

**KARIŐIK MODELLİ ÇİFT TARAFLI MONTAJ HATTI DENGEME
PROBLEMİ İÇİN ARI KOLONİSİ ALGORİTASI**

İsmail TÜRKDÖNMEZ

**YÜKSEK LİSANS TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

KASIM 2013

ANKARA

İsmail TÜRKDÖNMEZ tarafından hazırlanan “KARIŞIK MODELLİ ÇİFT TARAFLI MONTAJ HATTI Dengeleme Problemi İçin ARI Kolonisi Algoritması” adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Prof. Dr. Bilal TOKLU

Tez Danışmanı, Endüstri Mühendisliği Anabilim Dalı



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

Prof. Dr. Cevriye GENCER

Endüstri Mühendisliği Anabilim Dalı, G.Ü.



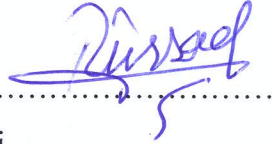
Prof. Dr. Bilal TOKLU

Endüstri Mühendisliği Anabilim Dalı, G.Ü.



Yrd. Doç. Dr. A. Kürşad TÜRKER

Endüstri Mühendisliği Anabilim Dalı, Kırıkkale Üniversitesi

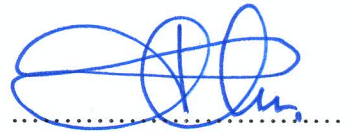


Tez Savunma Tarihi: 01/11/2013

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

Prof.Dr. Şeref SAĞIROĞLU

Fen Bilimleri Enstitüsü Müdürü Y.



TEZ BİLDİRİMİ

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

İsmail TRKDNMEZ

KARIŞIK MODELLİ ÇİFT TARAFLI MONTAJ HATTI Dengeleme PROBLEMİ İÇİN ARI KOLONİSİ ALGORİTMASI

(Yüksek Lisans Tezi)

İsmail TÜRKDÖNMEZ

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

Kasım 2013

ÖZET

Montaj hatlarını dengeleme, montaj hatlarının dizayn edilmesi ve yönetilmesi gibi diğer montaj hattı problemleri arasında en önemli problemlerdendir. Çift taraflı montaj hatları genellikle otobüs ve kamyonlar gibi büyük boyutlu üretimlerin montajında ortaya çıkar. Diğer taraftan, gerçek bal arılarının doğal yiyecek arama davranışlarını modellemeye çalışan Arı Algoritmaları sezgisel tabanlı sürü zekâsının nispeten yeni bir üyesidir. Bu çalışma çift taraflı karışık modellenli montaj hattı dengeleme problemini çözmek için bir yaklaşım sunmaktadır. İlk olarak, bir matematiksel programlama modeli sorunu tanımlamak için sunulmuştur. Daha sonra, bir arı kolonisi optimizasyon algoritması sorunu çözmek için önerilmektedir. Önerilen algoritmada, iki performans kriteri aynı anda dikkate alınır: ağırlıklı hat verimliliğini en büyükleme ve ağırlıklı düzgünlük endeksi en aza indirme. Önerilen yaklaşım örnek bir problem ile gösterilmiş ve performansı bir dizi test problemleri ile test edilmiştir.

Bilim Kodu : 906.1.148
Anahtar Kelimeler : Çift taraflı montaj hattı dengeleme, Karışık modellenli üretim, Arı kolonisi algoritması
Sayfa Adedi : 84
Tez Yöneticisi : Prof. Dr. Bilal TOKLU

**BEE COLONY ALGORITHM FOR MIXED-MODEL TWO SIDED
ASSEMBLY LINE BALANCING PROBLEM**

(M.Sc. Thesis)

İsmail TÜRKDÖNMEZ

GAZI UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

November 2013

ABSTRACT

Balancing of assembly lines is one of the most important problems among the other problems of assembly lines like designing and managing. Two-sided assembly lines are generally occurred in assembly of large-sized products such as buses and trucks. On the other hand, Bee Algorithms is a relatively new member of swarm intelligence based meta-heuristics which tries to model natural behavior of real honey bees in food foraging. This study presents an approach to address the two-sided mixed-model assembly line balancing problem. First, a mathematical programming model is presented to formally describe the problem. Then, a bee colony optimisation algorithm is proposed to solve the problem. In the proposed algorithm, two performance criteria are considered simultaneously: maximizing the weighted line efficiency and minimizing the weighted smoothness index. The proposed approach is illustrated with an example problem, and its performance is tested on a set of test problems.

Science Code	: 906.1.148
Key Words	: Two sided assembly line , Mixed model production, Bee colony algorithm
Page Number	: 84
Supervisor	: Prof. Dr. Bilal TOKLU

TEŞEKKÜR

Tez çalışmam süresince beni her konuda destekleyen ve yüreklendiren tez danışmanım, hocam Sayın Prof. Dr. Bilal TOKLU' ya, tezimin uygulama aşamasında bana yol gösteren, yoğunluğuna rağmen bana zaman ayırıp destek olan Sayın Dr. Hakan ÇERÇİOĞLU' na, bugünlere gelmemde sonsuz emekleri geçen ve sevgilerini hiçbir zaman esirgemeyen sevgili anneme ve babama, Ankara dışında olmam dolayısıyla ihtiyaç duyduğum her türlü işi yorulmadan bıkmadan yapan, işlerimin peşinde azimle koşan sevgili kardeşim Hatice TÜRKDÖNMEZ'e, yoğun iş temposuna rağmen kısıcık tatil günlerini, değerli zamanını gece gündüz demeden bana ayıran, çalışmama destek veren Murat ÖZYÜREK'e, ve son olarak bana en zor zamanlarımda destek olan, bu uzun tez sürecinde bana katlanan, sabrıyla hayran olduğum, sevgili eşim, hayat arkadaşım Dr. Yasemin ÖZYÜREK TÜRKDÖNMEZ'e teşekkürlerimle. Bu uzun soluklu çalışmayı yakın zamanda hayatımıza katılacak canım kızım ve aileme armağan ediyorum.

İÇİNDEKİLER

Sayfa

ÖZET	iv
ABSTRACT.....	v
TEŞEKKÜR	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	ix
ŞEKİLLERİN LİSTESİ.....	x
RESİMLERİN LİSTESİ.....	xi
SİMGELER VE KISALTMALAR.....	xii
1. GİRİŞ	1
2. MONTAJ HATTI KAVRAMI	5
2.1. Montaj Hatları	6
2.1.1. Geleneksel düz montaj hattı.....	6
2.1.2. U-tipi montaj hattı	7
2.1.3. Paralel montaj hatları	9
2.2. Çift Taraflı Montaj Hatları.....	11
2.2.1. Temel kavramlar	11
2.2.2. Çift taraflı montaj hattı nedir?.....	12
2.2.3. Çift taraflı montaj hattı dengeleme problemi ve çözüm yaklaşımları... 14	
3. LİTERATÜR ARAŞTIRMASI	20
4. KARIŞIK MODELLİ ÇİFT TARAFLI MONTAJ HATTI DENGELEME	27
5. ARI KOLONİSİ OPTİMİZASYON ALGORİTMASI	33
5.1. Arı Kolonisi Algoritmasına Giriş.....	33

Sayfa

5.2. Arı Kolonisi Algoritması Uygulamaları.....	39
5.3. Önerilen Arı Kolonisi Algoritması	46
5.4. Sayısal Örnek	60
5.5. Deneysel Çalışma	65
6. SONUÇ	72
KAYNAKLAR	73
EKLER.....	79
EK-1. Küçük boyutlu problemlerde tüm modeller için görev tamamlanma zamanları	80
EK-2. Problem-65’de tüm modeller için görev tamamlanma zamanları	81
EK-3. Problem 148’de tüm modeller için görev tamamlanma zamanları	82
ÖZGEÇMİŞ	84

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 3.1. Çift taraflı montaj hattı dengeleme üzerine yapılan çalışmalar	24
Çizelge 5.1. Sayısal örnek için görev işlem zamanları	61
Çizelge 5.2. Sayısal örnek için görev öncelikleri ve tarafları	62
Çizelge 5.3. Başlangıç hat dengesi.....	62
Çizelge 5.4. KÇMHD için AKA parametre değerleri.....	66
Çizelge 5.5. Küçük boyutlu test problemlerinin karşılaştırmalı sonucu	68
Çizelge 5.6. Büyük boyutlu test problemleri SA Algoritması [9] sonuçları.....	69
Çizelge 5.7. Büyük boyutlu test problemlerinin başlangıç hat denge sonuçları	69
Çizelge 5.8. Büyük boyutlu test problemlerinin önerilen AKA sonuçları.....	70

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Geleneksel düz montaj hattı	6
Şekil 2.2. U-tipi montaj hattı.....	8
Şekil 2.3. (a) Karışık modellenli geleneksel düz montaj hattı ve (b) karışık modellenli U-tipi montaj hattı	9
Şekil 2.4. Paralel montaj hatları	10
Şekil 2.5. Çift taraflı montaj hattı	13
Şekil 2.6. Öncelik şeması	15
Şekil 2.7. 5 görevli örnek problemin öncelik ilişkileri şeması, görev zamanları ve görevlerin operasyon tarafları	16
Şekil 2.8. Çevrim zamanı 5 olan örnek problem için iki çözüm.....	17
Şekil 2.9. Öncelik diyagramı.....	17
Şekil 2.10. Çevrim zamanı 20 birim için istasyonlara görev atamaları (a) tek taraflı montaj hattı (b) çift taraflı montaj hattı.....	18
Şekil 5.1. Arıların besin arama davranışı	37
Şekil 5.2. Arıların üçüncü ileri geçişi.....	40
Şekil 5.3. Bağımsız arılara görev verme	41
Şekil 5.4. Arıların dördüncü ileri geçişinden sonraki kısmi çözümleri.....	42
Şekil 5.5. Problemin çözüm adımları	46
Şekil 5.6. Arı Algoritması akış diyagramı	49
Şekil 5.7. Çözüm dizisi gösterimi	53
Şekil 5.8. Başlangıç çözümlerinin oluşturulması.....	54
Şekil 5.9. Sayısal örnek için öncelik diyagramı.....	61
Şekil 5.10. Başlangıç hat dengesi.....	65

RESİMLERİN LİSTESİ

Resim	Sayfa
Resim 5.1. Program boş başlangıç sayfası	67
Resim 5.2. Problem-1'in çevrim zamanı 4 için sonuç sayfası	67

SİMGELER VE KISALTMALAR

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

Simgeler	Açıklama
i, h, p, r	Görev
j, g	Çift istasyon
m	Ürün modeli
k, f	Hattın operasyon tarafı; $k, f = \begin{cases} 1\text{-sol taraf} \\ 2\text{-sağ taraf} \end{cases}$
(j, k)	Çift istasyon j ve onun operasyon tarafı k
I	Görev kümesi; $I = \{1, 2, \dots, i, \dots, nt\}$
J	Çift istasyon kümesi; $J = \{1, 2, \dots, j, \dots, nms\}$
M	Ürün modeli kümesi; $M = \{1, 2, \dots, m, \dots, npm\}$
A_L	Hattın sol tarafından yapılan görevler kümesi; $A_L \subset I$
A_R	Hattın sağ tarafından yapılan görevler kümesi; $A_R \subset I$
A_E	Hattın her iki tarafından yapılan görevler kümesi; $A_E \subset I$
$P(i)$	Görev i 'nin hemen öncülleri kümesi
$P_a(i)$	Görev i 'nin bütün öncülleri kümesi
$S(i)$	Görev i 'nin hemen ardılları kümesi

$Sa(i)$	Görev i 'nin bütün ardılları kümesi
P_0	Hiç öncülü olmayan görevler kümesi; $P_0 = \{i \in I \mid P(i) \emptyset\}$
t_{im}	Görev i 'nin, m modeli için operasyon zamanı
ψ	Çok büyük pozitif bir sayı
$C(i)$	Görev i 'nin operasyon tarafı karşısında yer alan görevler kümesi; $C(i) = \begin{cases} A_L & i \in A_R \\ A_R & \text{if } i \in A_L \\ \emptyset & i \in A_E \end{cases}$
$K(i)$	Görev i 'nin tercih edilen operasyon tarafı kümesi; $K(i) = \begin{cases} \{1\} & i \in A_R \\ \{2\} & \text{if } i \in A_L \\ \{1, 2\} & i \in A_E \end{cases}$
C	Çevrim zamanı
ε	Çok küçük pozitif bir sayı, $0 < \varepsilon \leq 1/(2 * nms+1)$
W_{jkm}	m modeli için istasyon (j,k) 'ya atanabilecek görevler kümesi
$\ W_{jkm}\ $	W_{jkm} 'deki görev sayısı
$\ M\ $	Model sayısı

CZ	Pozitif bölgeleme için görev çiftleri kümesi; $CZ = \{(i, h), \dots, (p, r)\}$
IZ	Negatif bölgeleme için görev çiftleri kümesi; $IZ = \{(i, h), \dots, (p, r)\}$
PZ	Görev çiftleri ve bölgesel kısıt için önceden belirlenmiş istasyonlar kümesi; $PZ = \{(r, (g, f)), \dots, (i, (j, k))\}$
SZ	Senkronizm kısıtı için senkronize görevler kümesi; $SZ = \{(i, h), \dots, (p, r)\}$
x_{ijk}	1, görev i istasyon (j, k) 'ya atanmış ise; 0, aksi halde
t_{im}^f	m modeli için görev i 'nin bitiş zamanı
F_j	1, eş istasyon j 'nin her iki tarafı kullanılıyor ise; 0, aksi halde
G_j	1, eş istasyon j 'nin tek tarafı kullanılıyor ise; 0, aksi halde
U_{jk}	1, istasyon (j, k) açılmış ise; 0, aksi halde
v_{jkm}	1, istasyon (j, k) m modeli için açılmış ise; 0, aksi halde
z_{ip}	1, görev i aynı istasyonda görev p 'den önce yapılıyorsa; 0, aksi halde

Kısaltmalar	Açıklama
AA	Arı algoritması
AKA	Arı kolonisi algoritması
ÇMHD	Çift taraflı montaj hattı dengeleme
KÇMH	Karışık modellenli çift taraflı montaj hattı
KÇMHD	Karışık modellenli çift taraflı montaj hattı dengeleme
MHD	Montaj hattı dengeleme
SA	Tavlama benzetimi
TALBP	Two sided assembly lines balancing problem
TZÜ	Tam Zamanında Üretim

1. GİRİŞ

Fayda üretmedeki asıl amaç, hedeflenen üretimi gerçekleştirirken verimlilik düzeyini yükseltmek, kaliteyi artırmak, kayıpları en aza indirmek (hatta kayıpsız çalışmak), maliyetleri düşürmek ve çalışma ortamını insana en uygun hale getirmektir. Bu hedeflere ulaşmak için ise organizasyonda kullanılan işgücü, makine, malzeme ve ekipmandan oluşan sistemin yeniden tasarlanması gerekmektedir. Çağımızın gereği olarak ortaya çıkan en kısa sürede, en kaliteli ve en uygun maliyetli üretimi gerçekleştirmek için oluşturulan sürekli üretim sistemlerinde, yüksek üretim hızıyla talebi karşılamanın en uygun yolu montaj hatlarının yapılandırılmasıdır.

Montaj hatları ilk olarak 1915 yılında Henry Ford'un, Taylor'un bilimsel yönetim ilkelerini hayata geçirerek Ford Model T arabalarının üretiminde kullanmasıyla ortaya çıkmıştır. Ford, otomobil imalinde uygulanması gereken bütün görevleri işçilere bölüştürerek, her işçinin devamlı aynı operasyonu yapmasını sağlamıştır. Bu ise işçilerin yaptıkları işlerde uzmanlaşmasını getirmiştir. Sonuç olarak çok sayıda aynı tip otomobili çok daha düşük maliyetlerde ve çok daha düşük sürelerde üretmek mümkün olmuştur. Bu imalat yapısı daha sonra seri imalat olarak isimlendirilmiştir.

Montaj hatlarının uygulanmak istenmesindeki esas amaçlar, her iş istasyonuna eşit miktarda iş dağıtımını yapmak, en az sayıda istasyon kullanmak, belirli istasyon için çevrim zamanını en küçükmek ve hattın etkinliğini en büyükmektir. Bu amaç doğrultusunda iş öğeleri, istasyon süreleri birbirine eşit ya da çok yakın olacak şekilde veya çevrim zamanını en küçükleyecek şekilde istasyonlar arasında paylaştırılır. Montaj hattı dengeleme ile görevler gruplandırılarak istasyonlar kurulur, bu istasyonların işlem süreleri birbirine yakın hale getirilir ve bu şartlar altında montaj hattının aksamadan çalışması sağlanarak kaynaklardan en büyük fayda elde edilir. Montaj hatlarında dengenin sağlanamadığı bir durumda bazı istasyonlarda diğerlerinden daha fazla iş yükü olacağı için, verimlilikte haliyle düşüşlerin olması ve bazı kayıpların ortaya çıkması kaçınılmaz bir durumdur.

Seri imalatın ana yapısını oluşturan geleneksel düz montaj hatları zamanla rekabet faktörleri, müşteri beklentileri ve teknolojik ilerlemelere bağlı olarak tasarım açısından değişik biçimlerde kullanılmıştır. Bunlar arasında en fazla öne çıkan şüphesiz ki, Toyota üretim sisteminin başarıya ulaşmasında önemli bir yeri olan U-tipi montaj hatlarıdır. Diğerleri ise; benzer üretim karakteristikleri gösteren birden fazla ürün modelinin aynı montaj hattında üretilbildiği karışık modelli montaj hatları, birbirine paralel birden fazla montaj hattının birlikte kullanılmasına olanak sağlayan paralel montaj hatları ve nispeten daha büyük boyutlara sahip olan otomobil, otobüs, kamyon gibi ürünlerin montaj işlemlerinin montaj hattının her iki tarafından yapılabilmesine imkân tanıyan çift taraflı montaj hatlarıdır.

Günümüz rekabet koşullarında kaynakların en uygun şekilde kullanılması çok büyük önem kazanmıştır. Bu da üretim faaliyeti içerisinde önemli bir yeri olan montaj hatlarına dikkat çekmiştir. Montaj hattının etkin kullanımı, bu hattın dikkate alınan amaçlar ve kısıtlar doğrultusunda en iyi şekilde dengelenmesi ile mümkün olabilmektedir.

Montaj hatları akış tipi üretim sistemlerindendir. Tesis içi yerleşim operasyonların teknolojik kısıtları dikkate alınarak yapılır.

Üretim her ne kadar çoğunluk olarak otomatize edilmiş olsa da planlama da dahil bazı konularda insan müdahalesi olmadan çalışması olanaksızdır. Bu da yanılma faktörünü beraberinde getirmektedir. Üretim esnasında montaj hatlarında birtakım aksaklıkların çıkması olasıdır. İş öğelerinin işlem süreleri arasındaki farklar nedeniyle montaj hatlarındaki iş istasyonlarında boş beklemeler, gecikmeler ve ara depolarda yığılmalar meydana gelmektedir. Bu sorunların giderilmesi için yapılan çalışmalara “Montaj Hatlarının Dengelenmesi” denilmektedir. Bu aksaklıkları gidermek adına doğadan da ilham alınarak pek çok uygulama gerçekleştirilmeye çalışılmıştır.

Montaj hattı dengeleme (MHD) problemi ilk olarak Helgeson (1954) tarafından çalışılmıştır ve Salveson (1955) tarafından matematiksel olarak formüle edilmiştir [15, 16].

Bu temel çalışmalardan sonra günümüze kadar MHD problemi ile ilgili çok sayıda matematiksel modeli, alt ve üst sınır hesaplamalarını, çözüm prosedürlerini, değişik amaç fonksiyonlarını, sezgisel ve meta-sezgisel algoritmaları içeren çok sayıda çalışma yapılmıştır. Literatürde MHD probleminde genel olarak dört problem tipi üzerinde durulmuştur:

Tip 1: Belirli çevrim zamanı için istasyon sayısı en küçükleme, *Tip 2:* Belirli istasyon sayısı için çevrim zamanı en küçükleme, *Tip E:* Hattın etkinliğinin en büyükleme ile istasyon sayısı ve çevrim zamanının eş zamanlı en küçükleme, *Tip F:* Belirli istasyon sayısı ve çevrim zamanı için makul bir çözümün bulunması [8].

MHD problemi kombinatoriyal optimizasyon problemlerinin NP-Zor sınıfına girmektedir. MHD probleminin kombinatoriyal yapısı problemin boyutu büyüdükçe en iyi sonuçların elde edilmesini zorlaştırmaktadır. Görevler arasındaki öncelik ilişkileri dikkate alınmadığında N görev için $N!$ adet farklı sıra elde edilmektedir. Eğer N görev arasında R adet öncelik ilişkisi dikkate alınırsa $N!/2^R$ adet farklı sıra bulunur. Sonuç olarak problemin boyutu arttıkça elde edilen çözüm alternatifleri de artacağından makul zamanda en iyi çözümlere ulaşmak mümkün olamamaktadır [1].

Bu tezin amacı; yukarıda verilen bilgiler doğrultusunda literatürde bahsi geçen bir problemi farklı bir çözüm yöntemi kullanarak çözümlmek ve mevcut çözümle karşılaştırmaktır. Bu nedenle, bu tezde, Özcan ve Toklu'nun (2009a) yapmış oldukları çalışmadaki ÇMHD problemi için çift istasyon sayısını (montaj hattının uzunluğunu) ve istasyon sayısını (operatör sayısını) en küçüklemeyi hedefleyerek arı kolonisi algoritmasıyla yeni bir çözüm geliştirmek ve sonuçlarını mevcut problemle karşılaştırmak amaçlanmıştır [9].

Tezin devamı şu şekilde planlanmıştır. İkinci bölümde montaj hattı kavramı ayrıntılı bir şekilde anlatılmış olup, en sık kullanılan montaj hattı tasarımları ele alınmıştır. Ardından çift taraflı montaj hatlarının farklılıkları ayrıntısıyla belirtilmiştir. Bununla birlikte çift taraflı montaj hatları ile ilgili temel kavram ve tanımlamaları verilmiştir. Daha sonra çift taraflı montaj hatları detaylı olarak ele alınmış ve örnek bir ÇMHD problemi tanımlanmıştır.

Üçüncü bölümde, Montaj hattı ve ÇMHD ile ilgili kapsamlı bir literatür araştırması verilmiştir.

Dördüncü bölümde karışık modellenmiş çift taraflı montaj hattı dengeleme (KÇMHD) ayrıntısıyla anlatılmış olup, Kim ve ark.'nın [11] geliştirdikleri model üzerine kurulmuş olan ve KÇMHD için Özcan ve Toklu'nun [9] geliştirdikleri matematiksel model anlatılmıştır.

Beşinci bölümde arı kolonisi optimizasyon algoritması tüm ayrıntılarıyla anlatılmış, kavramlar verilmiştir. Ayrıca bu bölümde problemi çözmede kullanılacak algoritma ayrıntısıyla anlatılmıştır.

Sonuç bölümünde ise oluşturulan algoritmaya dayalı kodlama sonucunda elde edilen netice ilgili çalışma ile karşılaştırılmış ve algoritmanın etkinliği tespit edilmeye çalışılmıştır.

2. MONTAJ HATTI KAVRAMI

Bir mamulü oluşturan parçaların önceden belirlenmiş bir sıraya uygun olarak birleştirilmesi işlemi montaj olarak adlandırılır ve montaj işlemlerinin tamamlanması ile ürün tam bir biçimde ortaya çıkar.

Montaj işlemlerinin üzerinde yapıldığı üretim hattına “montaj hattı” denir. Montaj hatlarının gelişmesi ve büyümesiyle montaj hattı dengeleme (Assembly Line Balancing) problemi ortaya çıkmıştır.

Montaj hatları, bir mamulü meydana getiren parçaların aralarında öncelik ilişkileri gözetilerek, bir hat boyunca sıralanan iş istasyonlarında birleştirildiği üretim sistemleridir.

Montaj hatlarının en belirgin özelliği malzemelerin üretim hattı boyunca iş gücünden de yararlanarak aktarılmasıdır. Montaj hattı, her birinde bir veya birden fazla işlemin yapıldığı iş istasyonları ve bunları birbirine bağlayan banttandır [2].

