

ROBOTİK HÜCRELERDE İKİ KRİTERLİ HAT DENGELEME

ELİF BÜŞRA ATASEVEN

**YÜKSEK LİSANS TEZİ
ENDÜSTRİ MÜHENDİSLİĞİ**

**TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

ARALIK 2014

ANKARA

Fen Bilimleri Enstitü onayı

Prof. Dr. Osman EROĞUL
Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

Prof. Dr. Tahir HANALIOĞLU
Anabilim Dalı Başkanı

Elif Büşra ATASEVEN tarafından hazırlanan ROBOTİK HÜCRELERDE İKİ KRİTERLİ HAT DENGEME adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Doç. Dr. Hakan GÜLTEKİN
Tez Danışmanı

Tez Jüri Üyeleri

Başkan : Yrd. Doç. Dr. Ayşegül ALTIN KAYHAN _____

Üye : Doç. Dr. Hakan GÜLTEKİN _____

Üye : Yrd. Doç. Dr. Hakkı Özgür ÜNVER _____

TEZ BİLDİRİMİ

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

Elif Büşra ATASEVEN

Üniversitesi : TOBB Ekonomi ve Teknoloji Üniversitesi
Enstitüsü : Fen Bilimleri
Anabilim Dalı : Endüstri Mühendisliği
Tez Danışmanı : Doç. Dr. Hakan GÜLTEKİN
Tez Türü ve Tarihi : Yüksek Lisans – Aralık 2014

Elif Büşra ATASEVEN

ROBOTİK HÜCRELERDE İKİ KRİTERLİ HAT DENGELEME

ÖZET

Bu çalışmada, çeşitli sayıda makine ve bir adet robotun bulunduğu bir üretim hattı ele alınmıştır. Robot parça üzerinde herhangi bir işlem yapmamakta sadece makineleri yükleme/boşaltma ve malzeme taşıma işlemlerini gerçekleştirilmektedir. Hatta tek tip parça üretimi yapılmaktadır. Bir parçanın üretilmesi için, hatta giren malzemeye makinelerde belirli sayıda işlem yapılmalıdır. Robot bir makineden diğer bir makineye sadece bir adet parça taşır ve tek seferde sadece bir makinede yükleme ve boşaltma işlemi yapar. Ele alınan problemde verilmek istenen kararlar; i- Hatta yer alacak makine sayısının belirlenmesi, ii- Parçaların gerektirdiği her bir işlemin işleneceği makinenin belirlenmesi, iii- Robot hareket sırasının belirlenmesidir. Problem, hem çevrim zamanı en küçüklenmesi ve hem de hatta kullanılan makine sayısının en küçüklenmesi olmak üzere iki kriterli bir eniyileme modeli olarak ele alınmıştır. Çözüm için amaç fonksiyonlarının önceliklendirilmesi yöntemi ve amaçlardan bir tanesinin modelde parametre olarak verilerek Pareto optimal çözümlerin belirlenmesi yöntemleri kullanılmıştır. Her iki yöntem için de matematiksel programlama formülasyonları geliştirilmiştir. Farklı problem parametreleri kullanılarak test çalışmaları yapılmıştır ve elde edilen sonuçlar detaylı olarak analiz edilmiştir.

Anahtar Kelimeler: Robotik hücre, montaj hattı dengeleme, çizelgeleme

University : TOBB University of Economics and Technology
Institute : Institute of Natural and Applied Sciences
Science Programme : Industrial Engineering
Supervisor : Assoc. Prof. Hakan GÜLTEKİN
Degree Awarded and Date : M.Sc. – December 2014

Elif Büşra ATASEVEN

BICRITERIA LINE BALANCING IN ROBOTIC CELLS

ABSTRACT

In this study, we consider a production line that consists of a number of machines and material handling robot. The robot does not perform any operations on the parts but loads/unloads the machines and transports the parts between the machines. The parts to be processed are assumed to be identical. A specific number of operations are to be performed on these parts on the machines. The robot can be transport a single part at a time and can load/unload a single machine at a time. The decisions to be made include i-the determination of the number of the machines on the line, ii-the determination of the assignment of operations to the machines, and iii-the determination of the robot move sequence. This problem is handled as a bicriteria problem that includes the minimization of the cycle time and the minimization of the number of machines used on the line. Two solution approaches are considered: Prioritization of the objectives and determination of the pareto optimal solutions by parameterizing one of the objectives. Mathematical programming formulations are developed for both of these methods. Computational studies are performed using different parameter combinations and the results are analyzed in detail.

Keywords: Robotic cells, assembly line balancing, scheduling

TEŐEKKÜR

Lisans ve yüksek lisans hayatım boyunca hep örnek aldığım, çeřitli projelerimde ve yüksek lisans eğitimde danışmanım olan, sabrıyla ve eğitimimdeki desteęiyle hep ilerlememi saęlayan hocam Doç. Dr. Hakan Gültekin'e çok teşekkür ederim.

Yüksek lisans derslerimi veren tüm hocalarıma ve tezimi okuyarak çok değerli yorumlarda bulunan Ayşegül Altın Kayhan ve Hakkı Özgür Ünver'e çok teşekkür ederim.

İş hayatı ve yüksek lisans çalışmalarımı beraber yürütürken yanlarında olamadığım ama hep desteklerini hissettiğim anneme, babama, kardeşime ve bu yoğun dönemimde sevgisiyle bana hep destek olan nişanlım Erdem Aksoy'a sonsuz teşekkürler.

Çalışarak yüksek lisans yapan öğrencileri desteklemek amaçlı BİDEB-2211 numaralı bursu veren TÜBİTAK'a teşekkürü borç bilirim.

İÇİNDEKİLER

ÖZET.....	iii
ABSTRACT.....	iv
TEŞEKKÜR.....	v
İÇİNDEKİLER.....	vi
ŞEKİLLERİN LİSTESİ.....	vii
TABLOLARIN LİSTESİ.....	viii
1. GİRİŞ.....	1
2. LİTERATÜR TARAMASI.....	5
2.1.Montaj Hattı Dengeleme.....	5
2.1.1.Basit Montaj Hattı Dengeleme.....	6
2.1.2.Genel Montaj Hattı Dengeleme.....	9
2.2.Robotik Hücre Çizelgeleme.....	10
2.3.Robotik Hücrelerde Hat Dengeleme.....	12
2.4.İki Kriterli Optimizasyon Problemleri.....	12
2.5.Yöntemin Belirlenmesi.....	Hata! Yer işareti tanımlanmamış.
3. PROBLEM TANIMI ve MATEMATİKSEL MODELLER.....	14
3.1.Problem Tanımı.....	14
3.2.Matematiksel Modeller.....	19
3.2.1.Önceliklendirilmiş Çözüm Yöntemi.....	19
3.2.2.Pareto Optimal Çözümlerin Belirlenmesi Yöntemi.....	25

4. TEST SONUÇLARI	28
4.1.Önceliklendirilmiş Çözüm Yöntemi	29
4.1.1.Toplam Makine Sayısı Önceliklendirilmiş Çözüm Yöntemi.....	31
4.1.2.Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemi	34
4.2.Pareto Optimal Çözümlerin Belirlenmesi	37
4.2.1.Çevrim Zamanını Parametrize Ederek Pareto Optimal Çözümleri Bulma Yöntemi	38
4.2.2. Toplam Makine Sayısını Parametrize Ederek Pareto Optimal Çözümleri Bulma Yöntemi	41
5. ÇÖZÜM YÖNTEMLERİNİN KARŞILAŞTIRILMASI	44
6. SONUÇ ve DEĞERLENDİRME	52
KAYNAKÇA	54
EKLER	57
A.Test Sonuçları	57
B.Verit Dosyası Örneđi – Veri-8	99
ÖZGEÇMİŞ	102

ŞEKİLLERİN LİSTESİ

Şekil 1.1: m Makineli Hücrede Robot Hareketi.....	2
Şekil 4.1: Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler Grafiği – Veri-11.....	40
Şekil 4.2: Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler Grafiği – Veri-11.....	42

TABLULARIN LİSTESİ

Tablo 2.1 Basit Montaj Hattı Dengeleme Probleminin Versiyonları.....	7
Tablo 4.1 Veri Dosyası - İşlem Sayısı Tablosu	299
Tablo 4.2 Veri-11 için Toplam Makine Sayısı Önceliklendirilmiş Çözüm Yöntemi Sonuçları.....	32
Tablo 4.3 Veri-11'in Farklı α Değerleri için Toplam Makine Sayısı Önceliklendirilmiş Çözüm Yöntemi Sonuçları.....	33
Tablo 4.4 Veri-11 için Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemi Sonuçları.....	35
Tablo 4.5 Veri-10 için Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemi Sonuçları.....	36
Tablo 4.6 Veri-11'in Farklı α Değerleri için Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemi Sonuçları	36
Tablo 5.1 Önceliklendirilmiş Çözüm Yönteminin CPLEX OPL Ortalama Çözüm Süreleri.....	46
Tablo 5.2 Pareto Optimal Çözümlerin Belirlenmesi Yönteminin CPLEX OPL Çözüm Süreleri.....	47
Tablo A.1 Toplam Makine Sayısı Önceliklendirilmiş Çözüm Yöntemin Ayrıntılı Sonuçları.....	57
Tablo A.2 Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemin Ayrıntılı Sonuçları.....	61
Tablo A.3 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-1.....	65
Tablo A.4 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-2.....	66
Tablo A.5 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-3.....	67
Tablo A.6 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-4.....	68

Tablo A.7 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-5.....	69
Tablo A.8 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-6.....	70
Tablo A.9 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-7.....	71
Tablo A.10 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-8.....	72
Tablo A.11 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-9.....	73
Tablo A.12 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-10.....	74
Tablo A.13 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-11.....	75
Tablo A.14 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-12.....	76
Tablo A.15 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-13.....	77
Tablo A.16 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-14.....	78
Tablo A.17 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-15.....	79
Tablo A.18 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-16.....	80
Tablo A.19 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-17.....	81
Tablo A.20 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-18.....	82
Tablo A.21 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-19.....	83
Tablo A.22 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-20.....	84

Tablo A.23 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-21.....	85
Tablo A.24 Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-22.....	86
Tablo A.25 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-1.....	87
Tablo A.26 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-2.....	87
Tablo A.27 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-3.....	88
Tablo A.28 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-4.....	88
Tablo A.29 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-5.....	89
Tablo A.30 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-6.....	89
Tablo A.31 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-7.....	90
Tablo A.32 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-8.....	91
Tablo A.33 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-9.....	91
Tablo A.34 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-10.....	92
Tablo A.35 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-11.....	92
Tablo A.36 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-12.....	93
Tablo A.37 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-13.....	93

Tablo A.38 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-14.....	94
Tablo A.39 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-15.....	94
Tablo A.40 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-16.....	95
Tablo A.41 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-17.....	96
Tablo A.42 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-18.....	96
Tablo A.43 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-19.....	97
Tablo A.44 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-20.....	97
Tablo A.45 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-21.....	98
Tablo A.46 Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-22.....	98

1. GİRİŞ

Henry Ford 1900'lü yılların başında seri montaj hattını geliştirerek araba üretim hızını artırmıştır. Bir işçinin üretim sürecindeki tüm işleri yapma görevini değiştirerek, bir işçiyi bir işte uzmanlaştırarak hızlı üretimi sağlamıştır. 1900'lü yılların başındaki bu gelişmelerin ardından montaj hattında üretim hızı ve verimlilik artışı için çeşitli çalışmalar yapılarak sürekli iyileştirme dönemine girilmiştir. İş istasyonu düzeni geliştirilmiş, kalite ve verimliliğin artırılması için her iş istasyonuna dengeli iş dağılımı yapılması gerekliliği fark edilmiştir. Bu sebeple montaj hattının dengelenmesi, verimliliğin artışı için en önemli çalışma alanlarından biri haline gelmiştir.

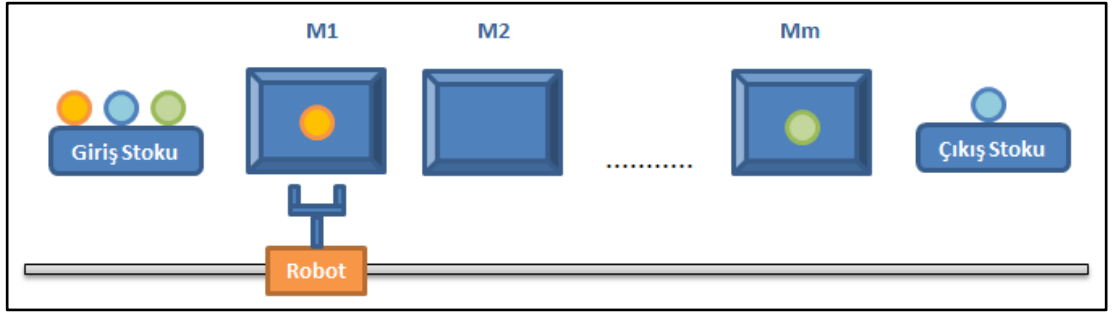
Gelişen teknoloji ile beraber montaj hatlarında makine yükleme boşaltma ve malzeme taşıma işlerinde kullanılan işçilerin yerini robotlar almaya başlamıştır. İş güvenliği, verimlilik artışı ve maliyetlerin düşürülmesi için işçiler başka alanlara kaydırılmıştır. Robotların makineleri yükleme boşaltma ve malzeme taşıma döngüsü, üretim hattının hızını etkilediği için robot hareketlerinin çizelgelenmesi ayrı bir çalışma konusu olmuştur.

Günümüze kadar montaj hattı dengeleme ve robot çizelgeleme problemleri ayrı ayrı ele alınmıştır. Ancak üretim hattının verimliliğini etkileyen bu iki problemin beraber ele alınması ve en iyi çözümün elde edilebilmesi gerekmektedir. Bu gereklilikten yola çıkarak bu çalışma hazırlanmış ve iki problem birlikte çözdürülerek en iyi sonuçların elde edilebilmesi için çeşitli çözüm yöntemleri geliştirilmiştir. Ele alınan çalışma ile literatürde bu alandaki boşluk doldurulmuştur.

Gerçek hayatta yeni kurulan veya iyileştirme yapılan bir montaj hattında, belirlenen amaçlara uygun en az makine sayısı ve çevrim zamanının belirlenmesi üretim verimliliği, maliyetin düşürülmesi ve kalitenin artırılması için çok önemlidir. Gelişen teknoloji ile beraber işçiler yerine robotların kullanılması piyasa rekabet koşullarına uygun çalışılabilmesi için hatta değer katan bir faaliyettir. Bu sebeple robot kullanılan montaj hattının dengelenmesi probleminin çözümü literatürdeki boşluğu

doldurmakla beraber gerçek hayat problemlerinin çözümlerine iyileştirme yapılabilmesi açısından önem kazanmaktadır.

Bu çalışmada tek tip parça üretimi yapılan çeşitli sayıda makine kullanılan bir montaj hattı ele alınmaktadır. Makineler hatta seri olarak dizilmiştir ve makineler arası parça taşıma ve makinelerin yükleme boşaltma işlemlerini yapan tek tutuculu bir robot kullanılmaktadır. Robotun hatta yerleşimi ve hareket düzeni Şekil 1.1’de gösterilmiştir.



Şekil 1.1. m-Makineli Montaj Hattında Robot Hareketi

Montaj hattında kullanılan robot parça üzerinde hiç bir işlem yapmamaktadır. Üretilmesi gereken tek tip parçaların makinelerde işlenmek üzere belirli sayıda işlemi bulunmaktadır. Üretim hattında kullanılan makinelerin CNC makineleri olduğu varsayılmıştır. Bunun bir sonucu olarak da makineler parça üzerinde yapılması gereken işlemlerin hepsini yapabilmektedir, karar verilmesi gereken hangi işin hangi makinede yapılacağıdır.

Ele alınan hattın verimli olarak çalışabilmesi için montaj hattı dengeleme problemlerinde olduğu gibi bu işlemlerin hangi makinelerde işleneceğinin belirlenmesi gerekmektedir. Problem varsayımları basit montaj hattı dengeleme probleminin varsayımları ile aynı olup, malzeme taşıma ve yükleme / boşaltma işlemlerinde robot kullanıldığı için genel montaj hattı dengeleme problemi kapsamına girmektedir.

Scholl ve Becker'in 2004 yılında hazırladıkları literatür tarama makalesinde [27] belirttikleri gibi basit montaj hattı dengeleme probleminin varsayımları aşağıdaki gibidir:

- Hatta tek tip parça üretimi yapılmaktadır.
- Parça üretimi için yapılması gereken işlemlerin işlem süreleri deterministiktir.
- İşlemler arasında öncelik ilişkileri mevcuttur.
- Her işlem sadece bir istasyona atanabilir. Bir istasyona birden fazla işlem atanabilir.

Robotlar sistemlerine yüklenen yazılım kodu ile çalışmaktadır. Yani bir üretim döngüsünde robotun yapması gereken işlemler kodlanır, robot koda göre çalışır ardından tekrar kod başa döner ve robot aynı işlemleri yapmaya başlar. Dolayısıyla robot hareketlerinin çizelgelenmesi bir döngüsel çizelgeleme problemidir.

Bir robot döngüsünde hattan çıkan parça sayısına göre döngüler adlandırılır. Bir döngüde n adet bitmiş parça elde ediliyorsa buna n -birim döngüsü denir [30]. Literatürde yapılan çalışmalar sonucu en iyi çözümlerin 1-birim döngülerde alındığı görülmüştür [9, 10, 11]. Bu çalışmada da 1-birim döngüleri ele alınmıştır. 1-birim döngüsünde, robotun başlangıç durumundan başlayıp tekrar başlangıç durumuna dönene kadar geçen süre bir parçanın üretim süresine eşit olmaktadır. Ardışık iki parça üretimi arasında geçen süreye çevrim zamanı denilmektedir. Ele alınan sistem için çevrim zamanı, 1-birim döngüyü tamamlamak için gereken toplam zamana eşittir. Üretim hızının artırılması için çevrim zamanının en küçüklenmesi gerekmektedir.

Montaj hattı dengeleme problemi NP-zor problemidir [5]. Bu çalışmada hem montaj hattı dengeleme problemi hem de robot hareketlerinin çizelgelenmesi problemi beraber ele alınarak modellenmiştir ve oluşturulan modellerden kısa sürede optimal çözümler elde edilmiştir.

Bölüm 2’de montaj hattı dengeleme ve robotik hücre çizelgeleme alanında yapılan çalışmalar taranmıştır. Bölüm 3’te ele alınan problem detaylı bir şekilde tanımlanmış ve problemin çözümü için geliştirilen matematiksel modeller anlatılmıştır. Bölüm 4’te yapılan test çalışmalarının sonuçları gösterilmiştir. Bölüm 5’te ise bu çalışmadan çıkarılan sonuçlar ve gelecek çalışmalar hakkında yorumlar yapılmıştır.

2. LİTERATÜR TARAMASI

1900'lü yılların başında Henry Ford tarafından kullanılmaya başlanılan montaj hatları, yıllar geçtikçe üretimde kalite ve verimliliğin artırılması ihtiyacı ve gelişen teknoloji ile sürekli iyileştirmeye çalışılmıştır. Bu sebeple ortaya çıkan montaj hattı dengeleme problemi ilk olarak 1950'li yıllarda çalışılmaya başlanmıştır. Önce basit montaj hattı ardından genel montaj hattı dengeleme problemlerinin çözümleri için matematiksel ve sezgisel yöntemler geliştirilmiştir. Bu problem üzerinde çok uzun yıllar çalışıldığı için çok geniş bir literatür vardır.

Bu çalışma kapsamında ele alınan montaj hattında malzeme taşıyıcı endüstriyel robotlar kullanılmıştır. Teknolojinin gelişmesiyle üretimde işçiler yerine robotların kullanılması, robot hareketleri çizelgelemesi problemini ortaya çıkarmıştır. Robotik hücre çizelgeleme problemi ile ilgili çalışmalar 1992 yılında başlamıştır [30].

Bu bölümde öncelikle montaj hattı dengeleme problemi için yapılan çalışmalar anlatılacaktır. Ardından robotik hücre çizelgeleme ve iki kriterli optimizasyon literatürü incelenecektir.

2.1. Montaj Hattı Dengeleme

Henry Ford, 14 Ocak 1914'te "Model T" adı verilen, siyah, kutu gibi, kullanılması ve onarımı kolay araba modelini montaj bandı seri olarak üretmeye başladı. Her bir işçi montajın sadece bir alanında çalışmaya başladı; üretim süresi büyük bir azalma gösterdi. Otomobil fiyatları yarı yarıya düştü. Montaj hattında kalite ve verimliliği artıran bu gelişme, günümüzde hala önemini korumaktadır. Seri üretim hatları Henry Ford zamanından itibaren sürekli geliştirilmiş, akış tipi üretim hattından paralel istasyonlu hatlara, müşteri isteklerine göre özelleştirilmiş karma ve çoklu hatlara, U-tipi yerleşimli hatlara dönüşmüştür. Üretim hatlarındaki bu değişiklikler beraberinde toplam iş yükünün hat boyunca tüm istasyonlara dengeli bir şekilde dağıtılması problemini getirmiştir. Bu probleme hat dengeleme problemi denilmektedir [5].

