

**PARALEL ÇOK İŞÇİLİ MONTAJ HATTI
DENGELEME PROBLEMLERİ İÇİN YENİ ÇÖZÜM
YAKLAŞIMLARI**

Talip KELLEGÖZ

**DOKTORA TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

MAYIS 2011

ANKARA

Talip KELLEGÖZ tarafından hazırlanan PARALEL ÇOK İŞÇİLİ MONTAJ HATTI Dengeleme problemleri için yeni çözüm yaklaşımları adlı bu tezin Doktora 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 Doktora tezi olarak kabul edilmiştir.

Prof. Dr. Serpil EROL
Endüstri Müh. Anabilim Dalı, Gazi Üni.

Prof. Dr. Bilal TOKLU
Endüstri Müh. Anabilim Dalı, Gazi Üni.

Prof. Dr. Levent KANDİLLER
Endüstri Müh. Anabilim Dalı, Çankaya Üni.

Prof. Dr. Hadi GÖKÇEN
Endüstri Müh. Anabilim Dalı, Gazi Üni.

Prof. Dr. Burak BİRGÖREN
Endüstri Müh. Anabilim Dalı, Kırıkkale Üni.

Tarih: 13/05/2011

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

Prof. Dr. Bilal TOKLU
Fen Bilimleri Enstitüsü Müdürü

TEZ BİLDİRİMİ

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

Talip KELLEĞÖZ

**PARALEL ÇOK İŞÇİLİ MONTAJ HATTI DENGELEME PROBLEMLERİ
İÇİN YENİ ÇÖZÜM YAKLAŞIMLARI**

(Doktora Tezi)

Talip KELLEGÖZ

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

Mayıs 2011

ÖZET

Otobüs, kamyon ve hâтта helikopter gibi büyük boyutlu karmaşık ürünlerin üretildiği montaj hatları, literatürde çalışılan klasik montaj hatlarından çok farklıdır. Bu tip ürünlerin üretim prosesleri, çoğu uzun görev zamanına sahip çok sayıda montaj görevi içerir. Dolayısıyla bu ürünlerin üretiminde basit veya çift taraflı hatların kullanılması yüzlerce istasyon ihtiyacını ortaya çıkaracaktır. Bu durum ise ürünün hattaki akış süresinin uzun olması, hattın kurulumu için geniş alan ihtiyacı, istasyonlardaki ekipman ve takımlar için yüksek yatırım maliyeti ve istasyonlarda bekleyen çok sayıda ara stok anlamına gelmektedir. Bu tip ürünlerin üretiminde bahsi geçen dezavantajların ortadan kaldırılması amacıyla paralel çok işçili montaj hatları kullanılabilir. Bu çalışma kapsamında bu tip hatlar ve bu hatlardaki dengeleme problemlerinden bir tanesi göz önünde bulundurulmuştur. Problem tanımı ve literatür araştırmasından sonra, ilgili problemin karışık tamsayılı doğrusal programlama modeli geliştirilmiştir. Deneysel çalışmalarda küçük boyutlu problemlerin sadece bir kısmının optimal çözümlerinin matematiksel model yoluyla bulunabileceği görülmüştür. Bu nedenle, problemin çözümü için yeni bir kurucu sezgisel önerilmiştir. Sezgiselin etkinliği optimal çözümü bilinen küçük boyutlu problemler üzerinde gösterildikten sonra, orta ve büyük boyutlu problemler için çözüm sonuçları ve CPU zamanları sunulmuştur. Ayrıca, dikkate alınan problemlerin optimal çözümlerinin bulunması amacıyla yeni bir dal sınır algoritması geliştirilmiştir.

Bu algoritma gerek probleme özgü bilgilerden, gerekse de kullanılan dallandırma stratejisinin özelliklerinden hareketle geliştirilmiş etkin baskınlık ve uygunluk kriterleri içermektedir. Algoritmanın performansını belirleyen diğer bir etkin bileşeni de birerleme prosesinde kullanılan sezgisel tabanlı yönlendirme mekanizmasıdır. Literatürde benzer bir problemin çözümü için önerilen dal sınır algoritması, göz önünde bulundurulan problemi çözecek şekilde modifiye edilerek önerilen algoritma ile karşılaştırılmıştır. Sonuçların analizi yoluyla, gerek optimum çözümleri her iki algoritma tarafından bulunan problemler için CPU zamanları açısından, gerekse de her iki algoritmanın sadece uygun çözümlerini bulabildiği problemler için çözüm kalitesi açısından, önerilen algoritmanın daha etkin olduğu gösterilmiştir.

Bilim Kodu : 906.1.141
Anahtar Kelimeler : Montaj hattı dengeleme, Paralel çok işçili istasyonlar, Matematiksel modeller, Kurucu sezgiseller, Dal-sınır metotları
Sayfa Adedi : 139
Tez Yöneticisi : Prof. Dr. Bilal TOKLU

**NEW SOLUTION APPROACHES FOR ASSEMBLY LINE BALANCING
PROBLEMS WITH PARALLEL MULTI-MANNED WORKSTATIONS**

(Ph.D. Thesis)

Talip KELLEGÖZ

**GAZİ UNIVERSITY
INSTITUTE OF SCIENCE AND TECHNOLOGY**

May 2011

ABSTRACT

Assembly lines of big-sized complex products; like buses, trucks, helicopters etc., are very different from the lines studied in the literature. These products' manufacturing processes have a lot of tasks most of which have long task times. If they are produced in simple or two-sided assembly lines, then hundreds of stations are needed. This means that a long product flow time, a large area for establishment of the line, a high budget for the investment of equipments and tools in stations and several works in process are also required. In order to eliminate these disadvantages, assembly lines with parallel multi-manned workstations can be used to produce this type of products. In this study, these lines and one of their balancing problems are considered. After the problem definition and literature review, it is modeled as a mixed integer programming formulation. In the experimental study, it has been seen that only some of small-sized problems can be solved optimally by using this formulation. Therefore, after presenting a detailed study of heuristic methods, a new efficient constructive heuristic algorithm is proposed. The efficiency of the proposed heuristic method is verified in small-sized problems whose optimal solutions are found. For medium and big-sized problems, heuristic results and CPU times are presented. A branch and bound based enumeration algorithm is also proposed for solving considered problems optimally. This algorithm includes some efficient dominance and feasibility criteria which are developed based on

problem-specific knowledge and the branching scheme used. A heuristic-based guidance for enumeration process is the one other efficient component of the algorithm. The branch and bound algorithm which was proposed for a special version of the studied problem in the literature has been modified to tackle the studied problem, and it is compared with the proposed algorithm. Through an analysis of the results, it has been seen that the proposed algorithm has better performance than the other one in terms of CPU times for the problems whose optimal solutions have been found by both algorithm, as well as the quality of feasible solutions of the problems whose optimal solutions could not be found by both algorithms for a given period of time.

Science Code : 906.1.141
Key Words : Assembly line balancing, Parallel multi-manned stations, Mathematical modeling, Constructive heuristics, Branch and bound methods
Page Number : 139
Adviser : Prof. Dr. Bilal TOKLU

TEŐEKKÜR

Çalıőmalarım boyunca deęerli yardım ve katkılarıyla beni yönlendiren Hocam Prof. Dr. Bilal TOKLU'ya, yine kıymetli tecrübe ve bilgilerinden faydalandığım Tez İzleme Kurul Üyesi hocalarım Prof. Dr. Hadi GÖKÇEN ve Prof. Dr. Levent KANDİLLER'e, manevi destekleriyle beni hiçbir zaman yalnız bırakmayan ve eğitimim için her türlü fedakârlığı fazlasıyla gösteren eşim ve çocuklarıma sonsuz teşekkürlerimi sunarım.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT	vi
TEŞEKKÜR	viii
İÇİNDEKİLER	ix
ÇİZELGELERİN LİSTESİ	xii
ŞEKİLLERİN LİSTESİ	xiii
SİMGELER VE KISALTMALAR	xiv
1. GİRİŞ.....	1
2. MONTAJ HATTI Dengeleme Problemleri	4
2.1. Temel Kavramlar ve Terminoloji.....	4
2.2. Montaj Hattı Düzenlemede Temel Prensipler.....	8
2.3. Montaj Hattı Dengeleme Problemlerinin Sınıflandırılması.....	9
2.3.1. Öncelik diyagramına yönelik özellikler (α).....	9
2.3.2. İş istasyonu ve hattın yapısına yönelik özellikler (β).....	12
2.3.3. Amaçlara yönelik özellikler.....	15
2.4. Paralel Çok İşçili Montaj Hattı Dengeleme Problemleri.....	16
2.5. Negatif Zaman Bölgeleme Kısıtları	21
3. LİTERATÜR ARAŞTIRMASI	22
3.1. Paralel İstasyonlu Montaj Hattı Dengeleme Problemleri	23
3.2. Paralel Görevli Montaj Hattı Dengeleme Problemleri	28
3.3. Paralel Montaj Hattı Dengeleme Problemleri.....	29
3.4. Çift Taraflı Montaj Hattı Dengeleme Problemleri	31

Sayfa

3.5. Çok İşçili/Grup Tabanlı Montaj Hattı Dengeleme Problemleri.....	37
3.6. Paralel Çok İşçili Montaj Hattı Dengeleme Problemleri.....	40
4. PARALEL ÇOK İŞÇİLİ MONTAJ HATTI DENGELEME PROBLEMLERİNİN ÇÖZÜMÜ İÇİN KARIŞIK TAMSAYILI MATEMATİKSEL PROGRAMLAMA MODELİ	45
4.1. Kullanılan Notasyonlar ve Temel Modelleme Yaklaşımı	45
4.2. Toplam İstasyon Sayısının En Küçüklenmesi Probleminin Matematiksel Modeli.....	47
4.3. Toplam İşçi Sayısının En Küçüklenmesi Probleminin Matematiksel Modeli.....	49
4.4. Bazı Bölgeleme Kısıtları.....	50
4.4.1. İşçi uyumsuz görevler.....	51
4.4.2. İstasyon uyumsuz görevler	51
4.4.3. Aynı işçiye atanacak görevler	51
4.4.4. Aynı istasyona atanacak görevler.....	52
4.4.5. Negatif zaman bölgeleme kısıtları	52
4.5. Test Problemi Sonuçları	52
5. PARALEL ÇOK İŞÇİLİ MONTAJ HATTI DENGELEME PROBLEMLERİNİN ÇÖZÜMÜ İÇİN DAL-SINIR ALGORİTMASI TEMELİNE DAYANAN ÇÖZÜM YÖNTEMİ	56
5.1. Problem için Geçerli Olan Bazı Alt Sınırlar	56
5.2. Problemin Çözümü için Önerilen Dal-Sınır Algoritması	60
5.2.1. Başlangıç işlemleri	62
5.2.2. Dallandırma işlemi	62
5.2.3. Sınırlama işlemi	70
5.2.4. Budama işlemi.....	70
5.2.5. Arama işleminin sezgisel olarak yönlendirilmesi	79

	Sayfa
5.3. Örnek Uygulama	84
5.4. Test Problemi Sonuçları	90
6. PARALEL ÇOK İŞÇİLİ MONTAJ HATTI Dengeleme PROBLEMLERİNİN ÇÖZÜMÜ İÇİN KURUCU BİR SEZGİSEL ALGORİTMA.....	98
6.1. Sezgisel Yöntemler ve Hat Dengelemede Kullanımı	98
6.2. Önerilen Sezgisel Yöntem	102
6.2.1. Çözümün daraltılması.....	103
6.2.2. Görev atamasının yapılması.....	104
6.3. Örnek Uygulama	108
6.4. Test Problemi Sonuçları	110
7. SONUÇ VE ÖNERİLER.....	114
KAYNAKLAR.....	120
EKLER.....	127
Ek-1 Örnek Dal-Sınır Algoritması Uygulamasına İlişkin Kısmi Çözümlere ait Gant Diyagramları	128
ÖZGEÇMİŞ.....	139

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 4.1. BMHDP-1 için literatürde önerilen karşılaştırma problemleri	53
Çizelge 4.2. Matematiksel model çözüm sonuçları	54
Çizelge 5.1. Alt sınır örneğine ilişkin hesaplama verileri.....	59
Çizelge 5.2. Bulunan tam çözüme ilişkin istatistik bilgiler	89
Çizelge 5.3. Yöntemlerin çözüm durumları açısından karşılaştırılması	93
Çizelge 5.4. Problem grupları.....	93
Çizelge 5.5. Problem boyutu $0 < n \leq 25$ olan problemlere ait sonuçlar	94
Çizelge 5.6. Problem boyutu $25 < n \leq 50$ olan problemlere ait sonuçlar	94
Çizelge 5.7. Problem boyutu $50 < n \leq 75$ olan problemlere ait sonuçlar	94
Çizelge 5.8. Problem boyutu $75 < n \leq 100$ olan problemlere ait sonuçlar	95
Çizelge 5.9. Problem boyutu $100 < n \leq 150$ olan problemlere ait sonuçlar	95
Çizelge 5.10. Problem boyutu $150 < n$ olan problemlere ait sonuçlar	95
Çizelge 5.11. Her iki yöntemin de uygun çözüm bulabildiği ortak problemlere ilişkin sonuçlar	96
Çizelge 5.12. Her iki yöntemin de uygun çözüm bulabildiği $n > 150$ boyutlu ortak problemlere ilişkin sonuçlar	97
Çizelge 5.13. Her iki yöntemin de optimal çözüm bulduğu ortak problemlere ilişkin CPU zamanı (saniye) sonuçları	97
Çizelge 6.1. BMHDP-1 için literatürde önerilen öncelik kurallarından bazıları.....	100
Çizelge 6.2. Öncelik kurallarının karşılaştırılması	111
Çizelge 6.3. Çözüm daraltma stratejilerinin karşılaştırılması	112
Çizelge 6.4. En iyi 4 öncelik kuralının stratejiler bazında karşılaştırılması	113
Çizelge 6.5. Optimal bilinen problemler için önerilen sezgiselin çözüm sonuçları.....	113

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Örnek öncelik diyagramı	5
Şekil 2.2. Örnek öncelik matrisi	6
Şekil 2.3. Tek, karışık ve çok modelli montaj hatları	10
Şekil 2.4. Seri ve U-tipi hat yerleşimi.....	14
Şekil 2.5. Paralel çok işçi montaj hatlarına ilişkin bir gösterim.....	16
Şekil 2.6. Problemin paralel işçili montaj hattı olarak uygun çözümlerinden birisi ..	17
Şekil 4.1. Matematiksel modele ilişkin şekilsel gösterim.....	46
Şekil 5.1. Örnek kısmi çözüme ait Gant diyagramı.....	58
Şekil 5.2. Verilen bir alt ağaçta uygun/optimal çözüm arayan prosedür	61
Şekil 5.3. Verilen bir dal sınır ağacında aramanın durdurulmasını sınavan fonksiyon.....	61
Şekil 5.4. Dallandırma işlemine ilişkin prosedür	68
Şekil 5.5. Mevcut düğümün budanıp budanmayacağını kontrol eden prosedür	72
Şekil 5.6. Geri dönüş (backtracking) işlemine ilişkin prosedür	73
Şekil 5.7. Önerilen sezgisel güdümlü dal sınır algoritması.....	83
Şekil 5.8. Örnek probleme ait öncelik diyagramı ve görevlerin $KumPA_i$ değerleri ...	84
Şekil 5.9. Örnek probleme ait öncelik diyagramının yeniden düzenlenmiş hali.....	85
Şekil 5.10. Örnek probleme ait dal-sınır ağacı (Kök düğüm numarası: 0)	86
Şekil 6.1. Önerilen sezgiselin ana akış şeması	103
Şekil 6.2. Örnek öncelik diyagramı	105
Şekil 6.3. Hu [74] sezgiseliyle yapılan atamanın Gant diyagramı	105
Şekil 6.4. Önerilen atama algoritmasının akış diyagramı	106
Şekil 6.5. Önerilen algoritma ile yapılan atamanın Gant diyagramı	107
Şekil 6.6. Başlangıç uygun çözümünün Gant diyagramı gösterimi	108
Şekil 6.7. Birinci çözüm daraltma işlemi sonucu ulaşılan çözümün Gant diyagramı gösterimi	109
Şekil 6.8. Üçüncü çözüm daraltma işlemi sonucu ulaşılan uygun çözümün Gant diyagramı gösterimi	109

SİMGELER VE KISALTMALAR

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

Simgeler	Açıklama
n	Problemdeki toplam görev sayısı
t_i	Problemdeki i . görevin görev zamanı
S_k	k . istasyona atanan görevlerin seti, yani istasyon yükü
$t(S_k)$	S_k 'da bulunan görevlerin toplam görev zamanı, yani k . istasyon zamanı
c	Çevrim zamanı
t_{Toplam}	Problemde toplam görev zamanı, $\sum_{i=1}^n t_i$
t_{Enk}	Problemdeki en küçük görev zamanı
t_{Enb}	Problemdeki en büyük görev zamanı
k_{Enb}	Maksimum istasyon sayısı
m_{Enb}	Her hangi bir istasyonda bulunabilecek maksimum işçi sayısı
TI	Negatif zaman bölgelemesi gerektiren görev çiftleri seti
M	Çok büyük sabit bir sayı
MM	Çok çok büyük sabit bir sayı
E_i	Problemdeki i . görevin atanabileceği en erken istasyon
L_i	Problemdeki i . görevin atanabileceği en geç istasyon
ES_i	Problemdeki i . görevin başlayabileceği en erken zaman
LS_i	Problemdeki i . görevin başlayabileceği en geç zaman
F_i	Problemdeki i . görevi direkt takip eden görevlerin seti
F_i^*	Problemdeki i . görevi takip eden tüm görevlerin seti
A	$j > i$ ve $j \in F_i$ olmak üzere (i, j) görev çiftleri seti
A^*	$j > i$ ve $j \in F_i^*$ olmak üzere (i, j) görev çiftleri seti
P_i	Problemdeki i . görevinin direkt takip ettiği görevlerin seti
P_i^*	Problemdeki i . görevi takip ettiği tüm görevlerin seti
UR_i	Problemdeki i göreviyle öncelik ilişkisi olmayan, eş zamanlı olarak yapılabilir görevlerin seti
II	Aynı işçiye atanması istenmeyen görev çiftlerinin oluşturduğu set

IS	Aynı işçiye atanması istenen görev çiftlerinin oluşturduğu set
SI	Aynı istasyona atanması istenmeyen görev çiftlerinin oluşturduğu set
SS	Aynı istasyona atanması istenen görev çiftlerinin oluşturduğu set
TI	Aynı zaman biriminde eş zamanlı olarak yapılması istenmeyen görev çiftlerinin oluşturduğu set
B_i	i görevinin başlama zamanı
k_i	i görevinin atandığı istasyon ($k_i = 1, 2, \dots, k_{Enb}$)
$k_{toplama}$	Toplam istasyon sayısı
m_i	i görevinin atandığı işçi ($m_i = 1, 2, \dots, m_{Enb}$)
os_l	l istasyonundaki toplam işçi sayısı ($l = 1, 2, \dots, k_{Enb}$; $os_l = 1, 2, \dots, m_{Enb}$)
AK_1	İstasyon sayısı alt sınır değeri
AS_1	İşçi sayısı alt sınır değeri 1
AS_2	İşçi sayısı alt sınır değeri 2
AS_3	İşçi sayısı alt sınır değeri 3
γ_{il}	i görevi l istasyonu veya sonrasına atanmışsa 1, değilse 0 değeri alan 0-1 değişken.
PA_i	Problemdeki i . görevin pozisyon ağırlığı
MI_i	Problemdeki i . görevin atanabileceği mümkün istasyon sayısı
r_i	Problemdeki i . görevi hemen takip eden önceliklerin sayısı
r_i^*	Problemdeki i . görevi birikimli takip eden önceliklerin sayısı
mc_{p-q}	$1 \leq p \leq q \leq k_{Enb}$ olmak üzere $p, p+1, \dots, q-1, q$ istasyonlarında bulunan ve en az 1 adet görev atanmış işçilerin sayısı
\bar{t}_i	Problemdeki i . görevin düzeltilmiş görev zamanı
tt_{p-q}	$1 \leq p \leq q \leq k_{Enb}$ olmak üzere $p, p+1, \dots, q-1, q$ istasyonlarına ataması yapılmış veya ilerleyen düğümlerde kesinlikle ataması bu istasyonlara yapılacak olan görevlerin \bar{t}_i değerleri toplamı
$CPU_{AğaçEnb}$	Dal sınır alt ağacında izin verilen maksimum arama süresi
$CPU_{Ağaç}$	Dal sınır alt ağacındaki arama süresi
$CPU_{ProbEnb}$	Problem için izin verilen maksimum arama süresi
CPU_{Prob}	Problem için arama süresi
$Sonuç_{Ağaç}$	Dal sınır alt ağacındaki arama sonucu
$Sonuç_{Prob}$	Problem için arama sonucu
ÇözümBulunamadı	Arama işleminde henüz uygun bir çözümün bulunamadığını belirten ifade
$Optimum\text{Çözüm}$	Arama işleminde problemin optimal çözümünün bulunduğu

	belirten ifade
<i>UygunÇözüm</i>	Arama işleminde probleme uygun bir çözüm bulunduğunu belirten ifade
<i>Çözümsüz</i>	Aranan bölgede problemin hiçbir uygun çözümünün bulunamadığını belirten ifade
<i>MevcutDüğüm</i>	Arama işlemindeki mevcut (geçerli) düğüm
$KumPA_i$	Problemdeki i . görevin birikimli pozisyon ağılık değeri
<i>EnbKumP/</i>	Problemdeki görevlerin $KumPA_i$ değerlerine göre büyükten küçüğe sıralanarak numaralarının belirlenmesi
<i>Seviye</i>	Dal sınır düğümünün seviyesi
G	Dal sınır düğümünde atanan görev
mK	Dal sınır düğümünde atamanın yapıldığı istasyon
mM	Dal sınır düğümünde atamanın yapıldığı işçi
mZ	Dal sınır düğümünde atanan mM işçisinin G 'ye başlama zamanı yani atama zamanı
AS	(Düğüm) alt sınır değeri
$İY_m$	Dal sınır düğümünde atamanın yapıldığı mK istasyonundaki m işçisine atanmış görevlerin toplam zamanı
$İZ_m$	Dal sınır düğümünde atamanın yapıldığı mK istasyonundaki m işçisine yapılabilecek bir sonraki atama zamanı
$İB_m$	Dal sınır düğümünde atamanın yapıldığı mK istasyonundaki m işçisinin atandığı en son görevi tamamlama zamanı yani boş kalma zamanı
yZ	Dal sınır düğümünün yavrusunda yapılacak atamanın atama zamanı, yani mevcut zaman
yK	Dal sınır düğümünün yavrusunda yapılacak atamanın istasyonu (Bir sonraki görev atamasının yapılacağı istasyon)
yM	Dal sınır düğümünün yavrusunda yapılacak atamanın işçisi
AL	Dal sınır düğümünün yavrusunda atanabilecek görevlerin listesi
<i>yIndeks</i>	Dal sınır düğümünün en son oluşturulan yavrusunda atanan görevin AL 'deki sırası
$K_{Uygunluk}$	Dal sınır düğümünde uygunluk kontrolü yapıp yapılmadığı gösteren ifade
$K_{Baskınlık}$	Dal sınır düğümünde baskınlık kontrolü yapıp yapılmadığı gösteren ifade
K_{AS}	Dal sınır düğümünün AS değerinin hesaplanıp hesaplanmadığını gösteren ifade
<i>enbG</i>	Kök düğümünden mevcut dal sınır düğümüne kadar ataması yapılan maksimum görev indeksi
<i>topG</i>	Kök düğümünden mevcut dal sınır düğümüne kadar ataması yapılan görevlerin toplam süresi

<i>topM</i>	Kök düğümünden mevcut dal sınır düğümüne kadar görev atanmış işçi sayısı
<i>Yavru</i>	Dal sınır düğümünün yavru düğüm referansı
<i>Ebeveyi</i>	Dal sınır düğümünün ebeveyn düğüm referansı
<i>GA</i>	Dal sınır ağacı kapsamında tutulan görev atamaları dizisi
<i>İşçiYük</i>	Dal sınır ağacı kapsamında işçilere atanan toplam yük miktarını tutan $m_{Enb} \times k_{Enb}$ boyutunda matris
<i>İşçiSayı</i>	Dal sınır ağacı kapsamında istasyon işçi sayılarının yönlendirilmesi amacıyla tutulan k_{Enb} boyutunda dizi
<i>İlkİşçiAtama</i>	Dal sınır ağacı kapsamında işçilere ilk görev atamasının hangi düğümde yapıldığını tutan $m_{Enb} \times k_{Enb}$ boyutunda matris
<i>ÜS</i>	Global üst sınır değeri
<i>Seviye_{Enk}</i>	Dal sınır ağacı kapsamında aranması gereken en küçük seviyeli düğüm
<i>ED</i>	Sezgisel yönlendirme için işçi etkinliği eşik değeri
<i>AS_{teorik}</i>	Problemin başlangıç teorik alt sınır değeri

Kısaltmalar**Açıklama**

BS	Becker ve Scholl [15] tarafından DA-MHDP'in çözümü için önerilen dal-sınır algoritması
BMHDP	Basit MHDP
BMHDP-1	Toplam istasyon sayısının en küçüklenmesi performans ölçütüne sahip BMHDP
ÇMHDP	Çift taraflı MHDP
DA-MHDP	Değişken çalışma alanlı montaj hattı dengeleme problemleri
KKPPP	Kaynak kısıtlı proje planlama problemi
KMHDP	Karışık modellenli MHDP
MHDP	Montaj hattı dengeleme problemi
Pİ-MHDP	Paralel çok işçili MHDP
Pİ-MHDP1	Toplam işçi sayısının en küçüklenmesi performans ölçütüne sahip Pİ-MHDP
PMÇP	Paralel makine çizelgeleme problemi

1. GİRİŞ

Üretim, insanlığın var oluşundan beri vardır. Nüfus artışı, buna bağlı olarak talepteki yükseliş ve küreselleşen dünya; pazar ve rekabet sınırlarını ortadan kaldırmış, üretim işletmelerinin mevcudiyetini sürdürebilmesinin ön koşulu yüksek kalite ve miktarlardaki üretimi daha düşük maliyetle pazara sunmak olmuştur. Yüksek miktar ve düşük maliyet hedefine, ilk uygulamasını 19. yüzyılın başlarında Henry Ford'un yaptığı montaj hatlarının kullanılmasıyla ulaşılabilmektedir. Günümüz dünyası sanayi devriminin yaşandığı bu yıllardan çok farklıdır. Gelişen teknolojinin yanı sıra bitmeyen müşteri istek ve beklentileri ürün talebinin de kendi içerisinde farklılaşmasına neden olmuş, yüksek kalite ve düşük maliyetin yanı sıra farklı müşteri tiplerine hitap eden ürünün değişik varyasyonlarının ihtiyaç duyulan tam zamanda pazara sunulması gerekmiştir. Bu ihtiyaca cevap verebilmek için sanayici ve araştırmacılar U-tipi yerleşim, karışık modelli montaj hatları gibi çeşitli üretim metotları ile tepkisel üretim, tam zamanında üretim gibi çeşitli üretim felsefelerini hataya geçirmişlerdir.

Montaj hattı dengeleme problemi (MHDP), üretim sistemlerinde sık karşılaşılan bir gerçek hayat problemi olması nedeniyle birçok akademisyen ve uygulayıcının çalışmalarına konu olmuş ve servis sistemlerinin de dâhil olduğu çok çeşitli alanlara uygulanabilen çeşitli çözüm yöntemleri geliştirilmiştir. MHDP'in görevleri arasındaki öncelik ilişkilerinin kaldırılmasıyla elde edilen özel bir versiyonu NP-zor olduğu bilinen kutu paketleme problemine dönüştüğünden, MHDP'de NP-zor problemler sınıfının bir üyesidir [1].

MHDP konusunda ilk analitik çalışmanın Salveson [2] tarafından yayınlanmasından beri bu alanda birçok çalışma yapılmış olmasına karşın önerilen yöntemlerin gerçek montaj hatlarında kullanılma oranı sınırlı kalmıştır. Erel ve Sarin [1] bu durumun temel nedenlerinin mevcut yöntemlerin yetersiz olması ve/veya gerçek montaj hatlarına uygulama imkânı verecek düzeyde esnek olmaması ve/veya uygulayıcıların yayınlanan algoritmalarından haberdar olmaması şeklinde olabileceğini belirtmiştir.

Gelişen teknolojinin diğer bir sonucu yüzlerce görevden oluşan karışık ürünlerin üretimini mümkün hale getirmesidir. Örneğin otobüs gibi boyutu büyük, görev sayısı fazla ve görev süreleri uzun olan ürünlerin üretiminin her istasyonda tek işçi bulunan klasik montaj hatlarında yapılması durumunda yüzlerce istasyon bulunan bir montaj hattına, buna bağlı olarak da ciddi boyutta bir alana ihtiyaç duyulacaktır. Ürünün yapısı göz önünde bulundurulduğunda istasyonlardan birinde aracın motor aksamının montajı yapılırken koltuk, direksiyon, far gibi bölümlerinin montajına izin verilmemesi hattın ekonomikliği ve ürünün hattaki akış süresi açısından da verimsiz bir yaklaşım olacaktır. Basit montaj hatlarında ürünün hattaki akış süresi toplam görev zamanından küçük olamayacağı için pazardaki talep değişimlerine cevap verme süresi de uzayacaktır. Günümüze kadar bu tip ürünleri üreten işletmeler iş istasyonlarında birden fazla işçinin çalışmasına izin vererek ve deneme-yanılma yoluyla hattı dengeleyerek üretimlerini yapmaktaydı. Son birkaç yıl içerisinde bu tip hatlarda ilk bilimsel çalışmalar yapılmış olmasına rağmen gerek yerleşim, gerek işlem zamanlarının değişken olma durumu gibi problem çeşitlerine, gerekse de diğer montaj hattı dengeleme modellerindeki çözüm yaklaşımlarına bakıldığında zaman yapılan çalışmaların yetersiz olduğu görülmektedir. Bu nedenle bu çalışma kapsamında her hangi bir istasyonunda, aynı parça üzerinde, eş zamanlı olarak farklı montaj görevlerini yapan birden fazla işçinin bulunmasına izin verilen problemler göz önünde bulundurulmuş, literatürde tespit edilen boşluk belirli ölçüde doldurulmaya çalışılmıştır.

Çalışmanın sunumu şu şekilde organize edilmiştir. İkinci bölüm montaj hatları ve hat dengeleme problemi ile ilgili tanım ve terminolojiyi içermektedir. Aynı bölüm kapsamında montaj hattı dengelemede temel prensipler, MHDP'in sınıflandırılması, bu çalışmanın konusu olan paralel çok işçili montaj hattı dengeleme probleminin (Pİ-MHDP) tanımı ve aynı iş istasyonunda eş zamanlı çalışan birden fazla işçinin bulunmasına izin verilmesi sonucu ortaya çıkan yeni bir bölgeleme kavramı ele alınmıştır.

Üçüncü bölümde, Pİ-MHDP ve bu probleme tanımsal açıdan benzerlik gösteren diğer montaj hattı dengeleme problemlerini de kapsayacak şekilde bir literatür araştırması verilmektedir.

Dördüncü bölümde ise toplam işçi sayısının en küçüklenmesi performans ölçütlü Pİ-MHDP'in (Pİ-MHDP1) çözümü için karışık tamsayılı doğrusal programlama modeli geliştirilmiş, çeşitli bölgeleme kısıtlarının bu modele nasıl dâhil edilebileceğinden bahsedilmiştir. Yine aynı bölüm kapsamında karşılaştırma problemleri üretilmiş ve bu problemler kullanılarak önerilen matematiksel modelin performansı test edilmiştir.

Beşinci bölümde ise matematiksel modelin etkisiz kaldığı orta boyuttaki problemlerin optimal çözümlerinin bulunmasında kullanılabilecek, problemin yapısından hareketle geliştirilmiş, dal-sınır tabanlı birerlemeye dayanan bir kesin çözüm algoritması sunulmuştur. Aynı bölüm kapsamında literatürde Pİ-MHDP1'e benzer bir problemin çözümü için önerilmiş başka bir algoritma Pİ-MHDP1'i çözecek şekilde modifiye edilerek önerilen algoritma ile karşılaştırılmış ve sonuçlar detaylı bir şekilde analiz edilerek sunulmuştur.

Kurucu sezgisel algoritmalar optimali garantilemeksizin çok büyük boyutlu problemlere kabul edilebilir sürede yine kabul edilebilir kalitede çözümler üretebilmektedir. İşte çalışmanın altıncı bölümünde, dört ve beşinci bölümlerde önerilen kesin çözüm yöntemlerinin yetersiz kaldığı durumlarda Pİ-MHDP1'in çözümü için kullanılabilecek sezgisel bir algoritma sunulmuştur. Matematiksel modelin çözüm sonuçlarından faydalanılarak önerilen sezgisel yöntemin performansı test edilmiş, sezgiselin büyük boyutlu karşılaştırma problemlerine ilişkin çözüm sonuçları da ayrıca sunulmuştur.

Yedinci ve son bölümde ise yapılan çalışma ve ulaşılan sonuçlar değerlendirilmiş, bu çalışmadan elde edilen tecrübeler doğrultusunda gelecekte yapılabilecek çalışmalardan bahsedilmiştir.

2. MONTAJ HATTI DENGELEME PROBLEMLERİ

2.1. Temel Kavramlar ve Terminoloji

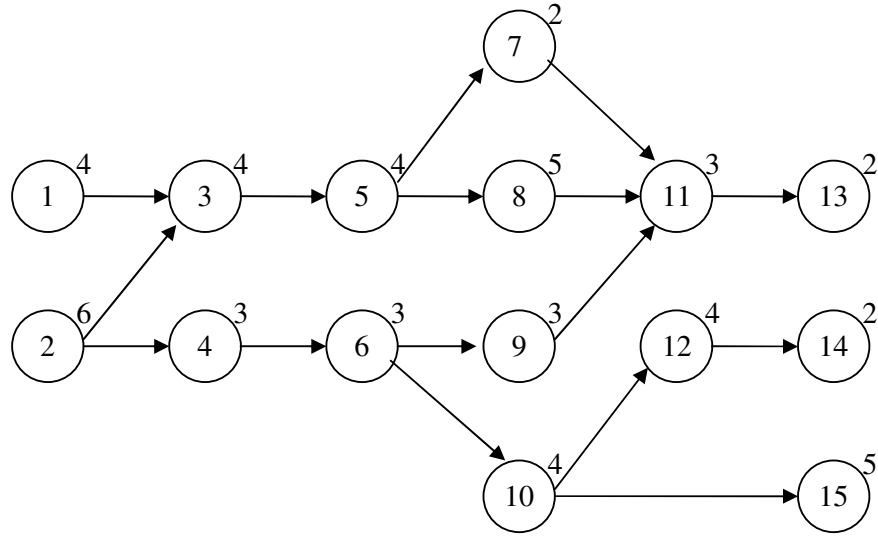
Nihai ürünü üretmek için ürünü oluşturan bileşen, alt montaj ve parçaların birleştirilmesi işlemi montaj olarak tanımlanır [3]. Seri üretim tiplerinden biri olan montaj hatları, montaj işlemleri yoluyla standart ürünlerin yüksek miktarlarda endüstriyel üretimini gerçekleştirmek için kullanılır. Montaj hatları konveyör veya benzer bir malzeme taşıma ekipmanı boyunca sıralanmış iş istasyonlarından oluşur [4]. Belirli bir zaman aralığıyla hattın başlangıcından serbest bırakılan parçalar hat boyunca ilerlerken her bir istasyonda bulunan işçi kendine verilen bir veya birkaç görevi montaj işlemleri yoluyla kendi istasyonunda bulunan parça üzerinde gerçekleştirir. Bu yolla hattın sonuna kadar ilerleyerek tüm görevleri tamamlanan parçalar nihai ürün olarak hattı terk ederler. MHDP açısından önemli olduğu düşünülen bazı kavramlar aşağıda sırasıyla açıklanmıştır.

Görev ve görev zamanı

Toplam iş yükünün gereksiz ek iş oluşturmadan bölünemeyen en küçük parçasıdır. Görevin yapılması için gereken süre görev zamanı (t_i) olarak adlandırılır [5].

Öncelik diyagramı, öncelik matrisi

Teknolojik kısıtlar dâhilinde bazı görevlerin birbirini takip etmesi gerektiğinden, görevlerin yaklaşık olarak hangi sıra ile yapılması gerektiği önceden belirlenebilir. Bu kısmi sıralama öncelik diyagramı vasıtası ile gösterilir. Öncelik diyagramında her düğüm bir görevi temsil eder ve her görev, ardılı ile bir ok aracılığıyla bağlanır. Okun çıktığı görev, öncül görevdir. Düğümlerin sağ üst kısımlarında, o görevin görev süresi gösterilir [6]. Şekil 2.1'de 15 görevli bir problemin öncelik diyagramı görülmektedir.



Şekil 2.1. Örnek öncelik diyagramı [5]

Görevler arasındaki öncelik ilişkilerini göstermenin diğer bir yolu ise öncelik matrisinin kullanılmasıdır. Öncelik matrisi bir üst üçgen matris olup öncelik diyagramında j görevi i görevini takip ediyorsa matrisin i . satır j . kolonuna 1, aksi halde 0 yazılarak oluşturulur [1]. Şekil 2.1’de verilen öncelik diyagramına ait öncelik matrisi Şekil 2.2’de sunulmuştur.

Esneklik katsayısı, sıralama gücü

Problemin öncelik yapısını ifade eden esneklik katsayısı, üretilebilecek uygun görev sıralarının sayısına ilişkin bir ölçüt olup öncelik matrisindeki 0 sayısının, n problemdeki toplam görev sayısı olmak üzere $n \times (n-1)/2$ değerine bölünmesiyle elde edilir [1].

Sıralama gücü ise öncelik diyagramı içerisindeki öncelik ilişkilerinin bağıl sayısını verir. Yüksek sıralama gücüne sahip problemlerin düşük sıralama gücü olanlara kıyasla daha karmaşık olması beklenir. Öncelik matrisindeki 1 sayısının $n \times (n-1)/2$ değerine bölünmesi ile elde edilir. Değeri 0-1 arasında değişir. Görevler arasında tek bir sıra olanaklı ise 1, öncelik ilişkisi söz konusu değilse 0 değerini alır [6].

		Ardıl görevler														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Öncül görevler	1	-		1		1		1	1			1		1		
	2		-	1	1	1	1	1	1	1	1	1	1	1	1	1
	3			-		1		1	1			1		1		
	4				-		1			1	1	1	1	1	1	1
	5					-		1	1			1		1		
	6						-			1	1	1	1	1	1	1
	7							-				1		1		
	8								-			1		1		
	9									-		1		1		
	10										-		1		1	1
	11											-		1		
	12												-		1	
	13													-		
	14														-	
	15															-

Şekil 2.2. Örnek öncelik matrisi

İş istasyonu, istasyon zamanı

Toplam iş yükünü oluşturan tüm görevlerin belirli bir alt setinin yapıldığı montaj hattı parçası iş istasyonu olarak adlandırılır. Bir iş istasyonuna atanan görev setine istasyon yükü (S_k), bu setin gerçekleştirilebilmesi için ihtiyaç duyulan zamana istasyon zamanı ($t(S_k)$) denir [5].

Çevrim zamanı

Hattaki her bir parça için her bir istasyona verilen maksimum zaman miktarı çevrim zamanı olarak adlandırılır ve c ile gösterilir. Çevrim zamanı ihtiyaç duyulan üretim oranına göre belirlenir [7].

İstasyon boş zamanı, toplam boş zaman (denge gecikmesi)

Çevrim zamanı ile her hangi bir istasyonun istasyon zamanı arasındaki farka istasyon boş zamanı denir. Hattın toplam boş zamanı ise hattaki tüm istasyonların boş

zamanları toplamına eşit olup, bu ifade aynı zamanda denge gecikmesi olarak da bilinir. Sabit bir çevrim zamanı için hattın boş zamanının en küçüklenmesi, hattaki toplam istasyon sayısının en küçüklenmesine eşdeğerdir [7].

Hat kapasitesi, hattın etkinliği

Hattaki toplam istasyon sayısı k olmak üzere hattın kapasitesi $T = k \cdot c$ ifadesiyle hesaplanır. Bu değer toplam görev zamanından (t_{Toplam}) küçük olamaz. Hattın etkinliği ise $E = t_{Toplam}/T$ formülüyle hesaplanmakta olup, toplam görev zamanının hat kapasitesine oranı olarak ifade edilir [5].

Düzensizlik indeksi

Toplam iş yükünün istasyonlara eşit olarak paylaşılıp paylaşılmadığının bir ölçütü olup, hattaki toplam istasyon sayısı k olmak üzere Eş. 2.1 ile hesaplanır [5].

$$DI = \sqrt{\sum_{i=1}^k (c - t(S_i))^2} \quad (2.1)$$

Bölgeleme kısıtları

Bu kısıtlar iki şekilde ifade edilebilir. Bunlardan ilki pozitif bölgeleme kısıtlarıdır ve burada belirli görevlerin aynı yerde veya aynı istasyonda yapılması gerekliliği vardır. İkincisi negatif bölgeleme kısıtlarıdır ki bu kısıtlar belirli görevlerin aynı istasyonda yapılmaması istenildiğinde kullanılırlar [8].

Gecikmeli (unpaced) hatlar, gecikmesiz (paced) hatlar

Gecikmeli hatlarda istasyon ilgili faaliyetlerini tamamladığında birim (iş parçası) bir sonraki istasyona iletilir. Görev zamanlarındaki değişkenliğin çok fazla olduğu durumlarda bu hatlar tercih edilirler. Gecikmesiz hatlarda ise her bir istasyona aynı

zaman miktarı (çevrim zamanı) izin verilir. Her bir çevrim zamanından sonra birim tamamlansın ya da tamamlanmasın bir sonraki istasyona iletilir [8].

Montaj hattı dengeleme problemi

Montaj işleminin yapılabilmesi için gerekli görevler, bu görevlerin yapılması için gerekli süreler ve aralarındaki öncelik ilişkileri verildiğinde, işlerin belirlenen bir performans ölçütünü en iyileyecek şekilde sıralı iş istasyonlarına atanması problemi montaj hattı dengeleme problemi olarak tanımlanmaktadır [9].

2.2. Montaj Hattı Düzenlemede Temel Prensipler

Montaj hatlarının düzenlenmesi aşağıdaki prensiplerin en iyi uyumuna göre yapılır [8]:

- *İşgücü prensibi*: Montaj hattında hangi miktarda bir işgücüne veya işçiye gereksinim vardır, bunlar nerelerde ve ne kadar bulundurulacaktır sorularına cevap aranır.
- *İş akışı prensibi*: İşlemler belli bir iş akışını sağlamalıdır. Bu durum süreklilik açısından da önemlidir.
- *En küçük hareket uzaklığı prensibi*: Montaj hattı üzerindeki taşıma mesafeleri; oluşturacağı maliyet, darboğaz ve depolama açısından, en küçük olmalıdır.
- *Sabit yol prensibi*: İşlemler son aşamaya kadar aynı yol üzerinde hareket etmelidir.
- *İşlemlerin eş zamanlı sürdürülmesi prensibi*: Montaj işlemleri ilk istasyondan son istasyona kadar eş zamanlı yapılır.
- *Parçaların değişebilirliği prensibi*: Montaj işlemi sırasında bazı parçalar işlem sürekliliğini bozmadan değiştirilebilmelidir.
- *Birim işlem prensibi*: Ürün üzerinde her işlem, en küçük işlem birimlerine ayrılmıştır.
- *En küçük işlem süresi*: Montaj işlemi, mümkün olan en kısa sürede

bitirilmelidir. Burada işlem süreleri sabit olduğuna göre en küçükleme kayıp zaman üzerinde yapılır.

2.3. Montaj Hattı Dengeleme Problemlerinin Sınıflandırılması

Montaj hattı dengeleme problemlerinin sınıflandırılması konusunda son zamanlarda Boysen ve ark. [10] detaylı bir çalışma yaparak bu problemlerin $[\alpha | \beta | \gamma]$ şeklinde bir gösterimle 3 temel kategoride sınıflandırılmasının uygun olacağı sonucuna varmışlardır. Bu sınıflandırmada α parametresi öncelik diyagramına ait özellikleri belirtirken, β parametresi istasyon ve hat yapısına ait özellikleri, γ parametresi ise amaçlara yönelik özellikleri ifade etmektedir. Her bir parametre ilgili problemi tanımlamak amacıyla birden fazla değer alabilmektedir. Şimdi sırasıyla bu özellikler incelenecek ve bu özellikler altında hangi problem sınıflarının yer alabileceğinden bahsedilecektir.

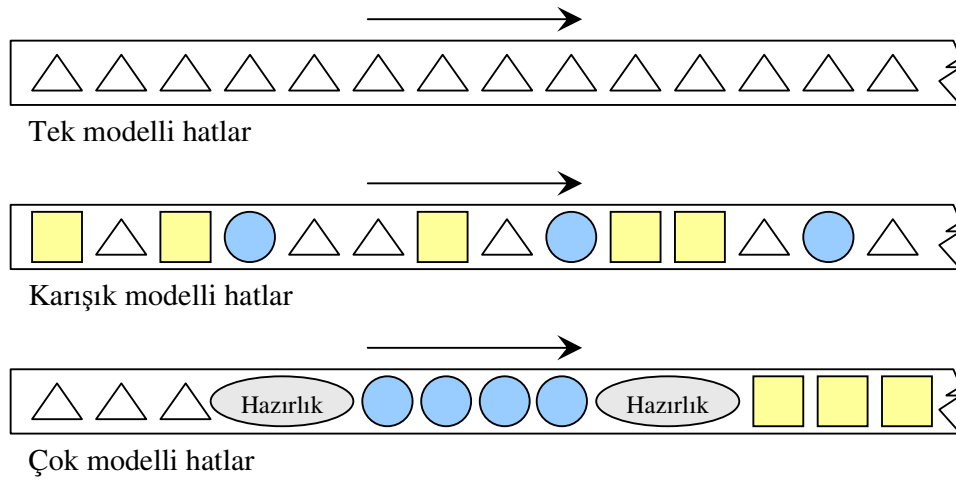
2.3.1. Öncelik diyagramına yönelik özellikler (α)

Ürün (α_1)

- Tek modellenli hatlar ($\alpha_1 = o$)
- Karışık modellenli hatlar ($\alpha_1 = mix$)
- Çok modellenli hatlar ($\alpha_1 = mult$)

Tek modellenli hatlarda aynı tipten tek bir ürün yüksek miktarlarda üretilir (kitle üretimi). Her bir istasyon kendisine atanan görev veya görevleri birbirinin aynı olan her bir parça için tekrarlar. Karışık modellenli hatlarda ise temel ürünün değişik versiyonları aynı hat üzerinde, güncel talebi karşılamak için karışık olarak üretilir. Üretim sürecinin benzer olmasından dolayı, model değişimi olduğunda hazırlık işlemi ya gerekmez veya çok az düzeyde yapılır. Bu tip hatlarda dengeleme probleminin yanı sıra kısa dönemli model sıralama problemi de söz konusudur. Çok

modelli hatlarda ise birbirine benzeyen birden çok ürün üretilir. Ürünlerdeki farklılık ürün değişiminde hattaki ekipmanların yeniden düzenlenmesini gerektirir. Bu nedenle ortaya çıkan hazırlık işlemlerinin en küçüklenmesi için ürünler partiler halinde üretilir. Parti büyüklükleri arttıkça stokta tutma maliyetleri de artar. Her bir ürünün üretim çevrimlerinin yanı sıra ekonomik parti büyüklükleri ve partilerin hattaki üretim sıraları belirlenmelidir. Şekil 2.3’de belirtilen hatlara ilişkin bir gösterim verilmiştir [5].



