

TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**ÇİFT TUTUCULU MALZEME TAŞIYICI ROBOTLU HÜCRELERDE
ENERJİ DUYARLI ÇİZELGELEME**

YÜKSEK LİSANS TEZİ
Nurdan EMİROĞLU

Endüstri Mühendisliği Anabilim Dalı

Tez Danışmanı: Doç. Dr. Hakan GÜLTEKİN
Eş Danışman: Prof. Dr. Sinan GÜREL

Temmuz 2019

Fen Bilimleri Enstitüsü Onayı

.....
Prof. Dr. Osman EROĞUL
Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

.....
Prof. Dr. Tahir HANALIOĞLU
Anabilimdalı Başkanı

TOBB ETÜ, Fen Bilimleri Enstitüsü'nün 161311047 numaralı Yüksek Lisans öğrencisi **Nurdan EMİROĞLU**'nin ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı "**ÇİFT TUTUCULU MALZEME TAŞIYICI ROBOTLU HÜCRELERDE ENERJİ DUYARLI ÇİZELGELEME**" başlıklı tezi **19 Temmuz 2019** tarihinde aşağıda imzaları olan jüri tarafından kabul edilmiştir.

Tez Danışmanı: **Doç. Dr. Hakan GÜLTEKİN**
TOBB Ekonomi ve Teknoloji Üniversitesi

Eş Danışman: **Prof. Dr. Sinan GÜREL**
Orta Doğu Teknik Üniversitesi

Jüri Üyeleri: **Doç. Dr. Ayşegül ALTIN KAYHAN (Başkan)**.....
TOBB Ekonomi ve Teknoloji Üniversitesi

Doç. Dr. Çağrı KOÇ
Ankara Sosyal Bilimler Üniversitesi

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, alıntı yapılan kaynaklara eksiksiz atıf yapıldığını, referansların tam olarak belirtildiğini ve ayrıca bu tezin TOBB ETÜ Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırlandığını bildiririm.

Nurdan EMİROĞLU

İMZA

ÖZET

Yüksek Lisans Tezi

ÇİFT TUTUCULU MALZEME TAŞIYICI ROBOTLU HÜCRELERDE ENERJİ DUYARLI ÇİZELGELEME

Nurdan Emiroğlu

TOBB Ekonomi ve Teknoloji Üniversitesi

Fen Bilimleri Enstitüsü

Endüstri Mühendisliği Anabilim Dalı

Tez Danışmanı: Doç. Dr. Hakan Gültekin

Eş Danışman: Prof. Dr. Sinan Gürel

Tarih: Temmuz 2019

Robotik hücreler, belirli sayıda makine ve bu makineler arası parça taşınmasını sağlayan bir elleçleme robotundan oluşan seri üretim sistemleridir. Sistemdeki robotlar aynı zamanda, makinelerin yükleme/boşaltma işlemlerini de gerçekleştirmektedir. Akademik literatürde ve iş dünyasında, robotik hücrelerde büyük çoğunlukla, üretim çizelgelerinin optimize edilmesinde sistemlerin çevrim zamanlarının minimizasyonuna odaklanılmaktadır. Bu amaçla paralel olarak, robot hareketlerinin maksimum hızda yapıldığı varsayılmakta, hızın dinamik olarak değiştirilmesiyle elde edilebilecek enerji tasarrufu göz önüne alınmamaktadır. Bu çalışma kapsamında, çift tutuculu robotların hareket hızlarının değiştirilebilir olmasından yola çıkılarak, robot hareket sıralaması ve robotun en uygun çalışma parametrelerinin belirlenmesi hedeflenmiştir. Diğer bir deyişle, problem enerji tüketimi ve üretim hızı amaçlarını eş zamanlı ele alan, iki kriterli bir yapıda incelenmiştir. Bu iki kriterli modelin çözümü için epsilon-kısıt yaklaşımı kullanılarak amaçlardan bir tanesi kısıt olarak yazılmıştır. Problem için doğrusal olmayan karma tamsayılı matematiksel model (MINLP) geliştirilmiştir. Modelin çözüm etkinliğini artırmak için problem karesel kısıtlı karma tamsayılı matematiksel model (MISOCP) olarak yeniden formüle edilmiştir. Yapılan bilgisayar deneyleri ile, MINLP

modelin iki ve daha çok makineli sistemlerde istenilen sürede optimal sonucu garanti edemediđi görölmüştür. MISOCP modelinin ise altı ve daha çok makineli sistemler için yetersiz kaldıđı görülerek, Etkin Çözüm Türetme Algoritması (ETA) adında sezgisel bir çözüm yöntemi geliştirilmiştir. Hesaplamalı çalışmalar yapılarak geliştirilen çözüm yöntemlerinin performansları oluşturulan veri kümeleri ile test edilmiştir. Ayrıca yapılan testler ile robot hareket hızlarının deđiştirilebilir olması durumunda ortalama %17.7 enerji tasarrufu sađlandığı gözlemlenmiştir.

Anahtar Kelimeler: Robotik hücre çizelgeleme, İki kriterli optimizasyon, Matematiksel modelleme, Sezgisel yöntemler



ABSTRACT

Master of Science

ENERGY CONSCIOUS SCHEDULING OF DUAL GRIPPER MATERIAL HANDLING ROBOTS IN ROBOTIC MANUFACTURING CELLS

Nurdan Emiroglu

TOBB University of Economics and Technology

Institute of Natural and Applied Sciences

Department of Industrial Engineering

Supervisor: Assoc. Prof. Hakan Gultekin

Co-Supervisor: Prof. Dr. Sinan Gurel

Date: July 2019

Robotic cells are serial production systems that consist of a number of machines and a material handling robot that transfers parts between the machines. At the same time, the robots in the system function to load/unload the machines. In the academic literature and the business world, minimization of the cycle time is the dominant objective for the optimization of production schedules. In parallel to this objective, robot movements are assumed to be at their maximum speeds and the energy savings that can be attained with the dynamic adjustment of the speeds is not taken into account. Within the scope of this study, considering the controllability of the move speeds of the dual-gripper robot, it is aimed to determine the optimal robot move sequence and the operation parameters of the robot. In other words, we considered the energy consumption and throughput rate objectives simultaneously as a bi-criteria optimization model. In order to solve this bi-criteria model we used the ε -constraint approach, one of the objectives is written as a constraint. For the problem, a mixed integer non-linear mathematical model (MINLP) is developed. In order to increase the solution efficiency, the problem is reformulated as a mixed integer second order constrained mathematical model (MISOCP). Computational tests revealed that the MINLP model cannot guarantee the optimal solution in reasonable times for systems with two or more machines.

Since it is also observed that the MISOCP model is not efficient for the systems with six or more machines, a heuristic solution method called Efficient Solution Generation algorithm (ETA) is developed. By using the created data sets, these three solution approaches are investigated through a computational study. Additionally, as a result of these tests, it is observed that an average energy savings of 17.7% can be attained by considering controllability of robot move speeds.

Keywords: Robotic cell scheduling, Bi-criterion optimization, Mathematical modeling, Heuristic methods



TEŞEKKÜR

Çalışmalarım ve yüksek lisans eğitimim boyunca değerli yardım ve katkılarıyla beni yönlendiren, kendimi geliştirmemde ve bu alanda çalışma isteğimi daima sürdürmemde emekleri olan değerli hocalarım Doç. Dr. Hakan GÜLTEKİN ve Prof. Dr. Sinan GÜREL'e,

Kıymetli zamanlarını ayırarak tezimi okuyan ve geri bildirimde bulunan tez jürimin üyeleri Doç. Dr. Ayşegül ALTIN KAYHAN, ve Doç. Dr. Çağrı KOÇ'a,

Yüksek lisans eğitimim boyunca bana burs sağlayan TOBB Ekonomi ve Teknoloji Üniversitesine, ve yine değerli tecrübelerinden faydalandığım TOBB Ekonomi ve Teknoloji Üniversitesi Endüstri Mühendisliği Bölümü öğretim üyelerine ve 215M845 kodlu proje kapsamında destek sağlayan TÜBİTAK'a,

Çalışmalarım boyunca her zaman yanımda olup beni motive eden, sevgilerini hiç esirgemeyen arkadaşlarıma,

Maddi manevi her konuda her zaman yanımda duran canım annem ve babam Kevser ve Tahir Necati TATAR'a, beni bu günlere sevgi ve saygı kelimelerinin anlamlarını bilecek şekilde yetiştirerek getiren babaannem ve anneanneme, neşe kaynağı kardeşlerim Mehmet Tarık, Ali Bahadır ve Zeynep Aybike'ye, beni ailelerine kabul ederek, kendi kızları gibi gören sevgili kayın annem ve babam Saliha CİRİT ve Mehmet EMİROĞLU'na,

Her zaman arkamda duran, bu çalışmada bana benden çok inanan ve motivasyon kaynağım olan, sevgisini ve huzurunu benden hiç esirgemeyen canım eşim Kadir Yavuz EMİROĞLU'na teşekkür ederim.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	iv
ABSTRACT	vi
TEŞEKKÜR	viii
İÇİNDEKİLER	ix
ŞEKİL LİSTESİ	x
ÇİZELGE LİSTESİ	xi
KISALTMALAR	xii
SEMBOLE LİSTESİ	xiii
1. GİRİŞ	1
2. LİTERATÜR ARAŞTIRMASI	5
3. PROBLEM TANIMI	13
3.1 Aktivite Tanımları	15
3.2 Örnek Aktivite Sıralaması	20
4. GELİŞTİRİLEN ÇÖZÜM YÖNTEMLERİ	23
4.1 Matematiksel Modeller	23
4.1.1 Karışık Tamsayılı Doğrusal Olmayan Matematiksel (MINLP) Model:	24
4.1.2 Tamsayılı İkinci Derece Konik Programlama (MISOCP) Mo- deli:	29
4.2 Etkin Çözüm Türetme Algoritması (ETA)	32
4.2.1 ETA: Aşama 1	41
4.2.2 ETA: Aşama 2	44
4.2.3 ETA: Aşama 3	49
4.2.4 ETA: Aşama 4	53
4.2.5 ETA: Aşama 5	56
5. HESAPLAMALI SONUÇLAR	57
5.1 Veri Kümeleri	57
5.2 MINLP, MISOCP ve ETA Koşturum Sonuçları	60
5.3 Aktivite Sıralamaları Sonuçları	73
5.4 Enerji Tasarrufu ile İlgili Sonuçlar	80
6. SONUÇ VE ÖNERİLER	83
KAYNAKÇA	85
ÖZGEÇMİŞ	91

ŞEKİL LİSTESİ

Şekil 1.1: m makineli çift tutuculu robotik hücre	3
Şekil 3.1: Robotun aksiyon süresine karşı enerji tüketimi [4]	14
Şekil 3.2: Robot döngüsünde olabilecek dört alternatif aktivite arası mesafeler .	19
Şekil 4.1: Etkin Çözüm Kümesi	34
Şekil 5.1: 2.4 Numaralı Döngü	74
Şekil 5.2: 2 makine, ÖT, DV veri kümesi için örnek etkin çözüm kümesi	75
Şekil 5.3: 3.1 Numaralı Döngü	76
Şekil 5.4: 4.1 Numaralı Döngü	78
Şekil 5.5: m makine	79
Şekil 5.6: m makine	79

ÇİZELGE LİSTESİ

Çizelge 5.1: Parametre Değerleri	59
Çizelge 5.2: 2 makine, İşlem Süresi DA, MISOCP, MINLP ve ETA karşılaştırması	62
Çizelge 5.3: 2 makine, İşlem Süresi GA, MISOCP, MINLP ve ETA karşılaştırması	63
Çizelge 5.4: 3 makine, ETA ve MISOCP karşılaştırması	64
Çizelge 5.5: 3 makine, ETA ve MISOCP karşılaştırması (Devamı)	65
Çizelge 5.6: 3 makine, ETA ve MISOCP karşılaştırması (Devamı)	66
Çizelge 5.7: 4 makine, ETA ve MISOCP karşılaştırması	68
Çizelge 5.8: 4 makine, ETA ve MISOCP karşılaştırması (Devamı)	69
Çizelge 5.9: 4 makine, ETA ve MISOCP karşılaştırması (Devamı)	70
Çizelge 5.10: 6 makine, ETA ve MISOCP karşılaştırması	71
Çizelge 5.11: 6 makine, ETA ve MISOCP karşılaştırması (Devamı)	72
Çizelge 5.12: 2 ve 3 makine, Enerji Tasarrufu ile ilgili sonuçlar (%)	80
Çizelge 5.13: 4 ve 6 makine, Enerji Tasarrufu ile ilgili sonuçlar (%)	81

KISALTMALAR

- MINLP : Karma Tamsayılı Doğrusal Olmayan Programla Modeli
(Mixed Integer Non-Linear Programming)
- MISOCP : Karma Tamsayılı İkinci Derece Konik Programlama Modeli
(Mixed Integer Second Order Constrained Programming)
- ETA : Etkin Çözüm Türetme Algoritması
- EÇK : Etkin Çözümler Kümesi



SEMBOL LİSTESİ

Bu çalışmada kullanılmış olan simgeler açıklamaları ile birlikte aşağıda sunulmuştur.

Kümeler	Açıklama
\mathcal{M}	Makineler kümesi
\mathcal{K}	Yükleme/boşaltma alternatifleri kümesi
\mathcal{KG}	Tutucu değişimi gerektiren ardışık yükleme ve boşaltma alternatifleri kümesi
\mathcal{P}	Aktivitelerin atanabileceği pozisyonların kümesi

Parametreler	Açıklama
ε	Makine yükleme/boşaltma zamanı
θ	Robotun aktif tutucusunu değiştirme süresi
$dist_{mn}$	M_m makinesinden M_n ($m \neq n$) makinesine mesafe
P_m	M_m makinesindeki parça işleme süresi
VUB	Robotun olabilecek en yüksek hızı
VLB	Robotun olabilecek en düşük hızı
C	Enerji tüketim fonksiyonu sabiti. Bu sabit, birim mesafe başına etki eden ağırlık ve sürtünme kuvvetlerinden kaynaklanmaktadır
G	Her tutucu değişiminde ortaya çıkan enerji tüketim fonksiyonu sabiti
B	Modelde kullanılan yeterince büyük bir sayı
s	Modelde kullanılan yeterince küçük bir sayı

1. GİRİŞ

Üretim endüstrisinde hız ve kalite açısından artan rekabet ve değişken müşteri talepleri sebebiyle firmaların otomatik sistem kullanımı gittikçe artmaktadır. Bu otomatik sistemlerden bir tanesi de endüstriyel robotlardır. World Robotics tarafından 2019 yılında yayınlanan rapora göre 2018’de dünya genelinde satın alınan robot sayısı 2015 yılına göre %30 artış göstermiştir [50].

Otomotiv sanayinin baskın olduğu ülkemizde de üretimde robot sistemlerinin kullanımını artış göstermektedir. 2008 yılında yapılan bir çalışmaya göre %19,6 oranı ile en çok kullanılan robotlar kaynak robotlarıdır. İkinci sırayı ise %18.3 ile malzeme elleçleme robotları almaktadır [36].

Malzeme elleçleme robotlarının kullanıldığı sistemlerden biri ise robotik hücrelerdir. Bu hücreler, giriş ve çıkış stoku, makine ya da makinelerden oluşur. Sistemde bulunan bir robot ise makineler arasında parça taşıması yapmakta aynı zamanda da makinelere işlenecek parçaları yüklemekte ve işlemi tamamlanan parçaları makineden boşaltmaktadır. Bu tip sistemlerde robotlar bir bilgisayar sistemine bağlı olarak komuta edilmektedir. Böylece, robotun hangi makineler arasında parça taşıma yapacağı, bu taşımanın hızı, ivmesi vb. nitelikleri kontrol edilebilmektedir. Robotun parça taşıırken aldığı yol ve taşıma hızına göre de robotun harcadığı enerji miktarı değişmektedir. Diğer bir deyişle, robotun hareket hızını kontrol ederek enerji tüketimini de kontrol etmek mümkündür.

Dünya genelinde artan karbon ayak izinin düşürülmesi bilinci ve sivil toplum kuruluşlarının enerji tasarrufu ile ilgili çalışmalarının yanında devletler de bu enerji tüketiminin azaltılması yönünde yasalar çıkarmaktadır. Türkiye de bu devletler arasındadır. Enerji Verimliliği Derneği (ENVERDER) enerji ve kaynakların verimli kullanılmasını sağlamak amacıyla kurulmuş çok uluslu ve çok ortaklı bir organizasyon olan Green Industry Platform’a üyedir. Bu derneğin raporuna göre, 2023’e kadar işletmelerin enerji

tüketiminin en az %15 oranında azaltılması hedeflenmiştir. Böylece 2023 yılına kadar 65 milyar TL tasarruf sağlanmasını hedeflenmektedir [18].

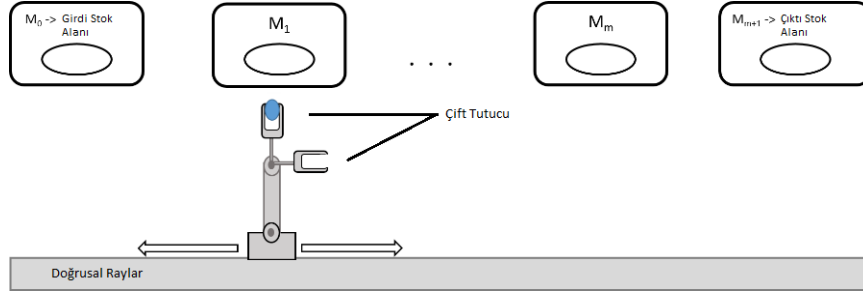
Enerji tasarrufu amacıyla sağlanan teşvik ve giderek artan enerji maliyetleri sebebiyle üretim endüstrisinde de enerji tasarrufu ile ilgili alınan tedbirler artmaktadır. Enerji tüketimlerinin önemli boyutlarda olduğu otomasyonlu üretim sistemlerinde de daha az enerji kullanma bilinci öne çıkmaya başlamıştır. Öyle ki bu alanda yapılan çalışmalar gün geçtikçe artmaktadır. Örneğin, ABD’de üretimde kullanılan enerji miktarı 2002’den 2005’e kadar %20 [3] oranında azaltılmıştır . Almanya’da 2004 yılı rakamları 1991’e göre %23 iyileşmiştir. Japonya’da da NEDO tarafından yapılan çalışma ile makinelerin işleme sürelerinin kısaltılması ile enerji tüketiminin en iyileştirilebileceği ile ilgili projeler sunulmuştur [17]. Otomotiv sektöründe üretim sistemindeki toplam enerji tüketiminin yaklaşık %8’i endüstriyel robotlardan kaynaklanmaktadır [41]. Bukata vd. [9] tarafından yapılan bir çalışmada bir robotik hücrede harcanan enerjinin %20’sinin tasarruf edilebileceği belirtilmiştir.

Bu çalışmanın amacı bir robotik hücrede üretim çıktı miktarını maksimize ederken aynı zamanda enerji tüketimini minimize etmek ve böylece sistemden en yüksek faydayı sağlamaktır. Sistemde çift tutuculu tek bir robot olduğu varsayılmaktadır.

Çift tutuculu tek bir robotun bulunduğu üretim hücresi Şekil 1.1’de gösterilmiştir. Bu tip robotlar aynı anda iki parça taşıyabilmekte veya bir makineyi boşalttıktan sonra, başka bir makineye gitmeye gerek olmadan aynı makineyi diğer tutucuda bulunan parça ile yeniden yükleyebilmektedirler. Herhangi bir anda, robotun tutucularından bir tanesi aktif diğeri pasif durumdadır. Pasif tutucudaki bir parçayı makinelere yükleyebilmek veya bu tutucuyla bir parçayı boşaltabilmek için aktif tutucunun değiştirilmesi gerekmektedir.

Makinelerin önünde ara stok alanı olmadığından, parçalar direkt olarak makinelere yüklenmektedir. İşlem tamamlandığında, parçalar robot tarafından boşaltılana kadar makine üzerinde bekleyebilmektedir. Fakat bu süre zarfında ilgili makine bloke olmakta ve makineye başka bir parça yüklenememektedir. İşlemi tamamlanan parça robot tarafından makineden boşaltılmaktadır. Robot aktivite sıralaması yapılırken, sistemde döngüsel üretim yapıldığı yani belirlenen aktivite sıralamasının sürekli olarak

tekrar ettiği varsayılmıştır.



Şekil 1.1: m makineli çift tutuculu robotik hücre

Doğrusal olarak dizilmiş makineler arasında parça transferini sağlayan robot, bir makineyi yükledikten sonra, işlem süresi boyunca makinenin önünde bekleyebilir ya da bu arada başka makineye giderek yükleme veya boşaltma yapabilir. Eğer makinenin önünde beklerse buna "tam bekleme" denir. Diğer taraftan yükleme sonrası başka bir makineye giderse, boşaltmak için bu makineye geri döndüğünde eğer parçanın işlemi tamamlanmışsa parçayı hemen boşaltırken eğer işlem tamamlanmamışsa, tamamlanana kadar beklemektedir. Bu bekleme ise "kısmi bekleme" adı verilir.

Literatürde bulunan birçok çalışmada amaç, aktivitelerin en kısa sürede tamamlanmasını sağlamak böylece de robotik hücrelerin üretim verimliliğini artırmaktır. Bu amaç, çevrim zamanının minimize edilmesine eşdeğerdir. Bu sebeple de robotların mümkün olan en hızlı şekilde hareket ettikleri sistemler üzerinde çalışılmaktadır. Ayrıca, robotun hareket hızından ve tutucu değişiminden kaynaklı enerji tüketim değerleri göz önünde bulundurulmamaktadır. Ancak, yukarıda bahsedilen kısmi bekleme durumlarında, robot daha düşük bir hızla hareket ettiğinde kısmi beklemenin elimine edilerek takip eden aktivitelerin yine aynı zamanda tamamlanması mümkündür. Diğer bir deyişle, boşaltma aktivitesi için bir makineye ilerlerken en hızlı halinden daha yavaş hareket etmesi sadece bekleme süresini azaltacak ve enerji tüketimini azaltmış olacaktır. Böylece üretim hızından ödün vermeden enerji tasarrufu sağlanmış olacaktır. Diğer taraftan düşük talep sezonları gibi çeşitli sebeplerle üretim hızlarının düşürülebileceği periyotlarda hem istenen üretim hızını sağlayıp hem de mümkün olan en düşük enerji tüketim değerini elde etmek önemli tasarruflar sağlayacaktır.

Bu çalışma kapsamında, belirtilen potansiyel enerji tasarruflarını göz önünde bulun-

durmak için, robotların hızlarının ayarlanabilir olduğu varsayılmıştır. Üretim hızının en yüksek seviyede tutulması aynı zamanda da enerji tasarrufunun mümkün olduğunca sağlanması için, bu tip hücrelerdeki robot hareket sıralaması ve robot hareket hızlarının belirlenmesi problemlerinin eş zamanlı olarak çözülmesi amaçlanmıştır. Belirtilen problem ilk olarak Karma Tamsayılı Doğrusal olmayan Programlama (MINLP) ile matematiksel olarak modellenmiştir. Oluşturulan veri kümeleri ile koşturum yapıldığında bu modelin çözüm süresi performansı bakımından küçük boyutlu veri kümelerinde dahi yetersiz olduğu görülmüştür. Bu sebeple, oluşturulan matematiksel modeldeki doğrusal olmayan amaç fonksiyonu ve kısıtlar değiştirilerek Karma Tamsayılı İkinci Dereceden Konik Programlama (MISOCP) şeklinde modellenmiş ve çözüm süresi performansının oldukça iyileştiği görülmüştür. MISOCP modelin performansının yetersiz kaldığı durumlar için ise meta sezgisel bir Etkin Çözüm Türetme Algoritması (ETA) geliştirilmiştir. Sunulan 3 çözüm yönteminin performans karşılaştırmaları yapılmıştır. Ayrıca oluşturulan veri kümeleri için elde edilen sonuçlar aktivite sıralaması ve enerji tüketimi tasarrufları açısından da değerlendirilmiştir.

Bölüm 2’de literatürde bulunan robotik hücrelerle ilgili çalışmalar incelenmiş, yapılan çalışmanın literatüre katkısı sunulmuştur. Bölüm 3’te ele alınan problem detaylı olarak açıklanmıştır. Ele alınan problemin çözülmesi için geliştirilen 3 çözüm yöntemi ise Bölüm 4’te yer almaktadır. Geliştirilen modeller ve algoritmanın performans açısından kıyaslanmaları da Bölüm 5’te bulunmaktadır. Ayrıca bu bölümde elde edilen sonuçlar aktivite sıralamaları açısından incelenmiş ve olası enerji tasarrufları yorumlanmıştır. Son olarak Bölüm 6’da ele alınan problem ve çözüm yöntemi özetlenmiş, elde edilen sonuçlar sıralanmıştır.

2. LİTERATÜR ARAŞTIRMASI

Literatürde robotik hücrelerle ilgili farklı ortam parametrelerine, şartlarına ve farklı amaç fonksiyonlarına yönelik çalışmalar yapılmıştır. Ele alınan farklı ortam özellikleri, makine sayısı, makinelerin dizilimi ve taşıyıcı robotun sahip olduğu tutucu sayısı gibi özelliklerdir.

Literatürde, makinelerin doğrusal ya da dairesel olarak dizildiği sistemler için araştırmalar bulunmaktadır. Şekil 1.1’de olduğu gibi doğrusal makine diziliminde robot düz bir yörünge üzerinde hareket ederek aktiviteleri gerçekleştirmektedir. Dairesel dizilimin bulunduğu üretim hücrelerinde ise robot makine diziliminin merkezinde bulunmakta ve parça taşıma/ yükleme/boşaltma aktivitelerini yapmaktadır. Makine diziliminin yanı sıra makineler arasındaki uzaklık tanımı da çalışmalar arasında değişmektedir. Şekil 1.1’de olduğu gibi makineler ardışık olarak numaralandırıldığında, $0 \leq i < j \leq m + 1$ olmak üzere i ve j makineleri arasındaki uzaklık $dist_{ij}$ ile gösterilebilir. Bu değer $dist_{i(i+1)} + dist_{(i+1)(i+2)} + \dots + dist_{(j-1)(j)}$ toplamına eşit ise bu uzaklık ölçümü "toplamsal uzaklık" olarak adlandırılmaktadır. Eğer ardışık makineler arası uzaklık birbirine eşit ve $dist_0$ ise i ve j makineleri arası mesafe $dist_{ij} = |i - j|dist_0$ olarak belirlenmektedir. Bu uzaklık tipi de "özdeş toplamsal uzaklık"tır. Bir başka uzaklık ölçümü ise tüm makineler arası uzaklıkların birbirine eşit olduğu "sabit uzaklık" ölçümüdür. Literatürdeki çalışmaların hemen hepsinde sabit robot hızları ele alındığından, robotun makineler arası ulaşım süresi δ_{ij} de toplamsal, özdeş toplamsal, genel ulaşım süresi olarak adlandırılmaktadır. Bu çalışmada ise belirtilen ölçüm kuralları yalnızca makineler arası uzaklıklar için geçerlidir. Robotların hareket hızı dinamik olarak değişebileceğinden δ_{ij} için kesin bir şey söylenememektir.

Robotik hücrelerle ilgili literatürdeki çalışmalar incelendiğinde robot türüne göre de sınıflandırma yapıldığı görülmektedir. Yapılan çalışmalarda tek ve çift tutuculu olmak üzere iki farklı tip robot ele alınmaktadır. "Tek tutuculu" bir robot aynı anda tek bir parça taşıyabildiğinden, bir boşaltma aktivesinden sonra yapabileceği tek aktivite ta-

şıdığı parçayı başka bir makineye yüklemektir. Aynı anda iki parça taşıyabilen "çift tutuculu" robotlar ise ardışık iki boşaltma aktivitesini ardışık olarak gerçekleştirebilmektedir.

Ortam şartlarına ise makineden alınma kriteri örnek olarak gösterilebilir. İşlemi biten parçanın türüne göre, makineden belli bir süre içinde (interval) ya da hemen (no-wait cell) alınması gerekebilir. Bu kriter genelde gıda ve kimya endüstrisi gibi alanlarda kullanılmaktadır. Bazı sistemlerde ise işlem süresi biten parçanın alınma süresi için bir kriter bulunmamaktadır (free-pick up cell). İşlemi biten parça, robotun boşaltma işlemine kadar makinede bekleyebilir.

Bazı sistemlerde ise üretilen parça tipi sayısı bir iken, farklı tip parça üretimi için yapılmış çalışmalar da mevcuttur. Tek tip özdeş parçalar üretilen sistemlerde verilmesi gereken tek karar robotun aktivite sıralamasıdır. Farklı tip parça üretiminde ise parçalar genellikle aynı makinelerde farklı işlem süresine sahiptir. Bu sebeple, farklı tip parça üretimi olan sistemlerde aktivite sıralaması ile birlikte parçaların hangi sıra ile işleme alınacakları da belirlenmelidir.

Ele alınan çalışmalar, belirlenen aktivite sıralamasının bir tekrarı sonucu sistem çıkan ürün miktarına göre de farklılıklar göstermektedir. Robotun bir döngüyü tamamlaması sonucunda n adet parçanın üretildiği döngülere " n -birim döngüsü" adı verilmektedir. Dolayısıyla bir tekrar sonucu 1 ürün üretilen döngüler 1-birim döngülerdir.

Robotik hücre sistemlerinde yoğunlaşılacak amaç fonksiyonları da farklılıklar göstermektedir. Bunlardan ilki ve en çok kullanılan döngüsel çevrimler için üretim miktarını en büyükmeyi amaçlayan çevrim zamanı minimizasyonudur. Sayısı artmakla birlikte, üretim sırasında harcanan enerji tüketiminin azaltılması üzerine yapılan az sayıda çalışma da literatürde bulunmaktadır. Ayrıca, üretim maliyetlerinin minimize edilmesi de ele alınan amaçlar arasındadır. Son olarak literatürde, belirtilen amaç fonksiyonlarını içeren iki kriterli amaç fonksiyonları kullanılarak problemlerin ele alındığı az sayıda çalışma da bulunmaktadır. Robotik Hücre çözelgesi ve ele alınan problem tipleriyle ilgili daha geniş bilgiye Crama vd. [12] , Geismar vd. [13] ve Brauner'in [5] literatür tarama makalelerinden ulaşılabilir.

Bu çalışmada ele alınan ortam parametreleri ise şu şekildedir, makineler doğrusal olarak dizilmiştir ve ardışık makineler arasındaki uzaklıklar birbirine eşittir. Aynı anda iki parça taşıyabilen çift tutuculu bir robot makineler arasında doğrusal olarak hareket etmekte ve makinelere parça yükleyip işlemi tamamlanan parçaları boşaltmaktadır. İşlemi tamamlanan parçalar boşaltılmak için robotun gelmesini bekleyebilirken, makinelerde işlem için bekleyen ya da işlemi tamamlanan parçalar için bir ara stok alanı bulunmamaktadır.

Literatürde yapılmış çalışmalarda belirlenen ortam parametrelerinin ele alınış şekli farklılıklar göstermektedir. Aşağıda bu tezde ele alınan problemle ilgili literatür üç alt başlık halinde sunulmuştur.

Tek tutuculu robotik hücreler ile ilgili çalışmalar

Robotik hücre alanındaki ilk çalışmalardan biri Sethi vd. [46] tarafından 1992 yılında yapılmıştır. Bu çalışmada, 2 ve 3 makineli tek tip parça üreten robotik hücrelerde robot aktivitelerinin çizelgelenmesi için analitik çözüm yöntemleri geliştirilmiştir. Ayrıca, tek tip parça üreten robotik hücrelerde 1-birim döngülerin, 2 ve 3 makineli sistemler için optimal olduğu ve m makineli bir robotik hücre için var olan en iyi 1-birim döngünün de polinom zamanda bulunabildiği gösterilmiştir. Ardından, Crama ve van de Klundert [10] toplanabilir hareket zamanının kullanıldığı tek tutuculu robot içeren ve tek tip parça üreten bir sistemde robot aktivitelerini çizelgeleme probleminin NP-Zor olduğunu göstermiştir. Aynı çalışmada, bu sistem için 1-birim döngüler arasından en iyi 1-birim döngüyü bulan $O(m^3)$ zamanlı bir algoritma da geliştirilmiştir. Hall vd. [33] 3 makine için 1-birim döngülerin daha karmaşık döngülerden daha iyi olduğunu ispat etmişlerdir . Crama ve Van de Klundert [11] bu ispatı basitleştirmiştir. Brauner ve Finke [6],[7] 4 veya daha fazla makineli hücreler için 1-birim döngülerin optimal olamayabileceğini göstermişlerdir . Geismar vd. [25] 1-birim döngülerin en iyi çözümden uzaklıklarının en kötü durumda bile 10/7 oranında olduğunu ispatlamışlardır. Bu da anlaması ve uygulaması kolay olan 1-birim döngülerin ele alınmış olmasını anlamlı kılmaktadır. Bu tez çalışmasında da 1- birim döngüler üzerinde çalışılmıştır.

Literatürde robotik hücre ortam parametrelerinin esnek değerler alabileceği çalışmalar da bulunmaktadır. Akturk vd. [2] tarafından yapılan çalışmada, makine işlem süre-

lerinin ayarlanabilir olduđu varsayılmıştır . Bu çalışmada ele alınan problemde, bir parçanın üretimi için gereken toplam süre sabittir ancak bu sürenin hangi makineye ne kadar paylaştırılacağı karar deđişkeni olarak belirlenmiştir. Her makinenin her operasyonu gerçekleştirebildiđi varsayılmıştır. Takip eden çalışmada Gultekin vd. [26] her operasyonun her makinede yapılabileceđi varsayımını kaldırmıştır. Bunun yerine her makinenin belirli işler yapabildiđi durumu ele almışlardır.

Robotik hücrelerle ilgili bazı çalışmalarda da farklı tip parçaların üretildiđi problemler ele alınmıştır. Bu problem tipi için, robotik hücrelerde hem robot hareket döngüsü hem de parça sıralaması bulunmaya çalışılır. Sethi vd. 1992 [46] yılında yaptıkları çalışmada 2 makineli bir robotik hücrede farklı tip parça üretimini de ele almıştır. Robot hareket çevrimi sabitlenip ve her bir parça için talep deđerleri verildiđinde, optimal parça sıralamasını belirleyen polinom zamanlı bir algoritma sunulmuştur. Ayrıca, robot hareket çevrimi ile parça sıralaması problemini eş zamanlı bulan bir algoritma Hall vd. [33] tarafından geliştirilmiştir. Çalışmada ayrıca 3 makineli bir robotik hücrede verilen bir döngü için (1-birim) parça sıralamasının belirlenmesi problemi incelenmiş ve robot çevrimi sabitlense bile parça sıralamasının bulunması probleminin olası 6 çevrimin 4'ü için NP-Zor olduđu ispatlanmıştır.