Montaj hatları günümüzde pek çok üretim dalında kullanılan üretim teknolojilerindendir. Günlük yaşantımızın her alanına girmiş pek çok sayıda ürün, (otomobiller, bilgisayarlar, bulaşık makinesi, televizyon vb.) montaj hatları kullanılarak üretilmektedir.

Montaj hatları ilk tasarlandığı 1915 yılından bu yana müşteri istek ve gereksinimlerindeki değişimler, teknolojik ilerlemeler ve işletmelerin etkili rekabet avantajı sağlama çabaları sonucunda değişik biçimlerde kullanılmıştır. Her bir montaj hattı tasarımının kendine özgü yapısı, diğerlerinden üstün ve zayıf yönleri bulunmaktadır.

İşletmeler amaçladıkları hedefler ölçüsünde kendi stratejilerine en uygun montaj hattını teknolojik altyapılarına bağlı olarak tasarlayabilirler.

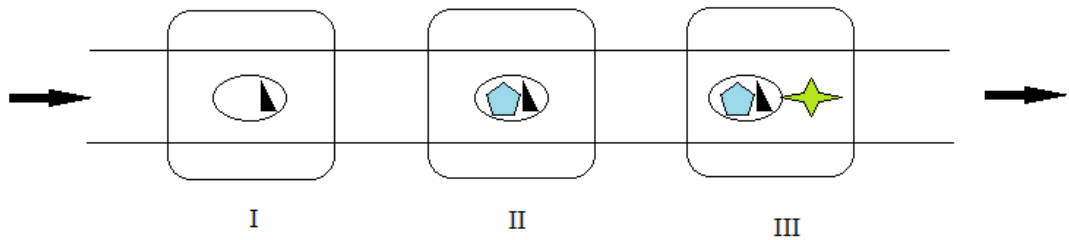
Bu bölümde, en sık kullanılan montaj hattı tasarımları ele alınmış olup, tasarım ve kullanım açısından çift taraflı montaj hatlarından farklılıkları belirtilmiştir. Ayrıca çift taraflı montaj hatları ile ilgili temel kavram ve tanımlamalar verildikten sonra çift taraflı montaj hatları detaylı olarak ele alınmıştır ve çift taraflı montaj hattı dengeleme (ÇMHD) problemi tanımlanmıştır.

2.1.Montaj Hatları

2.1.1. Geleneksel düz montaj hattı

Yukarıda da belirttiğimiz üzere montaj hatları işletmelerin rekabet ortamında daha güçlü bir üretim akışıyla ayakta kalması adına; ürün çeşidi, sahip olduğu teknolojik imkan ve stratejik hedeflerine bağlı olarak oluşturduğu bir üretim teknolojisidir. Montaj hattı tasarımındaki ana amaçlardan biri, her iş istasyonuna eşit miktarda iş dağıtımını yapabilmektir. Bu amaç doğrultusunda işler, istasyon süreleri birbirine eşit ya da çok yakın olacak şekilde istasyonlar arasında paylaştırılır.

Montaj hatları akış tipi üretim sistemlerindendir. Birbirlerine malzeme taşıma sistemleri ile bağlanmış sıralı istasyonlardan oluşur. İş parçaları bir istasyondan diğerine hareket ederek belirli montaj işlemleri görürler. Son istasyonda ise bitmiş ürün elde edilir. Şekil 2.1 bir montaj hattını göstermektedir.



Şekil 2.1. Geleneksel düz montaj hattı

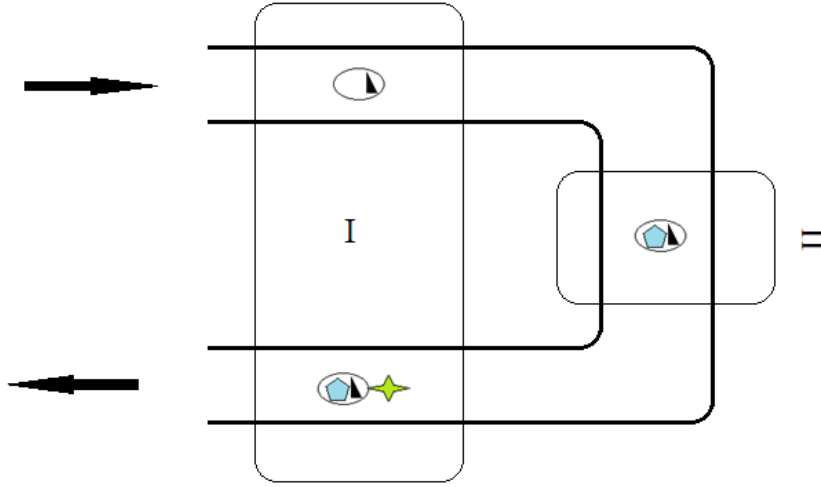
2.1.2. U-tipi montaj hattı

Geleneksel düz montaj hattından U-tipi montaj hattından teşkil olan bir sisteme dönüşüm kararı büyük bir yerleşim planı değişimi ve montaj operasyonları için yatırıma gerek duyar.

Yalın üretim ve tam zamanlı üretim felsefesi savunucuları işgücü verimliliğinin geliştirilmesi adına U-tipi montaj hattının geleneksel düz montaj hattına göre pek çok faydasının olduğunu savunmaktadırlar. Bu önerme firmaların neden geleneksel düz montaj hattından U-tipi montaj hattına geçmeyi göz önünde bulundurdıklarının en temel nedenini bize sunmaktadır. Şaşırtıcı olarak ufak deneysel ve tecrübeyle elde edilen datalar bu savı desteklemektedir [3].

Geleneksel düz montaj hatları ile U-tipi montaj hatlarının arasındaki asli farklılık montaj hattının tasarımıdır. Geleneksel düz montaj hatlarında istasyonlar düz bir hat boyunca yerleştirilirken, U-tipi montaj hatları hattın giriş ve çıkış yönleri aynı tarafta olacak biçimde ‘U’ biçiminde tasarlanmıştır. Pek çok firma, Tam Zamanında Üretim (TZÜ) sistemi prensiplerinin üretim ortamına adaptasyonunun neticesi olarak, U-tipi montaj hatlarını TZÜ sisteminin önemli bir bileşeni şeklinde kullanmaktadır.

U-tipi montaj hatları klasik düz hatlara nazaran üretim anlamında daha esnektir. Şöyle ki, müşteri taleplerinde oluşan ani değişimlere daha hızlı yanıt verebilmekte, işçiler arasında yardımlaşmayı ve ortaya çıkan sorunların daha çabuk çözülebilmesine meydan vermektedir. Şekil 2.2.’de örnek bir U-tipi montaj hattı verilmiştir.

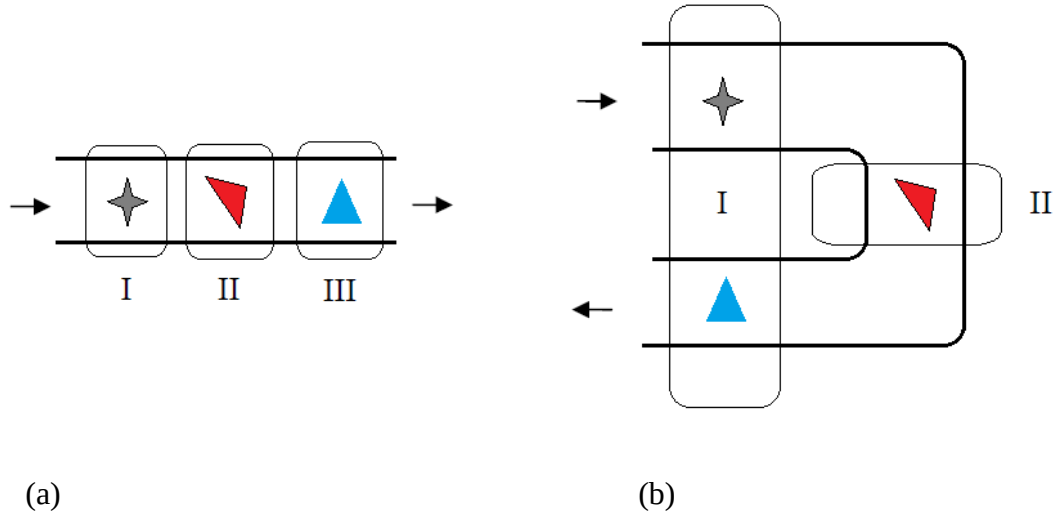


Şekil 2.2. U-tipi montaj hattı

Üreticiler, çeşitlilik gösteren müşteri taleplerine yanıt verebilmek için montaj hatlarını birden fazla ürün üretmek için de kullanmışlardır. Sadece tek bir ürün çeşidinin üretildiği montaj hatlarına tek modelli montaj hatları denir. Tek bir üründen yüksek miktarda üretim yapılan bu montaj hatlarında değişkenlik ve farklılık isteyen müşteri taleplerine yanıt vermek mümkün değildir.

Birbirlerine benzer üretim karakteristikleri gösteren iki veya daha fazla ürünün üretiminin yapıldığı montaj hatlarına ise karışık modelli montaj hatları adı verilir. Karışık modelli montaj hatları talep değişimlerine hızlı yanıt verebilmeye ve farklı özelliklerde ürün talep eden müşterileri tatmin etmeye olanak sağlamaktadır. Karışık modelli montaj hatları TZÜ sisteminin önemli bir parçası olarak görülmektedir [4].

Karışık modelli montaj hatları şu şekilde gösterilebilir:



Şekil 2.3. (a) Karışık modelli geleneksel düz montaj hattı ve (b) karışık modelli U-tipi montaj hattı

2.1.3. Paralel montaj hatları

Montaj hatları, yüksek boyutlu standardize edilmiş ürünlerin üretimi için konveyör bantları ya da diğer malzeme taşıma sistemleri ile birbirine bağlanmış seri iş istasyonlarını içerir.

İş parçalarının operasyonları görev istasyonlarında yüksek derecede uzmanlaşmış operatörler tarafından defalarca yapılır. Bu tür sistemlerin ana amacı standardize edilmiş ürünlerin kitle üretimi için maliyetleri düşürmektir. Maliyet kavramı içerisinde zaman yönetimi de önemli bir yer teşkil etmektedir.

Günümüzün üretim sistemlerinin gereksinimleri müşteri ihtiyaçlarının çeşitlenmesiyle değişmekte ve sistem kendisini bu istek karşısında geliştirmektedir. Ürün çeşitliliğinin artmasıyla montaj anlayışının düzenlenmesi değişmiştir. Bunun anlamı yüksek verimlilikteki ve esneklikteki montaj hatlarının farklı ürünlerin üretilmesi için tasarlanıp, dengelenmesi zorunluluk haline gelmiştir.

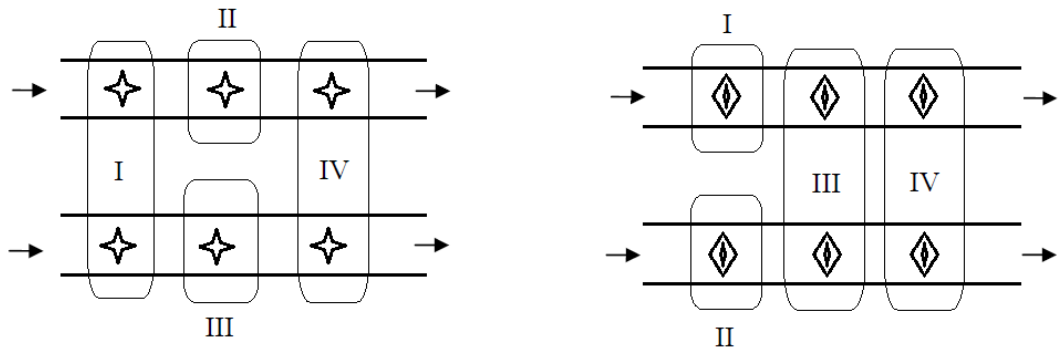
Bazı araştırmacılar montaj hatlarının verimliliğinin hatların komşuluklarının onları dengelerken göz önünde bulundurulmasıyla daha da geliştirileceğini tartışmaktadırlar [5, 6].

Hata hassasiyetini azaltmak, kapasite ve ürün çeşitliliği açısından esnekliği arttırmak için paralel hatlar endüstriyel ve akademik çevrelerdeki araştırmacıların dikkatini çekmektedir [7].

İki veya daha fazla montaj hattının birbirlerine paralel olarak yerleştirilip operatörlerin komşu montaj hatlarında çalışmalarını sağlamak montaj hatlarında kapasitenin artırılmasının bir yoludur. Bu tür montaj hatlarına paralel montaj hatları denir.

Paralel montaj hatları montaj hattı uzunluğunu kısaltır, ihtiyaç duyulan operatör sayısını azaltır ve montaj hatlarından birinde problem ortaya çıktığında diğer montaj hatlarının kesintisiz çalışmalarına ve problemin kısa zamanda çözülebilmesine olanak sağlar.

Paralel montaj hatlarında aynı veya farklı ürünler üretilbildiği gibi her bir montaj hattının çevrim zamanı da aynı veya farklı olabilir. Ayrıca paralel montaj hatlarında karışık modelli üretim de yapılabilir. Bu özelliklerinden dolayı üreticilere rekabet avantajı sağlamaktadır [8].



Şekil 2.4. Paralel montaj hatları

2.2. Çift Taraflı Montaj Hatları

2.2.1. Temel kavramlar

Montaj: Planlanan işlemler sonucu, planlanan parçaların birleştirilip kullanıma hazır hale getirilmesidir. Henüz tamamlanmamış ürünlere iş parçası adı verilir.

Görev (operasyon): Genellikle daha küçük parçalara bölünemeyen, montaj sürecinde yapılması gereken toplam işi oluşturan öğelerdir.

Görev (operasyon, işlem) zamanı: Belirli bir görevin yerine getirilmesi için gereken zamana denir.

İstasyon: Çift taraflı montaj hattı üzerinde sağ ve sol taraflara konumlandırılmış belirli bir görev grubunun (bir veya daha fazla görevin) yerine getirildiği bölümlerdir. Her bir istasyonda sadece bir operatörün çalışabildiği varsayılır. İstasyon sayısı operatör sayısını temsil etmektedir.

Çift (eş) istasyon: Çift taraflı montaj hattı üzerine yerleştirilmiş karşılıklı istasyonlar çiftine denir. Çift istasyon sayısı montaj hattının uzunluğunu temsil etmektedir.

İş yükü: Belli bir istasyona atanan görevler kümesi.

İstasyon zamanı: Belli bir istasyona atanan görevlerin görev süreleri toplamı.

Çevrim zamanı: İki ürünün montaj hattından arka arkaya tamamlanarak ayrılması arasından geçen zamandır. (Kısıtı: Çevrim zamanı en büyük görev süresinden daha küçük olamaz.)

Operasyon tarafı: Çift taraflı montaj hatlarında bazı görevler hattın sadece tek bir tarafına yapılabilirler. Bu görevlere L-tipi (sol taraf) ve R-tipi (sağ taraf) görevleri

denir. Bazı görevler ise hattın herhangi bir tarafında yapılabilirler. Bu görevlere ise E-tipi (her iki taraf) görevler adı verilir.

Öncelik şeması: Görevler arasındaki öncelik ilişkilerini, görev zamanlarını ve operasyon taraflarını göstermede kullanılan bir araçtır. Öncelik şemasında görevleri temsil eden düğümler, görev zamanları, görevlerin operasyon tarafları ve öncelik ilişkisini belirten oklar yer alır.

Operasyon sırası: Belli bir istasyona atanan görevlerin öncelik ilişkileri ve boş zamanlar dikkate alınarak oluşturulan uygulama sırası.

Başlama zamanı: Bir görevin bütün öncülleri tamamlandıktan sonra operasyon sırasına bağlı olarak uygulamaya başlandığı zaman.

Bitiş zamanı: Bir görevin operasyon sırasına bağlı olarak tamamlandığı zaman. (Kısıtı: Bir görevin bitiş zamanı çevrim zamanından büyük olamaz.)

Boş zaman: Çevrim zamanı ile istasyon zamanı arasındaki fark. Çift taraflı montaj hatlarında boş zamanlar iki ardışık görev arasında, bir istasyondaki ilk görevden önce veya bir istasyondaki son görevden sonra ortaya çıkar [9].

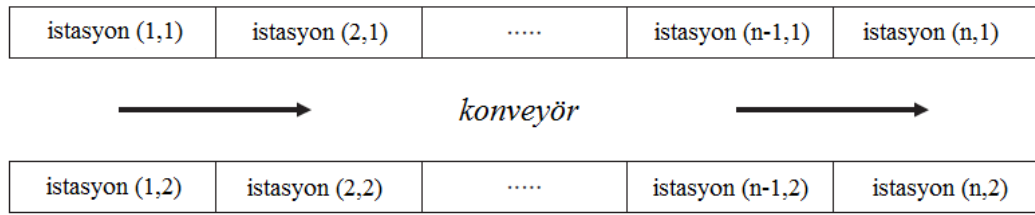
2.2.2. Çift taraflı montaj hattı nedir?

Çift taraflı montaj hatları genellikle otomobil, kamyon ve otobüs gibi yüksek hacimli geniş ölçülere sahip standardize edilmiş ürünler için dizayn edilirler.

Yüksek hacimli ve geniş ölçülere sahip ürünlerde ürünün yapısı gereği montaj işlemi hattın sadece tek tarafında yapılamamaktadır. Bu durumda hattın her iki tarafını da kullanmak daha uygun olmaktadır.

Çift taraflı montaj hatlarında bazı görevler hattın tek tarafında yerine getirilebilirken, bazıları da hattın her iki tarafında da gerçekleştirilebilir.

Hattın tek tarafında gerçekleştirilebilen görevlere L-tipi (sol taraf) ve R-tipi (sağ taraf) görevler denir. Her iki tarafta da gerçekleştirilen görevlere ise hattın herhangi bir tarafında yapılabilir anlamında E-tipi (either side - her iki taraf) görevler adı verilir. Örneğin bir otobüsün kapılarının montaj işlemi otobüsün sağ tarafından daha rahat yapılabilirken, camlarının montajı her iki taraftan da yapılabilir.



Şekil 2.5. Çift taraflı montaj hattı [14]

İşlem öncüllüğü ve çevrim zamanı kısıtları yoluyla operasyon yönünün değerlendirilmesi değişebilir. Boş zamanlar bazen aynı eşleşmiş istasyonlara atanan görevler için kaçınılmaz olabilir [11]. Örneğin, p ve h görevleri İstasyon 1'e, i görevi İstasyon 2'ye atanmıştır. i ve p görevleri h görevinin öncülleridir. Karşılıklı iki istasyon bir çift istasyonu (İstasyon 1 ve İstasyon 2) oluşturur, bunların her birine ise diğeri için eş istasyon adı verilir. Çift taraflı montaj hatlarında aynı istasyonda arka arkaya yapılan montaj görevleri arasında bazen boş zamanlar ortaya çıkabilir. i ve p görevleri h görevinin öncülleri olduğu için bu görevler tamamlanmadan h görevi başlayamaz. Bu yüzden çift taraflı montaj hatlarında görevlerin sıraya bağlı bitiş zamanları dikkate alınır [9].

Çift taraflı montaj hatlarının tek taraflı montaj hatlarına karşı pek çok üstün olduğu nokta vardır. Bunlar: (i) montaj hattı uzunluğu tek taraflı montaj hattına göre daha kısadır; (ii) malzeme taşıma maliyetleri, işçi hareketleri, hazırlık zamanları ve bir ürünün toplam üretim zamanı daha düşüktür; (iii) alet ve tamir maliyetleri daha düşük olabilmektedir. Tek taraflı montaj hatlarında önemli olan hangi işin hangi istasyonda işleneceği iken çift taraflı montaj hatlarında hem hangi işin hangi istasyonda işleneceği hem de hangi sırayla işleneceği belirlenmelidir. Diğer bir

deyişle çift taraflı montaj hattında aralarında yakın öncelik ilişkisi olan iki iş karşılıklı istasyonlara atandığında biri tamamlanmadan diğeri başlayamayacağından boş zaman ortaya çıkabilecektir. Dolayısıyla hattı dengelerken işlerin sırasına bağlı tamamlanma zamanları dikkate alınmalıdır. Böylece problem daha karmaşık ve çözülmesi zor bir hâl almaktadır [10]. Aynı iş parçasının iki operatör tarafından eş zamanlı montaj operasyonlarının yapılması ortaya çıkan boş zamanları azaltmakta ve dolayısıyla istasyon sayısının düşmesine neden olmaktadır.

Klasik tek taraflı montaj hattı dengelemesinde sadece öncelik ilişkileri ve çevrim süresi kısıtı olmak üzere iki temel kısıt varken, çift taraflı montaj hattı dengelenmesinde yalnızca bu kısıtları ele almak yeterli değildir. Çift taraflı montaj hattının içeriğindeki kısıtları şu şekilde sıralamak mümkündür:

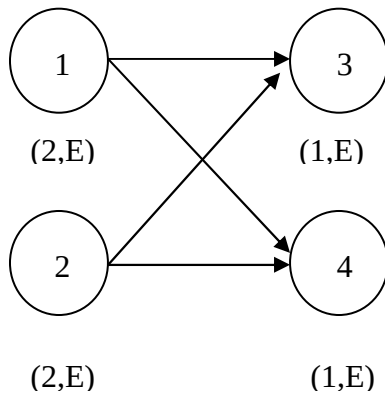
- Konumsal kısıtlar: Belirli bir işin işletmenin yerleşiminden dolayı önceden belirlenmiş bir istasyona atanması gerektiğini belirtir.
- Bölgesel kısıtlar: Hangi işlerin aynı istasyona atanması (pozitif bölgesel) ve hangi işlerin aynı istasyona atanmaması (negatif bölgesel) gerektiğini gösterir.
- Senkronizasyon kısıtları: Hattın her iki yönünde aynı anda gerçekleştirilmesi gereken, aynı süreye sahip işleri tanımlar.

2.2.3. Çift taraflı montaj hattı dengeleme problemi ve çözüm yaklaşımları

Çift taraflı montaj hatlarının tasarımı ve hattın dengelenmesi ilk olarak Bartholdi tarafından incelenmiştir. Bartholdi, İlk Uygun Kural adı verilen basit bir atama kuralını önermiş ve temel olarak interaktif bir program üzerine yoğunlaşmıştır. Bartholdi bu çalışmasında, aynı ürün ve aynı çevrim zamanı için bir çift taraflı montaj hattında ihtiyaç duyulan istasyon sayısının tek taraflı montaj hattı dengesi için gerekli istasyon sayısına eşit veya daha küçük olduğu tezini savunmuştur [10].

Tek modelli montaj hatları için çift taraflı montaj hattı problemleri ile ilgili tüm çalışmalar günümüzün piyasa eğilimleri altında rekabet eden firmalar için hayati bir faktördür ve otomotiv endüstrisinin son aşamadaki montaj hatlarının belirtilmesinde temel teşkil etmektedir. Tek modelli ürünleri üretmede kullanılan hatlar günümüz şartlarına bağlı olarak yüksek seviyedeki kitle tipi üretim için uygun değildir. Müşteri ihtiyaç ve isteklerinin çeşitliliği ve hatta karmaşıklık seviyesindeki talepler tek modelli üretim hatlarında çalışmayı imkânsız hale getirmektedir. Ancak günümüz koşullarına en uygun şekle sahip olan karışık modelli montaj hatları (benzer ürünlerin her hangi bir sırada veya karışık olarak üretilmeye başlanabilir) kitle tipi üretimin gereksinimlerini karşılar [12]. Çift taraflı montaj hatlarında boş zamanlar, görevlerin işlem sırasına ve karşılıklı istasyonlardaki görevlerin öncelik ilişkilerine göre çevrim zamanı başlangıcında ve çevrim zamanı sonunda ortaya çıkmaktadır. Çift taraflı montaj hatlarında bu ortaya çıkan boş zamanları azaltmak, aynı iş parçasının iki operatör tarafından eş zamanlı montaj operasyonlarının yapılmasıyla gerçekleşir ve dolayısıyla bu şekilde istasyon sayısı azalır [8].

Aşağıdaki şekilde verilen öncelik şemasında dört görev bulunmaktadır ve bütün görevler her iki tarafta da yapılabilen (E-tipi) görevlerdir. Çevrim zamanı üç zaman birimi olarak alındığında, tek taraflı montaj hattı dengesi için gerekli istasyon sayısının üç; çift taraflı montaj hattı dengesi için gerekli istasyon sayısının iki olduğu görülür [8].