Şekil 2.3. Tek, karışık ve çok modelli montaj hatları

Yapı (α_2)

- Doğrusal, yakınsayan, ıraksayan gibi özel yapıli öncelik diyagramları ($\alpha_2 = spec$)
- Öncelik diyagramında özel bir yapı yok ($\alpha_2 = o$)

Doğrusal öncelik diyagramlarında görevler birbiri ardına sıralanır, yani her bir görevin tek bir ardılı ve tek bir öncülü vardır [11]. Bazı problemlerde bu yapı görevlerin sadece bir alt seti için söz konusu olmaktadır. ıraksayan öncelik diyagramlarında son düğümlere yaklaşıldıkça görevlerin direkt ardıllarının sayısı artmaktadır. Bu tip yapının söz konusu olduğu problemlerde başlangıç istasyonlarına

atanabilecek görev sayısı düşükken, son istasyonlara doğru bu sayı artmaktadır [12]. Yakınsayan öncelik diyagramlarında ise bu yapı tam tersidir.

Görev zamanları (α_3)

- Stokastik görev zamanları ($\alpha_3 = t^{sto}$)
- Öğrenme etkisi gibi nedenlerle dinamik görev zamanları ($\alpha_3 = t^{dyn}$)
- Statik-deterministik görev zamanları ($\alpha_3 = o$)

Sıra bağımlı görev zamanları (α_4)

- Görev zamanlarının bir önceki pozisyonda yapılan göreve bağlı değişmesi ($\alpha_4 = \Delta t_{dir}$)
- Görev zamanlarının göreceli önceki pozisyonda yapılan göreve bağlı değişmesi ($\alpha_4 = \Delta t_{ind}$)
- Sıra bağımlı görev zamanı yok ($\alpha_4 = o$)

Atama kısıtları (α_5)

- Bazı görevlerin aynı istasyona birlikte atanması (pozitif bölgeleme) ($\alpha_5 = link$)
- Bazı görevlerin aynı istasyona birlikte atanamaması (negatif bölgeleme) ($\alpha_5 = inc$)
- Görevlerin istasyonlara atanmasını belirli görev özelliklerinin belirlemesi ($\alpha_5 = cum$)
- Bazı görevlerin atanacağı istasyonun bilinmesi ($\alpha_5 = fix$)
- Bazı görevlerin atanamayacağı istasyonların bilinmesi ($\alpha_5 = excl$)
- Bazı görevlerin belirli özellikte olan istasyonlara atanabilmesi ($\alpha_5 = type$)

- Bazı görevler arasında minimum uzaklık veya zaman kısıtının bulunması ($\alpha_5 = \min$)
- Bazı görevler arasında maksimum uzaklık veya zaman kısıtının bulunması ($\alpha_5 = \max$)
- Atama kısıtı yok ($\alpha_5 = 0$)

İşlem alternatifleri (α_6)

- İşlem alternatiflerinin sadece zaman ve maliyette değişime sebep olması ($\alpha_6 = pa$)
- İşlem alternatiflerinin zaman ve maliyetin yanı sıra görevler arasındaki önceliği de değiştirmesi ($\alpha_6 = pa^{prec}$)
- İşlem alternatiflerinin öncelik diyagramının belirli bir bölümünü tamamen değiştirmesi ($\alpha_6 = pa^{subgraph}$)
- İşlem alternatif yok ($\alpha_6 = 0$)

2.3.2. İş istasyonu ve hattın yapısına yönelik özellikler (β)

Ürün akışı (β_1)

- Gecikmesiz hatlar ($\beta_1 = \lambda v$)
 - Tüm parçalar için ortalama istasyon zamanı çevrim zamanıyla sınırlı ($\lambda = 0$)
 - İstasyon zamanı her bir model için belirli bir çevrim zamanıyla sınırlı ($\lambda = each$)
 - Stokastik görev zamanlı problemlerde çevrim zamanını sağlamak için bir olasılık değeri verilir ($\lambda = prob$)
 - Tüm istasyonlar için çevrim zamanı aynı ($v = 0$)
 - İstasyonlar arası çevrim zamanı farklılık gösterebilir ($v = div$)

- Gecikmeli hatlar ($\beta_1 = unpac^\lambda$)
 - Parça biter bitmez ya sonraki istasyona veya ara stoka gönderilen senkronizesiz hatlar ($\lambda = 0$)
 - Tüm istasyonlarda yapılan işlem bitince parçaların hareket ettiği senkronizeli hatlar ($\lambda = sync$)

Gecikmesiz hatlarda her bir parça için, atanan görevleri yerine getirmeleri amacıyla istasyonlara verilen çevrim zamanları birbirine eşittir. Parça ya konveyör üzerinde sabit bir hızla istasyonlar arasında ilerler veya çevrim zamanı bittikçe aynı anda bir sonraki istasyona gönderilir. Gecikmeli hatlarda ise görevlerini tamamlayan istasyonlar için sonraki istasyon meşgul olması halinde tamamlanan parçayı gönderebilecekleri ara stok alanları vardır [5].

Hat yerleşimi (β_2)

- Tek bir U-tipi hat ($\beta_2 = u$)
- n adet U-tipi hattın aynı anda göz önünde bulundurulması ($\beta_2 = u^n$)
- Seri düzenlenmiş hat (düz hat) ($\beta_2 = 0$)

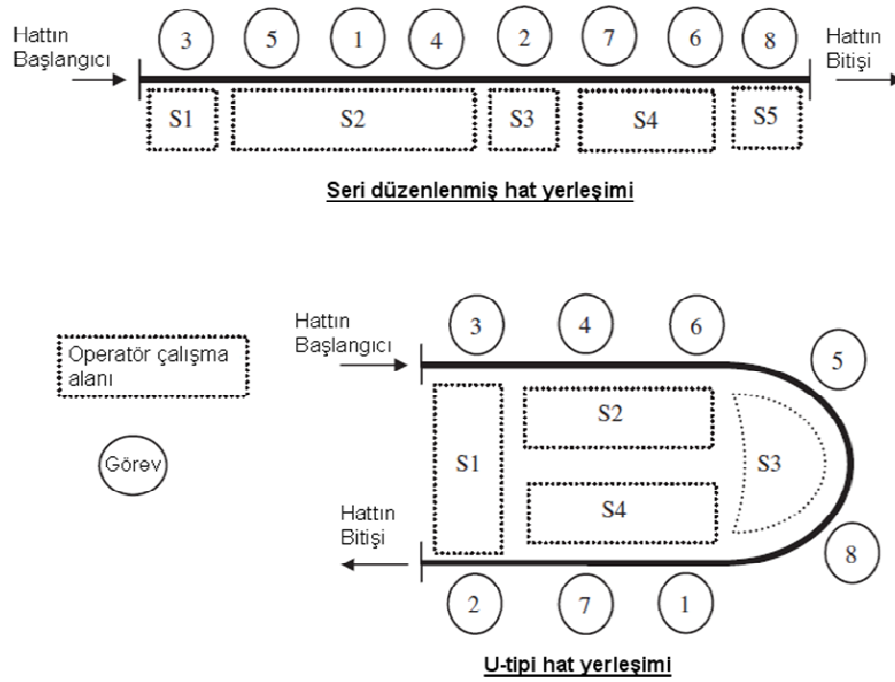
Şekil 2.4'den de görüldüğü gibi seri düzenlenmiş hatlarda işçi, atanan görevleri hattın sürekli bir parçası olan istasyonda yerine getirebilirken, U-tipi hatlarda ise yerleşim U şeklinde olup işçinin hattın karşı tarafına geçip oradaki görevleri de yerine getirmesine izin verilir. Yani hattın farklı parçalarındaki görevler aynı istasyonda gruplanabilmektedir. Montaj hatlarında bu yapının hem istasyon sayısını azaltıcı hem de işgücü verimliliğini artırıcı bir etkisi vardır [13].

Parallelleştirme (β_3)

- Birden fazla paralel hattın birlikte dengelenmesi veya hat sayısının karar probleminin bir parçası olması ($\beta_3 = pline^\lambda$)

- Hatta paralel istasyonların bulunmasına izin verilmesi ($\beta_3 = pstat^\lambda$)
- Hatta paralel görevlerin bulunmasına izin verilmesi ($\beta_3 = ptask^\lambda$)
- Birden fazla işçinin aynı parça üzerinde çalışması ($\beta_3 = pwork^\lambda$)
- Paralleleştirme yok ($\beta_3 = o$)

Paralleleştirmeye ilişkin detaylı açıklamaya literatür araştırmasına ilişkin bilgilerin sunulduğu Bölüm 3 kapsamında yer verilmiştir.



Şekil 2.4. Seri ve U-tipi hat yerleşimi

Kaynak ataması (β_4)

- Bir istasyon için önceden belirlenmiş kaynak setinden bir kaynağın seçilmesi ($\beta_4 = equip$)
- İstasyonun sahip olacağı kaynağı atan görevlerin belirlenmesi ($\beta_4 = res^\lambda$)
- Kaynak ataması yok ($\beta_4 = o$)

İstasyon bağımlı zaman artımı (β_5)

- İstasyon zamanının bir kısmının yürüme, taşıma gibi faaliyetlere ayrılması ($\beta_5 = \Delta t_{ump}$)
- İstasyon bağımlı zaman artımı yok ($\beta_5 = 0$)

Hat tasarımına ilişkin ilave durumlar (β_6)

- Ara stok alanlarının bulunması ($\beta_6 = buffer$)
- Ana hattı besleyen bir veya daha fazla sayıda besleyici (feeder) hattın bulunması ($\beta_6 = feeder$)
- Malzeme taşıma kutularının bulunması ($\beta_6 = mat$)
- Bazı görevlerin yapılabilmesini sağlamak amacıyla istasyonlara parçaya pozisyon aldırın (örneğin, parçanın kaldırılması gibi) ekipmanların atanması ($\beta_6 = change$)
- İlave durum yok ($\beta_6 = 0$)

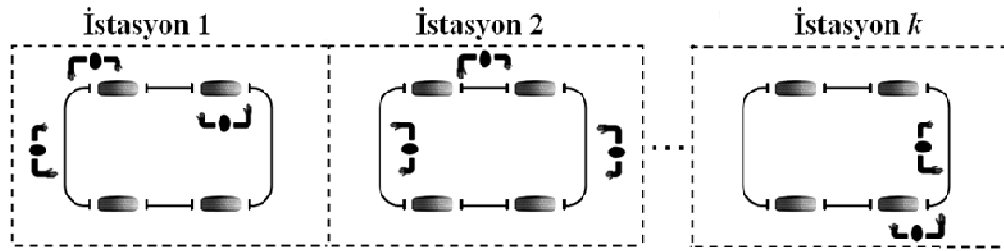
2.3.3. Amaçlara yönelik özellikler

- Verilen çevrim zamanı kısıtı altında istasyon sayısının en küçüklenmesi ($\gamma = m$)
- Verilen istasyon sayısı kısıtı altında çevrim zamanının en küçüklenmesi ($\gamma = c$)
- İstasyon sayısı ve çevrim zamanına ilişkin alt ve üst sınır değerleri verildiğinde hat etkinliğinin en büyüklenmesi ($\gamma = E$)
- Maliyetin en küçüklenmesi ($\gamma = Co$)
- Gelir-gider farkının yani karın en büyüklenmesi ($\gamma = Pr$)
- Farklı modellerdeki istasyon zamanının düzgünleştirilmesi ($\gamma = SSL^{stat}$)

- Hattaki tüm istasyon zamanlarının düzgünleştirilmesi ($\gamma = SSL^{line}$)
- Parça pozisyon değişimi gibi etkinliği belirleyen özelliklerin en küçüklenmesi veya en büyüklenmesi ($\gamma = score$)
- Amaç uygun çözümlerin bulunması ($\gamma = o$)

2.4. Paralel Çok İşçili Montaj Hattı Dengeleme Problemleri

Özellikle otomotiv sanayi olmak üzere birçok endüstride montajı yapılan ürün, üzerinde eş zamanlı olarak birden fazla işçinin çalışmasına olanak verecek ölçüde büyük boyutlara sahiptir. Bu olanağın sağlanmaması montaj hattının ve ürün temin zamanının çok uzun olmasına, üretkenliğin azalmasına sebep olacaktır [14]. Bu hatların her bir istasyonunda aynı parça üzerinde eş zamanlı olarak farklı montaj görevlerini yapan birden fazla işçi bulunur. Hattın başlangıcından serbest bırakılan ürün, işçilere atanan görevlerin yapılması amacıyla çevrim zamanı boyunca hattaki her bir istasyonda bekler ve bu zaman sonunda bir sonraki istasyona ilerler [7]. Bu hatlara ilişkin bir gösterim Şekil 2.5’de verilmiştir.

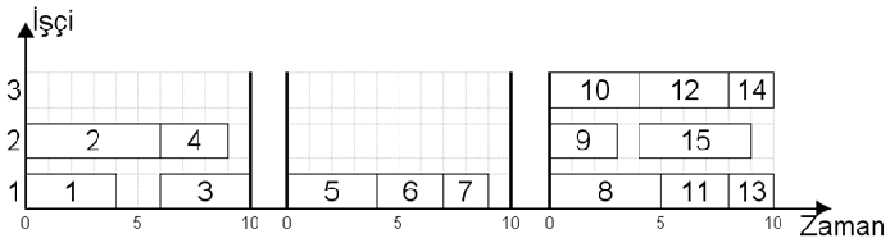


Şekil 2.5. Paralel çok işçili montaj hatlarına ilişkin bir gösterim

Dimitriadis [7], belirtilen sistemin kullanılabilmesi için parçanın, işçilerin birbirini engellemeyecek şekilde yeterince büyük boyutlara sahip olması gerektiğini ifade etmiştir. Bu modelin belirtilen koşul sağlanması durumunda daha kısa uzunluğa sahip hatlar oluşturabileceğini, bunun yanı sıra ara stok miktarını azaltacağını, aynı istasyonda çalışan işçilerin aparat paylaşımı kolaylaşacağı için aparat maliyetinin azalacağını ifade etmiştir. Paralel çok işçili montaj hatlarının birden fazla işçinin aynı görev üzerinde çalıştığı çok işçili/grup tabanlı montaj hatlarından ve paralel istasyon

bulunduran hatlardan farklı olduğunu belirtmiştir. Ayrıca, ilgili modelin çift taraflı hatlara benzediğini fakat çift taraflı hatlarda sadece 2 işçi bulunması ve pozisyon ve bölgeleme kısıtlarının söz konusu olması nedeniyle o hatlardan da ayrıldığını vurgulamıştır. Dimitriadis [7] ayrıca, göz önünde bulundurulacak problemin akademik literatürde hiç çalışılmadığını da vurgulamıştır.

Hattın daha iyi açıklanabilmesi amacıyla $n = 15$, $c = 10$, $t_{Toplam} = 54$ olan ve öncelik diyagramı Şekil 2.1’de verilen problemi göz önünde bulunduralım. Problemin paralel çok işçili montaj hattı olarak uygun çözümlerinden birisine ait Gant diyagramı Şekil 2.6’de sunulmuştur.



Şekil 2.6. Problemin paralel işçili montaj hattı olarak uygun çözümlerinden

Diyagrama dikkat edilirse çözümde 3 istasyon söz konusu olup istasyonlarda sırasıyla 2, 1 ve 3 adet işçi bulunmaktadır. Birinci istasyonun ilk işçisine 1 ve 3 görevleri, ikinci işçisine 2 ve 4 görevleri atanmıştır. İkinci istasyonun tek işçisine ise 5, 6 ve 7 görevleri atanmıştır. Son istasyonda bulunan işçilerden ilkinde 8, 11 ve 13 görevleri, ikincisine 9 ve 15 görevleri ve sonuncusuna da 10, 12 ve 14 görevleri atanmıştır. İkinci istasyonda tek bir işçi söz konusu olduğu için işçinin görevler arasında beklemesi gerekmemekte, tüm görevleri öncelik ilişkisi sağlanacak şekilde kesintisiz yapmaktadır. İlk ve son istasyonlarda ise durum farklıdır. İlk istasyonun birinci işçisi 1. görevi tamamlamış fakat 3. göreve başlayabilmesi için bu görevin öncülü olan 2 görevini diğer işçinin bitirmesini beklemiştir. Benzer durum son istasyonun ikinci işçisine atanan 15. görev için de söz konusudur. Yani, Pİ-MHDP’de öncelik ilişkileri nedeniyle işçilerin boş beklemesi söz konusu olabilmektedir. Dolayısıyla görevlerin istasyonlara atanmasının yanı sıra istasyonda

hangi işçi tarafından hangi zaman biriminde başlanarak yapılacağına da belirlenmesi gerekmektedir. Verilen problemin basit montaj hattı olarak ele alınması durumunda problemin kutu paketleme problemine indirgenmesiyle elde edilen toplam istasyon sayısı alt sınır değeri $AS_1 = [t_{\text{toplam}} / c]^+ = [54/10]^+ = 6$ istasyondur [5]. Bu problemde her bir istasyonda tek bir işçi bulunabileceğinden ihtiyaç duyulan işçi sayısının alt sınır değeri de 6'dır. Buna karşın şekilden de anlaşılacağı gibi Pİ-MHDP olarak ele alındığında aynı sayıda işçinin 3 istasyona atanmasıyla hat dengelenebilmektedir. Yani alan gereksinimi azalmaktadır. Ayrıca düz hatlarda parçanın akış süresi alt sınır değeri 60 iken incelenen problemde 30 olarak gerçekleşmektedir. Benzer şekilde basit montaj hattına kıyasla hem ara stok miktarı hem de aparat paylaşımı nedeniyle aparat yatırım maliyeti azaltılmış olacaktır.

Becker ve Scholl [15] göz önünde bulundurduğu özel bir Pİ-MHDP problemi için her hangi bir istasyonda bulunabilecek maksimum işçi sayısı ve maksimum istasyon sayısının alan sınırlaması nedeniyle bilindiğini varsayarak, çevrim zamanı ve toplam işçi sayısı parametrelerine bağlı olarak aşağıdaki dört problemin söz konusu olduğunu belirtmiştir:

1. Verilen çevrim zamanında işçi sayısının en küçüklenmesi,
2. Verilen işçi sayısında çevrim zamanının en küçüklenmesi,
3. Çevrim zamanı ve işçi sayısının hat etkinliğini en büyükleyecek şekilde belirlenmesi,
4. Verilen işçi sayısı ve çevrim zamanı için uygun bir dengenin araştırılması.

Düz montaj hatlarında işçi sayısı ve istasyon sayısının birbirine eşit olması nedeniyle işçi sayısının en küçüklenmesi bir biriyle doğru orantılı olacak şekilde hem istasyon sayısını hem de hattın kurulması için gereken alan ihtiyacını en küçüklemektedir. Buna karşın göz önünde bulunduran problemde işçi sayısı ve istasyon sayısı arasında doğrudan bir ilişki söz konusu değildir. Dolayısıyla işçi sayısının en küçüklenmesi alan ihtiyacını doğru orantılı olarak en küçüklemeyeceği için amaç fonksiyonu olarak alan ihtiyacını da içeren modelin tanımlanması gerekir. Bu durum

çift taraflı montaj hatlarında da ortaya çıkmaktadır. Bu nedenle Wu ve ark. [16], Özcan ve Toklu [17, 18] tarafından yapılan çalışmalarda birincil amaç alan ihtiyacının diğer bir ifadeyle hat uzunluğunun en küçüklenmesi, ikincil amaç ise işçi sayısının en küçüklenmesi olan çift taraflı montaj hattı dengeleme problemleri göz önünde bulundurulmuştur.

Çift taraflı montaj hatlarında her bir istasyonda en fazla iki işçi söz konusu olabilmekle birlikte görevlerin hangi tip işçiye atanacağı da belirlidir. Buna karşın göz önünde bulundurulan problemlerde işçi sayısı açısından böyle bir sınırlama söz konusu olmayıp, görevlerin atanmasında da işçi tipiyle ilgili her hangi bir kısıtlama yoktur. Bu nedenle, her hangi bir istasyonda bulunabilecek maksimum işçi sayısı da farklı bir amaç olarak ortaya çıkar. Her hangi bir istasyonda eşzamanlı çalışan işçi sayısının artmasıyla ara stok miktarı ve aparat paylaşımı nedeniyle aparat maliyeti azalırken, işçi sayısının azalmasıyla istasyonda ortaya çıkabilecek karmaşa da azalacaktır. Belirtilen açıklamalar ışığında probleme ilişkin amaçlarda yer alabilecek farklı parametreler aşağıdaki gibi tanımlanabilir:

1. İşçi sayısının en küçüklenmesi: Bu yolla hat etkinliğinin artırılması amaçlanmaktadır.
2. Çevrim zamanının en küçüklenmesi: Bu yolla hat etkinliğinin artırılması amaçlanmaktadır.
3. İstasyon sayısının (yani alan ihtiyacının veya hat uzunluğunun) en küçüklenmesi: Bu yolla ihtiyaç duyulan alan miktarının azaltılması amaçlanmaktadır.
4. İstasyonda bulunabilecek maksimum işçi sayısının (yani hat genişliğinin) en küçüklenmesi: Bu yolla işçi sayısından kaynaklanabilecek karmaşanın azaltılması amaçlanmaktadır.

Belirtilen açıklamalar ışığında Pİ-MHD problemlerinin temel varsayımları aşağıdaki şekilde ifade edilebilir:

1. Her hangi bir görev tek bir işçi tarafından yapılabilir.

2. Her hangi bir işçi birim zamanda tek bir görevi yapabilir.
3. İstasyonlarda birden fazla işçinin bulunmasına izin verilir.
4. İstasyonların işçi sayısı açısından bir birinden farklı olmasına izin verilir.
5. Her bir istasyonda gerekli tüm donanımlar mevcuttur.
6. Verilen çevrim zamanı süresince, aynı istasyonda çalışan işçiler istasyonda bekleyen aynı birim üzerinde kendilerine atanan görevleri eş zamanlı olarak gerçekleştirirler.
7. Görevler öncelik ilişkilerini sağlamak zorundadır. Aynı istasyonda farklı işçiler tarafından eş zamanlı olarak yapılan görevlerin öncelik ilişkisini sağlama zorunluluğu özellikle vurgulanmalıdır.

Bu çalışma kapsamında göz önünde bulundurulan ve çalışma kapsamında Pİ-MHDP1 olarak ifade edilen problemin varsayımları ise yukarıda belirtilen temel Pİ-MHDP varsayımlarına ilave olarak aşağıdaki şekilde ifade edilebilir:

1. Tek bir homojen ürünün (yani tek modelin) kitle üretimi söz konusudur.
2. Sabit çevrim zamanlı gecikmesiz hat vardır.
3. Görev zamanları deterministik olup görevlerin bölünmesine izin verilmez.
4. Öncelik ilişkileri dışında başkaca atama kısıtı bulunmaz.
5. İstasyonlar seri yerleştirilmiştir.
6. Çevrim zamanının istenen çıktı oranına göre belirlenmesi yoluyla bilindiği varsayılır.
7. İşçilerin görevleri yapması için ihtiyaç duyacağı seyahat zamanlarının ihmal edilebilecek düzeyde küçük olduğu ve görev zamanlarına eklendiği varsayılır.
8. Her bir istasyon ihtiyaç duyulan tüm donanım ve ekipmanları içerir.
9. En büyük alan kısıtı nedeniyle en büyük istasyon sayısı (k_{Enb}) bilinmemektedir.
10. Parçanın özellikleri ve istasyon yapısı nedeniyle her hangi bir istasyonda bulunabilecek en büyük işçi sayısının (m_{Enb}) bilindiği varsayılır.
11. Amaç, belirtilen kısıtlar altında ihtiyaç duyulan işçi sayısının en küçüklenmesidir.

2.5. Negatif Zaman Bölgeleme Kısıtları

Örneğin bir otobüs üretiminin direksiyon ve sürücü koltuğu montajlarını göz önünde bulunduralım. Bu görevlerin yapıldığı montaj pozisyonunda iki işçinin aynı anda çalışması söz konusu olamayacağından ya bu görevler aynı işçiye atanmalı veya farklı işçilere atanmışsa önce direksiyon montajı daha sonra sürücü koltuğu montajı yapılmalı ya da sürücü koltuğu montajından sonra direksiyon montajı yapılmalıdır. Yani, bu iki operasyon aynı montaj pozisyonunu gerektirdiği için aynı anda yapılamamalıdır. Bu durum esasında büyük boyutlu ürünlerin montajının yapıldığı hatlarda pratikte sıkça karşılaşılan bir durumdur. Belirtilen yeni yapının modellenebilmesi amacıyla montaj pozisyonları açısından aynı anda yapılamayacak zaman uyumsuz görev çiftlerini içeren bir setin (TI) olduğu varsayılmaktadır. Bu listenin elemanlarında bulunan görev çiftleri arasında öncelik ilişkisi bulunmaması gerekir. Öyle ki, öncelik söz konusu ise zaten görevlerin aynı anda yapılması mümkün değildir. Bu yaklaşım probleme yeni bir bölgeleme kısıtı olarak ilave edilebilir. Belirtilen bölgeleme kısıtı “Negatif Zaman Bölgeleme Kısıtı” olarak isimlendirilmiştir.

3. LİTERATÜR ARAŞTIRMASI

Montaj hatları ilk defa 1913 yılında Hanry Ford tarafından geliştirilip ABD'deki Ford firmasında kullanılmış olmasına rağmen 40 yıldan fazla bir süre hat dengeleme çalışmalarında sadece deneme-yanılma yöntemleri kullanılmıştır [1]. Montaj hattı dengeleme problemleri konusundaki ilk analitik çalışma Salveson [2] tarafından yayınlanmış olup, bu çalışmada istasyon sayısının en küçüklendiği tek modelli basit montaj hattı dengeleme problemi (BMHDP-1) tanımlanarak, ayrık istasyonlu çözümle sonuçlanabilecek bir doğrusal programlama formülasyonu sunulmuştur. Aynı problem konusunda pratik olarak uygulanabilecek ilk yöntem ise Jackson [19] tarafından geliştirilen dinamik programlama algoritmasıdır [20]. Yine aynı problemin ilk karışık tamsayı programlama modeli Bowman [21] tarafından geliştirilmiş, bu model daha sonra White [22] tarafından atamalar 0-1 değişkenlerle ifade edilecek şekilde iyileştirilmiştir [23]. NP-zor sınıfında bulunan montaj hattı dengeleme problemlerinin [24] en basit yapısına sahip olan BMHDP-1'in bölümlenme problemine (partition problem) nasıl indirgeneceğine ilişkin detaylı bilgi Queyranne [25]'in çalışmasında bulunabilir. Bunların yanı sıra montaj hattı dengeleme problemleri konusunda pek çok çalışma yapılmış olup, önerilen model ve yöntemler Baybars [24], Erel ve Sarin [1], Scholl ve Becker [26], Becker ve Scholl [27] ve Boysen ve ark. [28] tarafından yapılan literatür taraması çalışmalarında detaylı bir şekilde ele alınarak karşılaştırılmıştır.

Son dönemlerde Dimitriadis [7] tarafından ilk kez incelenip daha sonra Becker ve Scholl [15] tarafından yeniden ele alınan, her bir istasyonda birden çok işçinin bulunup aynı parça üzerindeki farklı görevleri paralel olarak yapmasına izin verilen yeni bir problem sınıfı ortaya çıkmıştır. Belirtilen model otobüs, kamyon, ekskavatör gibi büyük boyutlu ürünlerin montajının yapıldığı üretim sistemlerinde pratikte çok kullanılmasına rağmen, akademik literatürde hemen hemen hiç ele alınmamıştır [7]. Bu nedenle literatür taramasında görevlerin paralel yapılmasına izin verilmesi açısından bahsi geçen hatlara benzerlik gösteren modellerin incelendiği çalışmalara yoğunlaşmış olup bu çalışmalar aşağıdaki gibi 6 kategoride ele alınmıştır:

- Paralel istasyonlu montaj hattı dengeleme problemleri,
- Paralel görevli montaj hattı dengeleme problemleri,
- Paralel montaj hattı dengeleme problemleri,
- Çift taraflı montaj hattı dengeleme problemleri,
- Çok işçili/grup tabanlı montaj hattı dengeleme problemleri,
- Paralel çok işçili montaj hattı dengeleme problemleri.

3.1. Paralel İstasyonlu Montaj Hattı Dengeleme Problemleri

Basit montaj hattı dengeleme probleminin (BMHDP) temel varsayımlarından birisi görevlerin bölünemeyeceği şeklindedir [5]. Bu varsayımın doğal sonucu olarak problemin içerdiği maksimum görev zamanı çevrim zamanı için bir alt sınır değeri olacaktır. Eğer çevrim zamanını geçen bazı görevler varsa, bu çelişki istasyonların paralelleştirilmesi yoluyla çözülebilir [27]. Paralel olarak yerleştirilmiş istasyonların her birisi aynı montaj görevlerini farklı parçalara uygular. Paralel iki istasyona gelen parça sıralı olarak istasyonlardan birisine alınarak, ilgili istasyonda biten parça hatta ilerlemesine devam eder. Bu şekilde paralelleştirilmiş istasyonların yerel çevrim zamanları global çevrim zamanının iki katı olacaktır [5].

Freeman ve Jucker [29], istasyonların paralelleştirilmesi yaklaşımını ilk kez ortaya atmış, paralel istasyon kullanılmasının paralelleştirme için ilave sabit maliyet gerektirmesine karşın bazı durumlarda toplam maliyeti azaltabileceğini belirtmiştir.

Buxey [30], paralel istasyonlu, denge gecikmesini en küçüklemeyi amaçlayan problemin çözümü için iki adet sezgisel algoritma önermiştir. Görev zamanları ve pozisyon önceliklerine göre her bir istasyonu sıralı olarak ayrı ayrı en iyileyen ilk sezgisel tek bir çözüm üretirken, ikinci sezgisel uygun görev listesinden rastsal seçim yapmak yoluyla birden fazla çözüm üretip en iyisini seçmektedir. Önerilen modelde her hangi bir seri istasyona boş zaman olmaksızın görev atanamıyorsa ilgili istasyon yerine paralel istasyon kullanımı yoluna gidilmektedir.

Pinto ve ark. [31] istasyonların paralelleştirilmesine izin veren bir model tanımlamış, görevlerin istasyonlara atanması ve bir istasyonun paralelleştirilip paralelleştirilmeyeceğinin kararlaştırılması amacıyla dal-sınır tabanlı bir algoritma geliştirmiştir. Algoritma seri istasyon, paralel istasyon ve henüz karar verilmemişler olmak üzere üç küme üzerinden kısmi çözümleri oluşturarak arama yapmaktadır. Problemin amaç fonksiyonu olarak paralel istasyonların kurulması için katlanılan sabit maliyet ve istasyon sayısı ile çevrim zamanına bağlı hesaplanan işgücü maliyetinden oluşan toplam maliyetin en küçüklenmesi göz önünde bulundurulmuştur.

Sarker ve Shanthikumar [32], Buxey [30]'in incelediği problemi ele almış ve iki safhadan oluşan sezgisel bir algoritma geliştirmiştir. Algoritmanın ilk safhasında uygun görev listesindeki en büyük görev zamanına sahip görev mevcut istasyona atanarak başlangıç çözümü oluşturulmaktadır. Bu safhada mevcut istasyona atanan herhangi bir görevin görev zamanı çevrim zamanından büyükse paralel istasyon kullanımına gidilmektedir. İkinci safhada ise iki görevin yer değiştirmesi veya tek bir görevin istasyon değiştirmesi yoluyla çözüm iyileştirilmektedir.

Bard [33], parçaların bir istasyondan diğerine taşınması için gereken zaman olarak tanımlanan ölü zaman, paralel istasyon ve paralel görevler içeren modeli göz önünde bulundurmuş, problemin istasyon ve görev paralelleştirme maliyetleriyle işgücü maliyetleri toplamını minimum yapmak amacıyla dinamik programlama tabanlı bir algoritma geliştirmiştir.

Inman ve Leon [34] yüz yüze paralel istasyonlarda rastsal arızalanma, tamir ve işlem zamanlarını içeren stokastik durumu incelemiş, paralel istasyonların beslenmesine ilişkin 4 politikayı benzetim çalışmasıyla karşılaştırmıştır.

Askin ve Zhou [35], paralel istasyonların bulunmasına izin veren ve paralel istasyonlarda kullanılacak ekipman maliyetlerini de içeren maliyet tabanlı, karışık modelli montaj hattı dengeleme probleminin (KMHD) doğrusal olmayan tamsayı programlama modelini geliştirmiş, problemin çözümü için birleştirilmiş öncelik

diyagramlı ortak modeli kullanan öncelik kuralı tabanlı sezgisel bir yöntem önermiştir.

McMullen ve Frazier [36], görev zamanlarını talep frekanslarıyla çarparak paralel istasyonlu stokastik KMHDP problemini paralel istasyonlu stokastik BMHDP'ye dönüştürmüş, yeni problemi geliştirdiği sezgisel bir yöntemle çözerek orijinal probleme çözüm üretmeye çalışmıştır. Geliştirilen sezgiselde 10 adım bulunmakta olup, literatürde daha önce önerilmiş 7 farklı yöntemden birisi kullanılarak uygun görev listesinden görev seçimi yapılmaktadır.

McMullen ve Frazier [37], stokastik görev zamanları, paralel istasyonlar ve iki farklı amaç fonksiyonuna sahip KMHDP'in çözümü için tavlama benzetimi tabanlı bir sezgisel önermiştir. Amaç fonksiyonlarından ilki işgücü ve ekipman maliyetlerinin en küçüklenmesi, diğeri ise istenen çevrim zamanını elde etme ölçüsünün en iyilenmesidir. Önerilen 8 adımlı algorithmada başlangıç çözümü rastsal olarak, komşu çözümler ise iki görevin istasyon değiştirmesi yoluyla, bu mümkün değilse tek bir görevin farklı bir istasyona geçmesi yoluyla üretilmektedir. Deneysel çalışmalarda, sezgiselin amaç fonksiyonu çevrim zamanı olduğunda iyi çözümler üretirken, maliyet olduğunda ortalama kalitede çözümler ürettiği sonucuna varılmıştır.

Süer [3], paralel istasyonlu birbirinden bağımsız montaj hatları içeren paralel montaj hattı dengeleme probleminin çözümü için 3 safhalı sezgisel bir yöntem önermiştir. Problemden amaç toplam işçi sayısını en küçükleyecek hat sayısını ve her bir hattın konfigürasyonunu belirlemektir. Sezgiselin ilk safhasında görevler istasyonlarda gruplanmakta, ikinci safhasında her bir aşama için farklı sayıda paralel istasyon göz önünde bulundurularak alternatif hat dengeleri tespit edilmekte, üçüncü safhasında ise geliştirilen bir matematiksel model yardımıyla işçi sayısını minimum yapan hat sayısı ve her bir hattın konfigürasyonu belirlenmektedir.

Simaria ve Vilarinho [38], paralel istasyonlar içeren, istasyon sayısının en küçüklenmesi ve istasyonlar arasında yük dengesinin sağlanmasını eş zamanlı göz önünde bulunduran montaj hattı dengeleme probleminin matematiksel modelini

geliştirmiş, çözümü için tavlama benzetimi tabanlı sezgisel bir yöntem önermiştir. Yaklaşımında hangi görev zamanından sonra istasyonun paralelleştirileceği kullanıcı tarafından belirlenebilmektedir.

Vilarinho ve Simaria [39], paralel istasyonlu, bölgeleme kısıtları bulunan, amaç fonksiyonu istasyon sayısının en küçüklenmesi olan KMHDP'ni göz önünde bulundurmuş, ikincil amaç olarak dikey ve yatay dengesizliklerin en küçüklenmesini almış ve problemin matematiksel modelini geliştirmiştir. İstasyonların paralelleştirilmesi, Simaria ve Vilarinho [38] tarafından önerilen yönteme göre görev zamanı çevrim zamanını aşan görevleri içeren istasyonların paralel olmasına izin verilmesi şeklinde yapılmaktadır. Problemin çözümü için iki safhali, tavlama benzetimi tabanlı bir sezgisel önerilmiş olup ilk safhada birincil amaç en iyilenmeye, ikinci safhada ise birinci safhanın çözümü temel alınarak ikincil amaç en iyilenmeye çalışılmaktadır. Önerilen yöntem PC-kamera üretimi yapan bir firmanın montaj hattı probleminde uygulanmış ve çözümün tatmin edici olduğu belirtilmiştir.

Bukchin ve Rubinovitz [40], paralel istasyonlu ekipman seçimini içeren montaj hattı tasarım problemini toplam istasyon sayısının en küçüklenmesi ve toplam maliyetin en küçüklenmesi olmak üzere iki farklı amaç fonksiyonunu göz önünde bulundurarak incelemiş, p adet paralel istasyonun tek bir istasyondaki temel ekipmanın p katı hızında olan bir ekipmana karşılık geldiği noktasından hareketle paralel istasyon probleminin ekipman seçim probleminin özel bir hali olduğunu göstermiştir. Bunun sonucu olarak da paralel istasyon problemlerinin çözümünde ekipman seçim problemleri için geliştirilen yöntemlerin değişiklik yapılmadan kullanılabileceğini belirtmiştir. Ekipman seçim problemi için daha önce geliştirilen bir dal-sınır algoritmasını paralel istasyon probleminde adapte etmiş, paralel istasyonlu ekipman seçimini içeren montaj hattı tasarım probleminin tam sayılı doğrusal programlama modelini geliştirmiştir.

Vilarinho ve Simaria [41], bölgeleme kısıtları ve paralel istasyonların bulunmasına izin veren, birincil amacın istasyon sayısının en küçüklenmesi, ikincil amacın istasyon yüklerinin düzgülendirilmesi olan KMHDP'in çözümü için karınca kolonisi

temeline dayanan sezgisel bir yöntem önermiştir. İstasyonların paralelleştirilmesi Simaria ve Vilarinho [38] tarafında önerilen yönteme göre görev zamanı çevrim zamanını aşan görevleri içeren istasyonların paralel olmasına izin verilmesi şeklinde yapılmaktadır. Karıncaların ifade ettiği çözümlerin kalitesine göre görev sayısı boyutunda olan kare matriste tutulan fenomen bilgisi güncellenmekte, yeni çözüm oluşturulurken de mevcut istasyona atanabilecek uygun görevlerden birisinin seçim işlemi fenomen bilgisine göre oluşturulan olasılıklara göre yapılmaktadır. Karşılaştırma çalışmaları sonucunda önerilen yöntemin Vilarinho ve Simaria [39] tarafından önerilen tavlama benzetimi sezgiselinden daha iyi çözümlere ulaştığı belirtilmiştir.

Ege ve ark. [42] her bir aşamada istenilen sayıda paralel istasyon kullanımına izin veren, performans ölçütü istasyonların açılma ve gerekli ekipmanlarla donatılma maliyetinin en küçüklendiği MHDP'nin çözümü için dal-sınır tabanlı biri kesin diğeri sezgisel olan iki farklı algoritma önermiştir. Başlangıç üst sınır değerinin belirlenmesinde Askin ve Zhou [35] tarafından önerilen sezgisel kullanılmakta, düğümler görev atamalarına karşılık gelmekte, her bir düğümde ilgili görevin mevcut aşamaya atanması (çevrim zamanı aşıyorsa paralel istasyon açılması) veya yeni aşama açılarak atanması kararları verilmektedir. Mümkün olduğu sürece ikinci kararın göz ardı edilmesi yoluyla sezgisel algoritma ortaya çıkartılmıştır. Deneysel çalışmalarda orta boyutlardaki problemlerin optimal çözümlerinin, büyük boyutlardaki problemlerin ise kaliteli çözümlerinin makul zamanda bulunabildiği sonucuna varılmıştır.

Akpınar ve Bayhan [43], bazı bölgeleme kısıtları bulunan, paralel istasyon kullanımına izin veren, KMHDP'i göz önünde bulundurmuş, problemin çözümü için genetik algoritma tabanlı sezgisel bir yöntem önermiştir. İlk 3 tanesi literatürde yer alan sezgiseller yardımıyla kalanı ise rastsal olarak üretilmiş popülasyondan başlayan aramada rulet çemberi seçim mekanizması, iki noktalı çaprazlama, seçilen rastsal noktadan sonrasının rastgele karıştırıldığı özel bir mutasyon operatörü kullanılmaktadır. Deneysel çalışma literatürden alınmış test problemleri üzerinde yapılmış olup, önerilen algoritma Vilarinho ve Simaria [39] tarafından önerilen

tavlama benzetimi algoritmasıyla karşılaştırılmış ve daha iyi performansa sahip olduğu sonucuna varılmıştır.

3.2. Paralel Görevli Montaj Hattı Dengeleme Problemleri

Çevrim zamanından uzun görev zamanına sahip görevlerin orijinaliyle aynı öncelik ilişkisine sahip olacak şekilde alt görevlere ayrıştırılabileceği varsayılır. Hattın etkinliğini artırmak ve istenen çevrim zamanını sağlayan uygun bir denge bulmak için bu paralel görevler farklı istasyonlara atanır. Buna karşın, görevlerin bölünemeyeceğinden dolayı orijinal görev her bir çevrim zamanında sadece tek bir istasyonda yapılır. Bu durumu sağlamak amacıyla belirtilen görev alternatif olarak ilgili istasyonların her birinde yapılır ve bu nedenle yerel çevrim zamanının çevrimden çevrime farklılık göstermesine izin verilir [27].

Pinto ve ark. [44] amaç fonksiyonu tesis ve işçilik maliyetleri toplamının en küçüklenmesi olan, çevrim zamanını aşan görevlerin eşit uzunluklu 2 alt göreve ayrışmasına izin veren ve paralel görevlerin ikiden fazla istasyona atanmasını engelleyen problemin karışık tamsayılı matematiksel modelini oluşturmuştur. Bu modelin çözümü için Pinto ve ark. [31] tarafından önerilen metoda çok benzeyen bir dal-sınır algoritması geliştirmiştir.

Bukchin ve Rabinowitch [45], KMHDP'nde ortak görevlerin farklı istasyonlara atanmasına ve paralel görev kullanılmasına izin vermiş, görev paralelleştirme maliyetlerini de içeren toplam maliyetin en küçüklenmesi amacıyla dal-sınır tabanlı bir algoritma geliştirmiştir. Geliştirilen algoritma öncelikle derinlik stratejisini kullanmaktadır. Algoritmanın yetersiz kaldığı büyük boyutlu problemlerin çözümü için ise önerilen dal-sınır algoritması temeline dayanan sezgisel bir yöntem geliştirilmiş, deneysel çalışmalar sonucunda sezgiselin optimale yakın sonuçlar verdiği belirtilmiştir.

3.3. Paralel Montaj Hattı Dengele Problemleri

Paralel montaj hattı dengeleme problemlerinde birbirine paralel çalışarak aynı veya farklı ürünleri üreten birden çok montaj hattı aynı anda dengelenmektedir. Hat dengelenirken herhangi bir hattaki istasyona kendi hattına ait görevlerin yanı sıra komşu hatlardaki görevler de atanabilmektedir. Bu yolla istasyon sayısı en küçüklenip hattın etkinliği artırılmaya çalışılmaktadır [46].

Paralel montaj hattı dengeleme problemleri ilk kez Gökçen ve ark. [46] tarafından göz önünde bulundurularak problemin çözümü için pasif durum sezgiseli, aktif durum sezgiseli ve hatlarda farklı çevrim zamanlarının söz konusu olması durumunda kullanılabilir bir sezgisel olmak üzere üç farklı sezgisel yöntem önerilmiş ve probleminin 0-1 tamsayı programlama modeli geliştirilmiştir. Aynı ürünün üretildiği paralel hatların dengelenmesi amacıyla önerilen pasif durum sezgiselinde literatürde basit montaj hattı problemleri için önerilmiş sezgiseller kullanılarak her bir hat bağımsız olarak dengelendikten sonra boş zamanı çevrim zamanının yarısına eşit veya daha büyük olan istasyonların yükü paralel hatta bulunan ilişkili istasyona aktarılmaktadır. Aktif durum sezgiseli COMSOAL tabanlı bir yapıya sahip olup farklı ürün veya modellerin üretildiği hatlar için geliştirilmiştir. Bu sezgiselde mevcut istasyona sadece kendi görevleri ve komşu hatların görevlerinden atama yapılabilmekte olup oluşturulan birden fazla çözümün en iyisi seçilmektedir. Üçüncü sezgiselde ise öncelikle çevrim zamanları temel alınarak öncelik diyagramı birleştirilmekte, daha sonra oluşan probleme aktif durum sezgiseli uygulanmaktadır.

Benzer ve ark. [47], paralel montaj hattı dengeleme problemlerinin en kısa yol tabanlı şebeke modelini oluşturmuştur. Bu modelde yönlendirilmiş oklar görevlerin istasyonlara mümkün atamaları, ok uzunlukları ise ilgili atamadaki istasyon boş zamanını ifade ederken, başlangıç düğümünden bitiş düğümüne uzanan her bir rota uygun hat dengelerinden birisine karşılık gelmektedir.