Burada özetlenen tek tutuculu robotlu hücrelerle ilgili çalışmaların tamamı robot hızlarının sabit olduđu varsayımı ile çevrim zamanı minimizasyonu problemini ele almıştır. Bu çalışmada ise tek tutuculu robotlara göre daha yüksek üretim hızı deđerleri verebilen çift tutuculu robotlu hücreler ele alınmakta ve robot hızlarının deđiştirilebilir olduđu varsayımı ile çevrim zamanı minimizasyonu yanında enerji tüketimi minimizasyonu amaçları iki kriterli bir problem olarak ele alınmaktadır.

Çift tutuculu robotik hücreler ile ilgili çalışmalar

Çift tutuculu robotik hücreler üzerine yapılmış öncü çalışmalardan biri Sethi vd. [45] tarafından 2001 yılında gerçekleştirilmiştir. Bu çalışma ile tek tutuculu sistemde 2 makinede 2 adet 1-birim döngüsü mümkün iken, çift tutuculu robotik hücrelerde 2-makineli bir üretim hücresi için 52 farklı döngü olduđu ortaya çıkarılmıştır. Söz konusu çalışmada 2-makineli dairesel dizilimin olduđu sistemlerde parametrelerin belirli şartları sağlaması durumunda en iyi 1-birim döngüsü belirlenmiş, m -makineli sistemlerde

de çevrim zamanı için bir alt sınır geliştirilmiştir. Daha maliyetli olan çift tutuculu robotik hücrelerin, tek tutuculu robotik hücrelere göre avantajları incelenmiş ve hücre parametreleri ve makine sayısı verildiğinde çift tutuculu robot ile tek tutuculu robot performansını karşılaştıran basit bir sezgisel geliştirilmiştir.

Drobouchevitch vd. [15] m -makinelik çift tutuculu robotik hücre için 1-birim döngü sayısını hesaplayan bir algoritma geliştirmiştir. Bu algoritmaya göre, 10-makinelik sistemde 6.4×10^{11} adet 1-birim döngüsü bulunmaktadır. Ayrıca, tek ve çift tutuculu robotik hücreler verimlilik açısından karşılaştırılmış ve çift tutuculu robotik hücrelerin daha üretken olduğu gösterilmiştir. Geismar vd. [22] tek ve çift tutuculu robotik hücreler için ayrı ayrı algoritmalar geliştirmişlerdir. Drobouchevitch vd. [16] çift tutuculu robotlu hücrelerde farklı tip parçaların üretimini ele alarak robot hareket sıralaması ve parça sıralaması problemlerini ele almışlardır. Söz konusu çalışmada belirli 1-birim robot hareket dizilerinin altında en iyi parça sıralamasının bulunmasına odaklanılmıştır. En kötü durumda $3/2$ -yaklaşık performans sağlayan bir sezgisel algoritma geliştirilmiştir. Sriskandarajah vd. [48], 2-makinelik çift tutuculu ve farklı tip parça üreten bir sistem ele almışlardır. Robot döngüsü sabitlenmiş bile parça sıralama probleminin birçok durumda NP-Zor olduğunu ispatlamışlardır. Çözüm için bir sezgisel algoritma geliştirip, tek tutuculu robot yerine çift tutuculu robot kullanımının %18 ile %36 arasında bir iyileştirme sağladığını tespit etmişlerdir.

Geismar vd. [20] paralel makinelerin bulunduğu çift tutuculu robotlu hücreleri ele almışlardır. Uygulamada sıklıkla karşılaşılan bazı varsayımlar altında hem basit robotlu hücreler için hem de paralel makinelerin bulunduğu robotlu hücreler için en iyi çevrimleri belirlemişlerdir. Foumani ve Jenab [19], çift tutuculu robot olmamasına rağmen, "swap" adı verilen, robot üzerindeki parça ile makine üzerindeki parçanın değiştirilmesine olanak sağlayan bir özelliği ele almışlar ve bu özelliğin verimliliği artırdığını göstermişlerdir.

Gündoğdu ve Gultekin [29] tarafından yapılan bir çalışmada ise robotun belirli sayıda parça taşıyabilecek bir donanıma sahip olduğu özel bir hücre konfigürasyonu ele alınmıştır.

Bunların dışında, robotun birbiriyle bağımlı olarak hareket eden iki kolunun bulunduğu

robotik hücreler ([38],[21]), üretim ortamında darboğaz olan makinelere paralel aynı özellikte makinelerin eklenmesiyle ortaya çıkan hibrid robotik hücreler ([32], [23]), makineler arasında ara stok alanının bulunduğu robotik hücreler ([29], [16], [14]), robot sayısının birden fazla olduğu, ve/veya bir karar değişkeni olduğu robotik hücreler ([34], [35]) literatürde ele alınmıştır.

Yukarıda bahsedilen çalışmalarda robot hareket süreleri ve makine işlem süreleri birer problem parametresi olarak ele alınmış ve önceden bilindiği varsayılmıştır. Ayrıca bu çalışmalardaki ele alınan amaç fonksiyonu üretim hızını en büyükmek dolayısıyla çevrim zamanını en küçükmektir. Ancak bu tez çalışmasında robot hareket hızlarının değiştirilebilir olmasından yola çıkılarak çevrim zamanı ve enerji tüketimi minimizasyonu aynı anda ele alınmıştır.

Robot hareket hızlarının kontrolü ile ilgili çalışmalar

Literatürde amaç fonksiyonu sadece çevrim zamanı minimizasyonu olmayan, ortam parametrelerinin karar değişkeni olarak alındığı çalışmalar da bulunmaktadır. Bunlardan bazıları Gultekin vd. tarafından 2008 [27] ve 2010 [28] yıllarında yapılan çalışmalardır. Belirtilen çalışmalarda, robotik hücre sisteminde bulunan CNC makinelerinin hızları ele alınmıştır. Bu hızların belirli bir maliyet karşılığı azaltılıp artırılacağı varsayımı ile hem üretim maliyetini hem de çevrim zamanını minimize edecek iki-kriterli modeller ele alınmıştır. Ele alınan robotik hücreler 2 ve 3 makineli akış hücreleridir. İlk çalışmada hangi işlemin hangi makinede gerçekleşeceği belliyken, ikinci çalışma ile bu atama karar değişkeni olarak modele eklenmiştir.

Gurel ve Cincioğlu [30] makine hızlarının kontrol edilebildiği problem türü için, üretim maliyeti ve geciken iş sayısı hedeflerinin ödünleşmelerini incelemiş ve etkin çözüm kümesi üretmek üzere matematiksel modeller ve sezgisel tarama algoritmaları önermiştir.

Belirli kaynak kullanımını altında enerji tüketimini minimize etmeyi hedefleyen çalışmalar da literatürde bulunmaktadır. Araç hızlarının kontrol edilebilirliği üzerine ele alınan bu araç rotalama problemleri ile yakıt tüketiminin dolayısıyla karbondioksit salınımının minimize edilmesi amaçlanmıştır. Koç vd. [37] tarafından yapılan çalışmada

Yeşil Araç Rotalama problemleri olarak adlandırılan bu problem ele alınmıştır. Bu konu üzerinde yapılan çalışmaların kapsamlı bir taraması Lin vd. [39] tarafından yapılmıştır. Saka vd. [44] çalışmalarında hız kontrollü araç rotalama problemi için bir MISOCP modeli ve sezgisel tarama algoritması önermişlerdir. Akturk vd. [1] (2014) uçak yeniden çizelgeleme probleminde uçakların hızları da karar değişkeni olarak ele alınmıştır. Doğrusal olmayan bir yapıda olan amaç fonksiyonu karbondioksit salınımı maliyetini minimize etmektedir. Ele alınan problemin, ikinci-derece konik programlama eşitsizlikleri kullanarak etkin bir şekilde çözülebileceği gösterilmiştir.

Robotik hücrelerin enerji tüketimi ile ilgili çalışmalara da literatürde rastlanmıştır. Berlin [4] tarafından yapılan deneysel çalışmada robotun hızı en yüksek seviyedeysen, enerji tüketimi de en yüksek değerinde olduğu gösterilmiştir. Benzer deneysel çalışmalar farklı tiplerde robotlar için ve farklı özelliklerin (hız, ivmelenme, taşınan yükün ağırlığı vb.) enerji üzerine etkilerini incelemek için de yapılmıştır [42]. Smetanova'nın deneysel çalışmasında [47] IRB 4400/60 tipi robotla yaptığı denemelerde robotun farklı ağırlıklardaki yükleri taşıırken enerji tüketiminin nasıl değiştiği grafiksel olarak gösterilmiştir. Bu etkilerin doğrusal olmayan yapıda oldukları ve konveks bir fonksiyon şeklinde gösterilecekleri ifade edilmiştir.

Bryan vd. [8] belirli bir süre içerisinde, belirlenen başlangıç noktasından bitiş noktasına gidecek olan robotun optimal ivmelenme ve hız değerlerinin bulunmasına yönelik bir çalışma yapmışlardır. Bu tez çalışmasında ise robotun tam olarak hangi hareketleri yapacağı operasyonların sıralanması sonucu belirlenecektir. Ayrıca, her bir hareketi tamamlamak için belirli bir süre yoktur, bu süre de bir karar değişkenidir. Ayrıca bu projede çevrim zamanı ve enerji tüketimi minimizasyonu 2-kriterli bir şekilde ele alınacağı için problemler oldukça farklıdır. Meike ve Ribickis tarafından 2010 yılında [41], Pellicciari vd. [43] tarafından da 2013 yıllarında yapılan çalışmada robotun enerji tüketimini düşürmek için hız ve ivme değiştirme, yörünge planlama gibi çeşitli stratejiler önerilmiştir. Vergnano vd. [49] çok robotlu bir sistemde toplam enerjiyi minimize etmek üzere işlem ve robot hareket sıralarını belirlemek üzere bir karma tamsayı doğrusal olmayan programlama modeli geliştirmişlerdir. Ele alınan robotlar parça üzerinde işlem yapan robotlardır ve sistemde birden fazla robot bulunmaktadır. Ayrıca problem 2 kriterli olarak ele alınmamış ve çift tutuculu robotlar kullanılmamıştır. Bu sebeple,

mevcut tez çalışması ile büyük farklılıklar göstermektedir.

Son olarak Gurel vd. [31] tarafından 2019 yılında yayınlanan çalışmada tek tutuculu robotik hücre için hem enerji tüketimi hem de çevrim zamanı minimizasyonu hedeflenmiştir. 2 makineli sistemler için KKT şartları uygulanarak robot hızları analitik olarak belirlenmiştir.

Verilen literatür özetinde görüldüğü üzere, daha önce çift tutuculu robotlu hücrelerde robot hareket hızlarının kontrol edilebilir olması ve buna bağlı olarak çevrim zamanı ve enerji tüketimi hedefleri ele alınmamıştır. Literatürdeki çift tutuculu robotik hücrelerle ilgili çalışmaların tamamında robot hareket süreleri bilinen birer problem parametresidir. Bu sebeple bu tez çalışmasının özgün bir içerikte olduğu söylenebilir.

3. PROBLEM TANIMI

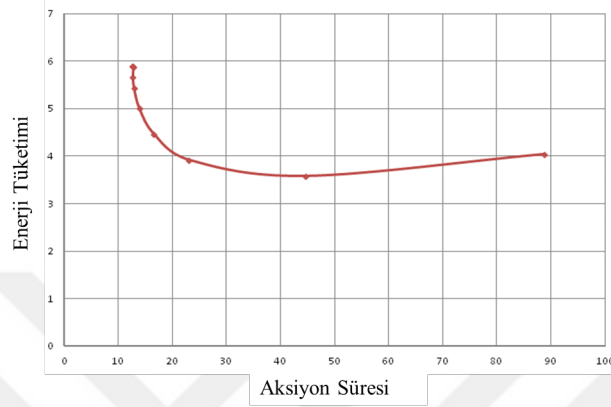
Giriş bölümünde anlatıldığı üzere bu tez çalışmasında çift tutuculu robotların kullanıldığı bir hücre konfigürasyonu ele alınmaktadır. Bu sistemde, m adet makine, giriş ve çıkış stokları doğrusal olarak dizilmiştir. Hedeflenen amaç ise en küçük çevrim zamanı ve en küçük enerji tüketimini aynı anda gerçekleştirmektir.

Çevrim zamanı bir parçanın üretilmesi için uzun dönemde gereken ortalama zaman olarak tanımlanmıştır. Bu süre hesaplanırken hem makinelerdeki işlem sürelerinin hem robotun makineleri yüklemesi ve boşaltması için gerekli zamanın ve hem de robotun parça transferleri için harcadığı sürelerin bilinmesi gerekmektedir. Robot belirli bir bilgisayar kodunu takip ettiği için robot hareketlerinin belirli bir döngü oluşturması ve bu döngünün sürekli tekrar edilmesi şeklindeki bir üretim şekli planlama açısından kolaylık sağlamaktadır. Bu durumda amaç üretim hızını maksimize edecek robot hareket döngüsünün bulunmasıdır. Diğer taraftan döngüsel çizelgelerin tek tip parça üreten sistemlerde en iyi sonucu verdiği ispatlanmıştır [24]. Literatür taramasında görüldüğü üzere 1-birim döngüler en çok ele alınan döngülerdir. Bu çalışmada da 1-birim döngüsel üretim modelleri ele alınacaktır.

Enerji tüketimi ise robot hareket hızlarına ve tutucu değişimine bağlı olarak hesaplanmaktadır. 2012 yılında yapılan deneysel çalışmada IRB4600 robotu için enerji tüketimi ve aksiyon süresi (belirli bir yolu kat etmek için gereken süre) arasındaki ilişki Şekil 3.1'de gösterildiği gibi bulunmuştur [4]. Görüldüğü üzere, aksiyon süresi en düşük seviyede (robotun hızı en yüksek seviyede), enerji tüketimi de en yüksek değerindedir. Buna rağmen literatürdeki çalışmalarda ve gerçek endüstriyel uygulamalarda ele alınan amaç, üretim hızı maksimizasyonu olduğu için genellikle robotun operasyonları en yüksek hızda yaptıkları varsayılmaktadır [40]. Ayrıca bu şekilden enerji tüketim fonksiyonunun dışbükey yapıda olduğu da görülmektedir.

Çevrim zamanını en küçük seviyede tutabilmek için, robot hareket hızlarının en yük-

sek seviyede olması gerekmektedir. Ancak bu durum, ikinci amaç fonksiyonu ile çelişmekte ve enerji tüketimini de artırmaktadır. Bunun yanında, en küçük çevrim zamanı değerine sahip olan çevrimde meydana gelen sistem beklemelerini göz önünde bulundurularak çevrim zamanını değiştirmeden daha az enerji tüketimi sağlayan ikinci bir çözüm elde etmek mümkündür.



Şekil 3.1: Robotun aksiyon süresine karşı enerji tüketimi [4]

İki kriterli olarak ele alınan bu problemde, ikinci çözüm ilk çözümü başatlayan bir çözümdür. İki kriterli bir optimizasyon probleminde, A ve B iki mümkün çözümü temsil etmek üzere, A çözümü her iki kriter açısından da B çözümünden daha kötü değilse ve bu kriterlerden en az birisi açısından B çözümünden daha iyiyse A çözümünün B çözümünü başatladığı (domine ettiği) söylenebilir. Eğer herhangi bir A çözümünü başatlayan başka bir mümkün çözüm yoksa, A çözümü başatlanmayan (Pareto etkin) bir çözümdür. Bu tanıma göre, yukarıda bahsedilen iki çözümden ikinci çözüm aynı çevrim zamanı değeri ile daha küçük bir enerji tüketimine sahip olduğundan Pareto Etkin Çözümler Kümesine (EÇK) dahil olmakta, ilk çözüm ise bu kümeye girememektedir. Çevrim zamanı daha büyük ve enerji tüketim değeri daha küçük olan başka bir çözüm de yine EÇK'ye girmektedir. Çünkü ikinci çözüm çevrim zamanı açısından tercih edilirken, üçüncü çözüm de enerji tüketimi açısından tercih edilmektedir. Bu çalışmada bir veri kümesi için ilk önce en küçük çevrim zamanına karşılık en küçük enerji tüketimini sağlayan çözüm bulunacak, sonraki aşamalarda da çevrim zamanı değeri artırılarak daha düşük enerji tüketimine sahip çözümler elde edilecektir. Böylece bir veri kümesi için EÇK'de yer alan belirli sayıda çözüm elde edilebilecektir.

3.1 Aktivite Tanımları

Robotik hücre çizelgeleme literatüründe, robot döngülerini tanımlamak için robot aktivite tanımları kullanılmıştır. Tek tutuculu robotlu sistemlerde robot bir makineyi boşalttığında yapabileceği mümkün tek aktivite bu parçayı bir sonraki makineye taşımak ve bu makineyi yüklemektir. Dolayısıyla, bu hareketler dizisi bir bütün olarak bir aktivite şeklinde tanımlanmıştır. Çift tutuculu robotlu hücrelerde ise bir makine boşaltıldıktan sonra robot diğer tutucusuyla başka aktiviteler yapabileceği için, aynı aktivite tanımı bütün olası robot döngülerini tanımlayabilmek için yetersizdir. Bu tip robotlu hücrelerde robot hareket döngülerini tanımlayabilmek için aşağıdaki tanımlar kullanılmıştır.

L_m : Robotun M_m makinesini yüklemesi aktivitesidir. Bu aktiviteyi yapabilmek için robot öncelikle mevcut bulunduğu pozisyondan M_m makinesine hareket eder. Eğer yüklemeyi yapmak için tutucusunu değiştirmesi gerekiyorsa, bu işlemi hareketi sırasında eş zamanlı olarak yapar. Bu durumda, robotun makineyi yükleyebilmek için hazır olma zamanını, bulunduğu yerden M_m 'e ulaşma zamanı ile robotun tutucusunu değiştirme zamanlarından hangisi daha büyükse, o belirler. Robot, makinenin önüne ulaştığında ve parçayı yüklemek için hazır olduğunda, makineyi yükler. Herhangi bir L_m aktivitesi sonrasında robotun tutucularından en az biri mutlaka boş olmak zorundadır.

U_m : Robotun M_m makinesini boşaltması aktivitesidir. Benzer şekilde robot yine öncelikle bulunduğu pozisyondan M_m makinesine hareket eder ve eğer gerekiyorsa, bu hareket esnasında tutucusunu değiştirir. Robotun makineyi boşaltabilmesi için makinedeki parçanın işleminin tamamlanmış olması gerekir. Değilse, robot gerektiği kadar makinenin önünde bekler. Herhangi bir makineyi boşaltabilmek için robotun tutucularından en az bir tanesi mutlaka boş olmalıdır. Herhangi bir U_m aktivitesinden sonra ise, robotun tutucularından bir tanesinde M_{m+1} makinesine yüklenecek (eğer M_m sistemdeki son makineyse çıkış stokuna bırakılacak) bir parça vardır.

Herhangi bir robot döngüsü yukarıdaki aktivitelerin uygun bir sıralaması şeklinde ta-

nımlanabilir. Uygun bir sıralama için olurluluk şartları şöyle sıralanabilir:

- Robotun zaten dolu olan bir makineyi yüklemeye çalışmaması ve zaten boş olan bir makineyi boşaltmaya çalışmaması,
- Robotun her iki tutucusu da doluysa bir başka boşaltma aktivitesi yapmaya çalışmaması,
- Robotun tutucularından herhangi birisinde M_m makinesini yüklemek için bir parça yoksa, bu makineyi yüklemeye çalışmaması,
- Bütün aktiviteler tamamlandıktan sonra, robotun döngüdeki başlangıç pozisyonuna geri dönmesi,
- Bütün makinelerin, döngünün başlangıcındaki durumlarına (dolu veya boş) gelmeleri ,
- Robotun her iki tutucusunun da yine döngü başlangıcındaki durumlarına (boş veya döngü başındakiyle aynı makineye yüklenmesi gereken bir parça taşınması) gelmiş olması gerekmektedir.

Dolayısıyla, robot hareket döngülerini ifade edebilmek için bu aktiviteleri robotun tutucularının durumlarıyla beraber ifade etmek gerekmektedir. Robotun tutucuları (g_1, g_2) şeklinde gösterilecektir. Burada g_i , robotun i tutucusunda g_i makinesine yüklenecek bir parça olduğu anlamına gelmektedir. $g_i = 0$ olması, tutucunun boş olduğunu ifade eder.

Yukarıdaki açıklamaların da ışığında, robot döngülerini ifade etmek üzere aşağıdaki aktivite gösterimleri kullanılacaktır:

- $L_m(0, g_2)$ ya da $L_m(g_1, 0)$: Robotun M_m makinesini tutucularından herhangi biriyle yüklediği aktivitedir. Tutucu durumu, aktivite hemen tamamlandıktan sonraki anı ifade etmektedir. Bu sebeple, herhangi bir L aktivitesinden sonra, tutuculardan en az bir tanesinin durumu 0 olmalıdır. Diğer tutucunun durumu $g_i = 0, 1, \dots, |\mathcal{M}| + 1$ durumlarından bir tanesi olabilir. $g_i = 0$ olması durumunda ise $L_m(0, 0)$ durumu gerçekleşmektedir.

- $U_m(m+1, g_2)$ ya da $U_m(g_1, m+1)$: Robotun M_m makinesini tutucularından herhangi birisiyle boşalttığı aktivitedir. Tutucu durumu, aktivite hemen tamamlandıktan sonraki anı ifade ettiği için herhangi bir U aktivitesinden sonra, tutuculardan bir tanesinin durumu $m+1$ olmalıdır. Diğer tutucunun durumu $g_i = 0, 1, \dots, |\mathcal{M}| + 1$ durumlarından herhangi bir tanesi olabilir.

Bu tanımlamalara göre, herhangi bir çift tutuculu m -makinelik robotik hücredeki bir 1-birim robot döngüsünde, çıkış stoku dahil olmak üzere her makine bir defa yüklenmeli (yani $m+1$ adet L aktivitesi kullanılmalı), giriş stoku dahil her makine bir defa boşaltılmalıdır (yani $m+1$ adet U aktivitesi kullanılmalıdır). Sonuçta bir döngüyü ifade edebilmek için toplamda $2(m+1)$ robot aktivitesi gereklidir. Aktivite sıralaması bir döngüyü tanımladığı için de döngüdeki ilk aktivitenin her zaman giriş stokundan yeni bir parça almak olduğu varsayılacaktır. Dolayısıyla, herhangi bir robot döngüsünün ilk aktivitesi ya $U_0(g_1, 1)$ ya da $U_0(1, g_2)$ olacaktır.

Aktivite ve döngü kavramını açıkladıktan sonra, bu aktivitelerin sıralanması ve robotun bu aktiviteleri hangi hızlarda yapacağı kararlarını vermek üzere geliştirilen matematiksel programlama formülasyonu açıklanabilir. Formülasyonun temel mantığı, m -makinelik bir döngüde, 1-birim döngülerini tanımlamak için gerekli olan $2(m+1)$ adet pozisyona toplamda $m+1$ adet L ve $m+1$ adet U aktivitesini, yukarıda bahsedilen olurluluk şartlarını sağlayacak şekilde atamaktır.

Tanımlanan aktivitelere göre, aktivitelerle ilgili dört farklı alternatif söz konusudur:

1. $L_m(0, g_2)$
2. $L_m(g_1, 0)$
3. $U_m(m+1, g_2)$
4. $U_m(g_1, m+1)$

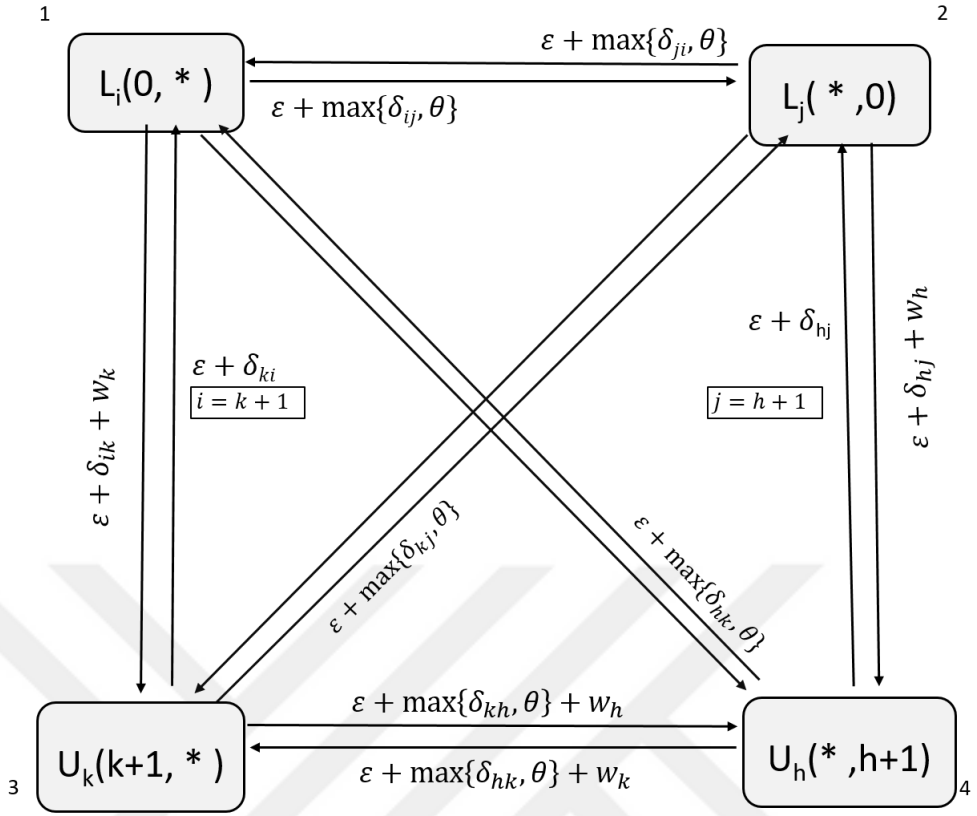
Örneğin 2 makinelik bir veri kümesi için 1 çevrimde $2 \cdot (2+1) = 6$ aktivite gerçekleşecektir.

Örnek bir aktivite sıralaması şu şekilde olabilir:

$$U_0(1, 2) \longrightarrow L_1(0, 2) \longrightarrow U_2(3, 2) \longrightarrow L_2(3, 0) \longrightarrow L_3(0, 0) \longrightarrow U_1(0, 2)$$

Herhangi bir döngünün ilk aktivitesinin robotun başlangıç stokundan bir parça alması aktivitesi olduğu varsayılmaktadır. Verilen örnekte bu işlem ilk tutucu ile gerçekleştirilmiştir. Birinci pozisyonda ilk tutucudaki parça 1. makineye, ikinci tutucudaki parça ise 2. makineye yüklenmek üzere beklemektedir. İkinci pozisyonda ilk tutucu ile 1. makineye yükleme aktivitesi gerçekleşmiştir. Dolayısıyla birinci tutucu boşalmıştır. Üçüncü ve dördüncü pozisyonlardaki aktiviteler 2. makinede gerçekleşmektedir. Bunlardan ilki makineden boşaltma ikinci ise yükleme aktiviteleridir. Farklı iki tutucu ile gerçekleşen bu aktiviteler arasında tutucu değişimi yapılmış, böylece dördüncü pozisyon sonunda 2. tutucu aktif hale gelmiştir. Ancak, beşinci pozisyondaki çıkış stokuna yükleme aktivitesi ilk tutucu ile gerçekleştirileceğinden dördüncü ve beşinci pozisyonlar arası da tutucu değişimi yapılmaktadır. Benzer şekilde altıncı pozisyondaki 1. makineden boşaltma işlemi de bir önceki pozisyondan farklı tutucu ile yapılacağından bu pozisyonlar arasında da aktif tutucu değişimi olmaktadır. Döngüsel üretim yapıldığından robot altıncı aktiviteden sonra ilk aktivitesi olan giriş stokundan parça alma işlemini tekrarlayacaktır. Görüleceği üzere, döngü başında ilk makine boş olarak yeni bir parçanın yüklenmesini beklerken, ikinci makinede bir parça yüklüdür ve işlemi devam etmektedir. Yine döngü başında, robotun ilk tutucusu boşken, ikinci tutucusunda ikinci makineye yüklenecek bir parça bulunmaktadır. Döngü tamamlanıp robot yine giriş stoğuna yöneldiğinde makinelerin ve robot tutucularının durumları döngü başındakiyle aynıdır. Dolayısıyla, bu robot hareket sırası olurlu bir döngü tanımlamaktadır ve sürekli tekrar edilebilir. Ayrıca, döngü boyunca giriş stoğundan yeni bir parça alınıp, çıkış stoğuna da üretilmiş bir parça bırakılmıştır. Bu sebeple, bu aktivite sıralaması bir 1-birim döngüsü tanımlamaktadır. Ardışık iki makineden boşaltma aktivitesi aynı tutucularla gerçekleşemeyeceğinden, robot başlangıç pozisyonuna dönerken de aktif tutucu değişimi yapmaktadır.

Bu aktivitelerin, döngüde birbirlerini takip etmeleri durumunda ortaya çıkacak süreler parametrik olarak yazılabilmektedir. Bu mesafeler Şekil 3.2’de gösterilmiştir. Dolayısıyla, matematiksel model hem bu aktiviteleri pozisyonlara atamakta hem de bu hareketleri gerçekleştirirken robotun kullandığı hızı belirlemektedir.



Şekil 3.2: Robot döngüsünde olabilecek dört alternatif aktivite arası mesafeler

Şekil 3.2’de görüleceği üzere aktivite arası geçen zamanlar ε , δ_{ij} ve w_i cinsinden yazılmıştır. Burada ε robotun yükleme ve boşaltma süresini belirten parametredir. δ_{ij} ise i ve j makineleri arasında robotun hareket süresini gösteren karar değişkenidir. w_i ise robotun boşaltma işleminden önce i makinesi önünde beklemesi gereken süreyi belirtmekte ve 0 ya da daha büyük değerler almaktadır. Şekil 3.2’e göre, tutucu değişiminin gerektiği aktiviteler arasında geçen zaman hesaplanırken δ_{ij} ve θ sürelerinden büyük olanı dikkate alınmıştır. Daha önce de belirtildiği gibi θ burada tutucu değişimi sırasında geçen süreyi ifade etmektedir. Örneğin, 1. tip aktivite olan $L_i(0, *)$ aktivitesinden sonra 2. tip $L_j(*, 0)$ aktivitesinin gerçekleşmesi için robot bulunduğu i makinesinden j makinesine gider bu esnada da aktif tutucusunu değiştirmek zorundadır. Bu değişikliği makine hareket zamanıyla aynı anda yapacağı için toplam süre hareket süresi ile tutucu değişim süresinden büyük olanına eşittir ($\max\{\delta_{ij}, \theta\}$). Bu süreye j makinesinin yüklenme süresi de eklenmektedir (ε).

3.2 Örnek Aktivite Sıralaması

Tanımlanan problemi daha iyi açıklayabilmek için aşağıdaki örnek kullanılmıştır. 5 makineli veri kümesi için kullanılan parametreler şu şekildedir:

- $\varepsilon = 1, \theta = 1.5, VUB = 3, VLB = 1$
- $P_1 = 50, P_2 = 60, P_3 = 20, P_4 = 40, P_5 = 30$
- Ardışık makineler arası uzaklık 5 birimdir.

Aşağıda 5 makineli bu örnek için rastgele seçilen olurlu bir aktivite sıralaması bulunmaktadır. Görüleceği gibi 1-birim olan bu döngü için çevrimde $2 \cdot (5+1) = 12$ aktivite bulunmaktadır.

$$U_0(1,0) \Rightarrow L_1(0,0) \Rightarrow U_3(4,0) \Rightarrow L_4(0,0) \Rightarrow U_5(6,0) \Rightarrow L_6(0,0) \Rightarrow U_4(5,0) \Rightarrow L_5(0,0) \Rightarrow U_2(3,0) \Rightarrow L_3(0,0) \Rightarrow U_1(2,0) \Rightarrow L_2(0,0)$$

Bu veri kümesi ve aktivite sıralaması için robot hızlarının $VUB=3$ değerine eşit olduğu varsayılmıştır. Bu durumda CT çevrim zamanı 76 değerindedir ve T_p ile belirtilen pozisyon başlama zamanları ise aşağıdaki gibi hesaplanmıştır.

$$T_1 : 0.00, T_2 : 2.67, T_3 : 9.67, T_4 : 12.33, T_5 : 15.00, T_6 : 17.67, T_7 : 53.33, T_8 : 56.00, T_9 : 62.00, T_{10} : 64.67, T_{11} : 69.00, T_{12} : 71.67$$

T_1 ilk aktivitenin başlama zamanıdır, bu sebeple 0 olarak belirlenir. T_2 yani ikinci pozisyondaki aktivitenin başlama zamanı hesaplanırken $\varepsilon = 1$ ve $\frac{dist_{01}}{3} = 1.67$ değerleri dikkate alınır, $T_2 = 2.67$ olarak hesaplanır.

Üçüncü pozisyonda makineden boşaltma işlemi gerçekleşmektedir. 3. makinedeki parçanın işlem süresinin henüz tamamlanmamış olma ihtimalinden dolayı T_3 iki aşamalı olarak hesaplanır. İlk olarak 3. makineye yükleme aktivitesinin sırası kontrol edilir. L_3 aktivitesi onuncu pozisyonda yani daha sonra gerçekleştiğinden T_3 aşağıdaki gibi hesaplanır.

$T_3 = \max = \{T_2 + \varepsilon + \frac{dist_{13}}{3}, -CT + T_{10} + \varepsilon + P_3\}$ max terimi içerisindeki ilk ifade, robotun makine önünde beklemesi gerekmeyen durumlarda geçerli olmaktadır. İkinci ifade ise P_3 işlem süresinin onuncu pozisyonda başladıktan sonra üçüncü pozisyona dönene kadar bitmediği yani robot beklemesinin gerçekleştiği durumlarda T_3 değerini belirleyecektir. Ancak CT ve T_{10} değerleri henüz bilinmemektedir. Başlangıçta bu değerler 0 olarak hesaplanacağından T_3 değeri sonraki iterasyonlarda belirlenmiş olacaktır.