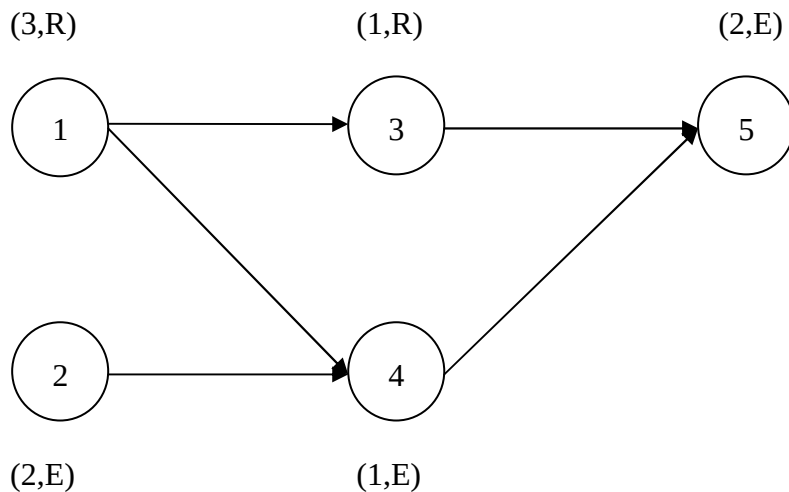
Şekil 2.6. Öncelik şeması [8]

Wu ve arkadaşlarının 2008 yılında yayınlanan bir çalışmalarına göre çift taraflı montaj hattı dengeleme problemleri (ÇMHDP) aynen tek taraflı olduğu gibi şu şekilde sınıflandırılabilir [13]:

1) Verilen bir çevrim zamanı için hattın uzunluğunu (çift istasyon sayısını) en küçüklemek. Tip 1 (TALBP I)

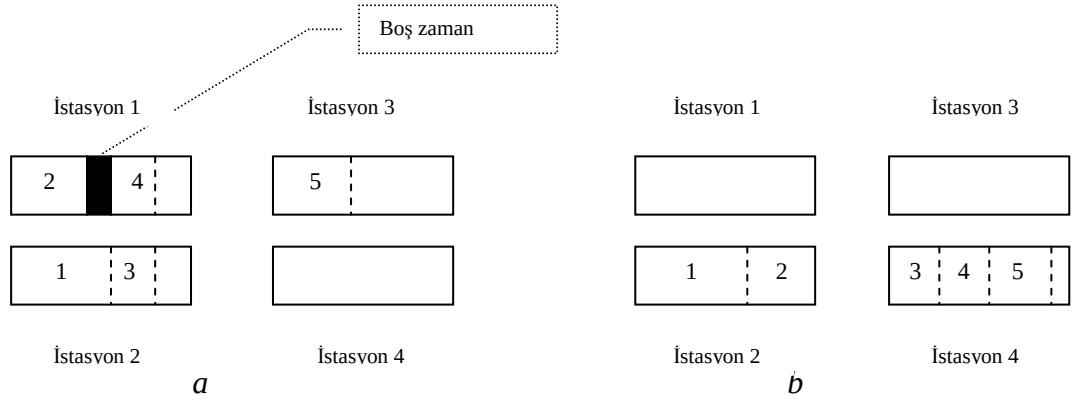
2) Verilen bir hat uzunluğu (çift istasyon sayısı) için çevrim zamanını en küçüklemek. Tip 2 (TALBP II)

ÇMHD kavramının daha iyi anlaşılması adına çeşitli çalışmalarda belirtilmiş örnekler kullanılmıştır. Bunlardan ilki Wu ve arkadaşlarının (2008) yılındaki çalışmalarında verdikleri örnek bir problem kullanılmıştır [13]. Örnek problemin öncelik ilişkileri, görev zamanları ve görevlerin operasyon tarafları Şekil 2.7’de verilmiştir. Görev sayısı 5’dir. Düğümlerin içerisindeki rakamlar görevleri, düğümlerin altında belirtilen (t_i, d_i) etiketleri sırasıyla i ’nci görevin görev zamanını ve operasyon tarafını (L: sol taraf, R: sağ taraf, E: herhangi bir taraf) göstermektedir. İki düğüm arasındaki yönlü ok öncelik ilişkisini ifade etmektedir.



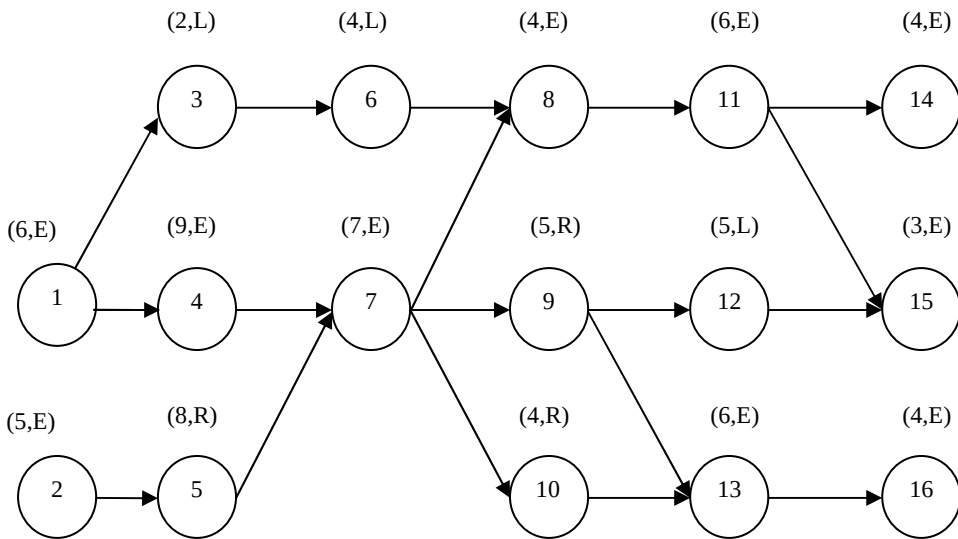
Şekil 2.7. 5 görevli örnek problemin öncelik ilişkileri şeması, görev zamanları ve görevlerin operasyon tarafları

Çevrim zamanı 5 olan örnek problem için iki adet çözüm bulunabilir. Her iki çözüm de minimal hat uzunluğuna ulaşır fakat ikincisi ilkinin nazaran daha uygundur çünkü daha az istasyon açılmıştır.



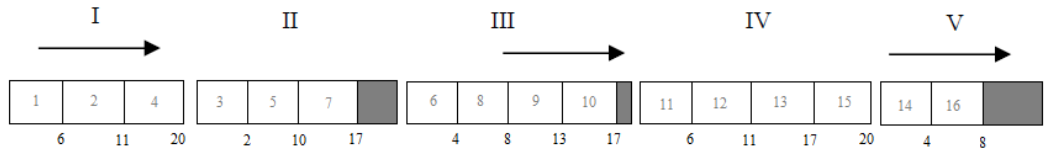
Şekil 2.8. Çevrim zamanı 5 olan örnek problem için iki çözüm. “a” yerleşiminde 3 istasyon açılmıştır; “b” yerleşiminde aynı hat üzerinde daha az istasyon açılmıştır (sadece 2 istasyon)

Konuyu daha iyi kavramak adına Lee ve arkadaşlarının 2001 yılındaki çalışmalarında verilen örnek bir problem kullanılmıştır [14].

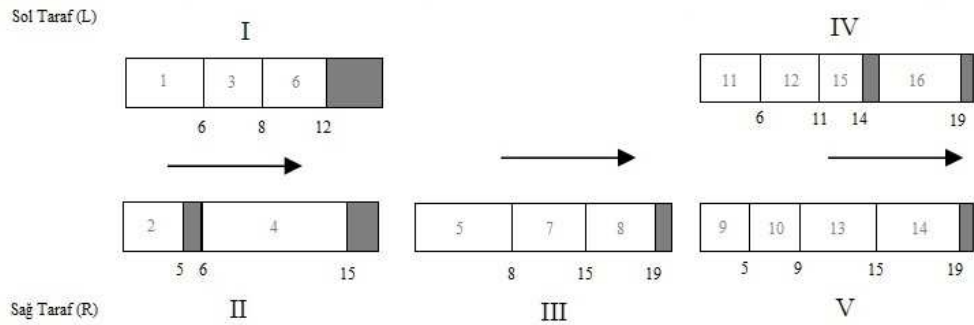


Şekil 2.9. Öncelik diyagramı

Örnek problemde verilen öncelik diyagramında görev sayısı 16 olup, öncelik ilişkileri, görev süreleri ve hattaki operasyon tarafları ilgili diyagramda belirtilmiştir. Şekil 2.10 çevrim zamanı 20 birim zaman için tek taraflı bir montaj hattı dengesini ve çift taraflı bir montaj hattı dengesini göstermektedir. Tek taraflı montaj hattı dengesinde bütün görevlerin R-tipi görevler olduğu varsayılmıştır. Her bir görevin başlama ve bitiş zamanları kutuların yanında verilmiştir.



(a)



(b)

Şekil 2.10. Çevrim zamanı 20 birim için istasyonlara görev atamaları (a) tek taraflı montaj hattı (b) çift taraflı montaj hattı

Şeklin (a) kısmında verilen tek taraflı montaj hattı dengesinde beş istasyon bulunmaktadır. Tek taraflı montaj hatlarında operatörler görevlerini herhangi bir kesintiye uğramaksızın yerine getirebilirler (aynı anda tek bir iş yapılacağı için), çünkü boş zamanlar çevrim zamanının sonuna doğru ortaya çıkmaktadır.

Şeklin (b) kısmında verilen çift taraflı montaj hattı dengelemesinde de beş istasyon vardır. Fakat burada tek taraflı montaj hattına nispeten hattın uzunluğu daha kısadır, hat üç adet çift istasyondan ibarettir. Bu tip hatların tek taraflıdan diğer bir farkı da bekleme zamanlarının (boş zaman) ortaya çıkma şeklindedir. Diyagramda da belirtilen öncelik ilişkilerinden dolayı operatörler görevlerini yaparlarken karşı tarafta yapılan bazı görevlerin bitmesini beklemektedirler. Örneğin, 16. görevin yapılabilmesi için 13. görevin tamamlanması gerekmektedir. Bu nedenle 15. görev ve 16. görev arasında bir birim boş zaman ortaya çıkmıştır.

Şekil 2.10'da verilen örnek çift taraflı montaj hattı dengesinde birincil amaç olarak çift istasyon sayısının (hattın uzunluğunun) en küçüklenmesi ikincil amaç olarak istasyon sayısının (operatör sayısının) en küçüklenmesi dikkate alınmıştır.

3. LİTERATÜR ARAŞTIRMASI

Montaj Hattı Dengeleme problemi ilk olarak Helgeson (1954) tarafından çalışılmıştır ve ilk defa Salveson (1955) tarafından matematiksel olarak ifade edilmiştir. Üretim karakteristiğinde ortaya çıkan ihtiyaçların etkisiyle yaşanan evrimle birlikte montaj hattı konusu daha da geliştirilerek farklı çözüm yollarını ve matematiksel modelleri içeren çok sayıda çalışma yapılmıştır [15, 16].

Talbot ve Patterson (1984) çalışmalarında tek modellenli üretim için geleneksel (düz) montaj hatlarını deterministik görev zamanlı matematiksel model kullanarak istasyon sayısının en küçüklenmesini amaçlamışlardır [17].

Agrawal (1985) yine tek modellenli deterministik görev zamanlı geleneksel (düz) montaj hatlarını birden çok amaç için optimize etmeye çalışmıştır. Bu çalışmada sezgisel model kullanılmıştır [18].

Rosenblatt ve Carlson (1985) ise kar en büyükleme esasına dayalı olarak tek modellenli geleneksel montaj hattını deterministik görev zamanıyla sezgisel ve matematiksel modeller kullanarak dengeleme çalışması yapmıştır [19].

Shtup ve Dar-El (1990) yine tek modellenli geleneksel montaj hatları için istasyon sayısını ve çevrim zamanını en küçükleyen bir matematiksel model geliştirmişlerdir. Ayrıca problemi sezgisel yaklaşım kullanarak çözümlenmişlerdir [20].

Plans ve Corominas (1999) tek modellenli geleneksel montaj hatlarında hat etkinliğinin en büyüklenmesini sağlayacak matematiksel bir model üzerinde durmuştur [21].

Bautista ve ark. (2000) çalışmalarında geleneksel montaj hatları için tek modellenli üretimde sezgiseller kullanarak istasyon sayısının en küçüklenmesi üzerinde durmuştur [22].

Ponnambalam ve ark. (2000) tek modellenli geleneksel montaj hatlarında düzgün iş yükü sağlayarak hat etkinliğini en büyükleyecek birçok amaçlı genetik algoritma geliştirmişlerdir [23].

Carnahan ve ark. (2001) fiziksel talep birleştirme kriterini kullanarak geleneksel tek modellenli montaj hattında çevrim zamanını en küçükleyecek bir genetik algoritma geliştirmişlerdir [24].

Ağpak ve Gökçen (2005) kaynak kısıtlı montaj hattı dengelenmesi problemi (RCALB) üzerinde durmuşlardır. Bu çalışmada geleneksel tek modellenli montaj hattında istasyon sayısını en küçükleyen deterministik görev zamanlı bir matematiksel model geliştirmişlerdir [25].

Mirales ve ark. (2007) dal sınır algoritmasını montaj hattına işçi ataması ve dengelemesi adına kullanmışlardır. Bu çalışmada tek modellenli geleneksel montaj hattı için çevrim zamanının en küçüklenmesi amaçlanmıştır [26].

Gökçen ve ark.(2006) yılındaki çalışmalarında tek modellenli üretim için paralel montaj hatlarının istasyon sayısının en küçüklemesini amaçlayan matematiksel bir model geliştirip, sezgisel yöntemlerle çözümlenmişlerdir [5].

Toklu ve Özcan (2008) basit U-tipi montaj hattı problemi için bir bulanık hedef programlama modeli geliştirmişlerdir. Bu model çok amaçlı olup görev zamanı deterministiktir [28].

Boysen ve ark (2008) montaj hatlarında hangi modelin ne zaman kullanılacağını anlatan bir çalışma yayınlamışlardır [29].

Otto ve Scholl (2011) çalışmalarında montaj hattı kavramına ergonomik riskleri de eklemiştir. Basit montaj hattı problemini ergonomik risklerle birleştirip Ergo SALBP adını verdikleri montaj hattını sezgisel yöntemlerle çözümleyip benzetimini yapmışlardır [27].

Wei ve ark. (2011) SALBP-E adı verilen basit montaj hattı dengeleme problemlerinde etkinliği esas alan bir model geliştirmişlerdir [30].

Mosadegh ve ark. (2012) meta sezgisel yöntemler kullanarak karışık modelli montaj hattı dengelemesi ve bir sıralama problemine çözüm getirmiştir. Bu çalışmada tavlama benzetimi ve genetik algoritma kullanılmıştır [31].

Literatürde montaj hattı dengeleme üzerine pek çok çalışma yapılmıştır. Ancak çift taraflı montaj hattı dengeleme pek az sayıda araştırmacı tarafından incelenmiş olup son yıllarda bu konu üzerindeki araştırma sayısı yavaş yavaş artmaktadır. Bu konuda yapılan çalışmalarda ağırlığı deterministik tek modelli düz çift taraflı montaj hattı dengeleme problemi oluşturmaktadır.

ÇMHD problemini ilk olarak Bartholdi (1993) incelemiştir. Çalışmasında bir gerçek hayat problemine yer veren Bartholdi (1993), bir interaktif program ile bir görev atama kuralı geliştirmiştir. Bu interaktif programı tasarlanmasındaki amacının da montaj hattı yöneticisine yardımcı olmak olduğunu belirtmiştir [10].

Lee ve ark. (2001) görevleri, iş ilişkilerine göre ve görev sarkmalarını (boş zamanları) en küçükleyecek şekilde ÇMHDP-I ve ÇMHDP-II için bir grup atama yöntemi geliştirmişlerdir [14].

Lapierre ve Ruiz (2004) ÇMHD problemleri için öncelik tabanlı bir sezgisel yöntem geliştirerek endüstriyel bir uygulama gerçekleştirmişlerdir. Burada montaj işlemlerinin farklı yüksekliklerde yapıldığı durumu incelemişlerdir [40].

Baykasoğlu ve Dereli (2008) bölgeleme kısıtı altında karınca kolonisi sezgiselinin ÇMHD problemine nasıl uygulanacağını gösteren ilk çalışmalardan birini sunmuşlardır [41].

Wu ve ark. (2008) dal-sınır algoritmasıyla ÇMHD problemine çözüm aramışlardır [13].

Xiaofeng ve ark. (2008) ÇMHD problemi için bir istasyon odaklı enumerativ algoritma önermişlerdir. Bu çalışmalarında zaman transfer fonksiyonu adı verilen bir fonksiyon kullanılmış olup önerilen algoritma Hoffmann sezgiseli ile entegre edilmiştir [37].

Kim ve ark. (2009) belirli sayıdaki çift istasyon sayısında çevrim zamanını en küçükleyen matematiksel modeli sunmuşlardır. Ayrıca bu problemi genetik algoritma tabanlı bir sezgisel algoritma ile çözmüşlerdir [11].

Simaria ve ark. (2009) karışık modelli ÇMHD problemi için bir matematiksel model sunmuşlardır. Bu çalışmada ortaya çıkan engel modelin karmaşıklığının yüksek dereceye sahip olmasıdır. Bu durum nedeniyle herhangi bir test problemini bu modeli kullanarak çözememişlerdir. Ayrıca bu çalışmalarında karınca kolonisi optimizasyonu tekniğine dayalı bir sezgisel yaklaşım sunmuşlardır [12].

Özcan ve Toklu (2009b) ÇMHD problemi için verilen bir çevrim zamanı için çift istasyon sayısını en küçükleyen bir matematiksel model, hedef programlama modeli ve bulanık hedef programlama modeli geliştirmişlerdir [32].

Özcan ve Toklu (2009a) ÇMHD problemi için verilen bir çevrim zamanı için istasyon sayısını en küçükleyecek bir matematiksel model önermişlerdir. Çok modelli ÇMHD problemi için bu çalışmada tavlama benzetimi kullanılmıştır [9].

Özcan, Gökçen ve Toklu (2010) iki ya da daha fazla çift taraflı montaj hattının eş anlı olarak dengelenmesi temeline dayanan paralel ÇMHD problemi için bir tabu araması algoritması sunmuşlardır [33].

Özcan ve Toklu (2010) daha önce ÇMHD problemlerinde çok fazla değinilmeyen sıralamaya bağlı hazırlık süresi konusu üzerinde durmuşlardır (TALBPS). Bu çalışmalarında bir karışık tamsayılı programlama modeli önerilmiş ve bu model çift istasyon sayısını en küçüklemeyi amaçlamıştır. Bu problemin çözümünde COMSOAL tekniği üzerine oturan (2-COMSOAL/S) tekniğini kullanmışlardır [34].

Özcan (2010) stokastik görev zamanlı ÇMHD problemine, rastgele-kısıtlı, kesikli-doğrusal, karışık tamsayılı programlama ve bir tavlama benzetimi algoritmasıyla çözüm bulmaya çalışmıştır [35].

Taha ve ark. (2011) ÇMHD problemlerinden oluşan bir problem setini genetik algoritma kullanarak çözmüşlerdir [36].

Özbakır ve Tapkan (2011) alan kısıtı altındaki ÇMHD problemini verilen bir çevrim zamanıyla istasyon sayısını en küçükleyecek şekilde düzenlemek için arı algoritmasını (BA) kullanmışlardır [39].

Chutima ve Chimklai (2012) çok amaçlı ÇMHD probleminde parçacık sürü optimizasyonunun özel bir dalı olan negatif bilgi parçacık sürü algoritmasını (PSONK) önermişlerdir. Sonuçlara Pareto optimalite testini uygulamışlardır. Ayrıca sonuçları Kesikli parçacık sürü optimizasyonu (DPSO), çok amaçlı genetik algoritma (NSGA II) ve COMSOAL tekniği ile elde edilen çeşitli senaryolarla karşılaştırmışlardır [38].

Çizelge 3.1’de literatürde yer alan konu ile ilgili çalışmaların özeti verilmiştir.

Çizelge 3.1. Çift taraflı montaj hattı dengeleme üzerine yapılan çalışmalar

Problem Çeşidi	Algoritma/Yöntem	Çalışmayı Yapan
ÇMHDP -I	İnteraktif program ve görev atama kuralı	Bartholdi (1993)
ÇMHDP -I, ÇMHDP -II	Grup atama prosedürü	Lee ve ark. (2001)
ÇMHDP -I *	Öncelik tabanlı bir sezgisel yöntem	Lapierre ve Ruiz (2004)

Çizelge 3.1.(Devam) Çift taraflı montaj hattı dengeleme üzerine yapılan çalışmalar

Problem Çeşidi	Algoritma/Yöntem	Çalışmayı Yapan
ÇMHDP -I	Karınca kolonisi optimizasyonu algoritması	Baykasoglu ve Dereli (2008)
ÇMHDP -I	Dal-sınır algoritmasıyla	Wu ve ark. (2008)
ÇMHDP -I	İstasyon odaklı enumerativ algoritma (Hoffmann sezgiseli)	Xiaofeng ve ark. (2008)
ÇMHDP -II	Matematiksel model, genetik algoritmalar	Kim ve ark. (2009)
ÇMHDP -I	Matematiksel model, tavlama benzetimi	Özcan ve Toklu (2009a)
ÇMHDP -I	Matematiksel model, hedef programlama modeli ve bulanık hedef programlama modeli	Özcan ve Toklu (2009b)
PÇMHDP -I	Tabu araması algoritması	Özcan, Gökçen ve Toklu (2010)
ÇMHDP -I	Genetik algoritma	Taha ve ark. (2011)
ÇMHDP -I	Arı algoritmasını (BA)	Özbakır ve Tapkan (2011)

Çizelge 3.1.(Devam) Çift taraflı montaj hattı dengeleme üzerine yapılan çalışmalar

Problem Çeşidi	Algoritma/Yöntem	Çalışmayı Yapan
ÇMHDP -I	Negatif bilgi parçacık sürü algoritmasını (PSONK), Pareto optimalite testini, Kesikli parçacık sürü optimizasyonu (DPSO), çok amaçlı genetik algoritma (NSGA II) ve COMSOAL tekniği	Chutima ve Chimklai (2012)

ÇMHDP -I *: Lapierre ve Ruiz (2004) bu çalışmasında istasyon sayısı açısından önerilen yöntemin daha iyi neticeler verdiği ancak firmaya ait çözümde daha uygun bir iş yükü dağılımının olduğu görülmüştür [40].

4. KARIŞIK MODELLİ ÇİFT TARAFLI MONTAJ HATTI DENGEME

Karışık modellenli çift taraflı montaj hatları (KÇMH) herhangi bir model dizisinde ve karışık modelde benzer üretim karakteristiklerine sahip ürün modellerinin ($\forall m \in M$) imalatı için tasarlanmıştır [9]. Bir başka deęişle farklı ürünlerin veya aynı ürünün deęişik modellerinin aynı hat üzerinde üretilebilmesi için tasarlanmıştır [42]. Her model kendi içinde bir öncelik ilişkisine sahiptir ve bunlar tek bir öncelik diyagramında birleştirilebilirler.

Bu bölümde sunulan matematiksel model Özcan ve Toklu çalışmasında önerilen matematiksel modele dayanmaktadır. Bu çalışmalarında Özcan ve Toklu 1.tip karışık modellenli çift taraflı montaj hattı dengeleme (KÇMHD-I / MTALB-I) problemini çözmek için yeni bir karışık tam sayılı programlama modeli geliştirmişlerdir [9]. Oluşturulan matematiksel model Kim ve arkadaşları tarafından sunulan 2.tip karışık modellenli çift taraflı montaj hattı dengelemenin (KÇMHD-II / MTALB-II) matematiksel formülasyonu üzerine kurulmuştur [11].