Scholl ve Boysen [4], daha önce Scholl ve Klein [48] tarafından BMHDP-1 için önerilen ve SALOME adı verilen dal-sınır algoritmasını modifiye ederek paralel montaj hattı problemlerinin çözümünde kullanmış, bu algoritmanın yetersiz kaldığı büyük boyutlu problemlerin çözümü için ise önerilen dal-sınır algoritması temeline dayanan sezgisel bir yöntem önermiştir.

Kara ve ark. [49] paralel montaj hattı dengeleme problemlerini incelemiş, Gökçen ve ark. [46]'nın önerdiği matematiksel modeli temel alarak kesin hedefler ve bulanık hedefler olması durumunda kullanılabilecek iki farklı hedef programlama modeli geliştirmiştir. Çelişen amaçlar olarak istasyon sayısı, çevrim zamanı ve istasyon yükleri olmak üzere üç farklı hedef öncelikleri verilen sıra olacak şekilde göz önünde bulundurulmuştur. Örnek problem üzerinde modellerin kullanımı açıklanmıştır.

Özcan ve ark. [50] karışık modelli paralel montaj hattı dengeleme problemlerini göz önünde bulundurmuş, performans ölçütü toplam istasyon sayısı ve düzgünlük indeksi olacak şekilde problem tanımını yaptıktan sonra çözümü için tavlama benzetimi tabanlı sezgisel bir algoritma önermiştir. Algoritmada çözüm, görev öncelikleri ve model sırasına karşılık gelen iki parçadan oluşmaktadır. Komşu çözüme ilerlemek amacıyla üretilen rastsal sayı önceden belirlenen değerden küçükse çözümün öncelik bölümü, değilse model sırası bölümü dikkate alınarak rastsal seçilen parçaların karşılıklı yer değiştirmesiyle komşu çözüm üretilmektedir. Yapılan deneysel çalışmalarda ulaşılan sonuçların problemin karmaşıklığı da göz önünde bulundurulduğunda başarılı sayılabileceği sonucuna varılmıştır.

Özbakır ve ark. [51] birinci amaç istasyonların boş zamanlarının en küçüklenmesi, ikinci amaç da hat etkinliğinin en büyüklenmesi olan çok amaçlı, çok ürünlü paralel montaj hattı dengeleme problemini göz önünde bulundurmuş, problemin çözümü için çoklu koloni tipi karınca kolonisi algoritması önermiştir. Algoritmanın deneysel sonuçları literatürde elde edilmiş diğer sonuçlarla karşılaştırılmış, kısa sürelerde kaliteli çözümlere ulaştığı belirtilmiştir.

3.4. Çift Taraflı Montaj Hattı Dengeleme Problemleri

Bu hatların bazı istasyonlarında hattın sağ ve sol tarafı olmak üzere karşılıklı iki istasyon çifti bulunur. Çiftlerden her birisi eş zamanlı olarak aynı parça üzerinde kendine atanan görevleri yerine getirir. Görevler, yerine getirileceği parça pozisyonuna göre sağ taraf görevleri, sol taraf görevleri, her iki tarafa da atanabilecek görevler (örneğin radyo montajı) ve her iki tarafta bulunan istasyonunda eşzamanlı olarak yerine getirmesi gereken görevler (örneğin arka koltuk montajı) olmak üzere 4 gruba ayrılır. Bu hatlarda dikkat edilmesi gereken durum, karşılıklı istasyonlara atanan görevlerin öncelik kısıtlarını sağlayacak şekilde gerçekleştirilmesi gerektiridir. Bu nedenle bazı durumlarda bir istasyon belirli bir göreve başlamak için karşı istasyonun öncül görevi bitirmesini bekleyebilir. Bu durum hatta boş zamanların ortaya çıkmasına neden olabilir [27].

Çift taraflı montaj hatları konusunda ilk çalışma Bartholdi [52] tarafından yapılmıştır. Çalışmada problem tanımı yapıldıktan sonra gerçek hayatta kullanılan bir hattın verileri sunulmuştur. Problemin çözümü için "ilk uygun pozisyon" adı verilen sezgisel bir algoritma önerilmiştir. Algoritma tüm öncülleri atanmış görevi mümkün olan ilk pozisyona atayarak çözüme gitmektedir. Çalışma kapsamında ayrıca sabit bir çevrim zamanı için çift taraflı montaj hatlarının en küçük istasyon sayısının tek taraflı hatlarına eşit veya daha küçük olacağı da gösterilmiş, kullanıcının bazı görevleri bazı istasyonlara sabitlemesine izin veren ve önerilen sezgiseli kullanan bir bilgisayar programı geliştirilmiştir.

Kim ve ark. [53] çift taraflı montaj hatlarında istasyon sayısının en küçüklenmesi için genetik algoritma tabanlı bir sezgisel önermiştir. Önerilen algoritmada istasyon tabanlı kodlama yapısı kullanılmış, kodlamadan görev atamaları bulunurken sezgisel bir algoritmadan faydalanılmış, başlangıç çözümleri basit montaj hatları için literatürde daha önce önerilmiş olan çeşitli görev seçim kurallarının rastsal uygulanması yoluyla oluşturulmuştur. Seçim işleminde turnuva seçim operatörü, çaprazlama ve mutasyon işlemleri için ise problem özelliklerine göre oluşturulan operatörler kullanılmıştır. Yapılan deneysel çalışmalar sonucunda yöntemin verilen

istasyon sınırları içerisinde her zaman uygun bir denge bulunduğu ifade edilmiş ve çözüm zamanının kabul edilebilir olduğu vurgulanmıştır.

Lee ve ark. [54], çift taraflı montaj hatları için iki farklı amaç tanımlamıştır. İlk amaç direkt ilişkili görevleri aynı istasyona atamaya çalışırken, ikinci amaç ilişkili görevleri karşılıklı istasyonlara atamaktan kaçınmakta, bu durum mümkün değilse öncülün bitiş zamanı ile ardılın başlama zamanı arasındaki boş zamanı en büyükmeyi denemektedir. Çalışma kapsamında ayrıca belirtilen iki amacı gözetererek, ilişkili görevlerin gruplandıktan sonra atanması temeline dayanan iki safhalı, öncelik kuralı tabanlı sezgisel bir algoritma geliştirilmiştir. Yapılan deneysel çalışmalar sonucunda yöntemin kabul edilebilir performans sergilediği belirtilmiştir.

Lapierre ve Ruiz [55], görevlerin ön, arka ve her iki yönde de olmak üzere üç farklı pozisyon özelliğinin yanı sıra üst ve alt olmak üzere iki farklı yükseklik özelliğine de sahip olduğu çift taraflı montaj hattı dengeleme problemine benzer bir problem üzerinde çalışmıştır. Problemden ön-yukarı, ön-aşağı, arka-yukarı ve arka-aşağı olmak üzere 4 farklı istasyon tipi söz konusu olup bazı görevler için aynı istasyona atanacak veya atanmasından kaçınılacak şekilde bölgeleme kısıtları dikkate alınmıştır. Problemin çözümü için geliştirilen istasyon tabanlı sezgisel algoritma her seferinde mevcut istasyona hem özellik hem de görev zamanı açısından atanabilecek görevlerin listesini oluşturmakta, literatürden alınan 4 farklı öncelik kuralından birisini kullanarak bu listeden seçilen görevi mevcut istasyona atamaktadır. Gerçek hayattan alınan problem üzerinde yapılan deneysel çalışma sonucunda algoritmanın hali hazırda kullanılan hat dengesine yakın kalitede çözüm ürettiği sonucuna varılmıştır.

Hu ve ark. [56], çift taraflı montaj hattı dengeleme problemi (ÇMHDP) için istasyon tabanlı atama temeline dayanan bir birerleme yöntemi geliştirmiştir. Öncelikle her bir görev için tanımlanan bir zaman transfer fonksiyonu yardımıyla en erken ve en geç başlama zamanları belirlenmekte, daha sonra istasyon tabanlı bir yaklaşımla sol tarafta bulunan istasyona öncelik verilerek görevler istasyonlara atanmaktadır. Son olarak, öncelik ilişkisini sağlamayan atamalar kontrol edilerek bu tip atamalar varsa

kaldırılmaktadır. Belirtilen atama yöntemi Hoffmann [57] tarafından önerilen sezgiselle birleştirilerek ÇMHDP'in çözümü için kullanılmıştır. Küçük boyutlu problemler kullanılarak yapılan deneysel çalışmalar sonucunda önerilen sezgiselin geliştirilen basit bir yöntemle hesaplanan alt sınır değerlerine çok yakın sonuçlar verdiği belirtilmiştir.

Baykasoğlu ve Dereli [58], bazı görevlerin aynı istasyona atanması ve atanmasından kaçınılması şeklindeki bölgeleme kısıtları bulunan ÇMHDP için karınca kolonisi algoritması tabanlı sezgisel bir yöntem geliştirmiştir. Algoritma 4 adımdan oluşmakta olup ilk adımda istasyon çifti açılmakta, ikinci adımda mevcut istasyona atanabilecek görev seti oluşturulmakta, üçüncü adımda ise karınca kolonisi algoritmasıyla bu setten rastsal bir görev seçilmekte ve son adımda seçilen görev mevcut istasyona atanmaktadır. Belirtilen 4 adım tüm görevler atanıncaya kadar tekrarlanmaktadır. En iyi çözüm güncellenerek durdurma koşulu sağlanıncaya kadar prosedür tekrarlanmaktadır.

Wu ve ark. [16], ilk amaç hat uzunluğunun en küçüklenmesi, ikinci amaç ise açılan istasyon sayısının yani işçi sayısının en küçüklenmesi şeklinde olan çok amaçlı ÇMHDP'in çözümü için görev tabanlı dallandırma yapan bir dal-sınır algoritması geliştirmiştir. Algoritma dallandırma sürecinde sağ ve sol taraf görevleri için tek düğüm oluştururken, her iki tarafa da atanabilecek görevler için sağ ve sol atama olmak üzere iki düğüm oluşturmaktadır. Alt sınır değeri amaç fonksiyonuna uygun şekilde iki parçalı bir fonksiyonla hesaplanmış, ilk parça da problemin kutu paketleme problemine indirgenmesiyle kapasite alt sınır değeri bulunurken ikinci parçada sayma tabanlı alt sınır değeri hesaplanarak amaç fonksiyonuyla aynı ağırlıklara sahip olacak şekilde birleştirilmiştir. Yapılan deneysel çalışmalar sonucunda, daha etkin alt sınır ve baskınlık kuralları belirlenmesi yoluyla yöntemin performansının artırılması gerektiği sonucuna varılmıştır.

Kim ve ark. [59], karşılıklı istasyon sayısının bilindiği ÇMHDP'de çevrim zamanının en küçüklenmesi performans ölçütünü göz önünde bulundurup problemin karışık tamsayı doğrusal programlama modelini geliştirmiş ve problemin çözümü için

durdun durum genetik algoritması tabanlı sezgisel bir yöntem önermiştir. Görev sayısı uzunluğunda oluşturulan grup tabanlı kromozomun her bir geninde ilgili görevin atandığı istasyon çifti yer almakta, her iki tarafa da atanabilecek görevlerin hangi istasyona atanacağı geliştirilen bir sezgiselle belirlenmektedir. Sezgisel öncülü atanmış tüm görevler içerisinde en erken başlama zamanına sahip olan görevi bu zamanı veren istasyona atamakta, bu durumu sağlayan birden fazla görev varsa kendisi ve tüm ardıllarının toplam görev zamanı en büyük olan göreve öncelik vermektedir. Seçim mekanizması olarak rulet çemberi, çaprazlama ve mutasyon operatörü olarak da daha önce literatürde önerilmiş operatörlerin bu problem için yeniden düzenlenmesiyle elde edilen operatörler kullanılmıştır. Karşılaştırma çalışmalarıyla geliştirilen metodun Bartholdi [52] tarafından önerilen sezgisel ve Kim ve ark. [53] tarafından önerilen genetik algoritmadan daha iyi performansa sahip olduğu sonucuna varılmıştır.

Özcan ve Toklu [17], çevrim zamanı bilinen, ilk amaç hat uzunluğunun en küçüklenmesi, ikinci amaç istasyon sayısının en küçüklenmesi olan karışık modelli ÇMHDP için Kim ve ark. [59]'nın önerdiği matematiksel modeli temel alarak karışık tamsayı programlama modelini geliştirmiştir. Ayrıca, Hu ve ark. [56]'nın geliştirdiği alt sınır hesaplama yöntemini bu problem için uyarlamış ve problemin çözümü için tavlama benzetimi tabanlı sezgisel bir algoritma önermiştir. Geliştirilen sezgiselde öncelik listesi tabanlı oluşturulan çözümün her bir pozisyonu bir göreve karşılık gelmekte olup pozisyonda tutulan değer, ilgili görevin atanma önceliğini belirlemektedir. Başlangıç çözümü her bir pozisyona ilgili görevin numarası yazılarak oluşturulmakta olup, uygun görevler içerisinde en büyük öncelik değerine sahip olan görev mevcut istasyon çiftine pozisyon kısıtına uygun şekilde atanmaktadır. Her iki tarafa da atanabilecek görevlerin ataması rastsal olarak yapılmaktadır. Komşu çözümler iki görevin istasyon değiştirmesi ve tek bir görevin farklı bir istasyona atanması işlemlerinden birisinin rastsal seçilmesiyle üretilmektedir. Deneysel çalışmalarda önerilen sezgiselin küçük boyutlu problemlerin tamamında matematiksel modelle bulunan optimal çözümünü bulduğu, büyük boyutlu problemlerde de alt sınır değerlerine yakın çözümler ürettiği görülmüştür.

Özcan ve Toklu [18], Kim ve ark. [59] tarafından ÇMHDP'nde çevrim zamanını en küçükleme için önerilen karışık tamsayılı matematiksel modeli temel olarak pozitif ve negatif bölgeleme kısıtlarının da bulunduğu 3 farklı matematiksel model geliştirmiştir. İlk modelin amaç fonksiyonu ağırlıklı yapıda olup sırasıyla hat uzunluğu ve toplam istasyon sayısının en küçüklenmesidir. Öncelikli hedef programlama yapısında olan ikinci modelde göz önünde bulundurulmuş hedefler öncelik sırasına göre hat uzunluğu, çevrim zamanı ve istasyonlara atanan görev sayılarının belirlenen bir değerden küçük olması şeklindedir. Üçüncü model ise ikinci modeldeki amaçların bulanık olması durumunda kullanılacak doğrusal üyelik fonksiyonu temeline dayanılarak oluşturulan bulanık karışık tamsayılı doğrusal programlama modelidir.

Özcan ve Toklu [60] toplam istasyon sayısı ve düzgünlük indeksi performans ölçütlerinin aynı anda göz önünde bulundurulduğu çok amaçlı ÇMHDP'in çözümü için tabu arama tabanlı sezgisel bir algoritma önermiştir. Çözüm görev sayısı kadar hane oluşmakta olup her bir hanede ilişkili görevin öncelik değeri yer almaktadır. Komşu çözüm üretme mekanizması olarak seçilen görevlerin öncelik değerlerinin karşılıklı yer değiştirmesi kullanılmaktadır. Amaç fonksiyonlarının birleştirilmesi amacıyla minimum sapma metodu adı verilen bir yöntem kullanılmış olup önerilen algoritma literatürden alınan çeşitli test problemleri kullanılarak literatürdeki bazı yöntemlerle karşılaştırılmış ve diğer yöntemlerden daha iyi çözümler ürettiği gösterilmiştir.

Simaria ve Vilarinho [61], iki tarafta eş zamanlı yapılan çok işçili görevlerin olduğu, karışık modelli, bölgeleme kısıtlarının bulunduğu, birincil amaç fonksiyonu toplam istasyon sayısının en küçüklenmesi, ikincil amaç fonksiyonu ise istasyonlar arasında ve istasyon içinde modeller arasında iş yükü dengesinin sağlanması şeklinde ağırlıklı amaç fonksiyonuna sahip karışık modelli ÇMHDP'ni göz önünde bulundurmıştır. Problemin karışık tamsayılı doğrusal olmayan matematiksel modelini geliştirmiş, çözümü için karınca kolonisi algoritması temeline dayanan ve 2-ANTBAL adı verilen sezgisel bir yöntem önermiştir. Sezgiselin her bir çözümü, her biri hattın bir tarafına karşılık gelen iki ayrı karınca ile ifade edilmekte, hattın her iki tarafındaki

görev sıraları ve çözüm kalitesi fenomen miktarını belirlemektedir. Başlangıç çözümlerinin rastsal oluşturulduğu algoritmanın yapılan karşılaştırma çalışmalarında Lee ve ark. [54] tarafından önerilen sezgiselden daha iyi performansa sahip olduğu sonucuna varılmıştır.

Xiaofeng ve ark. [62] toplam istasyon sayısının en küçüklendiği ÇMHDP çözümü konusunda Wu ve ark. [16] ile Hu ve ark. [56]'nın çalışmalarının devamı niteliğinde olan bir dal-sınır algoritma önermiştir. İstasyon tabanlı dallandırma işleminde Hu ve ark. [56]'nın yöntemi kullanılmıştır. Problemin basit montaj hattına indirgenmesi yoluyla literatürde BMHDP için önerilmiş alt sınır hesaplama metotları ile maksimum yük, Jackson, uygun set ve pozisyon sıralama baskınlık kurallarının yanı sıra görev zamanı artırımı isimli azaltım kuralı doğrudan kullanılmıştır. Deneysel çalışmalarda yöntemin çözüm sonuçları hem CPU zamanları hem de alt sınırdan sapmalar açısından değerlendirilmiş, algoritmanın etkin olmasına karşın alt sınır hesaplama ve baskınlık kurallarının geliştirilmesi gerektiği sonucuna varılmıştır.

Özcan [63] görev zamanları stokastik, performans ölçütü hat uzunluğu ve istasyon sayısının en küçüklmesi olan çok amaçlı ÇMHDP'i göz önünde bulundurmıştır. Problemin şans kısıtlı, parçalı doğrusal, karışık tamsayı programlama modelini geliştirmiş, çözümü için tavlama benzetimi tabanlı sezgisel bir algoritma önermiştir. Algoritmada çözüm, her biri bir göreve karşılık gelen hattın sağ taraf ve sol tarafına yönelik öncelik değer çiftlerini içermekte, komşu çözüme hareket ise seçilen iki öncelik değerinin yer değiştirmesi şeklinde yapılmaktadır. Deneysel çalışmalarda geliştirilen matematiksel modelin ve önerilen tavlama benzetiminin sonuçları COMSOAL algoritması çözümleri ve alt sınır değerleriyle karşılaştırılmış, önerilen metotların etkin çözümler ürettiği sonucuna varılmıştır.

Özcan ve Toklu [64] görevler arasında hazırlık zamanları içeren, hat uzunluğunun ilk, toplam istasyon sayısının ikinci amaç olarak alındığı çok amaçlı ÇMHDP'ni ilk kez incelemiştir. Problem tanımından sonra problemin karışık tamsayı matematiksel modeli geliştirilmiş, bu yöntemin yanı sıra problemin çözümü için COMSOAL tabanlı sezgisel bir yöntem de önerilmiştir. Deneysel çalışmalarda çeşitli test

problemleri kullanılarak yöntemlerin performansları incelenmiş, önerilen sezgisel etkin çözümler ürettiği sonucuna varılmıştır.

Özcan ve ark. [65] paralel çift taraflı montaj hatlarının çalışma prensiplerini geliştirmiş, her birinde farklı modellerin üretildiği paralel çift taraflı iki veya daha fazla sayıda hattan oluşan üretim sistemini göz önünde bulundurmıştır. Performans ölçütü olarak toplam istasyon sayısının en küçüklenmesi dikkate alınmış olup problemin çözümü için tabu arama tabanlı sezgisel bir algoritma önerilmiştir. Komşu üretme mekanizması olarak seçilen iki görevin atama öncelik değeri karşılıklı yer değiştirmektedir. Literatürden alınan test problemleri kullanılarak yapılan deneysel çalışmalarda önerilen yöntemin bulunduğu sonuçların hem alt sınır değerlerinden sapsması, hem de literatürden alınan bağımsız ÇMHDP çözümlerine göre kalitesinin yeterli olduğu sonucuna varılmıştır.

3.5. Çok İşçili/Grup Tabanlı Montaj Hattı Dengeleme Problemleri

İstasyonların paralelleştirilmesine benzer bir etkiye sahip olup, bazı istasyonların birleştirilerek ekip adı verilen işçi grupları tarafından işletilen daha büyük üretim birimleri şeklinde düzenlenmesini kapsar [27]. Bu tip problemler örneğin uçak üretimi gibi ağır sanayi endüstrilerinde görülmekte olup, bu sistemlerde ürünün boyutu ve maliyetler nedeniyle alan sınırlaması söz konusudur. Bu durumun doğal sonucu olarak da istasyon sayısı sınırlıdır. İstenen çıktı oranına ulaşmak için istasyonlar işçi grupları içerecek şekilde bütünleştirilmektedir. Bu problemlerde verilmesi gereken iki karar: (1) hangi görevin hangi istasyona atanacağı (2) her bir istasyondaki ekip boyutunun ne olacağı şeklindedir. Görevlerin daha küçük boyutlu görev elemanlarından oluştuğu, bu noktadan hareketle görev zamanlarının ekip boyutunun azalan bir fonksiyonu olduğu varsayılır [66]. Diğer bir ifadeyle istasyona atanan ekip boyutu büyüdükçe istasyonun yerel çevrim zamanı artmaktadır. Sonuç olarak bu sistemlerde, geleneksel montaj hatlarındaki “görevlerin bölünemeyen en küçük birimlerden oluştuğu” varsayımı ortadan kalkmaktadır [27].

Shtub [67], problemin çözümü için "dallandır ve tahmin et" adı verilen sezgisel bir algoritma geliştirmiştir. Algoritma izin verilen en küçük boyuta sahip ekiple mevcut istasyonu açmakta, BMHDP için önerilmiş bir sezgiseli kullanarak mevcut istasyona atanabildiği kadar görev atamaktadır. Bu noktada toplam istasyon sayısı tahmin edilerek, istenen istasyon sayısı ile karşılaştırılmakta, fazla miktarda istasyon gerekiyorsa mevcut istasyonun ekip boyutu bir birim artırılmakta, ekip boyutu izin verilen maksimum boyutu geçmişse uygun çözüm olmadığı sonucu çıkarılmaktadır. Eğer tahmin edilen istasyon sayısı izin verilenden küçük veya eşitse bir sonraki istasyon açılarak işlemlere devam edilmektedir. Tahmin edilen istasyon sayısının hesaplanmasında mevcut istasyondan sonra gelen tüm istasyonlara izin verilen maksimum boyutlu ekip atanacağı varsayılmaktadır. Çalışma kapsamında ayrıca problemin tamsayılı doğrusal programlama modeli de geliştirilmiştir. Önerilen sezgisel ve tamsayılı modellerle 8-11 görevli, küçük boyutlu 32 adet problem çözülmüş, problemlerin %75'inde optimalin bulunduğu belirtilmiştir.

Akagi ve ark. [68], problemin çözümü için iki aşamalı bir sezgisel yöntem önermiştir. "Paralele atama metodu" adı verilen sezgiselin ilk safhasında her bir istasyona atanacak ekip büyüklükleri belirlenmekte ve görevlerin istasyonlara atamaları yapılmaktadır. Ekipte bulunan her bir işçinin performansının eşit olduğu varsayılmakta ve istasyon zamanının ekip büyüklüğüne oranının çevrim zamanını geçmemesi sağlanmaktadır. Algoritma, mevcut istasyon için tek bir işçiden başlayarak maksimum işçi sayısına kadar 4 farklı seçim kuralıyla uygun görevlerden bir görev seçip atamasını yapmaktadır. İstasyon zamanının işçi sayısı ve çevrim zamanı çarpımına oranı (yani istasyon etkinliği) en yüksek olan atama ve ekip büyüklüğünü bu istasyon için sabitlemekte ve sonraki istasyona geçmektedir. Sezgiselin ikinci safhasında ise istasyona atanan görevler istasyondaki işçilere paylaştırılmaktadır. Bu amaçla, her seferinde her bir işçiye birer görev atanmakta, görevlerin bölünebildiği varsayımıyla yükü az olan işçinin yükü fazla olan işçiye aktarılması yoluyla yardım etmesi amaçlanarak işçiler arasında yük dengesi sağlanmaktadır.

Chakravarty ve Shtub [66] çevrim zamanı büyük, talep miktarı az olan ve öğrenme

etkisi nedeniyle çevrim zamanının zaman içerisinde azaldığı montaj hatlarının çözümünde, önceden belirlenen üretim programını sağlayacak şekilde çevrim zamanı değişikliği maliyeti ve işgücü maliyetleri toplamını minimum yapmak amacıyla iteratif bir doğrusal programlama yaklaşımı kullanmıştır. Öğrenme etkisinin çevrim zamanını azaltması nedeniyle ekip boyutları zaman içerisinde dinamik olarak değiştirilmektedir. Problemin çözümü, hatta giren ilk ürünün çevrim zamanından başlamak üzere birbirini takip eden 2 ürünün çevrim zamanı farkı önceden belirlenen bir değerden büyük olduğu müddetçe ilgili çevrim zamanı için geliştirilen bir doğrusal programlama modelinin çözülerek görev atamaları ve ekip boyutlarının bulunması şeklinde yapılmaktadır.

Wilson [69], problemin çözümü için yeni bir tamsayılı programlama modeli geliştirmiştir. Amaç fonksiyonu olarak toplam işçi sayısının en küçüklmesi göz önünde bulundurulmuş, önerilen modelin Shtub [67]'in önerdiği modelden daha fazla sayıda kısıt içermesine rağmen daha az sayıda değişken ve katsayı gerektirdiği ifade edilmiştir. Standart dal-sınır algoritmasıyla yapılan deneysel çalışmalarda önerilen modelin Shtub [67]'in modelinden daha kısa sürede optimal çözümü bulunduğu sonucuna varılmıştır.

Johnson [70], göz önünde bulundurduğu problemde her bir görevin özel eğitimli bir grup tarafından yapılabileceği varsayılmaktadır. Özel bir grup için açılan ve tek bir işçinin bulunduğu istasyona sadece o grubun yapabileceği görevler atanabilmektedir. Bir grubun istasyonları arasında başka bir gruba ait istasyonun bulunmasına izin verilmektedir. Çözüm işleminin ilk aşamasında, problem alt problemlere ayrılmaktadır. Eğer aynı grubu gerektiren ara görevler başka bir gruptaki görevin öncülü ve ardılı değilse bu gruba ait görevler ayrılarak bir alt problem oluşturulmakta ve bu alt problem Johnson [71]'in BMHDP-1 için önerdiği FABLE isimli dal-sınır algoritması ile çözülmektedir. Bir birinin içine giren gruplardan oluşturulan alt problemlerin çözümü için ise FABLE temeline dayanarak geliştirilen ve MABLE adı verilen bir dal-sınır algoritması kullanılmaktadır. Son aşamada ise alt problem çözümleri birleştirilerek nihai çözüm elde edilmektedir.

Bukchin ve ark. [72], göz önünde bulundurduğu iki aşamalı modelin ilk safhasında hammadde düğümleri atılmış ürün ağacı alt montaj bileşenlerine ayrılarak her bir bileşen bir işçi grubuna atanmaktadır. İşçi grubunun boyutu atanan alt montaj bileşeninin gerektirdiği toplam işlem zamanının çevrim zamanına bölünmesiyle belirlenmektedir. İkinci safhada ise her bir gruba tahsis edilen alt montaj bileşenlerinde bulunan görevlerin öncelik ilişkilerini sağlayacak şekilde grupta bulunan işçilere atamasının yapılması önerilmiş fakat atamanın nasıl yapılacağı belirtilmemiştir. Sadece, işçilerin paralel çalışacağı, alt montaj bileşeninde bulunan tüm görevleri yapabilecek şekilde eğitilmiş olmaları gerektiği, yarı-otomatize oldukları ve tüm grubun bileşen kalitesinden sorumlu olduğu ifade edilmiştir. Çalışmada önerilen modelin hem tanımında hem de çözüm metodolojisinde çeşitli eksiklikler bulunmaktadır.

Bukchin ve Masin [73]'in çalışması Bukchin ve ark. [72]'nin çalışmasının devamı niteliğinde olup, ürün ağacından hareketle ana parça alt montaj parçalarına bölünmekte, her bir alt montaj parçasına alt montajda bulunan operasyonların toplam üretim zamanının çevrim zamanına oranı sayısınca işçi içeren bir grup atanmaktadır. Her bir alt montajda bulunan görevlerin işçilere atanmasında ise BMHDP için literatürde önerilen yöntemlerin kullanılabilmesi belirtilmiştir. Belirtilen modelin en önemli avantajının akış zamanı ve ara stok miktarını azaltması olduğu ifade edilmiştir. Problemin çıktısı toplam grup sayısı, gruplar arasındaki öncelik ilişkisi, grup büyüklükleri ve gruplara atanan görevlerdir. Çok amaçlı olarak göz önünde bulundurulmuş problemin çözümü için geriye doğru hesaplamalı bir dal-sınır algoritması geliştirilmiştir. Bu yöntemden hareketle büyük boyutlu problemlerin çözümünde kullanılabilen bir de sezgisel yöntem önerilmiş, yapılan deneysel çalışmalar sonucunda metodun etkin sonuçlar ürettiği belirtilmiştir.

3.6. Paralel Çok İşçili Montaj Hattı Dengeleme Problemleri

Pİ-MHDP konusunda ilk çalışma Dimitriadis [7] tarafından yapılmış bu çalışmada olup tek modellenmiş Pİ-MHDP'de toplam istasyon sayısının (bununla bağlantılı olarak da toplam alan ihtiyacının) en küçüklenmesi amacıyla Hoffmann [57]'nin önerdiği

sezgiselin modifikasyonu temeline dayanan sezgisel bir algoritma geliştirilmiştir. Hoffmann [57]'nin önerdiği sezgiselde mevcut istasyona atanabilecek tüm görev setleri birerlenmekte, bu durum ise fazla miktarda hesaplama zamanı gerektirmektedir. Önerilen sezgiselde ise mevcut istasyondaki her bir işçi için önceden belirlenmiş kabul edilebilir bir boş zamanı sağlayan atama bulununca alt set araması durdurulmakta ve yeni istasyon açılmaktadır. Algoritma, mevcut istasyonda izin verilen en fazla sayıda işçi olduğunu varsayarak atanabilecek görev listesini oluşturmakta ve bu görevleri Hu [74] algoritmasını kullanarak işçilere atamaktadır. Bu atamanın işçi başına boş zaman miktarı önceden belirlenen miktara eşit veya küçükse atama kabul edilerek sonraki istasyona geçilmekte, değilse işçi sayısı bir azaltılarak aramaya devam edilmektedir. İşçi sayısı bir olmasına rağmen kabul edilebilir bir atama bulunamamışsa istasyona yapılan işçi başına en küçük boş zamanlı atama kabul edilerek bir sonraki istasyona geçilmektedir. Hu [74]'un algoritmasına göre atanabilecek görevler içerisinde son istasyona kritik yol uzunluğu en büyük olan görev mevcut istasyondaki iş yükü en az olan işçiye atanmaktadır. Önerilen algoritma literatürden alınan bir gerçek hayat problemi ve çeşitli karşılaştırma problemlerine uygulanmış, problemlerin önemli bir kısmında hattın etkinliği toplam boş zaman ve toplam işçi sayısı değişmeden hattın BMHDP-1'e kıyasla daha az istasyonla dengelendiği belirtilmiştir.

Becker ve Scholl [15], tek modelli, çevrim zamanı ve istasyon sayıları verilip işçi sayısının en küçüklendiği Pİ-MHDP'in özel bir sınıfının tanımını yapıp varsayımlarını belirterek ilgili problemi değişken çalışma alanlı montaj hattı dengeleme problemleri (DA-MHDP) olarak isimlendirmiştir. DA-MHDP'nin en temel özelliği birimin belirli sayıda montaj pozisyonlarına ayrılması ve bu pozisyonlarla ilişkili atama kısıtları içermesidir. Her hangi bir montaj pozisyonunda yapılan tüm montaj görevleri aynı istasyonda tek bir işçiye atanabilmekte, aynı işçi sadece komşu montaj pozisyonlarında yapılan görevleri yapabilmektedir. Göz önünde bulundurulan problem çeşitli bölgeleme kısıtlarının yanı sıra çevrim zamanını aşan montaj görevleri de içermektedir. Bahsi geçen çalışma kapsamında bu problemin karışık tamsayılı matematiksel modeli oluşturulmuş, basit montaj hatları için literatürde geliştirilmiş alt sınır hesaplama yöntemlerinden üç tanesi probleme

uygulanacak şekilde düzenlenmiş, görevler için istasyon aralıklarının hesaplanmasında kullanılacak formüller geliştirilmiştir. Problemin çözümü için Scholl ve Klein [48] tarafından BMHDP-1'in çözümü için önerilen SALOME isimli yöntemi temel alan bir dal-sınır algoritması geliştirmiştir. Algoritma istasyon tabanlı bir yapıda birerleme yapmakta olup lazer arama adı verilen stratejiden faydalanmaktadır. Dal sınır ağacının her bir seviyesi bir istasyona karşılık gelmekte, her bir düğüm ise ilgili istasyonun alternatif yüklerinden birisini ifade etmektedir. Global alt sınır değeri olarak geliştirilen üç yöntemle bulunan alt sınırlardan en iyisi kullanılmakta, yerel alt sınır değeri ise yöntemlerden sadece birisi kullanılarak hesaplanmaktadır. Her bir düğümde istasyonu belirlenen görevlerden oluşan bir kısmi çözüm ve henüz atanmamış görevlerin oluşturduğu bir artık problem söz konusudur. Kısmi çözüm için gerekli istasyon sayısı ile artık problemin alt sınır değeri toplanarak yerel alt sınır değeri bulunmakta, bu değer üst sınıra eşit veya büyükse düğüm budanmaktadır. Basit montaj hatları için önerilen "Maksimum yük", "Uygun set", "Jackson" ve "Boş istasyon" baskınlık kuralları ilgili problemde kullanılacak şekilde revize edilerek algoritma kapsamında kullanılmıştır. İstasyon yükleri belirlendikten sonra bu yükün uygun olup olmadığı, uygunsa görevlerin en az sayıda işçi arasında nasıl paylaşılacağı ve görev başlangıç sürelerinin ne olacağının belirlenmesi (istasyon problemi) için kaynak kısıtlı proje çizelgeleme problemleri için literatürde önerilen bir dal-sınır algoritması modifiye edilerek kullanılmıştır. İstasyon problemi için önerilen bu algoritmanın her bir dallandırmasında istasyona atanan görevlerden biri işçilerden birisine atanmakta olup lazer arama stratejisi kullanılmış ve alt sınır değerlerinin hesaplanmasında problemin kutu paketleme problemine gevşetilmesinden faydalanılmıştır. Problemin çözümü için önerilen dal-sınır algoritmasının istasyon probleminin çözüm süresi ve ilk istasyon için harcanan süre sınırlaması yoluyla sezgisel yöntem olarak kullanılabilceği belirtilmiştir. Önerilen yöntemler çift taraflı montaj hatları için Kim ve ark. [53] ve Baykasoğlu ve Dereli [58]'nin önerdiği sezgisellerle karşılaştırılmış ve bu sezgisellerden daha etkin oldukları sonucuna varılmıştır. Dal sınır algoritmasının küçük ve orta büyüklükteki problemlerde etkin olduğu belirtilirken, büyük boyutlu problemlerde sezgiselin kullanılmasının daha uygun olacağı ifade edilmiştir.

Cevikcan ve ark. [75], Pİ-MHDP'nin mevcut literatürünü özetlemiş, karışık modeli, model bazında farklı çevrim zamanları olan, pozitif ve negatif bölgeleme kısıtlarına sahip problem için 5 aşamalı sezgisel bir yöntem geliştirmiştir. Sezgiselin “Yatay dengeleme” adı verilen ilk adımında her bir modele ait görevler, öncelik ve çevrim zamanı kısıtlarını sağlayacak şekilde sıralı gruplara yani tek işçili seri istasyonlara ayrılmaktadır. Bu işlem için Bedworth ve Bailey [76] tarafından önerilen ve Kilbridge ve Wester [77] yöntemi temeline dayanan “bölgesel yaklaşım” sezgiseli kullanılmıştır. Bu sezgisel öncelik diyagramını çevrim zamanını sağlayacak şekilde bölgelere ayırmakta, en solda bulunan bölgeden başlayıp görev zamanı büyük olan görevlere öncelik vererek görev gruplarını oluşturmaktadır. Model grupları her bir model için ayrı ayrı oluşturulmaktadır. “Dikey dengeleme” isimli ikinci aşamada ise iş yükü ortalamasının üzerinde olan model gruplarına ait bazı görevler öncelik ve çevrim zamanı kısıtları sağlanacak şekilde iş yükü ortalamasının altında olan gruplara atanarak model grupları arasında iş yükü dengesi sağlanmaya çalışılmaktadır. Bu amaçla, her bir model için gruplardaki tüm görevler tek tek kontrol edilerek belirtilen kısıtları sağlayıp belirtilen amaca hizmet edenler farklı gruplara atanmaktadır. Üçüncü safha ise “Fiziksel istasyonların oluşturulması (ekip belirleme)” olarak isimlendirilmiştir. Bu aşamada sıralı görev gruplarından bazıları birleştirilerek tek bir istasyonda birden fazla işçi çalışacak şekilde ayarlanmakta, yani görevler işçilere atanmaktadır. Bu amaçla kaynak kısıtlı proje planlama problemi (KKPPP) için literatürde yer alan sezgisel bir yöntem modifiye edilerek kullanılmıştır. Sezgiselle başlanmadan önce işlem karmaşıklığını ortadan kaldırmak amacıyla PARETO analizi (ABC analizi) yoluyla modellerin, tüm modelleri en fazla temsil eden bir alt setinin bu aşamada göz önünde bulundurulmak üzere seçildiği belirtilmiştir. Sezgisel, her bir model için ilk istasyondan başlayarak maksimum işçi sayısına kadar tüm işçi sayılarındaki performans değerleri analiz edilerek ilgili istasyon için işçi sayısı belirlenmektedir. Örneğin birinci istasyon için 3 işçinin bulunduğu durumda istasyona ilk 3 görev grubunda bulunan görevler atanacaktır. Eğer bu durum seçilirse ve ikinci istasyonda 2 işçi bulunacaksa 4 ve 5. görev gruplarında bulunan görevler 2. istasyona atanacaktır. İstasyonu oluşturacak gruplar bu şekilde belirlendikten sonra görevlerin çizelgelenmesi şu yöntemle yapılmaktadır. İstasyona atanacak tüm görevler mevcut modelin öncelik diyagramında son düğüme olan en uzak mesafelerine göre (eşitlik

varsa görev zamanlarına göre) büyükten küçüğe sıralanarak öncelik listesi oluşturulur. Tüm görevler çizelgelenene kadar öncelik listesindeki ilk uygun eleman boş kalma süresi en küçük olan işçiye atanır. Çevrim zamanının %20 oranında geçilebileceği varsayılmış olup mevcut istasyon için en iyi işçi sayısına talepler bazında düzgünlük indeksi en iyi olan alternatifin seçilmesiyle karar verilmektedir. Ayrıca bu aşamada kullanılabilecek ikinci dereceden doğrusal olmayan bir matematiksel modelde geliştirilmiştir. Görev ve model sayısına göre matematiksel modelin boyutunun çok büyüdüğü ve doğrusal olmayan yapısı nedeniyle optimal çözümün bulunmasında kullanılamayacağı belirtilmiştir. Dördüncü ve “Model sıralama” olarak isimlendirilen aşamada ise modellerde bulunan parçalar verilen bir formül ile hesaplanan teslim tarihlerine göre küçükten büyüğe sıralanarak model sırası oluşturulmaktadır. Beşinci ve son aşamada ise talebin değişmesi veya talebe yeni modeller eklenmesi durumunda oluşturulan bir benzetim programıyla darboğaz ve boş istasyonlar arasında işçi transferinin analiz edilebileceği belirtilmiş, çalışmanın amacına hizmet etmediği gerekçesiyle son adımın detayına inilmemiştir. Önerilen sezgisel yöntem Dimitriadis [7] tarafından geliştirilen sezgiselle yine aynı çalışmada sunulan gerçek hayat problemi kullanılarak karşılaştırılmış, toplam alan gereksinimi ve ortalama akış zamanı açısından önerilen yöntemin daha etkin olduğu sonucuna varılmıştır. Ayrıca, önerilen yöntem traktör üretimi yapan bir firmanın gerçek hayat probleminin 35 görevli 6 modelli bir alt parçasına uygulanarak sonuçları çalışma kapsamında sunulmuştur.

Son zamanlarda yapılmış diğer bir çalışmada ise Fattahi ve ark. [78] toplam işçi sayısının en küçüklenmesini birincil amaç, toplam istasyon sayısının en küçüklenmesini de ikincil amaç olarak alan çok amaçlı Pİ-MHDP’ni göz önünde bulundurmıştır. Problemin karışık tam sayılı matematiksel modelini geliştirdikten sonra, çözümü için karınca kolonisi temeline dayanan sezgisel bir algoritma önermiştir. Önerilen algoritmada her bir karınca öncelik ilişkileri, görev zamanları ve çevrim zamanı kısıtlarını sağlayacak şekilde verilen görevleri istasyonlara atamaktadır. Sezgisel, Dimitriadis [7] tarafından önerilen sezgisel ile karşılaştırılmış ve daha iyi performansa sahip olduğu gösterilmiştir.

4. PARALEL ÇOK İŞÇİLİ MONTAJ HATTI DENGELEME PROBLEMLERİNİN ÇÖZÜMÜ İÇİN KARIŞIK TAMSAYILI MATEMATİKSEL PROGRAMLAMA MODELİ

4.1. Kullanılan Notasyonlar ve Temel Modelleme Yaklaşımı

Matematiksel modellerde kullanılan parametreler, setler ve karar değişkenleri aşağıdaki şekildedir.

Parametreler

t_i	: i görevinin görev zamanı
c	: Çevrim zamanı
m_{Enb}	: Her bir istasyonda izin verilen maksimum işçi sayısı
k_{Enb}	: İzin verilen maksimum istasyon sayısı
M	: Çok büyük sabit bir sayı
MM	: Çok çok büyük sabit bir sayı

Setler

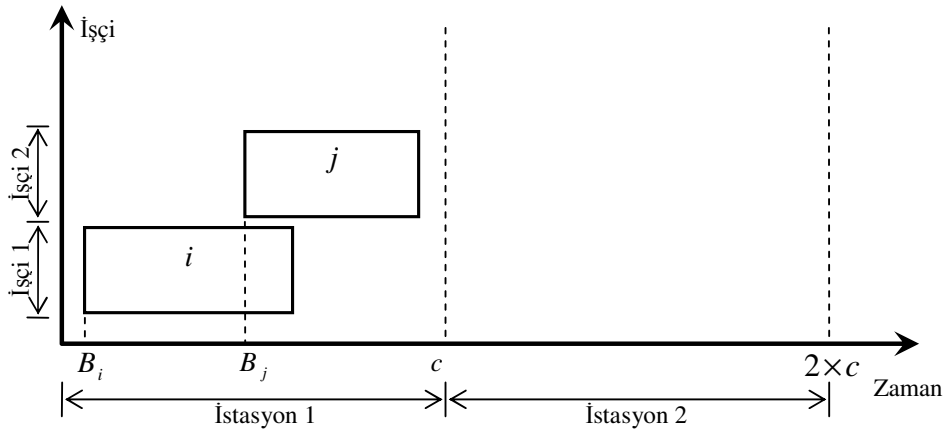
F_i	: Öncelik diyagramında i görevini direkt takip eden görevlerin seti
F_i^*	: Öncelik diyagramında i görevini takip eden tüm görevlerin seti
A	: $j > i$ ve $j \in F_i$ olmak üzere (i, j) görev çiftleri seti
A^*	: $j > i$ ve $j \in F_i^*$ olmak üzere (i, j) görev çiftleri seti
P_i	: Öncelik diyagramında i görevinin direkt takip ettiği görevlerin seti
P_i^*	: Öncelik diyagramında i görevinin takip ettiği tüm görevlerin seti
UR_i	: i göreviyle öncelik ilişkisi olmayan, eş zamanlı olarak yapılabilecek görevlerin seti
II	: Aynı işçiye atanması istenmeyen görev çiftlerinin oluşturduğu set (negatif bölgeleme)
IS	: Aynı işçiye atanması istenen görev çiftlerinin oluşturduğu set (pozitif bölgeleme)
SI	: Aynı istasyona atanması istenmeyen görev çiftlerinin oluşturduğu set (negatif bölgeleme)
SS	: Aynı istasyona atanması istenen görev çiftlerinin oluşturduğu set (pozitif bölgeleme)
TI	: Aynı zaman biriminde eş zamanlı olarak yapılması istenmeyen görev çiftlerinin oluşturduğu set (negatif bölgeleme)

Karar değişkenleri

B_i	: i görevinin başlama zamanı
k_i	: i görevinin atandığı istasyon ($k_i = 1, 2, \dots, k_{Enb}$)

- m_i : i görevinin atandığı işçi ($m_i = 1, 2, \dots, m_{Emb}$)
 k_{Toplam} : Toplam istasyon sayısı
 os_l : l istasyonundaki toplam işçi sayısı ($l = 1, 2, \dots, k_{Emb}$;
 $os_l = 1, 2, \dots, m_{Emb}$)
 γ_{il} : i görevi l istasyonu veya sonrasına atanmışsa 1, değilse 0 ($k_i \geq l$)
 $u1_{ij}, u2_{ij}$: $u1_{ij} = 0, u2_{ij} = 0$ ise i görevi j görevi bittikten sonra başlar,
 $u1_{ij} = 0, u2_{ij} = 1$ ise j görevi i görevi bittikten sonra başlar,
 $u1_{ij} = 1, u2_{ij} = 0$ ise i görevi j görevinin atandığı işçiden daha büyük indeksli bir işçiye atanır,
 $u1_{ij} = 1, u2_{ij} = 1$ ise j görevi i görevinin atandığı işçiden daha büyük indeksli bir işçiye atanır.