Literatürde bazı kısıtlayıcı varsayımlar ile tanımlanan basit montaj hattı ve günümüz ihtiyaçları için geliştirilen genel montaj hatları için yapılan çalışmalardan oluşmaktadır. Basit montaj hatları için Scholl ve Becker 2004 yılında bir literatür taraması yayınlamışlardır [27]. Genel montaj hattı dengeleme problemi içinse 2006 yılında Becker ve Scholl literatür tarama makalesi yayınlamışlardır [5]. Bu iki literatür tarama makalesi montaj hattı dengeleme problemleri için yapılan çalışmaları sınıflandırarak sunmaktadır.

2.1.1. Basit Montaj Hattı Dengeleme

Basit montaj hattı, akış tipi üretimin gerçekleştirildiği ve tek tip parça üretiminin yapıldığı hatlardır. Basit montaj hattı, montaj hatlarının en yalın hali olup araştırmacıların yarım yüzyıldan fazla zamandır ilgisini çekmiştir. Basit montaj hattı probleminin temel varsayımları aşağıdaki gibidir:

- Montaj hattında tek tip parçanın seri üretimi yapılır.
- Parça üzerinde yapılması gereken işlemlerin işlem süreleri deterministiktir.
- İşlemler arasında öncelik ilişkileri mevcuttur.
- Hatta kullanılan makineler seri olarak dizilmiştir.
- Tüm makineler eşit üretim kabiliyetine sahiptir.
- Her işlem hatta kullanılan istasyonlardan sadece birine atanır.
- Bir istasyona birden fazla işlem atanabilir [27].

Tablo 2.1’de özetlendiği gibi basit montaj hattı dengeleme probleminde makine sayısı ve çevrim zamanının verilmesine bağlı olarak problem 4 tipe ayrılmıştır. Tip-1 basit montaj hattı dengeleme probleminde çevrim zamanı verilmişken makine sayısı minimize edilir. Tip-2’de ise makine sayısı verilmiştir, çevrim zamanı en küçüklenir. Hem makine sayısının hem de çevrim zamanının en küçüklenmeye çalışıldığı problem Tip-E olarak geçmektedir. Bu çalışmada Tip-E problem ele alınmıştır. Basit montaj hattı dengeleme problemi için ele alınan son tip ise Tip-F’dir. Bu problemde

parametre olarak verilmiş makine sayısı ve çevrim zamanı için probleme olurlu çözüm elde edilip edilemeyeceği incelenir.

Tablo 2.1. Basit Montaj Hattı Dengeleme Probleminin Versiyonları [27]

Makine Sayısı	Çevrim Zamanı	
	Verilmiş	En Küçükle
Verilmiş	Tip-F	Tip-2
En Küçükle	Tip-1	Tip-E

Basit montaj hattı dengeleme problemi için matematiksel model kullanılarak çözüm yöntemleri elde edilmiştir. Scholl ve Klein 1997 yılında yaptıkları çalışmada [29] basit montaj hattı dengeleme problemi için yeni bir dal-sınır yöntemi geliştirmişlerdir ve bu yönteme SALOME adını vermişlerdir. Daha önce yapılmış çalışmalardan daha iyi sonuçlar elde etmişlerdir.

Scholl vd.'nin 2010 yılında yaptıkları çalışmada [28], daha önce geliştirdikleri SALOME [29] adlı algoritmaya atama kısıtları eklemişler ve bu yeni algoritmaya ABSALOM adını vermişlerdir. ABSALOM da bir dal-sınır yöntemidir ve problem için kesin çözüm aramaktadır. Bu problemin test çalışmasında kullanılan veri kümeleri bu çalışmada da kullanılacaktır.

Peeters ve Degraeve 2006 yılında yaptıkları çalışmada [25] basit hat dengeleme problemi için temel kısıtlar kullanılarak oluşturulan matematiksel modelden yola çıkarak Dantzig–Wolfe ayrışımını temel alan doğrusal programlama gevşetmesi ile yeni bir alt sınır bulmuşlardır.

Pastor ve Ferrer 2009 yılında yaptıkları çalışmada [24] Tip-1 ve Tip-2 basit montaj hattı dengeleme problemi için işlemlerin makinelere atanabilirlik sayısını, makine sayısı ve çevrim zamanından bağımsız olmasını sağlayan yeni kısıtlar tanımlamışlardır.

Gürsoy ve Nuriyev 2010 yılında sundukları çalışmada [18] basit montaj hattı dengeleme problemi için yalın üretim mantığını kullanarak matematiksel model geliştirmişlerdir.

Ritt ve Costa ise 2011 yılında yaptıkları çalışmada [26] o zamana kadar kullanılan basit montaj hattı dengeleme problemine ait matematiksel modelleri inceleyip çözüm aralığını daraltan kısıtlar bularak daha iyi sonuçlar elde etmişlerdir.

Basit montaj hattı dengeleme problemine sezgisel çözüm yöntemleri de geliştirilmiştir. İlk geliştirilen sezgisel çözüm yöntemlerinden biri Pozisyon Ağırlığı Yöntemidir (Ranked Positional Weight)'dir [19]. Pozisyon Ağırlığı Yönteminde bir işlemin kendi süresi ile ardılı olan işlem süreleri toplamı o işlemin pozisyon ağırlığını verir. Ardından işlemler pozisyon ağırlıklarına göre artandan azalana doğru sıralanır. İlk sıradaki işlemde başlayarak işlemler, çevrim zamanı aşılmayacak şekilde, istasyonlara atanır.

1963 yılında Hoffman tarafından geliştirilen ve kendi adının verildiği Hoffman Algoritmasında [20], öncelik ilişkileri matrisinde sütundaki tüm değerler toplanarak her birine kod numarası verilir. İlk dize seçilir ve bu dizedeki ilk 0 seçilir. Dizedeki bu eleman, o işlemin önceliği olmadığını gösterir. Bu işlem ilk istasyona atandıktan sonra kod numaraları yeniden oluşturulur ve tüm işlemler atanana kadar süreç devam eder.

Arcus'un 1965'de geliştirdiği COMSOAL yöntemi [3] işlemlerin öncelik ilişkileri matrisine göre oluşturulan atanabilir işler arasından rasgele seçim yaparak makinelere sırasıyla atar. Bu işlemi tekrarlayarak işlemler arasından en iyi çözümü ele alır.

Bautista ve Pereira 2009 yılında SALBP-1 problemi üzerinde çalışmışlar ve Sınırlı Dinamik Algoritma adlı sezgisel geliştirmişlerdir [4]. Dinamik Programlama çerçevesinde sezgisel kurallarla çözüm uzayını sınırlamışlardır. Geliştirdikleri

algoritma, literatürdeki diğer çalışmalarla karşılaştırıldığında en iyi sonuçları veren algoritmadır ve çözüm süreleri çok kısadır.

2.1.2. Genel Montaj Hattı Dengeleme

Genel montaj hattı dengeleme probleminde;

- Maliyet ve kar bazlı amaç fonksiyonu,
- Paralel istasyonlar ve işler,
- U-tipi yerleşimli hatlar,
- Atanabilirlik kısıtları,
- Stokastik işlem zamanları

gibi konuları ele alan çok sayıda çalışma vardır.

Ele alınan çalışmaya benzeyen ancak detaya inildiğinde farklı olan literatür robotik hat dengeleme problemi ile ilgilidir. Robotik hat dengeleme probleminde (Robotic Assembly Line Balancing Problem, RALBP), farklı robotlar farklı görevlere atanmıştır ve her robot kapasite ve özelliklerinden dolayı farklı montaj zamanlarına sahiptir [21]. Bu çalışmada ele alınan robotik hücreler ile RALBP problemi kapsamında ele alınan hücreler farklılık göstermektedir. Bu çalışmada robot malzeme taşıma ve makine yükleme boşaltma işlerini yaparken RALBP’da robotlar parça üzerinde işlem yapabilmektedir ve hatta birden fazla robot bulunabilmektedir. Ele alınan problemde hatta tek robot bulunmaktadır. Levitin vd. 2006 yılında RALBP üzerine genetik algoritma geliştirmişlerdir [22].

Ağpak ve Gökçen 2005 yılında yaptıkları çalışmada [1] hatta kullanılacak en küçük istasyon sayısını bulmak için 0-1 tamsayılı model geliştirmişlerdir. Çalıştıkları konu kaynak kısıtlı hat dengeleme problemidir (Resource Constrained Assembly Line Balancing, RCALBP). Bu tip problemlerde sınırlı sayıda çalışan işçi ve makine bulunmaktadır.

Hat dengeleme tasarım problemi de makinelerin yerleşimi üzerine kurulu olan bir problemdir. Atanacak işler birbirinin aynısı olmayan makinelerde üretilebilmektedir. Makine çeşitliliği de bir kısıt olarak modele dahil olur. Nicosia vd. 2002 yılında yaptıkları çalışmada [23] bu problem üzerine dinamik programlama algoritması geliştirmişlerdir.

2.2. Robotik Hücre Çizelgeleme

Teknolojinin getirdiği yeniliklerle beraber üretim hatlarında insan gücü yerine robotlar kullanılmaya başlanmıştır. Robot hareketleri çevrim zamanına etki ettiği için robot hareketlerinin çizelgelenmesi problemi ortaya çıkmıştır.

Tek tutuculu robotlarla çalışmaya başlayan robotik hücreler teknolojinin gelişmesiyle çift tutuculu robotlara, çift kollu robotlara dönüşmüştür. Bazı robotlar parça üzerinde işlem yapabilme yeteneğine sahiptir. Robotların hareket kabiliyeti geliştikçe montaj hattının dengelenmesi problemi daha zorlaşmıştır. Bu konuda Crama vd.'nin 2000 yılında [9] ve Geismar vd.'nin 2005 yılında [14] yaptıkları literatür taramaları robotik hücre çizelgeleme problemi çeşitlerini detaylı olarak analiz etmişlerdir.

Sethi vd.'nin 1992 yılında yaptığı çalışma [30] ile robotik hücre çizelgeleme literatürü başlamıştır. Tek parça üreten, iki veya üç makineli hücreler için uzun dönemde ortalama çıktı sayısını en büyükmeyi amaçlamışlardır. Bu amaç doğrultusunda çevrim zamanı için formüller geliştirmişlerdir. m makineli sistem için $m!$ tane 1-birim döngüsü bulunduğunu, iki makineli sistem için 1-birim döngülerinden birinin optimal olduğunu ispatlamışlardır.

Crama ve Klundert 1997 yılında yaptıkları çalışma [10] ile 3 makineli hücreler için 1-birim döngülerin optimal olduğunu ispatlamışlardır. m -makineli robotik hücreler için en iyi 1-birim döngüsünü veren algoritmayı ise 1999 yılında yaptıkları çalışmada geliştirmişlerdir [11].

Aktürk vd. 2005 yılında yaptıkları çalışmada [2] iki makineli robotik hücre ele almışlardır. Amaç optimal robot hareket döngüsünü ve en kısa çevrim zamanını verecek işlem atamasını bulmaktır. Verilen parametrelerle optimal 1-birim ve 2-birim döngülerini elde etmişlerdir.

Gültekin vd. 2010 yılında yaptıkları çalışma [17] ile literatürde ilk defa iki kriterli, hem işlemlerin makinelere atanması, makinelerin işlem süreleri ve robot hareketleri ile beraber çevrim zamanını hem de toplam üretim maliyetini en küçükleyen amaç fonksiyonunu ele almışlardır.

Brauner ve Finke 1999 ve 2001 yılında yaptıkları çalışmalarda [6, 7] 4 veya daha fazla makineli hücreler için 1-birim döngülerin optimal olmayabileceğini, k-birimlik döngülerin $k \geq 2$ için optimal sonuçlar verebileceğini göstermişlerdir.

Geismar vd.'nin 2005 yılında yaptıkları çalışmada [15] esnek robotik hücreler üzerinde çalışmışlardır. Her parçanın her makineye aynı sırada uğramasındansa makine sayısı ve işlem sıraları üzerinde değişiklik yapmışlardır. Yaptıkları değişikliğin verimlilik üzerindeki etkisini incelemişlerdir. İşlemlerin makinelerde gerçekleşme sırası değiştirilmiş ve her denemede makine sayısı artırılmıştır. 2 makineli robotik hücreler için verimlilikte bir artış olmamıştır. 3 ve 4 makineli robotik hücreler için verimlilik %14 civarında artmıştır.

Dawande vd. 2002 yılında yaptıkları çalışmada [12] ürün çıktı oranının artırılmasını amaçlayan robotik hücre üzerine çalışmışlardır ve 1-birim döngülerini kullanmışlardır. Gerçek hayatta da kullanılan sabit taşıma zamanlı robotları almışlardır yani robotik hücrelerde robotun herhangi iki makine arasındaki taşıma zamanı sabittir. Bu problem için optimal 1-birim döngüsünü bulan bir algoritma geliştirmişlerdir.

2.3. Robotik Hücrelerde Hat Dengeleme

Üstünoğlu 2012 yılında yaptığı çalışmada [31] tek tip parça üreten robotik üretim hattı ele almıştır. Tip-2 problemini robotik hücrelerde çalışmıştır. Matematiksel model kurmuş ardından problem NP-Zor olduğu için sezgisel algoritma geliştirmiştir. Geliştirdiği sezgisel algoritma, Crama ve van de Klundert algoritmasını kullanmaktadır.

2.4. İki Kriterli Optimizasyon Problemleri

Bu çalışmanın amaç fonksiyonu iki kriterli olduğu için iki kriterli optimizasyon literatürü de incelenmiştir. 2005 yılında Hoogeveen [21] çok kriterli çizelgeleme problemleri ile ilgili literatür taraması yayınlamıştır. Örnek iki problem ile iki kriterli çizelgeleme problemlerini tanıtmış ardından ilgili literatürü detaylı olarak incelemiştir.

Feng vd. 2014 yılında yayınlanan çalışmalarında [13] robotik hücreler için iki kriterli çizelgeleme üzerine çalışmışlardır. Amaç fonksiyonu çevrim zamanını ve toplam robot hareket süresini en küçükmektir. Ele aldıkları problem için karışık tamsayılı programlama modeli geliştirmişlerdir. Çözümlerinde Pareto kümelerini bulmuşlardır.

Gültekin vd. 2008 yılında yaptıkları çalışmada [16] tek tip parça üreten 2 ve 3 makineli robotik hücrelerde robot hareket sırasını ve en düşük üretim maliyetini bulmayı amaçlayan iki kriterli optimizasyon problemi üzerine çalışmışlardır. Bu çalışma robotik hücre çizelgeleme literatüründe maliyeti en küçükmeyi amaçlayan ilk yayındır.

Cheng vd. 1998 yılında [8] tek makinede iki kriterli çizelgeleme problemi üzerine çalışmışlardır. Amaçları iş sıralama ve kaynak ataması yapılmasıdır. Çözüm toplam ağırlıklandırılmış kaynak kullanımını ve iş tamamlanma zamanları kriterlerine göre değerlendirilmiştir ve amaç her iki kriterin de enküçülenmesidir. Çözüm için geliştirilen algoritmalar Pareto kümelerini yapılandırmıştır.

2.5. Yöntemin Belirlenmesi

Robotik hücrelerde iki kriterli hat dengeleme probleminin modellenmesi ve çözüm yöntemleri için detaylı bir literatür taraması yapılmıştır. Literatür taraması başlığı altında ele alınan probleme benzerlik gösteren çalışmalar anlatılmıştır.

Montaj hattı dengeleme problemi çok uzun yıllar çalışılmış ve iyileştirmeler yapılmış bir problemdir. Robotik hat dengeleme problemi ise gelişen teknoloji ile beraber montaj hatlarında robotların kullanılması ile başlamıştır. Her iki problem tipi için çeşitli çalışmalar yapılmıştır. Bu çalışmada bu iki problemi beraber ele alan tek bir matematiksel model kurulmuştur. Çözüm yönteminin geliştirilmesi aşamasında iki kriterli optimizasyon modellerinin çözüm yöntemleri de incelenmiş ve bu çalışmanın çözüm yöntemlerinden olan Pareto optimal çözümlerin belirlenmesi yönteminin kullanılmasına karar verilmiştir.

Literatür genel olarak incelendiğinde problem üretimde esneklik kapsamında robotik hat dengeleme problemine benzese bile, daha önce robotların parça üzerinde işlem yapmadığı ve hem çevrim zamanını hem de makine sayısını en küçükmeyi amaçlayan bir problem tanımı ile çalışılmamıştır. Bu iki farklı problem birlikte düşünüldüğünde sistemde düzenli bir malzeme akışının sağlanacağı, insan gücü ve tezgah kapasitelerinin daha verimli kullanılacağı düşünülmektedir.

3. PROBLEM TANIMI ve MATEMATİKSEL MODELLER

Bu bölümde robotik hücrelerde iki kriterli hat dengeleme problemi detaylı bir şekilde anlatılmıştır. Problem parametrelerinden ve karar değişkenlerinden bahsedilmiş ardından geliştirilen matematiksel modeller sunulmuştur.

3.1. Problem Tanımı

Bu çalışmada seri dizilmiş çeşitli sayıda makineden oluşan ve tek tip parça üretimi yapan bir montaj hattı ele alınmıştır. Hatta parça taşıma ve makine yükleme boşaltma işlemlerini tek tutuculu bir robot yapmaktadır. Robot doğrusal ray üzerinde hareket etmektedir. Parça üzerinde tüm işlemleri makineler yapmakta, robot parça üzerinde hiçbir işlem yapmamaktadır.

Hatta giren bir parçanın tamamlanarak çıkabilmesi için toplam K adet işlem başarıyla tamamlanmalıdır. Bu çalışmada k işinin işlem süresi o_k , $k = 1, \dots, K$ ile gösterilmektedir.

Montaj hattı dengeleme problemlerinde işlemler arasında öncelik ilişkileri vardır ve ele alınan problemde de yapılan varsayımlardan biridir. Bir işlemin yapılabilmesi için, öncesinde başka bir işlemin tamamlanmış olması gerekebilir. Öncelik ilişkileri b_{ij} parametresi ile aşağıdaki şekilde tanımlanmıştır:

$$b_{ij} = \begin{cases} 1, & i \text{ işlemi } j \text{ işleminin öncülü ise} \\ 0, & \text{diğer durumlarda} \end{cases}, \forall i, j \in \{1, \dots, K\}, i \neq j$$

Montaj hattında kullanılan robot, makineler ve giriş çıkış stokları arasında parça taşıma işlemini yapmaktadır. Robotun ardışık iki makine arasındaki hareket süresi δ ile gösterilmektedir. Robotun ardışık olmayan i makinesinden j makinesine gidiş süresi $|i - j|\delta$ kadardır. Hatta işlem görecekt parçaların bekletildiği giriş stoku 0., bitmiş parçaların bırakıldığı çıkış stoku ise $m + 1$. makine olarak gösterilmektedir.

Robot aynı zamanda makine yükleme boşaltma işlemlerini de yapmaktadır. Robot bir makineyi boşaltmak için makinenin önüne geldiğinde, parça üzerindeki işlem bitmişse hemen boşaltır, değilse işlem tamamlanana kadar bekler ve aldığı parçayı bir sonraki makineye taşır, o makineyi yükler. Makine yükleme veya boşaltma esnasında geçen süre ε ile gösterilmektedir.

Bu çalışmada ele alınan robot hareket döngüsü Dawande vd.'nin 2002 yılında yaptıkları çalışmada [12] optimal çözümler verdiğini ispatladığı “döngüsel robot hareket çizelgesi”dir. Robot belirli bir sistem durumundan başlar, yapması gereken aktiviteleri gerçekleştirir ve başlangıç durumuna geri döner. Bu döngü sürekli olarak tekrarlanır.

Crama ve van de Klundert 1997 yılında robot aktivitelerini tanımlamıştır [10]. Aktivite A_i robotun i . makineden parçayı aldığı, $(i+1)$. makineye götürdüğü ve $(i+1)$. makineye parçayı yüklediği hareketler bütünüdür. Bütün robot hareketleri bu şekilde tanımlanan aktivitelerle gösterilebilir. Hattan bir parçanın bitmiş ürün olarak çıkması için yapılan tüm döngüsel hareketler 1-birim döngüsü olarak adlandırılır. 1-birim döngülerinde bütün A_i aktiviteleri sadece bir kez tekrarlanır. Örneğin üç makineli bir montaj hattında robot tüm hareket döngülerinde giriş stokundan parça almaya başlasın. Sethi vd. m makineli sistemde $m!$ tane 1-birim döngüsü olduğunu ispatlamışlardır [30]. Bu durumda ele alınan 3 makineli örnek hatta 6 adet 1-birim döngüsü bulunmaktadır. Robot hareket döngüleri aşağıdaki şekilde gösterilir:

$$S_1 = A_0A_1A_2A_3$$

$$S_2 = A_0A_1A_3A_2$$

$$S_3 = A_0A_2A_1A_3$$

$$S_4 = A_0A_2A_3A_1$$

$$S_5 = A_0A_3A_1A_2$$

$$S_6 = A_0A_3A_2A_1$$

Yukarıda verilen robot hareket döngülerinden S_4 döngüsünde robot giriş stokundan bir parça alır, 1. makineye gider ve yükler. Sonra 2. Makineye gider, 2. Makinede işlemi bitmiş parçayı alır 3. Makineye gider ve 3. Makineyi yükler. 3. Makinede işlemin bitmesini bekler, 3. Makineyi boşaltır ve çıkış stokuna giderek parçayı çıkış stokuna bırakır. Ardından tekrar 1. Makineye gider işlemi bitmiş parçayı alır ve 2. Makineye yükler. Son olarak tekrar giriş stokunun önüne gider. Bu şekilde robot hareket döngüsü tamamlanmış olur.