Bu bölümde incelenilen KÇMHD problemi için aşağıdaki varsayımlar dikkate alınmıştır [9]:

- Benzer üretim karakteristiklerini gösteren ürün modelleri aynı çift taraflı montaj hattında üretilirler.
- Operatörler hattın her iki tarafında paralel olarak görevlerini yaparlar.
- Bazı görevler hattın sadece tek tarafında, bazıları da her iki tarafında yapılmayı gerektirirler.
- Farklı modellerin öncelik diyagramları bilinmektedir.
- Macaskill'in fikri olan birleştirilmiş öncelik diyagramı işletilmiştir [43].
- Görev zamanları deterministik olup, istasyon atamalarından bağımsızdır.
- Operatörlerin seyahat zamanları yok sayılmıştır.
- Paralel görevlere ve paralel istasyonlara izin verilmemiştir.
- Ara stoklara müsaade edilmemiştir.

- Farklı modeller arasında ortak görevler bulunmaktadır.
- Görevin tamamlanma süresi bir modelden diğerine farklılık gösterebilir ve aynı zamanda sıfıra da eşit olabilir.

Bir m modeli için i görevi ($\forall i \in I$) belirli bir çevrim zamanında (t_{im}) icra edilir. Ürün modelleri önceden belirlenmiş bir planlama ufku (P) boyunca üretilirler. Planlama ufkundaki m modeli için talep D_m şeklinde gösterilir. KÇMHD-1 problemi için gerekli olan hattın çevrim zamanı (C) aşağıdaki gibi hesaplanmıştır [12].

$$C = \frac{P}{\sum_{m \in M} D_m} \quad (4.1)$$

Birim model m sayısı (q_m) da şu şekilde hesaplanır:

$$q_m = \frac{D_m}{\sum_{m \in M} D_m} \quad \forall m \in M \quad (4.2)$$

Özcan ve Toklu çalışmalarında geliştirdikleri modeli anlatırken Kim ve arkadaşlarının geliştirdikleri modelin sadece literatürdeki ÇMHD problemi için yayınlanan matematiksel model olduğunu belirtmişlerdir [9]. Bu modelin öncelikli amacı verilen çift istasyon sayısına göre çevrim zamanını en küçüklemeektir. Ancak ÇMHD-II problemi için geliştirilen bu model, Özcan ve Toklu tarafından KÇMHD-I problemi için dönüştürülmüştür [9]. Bu amaçla, KÇMHD-I problemini çözmek için bir yeni karışık tamsayılı programlama formülasyonu geliştirmişlerdir. Önerilen modelle öncelikli amaçlarının verilen bir çevrim zamanıyla çift istasyon sayısını (hat uzunluğunu) en küçüklemeek olduğunu belirtmişlerdir.

KÇMHD-I için önerdikleri matematiksel model şöyledir:

$$\text{Minimize} \sum_{j \in J} (F_j + G_j) + \varepsilon \cdot \sum_{j \in J} \sum_{k=1,2} U_{jk} \quad (4.3)$$

Subject to

$$\sum_{j \in J} \sum_{k \in K(i)} x_{ijk} = 1 \quad \forall i \in I \quad (4.4)$$

$$\sum_{g \in J} \sum_{k \in K(h)} g \cdot x_{hgk} - \sum_{j \in J} \sum_{k \in K(i)} j \cdot x_{ijk} \leq 0 \quad \forall i \in I - P_0, \quad h \in P(i) \quad (4.5)$$

$$t_{im}^f \leq C \quad \forall i \in I, \quad m \in M \quad (4.6)$$

$$t_{im}^f - t_{hm}^f + \psi \cdot \left(1 - \sum_{k \in K(h)} x_{hjk} \right) + \psi \cdot \left(1 - \sum_{k \in K(i)} x_{ijk} \right) \geq t_{im}$$

$$\forall i \in I - P_0, \quad h \in P(i), \quad j \in J, \quad m \in M \quad (4.7)$$

$$t_{pm}^f - t_{im}^f + \psi \cdot (1 - x_{pj k}) + \psi \cdot (1 - x_{ijk}) + \psi \cdot (1 - z_{ip}) \geq t_{pm}$$

$$\forall i \in I, \quad m \in M, \quad p \in \{r \mid r \in I - (P_a(i) \cup S_a(i) \cup C(i)), i < r\},$$

$$j \in J, \quad k \in K(i) \cap K(p) \quad (4.8)$$

$$t_{im}^f - t_{pm}^f + \psi \cdot (1 - x_{pj k}) + \psi \cdot (1 - x_{ijk}) + \psi \cdot z_{ip} \geq t_{im}$$

$$\forall i \in I, \quad m \in M, \quad p \in \{r \mid r \in I - (P_a(i) \cup S_a(i) \cup C(i)), i < r\},$$

$$j \in J, \quad k \in K(i) \cap K(p) \quad (4.9)$$

$$\sum_{i \in I} x_{ijk} - \|W_{jkm}\| \cdot v_{jkm} \leq 0 \quad \forall j \in J, \quad m \in M, \quad k \in K(i) \quad (4.10)$$

$$\sum_{m \in M} v_{jkm} - \|M\| \cdot U_{jk} = 0 \quad \forall j \in J, \quad k = 1, 2 \quad (4.11)$$

$$\sum_{k=1,2} U_{jk} - 2 \cdot F_j - G_j = 0 \quad \forall j \in J \quad (4.12)$$

$$x_{ijk} \in \{0, 1\} \quad \forall i \in I, \quad j \in J, \quad k \in K(i) \quad (4.13)$$

$$z_{ip} \in \{0,1\} \quad \forall i \in I, p \in \{r \mid r \in I - (P_a(i) \cup S_a(i) \cup C(i)), i < r\} \quad (4.14)$$

$$F_j, G_j \in \{0,1\} \quad \forall j \in J \quad (4.15)$$

$$U_{jk} \in \{0,1\} \quad \forall j \in J, k = 1, 2 \quad (4.16)$$

$$v_{jkm} \in \{0,1\} \quad \forall j \in J, m \in M, k = 1, 2 \quad (4.17)$$

$$t_{im}^f \geq t_{im} \quad \forall i \in I, m \in M \quad (4.18)$$

Eş. 4.3 amaç fonksiyonu olup çift istasyon sayısını (hat uzunluğunu) öncelikli amaç olarak en küçükler. Ayrıca istasyon sayısını (operatör sayısını) ikincil amaç olarak en küçükler. Kısıtlara gelirse, Eş. 4.4 atama kısıtıdır, her bir görevin sadece bir istasyona atanmasını sağlar. Eş. 4.5 görevler arasındaki bütün öncelik ilişkilerini sağlayan bir öncelik ilişkileri kısıtıdır. Eş. 4.6 her model m için $\forall m \in M$ her bir görevin bitirilme zamanının çevrim zamanını aşamayacağını gösteren bir çevrim zamanı kısıtıdır. Eş.4.7, Eş.4.8 ve Eş.4.9 tüm m modeli için $\forall m \in M$ görevlerin sıraya bağlı bitiş zamanlarını kontrol eden kısıtlardır. Her i ve h görev çifti için, eğer h görevi i görevinin öncülü ise ve bu görevler aynı j çift istasyonuna atanmış ise Eş. 4.7 aktif hale gelir, $t_{im}^f - t_{hm}^f \geq t_{im}$ tüm m modeli için $\forall m \in M$. Eğer iki görev çifti arasında öncelik ilişkisi yok ise ve bu görevler aynı (j, k) istasyonuna atanmış ise, Eş. 4.8 ve Eş. 4.9 aktif hale gelir. Eğer i görevi p görevinden önce istasyona atandı ise, Eş. 4.8 tüm m modeli için $(\forall m \in M) t_{pm}^f - t_{im}^f \geq t_{pm}$ halini alır. Aksi durumda, Eş. 4.9 tüm m modeli için $(\forall m \in M) t_{im}^f - t_{pm}^f \geq t_{im}$ halini alır. Eş. 4.10 ve Eş. 4.11 Gökçen ve Erel tarafından geliştirilen karışık modellenli tek taraflı montaj hattı dengeleme problemi için istasyon kısıtlarıdır [44]. Bu kısıtlar istasyon sayısını tüm ürün modelleri için aynı yapmayı sağlarlar. Bu kısıtlar KÇMHD için tadil edilmiştir. Eğer tüm modeller için (j,k) istasyonuna görev atanırsa sıfırdan farklı bir değer olarak tüm modellerde bu istasyondan faydalanılır. Eğer tüm modeller için bir ya da birden fazla görev (j,k) istasyonuna atanırsa U_{jk} bir değerini aksi halde de sıfır değerini alacaktır.

Eş. 4.12 ile çift istasyon sayısı (hattın uzunluğu) bulunmaktadır. Eğer bir eş istasyonun her iki tarafı kullanılıyorsa Fj bir değerini Gj sıfır değerini alır. Yani çift istasyonlardan sol taraftaki $(j,1)$, aynı çift istasyonda sağ taraftaki $(j,2)$ değerini alıyorsa Fj bir değerini Gj sıfır değerini alır. Sadece tek tarafı kullanılıyor ise Fj sıfır değerini Gj bir değerini alır. Dolayısıyla, toplam hat uzunluğu $\forall j \in J$ için Fj ve Gj toplamına eşittir. Eş. 4.13 ve Eş.4.17 arasındaki kısıtlar ise işaret kısıtlarıdır. Eş. 4.18 m üretim modeli için i görevinin bitirilme zamanının, m modelinin i görevinin tamamlanma zamanına eşit ya da daha büyük olmasını sağlamaktadır.

Ayrıca Özcan ve Toklu bu karışık tamsayılı programlama modeline ek olarak bölgeleme kısıtları, konumsal kısıtlar ve senkronize (eşzamanlılık) kısıtları eklenebileceğini belirtmişlerdir [9]. Bazı görevler aynı istasyona atanmaya zorlanabilir, diğer görevlerin de aynı istasyona atanması yasaklanabilir. Bu kısıtlar sırasıyla pozitif ve negatif bölgeleme (zoning) kısıtları olarak bilinirler [9]. Aşağıdaki kısıtlardan Eş. 4.19 h ve i görevlerinin aynı (j,k) istasyonuna atanmasını sağlar. Eş. 4.20 ise h görevi ve i görevinin aynı (j,k) istasyonuna atanmamasını sağlar.

$$x_{hjk} - x_{ijk} = 0 \quad \forall (i,h) \in CZ, j \in J, k \in K(i) \cap K(h) \quad (4.19)$$

$$x_{hjk} + x_{ijk} \leq 1 \quad \forall (i,h) \in IZ, j \in J, k \in K(i) \cap K(h) \quad (4.20)$$

Kim ve arkadaşları konumsal kısıt adını verdikleri bir kısıtı modele eklenmiştir [45]. Konumsal kısıta göre, bazı görevlerin (sabit yerleri olan özellikle büyük boyutlu ürünler için) büyük ve ağır tesisler tarafından yapılması gereklidir. Bu tür görevler önceden belirlenmiş (j,k) istasyonuna atanmak zorundadır.

Aşağıda bu durumu içeren kısıt mevcuttur. Eş. 4.21 konumsal bir kısıttır ve r görevinin önceden belirlenmiş olan (g,f) istasyonuna atanmasını sağlar.

$$x_{rgf} = 1 \quad \forall (r,(g,f)) \in PZ \quad (4.21)$$

Bazı görevler karşılıklı istasyonlarda eş zamanlı olarak yapılmaya zorlanırlar. Bu tip görevler senkronize (eş zamanlılık) görevler olarak adlandırılmıştır [12].

Eş. 4.22 ve Eş. 4.23 h ve i görevlerinin tüm modeller için aynı başlangıç zamanına sahip birbirlerine direkt karşılıklı istasyonlara atanabilmesini sağlamaktadır.

$$x_{hjf} - x_{ijk} = 0 \quad \forall (i, h) \in SZ, j \in J, k \in K(i), f \in K(h), k \neq f \quad (4.22)$$

$$t_{hm}^f - t_{hm} - t_{im}^f + t_{im} = 0 \quad \forall (i, h) \in SZ, m \in M \quad (4.23)$$

5. ARI KOLONİSİ OPTİMİZASYON ALGORİTMASI

5.1. Arı Kolonisi Algoritmasına Giriş

Lucic ve Teodorovic'in tabiriyle (2002) yapay yaşam (*Artificial Life*), biyolojik bilgi birikimi ve tekniklerini zorlu mühendislik, yönetim, kontrol ve sayısal problemleri çözmeye yardımcı olmak adına kullanır. Doğal sistemler çok basit bireysel organizmaların dinamik olarak birbirleriyle etkileşerek oldukça karmaşık görevleri başarıyla yerine getirebileceklerini bize öğretir. Arı sistemi sürü zekâsı alanında yeni bir yaklaşım olarak nitelendirilebilir [46].

Sürü zekâsı olarak adlandırılan algoritmalar böcek hareketlerinin problemleri çözebilme kabiliyetini kullanarak onların davranışlarına odaklanarak etkin bir meta-sezgisel geliştirmeyi amaçlamıştır. Bu türdeki böcekler arasındaki iletişim sistemleri bilimsel optimizasyon problemlerine uyarlanmıştır. Yani bir nevi doğadan ilham alınmıştır. Bu türden iletişime arıların yapmış oldukları sallanma dansı adı verilen hareketler bütünü çok iyi bir örnek teşkil etmektedir. Bu dans ile gözcü arılar koloninin diğer üyelerine çiçek nektarının istikametini ve yolun uzunluğunu, ayrıca nektarın miktarını bildirirler [39].

1973 yılında Nobel ödülü alan Karl Von ve onun öğrencisi arıların dans dillerinin çözümlemesini gerçekleştirmişlerdir [53]. Arıların salınım dansında besin kaynağının istikameti konusunda güneşe göre konumlandıkları dans yönü ve besin kaynağının mesafesi konusunda da dans şekli ve titreme hareketinin sayısı önemli bir yer tutmaktadır.

Tarpy ile Wenner ve Wells kaşif arıların üzerindeki nektar kokusunun kaynak hakkında bilgisiz olan arıların besin kaynağının çeşidi konusunda bilgilendiren ilk ipucu olduğunu çalışmalarında tartışmışlardır [54, 55].

Bu başarılı mekanizmalar ile kaşif arılar, kolonilerindeki nektar konusunda tecrübesiz arıları çeşitli kaynakları toplamak adına verimli bir şekilde konumlandırır.

Tapkan mühendislik çalışmalarında ele alınan arı davranışlarının en önemlilerinin yiyecek arama ve çiftleşme olduğunu belirtmiştir. Bu davranışları ayrıntısı ile aşağıda anlatılmıştır [47].

Yiyecek Arama Davranışı:

Gözcü olarak alana dağılan arılar yiyecek bulmak için kovandan ayrılarak geniş alanları tararlar. Görevleri yeni bir yiyecek kaynağı bulmak ve bunu kolonideki diğer arılara haber vermektir. Alandan toplanan bilgilerle bu arılar kovanlarına dönerler. Arılar duymadıkları için besin kaynağının yerini, koloninin diğer üyelerine dans ederek tarif ederler. Bu dans yukarda da belirttiğimiz üzere içinde pek çok bilgiyi taşır. Tapkan arıların dans içeriği ile ilgili şunları belirtmiştir [47];

“Arının genel davranışlarından yiyecek kaynağı ile ilgili tüm bilgiler elde edilebilir. Örneğin nektar toplamış bir arı kovana döndüğünde sadece nektarı boşaltıp geri uçarsa bu, arının faydalandığı kaynak bilinen bir kaynaktır veya verimsizdir anlamına gelmektedir.”

Besin kaynağında toplanan arı ile kovanda dans eden gözcü arıların sayısı doğru orantılıdır. Ayrıca arılar buldukları ve yararlandıkları besin kaynaklarından üzerlerine koku bulaştırırlar. Ayrıca kaynaktan aldıkları nektarı diğer arılara tattırırlar. Bu bulaştırılan koku sayesinde diğer arılar bu kaynağın kullanılıp kullanılmadığını, tattıkları nektar örneğiyle de besinin çeşidini anlarlar ve ona göre enerjilerini ve zamanlarını bu kaynağa yoğunlaştırırlar ya da uzaklaşırlar.

Bu konu ile ilgi yapılan birkaç çalışma şöyledir:

Yonezawa ve Kikuchi arıların yiyecek arama davranışını incelemişlerdir. Grup zekâsına dayanan sistemlerin önemini gösteren bir algoritma geliştirmişler ve bir benzetim gerçekleştirmişlerdir [48].

Schmickl ve arkadaşları ise çok gözlemcili bir benzetim alanında bal arılarının yiyecek arama davranışının sağlamlığını incelemişlerdir [49].

Çiftleşme Davranışı:

Çiftleşme davranışında baş aktör olarak kraliçe arı vardır. Kraliçe erkek arıların toplandığı alanlara doğru uçar ve erkek arıların kendilerini bulmalarını sağlayan bir tür salgı yayar. Çiftleşme uçuşu adı verilen bu uçuş sırasında erkek arıların kraliçeyi fark etmeleri ile çiftleşme gerçekleşir. Bu konuda Adams ve arkadaşları, Laidlaw ve Page, Rinderer ve Collins çalışmalar yapmışlardır [50-52].

Ayrıca Özbakır ve Tapkan çalışmalarında arı sisteminin iki asli temel üzerine kurulduğunu belirtmişlerdir [39]:

Besin Kaynakları: Besin kaynağının değeri kovana olan yakınlık, enerji zenginliği ve bu enerjinin çıkarılma kolaylığı gibi pek çok faktöre bağlıdır.

Arayıcılar: Bu gruptaki arılar ikiye ayrılarak incelenmiştir.

- *Görevsiz arayıcılar:* Eğer bir arı hakkında arama alanındaki besin kaynakları adına herhangi bir bilgisi olmadığı varsayılırsa bu arı görevsiz arayıcı olarak aramaya başlar. Görevsiz arı için iki adet olasılık söz konusudur:
 (1) *Kaşif (öncü) arı:* Eğer bir arı herhangi bir ön bilgi olmadan rastgele bir arama yapıyorsa bu kaşif arıdır. Yapılan araştırmalara göre kovandaki kaşif arı oranı %5 ile %30 arasında değişmektedir. (2) *İzci (acemi) arı:* Kâşif arıların sallanım dansına katılıp bilgilendikten sonra arama faaliyetine katılan arılardır.
- *Görevli arayıcılar:* Kâşif arıların bilgilendirmesi sonucu besin kaynağı alanına çıkarak kaynak arayan izci (acemi) diğer bir adıyla gözcü arılar kaynağı bulduktan sonra görevli arayıcılar kadrosuna katılır. Besin kaynağı ile ilgili bilgileri hafızasına alarak kovana döner. Yüklediği nektarı kovandaki besin alanına boşaltır. Toplayıcı (arayıcı) arı için nektar kalıntı miktarı ile ilgili olası üç seçenek vardır. Bunlar:

1) Eğer nektar miktarı çok düşük seviyeye kadar iner ya da tükenirse arayıcı (toplayıcı) arı besin kaynağından vazgeçer ve görevsiz arı haline döner.

2) Eğer besin kaynağında yeterli miktarda nektar varsa, arayıcı (toplayıcı) arı besin kaynağı bilgilerini kovandaki arılarla paylaşmadan aramaya devam edebilir.

3) Ya da dans alanına giderek aynı besin kaynağı için kovandaki arıları bilgilendirmek adına salanlım dansını yapabilir. Bu seçenekler için olasılık değerleri yüksek besin kaynağı kalitesi ile ilgilidir. Kâşif arılarca tespit edilen zengin besin kaynakları hakkındaki bu bilgiler, dans alanında izci arılara sunulmakta ve izci arılar en kârlı kaynağı seçmektedirler.

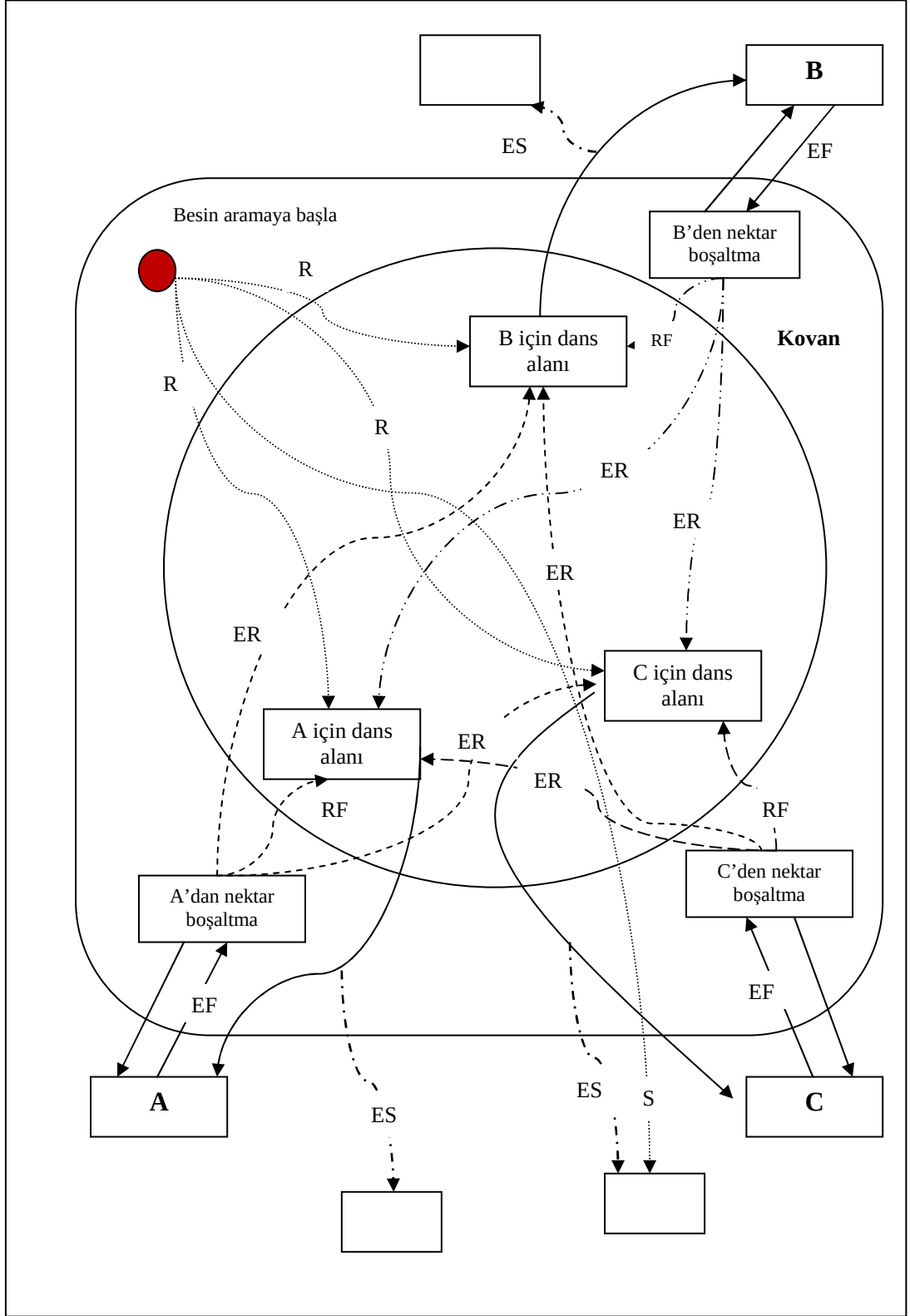
Tecrübeli arayıcılar: Bu tip arayıcılar besin kaynağının yeri ve kalitesi için geçmiş hafızalarını kullanırlar. Önceden keşfedilen besin kaynaklarının son durumunu denetleyen bir denetçi de olabilirler. Sallanım dansındaki bilgileri kullanarak yeniden arayıcı da olabilirler. Bunlar başka arılarca da kalitesi onaylanmış aynı besin kaynaklarını da tespit etmeye çalışıyorlar. Ya da bunlar tüm besin kaynakları tükenince yeni alanlar bulan izci (gözcü) arı da olabilirler. Bu arılar, başka görevli arılarca dans alanında anlatılan yeni besin kaynağı için izci arı da olabilirler.

Pozitif geri besleme: Yiyecek kaynaklarındaki nektar miktarı arttıkça o yiyecek kaynaklarını ziyaret eden izci arı sayısı da artar.

Negatif geri besleme: Terk edilen yiyecek kaynaklarındaki araştırma işlemi sonlandırılır.