Problemin tamsayı programlama modeli BMHDP için Scholl [5] tarafından önerilen üçüncü model temel alınarak oluşturulmuştur. Scholl [5], ilgili modeli akış tipi çizelgeleme modelleri için geliştirilen matematiksel modelleri temel alarak geliştirdiğini ifade etmektedir. Bilindiği gibi BMHDP’de her bir istasyonda sadece 1 adet işçi bulunmakta, yani istasyonlar işçilere karşılık gelmektedir. Buna karşın Pİ-MHDP, istasyonlarda birden fazla işçinin bulunmasına izin vermektedir. Aynı istasyonun farklı işçileri tarafından eş zamanlı olarak yapılan görevler dâhil tüm görevlerin öncelik ilişkilerini sağlaması gerekmektedir. Bu yapının i ve j görevleri arasında modellenebilmesi amacıyla şekilsel gösterimi Şekil 4.1’de sunulan yaklaşım kullanılmıştır.



Şekil 4.1. Matematiksel modele ilişkin şekilsel gösterim

Şekil 4.1 detaylı incelendiğinde modellemeye ilişkin aşağıdaki hususların vurgulandığı ortaya çıkmaktadır:

- Eğer j görevinin i görevinden sonra yapılması gerektiğine yönelik bir öncelik ilişkisi söz konusu ise $B_j \geq B_i + t_i$ olmalıdır.
- Eğer i ve j görevleri arasında öncelik ilişkisi söz konusu değilse bu durumda aşağıdaki şartlardan en az birisinin sağlanması gerekir:
 - i görevi, j görevi tamamlandıktan sonra başlar,
 - j görevi, i görevi tamamlandıktan sonra başlar,
 - i görevi ve j görevi farklı işçilere atanır. Yani, ya i görevinin atandığı işçi indeksi (m_i), j görevinin atandığı işçi indeksinden (m_j) büyük olmalı veya tam tersi olmalıdır.
- Verilen bir i görevinin başlama (B_i) ve bitiş saat zamanları ($B_i + t_i$) atandığı istasyonun (k_i) başlangıç ($(k_i - 1) \times c$) ve bitiş ($k_i \times c$) saat zamanları arasında yer almalıdır.

4.2. Toplam İstasyon Sayısının En Küçüklenmesi Probleminin Matematiksel Modeli

Yapılan ön çalışmalarda toplam istasyon sayısının en küçüklendiği Pİ-MHDP'in Pİ-MHDP1'e temel oluşturduğu görülmüştür. Bu nedenle öncelikle temel olan söz konusu model sunulacaktır. Model, karışık tamsayı doğrusal programlama modeli yapısında olup toplam kısıt sayısı $5 \times n + |A| + 4 \times (n \times (n - 1) / 2 - |A^*|)$ olmak üzere $O(n^2)$ iken toplam tamsayı değişken sayısı $2 \times n$ olmak üzere $O(n)$ ve 0-1 tamsayı değişken sayısı da $2 \times (n \times (n - 1) / 2 - |A^*|)$ olmak üzere $O(n^2)$ 'dir.

Amaç fonksiyonu

$$\min k_{Toplam} \tag{4.1}$$

Kısıt seti 1: Öncelik ilişkileri sağlansın.

$$B_j \geq B_i + t_i \quad ((i, j) \in A) \quad (4.2)$$

Kısıt seti 2: Görev, bulunduğu istasyonun başlangıç ve bitiş zamanları arasında yapılsın.

$$B_i \geq (k_i - 1) \times c \quad (i = 1, \dots, n) \quad (4.3)$$

$$B_i + t_i \leq k_i \times c \quad (i = 1, \dots, n) \quad (4.4)$$

Kısıt seti 3: Aralarında öncelik ilişkisi bulunmayan i ve j görevleri için:

- i görevi j görevi bittikten sonra başlasın (Eş. 4.5. aktif) veya,
- j görevi i görevi bittikten sonra başlasın (Eş. 4.6. aktif) veya,
- i görevi j görevinin atandığı işçiden daha büyük indeksli bir işçiye atansın (Eş. 4.7. aktif) veya,
- j görevi i görevinin atandığı işçiden daha büyük indeksli bir işçiye atansın (Eş. 4.8. aktif).

$$B_i \geq B_j + t_j - M(u_{1_{ij}} + u_{2_{ij}}) \quad ((i, j) \notin A^*) \quad (4.5)$$

$$B_j \geq B_i + t_i - M(u_{1_{ij}} + (1 - u_{2_{ij}})) \quad ((i, j) \notin A^*) \quad (4.6)$$

$$m_i \geq m_j + 1 - M((1 - u_{1_{ij}}) + u_{2_{ij}}) \quad ((i, j) \notin A^*) \quad (4.7)$$

$$m_j \geq m_i + 1 - M((1 - u_{1_{ij}}) + (1 - u_{2_{ij}})) \quad ((i, j) \notin A^*) \quad (4.8)$$

Kısıt seti 4: İstasyona atanacak işçilerin sayısı izin verilen sınırlar içerisinde olsun.

$$1 \leq m_i \leq m_{Enb} \quad (i = 1, \dots, n) \quad (4.9)$$

Kısıt seti 5: Toplam istasyon sayısı tüm görevlerin atandığı istasyonlardan büyük/eşit olsun.

$$k_{Toplam} \geq k_i \quad (i = 1, \dots, n) \quad (4.10)$$

Pozitiflik kısıtları

$$B_i, k_{Toplam} \geq 0 \quad ; \quad k_i, m_i \geq 0, tms \quad ; \quad u1_{ij}, u2_{ij} \quad 0-1 \quad (4.11)$$

4.3. Toplam İşçi Sayısının En Küçüklenmesi Probleminin Matematiksel Modeli

Pİ-MHDP1 için geliştirilen model karışık tamsayılı doğrusal programlama modeli yapısında olup toplam kısıt sayısı $6 \times n + 2 \times n \times k_{Enb} + |A| + 4 \times (n \times (n-1) / 2 - |A^*|)$ 'dir. Açık ki $n \geq k_{Enb}$ 'tür. Dolayısıyla modelin içerdiği kısıt sayısı $O(n^2)$ 'dir. Modelde bulunan toplam tamsayılı değişken sayısı ise $2 \times n$ olmak üzere $O(n)$ ve 0-1 tamsayılı değişken sayısı da $2 \times (n \times (n-1) / 2 - |A^*|) + n \times k_{Enb}$ olmak üzere $O(n^2)$ 'dir.

Amaç fonksiyonu

$$\min \sum_{i=1}^{k_{Enb}} os_i \quad (4.12)$$

Kısıt seti 1, Kısıt seti 2, Kısıt seti 3, Kısıt seti 4

Kısıt seti 6: Açılacak toplam istasyon sayısı izin verilen sınırlar içerisinde olsun

$$1 \leq k_i \leq k_{Enb} \quad (i = 1, \dots, n) \quad (4.13)$$

Kısıt seti 7: $k_i \geq l$ ise $\gamma_{il} = 1$ olsun.

$$k_i - l + 1 \leq M \times \gamma_{il} \quad (i = 1, \dots, n; l = 1, \dots, k_{Enb}) \quad (4.14)$$

Kısıt seti 8: os_l değişkenine l istasyonundaki maksimum işçi atansın.

$$os_l \geq (\gamma_{il} - 1) \times MM + (l - k_i) \times M + m_i \quad (i = 1, \dots, n; l = 1, \dots, k_{Enb}) \quad (4.15)$$

Pozitiflik kısıtları

$$B_i, os_l \geq 0 \quad (i = 1, \dots, n; l = 1, \dots, k_{Enb}) \quad (4.16)$$

$$k_i, m_i \geq 0 \text{ ve tms} \quad (i = 1, \dots, n) \quad (4.17)$$

$$u1_{ij}, u2_{ij}, \gamma_{il} \in \{0, 1\} \quad (i = 1, \dots, n; j = 1, \dots, n; l = 1, \dots, k_{Enb}) \quad (4.18)$$

4.4. Bazı Bölgeleme Kısıtları

Montaj hattı dengeleme problemlerinde sıkça ihtiyaç duyulan bazı önemli bölgeleme kısıtlarının yanı sıra bu çalışma kapsamında önerilen negatif zaman bölgeleme kısıtlarının önerilen modellere nasıl eklenebileceği dair bilgiler aşağıdaki alt bölümlerde sunulmuştur.

4.4.1. İşçi uyumsuz görevler

Aynı işçi tarafından yapılmasına izin verilmeyecek II setinin her bir (i, j) görev çifti için aşağıdaki kısıtların modele eklenmesi gerekir. Kısıtlardaki II_{ij} yardımcı karar değişkeni 0-1 tipinde olup kısıtlardan en az birisinin aktifliğinin garantilenmesi amacıyla kullanılmıştır.

$$(k_i - 1) \times m_{Enb} + m_i + 1 - (k_j - 1) \times m_{Enb} - m_j \leq M \times II_{ij} \quad ((i, j) \in II) \quad (4.19)$$

$$(k_j - 1) \times m_{Enb} + m_j + 1 - (k_i - 1) \times m_{Enb} - m_i \leq M \times (1 - II_{ij}) \quad ((i, j) \in II) \quad (4.20)$$

4.4.2. İstasyon uyumsuz görevler

Aynı istasyonda yapılmasına izin verilmeyecek SI setinin her bir (i, j) görev çifti için aşağıdaki kısıtların modele eklenmesi gerekir. Kısıtlardaki $I2_{ij}$ yardımcı karar değişkeni 0-1 tipinde olup kısıtlardan en az birisinin aktifliğinin garantilenmesi amacıyla kullanılmıştır.

$$k_j + 1 - k_i \leq M \times I2_{ij} \quad ((i, j) \in SI) \quad (4.21)$$

$$k_i + 1 - k_j \leq M \times (1 - I2_{ij}) \quad ((i, j) \in SI) \quad (4.22)$$

4.4.3. Aynı işçiye atanacak görevler

Aynı işçi tarafından yapılması istenen IS setinin her bir (i, j) görev çifti için aşağıdaki kısıtların modele eklenmesi gerekir.

$$(k_i - 1) \times m_{Enb} + m_i = (k_j - 1) \times m_{Enb} + m_j \quad ((i, j) \in IS) \quad (4.23)$$

4.4.4. Aynı istasyona atanacak görevler

Aynı istasyonda yapılması istenen SS setinin her bir (i, j) görev çifti için aşağıdaki kısıtların modele eklenmesi gerekir.

$$k_i = k_j \quad ((i, j) \in SS) \quad (4.24)$$

4.4.5. Negatif zaman bölgeleme kısıtları

Aynı anda eşzamanlı olarak yapılamayacak TI setinin her bir (i, j) görev çifti için kısıt seti 3 yerine aşağıdaki kısıtların modele eklenmesi gerekir. Kısıtlardaki $I_{3_{ij}}$ yardımcı karar değişkeni 0-1 tipinde olup kısıtlardan en az birisinin aktifliğinin garantilenmesi amacıyla kullanılmıştır.

$$B_i \geq B_j + t_j - M \times I_{3_{ij}} \quad ((i, j) \in TI) \quad (4.25)$$

$$B_j \geq B_i + t_i - M \times (1 - I_{3_{ij}}) \quad ((i, j) \in TI) \quad (4.26)$$

4.5. Test Problemi Sonuçları

Ele alınan modelin literatürü dar olduğu için karşılaştırma problemleri halen yayınlanmamıştır. Problemlerin üretilmesi amacıyla literatürde BMHDP-1 için önerilen çözüm yöntemlerinin testinde kullanılan ve t_{Enk} ve t_{Enb} sırasıyla problemdeki en küçük ve en büyük görev zamanları olmak üzere Çizelge 4.1'de verilen karşılaştırma problemleri baz alınarak aşağıda sunulan 6 adımlı bir çalışma yapılmıştır:

1. Karşılaştırma problemleri için literatürde kullanılan çevrim zamanları kullanılmıştır. Örneğin Arcus1 problemi için literatürde 16 farklı çevrim zamanı kullanılarak 16 farklı BMHDP-1 üretilmiştir.

Çizelge 4.1. BMHDP-1 için literatürde önerilen karşılaştırma problemleri

Problem	n	t_{Enk}	t_{Enb}	t_{Toplam}	Literatürde kullanılan çevrim zamanları
Arcus1	83	233	3691	75707	3786, 3985, 4206, 4454, 4732, 5048, 5408, 5824, 5853, 6309, 6842, 6883, 7571, 8412, 8898, 10816
Arcus2	111	10	5689	150399	5755, 5785, 6016, 6267, 6540, 6837, 7162, 7520, 7916, 8356, 8847, 9400, 10027, 10743, 11378, 11570, 17067
Barthold	148	3	383	5634	84, 85, 87, 89, 91, 93, 95, 97, 99, 101, 104, 106, 109, 112, 115, 118, 121, 125, 129, 133, 137, 142, 146, 152, 157, 163, 170
Barthol2	148	1	83	4234	403, 434, 470, 513, 564, 626, 705, 805
Bowman	8	3	17	75	20
Buxey	29	1	25	324	27, 30, 33, 36, 41, 47, 54
Gunther	35	1	40	483	41, 44, 49, 54, 61, 69, 81
Hahn	53	40	1775	14026	2004, 2338, 2806, 3507, 4676
Heskiaoff	28	1	108	1024	138, 205, 216, 256, 324, 342
Jackson	11	1	7	46	7, 9, 10, 13, 14, 21
Jaeschke	9	1	6	37	6, 7, 8, 10, 18
Kilbridge	45	3	55	552	56, 57, 62, 69, 79, 92, 110, 111, 138, 184
Lutz1	32	100	1400	14140	1414, 1572, 1768, 2020, 2357, 2828
Lutz2	89	1	10	485	11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21
Lutz3	89	1	74	1644	75, 79, 83, 87, 92, 97, 103, 110, 118, 127, 137, 150
Mansoor	11	2	45	185	48, 62, 94
Mertens	7	1	6	29	6, 7, 8, 10, 15, 18
Mitchell	21	1	13	105	14, 15, 21, 26, 35, 39
Mukherje	94	8	171	4208	176, 183, 192, 201, 211, 222, 234, 248, 263, 281, 301, 324, 351
Roszieg	25	1	13	125	14, 16, 18, 21, 25, 32
Sawyer	30	1	25	324	25, 27, 30, 33, 36, 41, 47, 54, 75
Scholl	297	5	1386	69655	1394, 1422, 1452, 1483, 1515, 1548, 1584, 1620, 1659, 1699, 1742, 1787, 1834, 1883, 1935, 1991, 2049, 2111, 2177, 2247, 2322, 2402, 2488, 2580, 2680, 2787
Tonge	70	1	156	3510	160, 168, 176, 185, 195, 207, 220, 234, 251, 270, 293, 320, 364, 410, 468, 527
Warnecke	58	7	53	1548	54, 56, 58, 60, 62, 65, 68, 71, 74, 78, 82, 86, 92, 97, 104, 111
Wee-Mag	75	2	27	1499	28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 45, 46, 47, 49, 50, 52, 54, 56

- Her bir karşılaştırma problemine görev zamanı sıfır olan tek bir batak düğüm eklenmiş, ilgili batak düğümün Becker ve Scholl [15] tarafından geliştirilen yöntemle atanabileceği ilk istasyon belirlenmiş, bu değer ele alınan problem için istasyon sayısı alt sınır değeri (AK_1) olarak alınmıştır.
- $OptB$ değeri BMHDP-1 için literatürden alınan optimal (veya üst sınır) istasyon sayısı olmak üzere her bir BMHDP-1 problemine karşılık gelen farklı Pİ-MHDP1 k_{Enb} değerleri $\{ AK_1, [(AK_1 + OptB) / 2]^+, OptB \}$ olarak alınmıştır.

4. Her bir problem için işçi sayısı alt sınır değeri $AS_1 = [t_{Toplam} / c]^+$ olarak tespit edilmiştir [5].
5. Her bir BMHDP-1 problemine karşılık gelen farklı Pİ-MHDP-1 m_{Enb} değerleri $\{[AS_1 / k_{Enb}]^+, 2 \times [AS_1 / k_{Enb}]^+\}$ olarak alınmıştır.
6. Belirtilen her bir c , k_{Enb} ve m_{Enb} kombinasyonu için birer adet olmak üzere toplam 1614 adet karşılaştırma problemi üretilmiş, bu problemlerden tekrar eden 63 adeti ve m_{Enb} değeri 1 olan 291 adetinin çıkartılmasıyla (BMHDP-1 olduğu için) 1260 adet Pİ-MHDP1 karşılaştırma problemi elde edilmiştir.

Karşılaştırma işlemlerinde P4 3GHz işlemcili 1GB dâhili bellek bulunan kişisel bilgisayarlar kullanılmıştır. Problemlerin çözümünde GAMS 22.5 yazılımı aracılığıyla Cplex çözücüsü kullanılmış ve her bir problem için 3600 CPU saniyesi çalıştırılmıştır. Çözüm sonuçlarına ilişkin bazı bilgiler Çizelge 4.2’de sunulmuştur.

Çizelge 4.2. Matematiksel model çözüm sonuçları

Problem	n	Optimal bulunan	Uygun çözüm bulunan	Çözumsuz	Uygun çözüm bulunamayan	Toplam
Arcus1	83		6		74	80
Arcus2	111		1	3	81	85
Barthol2	148				135	135
Barthold	148				40	40
Bowman	8	2				2
Buxey	29		19		15	34
Gunther	35		22	1	12	35
Hahn	53		7		5	12
Heskiaoff	28	10	13		5	28
Jackson	11	21		2		23
Jaeschke	9	14				14
Kilbridge	45		7		41	48
Lutz1	32	1	22		1	24
Lutz2	89		4	6	45	55
Lutz3	89		16		44	60
Mansoor	11	7				7
Mertens	7	18				18
Mitchell	21	15				15
Mukherje	94				65	65
Roszieg	25	9	17			26
Sawyer	30	4	28		12	44
Scholl	297				130	130
Tonge	70		8		72	80
Warnecke	58		1		79	80
Wee-mag	75				120	120
Toplam		101	171	12	976	1260
%		8	14	1	77	

Çizelge 4.2'den de görülebileceği gibi göz önünde bulundurulan problemin çözümünde matematiksel modelin kullanımı oldukça sınırlıdır. Toplam 101 adet problemin optimal çözümü bulunmuş, 12 adet problemin çözümsüz olduğu ispat edilmiş, 171 adet probleme ise verilen 1 saatlik süre içerisinde sadece en az birer adet uygun çözüm bulunabilmiştir. Dolayısıyla problemlerin %77'sine karşılık gelen 976 adet probleme uygun çözüm dahi bulunamazken, problemlerin %91'ine karşılık gelen 1147 adet problemin optimal çözümü bulunamamıştır. Optimalı bulunan en büyük boyutlu problemin görev sayısı ise 32 olmuş, bu grupta bulunan 24 farklı örneğin sadece bir tanesi çözülebilmştir.

5. PARALEL ÇOK İŞÇİLİ MONTAJ HATTI DENGELEME PROBLEMLERİNİN ÇÖZÜMÜ İÇİN DAL-SINIR ALGORİTMASI TEMELİNE DAYANAN ÇÖZÜM YÖNTEMİ

5.1. Problem için Geçerli Olan Bazı Alt Sınırlar

Alt sınır hesaplama işleminin, optimal çözümü mevcut yöntemlerle makûl zamanda bulunamayan problemlere veya alt problemlere uygulandığı aşikârdır. Daha önce de belirtildiği gibi bu işlemdeki temel yaklaşım, problemin çözümünü zorlaştıran bir veya birden fazla kısıtın ihmal edilmesi şeklindedir. Problemde yapılacak kısıt ihmali, amaç fonksiyonu değerinin aynı kalması veya daha da iyileşmesi anlamına gelecektir. Örneğin spesifik bir minimizasyon probleminde yapılacak bir kısıt ihmaliyle elde edilen problemin optimal amaç fonksiyonu değeri, orijinal problemin optimal amaç fonksiyonu değerinden büyük (yani kötü) olmayacaktır. Dolayısıyla ilgili değer orijinal problemin amaç fonksiyonu değeri için bir alt sınır değeri olacaktır.

Literatürde BMHDP-1 için önerilmiş çeşitli alt sınır hesaplama yöntemleri vardır (Bkz. Scholl [5]). Becker ve Scholl [15], problemin kutu paketleme probleme indirgenerek yeni probleme hesaplanan alt sınır değerinin kullanılması temeline dayanan ve sırasıyla AS_1 , AS_2 ve AS_3 olarak adlandırılan yöntemlerin inceledikleri DA-MHDP için de geçerli olduğunu ifade etmiştir. Daha önce de belirtildiği gibi DA-MHDP'nin bu çalışma kapsamında incelenen Pİ-MHDP1'den ayıran temel yapısı montaj pozisyonları ve bunlarla ilişkili atama kısıtlarını içermesidir. AS_1 , AS_2 ve AS_3 değerlerinin hesaplanmasında dikkate alınacak problem ise montaj pozisyonları gibi Pİ-MHDP1'de olmayan hiçbir ilave kısıtı kapsamamaktadır. Dolayısıyla bu alt sınır hesaplama yöntemleri Pİ-MHDP1 için de geçerlidir. Buna karşın Becker ve Scholl [15]'ün önerdiği dal-sınır algoritması istasyon tabanlı bir dallandırma yapısına sahiptir. Bu nedenle ifade edilen alt sınır hesaplama yöntemleri BMHDP-1'de uygulandığı gibi uygulanabilir. İlerleyen bölümlerde önerilecek yeni algoritma ise görev tabanlı bir strateji kullanmaktadır. Bu noktadan hareketle bahsi geçen alt sınır

hesaplama yöntemlerinin uygulanması bir miktar farklıdır. AS_1 'in uygulanmasına ilişkin detaylar bölümün ilerleyen aşamalarında sunulmuştur.

Bu çalışma kapsamında AS_1 'den hareketle oluşturulmuş yeni bir alt sınır hesaplama yöntemi geliştirilmiştir. Yöntemin ifade edilebilmesi amacıyla aşağıdaki tanımlamalara ihtiyaç vardır.

yK : Bir sonraki görev atamasının yapılacağı istasyon indeksi,

mc_{p-q} : $1 \leq p \leq q \leq k_{Enb}$ olmak üzere $p, p+1, \dots, q-1, q$ istasyonlarında bulunan ve en az 1 adet görev atanmış işçilerin sayısı,

tt_{p-q} : $1 \leq p \leq q \leq k_{Enb}$ olmak üzere $p, p+1, \dots, q-1, q$ istasyonlarına ataması yapılmış veya ilerleyen düğümlerde kesinlikle ataması bu istasyonlara yapılacak olan görevlerin verilen kısmi çözüme göre düzeltilmiş görev zamanları (\bar{t}_i) toplamı.

Verilen bir kısmi çözüm için tt_{p-q} ifadesi hesaplanırken, ataması yapılmış olan görevlerin E_i ve L_i değerleri ile en erken başlama zamanı (ES_i) ve en geç başlama zamanı (LS_i) değerleri atama bilgilerine göre alınıp, artık problemi oluşturan kalan görevlerin ilgili değerleri yeniden hesaplanır. Görev sürelerinin verilen bir kısmi çözüme göre düzeltilmesi işlemi şu şekilde yapılmaktadır. Eğer görev atanmamışsa, yani artık problem kapsamında bulunan bir görevse, görev süresi aynen korunur. Eğer görevin ataması yapılmışsa görev süresi şu şekilde düzeltilir:

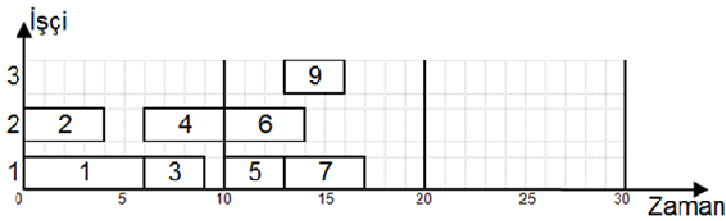
- Eğer görevin atandığı işçide daha önce atanmış başka bir görev varsa, işçinin önceki görevi tamamlama zamanı ile bu göreve başlama zamanı arasında kalan işçi bekleme süresi görev süresine eklenir.
- Eğer görev işçiye atanmış ilk görevse, işçinin istasyon başlangıç zamanından bu görevin başlama zamanı arasında kalan işçi bekleme süresi görev süresine eklenir.

- Eğer görev, ataması tamamlanmış istasyonda bulunan işçiye atanmış son görev ise, işçinin bu görevi tamamlamasından istasyon bitiş zamanına kadar bekleme süresi görev süresine eklenir.

Verilen bir kısmi çözüm için Eş. 5.1 kullanılarak hesaplanan değer işçi sayısı alt sınır değeri olacaktır:

$$AS = mc_{1-(yK-1)} + Enb\{mc_{yK-yK}, [tt_{yK-yK} / c]^+\} + [tt_{(yK+1)-k_{Enb}} / c]^+ \quad (5.1)$$

Örnek oluşturması amacıyla $c = 10$, $k_{Enb} = m_{Enb} = 3$, $yK = 2$ ve öncelik diyagramı Şekil 2.1’de verilen 15 görevli problemin, Gant diyagramı Şekil 5.1’deki kısmi çözümü için verilen ifadeyle alt sınır değeri hesaplanacaktır. Problemden bulunan görevlerin verilen kısmi çözüm için gerekli hesaplamaları Çizelge 5.1’de sunulmuştur.



Şekil 5.1. Örnek kısmi çözüme ait Gant diyagramı

Dikkat edilirse mevcut kısmi çözümde 1–7 ve 9 numaralı görevlerin ataması yapılmış olup bu görevlere ilişkin E_i , ES_i , L_i ve LS_i değerleri görevlerin kısmi çözümdeki atama bilgileri kullanılarak yazılmıştır. Ataması yapılmamış olan 8, 10–16 numaralı görevlerin belirtilen bilgileri ise daha önce belirtilen yöntemle uygun şekilde yeniden hesaplanmıştır. Mevcut istasyon $yK = 2$ olup bu istasyondan önce toplam $mc_{1-1} = 2$ adet işçiye atama yapılmıştır. Mevcut istasyonda ise toplam $mc_{2-2} = 3$ adet işçiye atama yapılmıştır. Bu istasyona ataması hali hazırda yapılmış veya ilerleyen düğümlerde kesinlikle ataması yapılacak görevler ise 5, 6, 7 ve 9 numaralı görevler olup bunların düzeltilmiş zamanları toplamı $tt_{2-2}=17$ ’dir.

Çizelge 5.1. Alt sınır örneğine ilişkin hesaplama verileri

Görev (i)	Atanmış	t_i	\bar{t}_i	ES_i	E_i	LS_i	L_i
1	Evet	6	6	0	1	0	1
2	Evet	4	4	0	1	0	1
3	Evet	3	4	6	1	6	1
4	Evet	4	6	6	1	6	1
5	Evet	3	3	10	2	10	2
6	Evet	4	4	10	2	10	2
7	Evet	4	4	13	2	13	2
8	-	5	5	14	2	20	3
9	Evet	3	6	13	2	13	2
10	-	2	2	14	2	23	3
11	-	4	4	20	3	24	3
12	-	3	3	20	3	25	3
13	-	5	5	20	3	25	3
14	-	2	2	23	3	28	3
15	-	2	2	24	3	28	3

İlerleyen düğümlerde mevcut istasyondan kesinlikle sonraki istasyonlara ataması yapılacak olan görevler ise 11-16 numaralı görevler olup bunların toplam görev zamanları ise $mc_{3-3} = 16$ 'dır. Dolayısıyla mevcut çözümün alt sınır değeri aşağıdaki gibi hesaplanacaktır:

$$\begin{aligned}
 AS &= mc_{1-1} + Enb\{mc_{2-2}, [tt_{2-2}/10]^+\} + [tt_{3-3}/10]^+ \\
 &= 2 + Enb\{3, [17/10]^+\} + [16/10]^+ = 7
 \end{aligned}
 \tag{5.2}$$

Düz hatlar için önerilen AS_1 yöntemi Pİ-MHDP1 için aşağıdaki şekilde güncellenebilir:

$$AS_1 = mc_{1-(yK-1)} + Enb\{mc_{yK-k_{Enb}}, [tt_{yK-k_{Enb}}/c]^+\}
 \tag{5.3}$$

Göz önünde bulundurulmuş kısmi çözüm için güncellenmiş AS_1 alt sınır değeri aşağıdaki gibi hesaplanacaktır:

$$AS_1 = mc_{1-1} + Enb\{mc_{2-3}, [tt_{2-3}/10]^+\} = 2 + Enb\{3, [40/10]^+\} = 6
 \tag{5.4}$$

Görüldüğü gibi verilen örnek için, önerilen yöntemle hesaplanan alt sınır değeri güncellenmiş AS_1 'den daha iyi sonuç üretmiştir.

$1 \leq p \leq q \leq k_{Emb}$ olmak üzere verilen bir $p - q$ istasyon grubu için önerilen alt sınır hesaplama yöntemi Eş. 5.5'de verildiği gibi genellenebilir:

$$AS_g = \text{Emb}\{mc_{1-(p-1)}, [tt_{1-(p-1)} / c]^+\} + \text{Emb}\{mc_{p-q}, [tt_{p-q} / c]^+\} + \text{Emb}\{mc_{(q+1)-k_{Emb}}, [tt_{(q+1)-k_{Emb}} / c]^+\} \quad (5.5)$$

5.2. Problemin Çözümü için Önerilen Dal-Sınır Algoritması

Önerilen dal-sınır algoritması, detayları ilerleyen bölümlerde anlatılacağı gibi, hafızada bir veya daha fazla sayıda her biri öncelik değeriyle ilişkilendirilmiş dal-sınır ağaç parçası tutmaktadır. Bu alt ağaçlardan önceliği en yüksek olanı seçilerek algoritması Şekil 5.2'de verilen "Ara()" prosedürü yoluyla arama yapılmaktadır. Ağaçtaki arama işleminin CPU üst sınırı $CPU_{AğaçEmb}$ parametresiyle gönderilen süreyle sınırlı olup, başlangıçta alt ağaçtaki arama sonucunun tutulduğu $SonuçAğaç$ değişkenine arama yapılan alt ağaçta henüz hiçbir uygun çözümün bulunmadığını belirten $ÇözümBulunamadı$ bilgisi aktarılmaktadır. Bu işlemden sonra algoritması Şekil 5.3'de verilen durdurma koşulu sağlanıncaya kadar (yani Durdur() fonksiyonu "Evet" bilgisi çevirinceye kadar) şu işlemler gerçekleştirilir:

- Eğer budama koşulu sağlanmıyorsa (yani budanır() fonksiyonu "Hayır" bilgisi çeviriyorsa) mevcut düğümden (*MevcutDüğüm*) Dallandır() prosedürü aracılığıyla dallandırma yapılır.
- Eğer budama koşulu sağlanıyorsa GeriDön() prosedürü yoluyla mevcut düğüm budanıp, mevcut düğümün ebeveynine dönülür.

```

Procedure Ara( $CPU_{AğaçEnb}$ );
begin
  •  $Sonuç_{Ağaç} \leftarrow ÇözümBulunamadı$ ;
  • while(Durdur() $=$ Hayır)do
    • if (Budandır() $=$ Hayır) then Dallandır();
      else GeriDön();
    end;
  end;

```

Şekil 5.2. Verilen bir alt ağaçta uygun/optimal çözüm arayan prosedür

Arama yapılan alt ağaçta arama işlemi, aşağıda sıralı olarak verilen koşullardan birisinin sağlanması durumunda durdurulmaktadır, yani algoritması Şekil 5.3’de verilen ve durdurma durumunu sınavan Durdur() fonksiyonu “Evet” sonucunu çevirmektedir:

- Optimum çözüm bulunmuşsa (yani $Sonuç_{Ağaç} = OptimumÇözüm$ ise),
- Uygun çözüm bulunmuşsa (yani $Sonuç_{Ağaç} = UygunÇözüm$ ise),
- Verilen alt ağaç ilerleyen bölümlerde detayları anlatılacak olan aranacak en düşük seviyeli düğüme kadar aranmasına rağmen uygun çözüm bulunamadı ise (yani $Sonuç_{Ağaç} = Çözümsüz$ ise),
- Alt ağaçta $CPU_{AğaçEnb}$ süresine eşit ya da daha fazla sürede arama yapılmışsa (yani $CPU_{Ağaç} \geq CPU_{AğaçEnb}$ ise).

```

bool Durdur();
begin
  • if( $Sonuç_{Ağaç} = OptimumÇözüm$ ) then Durdur $\leftarrow$  Evet;
    else if( $Sonuç_{Ağaç} = UygunÇözüm$ ) then Durdur $\leftarrow$  Evet;
    else if( $Sonuç_{Ağaç} = Çözümsüz$ ) then Durdur $\leftarrow$  Evet;
    else if( $CPU_{Ağaç} \geq CPU_{AğaçEnb}$ ) then Durdur $\leftarrow$  Evet;
    else Durdur $\leftarrow$  Hayır;
  end;

```

Şekil 5.3. Verilen bir dal sınır ağacında aramanın durdurulmasını sınavan fonksiyon

Önerilen algorithmada literatüde “Lazer Arama” olarak isimlendirilen ve her seferinde

tek bir alt düğümün oluşturularak, oluşturulan bu alt düğüm ile ilerlendiği arama stratejisi kullanılmıştır. Bu nedenle, dallandırma işlemi yoluyla her seferinde tek bir alt düğüm oluşturulup bu düğüm mevcut düğüm olarak alınmaktadır.

5.2.1. Başlangıç işlemleri

Başlangıç işlemleri kapsamında problemdeki tüm görevlerin takipçisi olan ve görev zamanı sıfır olarak alınan bir batak düğüm probleme eklenmektedir. Bu hayali görevin her hangi bir işçiye atanması tam bir çözümün birerlendiği anlamına gelmektedir. Yine başlangıç işlemlerinde görevler birikimli pozisyon ağırlık değerlerine göre büyükten küçüğe sıralanarak görev numaraları belirlenmektedir ($EnbKumP_t$ kriteri). Verilen bir i görevine ait birikimli pozisyon ağırlık değeri ($KumPA_i$) aşağıda verilen Eş. 5.6'nın kullanımı yoluyla hesaplanmaktadır [5]:

$$KumPA_i = t_i + \sum_{h \in F_i} KumPA_h \quad (5.6)$$

Sıralamada $KumPA_i$ değerleri eşit olan görevlerden orijinal numarası küçük olana öncelik verilmektedir.

5.2.2. Dallandırma işlemi

Dal-sınır algoritmalarının en önemli bileşenlerinden birisi olan dallandırma işlemi daha önce de belirtildiği gibi mevcut problemin çözümü daha kolay olan alt problemlere ayrıştırılmasına yönelik bir stratejidir. Bu yönü itibariyle kullanılan strateji doğrudan taranacak çözüm sayısını belirleyeceğinden, dal-sınır algoritmasının performansını doğrudan etkileyecektir. Bu nedenle dallandırma stratejisi dal-sınır algoritmasının etkinliği açısından oldukça önemlidir.

Önerilen algorithmada dallandırma işlemi görev tabanlı dallandırma stratejisine benzer bir yöntem kullanılarak yapılmaktadır. Basit montaj hatlarında bu stratejinin

kullanımı oldukça kolaydır. Mevcut istasyona atanabilecek görevlerin listesi oluşturulduktan sonra bu listenin her bir elemanı için, ilgili elemanın istasyonda atanabileceği en erken zamana yerleştirilmesi yoluyla bir adet alt problem oluşturulmaktadır [5]. Buna karşın göz önünde bulundurulacak problemde her bir istasyon üst sınır m_{Emb} olmak üzere birden fazla işçiye sahip olabilmektedir. Dolayısıyla mevcut istasyona atanabilecek görev listesinden seçilen elemanın hangi istasyonun hangi işçisine atanacağı ve atanacak işçinin bu göreve hangi zamanda başlayacağı sorularının da oluşturulacak dallandırma prosedürü tarafından karşılanması gerekir. Dolayısıyla kullanılan dallandırma stratejisi bu noktalarda görev tabanlı dallandırma stratejisinden ayrılmaktadır.

Pİ-MHDP1, görevlerin eş zamanlı yapılmasına izin vermesi açısından paralel makine çizelgeleme problemine (PMÇP) ve KKPPP'e benzerlik göstermektedir. PMÇP'nin Pİ-MHDP1'den en önemli farkı; görevler arasında öncelik ilişkilerinin bulunmaması, paralel çalışan makinelerin her birinin çizelgeleme periyodu boyunca kullanıma müsait olması veya bir birinden bağımsız bozulma periyotlarına sahip olması, performans ölçütlerinin kullanılan makine sayısı ile doğrudan ilişkisinin bulunmamasıdır. Pİ-MHDP1'e bu açılardan bakıldığında öncelik ve maksimum istasyon sayısı kısıtlarına ilave olarak istasyonlardaki işçilerin sadece istasyon başlangıç ve bitiş zamanları arasında operasyon yapabilmesi kısıtının altında, PMÇP'nde makinelere karşılık gelen işçilerin toplam sayısının en küçüklenmesi gerekmektedir. Çünkü problemlerdeki kaynaklar sırasıyla makineler ve işçilerdir.

PMÇP'nde kullanılan dallandırma stratejileri atanmamış operasyonların her birisinin makinelerden birine atanması yoluyla en fazla makine sayısı ve atanmamış görev sayısının çarpımı kadar düğüm oluşturulması temeline dayanmaktadır [79]. Bu problemlerde, seçilen operasyonun atandığı makinede yapılması makinenin bozulma gibi nedenlerle uygun olmaması durumlarının dışında hemen mümkün olmaktadır. Buna karşın incelenen problemlerde bu yapı bazı durumlarda öncelik kısıtları nedeniyle mümkün olmamakta, görevin istasyon başlangıç ve bitiş zamanları arasında yapılması gerekmekte, işçinin göreve başlayabilmesi için atanmış görevin

öncüllerinin tamamlanmasına deęin boş bekletilmesi hatta öncülleri tamamlanmış olsa bile işçinin ilerleyen zamanlarda atanabilecek hale gelecek başka bir görev için bekletilmesi gerekmektedir.

Kaynakların amaç fonksiyonunda deęil kısıtlarda göz önünde bulundurulduęu dięer bir problem sınıfı olan KKPPP için önerilen dal-sınır algoritmalarında kullanılıp, kısmi çözümün oluşturulması temeline dayanan ana dallandırma stratejileri ise mevcut kısmi çözüme (yani kısmi çizelgeye) her seferinde tek bir aktivitenin atanması ve her seferinde bir aktivite grubunun atanması şeklinde iki temel sınıfa ayrılabilir. İlk sınıfta bulunan stratejide dallandırma işlemi iki farklı yöntemle yapılmaktadır. İlk yöntemde tüm öncülleri atanmış faaliyetlerin her biri mümkün ilk başlama noktasına atanarak birer adet alt problem oluşturulurken, ikinci yöntemde her bir faaliyet en erken başlama noktasına atanarak ilki, listedeki dięer faaliyetlere atama öncelięi tanınarak da dięeri oluşturulup iki adet alt problem türetilir. Birden fazla faaliyetin kısmi çözüme eklenmesi yoluyla alt problemlerin oluşturulduęu ikinci sınıf stratejilerde ise yine iki farklı yöntem kullanılmaktadır. İlk yöntemde öncülleri mevcut zamanda tamamlanan faaliyetler ve mevcut zamanda halen devam eden faaliyetlerden oluşan iki farklı set oluşturulup ilk sette bulunan görevler mevcut zamana kaynak kısıtını ihmal etmeden atanması durumunda tüm görevler atanarak mevcut zaman atanana bu setteki en erken tamamlanma zamanına ilerletilir. Kaynak kısıtı ihmal ediliyorsa en küçük bekletme alternatif alt setlerinin tamamı birerlenerek her bir alt set için birer adet alt problem oluşturulur. Bir setin en küçük bekletme alt seti olabilmesi için kendisinin en küçük bekletme alt seti içermemesi gerekir. Kullanılan ikinci yöntemde ise dal sınır algoritmasına başlanmadan önce aynı anda yapılabilecek tüm faaliyet alt setleri birerlenir. Arama sürecindeki mevcut düęüm için düęüm mevcut zamanında işlemleri devam eden faaliyetler ve mevcut zamana atanabilecek (tüm öncülleri mevcut zamandan önce tamamlanmış) faaliyetler tespit edilir. Bu faaliyetlerin algoritma başlangıcında belirlenen her bir alt setinin mevcut zamana atanması yoluyla birer adet alt problem oluşturulur [80].

Pİ-MHDP-1'in çözümünde, daha önce belirtilen farklılıkları nedeniyle PMÇP ve KKPPP için literatürde önerilen dallandırma stratejilerinin direkt kullanılması

mümkün değildir. İstasyon başlangıç ve bitiş zamanları kısıtları altında PMÇP’de kesikli yapıda olması gereken makinelere, KKPPP’de yine kesikli yapıda bulunması gereken kaynaklara karşılık gelen işçi sayısının en küçüklenmesi için problemin tüm çözüm uzayını tarayacak şekilde ilave mekanizmaların geliştirilerek dallandırma stratejisine eklenmesi gerekmektedir.

Önerilen dal-sınır algoritmasında ağaçta bulunan her bir düğüm spesifik bir görevin atanmasına veya ilerleyen aşamalarda tanımlanacak olan işçi bekletme işlemine karşılık gelmekte olup aşağıdaki bilgileri tutar:

- Düğümün seviyesi (*Seviye*),
- Atanan görev (*G*),
- Atamanın yapıldığı istasyon ve işçi (*mK, mM*),
- Atanan işçinin *G*’ye başlama zamanı yani atama zamanı (*mZ*),
- Düğüm alt sınır değeri (*AS*),
- Toplam sayısı m_{Enb} olan ve mevcut istasyonda bulunan her bir işçi için atanan görevlerin toplam zamanı (yani işçi yükleri, $\dot{I}Y_m$), yapılabilecek bir sonraki atama zamanı ($\dot{I}Z_m$) ve atandığı en son görevi tamamlama zamanı yani boş kalma zamanı ($\dot{I}B_m$),
- Sonraki atama zamanı, yani mevcut zaman ($yZ = \min_{1 \leq m \leq m_{enb}} \{\dot{I}Z_m\}$)
- Sonraki atamanın yapılacağı istasyon indeksi (*yK*) ve işçi indeksi (*yM*),
- Bir sonraki düğümde atanabilecek görevlerin listesi (atama listesi, *AL*),
- En son oluşturulan alt düğümde atanan görevin *AL*’deki sırası (*yIndeks*),
- Düğümde uygunluk kontrolü yapıldıysa Evet yapılmasıysa Hayır değeri alan değişken ($K_{Uygunluk}$),
- Düğümde baskınlık kontrolü yapıldıysa Evet yapılmasıysa Hayır değeri alan değişken ($K_{Baskınlık}$),
- Düğümün alt sınır değeri hesaplandıysa Evet hesaplanmadıysa Hayır değeri alan değişken (K_{AS}),

- Kök düğümünden itibaren ataması yapılan maksimum görev indeksi ($enbG$),
- Kök düğümünden itibaren ataması yapılan görevlerin toplam süresi ($topG$),
- Kök düğümünden itibaren görev atanmış toplam işçi sayısı ($topM$),
- Alt düğüm referansı ($Yavru$) ve ebeveyn düğüm referansı ($Ebeveyn$),

Belirtilen parametrelerden zamanlara ilişkin olanlar her bir k istasyonu için 0 zamandan çevrim zamanına kadar değil, $(k-1) \times c$ zamanından $k \times c$ zamanına kadar değer alabilmekte olup bu yapı literatürde saat zamanı olarak isimlendirilmektedir. Çalışma kapsamında bu değerlerden ilki istasyon başlangıç zamanı, ikincisi ise istasyon bitiş zamanı olarak isimlendirilmiştir. Bu yaklaşım mevcut zamanda atanabilecek görevlerin listesinin oluşturulması gibi birçok işlemde kolaylık sağlamaktadır.

Ağaçtaki her bir düğüm kök düğümünden başlanıp ilgili düğüme kadar tüm düğümlerin incelenmesi yoluyla oluşturulabilecek bir kısmi çözümü ifade eder. Artık problemin içerdiği görevlerin toplam süresi t_{toplam} değerinden mevcut düğümün $topG$ değerinin çıkartılmasıyla bulunabilirken, görevlerin listesine ulaşmak amacıyla düğümlerden bağımsız olarak ağaç kapsamında tutulan ve görev atamaları (GA) adı verilen bir diziden faydalanılmaktadır. GA , problemdeki her bir görev için bir adet eleman içermekte olup ataması yapılmamış görevlerin elemanları boş bırakılırken ataması yapılan görevler için atanmanın yapıldığı düğümünün referansı kaydedilir. Bu yolla, örneğin, atanabilecek görevlerin listesi kök düğümünden itibaren tüm düğümlerin kontrol edilmesi yerine GA yardımıyla hızlıca oluşturulabilmektedir.

Lazer arama stratejisinin kullanılması ve düğümlerin görev atamalarına karşılık gelmesi nedeniyle ağaçta bulunan toplam düğüm sayısı daha sonra açıklanacak olan işçi bekletme işlemi de göz önünde bulundurulduğunda $n \times (m_{Enb} - 1)$ 'den fazla olmayacaktır. Öyle ki, atanacak her bir görev için tüm işçilerin bekletilmesi gereksiz boş zaman oluşturacaktır.

Mevcut düğümden sonra oluşturulacak düğüm kapsamında yapılacak atama, mevcut düğümdeki $\dot{I}Z_m$ değeri en küçük olan işçiye yapılmaktadır. Dolayısıyla atama işlemi zaman tabanlı takip edilmekte olup yM değeri $\dot{I}Z_m$ değeri en küçük olan işçiye eşit olacaktır. $\dot{I}Z_m$ değeri açısından birden fazla işçide bir beraberlik söz konusu ise indeksi en küçük olan işçi yM değerine aktarılır.