Yukarıdaki örnekte anlatıldığı gibi 1-birim robot hareket döngüsü tamamlandığında giriş stokundan bir parça alınmış ve çıkış stokuna bir parça bırakılmıştır. Yani bir parça üretimi tamamlanmıştır. 1-birim döngüsünün süresi bir parçanın üretim süresine yani çevrim süresine eşittir. Bu çalışmada çevrim zamanı C notasyonu ile gösterilmektedir. Bir döngünün çevrim zamanı hesaplamasının daha iyi anlaşılması için 3 makineli montaj hattında yukarıda verilen örnek döngülerden $S_4 = A_0A_2A_3A_1$ döngüsünü ele alalım. S_4 döngüsünde robot önce giriş stokundan bir parça alır (ϵ), M_1 makinesine gider (δ) ve makineyi yükler (ϵ). Parça makinede işlenmeye başlarken robot makine önünde beklemeden M_2 makinesine gider (δ). Dolu olan M_2 makinesinde parça işlemi bitmedi ise robot makine önünde bekler (w_2), parça işlemi bitince parçayı makineden boşaltır (ϵ), M_3 makinesine gider (δ) ve M_3 makinesini yükler (ϵ). Robot M_3 makinesinin önünde parça işleminin tamamlanmasını bekler (p_3). İşlem bitince robot M_3 makinesini boşaltır (ϵ), çıkış stokuna gider (δ) ve parçayı çıkış stokuna bırakır (ϵ). Ardından M_1 makinesine gider (3δ), dolu olan M_1 makinesinde parça işlemi bitmedi ise robot makine önünde bekler (w_1), parça işlemi bitince parçayı makineden boşaltır (ϵ), M_2 makinesine gider (δ) ve M_2 makinesini yükler (ϵ). Son olarak robot giriş stokuna geri döner (2δ).

Yukarıda anlatılan tüm işlemler S_4 döngüsü için bir çevrimde robotun yaptığı hareketler bütünüdür. Robot makineyi yükledikten sonra aynı makine önünde bekleyebilir. Bu durumda bekleme zamanı o makinenin işlem zamanına eşit olur. Eğer robot bir makineyi yükleyip dolu olan başka bir makinenin önüne gitti ise ve

işlem henüz bitmedi ise bu durumda robotun bekleme zamanı makinenin kalan işlem süresi kadardır. Sonuç olarak bir döngüde oluşan çevrim zamanı aşağıdaki gibidir:

$$\begin{aligned}
C &= 8\varepsilon + 10\delta + p_3 + w_1 + w_2 \\
w_1 &= \max\{0, p_1 - p_3 - 4\varepsilon - 6\delta - w_2\} \\
w_2 &= \max\{0, p_2 - 2\varepsilon - 4\delta\}
\end{aligned}$$

Bu çalışmada anlaşılması ve uygulanması kolay olan 1-birim döngüleri ele alınmıştır. 1-birim döngüleri bir çok durumda optimal çevrim zamanını vermektedir. Optimal olmadıkları durumda bile çok iyi sonuçlar elde edilmektedir.

Ele alınan problem, öncelik ilişkilerini sağlayarak işlemlerin en az sayıda makineye atanmasını aynı zamanda en kısa çevrim zamanını elde etmeyi amaçlamaktadır.

Yukarıda bahsedilen robot aktivitelerinin hangi sıra ile yapılması gerektiğine yani robot hareket çizelgesine karar verilmelidir. Robot aktivite sırasının belirlenebilmesi için her aktivitenin bir pozisyona atanması, tüm aktivite atamasının yapılmasının ardından bir çevrimde robotun hangi sıralamaya göre hareket etmesi gerektiği belirlenmektedir. Bunun için x_{ij} karar değişkeni tanımlanmıştır. x_{ij} değerleri A_i aktivitelerinin sıralamasını vermektedir.

$$x_{ij} = \begin{cases} 1, & i \text{ aktivitesi } j \text{ pozisyonuna} \\ & \text{atandı ise} \\ 0, & \text{diğer durumlarda} \end{cases}, \quad \forall i \in \{0, \dots, m\}, j \in \{1, \dots, m + 1\}$$

Model çıktularından biri de A_i aktivitelerinin başlangıç zamanlarıdır. Sıralaması belirlenen i aktivitesinin başlangıç anı T_i notasyonu ile gösterilmektedir. T_i modelin karar değişkenlerinden biridir.

Parçanın hattan tamamlanarak çıkabilmesi için K adet işlemin hepsinin tamamlanması gerekmektedir. Montaj hattında yapılması gereken her bir k işinin hangi makineye atandığı kararı için y_{ik} değişkeni tanımlanmıştır.

$$y_{ik} = \begin{cases} 1, & k \text{ işi } i \text{ makinesine atandı ise} \\ 0, & \text{diğer durumlarda} \end{cases}, \forall i \in \{1, \dots, m\}, k \in \{1, \dots, K\}$$

Bu çalışmada makine sayısı ve çevrim zamanı minimizasyonu birlikte yapılmaktadır. Makine sayısı minimizasyonu problemi için, makine sayısı üzerinde bir üst limit olduğu varsayılarak bunlardan kaç tanesinin kullanılacağına karar verilecektir. Hatta hangi makinelerin kullanılacağını belirlenmesi için z_i karar değişkeni tanımlanmıştır.

$$z_i = \begin{cases} 1, & i \text{ makinesi kullanılıyorsa} \\ 0, & \text{diğer durumlarda} \end{cases}, \forall i \in \{1, \dots, m\}$$

Makinelere işlem atamaları yapılırken, makine işlem zamanlarının çevrim zamanını aşmaması gerekmektedir. Makine işlem zamanı o makineye atanan işlemlerin işlem zamanları toplamına eşittir. Makine işlem zamanları p_i karar değişkeni ile gösterilmektedir.

$$p_i = \sum_{k=1}^K y_{ik} o_k, \forall i \in \{1, \dots, m\}$$

Robotlu hücrelerde montaj hattı dengeleme probleminin çözümü için yukarıda açıklanan parametre ve karar değişkenleri ile problem iki kriterli ele alınmaktadır. Bu çalışmada, iki kriterli problemleri çözmek için uygulanabilecek çözüm yöntemlerinden aşağıdaki ikisi ele alınmıştır:

1. Önceliklendirilmiş Çözüm Yöntemi,
2. Pareto Optimal Çözümlerin Belirlenmesi Yöntemi.

Yukarıda verilen iki matematiksel çözüm yöntemi Bölüm 3.2'de anlatılmaktadır.

3.2. Matematiksel Modeller

Robotik hücrelerde iki kriterli hat dengeleme probleminin temel olarak iki amaç fonksiyonu vardır; toplam makine sayısının ve çevrim zamanının en küçükleme. İki kriterli amaç fonksiyonunun çözümü için iki çeşit çözüm yöntemi kullanılmıştır. Kullanılan çözümler bu başlık altında anlatılmış, elde edilen sonuçlardan 4. Bölüm'de bahsedilmiştir.

3.2.1. Önceliklendirilmiş Çözüm Yöntemi

Önceliklendirilmiş çözüm yönteminde, amaç fonksiyonunda ele alınan iki kriterden daha önemli görülen kriter belirlenen yüksek bir katsayı ile çarpılarak öncelikli olarak en küçüklene çalışılır. Bu sayede önceliklendirilmiş kriter ilk olarak minimize edilirken ikincil kriter de ilk kriterin aldığı değerlere göre en küçüklene olur. Önceliklendirilmiş çözüm yönteminde önceliklendirme katsayısı α ile gösterilmiştir.

Ele alınan problemin amaç fonksiyonu hem makine sayısını hem de çevrim zamanını minimize etmektedir. Bu iki kriter çeşitli durumlarda öncelikli olabilmektedir. Örneğin montaj hattına ayrılan alanın darlığından dolayı, hatta yer alabilecek makine sayısı kısıtlıysa, bir makine daha fazla eklemek yeniden tesis tasarımı gibi sorunlara sebep olacaksa veya makineler çok maliyetli ise makine sayısının en küçüklene kriteri önceliklendirilebilir. Bununla beraber üretilen parça sayısının artırılması, daha fazla üretim gerçekleştirilmesi politikasına sahip bir şirketin çevrim zamanını azaltmak için bir makine daha fazla çalıştırmayı kabul ettiği durumlar vardır. Hala en az makineyi kullanmak isteyen şirket öncelikli olarak çevrim zamanını azaltmayı hedeflediğinde öncelikli kriter olarak çevrim zamanı ele alınır.

Verilen örneklerden yola çıkılarak, geliştirilen çözüm yönteminin gerçek hayatta kullanılabilmesi için her iki duruma da optimal çözüm verebilecek şekilde matematiksel modeller hazırlanmıştır.

3.2.1.1. Toplam makine sayısı önceliklendirilmiş çözüm yöntemi

Toplam makine sayısı önceliklendirilmiş çözüm yönteminde amaç öncelikli olarak hatta kullanılan toplam makine sayısını minimize etmek, ikincil olarak minimum makine sayısı içinde minimum çevrim zamanını elde etmektir. Hedeflenen değerleri elde edebilmek için amaç fonksiyonunda toplam makine sayısı belirlenmiş bir katsayı ile çarpılır. Bu şekilde amaç öncelikli olarak en küçük toplam makine sayısını bulmak olur. Ardından amaç fonksiyonunda yer alan çevrim zamanı en küçüklenir. Hatta kullanılan toplam makine sayısı önceliklendirilirken katsayısı sonsuz bir değer olarak belirlenmemektedir çünkü amaç her iki kriterin de en küçüklenmesidir. Eğer makine sayısının katsayısı sonsuz bir değer olursa çevrim zamanı en küçüklenmemiş olur.

Toplam makine sayısı önceliklendirilmiş çözüm yöntemi için geliştirilen matematiksel model aşağıdaki gibidir:

$$\min \quad \left(\alpha \sum_{i=0}^m z_i \right) + C \quad (3.1)$$

$$\text{st.} \quad \sum_{j=1}^{m+1} x_{ij} \leq 1, \quad \forall i \in \{0, \dots, m\} \quad (3.2)$$

$$\sum_{i=0}^m x_{ij} \leq 1, \quad \forall j \in \{1, \dots, m+1\} \quad (3.3)$$

$$p_i = \sum_{k=1}^K y_{ik} o_k, \quad \forall i \in \{1, \dots, m\} \quad (3.4)$$

$$p_i \leq C, \quad \forall i \in \{1, \dots, m\} \quad (3.5)$$

$$T_i \geq T_{i-1} + 2\varepsilon + \delta + p_i - M \left(1 - \left(x_{(i-1)j} - \sum_{k=1}^{j-1} x_{ik} \right) \right) \\ - M \left(1 - \sum_{k=j}^{m+1} x_{ik} \right) \forall i \in \{1, \dots, m\}, j \in \{1, \dots, m+1\} \quad (3.6)$$

$$T_i + C \geq T_{i-1} + 2\varepsilon + \delta + p_i - M \left(1 - \left(x_{(i-1)j} - \sum_{k=j+1}^{m+1} x_{ik} \right) \right) \\ - M \left(1 - \sum_{k=1}^j x_{ik} \right), \forall i \in \{1, \dots, m\}, j \in \{1, \dots, m+1\} \quad (3.7)$$

$$T_i \geq T_k + 2\varepsilon + p_i + \delta|k+1-i| - M \left(2 - (x_{ij} + x_{k(j-1)}) \right), \\ \forall i, k \in \{0, \dots, m\}, i \neq k, j \in \{2, \dots, m+1\} \quad (3.8)$$

$$C \geq T_i + 2\varepsilon + \delta + (i+1)\delta, \quad \forall i \in \{0, \dots, m\} \quad (3.9)$$

$$T_i \leq Mz_i, \quad \forall i \in \{0, \dots, m\} \quad (3.10)$$

$$\sum_{i=1}^m y_{ik} = 1, \quad \forall k \in \{1, \dots, K\} \quad (3.11)$$

$$b_{jk} \cdot y_{lk} \leq \sum_{i=1}^l y_{ij}, \quad \forall j, k \in \{1, \dots, K\}, \forall l \in \{1, \dots, m\} \quad (3.12)$$

$$x_{01} = 1 \quad (3.13)$$

$$y_{ik} \leq z_i, \quad \forall i \in \{1, \dots, m\}, \forall k \in \{1, \dots, K\} \quad (3.14)$$

$$z_{i+1} \leq z_i, \quad \forall i \in \{1, \dots, m-1\} \quad (3.15)$$

$$z_i \leq \sum_{j=2}^{m+1} x_{ij}, \quad \forall i \in \{1, \dots, m\} \quad (3.16)$$

$$\sum_{k>i}^{m+1} x_{lk} \leq \sum_{j=1}^{m+1} x_{ij}, \quad \forall i \in \{0, \dots, m\}, i \in \{0, \dots, m\}, i \neq l \quad (3.17)$$

$$x_{ik} \leq z_i, \quad \forall i \in \{1, \dots, m\}, k \in \{1, \dots, K\} \quad (3.18)$$

$$x_{ij} \in \{0, 1\}, \quad \forall i \in \{0, \dots, m\}, j \in \{1, \dots, m+1\} \quad (3.19)$$

$$y_{ik} \in \{0, 1\}, \quad \forall i \in \{0, \dots, m\}, k \in \{1, \dots, K\} \quad (3.20)$$

$$z_i \in \{0, 1\}, \quad \forall i \in \{0, \dots, m\} \quad (3.21)$$

$$T_i \geq 0, \quad \forall i \in \{0, \dots, m\} \quad (3.22)$$

$$p_i \geq 0, \quad \forall i \in \{0, \dots, m\} \quad (3.23)$$

Modelin (3.1) numaralı amaç fonksiyonu öncelikli olarak toplam makine sayısını ikincil olarak çevrim zamanını en küçüklemeyi hedeflemektedir. Toplam makine sayısı α ile çarpılarak önceliklendirilmiştir. α pozitif ve 1'den büyük bir değerdir. (3.2) numaralı kısıt ile her aktivitenin en fazla bir pozisyona atanmasını, (3.3) numaralı kısıt ise her pozisyona en fazla bir aktivitenin atanmasını sağlamaktadır. (3.4) numaralı kısıt ile her makinenin toplam işlem zamanları hesaplanmaktadır. Makinelerin işlem zamanları çevrim zamanını aşamaz bu sebeple modele (3.5) kısıtı eklenmiştir. (3.6) kısıtı aktivite sıralamasında A_{i-1} aktivitesi A_i 'den önce geliyorsa A_i aktivitesinin başlama zamanı, yani i makinesinin boşaltılmaya başlandığı an, i makinesinin doldurulma anından (yani $i-1$ makinesinden parçanın boşaltılmaya başlandığı an (T_{i-1}) ε kadar süre, parçanın i makinesine taşınması δ kadar süre, i makinesinin yüklenmesi ε kadar süre ve i makinesindeki işlem zamanı (p_i) toplamından) daha büyük ya da eşit olması gerektiğini göstermektedir. (3.7) kısıtı ise aktivite sıralamasında A_i aktivitesi A_{i-1} 'den önce geliyorsa i makinesinin boşaltılmaya başlama an (T_i) i makinesinin doldurulma anından (yani $i-1$ makinesinden parçanın boşaltılmaya başlandığı an (T_{i-1}) ε kadar süre, parçanın i makinesine taşınması δ kadar süre, i makinesinin yüklenmesi ε kadar süre ve i makinesindeki işlem zamanı (p_i) toplamından) bir çevrim süresi kadar geridedir. Bu sebeple çevrim süresi C ve T_i 'nin toplamı i makinesinin doldurulma anından daha büyük ya da eşit olmalıdır. (3.8) kısıtı herhangi bir k aktivitesi (A_k) herhangi bir i aktivitesi (A_i) önce geliyorsa A_i aktivitesinin başlama zamanı, yani i makinesinin boşaltılmaya başlandığı an, i makinesinin doldurulma anından (yani k makinesinden parçanın boşaltılmaya başlandığı an (T_k) ε kadar süre, parçanın i makinesine

taşınması $\delta|k + 1 - i|$ kadar süre, i makinesinin yüklenmesi ε kadar süre ve i makinesindeki işlem zamanı (p_i) toplamından) daha büyük ya da eşit olması gerektiğini göstermektedir. (3.9) numaralı kısıt ise çevrim zamanının robotun bütün aktiviteleri tamamlayıp giriş stokuna geri döndüğü ana eşit olmasını sağlamak için yazılmıştır. (3.10) kısıtı ile eğer hatta i makinesi kullanılmadıysa robotun o makineye uğramaması, dolayısıyla ilgili aktivitenin olmaması sağlanmaktadır. Problem varsayımlarından biri olan her işin sadece bir makinede yapılması kısıtı (3.11) ile eklenmiştir. Öncelik ilişkileri (3.12) numaralı kısıt ile sağlanmaktadır yani eğer j işlemi k işleminin öncülü ise ve k işlemi l makinesine atandı ise j işlemi l makinesinden daha önceki bir makinede yapılmalıdır. Robotun her zaman giriş stoku önünden harekete başlaması, yani A_0 aktivitesinin ilk sırada yapılması için (3.13) numaralı kısıt yazılmıştır. Robotun her döngüye giriş stoku önünden başlaması optimalliği etkilememektedir. i makinesi montaj hattında kullanılmadıysa ona işlem atanmaması için (3.14) numaralı kısıt eklenmiştir. Hatta sıralanmış makinelerin sıralı bir şekilde kullanımı, iki aktif makine arasında kullanılmayan bir makinenin olmaması için (3.15) kısıtının sağlanması gerekmektedir. Yani i makinesi kullanılmıyorsa, $i+1$ makinesi de kullanılmamalıdır. Bununla beraber eğer i makinesi kullanılıyorsa, makineyi boşaltmak için i aktivitesinin gerçekleşmesi gerekmektedir, bu ise (3.16) kısıtı ile sağlanmaktadır. i makinesi kullanılmıyorsa i aktivitesi yapılmayacağı için yani robot yükleme boşaltma işlemi yapmayacağı halde makineler arasında boş gezinmemesi için l aktivitesinin i aktivitesinden sonraki bir pozisyona atanması söz konusu değildir. Bu sebeple (3.17) numaralı kısıt eklenmiştir. i makinesi kullanılmıyorsa i aktivitesinin yapılmaması ise (3.18) ile sağlanmaktadır. (3.19) ile (3.23) arasındaki kısıtlar ise karar değişkenlerinin işaret kısıtlarıdır.

Toplam makine sayısı önceliklendirilmiş çözüm yönteminin optimal sonuçlar veren matematiksel modeli yukarıdaki gibi oluşturulmuştur. Çevrim zamanı önceliklendirilmiş çözüm yönteminin matematiksel modeli Bölüm 3.2.1.2’de verilmiştir.

3.2.1.2. Çevrim zamanı önceliklendirilmiş çözüm yöntemi

Amaç öncelikli olarak çevrim zamanını minimize etmek, ikincil olarak minimum çevrim zamanı değerinde minimum toplam makine sayısını elde etmektir. Hedeflenen değerleri elde edebilmek için amaç fonksiyonunda çevrim zamanı belirlenmiş bir katsayı ile çarpılır bu şekilde öncelik en küçük çevrim zamanını bulmak olur. Ardından amaç fonksiyonunda yer alan toplam makine sayısı küçüklenir. Geliştirilen matematiksel model aşağıdaki gibidir:

$$\begin{aligned} \min \quad & \alpha C + \sum_{i=0}^m z_i & (3.24) \\ \text{st. (3.2) ... (3.23)} \end{aligned}$$

Modelin (3.24) numaralı amaç fonksiyonu öncelikli olarak çevrim zamanını ikincil olarak toplam makine sayısını en küçüklemeyi hedeflemektedir. Çevrim zamanı α ile çarpılarak önceliklendirilmiştir. α pozitif ve 1'den büyük bir değerdir. Diğer kısıtlar bir önceki bölümde anlatılan kısıtlar ile aynıdır sadece amaç fonksiyonunda öncelik değiştirilmiştir.

Hem toplam makine sayısının hem de çevrim zamanının aynı amaç fonksiyonu içinde en küçüklendiği matematiksel modeller yukarıda anlatılmıştır. Bu modellerle çevrim zamanı ve toplam makine sayısı değerini veren tek bir optimal çözüm elde edilmektedir. Bu modeller literatürde Montaj Hattı Dengeleme Probleminin Tip-E modeline girmektedir.

Toplam makine sayısı ile çevrim zamanı arasındaki ödünleşimi görmek için ise Pareto optimal çözümlerin belirlenmesi yöntemi uygulanmalıdır. Bu ödünleşimi görmenin bir diğer yolu da yukarıdaki modellerde α parametresini farklı değerlerle denemektir. Fakat α sonsuz farklı değer alabileceği ve hangi değerlerin başatlanmayan çözüm verebileceğini kestirmek mümkün olmayacağı için Pareto optimal çözümlerin belirlenmesi yönteminin uygulanmasına karar verilmiştir. Pareto

optimal çözümlerin belirlenmesi yöntemi için geliştirilmiş matematiksel modeller 3.2.2 başlığı altında anlatılmıştır.