Sürekli değişim: Kâşif arılar yeni yiyecek kaynakları keşfetmek için rastgele arama yaparlar.

Çoklu etkileşim: Arılar yiyecek kaynağının pozisyonu hakkında edindikleri bilgileri dans alanında kovandaki diğer arılarla paylaşırlar. Karaboğa çalışmasında arı sisteminin kendi kendine örgütlenmesini böyle açıklamıştır [56].



Şekil 5.1. Arıların besin arama davranışı [57]

Besin toplayan arıların temel davranış özellikleri Lucic'in çalışmasında Şekil 5.1'deki gibi gösterilmiştir [57]. Bu çalışmasında yukarıdaki şekli şu şekilde açıklamaktadır.

Görevsiz arı ilk başta kovan etrafındaki besin kaynakları hakkında herhangi bir bilgisi olmadan besin aramaya başlar. Bu tip bir arı için iki seçenek vardır:

- *Kâşif (öncü) arılar*: Tamamen içgüdüsel olarak herhangi bir ön bilgi olmaksızın, kovan etrafında kendiliğinden besin aramaya başlayan arılardır (S).
- *İzci (gözcü) arılar*: Kâşif arılar tarafından yapılan sallanım dansını seyrederek besin kaynakları hakkında bilgi edinen ve daha sonra bu bilgiye göre hareket ederek besin kaynaklarına ulaşan arılardır (R).

Görevsiz arı besin kaynağını bulduktan sonra besin kaynağının yerini hafızasında tutarak o kaynağı tüketmeye başlar. Dolayısıyla görevli olmayan arı, bir görevli arı haline gelir. Besin kaynağından bir miktar nektar alarak kovana dönüp besin depolama alanına nektarı boşaltan bir arı için 4 alternatif vardır ve bu alternatiflerin gerçekleşme olasılığı yüksek oranda besin kaynağının kalitesine bağlıdır [47].

- *ES*: Nektar miktarı düşük seviyelere inmişse ya da tükenmişse, arı bu besin kaynağını terk eder, ardından görevli olmayan bir arıya dönüşür.
- *RF*: Aynı besin kaynağına dönmeden önce dans ederek kovandaki diğer arıları bilgilendirir.
- *EF*: Eğer besin kaynağında yeterli miktarda nektar varsa, bilgi paylaşımında bulunmadan besin kaynağını tüketmeye devam edebilir.
- *ER*: Tüketmekte olduğu besin kaynağının kalitesinden tatmin olmayan arı, dans alanında önerilen yeni bir besin kaynağını aramaya başlar.

5.2. Arı Kolonisi Algoritması Uygulamaları

Arı kolonisi optimizasyonu meta sezgiseli sürü zekâsı tekniğine ait bir yaklaşımdır. 1999 ile 2002 arasında Lucic ve Teodorovic arı kolonisi algoritmasının temeline giriş teşkil eden çalışmalar yapmışlardır [57, 60-62]. Arı kolonisi algoritması farklı kombinatoriyal optimizasyon problemlerinin çözümünü bulmak için uygulanan doğanın içerisinde uyarlanan bir meta sezgiseldir [59]. Yapay arılar sorunlara yardımlaşarak çözüm bulan karakterlerdir. Kolonide bulunan her yapay arı problem için bir çözüm üreticidir. Bu algoritma iki aşamadan (faz) oluşur. Bunlar:

- İleri geçiş (forward pass)
- Geri geçiş (backward pass)

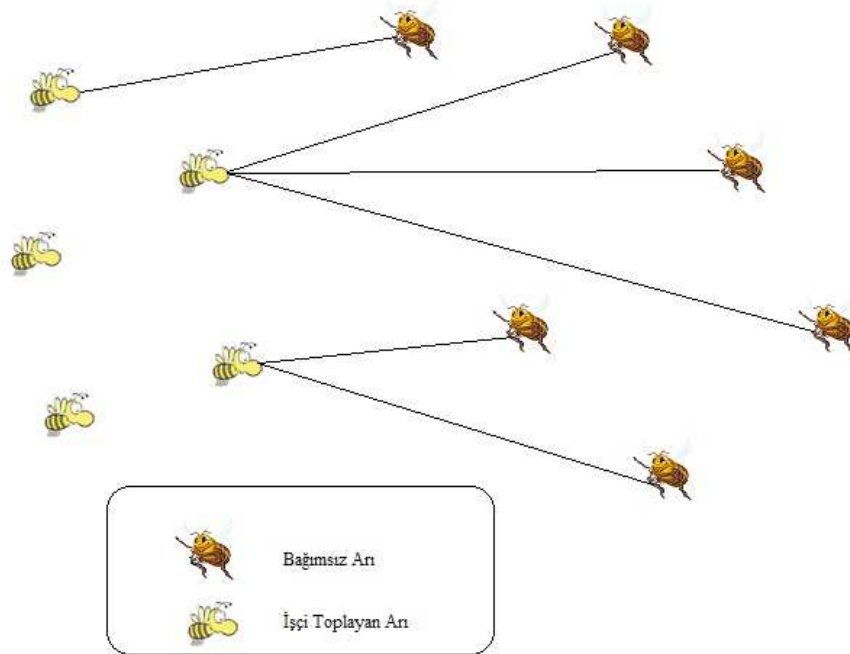
Her *ileri geçiş* (forward pass) fazında, tüm yapay arılar bir çözüm uzayı ararlar. Bunlar çözümü geliştiren önceden tanımlanmış olan yolları uygulayarak yeni çözümler geliştirmeye çalışırlar. Yani her ileri geçiş esnasında arı çözüm bileşenlerinden birini ziyaret ederek kısmi çözümü oluşturur ve kovana tekrar döner. Burada arılar çözüm hakkındaki var olan bilgilerini değiştirirler. Kısmi çözümler ileri geçiş safhasında arılar tarafından oluşturulduktan sonra ikinci faz olan geri geçiş safhası başlar. Bu fazda arılar kendi kısmi çözümlerinin kalitesi hakkındaki bilgiyi paylaşırlar [59].

Teodorovic ve arkadaşları yaptıkları çalışmada ileri geçiş ve geri geçiş safhalarını ayrıntısı ile anlatmışlardır [59]. Arama işleminin başlangıcında kolonide bulunan her arı analist olarak belirlenmiş arı ilk ileri geçiş esnasında çözüm bileşenlerinden (*NC*) birini ziyaret ederek kısmi çözümü oluşturur ve kovana tekrar döner. Mevcut arı kolonisi algoritmasına göre kovanın kesin olmayan konumu ve algoritmanın uygulanmasını etkilememesi nedenleriyle kovan doğal olmayan bir nesnedir. Kovan, Teodorovic ve arkadaşlarına göre sadece arıların besin aramadaki mevcut durum hakkında bilgi değişimini sağladıkları senkronizasyon noktası olarak belirlenmiştir [59].

İyi sonuca sahip arılar kendi çözümlerinde kalma ve duyurma konusunda daha şanslıdır. Kendi kısmi çözümüne *sadık* yapay arılar aynı zamanda çözümlerine işçi toplayabilen ve kısmi çözümlerini tanıtan arılardır. Eğer arı tarafından bir çözüm terk edilirse o arı artık bağımsız olur ve tanıtılan bir çözümü seçmek zorundadır. Bu karar da bir olasılık ile alınır ve en iyi tanıtılan çözümün seçilme şansı diğerlerine göre daha yüksektir. Geri geçiş safhasında tüm arılar iki gruba ayrılır. Bunlar;

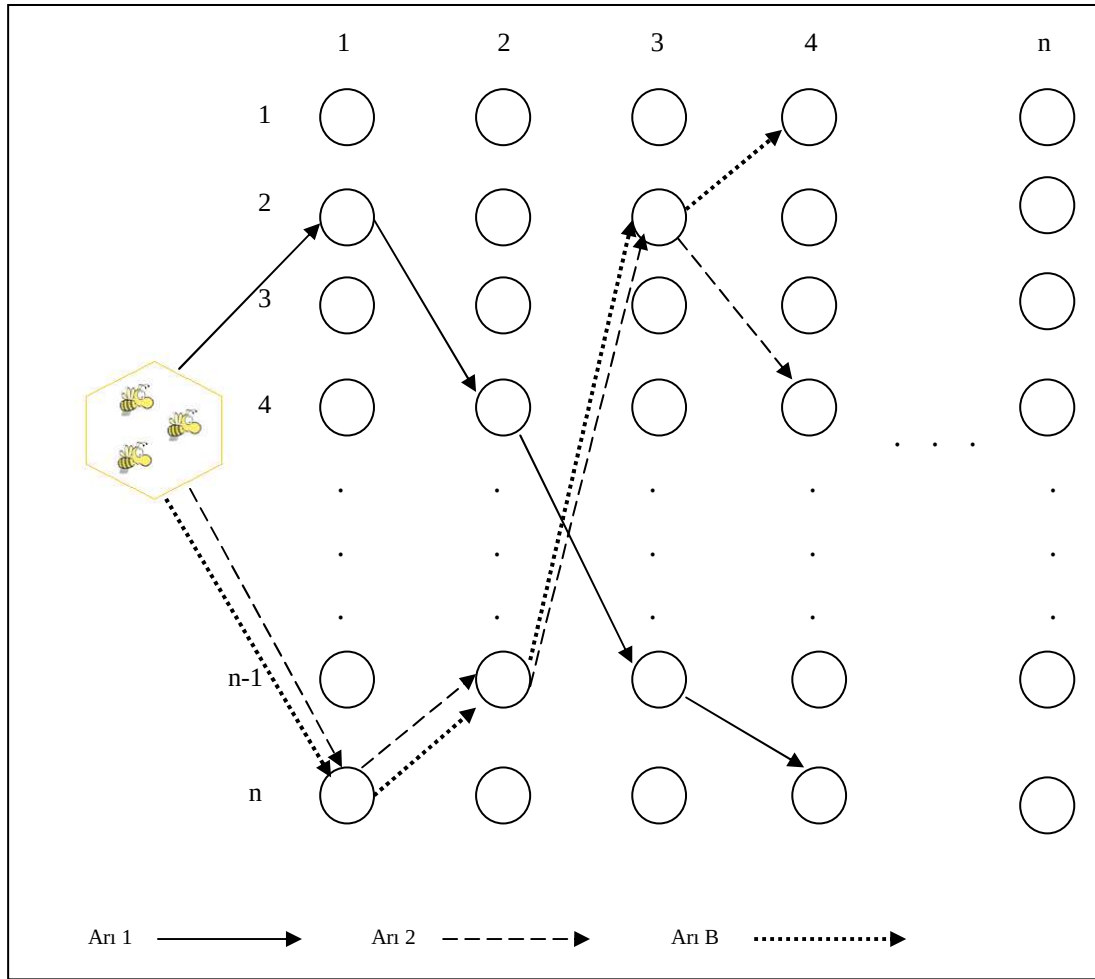
- R : işçi toplayan arılar
- $B-R$: bağımsız arılar

R ve $B-R$ değerleri bir geri geçişten diğerine değişiklik gösterebilir. Her bağımsız arı (gözcü) kendisine bir kâşif arının çözümünü bu aşamada seçmek zorundadır. Geri geçişteki sınıflandırmayı Teodorovic ve arkadaşları aşağıdaki şekille ifade etmişlerdir [59].



Şekil 5.3. Bağımsız arılara görev verme [59]

Bağımsız arılar (gözcü), kâşif arıların kısmî çözümlerinden bir tanesini seçtikten sonra beraber olarak kâşif arının çözüm noktasına kadar uçuşları devam eder. Bu noktadan sonra arılar bağımsızdır ve kendi kararını vererek yeni çözüm kümesi oluştururlar. Bu duruma yine aynı çalışmada Teodorovic ve arkadaşları şu şekilde örnek vermiştir [59].



Şekil 5.4. Arıların dördüncü ileri geçişinden sonraki kısmi çözümleri [59]

Yukarıdaki şekilde de görüleceği üzere Arı 2 kendi çözümünü terk etmiştir ve Arı B tarafından hâlihazırda üretilen yolda beraber gitmiştir. Yani bu da kısmi çözümün Arı B tarafından oluşturulduğu ve Arı 2'nin bunu kopyaladığı anlamına gelmektedir. Bu kısmi çözümün sonuna gelince tüm arılar bağımsız hareket ediyor. Neticede dördüncü ileri geçiş safhasında Arı 2 Kaynak 4'ü, Arı B ise Kaynak 1'i tercih ediyor.

Arı 1 ise başlangıçtaki bulduğu yolu değiştirmeyerek dördüncü ileri geçişte çözümünü geliştiriyor.

Arama algoritmasının ileri ve geri geçiş safhası her arının sadece bir adet oluşturabileceği gerekli tüm mümkün çözümler üretilinceye kadar devam eder. Tüm çözümler oluşturulduktan sonra en iyi çözüm seçilir ve global en iyi olan çözüm iterasyon sonundaki en iyi çözüm ile karşılaştırılarak güncellenir. Bu sayede arı kolonisi optimizasyon algoritmasının bir çevrimi tamamlanmış olur. Bundan sonra tüm arılara ait çözümler silinir ve yeni iterasyon başlamış olur. Arı kolonisi algoritması durma şartları (maksimum iterasyon sayısı, CPU zamanı vb.) sağlanana kadar tek tek iterasyonlar sürdürülerek devam eder. En son olarak en iyi sonuç (global en iyi) elde edilir. Olasılıkla ilişkili kararlar, sadakat kararı ve iş verme, genellikle rulet tekerleği prensibi ile çözülmektedir.

Öncelik değerlerinin ayarlanması gereken algoritma parametreleri şu şekildedir [59]:

B: Kovandaki arı sayısı, *NC*: Bir ileri geçiş esnasında geliştirici hareket sayısı (çözüm bileşenleri- uzaydaki çözümlerden ele alınacak düğüm sayısı)

Aramanın en başında tüm arılar kovandadır. Teodorovic ve arkadaşları algoritmanın taslak kodunu şu şekilde belirtmişlerdir [59]:

- 1) Başlat: tüm arılara boş birer başlangıç çözümü ata;
- 2) Tüm arılar için: // (İleri Geçiş)
 - a) $k = 1$; // (ileri geçişteki geliştirici hareketleri say)
 - b) Tüm mümkün düğümleri değerlendir;
 - c) Rulet tekerleği prensibi kullanarak bir düğümü seç;
 - d) $k = k + 1$; Eğer $k \leq NC$ adım (b)'ye git;
- 3) Tüm arıları kovanda topla.(Geri geçiş);
- 4) Her arının (kısmi) amaç fonksiyon değerini hesapla;

- 5) Tüm arılar, kendi keşfinde devam edip (sadık olup) ve görev veren arı mı olacağına ya da izci (gözcü) arı mı olacağına rastgele (randomly) karar verirler;
- 6) Tüm izci arılar için işveren arılardan rulet tekerleği prensibine göre yeni bir çözüm seç;
- 7) Eğer çözüm tamamlanmadıysa Adım 2'ye git;
- 8) Tüm çözümleri değerlendir ve en iyi çözümü bul;
- 9) Eğer durma kriteri sağlanmadıysa Adım 2'ye git;
- 10) Bulunan en iyi çözümü yaz.

Kâşif arıların çözümlerine göstermiş oldukları sadakat çözümün kalitesinin de belirleyici bir özelliğidir. Sadakat testi neticesinde kâşif arı;

- Ya kendi çözümlerini geliştirmeye devam eder
- Ya da bir gözcü arı olarak yeni bir çözüm almak zorunda kalır.

Adım (2) ve (4) probleme bağlıdır ve her biri arı kolonisi algoritması uygulaması ile çözümlenir. Adım (5) sadakat kararı ve Adım (6) görev verme işlemi gibi özellikli adımlar aşağıda belirtilmiştir.

Sadakat kararı:

B'inci arının (yeni ileri geçişin başlangıcında) daha önceki oluşturduğu kısmi çözüme sadakatinin olasılığı Teodorovic ve arkadaşları tarafından şu şekilde ifade edilmiştir [59]:

$$p_b^{u+1} = e^{-\frac{O_{\max} - O_b}{u}}, \quad b = 1, 2, \dots, B \quad (5.1)$$

O_b : b.-arı tarafından oluşturulmuş kısmi çözümün amaç fonksiyon değerinin normalize edilmiş hali,

O_{max} : Karşılaştırılmış olan kısmi çözümlerin normalize edilmiş değerlerinin maksimumu,

u : İleri geçişin sıra sayısı (örnek: $u=1$ ilk ileri geçişte, $u=2$ ikinci ileri geçişte vb ...)

Eş. 5.1'e göre daha iyi bir şekilde üretilen kısmi çözüm ya da amaç fonksiyon değeri (daha yüksek O_b değeri), arı b 'nin çözüme sadık olması hususunda daha yüksek olasılığın oluşmasını sağlar. İleri geçiş sıra sayısının artması yani diğer bir ifadeyle " u " teriminin artması olasılık değerini arttıracığından arıların kendi kısmi çözümlerine olan bağlılığını da yükseltecektir. Diğer bir ifadeyle arılar araştırma işleminin başında çözüm uzayının bulunmasında daha cesur olacaklardır. Daha fazla ileri geçiş yaptıklarında, arılar çözüm uzayını bulmada daha az cesarete sahip olacaklardır. Arama işleminin sonunda arılar halihazırda bilinen kısmi çözüme odaklanırlar yani kısacası ilerleyen aşamalarda kendi kısmi çözümlerine olan sadakatları fazla olacağından kendi çözümlerini geliştirmeye çalışacaklardır.

İş verme süreci:

Tüm bağımsız (kararsız) arılar için belli bir olasılıkla hangi işveren arıyı izleyeceklerine ilişkin eşitlik aşağıdadır. b -gözcü arı tarafından oluşturulan kısmi çözümün herhangi bir bağımsız (kararsız) arı tarafından seçilme olasılığı (p_b) Eş. 5.2 ile ifade edilmiştir:

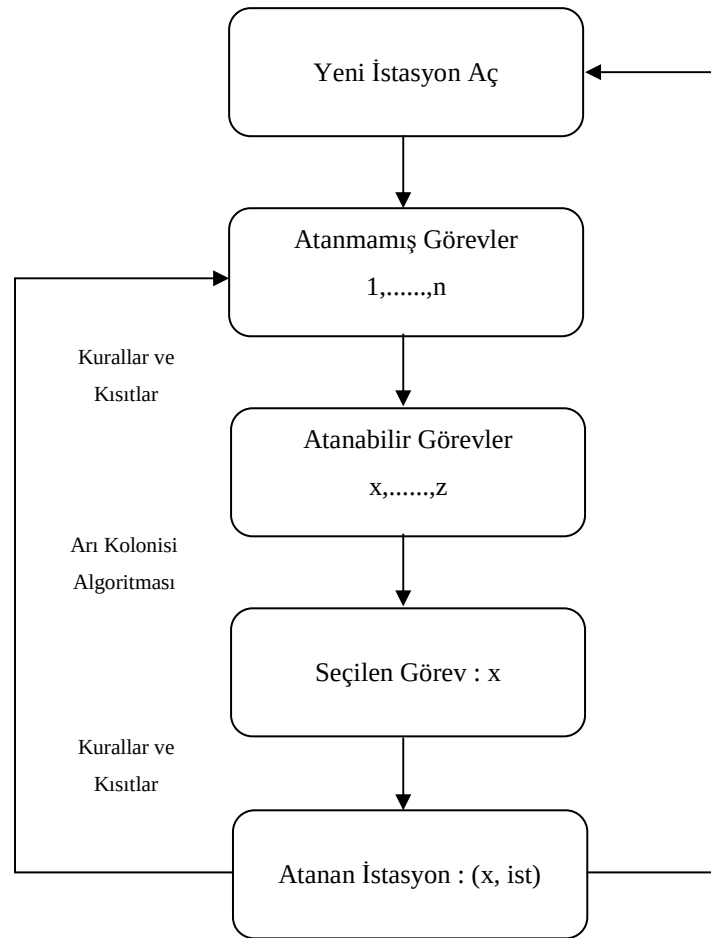
$$p_b = \frac{O_b}{\sum_{k=1}^R O_k} \quad , \quad b = 1, 2, \dots, R \quad (5.2)$$

O_k bu formülde k .(arının) tanıtılmış kısmi çözümünün amaç fonksiyon değerini göstermektedir. R ise kovanda bulunan işveren arı sayısını sembolize etmektedir. Eş. 5.2 ve rastgele üretilen bir sayı kullanılarak her bağımsız (kararsız) izci arı bir işveren arıya bağlanır.

5.3. Önerilen Arı Kolonisi Algoritması

Özbakır ve Tapkan yaptıkları çalışmada arı kolonisi algoritmasını tek modellenli üretimin yapıldığı çift taraflı montaj hatları için modifiye etmişlerdir [39].

Yapılan bu çalışma ile karışık modellenli üretimin yapıldığı çift taraflı montaj hatları için algoritma modifiye edilecektir. Problemin çözüm adımları akış diyagramında özetlenmiştir:



Şekil 5.5. Problemin çözüm adımları

Özbakır ve Tapkan'ın yapmış oldukları çalışmalarında alan kısıtlı tek modelli çift taraflı montaj hattının dengelenmesine yönelik arı kolonisi algoritmasıyla çözüm geliştirmişlerdir [39]. Bu tez çalışmasında ise amacımız Özcan ve Toklu tavlama benzetimi algoritması ile çözüm buldukları karışık modelli çift taraflı montaj hattı problemlerini önerdiğimiz arı kolonisi algoritması ile çözmek ve sonuçlarını ilgili çalışma ile karşılaştırmaktır [9]. Çözümü yapılacak olan karışık modelli çift taraflı montaj hattı dengeleme problemlerindeki varsayımlar şu şekildedir:

Problemin varsayımları

- Benzer üretim karakteristiklerine sahip ürün modelleri aynı çift taraflı montaj hattında üretilirler.
- Operatörler görevleri hattın her iki tarafında paralel olarak icra ederler.
- Bazı görevler hattın her iki tarafında icra edilebilirken, bazıları sadece tek tarafında icra edilebilir.
- Farklı modellerin öncelik diyagramları bilinmektedir. Macaskill'in kombine öncelik diyagramı kavramı kullanılmıştır [43].
- Görev zamanları deterministik ve istasyon atamalarından bağımsızdır.
- Farklı modeller arasında ortak görevler bulunmaktadır. Bir görevin tamamlanma süresi bir modelden diğerine farklılık gösterebilir ve aynı zamanda sıfıra da eşit olabilir.
- Operatörlerin seyahat süreleri dikkate alınmamıştır.
- Paralel görevlere, paralel istasyonlara ve ara stoklara izin verilmemiştir [9].

Pham ve arkadaşlarının algoritmada kullanmış olduğu ve bu çalışmada da kullanılacak parametrelerden bazıları şöyledir [63]:

- Kâşif arı sayısı: n
- n adet ziyaret edilen bölgeden seçilen bölge sayısı: m
- Seçilen m bölge içindeki en iyi bölge sayısı: e
- En iyi e bölgeye gönderilen izci (gözcü) arı sayısı: nep

- Seçilen diğer bölgelere ($m-e$) gönderilen izci arı sayısı: nsp
- Komşuluk arama boyutu: ngh
- ve son olarak da *durdurma* kriterini içermektedir.