Dallandırma işleminin prosedürü Şekil 5.4’de verilmiştir. Bu prosedürde bulunan $\dot{I}şçiYuk$, $\dot{I}şçiSayı$ ve $\dot{I}lk\dot{I}şçiAtama$ ifadelerine ilişkin açıklamalar ilerleyen bölümlerde sunulacaktır. Dallandırma işlemi, hiçbir görevin atanmadığı kök düğüm dışında kalan diğer düğümlerin oluşturulmasında kullanılır. Kök düğümde $\dot{I}Y_m$, $\dot{I}Z_m$ ve $\dot{I}B_m$ değerlerinin tamamına ve $Seviye$, $topG$ ve $topM$ değerlerine sıfır, $enbG$ değerine ise $-\infty$ atanır. AL listesi daha sonra açıklanacak yöntemle oluşturulduktan sonra $yIndeks$ değeri boşaltılır. yZ değerine sıfır, yK ve yM değerlerine 1 atanması yoluyla sonraki atamanın ilk istasyonun birinci işçisine sıfır zamanında yapılacağı belirtilir.

Dallandırma işlemi ağaçtaki seviyesi en büyük olan $MevcutDüğüm$ ’ün incelenmesiyle başlar. Yeni bir düğüm oluşturularak yeni düğümün $Ebeveyi$ değerine $MevcutDüğüm$ ’ün referansı, $MevcutDüğüm$ ’ün $Yavru$ değerine de yeni düğüm referansı kaydedilir. Yeni düğüme gerekli tüm bilgiler $MevcutDüğüm$ ’den kopyalanıp düğümde sırasıyla uygunluk kontrolü, baskınlık kontrolü ve alt sınır hesaplama işlemlerinin yapılmadığını (yani düğümün yeni oluşturulduğunu, geri gitme yoluyla mevcut düğüm seçilmediğini) belirtmek amacıyla $K_{Uygunluk}$, $K_{Baskınlık}$ ve K_{AS} ifadelerine “Hayır” değeri atanır. Bu işlemlerden sonra mevcut düğümün $yIndeks$ değeri bir artırılıp yine ebeveyn düğümün AL listesinde $yIndeks$ sırasında bulunan görev yeni oluşturulan düğüm yoluyla mevcut düğümün yK ve yM değerleri kullanılarak yK istasyonunun yM işçisine atanır. Yeni düğümdeki mZ değeri mevcut düğümdeki yZ değeri olup, yeni düğümdeki $enbG$, $topG$, $topM$

```

Procedure Dallandır();
begin
  • if (MevcutDüğün.Yavru=Boş Değer) MevcutDüğün.Yavru←yeni düğüm
    oluştur;
  • MevcutDüğün.Yavru bilgilerini MevcutDüğün'den kopyala;
  • MevcutDüğün.Yavru.Ebeveyi← MevcutDüğün;
  • MevcutDüğün← MevcutDüğün.Yavru;
  • MevcutDüğün.Seviye← MevcutDüğün.Seviye+1;
  • MevcutDüğün.KUygunluk ← Hayır;
  • MevcutDüğün.KBaskınlık ← Hayır;
  • MevcutDüğün.KAS ← Hayır;
  • MevcutDüğün.Ebeveyi.yIndeks← MevcutDüğün.Ebeveyi.yIndeks+1;
  • MevcutDüğün.G← MevcutDüğün.Ebeveyi.AL[MevcutDüğün.Ebeveyi.yIndeks];
  • MevcutDüğün.mK← MevcutDüğün.Ebeveyi.yK;
  • MevcutDüğün.mM← MevcutDüğün.Ebeveyi.yM;
  • update GA, İYm, İZm, İBm, İşçiYuk, İlkİşçiAtama, topG, enbG, topM,
    yK, yM, yZ;
  • MevcutDüğün.yIndeks←Boş değer;
  • MevcutDüğün.AL listesini oluştur;
  • while (MevcutDüğün.AL={}) do
    • MevcutDüğün'e zaman ilerletme işlemi uygula;
    • MevcutDüğün.AL listesini oluştur;
  • end;
  • if not ((MevcutDüğün.yM=1)AND(İY1=0)) then
    begin
    • if (İşçiSayı[MevcutDüğün.yK]≥MevcutDüğün.yM) OR
      (MevcutDüğün.İYMevcutDüğün.yM>0) then MevcutDüğün.AL listesinin
      sonuna Beklet ekle;
      else MevcutDüğün.AL listesinin başlangıcına Beklet ekle;
    • end;
  • end;

```

Şekil 5.4. Dallandırma işlemine ilişkin prosedür

değerleriyle birlikte $\dot{I}Y_m$, $\dot{I}Z_m$ ve $\dot{I}B_m$ değerleri güncellenir. Ağaç kapsamında tutulan GA güncellenerek yeni düğümün $yIndeks$ değeri boşaltılır ve AL listesi daha sonra açıklanacak yöntemle oluşturulur. Böylelikle yeni düğüm mevcut düğüm

olacaktır.

Atama işleminin bir sonraki düğümde yeni istasyona geçmesi yani yK değerinin bir birim artırılması AL listesinin oluşturulması sırasında yapılmaktadır. Bu işlem sonucunda yeni istasyonun açılmasına karar verilirse mevcut düğümün $\dot{I}Z_m$ ve $\dot{I}B_m$ değerleri açılacak yeni istasyonun başlangıç zamanına eşitlenirken, $\dot{I}Y_m$ değerleri sıfıra eşitlenir.

Yeni oluşturulan düğümün AL listesi aşağıdaki koşulları sağlayan görevleri numaralarına göre küçükten büyüğe içerecek şekilde oluşturulur:

- Ataması yapılmamış,
- Tüm öncülleri atanmış,
- Tüm öncülleri yZ zamanında veya daha öncesinde tamamlanmış,
- yZ zamanına atanması durumunda en geç yK istasyonunun istasyon bitiş zamanında tamamlanabilen.

Eğer atanmamış görev olmasına karşın belirtilen koşulları sağlayan hiçbir görev bulunamamışsa, yani atama listesi boşsa, mevcut düğüme zaman ilerletme işlemi uygulanır. Zaman ilerletme prosedürü $\dot{I}Z_m$ değeri yZ 'ye eşit olan tüm işçileri içeren bir R kümesini oluşturarak işleme başlar. R kümesindeki işçilerin $\dot{I}Z_m$ değerleri Eş. 5.7 yoluyla $\dot{I}Z_m$ değeri yZ 'den büyük olan en küçük $\dot{I}Z_m$ değerine eşitlenir:

$$\dot{I}Z_r = \min_{1 \leq m \leq m_{Eab}} \{\dot{I}Z_m \mid \dot{I}Z_m > yZ\} \quad (r \in R) \quad (5.7)$$

Böyle bir işçi bulunamazsa mevcut istasyon kapatılıp yeni bir istasyon açılarak daha önce ifade edilen yapıya uygun şekilde düğüm bilgileri güncellenir. Zaman ilerletme işleminden sonra AL tekrardan oluşturulur. AL yine boşsa listede en az bir adet görev bulunana kadar zaman ilerletme işlemi ardı ardına uygulanır.

Bilindiği gibi PM-MHDPI'in amaç fonksiyonu kullanılan işçi sayısının en küçüklenmesidir. Dolayısıyla bazı düğümlerde işçilerden bazılarının görev atanmasının yapılmaması yani işçinin belirli bir süre bekletilerek ilerleyen zamanlarda atanabilir hale gelecek görevlerin atanması veya işçinin hiç kullanılmaması gerekir. Bu mekanizmanın dallandırma prosedürüne eklenmesi amacıyla "işçi bekletme" kavramı geliştirilmiştir. İşçi bekletmede yZ zamanında yM işçisine görev atanması yapılmadan (boşluk yani *Beklet* değeri atanarak) yM işçisinin $\dot{I}Z_m$ değeri yZ 'den büyük olan en küçük işçi atama zamanına eşitlenir. Böyle bir işçi bulunamazsa işçi atama zamanı istasyon bitiş zamanı olarak alınır. Bekletilen işçiye daha önce hiç görev atanması yapılmamışsa bu işçiden sonraki bütün işçiler de bu işçi ile aynı zamana kadar bekletilir. Bunun nedeni atama yapılmayan işçilerin birbirinden farksız olduğudur. İşçi bekletme işleminin sadece hiç atama yapılmayan işçilere değil daha önce görev atanmış işçilere de uygulanması gerekir. Bekletme işleminin uygulanmadığı tek düğüm yeni açılan istasyonun ilk işçisine ilk atamanın yapılacağı düğümdür. Bu noktada yapılacak bekletmede, istasyondaki tüm işçilerin $\dot{I}Z_m$ değerleri sıfır olduğundan tüm işçiler istasyon bitiş zamanına kadar bekletilmiş olacak, istasyon hiç açılmayacak, bu yolla diğer görevlerin atanmasına fırsat vermediği gibi gereksiz boş zaman oluşturacaktır.

5.2.3. Sınırlama işlemi

Sınırlama işleminde mevcut düğüme karşılık gelen kısmi çözümün, artık problem de göz önünde bulundurulduğunda ne kadar iyi bir çözüm üreteceğine dair bir alt sınır değeri hesaplanmaktadır. Önerilen dal-sınır algoritması kapsamında düğümlerde alt sınır değerinin hesaplanmasında hesaplama külfetinin düşük olması nedeniyle formülasyonu Bölüm 5.2 kapsamında sunulan güncellenmiş AS_1 alt sınır hesaplama yöntemi kullanılmıştır.

5.2.4. Budama işlemi

Mevcut düğümün budanıp budanmayacağını kontrolüne ilişkin prosedürü Şekil

5.5’de verilmiştir. Verilen prosedüre göre sıralı olarak verilmiş aşağıdaki koşullardan birisinin sağlanması durumunda mevcut düğüm budanır:

- Bütün görevler atanarak tam çözüm bulunmuştur,
- Mevcut düğümün bütün alt düğümleri oluşturularak aranmıştır,
- Yapılan uygunluk kontrolü sonucunda mevcut düğümden aramaya devam etmenin uygun bir çözüme ulaştırmayacağı anlaşılmıştır,
- Yapılan baskınlık kontrolü sonucunda mevcut düğümün ifade ettiği kısmi çözümün daha önce aranmış başka bir kısmi çözüm tarafından bastırıldığı anlaşılmıştır,
- Mevcut düğüm için hesaplanan AS değeri global üst sınır değerine ($\dot{U}S$) eşit veya daha büyüktür.

Tam çözümün bulunması durumunda bu çözümün amaç fonksiyonu değeri $\dot{U}S$ değeriyle karşılaştırılarak daha iyi bir çözüm bulunmuşsa $\dot{U}S$ değeri güncellenir ve bulunan çözüm aday çözüm olarak hafızaya kaydedilir.

Uygunluk kontrol işlemleri sırasıyla aşağıda verilmiştir:

- Ataması yapılmamış olup mevcut düğümün atama listesinde de olmayan ancak kritik yol hesabıyla yapılan en son başlama zamanı mevcut düğümün yZ zamanına eşit veya küçük olan görevlerin bulunması (bahsedilen görevlerin bulunup bulunmadığı AL listesinin oluşturulması sırasında kontrol edilmektedir),
- Mevcut düğümden sonra yeni atama işlemlerine yeni açılacak istasyondan devam edilecek olması ancak yeni istasyonun k_{Enb} ’dan büyük olması,
- Kalan kapasitenin ataması yapılmamış görevlerin toplam süresinden küçük olması (verilen bir düğüm için kalan kapasite $k_{Enb} \times m_{Enb} \times c - \sum_{m=1}^{m_{Enb}} \dot{I}Z_m$ ifadesiyle hesaplanmaktadır).


```

bool Budandır();
begin
• if (Tüm görevler atanmış)then
  begin
  •  $SonuçAğaç \leftarrow UygunÇözüm;$ 
  • if ( $ÜS > MevcutDüğün.topM$ ) then  $ÜS \leftarrow MevcutDüğün.topM;$ 
  • if ( $ÜS = AS_{teorik}$ ) then  $SonuçAğaç \leftarrow OptimumÇözüm;$ 
  • Budandır←Evet;
  • Prosedürden çık;
  end;
• if ( $MevcutDüğün.yIndeks$  değeri  $MevcutDüğün.AL$  listesinin son elemanına eşit) then
  begin
  • Budandır←Evet;
  • Prosedürden çık;
  end;
• if ( $MevcutDüğün.K_{Uygunluk} = Hayır$ )then
  begin
  •  $MevcutDüğün.K_{Uygunluk} \leftarrow Evet;$ 
  • if ( $MevcutDüğün$  uygun değil) then
    begin
    • Budandır←Evet;
    • Prosedürden çık;
    end;
  end;
• if ( $MevcutDüğün.K_{Baskınlık} = Hayır$ )then
  begin
  •  $MevcutDüğün.K_{Baskınlık} \leftarrow Evet;$ 
  • if ( $MevcutDüğün$  bastırılır) then
    begin
    • Budandır←Evet;
    • Prosedürden çık;
    end;
  end;
• if ( $MevcutDüğün.K_{AS} = Hayır$ ) then  $MevcutDüğün.AS$  değerini hesapla;
• if ( $MevcutDüğün.AS \geq ÜS$ ) then
  begin
  • Budandır←Evet;
  • Prosedürden çık;
  end;
• Budandır←Hayır;
end;

```

Şekil 5.5. Mevcut düğümün budanıp budanmayacağını kontrol eden prosedür

Önerilen algoritma kapsamında 5 farklı baskınlık kontrolü uygulanmaktadır. Bu kontrollerden “Bekleme baskınlığı” ve “Atama baskınlığı” olarak isimlendirilen ilk

iki kontrol her atama gerçekleştikten sonra uygulanırken diğerleri her yeni istasyon açıldığı anda uygulanmaktadır. Dolayısıyla baskınlık kontrolü kapsamında yapılan işlemler sırasıyla aşağıdaki şekildedir:

- Bekleme baskınlık kontrolü,
- Atama baskınlık kontrolü,
- Uygun eş-set baskınlık kontrolü
- Jackson baskınlık kontrolü,
- Uygun alt-set baskınlık kontrolü.

Algoritma kapsamında budamadan kasıt, mevcut düğümün ebeveyninin mevcut düğüm olarak alınmasıdır. Eğer mevcut düğüm kök düğümse veya ilerleyen bölümlerde detaylarından bahsedilecek olan aranması gereken en düşük seviyeli düğüm ise ($Seviye_{Enk}$) arama işlemi durdurulur. Değilse, gerekli parametreler güncellenerek mevcut düğümün ebeveynine hareket edilir. GeriDön (backtracking) olarak isimlendirilen bu işleme ait prosedür Şekil 5.6'da sunulmuştur.

```

Procedure GeriDön();
begin
  • if ((  $MevcutDüğüm . Seviye=0$ ) OR (  $MevcutDüğüm . Seviye= Seviye_{Enk}$ 
    )) then
    begin
      •  $Sonuç_{Ağaç} \leftarrow Çözümsüz;$ 
      • Prosedürden çık;
    end;
  •  $İlkİşçiAtama, İşçiYuk, GA$  ifadelerini güncelle;
  •  $MevcutDüğüm \leftarrow MevcutDüğüm . Ebeveyr;$ 
end;

```

Şekil 5.6. Geri dönüş (backtracking) işlemine ilişkin prosedür

Bekleme Baskınlık Kontrolü

Mevcut düğüm tarafından i görevinin f işçisine atandığı varsayalım. Bu durumda

aşağıdaki özellikler sağlanıyorsa mevcut düğümün ifade ettiği kısmi çözüm daha önce taranmış başka bir kısmi çözüm tarafından bastırılır:

- i görevinin atandığı f işçisi daha önce bekletilmiştir, yani f işçisi i görevinin atandığı zamandan önce en az 1 zaman birimi boş beklemiştir.
- i görevinin tüm öncülleri i görevinin başlama zamanından en az 1 zaman birimi önce tamamlanmıştır, yani i görevi aynı f işçisine en az 1 zaman birimi daha erkene atabilir.

Dallandırma stratejisi gözden geçirildiğinde atama zamanlarının her zaman en az bir görevin tamamlanma zamanına veya istasyon başlangıç zamanlarına karşılık geldiği anlaşılmaktadır. Diğer bir ifadeyle aynı zaman diliminde bütün işçilerin bekletilmesi sadece yeni istasyon açılacaksa söz konusu olabilmektedir. Yine dallandırma stratejisi her bir görevin bitiş zamanını atama zamanı olarak incelemektedir. Dolayısıyla i görevinin atandığı f işçisi, ya boşta kalma zamanında i görevi atanarak bir düğüm oluşturulmuş (yani i görevinin tüm öncülleri f işçisinin boşta kalma zamanından daha önce tamamlanmıştır) veya f işçisi i görevinin tüm öncüllerinin tamamlandığı zamana kadar bekletilerek bir düğüm oluşturulmuştur (yani i görevinin tüm öncülleri f işçisinin boşta kalma zamanından sonra tamamlanmıştır). Belirtilen özelliklerin sağlanması durumunda her halükarda i görevi f işçisine mevcut çözümde atandığı zamandan daha erkene atanarak diğer işçilerdeki atamalar aynı kalacak şekilde en az bir kısmi çözüm daha önce oluşturulmuştur. Diğer işçilerdeki atamaların aynı kalması şu açıdan mümkün olacaktır. i görevi daha erken bir zamana atanabiliyorsa i görevinin takipçileri daha erken zamanda atanabilir olacak, diğer düğümlerdeki AL listesi en kötü şartlarda değişmeyecektir. i görevinin erken pozisyona atanması mümkün iken işçinin bekletilerek daha sonraki bir zamana atanmasının yapılması işçi etkinliği açısından amaç fonksiyonuna hizmet etmeyen bir sonuç doğuracaktır. Dolayısıyla mevcut kısmi çözümden daha iyi olan, en kötü şartlarda aynı kalitede tam çözüme ulaştıracak olan başka bir kısmi çözüm daha önce oluşturulduğundan mevcut düğüm elimine edilebilir.

Atama Baskınlık Kontrolü

Mevcut düğüm tarafından i görevinin, bu düğümün ebeveyni tarafından da j görevinin atandığı varsayalım. Bu durumda aşağıdaki özellikler sağlanıyorsa mevcut düğümün ifade ettiği kısmi çözüm daha önce oluşturulmuş başka bir kısmi çözüm tarafından bastırılır:

- i görevinin görev indeksi j görevinden küçüktür,
- i ve j görevleri aynı zamana atanmıştır,
- i ve j görevlerinin görev zamanları sıfırdan büyüktür. (yani probleme sonradan eklenen hayali görevler değildir).

Zamanları sıfırdan büyük olan iki farklı görevin aynı zamanda başlayacak şekilde atanabilmesi bu görevler arasında öncelik ilişkisinin olmadığını gösterir. Dolayısıyla mevcut düğümde atanan i görevi ebeveyn düğümde de AL listesinin bir elemanıdır. Mevcut düğümdeki i görevinin indeksi ebeveyn düğümdeki j görevinin indeksinden küçük olduğundan AL listesinde i görevi j 'den önce gelir. Dolayısıyla ebeveyn düğümün i 'yi, alt düğümünün de j 'yi aynı zamana atadığı başka bir kısmi çözüm daha önce oluşturulmuştur. Mevcut kısmi çözümle daha önce oluşturulmuş kısmi çözüm arasındaki tek fark i ve j görevlerinin atandıkları işçilerin farklı olmasından kaynaklanır. Buna karşın i ve j görevlerinin ilerleyen düğümlerde ardıllarının atanabilir oldukları zaman değişmeyeceğinden iki kısmi çözüm arasında takip eden arama bölgesi açısından hiçbir fark yoktur. Dolayısıyla belirtilen koşulları sağlayan düğümler elimine edilebilir.

Uygun eş-set Baskınlık Kontrolü

Problemde bulunan görevlerin bir alt-seti; bu alt-setteki tüm görevlerin öncülleri de aynı sette bulunuyorsa “uygun alt-set” olarak tanımlanmaktadır [81]. BMHDP-1’de her hangi bir uygun alt-set daha önce aynı ve daha az sayıda istasyona atanmış olarak

bir kısmi çözüm kapsamında oluşturulmuşsa bu alt setin ifade ettiği mevcut kısmi çözüm elimine edilebilir. Bu özelliğe uygun set baskınlık kuralı adı verilmekte olup bilindiği kadarıyla montaj hattı dengeleme problemlerinde ilk kez Nourie ve Venta [82] tarafından önerilen OptPack isimli dal-sınır algoritmasında kullanılmıştır.

Becker ve Scholl [15] inceledikleri DA-MHDP için uygun alt-set baskınlık kuralını şu şekilde tanımlamıştır; T_i kısmi çözümünün T_h kısmi çözümüne baskın olması için:

- $J(T_i)$ ve $J(T_h)$ sırasıyla T_i ve T_h 'ın içerdiği görevlerin uygun alt-setleri olmak üzere, $J(T_h) \subseteq J(T_i)$ olmalı,
- T_i kısmi çözümü en fazla T_h kısmi çözümü kadar işçiye ihtiyaç duymalı,
- T_i kısmi çözümü en fazla T_h kısmi çözümü kadar istasyona ihtiyaç duymalıdır.

Bu kural DA-MHDP'in yanı sıra performans ölçütü toplam işçi sayısının en küçüklenmesi olması koşuluyla hem Ç-MHDP hem de bu çalışma kapsamında ele alınan Pİ-MHDP1 için geçerli bir kuraldır. Çünkü tanım, DA-MHDP'i Ç-MHDP ve Pİ-MHDP1'den ayıran hiçbir özelliği içermemektedir.

Tanımından da anlaşılacağı gibi bu kural mevcut istasyonun kapatılması sırasında uygulanabilir. Kuralın uygulanabilmesi içinde istasyonların kapatılmasında bulunan kaliteli kısmi çözümlerin oluşturulacak bir yapı kapsamında hafızada saklanması gerekir. Bu yapıya yönelik etkin bir yöntem Nourie ve Venta [82] tarafından önerilmiş olup bu çalışma kapsamında da bu yöntem kullanılmıştır. Bu yöntemle oluşturulan yapıda istasyonların kapatılması esnasında eğer kaliteli ise istasyonlara atanan görev setleri ve bu setlerin ihtiyaç duyduğu istasyon ve işçi sayıları hafızada tutulmaktadır.

Uygun eş-set baskınlık kontrolü kapsamında sadece mevcut kısmi çözümde bulunan görevlerin tamamına yönelik bir kontrol uygulanmaktadır. Arama hafızası kontrol edilerek mevcut kısmi çözümde bulunan görevlerin tamamı daha önce aynı veya

daha az sayıda istasyonla, aynı zamanda da aynı veya daha az sayıda işçi ile oluşturulmuşsa mevcut kısmi çözüm elenmekte, elenemiyorsa mevcut kısmi çözüm ihtiyaç duyduğu istasyon ve işçi sayısı ile hafızaya kaydedilmektedir. Aynı yaklaşımın Becker ve Scholl [15] tarafından da aynı şekilde kullanılmıştır.

Jackson baskınlık kontrolü

Jackson [19] tarafından BMHDP-1 için geliştirilen bu kurala göre i görevinin j görevini potansiyel olarak bastırması için aşağıdaki şartların sağlanması gerekmektedir:

- $t_i \geq t_j$
- $F_j \subseteq F_i^*$

j görevini içerip i görevini içermeyen mevcut istasyon yükünden çevrim zamanını artırmadan j görevi çıkartılıp i görevi eklenebiliyorsa mevcut istasyon yükü elenir. Bu kuralın altında yatan gerçek j görevinin tüm takipçileri aynı zamanda i görevinin de takipçisi olması nedeniyle i görevi tamamlanmadan hiçbir j görevi takipçisinin operasyonuna başlanamamasıdır. Dolayısıyla i ve j görevlerinin sırası takipçilerin operasyona başlaması açısından önemli değildir. i görevinin görev zamanının j görevinden büyük olması koşulu ise i ve j görevlerinin yer değiştirmesi sonucu istasyon zamanının küçülmeyeceğini garantilemektedir [5].

Belirtilen kural Becker ve Scholl [15] tarafından DA-MHDP için modifiye edilerek en son istasyon yükü için (istasyon probleminin çözümünü müteakiben, tek bir istasyon göz önünde bulundurularak) kullanılmıştır.

Bu çalışma kapsamında göz önünde bulundurulmuş PM-MHDP1 için bu kural kısmi çözümün tamamı göz önünde bulundurulacak şekilde uygulanmıştır. Kurala göre i

görevinin j görevini potansiyel olarak demine etmesi durumunda aşağıdaki şartlar da sağlanıyorsa j görevini içeren kısmi çözüm elenmektedir:

- j görevi çıkartılıp i görevi eklendiğinde kısmi çözümün uygunluğu devam etmeli,
- yeni çözümün istasyon sayısı eski çözüme eşit veya daha küçük olmalı,
- yeni çözümün işçi sayısı eski çözüme eşit veya daha küçük olmalı.

İstasyonların kapatılması sırasında uygulanan bu kontrolde, atanabilecek görevler listesindeki her bir atanmamış i görevi kısmi çözümde bulunan her bir atanmış j göreviyle karşılaştırılır. i görevi j görevini potansiyel olarak bastırıyorsa j görevinin olmadığı i görevinin olduğu istasyon yükünün arama hafızasında kaydı olup olmadığı kontrol edilir. Kayıt varsa mevcut çözümün istasyon ve işçi yükleri arama hafızasında kayıtlı bilgilerle karşılaştırılıp yukarıda verilen özellikler sağlanıyorsa mevcut çözüm elenir.

Daha önce de belirtildiği gibi görevler başlangıç işlemleri kapsamında $EnbKumPA_i$ kriterine göre yeniden sıralanmaktadır. Jackson baskınlık ilişkisi ve $KumPA_i$ değerinin hesaplanma işlemleri göz önünde bulundurulduğunda baskın görevin sıralamada daha küçük numara alacağı görülebilir. Dolayısıyla baskın görevin bulunduğu kısmi çözüm diğer çözümden daha önce ziyaret edilecektir.

Uygun alt-set baskınlık kontrolü

Uygun eş-set baskınlık kontrolünün genellenmiş halidir. İlgili kontrol kapsamında sadece kısmi çözümlerdeki atanmış görev setlerinin birbirine eşit olduğu durum arama hafızasından kontrol edilirken, bu bölüm kapsamında mevcut kısmi çözümün daha önce oluşturulmuş daha iyi bir kısmi çözümün alt seti olup olmadığı kontrol edilmektedir. İstasyonların kapatılması sırasında uygulanan bu kontrol kapsamında yapılan işlemler sırasıyla aşağıdaki şekildedir:

- Mevcut kısmi çözümdeki işçi sayısı tespit edilerek aynı sayıdaki işçiye atanan en büyük toplam görev zamanı (*enbYük*) tespit edilir. Arama hafızası bu amaçla güncellenmiş olup *ÜS* boyutunda bir dizi yoluyla arama sürecinde her bir işçi sayısına atanmış maksimum toplam görev zamanı tutulmaktadır. Bu dizi arama hafızasının güncellenmesi sırasında güncellenmektedir.
- $Fark = enbYük - topG$ değeri bulunur.
- Atanmamış görevlerden uygunluk ilişkileri sağlanacak şekilde toplam görev zamanı *Fark*'a eşit veya daha küçük olan görev setleri birleştirilerek bu setler her seferinde mevcut kısmi çözümdeki görevlere dâhil edilip yeni bir görev seti oluşturulur. Oluşturulan yeni görev seti arama hafızasından kontrol edilerek daha önce mevcut çözümden daha iyi şartlarda kaydedilip kaydedilmediği belirlenir. Böyle bir kayıt varsa mevcut çözüm elenir.

5.2.5. Arama işleminin sezgisel olarak yönlendirilmesi

Önerilen algoritmanın şimdiye kadar anlatılan haliyle özellikle büyük boyutlu problemlere uygulanmasının çok hızlı bir şekilde uygun çözüm bulduğu, ancak ilk uygun çözümün bulunmasından sonra dal-sınır ağacının alt düğümlerinde (özellikle de alt sınır değeri yeni üst sınır değerinden bir küçük olan düğümlerde) oldukça uzun zaman kaybettiği görülmüştür. Bu nedenle de optimal bulunamayan ancak uygun çözümü, yani üst sınır değeri bulunan problemlerde bulunan üst sınır değerinin oldukça zayıf olduğu fark edilmiştir. Dolayısıyla yöntemin, üst sınır değerini arama işleminin ilk aşamalarında çok daha hızlı güncelleyecek (yani küçültecek), bu yolla da ağaçta çok daha fazla miktarda kısmi çözümü yani düğümü erken aşamalarda dolaylı olarak arayacak (yani ebeveyn düğümünü budayacak) bir mekanizmaya ihtiyacı vardır. Bunun yanı sıra bazı durumlarda ağacın belirli kesimlerinde oldukça uzun zaman arama yapılmasına karşın üst sınır güncellenememektedir. Oluşturulacak mekanizmanın bu dezavantajı da ortadan kaldıracak bir yapıda olması gerekmektedir. Bahsedilen eksikliklerin ortadan kaldırılması amacıyla önerilen algoritmaya “sezgisel güdüm” özelliği kazandırılmıştır. Sezgisel güdüm en genel anlamda arama işleminde üst sınır güncelleme işleminin yapılması veya belirli bir

süre aranmasına karşın üst sınır güncelleme işleminin yapılamaması durumunda ağacın sezgisel bir yöntemle belirlenen farklı bir noktasına hareket ederek taranmamış bölgesinin hafızaya kaydedilip, hareket edilen noktadan aramaya devam edilmesini sağlayan bir düğüm seçim ve yönlendirme mekanizmasıdır. Taranmamış bölgenin hafızaya kaydedilmesinin temel nedeni önerilen yöntemin kesin çözüm yöntemi olması nedeniyle ihtiyaç olması halinde (yani kaydedilen bölgenin dışında kalan aramada teorik alt sınır değerine eşit bir çözüm bulunamaması halinde) kaydedilen bölgeye tekrar dönülerek bu bölgenin de aranmasının sağlanmasıdır.

Daha önce anlatılan dallandırma işlemi incelendiğinde arama yapılan dal sınır ağacının, kök düğümünden mevcut düğüme kadar bir ipin üzerine dizili düğümlerin oluşturduğu yapıya benzediği görülebilir. Her bir düğümün diğer bilgilerinin yanı sıra kendine ait bir *AL* listesi, alt düğümü oluştururken kullandığı görevin *AL* listesindeki sırası, yani *yİndeks* bilgileri tutulmaktadır. Üst sınır güncellendikten sonra ağacın farklı bir noktasına atlama işlemi, sezgisel olarak belirlenmiş üst düğümlerden birisine dönülmesi işleminden ibarettir. Örneğin 20. seviyede bir düğümdeyken üst sınırın güncellendiği varsayalım ve sezgiselin uygulanması sonucunda 7. seviyedeki düğüme dönülmesi gerektiği sonucuna varılmış olsun. 7. düğüme dönülerek arama işlemine 7. düğümün *AL* listesinin bir sonraki elemanından devam edilmesi ve bu noktadan itibaren kalan tüm çözüm uzayının aranması problemin bütün çözüm uzayının aranmış olması sonucuna götürmeyecek, daha önce bulunulan 20. düğüm ile 7. düğüm arasındaki bölge aranmamış olacaktır. Dolayısıyla 20. düğümünden 7. düğüme hareket etmeden önce 7. düğümün *Seviye_{Enk}* olduğunun işaretlenerek hafızaya kaydedilmesi, gerektiği zaman kaydedilen ağaca döndüğünde 20. seviyedeki düğümünden başlanıp 7. seviyedeki *Seviye_{Enk}* düğümine kadar aranıp bu düğümde aramanın durdurulması mümkündür. Kayıt işleminden sonra arama yapılacak yeni düğüm noktasını içeren problem daha önce kaydedilmiş *Seviye_{Enk}* düğümünü içermelidir. Bu işleme hafızaya kaydedilen bilgi açısından bakıldığında çok da hantal olmayacağı görülebilir. Zira daha önce de belirtildiği gibi dal sınır ağacı düğümler içeren bir ip şeklindedir ve arama sürecinde hafıza gereksinimi açısından farklı kısmi çözümler çok farklılık arz etmemektedir.

Hafızada kayıtlı her bir dal sınır alt ağacının kendine ait bir $Seviye_{Enk}$ değeri bulunur. Çözüm sürecine başlandığı noktada $Seviye_{Enk}$ değeri kök düğüm olarak işaretlenir. Bu yolla tüm çözüm uzayının aranacağı ifade edilmiş olur. Başlangıçta tüm elemanları boş olan ve ağaç kapsamında tutulan $m_{Enb} \times k_{Enb}$ boyutundaki bir matriste (*İlkİşçiAtama*) istasyonun işçilerine ilk görev atamasının hangi düğümde yapıldığı düğüm referansları yoluyla tutulur. Bunun yanı sıra başlangıçta elemanları sıfır olan ve yine ağaç kapsamında tutulan, aynı boyuttaki matrise istasyondaki işçilere atanan toplam yük miktarları kaydedilir (*İşçiYük*). Bahsedilen dizilerin her ikisinin güncelleme işlemi her yeni düğüm oluşturmada yani yeni görev ataması yapıldığında ve mevcut düğümün budanıp geri dönüldüğü noktalarda yapılır. Bu dizilerin dışında başlangıçta m_{Enb} değerlerini içeren k_{Enb} boyutunda ve istasyondaki işçi sayılarını arama sürecinde yönlendirmeyi amaçlayan bir başka dizi de ağaç kapsamında tutulmaktadır (*İşçiSayı*). Bu dizi, atlama işleminin yapıldığı düğümlerde şu yöntemle güncellenir. İlk istasyondan başlamak üzere istasyonların ikinci işçisinden itibaren işçilerin etkinlik değerleri (yani işçiye atanan yükün çevrim zamanına oranı) önceden belirlenen bir etkinlik eşik değeriyle (ED) karşılaştırılır. Eğer işçinin etkinlik değeri ED değerinden küçükse, *İşçiSayı* dizisinin mevcut istasyona karşılık gelen hücrelerine ilgili istasyonun mevcut işçiden önce işçi sayısını azaltmaya yönlendirileceğini ifade eden mevcut işçi indeksinin bir küçüğü kaydedilir ve bir sonraki istasyona geçilir. Böyle bir ilişki bulunamıyorsa mevcut istasyon verisine m_{Enb} değeri kaydedilir. Bu ifadenin amacı ilgili istasyonun işçi sayısı azaltmaya yönlendirilmeyeceğidir. Yönlendirme işleminin ilk yapıldığı istasyondaki yönlendirme işçisine ilk görev atamasını yapan düğümün ebeveyni atlamanın yapılacağı düğümdür ($Seviye_{Enk}$). Dolayısıyla, bu düğüm işaretlenerek mevcut ağaç arama önceliği en yüksek olacak şekilde hafızaya kaydedilir. Atlama yapılacak düğümün daha önce işaretlenmiş düğümden daha büyük seviyeli bir düğüm olması gerekmektedir.

Mevcut ağaç hafızaya kaydedildikten sonra atlama düğümüne kadar her hangi bir kontrol yapılmaksızın geri dönme işlemi (backtracking) yapılır ve arama işlemine bu düğümden itibaren devam edilir. Atlama düğümü, daha önce de belirtildiği gibi etkinlik değeri ED 'den küçük olan bir işçiye atama yapacak şekilde alt düğüm oluşturmaktadır. Dolayısıyla ilk etapta ilgili işçiye atama yapılmasının engellenmesi gerekmektedir. Engelleme işlemi ise daha önce bahsedilen işçi bekletme işlemine karşılık gelmektedir. Dolayısıyla AL listesinin sonunda bulunan ve işçi bekletmeyi ifade eden $Beklet$ değerinin bir sonraki atama pozisyonuna taşınması gerekir. AL listesi $Beklet$ değerini içermiyor veya $Beklet$ değeri daha önce yapılmış olan bir atlama işlemiyle mevcut pozisyonun öncesine taşınmışsa AL listesinde herhangi bir değişiklik yapılmaz.

Verilen bir ağaçta arama işlemine $CPU_{AğaçEnb}$ süresi kadar devam edilmesine karşın üst sınır değeri güncellenememişse, mevcut ağaç arama önceliği en küçük olacak şekilde hafızaya kaydedilerek hafızadaki arama önceliği en büyük olan ağaç okunup arama işlemine bu ağaç üzerinden devam edilir. $CPU_{AğaçEnb}$ değerinin hesaplanmasında problem için izin verilen maksimum arama süresi ($CPU_{ProbEnb}$) ve problemin aranması sırasında mevcut zamana kadar geçen süre (CPU_{Prob}) olmak üzere aşağıda verilen Eş. 5.8 kullanılmıştır:

$$CPU_{AğaçEnb} = \min \{ 0, 10 \times CPU_{ProbEnb} ; CPU_{ProbEnb} - CPU_{Prob} \} \quad (5.8)$$

Bunun yanı sıra aranan ağaçta tüm düğümler birerlenmişse hafızada kayıtlı önceliği en yüksek ağaç okunarak aramaya devam edilir. Hafızada kayıtlı ağaç yoksa arama işlemine son verilir.

$İşçiSayı$ dizisi yoluyla istasyondaki işçi sayısının kontrolü şu şekilde yapılmaktadır. Eğer mevcut düğüm oluşturacağı yavru düğümü yoluyla mevcut istasyonda, bu dizide bulunan işçi sayısından sonraki bir işçiye atama yapıyorsa işçi bekletmeyi ifade eden $Beklet$ değeri mevcut düğümün AL listesinin ilk elemanı olacak, yani

```

Procedure ProblemiÇöz();
begin
  • Problemi oku, probleme sıfır görev zamanlı hayali batak görevi ekle;
  • Görevleri  $EnbKumP$  kriterine göre yeniden sırala;
  • Görevlerin en erken ve en geç başlama zamanları ile istasyon sınırlarını hesapla;
  •  $AS_{teorik} = \min\{AS_1, AS_2, AS_3\}$  olacak şekilde teorik alt sınır değerini hesapla;
  •  $\dot{U}S = k_{Enb} \times m_{Enb} + 1$  olacak şekilde başlangıç üst sınır değerini hesapla;
  •  $Sonuç_{Prob} \leftarrow \text{ÇözümBulunamadı}$ ;
  • Arama hafızasını temizle;
  • En büyük ve en küçük öncelik değerine sıfır ata,  $P_{Enb} \leftarrow 0$ ,  $P_{Enk} \leftarrow 0$ ;
  • Kök düğümü oluştur ve ağacı önceliği sıfır olacak şekilde hafızaya kaydet;
  • while( $CPU_{Prob} \leq CPU_{ProbEnb}$ )AND( $Sonuç_{Prob} \neq OptimumÇözüm$ )AND(Hafızada kayıtlı alt ağaç var)do
    • Ağaç  $\leftarrow$  Hafızadan önceliği en yüksek olan ağacı oku;
    •  $CPU_{AğaçEnb} = \min\left\{ \begin{array}{l} 0, 10 \times CPU_{ProbEnb}; \\ CPU_{ProbEnb} - CPU_{Prob} \end{array} \right\}$ 
    • Ağaç.Ara( $CPU_{AğaçEnb}$ );
    • if (Ağaç.  $Sonuç_{Ağaç} = UygunÇözüm$ )then
      begin
        •  $\dot{U}S \leftarrow$  Ağaç.  $\dot{U}S$ ;
        •  $Sonuç_{Prob} \leftarrow$  Ağaç.  $Sonuç_{Ağaç}$ ;
        •  $P_{Enb} \leftarrow P_{Enb} + 1$ ;
        •  $P_{Enk} \leftarrow P_{Enk} - 1$ ;
        •  $Seviye_{Enk}$  düğümünü belirleyerek işaretle, ağacı  $P_{Enk}$  öncelik değeriyle hafızaya kaydet;
        •  $Seviye_{Enk}$  düğümü mevcut düğüm oluncaya kadar GeriDön() prosedürünü ve diğer atlama işlemlerini uygula, ağacı  $P_{Enb}$  öncelik değeriyle hafızaya kaydet;
      end;
      else if(Ağaç.  $Sonuç_{Ağaç} = OptimumÇözüm$ )then
        begin
          •  $\dot{U}S \leftarrow$  Ağaç.  $\dot{U}S$ ;
          •  $Sonuç_{Prob} \leftarrow$  Ağaç.  $Sonuç_{Ağaç}$ ;
        end;
        else if(Ağaç.  $Sonuç_{Ağaç} = \text{ÇözümBulunamadı}$ )then
          begin
            •  $P_{Enk} \leftarrow P_{Enk} - 1$ ;
            • Ağacı  $P_{Enk}$  öncelik değeriyle hafızaya kaydet;
          end;
        • if( $Sonuç_{Prob} = \text{ÇözümBulunamadı}$ )AND(Hafızada kayıtlı alt ağaç yok)then
           $Sonuç_{Prob} \leftarrow \text{Çözümsüz}$ ;
        • else if( $Sonuç_{Prob} = UygunÇözüm$ )AND(Hafızada kayıtlı alt ağaç yok)then
           $Sonuç_{Prob} \leftarrow OptimumÇözüm$ ;
        • else if( $Sonuç_{Prob} = UygunÇözüm$ ) AND ( $\dot{U}S = AS_{teorik}$ )then
           $Sonuç_{Prob} \leftarrow OptimumÇözüm$ ;
      end;

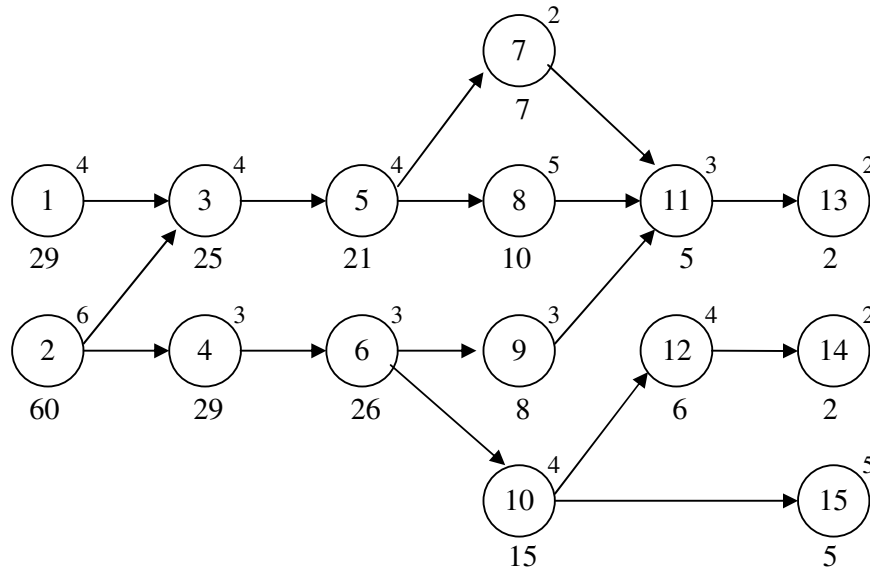
```

Şekil 5.7. Önerilen sezgisel güdümlü dal sınır algoritması

işçi oluşturulacak ilk düğümle bekletilecektir. Eğer atama işlemi dizideki işçi sayısı sınırı içerisinde yapılıyorsa *Beklet* değeri *AL* listesinin son elemanı olacak, yani işçi mevcut düğümde oluşturulacak son alt düğümle bekletilecektir. Önerilen dal-sınır algoritmasının prosedürü Şekil 5.7’de sunulmuştur. Prosedürde kullanılan *Sonuç_{Prob}* değişkeni alt-ağaç için arama sonucunu tutan *Sonuç_{Ağaç}* ile aynı değerleri alabilmekte olup problemin geneli için arama sonucunu tutmaktadır. Yapılan denemelerde iyi sonuç ürettiği görülen $ED = 0,75$ değeri algoritma kapsamında kullanılmıştır.

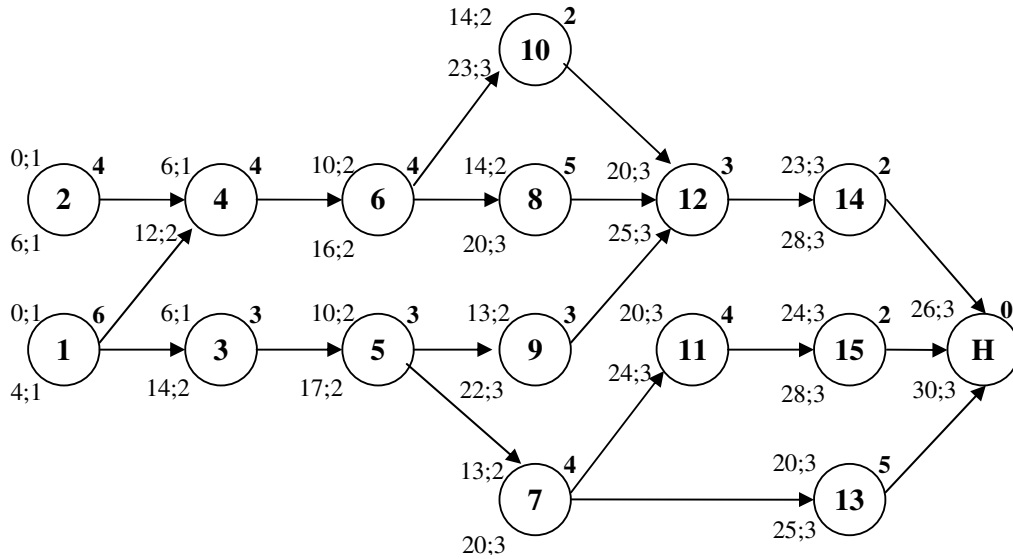
5.3. Örnek Uygulama

Önerilen dal-sınır algoritmasının daha iyi açıklanabilmesi amacıyla 15 görevli, $c = 10$, $t_{Toplam} = 54$, $k_{Enb} = 3$, $m_{Enb} = 3$ olan ve öncelik diyagramı Şekil 5.8’de verilen problem göz önünde bulundursun. Problem için başlangıç teorik alt sınır değeri $AS_{teorik} = 6$ olarak hesaplanmıştır. Öncelik diyagramında düğümlerin üzerinde yazılan rakamlar görev zamanlarını ifade ederken her bir görevin $KumPA_i$ değerleri hesaplanarak düğümlerin altında verilmiştir.