3.2.2. Pareto Optimal Çözümlerin Belirlenmesi Yöntemi

Ele alınan robotik hücrelerde iki kriterli hat dengeleme probleminin çözüm yöntemlerinden biri Pareto optimal çözümlerin belirlenmesi yöntemidir. Pareto optimal çözümlerin elde edilmesi için değişik metotlar vardır. Bunlardan bir tanesi Bölüm 3.2.1’de anlatılan modellerde α parametresini farklı değerlerle denemektir. Fakat α sonsuz farklı değer alabilir ve hangi değerlerin başatlanmayan çözüm verebileceğini kestirmek mümkün değildir. Diğer bir metot ise amaçlardan bir tanesini kısıta kaydırmak ve kısıt üst limiti için farklı değerler deneyerek Pareto optimal çözümlerin belirlenmesidir. Bu bölümde hem toplam makine sayısı hem de çevrim zamanı amaçları ayrı ayrı kısıta kaydırılarak oluşturulan matematiksel modeller anlatılacaktır. Bu modeller kısıta kaydırılan kriterin çeşitli değerleri için çözdürülür ve başatlanmayan (domine edilmemiş) çözümler kümesi elde edilir. Bu yöntem için hazırlanan matematiksel modeller literatürdeki Montaj Hattı Dengeleme Probleminin Tip-1 ve Tip-2 modellerine benzerlik göstermektedir.

Domine Edilmemiş Çözümler: $f(\cdot)$ ve $g(\cdot)$ en küçüklenmek istenen amaç fonksiyonları olsun. $f(x_2) \leq f(x_1)$ ve $g(x_2) \leq g(x_1)$ ve bunlardan en az birini mutlak küçük olarak sağlayan herhangi bir x_2 çözümü yoksa x_1 çözümü başatlanmayan (domine edilmemiş) bir çözümdür.

Bu bölümde anlatılacak matematiksel modellerin amacı kısıta kaydırılan kriterin farklı üst limit değerleri ayrı ayrı toplam makine sayısı ve çevrim zamanı için başatlanmayan çözümler kümesini bulmaktır.

Bu çözüm yöntemi de, önceliklendirilmiş çözüm yöntemi gibi, hem toplam makine sayısı hem de çevrim zamanı kriterleri için, gerçek hayat problemine uygun şekilde iki ayrı amaç fonksiyonu ile modellenmiştir.

3.2.2.1. Çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yöntemi

Robotik hücrelerde iki kriterli hat dengeleme probleminin çözümü için uygulanan çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yönteminde amaç fonksiyonu toplam makine sayısını minimize ederken, çevrim zamanı amaç fonksiyonundan çıkarılarak kısıt olarak yazılır. Bu kısıtta çevrim zamanına bir üst limit değeri verilir. Bu üst limit için farklı farklı değerler denenerek farklı başatlanmayan çözümlere ulaşılmaya çalışılır. Bölüm 4’de gösterildiği gibi, çeşitli çevrim zamanı üst limit değerleri için model çözdürülerek tüm başatlanmayan çözümler kümesi elde edilir.

Çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yönteminin matematiksel modeli aşağıda verildiği gibidir:

$$\begin{aligned} \min \sum_{i=0}^m z_i & \quad (3.25) \\ \text{st. (3.2) ... (3.23)} & \end{aligned}$$

Modelin (3.25) numaralı amaç fonksiyonu toplam makine sayısını minimize etmektedir. Çevrim zamanı olan C değeri modelde kısıt olarak yazılmakta, çeşitli çevrim zamanı üst limit değerleri için model çözdürülmektedir.

Çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yönteminin matematiksel modeli yukarıda verildiği gibidir. Diğer yöntemde ise amaç fonksiyonu çevrim zamanını en küçükleme iken toplam makine sayısı kısıta kaydırılmıştır. Bu yöntem toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yöntemidir ve bölüm 3.2.2.2’de anlatılmıştır.

3.2.2.2. Toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yöntemi

Robotik hücrede iki kriterli hat dengeleme problemi için hazırlanan bir diğer çözüm yöntemi ise toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yöntemidir. Bu yöntemde amaç fonksiyonu çevrim zamanını minimize ederken, toplam makine sayısı amaç fonksiyonundan çıkarılarak parametreye çevrilir. Toplam makine sayısı için farklı farklı değerler denenerek farklı başatlanmayan çözümlere ulaşılmaya çalışılır. Bölüm 4'te gösterildiği gibi, çeşitli makine sayısı değerleri için model çözdürülerek tüm başatlanmayan çözümler kümesi elde edilir.

Toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yönteminin matematiksel modeli aşağıda verildiği gibidir:

$$\begin{aligned} \min C & \hspace{20em} (3.26) \\ \text{st. } (3.2), \dots, (3.9), (3.11), \dots, (3.13), (3.17), (3.19), (3.20), (3.22), (3.23) \end{aligned}$$

Modelin (3.26) numaralı amaç fonksiyonu çevrim zamanını minimize etmekte, toplam makine sayısı modelde parametre olarak geçmektedir. Bu modelde makine sayısı parametre olarak alındığı için, makine sayısının karar değişkeni olarak geçtiği kısıtlar modelden çıkarılmıştır.

Bölüm 3'te anlatılan robotik hücrede iki kriterli hat dengeleme problemi için oluşturulmuş çözüm yöntemlerinin detaylı test çalışmaları yapılmıştır. Test çalışma sonuçları Bölüm 4'te verilmiştir.

4. TEST SONUÇLARI

3. bölümde verilen matematiksel modellerin test çalışmaları CPLEX OPL 12.4 çözücüsü ile yapılmıştır. Veri olarak montaj hattı dengeleme literatüründen [32] alınan 14 veri dosyası kullanılmıştır. Bu veri dosyalarında problemlere ait işlem sayıları, işlem süreleri ve öncelik ilişkileri bulunmaktadır. Veri-8 içindeki veriler Ek B'de örnek olarak verilmiştir.

Bu çalışmada geliştirilen çözüm yöntemlerinin daha detaylı test sonuçlarının alınabilmesi için 14 veri dosyası yeterli bulunmamış ve daha fazla sayıda veri üzerinde çalışılması ihtiyacı duyulmuştur. Literatürde [32] toplamda 26 adet veri dosyası vardır ancak bu çalışma kapsamında kullanılmayan diğer 12 adet veri dosyasında çok az işlem sayısı mevcuttur ve bu çalışma kapsamında yeterli görülmemiştir. Bu sebeple 14 veri dosyasına ek olarak Veri-1, Veri-2, Veri-11 ve Veri-12'nin işlem sayısı ve öncelik matrisleri alınmış, işlem süreleri ise rasgele (20-100) ve (0-10000) arasından türetilerek 8 veri dosyası daha elde edilmiş ve toplamda 22 veri dosyası üzerinde çalışılmıştır.

Bu çalışma kapsamında kurulan matematiksel modelin parametrelerinden olan işlem zamanları arasındaki fark büyüdükçe işlemlerin makinelere atanması zorlaşmaktadır. İşlem süreleri arasındaki fark azaldıkça hat dengeleme probleminin çözümü kolaylaşmaktadır. Bu sebeple literatürden alınan veri dosyalarına ek olarak üretilen verilerde işlem süreleri arasındaki farkın az / çok olmasına dikkat edilmiş ve rasgele (20-100) / (0-10000) arasından türetilmiştir.

Ele alınan 22 veri dosyasının işlem sayıları Tablo 4.1'de verilmiştir. Robotun hareket süresi δ ve makineleri yükleme boşaltma süresi ε değerleri literatürde yer almadığından, bu çalışma kapsamında türetilmiştir.

Tablo 4.1 Veri Dosyası - İşlem Sayısı Tablosu

Veri#	İşlem Sayısı	Veri#	İşlem Sayısı	Veri#	İşlem Sayısı	Veri#	İşlem Sayısı
1	83	7	53	13	25	19	89
2	111	8	28	14	297	20	89
3	148	9	45	15	83	21	94
4	148	10	32	16	83	22	94
5	29	11	89	17	111		
6	35	12	94	18	111		

Bölüm 3’de detayları anlatılan iki çeşit çözüm yönteminin ikişer farklı versiyonu için yapılan test çalışmaları ve kullanılan parametreler bundan sonraki bölümlerde anlatılmıştır. Matematiksel modeller Intel Core i7-2620M CPU 2.70 GHz işlemcili, 8 GB belleğe sahip bilgisayarda CPLEX OPL 12.4 kullanılarak çözdürülmüştür. Yapılan test çalışmalarında az işlem sayısına sahip veri kümeleri için çok hızlı en iyi sonuç alındığı, işlem sayısı arttıkça çözüm zamanının da arttığı gözlemlenmiştir. Oluşturulan modeller gerçek hayat problemlerine uygulanacak tasarım modelleridir ve tasarlanacak montaj hatlarının optimal sonuçlar vermesi çok önemlidir. Bu sebeple modelin çözüm süresinin kısa olmasından çok optimal sonuçlara ulaşılması hedeflenmiş ve bu bölümde verilen test sonuçlarından da anlaşılacağı gibi en iyi sonuçlara ulaşmak için modeller gerektiğinde uzun süreli çözdürülmüştür.

4.1. Önceliklendirilmiş Çözüm Yöntemi

Robotik hücrelerde iki kriterli hat dengeleme probleminin çözüm yöntemlerinden ilki olan önceliklendirilmiş çözüm yöntemi için oluşturulan matematiksel modelde yukarıda anlatılan veriler kullanılmış ayrıca ε , δ ve α değerleri türetilmiştir. ε ve δ değerlerinin, işlem zamanlarına kıyasla büyüklüğü/küçüklüğü problemin çözüm süresini etkilemektedir. Yapılan denemelerde aşağıdaki aralıkların en uygun değerler olduğu görülmüştür. ε ve δ değerlerinin çok küçük alınması literatürde “downhill cycle” adı verilen $A_0A_mA_{m-1}A_{m-2} \dots A_1$ aktivite sıralamasına sahip döngünün her zaman optimal olmasına sebep olmaktadır. Bu döngüde robot bir makineyi yükledikten sonra makine önünde beklemekten başka makinelere giderek yükleme

boşaltma işlemini gerçekleştirmektedir. Dolayısıyla robot çok fazla hareket etmektedir. ε ve δ değerlerinin işlem zamanı değerlerine kıyasla küçük olması ise bu döngünün daha iyi sonuçlar vermesini sağlamaktadır. Bu döngüde optimal işlem ataması ise makinelerdeki işlem zamanını dengeleyecek şekilde yapılan atamadır. Dolayısıyla klasik montaj hattı dengeleme problemleri için geliştirilen çözüm yöntemleri aynen kullanılabilir. Diğer taraftan ε ve δ değerleri işlem zamanlarına kıyasla çok büyük değerler alırsa literatürde “uphill cycle” adı verilen $A_0A_1A_2A_3 \dots A_m$ aktivite sıralamasına sahip döngünün her zaman optimal olmasına sebep olmaktadır. Bu döngüde robot birinci makineden başlayarak her makineyi önce yükler ardından işlemin tamamlanması için bekler ve sonra da makineyi boşaltır. Dolayısıyla işlem zamanının kısa ε ve δ değerlerinin uzun olması bu döngünün iyi olmasını sağlamaktadır. Bu döngüde robot bütün makinelerin önünde işlem süresi boyunca beklediği için işlemlerin makinelere atanmasının sonuca bir etkisi olmamaktadır.

Yukarıda bahsedilen, çözümü bilinen iki özel durum dışındaki döngüleri elde edebilmek için ε ve δ değerleri işlem zamanlarına bağlı olarak aşağıdaki şekilde türetilmektedir.

R işlem zamanlarının aritmetik ortalaması olmak üzere ε ve δ değerleri düzgün dağılımdan aşağıdaki aralıklar kullanılarak hesaplanmıştır:

$$\varepsilon: Uniform \left(0, \frac{R}{20} \right)$$

$$\delta: Uniform \left(\max \left\{ 0, (m-2) \frac{R - (m-2)\varepsilon}{100} \right\}, \frac{R - (m-2)\varepsilon}{20} \right)$$

Önceliklendirilmiş çözüm yönteminin test çalışmasında ε ve δ değerlerindeki rasgeleliği azaltmak için her kombinasyondan 5'er tekrar alınmıştır.

Matematiksel modelde kritere öncelik veren α katsayısının belirlenmesi gerekmektedir. Bu katsayı bir kriterin en küçüklenmesini diğerinden daha öncelikli yapmalıdır. α katsayısının nasıl belirlendiği 4.1.1 ve 4.1.2 bölümlerinde anlatılmıştır.

Toplam makine sayısı ve çevrim zamanı için kurulan önceliklendirilmiş çözüm yöntemi modellerinin çözdürülmesi sonucu elde edilen sonuçlar bundan sonraki bölümlerde verilmiştir.

4.1.1. Toplam Makine Sayısı Önceliklendirilmiş Çözüm Yöntemi

Robotik hücrelerde iki kriterli hat dengeleme problemi için oluşturulmuş toplam makine sayısı önceliklendirilmiş çözüm yönteminin amacı iki kriterden biri olan makine sayısının öncelikli olarak en küçüklenmesi ardından en küçük çevrim zamanının bulunmasıdır.

Bu yöntemde toplam makine sayısını önceliklendiren α katsayısı, toplam makine sayısını çevrim zamanı değerine göre önceliklendireceği için α katsayısı belirlenirken işlem sürelerine kıyasla bir değer bulunmalıdır. Bu çalışmada kullanılan veri dosyalarında işlem sayısı ve işlem süreleri büyüktür. Dolayısıyla α katsayısı belirlenirken, eğer işlem süreleri toplamından yüksek bir değer alınırsa model otomatik olarak tek makine kullanılan, tüm işlemlerin tek makinede yapıldığı bir çözüm vermektedir çünkü toplam makine sayısı toplam işlem zamanı değerine göre çok küçük kalmaktadır. Bu durumda da çevrim zamanı çok yüksek bir değer çıkmakta, gerçek hayat modellerine uygun değerler elde edilememektedir. Çünkü gerçek hayatta gelişen teknoloji ve rekabet ile beraber birim zamanda üretilen ürün sayısının artışı önem kazanmıştır bu sebeple çevrim zamanının en küçüklenmesi rekabet koşullarına uyum sağlanabilmesi için önemlidir. Eğer α katsayısı 1 alınırsa bu durumda makine sayısının en küçüklenmesi önceliklendirilmemiş olmakta ve bu yöntemin hedefine uygun değerler elde edilmemektedir. Bu sebeple α katsayısının çözüm yöntemi amacına uygun belirlenmesi gerekmektedir. Ele alınan çalışmanın çözüm yöntemlerinden biri olan toplam makine sayısı önceliklendirilmiş çözüm yönteminde α katsayısı olarak işlem sürelerinin en büyüğü alınmıştır. Çünkü hatta

bir makine daha fazla kullanılması en büyük işlem zamanı kadar amaç fonksiyonunu artıracaktır ancak o makinenin kullanılmaması daha düşük işlem zamanlı işlemlerin makinelere atanmasıyla çevrim zamanını artıracaktır fakat bu artış en büyük işlem zamanından daha düşük veya eşit olacaktır. Bu sebeple model daha az sayıda makine kullanımına yönelecektir. α katsayısı olarak işlem sürelerinin en büyüğü alınarak toplam makine sayısı değeri çözüm yönteminin amacına uygun olarak önceliklendirilmiştir. α katsayısının farklı değerleri ile model test edilmiştir.

Toplam makine sayısı önceliklendirilmiş çözüm yönteminde makine sayısı ve çevrim zamanı en küçüklenmesi çalışmasında elde edilen sonuçların detayları Ek A Tablo A.1’de verilmiştir. Her bir veri dosyası 5 farklı ε ve δ değerleri için çözdürülmüş, CPLEX çözüm zamanı, toplam makine sayısı ve çevrim zamanı değerleri Tablo A.1’e işlenmiştir. Tablo A.1’de görüldüğü üzere 22 veri dosyası için yapılan çalışmada optimal sonuçlar CPLEX 12.4 çözücüsüyle çok hızlı elde edilmiştir ancak en fazla işlem sayısına sahip Veri-14 veri dosyası için optimal sonucun bulunması diğer verilere göre daha uzun sürmüştür.

Toplam makine sayısı önceliklendirilmiş çözüm yönteminin Veri-11 için çözdürülmesi ile elde edilen sonuçlar Tablo 4.2’de verilmiştir.

Tablo 4.2. Veri-11 için Toplam Makine Sayısı Önceliklendirilmiş Çözüm Yöntemi Sonuçları

Veri#	Tekrar#	ε	δ	CPU (sn)	Makine Sayısı	Çevrim Zamanı
11	1	0	1	36,92	7	74
	2	1	1	55,43	7	78
	3	2	0	137,92	7	78
	4	0	2	81,00	7	78
	5	2	1	168,03	7	82

Tablo 4.2’de verilmiş olan sonuçlar toplam makine sayısı önceliklendirilmiş çözüm yönteminin Veri-11 ile çözdürülmesiyle elde edilmiştir. Örneğin Veri-11’in ilk tekrarında, $\varepsilon = 0$, $\delta = 1$ değerleri ile 36.92 saniyede optimal çözüme ulaşılmıştır.

Montaj hattında 7 makine kullanılması optimal çıkmış ve 7 makine için en küçük çevrim zamanı 74 birim zaman olarak bulunmuştur.

Toplam makine sayısı önceliklendirilmiş çözüm yönteminin, önceliklendirme katsayısı α 'nın değişik değerleri için Veri-11 ile çözdürülmesi sonucu elde edilen değerler Tablo 4.3'de verilmiştir. α 'nın farklı değerleri aşağıdaki gibi hesaplanmıştır:

1. İşlem sürelerinin ortalaması (6),
2. İşlem sürelerinin en büyüğü (10),
3. İşlem sürelerinin toplamı (485).

Bu üç değerden ikincisi test çalışmalarında kullanılan değerdir. Test çalışmalarında kullanılan değerden büyük ve küçük olan farklı iki değer için model test edilmiştir.

Tablo 4.3. Veri-11'in Farklı α Değerleri için Toplam Makine Sayısı Önceliklendirilmiş Çözüm Yöntemi Sonuçları

Veri#	α	ε	δ	CPU (sn)	Makine Sayısı	Çevrim Zamanı
11	6	0	1	232,85	9	58
	10			36,92	7	74
	485			0,95	1	489

Toplam makine sayısının önceliklendirme katsayısı küçük olan $\alpha = 6$ değeri için model daha fazla makine kullanan çözüme ulaşmıştır. Çünkü amaç fonksiyonunda toplam makine sayısının önceliği azaltılmıştır. Matematiksel modelin CPLEX OPL 12.4 çözdürücüsünde çözüm süresi uzamıştır. Bununla beraber önceliklendirme katsayısı büyük olan $\alpha = 485$ değeri için amaç fonksiyonunda toplam makine sayısı daha çok önceliklendirilmiştir. Bunun sonucu olarak da model daha az makine kullanımına yönelmiştir. Önceliklendirme katsayısı olarak işlem sürelerinin toplamı alındığında model tek bir makine kullanılan optimal çözüme ulaşmıştır. CPLEX OPL 12.4 çözdürücüsünde çözüm süresi azalmıştır.

22 veri dosyası için makine sayısı önceliklendirilmiş çözüm yönteminde optimal çözüme kısa sürede ulaşıldığı görülmüştür. Bununla beraber işlem sayısı büyüdükçe çözdürücünün çözüm zamanı da artmaktadır. Kurulan matematiksel model gerçek hayat tasarım problemlerinin çözülmesi için oluşturulduğundan büyük veri dosyaları için CPLEX OPL çözdürücüsünün uzun sürede çözümler vermesi kabul edilebilirdir çünkü robotik montaj hattı tasarımı için uzun vadeli karar verilecektir.

Ele alınan çalışmanın diğer kriteri olan çevrim zamanının önceliklendirilmesi yöntemi için elde edilen sonuçlar Bölüm 4.1.2’de verilmiştir.

4.1.2. Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemi

Çevrim zamanı önceliklendirilmiş çözüm yönteminin amacı öncelikli olarak çevrim zamanı değerinin ardından toplam makine sayısı değerinin en küçüklenmesidir.

Çevrim zamanı önceliklendirilmiş modelin çözdürülmesi için ε ve δ değerleri Bölüm 4.1’de anlatıldığı gibi hesaplanmıştır. α katsayısı çevrim zamanını önceliklendirecek çarpan olup eğer α çok büyük bir değer alırsa toplam makine sayısı değeri işlem sayısına yakın bir değer çıkmaktadır ve bu durum gerçek hayat problemi kısıtlarına uygun olmamaktadır. Çünkü gerçek hayat problemlerinde maliyet ve tesis planlama konuları makine sayısını kısıtlamakta bu durumda da her işlem için bir makine kullanılamamaktadır. Bununla beraber α katsayısı 1 alırsa çevrim zamanı önceliklendirilmiş çözüm yöntemi amacına uygun değerler elde edilmemektedir, çevrim zamanı ve toplam makine sayısı aynı oranda önceliklendirilmektedir. Çalışılan veri dosyalarında işlem süreleri yüksek değerler olduğu için çevrim zamanı toplam makine sayısından her zaman çok yüksek çıkmaktadır. Bu sebeplerden dolayı α katsayısı olarak çok yüksek olmayan bir değer alınmalı ancak 1 olmamalıdır. α katsayısının 3 alınmasına karar verilmiştir bu şekilde hem çevrim zamanı önceliklendirilmiş hem de toplam makine sayısının gerçek hayat problemleri için geçerli bir değer olması sağlanmıştır. α katsayısının farklı değerleri ile model test edilmiştir.