Dördüncü pozisyonda yükleme aktivitesi gerçekleştiğinden $T_4 = T_3 + \varepsilon + \frac{dist_{34}}{3}$ şeklinde hesaplanır. Beşinci ve altıncı aktivitelerin başlama zamanları da yukarıda açıklanan şekilde hesaplanmaktadır. Yedinci aktiviteye bakıldığında 4. makineden boşaltma işlemi olduğu görülmektedir. 4. makineye yükleme işlemi ise yedinci pozisyondan daha erken olan dördüncü sırada gerçekleşmektedir. Bu durumda T_7 değeri şöyle bulunur: $T_7 = \max = \{T_6 + \varepsilon + \frac{dist_{64}}{3}, T_4 + \varepsilon + P_4\}$ Burada CT değerinin hesaplamaya dahil edilmemesinin nedeni, 4. makine için işlem süresinin bir çevrim dahilinde bitmiş olmasıdır. Diğer pozisyonların başlama zamanları da benzer şekillerde hesaplanmaktadır. Bu hesaplamaların detayları Bölüm 4'te verilmiştir.

İşlem süresi bitmeyen parçaları makineden boşaltmak için p pozisyonunda robotun beklediği $Wait_p$ ve robotun boşaltma işlemine başlamadan önce işlenmesi biten parçaların bekleme $Idle_p$ süreleri meydana gelmektedir. Bu süreler ise şöyledir:

$$Wait_3 : 2.67 , Wait_7 : 31.33 , Idle_5 : 4 , Idle_9 : 5.3 , Idle_{11} : 15.33$$

Bu süreler hesaplanırken ise bekleme olmasaydı aktivitenin ne zaman başlamış olacağı belirlenir. Örneğin 3. makinedeki bekleme süresini gösteren $Wait_3 : 2.67$ değeri gerçekleşmeseydi $T_3 = T_2 + \varepsilon + \frac{dist_{13}}{3} = 2.67 + 1 + \frac{10}{3} = 7$ olacaktı. Ancak çevrim zamanı hesaplandıktan sonra T_3 değerinin 9.67 olduğu görülmektedir. Bu durumda $Wait_3 : 9.67 - 7 = 2.67$ olarak hesaplanmıştır. Makine beklemesi yani $Idle_p$ değeri ise aynı makineye yapılan yükleme işleminin başlama zamanına göre belirlenir. Örneğin beşinci pozisyonda oluşan $Idle_5$ makine bekleme süresi hesaplanırken sekizinci pozisyondaki L_5 aktivitesinin başlama zamanı ve işlem süresi göz önünde bulundurulur. $T_8 = 56$ ve $P_5 = 30$ olduğundan, robotun yükleme süresi de eklendiğinde $56+30+1 = 87$ birim sürede tamamlanması beklenmektedir. Döngüsel üretim olduğundan bu süre bir sonraki döngüye devredilecek ve T_5 için beklenen değer $87-76= 11$ olarak hesaplanacaktır. Ancak görüldüğü üzere $T_5 = 15$ olarak verilmiştir. Bu durum, aradaki 4 birimlik

fark kadar makine beklemesi olduđunun göstergesidir.



4. GELİŞTİRİLEN ÇÖZÜM YÖNTEMLERİ

Bu bölümde, Bölüm 3’te anlatılan problemin çözümü için geliştirilen çözüm yöntemleri anlatılmıştır.

4.1 Matematiksel Modeller

Önceki bölümde anlatıldığı gibi, en küçük enerji tüketimi ve çevrim zamanı değerine sahip aktivite sıralaması ve robot hız değerlerinin saptanması için bir matematiksel model geliştirilmiştir. \mathcal{M} kümesinde bulunan her bir m makinesi farklı P_m işlem süresine sahiptir. Ortamda bulunan m ve n makineleri arasındaki uzaklık ise $dist_{mn}$ olarak verilmiştir. Robotun bir makineye parça yükleme ve makineden parça boşaltma sırasında geçen süre ε ile tanımlanmıştır. Ardışık pozisyonlarda tutucu değişimi sırasında geçen süre ise θ olarak gösterilmektedir. Kurulan matematiksel model ile \mathcal{P} kümesinde bulunan her bir p pozisyonu için, robotun m makinesinde \mathcal{K} kümesinde bulunan aktivite tiplerinden hangisini gerçekleştireceği belirlenecek, böylece olurlu bir aktivite sıralaması elde edilecektir. Ayrıca robot bekleme zamanları da göz önünde bulundurularak robot hareket hızları da belirlenecektir. Modelde kullanılan kümeler, parametreler ve karar değişkenleri aşağıda gösterilmiştir.

Kümeler

- \mathcal{M} = Makineler kümesi
- \mathcal{K} = Yükleme/boşaltma alternatifleri kümesi
- $\mathcal{K}\mathcal{G}$ = Tutucu değişimi gerektiren ardışık yükleme ve boşaltma alternatifleri kümesi
- \mathcal{P} = Aktivitelerin atanabileceği pozisyonların kümesi

Parametreler

- ε = Makine yükleme/boşaltma zamanı
 θ = Robotun aktif tutucusunu değiştirme süresi
 $dist_{mn}$ = M_m makinesinden M_n ($m \neq n$) makinesine mesafe
 P_m = M_m makinesindeki parça işleme süresi
 VUB = Robotun olabilecek en yüksek hızı
 VLB = Robotun olabilecek en düşük hızı
 C = Enerji tüketim fonksiyonu sabiti. Bu sabit, birim mesafe başına etki eden ağırlık ve sürtünme kuvvetlerinden kaynaklanmaktadır
 G = Her tutucu değişiminde ortaya çıkan enerji tüketim fonksiyonu sabiti
 B = Modelde kullanılan yeterince büyük bir sayı
 s = Modelde kullanılan yeterince küçük bir sayı

Karar değişkenleri

$$y_{mpk} = \begin{cases} 1 & \text{Eğer } p \in \mathcal{P} \text{ pozisyonunda } m \in \mathcal{M} \text{ makinesinde } k \in \mathcal{K} \text{ tipi bir} \\ & \text{aktivite gerçekleşiyorsa} \\ 0 & \text{Diğer durumda} \end{cases}$$

$$x_{mn(p-1)} = \begin{cases} 1 & \text{Eğer } (p-1) \in \mathcal{P} \text{ pozisyonundaki } U_m \text{ ya da } L_m \text{ } m \in \mathcal{M} \text{ aktivite-} \\ & \text{leri ile ve } p \in \mathcal{P} \text{ pozisyonundaki } U_n \text{ ya da } L_n \text{ } n \in \mathcal{M} \text{ aktiviteleri} \\ & \text{arasında tutucu değişimi gerçekleşiyorsa} \\ 0 & \text{Diğer durumda} \end{cases}$$

- T_p = $p \in \mathcal{P}$ pozisyonundaki aktivitenin başlama zamanı
 v_{mnp} = $m \in \mathcal{M}$ ve $n \in \mathcal{M}$ makineleri arasında $p \in \mathcal{P}$ pozisyonunda robotun hızı
 δ_{mnp} = $m \in \mathcal{M}$ ve $n \in \mathcal{M}$ makineleri arasında $p \in \mathcal{P}$ pozisyonunda robotun hareket süresi

4.1.1 Karışık Tamsayılı Doğrusal Olmayan Matematiksel (MINLP) Model:

Daha önce de anlatıldığı gibi ele alınan problem birbiriyle çelişen iki farklı amaç fonksiyonuna sahiptir. Bunlardan ilki çevrim zamanının en küçüklenmesi (4.1) diğeri ise enerji tüketiminin olabilecek en düşük seviyede tutulmasıdır (4.2).

$$\text{Min } CT \quad (4.1)$$

$$\text{Min } \sum_m \sum_n \sum_p C \cdot \text{dist}_{mn} v_{mnp}^a + \sum_m \sum_n \sum_p x_{mnp} G \quad (4.2)$$

Görüleceği üzere amaç fonksiyonu (4.2) değeri iki ayrı terimin toplanması ile elde edilmektedir. Bunlardan ilki robotun hızı ve bu hız ile aldığı yol uzunluğundan kaynaklanan enerji tüketimidir. Bu terimde bulunan C sabiti ise ağırlık ve sürtünme kuvvetlerinden kaynaklanmaktadır. a değeri de hız değişkeni v_{mnp} 'nin ise üssel kuvvetidir. Enerji tüketim fonksiyonunun ikinci terimi ise tutucu değişiminden dolayı oluşan enerji tüketimini göstermektedir. Toplam tutucu değişimi, aktif tutucunun değişmesi için gereken enerji sabiti G ile çarpılarak bu terim elde edilmektedir.

Modeldeki kısıtlar ise aşağıdaki gibidir.

$$\delta_{mn(p-1)} \geq \theta - B(1 - x_{mn(p-1)}) \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p \neq 1 \quad (4.3)$$

$$\delta_{m1p} \geq \theta - B(1 - x_{m1p}) \quad \forall m, n \in \mathcal{M}, p = |\mathcal{P}| \quad (4.4)$$

$$\delta_{mnp} \geq \text{dist}_{mn} / v_{mnp} \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} \quad (4.5)$$

$$T_p \geq T_{p-1} + \varepsilon + \delta_{mn(p-1)}$$

$$-B(2 - \sum_h y_{m(p-1)h} - \sum_k y_{nph}) \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p \neq 1 \quad (4.6)$$

$$CT \geq T_p + \varepsilon + \delta_{m1p} - B(1 - \sum_k y_{mpk}) \quad \forall m, n \in \mathcal{M} : n = 1, p = |\mathcal{P}| \quad (4.7)$$

$$T_p \geq T_r + \varepsilon + P_m - B(2 - \sum_k y_{mpk} - \sum_h y_{mrh}) \quad \forall m \in \mathcal{M}, \forall p, r \in \mathcal{P} : p > r \quad (4.8)$$

$$CT + T_p \geq T_r + \varepsilon + P_m - B(2 - \sum_k y_{mpk} - \sum_h y_{mrh}) \quad \forall m \in \mathcal{M}, \forall p, r \in \mathcal{P} : p < r \quad (4.9)$$

$$y_{113} = 1 \quad (4.10)$$

$$y_{mp3} = 0 \quad \forall m \in \mathcal{M}, p = |\mathcal{P}|, \quad (4.11)$$

$$\sum_{s=r+1}^{p-1} y_{(n+1)s1} + 1 \geq y_{nr3} + y_{mp3} \quad \forall m, n \in \mathcal{M} : n \neq |\mathcal{M}|, \quad \forall p \in \mathcal{P} : p > r \quad (4.12)$$

$$\sum_{s=r+1}^P y_{(n+1)s1} + \sum_{s=1}^{p-1} y_{(n+1)s1} + 1 \geq y_{nr3} + y_{mp3} \quad \forall m, n \in \mathcal{M} : n \neq |\mathcal{M}|, \quad \forall p \in \mathcal{P} : p < r \quad (4.13)$$

$$\sum_{s=r+1}^{p-1} y_{(n+1)s2} + 1 \geq y_{nr4} + y_{mp4} \quad \forall m, n \in \mathcal{M}, \quad (4.14)$$

$$\forall p \in \mathcal{P} : p > r, \quad (4.15)$$

$$\sum_{s=r+1}^s y_{(n+1)s2} + \sum_{s=1}^{p-1} y_{(n+1)s2} + 1 \geq y_{nr4} + y_{mp4} \quad \forall m, n \in \mathcal{M} : n \leq m, \quad \forall p \in \mathcal{P} : p < r, \quad (4.16)$$

$$\sum_{p=1}^{|\mathcal{P}|} y_{mp3} = \sum_{p=1}^P y_{(m+1)p1} \quad \forall m \in \mathcal{M} : m \neq |\mathcal{M}| \quad (4.17)$$

$$\sum_{p=1}^{|\mathcal{P}|} y_{mp4} = \sum_{p=1}^P y_{(m+1)p2} \quad \forall m \in \mathcal{M} : m \neq |\mathcal{M}| \quad (4.18)$$

$$\sum_{m=1}^{|\mathcal{M}|} (\sum_{k=1}^4 y_{mpk}) = 1 \quad \forall p \in \mathcal{P} \quad (4.19)$$

$$\sum_{k=3}^4 (\sum_{p=1}^{|\mathcal{P}|} y_{mpk}) = 1 \quad \forall m \in \mathcal{M} : m \neq |\mathcal{M}| \quad (4.20)$$

$$\sum_{k=1}^2 (\sum_{p=1}^{|\mathcal{P}|} y_{mpk}) = 1 \quad \forall m \in \mathcal{M} : m \neq 1 \quad (4.21)$$

$$x_{nmp(p-1)} + 1 \geq y_{m(p-1)k} + y_{nph} \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} : p \neq 1, \quad \forall (k, h) \in \mathcal{KG} \quad (4.22)$$

$$x_{1n1p} \geq y_{nph} \quad \forall m, n \in \mathcal{M}, p = |\mathcal{P}|, \quad \forall (3, h) \in \mathcal{KG} \quad (4.23)$$

$$v_{mn(p-1)} \geq VLB(\sum_k^{|\mathcal{K}|} y_{m(p-1)k} + \sum_k^{|\mathcal{K}|} y_{nph} - 2) \quad \forall m, n \in \mathcal{M}, \quad \forall p \in \mathcal{P} : p \neq 1 \quad (4.24)$$

$$v_{m1p} \geq VLB(\sum_k^{|\mathcal{K}|} y_{mpk} - 1) \quad \forall m \in \mathcal{M}, p = |\mathcal{P}| \quad (4.25)$$

$$v_{mnp} \leq VUB \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} \quad (4.26)$$

$$y_{mpk}, x_{mnp} \in \{0, 1\} \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} \quad \forall k \in \mathcal{K} \quad (4.27)$$

$$v_{mnp}, \delta_{mnp}, T_p, CT \geq 0 \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} \quad (4.28)$$

İki amaç fonksiyonuna sahip olan bu modelin çözülmesi için ε -kısıt yöntemine başvurulmuştur. İlk olarak (4.3) ile (4.28) arasındaki kısıtlar (4.1). amaç fonksiyonu ile çözülmüştür. Matematiksel modelin çözümü sonucu elde edilen en iyi çevrim zamanı

değeri olan CT^* , çevrim zamanı üst limiti \overline{CT} olarak (4.30) kısıtı ile modele eklenmiştir. İkinci aşamada ise (4.2). amaç fonksiyonu (4.29) şeklinde düzenlenmiştir. CT değerinin amaç fonksiyonu 4.29'e eklenmesiyle, aynı enerji tüketim değerine sahip olabilecek farklı çözümlerden en küçük çevrim zamanına sahip çevrimin elde edilmesi sağlanmıştır. Yeterince küçük bir sayı olan s değeri ile çarpılması da çevrim zamanı amacının arka plana atılmasını, böylece enerji tüketiminin ön planda kalmasını sağlamıştır. Sonuç olarak aşağıda belirtilen matematiksel model oluşturulmuştur.

$$\text{Min} \sum_m \sum_n \sum_p C \cdot \text{dist}_{mn} v_{mnp}^a + \sum_m \sum_n \sum_p x_{mnp} G + s \cdot CT \quad (4.29)$$

Öyle ki;

$$CT \leq \overline{CT} \quad (4.30)$$

$$4.3 \text{ ile } 4.28 \text{ arası kısıtlar} \quad (4.31)$$

Amaç fonksiyonu 4.29'ün (4.3) ile (4.28) arasındaki kısıtlar ile birlikte çözülmüştür. Diğer bir ifadeyle ilk olarak, olası en küçük çevrim zamanı değeri bulunmuş olup, bu çevrim zamanını sağlayan en küçük enerji tüketimi değeri hesaplanmıştır. Ayrıca 4.30. kısıtta farklı \overline{CT} değerleri kullanılarak farklı pareto etkin çözümler elde edilebilmektedir.

Matematiksel modelde bulunan (4.3), (4.4) ve (4.5) numaralı kısıtlar, ardışık aktivitelere atanan iki makine arasında robotun yolculuk zamanını belirler. Kısıt (4.3) ve (4.4) ardışık iki aktivite arasında tutucu değişimi olduğunda geçen sürenin en az θ kadar olmasını sağlar. Tutucu değişimi gerektirmeyen aktivitelere veya θ değerinin yeterince küçük olduğu şartlarda bu iki kısıt gereksiz olacaktır. Böylece, kısıt (4.5) ile iki makine arasında geçen sürenin hıza bağlı olarak değeri ifade edilmiş olur.

Kısıt (4.6) önceki aktivite tamamlanmadan, bir sonraki aktivitenin başlayamaması kuralını sağlayan kısıttır. Şöyle ki p pozisyonundaki aktivitenin başlaması için, $(p-1)$ pozisyonunda başlayan aktiviteden sonra (T_{p-1}) en az yükleme ya da boşaltma için gereken süre ε ve iki makine arasındaki yolculuk zamanı kadar süre $(\delta_{mn(p-1)})$ geçmelidir. Kısıt (4.7) ise benzer bir mantıkla oluşturulmuş olup çevrim zamanını belirleyen kısıt-

tır. Çevrim süresinin tamamlanması için, son pozisyona atanan aktiviteden sonra (T_p) en az ε kadar ve robotun son pozisyonundaki m makinesi ile giriş stoku arasında almış olduğu kadar süre (δ_{m1p}) geçmelidir. Amaç fonksiyonu (4.29)'de çevrim zamanı değeri bulunduğu için bu iki kısıt doğru çalışmakta ve olası en küçük pozisyon başlama ve çevrim zamanı değerlerini vermektedir.

(4.8) ve (4.9) numaralı kısıtlar bir parçanın bir makinede boşaltılmadan önce makinedeki işleminin bitmesi gerektiğini yani o parçanın makinede en az işlem süresi (P_m) kadar zaman geçirmesi gerektiğini ifade eder. Bunlardan ilkinde, makinenin yükleme aktivitesinin aktivite sıralamasında boşaltmadan daha önce yer aldığı durum için, ikincisi ise boşaltmanın yüklemeye önce yer aldığı durum için yazılmıştır. Döngüsel çizelgeleme modelleri ele alındığı için bazı robot döngülerinde bir makine döngünün bir önceki tekrarında yüklenip bir sonraki tekrarında boşaltılmış olabilir. Bu durumda, robot hareket sıralamasında parçanın boşaltılması yüklenmesinden önceymiş gibi görülebilir. Kısıt (4.9) ikinci durumu doğru bir şekilde modellemek içindir. İki kısıt arasındaki fark, ikinci durumda çevrim zamanının kısıta eklenmiş olmasıdır. Amaç fonksiyonunda çevrim zamanı küçük bir katsayıyla da olsa cezalandırıldığı için bu kısıt doğru çalışmaktadır.

Kısıt (4.10), ilk pozisyona giriş stokundan yeni bir parçanın alınması aktivitesini atar. Bu bir döngü olduğu için, döngünün hangi aktiviteyle başladığının önemi yoktur. Önemli olan, aktivitelerin nasıl sıralandığıdır. Aktivite sıralamasının 3 numaralı aktiviteyle başladığı varsayıldığı için, son pozisyona 3 numaralı aktivitenin gelmemesi gerekmektedir. Bu koşul kısıt (4.11) ile sağlanmaktadır. (4.12) ve (4.13) numaralı kısıtlar, Şekil 3.2'de verilen aktivitelerden 3 numaralı aktivitenin ardışık iki tekrarı arasında 1 numaralı aktivitenin en az bir defa yapılmasını sağlarlar. Bu kısıtlardan ilki, döngü şeklindeki aktivite sıralamasının bir tarafını ele alırken, ikincisi de diğer tarafını ele almaktadır. Bu kısıtlar ortaya çıkan aktivite sıralamasının olurlu olması için gereklidir.

3 numaralı aktivite ile 1 numaralı aktivite arasındaki ilişkiye benzer şekilde (4.15) ve (4.16) numaralı kısıtlar, Şekil 3.2'de verilen aktivitelerden 4 numaralı aktivitenin ardışık iki tekrarı arasında 2 numaralı aktivitenin en az bir defa yapılmasını sağlarlar.

Kısıtlar (4.17) ve (4.18) döngüde kullanılan 1 numaralı aktivite sayısını 3 numaralı

aktivite sayısına, 4 numaralı aktivite sayısını da 2 numaralı aktivite sayısına eşitlemektedir. Diğer bir ifadeyle ilk ya da ikinci tutucu ile boşaltılan parça sayısı bu tutucu ile yükleme yapılan parça sayısına eşit olmalıdır. Bu da sonuçta çıkan döngünün olurlu olabilmesi için gereklidir. Kısıt (4.19), her pozisyona bir aktivitenin atanmasını sağlar. Kısıt (4.20), çıkış stoku hariç her makinenin döngü boyunca bir defa boşaltılmasını, Kısıt (4.21) ise giriş stoku hariç her makinenin bir defa yüklenmesini sağlar.

(4.22) ve (4.23) numaralı kısıtlar, robotun ardışık pozisyonlara atanan aktiviteler arasında tutucu değişimi yapıp yapmadığını belirlemektedir. Bu kısıtlardan ilki aynı çevrim içindeki tutucu değişimlerini kontrol ederken, ikincisi bir çevrimin son aktivitesi ile sonraki çevrimin ilk aktivitesi arasında değişim olup olmadığını belirler.

Kısıt (4.24) ve (4.25), robot eğer iki makine arasında yol almışsa, bu pozisyona ait hız değerinin alt limite uymasını sağlar. Bu kısıtlardan ilki, ilk ve son pozisyonlar için, ikincisi ise son pozisyondan tekrar giriş stokuna hareket için hız alt limitini sağlamaktadır. Amaç fonksiyonu (4.29)'de hız ve uzaklığa bağlı terimlerin bulunması sebebiyle, VLB değerinin 0'dan farklı olduğu durumlar için bu iki kısıt gereklidir. Kısıt (4.26) ise robot hareketlerinin belirli bir üst limitin altında olmasını sağlamaktadır. Son olarak kısıt (4.27) ve (4.28) de karar değişkenlerine ait işaret kısıtlarıdır.

Karma Tamsayılı Doğrusal Olmayan (MINLP) bir yapıda olan bu model GAMS arayüzü ile kodlanmış ve BARON çözücüsü ile denemeler yapılmıştır. Yapılan deneysel çalışmalardan 5 bölümünde bahsedilecektir. Fakat, bu modelde çözüm süresinin oldukça uzun olmasından dolayı bu modele alternatif olarak Karma Tamsayılı İkinci Derece Konik Programlama (MISOCP) modeli geliştirilmiştir. Çalışmanın bu kısmında geliştirilen MISOCP modeli açıklanacaktır.

4.1.2 Tamsayılı İkinci Derece Konik Programlama (MISOCP) Modeli:

Bir önceki bölümde görüldüğü üzere amaç fonksiyonu (4.29) ve kısıt (4.5) doğrusal olmayan terimler içermektedir. Bu nedenle, bu ifadelerin ikinci dereceden konik programlama yöntemleri kullanılarak ifade edilerek CPLEX çözücüsünde çözdürebilmek üzere yeni bir matematiksel model geliştirilmiştir.

Amaç Fonksiyonu

Amaç fonksiyonunda bulunan, v_{mnp}^a terimini ikinci dereceden şekilde yazabilmek için aşağıdaki gibi yeni karar değişkenleri tanımlanmıştır.

- $v_{mnp} = v_{mnp}^a$.
- $q = 1$ (karesel yazım formatını sağlayabilmek için q değişkeni tanımlanmıştır ve değeri 1'e eşitlenerek çarpımlarda etkisiz olması sağlanmıştır).

Tanımlanan değişkenler yardımıyla, enerji fonksiyonundak üssel kuvvet olan a 'nın aldığı bazı tam sayı değerlerine göre güncellenen amaç fonksiyonu ve kısıtlar aşağıda verilmiştir.

- $a = 2$:
Üssel kuvvet $a = 2$ olduğunda, kısıt (4.33) ve (4.34) modele eklenmelidir. q değişkeninin, eşitsizliğin sağ tarafına çarpan olarak eklenmesiyle, her iki taraf da karesel forma dönüştürülmüştür.

$$\text{Min } \sum_m \sum_n \sum_p C \cdot \text{dist}_{mn} v_{mnp} + \sum_m \sum_n \sum_p x_{mnp} G + s \cdot CT \quad (4.32)$$

Öyle ki;

$$q = 1 \quad (4.33)$$

$$v_{mnp} \cdot v_{mnp} \leq v_{mnp} \quad ; \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} \quad (4.34)$$

- $a = 3$:

Üssel kuvvet a 'nın 3 olduğu durumlar için amaç fonksiyonu (4.32) gibidir. Eklenenecek kısıtların formüle edilme adımları aşağıda verilmiştir.

$$v_{mnp} \geq v_{mnp}^3 \Rightarrow v_{mnp} \geq v_{mnp}^2 \cdot \frac{\text{dist}_{mn}}{\delta_{mnp}} \Rightarrow v_{mnp} \cdot \frac{\delta_{mnp}}{\text{dist}_{mn}} \geq v_{mnp} \cdot v_{mnp}$$

$$v_{mnp} \cdot v_{mnp} \leq v_{mnp} \frac{\delta_{mnp}}{\text{dist}_{mn}} \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} \quad (4.35)$$

Sonraki adımlarda, doğrusal olmayan $\frac{dist_{mn}}{\delta_{mnp}} = v_{mnp}$ terimi ikinci derece denklem şeklinde formüle edilmiştir.

$$\frac{dist_{mn}}{\delta_{mnp}} = v_{mnp} \Rightarrow dist_{mn} = v_{mnp} \cdot \delta_{mnp} \Rightarrow 4dist_{mn} = 4v_{mnp} \cdot \delta_{mnp}$$

$$\Rightarrow 4dist_{mn} - 2v_{mnp} \cdot \delta_{mnp} = 2v_{mn} \cdot \delta_{mnp}$$

$$\Rightarrow (v_{mnp}^2 + \delta_{mnp}^2) + 4dist_{mn} - 2v_{mnp} \cdot \delta_{mnp} = (v_{mnp}^2 + \delta_{mnp}^2) + 2v_{mn} \cdot \delta_{mnp}$$

$$\Rightarrow (v_{mnp}^2 + \delta_{mnp}^2 - 2v_{mnp} \cdot \delta_{mnp}) + 4dist_{mn} = (v_{mnp}^2 + \delta_{mnp}^2 + 2v_{mnp} \cdot \delta_{mnp})$$

$$\Rightarrow (v_{mnp} - \delta_{mnp})^2 + 4dist_{mn} = (v_{mnp} + \delta_{mnp})^2 \quad (4.36)$$

$$(v_{mnp} - \delta_{mnp})^2 + 4dist_{mn} \leq (v_{mnp} + \delta_{mnp})^2$$

$$\forall m, n \in \mathcal{M}, \quad \forall p \in \mathcal{P} \quad (4.37)$$

Hız üstel kuvveti $a = 3$ iken, amaç fonksiyonu (4.32) ve kısıtlar (4.35) ve (4.37) modele eklenmelidir.

- $a = 4$:

Yeni tanımlanacak, v_{mnp} basamak değişkeni ile, kısıtların ikinci dereceden yeniden yazımı şu şekildedir:

$$w_{mnp} = v_{mnp}^2$$

$$vv_{mnp} = w_{mnp}^2$$

$$v_{mnp} \cdot v_{mnp} \leq w_{mnp} \quad \forall m, n \in \mathcal{M} \forall p \in \mathcal{P} \quad (4.38)$$

$$w_{mnp} \cdot w_{mnp} \leq vv_{mnp} \quad \forall m, n \in \mathcal{M} \forall p \in \mathcal{P} \quad (4.39)$$

Üstel kuvvet 4 olduğunda uygulanan adımlar, $a = 2$ iken uygulanan işlemlerin tekrarlanması şeklindedir. Benzer şekilde, a 'nın diğer tam sayı değerleri için,

açıklanan formülasyonlar tekrar edilerek ikinci dereceden kısıt sistemleri elde edilebilmektedir.

Kısıt (4.5)

Kısıt (4.5)'deki doğrusal olmayan ifadenin farklı bir şekilde yazımı şu şekildedir.

$$v_{mnp} \geq dist_{mn}/\delta_{mnp} \quad \forall m, n \in \mathcal{M}, \forall p \in \mathcal{P} \quad (4.40)$$

Bu ifadenin dönüştürülme adımları ise (4.36) denklem grubuyla üssel kuvvet $a=3$ durumu için bir önceki aşamada gösterildiği gibidir. Karma tamsayılı doğrusal olmayan modelde kısıt (4.5) yerine eşitsizlik (4.37) eklendiğinde tamsayılı ikinci derece konik programlama modeli elde edilmektedir. Amaç fonksiyonun doğrusal, kısıtların ikinci derece konik eşitsizlikler olduğu bu formülasyonları CPLEX çözücüsü dal-sınır metodu ve "iç bölge yöntemleri" kullanarak çözülebilmektedir. Yapılan denemelerde MISOCP modelinin çözüm zamanını önemli ölçüde iyileştirdiği görülmüştür. Ancak makine sayısına bağlı olarak problem karmaşıklığının çok hızlı bir şekilde artmasıyla makine sayısının büyük olduğu problemler için MISOCP modeliyle makul sürede çözüm bulmanın mümkün olmadığı görülmüştür. Bu sebeple bir sonraki bölümde sezgisel bir algoritma önerilmiştir.

4.2 Etkin Çözüm Türetme Algoritması (ETA)

Yukarıda önerilen matematiksel programlama modellerinin çözebildiği problem büyüklükleri düşünülerek ve karar vericiye geniş bir etkin çözüm kümesi sunma gerekliliği dikkate alınarak hızlı çalışan bir sezgisel algoritma geliştirilmiştir. ETA ismi verilen algoritmanın amacı kısa sürede çok sayıda yaklaşık etkin çözümler bulmaktır. "Yaklaşık etkin çözüm" ifadesi elde edilecek çözümün sezgisel yolla bulunmuş olması nedeniyle kullanılmıştır.

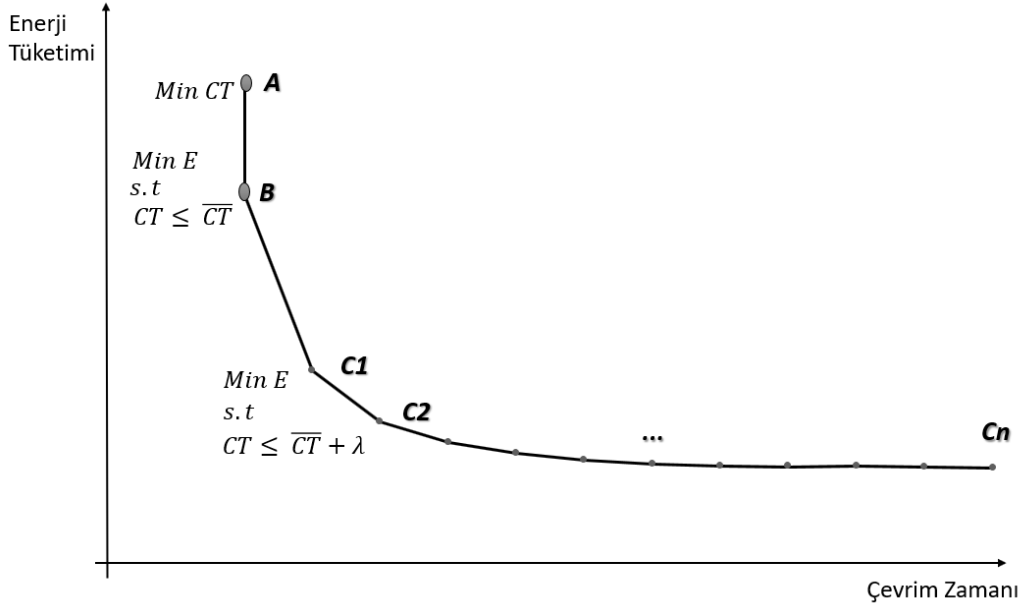
Adımları aşağıda sunulan ETA'nın genel çalışma mantığı şu şekildedir. ETA ilk olarak mümkün olan en kısa çevrim süresini bulmaya çalışır. Bunun için öncelikle robotun tüm hareketlerinde en yüksek hızda olduğunu varsayılır (Aşama 0). Böylece robotun makineler arası hareket süreleri minimum değerlerine eşitlenir. Elde edilen minimum

hareket süreleri dikkate alınarak minimum çevrim süresini veren aktivite sıralamasını bulmak bir sonraki aşamadır (Aşama 1). Bu aşamada Tavlama Benzetimi meta sezgiseli kullanılarak en düşük çevrim zamanı değerini veren aktivite sıralaması bulunmaya çalışılır. Minimum CT değerini sağlayan birden fazla aktivite sıralaması olan durumlarda ise bir çözüm havuzu oluşturularak elde edilen en iyi çevrimler burada tutulur.

En yüksek hız değerleri ile bulunan aktivite sıralamaları bir çevrim çizelgesi verir. Bu döngüler Şekil 4.1'deki A noktasına karşılık gelir. Bu döngülerde belli zamanlarında robotun boşta beklediği durumlar oluşabilir. Boşta bekleme sürelerinin bir kısmı robot yavaşlatılarak azaltılabilir, böylece çevrim süresi değişmeden enerji tüketimi azaltılabilir. Bu da ilk bulunan çözümün domine edilmesi anlamına gelir. Aşama 2'de hangi robot hareketlerinde çevrim süresi değişmeden robotun yavaşlatılabileceği belirlenir. Bu hareketler için robot hızları "esnek" hızlar olarak tanımlanmıştır. Azaltılmasıyla çevrim zamanını artıran hızlar ise "kritik" hızlar olarak ifade edilmiştir. Aşama 3'te seçilen aktiviteler için enerji tüketimini minimize eden hızlar belirlenerek ilk yaklaşık etkin çözüm bulunmuş olur. Bu çözüm Şekil 4.1'deki B noktasına karşılık gelir. Bazı durumlarda bu şekilde bir yavaşlatma ve enerji tasarrufu mümkün olmayabilir. Bu durumda Şekil 4.1'deki A ve B noktaları aynı noktaya karşılık gelmektedir.

En kısa çevrim süresini bulmaya çalışılan ilk yaklaşık etkin çözüm bulunduktan sonra ETA, çevrim süresini her iterasyonda arttırarak ve bu çevrim süresine karşılık gelen en düşük enerji tüketim değerine sahip robot hareket sıralaması ve hareket hızlarını belirleyerek yeni yaklaşık etkin çözümler bulur. Bu iteratif adımlar Aşama 4'ü oluşturmaktadır.

ETA'nın elde edeceği yaklaşık etkin çözüm kümesi örnek olarak Şekil 4.1'deki grafikteki gibi olacaktır. Aşama 1'de A noktası için çözüm havuzu oluşturulacak ve B noktası için alternatif çözümler belirlenecektir. Aşama 2 ve 3'te B noktası, Aşama 4'ün tekrarlanması ile de C_1, C_2, \dots, C_n noktaları türetilenektir.



Şekil 4.1: Etkin Çözüm Kümesi

ETA algoritmasının genel işleyişi aşağıdaki gibidir.