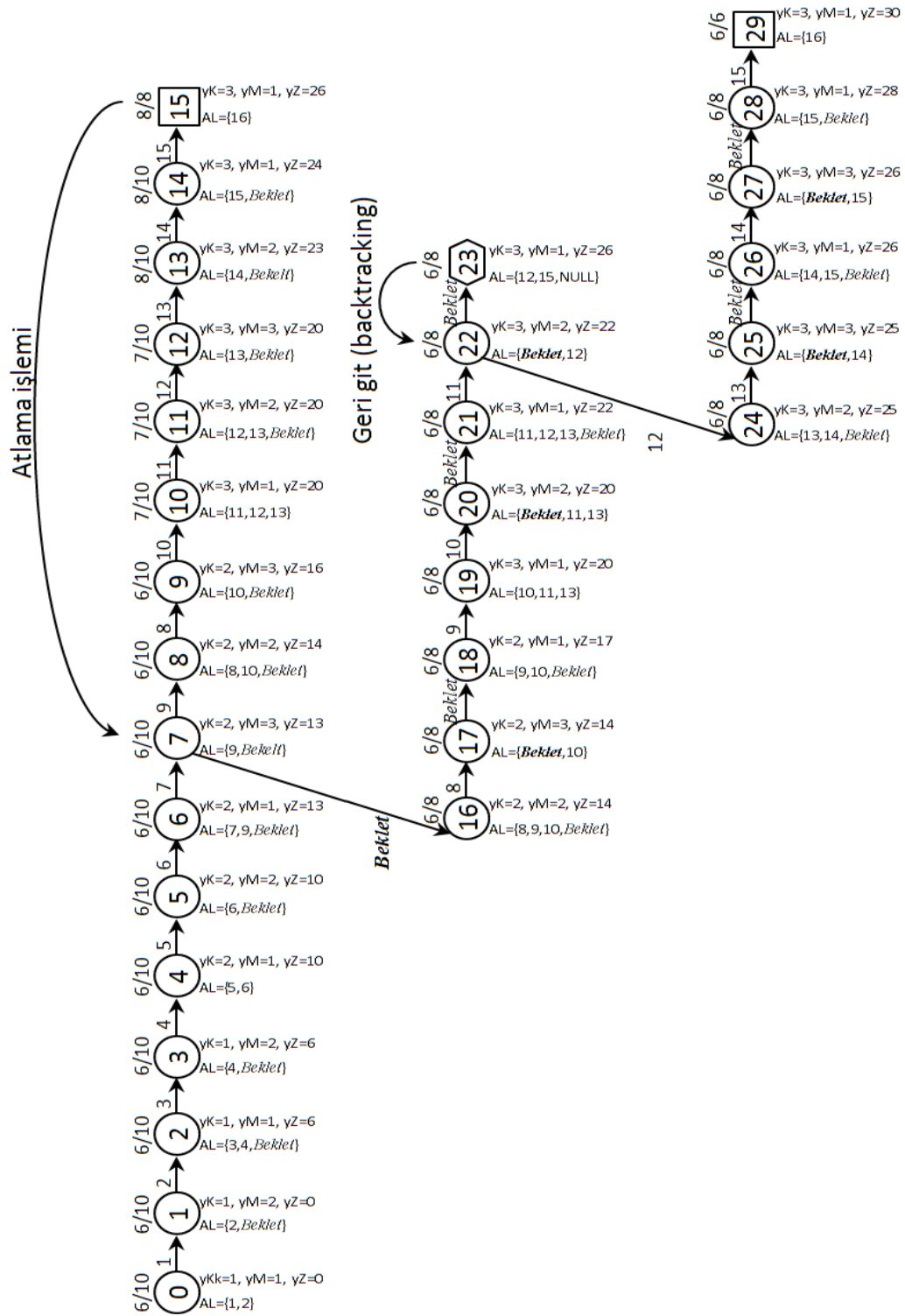
Şekil 5.8. Örnek probleme ait öncelik diyagramı ve görevlerin $KumPA_i$ değerleri

Başlangıç işlemlerinin uygulanması amacıyla öncelik diyagramına görev zamanı sıfır olan bir son düğüm eklenmiş ve görevlerin numaraları $EnbKumP_i$ kriterine göre yeniden düzenlenmiştir. Bu işlemler sonucu oluşan öncelik diyagramı Şekil 5.9'da verilmiştir. Öncelik diyagramında düğümlerin sol üst köşesinde bulunan değerler sırasıyla ilgili görevin en erken başlama zamanı ve istasyonu, sol alt köşesinde bulunan değerlerse en geç başlama zamanı ve istasyonudur. Başlangıç işlemleri kapsamında tüm istasyonların $İşçiSayı$ değerleri $m_{Enb} = 3$ olarak alınır.



Şekil 5.9. Örnek probleme ait öncelik diyagramının yeniden düzenlenmiş hali

Önerilen dal-sınır algoritmasının problemin çözümü için uygulanması sonucu elde edilen dal-sınır ağacı Şekil 5.10'da verilmiştir. Dal-sınır ağacının düğümlerine karşılık gelen kısmi çözümlerin Gant diyagramı gösterimleri ise Ek 1'de Şekil 1.1.-1.31. arasında sunulmuştur. Başlangıç düğümü 0 numaralı düğüm olan dal sınır ağacının düğümlerinin üzerinde bulunan rakamlardan ilki ilgili düğümün hesaplanan alt sınır değerini, ikincisi ise düğüm oluşturulduğu andaki global üst sınır değerini ifade etmektedir. Düğümlerin alt kısmında düğüme ilişkin AL , yK , yM ve yZ ifadeleri verilmiştir. Düğümler arası oklar ise atama işlemine karşılık gelip, ilgili atama işleminde ataması yapılan görev okların üzerine yazılmıştır. Bazı düğümlerin



Şekil 5.10. Örnek probleme ait dal-sınır ağacı (Kök düğüm numarası: 0)

atama listelerinde ve bazı okların üzerinde görülen *Beklet* ifadesi ilgili işçiye bekletme işleminin uygulanacağını/uygulandığını ifade etmektedir. Ek 1’de verilen Gant diyagramlarında işçi atama zamanının boş kalma zamanından büyük olması durumunda ($\dot{I}Z_m > \dot{I}B_m$) atama zamanı, $\dot{I}Z_m$, kesikli çizgiyle gösterilmiştir. Bunun yanı sıra yK istasyonundaki yM işçisinin yZ zamanı (yani atamanın yapılacağı istasyon, işçi ve zaman) “◀” simgesiyle işaretlenmiştir.

Ek-1 Şekil 1.1.’de kök düğüme (Düğüm 0) karşılık geldiği kısmi çözüme ilişkin Gant diyagramı verilmiştir. Bu diyagram kontrol edildiğinde tüm işçilerin atama zamanları, yükleri ve boş kalma zamanlarının sıfır olduğu görülebilir. Dolayısıyla bu düğüm için $yM = 1$, $yZ = 0$ ’dır. Ayrıca atamanın ilk istasyona yapılacağından dolayı $yK = 1$ ’dir. Kök düğümde atanan görev bulunmadığından öncelik diyagramı yardımıyla $AL = \{1,2\}$ olduğu tespit edilir. $yM = 1$ ve $yZ = 0$ olduğu için bu listeye işçi bekletme işlemine karşılık gelen *Beklet* ifadesi eklenmemiştir. Dolayısıyla bu listenin ilk sırasında bulunan 1 görevi $yM = 1$ işçisine $yZ = 0$ zamanında atanır.

Ek-1 Şekil 1.2.’de verilen Düğüm 1’in karşılık geldiği kısmi çözümün Gant diyagramı kontrol edildiğinde işçilerin atama zamanları, yükleri ve boş kalma zamanlarının sırasıyla birinci işçi için 6, 6 ve 6; ikinci ve üçüncü işçiler içinse 0, 0 ve 0 olduğu görülebilir. Dolayısıyla $yZ = 0$ ve $yM = 2$ olarak belirlenir. Bu noktada $AL = \{2, Beklet\}$ ’dir. 2 görevi tüm öncülleri $yZ = 0$ zamanından önce tamamlandığı için, *Beklet* değeri ise işçi bekletme işlemi uygulanması gerektiği için AL listesine dâhil edilmiştir. Dolayısıyla bu listenin ilk sırasında bulunan 2 görevi $yM = 2$ işçisine $yZ = 0$ zamanında atanır.

Ek-1 Şekil 1.3.’de verilen Düğüm 2’in karşılık geldiği kısmi çözüme ilişkin Gant diyagramı kontrol edildiğinde üçüncü işçinin atama zamanı sıfır olmasına karşın bu zamanda atanabilecek görev bulunmamaktadır. Dolayısıyla zaman ilerletme işlemiyle bu işçinin atama zamanı sıfır zamanından büyük olan en küçük işçi atama zamanına yani 4’e ilerletilir. Bu zamanda da atanabilecek görev yoktur. Dolayısıyla 2

ve 3. işçilerin atama zamanları 4'ten büyük olan en küçük işçi atama zamanına yani 6'ya ilerletilir. Bu işlemlerden sonra $yM=1$, $yZ=6$ ve $AL=\{3, 4, Beklet\}$ olarak belirlenir.

İşlemlere belirtilen yöntemle devam edilmesi durumunda Düğüm 15 ve karşılık geldiği kısmi çözümlerle ilişkili Gant diyagramı Ek-1 Şekil 1.16.'da atanmamış tek görev olarak hayali bitiş görevi kalacaktır. Dolayısıyla tam çözüm bulunmuş olur. Bulunan tam çözümdeki işçisi sayısı 8 olup bu düğümle global üst sınır değeri 10'dan 8'e inmiş olacaktır. Tam çözüm bulunması nedeniyle sezgisel olarak atlama işlemi uygulanır. Bulunan tam çözüme ilişkin bilgiler Çizelge 5.2'de sunulmuştur. Bu tabloda istasyonlardaki birinci işçiler hariç işçi etkinlik değeri sıfırdan büyük etkinlik eşik değeri olarak alınan $ED=0,75$ küçük veya eşit olan ilk işçi ikinci istasyondaki 3 numaralı işçidir. Daha önce de belirtildiği gibi istasyonlardaki ilk işçiler bu değerlendirmenin dışında bırakılmaktadır. Verilen dal-sınır ağacı kontrol edildiğinde bu işçiye ilk görev atamasını yapan düğümün 7 numaralı düğüm olduğu tespit edilir. İstasyon işçi sayıları ise şu şekilde hesaplanmıştır. Öncelikle tüm istasyonlardaki hesaplanan işçi sayıları m_{Emb} değerine eşit olarak alınır. İlk istasyondan başlanarak her bir istasyonun ikinci işçilerinden itibaren işçi etkinlik değeri kontrol edilir. İncelenen işçinin etkinlik değeri sıfır veya ED 'den küçükse istasyon işçi sayısı ilgili işçi indeksinin 1 eksiği olarak ayarlanıp sonraki istasyona geçilir. Birinci istasyonun ikinci işçisinin etkinliği ED 'den büyük olduğu için üçüncü işçisi kontrol edilir. Üçüncü işçinin etkinlik değeri sıfır olduğundan istasyon işçi sayısı 2 olarak kaydedilip ikinci istasyona geçilir. İkinci istasyonun ikinci işçisinin etkinlik değeri ED 'den büyük olduğundan üçüncü işçiye geçilir. Bu işçinin etkinlik değeri ED 'den küçük olduğundan istasyon işçi sayısı 2 olarak kaydedilip üçüncü istasyona geçilir. Üçüncü istasyonun ikinci işçisinin etkinlik değeri ED 'den küçük olduğundan istasyon işçi sayısı 1 olarak kaydedilerek işlem bitirilir.

Bu noktada mevcut dal-sınır ağacı önceliği en büyük olacak şekilde yeni istasyon işçi sayılarıyla birlikte arama yapılacak en küçük seviyeli düğüm 7 olacak şekilde hafızaya kaydedilir. Daha sonra 7 numaralı düğüme kadar geri dönüş işlemi

uygulanır. 7 numaralı düğümün *AL* listesinde işçi bekletmeyi ifade eden *Beklet* değeri mevcut dallandırma noktasında olduğundan değiştirilmez ve arama yapılacak en küçük seviyeli düğüm orijinal haline çevrilip (yani sıfır alınıp) yeni istasyon işçi sayılarıyla birlikte dal-sınır ağacı önceliği en büyük olacak şekilde hafızaya kaydedilir. Bu noktada hafızada kayıtlı iki adet dal-sınır ağacı vardır. Önceliği küçük olan ilk ağaç 7 numaralı düğümden 9 görevinin atanmasıyla oluşturulan, aranacak bölümü 7 numaralı düğümün altında kalan parça olan ağaçtır. İkinci ağacın ise önceliği daha büyüktür ve bu ağaçta arama işlemi 7 numaralı düğümden sıradaki atama işleminin yani işçi bekletme (*Beklet*) işleminin uygulanmasıyla başlayacak ve kök düğüme kadar bütün düğümlerde devam edilecektir.

Çizelge 5.2. Bulunan tam çözüme ilişkin istatistik bilgiler

	İstasyon 1			İstasyon 2			İstasyon 3		
	İşçi 1	İşçi 2	İşçi 3	İşçi 1	İşçi 2	İşçi 3	İşçi 1	İşçi 2	İşçi 3
Atanan görevlerin toplam süresi	9	8	0	7	9	5	6	5	5
İşçi etkinliği	0,90	0,80	0,00	0,70	0,90	0,50	0,60	0,50	0,50
Hesaplanan istasyon işçi sayısı	2			2			1		

Önceliği yüksek olan ikinci ağaç hafızadan okunarak (Bkz. Ek-1 Şekil 1.17.) sıradaki atama işleminin yani işçi bekletme işleminin uygulanmasıyla 16 numaralı düğüm oluşturulur. İşlemlere daha önce anlatılan şekilde devam edilerek Gant diyagramı Ek-1 Şekil 1.25.'de verilen kısmi çözüme karşılık gelen 23 numaralı düğüme kadar gelinir. Bu düğümden $yZ=26$ olup 12, 13, 14, 15 ve 16 numaralı görevler atanmamıştır. 12 ve 13 numaralı görevlerin en geç başlama zamanları 25 olduğundan $yZ=26$ ve daha sonrasına atanamazlar. Dolayısıyla bu düğümden aramaya devam edilmesi uygun olmayan çözümlere götürecektir. Bu nedenle 23 numaralı düğüm

budanarak 22 numaralı düğüme geri dönülür. 22 numaralı düğümden AL listesinde bulunan sonraki atanacak görev 12 görevidir. 12 numaralı görevin $yZ=22$ zamanına $yK=2$ istasyonunun $yM=2$ işçisine atanmasıyla 24 numaralı düğüm oluşturulur.

24 numaralı düğümden itibaren işlemlere belirtilen şekilde devam edilirse 29 numaralı düğümden yeni bir tam çözüme ulaşılabilecektir. Ulaşılan yeni çözümde kullanılan toplam işçi sayısı 6'dır. Güncellenen global üst sınırın yeni değeri 6 olup bu değer AS_{teorik} 'e eşit olduğu için problemin optimal çözümü bulunmuş olacaktır.

5.4. Test Problemi Sonuçları

Karşılaştırma işlemlerinde Bölüm 4 kapsamında üretilen test problemleri ve P4 3GHz işlemcili 1GB dâhili bellek bulunan kişisel bilgisayarlar kullanılmış olup karşılaştırılan algoritmalar Standart C++ programlama diliyle kodlanmıştır.

Becker ve Scholl [15] tarafından DA-MHDP'in çözümü için önerilen dal-sınır algoritması (BS), incelenen problemlerin çözümü için kullanılabilecek şekilde güncellenmiş ve önerilen algoritmayla karşılaştırılmıştır. BS'in güncellenmesi işlemi aşağıdaki şekilde yapılmıştır:

- DA-MHDP'de bulunan ve problemin uygun çözüm sayısını azaltıcı bir etkiye sahip olan montaj pozisyonları göz ardı edilmiştir. Bu nedenle de işçilerin komşu montaj pozisyonlarına atanabilmesi, görevlerin montaj pozisyonuyla ilişkili işçilere atanabilmesi gibi montaj pozisyonlarıyla ilişkili tüm kavramlar ve kısıtlar da göz ardı edilmiştir.
- Çevrim zamanını aşan görevlerin bulunabileceği, bununla bağlantılı olarak da bazı atölyelerin birden fazla istasyon boyunca uzayabileceği varsayımı göz ardı edilmiştir.
- Tüm pozitif ve negatif bölgeleme kısıtları göz ardı edilmiştir.

- BS başlangıç işlemleri kapsamında görev zamanı artırımı amacıyla EDAR adı verilen bir dal-sınır algoritması kullanılmaktadır. Bu yöntem ihtiyaç duyacağı ilave zaman nedeniyle kullanılmamış, görev zamanı artırımı göz ardı edilmiştir. EDAR'ın başlangıç işlemleri kapsamında probleme uygulanmaması nedeniyle karşılaştırmada BS açısından bir dezavantaj oluşturmayacağı açıktır. Zira EDAR'ın BS'de kullanılması önerilen algoritmada da kullanılmasını gerektirecektir, öyle ki EDAR yönteme değil probleme uygulanmaktadır. Dolayısıyla her iki algoritma kapsamında da görev zamanı artırımı uygulanmamıştır.

BS'in kodlanmasında mümkün mertebe dikkat gösterilmiştir. Karşılaştırmada kullanılan algoritmalara her bir problemin çözümü için 3600 CPU saniyesi verilmiştir. Karşılaştırma sonuçlarının sunulduğu tablolarda bulunan kısaltma ve notasyonlar aşağıdaki şekildedir:

- Çözümsüz : Problemin uygun çözüm bölgesinin boş olduğunu, yani çözümsüz olduğunu ifade etmektedir.
- Optimal çözüm : Problemin optimal çözümünün bulunduğunu ifade etmektedir.
- Uygun çözüm : Verilen süre içerisinde problemin en az bir adet uygun çözümünün bulunduğunu, ancak tüm çözüm uzayının taranamadığını ifade etmektedir.
- Çözüm bulunamadı : Verilen süre içerisinde tüm çözüm uzayının taranamadığı, taranan bölge içerisinde de probleme uygun bir çözümün bulunamadığını ifade etmektedir. Bu sonucun bulunduğu problemlerin çözümsüz olma olasılığı göz ardı edilmemelidir.
- Ortalama $Aralık_{AS}$: Yöntemin bulduğu en iyi uygun çözümdeki işçisi sayısı UB_a olmak üzere, yöntemlerin aşağıdaki formülasyonla hesaplanan $Aralık_{AS}$ değerleri ortalamasıdır. Optimum çözümü bulan yöntemler için $Aralık_{AS} = 0$ olarak alınmıştır.

$$Aralık_{AS} = 100 \times (UB_a - AS_{teorik}) / AS_{teorik} \quad (5.9)$$

Maksimum Açıklık : Problemlerin ($UB_a - AS_{teorik}$) farklarının en büyüğüdür.

Çizelge 5.3’de sunulan genel sonuçlar incelendiğinde, önerilen algoritma 1260 adet problemin 42’sinin çözümsüz olduğunu kanıtlamış, 483’ünün optimal çözümünü bulmuştur. Dolayısıyla önerilen algoritma, 525 adet problemi, yani bütün problemlerin %42’ni çözmüştür. BS ise 16 adet problemin çözümsüz olduğunu kanıtlarken optimal çözümünü bulduğu problem sayısı 292 adettir. Dolayısıyla bütün problemlerin %24’ne karşılık gelen 308 adet problemi çözmüştür. Önerilen algoritmanın çözümsüzlüğünü kanıtladığı 42 problemin 16’sının çözümsüzlüğü BS tarafından da kanıtlanırken, BS verilen süre içerisinde kalan 26 problemin bütün çözüm uzayını taramayı bitirememiştir. Benzer şekilde önerilen algoritmanın optimal çözümünü bulduğu 483 adet problemin sadece 288 adeti BS tarafından optimal olarak çözülmüş, kalan problemlerden 102 adetine uygun çözüm bulunurken 93 adetinin uygun çözümü dahi bulunamamıştır. Buna karşın BS’in optimal çözümünü bulduğu 292 adet problemde sadece 4 adetinin önerilen algoritma tarafından optimal çözümü bulunamamış, yine de problemlerin uygun çözümü bulunmuştur. Önerilen algoritmanın uygun çözüm bulduğu 665 adet problemin 4 adeti BS tarafından optimal olarak çözülmüş, 310 adetinin uygun çözümü bulunmuş, 351 adetinin uygun çözümü ise bulunamamıştır. BS’in uygun çözüm bulduğu 412 adet problemde 102 adeti önerilen algoritma tarafından optimal olarak çözülmüş, kalanlarının ise uygun çözümü bulunmuştur. Önerilen algoritma, 1260 problemde sadece 70 adetinin ne uygun çözümünü bulmuş ne de çözümsüz olduğunu gösterebilmiştir. Buna karşın BS aynı problemleri içeren 540 adete çözüm bulamamıştır. Bu analizden sonra karşılaştırma sonuçlarının problem boyutları bazında incelenmesinin uygun olacağına karar verilmiştir. Problemdeki görev sayısı problem boyutu olarak kabul edilerek tüm problemler boyutlarına göre Çizelge 5.4’de verildiği gibi 6 gruba ayrılmıştır.

Boyutu en küçük olan problem grubuna ilişkin sonuçlar Çizelge 5.5’de sunulmuştur. Bu çizelge incelendiğinde bu grupta bulunan 105 adet problemin tamamının optimal

Çizelge 5.3. Yöntemlerin çözüm durumları açısından karşılaştırılması

		BS				Toplam
		Çözümsüz	Optimal çözüm	Uygun çözüm	Çözüm bulunamadı	
Önerilen algoritma	Çözümsüz	16			26	42
	Optimal çözüm		288	102	93	483
	Uygun çözüm		4	310	351	665
	Çözüm bulunamadı				70	70
Toplam		16	292	412	540	1260

Çizelge 5.4. Problem grupları

Sıra No	Grup etiketi	Açıklama	Problem sayısı
1	$0 < n \leq 25$	Görev sayısı 1 ile 25 arasında olan problemler	105
2	$25 < n \leq 50$	Görev sayısı 26 ile 50 arasında olan problemler	213
3	$50 < n \leq 75$	Görev sayısı 51 ile 75 arasında olan problemler	292
4	$75 < n \leq 100$	Görev sayısı 76 ile 100 arasında olan problemler	260
5	$100 < n \leq 150$	Görev sayısı 101 ile 150 arasında olan problemler	260
6	$150 < n$	Görev sayısı 150'den fazla olan problemler	130
Genel Toplam			1260

çözümlerinin her iki yöntem tarafından da bulunduğu görülmektedir. Bu nedenle küçük boyutlu sayılabilecek problemlerin optimal çözümlerinin bulunmasında her iki yöntemin de yeterli olduğu söylenebilir. Diğer boyutlardaki problem çözümlerine ilişkin sonuçlar ise Çizelge 6.6-10.'da sunulmuştur. Önerilen algoritmanın (BS'in) optimal çözümü bulma oranı boyutlar bazında sırasıyla %94 (%61), %37 (%14), %30 (%12), %13 (%1) ve boyutu en büyük olan son grupta ise 130 adet problemin hiç birisinin optimal çözümü her iki yöntem tarafından da bulunamamıştır. Buna karşın önerilen algoritma tüm boyutlardaki problemlerde BS'den daha çok sayıda problemin optimal çözümünü bulmuş veya uygun çözümünü bulmuştur. Sonuç olarak çözüm kalitesi açısından önerilen algoritmanın daha iyi performans sergilediği söylenebilir. Buna karşın boyutun büyümesiyle önerilen algoritmanın da yetersiz

Çizelge 5.5. Problem boyutu $0 < n \leq 25$ olan problemlere ait sonuçlar

		BS		Toplam
		Çözümsüz	Optimal çözüm	
Önerilen algoritma	Çözümsüz	2		2
	Optimal çözüm		103	103
Toplam		2	103	105

Çizelge 5.6. Problem boyutu $25 < n \leq 50$ olan problemlere ait sonuçlar

		BS				Toplam
		Çözümsüz	Optimal çözüm	Uygun çözüm	Çözüm bulunamadı	
Önerilen algoritma	Çözümsüz	3				3
	Optimal çözüm		127	42	28	197
	Uygun çözüm				6	6
	Çözüm bulunamadı				7	7
Toplam		3	127	42	41	213

Çizelge 5.7. Problem boyutu $50 < n \leq 75$ olan problemlere ait sonuçlar

		BS				Toplam
		Çözümsüz	Optimal çözüm	Uygun çözüm	Çözüm bulunamadı	
Önerilen algoritma	Çözümsüz	7			24	31
	Optimal çözüm		32	22	23	77
	Uygun çözüm		1	137	45	183
	Çözüm bulunamadı				1	1
Toplam		7	33	159	93	292

Çizelge 5.8. Problem boyutu $75 < n \leq 100$ olan problemlere ait sonuçlar

		BS				Toplam
		Çözümsüz	Optimal çözüm	Uygun çözüm	Çözüm bulunamadı	
Önerilen algoritma	Çözümsüz	1			1	2
	Optimal çözüm		26	35	16	77
	Uygun çözüm		3	110	48	161
	Çözüm bulunamadı				20	20
Toplam		1	29	145	85	260

Çizelge 5.9. Problem boyutu $100 < n \leq 150$ olan problemlere ait sonuçlar

		BS				Toplam
		Çözümsüz	Optimal çözüm	Uygun çözüm	Çözüm bulunamadı	
Önerilen algoritma	Çözümsüz	3			1	4
	Optimal çözüm			3	26	29
	Uygun çözüm			59	154	213
	Çözüm bulunamadı				14	14
Toplam		3		62	195	260

Çizelge 5.10. Problem boyutu $150 < n$ olan problemlere ait sonuçlar

		BS		Toplam
		Uygun çözüm	Çözüm bulunamadı	
Önerilen algoritma	Uygun çözüm	4	98	102
	Çözüm bulunamadı		28	28
Toplam		4	126	130

kaldığı görülmekte olup, daha güçlü mekanizmalar içeren çözüm yöntemlerine ihtiyaç olduğu söylenebilir.

Çizelge 5.3 tekrardan incelendiğinde her iki yöntemin de verilen süre içerisinde sadece uygun çözüm bulabildiği 310 adet ortak problemin olduğu görülmektedir. Bu problemlerdeki yöntemlerin bulunduğu üst sınır değerlerinin hesaplanan AS_{teorik} 'den sapma oranları ve diğer detaylı bilgiler Çizelge 5.11'de sunulmuştur. Bu tablo incelendiğinde boyutu 150'den büyük olan 4 adet problem dışında tüm problemlerde önerilen algoritmanın AS_{teorik} 'a BS'den daha yakın veya aynı sonuç verdiği görülmektedir.

Çizelge 5.11. Her iki yöntemin de uygun çözüm bulabildiği ortak problemlere ilişkin sonuçlar

Problem Boyutu	Problem sayısı	Önerilen algoritma		BS	
		Ortalama Aralık AS	Maksimum Açıklık	Ortalama Aralık AS	Maksimum Açıklık
$50 < n \leq 75$	137	7,25	8	8,48	9
$75 < n \leq 100$	110	11,64	8	17,25	9
$100 < n \leq 150$	59	4,88	4	4,97	4
$150 < n$	4	7,45	5	6,39	3
Genel	310	8,36	8	10,89	9

Boyutu 150'den büyük olan 4 adet probleme ilişkin bilgiler ise Çizelge 5.12'de sunulmuştur. Bu çizelge incelendiğinde bu problemlerden sadece birisinde önerilen algoritmanın BS'den daha büyük üst sınır değeri bulunduğu görülebilir. Bu gruptaki problem sayısının az olması nedeniyle de BS'in performansı daha önde görülmektedir. Hâlbuki Çizelge 5.10'a bakıldığından BS'in bu grupta bulunan sadece bahsi geçen 4 adet probleme uygun çözüm bulduğu buna karşın önerilen algoritmanın bu problemleri de içeren toplam 102 probleme uygun çözüm bulduğu görülmektedir.

Yine Çizelge 5.3 tekrardan incelenirse her iki algoritmanın da çözümsüz veya optimal çözümünü bulduğu 304 adet ortak problemin olduğu görülmektedir. Bu problemler bazında algoritmaların CPU zamanları açısından karşılaştırılması Çizelge

Çizelge 5.12. Her iki yöntemin de uygun çözüm bulabildiği $n > 150$ boyutlu ortak problemlere ilişkin sonuçlar

Problem	n	c	k_{Enb}	m_{Enb}	AS_{teorik}	BS üst sınır değeri	Önerilen algoritma üst sınır değeri
Scholl	297	1699	42	2	41	43	43
Scholl	297	1659	42	2	42	45	45
Scholl	297	1483	47	2	47	50	52
Scholl	297	1659	30	2	42	45	45

5.13'de sunulmuştur. İlgili çizelgeden görülebileceği gibi önerilen algoritma ortalama olarak daha kısa sürelerde optimal çözüme ulaşmaktadır. Göz önünde bulundurulmuş problem bilindiği gibi orta ve uzun dönemli planlama periyodunda olan bir problemdir. Dolayısıyla çözüm zamanlarının makul olduğu söylenebilir. Planlama periyodu göz önüne alındığında daha büyük boyutlardaki problemlerin optimal çözümlerinin daha uzun süreler optimal çözümünün aranması önerilebilir.

Çizelge 5.13. Her iki yöntemin de optimal çözüm bulduğu ortak problemlere ilişkin CPU zamanı (saniye) sonuçları

Problem Boyutu	Problem Sayısı	Önerilen algoritma		BS	
		Ortalama CPU (sn)	En büyük CPU (sn)	Ortalama CPU (sn)	En büyük CPU (sn)
$0 < n \leq 25$	105	0,01	0,23	1,47	109,97
$25 < n \leq 50$	130	9,11	240,25	249,43	3162,53
$50 < n \leq 75$	39	262,58	2515,33	389,45	2324,25
$75 < n \leq 100$	27	68,15	529,08	327,19	1642,17
$100 < n \leq 150$	3	0,00	0,00	40,01	120,02
Genel	304	43,64	2515,33	186,59	3162,53

6. PARALEL ÇOK İŞÇİLİ MONTAJ HATTI DENGELEME PROBLEMLERİNİN ÇÖZÜMÜ İÇİN KURUCU BİR SEZGİSEL ALGORİTMA

6.1. Sezgisel Yöntemler ve Hat Dengelemede Kullanımı

Sezgisel yöntemlerin temel karakteristiği optimali garantilemeksizin çok hızlı bir şekilde kabul edilebilir kalitede çözümlere ulaşabilmeleridir. Kombinatoryal optimizasyon problemleri sınıfında bulunan özellikle orta ve büyük boyutlu bazı NP-zor problemlerin tek çözüm yöntemi sezgisel metotların kullanılmasıdır. Kombinatoryal optimizasyon problemlerinin çözümü için geliştirilen sezgisel yöntemler genel olarak kurucu ve iyileştirici sezgiseller olarak iki temel sınıfa ayrılır. Kurucu sezgiseller probleme ilişkin bazı özelliklerin kullanılmasıyla yeni bir çözüm üretirken iyileştirici sezgiseller mevcut bir veya birkaç çözümden başlayarak daha iyi bir çözüme ulaşmak için iteratif bazı işlemler uygular. Kurucu sezgiseller çok basit yapılı, uygulaması kolay ve çok hızlı çözüme ulaşabilmelerine rağmen ürettikleri çözümlerin kalitesi genellikle çok iyi değildir. Bu tip sezgiseller literatürde kombinatoryal optimizasyon problemleri için önerilen bir çok iyileştirici sezgiselin başlangıç çözüm üretici olarak kullanılmaktadır.

Montaj hattı dengeleme problemlerinin çözümüne yönelik doksanlı yılların ortalarına kadar çoğunlukla kurucu sezgisellerin geliştirilmesi üzerinde çalışılmasına rağmen, son dönemlerde araştırmacılar tabu arama, genetik algoritmalar gibi çeşitli meta sezgisel yöntemlere dayanan iyileştirici sezgisellerin geliştirilmesine odaklanmıştır [26].

Literatürde montaj hattı dengeleme problemleri için önerilen kurucu sezgiseller öncelik tabanlı kurucu sezgiseller ve kısmi birerlemeye dayalı kurucu sezgiseller olarak iki grupta incelenebilir [5]. Öncelik tabanlı kurucu sezgiseller görev zamanları ve görevler arası öncelik ilişkilerine dayanarak hesapladıkları görevlere ilişkin öncelik değerlerini probleme çözüm üretmek amacıyla kullanırlar. Görevlerin bu değerlere göre çoğu zaman büyükten küçüğe sıralanmasıyla bir öncelik listesi

oluşturulur. Bu listenin kullanılması yoluyla seçilen görevin hangi istasyona atanacağını belirlenmesinde ise istasyon tabanlı ve görev tabanlı olarak adlandırılan iki farklı strateji kullanılmaktadır. İstasyon tabanlı sezgisel yöntemler öncelik listesine göre seçilen görevin atamasını ilk istasyondan başlayarak yapar. Mevcut istasyona başkaca bir görev atanamıyorsa atama işlemine bir sonraki istasyondan devam eder. Görev tabanlı sezgisel yöntemler ise seçilen görevi öncelik ilişkileri ve diğer atama kısıtları tarafından belirlenen atanabileceği ilk istasyona atar. Bu sezgisellerden bazıları atanabilecek görev listesini her atamada güncellerken bazıları ise listedeki tüm görevler atandıktan sonra güncelleme işlemini gerçekleştirmektedir. Literatürde yapılan deneysel çalışmalar BMHDP-1 için geliştirilen istasyon tabanlı sezgisellerin görev tabanlı sezgisellerden daha kaliteli çözümler ürettiğini göstermiştir. Statik öncelik kuralı tabanlı sezgiseller öncelik listesini algoritmanın başlangıcında belirlerken, dinamik öncelik kurallarını kullanan sezgisellerde öncelikler mevcut kısmi çözüm göz önünde bulundurularak her atama yapıldıktan sonra tekrardan hesaplanmaktadır [26].

Kurucu sezgiseller ürettikleri çözüm sayıları açısından da iki gruba ayrılabilir. Tek geçişli olarak adlandırılan sezgiseller tek bir çözüm üretirken çok geçiş sezgiselleri her seferinde farklı öncelik kuralları veya stokastik tek bir öncelik kuralı kullanarak birden fazla çözüm üretir ve bunlar içerisinde en iyisini seçer [5]. Sezgisellerden bir kısmı çözümü baştan sonra oluştururken, diğer bir kısmı tüm ardılları atanmış görevleri göz önünde bulundurarak sondan başa doğru, bir kısmı da tüm öncülleri veya tüm ardılları atanmış görevlerin tamamını göz önünde bulundurarak çözüm üretme işlemini her iki yönde de eş zamanlı olarak gerçekleştirir.

BMHDP-1 için literatürde önerilmiş birçok öncelik listesi oluşturma yöntemi bulunmaktadır. Bu yöntemlerden en sık kullanılanları Çizelge 6.1'de sunulmuştur. Problemdeki E_i görevinin atanabileceği en erken istasyon (E_i) ve en geç istasyon (L_i) değerlerinin tespit edilmesinde Becker ve Scholl [15] tarafından önerilen yöntem kullanılmıştır. Öncelik kurallarına ilişkin detaylı bilgi için Hackman ve ark. [83],

Scholl [5] ve Scholl ve Becker [26] tarafından yapılan literatür taramalarına bakılabilir. Scholl ve VoB [84], BMHDP-1’de en iyi performansı EnbF^a ve EnbPA kurallarının sergilediğini belirtmiştir.

Çizelge 6.1. BMHDP-1 için literatürde önerilen öncelik kurallarından bazıları

Kural kodu	Açıklama
EnbZaman	Azalan görev zamanı t_i
EnbPA	Azalan pozisyon ağırlığı $PA_i = t_i + \sum_{h \in F_i^*} t_h$
EnbF ^a	Azalan takipçi sayısı $ F_i^* $
EnbF	Azalan hemen takipçi sayısı $ F_i $
EnkE	Artan en erken istasyon E_i
EnkL	Artan en geç istasyon L_i
EnkMI	Artan mümkün istasyon sayısı $MI_i = L_i - E_i + \varepsilon$
EnbOrtPA	Azalan ortalama pozisyon ağırlığı $PA_i / (F_i^* + 1)$
EnkOrtL	Artan ortalama en geç istasyon $L_i / (F_i^* + 1)$
EnbZaman/L	Azalan görev zamanının en geç istasyona oranı t_i / L_i
EnbF/MI	Azalan toplam takipçi sayısının boş istasyona oranı $ F_i / MI_i$
EnbZaman/MI	Azalan görev zamanının boş istasyona oranı t_i / MI_i
EnbTO	Azalan toplam takip eden öncelik sayısı r_i
EnbKumTO	Azalan birikimli takip eden öncelik sayısı r_i^*
EnbKumPA	Azalan birikimli pozisyon ağırlığı $KumPA_i = t_i + \sum_{h \in F_i} KumPA_h$
EnkNo	Artan görev numarası i
EnkZaman	Artan görev zamanı t_i
EnbE	Azalan en erken istasyon E_i
EnbL	Azalan en geç istasyon L_i

Verilen bir görev setindeki tüm görevlerin öncülleri ya atanmış veya aynı set tarafından içeriliyorsa BMHDP-1 için bu setin tek bir istasyona atanmasında tek bir kısıt söz konusu olup, bu kısıt ilgili setteki görevlerin toplam zamanının çevrim zamanından büyük olmamasıdır. Buna karşın Pİ-MHDP1’de, eğer istasyonda m adet işçi varsa, setin görev zamanının $m \times c$ ’den büyük olmaması ve öncelik ilişkilerinin atamaya engel teşkil etmemesi verilen setin ilgili istasyona uygun bir şekilde atanması için gerekli ama yeterli değildir. Zira proje süresinin (makespan) en

küçüklendiği KKPPP, NP-zor bir problemdir [85]. Söz konusu atama problem ise proje süresinin çevrim zamanına eşit veya küçük olduğu bir proje planının olup olmadığını sorgulayan KKPPP'e eşdeğerdir. NP-zor bir optimizasyon problemine amaç fonksiyonu kısıtı konularak elde edilen karar problemi NP-tam bir problemdir [86]. Dolayısıyla ilgili problem NP-zor yapıda olduğu için karar problemi yapısında olan söz konusu atama problemi NP-tam olacaktır. Sonuç olarak, bu problemin kesin çözümünün bulunabilmesi için birerleme algoritmalarının kullanılması gerekir. Pİ-MHDP1'e bu açıdan bakıldığı zaman üç alt problem içerdiği ortaya çıkmaktadır:

1. İstasyonlardaki işçi sayılarının (hat konfigürasyonu) belirlenmesi (konfigürasyon problemi),
2. Görevlerin hangi istasyonda yapılacağı belirlenmesi (paylaştırma problemi) ve
3. Görevlerin verilen istasyonda hangi işçiye ve zamana atanacağı belirlenmesi (atama problemi).

Hat konfigürasyonunun bilindiği varsayımıyla problemin sadece paylaştırma problemiyle sınırlı tutulması durumunda bile, yukarıda da bahsedildiği gibi, problem BMHDP-1 kadar kolay olmamakta, istasyona paylaştırılan görevlerin işçilere uygun bir şekilde atamasının yapılması gerekmektedir.

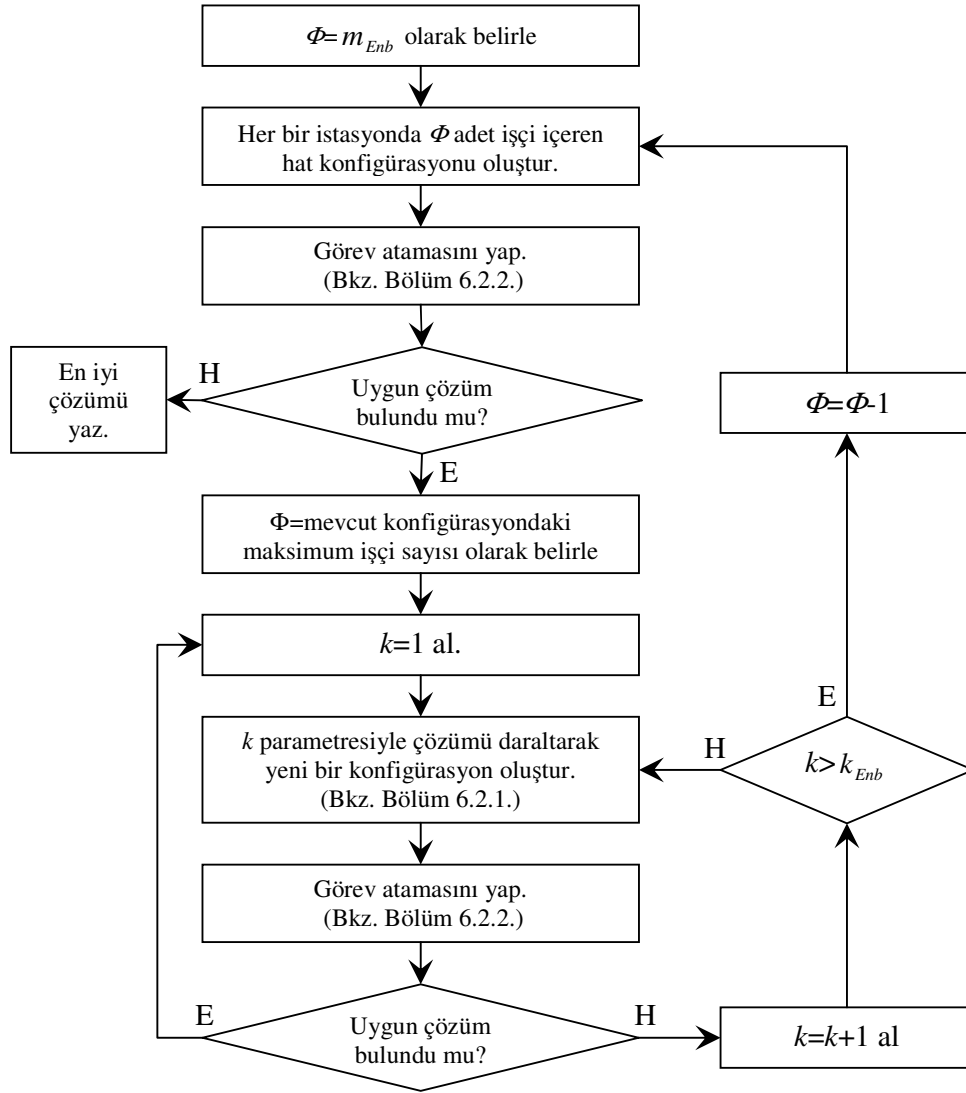
BMHDP-1'de her bir istasyon tek bir işçiye karşılık gelmekte olup istasyonlar dolayısıyla işçiler hat boyunca seri olarak yerleşmiştir. Dolayısıyla BMHDP-1 konfigürasyon problemini içermez. Pİ-MHDP1'de ise istasyonlar hat boyunca yine seri olarak sıralanmasına rağmen işçiler istasyon içerisinde paralel olarak görev yapmaktadır. Bu sistemlerde en fazla $k_{Enb} \times m_{Enb}$ adet işçi bulunabilir. Bu sayı problem için işçi sayısı üst sınır değeri olup bu durumda her bir istasyonda m_{enb} adet işçi bulunacaktır. Problemin işçi sayısı alt sınır değerlerinden birisi ise, daha önce de belirtildiği gibi, klasik BMHDP-1'de kullanılan ve öncelik ilişkileri ve görevlerin bölünemezliği kısıtlarının gevşetilmesi yoluyla hesaplanan AS_1 değeridir.

Esasında m_{Enb} değerini, başka her hangi bir kısıtlama yok ise, bir birine paralel yapılabilecek maksimum görev sayısı belirleyecektir. Bu sayı ise öncelik diyagramının esnekliğiyle yakından ilişkili olup esnekliğin artmasıyla genellikle büyüyecektir.

Sonuç olarak, klasik BMHDP-1’de öncelik değerlerine göre seçilen görev mümkün ilk istasyona atanırken, göz önünde bulundurulmuş Pİ-MHDP1’de mevcut istasyonda kaç işçinin bulunduğu ve atamanın bu işçilerden hangisine ve hangi zamanda yapılması gerektiğine karar verilmelidir. Dolayısıyla öncelik kuralı tabanlı sezgisellerin klasik BMHDP-1’e uygulandığı gibi incelenen probleme de uygulanma imkânı yoktur. Öyle ki, istasyonlardaki işçi sayılarının tespit edilmesinde (konfigürasyon problemi) ve görevin seçilen istasyonda hangi işçiye ve zamana atanacağını belirlenmesinde (atama problemi) kullanılmak üzere ilave mekanizmalara ihtiyaç vardır.

6.2. Önerilen Sezgisel Yöntem

Sezgiselin ana akış şeması Şekil 6.1’de sunulmuştur. Bu şemadan görüldüğü gibi sezgisel, her bir istasyonda maksimum sayıda işçi olduğu varsayımıyla ($\Phi = m_{Enb}$) oluşturulan hat konfigürasyonu ile işleme başlamaktadır. Bu konfigürasyona atama yapıldığında uygun bir çözüm bulunamıyorsa sezgiselin bu problemi çözebilecek kabiliyette olmadığı sonucuna varılmaktadır. Yapılan atama uygun bir çözüm üretmişse üretilen bu çözümün içerdiği konfigürasyondaki maksimum işçi sayısı saklanıp, konfigürasyondaki istasyonlardan birisi daha sonra anlatılacak yöntemle seçilerek bu istasyondaki işçi sayısı bir azaltılmakta (konfigürasyon daraltma işlemi) ve bulunan yeni konfigürasyona tekrardan atama yapılması yoluyla yeni bir uygun çözüm aranmaktadır. Konfigürasyon daraltma yöntemi tüm istasyonlara uygulanmasına rağmen uygun çözüme ulaşılamamışsa her bir istasyonda daha önce saklanan maksimum işçi sayısından bir eksik sayıda işçi olacak şekilde ($\Phi = \Phi - 1$) yeni bir konfigürasyon oluşturulup bu konfigürasyon baz alınarak işleme devam edilmektedir.



Şekil 6.1. Önerilen sezgiselin ana akış şeması

6.2.1. Çözümün daraltılması

Verilen bir k parametresiyle çözüm daraltma işlemi istasyon seçim listesinden k . sıradaki istasyonun seçilerek işçi sayısının bir azaltılması şeklinde yapılmaktadır. İstasyon seçim listesinin oluşturulmasında çeşitli stratejiler söz konusu olabilmekle birlikte çalışma kapsamında aşağıda açıklanan 4 farklı strateji incelenmiştir.