Çevrim zamanı önceliklendirilmiş çözüm yönteminde çevrim zamanı ve makine sayısı en küçüklenmesi çalışmasında elde edilen sonuçların detayları Ek A Tablo A.2’de verilmiştir. Her bir veri dosyası 5 farklı ε ve δ değerleri için çözdürülmüş, CPLEX çözüm zamanı, toplam makine sayısı ve çevrim zamanı değerleri Tablo A.2’ye işlenmiştir. Tablo A.2’de görüldüğü üzere 22 veri dosyası için yapılan çalışmada optimal sonuçlar CPLEX 12.4 çözücüsüyle hızlı elde edilmiştir ancak işlem sayısı yüksek veri dosyaları için optimal sonucun bulunması diğer verilere göre daha uzun sürmüştür.

Çevrim zamanı önceliklendirilmiş çözüm yöntemin Veri-11 için çözdürülmesi ile elde edilen sonuçlar Tablo 4.3’de verilmiştir.

Tablo 4.4. Veri-11 için Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemi Sonuçları

Veri#	Tekrar#	ε	δ	CPU (sn)	Sapma (%)	Makine Sayısı	Çevrim Zamanı
11	1	0	1	577,25	0	10	53
	2	1	1	3.295,26	0	9	62
	3	2	0	5.619,95	0	10	57
	4	0	2	6.123,89	0	10	60
	5	2	1	10.529,51	0	10	66

Tablo 4.3’de verilmiş olan sonuçlar öncelikli olarak çevrim zamanını ardından makine sayısını en küçüklemeyi hedefleyen matematiksel modelin Veri-11 ile çözdürülmesiyle elde edilmiştir. Örneğin Veri-11’in ilk tekrarında $\varepsilon = 0$, $\delta = 1$ değerleri ile 577,25 saniyede optimal çözüme ulaşılmıştır. Verilen parametrelerle montaj hattının en küçük çevrim zamanı 53 çıkmış, çevrim zamanı 53 değeri için kullanılabilir en az makine sayısı 10 bulunmuştur. Ayrıca Veri-11’in ikinci tekrarında $\varepsilon = 1$, $\delta = 1$ değerleri ile 3.295,26 saniyede optimal çözüme ulaşılmıştır ve montaj hattının en küçük çevrim zamanı 62 çıkmış, çevrim zamanı 62 değeri için kullanılabilir en az makine sayısı 9 bulunmuştur. İkinci tekrarda makine sayısının birinci tekrara göre daha az çıkmasının sebebi makine yükleme boşaltma süresini gösteren ε değerinin 1 artmış olmasıdır. Makine yükleme boşaltma süresi çevrim

zamanını etkilediği için model daha az makine kullanarak en küçük çevrim zamanını bulmuştur.

Çevrim zamanı önceliklendirilmiş çözüm yönteminde bazı veri dosyalarının çözüm süresi 2 günden fazla sürmüştür. Bu tip veriler için CPLEX OPL’de çözüm toleransı %0.05 olarak ayarlanmış ve optimal değerde sapma oranı %0.05 olarak çözüm elde edilmiştir. Bu şekilde çözdürülen Veri-10’un çözüm değerleri Tablo 4.4’de verilmiştir.

Tablo 4.5. Veri-10 için Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemi Sonuçları

Veri#	Tekrar#	ϵ	δ	CPU (sn)	Sapma (%)	Makine Sayısı	Çevrim Zamanı
10	1	41	11	686,63	0,05	10	1734
	2	14	18	168,33	0,05	10	1654
	3	44	14	457,08	0,05	10	1758
	4	37	12	1.623,55	0,05	10	1722
	5	34	16	304,45	0,05	10	1726

Çevrim zamanı önceliklendirilmiş çözüm yönteminin, önceliklendirme katsayısı α ’nın değişik değerleri için Veri-11 ile çözdürülmesi sonucu elde edilen değerler Tablo 4.6’da verilmiştir. $\alpha = 2, \alpha = 3$ ve $\alpha = 10$ değerleri kullanılmıştır. Bu üç değerden ikincisi test çalışmalarında kullanılan değerdir. Test çalışmalarında kullanılan değerden büyük ve küçük olan farklı iki değer için model test edilmiştir.

Tablo 4.6. Veri-11’in Farklı α Değerleri için Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemi Sonuçları

Veri#	α	ϵ	δ	CPU (sn)	Makine Sayısı	Çevrim Zamanı
11	2	0	1	124,06	8	96
	3			577,25	10	53
	10			1.824,58	10	53

Toplam makine sayısının önceliklendirme katsayısı küçük olan $\alpha = 2$ deęeri için model daha az makine kullanan çözüme ulaşmıştır. Çünkü amaç fonksiyonunda çevrim zamanının öncelięi azaltılmıştır ve optimal çevrim zamanı deęeri dięer çözümlere göre yüksek elde edilmiştir. Matematiksel modelin CPLEX OPL 12.4 çözdürücüsünde çözüm süresi kısalmıştır. Bununla beraber önceliklendirme katsayısı büyük olan $\alpha = 10$ deęeri için amaç fonksiyonunda çevrim zamanı daha çok önceliklendirilmiştir. Ancak çevrim zamanını azaltmak için bir makine daha fazla kullanılması robot hareketlerini artıracığından çevrim zamanını yükseltmektedir bu sebeple $\alpha = 10$ için çözdürücünün çözüm süresi uzamasına rağmen $\alpha = 3$ 'e göre daha iyi bir sonuç elde edilememiştir.

22 veri dosyası için çevrim zamanı önceliklendirilmiş çözüm yönteminde küçük veri dosyaları için optimal çözüme kısa sürede ulaşıldığı görülmüştür. Bununla beraber işlem sayısı büyüdükçe çözdürücünün çözüm süresi de artmaktadır. Kurulan matematiksel model gerçek hayat tasarım problemlerinin çözülmesi için oluşturulduğundan büyük veri dosyaları için CPLEX OPL çözdürücüsünün uzun sürede çözümler vermesi kabul edilebilirdir çünkü robotik montaj hattı tasarımı için uzun vadeli karar verilecektir. Ancak 3 günden daha fazla sürede çözüm veren veriler için sapma oranı %0.05 alınmıştır. Bu deęer çok küçük ve optimale çok yakındır.

Robotik hücrelerde iki kriterli hat dengeleme probleminin dięer çözüm yöntemi olan Pareto optimal çözümlerin belirlenmesi yönteminin test sonuçları Bölüm 4.2'de verilmiştir.

4.2. Pareto Optimal Çözümlerin Belirlenmesi

Pareto optimal çözümlerin belirlenmesi yönteminin amacı sırasıyla makine sayısı ve çevrim zamanı için domine edilmemiş çözümler kümesini bulmaktır. Çevrim zamanına ε, δ ve işlem zamanları etki etmektedir. Makine sayısının artmasıyla robotun toplam yükleme boşaltma ve toplam taşıma zamanı deęerleri artmakta bununla beraber makine işlem zamanları azalmaktadır. Yani bu iki toplam deęerleri

arasında ödünleşim vardır. Bu ödünleşim ilişkisini görebilmek için Pareto optimal çözümlerin belirlenmesi yönteminde ε ve δ değerleri küçük alınmıştır.

4.2.1. Çevrim Zamanını Parametrize Ederek Pareto Optimal Çözümleri Bulma Yöntemi

Çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yönteminde amaç kısıta kaydırılan çevrim zamanı için üst değerler verilerek en küçük toplam makine sayısının belirlenmesidir. Modelde kullanılan çevrim zamanı üst limiti azaltıldıkça hatta kullanılan makine sayısı artmaktadır. Elde edilen çevrim zamanı makine sayısı ikililerinden biri diğerinden daha iyi değerlendirmesi yapılamaz çünkü gerçek hayat modellerinde çevrim zamanı yüksek tutulurken az makine sayısı kullanılabilir veya düşük çevrim zamanı elde edilmesi için yüksek sayıda makine sayısı hatta çalıştırılabilir. Elde edilen çevrim zamanı makine sayısı ikililerinden oluşan başatlanmayan çözüm kümesini bulmak için çevrim zamanı C için üst limit verilerek model çözdürülmeye başlanır, en küçük makine sayısı değeri elde edildikten sonra çevrim zamanı azaltılarak model tekrar çözdürülür. Her adımda bu prosedür uygulanır. Çevrim zamanı üst limiti çözdürülen matematiksel model için olursuz olana kadar çalışılmaya devam edilir.

Modelin her adımında kullanılacak çevrim zamanı üst değeri aşağıdaki adımlar uygulanarak belirlenmiştir:

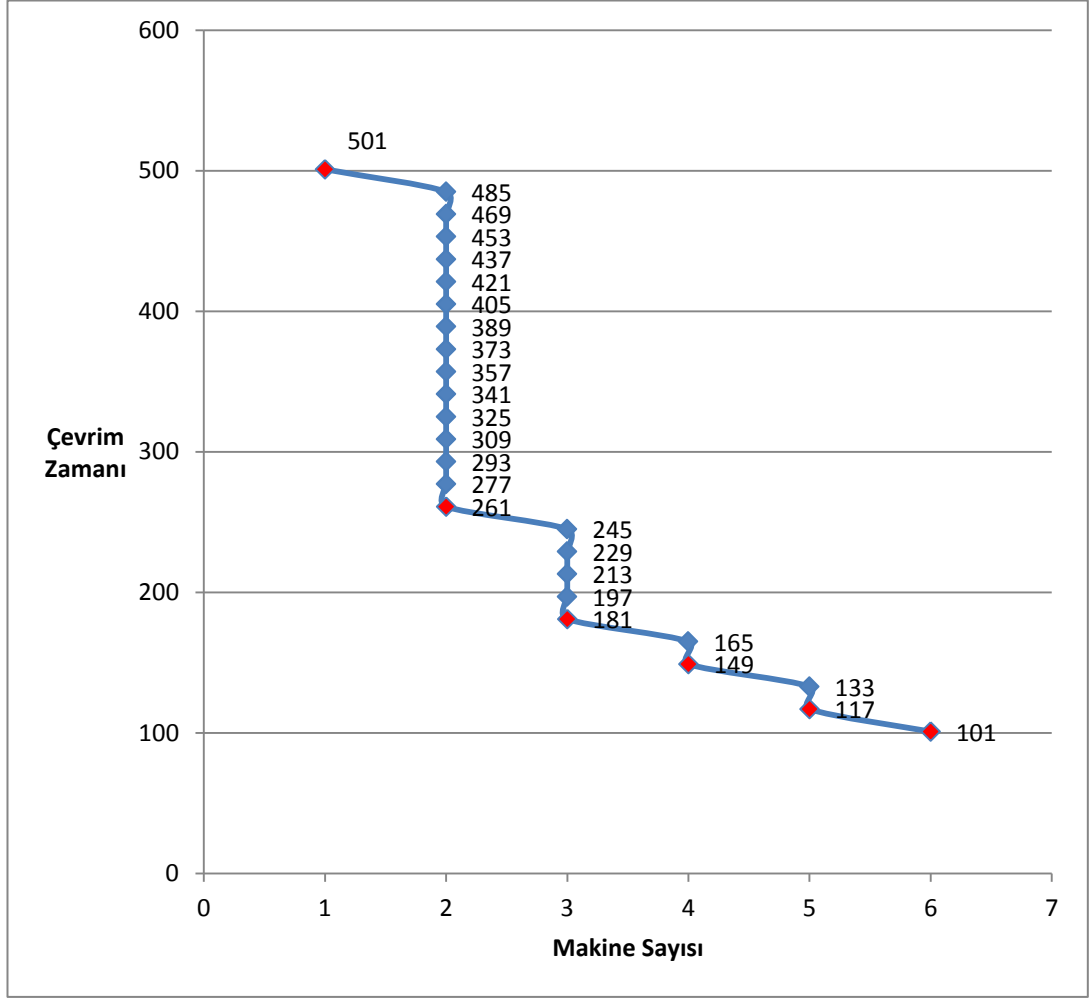
1. Toplam işlem süreleri hesaplanır ve üst limit olarak alınır.
2. İşlem süreleri ortalaması hesaplanır ve alt sınır olarak alınır.
3. Üst limit ve alt limit arasındaki fark hesaplanır.
4. Elde edilen fark θ 'ya bölünür ve bu değer çevrim zamanı üst değerinden her adımda çıkarılır (çevrim zamanı üst değeri azaltma sayısı).
5. İlk çevrim zamanı üst değerinin belirlenebilmesi için toplam işlem sürelerine, toplam makine sayısı 1 olacak şekilde ε ve δ değerlerinin toplamı eklenir.
6. İlk çevrim zamanı üst değeri ile model çözdürülür.

7. Her adımda çevrim zamanı üst değerinden azaltma sayısı çıkarılarak model çözdürülür.

Çevrim zamanı üst değeri azaltma sayısı hesaplanmasında kullanılan θ değeri 30 olarak alınmıştır. Çünkü daha küçük bir sayıya bölünmesi çevrim zamanı üst değerinin her adımda çok hızlı azalmasına bu da optimal makine sayılarının ardışık bir şekilde elde edilmesine engel olacaktır. Modelin amacı başatlanmayan çözümler kümesinin bulunması olduğundan en fazla başatlanmayan değerlerin elde edilmesi yani her çevrim zamanı üst değerine karşılık gelecek makine sayılarının ardışık bulunması önemlidir. Eğer üst limit ile alt limit arasındaki fark 30'dan daha büyük bir değere bölünürse bu durumda da çok fazla başatlanan çözümlere ulaşılabilecek ve Pareto optimal çözümlerin belirlenmesi formülasyonunun amacına uygun değerler elde edilmeyecek sadece çok daha fazla model çözdürülerek zaman kaybına sebep olacaktır.

Çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yönteminin CPLEX OPL 12.4'de çözdürülmesi ile elde edilen değerler Ek-A'da verilmiştir. Her bir veri dosyası çevrim zamanı değeri azaltılarak çözdürülmüş, CPLEX çözüm zamanı, toplam makine sayısı ve çevrim zamanı değerleri Ek A'daki tablolara işlenmiştir. Tablolarda görüldüğü üzere 22 veri dosyası için yapılan çalışmada optimal sonuçlar CPLEX 12.4 çözücüsüyle hızlı elde edilmiştir ancak işlem sayısı yüksek veri dosyaları için optimal sonucun bulunması diğer verilere göre daha uzun sürmüştür.

Çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yönteminin Veri-11 için çözdürülmesi ile elde edilen sonuçlar örnek olarak Şekil 4.1'de verilmiştir.



Şekil 4.1. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler Grafiği – Veri-11

Şekil 4.1’de verilmiş olan sonuçlar çevrim zamanı verilmiş problemde makine sayısını en küçükmeyi hedefleyen matematiksel modelin Veri-11 ile çözdürülmesiyle elde edilmiştir. Örneğin çevrim zamanı 501 değeri için model çözdürüldüğünde toplam makine sayısının en küçüklenmiş değeri 1 çıkmaktadır. Ardından çevrim zamanı değeri 16’şar azaltılmış olup 485 değeri ile model çözdürüldüğünde en küçüklenmiş makine sayısı değeri 2 çıkmaktadır. Bu şekilde çevrim zamanı değeri azaltılarak model çözdürülmüştür. Şekil 4.1’de kırmızı ile işaretlenmiş değerler başatlanmamış çözümler, mavi ile gösterilen değerler başatlanmış çözümlerdir. Örneğin çevrim zamanı 485 iken makine sayısı 2, çevrim zamanı 261 iken de makine sayısı 2’dir. Dolayısıyla ikinci çözüm (C=261) ilk

çözümü ($C=485$) domine etmektedir. Çevrim zamanı 101 iken makine sayısı 6 çıkmıştır ancak çevrim zamanı değeri 16 azaltılıp 85'e indirildiğinde model olursuz olmuştur çünkü çevrim zamanı değeri öncelik ilişkilerine göre tüm işlemlerin makinelere atanması için yeterli olmamıştır.

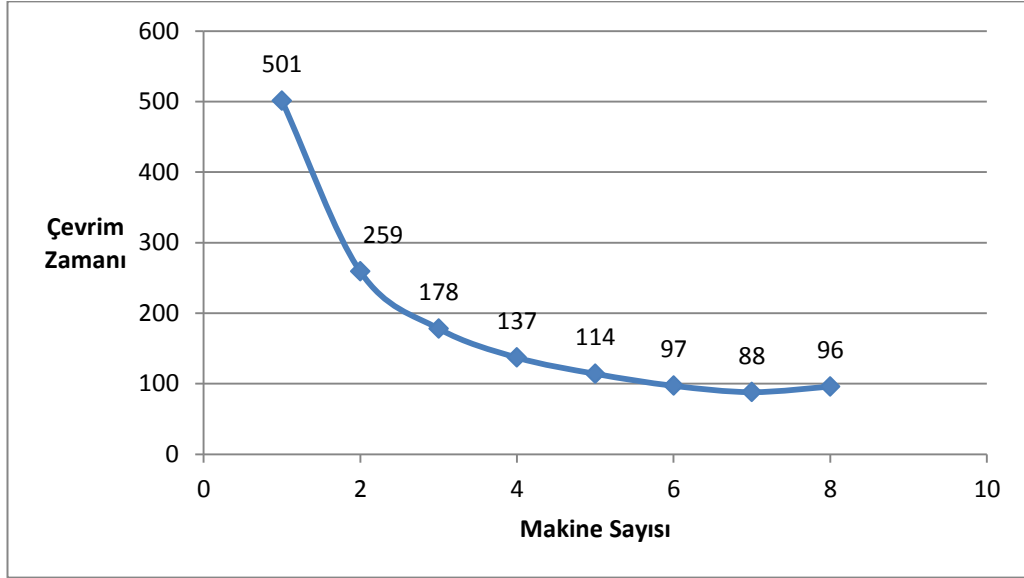
Çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yöntemi ile çözdürülen 22 veri dosyasının detaylı sonuçları Ek-A'da verilmiştir. Toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yönteminin test sonuçları ise Bölüm 4.2.2'de anlatılmıştır.

4.2.2. Toplam Makine Sayısını Parametrize Ederek Pareto Optimal Çözümleri Bulma Yöntemi

Toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yönteminin amacı parametre olarak verilen toplam makine sayısı için en küçük çevrim zamanını bulmaktır. Başlatılmayan çözümleri bulmak için makine sayısı $m = 1$ ile model çözdürülmeye başlanır, sonuç elde edildikten sonra makine sayısı bir artırılarak model tekrar çözdürülür. Her adımda bu prosedür uygulanır. Bir sonraki adımda elde edilen çevrim zamanı ve makine sayısı, bir önceki adımda elde edilen çevrim zamanı ve makine sayısı değerinden kötü olduğunda durulur. Çünkü bir sonraki adımda hem makine sayısı hem de çevrim zamanı değerleri, bir önceki adıma göre yüksektir ve o adımdan sonra da azalmayacaktır.

Toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yönteminin CPLEX OPL 12.4'de çözdürülmesi ile elde edilen değerler Ek-A'da verilmiştir. Her bir veri dosyası makine sayısı değeri azaltılarak çözdürülmüş, CPLEX çözüm zamanı, toplam makine sayısı ve çevrim zamanı değerleri Ek A'daki tablolara işlenmiştir. Tablolarda görüldüğü üzere 22 veri dosyası için yapılan çalışmada optimal sonuçlar CPLEX 12.4 çözücüsüyle hızlı elde edilmiştir ancak işlem sayısı yüksek veri dosyaları için optimal sonucun bulunması diğer verilere göre daha uzun sürmüştür.

Toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yönteminin Veri-11 için çözdürülmesi ile elde edilen sonuçlar Şekil 4.2’de verilmiştir.



Şekil 4.2. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler Grafiği – Veri-11

Şekil 4.2’de verilmiş olan sonuçlar makine sayısı verilmiş problemde çevrim zamanını en küçüklemeyi hedefleyen matematiksel modelin Veri-11 için çözdürülmesiyle elde edilmiştir. Örneğin makine sayısı 1 için model çözdürüldüğünde çevrim zamanının en küçüklenmiş değeri 501 çıkmaktadır. Ardından makine sayısı 2 için model çözdürüldüğünde en küçüklenmiş çevrim zamanı değeri 259 çıkmaktadır. Bu şekilde makine sayısı değeri artırılarak model çözdürülmüştür. Makine sayısı 8 iken çevrim zamanı değeri bir önceki çözümde elde edilenden yüksek çıkmıştır. Hem makine sayısı hem de çevrim zamanı artış göstermiştir. Bu noktadan sonra makine sayısı artırıldıkça çevrim zamanı değeri de artacağından bulunacak sonuçlar başatlanmış çözümler olacaktır. Bu sebeple makine sayısı 8’den sonra model çözdürülmemiştir.

Test sonuçları verilmiş iki tip çözüm yönteminin iki ayrı amaç fonksiyonu ile oluşturulmuş matematiksel modelleri yukarıda anlatılmıştır. Çeşitli veri dosyaları ile yapılan test çalışmaları sonucunda, büyük işlem sayısına sahip verilerle çözdürülen matematiksel modellerin optimal çözümlerine daha uzun sürede ulaşıldığı görülmüştür. Bununla beraber daha az işlem sayısına sahip veriler daha hızlı çözdürülmüştür.