Aşama 0: Tüm hızları, hız üst limiti olan VUB değerine eşitle

Aşama 1: Tavlama Benzetimi meta sezgisel algoritmasını ve belirlenen komşulukları kullanarak en küçük çevrim zamanını veren aktivite sıralamaları havuzunu belirle (Algoritma 2)

Aşama 2: Robot ve makinelerin boşta beklediği zamanları göz önünde bulundurarak, yavaşlatılması çevrim zamanını değiştiren (kritik) ve değiştirmeyen (esnek) hızları belirle (Algoritma 3 ve 4)

Aşama 3: Esnek olarak belirlenen hızları tüm beklemleri olabildiğince azaltacak kadar yavaşlat, böylece elde edilen çevrim zamanı alt limitini sağlayan en küçük enerji tüketimini hesapla (Algoritma 5 ve 6)

Aşama 4: Çevrim zamanı üst limitini λ kadar artır.

Aşama 4.1: Aşama 2'de belirlenen kritik hızları, yol aldıkları zaman λ kadar artacak miktarda yavaşlat (Algoritma 7), yeni çevrimin enerji tüketimini hesapla

→ Eğer elde edilen kritik hız değeri, esnek hızların en büyük değerinden de daha büyükse, Aşama 4.2'ye geç.

→ Değilse, yavaşlatma işlemini tüm hızlar için uygula (Algoritma 8) ve enerji tüketimlerini hesapla, Aşama 4.2'ye geç.

Aşama 4.2: Tavlama Benzetimi algoritması ile, çevrim zamanı alt limitine eşit olan döngüleri ve enerji tüketimlerini bul (Algoritma 9)

Aşama 4.3: Aşama 4.1 ve 4.2'de elde edilen döngülerden en az enerji tüketimini gerçekleştiren çözümü, etkin çözüm olarak belirle

Aşama 5: İstenen etkin çözüm sayısına Aşama 4'ü tekrarla

ETA algoritmasının aşamalarına geçmeden önce, algoritmanın genelinde kullanılan terimlerden de bu bölümde bahsedilmiştir. Bu terimler bir çevrimin her bir pozisyonunu için tanımlanmış olup şu şekildedir.

$McIndex_p$ = p pozisyonundaki aktivite tipini ve makineyi gösteren indistir. Negatif ifadeler yükleme, pozitifler ise boşaltmayı ifade eder. Örneğin, -1 indisi 1. makineye yükleme yapıldığını, 2 ise 2. makineden parça boşaltıldığını gösterir.

$gIndex_p$ = (g_1, g_2) şeklinde bulunulan pozisyonundaki tutucu durumlarını gösterir. $g_1, g_2 = 0, 1, \dots, m + 1$ durumlarından herhangi bir tanesi olabilir.

$actGripper_p$ = p pozisyonundaki aktif tutucuyu gösteren indistir.

V_p = p ve $p + 1$ pozisyonları arasındaki robot hızını gösterir.

VLB_p, VUB_p = p ve $p + 1$ pozisyonları arasında olması gereken hız alt ve

üst sınırlarını gösterir. VLB_p ve VUB_p değerleri

$VLB \leq VLB_p \leq VUB_p \leq VUB$ aralığındadır.

$dist_p$ = p ve $p + 1$ pozisyonları arasındaki makine uzaklığını gösterir.

- δ_p = p ve $p + 1$ arasındaki robotun ulaşım süresini belirtir.
- T_p = p pozisyonundaki aktivitenin başlama zamanını ifade eder.
- $Wait_p$ = p pozisyonunda boşaltma aktivitesi varsa ve işlem süresi bitmemişse robotun beklemesi gereken süreyi gösterir.
- $Idle_p$ = p pozisyonunda boşaltma aktivitesi varsa ve işlem süresi robot makineye gelmeden bitmişse, parçanın robotu bekleyeceği süreyi gösterir.

$$Flexibility_p = \begin{cases} 1 & \text{Eğer } V_p \text{ hız değeri çevrim zamanını artırmadan bir miktar} \\ & \text{azaltılabiliyorsa yani esnek hız ise} \\ 0 & \text{Diğer durumda} \end{cases}$$

ETA algoritmasının genelinde kullanılan,

- belirlenen bir aktivite sıralamasına ait tutucu indislerini bulma,
- tutucu indislerine göre çevrimin olurlu olup olmadığını belirleme
- eğer çevrim olurluysa çevrim zamanını ve enerji tüketimini bulma

yöntemlerinin detayları aşağıda anlatılmıştır.

Aktivite sıralamasının belirlenmesi ($MIndex_p$)

Aktivite sıralamasının belirlenmesi için başlangıç çözümünde Sethi vd. [45] tarafından belirli koşullar altında optimal olduğu kanıtlanan aktivite sıralaması ele alınmıştır. m makine olan bir sistemde, başlangıç aktivite sıralaması şöyledir;

$$U_0 \Rightarrow U_1 \Rightarrow L_1 \Rightarrow U_2 \Rightarrow L_2 \Rightarrow U_3 \cdots \Rightarrow L_{(m-1)} \Rightarrow U_m \Rightarrow L_m \Rightarrow L_{(m+1)}$$

Bu aktivite sıralaması sezgisel algorithmada $MIndex_p$ değerleriyle şu şekilde temsil edilecektir;

$$0 \Rightarrow 1 \Rightarrow -1 \Rightarrow 2 \Rightarrow -2 \Rightarrow 3 \cdots \Rightarrow -(m-1) \Rightarrow m \Rightarrow -m \Rightarrow (m+1)$$

Bu çözümden farklı olarak daha sonra anlatılacak olan komşuluklar ile de farklı aktivite sıralamaları elde edilmektedir. Aktivite sıralaması belirlendikten sonra, bu sıralamaya ait tutucu indisleri hesaplanmalıdır. Böylece, olurlu olduğu tespit edilen sıralama

için, tutucu değişimleri de göz önünde bulundurularak çevrim zamanı hesaplanabilecektir.

Aktif tutucuyu belirleme ($actGripper_p$)

Daha önce de belirtildiği üzere, gerçekleştirilen ilk aktivite $U_0(1,*)$, yani giriş stokundan ilk tutucu ile parça boşaltılması olarak belirlenmiştir ($actGripper_0 = 1$). Döngüsel çizelgeme yapılacağından, 1. pozisyona bu aktiviteyi sabitlemekte sorun yoktur. Sonraki pozisyonların her biri için ise aşağıdaki şartlar kontrol edilir.

- Eğer $n \neq m + 1$ iken, p . sırada m makinesinde boşaltma aktivitesi ve $p + 1$. sıradaki n makinesinde yükleme aktivitesi gerçekleşiyorsa,
- p ve $p + 1$. pozisyonlarda yükleme aktivitesi gerçekleşiyorsa,
- p ve $p + 1$. pozisyonlarda boşaltma aktivitesi gerçekleşiyorsa,
- Eğer $n \neq m + 1$ iken, $p - 1$. sırada m makinesinde yükleme, p . sıradaki n makinesinde boşaltma aktivitesi ve $p + 1$. sırada yine bir boşaltma aktivitesi varsa,

p ve $p + 1$. pozisyonlarda tutucu değişimi gerçekleşiyor demektir. Belirtilen şartlardan biri sağlandığında, $p + 1$ pozisyonundaki aktif tutucunun p pozisyonundaki tutucudan farklı olduğu belirlenmiş olur. Eğer yukarıdaki hiçbir durum gerçekleşmiyorsa, $p + 1$ pozisyonundaki aktif tutucunun p 'deki ile aynıdır.

Tutucu indislerinin belirlenmesi

Pozisyonlardaki aktif tutucular ($actGripper_p$) belirlendikten sonraki adım tutucu indislerinin ($gIndex_p$) belirlenmesidir. Böylece aktivite sıralamasının olurlu olup olmadığına karar verilecektir. Tutucu indisleri, gerçekleşen aktivite, aktif tutucu ve aktivitelerin gerçekleştiği makinelere bağlı olarak belirlenir. Şöyle ki, eğer p pozisyonunda m makinesinde boşaltma aktivitesi gerçekleşiyor iken $m + 1$ makinesine yükleme aktivitesi r pozisyonunda gerçekleşiyor ise, aktiviteyi gerçekleştiren tutucunun indisi, p ve $r - 1$ pozisyonları arasında $m + 1$ olarak belirlenir. Eğer p pozisyonunda yükleme aktivitesi gerçekleşiyor ve aynı tutucu ile boşaltma aktivitesi r pozisyonunda gerçekleş-

şiyor ise de aktiviteyi gerçekleştiren tutucunun indisi, p ve $r - 1$ pozisyonları arasında 0 olarak belirlenir.

Olurluluğun kontrol edilmesi

Belirlenen aktivite sıralaması için tutucu indisleri de elde edildikten sonra, olurluluğun kontrol edilmesi için aşağıdaki şartlar göz önünde bulundurulur.

- Eğer p pozisyonunda m makinesine yükleme aktivitesi gerçekleşiyor ve o pozisyonda aktif olan tutucunun $p - 1$ pozisyonunda indisi m değilse,
- Eğer, $p - 1$ pozisyonunda m makinesinden 1. tutucu (veya 2. tutucu) ile boşaltma işlemi gerçekleşiyor ve $p - 1$ pozisyonunda 1. tutucunun (veya 2.'nin) indisi $m + 1$ değilse,
- Eğer p pozisyonunda yükleme aktivitesi gerçekleşiyor ve o pozisyonda aktif olan tutucunun $p - 1$ pozisyonunda indisi 0 ise,
- Eğer p pozisyonunda boşaltma aktivitesi gerçekleşiyor ve o pozisyonda aktif olan tutucunun $p - 1$ pozisyonunda indisi 0 değilse,

aktivite sıralamasının olurlu olması mümkün değildir. Olurlu bir aktivite sıralaması için, yukarıdaki şartlardan hiçbirinin gerçekleşmemesi gerekmektedir.

Çevrim zamanının hesaplanması

Aktivite sıralamasının olurlu olduğundan emin olduktan sonra, aktif tutucu değişimlerini de göz önünde bulundurarak, ardışık pozisyonlarda bulunan makineler arasında alınan yol ve geçen zaman hesaplanmaktadır. Şöyle ki, eğer p ve $p + 1$ pozisyonlarındaki aktiviteler sırasıyla m ve n makinelerinde aynı tutucular ile yapılıyorsa δ_p olarak ifade edilen süre $\frac{dist_{mn}}{v_p}$ olarak belirlenir. Aktif tutucuların değiştiği durumda ise, $\frac{dist_{mn}}{v_p}$ ve θ değerlerinden büyük olanı makineler arası geçen süreyi belirlemektedir. Diğer bir deyişle, $\delta_p = \max\{\frac{dist_{mn}}{v_p}, \theta\}$ değerine eşittir.

Makineler arası geçen zaman hesaplandıktan sonra, pozisyonların başlama zamanları ve nihayetinde de aktivite sıralaması ile elde edilen çevrim zamanı aşağıda verilen algoritma ile belirlenmektedir.

Algoritma 1 Çevrim Zamanını Belirle

Girdi: Çevrim: aktivite sıralaması, $CT: 0$, $CTkontrol: 1$, $T_0:0$

```
1: while  $CT$  değeri,  $CTkontrol$  değerinden farklı iken do
2:   for  $p=1, \dots, \mathcal{P}$  do
3:      $CTkontrol \leftarrow CT$ 
4:     if  $p$ . pozisyonda  $m$  makinesinde yükleme aktivitesi varsa then
5:        $T_p \leftarrow T_{(p-1)} + \delta_{(p-1)} + \varepsilon$ 
6:       else if  $m$  makinesinden boşaltma işlemi  $r$  pozisyonunda yapılıyorsa then
7:         if  $r < p$  then
8:            $T_p \leftarrow \max\{T_{(p-1)} + \delta_{(p-1)} + \varepsilon; T_r + \varepsilon + P_m\}$ 
9:            $Idle_p \leftarrow \max\{0, T_p - (T_r + P_m + \varepsilon)\}$ 
10:        else
11:           $T_p \leftarrow \max\{T_{(p-1)} + \delta_{(p-1)} + \varepsilon; -CT + T_r + \varepsilon + P_m\}$ 
12:           $Idle_p \leftarrow \max\{0, -CT + T_p - (T_r + P_m + \varepsilon)\}$ 
13:        end if
14:         $Wait_p = T_p - (T_{(p-1)} + \delta_p + \varepsilon)$ 
15:      end if
16:      if  $p = |\mathcal{P}|$  then
17:         $CT \leftarrow T_p + \delta_p + \varepsilon$ 
18:      end if
19:    end for
20:  end while
```

Algoritma, ardışık iterasyonlardaki çevrim zamanı değerleri olan CT ve \overline{CT} birbirine eşit olana dek 1. satırında bulunan **while** döngüsü ile tekrar edilir. 5. satırda bulunan komut ile p pozisyonundaki yükleme aktivitesinin bir önceki aktiviteden $\delta_p + \varepsilon$ kadar daha sonra başlamasını sağlar. Bir boşaltma aktivitesinin başlama zamanını belirlemek için ise söz konusu makineye yükleme yapılan pozisyon bilinmelidir. Eğer bir çevrimde m makinesinden boşaltma, yükleme işleminden daha sonra gerçekleşiyorsa U_m aktivitesinin başlama zamanı 8. satırdaki formülle hesaplanır. Burada max ifadesinin kullanılmasının nedeni robotun makineye ulaşmadan ve işlem süresi tamamlanmadan

parçayı boşaltamamasıdır. Hareket süresi δ_p ve makine işlem süresi P_m değerlerinden büyük olanı T_p değerini belirlemektedir.

Döngüsel çizelgeleme ele alındığı için, bir çevrimde boşaltma aktivitesi her zaman yükleme aktivitesinden daha sonra gerçekleşmeyebilir. Bu durumlar için T_p değerinin belirlenmesinde 11. sıradaki formül kullanılmaktadır. Görüleceği üzere bu formülün 8. satırdakinden tek farkı max ifadesinin 2. teriminden CT değerinin çıkartılmasıdır. Bunun sebebi T_0 değerinin her çevrim için 0 olmasını sağlamaktır.

Algoritmanın 17. adımında ise CT değeri hesaplanmıştır. Bu değer son pozisyonun başlama zamanı T_p ile $\delta_p + \varepsilon$ değerlerinin toplanmasıyla elde edilmiştir. δ_p değeri $p = |\mathcal{P}|$ için, son pozisyonda aktivite gerçekleştirilen i makinesinden ilk aktivite yapılacak olan giriş stokuna olan uzaklık $dist_{i0}$ ' in V_p hızına bölümüyle hesaplanır. Giriş stokunda parça işlemesi gerçekleşmediğinden bu formülde $P_0 = 0$ değerinin bulunması gereksizdir. Ayrıca bu algoritma ile aktivitelerin başlangıç zamanları ve makinelerin işlem sürelerine göre sırasıyla robot ve makinelerin boşta bekleme süreleri olan $Wait_p$ ve $Idle_p$ değerleri de hesaplanmaktadır.

Olurlu olduğu tespit edilen bir döngünün çevrim zamanı hesaplandıktan sonra söz konusu çevrimin enerji tüketim değeri bulunacaktır.

Eneji tüketiminin hesaplanması

Daha önce açıklanan enerji tüketim fonksiyonu Eşitlik 4.2 kullanılarak, çevrim başına enerji tüketim değeri hesaplanır. Bütün pozisyonlar için $dist_p$ ve V_p değerleri tüketim için değerlendirmeye alınır. Bunun yanı sıra ardışık pozisyonlar arasında yer değiştiren bir tutucunun olup olmadığına bakılır ve tutucunun yer değiştirmesinin ortaya koyduğu tüketim de genel toplama eklenir.

Özetlemek gerekirse, yukarıda açıklanan yöntemler ile, bir aktivite sıralamasının olurlu olup olmadığı belirlendikten sonra çevrim zamanı ve enerji tüketimi hesaplanmaktadır. Açıklanan bu yöntem ve algoritmalar, ETA'nın genel olarak tüm aşamalarında kullanılacaktır.

Bir sonraki bölümde başlangıç aktivite sıralaması verilen ve hız değerleri VUB'ye

eşitlenen bir çevrim üzerinden en küçük çevrim zamanını veren çözümün bulunması adımları anlatılacaktır.

4.2.1 ETA: Aşama 1

Şekil 4.1'te A noktasının bulunması için bir Tavlama Benzetimi algoritması geliştirilmiştir. Uygulanan Tavlama Benzetimi algoritmasının aşamaları Algoritma 2 ile sunulmuştur. En küçük çevrim zamanını veren çözümün elde edilmesi için 3 farklı komşuluk tanımı Tavlama Benzetimi algoritması ile birlikte kullanılmıştır. Bu komşuluklar şu şekildedir:

Komşuluk 1: Uygulanan ilk komşuluk araması iki aktivitenin yer değiştirmesi şeklinde tanımlanmıştır. Seçilen iki rasgele aktivitenin yer değiştirmesi sonucunda oluşan çevrim eğer olurlu ise bu çözüm aday çözüm olarak kabul edilir. Eğer olurlu değil ise yer değiştirmeden bir önceki çevrime dönülerek yeni olurlu çözüm aranır. Bu komşuluğun Tavlama Benzetimi algoritmasında kullanılma oranı $\frac{1}{|P|}$ 'dir.

Komşuluk 2: Kullanılan diğer bir komşuluk tanımı ise rastgele seçilen başlangıç ve bitiş pozisyonları arasındaki aktivitelerin tersten sıralanması şeklindedir. Bu komşuluk aramasında da elde edilen çevrimin olurlu olup olmamasına göre aday çözüm kabul edilir ya da yeni çözüm aranır. Yer değiştirme komşuluğunda olduğu gibi bu komşuluk da $\frac{1}{|P|}$ oranında kullanılmaktadır.

Komşuluk 3: Son olarak, olası en küçük çevrim zamanlı döngüyü elde edebilmek için robot beklemelelerini göz önünde bulunduran bir araya ekleme metodu komşuluk aramasında kullanılmıştır. Bu metoda göre ilk önce çevrimdeki en yüksek robot bekleme değeri $Wait_max$ ve gerçekleştiği pozisyon p bulunur. Aktivite sıralamasının çevrim zamanı olan CT değerine göre $\frac{Wait_max}{CT}$ oranı belirlenir. Bu orandan küçük olan en büyük tamsayı değeri p aktivitesinin sağa doğru yer değiştireceği pozisyon sayısını vermektedir.

Örneğin 5 makineli bir küme için $Wait_max$ değeri 3. pozisyonda gerçekleşsin ve $\frac{Wait_max}{CT}$ değeri 4.5 olsun. Toplam 12 pozisyon olan bu örnekte, 3. pozisyondaki aktivite 4 pozisyon ilerletilerek 7 ve 8. pozisyonlar arasına alınır. Eğer bu yeni sıralama olurlu

değilse 3. pozisyondaki aktivite 1 sıra daha ilerletilerek 8 ve 9 arasına alınır. Olurlu çözüm buluna dek ya da pozisyon sayısınca gerçekleştirilen bu araya ekleme metodu, tüm olası pozisyonlarda da olurlu çözüm elde edilemezse ilk çözümü çıktı olarak vermektedir. Çevrimsel çizelgeleme yapıldığından hareket adım sayısı pozisyon sayısını aştığında 1. pozisyondan devam edilmektedir. Şöyle ki 4 makineli bir örnek için 10 pozisyon olduğundan, 6. pozisyondaki aktivite eğer 7 adım taşınıyorsa 3 ve 4. pozisyonlar arasına getiriliyor demektir.

Tavlama Benzetimi algoritmasına göre, başlangıç çözümünden sonra komşuluklar ile belirlenen yeni çözüm eğer daha önce elde edilen en iyi çözümden daha iyi bir çevrim zamanı değerine sahipse daha önceki çözümler kümesi temizlenilerek bu çözüm kümeye eklenir. Eğer çevrim zamanı değeri daha önceki çözümlerle aynı fakat aktivite sıralaması farklı ise çözüm kümesine bu çözüm de eklenir. Aynı CT değerine ancak farklı aktivite sıralamalarına sahip çevrimlerin çözüm havuzunda tutulmasının nedeni ise Şekil 4.1'deki A noktasında aynı çevrim zamanına sahip çevrimlerden hangisinin B noktasında daha düşük enerjiye sahip olacağına bilinmemesidir.

Elde edilen çevrimin CT değeri, çözüm havuzundaki en iyi CT değeri olan $bestCT$ 'ye eşit ya da küçük olması durumlarında yeni çözüm için belirtilen çözüm üzerinden komşuluk araması yapılır. Eğer çözüm daha kötü bir çevrim zamanı değerine sahipse, tavlama benzetimi için tanımlanan sıcaklık ve rasgele olasılık değerlerine göre bu çözüm üzerinden yeni çözüm araması yapıp yapılmayacağı belirlenir. Bitiş sıcaklık değeri ve sayma parametrelerine göre arama tamamlandığında en küçük çevrim zamanı çözümü için çözüm havuzu oluşturulmuş olacaktır. Bu havuzda bulunan çözümler için ETA algoritmasının 2. ve 3. aşamasındaki yöntemler uygulanarak hangi aktivite sıralamasının en küçük enerji değerini vereceği bulunur.

Tamamen rastgele yer değiştirmelerden oluştuğu için Komşuluk 1 ve 2, $\frac{1}{|\mathcal{P}|}$ oranında kullanılırken, Komşuluk 3 ise $\frac{|\mathcal{P}|-2}{|\mathcal{P}|}$ oranında arama yöntemi olarak kullanılmaktadır. $|\mathcal{P}|$ ise bir veri kümesi için $2m + 2$ değerini almaktadır. Bu sayede elde edilen çözümlerin kalitesinin artırılması sağlanmaya çalışılmıştır.

Algoritma 2 Aşama 1- Tavlama Benzetimi Algoritması

```
1: while sıcaklık < termin sıcaklığı do
2:   sayaç ← 0
3:   for sayaç < sayaç sınırı do
4:     Var olan çözüm  $Cycle_0$ 'ın çevrim zamanı değeri  $\hat{CT}_0$  ve enerji tüketim değeri  $\hat{E}_0$ 'yi hesapla
5:     if (sayaç %  $\mathcal{P}=1$ ) then
6:       Komşuluk 1 ile yeni olurlu bir çevrim belirle
7:     else if (sayaç %  $\mathcal{P}=2$ ) then
8:       Komşuluk 2 ile yeni olurlu bir çevrim belirle
9:     else
10:      Komşuluk 3 ile yeni olurlu bir çevrim belirle
11:    end if
12:    Elde edilen çözüm  $Cycle_1$ 'in çevrim zamanı değeri  $\hat{CT}_1$  ve enerji tüketim değeri  $\hat{E}_1$ 'yi hesapla
13:    if  $\hat{CT}_1 < bestCT$  then
14:      Çözüm havuzunu temizle,  $Cycle_1$  çözümünü havuza ekle
15:    else if  $\hat{CT}_1 = bestCT$  ve  $Cycle_1$  havuzda bulunan aktivite sıralamalarından farklı then
16:      Çözüm havuzunu temizlemeden  $Cycle_1$  çözümünü havuza ekle
17:    else
18:       $h \leftarrow [0,1]$  aralığında rasgele bir sayı
19:       $p \leftarrow \frac{1}{1 + \exp \frac{\hat{CT}_0 - \hat{CT}_1}{sıcaklık}}$ 
20:      if  $p > h$  then
21:         $Cycle_0 \leftarrow Cycle_1$ 
22:      end if
23:    end if
24:  end for
25:  sıcaklık ← sıcaklık · soğuma oranı
26: end while
```

Aşama 2 ve 3 sonucu elde edilecek en küçük çevrim zamanı ve enerji tüketimine sahip döngü Şekil 4.1'deki B noktası çözümünü verecektir. Bu durumda A noktası için ETA'nın vereceği sonuç B noktasındaki çevrimin hareket hızları yavaşlatılmamış versiyonu olacaktır.

Şekil 4.1'deki B noktası çözümünün elde edilmesi için kullanılan yöntemler devam eden bölümde açıklanmıştır.

4.2.2 ETA: Aşama 2

Bu bölümde verilen bir çevrim için robot hareket hızlarının esnek ya da kritik olduğunu belirlemek için kullanılan yöntemlerden bahsedilecektir. Geliştirilen algoritmalar ayrıca sözde kodlar şeklinde de sunulmuştur. Ayrıca Bölüm 3.2'da verilen aktivite sıralaması ve parametreler kullanılarak sayısal örneklerle de açıklanmıştır.

Kritik hızların belirlenmesinden önce, tutucu değişimleri göz önünde bulundurularak, pozisyonlar arası hız üst sınır değerleri belirlenmelidir. Şöyle ki, iki pozisyon arasında tutucu değişimi gerçekleştiğinde δ_p değeri $\frac{dist_p}{V_p}$ ve θ değerlerinden büyük olanına eşit olmaktadır. θ değerinin büyük olması durumunda ise, enerji verimi sağlayan bir çevrimin $\theta \leq \frac{dist_p}{V_p}$ eşitsizliğini sağlaması gerekmektedir. Bu durumda V_p hız değerinin üst limiti

$$VUB_p \leq \frac{dist_p}{\theta}$$

eşitsizliğini sağlamalı ve V_p değeri üst limit değerinden büyük olamayacağından VUB_p değerine eşitlenmelidir. Bu işlem enerji tüketiminin azaltılmasına yardımcı olsa da çevrim için hala yavaşlatılması mümkün olan hızlar olabilir. Bu durumda yavaşlatılacak hızlar aşağıdaki algoritma ile belirlenir.

Algoritma 3'ün 5. satırında bulunan kontrol ile V_p hız değerinin çevrim zamanını artırıp artırmadığı kontrol edilir. Yeni \hat{CT} değerinin çevrimin orijinal CT değerinden daha büyük olması 4. satırda σ kadar azaltılan V_p hızının kritik hız olduğunu göstermektedir. Eğer çevrim zamanı değeri değişmediyse V_p hızı esnek hız olarak tanımlanır ve $Flexibility_p$ değeri 1'e eşitlenir. V_{p+1} hızının esnek olup olmadığını kontrol etmeden

önce ise, 14. satırda V_p hızı σ eklenerek orijinal değerine eşitlenir.

Bölüm 3.2'deki örnek veri kümesi ve verilen çevrim için Algoritma 3 uygulandığında, V_3, V_7, V_8 ve V_9 hızları kritik olarak tanımlanmıştır. $V_1, V_2, V_4, V_5, V_6, V_{10}, V_{11}$ ve V_{12} hızları ise esnek hız olarak belirlenmiştir. Diğer bir deyişle, $V_1, V_2, V_4, V_5, V_6, V_{10}, V_{11}$ ve V_{12} hızları çevrim zamanını değiştirmeden, enerji tüketimini azaltmak için yavaşlatılabilir. Ancak esnek hızların ne kadar yavaşlatılabileceği makine ve robot beklmelerine göre belirlenmektedir.

Algoritma 3 Kritik Hızların Belirlenmesi

Girdi: *Cycle*: başlangıç aktivite sıralaması, *CT*: *Cycle* aktivite sıralamasının çevrim zamanı

```
1: for  $p=1, \dots, \mathcal{P}$  do
2:   if  $V_p > 0$  then
3:      $V_p \leftarrow (V_p - \sigma)$ 
4:      $\hat{CT} \leftarrow$  Değişen hızlarla Cycle çevriminin çevrim zamanını Algoritma 1 ile
      hesapla
5:     if  $\hat{CT} > CT$  then
6:        $Flexibility_p \leftarrow 0$ 
7:     else
8:        $Flexibility_p \leftarrow 1$ 
9:     end if
10:  else
11:     $Flexibility_p \leftarrow 0$ 
12:  end if
13:   $V_p \leftarrow (V_p + \sigma)$ 
14: end for
```

Bir sonraki aşamada bekleme ve hareket sürelerine göre hızların azami değerlerini belirleyecek olan kısıtlar oluşturulacaktır. ETA için "kısıt" tanımı hareket hızlarının çevrim zamanını aşmadan bekleme değerlerini olabildiğince azaltmak için ne kadar yavaşlatılabileceğini gösteren eşitsizlikler olarak tanımlanabilir. Ancak daha önce kısıt belirleme algoritması olan Algoritma 5'de kullanılacak olan *WaitIdle* dizisi Algoritma

4 ile oluşturulacaktır.

$WaitIdle$ dizisi veri kümesinde bulunan makine sayısı boyutunda olup, her bir makinenin boşaltılma aktivitesi sırasında meydana gelen bekleme zamanlarına göre değer alır. Eğer p sırasında U_m aktivitesi sırasında $Wait_p$ robot beklemesi gerçekleşiyorsa $WaitIdle_m$ değeri $Wait_p$ değerine eşitlenmektedir. Eğer bu aktivite sırasında makine beklemesi gerçekleşiyor ise $WaitIdle_m = -Idle_p$ değerini alır.

Algoritma 4 $WaitIdle$ dizisinin oluşturulması

Girdi: $Cycle$: başlangıç aktivite sıralaması

```
1: for  $p=1, \dots, \mathcal{P}$  do
2:   if  $Wait_p > 0$  then
3:      $WaitIdle_{McIndex_p} \leftarrow Wait_p$ 
4:   else if  $Idle_p > 0$  then
5:      $WaitIdle_{McIndex_p} \leftarrow (-Idle_p)$ 
6:   end if
7: end for
```

Bir sonraki aşamada, $WaitIdle$ dizisi Algoritma 4 ile $[-15.33, -5.33, 2.67, 31.33, -4]$ şeklinde hesaplanmıştır. Bu dizide pozitif elamanlar bulunduğu için kısıtların oluşturulması için Algoritma 5’da girdi olarak kullanılabilir. Hiçbir pozitif elemanın bulunmaması durumunda ise hızların yavaşlatılarak aynı çevrim zamanına sahip ancak daha düşük enerji tüketimli yeni bir çözüm elde edilmesi mümkün değildir. Bu durumda B noktasındaki çözüm A noktasındaki çözüme eşit olacaktır.

Algoritma 5 ise $WaitIdle$ dizisini kullanarak hangi hızın hangi bekleme süresini azaltabileceğini ve buna bağlı olarak ne kadar yavaşlatılabileceğini gösteren kısıtları oluşturmak için kullanılan algoritmadır.

8. satırda bulunan *if* kontrol komutu ile V_p hızının σ kadar azaltılmasının söz konusu $|WaitIdle_i|$ değerini azaltıp azaltmadığı kontrol edilir. $|WaitIdle_{2_i}|$ ’nin değeri $|WaitIdle_i|$ ’den küçük olması V_p değerinin i . kısıtta aktif olarak bulunduğunu gösterir. Bu sebeple 9. satırdaki atama işlemi yapılır. 12. satırda ise kısıttaki aktif hızların etki ettiği bekleme değeri kısıtın son elemanı olarak atanır.

Algoritma 5 Kısıtların Belirlenmesi

Girdi: *Cycle*: başlangıç aktivite sıralaması, *CT*: *Cycle* aktivite sıralamasının çevrim zamanı, *WaitIdle* ve *WaitIdle2* Dizileri, Const Dizisi

```
1: if  $\exists \text{WaitIdle}_i > 0$  &  $\exists \text{Flexibility}_p = 1$  then
2:   for  $j = 1, \dots, |\text{WaitIdle}|$  do
3:     if  $\text{WaitIdle}_p \neq 0$  then
4:       for  $p = 1, \dots, \mathcal{P}$  do
5:          $V_p \leftarrow V_p - \sigma$ 
6:         Algoritma 1 ile yeni çevrim zamanını ve
7:         WaitIdle2 dizisinin elemanlarını belirle
8:         if ( $\text{WaitIdle}_j \neq 0$  ve  $|\text{WaitIdle}_j| > |\text{WaitIdle2}_j|$ ) then
9:            $\text{Const}_j \leftarrow \text{Const}_j \cup V_p$ 
10:        end if
11:       end for
12:        $\text{Const}_j \leftarrow |\text{WaitIdle}_j|$ 
13:        $V_p \leftarrow V_p + \sigma$ 
14:       Algoritma 1'i kullanarak orijinal çevrim zamanını ve WaitIdle dizisini
        yeniden hesapla
15:     end if
16:   end for
17: end if
```

Bu algoritmanın çıktıları olan bu kısıt dizileri $|\mathcal{P}| + 1$ boyutundadır ve en fazla makine sayısı m kadar kısıt dizisi vardır. Dizinin ilk $|\mathcal{P}|$ elemanı ikili değerler almakta olup V_p hızının o kısıtta yer alıp almadığını göstermektedir. Dizinin $(|\mathcal{P}| + 1)$. elemanı ise kısıtta aktif olan hızların etki ettiği bekleme süresi değerine eşittir. Örneğin 3 makinelik bir veri kümesi için elde edilecek bir kısıt şu formatta olacaktır:

$\text{Const}_j: [0, 0, 1, 0, 0, 0, 1, 1, |\text{WaitIdle}_j|]$

3, 7 ve 8. robot hareket hızlarının değerinin 1 olduğu yani bu kısıtta aktif olduğu gö-

rılmaktadır. Bu hızlar yavaşlatılmadan önce geçerli olan eşitsizlikler şu şekildedir:

$$\frac{dist_3}{v_3} \leq \delta_3; \quad \frac{dist_7}{v_7} \leq \delta_7; \quad \frac{dist_8}{v_8} \leq \delta_8$$

Ancak V_3, V_7, V_8 hareket hızlarının yavaşlatılması ve bu 3 eşitliğin birlikte $|WaitIdle_i|$ kadar genişletilmesi için 3 eşitsizlik toplanmış ve yeni eşitsizliğin sağ tarafına $|WaitIdle_i|$ değeri eklenmiştir.

Sonuç olarak, bu kısıt dizisinin matematiksel olarak gösterimi şöyle yapılmaktadır:

$$\frac{dist_3}{v_3} + \frac{dist_7}{v_7} + \frac{dist_8}{v_8} \leq \delta_3 + \delta_7 + \delta_8 + |WaitIdle_i|$$

Yukarıda Algoritma 3 ile Bölüm 3.2'deki örnek veri kümesi için kritik ve esnek hızlar hesaplanmıştır. Yine aynı örnek küme için *WaitIdle* dizisinin de kullanılmasıyla Algoritma 5 ile elde edilen kısıt dizileri şöyledir:

- $Const_1: [1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 15.33]$
- $Const_2: [0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 5.33]$
- $Const_3: [1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 2.67]$
- $Const_4: [0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 31.33]$
- $Const_5: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 4]$

Dizi şeklinde ifade edilen bu kısıtların matematiksel olarak gösterimi ise aşağıda verilmiştir.