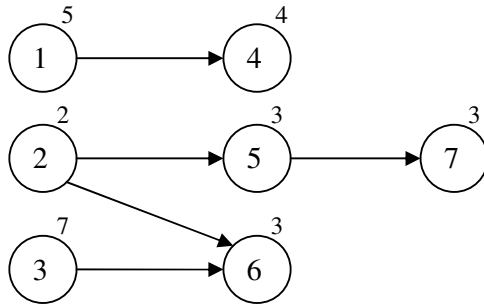
1. *Boş zamana göre seçim:* Bu stratejide, istasyonun toplam boş zamanı istasyondaki işçi sayısına bölünerek her bir istasyon için ortalama işçi boş zamanı hesaplanmakta, istasyonların bu zamanlara göre büyükten küçüğe sıralanması yoluyla istasyon seçim listesi oluşturulmaktadır. Bu stratejiyle ortalama boş zamanı en fazla olan istasyondan bir işçi çıkartılarak toplam yükün kalan işçilere dağıtılması yoluyla işçi sayısının azaltılması hedeflenmiştir.
2. *İşçi yüküne göre seçim:* Bu stratejide ise istasyonlar içerdikleri en az işçi yüküne göre küçükten büyüğe sıralanarak istasyon seçim listesi oluşturulmaktadır. Bu stratejiyle iş yükü en az olan işçi istasyondan çıkartılarak iş yükünün istasyondaki diğer işçilere dağıtılması yoluyla işçi sayısının azaltılabileceği düşünülmüştür.
3. *Soldan sağa seçim:* İstasyon seçim listesi ilk istasyon birinci sıraya, ikinci istasyon ikinci sıraya şeklinde devam edilerek oluşturulmaktadır. Bu stratejinin iraksayan yapıda öncelik ilişkisi bulunan problemlerin çözümünde kolaylık sağlayabileceği düşünülmüştür.
4. *Sağdan sola seçim:* Son istasyon birinci sırada olacak şekilde seçim listesi sondan başa oluşturulmaktadır. Bu stratejinin ise yakınsayan yapıda öncelik ilişkisi bulunan problemlerin çözümünde kolaylık sağlayabileceği düşünülmüştür.

Belirtilen stratejilerin performansları bölümün ilerleyen kısımlarında karşılaştırılmıştır.

6.2.2. Görev atamasının yapılması

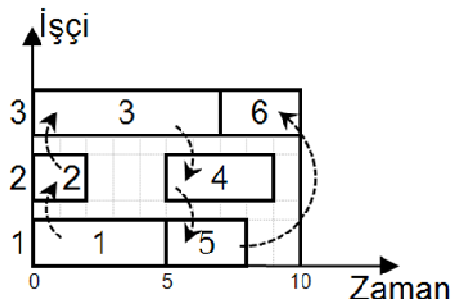
Seçilen bir görevin belirli bir istasyona atanması problemi daha önce de belirtildiği gibi KKPPP'e benzerlik göstermektedir. Önerilen sezgisel kapsamında görev ataması Hu [74] tarafından KKPPP'in çözümü için önerilen sezgisele benzer bir yöntemle

yapılmaktadır. Hu [74] sezgiseline göre atanabilecek görevler içerisinde son istasyona kritik yol uzunluğu en büyük olan görev (önerilen sezgiselde oluşturulan seçim listesinde en önde bulunan görev) en az kullanılan kaynağın mümkün en erken zamanına atanmaktadır. Bu sezgiselin performansının düşük olduğu bir durumun anlatılabilmesi amacıyla Şekil 6.2'deki öncelik diyagramıyla verilen görevlerin, $c=10$ olan, öncelik listesi 1-2-3-4-5-6-7 şeklinde belirlenmiş ve $m=3$ işçinin bulunduğu boş bir istasyona atanacağını varsayalım.



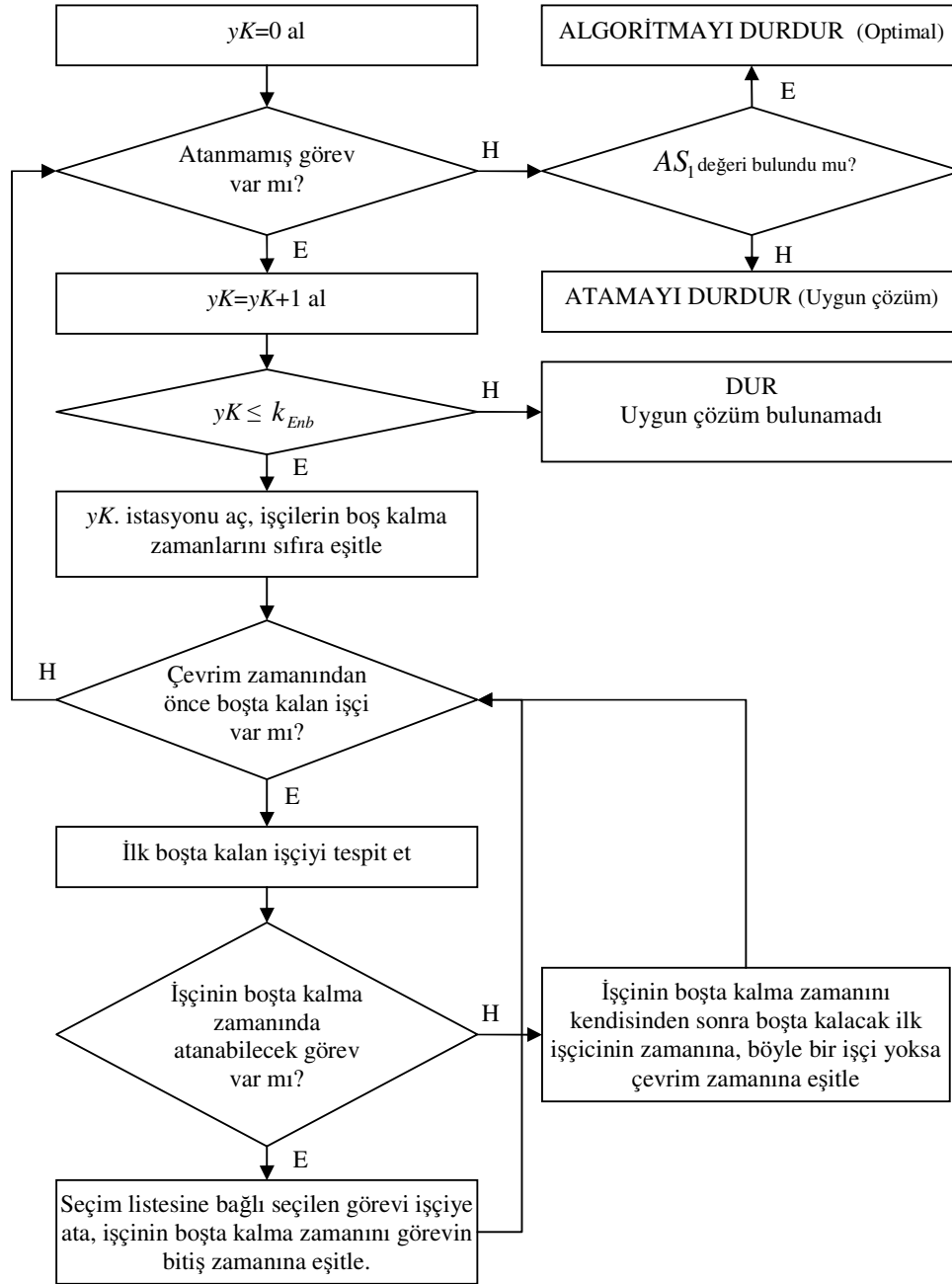
Şekil 6.2. Örnek öncelik diyagramı

Hu [74] sezgiselinin kullanılmasıyla yapılan atamanın Gant diyagramı Şekil 6.3'de verilmiştir. Bu diyagrama dikkat edilecek olursa 7 numaralı görev ilgili istasyona atanmamıştır. Bunun temel nedeni sezgiselin zaman bazlı değil, görev bazlı çözüm kurmasından kaynaklanmaktadır. Bu noktadan hareketle önerilen atama yöntemi zaman bazlı atama yapacak şekilde tasarlanmıştır.



Şekil 6.3. Hu [74] sezgiseliyle yapılan atamanın Gant diyagramı

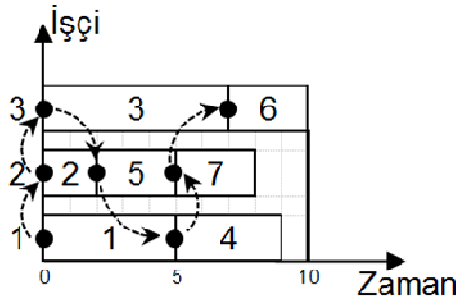
Akış diyagramını Şekil 6.4’de sunulan önerilen atama yönteminde her bir işçinin boşta kalma zamanını tutan bir listeden faydalanılmaktadır. Küçükten büyüğe sıralı bu liste başlangıçta her bir işçi için birer adet sıfır değerini içermekte olup atama işlemi bu



Şekil 6.4. Önerilen atama algoritmasının akış diyagramı

listenin ilk elemanının ifade ettiği işçiye ilgili zaman noktasına yapılmaktadır. İlgili atama zamanı ve işçi numarası listeden alındıktan sonra (aynı zamanda bu kayıt listeden silinir) atama zamanda tüm öncülleri atanmış ve tamamlanmış, ancak kendisi atanmamış görevler içerisinde seçim kuralına göre bir görev belirlenerek ilgili işçi ve zamana atanır. Belirtilen şartları sağlayan hiçbir görev yoksa belirlenen işçiye belirlenen zamanda görev ataması yapılmaz. Bu noktada görev ataması yapılmışsa atanan görevin tamamlanma zamanı, atama yapılmamışsa listede bulunan ve ilgili zamandan farklı en küçük zaman işçiyle bağlantılı olacak şekilde listeye tekrar eklenmektedir.

Önerilen atama yöntemiyle görev ataması yapılması durumunda oluşan Gant diyagramı Şekil 6.5’de sunulmuştur. Şekilden de görülebileceği gibi 7 numaralı görevin de istasyona atanmasıyla istasyon verimliliği artmıştır.



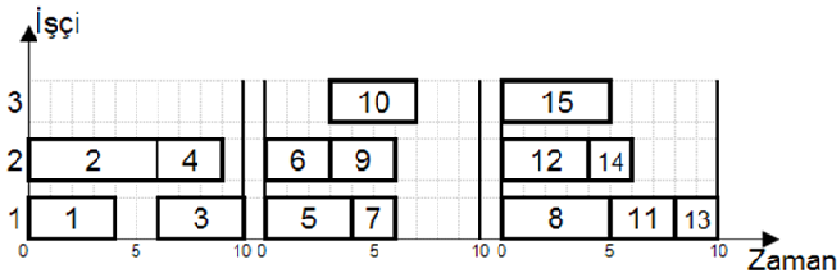
Şekil 6.5. Önerilen algoritma ile yapılan atamanın Gant diyagramı

Önerilen atama algoritmasının aynı anda yapılamayacak görevler (TI listesinin elemanları) bulunduran, yani negatif zaman bölgeleme kısıtları içeren probleme uygulanması durumunda algorithmada şu şekilde bir değişikliğin yapılması gerekir. İşçinin boşa kalma zamanında atanabilecek görevlerin listesi oluşturulduktan sonra bu listede bulunan ve negatif zaman bölgeleme kısıtlarına bağlı olarak aynı anda yapılamayacak görevi halen devam eden görevler listeden çıkartılır. Bu yolla negatif zaman bölgeleme kısıtları da sağlanmış olur. Aynı yaklaşım diğer tüm bölgeleme kısıtları için de kullanılabilir.

6.3. Örnek Uygulama

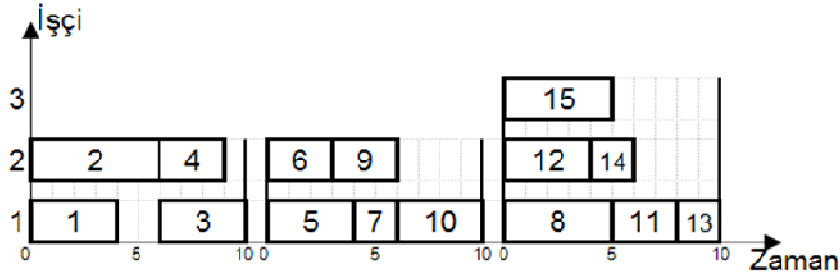
Önerilen sezgisel yöntemin daha iyi açıklanabilmesi amacıyla 15 görevli, $c=10$, $k_{Enb} = 3$, $m_{Enb} = 3$ olan ve öncelik diyagramı Şekil 2.1'de verilen problem göz önünde bulundursun. Problemin işçi sayısı açısından alt sınır değeri $AS_1 = [t_{Toplam} / c]^+ = [54 / 10]^+ = 6$ olacaktır. EnkNo kodlu artan görev numarası kuralı uygulanacak olursa seçim listesi 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 şeklinde olacaktır. Çözüm daraltma işleminde boş zamana göre seçim stratejisi kullanılacaktır.

$\Phi = m_{Enb} = 3$ olarak alınırsa hattın başlangıç konfigürasyonunda istasyonlarda bulunan işçi sayıları (3, 3, 3) şeklinde karşımıza çıkacaktır. Bu konfigürasyona, verilen atama yöntemiyle atama yapılırsa elde edilen çözümün Gant diyagramı Şekil 6.6'da gösterildiği gibi olacaktır.



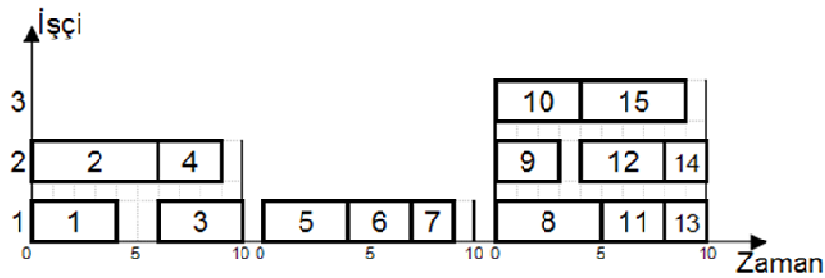
Şekil 6.6. Başlangıç uygun çözümünün Gant diyagramı gösterimi

Bu çözümde istasyonlardaki maksimum işçi sayısı 3 olduğu için $\Phi=3$ olarak saklanır ve $k=1$ alınır. Verilen çözümde istasyonlardaki işçilerin ortalama boş zamanları sırasıyla 1,50; 4,67 ve 3,00 olarak ortaya çıkmıştır. $k=1$ parametresiyle boş zamanı en büyük olan ikinci istasyondaki işçi sayısı 1 azaltılarak (çözüm daraltma işlemi) yeni konfigürasyon (2, 2, 3) olacak şekilde belirlenir. Bu konfigürasyona yapılan atama sonucu ulaşılan çözümün Gant diyagramı gösterimi Şekil 6.7'de sunulmuştur.



Şekil 6.7. Birinci çözüm daraltma işlemi sonucu ulaşılan çözümün Gant diyagramı gösterimi

Çözüm daraltma işlemi sonucunda uygun bir atama bulunduğu için tekrardan $k=1$ olarak alınır. Verilen çözümde istasyonlardaki işçilerin ortalama boş zamanları sırasıyla 1,50; 2,00 ve 3,00 olarak ortaya çıkmıştır. $k=1$ parametresiyle boş zamanı en büyük olan üçüncü istasyondaki işçi sayısı 1 azaltılarak (çözüm daraltma işlemi) yeni konfigürasyon (2, 2, 2) olacak şekilde belirlenir. Bu konfigürasyona atama yapılırsa uygun bir çözümün bulunamayacağı görülebilir. Dolayısıyla $k=2$ alınarak boş zamanı ikinci en büyük olan ikinci istasyondaki işçi sayısı 1 azaltılır (çözüm daraltma işlemi) ve yeni konfigürasyon (2, 1, 3) olacak şekilde belirlenir. Bu konfigürasyona yapılan atama sonucu ulaşılan çözümün Gant diyagramı gösterimi Şekil 6.8’de sunulmuştur.



Şekil 6.8. Üçüncü çözüm daraltma işlemi sonucu ulaşılan uygun çözümün Gant diyagramı gösterimi

Bulunan çözüm alt sınır değeriyle aynı işçi sayısına sahip olduğundan atama algoritmasına göre süreç sona erdirilir. Kullanılan sezgisel, problemin optimal çözümünü bulmuştur.

6.4. Test Problemi Sonuçları

Karşılaştırma işlemlerinde Bölüm 4 kapsamında üretilen test problemleri, P4 3GHz işlemcili 1GB dâhili bellek bulunan kişisel bilgisayarlar kullanılmış olup, sezgiseller Standart C++ programlama diliyle kodlanmış ve derlenmiştir.

Çizelge 6.1’de sunulan 19 farklı öncelik kuralı ve Bölüm 6.1’de ifade edilen 4 farklı çözüm daraltma stratejisi ikili şekilde kullanılarak önerilen sezgiselin 76 farklı versiyonuyla karşılaştırma problemleri çözülmüştür. Sezgisellerden hiç birisi karşılaştırma problemlerinden 269 tanesinin uygun çözümünü bulamamıştır. Bu problemlerin çözümsüz olma durumu da söz konusu olabilir. Bu problemler değerlendirme dışı bırakılmıştır. Kalan problemler ise en az bir sezgisel tarafından uygun çözümü bulunan problemler olup her bir sezgiselin ilgili problem için performans değeri ($Aralık_{Min}$) Eş. 6.1 kullanılarak hesaplanmıştır. Bu eşitlikteki Çözüm ifadesi ele alınan sezgiselin ilgili problem için bulunduğu toplam işçi sayısını, Min ifadesi ise ilgili problem için tüm sezgisellerin bulunduğu toplam işçi sayılarının en küçüğünü temsil etmektedir. Her hangi bir probleme uygun çözüm bulamayan sezgiselin $Aralık_{Min}$ değeri ilgili problem için 100 olarak alınmıştır.

$$Aralık_{Min} = 100 \times (\text{Çözüm} - Min) / \text{Çözüm} \quad (6.1)$$

Deneme sonuçlarının öncelik kuralları açısından değerlendirilmesine ilişkin bilgiler Çizelge 6.2’de sunulmuştur. $Aralık_{Min}$ değerleri incelendiğinde en iyi performans sergileyen ilk 4 kural EnbKumPA, EnbF/MI, EnbPA ve EnbZaman/MI olarak ortaya çıkmıştır.

Çizelge 6.2. Öncelik kurallarının karşılaştırılması

Öncelik Kuralı	<i>Ortalama Aralık_{Min}</i> *	<i>Ortalama Aralık_{Min}</i> **	<i>En küçük Aralık_{Min}</i> **	<i>En büyük Aralık_{Min}</i> **	Çözümsüz Deneme Sayısı	% Çözümsüz	Ortalama CPU (sn)
EnbTO	2,99	2,91	0,00	25,00	4	0,09	0,05
EnbOrtPA	4,03	3,86	0,00	30,00	8	0,18	0,05
EnbKumTO	3,03	2,95	0,00	25,00	4	0,09	0,05
EnbKumPA	2,04	2,04	0,00	33,33	0	0,00	0,05
EnbE	5,35	5,35	0,00	40,00	0	0,00	0,05
EnbFa	3,53	2,92	0,00	25,00	28	0,63	0,05
EnbF/MI	2,72	2,55	0,00	25,00	8	0,18	0,05
EnbF	6,27	5,85	0,00	40,00	20	0,45	0,05
EnbL	9,17	9,17	0,00	43,75	0	0,00	0,04
EnbPA	2,83	1,60	0,00	33,33	56	1,25	0,05
EnbZaman	16,61	4,05	0,00	38,46	584	13,08	0,05
EnbZaman/L	3,31	3,14	0,00	25,00	8	0,18	0,06
EnbZaman/MI	2,62	2,53	0,00	25,00	4	0,09	0,06
EnkOrtL	2,84	2,67	0,00	25,00	8	0,18	0,05
EnkE	8,37	7,95	0,00	39,13	20	0,45	0,04
EnkL	3,57	3,22	0,00	33,33	16	0,36	0,05
EnkMI	3,46	3,20	0,00	33,33	12	0,27	0,05
EnkNo	5,63	5,63	0,00	36,36	0	0,00	0,05
EnkZaman	10,00	10,00	0,00	40,00	0	0,00	0,04

* Çözümsüz denemeler dâhil ($Aralık_{Min}$ değeri 100 olacak şekilde)

** Çözümsüz denemeler dâhil değil

Deneme sonuçlarının çözüm daraltma stratejileri açısından değerlendirilmesine ilişkin bilgiler Çizelge 6.3'de sunulmuştur. Çözüm daraltma stratejilerinin çözüm kalitesi üzerindeki etkinliği öncelik kurallarına kıyasla daha küçük olmasına karşın en iyi performansa soldan sağa seçim stratejisinde ulaşıldığı görülmektedir. Problemin uygun çözümünün bulunamaması Şekil 6.1'den de görülebileceği gibi çözüm daraltma işleminden bağımsızdır. Öyle ki, her bir istasyonda maksimum sayıda işçi olduğu varsayımıyla sezgisel çalıştırıldığında uygun bir çözüm bulunamamışsa problemin çözümünün bulunamadığı sonucuna varılmakta, çözüm daraltma işlemi uygulanmamaktadır. Bu durum Çizelge 6.3'den de görülebileceği gibi uygun çözümü bulunamayan problem sayısının tüm çözüm daraltma stratejilerinde bir birine eşit olması sonucunu doğurmaktadır.

Çizelge 6.3. Çözüm daraltma stratejilerinin karşılaştırılması

Çözüm Daraltma Stratejisi	Ortalama Aralık _{Min} *	Ortalama Aralık _{Min} **	En küçük Aralık _{Min} **	En büyük Aralık _{Min} **	Çözümsüz Deneme Sayısı	% Çözümsüz	Ortalama CPU (sn)
Soldan sağa seçim	5,33	4,45	0,00	43,75	195	0,92	0,07
Sağdan sola seçim	5,06	4,18	0,00	43,75	195	0,92	0,06
Boş zamana göre seçim	5,17	4,29	0,00	43,75	195	0,92	0,04
İşçi yüküne göre seçim	5,16	4,28	0,00	43,75	195	0,92	0,03

* Çözümsüz denemeler dâhil ($Aralık_{Min}$ değeri 100 olacak şekilde)

** Çözümsüz denemeler dâhil değil

Çizelge 6.4’de en iyi 4 öncelik kuralının çözüm daraltma stratejileri bazında performans değerlendirmesi verilmiştir. İlgili tablo incelendiğinde çözümsüz denemeler dâhil edilmediğinde EnbPA kuralının sağdan sola seçim stratejisiyle iyi performans sergilediği, çözümsüz denemeler de dâhil edildiğinde ise EnbKumPA kuralının yine sağdan sola seçim stratejisiyle iyi performans sergilediği görülmektedir. Bu sonuçlardan sonra sadece “EnbPA/sağdan sola seçim” algoritmasının çözüm bulduğu problemler için “EnbKumPA/sağdan sola seçim” algoritmasının performans değerinin karşılaştırılmasına karar verilmiş ve bu değer 1,80 olduğu görülmüştür. Yapılan değerlendirme çalışmaları ışığında problem çözümünde öncelikle “EnbPA/sağdan sola seçim” sezgiselinin kullanılması, çözüm bulunamaması halinde de “EnbKumPA/sağdan sola seçim” sezgiselinin kullanılmasının uygun olacağı sonucuna varılmıştır.

Bölüm 4 kapsamında matematiksel model çalışması sonucu optimal çözümü bulunan 101 adet problem için önerilen sezgiselin karşılaştırma sonuçları Çizelge 6.5’de sunulmuştur. Çizelgeden de görülebileceği gibi önerilen sezgiselin küçük boyutlu problemlerin bilinen optimal çözümlerinden sapması kabul edilebilir düzeyde olmaktadır. Mansoor grubundaki problemlerde ortalama $Aralık_{Min}$ değeri büyük görünmesine rağmen algoritmanın bulduğu çözümlerin optimal çözümlerden olan en büyük işçi farkı sadece 1 adettir.

Çizelge 6.4. En iyi 4 öncelik kuralının stratejiler bazında karşılaştırılması

Sezgisel Bileşenleri	<i>Ortalama Aralık_{Min}*</i>	<i>Ortalama Aralık_{Min}**</i>	<i>En küçük Aralık_{Min}**</i>	<i>En büyük Aralık_{Min}**</i>	Çözümsüz Deneme Sayısı	% Çözümsüz	Ortalama CPU
EnbKumPA							
Soldan sağa seçim	2,29	2,29	0,00	33,33	0	0,00	0,07
Sağdan sola seçim	1,78	1,78	0,00	33,33	0	0,00	0,07
Boş zamana göre seçim	2,04	2,04	0,00	33,33	0	0,00	0,04
İşçi yüküne göre seçim	2,04	2,04	0,00	33,33	0	0,00	0,04
EnbF/MI							
Soldan sağa seçim	2,94	2,76	0,00	25,00	2	0,18	0,07
Sağdan sola seçim	2,51	2,34	0,00	25,00	2	0,18	0,06
Boş zamana göre seçim	2,74	2,56	0,00	25,00	2	0,18	0,04
İşçi yüküne göre seçim	2,71	2,53	0,00	25,00	2	0,18	0,04
EnbPA							
Soldan sağa seçim	3,03	1,80	0,00	33,33	14	1,25	0,07
Sağdan sola seçim	2,59	1,35	0,00	33,33	14	1,25	0,06
Boş zamana göre seçim	2,84	1,60	0,00	33,33	14	1,25	0,04
İşçi yüküne göre seçim	2,87	1,63	0,00	33,33	14	1,25	0,04
EnbZaman/MI							
Soldan sağa seçim	2,70	2,62	0,00	25,00	1	0,09	0,08
Sağdan sola seçim	2,63	2,54	0,00	25,00	1	0,09	0,08
Boş zamana göre seçim	2,53	2,44	0,00	25,00	1	0,09	0,04
İşçi yüküne göre seçim	2,60	2,52	0,00	25,00	1	0,09	0,04

* Çözümsüz denemeler dâhil (*Aralık_{Min}* değeri 100 olacak şekilde)

** Çözümsüz denemeler dâhil değil

Çizelge 6.5. Optimali bilinen problemler için önerilen sezgiselin çözüm sonuçları

Problem	<i>n</i>	Örnek sayısı	<i>Ortalama Aralık_{Min}</i>	<i>Ortalama fark</i>	<i>En büyük fark</i>
Bowman	8	2	0	0	0
Heskiaoff	28	10	0	0	0
Jackson	11	21	2,38	0,14	1
Jaeschke	9	14	0	0	0
Lutzl	32	1	0	0	0
Mansoor	11	7	24,05	1,00	1
Mertens	7	18	0	0	0
Mitchell	21	15	2,78	0,13	1
Roszieg	25	9	5,29	0,44	1
Sawyer	30	4	0	0	0
Genel		101	3,05	0,16	1

7. SONUÇ VE ÖNERİLER

Bu çalışma kapsamında ele alınan montaj hattı modelinin klasik hatlardan en bariz farkı, öncelik ilişkileri sağlanacak şekilde aynı parçanın farklı montaj görevlerini eş zamanlı olarak gerçekleştiren birden fazla işçinin aynı istasyonda bulunmasına izin verilmesidir. Bu tip hatlara özellikle görev sayısı fazla, birden fazla işçinin birbirini engellemeden aynı parça üzerinde eş zamanlı çalışmasına izin verecek ölçüde büyük boyutlu ürünlerin üretildiği sistemlerde sıkça karşılaşılmaktadır. Buna karşın bu problemler akademik ilgiyi yeni yeni çekmeye başlamıştır.

Çalışmanın ilk bölümünde giriş bilgileri sunulduktan sonra 2. bölümünde montaj hattı dengeleme problemlerinin yanı sıra ele alınan problemlere ilişkin bazı kavramlardan bahsedilmiş, montaj hattı dengeleme problemlerinin sınıflandırması yapılmıştır. Bölüm 3 kapsamında ise ele alınan probleme benzerlik gösteren bazı montaj hattı dengeleme problemleri için literatür çalışmaları özetlenmiştir.

Bölüm 4’de ise problemin kesin çözümünün bulunmasında kullanılabilecek matematiksel model geliştirme çalışması yapılmıştır. İncelenen problemde performans ölçütü toplam işçi sayısının en küçüklenmesidir. Buna karşın hat uzunluğunun yani toplam istasyon sayısının en küçüklenmesi performans ölçütlü hat dengeleme problemi matematiksel model açısından incelenen probleme temel oluşturmaktadır. Bu nedenle öncelikle hat uzunluğunun en küçüklenmesine yönelik hat dengeleme probleminin karışık tam sayılı doğrusal programlama formundaki matematiksel modeli geliştirilmiş, modelde bulunan farklı tipteki karar değişkenleri ve kısıtların sayıları hesaplanmıştır. İkinci aşamada ise bu model temel alınarak toplam işçi sayısının en küçüklenmesi performans ölçütlü hat dengeleme probleminin yine aynı formda olan matematiksel modeli kurularak karar değişkeni ve kısıt sayıları hesaplanmıştır. Hat dengelemede bölgeleme kısıtlarının incelenen modellerin gerçek hayata uygulanabilirliği açısından özel bir önemi vardır. Bölgeleme kısıtları, en genel anlamda, görevlerin kaynaklara atanmasını sınırlandırır. Daha önce de belirtildiği gibi incelenen modelde aynı parçaya ait birden fazla görevin eş zamanlı yapılması mümkün olmaktadır. Problemin bu özelliğinden hareketle yeni bir bölgeleme kısıtı

ortaya çıkmaktadır. Çalışma kapsamında zaman bölgeleme kısıtları olarak adlandırılan bu yapıda iki görevin eş zamanlı yapılması ön görülmekte (pozitif zaman bölgeleme kısıtları) veya engellenmektedir (negatif zaman bölgeleme kısıtları). Bu kısıtlarda dâhil olmak üzere hat dengeleme de sıkça karşılaşılan diğer bölgeleme kısıtlarının önerilen matematiksel model kapsamında nasıl kullanılabileceği aynı bölüm kapsamında tek tek ifade edilmiş, uygun yapıda kısıtlar sunulmuştur. İncelenen problem montaj hattı dengeleme literatüründe yeni yeni çalışılmaya başlamış, literatürde henüz karşılaştırma problemleri yayınlanmamıştır. Bu nedenle aynı bölüm kapsamında çalışmada önerilen çözüm yöntemlerinin değerlendirilmesinde kullanılacak çeşitli boyutlarda 1260 adet karşılaştırma problemi üretilmiştir. Önerilen matematiksel modele her bir problem için 1 CPU saati zaman verilerek tüm problemler çözülmüş ve sonuçlar sunulmuştur. Sonuçlar incelendiğinde matematiksel modelin küçük boyutlu problemlerin optimal çözümlerini rahatlıkla bulabildiği, problem boyutunun büyümesiyle etkisini yitirdiği görülmüştür. Matematiksel model yoluyla çözülebilen en büyük boyutlu problemin görev sayısı 32 adettir. Bu sınıfta bulunan 24 adet problemde sadece birisine önerilen yöntemle optimal çözüm bulunabilmiştir. Bu sonuçların görülmesinden sonra optimal çözümlerin bulunmasında kullanılabilecek, probleme özgü bilgileri kullanan özel amaçlı bir dal sınır algoritmasının geliştirilmesine yönelik çalışmaların yapılmasına karar verilmiştir.

Bölüm 5'e ilgili problemlerin alt sınır hesaplamalarına ilişkin bazı bilgiler verilerek başlanmıştır. Problemin alt sınır değerinin hesaplanmasında kullanılabilecek yeni bir yöntem önerilerek diğer hat dengeleme problemlerinde kullanılan alt sınır hesaplama yöntemlerinin incelenen problemlerde daha etkin olarak kullanılabilmesi için yapılabilecek modifikasyonlar irdelenmiştir. Ayrıca, önerilen alt sınır hesaplama yönteminin nasıl kullanılabileceğine ilişkin açıklayıcı bir örnek verilmiştir. İncelenen örnek problem için önerilen yeni alt sınır hesaplama yönteminin hat dengeleme problemlerinde çok sık kullanılan diğer bir yöntemden daha iyi değer hesapladığı görülmüştür. Bu çalışmadan sonra incelenen problemlerin optimal çözümlerinin bulunmasında kullanılabilecek dal sınır tabanlı özel bir algoritma sunulmuştur. Algoritma, dallandırma işleminde görev tabanlı dallandırma stratejisinden hareketle

geliştirilmiş özel bir zaman tabanlı dallandırma stratejisi kullanmaktadır. Geliştirilen dallandırma stratejisi ve probleme özgü bilgilerin analizi yoluyla iki adet baskınlık kuralı geliştirilmiş, bunlara ilave olarak literatürde bulunan üç adet baskınlık kuralı da algoritma kapsamında kullanılmıştır. Ayrıca algoritma çeşitli uygunluk kurallarını da kapsamaktadır. Algoritmanın en önemli bileşenlerinden bir tanesi, algoritma performansı üzerinde oldukça büyük etkiye sahip olan “sezgisel güdüm” mekanizmasıdır. Bu mekanizmayla lazer arama stratejisinin birlikte kullanımının özellikle büyük boyutlu problemlere daha kaliteli çözümlerin üretilmesini sağladığı deneysel çalışmalarda görülmüştür. Literatürde, incelenen probleme çeşitli benzerlikler gösteren DA-MHDP’in çözümü için önerilmiş dal-sınır algoritması, BS, modifiye edilerek önerilen algoritma ile karşılaştırılmış, gerek çözüm kalitesi gerekse de çözüm zamanı açısından önerilen algoritmanın daha iyi performans sergilediği görülmüştür. Ulaşılan bu sonucun öngörülen bazı nedenleri şu şekilde açıklanabilir. BS, basit montaj hatlarının çözümünde çok başarılı sonuçlar veren SALOME isimli dal-sınır algoritması temel alınarak geliştirilmiştir. Basit montaj hatlarında istasyon problemi diye bir zorluk söz konusu değildir. Öyle ki, toplam zamanı çevrim zamanını aşmayan ve öncelik ilişkilerini ihmal etmeyen görev grubu boş bir istasyona direkt atanabilir. Buna karşın incelenen problemde toplam zamanı istasyon kapasitesini aşmayan ve öncelik ilişkilerini sağlayan görev grubunun boş bir istasyona atanıp atanamayacağı problemi NP-tam bir problemdir. Dolayısıyla, “atanabileceği en az işçi sayısının bulunması” problemi de NP-zor bir problemdir. Bu nedenle BS, “istasyon problemi” olarak isimlendirilen bu problemin çözümü için özel geliştirilmiş bir dal-sınır algoritması kullanmaktadır. Eğer ilgili istasyon yükü optimal çözümün hatta uygun bir çözümün parçası değilse, istasyon probleminin çözümü için harcanan süre boşa gitmektedir. Özellikle de başlangıç istasyonlarında bu durum oldukça külfetli olmaktadır. Bu nedenle BS, özellikle büyük boyutlu problemlerin çoğunda uygun çözüm dahi bulamamaktadır. Önerilen algoritma ise kısa süre içerisinde uygun çözüme ulaşması, daha sonra da ulaştığı çözümü aday çözüm olarak kullanması nedeniyle daha fazla miktarda düğümü harici olarak budayabilmektedir. Sezgisel güdüm özelliği de algoritma performansını son derece artırmıştır. Geliştirilen görev tabalı dallandırma stratejisi ve probleme has bilgilerden

hareketle ortaya çıkartılan yeni uygunluk ve baskınlık kurallarının da algoritma performansı üzerinde son derece etkin olduğu ön denemelerde görülmüştür.

Bölüm 6 kapsamında ise problemin çözümünde kullanılabilecek kurucu sezgisel bir yöntem geliştirilmiştir. Çalışma kapsamında da belirtildiği gibi incelenen problem esasında 3 alt problemden oluşmaktadır. Bu problemler; istasyonlardaki işçi sayılarının belirlenmesi, görevlerin hangi istasyonda yapılacağına belirlenmesi ve son olarak da atandığı istasyonlarda hangi işçinin hangi göreve hangi zamanda başlayacağına belirlenmesidir. İşte yeni sezgisel bütün bu alt problemlerle başa çıkabilecek mekanizmalar içermekte olup çok geçişli, istasyon tabanlı, statik, öncelik kuralına dayalı bir yapıdadır. Algoritma kapsamında görev önceliklerinin belirlenmesi amacıyla literatürde diğer hat dengeleme problemlerinde kullanılan öncelik kurallarından faydalanılmıştır. Sezgiselin matematiksel model yoluyla optimal çözümleri bulunan küçük boyutlu problemler üzerinde performansı incelenmiş ve kabul edilebilir kalitede, çok hızlı çözümler ürettiği görülmüştür. Daha büyük boyutlu diğer problemlerin çözümleri sezgiselin kullanımı yoluyla bulunarak test sonuçları sunulmuştur.

Daha önce de belirtildiği gibi göz önünde bulundurulan problemle gerçek hayatta sık karşılaşılan akademik alandaki çalışmalar yeni yeni başlamaktadır. Söz konusu problem montaj hattı dengeleme problemlerinin ciddi bir bölümünü kapsamaktadır ve hat yerleşimi, paralelleştirme, farklı görev zamanı tipleri, farklı ürün sayıları, farklı bölgeleme kısıtları gibi çeşitli yaklaşımlarla oluşturulmuş birçok versiyonu göz önünde bulundurularak problem tanımı yapıp çözümü için matematiksel modeller, kesin algoritmalar ve sezgisel metotların geliştirilmesi konularında çalışmaların yapılması gerekmektedir.

Bu çalışma kapsamında matematiksel model ve dal-sınır tabanlı olmak üzere iki farklı kesin çözüm metodu ve bir adet de sezgisel çözüm metodu önerilmiştir. Kesin çözüm metodlarından matematiksel modelin kullanımının oldukça sınırlı olduğu görülmüş, orta boyutlardaki problemlerin optimal çözümlerinin dahi izin verilen zaman içerisinde bulunamadığı görülmüştür. Önerilen dal-sınır tabanlı algoritma ise optimal

çözümün bulunmasında daha başarılı bir grafik çizmekte ve orta boyutlu problemlerin çoğunun optimal çözümlerini bulabilmektedir. Buna karşın problem boyutunun büyümesiyle bu algoritma da etkinliğini kaybetmektedir. Çalışma kapsamında da belirtildiği gibi incelenen problem orta ve uzun dönemli planlama periyodunda bulunan bir problemidir. Çözüm için 1 CPU saati zaman verilmesine karşın, problemin planlama periyodu göz önüne alındığında çözüm için algoritmaya daha uzun zamanlar izin verilebilir. Bu zaman çoğu durumda 1 aya kadar çıkartılabileceği gibi paralelleştirme yoluyla birden fazla bilgisayarın aynı problemi çözmesi sağlanabilir. Algoritmanın yapısı böyle bir paralelleştirmeye uygundur, öyle ki dal sınır ağacı alt ağaçlardan oluşan bir yapıya sahiptir. Bu yolla gerçek hayat problemlerinin optimal çözümlerinin daha da büyük olan belirli boyuta kadar bulunabileceği düşünülmektedir. Buna karşın problem NP-zor yapıdadır. Bu özelliği nedeniyle mevcut imkânlarla çözülebilecek problemler boyut açısından bir üst sınıra kesinlikle sahiptir. Her ne kadar sezgisel yöntem bu tip durumlara çözüm getirebilecek bir özelliğe sahip olsa da, yöntemin kurucu bir yapıda olması nedeniyle bulunacak çözümlerin kalitesi de eleştiriye açıktır. Bu nedenle problemin çözümü için kullanılacak etkin meta sezgisel metotların geliştirilmesi gerekmektedir. Çalışma kapsamında da belirtildiği gibi incelenen problem görevlerin paralel yapılmasına izin vermesi nedeniyle paralel makine çizelgeleme problemi ve kaynak kısıtlı proje planlama problemine benzerlik göstermesine rağmen teorik yapısı itibarıyla bu problemlerden oldukça farklıdır. Bu nedenle etkin iyileştirici sezgisel algoritmalar geliştirmek için bütün meta sezgisel metotların ortak bileşeni olan çözümün kodlanması konusunda bile detaylı çalışmaların yapılması gerekmektedir. Gerek tek noktadan arama yapan algoritmalarındaki komşu üretim mekanizmaları, gerekse de popülasyon tabanlı algoritmalarda ki yeni bireyler üreten bileşenlerin tamamının probleme özgü bilgiler kullanılarak tasarlanması gerekmektedir. Dolayısıyla bu çalışmaların devamı olarak aynı problemin çözümü için meta sezgisel yöntemlerin kullanımına yönelik araştırmalar yapılması planlanmaktadır. Sonraki aşamalar da ise gerek önerilen kesin çözüm yöntemleri, gerekse de sezgisel yöntemlerin aynı sınıfta bulunan komşu problemler üzerinde kullanımları konusunda çalışmalar yapılacaktır. Bu problemler gerek hat yerleşimi açısından gerekse de işlem zamanları açısından farklılık göstermektedir. Diğer bir önemli unsur ise problemleri

gerçek hayat problemlerine yaklařtıran bölgeleme kısıtlarıdır. Bu kısıtları içeren problemler konusunda da ilave çalışmaların yapılmasına ihtiyaç vardır.

KAYNAKLAR

1. Erel, E., Sarin, S.C., "A survey of the assembly line balancing procedures", *Production Planning and Control*, 9: 414-434 (1998).
2. Salveson, M.E., "The assembly line balancing problem", *The Journal of Industrial Engineering*, 6: 18-25 (1955).
3. Süer, G.A., "Designing Parallel Assembly Lines", *Computer and Industrial Engineering*, 35: 467-470 (1998).
4. Scholl, A., Boysen, N., "Designing parallel assembly lines with split workplaces: Model and optimization procedure", *International Journal of Production Economics*, 119: 90-100 (2009).
5. Scholl, A., "Balancing and sequencing assembly lines, 2nd ed.", *Springer-Verlag Co.*, Newyork, 1-288 (1999).
6. Bayraktaroğlu, A.E., "Basit U-tipi montaj hattı dengelemede analitik yöntemlerin karşılaştırılması", Yüksek Lisans Tezi, *İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü*, İstanbul, 3-60 (2007).
7. Dimitriadis, S.G., "Assembly line balancing and group working: A heuristic procedure for workers' groups operating on the same product and Workstation", *Computers and Operations Research*, 33: 2757-2774 (2006).
8. Çerçioğlu, H., "Stokastik paralel montaj hattı dengeleme problemi için yeni modeller", Doktora Tezi, *Gazi Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, 1-50 (2009).
9. Ağpak, K., Gökçen, H., Saray, N.N., Özel, S., "Stokastik görev zamanlı tek modelli U tipi montaj hattı dengeleme problemleri için bir sezgisel", *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 17: 115-124 (2002).
10. Boysen, N., Fliedner, M., Scholl, A., "A classification of assembly line balancing problems", *European Journal of Operational Research*, 183: 674-693 (2007).
11. Park, K., Park, S., Kim, W., "A heuristic for an assembly line balancing problem with incompatibility, range, and partial precedence constraints", *Computers and Industrial Engineering*, 32: 321-332 (1997).
12. Scholl, A., "Data of assembly line balancing problems", *Darmstadt Technical University*, Darmstadt, 1-32 (1997).
13. Agrawal, S., Tiwari, M. K., "A collaborative ant colony algorithm to stochastic mixed-model U-shaped disassembly line balancing and sequencing problem", *International Journal of Production Research*, 46: 1405-1429 (2008).

14. Falkenauer, E., "Line Balancing in the Real World", *Proceedings of the International Conference on Product Lifecycle Management*, Lumiere University of Lyon-France, 1-10 (2005).
15. Becker, C., Scholl, A., "Balancing assembly lines with variable parallel workplaces: Problem definition and effective solution procedure", *European Journal of Operational Research*, 199: 359-374 (2009).
16. Wu, E.F., Jin, Y., Bao, J.S., Hu, X.F., "A branch-and-bound algorithm for two-sided assembly line balancing", *The International Journal of Advanced Manufacturing Technology*, 39: 1009-1015 (2008).
17. Özcan, U., Toklu, B., "Balancing of mixed-model two-sided assembly lines", *Computers and Industrial Engineering*, 57: 217-227 (2009).
18. Özcan, 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 (2009).
19. Jackson J.R., "A computing procedure for a line balancing problem", *Management Science*, 2: 261-271 (1956).
20. Johnson R.V., "Assembly line balancing algorithms: Computation comparisons", *International Journal of Production Research*, 19: 277-287 (1981).
21. Bowman, E. H., "Assembly Line Balancing by Linear programming", *Operations Research*, 8: 385-389 (1960).
22. White, W.W., "Comments on a Paper by Bowman", *Operations Research*, 9: 274-276 (1961).
23. Esin, C.E., "Çok-adamlı operasyonlar içeren montaj hattı dengeleme", Yüksek Lisans Tezi, *Ortadoğu Teknik Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, 1-78 (2007).
24. Baybars, İ., "A Survey of Exact Algorithms for the Simple Assembly Line Balancing Problem", *Management Science*, 32: 909-932 (1986).
25. Queyranne, M., "Bounds for assembly line balancing heuristics", *Operations Research*, 33: 1353-1359 (1985).
26. Scholl, A., Becker, B., "State-of-the-art exact and heuristic solution procedures for simple assembly line balancing", *European Journal of Operational Research*, 168: 666-693 (2006).
27. Becker, C., Scholl, A., "A survey on problems and methods in generalized assembly line balancing", *European Journal of Operational Research*, 168: 694-715 (2006).

