing problem, Journal of Industrial Engineering, 6, 3, 18-25.

Samadhi, T.M.M.A., Sumihartati, A., 2016, Model for assembly line re-balancing considering additional capacity and outsourcing to face demand fluctuations, IOP Conference Series: Materials Science and Engineering, 114, 1, 1-10.

Sancı, E., Azizoğlu, M., 2017, Rebalancing the assembly lines: Exact solution approaches, International Journal of Production Research 55, 20, 5991–6010.

Scholl, A., Becker, C., 2006, state-of-the-art exact and heuristic solution procedures for simple assembly line balancing, European Journal of Operational Research, 168, 3, 666-693.

Serin, F., Mete, S., Çelik, E., 2019, An efficient algorithm for U-type assembly line re-balancing problem with stochastic task times, Assembly Automation, 39, 4, 581-595, doi:10.1108/AA-07-2018-106

Sivasankaran P., Shahabudeen, P., 2014, Literature review of assembly line balancing problems, International Journal Of Advanced Manufacturing Technology, 9-12, 1665-1694.

Soba, M., 2008, Esnek üretim sistemleri ve işletmelerin rekabet gücüne etkileri, Uşak Üniversitesi Sosyal Bilimler Dergisi, 1, 2, 103-123.

Sungur, B., Yavuz, Y., 2015, Assembly line balancing with hierarchical worker assignment, Journal of Manufacturing Systems, 37, 290-298.

Supnick, F., Solinger, J., 1960, An extremal production-line problem, Operations Research, 8, 3, 381-384.

Szeto, W.Y., Wu, Y., Ho, S.C., 2011, An artificial bee colony algorithm for the capacitated vehicle routing problem, European Journal Of Operational Research, 215, 1, 126–135.

Tanrıtanır, E., 1990, Üretim sistemleri ve imalat sistemleri, İstanbul Üniversitesi Orman Fakültesi Dergisi, 40, 1, 127-144.

Tempelmeier, H., 2003, Practical considerations in the optimization of flow production systems, International Journal of Production Research 41, 1, 149–170.

Vilà, M., Pereira, J., 2014, A branch-and-bound algorithm for assembly line worker assignment and balancing problems, Computers & Operations Research, 44, 105-114.

White, W.W., 1961, Comments on a paper by Bowman. Operations Research, 9, 2, 274-276.

KAYNAKLAR DİZİNİ (devam)

Yağmahan, B, Emel, E., 2015, Yasak arama algoritmasına dayalı pürüzsüzleştirme yaklaşımı içeren karışık modelli montaj hattı dengeleme, Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, 30, 1, 57-69.

Yang, C., Gao, J., Sun, L., 2013, A multi-objective genetic algorithm for mixed-model assembly line rebalancing, Computers and Industrial Engineering 65, 1, 109–116.

Yılmaz, E., Erol, R., 2005, Montaj hatlarının değişen koşullar altında yeniden dengelenmesi, Çukurova Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 20, 1, 213-227.

Zacharia, P.T., Nearchou, A.C., 2016, A population-based algorithm for the biobjective assembly line worker assignment and balancing problem, Engineering Applications of Artificial Intelligence, 49, 1-9.

Zha, J., Yu, J.J. 2014., A hybrid ant colony algorithm for U-line balancing and rebalancing in just-in-time production environment, Journal of Manufacturing Systems 33, 1, 93–102.

Zhang Y., Hu X., Wu, C., 2018a, A modified multi-objective genetic algorithm for two-sided assembly line re-balancing problem of a shovel loader, International Journal of Production Research, 56, 9, 3043-3063.

Zhang, Y., Hu, X., Wu, C., 2018b, Heuristic algorithm for type II two-sided assembly line rebalancing problem with multi-objective, MATEC Web of Conferences, 175.

Zhang Y., Hu X., Wu, C., 2018c, Push rule-based heuristic method for multi-objective two-sided assembly line rebalancing problem, International Conference on Physics, Mathematics, Statistics Modelling and Simulation.

Zhang Y., Hu X., Wu, C., 2019a, Improved imperialist competitive algorithms for rebalancing multi-objective two-sided assembly lines with space and resource constraints, International Journal of Production Research, doi: 10.1080/00207543.2019.1633023 (in press).

Zhang, Z., Tang, Q., Han, D., Li, Z. 2019b, Enhanced migrating birds optimization algorithm for U-shaped assembly line balancing problems with workers assignment, Neural Computing & Applications, 31, 11, 7501-7515.