Algoritma n adet kâşif arının arama uzayında rassal olarak yerlerini almasıyla başlamaktadır. Algoritmanın temel adımları aşağıdadır:

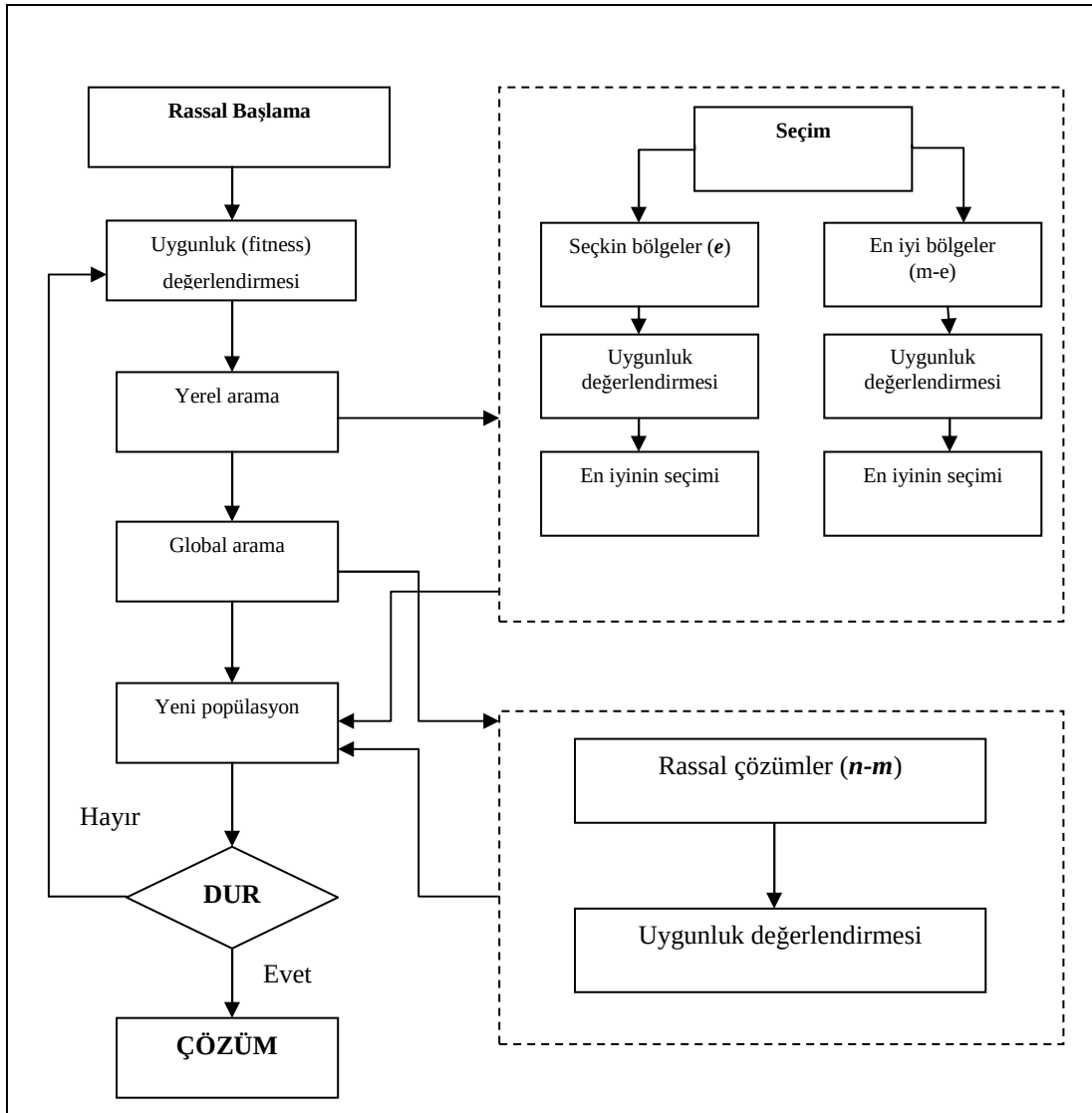
1. Başlat: rastgele çözümlerle başlangıç arı popülasyonunu oluştur;
2. Değerlendir: popülasyonun uygunluğunu denetle;
3. *While*: durdurma kriterine rastlanmadıysa // yeni arı popülasyonu oluştur;
4. Seç: seçkin arıları belirle;
5. Seç: komşuluk aramaları için bölgeler seç;
6. İş ver: seçilen bölgeler etrafında arılara iş ver ve uygunluğu denetle;
7. Seç: her bir bölgedeki en iyi uygunluk değerine sahip arıyı seç;
8. Ata: kalan arıları rastgele arama için ata ve uygunluklarını değerlendir
9. *End While*: en iyi çözümü bulunca bitir. Bulunan en iyi çözümü yaz.

Algoritmanın açıklaması şu şekilde yapılmıştır [39]. Problemden yer alan kâşif arılarca ziyaret edilen noktaların uygunlukları (fitness) algoritmanın 2. adımında değerlendirilir. Algoritmanın 4. adımında en yüksek uygunluk değerine sahip arılar seçkin arılar olarak belirlenir ve bu bölgeler (m) komşuluk araması için seçilir.

5. ve 6. adımlarda algoritma seçilen en iyi e bölge için daha fazla arının komşuluk ilişkisini sağlama adına arama işlemi yürütür. Bunun amacı bu bölgelere diğer bölgelere ($m-e$) nispeten daha fazla arı gönderilerek daha detaylı arama yapmaktır. Yeni popülasyonun oluşturulması için her bölgedeki en iyi uygunluk değerine sahip arı algoritmanın 7. adımında seçilir. 7. adıma göre her bölgede sadece bir arı en yüksek uygunluğa sahip olmaktadır.

Algoritmanın 8. adımında ise popülasyondaki diğer arılar ($n-m$) yeni potansiyel çözümler elde etmek için rastgele olarak araştırma uzayına atanırlar. Bu adımlar

durdurma kriterine ulaşıncaya kadar devam eder. Tüm iterasyonların sonucunda yeni popülasyon, seçilen her bölgenin temsilcileri ve rastgele arama yapan kâşif arılar olmak üzere iki parçadan oluşacaktır.



Şekil 5.6. Arı Algoritması akış diyagramı [64]

Çift taraflı montaj hattı probleminin çözümü için modifiye edilen algorithmada kullanılan parametreler şöyle açıklanmışlardır:

S : Kaşif arı sayısı ($s = 1, \dots, S$)

P : Görevlendirilmiş (gözcü) arı sayısı ($p = 1, \dots, P$)

e : En iyi görevlendirilmiş (gözcü) arı sayısı

nep : Tüm e görevlendirilmiş arılar için izci (onlooker) arı sayısı

nsp : Tüm $P-e$ görevlendirilmiş arılar ($nsp < nep$) için izci (onlooker) arı sayısı

$MaxIter$: İterasyon limiti (durdurma kriteri)

σ^p : p 'inci görevlendirilmiş arının çözümü

σ^{ls} : Yerel arama ile bulunan komşu çözüm, $ls = \{kaydırma, değiştirme\}$

σ^{best} : En iyi çözüm

$fit(\sigma^p)$: p 'inci görevlendirilmiş arının uygunluk fonksiyonu değeri.

I : Görev kümesi

Modifiye edilen arı algoritması (S , P , e , nep , nsp , $MaxIter$) parametrelerinin başlatılması ve kâşif arılar tarafından S adet başlangıç sonucu üretilmesiyle devam eder.

Başlangıç çözümleri sezgisel yöntemlerle elde edilir. Üretilen çözümler kümesindeki P adet iyi çözüm görevlendirilmiş arılar kadar hesaplanır. İyi çözümlerdeki P sayısından, çözümlerdeki e sayısı en iyi sonuç olarak seçilir. Daha detaylı bir komşuluk araması için bu en iyi çözümlere nep adet izci (onlooker) arı gönderilir.

Geriye kalan $P-e$ sayıda çözüme ise daha az sayıda izci (onlooker) arı gönderilir. Komşuluk mekanizmalarından (kaydırma ve değiştirme) birisi bölgesel aramalar için her izci arıya %50 olasılıkla uygulanır. En iyi izci arı orijinal görevlendirilmiş arı ile

karşılaştırılır, eğer izci arı daha iyi ise görevlendirilmiş arı bu yeni çözüm ile geliştirilir. Görevlendirilmiş arı en iyi sonuçla karşılaştırılır, eğer bu arının çözümü daha iyi ise en iyi sonuç bu çözümle geliştirilir. Global arama için, $S-P$ adet kâşif arı çözümü sezgisel metotlar kullanılarak üretilir [39].

Çift taraflı montaj hattına uyarlanan algoritmanın ayrıntılı adımları şu şekilde belirtilmiştir:

1. Parametreleri başlangıç durumuna getir
2. Sezgisel yöntemlerle kâşif arıları başlat
3. Kâşif arıların uygunluk fonksiyonunu değerlendir

$$\min f(\sigma^s) = \left(\frac{\sum_{m \in M} q_m \left(\sum_{j \in J} \sum_{k=1,2} ({}_m WL_j^k - WL_{\max})^2 \right)}{NS} \right) \quad (5.3)$$

4. $I = 0$

5. Do

Artan şekilde sırala $s=1 \dots S(\sigma^S)$ ve en iyi P adet çözümü görevli arı olarak belirle

En iyi e adet görevli arıyı seç

En iyi e adet görevli arının her birine nep adet izci arı ata

Kalan $P-e$ adet görevli arının her birine nsp adet izci arı ata

$t=0$

Do $k=0$

Do

e adet görevli arıya atanan izci arıya %50 olasılıkla komşuluk yapısından birini uygula

If $f(\sigma^{kaydırma}) < f(\sigma^p)$ then $\sigma^p = \sigma^{kaydırma}$

If $f(\sigma^{değiştirme}) < f(\sigma^p)$ then $\sigma^p = \sigma^{değiştirme}$

$k=k+1$

While $(k < nep)$ $k = 0$

Do

P -e adet görevli arıya atanan izci arıya %50 olasılıkla komşuluk yapısından birini uygula

If $f(\sigma^{kaydırma}) < f(\sigma^p)$ then $\sigma^p = \sigma^{kaydırma}$

If $f(\sigma^{değiştirme}) < f(\sigma^p)$ then $\sigma^p = \sigma^{değiştirme}$

$k=k+1$

While ($k < nsp$) $t=t+1$

While ($t < p$)

En iyi sonucu geliştir

If $\min_{p=1,\dots,p} f(\sigma^p) \leq f(\sigma^{eniyi})$ then $\sigma^{eniyi} = \sigma^p$

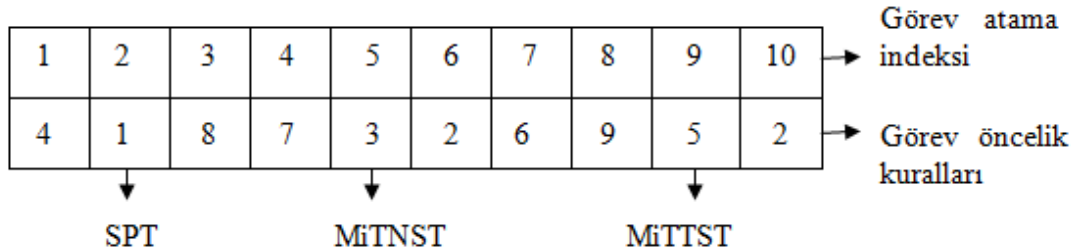
S-P kâşif arıyı sezgisel yöntemlerle başlat

While ($I < MaxIter$)

İlgili problemlerin çözümü için önerilen arı algoritmasında çözüm dizilerinin gösterim şekli olarak sezgisel tabanlı gösterim kullanılmıştır. Bu gösterim şeklinde çözümler dolaylı bir yolla temsil edilmekte yani çözüm dizisinin her değeri bir sezgisel kuralı ifade etmektedir. Daha açık bir ifade ile kodlama için, çözümlerin dolaylı bir şekilde temsil edildiği sezgisel tabanlı gösterimler kullanılmaktadır. İlk olarak görevlerin öncelik değerleri kodlanır ve bu kurallar problemlere çözüm üretmek için uygulanır. Sezgisel tabanlı gösterimlerde her çözüm dizisi bir sezgisel kural ile temsil edilir. Bunun anlamı şudur; i 'nci çözüm dizisinin değeri i 'nci görevin atanmasında kullanılacak sezgisel kuralı gösterir [39].

- 1) En kısa işlem süresi (SPT): En kısa işlem süresine sahip işi seçer.
- 2) En uzun işlem süresi (LPT): En çok işlem süresine sahip işi seçer.
- 3) Kendini takip eden iş sayısı toplamının en küçüğü (MiTNST): Kendini takip eden iş sayısı en az olan işi seçer.
- 4) Kendini takip eden iş sayısı toplamının en büyüğü (MaTNST): Kendini takip eden iş sayısı en çok olan işi seçer.
- 5) Kendini takip eden işlerin toplam işlem süresinin en küçüğü (MiTTST): Kendinden sonra gelen işlerin işlem süresi toplamı en az olan işi seçer.
- 6) Kendini takip eden işlerin toplam işlem süresinin en büyüğü (MaTTST): Kendinden sonra gelen işlerin işlem süresi toplamı en çok olan işi seçer.

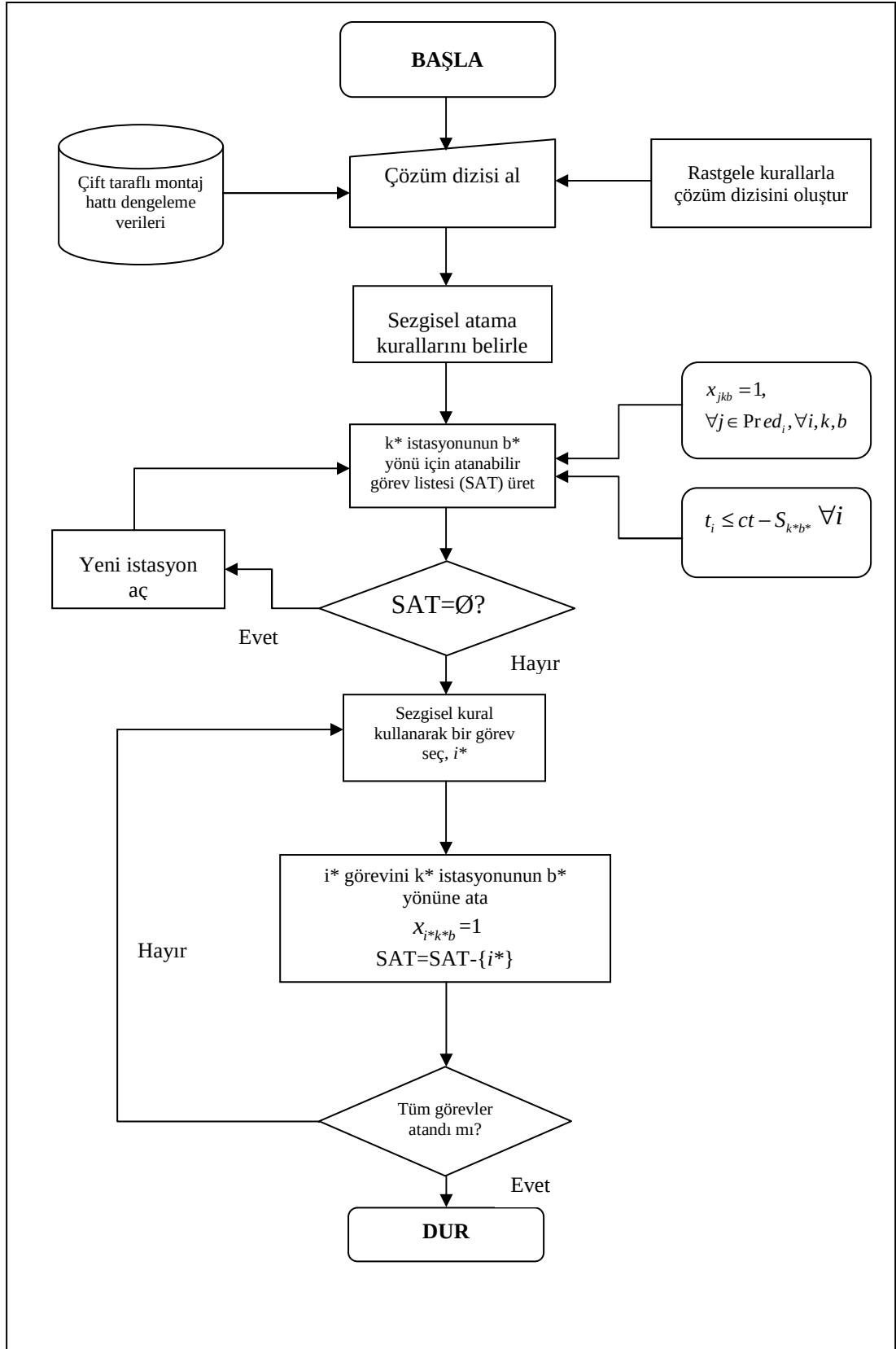
- 7) Maksimum sıralı konumsal ağırlık (MaRPW): Sıralı konumsal ağırlık değeri en büyük olan işi seçer. Bir işe ait sıralı konumsal ağırlık değeri ise o işin işlem süresi ile kendinden sonra gelen işlerin işlem sürelerinin toplamı ile elde edilir.
- 8) Maksimum ortalama sıralı konumsal ağırlık (MaARPW): Ortalama sıralı konumsal ağırlık değeri en büyük olan işi seçer. Bir işe ait ortalama sıralı konumsal ağırlık değeri ise o işin işlem süresi ile kendinden sonra gelen işlerin işlem sürelerinin toplamının iş sayısına bölünmesi ile elde edilir.
- 9) Alan önceliği (PZ): Pozitif bölgesel kısıta sahip bir işin kendinden önceki işini seçer.



Şekil 5.7. Çözüm dizisi gösterimi

Şekil 5.7.'de 10 görev içeren bir veri kümesinin ilgili çözüm dizisi örneğini göstermektedir. Örnek gösterime göre seçilen bir görev 2.dereceye atanırsa SPT (En kısa işlem süresi) sezgiseli kullanılır. Eğer seçilen bir görev 5.dereceye atanırsa MiTNST (Kendini takip eden iş sayısı toplamının en küçüğü) sezgiseli kullanılır. Eğer seçilen bir görev 9.dereceye atanırsa MiTTST (Kendini takip eden işlerin toplam işlem süresinin en küçüğü) sezgiseli kullanılır. Son sezgisel kuralı da kullanarak algoritma kısıtlandırılmış görevlerden önceliği olan görevlerin önceliklerini verir [39]. Çift taraflı montaj hattı problemlerinde herhangi bir özel kısıta sahip olmayan çözüm dizilerinin oluşturulması esnasında (1)-(8) kuralları kullanılır ve çözüm dizisi uzunluğu iş sayısına eşit olur.

Problem için başlangıç çözümlerinin oluşturulması aşaması Şekil 5.8'de açıklanmaktadır.



Şekil 5.8. Başlangıç çözümlerinin oluşturulması

Diyagramda da görüleceği üzere, atama yapılacak istasyonun kapasitesini aşmayan ve varsa önceki işleri bir istasyona atanmış işlerden (*Predi*, *i* işinden hemen önceki işler kümesini göstermek üzere) atanabilir işler listesi öncelikli olarak oluşturulur. Eğer atanabilir işler listesi boş ise yeni bir istasyon açılır, diğer durumda verilen çözüm dizisinde sıradaki kural kullanılarak bir iş seçilir ve ataması gerçekleştirilir. Bir önceki adımda atanan iş, atanabilir işler listesinden çıkarılır. Bu adımlar bütün işler bir istasyona atanana kadar tekrarlanır.

I : görevlerin kümesi

J : iş istasyonlarının kümesi

$Pred_t$: i görevinin hemen ardılarının kümesi

t_i : i görevinin işlem zamanı

S_i : i görevinin görev tarafı, $\{L,R,E\}$

$Idle_{j, s_i}$: S_i tarafındaki j istasyonundaki boş zaman

Uygunluk fonksiyonu:

Uygunluk fonksiyonu tek modellenli çözümlerde, n istasyon sayısı, ct çevrim zamanı ve S_k k 'inci istasyon zamanı olmak üzere istasyon sayısının en küçüklenmesi ve dengelenmiş istasyonları sağlamayı amaçlamıştır [39]

$$f = \sqrt{\frac{\sum_{k=1}^n (ct - S_k)^2}{n}} + \frac{\sum_{k=1}^n (ct - S_k)}{n} \quad (5.4)$$

Uygunluk fonksiyonunun ikinci kısmı çözüm içerisindeki istasyon sayısını en küçüklerken, ilk kısım da aynı istasyon sayısı için çözümler arasındaki en iyi dengeyi bulmayı amaçlar [65].

Bu çalışmada tek modellenli problemlerin çözümünün aksine karışık modellenli çözümlerde WSI' yi (ağırlıklandırılmış düzgünlük indeksi) uygunluk fonksiyonu olarak algoritmaya eklenmiştir.

Komşuluk yapıları:

Yapılan çalışmalarda komşuluk yapısı olarak algoritmanın performansının öne çıkabilmesi adına basit yapılar kullanılmıştır. Bu amaçla bu çalışmada da kaydırma ve değiştirme komşuluk yapıları %50 olasılıkla değişimli olarak kullanılmıştır. Diğer taraftan önceki çalışmalarda olduğu gibi her bir komşuluk aramasında dizi uzunluğunun %10'u kadar kaydırma ya da değiştirme yapılmıştır.

Bu çalışmada iterasyon sayısı durdurma kriteri olarak alınmış ve görev sayısına eşit kabul edilmiştir. Yeni bir komşu çözüm şu şekilde üretilir:

Komşuluk üretme işlemi;

1. Rassal bir değer üret, p_1 ([0,1] değerleri arasında)
2. Eğer $p_1 \leq 0,5$ Adım 4'e git
3. Öncelik listesinden iki farklı pozisyon seç ve içeriklerini değiştir.
4. Öncelik listesinden bir pozisyon seç ve bu seçilen pozisyondaki değeri diğer seçilen değerin soluna ekle.

Fizibil çözüm oluşturma:

Bu bölümde anlatılacak istasyon odaklı işlemler bir fizibil çözüm üretmek için kullanılır. Problemi çözmede kullanacağımız arı algoritmasının her adımında, bu işlem ilk çift istasyonla (NM) başlar (sol taraf (NL) ve sağ taraf istasyonlar (NR)). Diğer çift istasyonlar ardışık olarak kabul edilir.

Atanabilir görevler listesi (SAT) öncelik ilişkilerini sağlayan görevler kümesinden üretilir. SAT içerisindeki en yüksek önceliğe sahip bir görev (en küçük görev rakamı değeriyle) seçilir ve tercih edilen operasyon yönündeki çift istasyona bu görev atanır. Sol ve sağ taraf istasyonları olabildiğince azami şekilde doldurulur, ardından bu çift istasyon kapatılır ve diğer çift istasyona geçilir. Bu işlem tüm görevler atanana kadar devam eder.

ÇMHD-I için işlem basamakları şu şekildedir:

1. Set $NM=1$, $NL=0$, $NR=0$, ${}_mW_1^1 = 0$ ve ${}_mW_1^2 = 0$ tüm $m \in M$.
2. $SAT(SAT=\{i|(tüm\ p \in P(i)\ \text{atanmış\ görevler\ yada}\ P(i)=\{\emptyset\})\ \text{ve}\ i\ \text{görevi\ henüz}\ \text{atanmamış})\}$ Eğer $SAT=\{\emptyset\}$ ise, Adım 6'ya git.
3. SAT içerisindeki görevleri, öncelik listesindeki öncelik değerlerinin artan sırasına göre sırala.
4. SAT içerisindeki ilk görev h' yi şu şekilde ata;

4.1. Eğer görev $h \in A_L$ ise,

4.1.1. Eğer $t_{hm} + {}_mWL_{NM}^1 \leq C$ ve $t_{hm} + t_{rm}^f \leq C (t_{rm}^f = \max\{t_{pm}^f | p \in P(h)\})$ hâlihazırdaki çift istasyonun sağ tarafına atanmış görev) tüm $m \in M$, ardından h görevini sol taraf istasyonuna ata; $TL_{NM}^1 = TL_{NM}^1 + \{h\}$ ve set $t_{hm}^f = \max\{(t_{hm} + {}_mWL_{NM}^1), (t_{hm} + t_{rm}^f)\}$ tüm $m \in M$ için.

Set ${}_mWL_{NM}^1 = t_{hm}^f$ tüm $m \in M$ için ve Adım 2'ye git; aksi halde Adım 5'e git.

4.2. Eğer görev $h \in A_R$ ise,

4.2.1. Eğer $t_{hm} + {}_mWL_{NM}^2 \leq C$ ve $t_{hm} + t_{rm}^f \leq C (t_{rm}^f = \max\{t_{pm}^f | p \in P(h)\})$ hâlihazırdaki çift istasyonun sol tarafına atanmış görev) tüm $m \in M$, ardından h görevini sağ taraf istasyonuna ata; $TL_{NM}^2 = TL_{NM}^2 + \{h\}$ ve set $t_{hm}^f = \max\{(t_{hm} + {}_mWL_{NM}^2), (t_{hm} + t_{rm}^f)\}$ tüm $m \in M$ için.