28. Boysen, N., Fliedner, M., Scholl, A., “Sequencing mixed-model assembly lines: Survey, classification and model critique”, *European Journal of Operational Research*, 192: 349–373 (2009).
29. Freeman D.R., Jucker, J.V., “The line balancing problem”, *Journal of Industrial Engineering*, 18: 361-364 (1967).
30. Buxey, G.M., “Assembly line balancing with multiple stations”, *Management Science*, 20: 1010–1021 (1974).
31. Pinto, P.A., Dannenbring, D.G., Khumawala, B.M., “Branch and bound and heuristic procedures for assembly line balancing with paralleling of stations”, *International Journal of Production Research*, 19: 565–576 (1981).
32. Sarker, B.R., Shanthikumar, J.G., “A generalized approach for serial or parallel line balancing”, *International Journal of Production Research*, 21: 109–133 (1983).
33. Bard, J.F., “Assembly line balancing with parallel workstations and dead time”, *International Journal of Production Research*, 27: 1005–1018 (1989).
34. Inman, R.R., Leon, M., “Scheduling duplicate serial stations in transfer lines”, *International Journal of Production Research*, 32: 2631–2644 (1994).
35. Askin, R.G., Zhou, M., “A parallel station heuristic for the mixed-model production line balancing problem”, *International Journal of Production Research*, 35: 3095–3105 (1997).
36. McMullen, P.R., Frazier, G.V., “A heuristic for solving mixed-model line balancing problems with stochastic task and parallel stations”, *International Journal of Production Economics*, 51: 177–190 (1997).
37. McMullen, P.R., Frazier, G.V., “Using simulated annealing to solve a multiobjective assembly line balancing problem with parallel workstations”, *International Journal of Production Research*, 36: 2717–2741 (1998).
38. Simaria, A.S., Vilarinho, P.M., “The simple assembly line balancing problem with parallel workstations - A simulated annealing approach”, *International Journal of Industrial Engineering : Theory Applications and Practice*, 8: 230-240 (2001).
39. Vilarinho, P.M., Simaria, A.S., “A two-stage heuristic method for balancing mixed-model assembly lines with parallel workstations”, *International Journal of Production Research*, 40: 1405–1420 (2002).
40. Bukchin, J., Rubinovitz, J., “A weighted approach for assembly line design with station paralleling and equipment selection”, *IIE Transactions*, 35: 573–585 (2003).

41. Vilarinho, P.M., Simaria, A.S., “ANTBAL: An ant colony optimization algorithm for balancing mixed-model assembly lines with parallel workstations”, *International Journal of Production Research*, 44: 291-303 (2006).
42. Ege, Y., Azizoglu, M., Ozdemirel, E., “Assembly line balancing with station paralleling”, *Computers and Industrial Engineering*, 57: 1218-1225 (2009).
43. Akpınar, S., Bayhan, G.M., “A hybrid genetic algorithm for mixed model assembly line balancing problem with parallel workstations and zoning constraints”, *Engineering Applications of Artificial Intelligence*, 24: 449-457 (2011).
44. Pinto, P.A., Dannenbring, D.G., Khumawala, B.M., “A branch and bound algorithm for assembly line balancing with paralleling”, *International Journal of Production Research*, 13: 183–196 (1975).
45. Bukchin, Y., Rabinowitch, I., “A branch-and-bound based solution approach for the mixed-model assembly line-balancing problem for minimizing stations and task duplication costs”, *European Journal of Operational Research*, 174: 492-508 (2006).
46. Gökçen, H., Ağpak, K., Benzer, R., “Balancing of parallel assembly lines”, *International Journal of Production Economics*, 103: 600-609 (2006).
47. Benzer, R., Gökçen, H., Çetinyokus, T., Çerçioğlu, H., “A network model for parallel line balancing problem”, *Mathematical Problems in Engineering*, 2007: 1-12 (2007).
48. Scholl, A., Klein, R., “SALOME: a bidirectional branch and bound procedure for assembly line balancing”, *INFORMS Journal on Computing*, 9: 319-334 (1997).
49. Kara, Y., Gökçen, H., Atasagun, Y., “Balancing parallel assembly lines with precise and fuzzy goals”, *International Journal of Production Research*, 48: 1685–1703 (2010).
50. Özcan, U., Çerçioğlu, H., Gökçen, H., Toklu, B., “Balancing and sequencing of parallel mixed-model assembly lines”, *International Journal of Production Research*, 48: 5089–5113 (2010).
51. Özbakır, L., Baykasoğlu, A., Görkemli, B., Görkemli, L., “Multiple-colony ant algorithm for parallel assembly line balancing problem”, *Applied Soft Computing*, 11: 3186-3198 (2011).
52. Bartholdi, J.J., “Balancing two-sided assembly lines: A case study”, *International Journal of Production Research*, 31: 2447–2461 (1993).

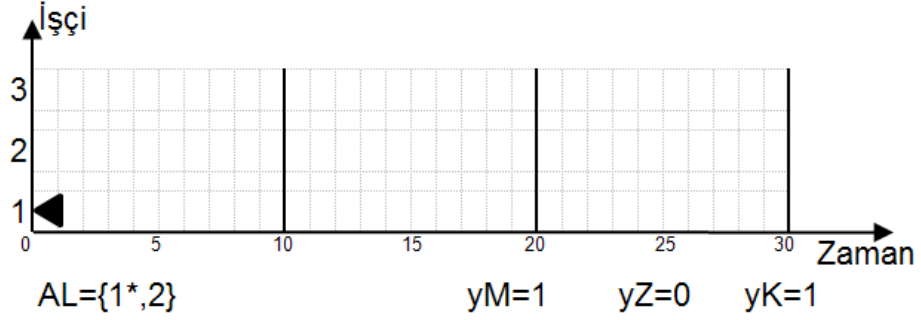
53. Kim, Y.K., Kim, Y., Kim, Y.J., “Two-sided assembly line balancing: a genetic algorithm approach”, *Production Planning and Control*, 11: 44–53 (2000).
54. Lee, T.O., Kim, Y., Kim, Y.K., “Two-sided assembly line balancing to maximize work relatedness and slackness”, *Computers and Industrial Engineering*, 40: 273–292 (2001).
55. Lapierre S.D., Ruiz, A.B., “Balancing assembly lines: an industrial case study”, *Journal of The Operations Research Society*, 55: 589–597 (2004).
56. Hu., X., Wu, E., Jin, Y., “A station-oriented enumerative algorithm for two-sided assembly line balancing”, *European Journal of Operational Research*, 186: 435-440 (2008).
57. Hoffmann, T.R., “Assembly line balancing with a precedence matrix”, *Management Science*, 9: 551–562 (1963).
58. Baykasoğlu, A., Dereli, T., “Two-sided assembly line balancing using an ant-colony-based heuristic”, *The International Journal of Advanced Manufacturing Technology*, 36: 582-588 (2008).
59. 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).
60. Özcan, U., Toklu, B., “A tabu search algorithm for two-sided assembly line balancing”, *International Journal of Advanced Manufacturing Technology*, 43: 822–829 (2009).
61. 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).
62. Xiaofeng, H., Erfei, W., Jinsong, B., Ye, J., “A branch-and-bound algorithm to minimize the line length of a two-sided assembly line”, *European Journal of Operational Research*, 206: 703-707 (2010).
63. Özcan, 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).
64. Özcan, U., Toklu, B., “Balancing two-sided assembly lines with sequence-dependent setup times”, *International Journal of Production Research*, 48: 5363–5383 (2010).
65. Özcan, U., Gökçen, H., Toklu, B., “Balancing parallel two-sided assembly lines”, *International Journal of Production Research*, 48: 4767–4784 (2010).

66. Chakravarty, A.K., Shtub, A., “Dynamic manning of long cycle assembly lines with learning effect”, *IIE Transactions*, 18: 392–397 (1986).
67. Shtub, A., “Serial production systems”, Unpublished Ph.D. dissertation, *University of Washington*, Washington, 1-85 (1982).
68. Akagi, F., Hirokazu, O., Kikuchi, S., “A method for assembly line balancing with more than one worker in each station”, *International Journal of Production Research*, 21: 755-770 (1983).
69. Wilson, J.M., “Formulation of a problem involving assembly lines with multiple manning of work stations”, *International Journal of Production Research*, 24: 59-63 (1986).
70. Johnson, R.V., “Balancing assembly lines for teams and work groups”, *International Journal of Production Research*, 29: 1205–1214 (1991).
71. Johnson, R.V., “Optimally balancing large assembly lines with FABLE”, *Management Science*, 34: 240–253 (1988).
72. Bukchin, J., Dar-El, E.M., Rubinovitz, J., “Team oriented assembly system design: A new approach”, *International Journal of Production Economics*, 51: 47–57 (1997).
73. Bukchin, J., Masin, M., “Multi-objective design of team oriented assembly systems”, *European Journal of Operational Research*, 156: 326–352 (2004).
74. Hu, T.C., “Parallel sequencing and assembly line problems”, *Operations Research*, 9: 841-848 (1961).
75. Cevikcan, E., Durmusoglu, M.B., Unal, M.E., “A team-oriented design methodology for mixed model assembly systems”, *Computers and Industrial Engineering*, 56: 576-599 (2009).
76. Bedworth, D.D., Bailey, J., “Integrated production control systems, analysis, design”, *John Wiley and Sons*, Hoboken, USA, 370–374 (1987).
77. Kilbridge, M.D., Wester, L., “A heuristic method of assembly line balancing”, *Journal of Industrial Engineering*, 12: 292–298 (1961).
78. Fattahi, P., Roshani, A., Roshani, A., “A mathematical model and ant colony algorithm for multi-manned assembly line balancing problem”, *International Journal of Advanced Manufacturing Technology*, 53:363-378 (2011).
79. Shim, S.O., “Generating subproblems in branch and bound algorithms for parallel machines scheduling problem”, *Computers and Industrial Engineering*, 57: 1150-1153 (2009).

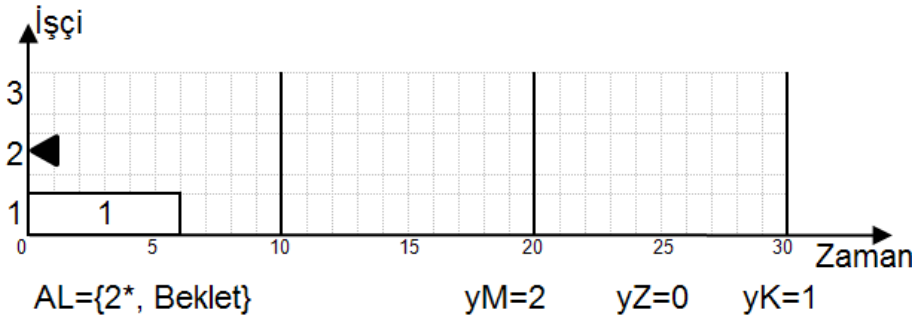
80. Artigues, C., Demassey, S., Néron E., “Resource-Constrained Project Scheduling: Models, Algorithms, Extensions and Applications”, *John Wiley and Sons, Inc.*, Hoboken, USA, 21-277 (2008).
81. Schrage L., Baker K.R., “Dynamic programming solution of sequencing problems with precedence constraints”, *Operations Research*, 26: 444-449 (1978).
82. Nourie F.J., Venta E.R., “Finding optimal line balances with OptPack”, *Operations Research Letters*, 10: 165-171 (1991).
83. Hackman S.T., Magazine M.J., Wee T.S., “Fast, effective algorithms for simple assembly line balancing problems”, *Operations Research*, 37: 916–924 (1989).
84. Scholl, A., VoB, “Simple Assembly Line Balancing – Heuristic Approaches”, *Journal of Heuristic*, 2: 217-244 (1996).
85. Franck, B., Neumann, K., Schwindt, C., “Truncated branch-and-bound, schedule-construction, and schedule-improvement procedures for resource-constrained project scheduling”, *OR Spektrum*, 23: 297-324 (2001).
86. Korte, B., Vygen, J., “Combinatorial Optimization: Theory and Algorithms 3rd ed.”, *Springer-Verlag*, Germany, 91-115 (2006).

EKLER

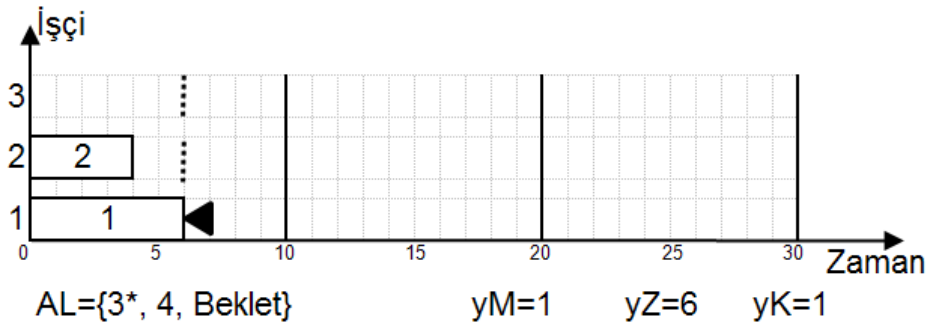
Ek-1 Örnek Dal-Sınır Algoritması Uygulamasına İlişkin Kısmi Çözümlere ait Gant Diyagramları



Şekil 1.1. Seviye 0 Düğüm 0 (kök düğüm)'a ilişkin Gant diyagramı

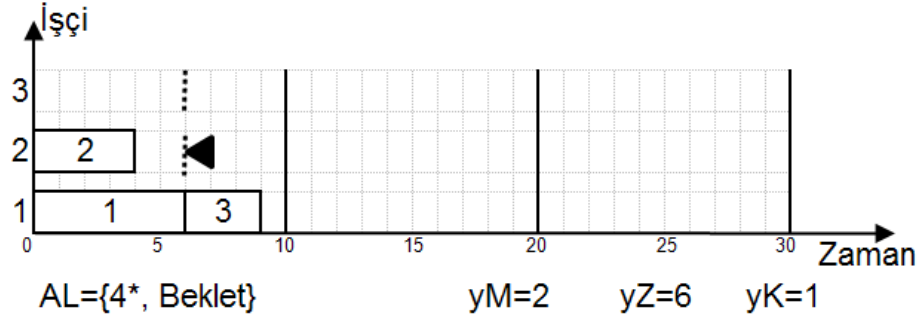


Şekil 1.2. Seviye 1 Düğüm 1'e ilişkin Gant diyagramı

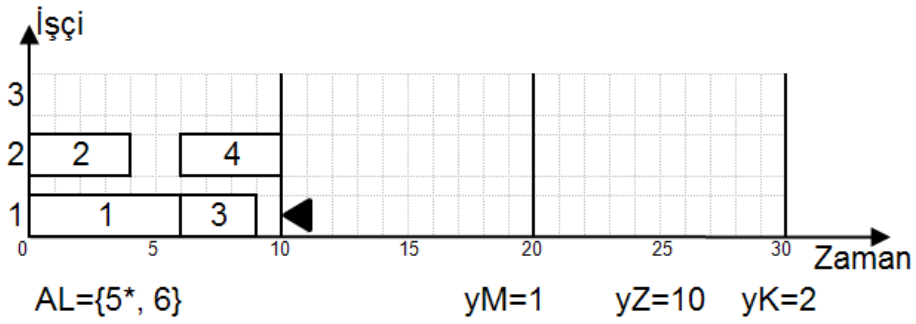


Şekil 1.3. Seviye 2 Düğüm 2'e ilişkin Gant diyagramı

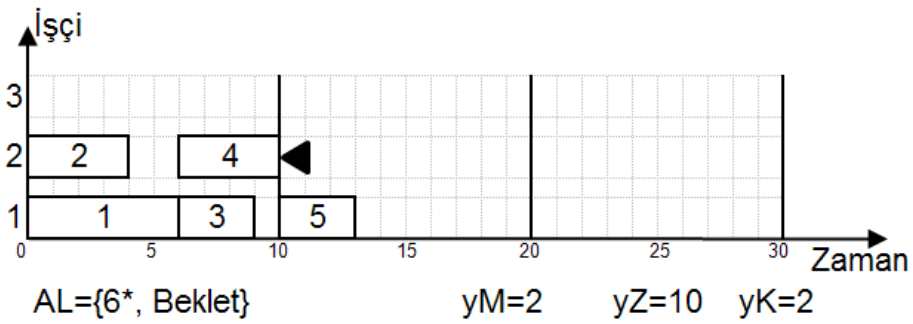
Ek-1 (Devam) Örnek Dal-Sınır Algoritması Uygulamasına İlişkin Kısmi Çözümlere ait Gant Diyagramları



Şekil 1.4. Seviye 3 Düğüm 3'e ilişkin Gant diyagramı

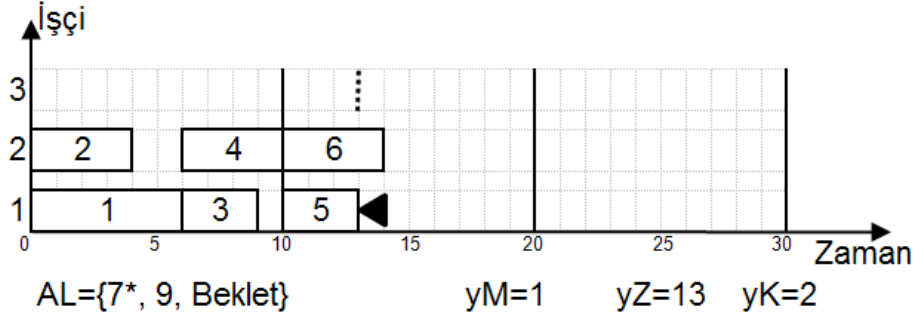


Şekil 1.5. Seviye 4 Düğüm 4'e ilişkin Gant diyagramı

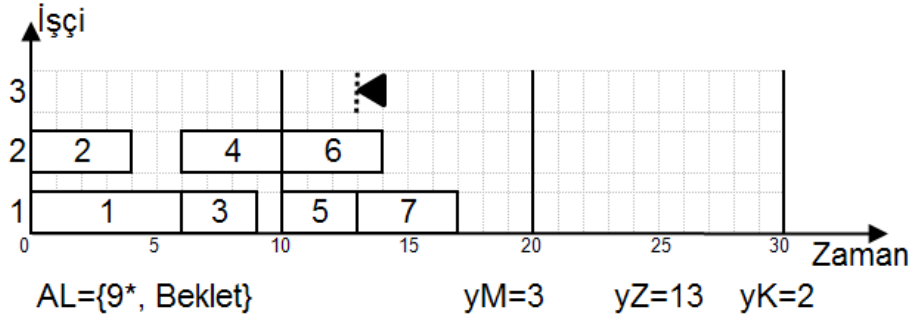


Şekil 1.6. Seviye 5 Düğüm 5'e ilişkin Gant diyagramı

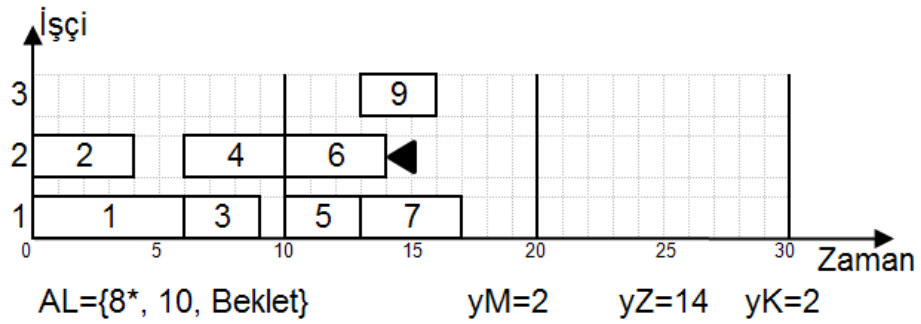
Ek-1 (Devam) Örnek Dal-Sınır Algoritması Uygulamasına İlişkin Kısmi Çözümlere ait Gant Diyagramları



Şekil 1.7. Seviye 6 Düğüm 6'a ilişkin Gant diyagramı

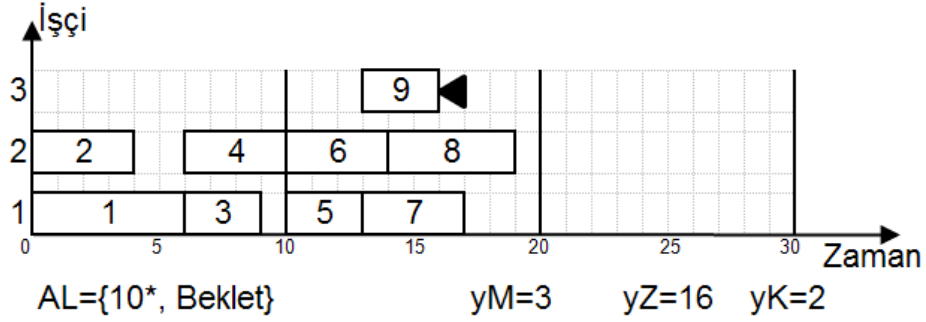


Şekil 1.8. Seviye 7 Düğüm 7'e ilişkin Gant diyagramı

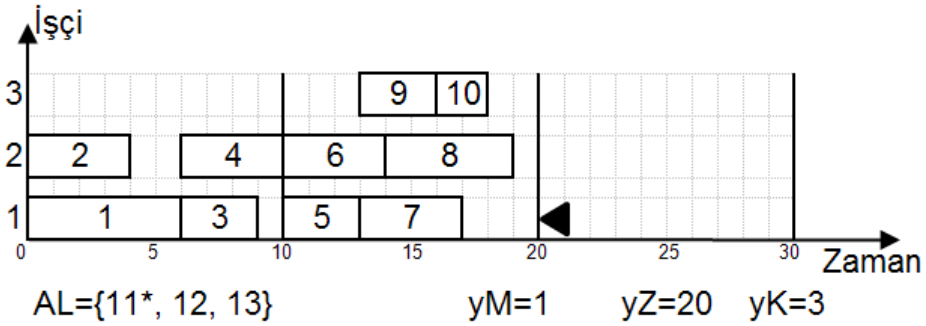


Şekil 1.9. Seviye 8 Düğüm 8'e ilişkin Gant diyagramı

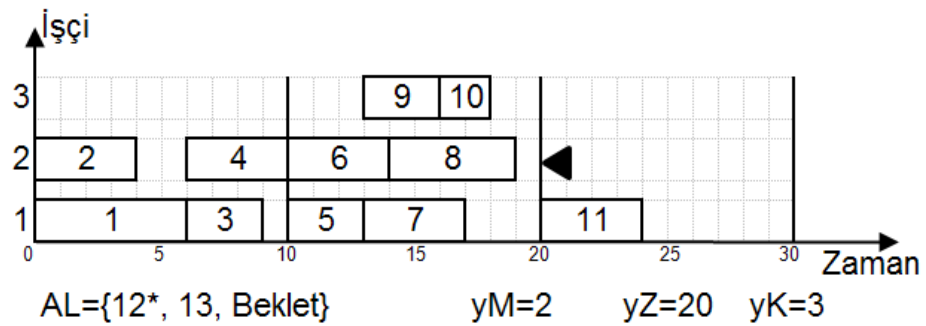
Ek-1 (Devam) Örnek Dal-Sınır Algoritması Uygulamasına İlişkin Kısmi Çözümlere ait Gant Diyagramları



Şekil 1.10. Seviye 9 Düğüm 9'a ilişkin Gant diyagramı

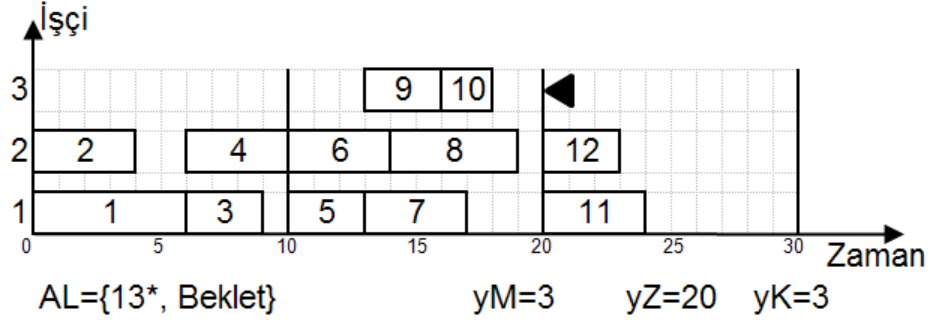


Şekil 1.11. Seviye 10 Düğüm 10'a ilişkin Gant diyagramı

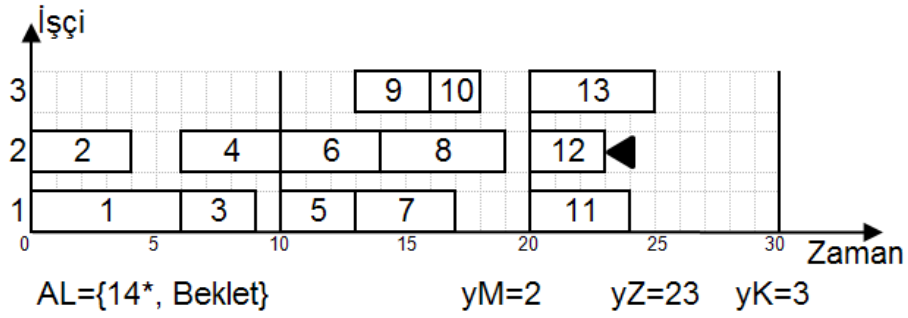


Şekil 1.12. Seviye 11 Düğüm 11'e ilişkin Gant diyagramı

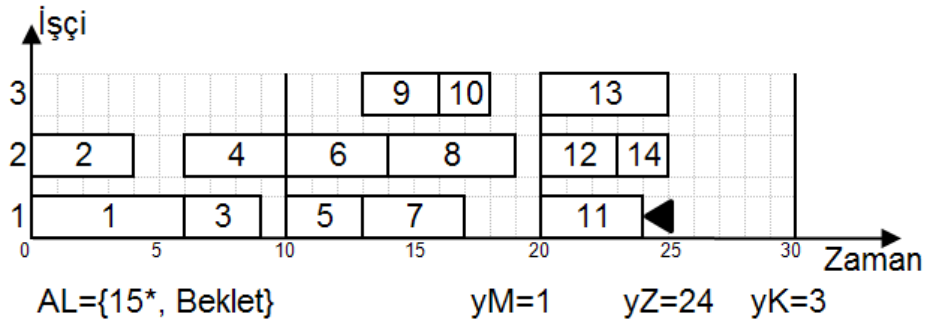
Ek-1 (Devam) Örnek Dal-Sınır Algoritması Uygulamasına İlişkin Kısmi Çözümlere ait Gant Diyagramları



Şekil 1.13. Seviye 12 Düğüm 12'e ilişkin Gant diyagramı

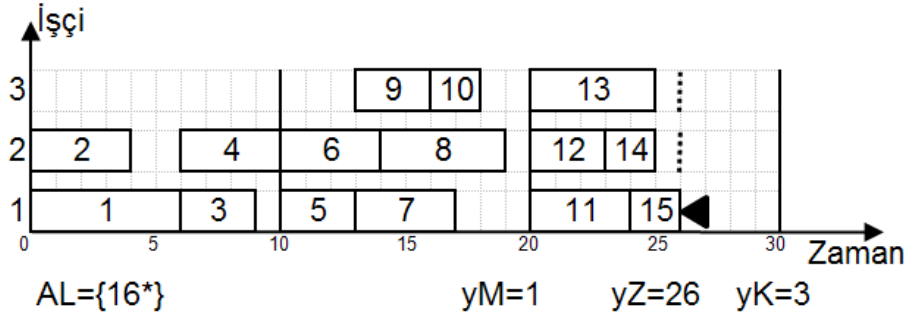


Şekil 1.14. Seviye 13 Düğüm 13'e ilişkin Gant diyagramı

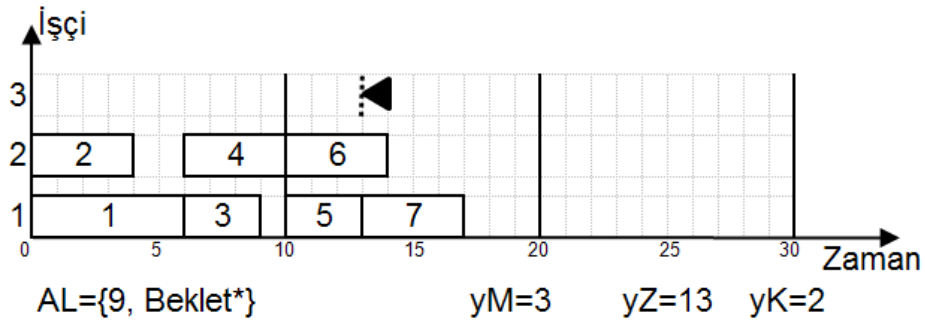


Şekil 1.15. Seviye 14 Düğüm 14'e ilişkin Gant diyagramı

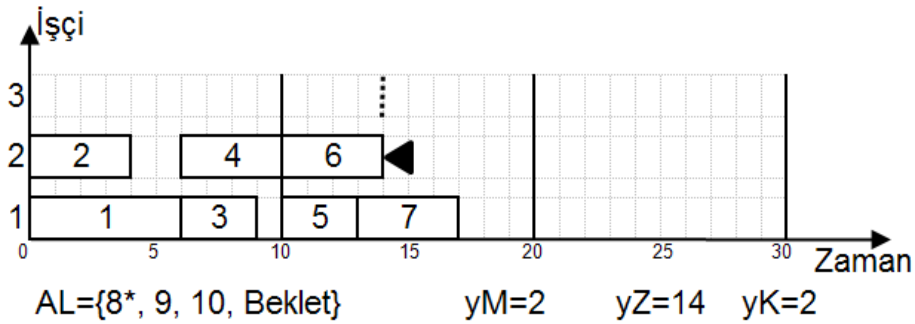
Ek-1 (Devam) Örnek Dal-Sınır Algoritması Uygulamasına İlişkin Kısmi Çözümlere ait Gant Diyagramları



Şekil 1.16. Seviye 15 Düğüm 15'e ilişkin Gant diyagramı

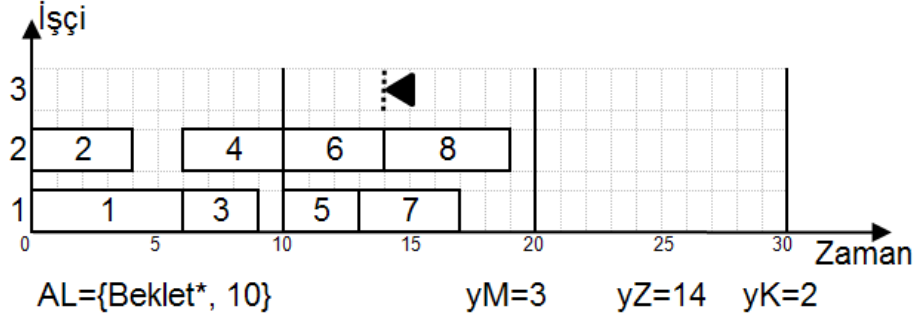


Şekil 1.17. Seviye 7 Düğüm 7'de seçim sırasının değiştirilmesi sonucu oluşan Gant diyagramı

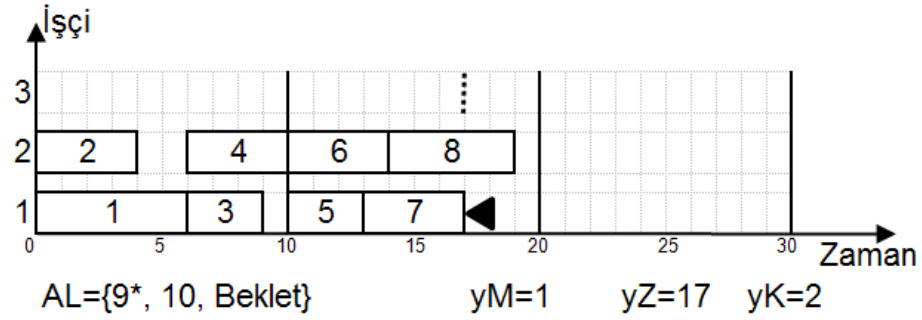


Şekil 1.18. Seviye 8 Düğüm 16'a ilişkin Gant diyagramı

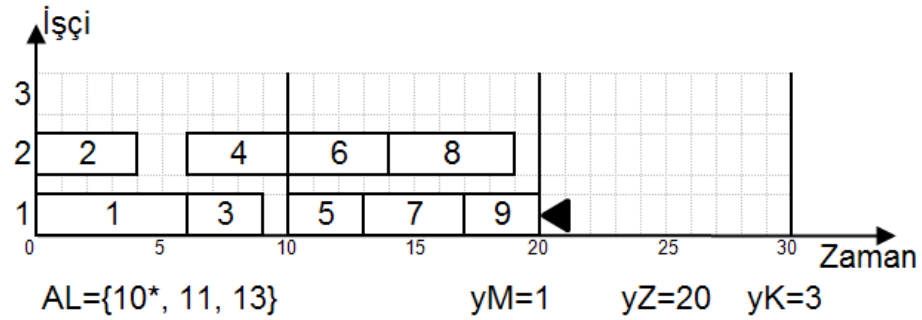
Ek-1 (Devam) Örnek Dal-Sınır Algoritması Uygulamasına İlişkin Kısmi Çözümlere ait Gant Diyagramları



Şekil 1.19. Seviye 9 Düğüm 17'e ilişkin Gant diyagramı

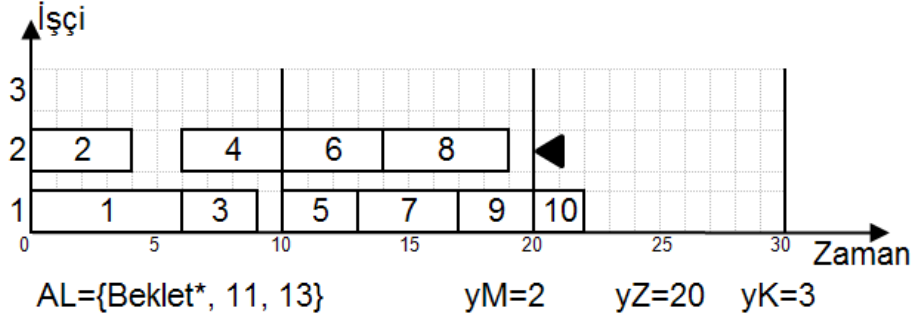


Şekil 1.20. Seviye 10 Düğüm 18'e ilişkin Gant diyagramı

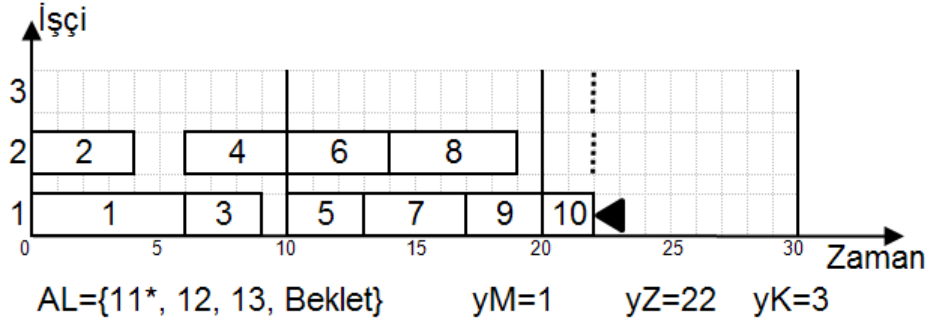


Şekil 1.21. Seviye 11 Düğüm 19'a ilişkin Gant diyagramı

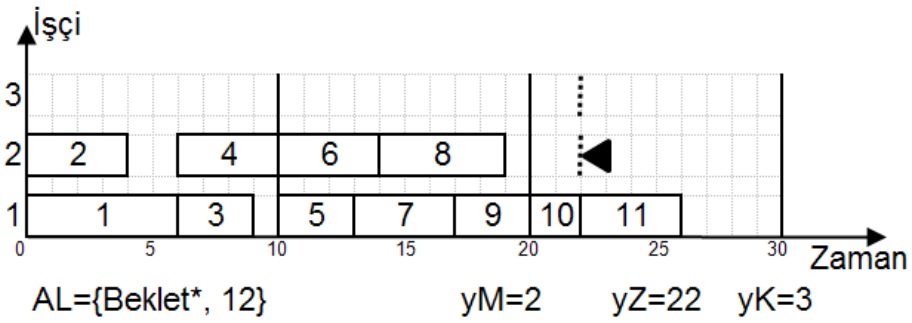
Ek-1 (Devam) Örnek Dal-Sınır Algoritması Uygulamasına İlişkin Kısmi Çözümlere ait Gant Diyagramları



Şekil 1.22. Seviye 12 Düğüm 20'e ilişkin Gant diyagramı

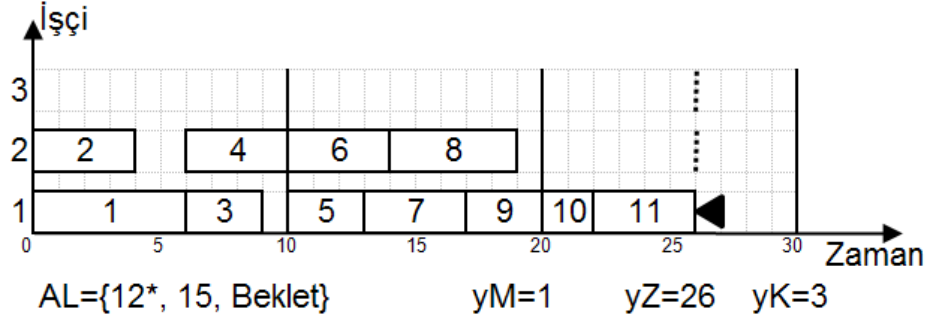


Şekil 1.23. Seviye 13 Düğüm 21'e ilişkin Gant diyagramı

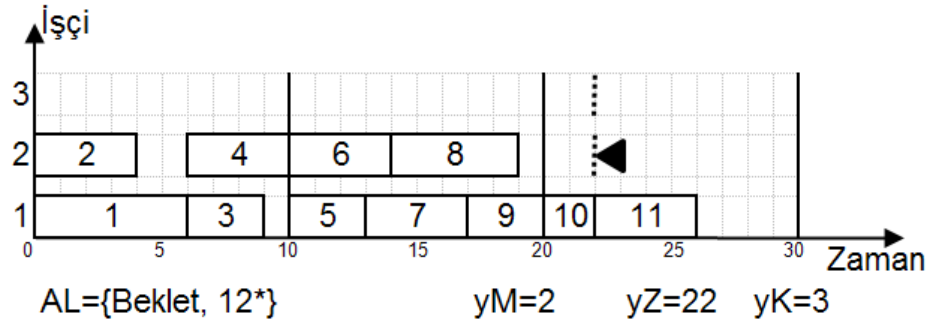


Şekil 1.24. Seviye 14 Düğüm 22'e ilişkin Gant diyagramı

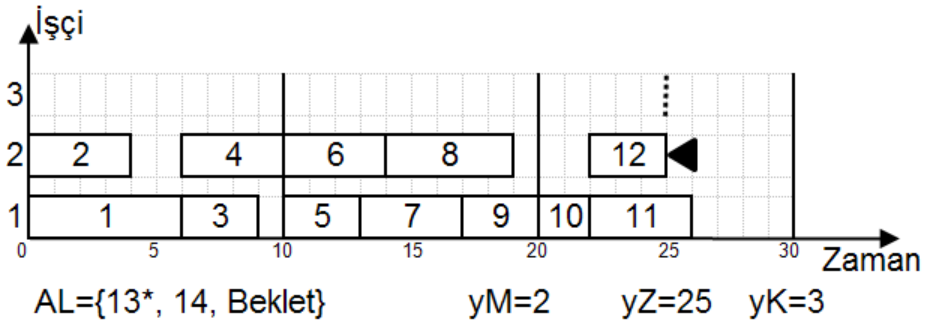
Ek-1 (Devam) Örnek Dal-Sınır Algoritması Uygulamasına İlişkin Kısmi Çözümlere ait Gant Diyagramları



Şekil 1.25. Seviye 15 Düzüm 23'e ilişkin Gant diyagramı

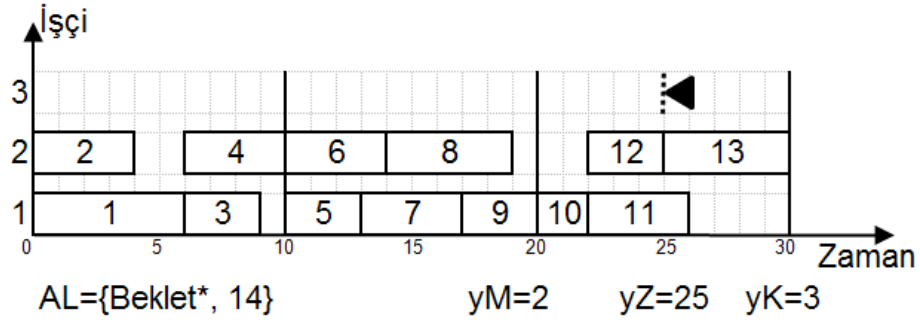


Şekil 1.26. Seviye 14 Düzüm 22'de seçim sırasının değiştirilmesi sonucu oluşan Gant diyagramı

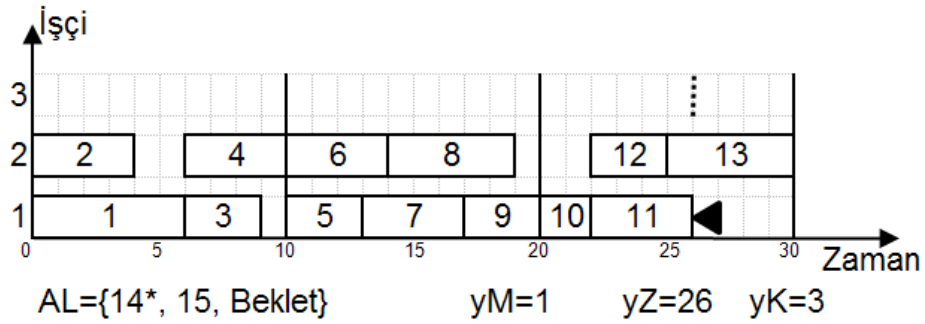


Şekil 1.27. Seviye 15 Düzüm 24'e ilişkin Gant diyagramı

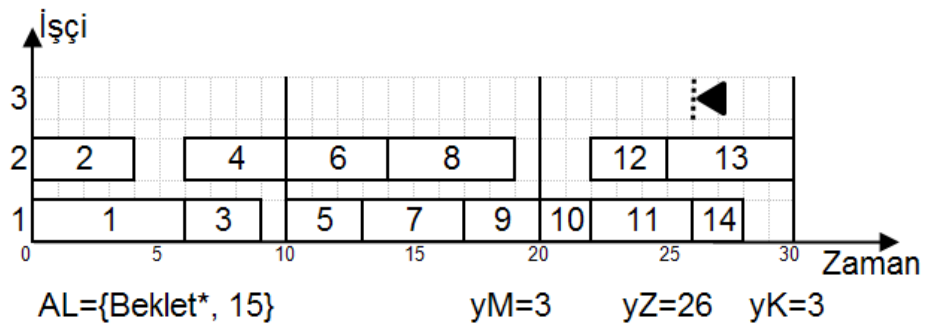
Ek-1 (Devam) Örnek Dal-Sınır Algoritması Uygulamasına İlişkin Kısmi Çözümlere ait Gant Diyagramları



Şekil 1.28. Seviye 16 Düğüm 25'e ilişkin Gant diyagramı

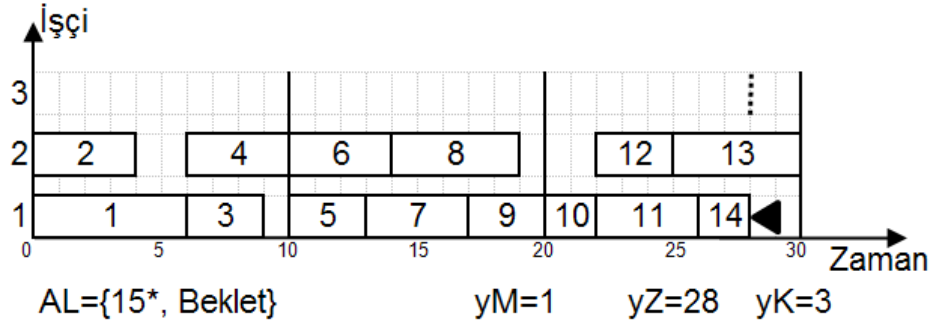


Şekil 1.29. Seviye 17 Düğüm 26'a ilişkin Gant diyagramı

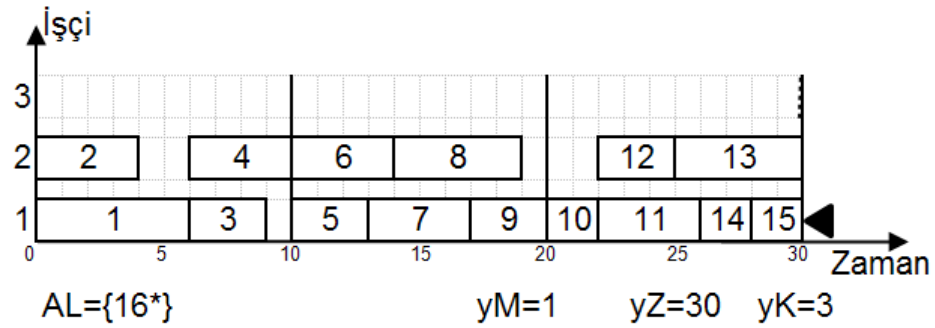


Şekil 1.30. Seviye 18 Düğüm 27'e ilişkin Gant diyagramı

Ek-1 (Devam) Örnek Dal-Sınır Algoritması Uygulamasına İlişkin Kısmi Çözümlere ait Gant Diyagramları



Şekil 1.31. Seviye 19 Düzüm 28'e ilişkin Gant diyagramı



Şekil 1.31. Seviye 20 Düzüm 29'a ilişkin Gant diyagramı

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : KELLEGÖZ, Talip
 Uyruğu : T.C.
 Doğum tarihi ve yeri : 06.10.1978 Çorum
 Medeni hali : Evli
 Telefon : 0 (318) 357 36 44
 E-mail : tkellegoz@kku.edu.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Yüksek Lisans	Kırıkkale Üniversitesi Endüstri Mühendisliği A.B.D.	2006
Lisans	Gazi Üniversitesi Endüstri Mühendisliği	2000
Lise	Şehit Nedim Tuğaltay Lisesi	1994

İş Deneyimi

Yıl	Yer	Görev
2007–2011	Gazi Üniversitesi	Araştırma Görevlisi
2004–2007	Kırıkkale Üniversitesi	Araştırma Görevlisi
2003–2004	Ondata Yazılım Ltd. Şti.	Yazılım Mühendisi
2000–2003	Naksan Plastik A.Ş.	Üretim Planlama Mühendisi

Yabancı Dil

İngilizce

Hobiler

Voleybol, bilgisayar teknolojileri, doğa sporları