Test çalışmaları için literatürden alınan 14 veri dosyasına ek olarak işlem zamanları rasgele (20-100) ve (0-10000) arasından türetilen veri dosyalarının çözüm sürelerinin farklı olduğu görülmüştür. İşlem süreleri rasgele (20-100) arasından türetilen veriler her iki çözüm yönteminde de daha hızlı sonuçlar vermiştir. İşlem süreleri rasgele (0-10000) arasından türetilen veriler ise her iki çözüm yönteminde de daha uzun sürede sonuçlar vermiştir.

Bölüm 5'te, bu çalışmada kullanılan iki çeşit çözüm yönteminin yapılan testlerden elde edilen sonuçlara göre detaylı karşılaştırması yapılmış, elde edilen sonuçlar yorumlanmıştır.

5. ÇÖZÜM YÖNTEMLERİNİN KARŞILAŞTIRILMASI

Bu çalışma kapsamında ele alınan robotik hücrelerde iki kriterli hat dengeleme probleminin çözümünde iki yöntem kullanılmıştır:

1. Önceliklendirilmiş çözüm yöntemi,
2. Pareto optimal çözümlerin belirlenmesi yöntemi.

İlk yöntemde ele alınan problem iki kriterli olduğu için kriterlerden birinin önceliklendirilerek en küçüklendiği ardından diğer kriterin elde edilen değerler içinde en küçüklendiği bir çalışma yapılmıştır. Bu çalışma sonucunda iki ayrı amaç fonksiyonuna ancak aynı kısıtlara sahip iki çeşit matematiksel model türetilmiş, hem montaj hattı dengeleme probleminin çözümü hem de robotun hareket çizelgesi beraber elde edilmiştir. Literatürde hem çevrim zamanı ve makine sayısını en küçükleyen montaj hattı dengeleme problemi hem de robotik hücre çizelgeleme problemi aynı matematiksel model içinde modellenip tek bir parametre girişiyle optimal çözüme ulaştıran matematiksel model çalışması bulunmamaktadır. Bu sebeple ele alınan çalışma literatüre katkı sağlamıştır.

Diğer bir çözüm yöntemi olan Pareto optimal çözümlerin belirlenmesi yönteminde amaç fonksiyonlarından biri kısıta kaydırılarak diğer amaç fonksiyonu en küçüklenmiştir. Bu yöntemle elde edilen sonuçlar domine edilmemiş çözümler kümesini vermiştir. Pareto optimal çözümlerin belirlenmesi yönteminde elde edilen sonuçlar optimal çevrim zamanı toplam makine sayısı ikililerini bulmakta, karar vericinin bu optimal değerler arasından seçim yapmasına olanak sağlamaktadır. Karar vericinin kullanılacak toplam makine sayısında esnekliği varsa bu yöntem ile en uygun çözümü seçebilmektedir. Pareto optimal çözümlerin belirlenmesi yönteminde önceliklendirilmiş çözüm yöntemine göre daha kapsamlı sonuçlar elde edilmektedir.

Geliştirilen matematiksel modellerin çözüm yöntemleri gerçek hayatta uygulanabilir yöntemlerdir. Gerçek hayat problemleri bu yöntemlerle çözdürüldüğünde uzun vadeli

tasarım kararları verilecektir. Bu sebeple çok kısa sürede modellerin çözdürülmesi beklenmez, çözümlere ulaşabilmek için gerekirse birkaç gün beklenebilir. Bu çalışmada ele alınan çözüm yöntemlerinin CPLEX OPL çözüm süreleri, işlem sayısına ve öncelik matrisine göre değişkenlik göstermektedir. 22 veri dosyası için önceliklendirilmiş çözüm yöntemine ait CPLEX OPL 12.4 çözdürücüsünün ortalama çözüm süreleri ve Pareto optimal çözümlerin belirlenmesi yöntemine ait CPLEX OPL 12.4 çözdürücüsünün toplam çözüm süreleri Tablo 5.1 ve Tablo 5.2’de verilmiştir.

Tablo 5.1’de önceliklendirilmiş çözüm yöntemi ile CPLEX OPL’de çözdürülen 22 adet veri dosyalarına ait ortalama çözüm süreleri verilmiştir. Önceliklendirilmiş çözüm yönteminde ε ve δ değerleri arasındaki rasgeleliği ortadan kaldırmak için 5 farklı replikasyon yapılmıştır. Her bir veri dosyası 5 farklı ε ve δ değerleri ile çözdürüldü ve Tablo 5.1’de her bir veri dosyası için 5 farklı replikasyonun çözüm sürelerinin ortalaması yer almaktadır. Tablo 5.1’den görüleceği üzere toplam makine sayısı önceliklendirilmiş çözüm yönteminin test sonuçları çevrim zamanı önceliklendirilmiş çözüm yöntemine göre daha hızlı alınmıştır.

Çevrim zamanı önceliklendirilmiş çözüm yönteminde Veri-10 gibi bazı modellerin çözümü 1.5-2 günden daha uzun sürdüğü için çözüm toleransı %0,05 yapılarak CPLEX OPL çözdürücüsünde çözdürülmüştür. Bu çözüm yönteminde optimal çözümden %0,05 uzak olan 5 veri dosyası vardır, diğer çözüm yöntemlerinde elde edilen değerler optimaldir. Bu sebeple çevrim zamanı önceliklendirilmiş çözüm yönteminde optimal çözümlere ulaşmak diğer yöntemlere göre zordur.

Tablo 5.1. Önceliklendirilmiş Çözüm Yönteminin CPLEX OPL Ortalama Çözüm Süreleri (sn)

Veri#	Toplam Makine Sayısı Önceliklendirilmiş Çözüm Yöntemi	Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemi
Veri 1	39,83	127,24
Veri 2	22,07	3,70
Veri 3	122,48	6.893,60
Veri 4	23,22	3.043,60*
Veri 5	3,72	5.555,10
Veri 6	4,39	21.892,60
Veri 7	1,95	1.369,22
Veri 8	3,04	326,30
Veri 9	3,81	1.963,40
Veri 10	1,93	1648,01*
Veri 11	95,86	5.229,17
Veri 12	98,11	2.180,65*
Veri 13	2,19	755,11
Veri 14	1.965,59	12.477,73*
Veri 15	184,95	298,44
Veri 16	130,68	1.062,65
Veri 17	52,94	2.224,72*
Veri 18	91,53	2.591,22*
Veri 19	75,96	415,44
Veri 20	239,88	534,27
Veri 21	606,13	1.282,50
Veri 22	271,52	494,62

* Çözüm toleransı %0,05 olan çözümlerin OPL süresi

Tablo 5.2’de Pareto optimal çözümlerin belirlenmesi yöntemi ile CPLEX OPL’de çözdürülen 22 adet veri dosyalarına ait toplam çözüm süreleri verilmiştir. Pareto optimal çözümlerin belirlenmesi yönteminde sonuca ulaşabilmek için birden fazla model çözdürülmektedir. Bu sebeple tüm çözdürülen modellerin çözüm süreleri toplanmış ve Tablo 5.2’de verilmiştir.

Tablo 5.2. Pareto Optimal Çözümlerin Belirlenmesi Yönteminin CPLEX OPL Çözüm Süreleri (sn)

Veri#	Çevrim Zamanını Parametrize Ederek Pareto Optimal Çözümleri Bulma Yöntemi	Toplam Makine Sayısını Parametrize Ederek Pareto Optimal Çözümleri Bulma Yöntemi
Veri 1	188,22	1257,77
Veri 2	24,92	746,06
Veri 3	157,74	7857,45
Veri 4	3309,03	1002,34
Veri 5	60,77	19,06
Veri 6	522,51	183,83
Veri 7	36,45	31.940,95
Veri 8	81,06	30.220,37
Veri 9	131,14	276,14
Veri 10	124,44	7.689,05
Veri 11	326,46	215,27
Veri 12	461,99	1.046,83
Veri 13	23,42	10,64
Veri 14	1818,85	15.244,11
Veri 15	345,13	2.796,95
Veri 16	4368,05	15.719,51
Veri 17	48,32	9.563,13
Veri 18	84,86	1.192,09
Veri 19	640,62	8.199,30
Veri 20	3391,12	2.523,60
Veri 21	19665,13	4.426,70
Veri 22	1385,28	2.099,84

Tablo 5.2’den görüldüğü üzere başatlanmayan çözüm kümelerinin bulunması için uygulanan çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yönteminde parametre olarak verilen çevrim zamanının yüksek değerleri için çok hızlı optimal çözümlere ulaşılmış, çevrim zamanı değeri azaltıldıkça optimal makine sayısının bulunması zorlaştığı için çözüm süreleri uzamıştır. Bu yöntemin 22 veri dosyası için ortalama çözüm sürelerini küçük çevrim zamanı üst limiti değeri ile çözdürülen modellerin çözüm süreleri artırmıştır.

Başatlanmayan çözüm kümelerinin bulunması için uygulanan ikinci yöntem olan toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yönteminde parametre olarak verilen makine sayısının küçük değerleri için çok hızlı optimal çözümlere ulaşılmış, makine sayısı değeri arttıkça optimal çevrim zamanının bulunması zorlaştığı için çözüm süreleri uzamıştır. Bu yöntemin 22 veri dosyası için ortalama çözüm sürelerini yüksek makine sayısı parametresi ile çözdürülen modellerin çözüm süreleri artırmıştır.

Çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yönteminde başatlanmayan çözümlerin elde edilebilmesi için toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yöntemine göre daha çok model çözdürülmesi gerekmiştir. Çünkü parametre olarak verilen çevrim zamanı değerleri için optimal makine sayısı birden fazla çözümde aynı çıkabilmektedir. Yani başatlanmayan çözüme ulaşana kadar o makine sayısı için başatlanmış çözümlerde elde edilebilmektedir. Ayrıca çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yönteminde çevrim zamanı aralıklarının belirlenmesi gerekmektedir. Çevrim zamanı aralıkları büyük olursa tüm başatlanmayan çözümler elde edilememektedir, optimal makine sayıları ardışık olmamakta yani bazı makine sayıları atlanmaktadır. Eğer çevrim zamanı aralıkları küçük seçilirse bu seferde çok fazla model çözdürülerek değer katmayan başatlanan çözümler bulunmaktadır.

Toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yönteminde ise her iterasyonda bir tane başatlanmayan çözüm elde edilmekte bu da toplam çözüm süresinin kısılmasını, daha etkin çalışılmasını sağlamaktadır.

Çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yönteminde çevrim zamanı üst limit değeri belirli aralıklarla azaltılmaktadır. Ancak bu aralıkların birer birer azaltılmamasından kaynaklı toplam makine sayısı için elde edilen başatlanmamış çözümler en iyi çözüm olmayabilmektedir. Örneğin Veri-11 için çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yönteminde çevrim zamanı 261 iken makine sayısı 2 olarak elde edilmiştir. Ancak toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yönteminde makine

sayısı 2 için bulunan çevrim zamanı değeri 259'dur. Bu da çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yönteminde çevrim zamanı üst limitinin tek tek azaltılmamasından kaynaklı gerçek başatlanmayan çözümlerin bulunmama riski olduğunu göstermektedir. Ancak çevrim zamanı üst limitinin tek tek azaltılmasının da çok fazla sayıda başatlanan çözümler bulunmasına sebep olacaktır.

Çalışılan matematiksel modeller ve onların çözüm yöntemleri gerçek hayat problemlerine uygun olarak oluşturulmuştur. Çözüm yöntemlerinin geliştirilmesinde örnek alınan gerçek hayat problemleri aşağıdaki gibidir:

- **Toplam Makine Sayısı Önceliklendirilmiş Çözüm Yöntemi:** Yeni bir tesis kuran orta ölçekli bir firma yatırım miktarının sınırlı olması sebebiyle alacağı makine sayısını kısıtlamak zorundadır. Ancak aynı zamanda gelişen teknoloji ile beraber artan rekabet koşullarına uymak zorunda bu sebeple montaj hattında insan gücü yerine kalite ve güvenliğin artırılması amacıyla robot kullanmalı aynı zamanda bir birim ürün üretimi için geçen süreyi yani çevrim zamanını azaltmak zorundadır. Bu orta ölçekli firmanın parametreleri ile çalıştıracağı model **Toplam Makine Sayısı Önceliklendirilmiş Çözüm Yöntemidir.**
- **Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemi:** Yeni tesis kuran büyük ölçekli bir firma hatta kuracağı makine sayısını belirlerken en az maliyet oluşturacak şekilde planlama yapmalıdır. Ancak büyük ölçekli firmaların en önemli önceliği birim zamanda ürettiği ürün sayısıdır. Bu sebeple öncelikli olarak çevrim zamanını azaltmalı ve piyasadaki rekabet koşullarına uymalı ardından hedef çevrim zamanına uygun en az makine sayısını bulmalıdır. Aynı zamanda üretimde kalite, hız ve güvenliğin artması için robot kullanması gerektiğinden büyük ölçekli firmanın parametreleri ile çalıştıracağı model **Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemidir.**

- Çevrim Zamanını Parametrize Ederek Pareto Optimal Çözümleri Bulma Yöntemi: Piyasadaki rekabet koşullarına göre üretim sürecinde iyileştirme yapmak isteyen bir firma bir birim ürünün üretilmesi için gerekli olan çevrim zamanı ve toplam makine sayısını belirlemek istemektedir. Firmanın kullanacağı makine sayısında esnekliği vardır. Çevrim zamanı değerlerine karşılık gelen optimal toplam makine sayısı değerlerini belirlemek istemektedir. Ayrıca robotik hücrede üretim yapacak olan firma robotun makine yükleme boşaltma ve makineler arasındaki hareket süresini belirledikten sonra elindeki üretim parametreleri ile beraber toplam makine sayısını çevrim zamanı ikililerini bulmak ve aralarından seçim yapmak için çevrim zamanını parametrize ederek Pareto optimal çözümleri bulma yöntemini kullanmalıdır.
- Toplam Makine Sayısını Parametrize Ederek Pareto Optimal Çözümleri Bulma Yöntemi: Piyasadaki rekabet koşullarına göre üretim sürecinde iyileştirme yapmak isteyen bir firma bir birim ürünün üretilmesi için gerekli olan çevrim zamanı ve toplam makine sayısını belirlemek istemektedir. Firmanın kullanacağı makine sayısında esnekliği vardır. Toplam makine sayısı değerlerine karşılık gelen optimal çevrim zamanı değerlerini belirlemek istemektedir. Ayrıca robotik hücrede üretim yapacak olan firma robotun makine yükleme boşaltma ve makineler arasındaki hareket süresini belirledikten sonra elindeki üretim parametreleri ile beraber optimal toplam makine sayısını çevrim zamanı ikililerini bulmak ve aralarından seçim yapmak için toplam makine sayısını parametrize ederek Pareto optimal çözümleri bulma yöntemini kullanmalıdır.

Yukarıda verilen gerçek hayat problemleri örnekleri yeni kurulan veya kurulmuş ve tesis alanı ve planlaması belirli firmalara aittir. Bu örnekler gerçek hayatta sıklıkla karşılaşılan ve tasarım planlaması aşamasında karar verilmesi zor kararlara aittir. Geliştirilen matematiksel modeller gerçek hayat problemlerini ve günümüzde gelişen teknoloji ile beraber kullanılmaya başlanan robotik üretim hatlarını ele almaktadır. Montaj hattı dengeleme uzun yıllardır çalışılan bir problem olup robotik hücreler de

günümüzde sıklıkla karşılaşılan üretim hatlarındandır. Robotik hücrelerde kullanılan robotlar kalite, güvenlik ve üretim hızını artırmaktadır ancak en iyi çevrim zamanının, en iyi makine sayısının belirlenmesi problemi bu tip hücreler için de geçerlidir. Bu sebeple bu çalışmada çeşitli problemler için matematiksel modeller geliştirilmiş ve çözüm yöntemleri bulunmuştur.

6. SONUÇ ve DEĞERLENDİRME

Bu çalışmada tek tip parça üretimi yapan, K adet işlemin ve çeşitli sayıda makinenin olduğu, makine yükleme boşaltma ve makineler arası parça taşıma işlerini tek tutuculu bir robotun yaptığı bir montaj hattı ele alınmıştır. Robot parça üzerinde hiçbir işlem yapmamaktadır. Parça üzerinde yapılması gereken işlemler arasında öncelik ilişkisi mevcuttur. Bu çalışmanın amaç fonksiyonu iki kriterlidir; hem çevrim zamanının hem de toplam makine sayısının en küçüklenmesi amaçlanmıştır. En küçük toplam makine sayısı ve çevrim zamanı değerlerinin elde edilmesi ve robotun hareket sırasının belirlenmesi bu çalışmanın sonuçlarıdır. Robot hareketleri için 1-birim döngüsü ele alınmıştır. Montaj hattında kullanılan robot bir birim ürünün üretimi için gerçekleştirdiği hareketleri her döngüde tekrarlamaktadır.

Montaj hattı dengeleme, robotik hücre çizelgeleme ve iki kriterli optimizasyon üzerine yapılmış olan literatür incelendikten sonra detaylı problem tanımı yapılmıştır. İki kriterli robotik montaj hattı dengeleme problemi için iki ayrı çözüm yöntemi geliştirilmiştir; bunlar önceliklendirilmiş çözüm yöntemi ve Pareto optimal çözümlerin belirlenmesi yöntemleridir. Ardından literatürden alınan ve rasgele üretilmiş işlem zamanlarını içeren 22 çeşit veri dosyası ile geliştirilen yöntemler CPLEX OPL 12.4 çözdürücüsünde çözdürülmüştür. Çevrim zamanı önceliklendirilmiş çözüm yönteminin diğer çözüm yöntemlerine göre daha uzun sürede çözümler verdiği görülmüştür. Pareto optimal çözümlerin belirlenmesi yönteminde ise önceliklendirilmiş çözüm yöntemine göre daha kapsamlı sonuçlar elde edilmiştir.

Robot kullanan montaj hattında hem çevrim zamanı hem de makine sayısı en küçüklenmesi daha önce çalışılmamış bir problem olup yapılan çalışma ile literatürde bu alandaki boşluk doldurulmuştur.

Bu çalışmanın çözüm yöntemlerinden olan çevrim zamanı önceliklendirilmiş çözüm yönteminde optimal sonuçlara ulaşmak çok uzun zaman almıştır. Bu yöntem için bir sezgisel model geliştirilip daha kısa sürede optimale yakın sonuçlar elde edilebilir.

Bu çalışmada kullanılan makineler tüm işlemleri yapabilme kabiliyetine sahiptir. Ancak parça üretiminde gerekli özel işlemlerin sadece bazı makinelerde yapılabileceği kısıtı ele alınarak bir çalışma geliştirilebilir. Ele alınan probleme atama kısıtları eklenerek işlemlerin belirli makinelere atanması sağlanabilir. Ancak atama kısıtları ile kurulan matematiksel model daha karmaşık olacağı için çözümleri elde etmek daha zor olabilir.

Bu çalışmada tek tip parça üretimi yapan montaj hattı ele alınmıştır. İleriki çalışmalarda tek tip değil farklı tip parça üretilen sistemlerin ele alınması durumunda robot hareket sıralaması, işlemlerin makinelere atanması sonuçlarının yanı sıra bir de parça sıralamasının belirlenmesi gerekecektir. Ayrıca tek tip parça üretiminde çok iyi sonuçlar veren 1-birim döngüleri farklı tip parça üretiminde o kadar iyi sonuçlar vermemektedir. Dolayısıyla sadece 1-birim döngüler değil k-birim döngüleri de ele alınmalıdır. Bu da problemin modellenmesini ve çözdürülmesini zorlaştıran önemli unsurlardır.

Ele alınan montaj hattında kullanılan robotlar sadece parça taşıma ve makine yükleme boşaltma işlemlerini yapmaktadır. Robotunda parça üzerinde işlem yapabildiği robotik montaj hattı dengeleme problemi iki kriterli olarak çalışılabilir. Ancak bu durumda robotun parça taşıma ve makine yükleme boşaltma sürelerinin yanı sıra parça işleme süresinin de eklenmesi ve işlem atamasının gerçekleştirilmesi modellemeyi ve çözümü zorlaştıracaktır.

Bu çalışma kapsamında ele alınan montaj hattında tek tutuculu bir robot kullanılmaktadır. Tek tutuculu yerine çift tutuculu bir robotun kullanıldığı montaj hattı üzerine çalışılabilir. Ancak çift tutuculu robotun kullanıldığı bir hücrede sadece 2 makine ile çalışıldığında bile döngü sayısı çok artmaktadır. Ayrıca çok tip parça üretimi yapıldığında ise problemin çözümü daha da zorlaşmaktadır. Bu sebeple bu tarz bir çalışmada model çözümünü elde etmek zor olacaktır.