Kısıt 4.41 ile ifade edilmek istenen V_1, V_{11} ve V_{12} 'nin 15.33'lük bekleme zamanını ortadan kaldırmak için yavaşlatılabileceğidir. $Const_1$ 'e bakıldığında bu dizinin 1, 11 ve 12. elemanlarının 1 değerini aldığı görülmektedir. Ancak, Kısıt 4.42'ye göre V_{11}, V_{10} ile birlikte 5.33'lük birimlik başka bir beklemede de etki sahibidir. Benzer şekilde, Kısıt

4.43 ve 4.44 ile ifade edilmek istenen de sırasıyla şudur: V_1, V_2, V_{10}, V_{11} ve V_{12} hızları 2.67 miktarındaki beklemeyi, V_4, V_5 ve V_6 hızları da 31.33 miktarındaki beklemeyi azaltmak için yavaşlatılabilir.

$$\frac{dist_1}{v_1} + \frac{dist_{11}}{v_{11}} + \frac{dist_{12}}{v_{12}} \leq \delta_1 + \delta_{11} + \delta_{12} + 15.33 \quad (4.41)$$

$$\frac{dist_{10}}{v_{10}} + \frac{dist_{11}}{v_{11}} \leq \delta_{10} + \delta_{11} + 5.33 \quad (4.42)$$

$$\frac{dist_1}{v_1} + \frac{dist_2}{v_2} + \frac{dist_{10}}{v_{10}} + \frac{dist_{11}}{v_{11}} + \frac{dist_{12}}{v_{12}} \leq \delta_1 + \delta_1 + \delta_{10} + \delta_{11} + \delta_{12} + 2.67 \quad (4.43)$$

$$\frac{dist_4}{v_4} + \frac{dist_5}{v_5} + \frac{dist_6}{v_6} \leq \delta_4 + \delta_5 + \delta_6 + 31.33 \quad (4.44)$$

$Const_5$ incelendiğinde ise, söz konusu dizinin ilk 12 elementinin 0 olduğu görülmektedir. Yani herhangi bir hareketin yavaşlatılması 4 birimlik beklemeyi ortadan kaldırmamaktadır. Bu sebeple $Const_5$ için matematiksel gösterim yapılmamıştır.

4.2.3 ETA: Aşama 3

Aşama 2'de kısıtları temsil etmek için oluşturulan diziler Algoritma 6'in girdilerini oluşturmaktadır.

Algoritmanın uygulanan ilk adımında bağlayıcı olarak işaretlenen kısıtlardaki hız değerlerinin $Flexibility_p$ değerleri 101 olarak işaretlenmiştir. Buradaki 101 değerinin sayısal olarak bir anlamı yoktur. Yalnızca, sabitlenmemiş hız değerleri hesaplanırken bu hızların sabitlendiğini göstermek ve sabitlenen yeni hız değerlerinin kullanılmasında kolaylık sağlaması için belirteç olarak kullanılmıştır.

Eğer bağlayıcı olarak işaretlenmemiş kısıtlar var ise bu kısıtlar için V_{temp} değerinin hesaplanması için 10. sıradaki for döngüsü kullanılır. Satır 16'daki kontrol ile p . pozisyondaki hız değerinin j kısıtına dahil olup olmadığı, bu hızın daha önceki iterasyonlarda alt ya da üst sınıra sabitlenip sabitlenmediği kontrol edilir. Daha sonra, bu kontrolü geçen hız değerlerinin önceki iterasyonlarda bağlayıcı olarak işaretlenen kısıtlara

sıtlarda bulunup bulunmadığı, yani sabitlenip sabitlenmediği 17. satırda kontrol edilir. Eğer bu hız değeri henüz sabitlenmemişse $tempDist$ satır 18'deki gibi hesaplanır. Eğer bu hız daha önce sabitlenmiş ise, kısıtta bulunan diğer sabitlenmemiş hızların gerektiğinden daha fazla yavaşlatılmasını engellemek için $Dist_p/V_p$ miktarı sağ taraf değeri $tempRHS$ 'den çıkarılır (satır 20). $tempDist$ ve $tempRHS$ değerlerinin hesaplaması tamamlandıktan sonra satır 25'deki formül kullanılarak V_{temp} değerleri hesaplanmaktadır.

Bağlayıcı olmayan tüm hız değerleri için V_{temp} değerleri hesaplandığında, elde edilen en büyük hız değeri ve bu hız değerinin hangi kısıt için hesaplandığı tespit edilir.

Örnek üzerinden devam edilecek olursa, **Aşama 2**'de Bölüm 3.2'te verilen veri kümesi için kısıtlar belirlenmişti. Matematiksel olarak da gösterilen 4.41 ile 4.44 arasındaki kısıtlar için hız değerleri Algoritma 6'in 25. satırındaki formül ile sırasıyla 0.91, 1.45, 2.5 ve 0.53 şeklinde hesaplanmıştır. 26. satırda bulunan $maxVLB$ değeri hesaplanan en yüksek hız değeri olan 2.5'e eşitlenmiştir. Daha sonra bu hızın ait olduğu kısıt 30. satırda tespit edilecektir.

Bu sayede $maxVLB$ değerine sahip olan kısıttaki hızlar sabitlenmiş olacaktır. Verilen sayısal örnek üzerinden devam edilirse, 2.5 olan V_{temp} değeri 4.43 numaralı kısıta ait olduğundan V_1, V_2, V_{10}, V_{11} ve V_{12} hareket hızları 2.5'e eşitlenmiştir. Hızları sabitlenen kısıtlar 31. satırda bağlayıcı olarak işaretlenmekte, böylece sonraki iterasyonlarda hareketlerin yavaşlatılmasının çevrim zamanını artırmasının önüne geçilmektedir. Ayrıca Kısıt 4.43 için hız değerleri 2.5'e sabitlendiğinde eşitsizliğin iki tarafının da 16 değerini alarak bağlayıcı olduğu görülmektedir.

Hız ataması yapılırken sabitlenecek hızın daha önce sabitlenip sabitlenmediği 33. satırda kontrol edilir. Eğer sabitleme işlemi yapılmadıysa $maxVLB$ değerinin VUB_p ve VLB_p arasında olup olmadığı kontrol edilir. Eğer $maxVLB$ hız üst limitinden büyükse V_p değeri VUB_p 'ye eşitlenir, eğer alt limitten küçük ise de V_p değeri VLB_p limitine eşitlenir. $Flexibility_p$ değerinin bu durumlarda 100'e eşitlenmesinin sebebi bu hızların sonraki iterasyonlarda yeniden hız hesaplarına dahil edilmemesini belirtmektir. 100 değerinin sayısal bir karşılığı yoktur, algoritmanın oluşturulmasında kolaylık sağlamak için bu değer ele alınmıştır.

Algoritma 6 Esnek Hızları Yavaşlat

Girdi: *Cycle*: başlangıç aktivite sıralaması, *Const* dizisi, *Vtemp* dizisi

```
1: while Tüm kısıtlar bağlayıcı olarak işaretlenmediği sürece do
2:   for  $j = 1..|Const|$  do
3:     if  $Const_j$  bağlayıcı olarak işaretlenmişse then
4:       if Eğer  $V_p \in Const_j$  then
5:          $Flexibility_p \leftarrow 101$ 
6:       end if
7:     end if
8:   end for
9:    $maxVLB \leftarrow 0$ 
10:  for  $j = 1..|Const|$  do
11:    if  $Const_j$  bağlayıcı olarak işaretlenmemişse then
12:       $tempDist \leftarrow 0$ 
13:       $tempDelta \leftarrow 0$ 
14:       $tempRHS \leftarrow Const_j$ 'nin sağ taraf değeri
15:      for  $p = 1, \dots, \mathcal{P}$  do
16:        if  $V_p \in Const_j, VUB_p > 0$  ve  $Flexibility_p \neq 100$  then
17:          if  $Flexibility_p \neq 101$  then
18:             $tempDist \leftarrow tempDist + Dist_p$ 
19:          else
20:             $tempRHS \leftarrow tempRHS - (Dist_p/V_p)$ 
21:          end if
22:           $tempDelta \leftarrow tempDelta + Delta_p$ 
23:        end if
24:      end for
25:       $Vtemp_j \leftarrow tempDist / (tempRHS + tempDelta)$ 
26:       $maxVLB \leftarrow Max\{maxVLB, Vtemp_j\}$ 
27:    end if
28:  end for
29:  for  $j = 1..|Const|$  do
30:    if  $Vtemp_j = maxVLB$  then
```

```

31:      Constj'yi bağlayıcı olarak işaretle
32:      for  $p = 1, \dots, \mathcal{P}$  do
33:          if  $V_p \in \text{Const}_j$  ve  $\text{Flexibility}_p \neq 101$  then
34:              if  $\text{Flexibility}_p \neq 100$  then
35:                   $VLB_p \leftarrow \text{Max}\{VLB_p, \text{maxVLB}\}$ 
36:                  if  $McIndex_p = McIndex_{p+1}$  then
37:                       $VUB_p \leftarrow 0$ 
38:                  else if  $\text{actGripper}_p \neq \text{actGripper}_{p+1}$  then
39:                       $VUB_p \leftarrow \text{Min}\{VUB_p, \text{Dist}_p/\theta\}$ 
40:                  end if
41:                  if  $VLB_p > VUB_p$  then
42:                       $\text{Flexibility}_p \leftarrow 100$ 
43:                      Constj'yi bağlayıcı değil olarak işaretle
44:                  end if
45:                   $V_p \leftarrow \text{Min}\{VUB_p, VLB_p\}$ 
46:              end if
47:          end if
48:      end for
49:      else if Constj'yi bağlayıcı değil olarak işaretlenmişse then
50:           $V_{temp_j} \leftarrow 0$ 
51:      end if
52:  end for
53: end while

```

Kısıt 4.41 ve 4.42'de bulunan tüm hız değerleri Kısıt 4.43'ün bağlayıcı olmasıyla hesaplanmış durumdadır. Bu sebeple bu kısıtlar için yeniden V_{temp} hesaplaması yapılmayacaktır. Söz gelimi, Kısıt 4.41'de V_4 değeri de bulunsaydı, bu kısıtın hesaplanmasında aşağıdaki denklem kullanılacaktı:

$$\frac{dist_1}{2.5} + \frac{dist_{11}}{2.5} + \frac{dist_{12}}{2.5} + \frac{dist_4}{v_4} \leq \delta_4 + \delta_1 + \delta_{11} + \delta_{12} + 15.33 - \frac{dist_1}{3} - \frac{dist_{11}}{3} - \frac{dist_{12}}{3}$$

Böylece, V_1, V_{11} ve V_{12} hızlarının yavaşlamasıyla azalmış olan 15.33'lük bekleme değerinin güncellenmesi sağlanmış olup, V_4 'ün yavaşlatılmasının çevrim zamanı artışına sebep olmaması sağlanmıştır. Kısıt 4.44 için ise V_{temp} değeri 0.53 olarak hesaplanmıştır. Bu değer verilen VLB değerinden düşük olduğundan V_4, V_5 ve V_6 hızlarının değeri VLB değeri yani 1'e sabitlenmiştir. Esnek olan tüm hızların değerleri sabitlendiğinden Algoritma 6 için daha fazla iterasyon gerekmemektedir.

Özetlemek gerekirse, Aşama 1'de elde edilen çözüm kümesi için Aşama 2 ve 3'te

enerji tüketimini en aza indirecek hız değerleri hesaplanmıştır. Bu çözümlerden en küçük enerji tüketimine sahip olan çevrim Şekil 4.1'deki B noktasına karşılık gelmektedir. B noktasındaki çözüm için elde edilen aktivite sıralamasının hareketlerinin yavaşlatılmamış hali de Şekil 4.1'deki A noktası çözümünü vermektedir. Aşama 4'te ise çevrim zamanı limiti artırılarak elde edilen daha küçük enerji tüketimine sahip pareto optimal noktalar bulunacaktır.

4.2.4 ETA: Aşama 4

Aşama 1, 2 ve 3 sonucu Şekil 4.1'deki A ve B noktaları elde edilmiştir. Bu aşamada ise Şekil 4.1'deki diğer yaklaşık pareto etkin çözümler elde edilecektir. ETA sezgiseli ile karar vericinin istediği sayıda yaklaşık pareto etkin çözüm türetilebilmektedir.

Şekil 4.1'deki B noktasından C_1 noktasına geçerken iki çözüm alternatifi bulunmaktadır. Bunlardan ilki, aktivite sıralamasını değiştirmeden kritik hızları belirli bir miktar yavaşlatarak çevrim zamanını artırmaktır. İkinci alternatif ise Aşama 1'de geliştirilen arama algoritmasını kullanarak farklı bir aktivite sıralaması bulmaktır.

Kritik hızların yavaşlatılması işlemi, esnek hızların yavaşlatılmasına benzer bir şekilde yapılmaktadır. Bu algoritmanın sözde kodu Algoritma 7 ile gösterilmiştir.

Algoritma 7 uygulandıktan sonra, hesaplanan kritik hız değerleriyle, çevrimde bulunan esnek hız değerleri karşılaştırılır. Eğer kritik hız değerlerinden daha büyük değerde bir esnek hız var ise bu daha küçük enerji tüketimi için o hızın da yavaşlatılması gerektiği anlamına gelmektedir. Kritik ve esnek hızlar arasındaki bu kontrol gerçekleştirildikten sonra, eğer esnek hızların da yavaşlatılması gerektiği kararına varıldığında Algoritma 8 uygulanmıştır.

Görüldüğü üzere Algoritma 7 ile 8 arasındaki fark 6. satırlarda gerçekleşen kontrol işlemidir. Algoritma 7'de sadece kritik hızların yavaşlatılma işlemine dahil edilmesi sağlanmışken, Algoritma 8'de alt ya da üst sınır limitinde bulunmayan tüm hız değerleri yavaşlatılmıştır.

Algoritma 7 Kritik Hızların Yavaşlatılması

Girdi: *Cycle*: başlangıç aktivite sıralaması

```
1: while tempV  $\neq$  tempV2 do
2:   tempV2  $\leftarrow$  tempV
3:   tempDist  $\leftarrow$  0
4:   tempDelta  $\leftarrow$  0
5:   for  $p = 1, \dots, \mathcal{P}$  do
6:     if  $Flexibility_p = 1$  then
7:       tempDist  $\leftarrow$  tempDist +  $Dist_p$ 
8:       tempDelta  $\leftarrow$  tempDelta +  $Delta_p$ 
9:     end if
10:  end for
11:  tempV  $\leftarrow$  tempDist / (tempDelta +  $\lambda$ )
12:  for  $p = 1, \dots, \mathcal{P}$  do
13:    if  $Flexibility_p = 1$  then
14:      if tempV >  $VUB_p$  then
15:         $V_p \leftarrow VUB_p$ 
16:      else if tempV <  $VLB_p$  then
17:         $V_p \leftarrow VLB_p$ 
18:      else
19:         $V_p \leftarrow tempV$ 
20:      end if
21:    end if
22:  end for
23:  tempV2  $\leftarrow$  tempV
24: end while
```

Algoritma 8 Kritik Hızların Yavaşlatılması-Versiyon 2

Girdi: *Cycle*: başlangıç aktivite sıralaması

```
1: while tempV  $\neq$  tempV2 do
2:   tempV2  $\leftarrow$  tempV
3:   tempDist  $\leftarrow$  0
4:   tempDelta  $\leftarrow$  0
5:   for  $p = 1, \dots, \mathcal{P}$  do
6:     if Flexibility $p$   $\neq$  100 then
7:       tempDist  $\leftarrow$  tempDist + Dist $p$ 
8:       tempDelta  $\leftarrow$  tempDelta + Delta $p$ 
9:     end if
10:  end for
11:  tempV  $\leftarrow$  tempDist / (tempDelta +  $\lambda$ )
12:  for  $p = 1, \dots, \mathcal{P}$  do
13:    if Flexibility $p$   $\neq$  100 then
14:      if tempV > VUB $p$  then
15:         $V_p \leftarrow VUB_p$ 
16:        Flexibility $p$   $\leftarrow$  100
17:      else if tempV < VLB $p$  then
18:         $V_p \leftarrow VLB_p$ 
19:        Flexibility $p$   $\leftarrow$  100
20:      else
21:         $V_p \leftarrow$  tempV
22:      end if
23:    end if
24:  end for
25:  tempV2  $\leftarrow$  tempV
26: end while
```

Diğer alternatif çözüm için ise, Aşama 4.2’de aynı çevrim zamanıyla daha küçük enerji tüketimini sağlayan olası bir aktivite sıralaması aranmaktadır. Tavlama Benzetimi Al-

goritması ve Aşama 1’de anlatılan komşuluklar kullanılarak, verilen çevrim zamanı limitinin altındaki çevrimler araştırılır. C_1 noktası için \overline{CT} üst limiti ile aranan çözümün çevrim zamanı değeri CT ise bu çözümün kabul edilip edilmeyeceğinin karar aşamaları şu şekildedir:

Algoritma 9 Alternatif Çözümün Belirlenmesi

- 1: **if** $CT < \overline{CT}$ **then**
 - 2: $\lambda \leftarrow \overline{CT} - CT$
 - 3: Sırasıyla, Algoritma 3,4,5 ve 6’i kullanarak, mevcut çözümün esnek hızlarını çevrim zamanını değiştirmeden yavaşlat
 - 4: λ artış miktarını kullanarak Algoritma 7 ve gerekiyorsa Algoritma 8’i kullanarak hızları yavaşlat
 - 5: Hız değerleri yavaşlamış olan bu çevrimin enerji tüketimi en düşük değerde ise bu çözümü en iyi çözüm olarak belirle
 - 6: **end if**
-

Tavlama Benzetimi Algoritması sonunda elde edilen en düşük enerji tüketimine sahip olan çevrim Şekil 4.1’deki C_1 noktası için ikinci alternatif çözümdür.

Aşama 4.1 ve 4.2 ile elde edilen alternatif çözümlerden en küçük enerji tüketimini sağlayan çözüm C_1 noktası için pareto etkin çözüm olarak kabul edilir.

4.2.5 ETA: Aşama 5

Bu aşamada **Aşama 4**’deki işlemlerin istenen yaklaşık pareto etkin çözüm sayısınınca tekrarlanması sağlanır. Diğer bir deyişle Aşama 5’te, Şekil 4.1’deki B ve C_1 noktaları arasında **Aşama 4**’te uygulan işlemler, C_1 ve C_2 noktaları arasında uygulanır. **Aşama 4**’ün bir sonraki tekrarında ise C_2 noktasından C_3 noktasına geçiş sağlanmıştır. Genellemek gerekirse, Aşama 5’te istenen pareto etkin çözümü sayısınınca, çevrim zamanı sınırı λ kadar artırılarak **Aşama 4** tekrar edilir.

5. HESAPLAMALI SONUÇLAR

Geliştirilen çözüm yöntemlerinin etkinliklerini ölçebilmek için oluşturulan örnek veri kümeleriyle performans testleri yapılmıştır. Bu bölümde, MINLP ve MISCOP matematiksel modelleri ile ETA sezgisel algoritmasının çözüm süresi ve optimale yakınlık açısından karşılaştırmaları verilmiştir. Ayrıca veri kümeleri için elde edilen aktivite sıralamaları incelenmiş ve çözümlerde sıklıkla rastlanan çevrimler yorumlanmıştır. Son olarak, Şekil 4.1'deki noktalar arasındaki geçişlerde sağlanan enerji tasarrufu değerlendirmeleri de bu bölümde bulunmaktadır.

5.1 Veri Kümeleri

Geliştirilen modellerin hesaplama performanslarını test etmek için veri kümeleri oluşturulmuştur. Deneylerde kullanılan veri kümelerinin boyutu makine sayısı bazında 2, 3, 4 ve 6 olarak belirlenmiştir. Ayrıca, makineler arası mesafeler oluşturulurken, literatürde sıkça kullanılan "özdeş toplamsal" uzaklık alternatifi kullanılmıştır. Diğer bir deyişle, ardışık makinelerin arasındaki mesafenin özdeş olduğu ve ardışık olmayan makineler arasındaki mesafenin, ardışık makineler arası mesafenin toplamına eşit olduğu varsayımı kullanılarak uzaklık parametreleri oluşturulmuştur. Doğrusal makine dizilimi göz önünde bulundurulduğundan M_i ile M_j makineleri arasındaki mesafe $dist_{ij} = |i - j|dist_0$ şeklinde hesaplanmış olur. Burada $dist_0$ ardışık makineler arası mesafedir. Makineler arası işlem süreleri varyansının problem karmaşıklığına etkisi olabileceği ve çözüm yöntemlerinin performansını etkileyebileceği öngörüldüğünden rastgele türetme işleme yapılırken dar ve geniş aralıklı Düzgün (Uniform) dağılım kullanılmıştır. Bu sayede işlem süreleri varyansı düşük ve yüksek problem durumları oluşturulmuştur.

Diğer taraftan makine işlem süreleri ve makineler arası mesafeler yakından ilişkilidir. Eğer işlem süreleri robotun hareket sürelerine göre çok düşük olursa, yani, mesafeler

baskın olursa makineler işlemlerini hızlıca bitirip robotun boşaltma için gelmesini bekleyeceklerdir. Bu durumda, alternatif döngülerden sadece birkaç tanesi her zaman optimali verecektir. Robotun hiçbir zaman bekleme süresi olmayacak, dolayısıyla robotu yavaşlatmanın bir faydası da olmayacaktır. Benzer bir durum, işlem sürelerinin baskın olması durumunda da geçerlidir. Bu durumda da robot sürekli en yavaş hızında hareket edecek ve yine optimal döngü her zaman aynı olacaktır. Hem problem karmaşıklığının yüksek olduğu durumları test edebilmek hem de anlamlı çıkarımlar yapabilmek için işlem sürelerinin makineler arası mesafe ve robotun hareket hızı alt ve üst limitine kıyasla uygun değerlerde türetilmesi gerekmektedir.

Ayrıca, aktif robot tutucusu değişim süresi olarak verilen θ parametresinin makineler arası hareket süresine kıyasla küçük olması bu parametrenin hiçbir zaman etkili olmamasına sebep olurken, çok yüksek olması da tutucu değiştirmenin her zaman daha baskın olması anlamına gelmektedir. Bu sebeple, bu parametre de mesafe ve hız alt ve üst limitlerine bağlı olarak küçük ve büyük iki farklı değerde türetilmiştir.

Sonuç olarak işlem süresi ve theta için ikişer seviye ve 2,3,4 ve 6 makineli problemler düşünülerek toplam 16 veri kümesi oluşturulmuş, her bir veri grubu için rastgelelik unsuru göz önünde bulundurularak onar veri kümesi toplamda 160 veri kümesi oluşturulmuştur. Testlerde kullanılan parametrelerin dağılımı Çizelge 5.1’de gösterilmiştir. Bu tabloda “DD” kısaltması düzgün dağılımı, "TB" kısaltması ise bir veri kümesi için makineler arası toplam uzaklığın hız üst sınırına oranını ifade eden temel birimi göstermektedir. Bu birim robotun üst limit hızıyla tüm makineleri sırasıyla ziyaret edip başlangıç noktasına geri dönmesi için gereken süreyi vermektedir.

Tüm veri setleri için pareto noktalar arasındaki artış miktarı olan λ değeri bir veri kümesi için (En büyük CT değeri- En küçük CT değeri)/ ψ şeklinde hesaplanmıştır. m makineli bir veri kümesi için en büyük çevrim zamanı değeri aşağıdaki döngü kullanılarak elde edilmektedir.

$$U_0(1,0) \longrightarrow L_1(0,0) \longrightarrow U_1(2,0) \longrightarrow L_2(0,0) \longrightarrow U_2(3,0) \longrightarrow \dots \longrightarrow L_n(0,0) \\ \longrightarrow U_n(n+1,0) \longrightarrow L_{(n+1)}(0,0) \longrightarrow \dots \longrightarrow L_{(m-1)} \longrightarrow U_m \longrightarrow L_m \longrightarrow \dots \longrightarrow L_{m+1}$$

Çizelge 5.1: Parametre Değerleri

Parametre	Dağılım	
	Özdeş Toplamsal (ÖT)	
Mesafe	DD[5,50]	
	Dar Aralık (DA)	Geniş Aralık (GA)
İşlem Süresi	DD[$TB \cdot \frac{3}{4}$, $TB \cdot \frac{5}{4}$]	DD[$TB \cdot \frac{1}{2}$, $TB \cdot \frac{3}{2}$]
	Küçük	Büyük
θ	$TB \cdot \frac{1}{4}$	$TB \cdot \frac{1}{2}$
ε	1	
Hız Limiti	Alt sınır (VLB) =1 Üst sınır (VUB)=3	
C	1	
G	2	

Bu çevrimin olabilecek en az hareketi yapması ve hiç tutucu değişimi gerçekleştirilmesi sebebiyle, tüm hızlar alt sınır değeri olan VLB'ye eşitlendiğinde en büyük çevrim zamanını ve en düşük enerji tüketim değerini vereceği görülmektedir.

Bu döngüde giriş stokunda parça alan robot bir n makinesine önce yükleme işlemi yapıp, işlem süresinin tamamlanmasını bekleyerek boşaltma işlemi gerçekleştirmekte daha sonra $n + 1$ makinesine geçmektedir. En yüksek çevrim zamanının elde edilmesi hedeflendiğinden robotun hız alt limitinde hareket ettiği varsayılmıştır. Bu durumda ardışık makineler arası uzaklığın eşit ve $dist_0$ olduğu bir veri kümesinin en büyük çevrim zamanı değeri aşağıdaki formülle hesaplanmaktadır.

$$\sum_{i=1}^m P_i + 2(m+1)\varepsilon + 2m\left(\frac{dist_0}{VLB}\right) \quad (5.1)$$

ψ değeri en düşük ve en yüksek çevrim zamanı değerleri arasında eşit aralıklarla bu-

lunmak istenen pareto optimal çözüm sayısını vermektedir. Bu çalışmada ψ değeri 10 olarak belirlenmiş, yani Şekil 4.1'deki B ile C_9 noktası dahil 10 pareto optimal nokta işaretlenmiştir. 2 ve 3 makine için en küçük çevrim zamanı noktası olan A noktası ve en büyük çevrim zamanı noktası olan C_{10} noktası da dahil olmak üzere 12 nokta için koşturum yapılmıştır. Ancak C_5 ile C_9 arasındaki noktaların çözüm değerlerinin birbirini tekrar ettiği tespit edilip, bu noktalara sonuç tablolarında yer verilmemiştir. Ayrıca, 2 ve 3 makinelik problemlerde C_{10} noktası için MISOCP ve ETA ile elde edilen çözümlerin birbirine eşit olduğu görülmüştür. Bu sebeple C_{10} noktasına da sonuç tablolarında yer verilmemiştir.

4 ve 6 makine için ise çözüm süreleri daha uzun olduğu için Şekil 4.1'deki B ile C_9 arasındaki noktalardan yalnızca B ile C_4 noktaları için koşturum yapılmıştır. Bu veri kümelerinin sonuç tablolarında bulunan C_5 noktaları ise en büyük çevrim zamanı noktasını belirtmektedir.

5.2 MINLP, MISOCP ve ETA Koşturum Sonuçları

Öncelikle, geliştirilen MINLP ve MISOCP modellerinin performansları birbirleriyle karşılaştırılmıştır. Daha sonra MISOCP modelin çok daha hızlı ve verimli çalıştığı göz önünde bulundurularak, geliştirilen ETA sezgiselinin sonuçları MISOCP ile karşılaştırılmıştır. $m = 2$ durumunda bu modellerin performans karşılaştırması Çizelge 5.2 ve 5.3'te verilmiştir. Çizelge 5.2 işlem sürelerinin dar aralık, Çizelge 5.3 ise geniş aralık ile belirlendiği 2 makineli veri kümeleri için alınan sonuçları göstermektedir.

Tabloda görüleceği üzere tüm veri kümeleri için, Şekil 4.1'deki A, B, C_1, C_2, C_3, C_4 ve C_5 noktaları için alınan çözüm değerleri ve süreleri karşılaştırılmıştır. Hatırlanacağı üzere A noktası, tüm hızlar üst limit değerine eşit olduğunda elde edilecek en küçük çevrim zamanlı çözüm noktasıdır, yani bu noktanın çözümü Karma Tamsayı Programlama (MIP) ile bulunabilmektedir. Tüm veri kümeleri için B, C_1, C_2, C_3, C_4 ve C_5 noktalarından önce A noktasının çözümü GAMS ara yüzü ve CPLEX çözücüsü elde edilmekte, MISOCP ve MINLP modeller ile sonuçlar daha sonra alınmaktadır. Bu nedenle MISOCP ve MINLP modellerinin kıyaslaması yapılırken A noktası göz önünde bulundurulmamıştır. ETA ile MISOCP yöntemlerinin kıyaslamasında ise tüm noktalar

açısından çıkarımlar yapılmıştır.

MISOCP modeli GAMS ara yüzü ve CPLEX çözücüsü ile çözdürülürken MINLP modeli yine GAMS ara yüzü ve BARON çözücüsü ile çözülmüştür. Modeller için 1 saat (3600 saniye) zaman limiti kullanılmıştır. Tabloda çözücünün zaman limiti ile durduğu durumlar ZL ile gösterilmiştir. Her iki çözücü de kullandıkları algoritmalarla verilen problem için optimal çözümü garanti etmektedir. Ancak, sonuçlar incelendiğinde, konik modelin en kötü durumda 14.82 saniyede optimal çözümü bulduğu görülmektedir. Tabloya göre, MINLP ile alınan 240 sonuçtan 97'sinde problem için 1 saat içerisinde herhangi bir olurlu çözüm bulamamıştır. Bu nedenle bu durumlar için sapma miktarı da hesaplanamamıştır. Çizelge 5.2 ve 5.3'de bu durumlar ÇY ile gösterilmiştir. Konik model sonuçları ile karşılaştırıldığında, 2 makineli veri kümelerinde için bile MINLP çözümü uzun zaman aldığından 3, 4 ve 6 makineli veri kümeleri için MINLP koşturumu yapılmamıştır.

ETA ile alınan sonuçlar optimal sonuçlar ile karşılaştırıldığında ise geliştirilen bu algoritmanın 2 makineli örnekler için oldukça iyi sonuçlar verdiği görülmektedir. Şöyle ki, tüm veri kümeleri için sezgisel algoritma ile hesaplanan Şekil 4.1'deki A noktası çözümleri ve en büyük çevrim zamanı çözümleri MISOCP model ile aynı olarak bulunmuştur. Bu sayede, iki yöntem ile hesaplanan pareto optimal noktalardaki çevrim zamanı üst sınırları birbirine eşit bulunmuş olup, bulunan sonuçların enerji verimliliği kıyaslanabilmiştir. Bulunan tüm çevrim zamanı limitleri birbirine eşit olduğundan bu değerler tablolarda gösterilmemiştir.

Çözümlerin Enerji tüketim değerlerinin optimal değerlerle kıyaslaması yapıldığında ise, Çizelge 5.2'ye bakıldığında en yüksek hata oranının 6. veri kümesinin C_1 noktasında ortaya çıkan %0.07 olduğu görülmüştür. Diğer hata oranlarının $[-0.1, 0.1]$ arasında değiştiği görülmektedir. Ancak bu farklılıklar, ETA'nın performansından değil, kullanılan ara yüzlerin hesaplamalar esnasında yaptığı rasyonel sayıları yuvarlama farklılıklarından kaynaklanmaktadır. Çünkü ETA ve MISOCP modellerin sonuçları hareket hızları ve aktivite sıralamaları açısından karşılaştırıldığında sonuç neredeyse aynı olduğu, farklılıkların binde birler basamağından sonra ortaya çıktığı belirlenmiştir. Bu sebeple %0.1 oranında enerji tüketimi farklılıkları ortaya çıktığı gibi, ETA ile elde

edilen enerji değerinin -%0.1 oranında daha düşük olduğu durumlar da oluşmuştur.