Set ${}_mWL_{NM}^2 = t_{hm}^f$ tüm $m \in M$ için ve Adım 2'ye git; aksi halde Adım 5'e git.

4.3. Eğer görev $h \in A_E$ ise,

4.3.1. Rassal bir sayı üret, p_2 [0,1] arasında bir rakam. Eğer $p_2 \leq 0.5$ ise Adım 4.1.1.'e git; diğer durumda Adım 4.2.1.'e git.

5. SAT içerisindeki hiçbir görev çift istasyonun her iki tarafına da atanamazsa yeni bir çift istasyon aç. $TL_{NM}^1 \neq \{\emptyset\}$, $NL=NL+1$. Eğer $TL_{NM}^2 \neq \{\emptyset\}$, $NR=NR+1$. Set $NM=NM+1$, ${}_mWL_{NM}^1 = 0$ ve ${}_mWL_{NM}^2 = 0$ tüm $m \in M$, ve Adım 2'ye git.

6.Dur

${}_mWL_{NM}^1$ ve ${}_mWL_{NM}^2$ tüm $m \in M$ için çift istasyonunun sırasıyla sol taraf ve sağ taraf istasyonun kaçınılmaz boş zamanları da içeren istasyon yüklemesini gösterir. TL_{NM}^1 ve TL_{NM}^2 sırasıyla halihazırda kullanılan çift istasyonun sol ve sağ taraf istasyonuna atanan görev kümesini gösterir. Yukarıdaki istasyon odaklı işlemlerle elde edilen fizibil çözüm, istasyonlara görev atamalarını ve istasyonun içinde görevlerin operasyon sıralarını verir. Kullanılan çift istasyon sayısı NM , kullanılan sol istasyon sayısı NL , kullanılan sağ istasyon sayısı ise NR ile gösterilmiştir. Toplam istasyon sayısı $(NL+NR)$ NS 'dir [9].

Amaç fonksiyonu:

ÇMHD-I problemi için önerilen yaklaşımın amacı çift istasyon sayısını ve verilen bir çevrim zamanı için istasyon sayısını en küçükmektir. Uygun çözümü oluşturma işleminde çift istasyonların ardıl oldukları gözönüne alınmıştır (bir çift istasyon maksimum derecede yüklendiği zaman bu çift istasyon kapatılır ve yenisi açılır). Bu nedenle istasyon sayısının en küçüklenmesi istasyon çiftleri sayısının da en küçüklenmesini sağlayacaktır. İstasyonların sayısını en küçükmek hattın etkinliğini en büyükmekle eşdeğerdir. Problemin karışık modelli nitelikte olmasının gözönüne alınmasıyla, ağırlıklandırılmış hat etkinliği (WLE) şu şekilde hesaplanır [9]:

$$WLE = \left(\frac{\sum_{m \in M} q_m \left(\sum_{i \in I} t_{im} \right)}{C.NS} \right) . 100 \quad (5.5)$$

İstasyonlar arasındaki işyükü eşitliği de problemin karışık modelli olması gözönünde bulundurularak verilmiştir. İstasyonlar arasındaki işyükü eşitliği, istasyonların boş zamanlarını olabildiğince eşit olarak dağıtmayı amaçlar. Eş. 5.6 verilen hat dengeleme için istasyonlar arasındaki ağırlıklandırılmış düzgünlük indeksinin (WSI) hesaplanmasıdır [9].

WL_{max} : maksimum istasyon zamanı

$$WSI = \sqrt{\frac{\sum_{m \in M} q_m \cdot \left(\sum_{j \in J} \sum_{k=1,2} ({}_m WL_j^k - WL_{max})^2 \right)}{NS}} \quad (5.6)$$

WLE ' inin en büyüklenmesiyle, hat uzunluğu ve operatör sayısının en küçüklenmesi amaçlanır. WSI ' nin en küçüklenmesiyle ile de istasyonlar arasındaki işyükü farklılığının azaltılması ve bu işlerin istasyonlara olabildiğince dengeli dağıtılması amaçlanır. Bu yaklaşım için amaç fonksiyonu şu şekildedir [9] :

WLE_0 ve WSI_0 başlangıç çözümünden elde edilen amaç fonksiyon değerleridir.

$$\text{Minimize } E = \frac{WLE_0}{WLE} + \frac{WSI}{WSI_0} \quad (5.7)$$

Eş. 5.7 WLE 'nin en büyüklenmesini ve WSI 'nin en küçüklenmesini garanti eder. Önceki bölümlerdeki karışık tamsayılı programlama modelinin amaç fonksiyonu olan (Eş. 4.3), öncelikle çift istasyon sayısını en küçüklemeyi, ikincil olarak da verilen bir çevrim zamanında istasyon sayısını en küçüklemeyi amaçlar.

Önerilen arı kolonisi algoritmasında, iki performans kriteri dikkate alınmıştır. Bunlar Eş. 5.5'de verilen WLE ve Eş. 5.6 'da verilen WSI 'dir. Bunlar tek bir amaç fonksiyonunda (Eş. 5.7) birleştirilmiştir. WLE 'nin en büyüklenmesi istasyon sayısının en küçüklenmesini sağlar. Çünkü çözüm işleminde (çift istasyonların sıralı

olduğu dikkate alınır) WLE 'nin en büyüklenmesi çift istasyon sayısının azaltılmasına neden olacaktır. Eğer WLE 'nin en büyüklenmesi amaç fonksiyonu olarak direkt kullanılırsa çözümler arasında güçlü bir ayrım elde edilemez. Çünkü WLE elde edilen iki çözümün kalitesini karşılaştırmak için yeterli değildir [9]. Bu zorluktan kurtulmak adına bu çalışmada WSI performans kriteri olarak belirlenmiştir. Eş. 4.3 ve Eş. 5.7 çift istasyon sayısını ve istasyon sayısını azaltmayı amaçlamaktadır.

Alt sınır hesaplamaları:

ÇMHD-I problemi için alt sınır (LB) Hu ve ark.(2008) tarafından verilmiştir [37]. Bu LB KÇMHD-I için modifiye edilmiştir [9]. KÇMHD-I problemi için istasyon sayısındaki LB hesaplaması aşağıda verilmiştir.

$$Max = \max \left\{ \left[\frac{\sum_{i \in A_L} \sum_{m \in M} q_m \cdot t_{im}}{C} \right], \left[\frac{\sum_{i \in A_R} \sum_{m \in M} q_m \cdot t_{im}}{C} \right] \right\} \quad (5.8)$$

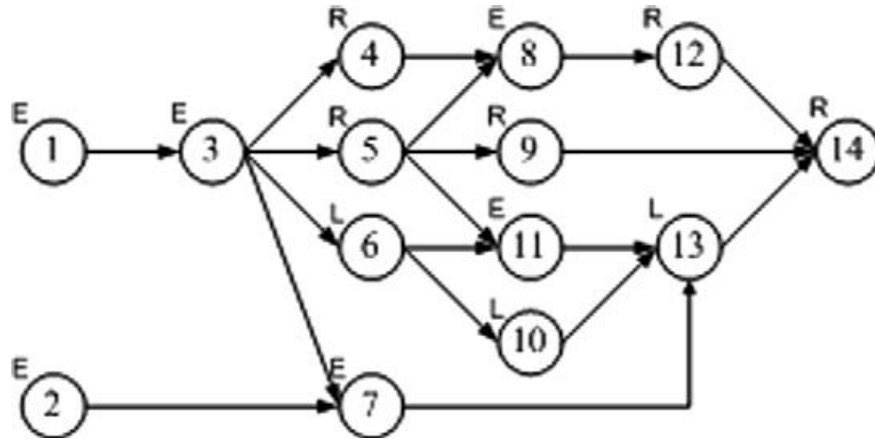
$$LB = 2 \cdot Max + \max \left\{ 0, \left[\frac{\sum_{i \in A_E} \sum_{m \in M} q_m \cdot t_{im} - \left(Max \cdot C - \sum_{i \in A_L} \sum_{m \in M} q_m \cdot t_{im} \right) - \left(Max \cdot C - \sum_{i \in A_R} \sum_{m \in M} q_m \cdot t_{im} \right)}{C} \right] \right\} \quad (5.9)$$

5.4. Sayısal Örnek

Bu bölümde arı kolonisi algoritmasının bazı özelliklerini göstermek adına belirli bir akışta ilerleyen, belirli özelliklere sahip bir sayısal örnek verilmiştir [12].

- i. İki model, A ve B, eş zamanlı olarak planlama ufkunda bir hatta montaj edilmektedir. (çevrim zamanı $C=20$, $q_a=\%50$ ve $q_b=\%50$)

- ii. Aşağıdaki şekil 14 görevli bir kombine öncelik diyagramı olup, her görevin atanacağı yönlerde belirtilmiştir.(L-sol, R-sağ, E-her iki yön) Ayrıca aşağıda verilen çizelgede her iki model için görev işlem zamanları da belirtilmiştir.
- iii. Görev 9 ve 10 senkronize görevlerdir (bu görevler hattın her iki tarafında eş zamanlı olarak yapılmalıdır).



Şekil 5.9. Sayısal örnek için öncelik diyagramı

Çizelge 5.1. Sayısal örnek için görev işlem zamanları

Görev	m_A	m_B
1	0	2
2	8	8
3	7	7
4	7	5
5	2	2
6	6	0
7	4	0
8	0	2
9	3	2
10	3	2
11	6	6
12	3	3
13	5	5
14	4	6

Öncelikle görevler arasında öncelik ilişkisini ve görev yönünü daha iyi görebilmek adına aşağıdaki tablo oluşturulmuştur:

Çizelge 5.2.Sayısal örnek için görev öncelikleri ve tarafları

Görev	Öncül İşlemler	Görev Tarafları (R:sağ, L:sol, E: her iki yön)
1	-	E
2	-	E
3	1	E
4	3	R
5	3	R
6	3	L
7	2,3	E
8	4,5	E
9	5	R
10	6	L
11	5,6	E
12	8	R
13	7,10,11	L
14	9,12,13	R

Çizelge 5.3.Başlangıç hat dengesi

A.1	A.2	A.3	A.4	A.5	A.6
$NM=1,$ $NL=0,$ $NR=0,$ ${}_AW_1^1 = 0$ $,$ ${}_BW_1^1 = 0$ $,$ ${}_AW_1^2 = 0$ $,$ ${}_BW_1^2 = 0$	$SAT=\{1,2\}$ $SAT=\{2,3\}$	$\{1,2\}$ $\{2\}$	Select task 1, $P(1)=\{ \emptyset \},$ task $1 \in A_E,$ $p_2 \leq 0.5,$ $0+0 \leq 20,$ $2+0 \leq 20,$ $TL_1^1 = TL_1^1 + \{1\}, t_{1A}^f = 0,$ $t_{1B}^f = 2,$ ${}_AWL_1^1 = 0,$ ${}_BWL_1^1 = 2$ Select task 2, $P(2)=\{ \emptyset \},$ task $2 \in A_E,$ $p_2 > 0.5,$ $8+0 \leq 20,$ $8+0 \leq 20,$ $TL_1^2 = TL_1^2 + \{2\},$ $t_{2A}^f = 8, t_{2B}^f = 8$ ${}_AWL_1^2 = 8, {}_BWL_1^2 = 8$		

Çizelge 5.3.(Devam) Başlangıç hat dengesi

A.1	A.2	A.3	A.4	A.5	A.6
	SAT={9,10,11,12}	{9,10,11,12}	Select task 9, $P(9)=\{5\}$, task $9 \in A_R$, $3+17 \leq 20$, $2+17 \leq 20$, $TL_1^2 = TL_1^2 + \{9\}$, $t_{9A}^f = 20$, $t_{9B}^f = 19$ $_A WL_1^2 = 20$, $_B WL_1^2 = 19$		
	SAT={10,11,12}	{10,11,12}	Select task 10, $P(10)=\{6\}$, task $10 \in A_L$, $3+17 \leq 20$, $2+9 \leq 20$, $TL_1^1 = TL_1^1 + \{10\}$, $t_{10A}^f = 20$, $t_{10B}^f = 11$ $_A WL_1^1 = 20$, $_B WL_1^1 = 11$		
	SAT={11,12}	{11,12}	Select task 11, $P(11)=\{5,6\}$, task $11 \in A_E$, $p_2 \leq 0.5$ $6+16 > 20$, $6+13 \leq 20$, adım5'e git	Hiçbir görev Seçileme di NL=1, NR=1	
$NM=2$, $_A W_2^1 = 0$, $_B W_2^1 = 0$, $_A W_2^2 = 0$, $_B W_2^2 = 0$	SAT={11,12}	{11,12}	Select task 11, $P(11)=\{5,6\}$, task $11 \in A_E$, $p_2 \leq 0.5$ $6+0 \leq 20$, $6+0 \leq 20$, $TL_2^1 = TL_2^1 + \{11\}$, $t_{11A}^f = 6$, $t_{11B}^f = 6$ $_A WL_2^1 = 6$, $_B WL_2^1 = 6$		
	SAT={12,13}	{12,13}	Select task 12, $P(12)=\{8\}$, task $12 \in A_R$ $3+0 \leq 20$, $3+0 \leq 20$, $TL_2^2 = TL_2^2 + \{12\}$, $t_{12A}^f = 3$, $t_{12B}^f = 3$ $_A WL_2^2 = 3$, $_B WL_2^2 = 3$		
	SAT={13}	{13}	Select task 13, $P(13)=\{7,10,11\}$, task $13 \in A_L$ $5+6 \leq 20$, $5+6 \leq 20$, $TL_2^1 = TL_2^1 + \{13\}$, $t_{13A}^f = 11$, $t_{13B}^f = 11$ $_A WL_2^1 = 11$, $_B WL_2^1 = 11$		

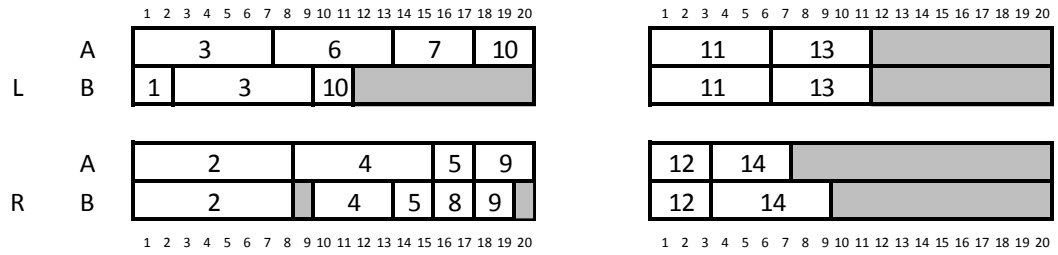
Çizelge 5.3.(Devam) Başlangıç hat dengesi

A.1	A.2	A.3	A.4	A.5	A.6
$NM=2,$ ${}_A W_2^1 = 0$ $,$ ${}_B W_2^1 = 0$ $,$ ${}_A W_2^2 = 0$ $,$ ${}_B W_2^2 = 0$	$SAT=\{14\}$ $SAT=\{\emptyset\}$	$\{14\}$	Select task 14, $P(14)=\{9,12,13\},$ $task\ 14 \in A_R$ $4+3 \leq 20,$ $6+3 \leq 20,$ $TL_2^2 = TL_2^2 + \{14\},$ $t_{14A}^f = 7, t_{14B}^f = 9$ ${}_A WL_2^2 = 7, {}_B WL_2^2 = 9$		Dur

$$WLE=66,875$$

$$WSI=2,485$$

$$LB=4$$



Şekil 5.10. Başlangıç hat dengesi

Bu başlangıç dengesinden sonra oluşturulan algoritmanın durdurma kriterine rastlayıncaya kadar uygulanması sonucu elde edilen sonuçlar aşağıdadır;

$$WLE=80,275$$

$$WSI=1,305$$

$$NM[NS]=2[4]$$

5.5. Deneysel Çalışma

Çalışmanın bu aşamasında ortaya konulan algoritmanın etkinliği test edilmiştir. Ozcan ve Toklu'nun çalışmalarında elde ettikleri sonuçlarla birebir karşılaştırma

yapabilmek adına arı kolonisi algoritmasının özel kısıt içermeyen KÇMHD için ayrı ayrı çalıştırılmış ve sonuçlar ilgili çalışma ile karşılaştırılmıştır [9].

Öncelikle algoritma parametreleri literatürde sunulan genel öneriler ve önceki deneyimlere göre belirlenmiş olup Çizelge 5.4’de verildiği gibidir. Çalışmamızda kullanılan kontrol parametresi değerlerinin belirlenmesi aşamasında $\{S,P,e,nep,nsp\}$: $\{20,10,5,4,2\}$, $\{15,5,3,4,2\}$, $\{10,5,3,2,1\}$, $\{5,3,2,2,1\}$ olmak üzere 4 farklı parametre kombinasyonu önceki çalışmalarda oluşturulmuştur [39]. Test problemleri üzerinde yapılan analizler neticesinde 3. parametre kombinasyonun en iyi performansı gösterdiği görüldüğünden, algoritmanın parametreleri olarak bu değerler kullanılmıştır.

Çizelge 5.4. KÇMHD için AKA parametre değerleri

Parametre	Değer
<i>S</i>	10
<i>P</i>	5
<i>e</i>	3
<i>nep</i>	2
<i>nsp</i>	1
<i>MaksIter</i>	100

Özcan ve Toklu’nun çalışmalarında yararlandıkları test problemleri küçük boyutlu (p9, p12, p16, p24) ve büyük boyutlu (p65, p148) problemleri içermekte olup, problem tanımlarındaki sayılar ilgili pozisyonadaki görev sayısını göstermektedir [9].

KÇMHD’nin çözümü için önerilen arı kolonisi algoritması, C++ programlama dilinde kodlanarak 1.73 GHz CPU ve 4GB RAM özelliklere sahip Intel Core i7-740QM bilgisayar kullanılarak test problemleri üzerinde analiz edilmiştir. İlgili programlama dilinde kodlanarak oluşturulan programın ekran görüntüsü şu şekildedir:

Çözülecek problemi seğiniz: Seğiniz Çevrim zamanını seğiniz: Seğiniz

Öncelik İlişkileri

Öncüller	Ardıllar
----------	----------

Görev Süreleri

Görev	Model A	Model B
-------	---------	---------

Görev Tarafları

Görev	Taraflı
-------	---------

Problemi Çöz

LB CPU NS NM WLE WSI

Resim 5.1. Program boş başlangıç sayfası

Çözülecek problemi seğiniz: Problem 1 Çevrim zamanını seğiniz: 4

Öncelik İlişkileri

Öncüller	Ardıllar
1	4
2	5
2	6
3	6
4	7

Görev Süreleri

Görev	Model A	Model B
1	2	0
2	3	1
3	0	1
4	3	0
5	1	3
6	1	1

Görev Tarafları

Görev	Taraflı
1	2
2	1
3	0
4	2
5	1

Problemi Çöz

LB 4 CPU 0.15 NS 4 NM 3 WLE 78.12 WSI 1.275

Resim 5.2. Problem-1'in çevrim zamanı 4 için sonuç sayfası

AKA her problem kümesinin bütün çevrim süreleri için 5'er kez çalıştırılmıştır. Minimum, ortalama ve maksimum istasyon sayıları, minimum CPU süreleri ile birlikte ilgili çizelgelerde sunulmuştur.

Çizelge 5.5 ve Çizelge 5.8 'de görüldüğü gibi önceden çözümü başka yaklaşımlarla denenmiş problemler için önerilen algoritma bu test problemlerinin üzerinde başarılı sonuçlar vermiştir. Ancak küçük boyutlu problemlerin sonuçları Özcan ve Toklu'nun çalışmalarıyla büyük bir farklılık gösterememiş olup, sonuçları geliştirecek bir netice elde edilememiştir [9]. Büyük boyutlu problemlerde ise P65 için bir adet ve P148 için ise üç adet olmak üzere toplam dört adet test probleminin sonuçları geliştirilmiştir. Bu sonuçların bilinen en iyi sonuçlar olduğu söylenebilir.

Önerilen yaklaşım 31 test problemi arasından 15 test probleminin *LB* değerlerine ulaşmıştır.

Çizelge 5.5. Küçük boyutlu test problemlerinin karşılaştırmalı sonucu

Problem	M#	C	LB	NM[NS]	WLE%	WSI	CPU(s)
P9	2	4	4	3[4]	78,12	1,275	0,15
		5	3	2[3]	83,33	1,080	0,88
		6	3	2[3]	70,15	1,080	0,83
P12	2	5	5	3[5]	84,00	1,183	1,16
		6	4	2[4]	87,50	0,870	1,25
		7	3	2[4]	75,00	0,870	1,63
		8	3	2[3]	87,50	1,291	1,99
P16	2	15	5	4[6]	75,00	3,708	3,27
		16	5	4[6]	70,31	4,615	3,89
		18	4	3[5]	75,00	5,909	4,01
		19	4	3[5]	71,05	5,909	3,66
		21	4	2[4]	80,35	5,160	3,15
		22	4	2[4]	76,40	4,859	3,24
P24	2	20	7	4[7]	89,28	2,903	8,75
		24	6	3[6]	86,80	2,858	9,02
		25	5	3[6]	83,33	2,799	8,99
		30	5	3[5]	83,33	2,450	7,98
		35	4	2[4]	89,28	2,644	8,02
		40	4	2[4]	78,12	2,291	8,16

Çizelge 5.6. Büyük boyutlu test problemleri SA Algoritması [9] sonuçları

Problem	M#	C	LB	SA Algoritması [9]	
NM[NS]					
				En İyi	Ortalama
P65	3	326	8	5[9]	5[9]
		381	7	4[8]	4[8]
		435	6	4[7]	4[7]
		490	6	3[6]	3[6]
		544	5	3[6]	3[6]
P148	4	204	13	9[17]	9[17]
		255	11	7[14]	7[14]
		306	9	6[12]	6[12]
		357	8	5[10]	5[10]
		408	7	5[10]	5[10]
		459	6	4[8]	4[8]
		510	6	4[8]	4[8]

Çizelge 5.7. Büyük boyutlu test problemlerinin başlangıç hat denge sonuçları

Problem	M#	C	LB	Başlangıç Hat Denge Sonuçları		
				NM[NS]	WLE%	WSI
P65	3	326	8	6[12]	65,98	144,325
		381	7	5[9]	72,35	132,333
		435	6	4[7]	81,86	91,177
		490	6	4[7]	78,65	123,264
		544	5	4[7]	69,58	185,336
P148	4	204	13	12[23]	51,70	103,258
		255	11	9[18]	57,28	123,746
		306	9	8[16]	53,70	156,790
		357	8	7[13]	55,27	175,824
		408	7	6[12]	53,70	170,197
		459	6	5[10]	59,39	181,354
		510	6	5[10]	53,25	224,897

Çizelge 5.8. Büyük boyutlu test problemlerinin önerilen AKA sonuçları

Problem	M#	C	LB	Önerilen AKA						CPU(s)
				NM[NS]		WLE%		WSI		
				En İyi	Ortalama	En İyi	Ortalama	En İyi	Ortalama	
P65	3	326	8	5[9]	5[9]	84,93	84,93	62,765	75,282	765
		381	7	4[8]	4[8]	81,65	81,65	68,998	74,521	789
		435	6	4[7]	4[7]	81,83	81,83	85,773	91,258	824
		490	6	3[6]	3[6]	84,81	84,81	76,591	88,005	596
		544	5	3[5]	3[5]	81,83	81,83	69,275	74,238	688
P148	4	204	13	9[17]	9[17]	76,01	76,01	60,465	64,991	3876
		255	11	7[13]	7[13]	80,08	80,08	70,326	75,269	4028
		306	9	6[11]	6[11]	76,53	76,53	81,478	84,335	3982
		357	8	5[10]	5[10]	73,64	73,64	102,768	105,241	3889
		408	7	5[9]	5[9]	70,72	70,72	81,128	88,524	3870
		459	6	4[8]	4[8]	71,62	71,62	103,301	109,365	3881
		510	6	4[8]	4[8]	64,46	64,46	81,284	84,657	3625

Yapılan deneysel çalışma ile önerilen algoritmanın büyük boyutlu problemlerde etkili sonuçlar verdiği, küçük boyutlu problemlerde de önceki çalışmalara yakın neticeler verdiği görülmüştür. Ortaya konulan algoritma ile P65 ve P148 deneysel çalışmasında belirli çevrim zamanlarında önceki çalışmadan daha düşük sayıda istasyona yerleştirme yapıldığından hattın etkinliği geliştirilmiştir.