KAYNAKÇA

- [1] Ağpak, K., Gökçen, H., Assembly line balancing: Two resource constrained cases. *Int J. Production Economics*, 96:129-140, 2005.
- [2] Aktürk, M.S., Gültekin, H. ve Karasan, O.E., Robotic cell scheduling with operational flexibility. *Discrete Applied Mathematics*, 145 (3):334-348, 2005.
- [3] Arcus, A.L., A computer method of sequencing operations for assembly lines. *International Journal of Production Research*, 4 (4):259-277, 1965.
- [4] Bautista, J., Pereira, J., A dynamic programming based heuristic for the assembly line balancing problem. *European Journal of Operational Research*, 194:787-794, 2009.
- [5] Becker, C., Scholl, A., A Survey on Problems ve Methods in Generalized Assembly Line Balancing. *European Journal of Operations Research*, 168:694-715, 2006.
- [6] Brauner, N., Finke, G., On the conjecture in robotic cells:new simplified proof for the three-machine cases. *INFOR*, 37:20-36, 1999.
- [7] Brauner, N., Finke, G., On cycles and permutations in robotic cells. *Mathematical and Computer Modelling*, 34:565-591, 2001.
- [8] Cheng, E., Janiak, A. ve Kovalyov, M., Bicriterion Single Machine Scheduling with Resource Dependent Processing Times. *SIAM Journal on Optimization*, 8(2), 617-630, 1998.
- [9] Crama, Y., Kats, Y., van de Klundert, J. ve Levner, E., Cyclic scheduling in robotic flowshops. *Annals of Operations Research*, 96:97-124, 2000.
- [10] Crama, Y., van de Klundert, J., Cyclic scheduling of identical parts in robotic cells. *Operations Research*, 45: 952-95, 1997.
- [11] Crama, Y., van de Klundert, J., Cyclic scheduling in 3-machine robotic flowshops. *Journal of Scheduling*, 4: 35-54, 1999.
- [12] Dawande, M., Sriskandarajah, C. ve Dosyahi, S.P., On throughput maximization in constant travel-time robotic cells. *Manufacturing and Service Operations Management*, 4 (4):296-312, 2002.
- [13] Feng, J., Che, A., Wang, N., Bi-objective cyclic scheduling in a robotic cell with processing time windows and non-Euclidean travel times. *International Journal of Production Research*, 52 (9): 2505-2518, 2014.

- [14] Geismar, H.N., Dawande, M., Sethi, S.P. ve Sriskandarajah, C., Sequencing and scheduling in robotic cells: recent development. *Journal of Scheduling*, 8:387-426, 2005.
- [15] Geismar, H.N., Dawande, M., Dosyahi, S.P. ve Sriskandarajah, C., A note on productivity gains in flexible robotic cells. *International Journal of Flexible Manufacturing Systems*, 17 (1):5-21, 2005.
- [16] Gültekin, H., Aktürk, M.S. ve Karasan, O.E., Bicriteria robotic cell scheduling. *Journal of Scheduling*, 11 (6): 457-473, 2008.
- [17] Gültekin, H., Aktürk, M.S. ve Karasan, O.E., Bicriteria robotic operation allocation in a flexible manufacturing cell. *Computers&Operations Research*, 37: 779-789, 2010.
- [18] Gürsoy, A., Nuriyev, U., A mathematical model for assembly line balancing problem with lean manufacturing. *The Third International Conference "Problems of Cybernetics and Informatics"*, Baku, Azerbaijan, September 6-8, 2010.
- [19] Helgeson, W.B., Birnie, D.P., Assembly line balancing using ranked positional weight technique. *Journal of Industrial Engineering*, 12:394-398, 1961.
- [20] Hoffman, T.R., Assembly line balancing with a precedence matrix. *Management Science*, 9 (4):551-562, 1963.
- [21] Hoogeveen, H., Multicriteria scheduling. *European Journal of Operational Research* 167:592-623, 2005.
- [22] Levitin, G., Rubinovitz, J. ve Shnits, B., A genetic algorithm for robotic assembly line balancing. *European Journal of Operational Research*, 168:811-825, 2006.
- [23] Nicosia, G.,Pacciarelli, D. ve Pacici, A., Optimally balancing assembly lines with different workstations. *Displied Applied Mathematics*, 118:99-113, 2002.
- [24] Pastor, R., Ferrer, L., An improved mathematical program to solve the simple assembly line balancing problem. *International Journal of Production Research*, 47 (11):2943-2959, 2009.
- [25] Peeters, M., Degraeve, Z., A linear programming based lower bound for the simple assembly line balancing problem. *European Journal of Operational Research*, 168:716-731, 2006.
- [26] Ritt, M., Costa, A.M., A comparison of formulations for the simple assembly line balancing problem. 2011.

- [27] Scholl, A., Becker, C., State-of-The-Art Exact and Heuristic Solution Procedures for Simple Assembly Line Balancing, *European Journal of Operational Research*, 168:666-693, 2004.
- [28] Scholl, A. Fliedner, M. ve Boysen, N., ABSALOM: Balancing assembly lines with assignment restrictions. *European Journal of Operational Research*, 200:688-701, 2010.
- [29] Scholl, A., Klein, R., SALOME: A bidirectional branch and bound procedure for assembly line balancing. *Inform. J. Computing*, 9:319-334, 1997.
- [30] Sethi, S.P., Sriskandarajah, C., Sorger, G., Blazewicz, J. ve Kubiak, W., Sequencing of parts and robot moves in robotic cells. *The International Journal of Flexible Manufacturing Systems*, 4: 331-358, 1992.
- [31] Üstünođ, S., Gültekin, H., Robotik Hücrelerde Hat Dengeleme. 2012.
- [32] Veri erişim adresi: www.assembly-line-balancing.de, erişim tarihi: 2 Şubat 2014.

EKLER

A. Test Sonuçları

Tablo A.1. Toplam Makine Sayısı Önceliklendirilmiş Çözüm Yöntemin Ayrıntılı Sonuçları

Veri#	Tekrar#	ϵ	δ	CPU (sn)	Sapma (%)	Makine Sayısı	Çevrim Zamanı
1	1	36	39	14,49	0	5	15456
	2	1	45	21,18	0	5	15328
	3	12	38	52,17	0	5	15344
	4	20	36	34,09	0	5	15368
	5	23	46	77,24	0	5	15418
2	1	5	62	13,21	0	4	46359
	2	36	57	31,12	0	4	46463
	3	63	45	13,15	0	4	46523
	4	25	47	17,18	0	4	46379
	5	2	53	35,68	0	4	46311
3	1	0	2	81,61	0	7	613
	2	0	1	73,30	0	7	609
	3	1	1	89,12	0	7	613
	4	1	2	187,01	0	7	617
	5	1	0	181,34	0	7	609
4	1	0	2	25,88	0	4	1416
	2	2	2	24,70	0	4	1425
	3	1	2	25,23	0	4	1421
	4	1	0	21,89	0	4	1413
	5	3	2	18,41	0	4	1429
5	1	2	0	4,13	0	4	90
	2	1	0	3,40	0	4	86
	3	2	1	3,98	0	4	94
	4	0	1	4,01	0	4	86
	5	1	1	3,07	0	4	90
6	1	0	1	5,71	0	4	124
	2	1	1	3,63	0	4	129
	3	0	2	2,20	0	3	169
	4	1	0	6,71	0	3	165
	5	2	1	3,71	0	3	173

Veri#	Tekrar#	ϵ	δ	CPU (sn)	Sapma (%)	Makine Sayısı	Çevrim Zamanı
7	1	18	0	1,81	0	3	4859
	2	9	7	2,11	0	3	4851
	3	11	0	2,23	0	3	4831
	4	33	3	1,65	0	3	4931
	5	43	11	1,95	0	3	5003
8	1	3	2	3,12	0	3	362
	2	2	2	3,54	0	3	358
	3	3	1	3,12	0	3	358
	4	0	2	3,34	0	3	350
	5	1	1	2,07	0	3	349
9	1	0	1	3,54	0	3	188
	2	2	1	5,01	0	3	196
	3	1	0	3,76	0	3	188
	4	2	0	2,61	0	3	192
	5	1	1	4,13	0	3	192
10	1	41	11	1,78	0	3	4984
	2	14	18	1,54	0	3	4904
	3	44	14	2,29	0	3	5008
	4	37	12	1,93	0	3	4972
	5	34	16	2,11	0	3	4976
11	1	0	1	36,92	0	7	74
	2	1	1	55,43	0	7	78
	3	2	0	137,92	0	7	78
	4	0	2	81,00	0	7	78
	5	2	1	168,03	0	7	82
12	1	2	2	54,27	0	5	860
	2	4	2	171,04	0	5	868
	3	0	2	100,32	0	5	852
	4	6	1	85,82	0	5	872
	5	5	1	79,08	0	5	868
13	1	1	1	1,64	0	3	49
	2	1	0	1,89	0	3	45
	3	0	1	2,06	0	3	46
	4	2	1	3,31	0	3	54
	5	0	2	2,04	0	3	49

Veri#	Tekrar#	ϵ	δ	CPU (sn)	Sapma (%)	Makine Sayısı	Çevrim Zamanı
14	1	14	9	2.358,78	0	7	10044
	2	9	12	3.623,40	0	7	10036
	3	13	7	1.875,77	0	7	10032
	4	22	7	832,58	0	7	10068
	5	7	6	1.137,4	0	7	10026
15	1	48	178	187,70	0	6	55782
	2	24	163	245,28	0	6	55620
	3	109	174	191,99	0	6	56010
	4	113	194	50,84	0	6	56102
	5	59	145	248,96	0	6	55694
16	1	1	3	61,75	0	7	759
	2	3	3	99,26	0	7	767
	3	2	3	56,40	0	7	763
	4	0	4	338,23	0	7	759
	5	2	1	97,75	0	7	755
17	1	19	212	33,70	0	5	122543
	2	94	226	25,37	0	5	122892
	3	110	240	56,15	0	5	123010
	4	15	188	63,20	0	5	122435
	5	52	178	86,30	0	5	122533
18	1	3	3	174,49	0	5	1465
	2	1	3	86,03	0	5	1457
	3	3	0	48,50	0	5	1453
	4	2	3	42,11	0	5	1461
	5	2	1	106,53	0	5	1453
19	1	19	220	109,15	0	6	69426
	2	51	191	75,93	0	6	69438
	3	32	172	73,99	0	6	69286
	4	10	199	73,12	0	6	69306
	5	28	170	47,61	0	6	69269
20	1	0	3	124,02	0	7	812
	2	1	4	161,09	0	7	820
	3	2	3	75,47	0	7	820
	4	3	3	723,55	0	7	824
	5	3	2	115,27	0	7	820

Veri#	Tekrar#	ϵ	δ	CPU (sn)	Sapma (%)	Makine Sayısı	Çevrim Zamanı
21	1	17	164	325,75	0	7	61457
	2	15	151	104,90	0	7	61399
	3	37	146	549,48	0	7	61459
	4	53	153	1.714,40	0	7	61558
	5	42	149	336,14	0	7	61498
22	1	1	3	191,52	0	7	786
	2	2	3	366,52	0	7	791
	3	1	0	121,12	0	7	775
	4	1	2	176,38	0	7	783
	5	0	2	502,08	0	7	779

Tablo A.2. Çevrim Zamanı Önceliklendirilmiş Çözüm Yöntemin Ayrıntılı Sonuçları

Veri#	Tekrar#	ε	δ	CPU (sn)	Sapma (%)	Makine Sayısı	Çevrim Zamanı
1	1	36	39	132,57	0	7	12156
	2	1	45	115,27	0	7	12040
	3	12	38	96,42	0	7	12056
	4	20	36	194,95	0	7	12080
	5	23	46	96,99	0	7	12132
2	1	5	62	3,51	0	4	46359
	2	36	57	3,68	0	4	46463
	3	63	45	3,49	0	4	46523
	4	25	47	4,15	0	4	46379
	5	2	53	3,67	0	4	46311
3	1	0	2	2.062,53	0	11	393
	2	0	1	14.238,19	0	11	389
	3	1	1	12.854,22	0	11	393
	4	1	2	3.625,47	0	11	397
	5	1	0	1.687,61	0	11	389
4	1	0	2	3.833,37	0,005	10	574
	2	2	2	1.841,75	0,005	10	580
	3	1	2	2.236,58	0,005	10	577
	4	1	0	3.021,80	0,005	10	568
	5	3	2	4.284,51	0,005	10	586
5	1	2	0	2.789,66	0	10	44
	2	1	0	1.431,30	0	10	38
	3	2	1	3.063,89	0	7	60
	4	0	1	179,05	0	10	40
	5	1	1	20.311,59	0	8	50
6	1	0	1	35.318,28	0	10	54
	2	1	1	2.033,67	0	9	62
	3	0	2	659,17	0	8	71
	4	1	0	687,02	0	10	54
	5	2	1	70.764,87	0	8	75
7	1	18	0	681,16	0	10	1847
	2	9	7	1.010,94	0	10	1839
	3	11	0	52,41	0	10	1819
	4	33	3	1.633,61	0	10	1919
	5	43	11	3.468,00	0	10	1991

Veri#	Tekrar#	ϵ	δ	CPU (sn)	Sapma (%)	Makine Sayısı	Çevrim Zamanı
8	1	3	2	61,67	0	7	183
	2	2	2	33,98	0	7	179
	3	3	1	176,39	0	7	179
	4	0	2	1.343,61	0	8	170
	5	1	1	15,83	0	7	171
9	1	0	1	149,09	0	10	60
	2	2	1	8.813,16	0	9	76
	3	1	0	411,06	0	10	60
	4	2	0	193,16	0	10	64
	5	1	1	250,53	0	10	64
10	1	41	11	1.686,63	0,005	10	1734
	2	14	18	1.168,33	0,005	10	1654
	3	44	14	2.457,08	0,005	10	1758
	4	37	12	1.623,55	0,005	10	1722
	5	34	16	2.304,45	0,005	10	1726
11	1	0	1	577,25	0	10	53
	2	1	1	3.295,26	0	9	62
	3	2	0	5.619,95	0	10	57
	4	0	2	6.123,89	0	10	60
	5	2	1	10.529,51	0	10	66
12	1	2	2	2.314,30	0,005	10	440
	2	4	2	1.160,94	0,005	10	448
	3	0	2	1.577,30	0,005	10	432
	4	6	1	3.591,92	0,005	10	452
	5	5	1	2.258,78	0,005	10	448
13	1	1	1	596,70	0	5	34
	2	1	0	91,70	0	8	20
	3	0	1	1.787,70	0	6	25
	4	2	1	1.230,89	0	4	44
	5	0	2	68,55	0	4	40
14	1	14	9	10.344,11	0,005	12	5962
	2	9	12	11.131,87	0,005	12	5955
	3	13	7	11.221,89	0,005	12	5952
	4	22	7	12.630,14	0,005	12	5987
	5	7	6	17.060,66	0,005	12	5927

Veri#	Tekrar#	ϵ	δ	CPU (sn)	Sapma (%)	Makine Sayısı	Çevrim Zamanı
15	1	48	178	277,62	0	11	30994
	2	24	163	173,27	0	11	30792
	3	109	174	340,05	0	11	31195
	4	113	194	451,16	0	11	31275
	5	59	145	250,09	0	11	30870
16	1	1	3	991,58	0	10	566
	2	3	3	1.181,28	0	10	574
	3	2	3	695,72	0	10	570
	4	0	4	2.222,53	0	10	556
	5	2	1	222,12	0	10	562
17	1	19	212	1.967,94	0,005	11	50396
	2	94	226	3.101,59	0,005	11	50803
	3	110	240	2.171,80	0,005	11	50909
	4	15	188	1.676,45	0,005	11	50294
	5	52	178	2.205,80	0,005	11	50433
18	1	3	3	1.384,42	0,005	10	745
	2	1	3	1.271,09	0,005	10	738
	3	3	0	4.608,59	0,005	10	733
	4	2	3	3.368,02	0,005	10	742
	5	2	1	2.323,98	0,005	10	734
19	1	19	220	433,19	0	10	42615
	2	51	191	263,14	0	10	42623
	3	32	172	868,88	0	10	42354
	4	10	199	352,61	0	10	42493
	5	28	170	159,39	0	10	42424
20	1	0	3	308,14	0	10	573
	2	1	4	537,99	0	10	581
	3	2	3	345,61	0	10	581
	4	3	3	157,69	0	10	585
	5	3	2	1.321,91	0	10	581
21	1	17	164	1.129,58	0	11	43404
	2	15	151	1.424,94	0	11	43288
	3	37	146	1.115,74	0	11	43441
	4	53	153	1.507,56	0	11	43399
	5	42	149	1.234,67	0	11	43456

Veri#	Tekrar#	ϵ	δ	CPU (sn)	Sapma (%)	Makine Sayısı	Çevrim Zamanı
22	1	1	3	703,64	0	10	558
	2	2	3	519,48	0	10	562
	3	1	0	310,06	0	10	546
	4	1	2	583,92	0	10	554
	5	0	2	356,00	0	10	550

Tablo A.3. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-1

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	75723	0	0,56
2	73229	0	0,51
2	70735	0	0,72
2	68241	0	0,81
2	65747	0	0,7
2	63253	0	1
2	60759	0	0,69
2	58265	0	0,84
2	55771	0	1,03
2	53277	0	0,91
2	50783	0	0,66
2	48289	0	0,78
2	45795	0	0,69
2	43301	0	0,87
2	40807	0	0,92
2	38313	0	0,67
3	35819	0	3,53
3	33325	0	2,95
3	30831	0	1,44
3	28337	0	0,92
3	25843	0	0,9
4	23349	0	8,16
4	20855	0	32,48
5	18361	0	58,63
5	15867	0	7,46
6	13373	0	59,39
	10879	0	Olursuz!

Tablo A.4. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-2

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	150415	0	0,51
2	145446	0	0,64
2	140477	0	1,01
2	135508	0	0,84
2	130539	0	1,03
2	125570	0	0,8
2	120601	0	0,76
2	115632	0	0,7
2	110663	0	0,97
2	105694	0	0,97
2	100725	0	0,64
2	95756	0	0,97
2	90787	0	1,03
2	85818	0	0,92
2	80849	0	0,98
2	75880	0	0,95
3	70911	0	2,84
3	65942	0	2,73
3	60973	0	2,15
4	56004	0	1,94
4	51035	0	1,54
	46066	0	Olursuz!

Tablo A.5. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-3

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	4250	0	0,51
2	4109	0	0,62
2	3968	0	0,64
2	3827	0	0,64
2	3686	0	0,64
2	3545	0	0,64
2	3404	0	1
2	3263	0	0,62
2	3122	0	0,64
2	2981	0	0,61
2	2840	0	0,64
2	2699	0	1,03
2	2558	0	0,8
2	2417	0	0,66
2	2276	0	0,72
2	2135	0	1
3	1994	0	2,06
3	1853	0	2,01
3	1712	0	2,12
3	1571	0	3,6
3	1430	0	2,28
4	1289	0	5,02
4	1148	0	3,6
5	1007	0	7,04
5	866	0	6,16
6	725	0	7,14
8	584	0	49,69
10	443	0	55,61
	302	0	Olursuz!

Tablo A.6. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-4

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	5650	0	1,01
2	5463	0	1,48
2	5276	0	1,03
2	5089	0	1,03
2	4902	0	1,03
2	4715	0	1,03
2	4528	0	1,03
2	4341	0	1,47
2	4154	0	1,48
2	3967	0	1,03
2	3780	0	1,56
2	3593	0	1,03
2	3406	0	1,03
2	3219	0	1,5
2	3032	0	1,48
2	2845	0	1,11
3	2658	0	2,61
3	2471	0	3,09
3	2284	0	2,65
3	2097	0	3,09
3	1910	0	3,09
4	1723	0	9,38
4	1536	0	14,96
5	1349	0	16,15
5	1162	0	16,25
6	975	0	51,89
8	788	0	109,26
10	601	0	3057,28
	414	0	Olursuz!

Tablo A.7. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-5

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	340	0	0,52
2	329	0	0,61
2	318	0	0,64
2	307	0	0,62
2	296	0	0,72
2	285	0	0,59
2	274	0	0,77
2	263	0	0,62
2	252	0	0,59
2	241	0	0,61
2	230	0	0,62
2	219	0	0,61
2	208	0	0,61
2	197	0	0,61
2	186	0	0,61
3	175	0	1,44
3	164	0	1,68
3	153	0	1,26
3	142	0	2,45
3	131	0	1,54
4	120	0	2,62
4	109	0	2,79
4	98	0	4,18
5	87	0	21,82
6	76	0	11,64
	65	0	Olursuz!

Tablo A.8. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-6

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	499	0	0,52
2	483	0	0,62
2	467	0	0,62
2	451	0	0,64
2	435	0	0,64
2	419	0	0,61
2	403	0	0,76
2	387	0	0,62
2	371	0	0,76
2	355	0	0,62
2	339	0	0,62
2	323	0	0,61
2	307	0	0,62
2	291	0	0,61
2	275	0	0,59
2	259	0	1,02
3	243	0	2,01
3	227	0	0,67
3	211	0	2,57
3	195	0	1,2
3	179	0	2,06
4	163	0	2,46
4	147	0	5,82
5	131	0	4,17
5	115	0	3,84
7	99	0	128,78
8	83	0	358,45
	67	0	Olursuz!

Tablo A.9. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-7

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	14042	0	0,56
2	13583	0	0,61
2	13124	0	0,52
2	12665	0	0,51
2	12206	0	0,51
2	11747	0	0,64
2	11288	0	0,75
2	10829	0	0,64
2	10370	0	0,65
2	9911	0	0,69
2	9452	0	0,64
2	8993	0	0,64
2	8534	0	0,66
2	8075	0	0,61
2	7616	0	0,62
2	7157	0	0,64
3	6698	0	2,56
3	6239	0	0,7
3	5780	0	0,73
3	5321	0	0,75
3	4862	0	0,83
4	4403	0	0,69
4	3944	0	0,67
5	3485	0	4,68
5	3026	0	1,03
6	2567	0	1,64
8	2108	0	12,28
	1649	0	Olursuz!