Çizelge 5.2: 2 makine, İşlem Süresi DA, MISOCP, MINLP ve ETA karşılaştırması

Veri Kümesi	Etkin Çözüm No.	Küçük θ						Büyük θ				
		MISOCP		MINLP		ETA		MISOCP		MINLP		ETA
		Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)	Çözüm Süresi (sn)	Enerji Sapma (%)	Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)	Çözüm Süresi (sn)	Enerji Sapma (%)	
1	A	0.5			1.3	0.0	0.5			0.6	0.0	
	B	4.7	ZL	ÇY	0.0	0.0	4.5	ZL	ÇY	0.0	0.0	
	C ₁	3.6	ZL	ÇY	0.5	0.0	5.3	ZL	ÇY	0.2	0.0	
	C ₂	6.6	ZL	104.3	0.6	0.0	6.1	ZL	92.8	0.3	0.0	
	C ₃	7.8	ZL	35.5	1.0	-0.1	5.6	ZL	6.1	0.6	0.0	
	C ₄	7.3	ZL	0.1	0.7	0.0	5.8	ZL	0.1	0.5	0.0	
C ₅	6.8	ZL	24.1	0.7	0.0	6.5	ZL	ÇY	0.8	0.0		
2	A	0.5			0.8	0.0	0.5			0.5	0.0	
	B	5.1	ZL	ÇY	0.0	0.0	5.2	ZL	ÇY	0.0	0.0	
	C ₁	4.8	ZL	45.3	0.4	0.0	4.6	ZL	ÇY	0.3	0.0	
	C ₂	14.3	ZL	26.3	0.6	0.0	8.9	ZL	ÇY	0.4	0.0	
	C ₃	10.5	ZL	70.9	0.6	0.0	5.9	ZL	123.2	0.5	-0.1	
	C ₄	9.3	ZL	121.1	0.9	-0.1	8.8	ZL	176.4	0.6	0.0	
C ₅	8.2	ZL	ÇY	0.8	0.0	6.0	ZL	85.3	0.7	0.0		
3	A	0.5			0.6	0.0	0.5			0.5	0.0	
	B	4.3	ZL	52.5	0.0	0.0	3.6	ZL	ÇY	0.0	0.0	
	C ₁	5.8	ZL	158.6	0.4	0.0	5.8	ZL	0.5	0.3	0.0	
	C ₂	7.0	ZL	4.1	0.7	0.0	6.3	ZL	ÇY	0.9	0.0	
	C ₃	6.8	ZL	0.0	0.6	0.0	2.8	ZL	35.9	0.4	-0.1	
	C ₄	8.5	ZL	0.1	1.0	-0.1	7.8	ZL	75.0	0.6	0.0	
C ₅	4.8	ZL	6.5	1.8	0.0	7.7	ZL	75.0	0.7	0.0		
4	A	0.5			0.5	0.0	0.6			0.5	0.0	
	B	6.0	ZL	ÇY	0.0	0.0	5.0	ZL	ÇY	0.0	0.0	
	C ₁	5.7	ZL	63.5	0.3	0.1	4.0	ZL	ÇY	0.2	0.0	
	C ₂	5.2	ZL	8.2	0.5	0.0	5.1	ZL	93.8	0.3	0.0	
	C ₃	3.2	ZL	48.8	0.6	0.1	6.7	ZL	121.0	0.3	0.0	
	C ₄	8.5	ZL	ÇY	1.0	-0.1	4.6	ZL	198.4	0.6	0.0	
C ₅	7.1	ZL	10.5	1.0	0.0	6.6	ZL	80.0	0.5	0.0		
5	A	0.4			0.5	0.0	0.6			0.6	0.0	
	B	3.9	ZL	ÇY	0.0	0.0	5.3	ZL	ÇY	0.0	0.0	
	C ₁	5.2	ZL	ÇY	0.6	0.0	6.7	ZL	ÇY	0.9	0.0	
	C ₂	5.9	ZL	ÇY	1.2	0.0	7.6	ZL	26.7	1.4	0.0	
	C ₃	4.8	ZL	ÇY	2.2	0.0	3.6	ZL	126.0	0.4	0.0	
	C ₄	4.0	ZL	ÇY	2.8	0.0	4.5	ZL	0.0	0.7	0.0	
C ₅	6.0	ZL	3.8	4.3	0.0	7.0	ZL	0.0	0.8	0.0		
6	A	0.6			0.9	0.0	0.8			0.7	0.0	
	B	2.6	ZL	ÇY	0.0	0.0	4.9	ZL	ÇY	0.0	0.0	
	C ₁	3.8	ZL	158.9	1.8	0.7	8.6	ZL	37.7	0.5	0.0	
	C ₂	2.7	ZL	27.3	0.7	0.0	3.8	ZL	113.1	0.4	-0.1	
	C ₃	9.3	ZL	72.8	0.8	0.0	6.3	ZL	ÇY	0.5	0.0	
	C ₄	5.0	ZL	18.2	5.4	0.0	6.3	ZL	6.6	0.9	0.0	
C ₅	7.1	ZL	18.2	4.1	0.0	7.6	ZL	6.6	0.8	0.0		
7	A	0.4			0.6	0.0	0.6			0.5	0.0	
	B	2.6	ZL	ÇY	0.0	0.0	5.7	ZL	0.0	0.0	0.0	
	C ₁	7.0	ZL	ÇY	2.0	0.0	4.9	ZL	110.1	0.3	0.0	
	C ₂	7.1	ZL	200.9	0.7	0.0	7.4	ZL	241.8	0.3	0.0	
	C ₃	3.3	ZL	208.3	0.8	0.0	5.0	ZL	234.6	0.4	0.0	
	C ₄	8.3	ZL	23.4	2.9	-0.1	7.7	ZL	0.0	0.7	0.0	
C ₅	5.1	ZL	37.5	1.4	0.0	5.9	ZL	0.0	0.7	0.0		
8	A	0.4			0.5	0.0	0.6			0.6	0.0	
	B	3.6	ZL	ÇY	0.0	0.0	2.3	ZL	ÇY	0.0	0.0	
	C ₁	7.4	ZL	83.6	0.5	0.0	4.7	ZL	207.2	0.3	0.0	
	C ₂	7.7	ZL	161.8	1.3	0.0	7.5	ZL	380.5	0.3	0.0	
	C ₃	7.8	ZL	253.5	0.8	0.0	7.3	ZL	110.5	0.7	0.0	
	C ₄	9.1	ZL	358.3	1.1	0.0	5.8	ZL	3.9	0.7	0.0	
C ₅	7.4	ZL	0.9	1.1	0.0	2.8	ZL	5.4	0.7	0.0		
9	A	0.6			0.5	0.0	0.6			0.6	0.0	
	B	4.4	ZL	ÇY	0.0	0.0	4.1	ZL	ÇY	0.0	0.0	
	C ₁	7.8	ZL	ÇY	1.1	0.0	4.2	ZL	117.0	0.4	0.0	
	C ₂	7.0	ZL	18.6	0.4	-0.1	5.0	ZL	230.5	0.4	0.0	
	C ₃	5.9	ZL	62.1	0.8	0.0	5.3	ZL	345.7	0.9	0.0	
	C ₄	5.8	ZL	111.7	0.9	0.0	5.8	ZL	404.7	0.6	0.0	
C ₅	5.5	ZL	2.1	2.2	0.0	6.5	ZL	78.4	0.7	0.0		
10	A	0.4			0.9	0.0	0.5			0.5	0.0	
	B	4.8	ZL	ÇY	0.0	0.0	2.5	ZL	ÇY	0.0	0.0	
	C ₁	6.2	ZL	147.9	0.5	0.0	3.2	ZL	ÇY	0.3	0.0	
	C ₂	7.7	ZL	73.5	0.5	0.0	3.1	ZL	130.8	0.4	0.0	
	C ₃	7.7	ZL	ÇY	0.7	-0.1	4.2	ZL	230.6	0.4	0.0	
	C ₄	7.8	ZL	31.2	1.0	-0.1	6.6	ZL	329.8	0.7	0.0	
C ₅	8.1	ZL	50.2	0.9	0.0	7.4	ZL	76.1	0.6	0.0		

Çizelge 5.3: 2 makine, İşlem Süresi GA, MISOCP, MINLP ve ETA karşılaştırması

Veri Kümesi	Etkin Çözüm No.	Küçük θ					Büyük θ					
		MISOCP Çözüm Süresi (sn)	MINLP Çözüm Süresi (sn)	Enerji Sapma (%)	ETA Çözüm Süresi (sn)	Enerji Sapma (%)	MISOCP Çözüm Süresi (sn)	MINLP Çözüm Süresi (sn)	Enerji Sapma (%)	ETA Çözüm Süresi (sn)	Enerji Sapma (%)	
11	A	0.6			0.4	0.0	0.7			0.5	0.0	
	B	4.9	ZL	ÇY	0.0	0.0	4.9	ZL	ÇY	0.0	0.0	
	C ₁	7.3	ZL	ÇY	0.4	0.0	4.8	ZL	ÇY	0.2	0.0	
	C ₂	7.1	ZL	ÇY	0.7	0.0	5.5	ZL	94.7	0.3	0.0	
	C ₃	5.6	ZL	ÇY	0.7	-0.1	3.9	ZL	113.4	0.5	0.0	
	C ₄	5.3	ZL		80.6	1.7	0.0	8.0	ZL	176.1	0.5	0.0
C ₅	5.3	ZL		ÇY	0.7	0.0	2.3	ZL	87.8	0.5	0.0	
12	A	0.5			26.2	0.0	0.6			0.5	0.0	
	B	6.4	ZL	57.3	0.0	0.0	4.8	893.2	ÇY	0.0	0.0	
	C ₁	7.2	ZL	104.1	3.6	0.0	6.3	ZL	ÇY	0.4	0.1	
	C ₂	4.4	ZL	2.1	4.2	0.0	7.2	ZL	20.7	0.4	0.1	
	C ₃	4.0	ZL	34.8	11.5	0.1	6.8	ZL	ÇY	0.6	0.0	
	C ₄	5.6	ZL	70.3	4.6	0.1	6.1	ZL	16.6	0.6	0.0	
C ₅	6.6	ZL	0.0	13.1	0.0	3.7	ZL	0.2	0.6	0.0		
13	A	0.5			6.3	0.0	0.8			2.7	0.0	
	B	4.9		764.01	ÇY	0.0	0.0	2.3	ZL	ÇY	0.0	0.0
	C ₁	3.1	ZL	ÇY	26.9	0.1	7.0	ZL	ÇY	1.7	0.0	
	C ₂	7.6	ZL	ÇY	5.4	0.0	8.1	ZL	36.0	3.8	0.0	
	C ₃	7.3	ZL	ÇY	5.3	0.0	4.6	ZL	80.4	2.8	0.0	
	C ₄	2.7	ZL	ÇY	16.8	0.0	11.6	ZL	118.9	7.6	0.0	
C ₅	3.3	ZL	ÇY	35.8	16.3	0.0	7.2	ZL	118.9	9.8	0.0	
14	A	0.5			0.9	0.0	0.7			0.5	0.0	
	B	4.5	ZL	ÇY	0.0	0.0	3.7	ZL	ÇY	0.0	0.0	
	C ₁	7.2	ZL	ÇY	0.4	0.0	6.7	ZL	ÇY	0.3	0.0	
	C ₂	9.6	ZL	ÇY	2.8	0.1	8.4	ZL	5.7	1.3	0.0	
	C ₃	6.6	ZL	ÇY	1.5	0.0	5.5	ZL	34.7	0.7	0.0	
	C ₄	6.9	ZL	ÇY	1.1	0.0	5.2	ZL	16.3	0.6	0.0	
C ₅	4.3	ZL	ÇY	2.5	0.0	4.0	ZL	16.3	0.8	0.0		
15	A	0.5			2.4	0.0	0.6			0.8	0.0	
	B	4.6	ZL	ÇY	0.0	0.0	2.7	413.7	ÇY	0.0	0.0	
	C ₁	3.3	ZL	ÇY	1.0	0.0	6.9	ZL	138.8	1.4	0.0	
	C ₂	7.7	ZL	ÇY	1.1	0.0	7.7	ZL	233.0	1.0	0.0	
	C ₃	6.7	ZL	ÇY	1.5	0.1	6.6	ZL	108.4	3.2	0.0	
	C ₄	6.2	ZL	ÇY	4.8	0.0	5.2	ZL	116.1	2.5	0.0	
C ₅	4.0	ZL	ÇY	3.1	0.0	8.7	ZL	2.8	2.5	0.0		
16	A	0.6			2.2	0.0	0.5			0.5	0.0	
	B	6.5	ZL	ÇY	0.0	0.0	5.4	377.4	ÇY	0.0	0.0	
	C ₁	5.6	ZL	179.9	0.9	0.1	6.3	ZL	77.1	0.3	0.1	
	C ₂	5.7	ZL	1.1	0.8	0.0	7.6	ZL	ÇY	0.4	0.1	
	C ₃	6.5	ZL	ÇY	0.8	0.0	8.3	ZL	117.5	0.4	0.1	
	C ₄	6.0	ZL	15.2	1.1	0.1	6.4	ZL	0.4	0.7	0.0	
C ₅	7.2	ZL	199.0	1.3	0.0	6.1	ZL	0.4	1.4	0.0		
17	A	0.6			3.7	0.0	0.8			0.6	0.0	
	B	5.8		659.15	ÇY	0.0	0.0	2.8	789.8	ÇY	0.0	0.0
	C ₁	4.0	ZL	137.4	1.3	0.0	7.4	ZL	0.1	1.1	0.0	
	C ₂	8.3	ZL	34.2	2.7	0.0	7.9	ZL	52.6	0.4	0.1	
	C ₃	8.1	ZL	78.8	11.8	0.0	6.1	ZL	106.4	0.6	0.0	
	C ₄	4.2	ZL	128.5	20.4	0.0	5.3	ZL	157.8	2.4	0.0	
C ₅	7.2	ZL	160.6	9.3	0.0	8.1	ZL	152.7	0.8	0.0		
18	A	0.4			7.8	0.0	0.6			3.9	0.0	
	B	3.0	ZL	ÇY	0.0	0.0	3.3	ZL	ÇY	0.0	0.0	
	C ₁	5.4	ZL	ÇY	28.8	0.0	4.8	ZL	ÇY	17.4	0.0	
	C ₂	6.6	ZL	ÇY	17.8	0.0	9.2	ZL	ÇY	3.9	0.0	
	C ₃	5.4	ZL	ÇY	21.5	0.0	6.1	ZL	84.4	2.7	0.0	
	C ₄	7.2	ZL	99.9	14.4	0.1	9.6	ZL	51.7	10.2	0.0	
C ₅	8.2	ZL	127.8	22.0	0.0	3.4	ZL	12.0	8.7	0.0		
19	A	0.6			0.8	0.0	0.5			0.8	0.0	
	B	3.1	ZL	ÇY	0.0	0.0	5.0	ZL	ÇY	0.0	0.0	
	C ₁	6.8	ZL	ÇY	0.5	0.0	4.0	ZL	2.6	0.3	0.0	
	C ₂	8.2	ZL	ÇY	0.6	0.0	4.5	ZL	44.2	0.4	0.0	
	C ₃	5.7	ZL	ÇY	1.0	0.1	6.6	ZL	86.0	0.7	-0.1	
	C ₄	4.4	ZL	ÇY	1.0	-0.1	7.4	ZL	104.1	0.7	0.0	
C ₅	7.5	ZL	ÇY	1.1	0.0	2.1	ZL	104.1	0.7	0.0		
20	A	0.4			4.4	0.0	0.6			13.3	0.0	
	B	6.0	ZL	ÇY	0.0	0.0	2.6	ZL	ÇY	0.0	0.0	
	C ₁	5.3	ZL	ÇY	1.1	0.0	8.0	ZL	ÇY	19.2	0.0	
	C ₂	5.2	ZL	ÇY	2.9	0.0	5.6	ZL	ÇY	19.6	-0.1	
	C ₃	7.8	ZL	ÇY	4.1	0.0	5.3	ZL	17.0	3.4	0.0	
	C ₄	5.5	ZL	ÇY	2.8	0.0	6.5	ZL	32.9	8.7	0.0	
C ₅	4.6	ZL	27.8	6.2	0.0	4.5	ZL	ÇY	8.3	0.0		

ETA ile elde edilen sonuçların mutlak ortalama hata oranı ise yaklaşık %0.01 olarak hesaplanmıştır. Çözüm süresi açısından değerlendirildiğinde ise 2-makineli örneklerde ETA ve MISOCP arasında çok büyük bir fark olmamakla birlikte bu süreler sırasıyla 2.3 ve 5.1 saniyedir.

Çizelge 5.4, 5.5 ve 5.6’da 3 makineli veri setleri için yapılan MISOCP ve ETA için karşılaştırmalar görülmektedir. Ortalama çözüm süresi bakımından karşılaştırıldığında sezgisel modelin performansının matematiksel modele göre oldukça iyi olduğu görülmektedir. Nitekim bu süreler sırasıyla 2.86 ve 24.12 şeklindedir. Küçük ve büyük θ değerleri için ortalama çözüm süresi değerleri Çizelge 5.6’da görülmektedir. Hata oranı açısından karşılaştırıldığında da sezgisel modelin 1 etkin çözüm noktası hariç matematiksel model ile bulunan optimal sonuçlara oldukça yaklaştığı söylenebilir. Bu noktadaki hata oranı %3.4 iken diğer tüm noktalar için maksimum hata oranı %0.01’dir. ETA için alınan çözümlerdeki hız değerleri incelendiğinde bu değerlerin matematiksel model sonuçlarındaki hız değerleriyle yaklaşık olarak aynı olduğu görülmüştür. Dolayısıyla %0.01 ve -%0.01’lik hata oranı değerleri rasyonel sayı yuvarlamaları sonucunda ortaya çıkan farklılıklar sebebiyle oluşmuştur. Yani ETA modelinin 3 makineli veri setleri için %99 oranında pareto etkin çözüm noktalarında optimal değeri verdiği çıkarımı yapılabilmektedir.

Çizelge 5.4: 3 makine, ETA ve MISOCP karşılaştırması

Veri Kümesi	P	Etkin Çözüm No.	Küçük θ			Büyük θ		
			MISOCP	ETA		MISOCP	ETA	
			Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)	Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)
1	DA	A	3.3	1.6	0.0	4.2	0.8	0.0
		B	11.5	0.0	0.0	10.8	0.0	0.0
		C ₁	14.5	1.0	0.0	15.6	0.4	0.0
		C ₂	41.7	2.0	0.0	24.9	1.0	0.0
		C ₃	26.2	2.5	0.0	23.3	1.2	0.0
		C ₄	33.2	2.4	0.0	43.8	1.5	0.0
		C ₅	22.5	2.4	0.0	32.7	1.8	0.0
2	DA	A	5.5	0.9	0.0	4.3	0.9	0.0
		B	15.1	0.0	0.0	14.7	0.0	0.0
		C ₁	21.2	1.4	0.0	25.3	0.4	0.0
		C ₂	26.6	1.3	0.0	34.2	1.3	0.0
		C ₃	25.2	1.6	-0.1	40.3	1.2	0.0
		C ₄	20.2	1.5	-0.2	31.6	1.1	0.0
		C ₅	22.9	1.4	0.0	24.6	1.7	0.0
3	DA	A	4.4	1.1	0.0	4.8	1.1	0.0
		B	11.3	0.0	0.0	9.7	0.0	0.0
		C ₁	28.4	0.9	0.0	18.5	0.4	0.0
		C ₂	33.8	1.6	0.0	45.2	1.2	0.0
		C ₃	25.6	2.5	0.0	32.5	1.3	0.0
		C ₄	25.6	2.5	0.1	58.3	2.0	0.0
		C ₅	29.9	2.4	0.0	18.9	2.3	0.0
4	DA	A	3.7	1.9	0.0	5.0	1.5	0.0
		B	19.8	0.0	0.0	20.9	0.0	0.0
		C ₁	27.4	1.9	0.0	14.4	0.4	0.0
		C ₂	35.1	1.2	0.0	30.4	0.9	0.0
		C ₃	30.9	2.6	0.1	34.2	1.3	0.0
		C ₄	46.0	3.4	0.0	27.4	2.4	-0.1
		C ₅	31.1	4.6	0.0	21.8	2.8	0.0

Çizelge 5.5: 3 makine, ETA ve MISOCP karşılaştırması (Devamı)

Veri Kümesi	P	Etkin Çözüm No.	Küçük θ			Büyük θ		
			MISOCP	ETA		MISOCP	ETA	
			Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)	Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)
5	DA	A	3.2	0.9	0.0	3.5	0.9	0.0
		B	24.0	0.0	0.0	17.4	0.0	0.0
		C ₁	47.4	1.0	0.0	18.1	0.4	0.0
		C ₂	19.3	1.6	0.0	47.0	1.2	0.0
		C ₃	33.2	2.5	0.0	20.4	1.4	0.0
		C ₄	22.2	1.7	0.0	19.1	1.4	0.0
		C ₅	27.4	1.8	0.0	33.7	1.9	0.1
6	DA	A	3.0	2.9	0.0	3.7	3.1	0.0
		B	7.0	0.0	0.0	6.5	0.0	0.0
		C ₁	47.1	3.7	-0.1	24.8	3.0	0.0
		C ₂	43.6	3.3	0.0	20.0	1.5	0.0
		C ₃	42.7	4.7	0.0	34.4	1.5	0.0
		C ₄	30.1	4.0	0.0	30.7	1.8	-0.1
		C ₅	35.2	6.4	0.0	27.3	10.6	0.0
7	DA	A	3.4	3.5	0.0	4.6	2.4	0.0
		B	13.1	0.0	0.0	9.1	0.0	0.0
		C ₁	19.9	11.7	0.0	12.0	1.1	0.0
		C ₂	51.8	4.3	0.0	11.6	1.1	0.0
		C ₃	32.6	5.0	0.0	18.4	1.2	0.0
		C ₄	50.2	6.9	0.0	34.5	1.5	0.0
		C ₅	20.9	6.7	0.0	25.1	3.0	0.0
8	DA	A	3.4	1.3	0.0	3.8	0.8	0.0
		B	9.2	0.0	0.0	10.6	0.0	0.0
		C ₁	23.0	1.5	0.0	13.5	1.0	0.0
		C ₂	29.3	1.3	0.0	21.2	0.9	0.0
		C ₃	42.0	2.7	0.0	36.1	1.2	0.0
		C ₄	30.2	1.9	0.0	45.7	1.2	0.0
		C ₅	36.7	2.6	0.0	24.4	1.8	0.0
9	DA	A	2.3	2.8	0.0	4.0	1.5	0.0
		B	29.2	0.0	0.0	9.1	0.0	3.4
		C ₁	45.0	12.6	0.0	19.5	1.2	0.0
		C ₂	63.9	2.1	0.0	25.5	1.0	-0.1
		C ₃	66.0	3.1	-0.1	25.1	1.3	0.0
		C ₄	51.1	6.3	0.0	22.9	3.2	0.0
		C ₅	29.5	7.0	0.0	24.5	3.8	0.0
10	DA	A	3.2	0.9	0.0	3.6	1.0	0.0
		B	17.7	0.0	0.0	36.8	0.0	0.0
		C ₁	20.4	3.3	0.0	45.6	2.7	0.0
		C ₂	34.7	2.3	-0.1	39.3	1.4	0.0
		C ₃	25.6	3.1	0.0	49.7	1.1	0.0
		C ₄	24.6	1.9	0.0	49.1	1.6	0.0
		C ₅	23.7	1.9	0.0	23.3	2.1	0.0
11	GA	A	3.6	9.3	0.0	4.6	3.4	0.0
		B	16.7	0.0	0.0	13.5	0.0	0.0
		C ₁	26.0	25.9	0.0	22.7	1.4	-0.1
		C ₂	28.5	25.4	0.0	22.3	2.7	0.0
		C ₃	40.1	9.3	0.0	27.9	1.5	0.0
		C ₄	40.2	8.4	0.1	29.8	1.7	-0.1
		C ₅	29.3	6.5	0.0	23.8	1.7	0.0
12	GA	A	3.5	1.9	0.0	3.4	0.9	0.0
		B	11.7	0.0	0.0	9.8	0.0	0.0
		C ₁	22.9	1.0	0.0	18.3	1.0	0.0
		C ₂	40.0	1.0	0.0	29.6	0.6	0.0
		C ₃	26.4	6.9	0.0	29.4	1.0	0.0
		C ₄	23.4	2.3	0.0	66.0	1.4	-0.1
		C ₅	29.7	2.2	0.0	28.4	1.5	0.0

Çizelge 5.6: 3 makine, ETA ve MISOCP karşılaştırması (Devamı)

Veri Kümesi	P	Etkin Çözüm No.	Küçük θ			Büyük θ		
			MISOCP	ETA		MISOCP	ETA	
			Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)	Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)
13	GA	A	3.2	7.3	0.0	5.3	3.5	0.0
		B	25.6	0.0	0.0	12.1	0.0	0.0
		C ₁	25.5	30.5	0.0	27.0	1.4	0.0
		C ₂	24.5	16.2	0.0	31.6	1.8	0.0
		C ₃	30.2	16.8	0.1	30.0	3.9	0.0
		C ₄	22.9	7.0	0.0	26.0	4.7	0.0
		C ₅	34.5	47.6	0.0	17.5	4.1	0.0
14	GA	A	4.6	0.9	0.0	4.6	0.8	0.0
		B	19.8	0.0	0.0	11.3	0.0	0.0
		C ₁	26.1	1.0	0.0	11.2	0.7	-0.1
		C ₂	30.8	1.4	0.0	20.5	0.9	0.0
		C ₃	21.6	1.8	-0.1	28.5	0.9	0.0
		C ₄	30.4	1.6	-0.1	25.2	1.9	0.0
		C ₅	32.3	2.2	0.0	26.8	1.9	0.0
15	GA	A	3.8	0.9	0.0	4.7	0.9	0.0
		B	11.2	0.0	0.0	17.4	0.0	0.0
		C ₁	22.5	0.9	0.0	44.3	0.9	0.0
		C ₂	40.5	2.2	0.0	25.1	1.3	0.0
		C ₃	26.9	1.7	-0.1	25.1	1.2	0.0
		C ₄	28.4	1.8	0.0	18.7	1.3	0.1
		C ₅	16.9	3.1	0.0	21.5	1.5	0.0
16	GA	A	3.5	3.5	0.0	3.3	5.4	0.0
		B	16.4	0.0	0.0	10.9	0.0	0.0
		C ₁	47.1	8.1	0.0	22.7	1.4	0.0
		C ₂	36.3	6.2	0.0	46.3	1.7	0.0
		C ₃	26.5	5.9	0.1	26.7	2.4	0.0
		C ₄	57.4	9.0	0.0	29.8	3.8	-0.1
		C ₅	25.5	15.5	0.0	29.9	7.2	0.0
17	GA	A	3.8	2.6	0.0	3.5	0.9	0.0
		B	22.2	0.0	0.0	5.7	0.0	0.0
		C ₁	24.4	1.4	0.0	17.9	0.9	0.0
		C ₂	38.1	5.7	0.1	39.5	1.0	0.0
		C ₃	63.1	5.9	0.1	22.9	1.5	-0.1
		C ₄	24.9	4.7	0.0	39.2	1.6	0.0
		C ₅	29.9	7.5	0.0	32.7	2.2	0.0
18	GA	A	3.6	1.0	0.0	4.0	0.9	0.0
		B	27.4	0.0	0.0	7.9	0.0	0.0
		C ₁	20.4	1.1	0.0	23.6	0.8	0.0
		C ₂	24.0	2.6	0.0	17.6	1.2	0.0
		C ₃	46.8	2.3	0.0	17.6	1.0	-0.1
		C ₄	28.8	1.9	0.0	27.4	1.3	0.0
		C ₅	33.4	2.2	0.0	21.5	1.3	0.0
19	GA	A	3.6	3.9	0.0	3.3	1.8	0.0
		B	10.2	0.0	0.0	18.2	0.0	0.0
		C ₁	38.7	2.4	0.0	36.5	1.5	0.0
		C ₂	33.3	26.3	0.0	46.5	3.7	0.0
		C ₃	37.8	9.1	0.0	29.7	4.8	0.1
		C ₄	22.4	13.4	0.0	25.0	4.7	0.0
		C ₅	27.0	9.4	0.0	18.9	3.3	0.0
20	GA	A	3.5	1.0	0.0	3.8	0.8	0.0
		B	39.1	0.0	0.0	14.8	0.0	0.0
		C ₁	22.9	3.4	0.0	20.6	1.2	0.0
		C ₂	31.4	1.8	0.0	22.4	1.1	0.0
		C ₃	27.5	2.6	-0.1	32.3	1.1	0.0
		C ₄	17.7	1.7	0.0	16.1	1.4	0.0
		C ₅	26.8	1.8	0.0	23.3	1.6	0.0
En Küçük			2.25	0.00	0.0	3.28	0.00	0.00
Ortalama			25.72	4.19	0.0	22.52	1.53	0.02
En Büyük			66.00	47.61	0.1	66.00	10.60	3.4

4 makineden oluşan veri kümeleri için sonuçlar incelendiğinde de benzer sonuçlar çıkarılmaktadır. Çizelge 5.7, 5.8 ve 5.9'da verilen değerlere göre matematiksel model için ortalama çözüm süresi yaklaşık 10 dakikadır. Sezgisel model için ise bu süre 24 saniyedir. Ayrıca, sezgisel model ile alınan çözümlerden en uzun süren koşturum 234 saniyedir. Bu süre sezgisel model için 6. veri kümesinin C_3 noktasında küçük θ değeri için ortaya çıkmıştır. matematiksel model ile bu noktadaki çözüm süresi de 1037 saniye ile kendi ortalamasından %70 fazladır.

4 makine için gerçekleştirilen koşturumlardan yalnızca 1 tanesinde matematiksel model zaman limitine takılarak optimal çözümü elde edememiştir. Küçük θ değerlerine sahip 10. veri kümesinin C_1 noktasında ortaya çıkan bu durum için matematiksel modelin hata yüzdesi %1286 değerindedir. Buna karşılık ETA modelinin bu noktada çözüm süresi 66.6 saniye olup elde edilen çözüm matematiksel modelden %3.9 daha iyidir.

Çözüm kalitesi açısından değerlendirildiğinde sezgisel modelin küçük ve büyük θ değerleri için ortalama %0.03 ve %0.06 hata yüzde oranlarına sahip olduğu Çizelge 5.9'da görülmektedir. Yukarıda bahsedilen 10. veri kümesinin C_1 noktasında gerçekleşen -%3.9 değeri bu ortalamaya katılmamıştır. Ayrıca, daha önce açıklanan 2 ve 3 makineli veri kümeleri koşturumlarında olduğu gibi, rasyonel sayıların yüzde birler basamağında oluşan farklılıklar nedeniyle ortaya çıkan -%0.01'lik hata oranı değerleri de mutlak değerleri alınarak bu ortalamaya dahil edilmiştir. Yine Çizelge 5.9'da görüleceği gibi en büyük hata oranı ETA için %4 değerindedir.

6 makineli veri setleri için MISOCP ve ETA modelleri ile alınan sonuçlar ise Çizelge 5.10 ve 5.11'de gösterilmiştir. Matematiksel model 6 makine için B, C_1, C_2, C_3, C_4 noktalarında zaman limitine takılarak durmuş, optimal sonucu elde edememiştir. C_1, C_2, C_3, C_4 noktaları için MISOCP ve ETA'da aynı çevrim zamanı sınırı ile enerji tüketim değeri bulunmaya çalışılmıştır. Yani, eğer matematiksel sezgisel model ile elde edilen A noktası çevrim zamanı değeri ETA ile elde edilen değerden küçükse, küçük olan CT değeri baz alınmış ve artırım miktarı olan λ küçük olan çevrim zamanı değerine göre hesaplanmıştır. Böylece iki modelin bu noktalar için kıyaslanması olanaklı hale getirilmiştir.

Çizelge 5.7: 4 makine, ETA ve MISOCP karşılaştırması

Veri Kümesi	P	Etkin Çözüm No.	Küçük θ			Büyük θ		
			MISOCP	ETA		MISOCP	ETA	
			Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)	Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)
1	DA	A	120.5	16.3	0.0	146.8	16.6	0.0
		B	238.4	0.0	0.0	159.5	0.0	0.0
		C ₁	1122.2	14.5	-0.1	509.2	11.1	0.1
		C ₂	854.3	17.9	0.0	680.3	22.8	0.1
		C ₃	1460.1	31.3	0.0	469.8	26.0	0.0
		C ₄	842.0	33.6	0.0	966.8	25.9	0.0
		C ₅	13.6	0.0	0.0	43.0	0.0	0.0
2	DA	A	98.9	13.8	0.0	95.9	14.5	0.0
		B	300.6	0.0	0.0	181.3	0.0	0.0
		C ₁	932.5	27.4	0.0	445.5	9.9	0.0
		C ₂	1609.9	25.6	-0.1	575.5	23.7	0.0
		C ₃	929.6	29.4	0.1	822.6	25.0	0.0
		C ₄	517.7	30.0	0.0	971.9	23.8	-0.1
		C ₅	18.1	0.0	0.0	41.9	0.0	0.0
3	DA	A	119.7	30.3	0.0	43.6	15.6	0.0
		B	196.1	0.0	0.0	113.7	0.0	0.0
		C ₁	721.2	23.6	0.0	468.5	28.0	0.0
		C ₂	835.2	67.0	0.0	1220.5	57.1	0.0
		C ₃	766.2	57.4	0.0	425.3	29.7	0.0
		C ₄	684.3	72.0	0.0	602.9	27.6	0.1
		C ₅	16.2	0.0	0.0	10.8	0.0	0.0
4	DA	A	81.3	13.3	0.0	93.5	14.1	0.0
		B	510.8	0.0	0.0	859.5	0.0	0.0
		C ₁	1153.4	21.2	0.0	1456.2	10.1	0.0
		C ₂	1436.1	27.6	0.0	971.1	27.9	0.0
		C ₃	694.8	29.0	0.1	986.7	24.3	-0.1
		C ₄	728.7	30.8	0.0	982.7	23.8	-0.1
		C ₅	18.5	0.0	0.0	18.9	0.0	0.0
5	DA	A	39.8	18.4	0.0	30.9	14.6	0.0
		B	145.1	0.0	0.0	245.3	0.0	0.0
		C ₁	1103.7	23.7	0.0	545.2	11.3	0.0
		C ₂	575.4	38.7	0.0	754.3	13.8	0.0
		C ₃	1319.9	30.1	-0.1	1498.8	25.3	0.0
		C ₄	911.7	36.6	0.0	1527.8	26.7	0.0
		C ₅	9.6	0.0	0.0	6.9	0.0	0.0
6	DA	A	92.4	32.5	0.0	38.2	17.4	0.0
		B	131.8	0.0	0.0	120.8	0.0	0.0
		C ₁	1279.7	94.9	0.0	542.2	44.3	0.0
		C ₂	1037.1	234.3	0.0	1106.1	45.7	0.0
		C ₃	1236.5	81.0	-0.1	1175.1	114.5	0.0
		C ₄	810.5	59.2	0.0	611.1	38.2	0.0
		C ₅	19.7	0.0	0.0	11.5	0.0	0.0
7	DA	A	110.3	25.5	0.0	32.7	16.4	0.0
		B	203.6	0.0	0.0	157.7	0.0	0.0
		C ₁	885.4	70.5	0.0	398.3	13.8	0.0
		C ₂	716.3	52.1	0.0	843.6	27.1	0.0
		C ₃	537.8	51.4	-0.1	1284.0	32.6	0.0
		C ₄	1132.8	66.4	0.0	1160.3	29.3	0.0
		C ₅	30.1	0.0	0.0	10.7	0.0	0.0
8	DA	A	115.9	19.1	0.0	30.0	19.9	0.0
		B	167.1	0.0	0.0	230.2	0.0	0.0
		C ₁	1135.5	31.0	0.0	644.2	10.7	0.0
		C ₂	1489.8	33.5	0.0	1249.2	25.2	0.0
		C ₃	1441.9	39.2	0.0	552.5	70.4	0.0
		C ₄	1962.0	76.6	0.0	429.2	41.4	0.0
		C ₅	50.2	0.0	0.0	4.8	0.0	0.0

Çizelge 5.8: 4 makine, ETA ve MISOCP karşılaştırması (Devamı)