Karışık modelli montaj hatlarında yerleştirilen hat sayısı düştükçe ya da hattaki boş zamanlar azaldıkça WSI indeksinin düşeceği ve WLE indeksinin de yükseleceği gözönüne alındığında, çözümü yapılan test problemlerinde daha iyi bir hat yerleştirmesi ile bu indeks değerlerinin iyileştirildiği görülmüştür. Önerilen algoritma ile çözümlerde daha dengeli bir yerleştirme ile büyük boyutlu problemlerde göreceli olarak WSI ve WLE indeks değerlerinin geliştirilmesi sağlanırken, küçük boyutlu problemlerde önceki çalışmaya yakın neticeler elde edilmiştir.

Önerilen yaklaşımın hesaplama zamanı 0.15 s ve 4028 s aralığında (minimum) bulunmuştur. Önerilen yaklaşımın denenmiş yöntemlere nazaran daha fazla hesaplama zamanına ihtiyaç duyduğu söylenebilir.

ÇMHD problemleri literatürde yer alan zor problemlerden biridir. Problemin karışık modelli olması çözümü daha da zorlaştırmaktadır. Önceki bölümlerde de belirtildiği üzere çift taraflı montaj hatlarında karışık modellerin dâhil olduğu çalışmalar oldukça azdır. Önerilen arı kolonisi algoritmasının deneysel çalışma bölümünde bahsedilen problemlere uygulanmasıyla tatmin edici sonuçlar elde edilmiştir. Algoritmanın ve komşuluk yapılarının en basit hallerinin kullanıldığı da göz önüne alındığında önerilen algoritma KÇMHD problemleri üzerinde iyi bir performansa sahip olduğu söylenebilir.

6. SONUÇ

Çift taraflı montaj hatlarında karışık modelli üretim problemlerine yönelik çalışmalar literatürde sınırlıdır. Bu tez çalışmasında, çift taraflı montaj hattı dengeleme problemlerinde karışık modeller için, literatürde yeni bir konu olan arı kolonisi algoritması yaklaşımıyla çözümler önerilmiştir. Önerilen algorithmada verilen bir çevrim zamanı için çift istasyon sayısını öncelikli amaç olarak, istasyon sayısını ikincil amaç olarak en küçüklemelektir.

Literatürde, karışık modelli üretim yapılan çift taraflı montaj hatları için karınca kolonisi algoritması kullanılarak çözüm önerilmiştir [12]. Bu çalışma ile önerilen yaklaşım eş zamanlı olarak hattın etkinliğini en büyüklemele ve düzgünlük indeksini en küçüklemelektir.

Önerilen arı algoritması yaklaşımı test problemleri kullanılarak performansları literatürdeki diğer yaklaşımlarla karşılaştırmalı olarak incelenmiştir. Test sonuçlarına göre önerilen arı kolonisi algoritması etkili sonuçlar vermektedir. Kullanılan ve daha önceki çalışmalarda önerilen matematiksel model ile sadece küçük boyutlu problem seti çözülebilmıştır. Bu çalışma ile önerilen algoritma ile büyük boyutlu problemlerde etkili sonuçlar elde edilmiştir. Önerilen yaklaşımın denenmiş yöntemlere nazaran daha fazla hesaplama zamanına ihtiyaç duyduğu söylenebilir.

Literatürde de görüldüğü gibi KÇMHD üzerine yapılan çalışmalar oldukça az olup tek modelli hatlar üzerinde etkin performans gösteren algoritmalar karışık modelli hatlar üzerinde de incelenebilir.

Bu bilgiler ışığında ileride daha karmaşık problemler ele alınabilir. Yapılan çalışmalarla oluşturulan çözüm yaklaşımları geliştirilebilir ve önerilen yaklaşımların özellikle gerçek hayat problemleri üzerinde test edilebilir. İmalat sanayinin günümüz rekabet koşullarında ihtiyaç duyduğu etkin ve verimli çalışma koşullarını sağlayacak hat yerleştirmeleri ile yapılan çalışmalar uygulamaya geçirilebilir.

KAYNAKLAR

1. Ajenblit, D.A., Wainwright, R.L., “Applying genetic algorithms to the U-shaped assembly line balancing problem”, ***Proceedings of the 1998 IEEE International Conference on Evolutionary Computation***, Anchorage, AK, 96-101 (1998).
2. Acar, N., Ertas, S., “Kesikli Seri Üretim Sistemlerinde Planlama ve Kontrol çalışmaları”, ***MPM Yayınları***, Ankara, 156 (1986).
3. Aase, G.R. , Olson. J.R. , Schniederjans. M.J., “U-shaped assembly line layouts and their impact on labor productivity: An experimental study”, ***European Journal of Operational Research***, 156: 698–711 (2004)
4. Hwang, R.K. , Katayama, H. , “A multi-decision genetic approach for workload balancing of mixed-model U-shaped assembly line systems”, ***International Journal of Production Research***, 47: 3797-3822 (2009).
5. Gökçen, H., Ağpak, K., Benzer, R. , “Balancing of parallel assembly lines”, ***International Journal of Production Economics***, 103 : 600–609 (2006).
6. Scholl, A., Boysen, N., “Designing parallel assembly lines with split workplaces: Model and optimization procedure”, ***International Journal of Production Economics***, 119 : 90–100 (2009).
7. Ozbakir, L., Baykasoglu, A., Gorkemli, B., Gorkemli, L., “Multiple-colony ant algorithm for parallel assembly line balancing problem”, ***Applied Soft Computing***, 11: 3186–3198 (2011)
8. Ozcan, U., “Çift taraflı montaj hattı dengeleme problemleri için yeni çözüm önerileri”, Doktora Tezi, ***Gazi Üniversitesi Fen Bilimleri Enstitüsü***, Ankara, 10-16 (2009).
9. Ozcan, U., Toklu, B., “Balancing of mixed-model two-sided assembly lines”, ***Computers and Industrial Engineering***, 57 : 217-227 (2009a)
10. Bartholdi, J.J., “Balancing two-sided assembly lines: A case study”, ***International Journal of Production Research***, 31: 2447-2461 (1993)
11. Kim, Y.K., Song, W.S., Kim, J.H., “A mathematical model and a genetic algorithm for two-sided assembly line balancing”, ***Computers and Operations Research***, 36:853-865 (2009)
12. Simaria, A.S., Vilarinho, P.M., “2-ANTBAL: An ant colony optimisation algorithm for balancing two-sided assembly lines”, ***Computers and Industrial Engineering***, 56: 489–506 (2009)

13. Wu, E-F., Jin, Y., Bao, J-S., Hu, X-F., "A branch-and-bound algorithm for two-sided assembly line balancing", *International Journal of Advanced Manufacturing Technology*, 39 (9-10): 1009-1015 (2008).
14. Lee, T.O., Kim, Y., Kim, Y.K., "Two-sided assembly line balancing to maximize work relatedness and slackness", *Computers and Industrial Engineering*, 40: 273-292 (2001).
15. Helgeson, W.B., Salvesson, M.E., Smith, W.W., "How to balance an assembly line", Technical Report, *Carr Press*, New Caraan, Conn (1954).
16. Salvesson, M.E., "The assembly line balancing problem", *The Journal of Industrial Engineering*, 6 (3): 18-25 (1955).
17. Talbot, F.B., Patterson, J.H., "An integer programming algorithm with network cuts for solving the simple assembly line balancing problem", *Management Science*, 30: 85-99 (1984).
18. Agrawal, P.K., "The related activity concept in assembly line balancing", *International Journal of Production Research*, 23: 403-421 (1985).
19. Rosenblatt, M.J., Carlson, R.C., "Designing a production line to maximize profit", *IIE Transactions*, 17: 117-121 (1985).
20. Shtub, A., Dar-El, E.M., "An assembly chart oriented assembly line balancing approach", *International Journal of Production Research*, 6: 1137-1151 (1990).
21. Plans, J., Corominas, A., "Modelling and solving the SALB-E problem", *Proceedings of the 1999 IEEE International Symposium on Assembly and Task Planning*, Porto, Portugal - July (1999).
22. Bautista, J., Suarez, R., Mateo, M., Companys, R., "Local search heuristics for the assembly line balancing problem with incompatibilities between tasks", *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, San Francisco, CA, 2404-2409 (2000).
23. Ponnambalam, S.G., Aravindan, P., Naidu, G.M., "A multi-objective genetic algorithm for solving assembly line balancing problem", *International Journal of Advanced Manufacturing Technology*, 16: 341-352 (2000).
24. Carnahan, B.J., Norman, B.A., Redfern, M.S., "Incorporating physical demand criteria into assembly line balancing", *IIE Transactions*, 33: 875-887 (2001).
25. Ağpak, K., Gökçen, H., "Assembly line balancing: Two resource constrained cases", *International Journal of Production Economics*, 96: 129-140 (2005).

26. Miralles, C., García-Sabater, J.P., Andrés, C., Cardós, M., “Branch and bound procedures for solving the Assembly Line Worker Assignment and Balancing Problem: Application”, ***Discrete Applied Mathematics***, doi: 10.1016/j.dam.2005.12.012 (2007).
27. Otto, A., Scholl, A., “Incorporating ergonomic risks into assembly line balancing”, ***European Journal of Operational Research***, 212: 277-286 (2011).
28. Toklu, B., Özcan, U., “A fuzzy goal programming model for the simple U-line balancing problem with multiple objectives”, ***Engineering Optimization***, 40(3): 191- 204 (2008).
29. Boysen, N., Fliedner, M., Scholl, A., “Assembly line balancing: Which model to use when?”, ***Int. J. Production Economics*** 111: 509–528 (2008).
30. Wei, N.C., Chao, I.M., “A solution procedure for type E simple assembly line balancing problem”, ***Computers and Industrial Engineering***, 61 : 824-830 (2011).
31. Mosadegh, H., Zandieh, M., Ghomi, S.M.T.F., “ Simultaneous solving of balancing and sequencing problems with station-dependent assembly times for mixed-model assembly lines”, ***Applied Soft Computing***, 12 : 1359–1370 (2012).
32. Ozcan, U., Toklu, B., “Multiple-criteria decision-making in two-sided assembly line balancing: A goal programming and a fuzzy goal programming models”, ***Computers and Operations Research***, 36 : 1955 – 1965 (2009b).
33. Ozcan, U., Gokcen, H., Toklu, B., “Balancing parallel two-sided assembly lines”, ***International Journal of Production Research***, 48(16) : 4767-4784 (2010)
34. Ozcan, U., Toklu, B., “Balancing two-sided assembly lines with sequence-dependent setup times”, ***International Journal of Production Research***, 48(18) : 5363-5383 (2010)
35. Ozcan, U., “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 : 81-97 (2010)
36. Taha, R.B., El-Kharbotly, A.K., Sadek, Y.M., Afia, N.H., “A Genetic Algorithm for solving two-sided assembly line balancing problems”, ***Ain Shams Engineering Journal***, 2 : 227 – 240 (2011)
37. Xiaofeng, H., Erfei, W., Ye, J., “A station-oriented enumerative algorithm for two-sided assembly line balancing”, ***European Journal of Operational Research***, 186 : 435–440 (2008)

38. Chutima, P., Chimklai, P., “Multi-objective two-sided mixed-model assembly line balancing using particle swarm optimisation with negative knowledge”, ***Computers and Industrial Engineering***, 62 : 39-55 (2012)
39. Ozbakır, L., Tapkan, P., “Bee colony intelligence in zone constrained two-sided assembly line balancing problem” ***Expert Systems with Applications***, 38:11947-11957 (2011)
40. Lapierre, S.D., Ruiz, A.B., “Balancing Assembly Lines: An Industrial Case Study” , ***Journal of Operational Research Society***, 55(6), 589-597 (2004).
41. Baykasoglu, A., Dereli, T., “Two-sided assembly line balancing using an ant colony based heuristic”, ***International Journal of Advanced Manufacturing Technology***, 36 (5-6):582-588 (2008).
42. Kara, Y., Ozcan, U., Peker, A., “Balancing and sequencing mixed-model just-in-time U-lines with multiple objectives”, ***Applied Mathematics and Computation*** 184 : 566–588 (2007)
43. Macaskill, J. L. C., “ Production-line balances for mixed model lines”, ***Management Science***, 19: 423-434 (1972).
44. Gokcen, H., and Erel, E., “A goal programming approach to mixed-model assembly line balancing problem”, ***International Journal of Production Economics***, 48: 177-185 (1997).
45. Kim, Y. K., Kim, Y., Kim, Y. J., “Two-sided assembly line balancing: a genetic algorithm approach”, ***Production Planning and Control***, 11(1) : 44-53 (2000).
46. Lucic, P., Teodorovic, D., “Transportation Modeling: An Artificial Life Approach”, ***In Proceedings of the 14th international conference on tools with artificial intelligence***, Washington, DC, USA, 216-223 (2002).
47. Tapkan, P. Z., “Kombinatorial Optimizasyon Problemlerinde Arı Sistemi Yaklaşımı”, Doktora Tezi, ***Erciyes Üniversitesi Fen Bilimleri Enstitüsü***, Kayseri, 10-16 (2010).
48. Yonezawa, Y., Kikuchi, T., “Ecological Algorithm for Optimal Ordering Used by Collective Honey Bee Behavior”, ***In Proceedings of 7th International Symposium on Micro Machine and Human Science***, Nagoya , Japan, 249-256 (1996).
49. Schmickl, T., Thenius, R., Crailsheim, K., “Simulating Swarm Intelligence in Honey Bees: Foraging in Differently Fluctuating Environments”, ***In Proceedings of Genetic and Evolutionary Computation Conference***, Washington DC, USA, 273-274, 2005.

50. Adams, J., Rothman, E.D., Kerr, W.E., Paulino, Z.L., “Estimation of the Number of Sex Alleles and Queen Matings from Diploid Male Frequencies in a Population of *Apis Mellifera*”, ***Genetics***, 86(3): 583–596 (1977).
51. Laidlaw, H. H., Page, R. E., “Mating Designs, In: Rinderer, T.E., (eds.), *Bee Genetics and Breeding*”, ***Academic Press, Inc***, 323-341 (1986).
52. Rinderer, T. E., Collins, A. M., “Behavioral Genetics, In: Rinderer, T.E., (eds.), *Bee Genetics and Breeding*”, ***Academic Press, Inc***, 155-176 (1986).
53. Karl von, F., “The dance language and orientation of bees”, ***The Belknap Press of Harvard University Press***, (1967).
54. Tarpy, D. R., “The dance language of the honey bee,” Doktora Tezi, ***NC State University***, North Carolina State (2009).
55. Wenner, A. M., Wells, P. H. , “Anatomy of a controversy: The Question of a Language among bees” ***Columbia University Press***, (1990).
56. Karaboga, D., “An Idea Based on Honey Bee Swarm for Numerical Optimization”, Technical Report, ***Erciyes University Engineering Faculty Computer Engineering Department***, Kayseri (2005).
57. Lucic, P., “Modeling Transportation Problems Using Concepts of Swarm Intelligence and Soft Computing”, Ph.D. Dissertation, ***Civil Engineering, Faculty of the Virginia Polytechnic Institute and State University***, Virginia (2002).
58. Davidovic, T., Ramljak, D., Selmic, M., Teodorovic, D., “Bee Colony Optimization For the P-center Problem”, ***Computers and Operations Research*** 38 : 1367–1376 (2011).
59. Teodorovic, D., Davidovic, T., Selmic, M., “Bee Colony Optimization: The Applications Survey”, ***ACM Transactions on Computational Logic***, 1-20 (2011).
60. Lucic, P., Teodorovic, D., “Bee system: modeling combinatorial optimization transportation engineering problems by swarm intelligence” ***TRISTAN IV Triennial Symposium on Transportation Analysis***, Sao Miguel, Azores Islands, 441-445 (2001).
61. Lucic, P., Teodorovic, D., “Computing with bees: attacking complex transportation engineering problems” ***International Journal on Artificial Intelligence Tools***, 12: 375-394 (2003a).
62. Lucic, P., Teodorovic, D., “Vehicle routing problem with uncertain demand at nodes: the bee system and fuzzy logic approach”, ***In Fuzzy Sets based Heuristics for Optimization***, 67-82 (2003b).

63. Pham, D.T., Koc,E., Ghanbarzadeh, A., Otri, S., Rahim, S., and Zaidi, M. “The bees algorithm-A novel tool for complex optimisation problems”, ***In Proceedings of IPROMS conference***, 454-461 (2006).
64. Pham, D.T., Castellani, M., “The Bees Algorithm: Modelling Foraging Behaviour to Solve Continuous Optimization Problems, In Proceedings of the Institution of Mechanical Engineers”, ***Journal of Mechanical Engineering Science***, 223(12): 2919-2938, (2009).
65. Leu, Y. Y., Matheson, L. A., and Rees, L. P., “Assembly line balancing using genetic algorithms with heuristic-generated initial populations and multiple evaluation criteria”, ***Decision Sciences***, 25: 581-606 (1994).

EKLER

EK-1. Küçük boyutlu problemlerde tüm modeller için görev tamamlanma zamanları

P-9	P-12		P-16		P-24	
	Model A	Model B	Model A	Model B	Model A	Model B
Görev						
1	2	0	2	3	6	0
2	3	1	3	3	5	2
3	0	1	2	0	2	0
4	3	0	3	2	0	9
5	1	3	1	2	8	0
6	1	1	1	0	4	8
7	2	2	3	2	7	7
8	0	3	3	1	4	3
9	1	1	2	1	0	5
10			2	3	4	1
11			0	2	6	3
12			0	1	0	5
13					6	9
14					4	5
15					3	8
16					4	7
17						2
18						7
19						9
20						9
21						8
22						0
23						9
24						9

EK-2. Problem-65’de tüm modeller için görev tamamlanma zamanları

P-65							
	Model A	Model B	Model C		Model A	Model B	Model C
Görev				Görev			
1	35	26	28	34	41	39	52
2	14	15	38	35	21	63	34
3	1	54	58	36	33	28	8
4	18	1	11	37	147	103	79
5	36	33	16	38	43	12	85
6	29	26	2	39	15	24	19
7	159	61	88	40	27	1	7
8	70	5	54	41	4	5	17
9	24	16	32	42	25	80	104
10	99	40	43	43	22	18	28
11	56	56	40	44	3	7	23
12	51	47	12	45	44	35	15
13	94	132	33	46	26	34	20
14	29	6	54	47	2	19	10
15	39	1	33	48	41	44	18
16	15	16	121	49	9	3	16
17	11	2	12	50	8	46	5
18	74	58	184	51	20	13	40
19	34	14	10	52	35	13	31
20	19	19	77	53	14	5	2
21	14	14	24	54	38	93	35
22	12	24	8	55	11	23	12
23	19	61	8	56	56	7	79
24	38	76	22	57	23	98	67
25	89	43	33	58	52	64	23
26	66	45	45	59	16	2	19
27	27	6	35	60	10	24	3
28	189	249	227	61	98	66	37
29	2	48	82	62	3	11	5
30	21	33	25	63	117	15	3
31	6	5	4	64	54	85	143
32	18	12	3	65	35	26	55
33	13	53	3				

EK-3. Problem 148’de tüm modeller için görev tamamlanma zamanları

P-148																	
	A	B	C	D		A	B	C	D		A	B	C	D			
Görev						Görev						Görev					
1	12	10	10	6	51	18	15	25	7	101	7	11	13	6			
2	10	24	1	24	52	6	7	10	7	102	17	4	6	19			
3	7	6	1	4	53	51	61	28	74	103	8	3	14	7			
4	42	38	19	46	54	13	18	23	10	104	1	20	14	35			
5	26	3	29	12	55	3	3	2	5	105	53	49	11	7			
6	5	7	1	6	56	7	17	11	26	106	1	5	12	6			
7	19	13	25	26	57	7	3	9	10	107	9	8	6	7			
8	11	11	32	32	58	33	42	9	43	108	7	12	2	37			
9	20	33	30	8	59	4	14	2	2	109	32	33	14	19			
10	21	29	8	16	60	3	2	2	1	110	20	28	20	8			
11	3	18	12	1	61	1	2	1	3	111	6	9	14	9			
12	7	2	2	10	62	5	5	3	5	112	87	70	94	9			
13	10	2	10	13	63	13	13	7	15	113	7	3	12	11			
14	6	15	16	7	64	26	4	22	25	114	20	17	9	16			
15	16	10	10	35	65	1	4	4	3	115	15	5	3	11			
16	23	23	39	18	66	19	15	14	9	116	2	12	22	4			
17	6	3	2	6	67	8	4	5	5	117	5	10	14	19			
18	3	12	23	7	68	14	3	10	6	118	9	9	22	9			
19	20	10	8	23	69	4	6	3	13	119	22	23	47	24			
20	6	6	4	2	70	12	7	6	12	120	23	21	14	30			
21	5	6	7	7	71	27	46	48	34	121	22	18	8	20			
22	1	5	8	4	72	14	4	14	25	122	20	13	25	8			
23	10	8	8	7	73	23	37	27	19	123	14	18	6	8			
24	6	6	5	2	74	29	4	31	29	124	1	5	15	12			
25	3	11	2	5	75	51	17	24	34	125	12	13	6	11			
26	10	13	5	13	76	5	1	4	5	126	39	42	5	27			
27	4	8	7	3	77	23	10	28	24	127	24	5	26	39			
28	24	17	14	11	78	6	8	8	4	128	5	4	9	9			
29	2	10	6	9	79	15	107	90	78	129	8	7	8	2			
30	18	25	2	7	80	4	1	2	2	130	6	9	2	5			
31	3	4	9	4	81	14	12	4	8	131	9	11	11	15			
32	1	6	8	6	82	7	9	6	3	132	3	16	3	12			
33	13	2	4	11	83	20	9	8	17	133	15	13	15	9			
34	20	16	7	30	84	7	13	7	24	134	19	16	13	10			
35	40	23	5	33	85	13	8	9	18	135	35	8	4	12			

EK-3. (Devam) Problem 148’de tüm modeller için görev tamamlanma zamanları

P-148															
	A	B	C	D		A	B	C	D		A	B	C	D	
Görev					Görev					Görev					
36	20	23	24	11	86	13	21	12	8	136	50	62	23	45	
37	3	2	5	2	87	42	13	13	9	137	18	2	18	12	
38	75	9	37	23	88	9	1	18	20	138	4	10	15	15	
39	7	6	3	6	89	5	10	6	12	139	23	1	24	34	
40	12	5	2	14	90	15	9	2	9	140	1	22	3	7	
41	38	15	12	24	91	40	83	37	93	141	89	123	7	9	
42	5	6	2	9	92	6	22	34	10	142	85	48	62	76	
43	8	29	20	25	93	25	4	27	17	143	48	30	22	6	
44	62	23	37	6	94	29	43	27	12	144	70	98	131	100	
45	52	34	78	10	95	18	2	10	16	145	90	112	58	110	
46	7	5	3	2	96	9	13	13	17	146	47	45	48	13	
47	21	10	42	6	97	6	12	5	2	147	61	6	20	75	
48	1	5	8	13	98	31	5	23	32	148	65	18	15	19	
49	26	20	41	40	99	13	49	44	24						
50	23	25	34	12	100	20	31	34	16						

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : TÜRKDÖNMEZ, İsmail
 Uyruğu : T.C.
 Doğum tarihi ve yeri : 20.06.1984 Kayseri
 Medeni hali : Evli
 Telefon : 0 (541) 827 47 12
 e-mail : ismailturkdonmez@yahoo.com

Eğitim Derece

Eğitim Birimi

Mezuniyet Tarihi

Lisans	Gazi Üniversitesi / End. Müh. Bölümü	2008
Lise	Etimesgut Anadolu Lisesi	2002

İş Deneyimi Yıl

Yer

Görev

2008-2009	T.Halk Bankası A.Ş.	Ticari-Kobi Krediler Görevlisi
2010-2011	DAF-Tırsan Otomotiv A.Ş.	Saha Planlama Mühendisi
2011-	T. Vakıflar Bankası A.Ş. Bölge İstihbarat Müdürlüğü	Uzman

Yabancı Dil

İngilizce

Hobiler

Kitap okumak, seyahat etmek, fotoğrafçılık