Tablo A.10. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-8

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	1040	0	0,52
2	1007	0	0,61
2	974	0	0,61
2	941	0	0,61
2	908	0	0,59
2	875	0	0,61
2	842	0	0,59
2	809	0	0,62
2	776	0	0,61
2	743	0	0,64
2	710	0	0,59
2	677	0	0,61
2	644	0	1,03
2	611	0	0,59
2	578	0	0,59
2	545	0	0,62
3	512	0	1,28
3	479	0	1,53
3	446	0	1,54
3	413	0	1,15
3	380	0	1,83
4	347	0	2,32
4	314	0	2,78
4	281	0	2,25
5	248	0	3,09
6	215	0	8,88
7	182	0	44,37
	149	0	Olursuz!

Tablo A.11. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-9

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	568	0	0,55
2	550	0	0,51
2	532	0	0,52
2	514	0	0,56
2	496	0	0,52
2	478	0	0,56
2	460	0	0,51
2	442	0	0,51
2	424	0	0,51
2	406	0	0,59
2	388	0	0,52
2	370	0	0,58
2	352	0	0,51
2	334	0	0,64
2	316	0	0,62
2	298	0	1,03
3	280	0	1,23
3	262	0	1,75
3	244	0	2,54
3	226	0	2,82
3	208	0	2,26
4	190	0	7,21
4	172	0	3,87
4	154	0	4,96
5	136	0	5,79
6	118	0	27,8
7	100	0	61,67
	82	0	Olursuz!

Tablo A.12. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-10

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	14156	0	0,51
2	13699	0	0,64
2	13242	0	0,69
2	12785	0	0,64
2	12328	0	0,64
2	11871	0	0,7
2	11414	0	0,65
2	10957	0	0,64
2	10500	0	0,62
2	10043	0	0,59
2	9586	0	0,62
2	9129	0	0,61
2	8672	0	0,59
2	8215	0	0,61
2	7758	0	0,59
2	7301	0	0,61
3	6844	0	1,54
3	6387	0	1,73
3	5930	0	0,7
3	5473	0	1,54
3	5016	0	1,03
4	4559	0	3,74
4	4102	0	1,62
4	3645	0	0,65
5	3188	0	1,86
6	2731	0	3,09
7	2274	0	5,18
9	1817	0	91,81
	1360	0	Olursuz!

Tablo A.13. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-11

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	501	0	0,58
2	485	0	0,59
2	469	0	0,61
2	453	0	0,59
2	437	0	0,59
2	421	0	0,51
2	405	0	0,75
2	389	0	0,75
2	373	0	0,75
2	357	0	0,67
2	341	0	0,75
2	325	0	0,67
2	309	0	0,59
2	293	0	0,67
2	277	0	0,59
2	261	0	0,64
3	245	0	5,21
3	229	0	3,17
3	213	0	5,12
3	197	0	0,76
3	181	0	0,95
4	165	0	16,69
4	149	0	15,63
5	133	0	32,54
5	117	0	31,06
6	101	0	205,03
	85	0	Olursuz!

Tablo A.14. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-12

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	4224	0	0,52
2	4085	0	0,7
2	3946	0	0,78
2	3807	0	0,73
2	3668	0	0,95
2	3529	0	0,75
2	3390	0	1,03
2	3251	0	0,7
2	3112	0	0,72
2	2973	0	0,75
2	2834	0	0,67
2	2695	0	0,72
2	2556	0	0,66
2	2417	0	0,69
2	2278	0	1,03
2	2139	0	0,62
3	2000	0	3,6
3	1861	0	6,61
3	1722	0	2,57
3	1583	0	6,82
3	1444	0	2,57
4	1305	0	11,05
4	1166	0	16,57
5	1027	0	43,74
5	888	0	108,02
6	749	0	20,34
8	610	0	160,7
10	471	0	67,38
	332	0	Olursuz!

Tablo A.15. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-13

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	141	0	0,51
2	137	0	0,59
2	133	0	0,58
2	129	0	0,61
2	125	0	0,58
2	121	0	0,59
2	117	0	0,62
2	113	0	0,62
2	109	0	0,61
2	105	0	0,59
2	101	0	0,61
2	97	0	0,61
2	93	0	0,61
2	89	0	0,59
2	85	0	0,62
2	81	0	0,61
3	77	0	0,98
3	73	0	1,31
3	69	0	1,59
3	65	0	1,53
3	61	0	1,03
4	57	0	3,12
4	53	0	4,31
	49	0	Olursuz!

Tablo A.16. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-14

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	69676	0	1,83
2	67361	0	3
2	65046	0	2,89
2	62731	0	4,43
2	60416	0	3,63
2	58101	0	3,92
2	55786	0	3,54
2	53471	0	3,25
2	51156	0	2,56
2	48841	0	2,53
2	46526	0	2,14
2	44211	0	2,28
2	41896	0	2,28
2	39581	0	2,04
2	37266	0	2,11
2	34951	0	2,06
3	32636	0	7,39
3	30321	0	9,33
3	28006	0	3,35
3	25691	0	10,8
3	23376	0	5,13
4	21061	0	118,73
4	18746	0	59,5
5	16431	0	437,01
5	14116	0	82,37
6	11801	0	410,8
8	9486	0	276,89
10	7171	0	353,06
	4856	0	Olursuz!

Tablo A.17. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-15

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	329220	0	0,51
2	318378	0	0,67
2	307536	0	1
2	296694	0	0,64
2	285852	0	0,62
2	275010	0	1,03
2	264168	0	0,78
2	253326	0	0,8
2	242484	0	0,64
2	231642	0	0,66
2	220800	0	0,64
2	209958	0	0,67
2	199116	0	0,67
2	188274	0	0,64
2	177432	0	0,76
2	166590	0	0,65
3	155748	0	2
3	144906	0	1,53
3	134064	0	3,71
3	123222	0	3,09
3	112380	0	2,06
4	101538	0	7,38
4	90696	0	5,84
5	79854	0	11,84
5	69012	0	21,73
6	58170	0	32,93
7	47328	0	20,67
10	36486	0	220,97
	25644	0	Olursuz!

Tablo A.18. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-16

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	5211	0	0,52
2	5039	0	0,94
2	4867	0	0,64
2	4695	0	0,61
2	4523	0	0,67
2	4351	0	0,98
2	4179	0	0,83
2	4007	0	0,87
2	3835	0	0,7
2	3663	0	0,89
2	3491	0	0,69
2	3319	0	0,62
2	3147	0	0,69
2	2975	0	1,09
2	2803	0	1
2	2631	0	0,66
3	2459	0	3,09
3	2287	0	3,21
3	2115	0	4,12
3	1943	0	1,98
3	1771	0	4,84
4	1599	0	32,15
4	1427	0	5,2
5	1255	0	36,58
5	1083	0	20,08
6	911	0	37,97
8	739	0	242,19
10	567	0	3964,24
	395	0	Olursuz!

Tablo A.19. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-17

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	542556	0	0,83
2	524634	0	0,75
2	506712	0	0,66
2	488790	0	0,86
2	470868	0	0,98
2	452946	0	1,08
2	435024	0	1,01
2	417102	0	0,84
2	399180	0	0,72
2	381258	0	0,7
2	363336	0	0,75
2	345414	0	1,03
2	327492	0	0,62
2	309570	0	0,62
2	291648	0	0,91
2	273726	0	0,87
3	255804	0	1,89
3	237882	0	2,57
3	219960	0	4,48
3	202038	0	1,39
3	184116	0	0,89
4	166194	0	4,1
4	148272	0	3,59
6	130350	0	16,18
	112428	0	Olursuz!

Tablo A.20. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-18

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	7203	0	0,83
2	6965	0	0,9
2	6727	0	0,87
2	6489	0	0,72
2	6251	0	0,9
2	6013	0	0,83
2	5775	0	0,73
2	5537	0	1,04
2	5299	0	0,75
2	5061	0	0,72
2	4823	0	0,7
2	4585	0	0,7
2	4347	0	0,69
2	4109	0	0,86
2	3871	0	0,86
2	3633	0	0,95
3	3395	0	2,18
3	3157	0	3,09
3	2919	0	2,06
3	2681	0	2,06
3	2443	0	1,17
4	2205	0	8,85
4	1967	0	5,18
5	1729	0	17,58
5	1491	0	29,64
	1253	0	Olursuz!

Tablo A.21. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-19

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	410543	0	0,52
2	397012	0	0,64
2	383481	0	0,69
2	369950	0	0,64
2	356419	0	0,62
2	342888	0	0,69
2	329357	0	0,69
2	315826	0	0,69
2	302295	0	0,67
2	288764	0	0,67
2	275233	0	0,69
2	261702	0	0,69
2	248171	0	0,7
2	234640	0	0,69
2	221109	0	0,67
2	207578	0	0,66
3	194047	0	2,57
3	180516	0	2,43
3	166985	0	1,73
3	153454	0	1,03
3	139923	0	1,03
4	126392	0	28,44
4	112861	0	5,63
5	99330	0	4,4
5	85799	0	3,17
6	72268	0	7,75
8	58737	0	227,79
10	45206	0	344,03
	31675	0	Olursuz!

Tablo A.22. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-20

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	5593	0	0,58
2	5409	0	0,51
2	5225	0	0,66
2	5041	0	0,66
2	4857	0	0,61
2	4673	0	0,59
2	4489	0	0,64
2	4305	0	0,66
2	4121	0	0,66
2	3937	0	0,7
2	3753	0	0,84
2	3569	0	0,67
2	3385	0	0,72
2	3201	0	0,59
2	3017	0	0,64
2	2833	0	0,67
3	2649	0	3,73
3	2465	0	3,71
3	2281	0	6,35
3	2097	0	0,72
3	1913	0	0,73
4	1729	0	8,61
4	1545	0	10,64
5	1361	0	19,97
5	1177	0	14,06
6	993	0	22,5
8	809	0	593,19
9	625	0	2696,51
	441	0	Olursuz!

Tablo A.23. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-21

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	425074	0	0,52
2	411056	0	0,62
2	397038	0	0,8
2	383020	0	0,79
2	369002	0	0,62
2	354984	0	0,62
2	340966	0	0,62
2	326948	0	0,62
2	312930	0	0,62
2	298912	0	1
2	284894	0	0,62
2	270876	0	0,64
2	256858	0	0,62
2	242840	0	0,66
2	228822	0	0,64
2	214804	0	0,64
3	200786	0	2,17
3	186768	0	2,11
3	172750	0	1,54
3	158732	0	1,56
3	144714	0	2,01
4	130696	0	20,12
4	116678	0	1,03
5	102660	0	3,84
5	88642	0	7,32
6	74624	0	12,07
8	60606	0	52,68
10	46588	0	19548,03
	32570	0	Olursuz!

Tablo A.24. Çevrim Zamanını Parametrize Ederek Bulunan Pareto Optimal Çözümler - Veri-22

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	5404	0	0,55
2	5226	0	0,75
2	5048	0	0,81
2	4870	0	0,75
2	4692	0	0,79
2	4514	0	1,01
2	4336	0	0,69
2	4158	0	1,03
2	3980	0	0,51
2	3802	0	0,59
2	3624	0	0,76
2	3446	0	1,03
2	3268	0	0,61
2	3090	0	0,61
2	2912	0	0,7
2	2734	0	0,62
3	2556	0	3,87
3	2378	0	7,04
3	2200	0	8,72
3	2022	0	4,87
3	1844	0	6,43
4	1666	0	16,38
4	1488	0	11,22
5	1310	0	73,8
5	1132	0	15,43
6	954	0	34,99
8	776	0	708,79
10	598	0	481,93
	420	0	Olursuz!

Tablo A.25. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-1

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	75723	0	0
2	37872	0	1,61
3	25254	0	2,14
4	18944	0	6,54
5	15158	0	110,03
6	12639	0	132,55
7	11872	0	6,6
8	11872	0	80,14
9	11872	0	251,23
10	11872	0	282,64
11	12058	0	384,29

Tablo A.26. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-2

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	150415	0	0
2	75216	0	2,06
3	56651	0	1,03
4	46107	0	1,64
5	46107	0	3,34
6	46107	0	4,6
7	46107	0	6,86
8	46107	0	8,08
9	46107	0	5,55
10	46107	0	12,84
11	46107	0	35,93
12	46107	0	46,91
13	46107	0	23,4
14	46107	0	77,89
15	46107	0	110,06
16	46107	0	63,87
17	46107	0	126,16
18	46115	0	215,84

Tablo A.27. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-3

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	4250	0	0
2	2133	0	1,15
3	1428	0	1,72
4	1075	0	2,42
5	863	0	5,29
6	722	0	5,09
7	621	0	10,73
8	546	0	24,54
9	487	0	79,33
10	440	0	133,07
11	402	0	1359,64
12	402	0	1895,28
13	402	0	1825,35
14	406	0	2513,84

Tablo A.28. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-4

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	5650	0	0
2	2833	0	1,17
3	1894	0	1,92
4	1425	0	3,07
5	1143	0	3,39
6	955	0	12,67
7	821	0	11,23
8	721	0	26,96
9	643	0	117,17
10	581	0	162,32
11	529	0	317,82
12	535	0	344,62

Tablo A.29. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-5

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	340	0	0
2	178	0	1,15
3	124	0	1,12
4	98	0	1,79
5	81	0	2,15
6	76	0	3,88
7	84	0	8,97

Tablo A.30. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-6

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	499	0	0
2	258	0	0,64
3	177	0	1,04
4	137	0	1,89
5	113	0	72,01
6	100	0	8,24
7	88	0	5,63
8	96	0	94,38

Tablo A.31. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-7

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	14042	0	0
2	7030	0	1,61
3	4803	0	1,61
4	3693	0	2,17
5	2839	0	1,9
6	2416	0	3,6
7	2352	0	5,76
8	1923	0	9,89
9	1843	0	26,51
10	1791	0	29,2
11	1791	0	146,52
12	1791	0	1032,27
13	1791	0	4431,41
14	1791	0	5108,32
15	1791	0	4989,43
16	1791	0	5894,29
17	1807	0	10256,46

Tablo A.32. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-8

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	1040	0	0
2	528	0	1,54
3	358	0	1,84
4	272	0	1,81
5	221	0	2,17
6	187	0	3,7
7	179	0	5,05
8	179	0	30,65
9	179	0	2185,38
10	179	0	2056,63
11	179	0	2156,81
12	179	0	3251,03
13	179	0	5683,52
14	179	0	6214,46
15	188	0	8625,78

Tablo A.33. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-9

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	568	0	0
2	292	0	1,67
3	200	0	1,65
4	154	0	2,43
5	127	0	3,18
6	108	0	235,75
7	95	0	11,09
8	96	0	20,37

Tablo A.34. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-10

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	14156	0	0
2	7092	0	1,03
3	4792	0	1,04
4	3590	0	2,06
5	2888	0	2,22
6	2412	0	2,92
7	2112	0	5,63
8	1876	0	13,62
9	1654	0	18,42
10	1542	0	71,09
11	1488	0	155,53
12	1416	0	184,75
13	1416	0	260,04
14	1416	0	325,62
15	1416	0	468,73
16	1416	0	827,96
17	1416	0	1004,49
18	1416	0	1394,53
19	1416	0	1904,05
20	1424	0	1045,32

Tablo A.35. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-11

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	501	0	0
2	259	0	1,62
3	178	0	1,95
4	137	0	2,34
5	114	0	2,95
6	97	0	4,48
7	88	0	7,72
8	96	0	194,21

Tablo A.36. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-12

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	4224	0	0
2	2120	0	1,09
3	1419	0	2,45
4	1068	0	4,21
5	860	0	5,8
6	720	0	15,33
7	637	0	27,14
8	548	0	41,59
9	487	0	115,52
10	440	0	280,38
11	447	0	553,32

Tablo A.37. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-13

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	141	0	0
2	79	0	1,14
3	58	0	1,56
4	52	0	2,43
5	60	0	5,51

Tablo A.38. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-14

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	69671	0	0
2	34844	0	1,89
3	23235	0	2,26
4	17430	0	5,85
5	13948	0	9,33
6	11626	0	76,55
7	9970	0	230,72
8	8725	0	427,76
9	7762	0	621,09
10	7002	0	383,92
11	6365	0	473,64
12	5841	0	2862,3
13	5857	0	10148,8

Tablo A.39. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-15

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	329220	0	0
2	164624	0	1,61
3	109760	0	2,62
4	82323	0	2,68
5	65863	0	25,72
6	54889	0	107,52
7	47050	0	590,68
8	41353	0	15,96
9	36764	0	41,98
10	33080	0	396,69
11	30086	0	360,53
12	30154	0	1250,96

Tablo A.40. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-16

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	5211	0	0
2	2614	0	1,78
3	1748	0	1,97
4	1315	0	2,48
5	1055	0	9,34
6	882	0	15,87
7	759	0	46,3
8	666	0	52,29
9	596	0	31,7
10	566	0	2239,67
11	566	0	2740,01
12	566	0	1896,21
13	566	0	2014,75
14	566	0	2131,62
15	566	0	3547,87
16	572	0	987,65

Tablo A.41. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-17

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	542556	0	0
2	271294	0	1,61
3	180872	0	2,17
4	135661	0	3,59
5	121630	0	3,17
6	90832	0	5,41
7	77874	0	11,09
8	68136	0	8,66
9	60572	0	23,28
10	54537	0	135,98
11	49542	0	621,63
12	45449	0	2314,61
13	45449	0	3450,81
14	45457	0	2981,12

Tablo A.42. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-18

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	7203	0	0
2	3610	0	1,7
3	2412	0	2,36
4	1813	0	2,62
5	1457	0	7,74
6	1440	0	5,63
7	1045	0	18,69
8	918	0	14,75
9	818	0	21
10	738	0	86,56
11	673	0	392,81
12	681	0	638,23

Tablo A.43. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-19

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	410543	0	0
2	205296	0	2,11
3	136872	0	2,04
4	102692	0	5,18
5	82362	0	3,77
6	68486	0	6,43
7	58880	0	89,65
8	51612	0	31,89
9	45707	0	535,77
10	41657	0	121,6
11	37810	0	979,17
12	37810	0	1247,63
13	37810	0	1561,21
14	37818	0	3612,85

Tablo A.44. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-20

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	5593	0	0
2	2805	0	0,61
3	1875	0	2,29
4	1411	0	2,95
5	1133	0	42,88
6	947	0	5,2
7	816	0	6,97
8	719	0	44,09
9	640	0	77,81
10	576	0	429,14
11	576	0	671,65
12	580	0	1240,01

Tablo A.45. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-21

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	425074	0	0
2	212547	0	0,67
3	141712	0	2,2
4	106288	0	2,31
5	85036	0	4,71
6	70866	0	14,15
7	60744	0	12,57
8	53297	0	18,27
9	47446	0	18,69
10	42703	0	103,58
11	38852	0	150,7
12	38600	0	825,53
13	38610	0	3273,32

Tablo A.46. Toplam Makine Sayısını Parametrize Ederek Bulunan Pareto Optimal Çözümler – Veri-22

Makine Sayısı	Çevrim Zamanı	Sapma (%)	CPLEX Süresi (sn)
1	5404	0	0
2	2710	0	1,78
3	1812	0	2,06
4	1363	0	2,73
5	1094	0	6,43
6	914	0	8,91
7	787	0	19,63
8	691	0	26,21
9	619	0	82,51
10	558	0	287,6
11	558	0	405,89
12	568	0	1256,09

B. Veri Dosyası Örneği – Veri-8

Literatürden alınan [32] veri dosyasının içeriği birebir kopyalanmış, yanlarına açıklamalar yazılmıştır.

28 – Toplam işlem sayısını göstermektedir.

70 – Alt alta 28 adet işlem süresi bulunmaktadır.

59

33

6

1

27

17

62

31

53

21

19

108

52

5

8

97

8

47

67

17

8

3

21

107

3

2

72

1,3 – İşlemler arasındaki öncelik ilişkilerini göstermektedir.

1,4

1,5

1,8

1,19

1,21

1,22

1,23

1,24

1,26

2,6

2,17

3,28

4,28

5,28

6,7

7,18

8,9

9,10

10,11

10,12

11,15

12,13

13,14

13,16

14,28

15,28

16,28

17,28

18,28

19,20

20,28

21,28

22,28

23,28

24,25

25,28

26,27

27,28

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : ATASEVEN, Elif Büşra
Uyruğu : T.C.
Doğum tarihi ve yeri : 18.08.1989 - Gölcük
Medeni hali : Bekar
Telefon : 0 (533) 682 98 58
e-mail : ebusraataseven@gmail.com.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Y.Lisans	TOBB Ekonomi ve Teknoloji Üniversitesi Endüstri Mühendisliği	2014
Lisans (Çift Anadal)	TOBB Ekonomi ve Teknoloji Üniversitesi Endüstri Mühendisliği	2012
Lisans (Anadal)	TOBB Ekonomi ve Teknoloji Üniversitesi Matematik	2012

İş Deneyimi

Yıl	Yer	Görev
2013-Halen	Türk Havacılık ve Uzay Sanayi A.Ş. – TAI	Endüstri Mühendisi
2012-2013	Türksat Uydu Haberleşme Kablo TV ve İşletme A.Ş	İş Analisti

Yabancı Dil

İngilizce, İspanyolca

Yayımlar

Ataseven, E.B., Gültekin, H., Robotlu Hücrede Montaj Hattı Dengeleme. 14. Üretim Araştırmaları Sempozyumu, İstanbul, Türkiye, Eylül 3-4-5, 2014.