Veri Kümesi	P	Etkin Çözüm No.	Küçük θ			Büyük θ		
			MISOCP	ETA		MISOCP	ETA	
			Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)	Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)
9	DA	A	39.2	16.2	0.0	44.4	16.0	0.0
		B	530.3	0.0	0.0	211.8	0.0	0.0
		C ₁	1214.2	25.2	0.0	264.4	22.9	-0.1
		C ₂	732.0	26.2	0.0	520.7	29.0	0.0
		C ₃	960.2	28.6	0.0	1433.6	26.0	0.0
		C ₄	357.3	60.9	0.0	1121.4	35.6	0.0
		C ₅	8.1	0.0	0.0	9.4	0.0	0.0
10	DA	A	96.8	22.8	0.0	46.4	15.7	0.0
		B	233.2	0.0	0.0	405.5	0.0	0.0
		C ₁	3600.1	66.6	-3.9	890.0	28.4	0.0
		C ₂	826.5	45.0	-0.1	1096.0	30.8	0.0
		C ₃	1713.3	58.6	0.0	789.0	36.0	0.0
		C ₄	880.6	61.4	0.0	1070.5	30.6	0.0
		C ₅	21.8	0.0	0.0	11.0	0.0	0.0
11	GA	A	51.1	12.4	0.0	36.4	15.5	0.0
		B	329.6	0.0	0.0	197.2	0.0	0.0
		C ₁	832.0	23.1	0.0	567.3	28.5	0.0
		C ₂	438.9	33.2	0.0	541.1	26.8	-0.1
		C ₃	824.4	32.2	-0.1	749.3	23.2	0.0
		C ₄	682.1	35.8	0.0	975.7	25.0	-0.1
		C ₅	6.3	0.0	0.0	5.9	0.0	0.0
12	GA	A	40.0	33.6	0.0	30.3	18.2	0.0
		B	144.7	0.0	0.0	134.1	0.0	0.0
		C ₁	1363.4	39.9	0.0	255.1	12.0	0.0
		C ₂	1403.6	68.5	0.0	1028.7	29.7	0.0
		C ₃	1496.6	96.1	-0.1	560.8	27.7	0.0
		C ₄	728.6	106.9	0.0	766.5	31.5	-0.1
		C ₅	9.2	0.0	0.0	9.4	0.0	0.0
13	GA	A	55.5	16.1	0.0	41.4	21.9	0.0
		B	229.5	0.0	0.0	274.1	0.0	0.0
		C ₁	732.6	24.5	0.0	380.1	10.2	0.3
		C ₂	822.5	28.4	0.0	361.0	13.6	0.0
		C ₃	555.7	198.5	0.0	1422.1	26.4	0.0
		C ₄	868.4	36.3	0.0	577.5	27.8	-0.1
		C ₅	8.6	0.0	0.0	13.0	0.0	0.0
14	GA	A	42.9	32.6	0.0	31.3	27.5	0.0
		B	432.8	0.0	0.0	210.0	0.0	0.0
		C ₁	800.1	37.5	0.0	728.2	16.5	1.3
		C ₂	1197.7	59.2	0.0	1126.7	24.7	0.0
		C ₃	1375.9	62.9	0.1	653.1	32.9	0.0
		C ₄	1327.6	72.5	0.0	588.5	45.9	0.0
		C ₅	5.6	0.0	0.0	7.0	0.0	0.0
15	GA	A	33.0	16.1	0.0	29.6	14.7	0.0
		B	555.8	0.0	0.0	280.8	0.0	0.0
		C ₁	1210.0	137.2	0.0	970.6	25.4	4.0
		C ₂	677.6	135.2	0.0	487.0	23.4	0.0
		C ₃	450.2	33.3	-0.1	591.0	25.6	0.0
		C ₄	751.8	35.4	0.0	1414.2	24.8	0.0
		C ₅	8.6	0.0	0.0	9.7	0.0	0.0
16	GA	A	35.5	16.3	0.0	39.5	15.1	0.0
		B	480.4	0.0	0.0	217.8	0.0	0.0
		C ₁	611.0	23.1	0.0	593.5	26.5	1.3
		C ₂	1390.4	19.0	0.0	572.7	12.4	0.0
		C ₃	628.4	36.8	0.0	1506.4	21.9	0.0
		C ₄	557.5	35.2	0.0	1382.5	24.8	0.1
		C ₅	8.4	0.0	0.0	4.7	0.0	0.0

Çizelge 5.9: 4 makine, ETA ve MISOCP karşılaştırması (Devamı)

Veri Kümesi	P	Etkin Çözüm No.	Küçük θ			Büyük θ		
			MISOCP	ETA		MISOCP	ETA	
			Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)	Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)
17	GA	A	104.2	19.1	0.0	38.2	19.7	0.0
		B	217.7	0.0	0.0	181.9	0.0	0.0
		C ₁	460.7	26.8	0.0	623.1	12.3	0.0
		C ₂	682.8	31.6	0.0	797.4	25.7	0.0
		C ₃	2130.0	29.6	0.0	904.8	24.8	0.0
		C ₄	815.8	45.4	0.0	407.6	34.3	0.0
		C ₅	20.3	0.0	0.0	10.7	0.0	0.0
18	GA	A	42.1	14.0	0.0	39.5	16.1	0.0
		B	353.2	0.0	0.0	220.8	0.0	0.0
		C ₁	1542.4	24.1	0.0	392.9	10.6	0.0
		C ₂	1216.9	31.6	0.0	630.3	22.9	0.0
		C ₃	813.7	31.9	0.0	1095.7	25.6	0.0
		C ₄	747.4	29.3	0.0	2963.8	23.3	0.2
		C ₅	5.8	0.0	0.0	6.3	0.0	0.0
19	GA	A	41.5	14.5	0.0	49.5	14.8	0.0
		B	101.6	0.0	0.0	199.7	0.0	0.0
		C ₁	933.0	22.9	0.0	1452.1	10.1	0.0
		C ₂	1276.8	28.1	0.0	2801.3	27.8	0.0
		C ₃	814.0	27.5	-0.1	1101.9	26.6	0.0
		C ₄	720.5	30.6	0.0	1482.8	24.9	0.0
		C ₅	9.6	0.0	0.0	4.3	0.0	0.0
20	GA	A	90.3	15.8	0.0	28.8	14.0	0.0
		B	142.5	0.0	0.0	351.6	0.0	0.0
		C ₁	2454.2	22.8	1.8	743.2	14.1	0.0
		C ₂	765.7	28.3	-0.1	880.0	25.5	0.0
		C ₃	1749.8	32.5	-0.1	1887.2	25.5	-0.1
		C ₄	1662.2	36.8	0.0	517.0	28.4	0.0
		C ₅	29.1	0.0	0.0	9.9	0.0	0.0
En Küçük		5.59	0.00	0.0	4.28	0.00	0.00	
Ortalama		650.53	30.14	0.03	555.09	17.82	0.06	
En Büyük		3600.11	234.26	1.8	2963.84	114.45	4.0	

Şekil 4.1'deki A ve C₅ noktaları ise bu veri kümeleri için en küçük ve en büyük çevrim zamanı çözümleridir. 2. dereceden kısıtlar içermeyen bu problemler Karma Tamsayı Programlama(MIP) olarak modellenmiş ve CPLEX çözücüsü ile çözdürülmüştür. En küçük çevrim zamanı noktası olan A için 6-makineli problemlerde zaman limiti dahilinde MIP ile optimal çözümler elde edilememiştir. Bu nokta için verilen CT-Sapma değerleri MIP ve ETA ile elde edilen çözüm değerlerinden daha küçük olanına göre hesaplanmıştır. C₅ noktası için ise için optimal çözümler zaman limiti dahilinde MIP model ile bulunmuştur. Çizelgelerde, matematiksel model ile çözüm elde edilemeyen noktalarda ETA ile çözüm elde edilmesine rağmen, modelin hata yüzde oranı kontrol edilemeyeceğinden bu noktaların değerleri "ÇY" olarak gösterilmiştir.

Çizelge 5.10: 6 makine, ETA ve MISOCP karşılaştırması

Veri Kümesi	P	Etkin Çözüm No.	Küçük θ				Büyük θ			
			MISOCP		ETA		MISOCP		ETA	
			Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)	CT Sapma (%)	Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)	CT Sapma (%)
1	DA	A	ZL	151.4	0.0	0.0	ZL	165.2	0.0	0.0
		B	ZL	0.0	ÇY	ÇY	ZL	0.0	0.0	0.0
		C ₁	ZL	362.9	-7.6	0.0	ZL	353.6	-16.0	0.0
		C ₂	ZL	367.8	-1.3	0.0	ZL	367.6	-26.4	0.0
		C ₃	ZL	372.1	28.3	0.0	ZL	368.4	-26.3	0.0
		C ₄	ZL	373.4	-19.3	-12.1	ZL	369.6	-8.4	-7.6
		C ₅	758.4	0.0	0.0	987.7	0.0	0.0	0.0	
2	DA	A	ZL	177.7	0.0	4.5	ZL	158.7	-0.2	-10.9
		B	ZL	0.0	7.1	4.5	ZL	0.0	-7.5	-10.9
		C ₁	ZL	365.7	-11.3	0.0	ZL	346.1	-3.8	-7.6
		C ₂	ZL	365.7	-34.3	0.0	ZL	360.3	-3.6	-5.6
		C ₃	ZL	369.4	24.4	0.0	ZL	363.5	-70.2	-4.1
		C ₄	ZL	372.4	14.0	-10.5	ZL	364.3	-0.2	-12.1
		C ₅	2592.3	0.0	0.0	1656.5	0.0	0.0	0.0	
3	DA	A	ZL	144.3	18.7	3.7	ZL	183.6	-7.7	-4.2
		B	ZL	0.0	ÇY	ÇY	ZL	0.0	32.0	-4.2
		C ₁	ZL	363.2	1.5	0.0	ZL	354.3	29.2	-3.2
		C ₂	ZL	362.7	-0.2	0.0	ZL	361.9	-27.6	-2.5
		C ₃	ZL	361.2	-15.6	0.0	ZL	360.7	-16.5	-1.9
		C ₄	ZL	362.7	-14.7	-5.6	ZL	362.8	-14.0	-1.5
		C ₅	1429.8	0.0	0.0	2906.5	0.0	0.0	0.0	
4	DA	A	ZL	148.8	0.0	0.0	ZL	168.2	0.0	0.0
		B	ZL	0.0	ÇY	ÇY	ZL	0.0	0.0	0.0
		C ₁	ZL	360.1	0.9	0.0	ZL	294.8	-28.5	0.0
		C ₂	ZL	362.7	-6.8	0.0	ZL	362.5	-34.3	0.0
		C ₃	ZL	363.6	-18.2	0.0	ZL	365.2	-34.6	0.0
		C ₄	ZL	365.0	-7.7	-11.7	ZL	362.1	-7.3	-6.7
		C ₅	1083.1	0.0	0.0	1183.1	0.0	0.0	0.0	
5	DA	A	ZL	154.0	-0.4	-2.5	ZL	164.9	0.0	0.6
		B	ZL	0.0	17.8	-2.5	ZL	0.0	-1.5	0.6
		C ₁	ZL	364.1	36.4	-1.8	ZL	362.5	-19.7	0.0
		C ₂	ZL	367.1	-6.5	-1.3	ZL	358.7	33.8	0.0
		C ₃	ZL	361.0	-19.3	-1.0	ZL	364.3	45.3	0.0
		C ₄	ZL	366.2	-2.1	-4.4	ZL	360.8	39.1	0.0
		C ₅	1615.8	0.0	0.0	1488.3	0.0	0.0	0.0	
6	DA	A	ZL	169.9	0.0	1.7	ZL	154.7	-8.3	13.3
		B	ZL	0.0	-4.1	1.7	ZL	0.0	ÇY	ÇY
		C ₁	ZL	364.2	-3.8	0.0	ZL	362.1	-23.0	0.0
		C ₂	ZL	369.1	-32.2	0.0	ZL	363.4	-3.2	0.0
		C ₃	ZL	383.2	-41.8	0.0	ZL	371.0	-22.0	0.0
		C ₄	ZL	379.7	-48.8	-3.2	ZL	364.4	57.3	0.0
		C ₅	1407.2	0.0	0.0	1222.9	0.0	0.0	0.0	
7	DA	A	ZL	153.8	0.0	1.2	ZL	158.3	-7.7	-4.6
		B	ZL	0.0	ÇY	ÇY	ZL	0.0	13.7	-4.6
		C ₁	ZL	362.3	-24.0	0.0	ZL	342.0	5.8	-3.3
		C ₂	ZL	361.1	-27.9	0.0	ZL	362.4	-61.4	-2.4
		C ₃	ZL	363.4	-27.5	0.0	ZL	364.8	-68.1	-1.8
		C ₄	ZL	368.3	27.6	-13.0	ZL	360.1	-9.0	-6.9
		C ₅	1364.1	0.0	0.0	878.3	0.0	0.0	0.0	
8	DA	A	ZL	168.4	8.4	9.5	ZL	165.4	0.0	-5.2
		B	ZL	0.0	ÇY	ÇY	ZL	0.0	-13.2	-5.2
		C ₁	ZL	360.2	-51.6	0.0	ZL	341.5	-55.8	-3.6
		C ₂	ZL	360.8	-33.6	0.0	ZL	362.1	-14.7	-2.6
		C ₃	ZL	365.1	-48.3	0.0	ZL	368.3	-10.9	-1.9
		C ₄	ZL	365.0	-59.4	-10.3	ZL	366.2	-36.4	-4.0
		C ₅	1159.6	0.0	0.0	1354.5	0.0	0.0	0.0	
9	DA	A	ZL	166.5	8.4	11.0	ZL	166.3	0.0	11.3
		B	ZL	0.0	7.7	11.0	ZL	0.0	ÇY	ÇY
		C ₁	ZL	362.0	-18.2	0.0	ZL	301.6	-27.9	0.0
		C ₂	ZL	362.3	-31.2	0.0	ZL	368.4	-43.5	0.0
		C ₃	ZL	361.3	-24.5	0.0	ZL	362.7	10.2	0.0
		C ₄	ZL	364.4	-31.3	-11.4	ZL	362.8	-4.8	-6.9
		C ₅	1130.8	0.0	0.0	1305.3	0.0	0.0	0.0	
10	DA	A	ZL	199.5	-0.1	7.3	ZL	164.9	0.0	3.4
		B	ZL	0.0	-16.1	20.0	ZL	0.0	6.2	15.6
		C ₁	ZL	365.0	-38.7	0.0	ZL	334.6	-1.7	0.0
		C ₂	ZL	378.5	-18.3	0.0	ZL	364.8	-20.0	0.0
		C ₃	ZL	382.9	-11.4	0.0	ZL	371.8	45.3	0.0
		C ₄	ZL	385.0	22.2	-12.3	ZL	364.1	10.8	-2.6
		C ₅	1866.8	0.0	0.0	1594.1	0.0	0.0	0.0	

Çizelge 5.11: 6 makine, ETA ve MISOCP karşılaştırması (Devamı)

Veri Kümesi	P	Etkin Çözüm No.	Küçük θ				Büyük θ				
			MISOCP		ETA		MISOCP		ETA		
			Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)	CT Sapma (%)	Çözüm Süresi (sn)	Çözüm Süresi (sn)	Enerji Sapma (%)	CT Sapma (%)	
11	GA	A	ZL	165.2	0.0	2.6	ZL	182.2	0.0	2.6	
		B	ZL	0.0	ÇY	ÇY	ZL	0.0	-7.4	2.6	
		C ₁	ZL	362.3	38.3	0.0	ZL	317.4	31.2	0.0	
		C ₂	ZL	362.3	4.3	0.0	ZL	361.8	-10.6	0.0	
		C ₃	ZL	365.4	-12.3	0.0	ZL	368.4	-5.4	0.0	
		C ₄	ZL	360.7	31.6	-11.8	ZL	363.4	-7.4	-11.2	
		C ₅	2290.5	0.0	0.0	1099.9	0.0	0.0	0.0		
12	GA	A	ZL	188.2	0.0	8.7	ZL	166.2	-7.7	2.0	
		B	ZL	0.0	-5.6	8.7	ZL	0.0	4.6	2.0	
		C ₁	ZL	360.5	-8.6	0.0	ZL	333.0	-2.1	0.0	
		C ₂	ZL	377.8	-2.7	0.0	ZL	366.4	2.8	0.0	
		C ₃	ZL	379.9	-1.2	-0.6	ZL	365.6	4.5	0.0	
		C ₄	ZL	380.6	28.1	-14.4	ZL	365.0	20.0	-5.3	
		C ₅	1276.5	0.0	0.0	1291.2	0.0	0.0	0.0		
13	GA	A	ZL	204.8	7.5	6.7	ZL	159.4	-14.3	2.3	
		B	ZL	0.0	17.0	6.7	ZL	0.0	41.0	2.4	
		C ₁	ZL	362.7	-25.4	0.0	ZL	348.6	-8.9	0.0	
		C ₂	ZL	365.2	4.8	0.0	ZL	365.2	-39.1	0.0	
		C ₃	ZL	365.1	16.7	0.0	ZL	363.3	-32.5	0.0	
		C ₄	ZL	366.2	27.0	-12.8	ZL	361.3	34.4	-6.8	
		C ₅	1230.0	0.0	0.0	1404.6	0.0	0.0	0.0		
14	GA	A	ZL	178.8	8.5	6.4	ZL	159.3	-7.7	2.7	
		B	ZL	0.0	ÇY	ÇY	ZL	0.0	-16.3	2.7	
		C ₁	ZL	359.3	-22.5	0.0	ZL	349.7	-42.1	0.0	
		C ₂	ZL	379.6	10.3	0.0	ZL	366.7	-18.7	0.0	
		C ₃	ZL	380.9	5.8	-0.1	ZL	365.8	-50.6	0.0	
		C ₄	ZL	379.7	44.0	-9.3	ZL	363.6	12.7	-5.4	
		C ₅	1656.2	0.0	0.0	1291.7	0.0	0.0	0.0		
15	GA	A	ZL	228.4	0.1	9.3	ZL	165.7	0.0	7.4	
		B	ZL	0.0	ÇY	ÇY	ZL	0.0	ÇY	ÇY	
		C ₁	ZL	361.0	-37.1	0.0	ZL	359.8	24.0	0.0	
		C ₂	ZL	362.8	0.0	0.0	ZL	364.0	9.6	0.0	
		C ₃	ZL	361.9	10.4	0.0	ZL	368.5	26.0	0.0	
		C ₄	ZL	366.4	36.4	-11.0	ZL	363.1	7.9	-9.0	
		C ₅	1360.4	0.0	0.0	1611.3	0.0	0.0	0.0		
16	GA	A	ZL	226.1	-8.3	5.3	ZL	167.6	16.5	-8.6	
		B	ZL	0.0	-20.8	5.3	ZL	0.0	30.2	-8.6	
		C ₁	ZL	366.5	-6.4	0.0	ZL	341.3	19.9	-5.9	
		C ₂	ZL	362.0	-7.3	-0.7	ZL	365.3	31.0	-4.2	
		C ₃	ZL	399.6	6.8	-1.5	ZL	373.1	-0.1	-3.1	
		C ₄	ZL	400.7	-10.5	-14.4	ZL	374.6	24.3	-16.6	
		C ₅	1667.9	0.0	0.0	1355.3	0.0	0.0	0.0		
17	GA	A	ZL	210.1	8.3	6.4	ZL	180.9	16.4	-8.4	
		B	ZL	0.0	23.3	6.4	ZL	0.0	53.1	-8.4	
		C ₁	ZL	363.0	-41.9	0.0	ZL	349.6	-5.9	-5.9	
		C ₂	ZL	363.3	-15.7	0.0	ZL	359.8	26.8	-4.3	
		C ₃	ZL	364.6	18.6	0.0	ZL	363.0	-39.2	-3.2	
		C ₄	ZL	370.7	13.6	-13.5	ZL	366.4	-33.7	-7.3	
		C ₅	1514.1	0.0	0.0	1347.9	0.0	0.0	0.0		
18	GA	A	ZL	222.8	9.0	4.7	ZL	168.3	24.9	-2.8	
		B	ZL	0.0	-8.4	4.7	ZL	0.0	39.5	-2.8	
		C ₁	ZL	362.9	-7.0	0.0	ZL	366.0	-19.7	-1.9	
		C ₂	ZL	371.9	-22.8	0.0	ZL	366.2	-1.3	-1.4	
		C ₃	ZL	366.6	-0.9	0.0	ZL	364.8	-31.7	-1.0	
		C ₄	ZL	375.4	13.6	-4.6	ZL	372.2	24.2	-11.2	
		C ₅	1433.8	0.0	0.0	1857.8	0.0	0.0	0.0		
19	GA	A	ZL	218.6	-8.3	6.0	ZL	172.5	15.4	12.0	
		B	ZL	0.0	-24.6	6.0	ZL	0.0	ÇY	ÇY	
		C ₁	ZL	358.5	-18.1	0.0	ZL	339.5	-32.9	0.0	
		C ₂	ZL	362.2	3.1	0.0	ZL	360.2	1.5	0.0	
		C ₃	ZL	363.0	5.7	0.0	ZL	361.9	-39.8	0.0	
		C ₄	ZL	363.5	13.1	-9.3	ZL	365.3	1.6	-8.7	
		C ₅	1959.5	0.0	0.0	1167.5	0.0	0.0	0.0		
20	GA	A	ZL	207.1	-16.6	4.8	ZL	171.8	0.0	2.5	
		B	ZL	0.0	ÇY	ÇY	ZL	0.0	5.2	2.5	
		C ₁	ZL	362.4	0.0	0.0	ZL	359.4	0.0	0.0	
		C ₂	ZL	368.6	-63.8	0.0	ZL	364.0	-60.3	0.0	
		C ₃	ZL	368.2	-20.3	0.0	ZL	361.6	-12.3	0.0	
		C ₄	ZL	371.2	32.9	-14.4	ZL	366.3	12.9	-11.6	
		C ₅	1467.5	0.0	0.0	1238.9	0.0	0.0	0.0		
En Küçük Ortalama				758.4	0.0	-63.8	-14.4	878.3	0.0	-70.2	-16.6
En Büyük				1513.2	236.4	-3.9	-0.4	1412.2	229.2	-3.3	-1.6
				ZL	400.7	44.0	20.0	ZL	374.6	57.3	15.6

Sezgisel modelin performansını çözüm süresi olarak kıyaslandığında 233.4 sn ile matematiksel modelin hayli düşük olduğu söylenebilir. Ancak buna bağlı olarak, enerji ve minimum çevrim zamanı sonuçlarında dalgalanmalar görülmektedir. Küçük θ değerine sahip veri kümelerinde ETA modelinin en fazla %20 olduğu ortalama hata yüzde oranının ise -%0.4 olduğu Çizelge 5.11’de verilmiştir. ETA’nın MISOCP’den daha iyi sonuç verdiği çözüm noktaları için ortalama %19 oranında daha iyi sonuçlar verdiği görülmüştür. Ters durumlar için de MISOCP modeli ortalama %15 daha iyi çözümler vermektedir. Veri kümeleri ile büyük θ kullanıldığında ise ETA’nın performansının kötüleştiği görülmüştür. Bu veri kümeleri için alınan 140 çözümde enerji sapma değerleri %20’den fazla olan nokta sayısı ise 21’dir. Ayrıca, kümelerinde ETA modeli ile %70.2 oranına varan daha iyi sonuçlar elde edildiği gibi, % 57.3 oranında matematiksel modelin daha iyi çözümler verdiği noktalar da bulunmaktadır.

5.3 Aktivite Sıralamaları Sonuçları

2 makineli veri kümelerinden elde edilen çözümler incelendiğinde optimal çözümlerde aşağıda aktivite sıralamaları verilen 6 çevrimin elde edildiği görülmektedir.

$$2.1 \quad U_0(1,2) \longrightarrow L_1(0,2) \longrightarrow U_2(3,2) \longrightarrow L_3(0,2) \longrightarrow L_2(0,0) \longrightarrow U_1(0,2)$$

$$2.2 \quad U_0(1,0) \longrightarrow U_1(1,2) \longrightarrow L_1(0,2) \longrightarrow U_2(3,2) \longrightarrow L_3(0,2) \longrightarrow L_2(0,0)$$

$$2.3 \quad U_0(1,2) \longrightarrow L_1(0,2) \longrightarrow U_2(3,2) \longrightarrow L_2(3,0) \longrightarrow L_3(0,0) \longrightarrow U_1(0,2)$$

$$2.4 \quad U_0(1,0) \longrightarrow L_1(0,0) \longrightarrow U_2(3,0) \longrightarrow L_3(0,0) \longrightarrow U_1(2,0) \longrightarrow L_2(0,0)$$

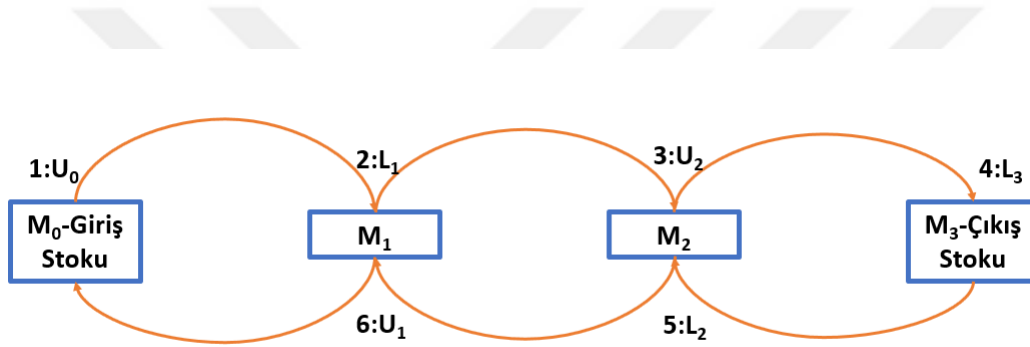
$$2.5 \quad U_0(1,2) \longrightarrow L_1(0,2) \longrightarrow U_2(3,2) \longrightarrow L_2(3,0) \longrightarrow U_1(3,2) \longrightarrow L_3(0,2)$$

$$2.6 \quad U_0(1,0) \longrightarrow L_1(0,0) \longrightarrow U_1(2,0) \longrightarrow L_2(0,0) \longrightarrow U_2(3,0) \longrightarrow L_3(0,0)$$

$m = 2$ olan 40 veri kümesi için, minimum çevrim zamanı \overline{CT} ve $maxCT$ değerleri elde edildikten sonra, $[\overline{CT}, maxCT]$ aralığı belirlenmiştir. Bu aralık 10 eşit parçaya ayrılarak, artış miktarı olan λ değeri hesaplanmış ve etkin çözümler her adımda \overline{CT} λ birim artırılarak elde edilmiştir. Artırma işlemi 10 kez yapılmış, ayrıca minimum enerjiyi sağlayan maksimum çevrim zamanı da çözüm kümesine eklenmiştir. Sonuç

olarak bir veri kümesi için en küçük ve en büyük çevrim zamanı çözümlerini de içeren 12 nokta için Pareto etkin çözüm araştırması yapılmıştır. Toplamda gerçekleştirilen 480 koşturum sonucu, 2 makine için elde edilen sonuçlar şu şekilde özetlenebilir.

2.1 numaralı döngü, en küçük çevrim zamanı ve enerji tüketiminin sağlandığı A noktası için (Şekil 4.1) veri kümelerinin %57'inde optimal değerleri veren döngüdür. Şekil 5.1 ile bu döngünün doğrusal makine dizilimi olduğunda nasıl gerçekleştiği gösterilmiştir. Görüldüğü üzere robot, bir sonraki pozisyon için en yakındaki makineyi tercih etmiş böylece olası zaman kayıplarının önüne geçmiştir. Büyük ve küçük θ değerleri için elde edilen çözümlerin %85'inde ortaya çıkan bu döngü, ara çözümlerin ise %95'inde optimal aktivite sıralaması olarak elde edilmektedir.



Şekil 5.1: 2.4 Numaralı Döngü

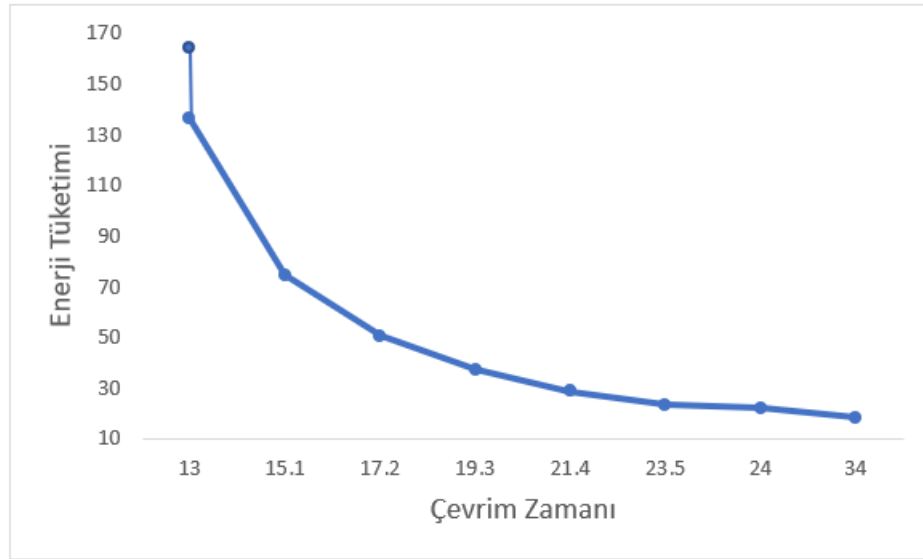
Küçük θ değerine sahip veri kümelerinde, EÇK'de A ve B noktaları için elde edilen diğer iki aktivite sıralaması da döngü 2.2 ve 2.3 şeklindedir. Yalnızca A ve B noktalarında ortaya çıkan bu iki çevrim, hiç bir C noktasında elde edilmemiştir. Bunun sebebi olarak, iki döngü için de 4 defa aktif tutucu değişimi gerçekleşmesi gerekçe olarak gösterilebilir. C noktalarında genişleyen çevrim zamanı limiti ile tutucu değişimi gereksinimi azalmış böylece 2.2 ve 2.3 numaralı döngüler yerini 2.1 numaralı döngüye bırakmıştır. Büyük θ değerine sahip veri kümelerinde, çevrim zamanında oluşturacakları artış daha büyük olacağından, 4 defa tutucu değişimi gerektiren bu çevrimler hiç kullanılmamıştır.

Çevrim 2.4 ve 2.5 ise yalnızca büyük θ değerine sahip kümelerde açığa çıkan çözümlerdir. Bu aktivite sıralamalarının belirtilen veri kümeleri için A ve B noktası çözümlerinde kullanılma oranı %40 iken C noktası çözümlerinde bu çevrimlere rastlanmamış-

tır.

2.6 numaralı döngü ise çevrim zamanı limiti yüksek olduğunda ortaya çıkan ve tek tutucu kullanımının yeterli olduğu (UPHILL) döngüdür. Bu çevrim tüm veri setlerinde, yalnızca en büyük çevrim zamanı ve en küçük enerji tüketimine sahip son C noktasında ortaya çıkmamıştır.

Şekil 5.2, 2 makineden oluşan, makineler arası uzaklığın özdeş toplamsal ve işlem süreleri varyansının düşük olduğu bir veri kümesi için alınan etkin çözümler kümesini göstermektedir. Bu şekilde görülebileceği üzere çevrim zamanı üst limiti 13 olan iki çözüm bulunmaktadır. Bunlardan ilki Şekil 4.1'deki A noktasına karşılık gelen, yaklaşık 170 birim enerji tüketimi değerine sahip olan ve çevrim zamanı minimizasyonu modeli ile elde edilen çözümdür. Diğeri ise, bu ilk modelden elde edilen çevrim zamanı değerinin enerji minimizasyonu modeline üst limit olarak eklenmesiyle elde edilen B noktası çözümdür. Bu ikinci çözüm yaklaşık 130 birim ile daha düşük enerji tüketim değerine sahiptir. Dolayısıyla, ikinci çözüm ilk çözümü domine etmektedir ve ilk çözüm etkin çözümler kümesinde yer almamaktadır.



Şekil 5.2: 2 makine, ÖT, DV veri kümesi için örnek etkin çözüm kümesi

Grafik incelendiğinde, Pareto önyüzde 2 farklı döngünün önyüzün farklı noktalarında ortaya çıktığı görülmektedir. Bunlardan ilki yukarıda açıklanan 2.1 numaralı döngüdür. Çevrim zamanı üst limitinin belirli bir değerden düşük olduğu noktalarda hep bu döngü en iyi sonuçları vermiştir. Çevrim zamanı üst limiti 34' e ulaştığı anda ise 2.6 numaralı

döngü en iyi sonuçları vermeye başlamıştır. Ayrıca, robotun tüm hızlarını üst limitte kullanmasına göre, hız ayarlaması yapılarak %18 enerji tasarrufu sağlanmıştır. Enerji tasarrufu sonuçları ile ilgili detaylı açıklama Bölüm 5.4'de verilecektir.

3 makineli veri kümeleri için pareto optimal noktalarda ortaya çıkan çevrimler incelenecek olursa, aşağıdaki 5 döngünün çözümlerde yer aldığını görülebilir:

$$\mathbf{3.1} \quad U_0(1,2) \longrightarrow L_1(0,2) \longrightarrow U_2(3,2) \longrightarrow L_2(3,0) \longrightarrow U_3(3,4) \longrightarrow L_4(3,0) \\ \longrightarrow L_3(0,0) \longrightarrow U_1(0,2)$$

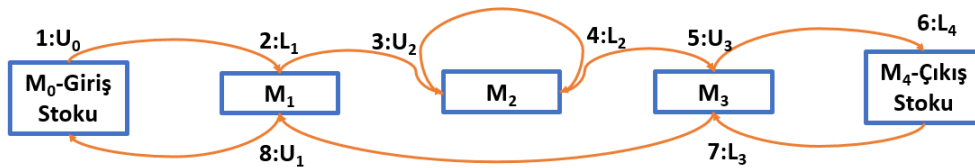
$$\mathbf{3.2} \quad U_0(1,3) \longrightarrow L_1(0,3) \longrightarrow U_1(2,3) \longrightarrow L_2(0,3) \longrightarrow U_3(4,3) \longrightarrow L_4(0,3) \\ \longrightarrow L_3(0,0) \longrightarrow U_2(0,3)$$

$$\mathbf{3.3} \quad U_0(1,2) \longrightarrow L_1(0,2) \longrightarrow U_2(3,2) \longrightarrow L_3(0,2) \longrightarrow U_3(4,2) \longrightarrow L_4(0,2) \\ \longrightarrow L_2(0,0) \longrightarrow U_1(0,2)$$

$$\mathbf{3.4} \quad U_0(1,0) \longrightarrow L_1(0,0) \longrightarrow U_3(4,0) \longrightarrow L_4(0,0) \longrightarrow U_2(3,0) \longrightarrow L_3(0,0) \\ \longrightarrow U_1(2,0) \longrightarrow L_2(0,0)$$

$$\mathbf{3.5} \quad U_0(1,0) \longrightarrow L_1(0,0) \longrightarrow U_1(2,0) \longrightarrow L_2(0,0) \longrightarrow U_2(3,0) \longrightarrow L_3(0,0) \\ \longrightarrow U_3(4,0) \longrightarrow L_4(0,0)$$

3 makineli veri kümeleri ile alınan 480 koşturumda %34 oranı ile en fazla karşılaşılan aktivite sıralaması 3.1 numaralı sıralama olmuştur. Şekil 5.3 bu çevrimin doğrusal makine dizilimi için temsilini göstermektedir. 2 makineli kümeler için elde edilen 2.1 numaralı sonuç ile benzerlik taşımaktadır. Şöyle ki, Giriş Stoku ile Çıkış Stoku arasındaki pozisyonlarda makine numarası ya artış göstermiş ya da sabit kalmıştır. Çıkış ve Giriş stokları arasında ise makine numaraları azalan bir davranış göstermiştir. Bu durum, robotun bulunduğu doğrultuda hep ileri gitme yönünde bir davranış gösterdiği şeklinde yorumlanabilir. Bu döngü için aktif tutucu değiştirme sayısı ise dördüttür.



Şekil 5.3: 3.1 Numaralı Döngü

3.2 ve 3.3 numaralı döngüler de %17 ve %11 oranında açığa çıkmış döngülerdir. Bu döngüler için de robotun bulunduğu yönde hep ileri gittiği çıkarımı yapılabilir.

Küçük θ değerine sahip 3 makineli kümeler için, 3.1. çevrimin A ve B noktalarında optimal olarak elde edilme oranı %95 olarak tespit edilmiştir. Büyük θ değerine sahip tüm örnekler için ise A ve B noktalarında 3.4 numaralı çözüm elde edilmiştir. Görüleceği üzere bu aktivite sıralamasında tutucu değişimi gerçekleşmemektedir. Bu sebeple, büyük θ ile oluşturulmuş örneklerde en küçük çevrim zamanını sağlayan sıralama olarak açığa çıkmıştır. C noktalarında çevrim zamanı üst limitinin gevşemesiyle, tutucu değişimi gerektiren döngüler bu tip örneklerde de görülmeye başlanmıştır. Çevrim 3.5 ise çevrim zamanı limitinin yeterince geniş olduğunda ortaya çıkan döngüdür.

2 ve 3 makineli veri kümelerinin yanı sıra 4 makineli örnekler için de açığa çıkan çözümler incelenmiştir. 4 makineli örnekler için en küçük ve en büyük çevrim zamanı noktasını içeren 7 pareto optimal nokta için çözümler alınmıştır. 40 veri kümesi için yapılan 280 koşturumun sonuçları incelendiğinde, 19 farklı çevrimin açığa çıktığı, bunlardan 5 tanesinin ise %85 oranında çözümlerde görüldüğü tespit edilmiştir. Bu 5 çevrim şu şekildedir:

$$\mathbf{4.1} \quad U_0(1,2) \longrightarrow L_1(0,2) \longrightarrow U_2(3,2) \longrightarrow L_2(3,0) \longrightarrow U_3(3,4) \longrightarrow L_3(0,4) \\ \longrightarrow U_4(5,4) \longrightarrow L_5(0,4) \longrightarrow L_4(0,0) \longrightarrow U_1(0,2)$$

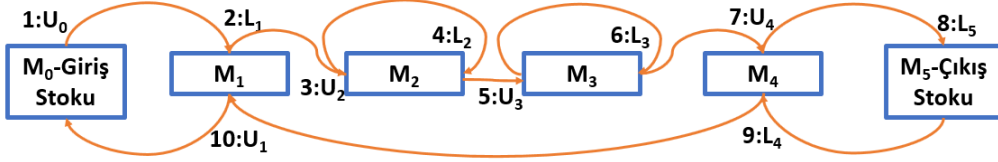
$$\mathbf{4.2} \quad U_0(1,0) \longrightarrow L_1(0,0) \longrightarrow U_1(2,0) \longrightarrow L_2(0,0) \longrightarrow U_2(3,0) \longrightarrow L_3(0,0) \\ \longrightarrow U_3(4,0) \longrightarrow L_4(0,0) \longrightarrow U_4(5,0) \longrightarrow L_5(0,0)$$

$$\mathbf{4.3} \quad U_0(1,4) \longrightarrow L_1(0,4) \longrightarrow U_2(3,4) \longrightarrow L_3(0,4) \longrightarrow U_4(5,4) \longrightarrow L_5(0,4) \\ \longrightarrow L_4(0,0) \longrightarrow U_3(0,4) \longrightarrow U_1(2,4) \longrightarrow L_2(0,4)$$

$$\mathbf{4.4} \quad U_0(1,0) \longrightarrow L_1(0,0) \longrightarrow U_4(5,0) \longrightarrow L_5(0,0) \longrightarrow U_3(4,0) \longrightarrow L_4(0,0) \\ \longrightarrow U_2(3,0) \longrightarrow L_3(0,0) \longrightarrow U_1(2,0) \longrightarrow L_2(0,0)$$

$$\mathbf{4.5} \quad U_0(1,2) \longrightarrow L_1(0,2) \longrightarrow U_2(3,2) \longrightarrow L_3(0,2) \longrightarrow U_4(5,2) \longrightarrow L_5(0,2) \\ \longrightarrow U_3(4,2) \longrightarrow L_4(0,2) \longrightarrow L_2(0,0) \longrightarrow U_1(0,2)$$

Şekil 5.4’de 4.1 numaralı döngüdeki aktivite sıralamasının gösterimi yapılmıştır. Bu sıralama, 4 makineli sonuçlarda %26 oranı en çok açığa çıkan çözümdür. Şekil 5.4’e bakıldığında robotun çıkış stokuna kadar hep ilerideki bir makineye, çıkış stokundan sonra da geridekine hareket etme durumunun bu çevrimde de ortaya çıktığı görülmektedir.



Şekil 5.4: 4.1 Numaralı Döngü

4.2. çevrim ise tüm veri kümeleri için en büyük çevrim zamanı ve en küçük enerji tüketimi çözümü için elde edilen döngüdür. Diğer çözümlerin açığa çıkma oranları ise sırasıyla, %10, %20 ve %14’tür. Bu 3 döngü için de yukarıda bahsedilen durum geçerli değildir.

2, 3 ve 4 makineli veri kümeleri için alınan çözümlerde en fazla sayıda ortaya çıkan çözümler sırasıyla Şekil 5.1, 5.3 ve 5.4 ile gösterilmiştir. Bu üç çevrimin ortaya koyduğu örüntü incelendiğinde aşağıdaki çıkarımlar elde edilebilir.

1. Robot giriş stokundan sonra 1. makineye yükleme aktivitesini gerçekleştirmektedir: $U_0 \rightarrow L_1$
2. L_1 aktivitesinden sonra robot çıkış stokundan bir önceki makineye kadar ardışık makinelerde sırasıyla boşaltma ve yükleme aktivitelerini gerçekleştirmektedir :
 $U_0 \rightarrow L_1 \rightarrow U_2 \rightarrow L_2 \rightarrow \dots \rightarrow U_{(m-1)} \rightarrow L_{(m-1)}$
3. Robot çıkış stokundan bir önceki m makinesine geldiğinde ise sadece makineden boşaltma aktivitesini yapmakta ve daha sonra da yükleme aktivitesi için çıkış stokuna ilerlemektedir:
 $U_0 \rightarrow L_1 \rightarrow U_2 \rightarrow L_2 \rightarrow \dots \rightarrow U_{(m-1)} \rightarrow L_{(m-1)} \rightarrow U_m \rightarrow L_{(m+1)}$
4. Bir sonraki sırada ise robot, yükleme işlemini henüz gerçekleştirmediği m maki-

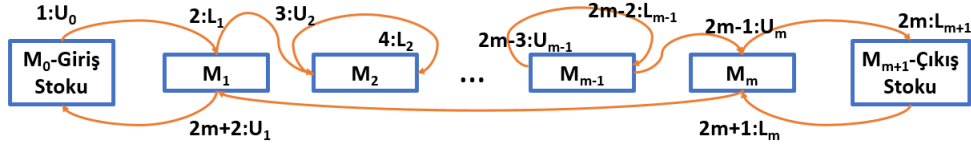
nesine hareket ederek L_m aktivitesini yapmaktadır:

$$U_0 \rightarrow L_1 \rightarrow U_2 \rightarrow L_2 \rightarrow \dots \rightarrow U_{(m-1)} \rightarrow L_{(m-1)} \rightarrow U_m \rightarrow L_{(m+1)} \rightarrow L_m$$

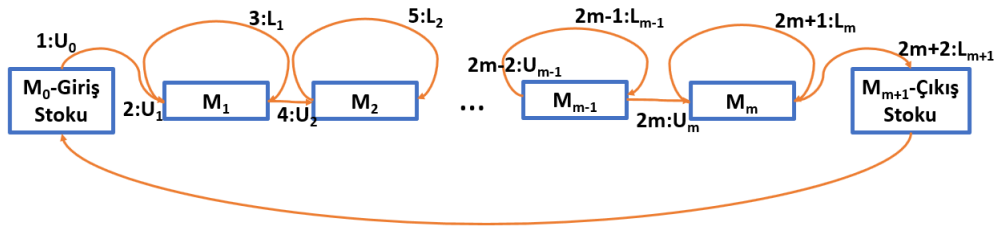
5. Son pozisyonda ise 1. pozisyonda yüklenmiş olan 1. makineden parça boşaltılması yapılmaktadır. Bu aktiviteden sonra robot tekrar ilk pozisyona dönerek bir sonraki çevrime başlamış olur.

$$U_0 \rightarrow L_1 \rightarrow U_2 \rightarrow L_2 \rightarrow \dots \rightarrow U_{(m-1)} \rightarrow L_{(m-1)} \rightarrow U_m \rightarrow L_{(m+1)} \rightarrow L_m \rightarrow U_1$$

Şekil 5.5 ile görselleştirilen bu çevrim incelendiğinde Sethi vd. [45] tarafından belirli şartlar altında dairesel makine dizilimi için optimal olduğu ispatlanan Şekil 5.6 deki döngü ile benzer adımlar izlediği görülebilir.



Şekil 5.5: m makine



Şekil 5.6: m makine

Bu iki çevrimdeki farklılıklar yalnızca giriş stokundan bir sonraki makine ile çıkış stokundan bir önceki makinelerdeki aktivitelerin sıralamasından kaynaklanmaktadır. Bu örüntüden yola çıkılarak 6 ve daha çok makineli veri kümeleri için aktivite sıralaması elde edilebilir. Bu sayede MISOCP ve ETA modeli için etkin çözüm elde etmenin zorlaştığı noktalar için çözüm elde etmede kolaylık sağlanabilir.

5.4 Enerji Tasarrufu ile İlgili Sonuçlar

Bu bölümde, tüm makine kümeleri için elde edilen çözümler pareto optimal noktaları arasında geçişlerdeki enerji tasarrufları bakımından elde edilen sonuçlar açıklanacaktır.

Çizelge 5.12’de 2 ve 3 makineli veri setlerinin küçük ve büyük θ değerleri için enerji tüketimi tasarrufu sonuçları bulunmaktadır. Bu kümeler için matematiksel model ile zaman limiti içinde optimal sonuçlar elde edildiğinden, tasarruf yüzdeleri matematiksel model sonuçları üzerinden elde edilmiştir. Tabloda $A \rightarrow B$ sütunu altında bulunan değerler Şekil 4.1’de gösterilen A ve B noktaları arasındaki enerji tüketim farkının yüzdesidir. Bu yüzdeler, hızlar üst limit değerine eşitken elde edilen en küçük çevrim zamanlı A noktasından, hızların ayarlanabilir olduğu ve en küçük çevrim zamanına sahip olan B noktasına geçildiğinde sağlanan enerji tasarrufunu göstermektedir.

Çizelge 5.12: 2 ve 3 makine, Enerji Tasarrufu ile ilgili sonuçlar (%)

Veri Kümesi	2 Makine				3 Makine			
	Küçük θ		Büyük θ		Küçük θ		Büyük θ	
	A \rightarrow B	B \rightarrow C ₁	A \rightarrow B	B \rightarrow C ₁	A \rightarrow B	B \rightarrow C ₁	A \rightarrow B	B \rightarrow C ₁
1	18.1	45.3	0.0	45.4	17.4	53.6	0.0	40.2
2	45.9	41.0	29.4	52.9	18.1	54.5	0.0	41.6
3	45.9	41.0	29.4	53.0	18.3	52.9	0.0	38.5
4	18.1	46.4	0.0	47.8	18.3	55.5	0.0	41.1
5	18.4	61.9	46.7	48.9	18.7	55.1	0.0	40.1
6	32.4	58.3	47.5	49.5	18.6	56.2	16.3	38.4
7	44.9	40.8	29.5	52.4	18.6	53.5	0.6	38.3
8	18.4	60.4	38.8	48.5	18.6	56.7	12.0	39.7
9	18.3	61.8	48.2	48.1	19.0	57.0	15.1	39.5
10	40.2	43.3	12.7	68.9	18.7	55.9	10.5	39.0
11	18.1	45.3	18.0	53.7	18.0	50.9	6.2	35.9
12	45.5	38.7	29.1	50.2	17.4	50.7	0.0	44.6
13	18.2	60.2	49.7	44.6	40.4	50.9	24.8	42.9
14	42.0	52.9	13.8	81.5	18.1	55.2	0.0	42.3
15	41.5	53.4	13.9	81.6	26.0	56.0	26.5	40.2
16	45.5	40.3	29.1	52.3	36.1	49.2	31.6	37.7
17	18.1	60.1	40.9	47.0	37.6	51.3	23.1	39.8
18	26.7	57.6	55.7	43.6	49.1	48.2	32.3	42.6
19	36.9	54.7	59.8	44.3	18.6	67.9	30.7	48.0
20	30.1	56.4	58.0	43.5	39.2	47.8	25.8	48.6
En Küçük	18.1	38.7	0.0	43.5	17.4	47.8	0.0	35.9
Ortalama	31.2	51.0	32.5	52.9	24.2	53.9	12.8	40.9
En Büyük	45.9	61.9	59.8	81.6	49.1	67.9	32.3	48.6

Diğer bir deyişle, robotların ayarlanabilir hızlara sahip olduğu sistemlerin, sabit hızlı

sistemlere göre ne kadar daha az enerji tükettiğini göstermektedir. 2 makineli veri kümeleri için bu değer ortalama %31.8'dir. 3 makineli veri setleri için ise bu değer %18.5'tir.

Çizelge 5.12'de $B \rightarrow C_1$ sütunu altında bulunan değerler ise, yine Şekil 4.1'deki B ve C_1 noktaları arasındaki geçişteki enerji tasarrufu değerlerini göstermektedir. Bu iki nokta birbirini domine edemeyen iki noktadır. Ancak, bir çevrim zamanı kısıtından bir λ miktarı genişletilmesiyle büyük oranlarda enerji tasarrufu sağlanabildiğini göstermek için bu iki nokta arasındaki enerji farkı yüzdeleri de analizlere eklenmiştir. Bir veri kümesi için λ değeri daha önceki bölümlerde de anlatıldığı üzere o kümenin en büyük ve en küçük çevrim zamanlarının farkının 10'a bölünmesiyle elde edilir. Örneğin, küçük θ değerli 1. veri kümesi için en küçük çevrim zamanı 13, en büyük ise 34'tür. Bu küme için çevrim zamanı üst limiti 13'ten 15.1'e genişletildiğinde %45.3 oranında enerji tasarrufu sağlanmıştır.

Çizelge 5.13 ise 4 ve 6 makine için enerji tüketimi farkı yüzdelerini vermektedir.

Çizelge 5.13: 4 ve 6 makine, Enerji Tasarrufu ile ilgili sonuçlar (%)

Veri Kümesi	2 Makine				3 Makine			
	Küçük θ		Büyük θ		Küçük θ		Büyük θ	
	A \rightarrow B	B \rightarrow C_1	A \rightarrow B	B \rightarrow C_1	A \rightarrow B	B \rightarrow C_1	A \rightarrow B	B \rightarrow C_1
1	5.9	67.4	0.0	57.8	0.0	68.7	0.0	50.3
2	0.0	73.4	0.0	41.7	12.3	62.2	16.7	59.9
3	9.9	65.7	0.0	45.3	-18.7*	72.0	24.3	45.4
4	0.0	71.1	0.0	41.8	0.0	65.5	0.0	61.4
5	0.0	71.0	0.0	41.3	19.1	71.7	4.7	48.0
6	0.0	75.4	0.0	46.5	11.3	67.1	32.4	38.5
7	0.0	73.1	0.0	44.1	3.1	73.0	14.8	58.2
8	0.0	74.0	0.0	44.8	-5.1*	63.4	19.2	60.8
9	5.9	68.2	0.0	58.1	4.6	65.2	18.8	54.2
10	0.0	72.8	0.0	43.6	19.2	65.9	24.0	48.2
11	5.9	63.3	0.0	44.1	7.4	76.8	7.4	68.5
12	0.0	72.9	0.0	44.4	5.6	65.3	14.8	49.5
13	8.5	59.9	0.0	42.8	11.2	72.8	29.1	55.2
14	4.9	72.4	18.3	46.7	1.6	71.7	9.4	62.3
15	12.0	69.6	17.2	45.4	11.8	66.4	28.1	58.0
16	39.0	42.8	25.6	41.7	16.8	67.1	10.5	71.6
17	0.0	73.6	0.0	46.2	26.2	64.7	23.9	57.6
18	3.0	69.4	3.0	38.9	8.0	69.5	10.5	68.6
19	5.1	71.3	27.6	51.6	20.9	62.7	-15.4*	70.8
20	5.1	73.2	17.2	63.7	18.5	64.8	21.3	73.8
En Küçük	0.0	42.8	0.0	38.9	0.0	0.0	0.0	0.0
Ortalama	5.3	69.0	5.4	46.5	0.0	68.7	0.0	50.3
En Büyük	39.0	75.4	27.6	63.7	12.3	62.2	16.7	59.9

Çizelge 5.13'te 4 makine için verilen sonuçların 1 tanesi hariç hepsi matematiksel model ile elde edilen optimal değerlerle hesaplanmıştır. Yalnızca Küçük θ 'ya sahip 10. veri kümesinin C_1 enerji değeri sezgisel modelin sonuçlarından alınmıştır. Çünkü hatırlanacağı üzere bu noktada MISOCP modeli zaman limitine takılmış olup, sezgisel model ile daha iyi bir sonuç elde edilmiştir. Dolayısıyla bu veri kümesi için $B \rightarrow C_1$ değeri, matematiksel modelden alınan B noktası ve ETA sonuçlarından alınan C_1 noktası enerji değerine göre hesaplanmıştır.

6 makine için ise matematiksel model ile zaman limiti içerisinde optimal değerler elde edilemediğinden, enerji tasarrufu değerleri ETA ve MISOCP sonuçları karşılaştırılarak hesaplanmıştır. Yani, hangi model ile elde edilen enerji değeri daha küçükse fark değerleri ona göre hesaplanmıştır. Çizelge 5.13'ye bakıldığında * ile gösterilen bazı yüzdelerin negatif işaretli olduğu görülmektedir. $A \rightarrow B$ sütunlarında gerçekleşen bu negatif yüzdelerin sebebi, B noktası için MISOCP ile zaman limiti içerisinde herhangi bir çözüm elde edilememesi ve ETA ile elde edilen çözümün kalitesinin daha kötü olup A noktasından daha çok enerjili bir çözüm elde edilmiş olmasıdır. Ortalama değerler hesaplanırken bu oranlar ortalamaya dahil edilmemiştir. 4 makine kümelerinde ortalama enerji tasarrufu değerleri $A \rightarrow B$ için %5.3 iken, $B \rightarrow C_1$ için %57.8'dir.

4 makineli kümeler gibi, 6 makineli örneklerde de $B \rightarrow C_1$ noktaları arasındaki geçişte yüksek oranda enerji tasarrufu gerçekleşmektedir. Elde edilen sonuçlara göre, bu geçişte ortalama olarak %62.9 enerji tasarrufu sağlanmıştır. $A \rightarrow B$ arasındaki geçişte ise nispeten daha düşük bir enerji tasarrufu sağlanmıştır. Ortalama %15.3 olan bu değer yine de göz önünde bulundurmaya yeterli olacak bir tasarrufun olduğu göstermektedir.

Çizelge 5.12 ve 5.13'de bazı değerlerin ise 0 olduğu görülmektedir. $A \rightarrow B$ sütunlarında bulunan bu 0 değerleri ayarlanabilir hızların o veri setleri için enerji tasarrufu için yeterli olmadığını göstermektedir.

6. SONUÇ VE ÖNERİLER

Bu çalışmada tek tip parça üreten çift tutuculu robotlu hücrelerdeki robot hareket sıralamasının ve robot hareket hızlarının belirlenmesi problemleri eşzamanlı olarak ele alınmıştır. m -makineli genel robotik hücrelerdeki problemi çözebilmek için bir Karma Tamsayı Doğrusal Olmayan matematiksel model geliştirilmiş ve yapılan testlerle doğrulaması gerçekleştirilmiştir. Fakat küçük problem boyutlarında bile makul sürelerde çözüm çıkartamayan bu modele alternatif olarak İkinci Derece Konik programlama formülasyonu geliştirilmiştir. Bu modelin de doğrulaması yapıldıktan sonra her iki model birbiriyle çözüm süresi açısından karşılaştırılmıştır. Konik model çözüm süresini önemli derecede düşürmüştür. Buna rağmen problem büyüklüğü arttıkça konik modelin de çözüm süresi hızlı bir şekilde artmaktadır. Bu sebeple problem için ayrıca bir sezgisel algoritma geliştirilmiştir. Etkin Çözüm Türetme Algoritması (ETA) adı verilen bu sezgisel algoritma JAVA programlama dilinde kodlanmıştır.

Test problemleri türetilerek sezgiselin ve matematiksel modellerin kapsamlı performans testleri yapılmıştır. Yapılan testlerde çift tutuculu robotların, esnek hızlı robotik hücrelerde kullanıldığında ortalama %17.7 enerji tasarrufu sağladığı gösterilmiştir. Bu ortalamanın 2,3, 4 ve 6 makineli kümeler için ise değerleri sırasıyla %31.8, %18.5, %5.4 ve %15.3'tür. En küçük çevrim zamanı ve en küçük enerji tüketimi çözümünde çevrim zamanı limiti λ kadar artırıldığında ise enerji tüketiminde ortalama %55 tasarruf sağlanabileceği görülmüştür.

İlerleyen çalışmalarda, 6 makineli veri kümelerinde ETA'nın verimliliğini artırmak amacıyla geliştirmeler yapılabilir. 2, 3 ve 4 makineli kümeler için baskın aktivite sıralamalarında açığa çıkan sonuçlarda robotun bir sonraki makineye gitme eğiliminde olması çıkarımı kullanılarak başlangıç çözümleri ve komşuluk aramalarında iyileştirmeler sağlanabilir. Ayrıca, Şekil 5.5'te verilen çevrim geliştirilen sezgisel yöntem için başlangıç çözümü olarak verilmesinin, elde edilen minimum çevrim zamanı çözümlerinin kalitesini artırıp artırmayağı test edilmelidir. Ayrıca bu çevrimin diğer Pareto

etkin noktalar için de aday çözüm kabul edilmesi çözümlerin hata yüzde oranlarının geliştirilmesinde rol oynayabilir.

Şekil 5.5 ile verilen çevrimin hangi ortam şartları altında optimal aktivite sıralamasını vereceği araştırılabilir. Bu aktivite sıralamasının optimal çevrim olup olmasında makine işlem süreleri varyansının ya da makineler arası uzaklıkların nasıl dağıldığının etkili olup olmadığı incelenebilir.

Bunların yanında, makineden alınma kriterinin hemen ya da belirli bir süre olduğu üretim sistemleri için geliştirilen çözüm yöntemleri de farklılaştırılarak uygulanabilir. Ancak bu sistemler için makine bekleme süreleri de göz önünde bulundurulacağından problemin karmaşıklığı artmaktadır.

Ayrıca, değişik tip parça üretimi, 1-birim yerine k-birim döngülerin ele alınması, robot hızı yanında makine hızlarının da kontrol edilebilir olması, robot hareket ivmelerinin de göz önünde bulundurulması, makineler arasında ara stok alanı bulunması gibi farklı ortam şartlarına göre de çalışmalar çeşitlendirilebilir.

KAYNAKÇA

- [1] Akturk, M. S., Gultekin, H., and Karasan, O. E. (2005). Robotic cell scheduling with operational flexibility. *Discrete Applied Mathematics*, 145(3):334–348.
- [2] Amerika Birlesik Devletleri Nüfus Sayım Bürosu, (2005). Annual survey of manufacturers. <https://www2.census.gov/programs-surveys/asm/tables/2005/a>. Alındığı tarih:2019-06-10.
- [3] Berlin, H. (2019). Power and productivity for a better world. <http://www.abb.com/blog/gad00540/1dde6.aspx?tag=productivity>. Alındığı tarih: 2019-06-10.
- [4] Brauner, N. (2008). Identical part production in cyclic robotic cells: Concepts, overview and open questions. *Discrete Applied Mathematics*, 156(13):2480–2492.
- [5] Brauner, N., Finke, G. (1999). Optimal moves of the material handling system in a robotic flow-shop. In *Proceedings IEPM*, volume 99, pages 409–417.
- [6] Brauner, N. and Finke, G. (2001). Cycles and permutations in robotic cells. *Mathematical and Computer Modelling*, 34(5):565–591.
- [7] Bryan, C., Grenwalt, M., and Stienecker, A. (2010). Energy consumption reduction in industrial robots. In *Proceedings ASEE North Central Sectional Conference*.
- [8] Bukata, L., Šucha, P., Hanzálek, Z., and Burget, P. (2017). Energy optimization of robotic cells. *IEEE Transactions on Industrial Informatics*, 13(1):92–102.

- [9] **Crama, Y. and Klundert, J. V.D.** (1997). Cyclic scheduling of identical parts in a robotic cell. *Operations Research*, 45(6):952–965.
- [10] **Crama, Y. and van De Klundert, J.** (1999). Cyclic scheduling in 3-machine robotic flow shops. *Journal of Scheduling*, 2(1):35–54.
- [11] **Crama, Y. and van De Klundert, J.** (1997). Cyclic scheduling of identical parts in a robotic cell. *Operations Research*, 45(6):952–965.
- [12] **Dawande, M., H. N., G., Sethi, S. P., and Sriskandarajah, C.** (2005). Sequencing and scheduling in robotic cells: Recent developments. *Journal of Scheduling*, 8(5):387–426.
- [13] **Drobouchevitch, I., Geismar, N., and Sriskandarajah, C.** (2010). Throughput optimization in robotic cells with input and output machine buffers: A comparative study of two key models. *European Journal of Operational Research*, 206(3):623–633.
- [14] **Drobouchevitch, I., Sethi, S. P., Sidney, J., and Sriskandarajah, C.** (2004). A note on scheduling multiple parts in two-machine dual gripper robotic cell: Heuristic algorithm and performance guarantee. *International Journal of Operations and Quantitative Management*, 10(4):297–314.
- [15] **Drobouchevitch, I., Sethi, S. P., Sidney, J., and Sriskandarajah, C.** (2006). Scheduling dual gripper robotic cell: One-unit cycles. *European Journal of Operational Research*, 171(2):598–631.
- [16] **Enerji ve Tesisat, (2014).** Sanayide enerji verimliliği <http://www.enerjivetesisat.com/enerjihaberleri/enerji/3532-sanayide-enerji-verimlilik-cokun-karabalmakale>. Alındığı tarih: 2019-06-10.
- [17] **Enerji Verimliliği Derneği, (2015).** Enerji verimliliği strateji belgesi. <http://www.enver.org.tr/UserFiles/Article/7d3a2037-d5fe-4c28-8031-363aefd325d1.pdf>. Alındığı tarih: 2019-06-10
- [18] **Foumani, M. and Jenab, K.** (2012). Cycle time analysis in reentrant robotic cells with swap ability. *International Journal of Production Research*, 50(22):6372–6387.

- [19] **Geismar, H., Dawande, M., and Sriskandarajah, C.** (2006). Throughput optimization in constant travel-time dual gripper robotic cells with parallel machines. *Production and Operations Management*, 14(2): 311–328.
- [20] **Geismar, H., Manoj, U., Sethi, S. P., Sriskandarajah, C., and Ramanan, N.** (2012). Scheduling robotic cells served by a dual-arm robot. *IIE Transactions*, 44(3):230–248.
- [21] **Geismar, H., Pinedo, M., and Sriskandarajah, C.** (2008). Robotic cells with parallel machines and multiple dual gripper robots: a comparative overview. *IIE Transactions*, 40(12):297–314.
- [22] **Geismar, H., Sriskandarajah, C., and Ramanan, N.** (2004). Increasing throughput for robotic cells with parallel machines and multiple robots. *IEEE Transactions on Automation Science and Engineering*, 1(1):84–89.
- [23] **Geismar, H. N., Dawande, M., and Sethi, S. P.** (2005). Dominance of cyclic solutions and challenges in the scheduling of robotic cells. *SIAM Review*, 47(4):709–721.
- [24] **Geismar, H. N., Dawande, M., and Sriskandarajah, C.** (2007). A $(10/7)$ -approximation algorithm for an optimum cyclic solution in additive travel-time robotic cells. *IIE Transactions*, 39(2):217–227.
- [25] **Gultekin, H., Akturk, M. S., and Karasan, O. E.** (2006). Cyclic scheduling of a 2-machine robotic cell with tooling constraints. *European Journal of Operational Research*, 174(2):777–796.
- [26] **Gultekin, H., Akturk, M. S., and Karasan, O. E.** (2008). Bicriteria robotic cell scheduling. *Journal of Scheduling*, 11(6):457–473.
- [27] **Gultekin, H., Akturk, M. S., and Karasan, O. E.** (2010). Bicriteria robotic operation allocation in a flexible manufacturing cell. *Computers & operations research*, 37(4):779–789.
- [28] **Gundogdu, E. and Gultekin, H.** (2015). Scheduling in two-machine robotic

cells with a self-buffered robot. *IEEE Transactions*, 48(2):170–191.

- [29] **Gurel, S., Cinciog˘lu, D.** (2015). Rescheduling with controllable processing times for number of disrupted jobs and manufacturing cost objectives. *International Journal of Production Research*, 53(9):2751–2770.
- [30] **Gürel, S., Gultekin, H., and Akhlaghi, V. E.** (2019). Energy conscious scheduling of a material handling robot in a manufacturing cell. *Robotics and Computer-Integrated Manufacturing*, 58:97–108.
- [31] **Hall, N., Potts, C., and Sriskandarajah, C.** (2000). Parallel machine scheduling with a common server. *Discrete Applied Mathematics*, 102(3):223–243.
- [32] **Hall, N. G., Kamoun, H., and Sriskandarajah, C.** (1997). Scheduling in robotic cells: Classification, two and three machine cells. *Operations Research*, 45(3):421–439.
- [33] **Kats, V. and Levner** (1997). A strongly polynomial algorithm for no-wait cyclic robotic flowshop scheduling. *Operations Research Letters*, 21(4):171–179.
- [34] **Kats, V. and Levner** (1998). Minimizing the number of vehicles in periodic scheduling: The non-euclidean case. *European Journal of Operational Research*, 107(2):371–377.
- [35] **Kilic, H. S. and Durmusoglu, M. B.** (2015). Advances in assembly line parts feeding policies: a literature review. *Assembly Automation*, 35(1):57–68.
- [36] **Koc, C. and Karaoglan, I.** (2016). The green vehicle routing problem: A heuristic based exact solution approach. *Applied Soft Computing*, 39:154–164.
- [37] **Lee, T.** (2008). A review of scheduling theory and methods for semiconductor manufacturing cluster tools. In *Proceedings of the 40th Conference on Winter Simulation, Miami, Florida*.

- [38] **Lin, C., Choy, K., Ho, G., Chung, S., and Lam, H.** (2014). Survey of green vehicle routing problem: Past and future trends. *Expert Systems with Applications*, 41(4, Part 1):1118 – 1138.
- [39] **Meike, D., Pellicciari, M., and Berselli, G.** (2014). Energy efficient use of multirobot production lines in the automotive industry: Detailed system modeling and optimization. *IEEE Transactions on Automation Science and Engineering*, 11(3):798–809.
- [40] **Meike, D. and Ribickis, L.** (2011). Energy efficient use of robotics in the automobile industry. In *Advanced Robotics (ICAR), 2011 15th International Conference on*, pages 507–511. IEEE.
- [41] **Paryanto, Brossog, M., Bornschlegl, M., and Franke, J.** (2015). Reducing the energy consumption of industrial robots in manufacturing systems. *International Journal of Advanced Manufacturing Technology*, 78(5-8):1315 – 1328.
- [42] **Pellicciari, M., Berselli, G., Leali, F., and Vergnano, A.** (2013). A method for reducing the energy consumption of pick-and-place industrial robots. *Mechatronics*, 23(3):326–334.
- [43] **Saka, O. C., Gürel, S., and Woensel, T. V.** (2016). Using cost change estimates in a local search heuristic for the pollution routing problem. *OR Spectrum*, 39(2):557–587.
- [44] **Sethi, S. P., Sidney, J. B., and Sriskandarajah, C.** (2001). Scheduling in dual gripper robotic cells for productivity gains. *IEEE Transactions on Robotics and Automation*, 17(3):324–341.
- [45] **Sethi, S. P., Sriskandarajah, C., Sorger, G., Blazewicz, J., and Kubiak, W.** (1992). Sequencing of parts and robot moves in a robotic cell. *International Journal of Flexible Manufacturing Systems*, 4(3 – 4):331–358.
- [46] **Smetanová, A.** (2010). Optimization of energy by robot movement. *Modern Machinery Science Journal*, 3(1):172–176.
- [47] **Sriskandarajah, C., Drobouchevitch, I., Sethi, S. P., and Chandrasekaran, R.** (2004). Scheduling multiple parts in a robotic cell served by a dual-

gripper robot. *Operations Research*, 52(1):65–82.

- [48] **Vergnano, A., Thorstensson, C., Lennartson, B., Falkman, P., Pellicciari, M., Leali, F., and Biller, S.** (2012). Modeling and optimization of energy consumption in cooperative multi-robot systems. *Automation Science and Engineering, IEEE Transactions on*, 9(2):423–428.
- [49] **World Robotics,** (2019). World robotics 2019 preview. https://ifr.org/downloads/press2018/IFR_World_Robotics_Outlook_2019_-_Chicago.pdf. Alındığı tarih: 2019-06-10.



ÖZGEÇMİŞ

Ad-Soyad : Nurdan Emirođlu
Uyruđu : TC
Dođum Tarihi ve Yeri : 29 Mart 1993, İskilip/Çorum
E-posta : nurdantemiroglu@gmail.com

ÖĐRENİM DURUMU:

- **Lisans** : 2016, İhsan Dođramacı Bilkent Üniversitesi, Endüstri Mühendisliđi
- **Yüksek Lisans** : 2019, TOBB Ekonomi ve Teknoloji Üniversitesi, Endüstri Mühendisliđi

MESLEKİ DENEYİM VE ÖDÜLLER:

Yıl	Yer	Görev
2016-2019	TOBB ETU	Tam Burslu Yüksek Lisans Öğrencisi

YABANCI DİL: İngilizce

TEZDEN TÜRETİLEN YAYINLAR, SUNUMLAR VE PATENTLER:

- **Tatar, N.**, Gultekin, H., Gürel, S., Çift Tutuculu Malzeme Taşıyıcı Robotlu Hücrelerde Enerji Duyarlı Çizelgeleme, YAEM 2018, Eskişehir, Turkey
- **Tatar, N.**, Gultekin, H., Gürel, S., Energy Concious Scheduling of Robot Moves in Dual Gripper Robotic Cells, 16th annual PMS 2018, Rome, Italy
- **Tatar, N.**, Gultekin, H., Gürel, S., Çift Tutuculu Robotlu Hücrelerde Robot Hareket Hızlarının Ve Sıralamalarının Belirlenmesi, YAEM 2017, Istanbul, Turkey